Simultaneous Precise Solutions to the Visibility Problem of Sculptured Models

Joon-Kyung Seong[1], Gershon Elber[2], and Elaine Cohen[1]

[1] University of Utah, Salt Lake City, UT84112, USA
seong@cs.utah.edu, cohen@cs.utah.edu
[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 θ 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.

M.-S. Kim and K. Shimada (Eds.): GMP 2006, LNCS 4077, pp. 451–464, 2006.

- 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 [24,1,9,18,14,19]. 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 polygonal 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 the hidden curves from line drawings of surfaces (not shaded drawings). Krishnan and Manocha [16] presented an algorithm for the elimination of hidden surfaces using a combination of symbolic techniques and results from numerical linear algebra.

Elber et al. [6] presented an algorithm for computing two-dimensional visibility charts for planar curves. The visibility charts, however, are constructed by discretizing a continuous set of view directions [6]. Our algorithm is an extension of that work into the computation of continuous visibility charts. Krishnan and Manocha [16] solves the hidden surface removal problem for a discrete set of view directions only.

Our approach is unique in that of solving the visibility problem for all view directions in the domain, simultaneously.

Summary of Our Approach

We reduce the solution to the visibility problem to the problem of finding the zeros of a set of polynomial equations in the parameter space. For the curve case, visible curve locations are computed by solving 2 polynomial equations in 3 variables (t and r for curve parameters and θ for a view direction). Since one of the two equations is an inequality constraint, this framework yields 2-manifold surfaces as a 0-set in a 3-D parameter space. A curve's location is invisible if its corresponding parameter belongs to the projected region of the two-manifolds onto the $t\theta$-plane. The problem for 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 this 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.

The presented approach for the hidden curve removal can be extended to the surface case where we have 2 degrees of freedom (dof) for a view direction and two for surface parameters. Similarly to the curve case, visible surface's locations are computed by solving 3 polynomial equations in 6 variables, one of which is an inequality constraint. Assuming a freeform surface $S(u, v)$ is used to parameterize for all possible view directions $\mathcal{V}(\theta, \varphi)$, the 0-set of the 3 equations is constructed as four-manifolds in a 6-dimensional parameter space, and its projection into the $uv\theta\varphi$-domain prescribes the hidden parts of the surface $S(u, v)$. A surface's location, $S(u_0, v_0)$, is invisible from viewing direction $\mathcal{V}(\theta_0, \varphi_0)$ if its corresponding parameter, $(u_0, v_0, \theta_0, \varphi_0)$, belongs to the projected region of the 0-set. The boundary of the projected region is computed by introducing one more equation to the set of 3 equations, therefore generating 3-manifolds in the 4-dimensional parameter space. The visibility charts for the surface case are then constructed using the 3-manifolds in the $uv\theta\varphi$-parameter domain. A particular visibility query, which specifies θ and φ for a view direction, is resolved by extracting one-manifold curves in the surface's uv-parameter domain. Those curves in the uv-domain trim away hidden surface regions and thus only the visible surfaces are rendered from that view direction.

The topological structure of the visibility chart is further analyzed in the same framework, which provides a reliable solution to the computation of the visibility chart. The number of connected curve segments that delineate the hidden parts from the visible ones changes at critical points where the global topology changes in the visibility chart. *Aspect graphs* [3] are used in computer vision to topologically analize the visibility problem. In this paper, algebraic constraints for these critical points are derived as a set of 3 polynomial equations in 3 variables for the curve case and precomputed for the global analysis of the visibility chart. Based on this topological information, it becomes easier to analyze the global arrangement of the visibility chart, avoiding the computation of complex combinatorial curve-curve intersections.

The rest of this paper is organized as follows. In Section 2, the hidden curves removal algorithm is discussed for planar curves. Section 3 presents its extension to the elimination of hidden surfaces. Some examples are presented in Section 4 and finally, in Section 5, this paper is concluded.

2 Continuous Visibility for Planar Curves

Let $\mathcal{V}(\theta)$ be a one-parameter family of viewing directions. The visibility for a planar curve $\mathbf{C}(t)$ is then solved by lifting the problem into a higher dimension, where the answer is represented using simultaneous solution of two polynomial equations.

Lemma 1. *A planar curve point* $\mathbf{C}(t)$ *is visible if and only if it satisfies the following two polynomial equations for all r,*

$$\mathcal{F}(t,r,\theta) = \mathcal{V}(\theta) \times (\mathbf{C}(t) - \mathbf{C}(r)) = 0,$$
$$\mathcal{G}_1(t,r,\theta) = \langle \mathcal{V}(\theta), \mathbf{C}(t) - \mathbf{C}(r) \rangle \leq 0.$$

Proof. Two equations, $\mathcal{F}(t,r,\theta) = 0$ and $\mathcal{G}_1(t,r,\theta) \leq 0$, are satisfied only if $\mathbf{C}(t)$ is closer to the view source than $\mathbf{C}(r)$ while two curve points are on the same line to the view direction $\mathcal{V}(\theta)$. Therefore, there may be no other curve point $\mathbf{C}(r)$ that blocks $\mathbf{C}(t)$ from $\mathcal{V}(\theta)$ if $\mathbf{C}(t)$ satisfies the above two equations for all r, which implies that $\mathbf{C}(t)$ is visible from the viewing direction. ■

Figure 1 demonstrates Lemma 1. Given a viewing direction $\mathcal{V}$, two curve points $\mathbf{C}(t)$ and $\mathbf{C}(r)$ in Figure 1(a) satisfy the first equation $\mathcal{F}(t,r,\theta) = 0$. This means that the vector from $\mathbf{C}(t)$ to $\mathbf{C}(r)$ is parallel to the view direction. The second condition is satisfied only if $\mathbf{C}(t)$ is closer to the view source than $\mathbf{C}(r)$. Thus, the curve point $\mathbf{C}(t)$ is visible for the view direction $\mathcal{V}$, while $\mathbf{C}(r)$ is not. For the curve point $\mathbf{C}(t)$ to be visible, $\mathcal{G}_1(t,r,\theta) \leq 0$ should be satisfied for all r. This implies that if there is any value of r such that $\mathcal{G}_1(t,r,\theta) > 0$, then the curve point $\mathbf{C}(t)$ is not visible. In Figure 1(b), $\mathbf{C}(t)$ is potentially visible from $\mathcal{V}$ if one considers the curve point $\mathbf{C}(s)$ as its corresponding pair. The point $\mathbf{C}(t)$, however, is not visible since there exists another curve point $\mathbf{C}(r)$ that fails at the second constraint of Lemma 1.

Elber et al. [6] solves two polynomial equations in two variables for a discrete set of view directions. If $\mathcal{V}$ is one such direction,

$$\mathbf{C}'(t) \times \mathcal{V} = 0,$$
$$(\mathbf{C}(t) - \mathbf{C}(r)) \times \mathcal{V} = 0.$$

Solution points of these two equations prescribe the visible portion of $\mathbf{C}$ for each $\mathcal{V}$, providing only a discrete solution. In this paper, we solve the problem of computing visible regions for all possible view directions $\mathcal{V}(\theta)$ in the domain, simultaneously, providing a continuous solution to the visibility problem.

For the clarity of explanation, we consider *invisible* curve segments instead.

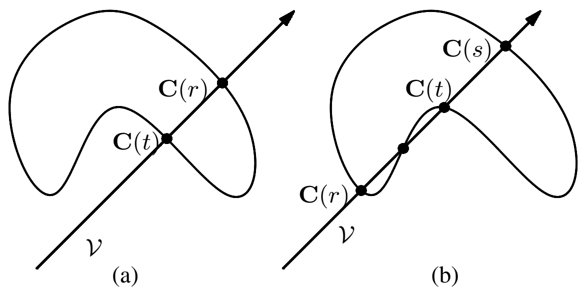

Fig. 1. (a) Given a viewing direction $\mathcal{V}$, a planar curve point $\mathbf{C}(t)$ is visible while $\mathbf{C}(r)$ is not. (b) A point $\mathbf{C}(t)$ has another curve point $\mathbf{C}(r)$ which makes it invisible from the view direction $\mathcal{V}$.

Corollary 1. *A planar curve point* $\mathbf{C}(t)$ *is* invisible *if and only if there exists another curve point* $\mathbf{C}(r)$ *such that the following two polynomial equations hold*

$$\mathcal{F}(t, r, \theta) = \mathcal{V}(\theta) \times (\mathbf{C}(t) - \mathbf{C}(r)) = 0, \tag{1}$$

$$\mathcal{G}_2(t, r, \theta) = \langle \mathcal{V}(\theta), \mathbf{C}(t) - \mathbf{C}(r) \rangle > 0. \tag{2}$$

Now, any r for which $\mathcal{G}_2(t, r, \theta) > 0$ holds renders curve point $\mathbf{C}(t)$ invisible. As this second equation, $\mathcal{G}_2(t, r, \theta) > 0$, is an inequality constraint, the solution of both constraints is a 2-manifold in 3-D parameter space. Furthermore, the solution is symmetric with respect to the $t = r$ plane so, we can consider one more inequality constraint, $t > r$, to speed up the equation-solving process by purging half the solution domain.

Denote by $\mathcal{M}$ the solution of Equations (1) and (2) that determines the hidden parts of the planar curve $\mathbf{C}(t)$. The projection of $\mathcal{M}$ into the $t\theta$-plane characterizes the regions where the curve is not visible. That is, if a parameter (t, θ) falls into the projected region of $\mathcal{M}$, then the corresponding curve point $\mathbf{C}(t)$ is not visible for the viewing direction $\mathcal{V}(\theta)$. Its complement, the uncovered region (under this projection) in the $t\theta$-plane, determines all the visible sections of $\mathbf{C}$ along continuously varying view directions. Figure 2 shows an example of such a visibility chart. Gray regions in Figure 2(a) represents the 2D projection of $\mathcal{M}$ for the planar curve $\mathbf{C}(t)$. Given a viewing direction $\mathcal{V}$, one can extract a set of visible curve segments from the uncovered (white) regions (see Figure 2(b)).

As one can see from Figure 2(b), visibility queries are resolved by extracting corresponding white regions from the visibility chart. Thus, solving the visibility problem for planar curves can be reduced to that of finding boundary curves of the projected regions of $\mathcal{M}$ in the parameter space. Since the projection is performed to the $t\theta$-plane, the boundary of the projected region under this projection occurs either at the boundaries of the zero-set $\mathcal{M}$ or at its local extrema. Since $\mathcal{M}$ is continuous and closed, it has no boundary and hence, the visibility problem reduces to finding r-extrema of the zero-set $\mathcal{M}$ which are the r-directional silhouettes of $\mathcal{M}$.

Definition 1. *Given a one-parameter family of viewing directions* $\mathcal{V}(\theta)$*, a* C^1*-continuous planar curve* $\mathbf{C}$*, and the solution manifold* $\mathcal{M}$ *of Equations (1) and (2) for* $\mathbf{C}$*;*

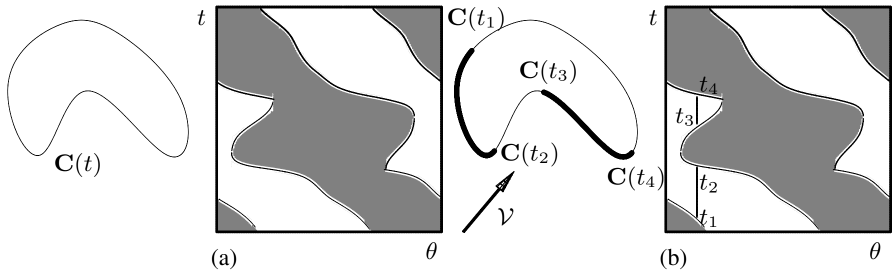

Fig. 2. (a) Given a planar curve $\mathbf{C}(t)$, the gray region in the $t\theta$-plane represents hidden curves of $\mathbf{C}$. (b) Visible curve segments can be extracted from the uncovered (white) regions.

1. *The r-directional silhouette curves, S^r, comprise the set of points on $\mathcal{M}$ whose r-directional partial derivative vanishes (bold lines in Figure 3(a) shows the projection of S^r in the $t\theta$-plane).*
2. *Denote by $S_I^r \subset S^r$ the set of points that falls in the interior of the projection of $\mathcal{M}$, among the set of r-directional silhouettes S^r (see dotted line segments in Figure 3(b)).*

Then, the sought boundary of $\mathcal{M}$, $\partial\mathcal{M}$, that delineates the visible segments of $\mathbf{C}$ from all possible views, can be computed using the two sets S^r and S_I^r as:

$$\partial\mathcal{M} = S^r - S_I^r.$$

Figure 3(c) presents $\partial\mathcal{M}$ in bold lines and $\mathcal{M}$ as a shaded region.

The r-directional silhouette curves, S^r, of $\mathcal{M}$ can be computed by finding the simultaneous solution of Equations (1), (2) and (3), where

$$\frac{\partial\mathcal{F}}{\partial r}(t, r, \theta) = 0. \tag{3}$$

Having two equality equations in three variables, solutions of the three equations are curves in the $tr\theta$-parameter space. As $\mathcal{F}$ and $\mathcal{G}_2$ are piecewise rational functions, the solution can be constructed by exploiting the convex hull and subdivision properties of NURBS, yielding a highly robust divide-and-conquer computation [5]. The solver [5] recursively subdivides rational functions along all parameter directions until a given maximum depth of subdivision or some other termination criteria is reached. At the end of the subdivision step, a discrete set of points are numerically improved into a highly precise solutions using a multivariate Newton-Raphson iterative stage. Finally, these discrete points are connected into a set of piecewise linear curves in the parameter space (See [23] for more details).

An entire curve segment or any portion of the curve segment in S^r can fall inside the projected region of $\mathcal{M}$ (see Figure 3(a)). We need to trim away S_I^r from S^r since they correspond to interior curve segments. An efficient and robust algorithm for purging S_I^r away is presented in this section and is based on the analysis of a topological change in the visibility charts. Given a continuous one parameter family of view directions

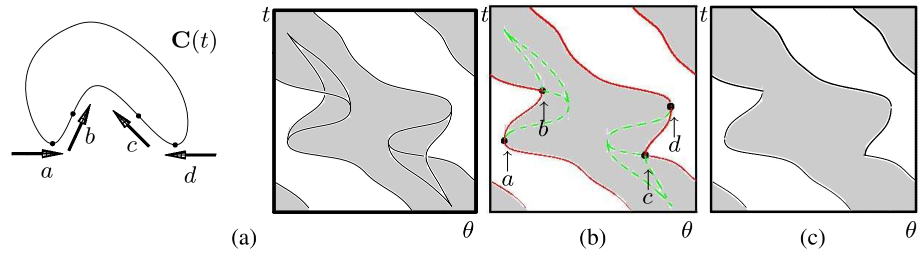

Fig. 3. (a) r-direction silhouette curves S^r projected into the $t\theta$-plane. (b) Dotted line segments represent S^r_I and (c) $\partial\mathcal{M} = S^r - S^r_I$ is shown in bold. Critical points are computed using a topological analysis and shown in (b). Their corresponding curve points and view directions are also shown in (a).

$\mathcal{V}(\theta)$, a topological change (i.e. a change in the number of connected components) can occur either globally or locally. Global topological changes occur where the viewing direction is parallel to a bi-tangent line segment of $\mathbf{C}$ connecting two (or more) points. Topological changes occur locally where the viewing direction is parallel to the tangent direction of $\mathbf{C}$, at an inflection point.

The bi-tangent line segment of $\mathbf{C}$ touches tangentially the curve at two or more different points. Bi-tangent directions can be computed by simultaneously solving the following three equations, in three variables:

$$\mathcal{F}(t, r, \theta) = 0,$$

$$\frac{\partial \mathcal{F}}{\partial t}(t, r, \theta) = \langle \mathcal{V}(\theta), N(t) \rangle = 0, \tag{4}$$

$$\frac{\partial \mathcal{F}}{\partial r}(t, r, \theta) = \langle \mathcal{V}(\theta), N(r) \rangle = 0. \tag{5}$$

Equations (4) and (5) constrain the viewing direction $\mathcal{V}(\theta)$ to touch $\mathbf{C}$ tangentially at two different points $\mathbf{C}(t)$ and $\mathbf{C}(r)$, respectively. The bi-tangent direction of $\mathbf{C}$ itself can be computed using two polynomial equations in two variables. In this context, however, the viewing direction $\mathcal{V}(\theta)$, which is parallel to the bi-tangent direction, must be computed for further processing. Inflection points of a planar curve occur at points where the sign of the curvature, a rational form if $\mathbf{C}$ is rational, changes. Solution points of $t = r$ clearly satisfy all the above equations and must be purged away.

Let $\mathcal{T}$ be a set of points (t, r, θ) in the $tr\theta$-parameter space that correspond to either bi-tangents or inflection points. We constrain point $(t, r, \theta) \in \mathcal{T}$ to be outside the projected region. The black bold dots in Figure 3(b) represents these critical points, at which the topological structure of the visibility chart changes. Thus, the r-directional silhouette curves, S^r, are trimmed at such critical points $(t, r, \theta) \in \mathcal{T}$. The curve segments S^r_I (Dotted line segments in Figure 3(b)) can be determined using a simple visibility check of a single point, testing whether the segment falls inside the projected region of $\mathcal{M}$ or not. Figure 3(c) shows the visible boundaries $\partial\mathcal{M}$ of the projected regions as a set of piecewise curves.

3 Continuous Visibility for Freeform Surfaces

The presented algorithm for computing visibility of planar curves can be extended for computing the hidden surfaces. Given two-parameters family of viewing directions $\mathcal{V}(\theta, \varphi)$, the visibility problem for the surface case is solved in a six-dimensional parameter space, $(u, v, s, t, \theta, \varphi)$. Much like the curve case, this higher dimensional formulation simultaneously considers all view directions in the domain, and provides a reliable solution to a particular visibility query. We first present a set of conditions for determining whether a surface location $\mathbf{S}(u, v)$ is visible or not.

Lemma 2. *A surface point $\mathbf{S}(u, v)$ is invisible if and only if there exists another surface point $\mathbf{S}(s, t)$ such that*

$$\mathcal{F}(u, v, s, t, \theta, \varphi) = \left\langle \mathbf{S}(u, v) - \mathbf{S}(s, t), \frac{\partial \mathcal{V}}{\partial \theta}(\theta, \varphi) \right\rangle = 0, \tag{6}$$

$$\mathcal{G}(u, v, s, t, \theta, \varphi) = \left\langle \mathbf{S}(u, v) - \mathbf{S}(s, t), \frac{\partial \mathcal{V}}{\partial \varphi}(\theta, \varphi) \right\rangle = 0, \tag{7}$$

$$\mathcal{H}(u, v, s, t, \theta, \varphi) = \langle \mathbf{S}(u, v) - \mathbf{S}(s, t), \mathcal{V}(\theta, \varphi) \rangle > 0, \tag{8}$$

where $\mathcal{V}(\theta, \varphi)$ is a polynomial approximation to the sphere that spans all possible viewing directions.

Proof. By Equations (6) and (7), the two surface points $\mathbf{S}(u, v)$ and $\mathbf{S}(s, t)$ are on the same line with the same direction to the view direction $\mathcal{V}(\theta, \varphi)$. By satisfying Equation (8), $\mathbf{S}(s, t)$ is closer to the view source than $\mathbf{S}(u, v)$, which makes $\mathbf{S}(u, v)$ invisible for that view direction. ∎

Since Equation (8) is an inequality constraint, the simultaneous zeros of the three Equations (6) – (8) are 4-manifolds in a six-dimensional parameter space. Let $\mathcal{M}$ be the 4-manifold zero-set of Equations (6) – (8). Then, similarly to the curve case, the projection of the zero-set into the $uv\theta\varphi$-domain prescribes the hidden parts of the surface $\mathbf{S}(u, v)$. If (u, v, θ, φ) falls into the interior of the projected region of $\mathcal{M}$, then the corresponding surface location, $\mathbf{S}(u, v)$, is not visible from viewing direction $\mathcal{V}(\theta, \varphi)$. In other words, the uncovered region (under this projection), in the $uv\theta\varphi$-domain, determines all the visible sections of $\mathbf{S}(u, v)$ along continuously varying viewing directions. In Figure 4(a), a shaded region depicts the projection of the zero-set, $\mathcal{M}$, into the $uv\theta\varphi$-parameter space. A parameter $(u_1, v_1, \theta_1, \varphi_1)$ falls into the projected region in Figure 4(a) and thus, its corresponding surface point $\mathbf{S}(u_1, v_1)$ is invisible for viewing direction $\mathcal{V}(\theta_1, \varphi_1)$ (see Figure 4(b)). On the other hand, point $\mathbf{S}(u_2, v_2)$ is visible since parameter $(u_2, v_2, \theta_1, \varphi_1)$ is located outside the projected region.

Projected into the $uv\theta\varphi$ four-dimensional space, the boundaries of the projection of the zero-set $\mathcal{M}$ can be determined as the st-directional silhouettes of $\mathcal{M}$, by finding all the simultaneous zeros of Equations (6) – (9), where

$$\mathcal{I}(u, v, s, t, \theta, \varphi) = \langle \mathcal{V}(\theta, \varphi), \mathbf{N}(s, t) \rangle = 0, \tag{9}$$

and $\mathbf{N}(s, t)$ is a normal vector field of $\mathbf{S}(s, t)$. The common zero-set of Equations (6) – (9) is now a 3-manifold in a six-dimensional space, which is the boundary of the

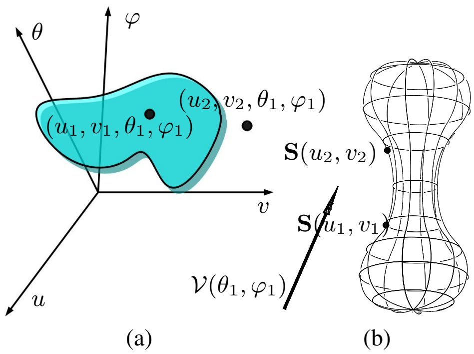

Fig. 4. (a) A shaded volume depicts a projection of the solution $\mathcal{M}$ into the $uv\theta\varphi$-parameter space. (b) $\mathbf{S}(u_1, v_1)$ is invisible for a viewing direction $\mathcal{V}(\theta_1, \varphi_1)$ since $(u_1, v_1, \theta_1, \varphi_1)$ falls into the projected volume. Compare it with $\mathbf{S}(u_2, v_2)$.

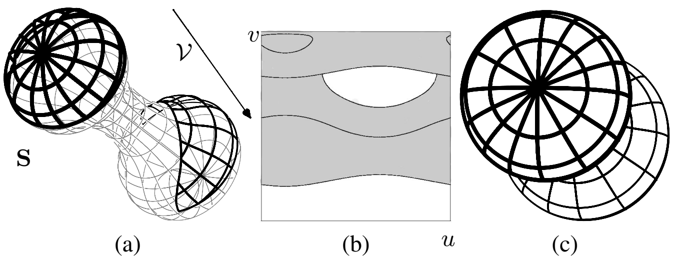

Fig. 5. (a) A surface $\mathbf{S}$ with a viewing direction $\mathcal{V}$. (b) A set of trimming curves in the uv-parameter domain. (c) Visible parts of $\mathbf{S}$ are shown for the given view direction.

projected volume of $\mathcal{M}$. Given a particular viewing query $\mathcal{V}(\theta_0, \varphi_0)$, two of the solution space's remaining degrees-of-freedom are fixed and we can extract 1-manifold solution curves from the projected region of $\mathcal{M}$. These curves in the parameter space correspond to curves that delineate the hidden surfaces from the visible ones.

It is quite difficult to either visualize or contour 3-manifolds in a six-dimensional space. By fixing a particular viewing direction, 1-manifold curves in a six-dimensional space result. So it is possible to use the algorithm presented by Seong et al [23] to extract all the visible parts of $\mathbf{S}(u, v)$. Figure 5(a) shows a surface $\mathbf{S}$ with a viewing direction $\mathcal{V}$. The boundary curves of visible sections in the uv-domain are computed using our approach (see Figure 5(b)). In Figure 5(c), gray-colored trimming surfaces represent hidden surfaces of the original surface and the bold ones are visible sections for the viewing direction. Shaded regions in the parameter domain (Figure 5(b)) correspond to the hidden surfaces in Euclidean space (Figure 5(c)).

4 Experimental Results

We now present examples of computing a visibility chart in a continuous domain for both planar curves and 3D surfaces. For all the figures, the gray-colored region

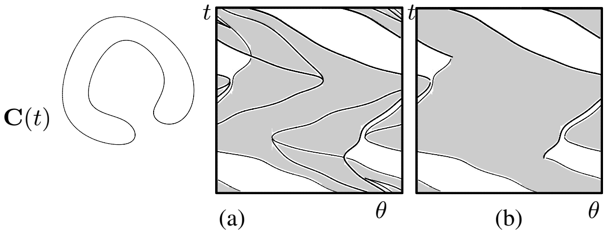

Fig. 6. (a) Given a planar curve $\mathbf{C}(t)$, the projected region of $\mathcal{M}$ and projected r-directional silhouette curves S^r are shown in gray and bold lines, respectively. (b) A set of visible segments, S_v^r, is shown in bold lines.

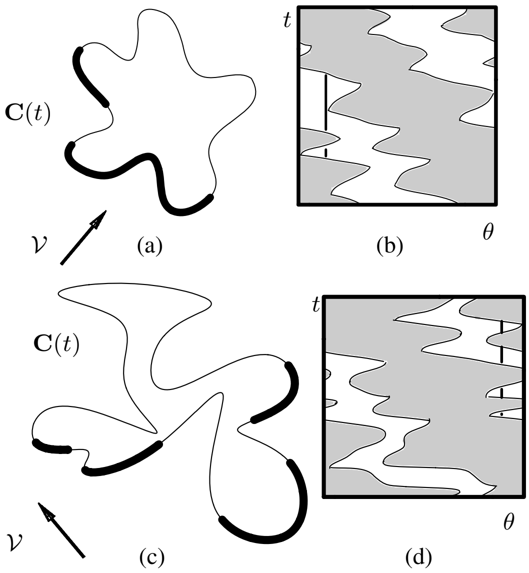

Fig. 7. (a), (c) A planar curve $\mathbf{C}(t)$ and the visible curve segments that are shown in bold lines. (b), (d) A continuous visibility charts computed by solving Equations (1) – (3).

represents the projection of the zero-set of the corresponding set of polynomial equations in the parameter space and characterizes hidden parts of planar curves or surfaces. Bold lines in curves or surfaces represents visible parts from the given view direction.

Figure 6 shows a planar curve and its visibility charts in a continuous domain. Bold lines in Figure 6(a) represent a set of r-directional silhouettes of the zero-set manifold. The boundary curves of the projected region are computed based on a topological analysis of the visibility charts and shown in Figure 6(b).

In Figures 7, (a) and (c) show two planar curves and (b) and (d) are the visibility charts for all viewing directions. For a particular viewing direction, $\mathcal{V}$, a set of visible curve segments are shown in bold lines in Figures 7(a) and (c). Figures 7(b) and (d) show the corresponding parameter domain in thick lines. The computation time for generating the visibility charts over all possible view directions for the curve case vary

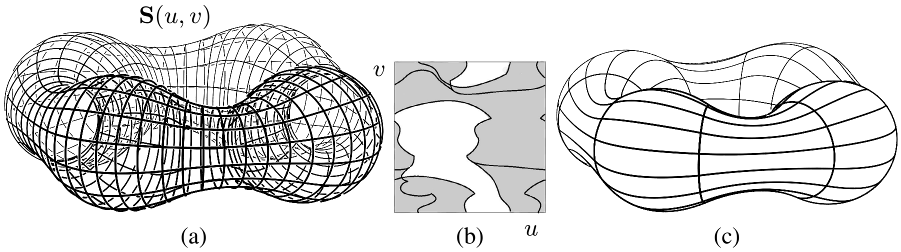

Fig. 8. (a) An envelope surface generated by sweeping a scalable ellipsoid along a space trajectory is shown. (b) A set of trimming curves in the uv-parameter domain is presented in bold lines. (c) Visible parts of the surface are shown for the given viewing direction.

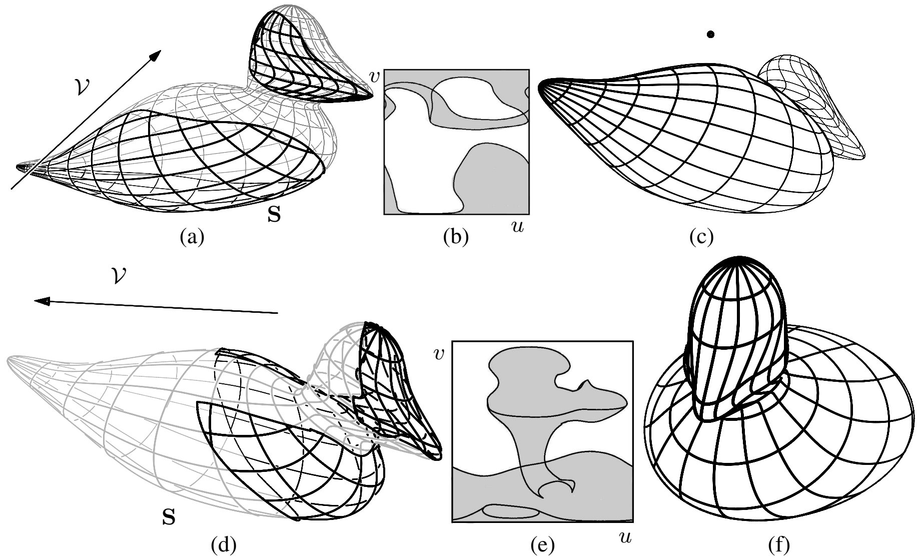

Fig. 9. (a), (d) A surface $\mathbf{S}$ is shown with a view direction. (b), (e) A set of trimming curves in the uv-parameter domain is presented in bold lines. (c), (f) Visible parts of the surface are shown for the given viewing direction.

according to the curve's complexity, taking from 1.3 to 6 seconds on a Pentium IV 2GHz desktop machine.

Figure 8(a) shows an envelope surface generated by sweeping a scalable ellipsoid along a space trajectory. A set of trimming curves is shown in Figure 8(b), which is the result of solving Equations (6) – (9) after fixing a viewing direction. Each trimmed surface sub-region is tested for visibility using a simple ray-surface intersection method. Figure 8(c) draws visible surface patches only.

The original surfaces in Figure 9(a) and (d) are bi-quartic NURBS having about 250 control points and shown with different view directions. Figure 9(b) and (e) show a set of trimming curves which are boundaries between visible parts and hidden surfaces in

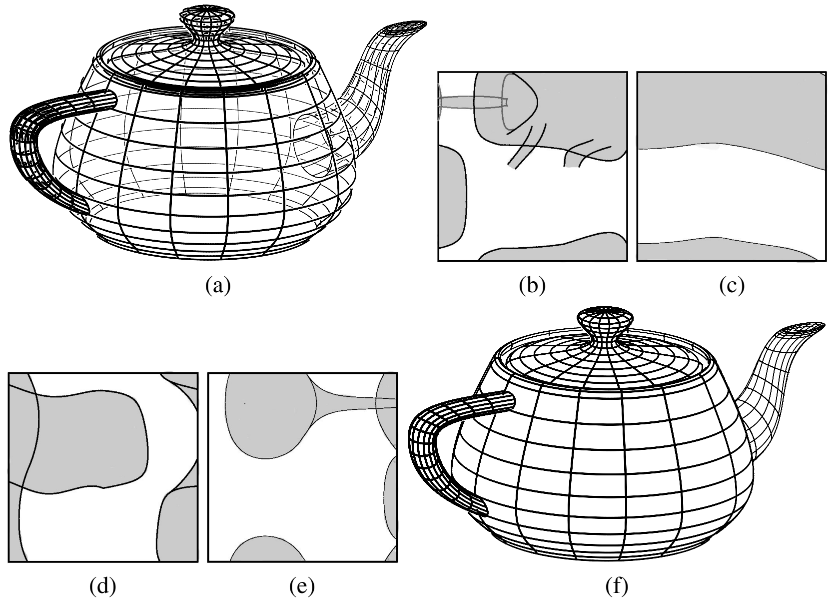

Fig. 10. (a) A teapot is presented by four surface patches. A set of trimming curves in the parameter domain of the body (b), handle (c), spout (d) and the cap (e). Trimmed surfaces are shown in (f) which are visible for the viewing direction.

the uv-parameter domain. Figure 9(c) and (f) show visible surface patches only along a specified viewing direction. On a 2GHz Pentium IV machine, computing the trimming curves in the uv-domain for Figures 8 – 10 took about 13 to 45 seconds.

The teapot in Figure 10 is represented by four bi-cubic NURBSs surfaces which are open (Figure 10(a)). Each of the four surface patches can be hidden by any of the other ones according to the viewing direction. In Figure 10(a), part of the body is blocked by both a handle and a cap for the given viewing direction (a figure is generated along the viewing direction). Furthermore, it blocks itself and makes shadow regions. Figure 10(b) shows the trimming curves in the parameter domain of the body. They are comprised of three set of curves. Trimming curves generated due to a cap are represented by gray-colored lines in Figure 10(b) and four open curve segments located in the middle part of the domain are generated by the handle. Since the surface patch of the handle is not closed, the trimming curves are also open. Thus, the geometric intersection curve between the handle and the body is needed for a proper trimming. All the other trimming curves in Figure 10(b) stems from the body itself. Figure 10(c)–(e) show a set of trimming curves for the handle, spout and the cap, respectively. Finally, Figure 10(f) draws all the visible parts.

5 Conclusion and Future Work

We have presented a robust and efficient scheme for computing hidden curve/surface removal, in the continuous domain. The approach is based on the derivation of a set of algebraic constraints that determine the visibility of curve's or surface's locations. All view directions in the domain are considered simultaneously, and the algorithm provides a continuous chart for the visibility from all possible views. By simultaneously solving 2 polynomial equations for a curve case and 3 polynomial equations for a surface case, in the parameter space, the presented approach can detect all the hidden parts of the sculptured model for continuously varying view directions. The zero-set of the polynomial equations prescribes the hidden parts of the model and we construct a visibility chart by projecting the zero-set into an appropriate parameter space. Furthermore, the topological structure of the visibility chart is analyzed in the same framework, providing a reliable solution to the computation of the visibility chart.

The presented approach can be applied to trimmed models as well. The original trimming curves need to be considered in the computation of the boundary curves between visible and invisible parts in the case of trimmed models. Visibility computations for perspective views are desirable extensions to the method presented. To this end, we need to deal with even higher-dimensional solution spaces.

Acknowledgments

All the algorithms and figures presented in this paper were implemented and generated using the IRIT solid modeling system [12] developed at the Technion, Israel. This work was supported in part by NSF IIS0218809. All opinions, findings, conclusions or recommendations expressed in this document are those of the author and do not necessarily reflect the views of the sponsoring agencies.

References

1. A. Appel. The Notion of quantitative Invisibility and the Machine Rendering of Solids. *Proceedings ACM National Conference*, 1967.
2. W.J. Bouknight. A procedure for Generation of Three-Dimensional Half-toned Computer Graphics Representations. *CACM*, Vol. 13, No. 9, 1969.
3. K. Bowyer and C. Dyer. Aspect graphs: An introduction and survey of recent results. *Proc. SPIE Conf. on Close Range Photogrammetry Meets Machine Vision*, 1395, pp. 200–208, 1990.
4. E. Catmull. A Subdivision Algorithm for Computer Display of Curved Surfaces. *Ph.D Thesis*, Report UTEC-CSc-74-133, Computer Science Department, University of Utah, Salt Lake City, UT, 1974.
5. G. Elber and M. S. Kim. Geometric Constraint Solver Using Multivariate Rational Spline Functions. *Proc. of International Conference on Shape Modeling and Applications*, pp 216–225, MIT, USA, June 15-17, 2005.
6. G. Elber, R. Sayegh, G. Barequet and R. Martin. Two-Dimensional Visibility Charts for Continuous Curves. *Proc. of ACM Symposium on Solid Modeling and Applications*, Ann Arbor, MI, June 4-8, 2001.
7. G. Elber and E. Cohen. Hidden curve removal for free form surfaces. *Computer Graphics*, Vol. 24, No. 4, pp. 95–104, 1990.
8. J. Foley, A. Van Dam, J. Hughes, and S. Feiner. Computer Graphics: Principles and Practice. Addison Wesley, Reading, Mass., 1990.

9. R. Galimberti and U. Montanari. An algorithm for Hidden Line Elimination. *CACM*, Vol. 12, No. 4, pp. 206–211, 1969.
10. C. Hornung. A Method for Solving the Visibility Problem. *CG&A*, pp. 26–33, July 1984.
11. C. Hornung, W. Lellek, P. Pehwald, and W. Strasser. An Area-Oriented Analytical Visibility Method for Displaying Parameterically Defined Tensor-Product Surfaces. *Computer Aided Geometric Design*, Vol. 2, pp. 197–205, 1985.
12. IRIT 9.0 User's Manual, October 2000, Technion. http://www.cs.technion.ac.il/∼irit.
13. T. Ju, F. Losasso, S. Schaefer, and J. Warren. Dual Contouring of Hermite Data. *In Proceedings of SIGGRAPH 2002*, pp. 339–346, 2002.
14. T. Kamada and S. Kawai. An Enhanced Treatment of Hidden Lines. *ACM Transaction on Graphics*, Vol. 6, No. 4, pp. 308–323, 1987.
15. M. S. Kim and G. Elber. Problem Reduction to Parameter Space. *The Mathematics of Surface IX (Proc. of the Ninth IMA Conference)*, London, pp 82–98, 2000.
16. S. Krishnan and D. Manocha. Global Visibility and Hidden Surface Algorithms for Free Form Surfaces. Technical Report: TR94-063, University of North Carolina, 1994
17. L Li. Hidden-line algorithm for curved surfaces. *Computer-Aided Design*, Vol. 20, No. 8, pp. 466–470, 1988.
18. P. Loutrel. A Solution to the Hidden-line Problem for Computer Drawn Polyhedra. *IEEE Transactions on Computers*, Vol. C-19, No. 3, pp. 205–213, 1970.
19. M. Mckenna. Worst-Case Optimal Hidden-Surface Removal. *ACM Transaction on Graphics*, Vol. 6, No. 1, pp. 19–28, 1987.
20. M. Mulmuley. An efficient algorithm for hidden surface removal. *Computer Graphics*, Vol. 23, No. 3, pp. 379–388, 1989.
21. T. Nishita, S. Takita, and E. Nakamae. Hidden Curve Elimination of Trimmed Surfaces Using Bezier Clipping. *Proc. of the 10th International Conference of the Computer Graphics on Visual Computing*, pp. 599–619, Tokyo Japan, 1992.
22. Y. Ohno. A Hidden Line Elimination Method for Curved Surfaces. *Computer-Aided Design*, Vol. 15, No. 4, pp. 209–216, 1983.
23. J. K. Seong, G. Elber, and M. S. Kim. Contouring 1- and 2-Manifolds in Arbitrary Dimensions. *Proc. of International Conference on Shape Modeling and Applications*, pp 216–225, MIT, USA, June 15-17, 2005.
24. I. Sutherland, R. Sproull, and R. Schumacker. A Characterization of ten Hidden-Surface Algorithms. *Computer Surveys*, Vol. 6, No. 1, pp. 1–55, 1974.
25. K. Weiler and P. Atherton. Hidden Surface Removal Using Polygon Area Sorting. *SIGGRAPH77*, pp. 214–222, 1977.
26. T. Whitted. An Improved Illumination Model for Shaded Display. *ACAM*, Vol. 23, No. 6, pp. 343–349, 1980.