MODELING THE RIM APPEARANCE

by

W. Brent Seales
and
Charles R. Dyer

Computer Sciences Technical Report #931
May 1990

Modeling the Rim Appearance

W. Brent Seales

Charles R. Dyer

Department of Computer Science
University of Wisconsin
Madison, Wisconsin 53706 USA

Abstract

Representing the rim of a 3D solid shape and the occluding contour that it generates is a difficult problem because of their dependence on viewpoint and on the local and global geometry of the shape. Describing how the rim and occluding contour change as a function of viewpoint organizes the features of the occluding contour for indexing and matching. This organization makes the features of the occluding contour explicit for matching in a dynamic context where image features are changing over time, and in a static context where matching methods must iteratively refine an estimation of viewpoint.

This paper introduces a novel, viewer-centered approach to modeling the geometry of the visible occluding contour of solid 3D shape. The *rim appearance representation* models the exact appearance of the occluding contour formed by the edges of a polyhedron which is assumed to be an approximation of a smooth shape. An algorithm is presented for constructing the rim appearance representation. Bounds on space and time are given, and implementation results show that the rim appearance representation is significantly smaller than the aspect graph or the aspect representation.

The support of the National Science Foundation under Grant No. IRI-8802436 is gratefully acknowledged.

1. Introduction

3D model-based vision depends on modeling the salient, observable features of 3D shape. One of the characteristic features of the appearance of 3D shape is the occluding contour which is generated by the projection of points on a shape which are tangent to the viewing direction. The projection process generates occluding contours, and the opacity of solid shape causes the creation of T-junctions in the image plane. The arrangement of these contours and T-junctions is directly related to the 3D properties of the shape and provides strong information for recognition [Marr77, Koen84]. Object-centered object models do not explicitly represent the properties of the occluding contour since the occluding contour is not generated by any specific set of object features. Viewer-centered models of 3D shape are better suited to encode the dynamic, viewpoint-dependent nature of the occluding contour.

This paper introduces a novel approach to explicitly modeling the geometry of the visible occluding contour for polyhedra. The exact appearance of the occluding contour is modeled in a viewer-centered representation called the *rim appearance representation*. This representation models the occluding contour formed by the edges in a polyhedron which is assumed to be an approximation of a smooth 3D shape. The rim appearance representation explicitly represents self-occlusion and the exact appearance of the occluding contour. Since much of the observable geometry of the self-occlusion of 3D shape is preserved in polyhedral approximations, it is not necessary to assume a polyhedral world. Further, the geometry of self-occlusion includes viewpoint so that the dynamic changes in occlusion relationships are made explicit in this representation. The visible occluding contours of a 3D shape are represented as a piecewise-continuous function of viewpoint.

The 3D points which generate the occluding contour change with viewpoint, but contours in the image plane deform smoothly, and features such as T-junctions and contour extrema span sets of viewpoints despite the change in points which generate them. The smooth evolution of the occluding contour is discontinuous at a finite set of isolated viewpoints where topological changes in the structure of the occluding contour occur. The rim appearance representation encodes an approximation of both the smooth evolution and the points of discontinuity as explicit, connected structures which can be organized and manipulated directly.

Recent work in the modeling of dynamic geometric features has concentrated on the aspect graph, a graph which enumerates all of the topologically-distinct 2D views of a 3D object, as well as the transitions between views. The aspect graph has been constructed for polyhedra [Plan90, Gigu88, Bowy89], solids of revolution [Egge89, Krie89], and piecewise-smooth objects [Srip89, Ponc90]. The rim appearance representation is different from the aspect graph in two major ways. First, the rim appearance representation stores only the appearance of the rim, that is, the occluding contour. The behavior of the occluding contour of a polyhedron is a strict subset of all of the topologically-distinct changes in the appearance of the polyhedron. Thus the rim appearance representation is significantly smaller than the aspect graph for polyhedra. In addition, the rim appearance representation models only those features which are observed in the images of smooth objects. Topologically-distinct views in the aspect graph which are artifacts of the polyhedral approximation are omitted. Although aspect graphs have been constructed for piecewise-smooth shapes, the numerical complexity and implicit form of the representation makes it difficult to extract and represent the features of interest. This problem is avoided by relying on polyhedra.

The second way in which the rim appearance representation differs from the aspect graph is that the rim appearance representation stores the dynamics of the occluding contour at a level of abstraction above the individual edges which form it. Although the individual edges of a polyhedron and the self-occlusion of those edges change with viewpoint, the occluding contour itself is represented as an explicit object. The interaction between parts of the occluding contour is represented as a *contour event*, a higher-level event than the interaction of the individual edges of a polyhedron. This abstraction provides a natural way to organize geometric constraints based on the occluding contour.

This paper analyzes the geometry of the polyhedral rim and presents an algorithm for constructing the rim appearance representation. Section 2 defines more formally the rim and the occluding contour, and presents properties of the rim which make it both desirable but difficult to represent. The basis for the rim appearance representation is the apparent intersection of two and of three linear edges. The geometry of these apparent intersections and the definition of the polyhedral rim are discussed in Section 3. Section 4 defines aspect space, describes the algorithm for constructing the rim appearance representation, and details the key features of the occluding contour which are modeled. The organization of the features of the occluding contour is discussed in Section 5. Section 6 shows results from an implementation which constructs the rim appearance representation, and Section 7 gives concluding remarks.

2. Representing the Visible Occluding Contour

The rim appearance representation is based on the edges and vertices of an underlying 3D polyhedral object which is assumed to be an approximation of some real, smooth object. The choice of features to represent is derived from an understanding of the geometry of the occluding contour for smooth shapes. The definitions below distinguish between the rim and the occluding contour for smooth shape and provide a motivation for the basis of the rim appearance representation for polyhedra. The geometric definition of the polyhedral rim is presented in Section 3.

There are two distinct sets of points involved in the definition of the occluding contour: the rim points, a set of points on the 3D object, and the occluding contour, a set of 2D points in the projected image. The *rim*, or contour generator, is the set of 3D points on the smooth surface of the object at which the line of sight of the viewer is tangent. The *occluding contour* is the set of points formed by the projection of the visible rim points into the 2D image. That is, some rim points may be occluded by part of the object when projected into the occluding contour. Unless otherwise noted, the term occluding contour will refer to the projection of only the visible rim points.

The occluding contour of 3D shapes is difficult to represent for two primary reasons. First, the occluding contour is dependent on viewpoint. A point is on the rim, by definition, if the normal vector to the surface at that point is perpendicular to the viewing direction. A small change in viewpoint can completely change the set of rim points, and hence the occluding contour. Second, opaque shapes cause self-occlusion. For transparent objects, the space curves defined by the complete set of rim points for a particular viewpoint are in general closed. The *visible* rim curves for opaque objects, however, are not closed contours. In general, the locus of visible rim points for smooth 3D shapes forms a set of disjoint continuous curves in $\mathbb{R}^3$, and that locus varies continuously with viewpoint.

Methods which attempt to model the occluding contour have concentrated on both qualitative features of the contour [Koen84, Rich88] and quantitative representations [Ponc89, Vail89, Broo83]. The difficulty in predicting the appearance of the occluding contour is directly related to the type of model representation chosen [Basr88]. The occluding contour of polyhedral objects is a result of the projection of linear rim edges which do not change continuously with viewpoint. The discrete nature of the polyhedral rim makes the extraction of edges on the occluding contour quite easy. Other object models which attempt to more realistically model smooth curved surfaces increase the complexity of computing the exact occluding contour [Besl88]. Generic smooth surfaces [Koen84] have desirable mathematical properties but are impractical as computer models. Parametric patches approximate generic surfaces but are still numerically very complex [Ponc89]. Generalized cylinders [Marr77, Broo83] are difficult to represent because of non-linear deformations and arbitrary spines. Simplifying assumptions about the form of the spine and the type of sweeping rules reduces the complexity. The implicit form, however, makes it difficult to extract the occluding contour.

The class of polyhedral 3D shapes is the earliest and most fundamental representation for solid shape in computer vision and computer graphics. Pioneering work in computer vision and early work in realistic image synthesis was based almost solely on the geometry of polyhedra. Much of the work in 3D recognition still relies on the simplicity and compactness of the polyhedron (for example [Thom87, Murr89]). Experience in both computer vision and graphics has shown that the speed and simplicity of linear representations can compensate for size increases and approximation error, provided the appropriate feature information is retained [Lowe89]. In the present context, it is important to preserve the occluding contour and the interactions between contours as a result of self-occlusion. The local approximation of smooth surfaces with planar patches preserves these contour features while affording linearity. Further, the approximation can be made arbitrarily close by selective linear refinement where necessary.

The rim of a polyhedron is well-defined, and can be computed exactly for a given viewpoint. The set of visible rim edges are exactly those edges which satisfy a set of visibility conditions. Polyhedra eliminate the continuous property of the rim because the set of rim edges corresponding to a given viewpoint changes discretely as the visibility of edges changes. These characteristics make polyhedra acceptable as the basis for an explicit approximate model of the occluding contour.

3. The Geometry of Self-Occlusion

The underlying model for the rim appearance representation is the polyhedron. The geometry of a polyhedron is simple because of its linearity and compact representation, but polyhedra introduce geometric features such as the edge-vertex and edge-edge-edge events which are not present in smooth shapes. Because it is assumed that the polyhedron is an approximation of a smooth object, it is necessary to make the correspondence between the geometry of the polyhedron and the visual features in the occluding contour of the smooth object.

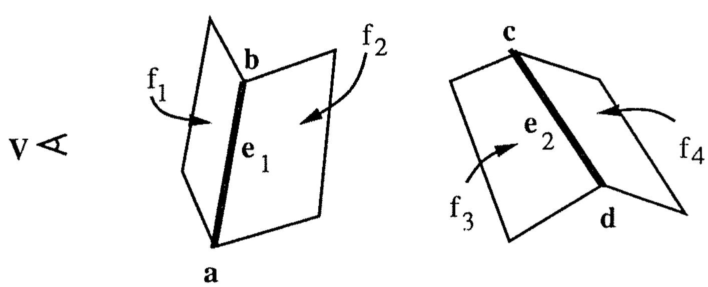


Fig. 1. Two edges defined by the faces which meet to form them.

This section is divided into three parts. First, the rim of a polyhedron is defined. A polyhedral rim edge is defined by the local orientation of the surfaces which meet to form the edge. This definition approximates the behavior of the smooth rim as viewpoint changes. Second, the visual events which cause topological changes in the projected line drawing of a polyhedron are described. Finally, the geometry of a T-junction formed by the apparent intersection of two edges is presented. These geometric properties are the basis for the rim appearance representation which is described in detail in Sections 4 and 5.

3.1. The Polyhedral Rim

The faces of a polyhedron are planar and meet at edges where there is a surface orientation discontinuity. A face does not turn away from the viewer smoothly, but rather disappears instantaneously at the viewpoint where the face is edge-on to the viewer. The smooth occluding contour is approximated by defining an edge to be on the rim when the two faces which meet at the edge are oriented such that exactly one of the faces is turned away from the viewer.

The geometric conditions for a rim edge can be specified in terms of edges, vertices and surface normals. Specifically, let e_1 and e_2 be two edges in $\mathbb{R}^3$ as shown in Fig. 1. The faces f_1 and f_2 meet to form e_1, and the faces f_3 and f_4 meet at edge e_2. We denote the directed unit normal to f_i as $\mathbf{n}_i$, and the viewpoint as $\mathbf{V}$. A rim edge is defined as follows:

Definition: An edge of a polyhedron is on the rim if and only if exactly one of the two faces which form the edge is visible, i.e.,

$$e_1 \text{ on rim} \Leftrightarrow (\text{visible}\,(f_1)\text{ XOR visible}(f_2))$$

f_i is visible from $\mathbf{V}$ means that $\mathbf{V}$ is in the positive halfspace defined by the directed normal of f_i. For example, e_1 in Fig. 1 is on the **rim** when either of the following two geometric conditions hold:

$$\begin{matrix} (\mathbf{V}-\mathbf{a})\cdot\mathbf{n}_1 > 0 \\ (\mathbf{V}-\mathbf{a})\cdot\mathbf{n}_2 < 0 \end{matrix} \quad \text{or} \quad \begin{matrix} (\mathbf{V}-\mathbf{a})\cdot\mathbf{n}_1 < 0 \\ (\mathbf{V}-\mathbf{a})\cdot\mathbf{n}_2 > 0 \end{matrix} \qquad (1)$$

These rim constraints define the rim locally. Since rim edges can interact globally, this definition does not say anything about the visibility of e_1 from **V**. Hence the local definition of a rim edge must be augmented with the geometric constraints which arise from the global interaction of rim edges.

3.2. Visual Events

The projected edges of a polyhedron form a line drawing in the image plane. Viewpoints where the connectivity, or topology, of the line drawing changes are viewpoints where visual events occur. A viewpoint which causes a topological change in the line drawing from any infinitesimal change in viewing direction is called a visual event. It has been shown that all such visual events can be found for polyhedra by computing the edge-edge-edge event (EEE-event) [Plan90, Gigu88]. The EEE-event is the apparent intersection of three non-adjacent edges. A degenerate case of the EEE-event is the edge-vertex event (EV-event), where two of the edges actually meet at a vertex. See [Plan88, Gigu88] for more details on properties of EEE-events.

3.3. The Geometry of the T-junction

The primary event affecting the appearance of the visible rim is the formation of a T-junction which results from self-occlusion. The T-junction is not an infinitesimal visual event; T-junctions persist over spans in viewpoint and are bounded by the visual events introduced above. A quantitative representation of a T-junction which occurs as a result of self-occlusion must include both the bounding visual events where the T-junction appears and disappears, as well the quantitative appearance of the T-junction in the image plane. This section analyzes the geometry of a T-junction between two edges. The geometry is used for rim edges to compute the exact appearance of the occluding contour. The goal of computing the interaction between edges which form T-junctions is to encode the dynamic properties of a T-junction, represent the exact appearance of the rim, and to make explicit the dynamic changes which occur in the occluding contour with respect to viewpoint.

In order to determine whether two edges in $\mathbb{R}^3$ form a T-junction from some viewpoint we first note some geometric constraints. Two edges which form a T-junction can be labeled *occluding* and *occluded*

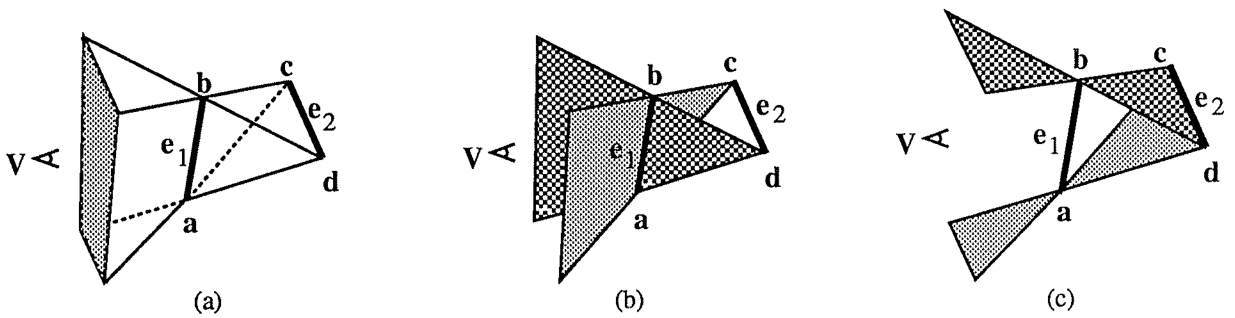


Fig. 2. (a) The two edges e_1 and e_2 form a T-junction in viewpoint space defined by four planes which form a tetrahedron from the four vertices belonging to the two edges. (b), (c) The boundaries of the space where the T-junction occurs.

where the occluding edge is the one closest to the viewer. One basic constraint is that only *rim* edges can be labeled as occluding edges. Further, only convex edges can be occluding edges. Convex edges are those which are formed by faces which meet at a convex angle. This restricts the number of potential T-junctions since only a subset of the edges is convex, and each of these is on the rim from a restricted set of viewing directions.

For smooth objects, the occluded edge can be either a smooth, occluding boundary or a surface discontinuity. Since we are assuming that the polyhedron approximates a smooth object, we restrict the occluded edge to also be on the rim. This restriction can be relaxed to also include edges which correspond to true surface discontinuities in a piecewise-smooth object.

The geometry of the visibility of two edges which form a T-junction in the image plane forms a partition of viewspace. Two edges e_1 and e_2 form a partition of space into a region where the edges will project to form a T-junction. The planes which bound this region are defined by the vertices at the endpoints of the two edges. These four planes form a tetrahedron with four triangular sides (see Fig. 2(a)). Each of the planar sides is defined by a set of three vertices: plane $(\mathbf{a},\mathbf{b},\mathbf{d})$ and plane $(\mathbf{a},\mathbf{b},\mathbf{c})$ share the common edge $\overline{\mathbf{ab}} = e_1$; plane $(\mathbf{c},\mathbf{d},\mathbf{a})$ and plane $(\mathbf{c},\mathbf{d},\mathbf{b})$ share the common edge $\overline{\mathbf{cd}} = e_2$.

A viewpoint which causes two edges to form a T-junction must lie within the region bounded by the planes which form the tetrahedron above. This restriction is expressed by the following four T-junction inequalities:

$$
\begin{array}{l}
[(\mathbf{c}-\mathbf{b}) \times (\mathbf{d}-\mathbf{b})] \cdot (\mathbf{V}-\mathbf{b}) < 0 \\
[(\mathbf{c}-\mathbf{a}) \times (\mathbf{d}-\mathbf{a})] \cdot (\mathbf{V}-\mathbf{a}) < 0 \\
[(\mathbf{a}-\mathbf{c}) \times (\mathbf{b}-\mathbf{c})] \cdot (\mathbf{V}-\mathbf{a}) > 0 \\
[(\mathbf{a}-\mathbf{d}) \times (\mathbf{b}-\mathbf{d})] \cdot (\mathbf{V}-\mathbf{a}) > 0
\end{array}
\tag{2}
$$

Thus e_1 occludes e_2 from viewpoint **V** if and only if (1) both e_1 and e_2 are on the rim (the rim constraints shown in Eq. (1) are satisfied), and (2) the viewpoint is inside the region bounded by the four planar sides of the tetrahedron (the T-junction constraints shown in Eq. (2) are satisfied). Geometrically, the T-junction inequalities determine the visibility of the four planes defining the tetrahedron. The two faces of the tetrahedron containing e_1 must be visible from **V**; the other two planes which contain e_2

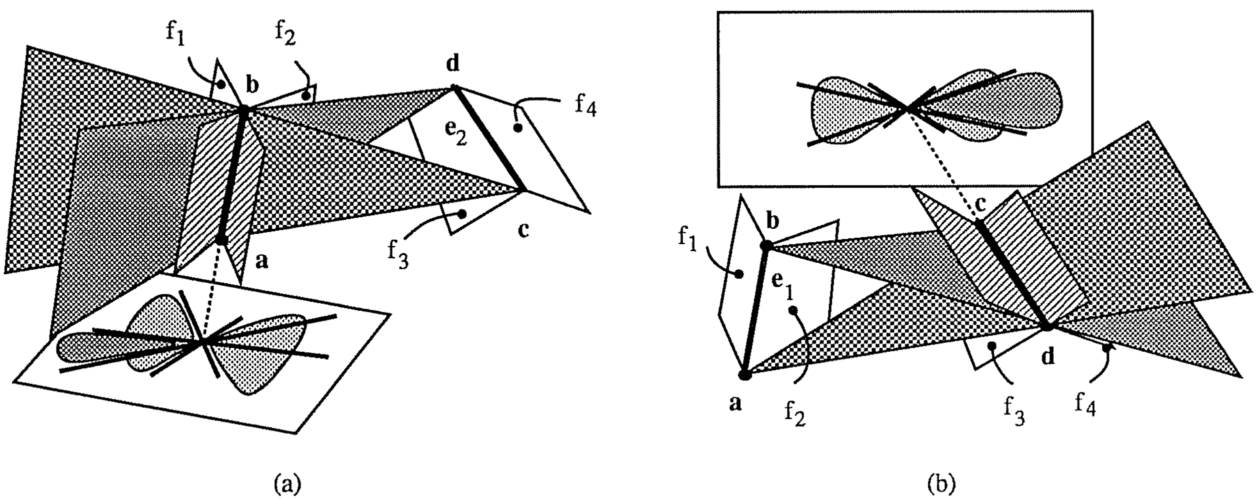


Fig. 3. (a) The regions on the plane normal to the edge e_1 defined by the intersection of the four planes meeting at e_1 determine whether the rim constraints and the T-junction constraints are satisfied for e_1. (b) The same geometry for the four planes meeting at e_2 determine whether the rim constraints and the T-junction constraints are satisfied for e_2.

must not be visible. Visible means that the viewpoint **V** is in the plane's positive halfspace. The tetrahedron is oriented so that the *outside* is beyond e_1 and e_2, and the *inside* is between them. The viewpoints in the shaded region in Fig. 2(a) are those which satisfy Eq. (2).

Eqs. (1) and (2) specify the geometric relationship between two edges which potentially form a T-junction. That is, given the set of all **V** that satisfies the rim constraints for e_1 and e_2, it can be determined if there is *any* **V** which also satisfies the T-junction constraints. A direct and efficient method for solving this problem uses the geometry of the regions in $\mathbb{R}^3$ formed by two edges. The two faces f_1 and f_2 and the two planes defined by **a**,**b**,**c** and **a**,**b**,**d** all intersect at e_1 (see Fig. 3(a)). The viewpoints which satisfy the rim constraints for e_1 are bounded by the planes for f_1 and f_2. The viewpoints which satisfy the T-junction constraints must lie within the two planes defined by **a**,**b**,**c** and **a**,**b**,**d**. Hence, the intersection of these two regions must be non-empty in order for a T-junction to occur. Call the necessity of this non-empty region condition (1). Likewise, the two faces f_3 and f_4 and the two planes defined by **c**,**d**,**a** and **c**,**d**,**b** all intersect at e_2 (see Fig. 3(b)). The viewpoints which satisfy the rim constraints for e_2 are bounded by the planes for f_3 and f_4. The viewpoints which satisfy the T-junction constraints must lie within the two planes defined by **c**,**d**,**a** and **c**,**d**,**b**. Hence, the intersection of these two regions must also be non-empty in order for a T-junction to occur. Call the necessity of this non-empty region condition (2).

The geometric constraints in Eqs. (1) and (2) are mutually satisfied if and only if the two edges form a T-junction. That is, all the constraints are satisfied if and only if condition (1) and condition (2) hold. So we can prove the following:

Theorem: The rim conditions (1) and (2) hold if and only if edges e_1 and e_2 form a T-junction for some viewing direction.

Determining whether conditions (1) and (2) are satisfied for two specific edges is done using the four planes involved with each edge and the plane orthogonal to the edge. Fig. 3(a) shows the intersection of the four planes with the horizontal plane normal to e_1. The two large shaded regions on this horizontal plane which are opposite each other represent areas where e_1 is on the rim. The longer shaded area shows where e_1 occludes e_2. Any overlap in these shaded areas in the plane orthogonal to e_1 implies that condition (1) is satisfied. This overlap is computed efficiently by the projection into the orthogonal plane of the vectors in each of the four support planes. The test for this overlap amounts to an ordering of four vectors in the orthogonal plane. Fig. 3(b) shows the analogous geometry for condition (2).

In summary, the analog of rim points in polyhedra is the set of edges satisfying the geometric property that only one of the two faces meeting at the edge is turned toward the viewer. The visual events which affect a change in the topology of the edges in the line drawing of a polyhedron are completely determined by the EEE-event, where the EV-event is a special case of the EEE-event. Thus the definition of the rim edge can be combined with the visual event computation in order to compute exactly those viewpoints where the polyhedral rim changes topologically. The T-junction is a persistent visual interaction, and the viewpoints where T-junctions are created and annihilated are bounded by the EEE-event. The geometry of the T-junction combined with the definition of the rim provides a fast test to determine whether two rim edges interact to form a T-junction. Once two edges are known to form a T-junction, a quantitative description of the T-junction in the image plane can be computed. This description is incorporated into an algorithm for constructing the rim appearance representation which is described in the next section.

4. Constructing the Rim Appearance Representation

The rim appearance representation models the occluding contour as a function of viewpoint. Representing the rim as a function of viewpoint is related to the aspect representation [Plan88], a complete representation of appearance as a function of viewpoint. The relationship of the rim appearance representation to the aspect representation and the aspect graph depends on a single key component: *aspect space*. Aspect space, the cross-product space of the image plane $\times$ viewpoint space, is the central component in the construction of the rim appearance representation. Aspect space has been shown to be useful for encoding the exact appearance of polyhedra as a function of viewpoint [Plan88], constructing the aspect graph [Plan90, Plan88], and for the interactive animation of polyhedral scenes [Seal90]. Section 4.1 presents the important properties of aspect space, and differentiates between two representations which have been constructed using aspect space, the aspect representation and the aspect graph. Section 4.2 describes how the properties of aspect space are used as the basis of the rim appearance representation. An algorithm for constructing the rim appearance representation for a polyhedral model is given in Section 4.3.

4.1. Aspect Space and the Aspect Representation

Aspect space, the cross-product space of the image plane $\times$ viewpoint space, makes use of a multi-dimensional space to encode the appearance of objects for all viewpoints. The dimensionality of aspect space is dependent on the projection model and the geometry of objects. Aspect space for orthographic

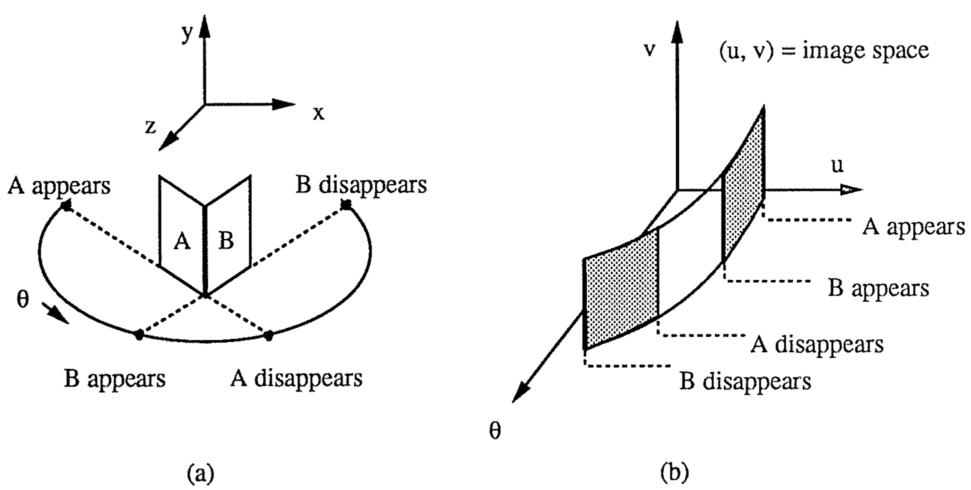


Fig. 4. (a) An edge formed by two faces A and B. (b) The surface in aspect space corresponding to the edge in (a). The visibility of the edge in (a) as a part of the rim is represented by the shaded parts of the surface.

projection is 4D since there are two degrees of freedom in viewpoint space and two degrees of freedom in the image plane. A complete discussion of 4D aspect space can be found in Plantinga's thesis [Plan88]. Features such as a vertex, an edge or a face correspond to particular structures which are contained in aspect space. These structures in aspect space represent the appearance of the feature for all viewpoints.

For example, the unit circle $\mathbf{S}^1$ under orthographic projection is a 1D space of viewpoints. Since image space is a plane, aspect space in this instance is $\mathbb{R}^2 \times \mathbf{S}^1$. We can represent this as a subspace of $\mathbb{R}^3$, specifically $\mathbb{R}^2 \times (-\pi, \pi]$. Thus a point in aspect space corresponds to a point in the image plane and a particular viewpoint.

For this 3D aspect space, a vertex (x_0, y_0, z_0) generates a 1D curve in aspect space. Points on this curve correspond to the location of the vertex in the image plane for a specific viewpoint. The equations for this curve are derived from the image coordinates of the point after a rotation by θ about the y-axis (see Fig. 4(a)). Denoting coordinates in the image plane (u, v), the result is

$$u = x_0 \cos\theta - z_0 \sin\theta$$
$$v = y_0$$

These equations specify the appearance of a vertex as a 1D curve in aspect space $(u(\theta), v, \theta)$, $-\pi < \theta \leq \pi$.

A line segment connecting two vertices $\mathbf{p}_1 = (x_1, y_1, z_1)$ and $\mathbf{p}_1 + \mathbf{a}_1 = (x_1 + a_1, y_1 + b_1, z_1 + c_1)$ can be represented parametrically as $\mathbf{p}(s) = \mathbf{p}_1 + s\, \mathbf{a}_1$, $0 \leq s \leq 1$. The line segment appears in the image at the points

$$u = (x_1 + a_1\, s) \cos\theta - (y_1 + b_1\, s) \sin\theta$$
$$v = z_1 + c_1\, s$$

which specifies the appearance of an edge as the 2D surface in aspect space $(u(s, \theta), v(s), \theta)$. This is the form of the 2D surface in aspect space shown in Fig. 4(b) which corresponds to the appearance at all viewpoints of the bold edge in Fig. 4(a).

We will use the term *visibility structure* to denote the structure in aspect space which corresponds to the visibility of a particular model feature. The dimensionality of the visibility structure for a feature depends on both the dimension of the feature and the dimension of viewpoint space. For visual simplicity we have illustrated visibility structures using a 1D viewpoint space so that the visibility structure for edges is a 2D surface and the visibility structure for a face is a 3D volume. With two degrees of freedom in viewpoint, the dimensionality of the visibility structure for faces, edges and vertices increases by 1.

A fundamental property of aspect space is that occlusion is equivalent to set subtraction in aspect space. Consider two faces and their corresponding visibility structures. A point which lies within the visibility structure for both faces is a single image point generated from both faces. Since only one face can be visible, the point is removed from the visibility structure for the face which is occluded. Thus, the exact set of visible points of a face from all viewing directions can be computed by performing set subtraction in aspect space. The visibility structure for a face is a volume bounded by algebraic surfaces. The intersection operations for two such structures can be done in closed form in a way similar to polyhedral intersection.

The aspect representation, or *asp* for short, is the explicit boundary model of the visibility structures in aspect space which correspond to the visibility of each vertex, edge and face of a polyhedral object [Plan88, Plan90]. The asp for a polyhedron quantifies the appearance of each face, taking into account the occlusion relationships between faces. The asp for a single planar face can be described exactly by computing the equations of its boundaries in aspect space. For example, the asp for face A in Fig. 4(a) is the volume in aspect space which is bounded by the surfaces swept out by the edges around face A. The surface in Fig. 4(b) is one of these boundaries which corresponds to the common edge of face A and face B. The asp was originally introduced as an intermediate structure for constructing the aspect graph, a graph which represents each topologically-distinct view of an object as a node, and the transitions between views as arcs. The construction of the asp is a well-defined procedure with known, tight bounds on the time and space requirements.

The asp for a polyhedron has several basic properties. First, the asp encodes the appearance of the polyhedron with hidden surfaces removed. The asp for a face is the volume in aspect space with occluded

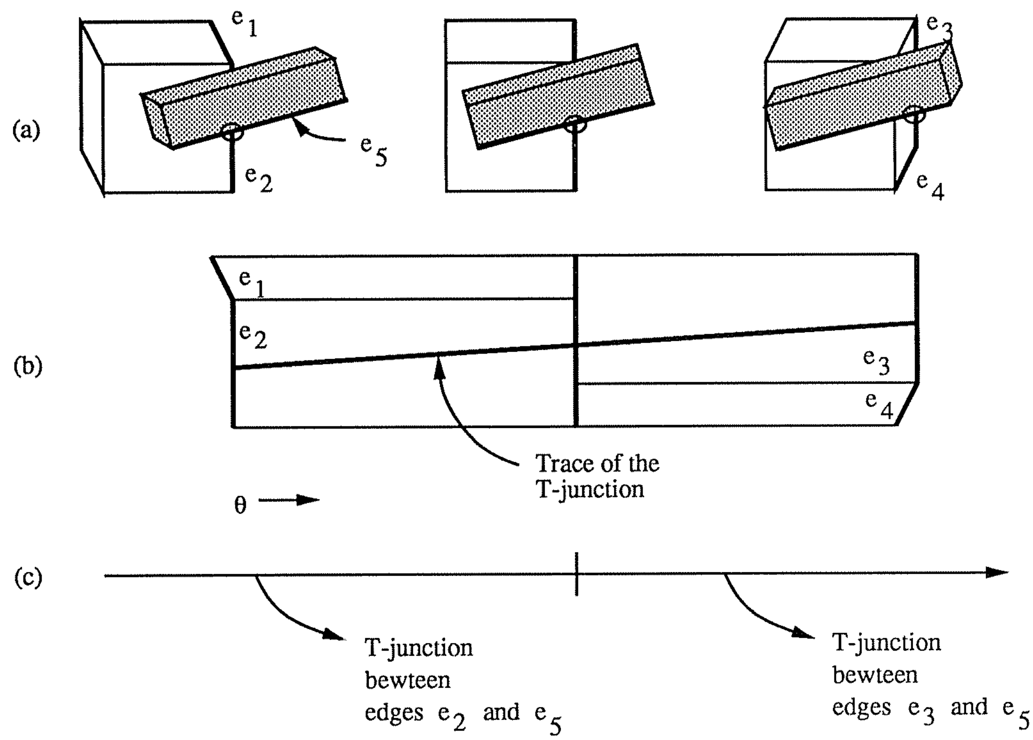


Fig. 5. (a) The rim edges of a cube being occluded by a rectangular solid. (b) The 2D surfaces in aspect space for the rim edges of the cube. The line which spans the surfaces is the trace of the T-junction between e_5 and the rim edges of the cube. (c) A linked set of intervals along the viewpoint axis represents the edges which form the T-junction.

portions removed so that global visibility information is included in the asp description. Second, a cross-section of the asp for a face yields the appearance of that face with hidden parts removed. Third, the bounding surfaces and curves of an asp volume, represented as algebraic surfaces, correspond to visual events which are not just intrinsic to the object (e.g. vertices), but also include the apparent intersection of pairs of edges (the EE-event) and the apparent intersection of triples of edges (the EEE-event).

4.2. The Appearance of the Rim

The rim appearance representation is the organized collection of the visibility structures in aspect space corresponding to the rim edges of a polyhedron. The visibility structure for a rim edge is a section of the visibility structure for an edge corresponding to the viewpoints where the edge is on the rim. Fig. 4(b) shows the visibility structure in aspect space for the bold edge in Fig. 4(a). The shaded sections of the visibility structure are those sections where the edge is on the rim. A portion of the visibility structure is cut away at the viewpoints where the edge leaves the rim so that only the visibility of the edge as a part of the rim is represented.

The asp is the visibility structure which represents the exact appearance of a face as a function of viewpoint. The asp, however, does not make explicit the visual events which affect the rim. In contrast, there are three characteristics which we feel are necessary in a representation of the rim appearance:

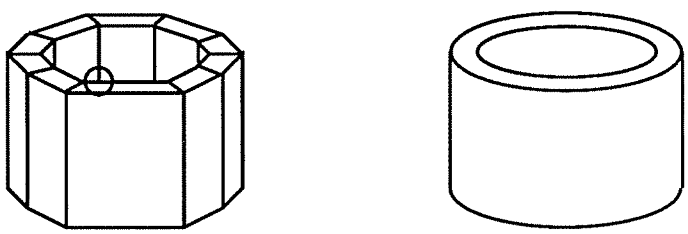
- The visibility structures for rim edges must be connected together over viewpoints where the set of rim edges changes
- Only visual events which involve rim edges should be included
- The exact appearance of the rim must be encoded, including an explicit representation of T-junctions and contour splits and merges

The following paragraphs describe how the rim appearance representation meets these goals.

Each visibility structure encodes the exact visibility of a single rim edge for all viewpoints, and structures are connected together in aspect space where they are adjacent. This explicit connection allows visual events involving the visible rim (i.e., T-junctions) to be explicitly represented as piecewise-continuous objects. Fig. 5 illustrates how the set of rim edges changes with viewpoint. As viewpoint changes, the edges which participate in the T-junction change because the set of rim edges changes. Fig. 5(b) shows how the visibility structures for the rim edges e_1 and e_2 are connected to the adjacent visibility structures for e_3 and e_4. Visibility structures for individual rim edges are connected together in aspect space and are associated as part of the same contour.

The visibility structure for an edge can be constructed so that it encodes only the visibility of the edge as part of the rim. Thus only visual events which involve rim edges are computed and stored. Fig. 6(a) shows a cylindrical shape which is approximated by a non-convex polyhedron. The visual event circled in Fig. 6(a) is part of the appearance of one of the faces of the model, and is represented in the asp. This feature will never be observable in an image, however, and hence is not represented in the rim appearance representation.

A cross-section of the rim appearance representation for a fixed viewpoint corresponds to the exact appearance of the occluding contour in the image plane. Unlike the asp, the rim appearance representation encodes the appearance of the visible occluding contour, not of individual faces. The visual events involving the rim throughout viewpoint space can be extracted and organized explicitly to define the T-junctions which occur in the projected model. In fact, constraints between pairs of these visual events can be formulated explicitly in terms of orientation, spatial relationship, and derivatives with respect to change in viewing direction. By representing the occluding contour as a function of viewpoint, the evolution of the appearance of the occluding contour over viewpoint can be extracted and summarized hierarchically from cross-sections and from visual event properties. Thus properties of the occluding



(a) (b)

Fig. 6. (a) A polyhedral approximation of a curved object. (b) The actual smooth object. The visual events like the one circled in (a) will never be produced by the object in (b). Such visual events are not represented in the rim appearance representation.

contour such as extrema and merging and splitting can be explicitly represented and organized.

Several snapshots of the appearance of the rim for a polyhedral model of a candlestick are shown in Fig. 7. These views were generated by an implementation which constructs the rim appearance representation for one degree of freedom in viewpoint. The rim appearance representation encodes each of the T-junctions between the contours of the candlestick as a single structure. The visual events which affect the T-junction are stored explicitly. For example, the top sequence in Fig. 7 shows the migration of the circled T-junction as viewpoint changes. The second view shows a triple-contour intersection (EEE-event) where two T-junctions temporarily coincide. The third view shows that the T-junction has evolved from an interaction between the base and top of the candlestick to the interaction between the base and the middle of the candlestick.

The bottom row of snapshots in Fig. 7 illustrates the splitting and merging of an occluding contour. The occluding contour corresponding to the top section of the candlestick in the leftmost view is unbroken. A slight change in viewpoint causes the contour to break at a concave edge. Likewise, part of the contour corresponding to the middle section splits at a concave edge. In the third view, the contours have merged again into a single silhouette (except for the contour at the hole in the candlestick). The merging occurs at viewpoints where the T-junctions end (EV-events) and the rim edges are connected spatially into a single circuit.

4.3. An Algorithm

The algorithm for constructing the rim appearance representation is similar to the algorithm for constructing the asp. The critical differences are (1) the connection of visibility structures in aspect space based on how the set of rim edges changes with respect to viewpoint, and (2) the restriction of the visibility structure of an edge to the rim. The steps below outline an algorithm to construct the rim appearance representation:

(1) **Compute the visibility structure for all rim edges.** The local definition of the rim restricts the

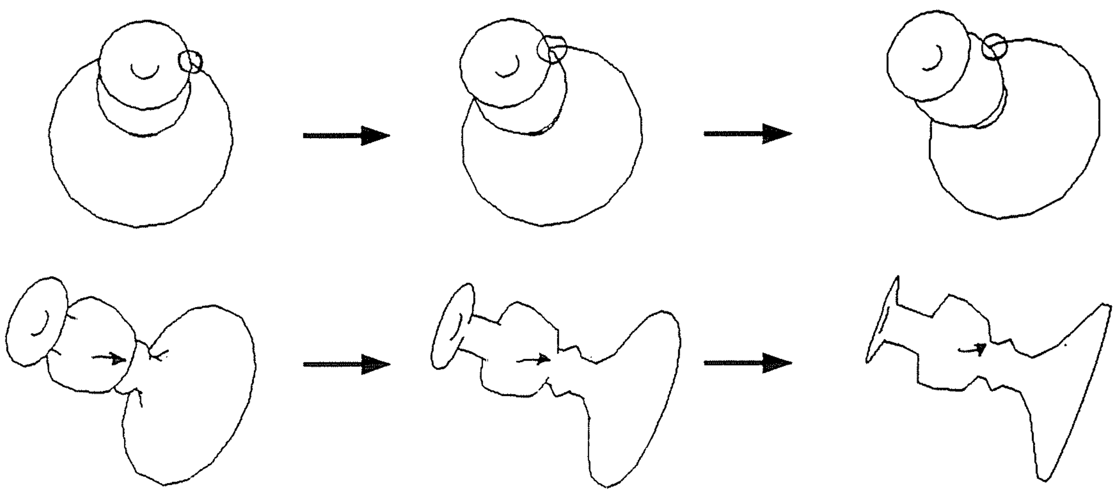


Fig. 7. The rim appearance representation encodes the T-junctions and the contours in this model as a piecewise-continuous function of viewpoint.

regions in aspect space where an edge is visible as a rim edge. The two disjoint regions in aspect space where each edge is locally defined to be on the rim are constructed.

(2) **Determine potential EE-events between rim edges.** Since only rim edges can cause visual events (of interest) with rim edges, the efficient test given in Section 3 is used to determine those pairs of edges which form a T-junction from some set of viewpoints. For such edges, their visibility structures are constructed and then modified as follows:

(a) Given the visibility structure for a single rim edge, subtract the visibility structures for all other intersecting rim edges. The resulting visibility structure is the appearance of the rim edge including the boundaries in aspect space corresponding to T-junctions formed with other rim edges. The algebraic boundaries generated by this process are fully specified by the equations for the EV- and EEE-events [Plan88].

(b) The final visibility structure for a single edge must be divided into a set of potentially disjoint volumes. This division is based on the global visibility of the edge. The global visibility test must be made to determine if the edge is fully visible or completely occluded. However, since visibility can only change at rim edge boundaries, this global test needs to be performed only once for the entire visibility structure for an edge. The result of this test can be propagated to adjacent sections of the visibility structure to determine which pieces need to be cut away as a result of total occlusion. Note that the fact that the visibility can only change at EE-events implies that the property of being totally occluded or totally visible must remain constant across a connected section of the visibility structure.

(3) **Connect the visibility structures for individual rim edges across both spatial and viewpoint dimensions in aspect space.** Spatial connections are specified by the connectivity information in the polyhedral model. Thus the visibility structures for two spatially-connected rim edges can share the curve in aspect space defined by the visibility of their common vertex. The visibility structures for two distinct edges are connected in viewpoint space at those viewpoints where one edge leaves the rim and the other edge becomes part of the rim. This "sewing" step is necessary because given this information, a single T-junction can be described across viewpoint as a piecewise-connected curve or surface which is independent of its constituent edges. That is, the events where edges disappear from the rim can be ignored and the connectivity information can be used to describe the T-junctions across these boundaries.

5. Explicit Representation of T-junctions and Contour Features

The rim appearance representation encodes the visual events which affect the rim edges of polyhedra. The interaction between edges is computed as the basis for the rim appearance representation. These visual events are used to construct an approximation of the continuously changing appearance of the occluding contour for smooth shapes. Thus we can distinguish between the edge events of the polyhedron and the abstract *contour event*, i.e., the singularities and interactions of the continuously changing rim which projects to the occluding contour. The contour event is a higher-level event than the edge events in the sense that the rim points which project to the occluding contour change with viewpoint and are not associated with a static set of points or edges. For example, Fig. 5(a) illustrates a T-junction between the rectangular solid in front and the cube behind. The T-junction is a contour feature which persists despite a change in the rim edges of the cube. The middle view is exactly the viewpoint where e_1 and e_2 are leaving the rim, and e_3 and e_4 are being added to the rim. As viewpoint changes, the edges on the rim of the cube change. The T-junction persists, however, and is represented as a contour feature which is independent of the individual edges which form it.

Contour features have been cataloged previously for generic surfaces [Koen84], piecewise-smooth surfaces [Rieg87] and developable surfaces [Srip89]. Each visual event in these catalogs occurs when the viewing direction has high-order contact with the surface along certain characteristic curves [Ponc90]. For example, lips and beaks (local events) occur along the asymptotic directions at parabolic surface points. A T-junction is created by a swallowtail, which is the result of the local properties of a saddle point. Global events such as the apparent intersection of three curves and non-transverse intersections also generate T-junctions. Recent work on constructing the aspect graph for smooth objects has made use of the catalog of visual events in order to partition viewspace into regions of constant aspect [Egge89, Krie89, Srip89].

The visual events which occur between rim edges in polyhedra are the only viewpoints where the topology of the occluding contour can change. In this sense these visual events correspond to topological changes in the occluding contours of smooth, continuous 3D shapes. The rim appearance representation organizes the low-level edge and vertex events in order to represent the topological changes of the polyhedron's occluding contour. Since the edges which form the occluding contour change with viewpoint, changes in the occluding contour are described in the rim appearance representation as a set of individual events which have been connected together spatially and across viewpoint. This second level of connected events forms an explicit representation of the contour event. The explicit representation of contour events such as T-junctions which change across viewpoint encodes the viewpoints where the feature is created or annihilated as well as the dynamics of the feature across viewpoint. Thus the rim appearance representation encodes all contour events as a continuous function of viewpoint.

The rim appearance representation organizes the contour events which are preserved in polyhedra. In the smooth case, the locus of viewpoints where visual events occur can be specified as solutions to algebraic equations. The solutions to such equations, however, are implicit and numerically complex [Ponc89]. The polyhedral approximation eliminates the numerical complexity and provides a piecewise organization that preserves many of the interesting and detectable occluding contour features.

The following sections show specifically how individual edge events are organized into higher-level contour events. These contour events consist of piecewise continuous T-junctions and the splitting and merging of the occluding contour.

5.1. T-junctions

The difficulty in representing T-junctions is the numerical complexity of the locus of surface points which define the viewing directions where T-junctions occur. In polyhedra, T-junctions are formed by the apparent intersection of two edges, the EE-event. The connection of multiple EE-events across viewpoint establishes a piecewise representation for a single T-junction between two occluding contours.

There are three properties of a T-junction which are represented in the rim appearance representation: the image location of the apparent intersection point, the orientation of the T-junction, and the viewpoints where a T-junction is created or annihilated. These properties are directly available from the geometry of the two edges which form a T-junction; the piecewise connection of EE-events represents the change in the T-junction as the rim changes.

Fig. 5 shows part of the rim appearance representation for several rim edges on a polyhedron. The 2D surface in aspect space shown in Fig. 5(b) corresponds to the rightmost two rim edges on the cube. The 1D curve which cuts across the surfaces for e_2 and e_3 is the trace of the T-junction which is formed by e_5 and the rim edges on the occluded cube. Notice that as e_2 leaves the rim of the cube and is replaced by e_3, the 1D curve representing the T-junction continues unbroken across the rim change. This connection is included in the construction of the rim appearance representation and is used to establish a representation for one particular T-junction which changes dynamically across viewpoint. Fig. 5(c) shows how the T-junction is represented as a connected set of adjacent viewpoint intervals. The boundaries between these 1D viewpoint intervals correspond to visual events and changes in the set of rim edges which form the T-junction.

The properties of a T-junction such as orientation and location in the image plane are immediately available from the geometry of the edges which form it. The viewpoints where the T-junction is created or annihilated occur where no continuation of the T-junction to other rim edges is possible. The creation and annihilation viewpoints correspond to degenerate views which produce topological changes in the occluding contour.

5.2. Contour Splitting and Merging

The occluding contours of smooth, transparent shapes are always closed curves which may contain non-smooth singularities such as cusps. These curves may overlap in the image plane and can have self-intersections. For closed transparent polyhedra, the occluding contour is always a set of closed, linked edges which can overlap and can self-intersect. For opaque objects, on the other hand, self-occlusion (the overlapping of two different closed contour loops in the image plane) causes a contour to split into disjoint parts, or to merge into a larger piece. A swallowtail illustrates the way a contour can break at the T-junction which is created by the self-occlusion of an opaque surface.

The rim appearance representation computes all of the visual events which affect the visibility of rim edges. These events include T-junctions which cause the occluding contour to split and merge, as well as the EV- and EEE-events where T-junctions begin and end. The topological behavior of the occluding contour is computed and stored by organizing these events as a function of viewpoint. The topological characteristics are precomputed, and the exact appearance of the occluding contour for any viewpoint is directly available from the rim appearance representation.

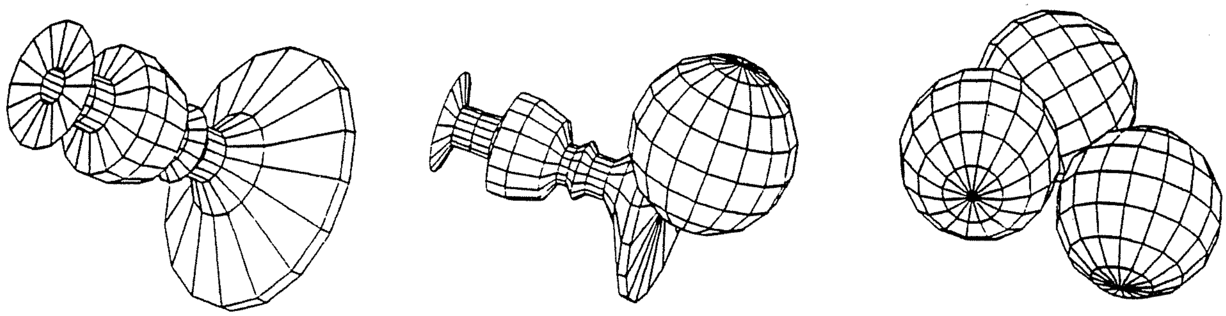
As an example, consider the middle section of the occluding contour shown in the bottom set of views in Fig. 7. The occluding contour corresponds to the convex portion of the middle of the candlestick. As viewpoint changes, the contour splits when some edges leave the rim. In the rightmost view, the middle contour merges with the others when the T-junctions disappear. These splits and merges are modeled directly in the rim appearance representation.

The high-level contour representation which is constructed connects rim edges together spatially. The viewpoints where merges and splits occur as a result of visual events are explicitly computed so that the higher-level behavior of the occluding contour as a function of viewpoint is encoded. For example, the rim edges for a convex polyhedron make up a single loop of edges. Since there are no EV- and EEE-events which cause the rim of a convex solid to split and merge, the behavior of the occluding contour is constructed as an unbroken loop of edges which never splits or merges. This higher-level structure provides a representation of the topological features of the occluding contour which is independent of the constituent individual edges.

In summary, the rim appearance representation explicitly represents the visual events which cause a change in the topology of the occluding contour. The interaction of occluding contours in the form of a T-junction is represented as a piecewise-continuous structure which is connected across the changes in the participating rim edges. The locus of viewpoints which corresponds to these visual events in the smooth case is numerically complex and difficult to make explicit. The rim appearance representation avoids these problems by using the discrete rim of polyhedra. The higher-level contour is represented as a connected set of edges which splits and merges as a function of viewpoint. This higher-level representation is important because its topology corresponds to the topology of the occluding contour of the smooth object which the polyhedron models. The splitting and merging of the occluding contour as well as the T-junctions which bound these splits and merges correspond to the dynamic behavior of the occluding contour of a smooth object. The topological behavior of the occluding contour and the explicit quantitative representation of the appearance of the occluding contour can then be used together as the basis for a model-based recognition system.

6. Theoretical and Empirical Analysis

The complexity of constructing the rim appearance representation is bounded by the number of visual events which affect the appearance of the rim. The number of vertices in the visibility structure of the rim appearance representation in aspect space determines the construction time and space complexity. Under orthographic projection with two degrees of freedom in viewpoint, each vertex in the rim appearance representation is generated by the apparent intersection of four object edges. Thus in the worst case, the rim appearance representation can be constructed in space $O(\mathbf{n}^4)$ for a polyhedron with $\mathbf{n}$ faces. Since



Model	Faces	Edges	Total Events (Aspect Rep)	Rim Events (Rim Rep)	Rim Event Percentage	Ave. Events/Edge (Aspect Rep)	Ave. Events/Edge (Rim Rep)
Candlestick	288	560	3079	838	27.2	5.5	1.5
Candlestick with Sphere	416	800	5157	1277	24.8	6.5	1.6
3 Spheres	384	720	2828	516	18.2	3.9	0.7

Fig. 8. The aspect representation and the rim appearance representation were created for the three polyhedral models above. The sixth column shows that the number of visual events computed for each of the models is reduced by 75% when computing the visual events involving only the rim edges. The visual events were generated under orthographic projection with one degree of freedom in viewpoint.

the algorithm must compute the intersection of the visibility structure for each rim edge with every other, the construction time is bounded by $O(\mathbf{n}^5)$.

These complexity bounds are the same as those for constructing the aspect representation [Plan88]. Highly pathological polyhedral shapes such as picket fences and grids can approach the worst-case behavior. The rim appearance representation, however, has a much better average case behavior because of the elimination of many of the visual events which occur in polyhedra but are not related to the rim or the occluding contour.

The construction algorithm for the rim appearance representation has been implemented for 3D aspect space. The implementation is written in C and uses an X-windows interface for object display. The table in Fig. 8 shows results generated by the program for three polyhedral models. The number of visual events stored in the rim appearance representation is compared to the number of visual events computed for the aspect representation. The storage requirement for a single visual event is approximately 20 bytes. The size of the aspect graph for the models is not shown in the table because the aspect graph is always as large or larger than the aspect representation.

The columns of the table in Fig. 8 show the total number of visual events in the aspect representation and the number of these events which are rim events. Column 6 shows the percentage of the total events which are rim events. The rightmost two columns report the average number of events per edge in the aspect representation and the average number of events per edge in the rim appearance representation. As the sixth column shows, there was approximately a 75% reduction in the number of visual events to be considered in the rim appearance representation. In the case of the candlestick, the number of visual events per edge decreased from 5.5 in the aspect representation to 1.5 in the rim appearance representation. These results indicate that the rim appearance representation saves significant time and space over aspect representation (and hence the aspect graph) while preserving the completeness of the exact appearance of the occluding contour at all viewpoints.

7. Concluding Remarks

This paper has described a novel, viewer-centered model of the occluding contour of 3D objects called the rim appearance representation. This representation models the exact appearance of the rim edges of a polyhedral approximation of a smooth object. The appearance of the polyhedral rim and the features of self-occlusion which affect its appearance correspond directly to the contour events generated by smooth objects. The linear edges of polyhedra make it possible to represent both T-junctions and occluding contour events as explicit objects which are organized in terms of sets of interacting rim edges. Representing these features explicitly is difficult in general because of the continuously changing rim points of smooth surfaces.

A viewer-centered model of the occluding contour of 3D shape is important for model-based vision. The occluding contour of 3D shape is related to 3D surface properties, and the features of self-occlusion provide strong constraints for recognition. Features such as T-junctions and contour events are intrinsic to projected shape, and can be represented and used for indexing and matching in a model-based system. The relationships between occluding contour features can further constrain the matching process. The viewer-centered approach to modeling 3D shapes makes the change in features with respect to viewpoint explicit. This change can be used in a dynamic context where image features are changing over time, or where matching methods must iteratively refine an estimation of viewpoint [Lowe89]. In addition, an explicit model which includes features of self-occlusion for solid shape makes the prediction of the appearance of the model a faster process which can speed up model matching [Basr88].

Although the rim appearance representation can be large for worst-case examples, the average case requires much less time and space. Consequently, this representation is much smaller on average than other representations such as aspect graphs or the aspect representation. The organization of individual rim edges into contour-level events gives a natural abstraction and useful structure to the visual events which affect the rim. Finally, it should be noted that this model of the occluding contour is intended to be used in conjunction with other available surface and geometric information. The integration of this 3D viewer-centered approach with other geometric features will provide a strong foundation for more sophisticated, shape-based approaches to 3D vision.

References

[Basr88] Basri, R. and S. Ullman, The alignment of objects with smooth surfaces, *Proc. 2nd Int. Conf. Computer Vision*, 1988, 482-488.

[Besl88] Besl, P. J., Geometric modeling and computer vision, *Proc. IEEE* **76**, 1988, 936-958.

[Bowy89] Bowyer, K., D. Eggert, J. Stewman, and L. Stark, Developing the aspect graph representation for use in image understanding, *Proc. Image Understanding Workshop*, 1989, 831-849.

[Broo83] Brooks, R. A., Model-based three-dimensional interpretations of two-dimensional images, *IEEE Trans. on Pattern Analysis and Machine Intell.* **5**, 1983, 140-150.

[Egge89] Eggert, D. and K. Bowyer, Computing the orthographic projection aspect graph of solids of revolution, *Proc. IEEE Workshop on Interpretation of 3D Scenes*, 1989, 102-108.

[Gigu88] Gigus, Z. and J. Malik, Computing the aspect graph for line drawings of polyhedral objects, *Proc. Computer Vision and Pattern Recognition*, 1988, 654-661.

[Koen84] Koenderink, J. J., What does the occluding contour tell us about solid shape?, *Perception* **13**, 1984, 321-330.

[Krie89] Kriegman, D. J. and J. Ponce, Computing exact aspect graphs of curved objects: Solids of revolution, *Proc. IEEE Workshop on Interpretation of 3D Scenes*, 1989, 116-122.

[Lowe89] Lowe, D. G., Fitting parameterized 3-D models to images, Technical Report 89-26, Computer Science Department, University of British Columbia, Vancouver, B. C., 1989.

[Marr77] Marr, D., Analysis of occluding contour, *Proc. Royal Society London* **197**, 1977, 441-475.

[Murr89] Murray, D., D. Castelow, and B. Buxton, From image sequences to recognized moving polyhedral objects, *Int. J. Computer Vision* **3**, 1989, 181-208.

[Plan88] Plantinga, W. H., The asp: A continuous, viewer-centered object representation for computer vision, Ph.D. Dissertation, Computer Science Department, University of Wisconsin-Madison, 1988.

[Plan90] Plantinga, W. H. and C. R. Dyer, Visibility, occlusion, and the aspect graph, *Int. J. Computer Vision*, to appear, 1990.

[Ponc89] Ponce, J. and D. J. Kriegman, On recognizing and positioning curved 3D objects from image contours, *Proc. Image Understanding Workshop*, 1989, 461-470.

[Ponc90] Ponce, J. and D. J. Kriegman, Computing exact aspect graphs of curved objects: parametric surfaces, Technical Report UIUCDCS-R-90-1579, Department of Computer Science, University of Illinois at Urbana-Champaign, 1990.

[Rich88] Richards, W., B. Dawson, and D. Whittington, Encoding contour shape by curvature extrema, in *Natural Computation*, W. Richards, ed., MIT Press, Cambridge, Mass., 1988, 83-98.

[Rieg87] Rieger, J. H., On the classification of views of piecewise smooth objects, *Image and Vision Computing* **1**, 1987, 91-97.

[Seal90] Seales, W. B. and C. R. Dyer, Shaded rendering and shadow computation for polyhedral animation, *Proc. Graphics Interface '90*, 1990.

[Srip89] Sripradisvarakul, T. and R. Jain, Generating aspect graphs for curved objects, *Proc. Workshop on Interpretation of 3D Scenes*, 1989, 109-115.

[Thom87] Thompson, D. W. and J. L. Mundy, Three-dimensional model matching from an unconstrained viewpoint, *Proc. IEEE Int. Conf. Robotics and Automation*, 1987, 208-220.

[Vail89] Vaillant, R. and O. Faugeras, Using occluding contours for recovering shape properties of objects, *Proc. IEEE Workshop on Interpretation of 3D Scenes*, 1989, 26-32.