Polytechnic
UNIVERSITY
Brooklyn · Long Island · Westchester

A Survey of Aspect Graphs

Robert D. Schiffenbauer

Department of Computer and Information Science

Technical Report
TR-CIS-2001-01
02/28/2001

A Survey of Aspect Graphs

By Robert D. Schiffenbauer
Polytechnic University
Brooklyn, New York

February 2001

This survey is submitted as one of the requirements for the Ph.D. degree in Computer Science at Polytechnic University, Brooklyn, New York.

Table of Contents

ACKNOWLEDGMENT

Special thanks to my advisor, Professor Boris Aronov of Polytechnic University, for providing copious amounts of valuable and much needed guidance while a rather lengthy sequence of revisions gradually converged to this final document.

1 Introduction

One approach to recognizing three-dimensional objects by computer is to obtain a series of two-dimensional views of a known object (Fig 1), maintain them in some convenient representation in storage, and then match one or more two-dimensional views of an unknown object against the stored views of the known object, thereby reducing the three-dimensional matching problem to a series of two-dimensional ones.

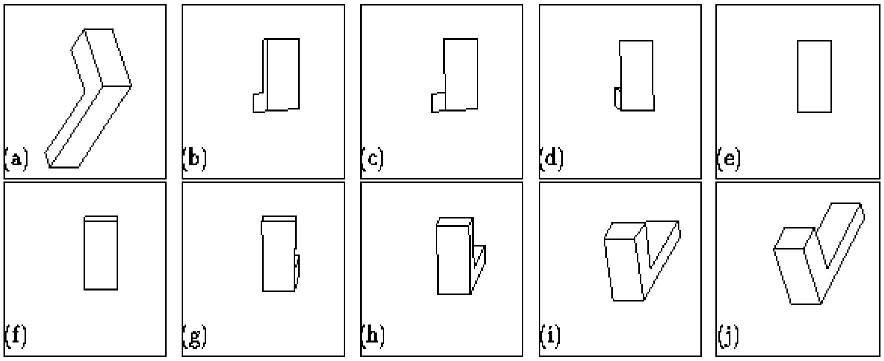

Fig 1. A series of two-dimensional views of a known (L-shaped) object. (From [28])

If some sufficiently high matching threshold is surpassed in comparing the views of the unknown object with the stored views, then the unknown object is presumed to be of the same type as the known object. This survey will look more closely at the problem of computing and storing the two-dimensional views of the known object. The *aspect graph* is a data structure that incorporates information about these views. The nodes of this graph represent the collection of *stable* views of the object which, in a sense to be made more precise below (section 2), are the only significant views associated with that object. Stored with each node is enough information to allow for the reconstruction of the stable view it represents. The aspect graph also maintains view adjacency information. Two stable views are said to be *adjacent* when it is possible for a moving viewer to pass from one view to the other with no intervening stable views. In the aspect graph, each pair of nodes representing adjacent stable views is joined by an edge.

In section 2 we define more precisely the problem domain for which aspect graphs are utilized and provide definitions for many terms that will be encountered throughout this survey. In section 3 we present an extensive examination of recent research efforts pertaining to the aspect graph for general polyhedral objects. We provide detailed analyses of the complexities of aspect graphs for collections of convex and non-convex polyhedra. Several algorithms found in the literature for the computation of aspect graphs induced by such objects are discussed in depth and their run-time complexities are compared. The study of aspect graphs for the special case of polyhedral terrains has proved a particularly fruitful area of research, and a thorough treatment is provided in section 4. Again, complexities of these aspect graphs and algorithms to compute them in this special case are examined in detail. Finally, in section 5 we summarize a number of research efforts in several related areas. One such effort concerns the computation of aspect graphs for object silhouettes. The complexities of these aspect graphs are often much lower than those induced by the entire scene. Much effort has been expended regarding the construction of useful approximations to aspect graphs, such as scale space, finite resolution and weighted aspect graphs. These data structures, which may be simpler or more easily computed than ordinary aspect graphs, nevertheless often embody enough information to be useful in various circumstances. We also discuss structures that provide useful alternatives to aspect graphs in certain cases such as the 3d visibility complex or a uniform tessellation of the viewpoint space. We mention briefly here the related issue of conservative visibility. Additional topics cited briefly in the final section include aspect graphs for objects with smooth surfaces, simplified algorithms for computing aspect graphs and the aspect graph in two dimensions.

2 Definitions

Our problem domain is defined by three parameters [25]; the type of object (or objects) that make up the scene whose aspect graph is to be determined, the set of available viewpoints from which the scene may be viewed (the *viewpoint space*) and a precise definition of what constitutes a *view* of the scene from a given viewpoint.

In this survey we focus primarily on the aspect graphs of scenes composed of collections of opaque three-dimensional polyhedral objects (with planar faces). This includes as a special case the objects known as polyhedral terrains. A large body of work exists concerning the determination of aspect graphs for scenes involving other types of three-dimensional objects, for example solids of revolution, and piecewise-smooth surfaces. However, the machinery involved is more complex than in the case of polyhedral objects and usually requires analytical methods for finding solutions to systems of simultaneous polynomial equations [22]. We will return briefly to these considerations in section 5.4.

In most research the viewpoint space is modeled in one of two ways. In the *orthographic model*, the viewpoint space consists of all points on the sphere at infinity. Each viewpoint represents a particular *direction* of observation and is defined by two parameters: θ, its *longitude*, or angle of rotation about the vertical axis, and φ, its *azimuth*, or angle from the positive vertical axis (Fig 2). Note that there exists a natural mapping from this set of viewpoints to points on the unit sphere $\mathbf{S}^2$ about the origin.

In this model all *lines of sight* emanating from a given viewpoint are *parallel* and oriented in a direction opposite to that defined by the viewpoint. Any line of sight that meets an object terminates there, and is thus a ray (all other lines of sight are complete lines). The *visible* features of the objects (vertices, edges or portions of edges, faces or portions of faces) with respect to a given viewpoint are those encountered by this collection of rays. An object feature which is not visible from the viewpoint and for which the volume immediately in front of the feature (with respect to the viewpoint) is exterior to the object itself is said to be *occluded* from that viewpoint. Thus, while convex objects possess invisible features, they have no occluded features. An occluded feature f_1 is said to be *(partially) occluded by* a feature f_2 (with respect to a given viewpoint) if some line of sight meeting f_2 can be extended to a complete line intersecting f_1.

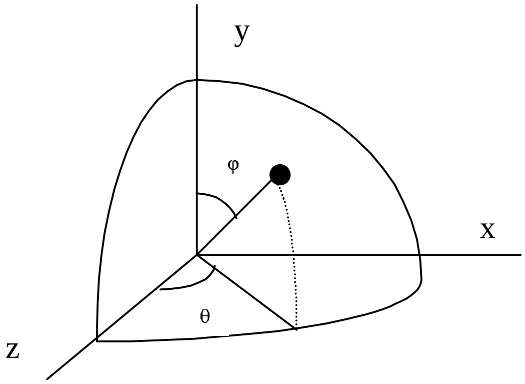

Fig 2. A point (θ, φ) on S^2 representing a viewpoint in the orthographic model. (From [25])

In the *perspective model*, the viewpoint space consists of all points in $\mathbf{R}^3$ with the exception of those points occupied by the objects themselves. Thus each viewpoint is defined by its *x*-, *y*-, and *z*-coordinates. In this model, lines of sight emanate in all directions from a given viewpoint. Here, however, any line of sight that meets an object is a line segment. Each such segment originates at the viewpoint and terminates at its point of intersection with the object. All other lines of sight are rays originating at the viewpoint. *Visible* features with respect to a given viewpoint are those encountered by the collection of line segments. *Occluded* features are defined similarly to the orthographic model. Finally, an occluded feature f_1 is said to be *(partially) occluded by* a feature f_2 (with respect to a given viewpoint) if some line of sight meeting f_2 can be extended to a ray (emanating from the viewpoint) intersecting f_1.

Suppose that, in the orthographic model, each line of sight that is a ray is extended to a complete line. Similarly suppose that, in the perspective model, each line of sight that is a line segment is extended to a ray (with endpoint at the viewpoint). The resulting collections of extended rays and line segments will both be referred to as *extended sight lines*.

In certain contexts the viewpoint space is limited to a line or line segment in $\mathbf{R}^3$ or a geodesic curve (i.e. an arc of a great circle) in $\mathbf{S}^2$. Such curves are generically referred to as *flight paths*.

We now turn to a more precise formulation of the third parameter defining our problem domain, the concept of *view*. The definition we present is the most common, but by no means the only one to be found in the literature. In the rest of this section we refer to a single object, but the discussion applies equally well to collections of objects.

In the orthographic model we select a viewpoint and project object vertices, edges and edge portions onto a plane *P* at infinity that is orthogonal to the lines of sight. A vertex or point on an edge that is met by a line of sight is projected onto *P* at the point where its extended sight line intersects *P*. Since all lines of sight are parallel, the resulting projection has no distortions due to perspective effects; features along two lines of sight *l* and *l'* separated by a distance *d* will have projections also separated by a distance *d* in the projective plane. In the perspective model we select a viewpoint and project these same features onto a sphere *S* of infinitesimal radius with center at that viewpoint. For each line of sight *l*, the object vertex or edge point met by it, if any, is projected onto *S* at the point where *l* intersects *S*. As its name implies, perspective distortions may be seen in the projection obtained under the perspective model. For example, parallel edges receding from a viewpoint will appear to meet in the projection at what in common parlance is called a *vanishing point*.

The projection of an object onto the plane *P* or sphere *S* with respect to a viewpoint is called the object's *image* from that viewpoint, and *P* and *S* are referred to as the *image plane* and *image sphere* respectively. There are two important types of images, opaque images and transparent images.

The *opaque image* of an object with respect to a given viewpoint is the projection of object vertices and edge portions visible from that viewpoint onto the image plane or sphere in the manner described above. Object vertices and edge portions that are occluded by the interior of object faces do not project onto the opaque image. In contrast, the *transparent image* of an object with respect to a given viewpoint is the projection of *all* object vertices and edges onto the image plane or sphere under the assumption that object faces are *transparent*. In other words, object faces do not cause occlusions. In terms of the above discussion, we expand all lines of sight that intersect the interior of an object face until they encounter an object edge or vertex or else pass to infinity. We then project all object features (vertices and edges) encountered by this new collection of sight lines onto the image plane or sphere. Thus the transparent image may be regarded as the projection of the *wire frame* of the object under consideration.

Opaque and transparent images are composed of three types of features, *vertices*, *edges* and *T-junctions*. Object vertices project to image vertices, and portions of object edges project to image edges. An additional set of vertices, referred to as *T-junctions*, arises due to the intersections of the projections of non-adjacent object edges. T-junctions are but one instance where image features intersect due to the *alignment* of the object features that project to them along an extended sight line emanating from the viewpoint. We will have occasion to speak more about this in section 3.1. In general, T-junctions involving two edges appear T-shaped in the opaque view, since the face adjacent to one of the object edges occludes a portion of the other object edge (Fig 3). Since object faces do not occlude in the transparent view, T-junctions involving two edges are in this case X-shaped in appearance.

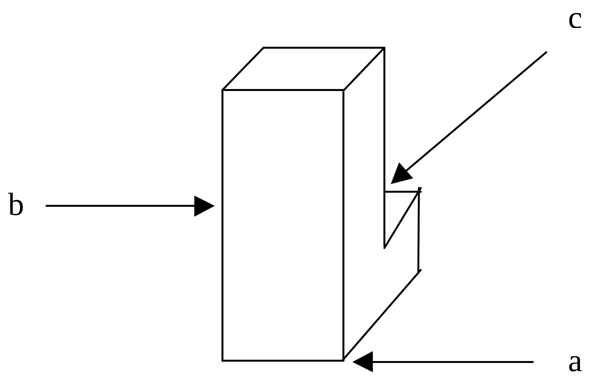

Fig 3. An opaque image showing (a) a vertex, (b) an edge and (c) a T-junction. (From [28])

The *labeled image structure graph* (LISG) of the (opaque or transparent) image is constructed by mapping image vertices (including T-junctions) to nodes and image edges to arcs, and labeling these nodes and arcs (and the faces induced by them) in some consistent way [18]. One reasonable labeling scheme involves the assignment of a distinct label to every vertex, edge and face of the original object. Image vertices that are not T-junctions and image edges receive the label of the corresponding object feature that projects to them. A T-junction is assigned a set of labels, one label from each image edge that gives rise to it. Finally, although object faces are *not* considered to project into the image, we may assign to each image face the label of the object face that would be seen there, if any, from the viewpoint (in the transparent case, we assign the label of the *closest* face, relative to the viewpoint, that would be seen there, if any). Under such a labeling scheme the LISG thus becomes an embedded, labeled, undirected planar graph. The *opaque* (*transparent*) *view* from a given viewpoint is the LISG of the opaque (transparent) image as seen from that viewpoint.

Two embedded, undirected, planar graphs A and B are *isomorphic* [18] when they possess the same qualitative (topological) structure, that is, when there exists a one-to-one correspondence, M, called an *isomorphism*, mapping the nodes (arcs and faces) of A to the nodes (respectively, arcs and faces) of B such that given any node *n* of A, if the arcs and faces adjacent to n (proceeding, say, in the clockwise direction) are a_1, f_1, a_2, f_2, . . . a_k, f_k and the arcs are adjacent to nodes n_1, n_2, . . ., n_k, respectively, then the arcs and faces adjacent to M(n) in B are M(a_1), M(f_1), M(a_2), M(f_2), . . ., M(a_k), M(f_k) (also proceeding in the clockwise direction) and the arcs are adjacent to nodes M(n_1), M(n_2), . . ., M(n_k), respectively. In addition, for a labeled graph, under the rules for assigning labels to nodes and arcs, the nodes n and M(n) must be labeled identically, as must the arcs a and M(a) and the faces f and M(f). Two views are called *distinct* when their corresponding LISG's are *not* isomorphic.

The aspect graph embodies only *opaque* views of an object. As we shall see, however, the transparent view is important because most algorithms compute the collection of transparent views of an object as an intermediate step towards computing the collection of opaque views.

To compute the aspect graph it is first necessary to partition the viewpoint space (in the orthographic case this may be regarded as *either* the sphere at infinity *or* $\mathbf{S}^2$, by virtue of the natural mapping that exists between the two) into maximally connected regions of viewpoints from which the corresponding opaque views are isomorphic. A representative viewpoint is selected from each full-dimensional cell (a two-dimensional region in the orthographic model, and a three-dimensional region in the perspective model) of this *viewpoint space partition* (VSP) and its associated opaque view is constructed. The aspect graph is the dual of the VSP. It possesses a node for each full-dimensional cell in the VSP and an arc between nodes representing *adjacent* cells (those which are separated by a region of one lower dimension in the partition). In addition, it stores at each node the opaque view of the representative viewpoint (or at least enough information to reconstruct that view [17]) (Fig 4). The aspect graph thus contains the information necessary to allow the type of two-dimensional image matching described in the introduction.

The viewpoints within the full-dimensional cells of the VSP, termed *non-critical regions*, are called *general viewpoints*. These are viewpoints for which any infinitesimal movement in viewpoint space leads to a viewpoint with an isomorphic opaque view. It is in this sense that the views associated with non-critical regions are considered *stable*; there exists a volume of viewpoint space for each such view from which that view persists (up to isomorphism) as the viewpoint is varied. In the orthographic model non-critical regions are two-dimensional cells and are bounded by one-dimensional curves. In the perspective model non-critical regions are three-dimensional cells and are bounded by two-dimensional surfaces. These curves and surfaces, termed *actual critical regions*, consist of viewpoints for which there exists some infinitesimal movement in viewpoint space leading to a viewpoint with a non-isomorphic opaque view. The viewpoints at these critical curves and surfaces are called *accidental viewpoints*. Views associated with actual critical regions are considered unstable. Because they are associated with regions in viewpoint space which are infinitesimally small they are considered insignificant for image matching, and thus are not stored in the aspect graph.

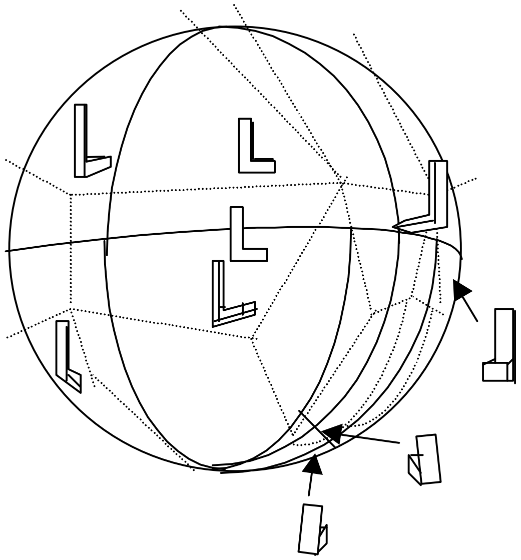

Fig 4. One hemisphere of S^2, showing a VSP (solid lines), the corresponding aspect graph (dashed lines) and selected opaque views of an object for several regions in the VSP in the orthographic model. (From [18])

Finally, we will also have occasion to discuss the partition of viewpoint space induced by the set of *transparent* views of the object. This partition, consisting of maximally connected regions of viewpoints from which the corresponding transparent views are isomorphic, is a *refinement* of the VSP. Just as for the VSP, we will refer to the full-dimensional cells of this partition as *non-critical regions* (containing *general viewpoints*). We refer to the curves or surfaces of this partition as *potential critical regions* (containing *accidental viewpoints*) since they do not necessarily appear in the VSP.

Note that the VSP is the *arrangement* [20] of the actual critical regions in viewpoint space. Similarly, the partition induced by the set of transparent views is the arrangement of potential critical regions in viewpoint space.

3 Aspect Graphs and General Polyhedra

We begin this section with a discussion of the nature of the potential critical regions induced by the transparent views of a single polyhedral object. We discuss three algorithms that use very different approaches to compute the actual critical regions of the VSP given that the set of potential critical regions has already been determined. Bounds on the complexity of the VSP in both the orthographic and perspective models are established next, and a comparison of the run-time complexities of the algorithms is presented. Subsequent to this, bounds on the time and space complexities involved in computing, storing and retrieving the stable opaque views associated with each node in the aspect graph are considered. Bounds on the complexity of the VSP induced in the special case where the scene consists of k convex polyhedral objects are established next.

Following this, several algorithms are delineated for computing the potential critical regions along a flight path in $\mathbf{R}^3$ induced by a collection of polyhedral objects. Finally, we consider a special type of object, called an *articulated assembly*, consisting of a collection of moving polyhedral parts. Each specific *configuration* of such an object yields a distinct VSP.

3.1 Critical Events and Potential Critical Regions

We begin by describing how to find the set of potential critical regions induced by an object or collection of objects, since the algorithms to be described require that these be determined before the actual critical regions can be computed. Potential critical regions, occurring at viewpoints for which there is a qualitative (topological) structural change in the LISG of the transparent view as the viewpoint moves infinitesimally in some direction, are induced by two kinds of *critical events*. An *EV-event* occurs when an *image* vertex intersects an *image* edge. This happens when the corresponding *object* vertex and non-adjacent *object* edge are aligned along an extended sight line from the viewpoint. An *EEE-event* occurs when three *image* edges (or, equivalently, an image edge and a T-junction formed by two other image edges) intersect at a point. Such an event happens when the three corresponding pairwise non-adjacent *object* edges are aligned along an extended sight line from the viewpoint (Fig 5).

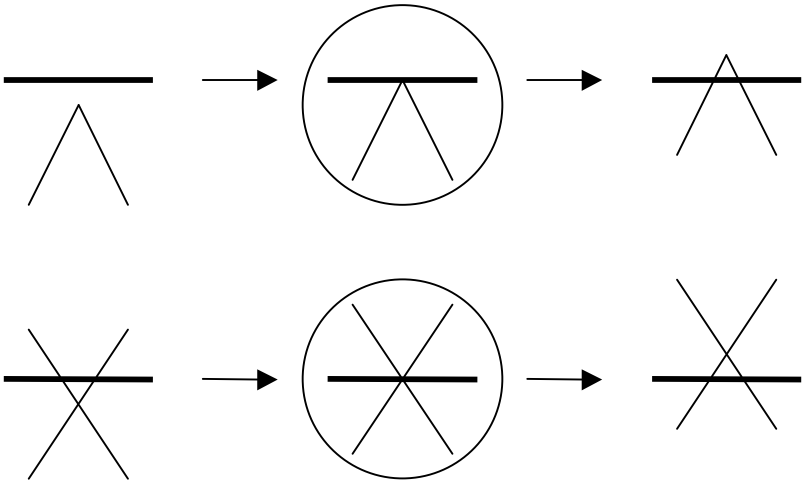

Fig 5. An EV-event (top row) and an EEE-event (bottom row). (From [17])

It can be shown [18] that these events are fundamental in the sense that any other event that induces structural changes in the LISG of the transparent image reduces to one of these. For example, the intersection of two image T-junctions as seen from a given viewpoint may be regarded as the simultaneous occurrence of several EEE-events. Again, the intersection of two image vertices is an EV-event involving one of these vertices and either of the edges adjacent upon the other vertex.

To determine the potential critical regions induced by EV-events [25], consider an object vertex *v* and an object edge *e*. In $\mathbf{R}^3$ it is clear that the only viewpoints that possess extended sight lines passing through both *v* and *e* lie in a planar region which is part of the affine hull of *v* and *e*.

This region is bounded by the lines passing through *v* and either endpoint of *e*. All viewpoints between *v* and *e* are discarded, as no extended sight line emanating from such viewpoints (recall that these are rays in the perspective model) can pass through both features simultaneously. This region is a potential critical surface in the perspective model. This surface intersects the sphere at infinity along two diametrically opposite arcs of a great circle, yielding two potential critical curves in the orthographic model (Fig 6).

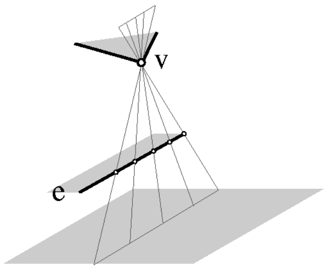

Fig 6. The potential critical region induced in $\mathbf{R}^3$ by a vertex *v* and an edge *e* (excluding the region between the features) and (at the bottom of the diagram) its intersection with a portion of the sphere at infinity. (From [14])

Special considerations exist for the EV-events induced by the vertices and edges bounding a single object face. In $\mathbf{R}^3$, every viewpoint lying on the plane containing this object face (recall that all viewpoints are exterior to the object itself) will possess an extended sight line intersecting at least one edge and one vertex adjacent to that face. Thus this planar region is a potential critical surface in the perspective model. The intersection of this surface with the sphere at infinity is a great circle, hence this is a potential critical curve in the orthographic model.

To determine the potential critical regions induced by EEE-events [25], consider three object edges e_1, e_2 and e_3. For a selected point *q* on one of the edges (say e_1), the only viewpoints in $\mathbf{R}^3$ that possess extended sight lines passing through *q* and points on both of the other two edges are those along the line of intersection of the two planes defined by *q* and those edges (Fig 7).

As *q* is allowed to vary among all points on e_1, these lines of intersection sweep out a quadric surface, the nature of which depends on the relative orientation of the three edges. For example, if the edges are skew to each other then the induced quadric surface is a portion of a hyperboloid of one sheet [18]. As in the case of EV-events, however, viewpoints on this surface lying between any two of the edges are discarded, as no extended sight line emanating from such viewpoints can pass through all three features simultaneously. This quadric surface constitutes a potential critical surface in the perspective model. The intersection of this surface with the sphere at infinity yields a potential critical curve in the orthographic model (Fig 8).

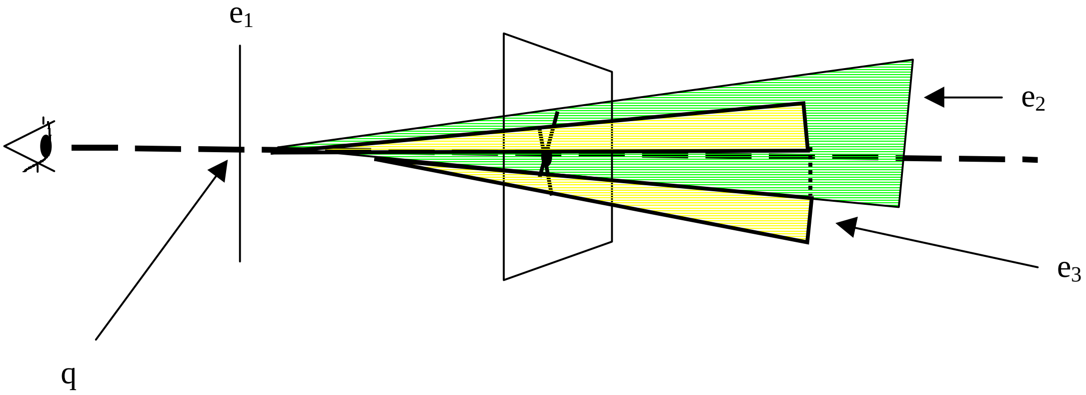

Fig 7. An extended sight line passing through *q* and points on *e_2* and *e_3*. (From [18])

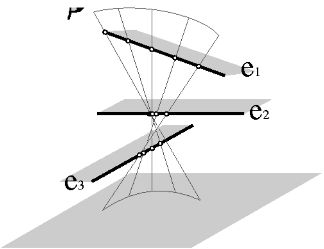

Fig 8. The potential critical region induced in R^3 by three edges *e_1*, *e_2* and *e_3* (excluding the region between the features) and (at the bottom of the diagram) its intersection with a portion of the sphere at infinity. (From [14])

We note that each potential critical region is an algebraic *surface patch* of constant maximum degree, and is bounded by a constant number of algebraic surface patches, each of which is also of constant maximum degree. We will refer to such surface patches as *well-behaved.* A property of such patches in two dimensions is that any pair intersects in at most a constant number of points. A property of such patches in three dimensions is that any pair intersects in at most a constant number of algebraic curves and any triple intersects in at most a constant number of points.

3.2 Actual Critical Regions and Algorithms to Compute the VSP

The actual critical regions of the VSP are composed of viewpoints for which there is a qualitative structural change in the LISG of the opaque view as the viewpoint moves infinitesimally in some direction. Equivalently, these regions consist of the set of accidental viewpoints for which there exists at least one critical event such that the object features associated with that event are *not* occluded by any other object features (Fig 9). Therefore the actual critical regions of the VSP are contained within potential critical regions, and the arrangement of potential critical regions is a refinement of the VSP.

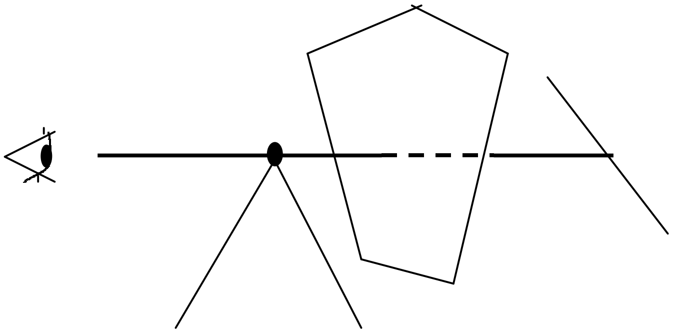

Fig 9. The occlusion of an edge that would have been involved in an EV-event by an object face. (From [17])

Yet another equivalent formulation is that actual critical regions are made up of those accidental viewpoints for which there exists an extended sight line with the following properties (Fig 10):

- The line intersects every feature associated with some critical event.
- The line is tangent to the object(s) at the features associated with the event except, possibly, at the feature that is furthest from the viewpoint (it may be tangent to the object or it may penetrate the object at that feature).
- The line does not intersect any other object feature between the viewpoint and that furthest feature.

We note here that, in three dimensions, actual critical regions are not necessarily bounded by a constant number of surface patches. Thus, while potential critical regions are well-behaved, actual critical regions may not necessarily be so. It is not true, for example, that any pair of actual critical regions intersects in at most a constant number of algebraic curves. However, because actual critical regions lie on potential critical regions, which are indeed well-behaved, these intersections consist of a collection of curve *segments* that lie on a constant number of algebraic curves. In two dimensions each actual critical region is a curve (bounded by two points) lying on a well-behaved (potential critical) curve and thus is well-behaved.

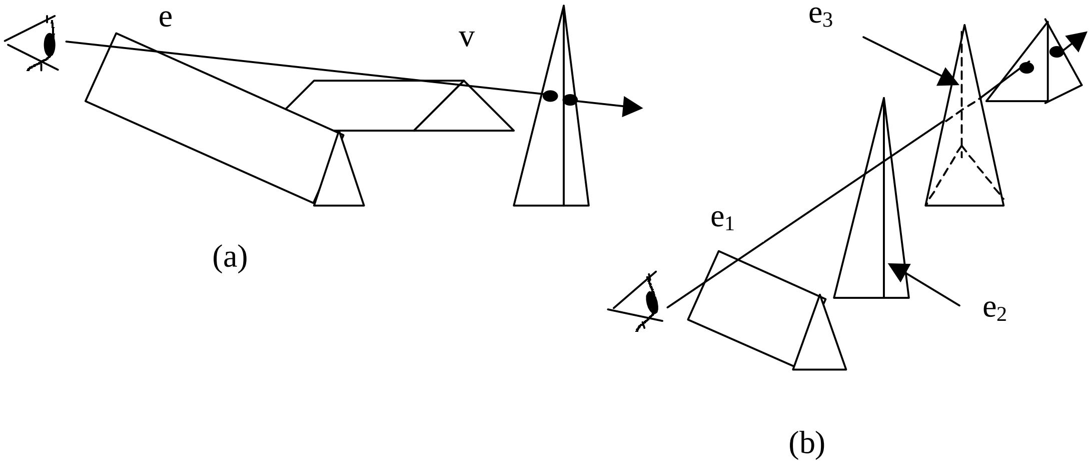

Fig 10. Accidental viewpoint belonging to actual critical regions. In each case the event shown (an EV-event in (a) and an EEE-event in (b)) is visible in the opaque view. (From [27])

We now describe the different approaches used by the three algorithms, mentioned earlier, for computing actual critical regions directly from potential critical regions. The method of the first algorithm [17] is to prune each potential critical region as it is computed (or to discover that a potential critical region need not be computed in the first place) using various ad hoc methods *prior* to computing the entire VSP. For example, it is not necessary to compute the potential critical region induced by an EV-event when the vertex *v* lies in the *shadow volume* of an edge *e*. This is the intersection of the negative half-spaces of the two faces adjacent to *e*. The *negative half-space* of a face consists of all points in $\mathbf{R}^3$ whose associated vectors have negative dot product with the outward normal of the face (in other words it is the half-space behind the face). A line segment extended from any point *p* in the shadow volume of *e* to a point *p*' on *e* will, in a region near *p*', lie in the interior of the object whose boundary contains *e*. Therefore, no extended sight line (from any viewpoint) intersecting both *v* and *e* with the three properties discussed above may exist, and there is no need to compute the potential critical region induced by *v* and *e* since there can be no corresponding actual critical region.

In the orthographic model, pruning of a potential critical curve amounts to identifying all viewpoints on that curve from which some feature involved in the associated critical event is occluded by an object feature not associated with the event. Points with this property are pruned out, and the remaining set of viewpoints constitutes the actual critical curve.

Practically, all such points can be identified by first finding all *event occlusion endpoints* (EOE points) on the potential critical curve. EOE points are those viewpoints for which the extended sight line associated with the critical event is tangent to some edge adjacent to an object face before intersecting at least one of the features involved in the event. The object face is not adjacent to any of the features defining the event. This extended sight line thus intersects *four* object edges (note that, for EV-events, we regard the intersection of the extended sight line and the vertex associated with the event as an intersection with each of the edges adjacent to that vertex), is tangent to the object at the three edges closest to the viewpoint, and may penetrate the object only at or beyond the fourth edge (Fig 11). We will have more to say about lines with these properties in section 4.4, which deals with polyhedral terrains.

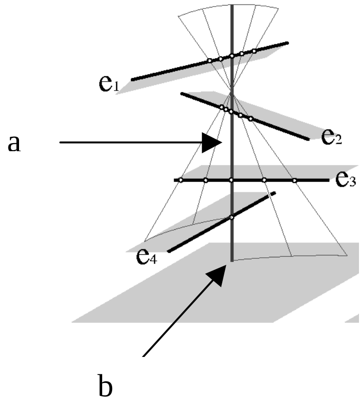

Fig 11. An extended sight line intersecting four object edges (a) and its associated EOE point on (a small portion of) the sphere at infinity (b). (From [14])

In $\mathbf{R}^3$ at most two lines can intersect four lines (or line segments) in general position in the manner described (*general position* with respect to lines, or line segments, implies that they are not all coplanar and that no three of them have a common point of intersection). Thus there can be at most two extended sight lines associated with the four edges in question. Since these two lines intersect the sphere at infinity in at most four points we note that each object edge contributes no more than four EOE points to each potential critical curve.

We also note that any EOE point on a potential critical curve p_1 occurs at the intersection of that curve with some other potential critical curve p_2. This is so because the edge adjacent to the occluding face combines with several of the features which induce p_1 to form the critical event which induces p_2.

Once the set of EOE points for the potential critical curve is determined, they are sorted along that curve. The EOE points are then classified into two groups, those for which an occlusion is *about to occur* as the curve is traversed in an arbitrary, fixed direction, and those for which an occlusion is *about to cease*. If the number of occluding faces is first calculated for the initial point on the curve at which the traversal begins, then the collection of segments between consecutive EOE points along the curve for which no occlusions occur is easy to find. This collection constitutes the actual critical region for the event. By analyzing the potential critical curves induced by all events in a similar manner, we may obtain the entire set of actual critical regions for the object.

Another approach for handling occlusion, used by the second algorithm we discuss, is to compute the entire arrangement of potential critical regions in $\mathbf{S}^2$ before any pruning is performed [18]. In the orthographic model, this involves finding the intersections of the potential critical curves in the arrangement and determining the curve segments that are bounded by these intersections. The two cells adjacent to each segment possess distinct transparent views, but their opaque views may be isomorphic. The opaque views of adjacent cells will be isomorphic when, for each critical event associated with the segment between the cells (there may be more than one if different events induce overlapping potential critical curves), at least one of the features involved in the event is occluded by a feature not associated with the event from all viewpoints along the segment. If this occurs, the segment is pruned from the partition. When all such segments are removed, what remains constitutes the entire set of actual critical regions for the object.

The approach taken by the third algorithm requires the use of geometric spaces of dimension greater than three [25]. Given a d-dimensional viewpoint space and a two-dimensional image space (corresponding to an image plane or sphere), we define the (d+2)-dimensional *aspect space* to be the Cartesian product of these two spaces. Thus, the aspect space is four-dimensional in the orthographic model and five-dimensional in the perspective model. A point in aspect space, denoted by (θ, φ, u, v) in the orthographic model ((x, y, z, u, v) in the perspective model), represents the contents of point (u, v) in the image space as seen from viewpoint (θ, φ) (respectively (x, y, z)).

A point on an object feature is said to occupy a point (θ, φ, u, v) (or (x, y, z, u, v)) in aspect space when, with respect to the viewpoint (θ, φ) (respectively (x, y, z)), its projection (in the *transparent* image) lies at the point (u, v) in the image space. The set of points in aspect space occupied by an object feature is called its *aspect space volume*. For example, a vertex is a zero-dimensional feature which can be observed (in the transparent image) from all points in the two-dimensional orthographic viewpoint space leading to an aspect space volume of two (zero plus two) dimensions. Likewise, an edge is a one-dimensional feature which can be observed from all points in the two-dimensional orthographic viewpoint space leading to an aspect space volume of three dimensions. In the perspective model aspect space volumes for vertices and edges are of three and four dimensions, respectively.

Note that aspect space volumes may be associated with EV- and EEE-events as well as object features. As we have seen, the (zero-dimensional) points of intersection in the image space associated with these events are observable from a one-dimensional curve of viewpoints in the orthographic model and from a two-dimensional surface of viewpoints in the perspective model. Thus the induced aspect space volumes are one-dimensional in the orthographic model and two-dimensional in the perspective model. We emphasize the fact that aspect space volumes associated with EV- and EEE-events share a similarity with the potential critical regions examined in section 3.1 in that they embody the collection of viewpoints from which structural changes in the corresponding transparent image may be observed**.** One could project these volumes into viewpoint space, thus generating the set of potential critical regions for the object. These regions may then be pruned, as described earlier, to remove regions from which occlusion occurs, thus yielding the entire set of actual critical regions.

An alternative approach, however, is to perform this pruning in aspect space *before* projecting into viewpoint space. The virtue of this alternative (indeed the reason for making use of the aspect space construction at all) becomes apparent when it is noted that occlusion is particularly easy to quantify in aspect space. Two object features f_1 and f_2 occlude whenever they occupy the same point in the image space (in the transparent image) with respect to a given viewpoint. Equivalently, occlusion occurs whenever the aspect space volumes for the two features have non-empty intersection (note that analytical means must be employed to determine those connected sub-regions of this intersection for which f_1 occludes f_2 and vice versa).

This suggests the following course of action. Using analytical methods (parametric equations for aspect space volumes arising from the different kinds of features and events are derived in [25]) compute the aspect space volume for each object face. Then, for a selected face f, determine the intersection of its volume with each of the other aspect space volumes just computed. Use set subtraction to remove from the aspect space volume of f those sub-regions of these intersections for which f is *occluded by* another object face (note that pruning of occluded features and events involving f is accomplished by this subtraction; we may think of the remaining volume as the aspect space volume for f corresponding to the *opaque* image). The resulting volumes for all faces may be determined similarly, and the union of these yields a final volume for the entire object. Finally, in the orthographic model, the one-dimensional curves of this volume are projected into viewpoint space, while, in the perspective model, the two-dimensional surfaces of this volume are so projected. These curves and surfaces are the ones associated with visible EV- and EEE-events only, since all occlusions have been eliminated by the set subtraction step. Thus these projections constitute the entire collection of actual critical regions that is sought.

3.3 Complexity of the VSP

This section provides an analysis of the complexity of the VSP for a single polyhedral object with $O(n)$ vertices, edges and faces. For any such object there exist $O(n^2)$ vertex-edge pairs and $O(n^3)$ edge triples. Thus the number of potential critical regions induced by EV-events is $O(n^2)$, and those induced by EEE-events is $O(n^3)$. We make use of the following fact [20]: in $\mathbf{R}^{\mathrm{d}}$, a set of $O(n)$ well-behaved $(d\text{-}1)$-dimensional surface patches induces an arrangement of complexity $O(n^d)$. Recall that the potential critical regions generated by EV- and EEE-events are all well-behaved. The entire set of $O(n^3)$ potential critical regions therefore induces an arrangement of complexity $O(n^6)$ in the orthographic model and $O(n^9)$ in the perspective model. It can be shown (we omit the details) that these also serve as bounds for the complexity of the VSP itself. (In fact, in any situation where an upper bound can be established for the number of actual critical regions, the upper bound complexity of the arrangement of an equal number of potential critical regions containing these actual critical regions serves as an upper bound on the complexity of the VSP induced by the actual critical regions. This result will be used repeatedly throughout this survey.)

In the case of a single convex polyhedron, the above bounds may be improved upon. Here there exist no EEE-events (since no three pairwise non-adjacent object edges may be met by any extended sight line). Further, all EV-events that induce actual critical regions arise due to vertex-edge pairs that lie on the same object face only (for a given edge, all vertices not lying on the same face lie in the shadow volume of the edge and thus do not generate actual critical regions – see section 3.2). Therefore, all actual critical regions lie, in the orthographic model, on one of the n great circles at infinity coplanar with the object faces or, in the perspective model, on one of the n planes containing the object faces. These actual critical regions are thus well-behaved. The result mentioned above may then be used directly to conclude that the VSP for convex objects in the orthographic model is $O(n^2)$, while in the perspective model it is $O(n^3)$.

A lower bound on the worst-case complexity of the VSP for a polyhedron of complexity $\Theta(n)$ is established in [25]. For the convex case, it is shown that there exists a convex polyhedron such that there will be $\Omega(n^2)$ vertices in the VSP of the orthographic model and $\Omega(n^3)$ vertices in the VSP of the perspective model (Fig 12). In the orthographic model, this can be seen by noting that each of the $\Theta(n)$ great circles, which are the actual critical regions induced by the $\Theta(n)$ faces on one of the 'orthogonal' bands, will intersect each of the $\Theta(n)$ great circles associated with the faces on the other 'orthogonal' band. There will thus be $\Theta(n^2)$ distinct points of intersection. In the perspective model, the $\Theta(n^3)$ triples consisting of two planes containing two faces in one band and a plane containing a face in the other band will intersect in $\Theta(n^3)$ distinct points. Because of the previous upper bound results, these bounds for the complexity of the VSP are tight in the worst case: $\Theta(n^2)$ in the orthographic model and $\Theta(n^3)$ in the perspective model.

In the non-convex case, a polyhedron of complexity $\Theta(n)$ can be constructed [25] consisting of two grids of $\Theta(n)$ by $\Theta(n)$ 'strips' (of negligible thickness), behind, and slightly skew to, two sets of $\Theta(n)$ parallel rectangular faces with a narrow gap between each pair of consecutive faces (Fig 13). Since the strips are arbitrarily thin, we may consider each of the $\Theta(n^2)$ junctions of the grids at which two edges appear to overlap to be pseudo-vertices. These pseudo-vertices along with the edges of the rectangular faces induce $\Theta(n^3)$ nearly planar actual critical regions. The grids and rectangular faces on the left are positioned so that, as the viewpoint moves along a horizontal line in viewpoint space, each of the junctions aligns with the edges of one gap before any of them are seen to align with the edges of another gap. In this way it is ensured that each of the $\Theta(n^3)$

actual critical regions are met at a distinct point along the line. The grids and faces on the right are positioned so that the same is true as the viewpoint moves along a vertical line. Thus a 'grid' of $\Theta(n^3)$ by $\Theta(n^3)$ nearly planar actual critical regions is induced in some region of viewpoint space. This therefore yields a VSP with complexity $\Omega(n^6)$. In the perspective model, a third set of grids and faces must be added, and the three sets positioned orthogonally to each other to create a region of viewpoint space with a grid containing $\Theta(n^3)$ by $\Theta(n^3)$ by $\Theta(n^3)$ nearly planar actual critical regions. Thus a VSP of complexity $\Omega(n^9)$ is constructed. Note that by extending imaginary prisms (with total complexity $O(n)$) of negligible thickness between the various strips and rectangular faces one may construct a single connected polyhedral object. With care, the addition of these prisms will not lower the worst-case complexities just presented. So assuming that the scene consists of a single, non-convex, connected polyhedron does not affect the worst-case complexity of the induced VSP (which, because of the previous upper bound results, are tight): $\Theta(n^6)$ in the orthographic model and $\Theta(n^9)$ in the perspective model.

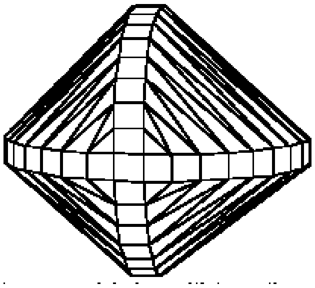

Fig 12. A convex polyhedron (with $\Theta(n)$ faces in each 'orthogonal' band) which induces a VSP with complexity $\Theta(n^2)$ in the orthographic model and $\Theta(n^3)$ in the perspective model. (From [25])

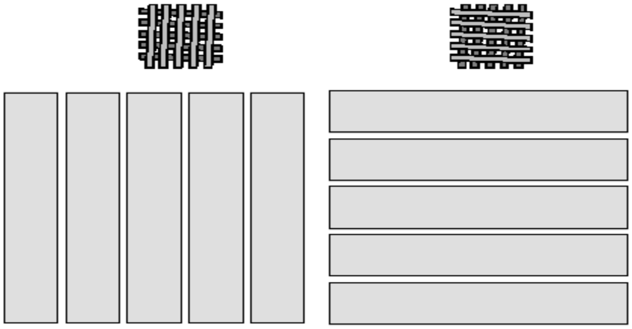

Fig 13. A non-convex polyhedron (with $\Theta(n)$ faces in total) which induces a VSP with complexity $\Theta(n^6)$ in the orthographic model. (From [25])

The lower bound construction presented above for the non-convex case requires that some object edges (those of the grids) be slightly askew to other object edges (those of the rectangular faces). That such a condition is unnecessary to obtain these bounds, at least in the orthographic case, is demonstrated in [29]. Here a set of $4n$ axis-parallel lines is presented for which there are $\Omega(n^6)$ distinct vertices in the VSP, each formed by the intersection of a pair of actual critical regions associated with the EEE-events induced by these lines. Although the proof is for a collection of lines, each such line may be replaced by a long axis-parallel prism of negligible thickness (so that for each EEE-event induced by the lines there is a corresponding EEE-event induced by the prisms), and all such prisms connected, as above, by additional (axis-parallel) prisms of negligible thickness. As before, these additional prisms will not lower the worst-case complexity of the VSP. The resulting connected, non-convex, polyhedron (of complexity $\Theta(n)$) all of whose faces are bounded by axis-parallel rectangles, thus induces a VSP of complexity $\Theta(n^6)$.

This complexity is obtained by deriving equations for the actual critical curves on a *plane* (rather than the sphere) at infinity induced by the EEE-events arising from the $4n$ axis-parallel lines (these critical curves are shown to be hyperbolae in all cases), selecting $\Theta(n^6)$ pairs of these curves such that each pair intersects in exactly one point on the plane, and demonstrating that these points of intersection are all distinct. This establishes the existence of $\Omega(n^6)$ distinct points of intersection of critical curves on the plane at infinity. Thus, because of the argument in the previous paragraph, the assertion regarding the complexity of the VSP in the orthographic case is immediate.

A recent construction [4] demonstrating a lower bound on the worst-case complexity of the VSP induced by a scene consisting of $\Theta(k)$ convex polyhedra with a total of $\Theta(n)$ vertices, edges and faces (section 3.6) can be modified to obtain a lower bound of $\Omega(n^9)$ on the worst-case complexity for the VSP induced by axis-parallel lines in the perspective model. We omit the details.

3.4 Comparative Run-Time Complexities of the Algorithms

We now present a comparison of the worst-case run-time complexities of the three algorithms discussed in section 3.2 for computing the VSP of a general polyhedral scene in the orthographic model. Subsequent to this we discuss the complexity of the aspect space algorithm in the perspective model. In the orthographic model, arrangements are represented using a standard incidence graph data structure and are constructed using a standard two-dimensional sweep algorithm [20]. This algorithm constructs arrangements of m curves having a total of k points of intersection in time $O((m + k) \log m)$. In our examples the curves are well behaved, thus $k = O(m^2)$ and the running time is $O(m^2 \log m)$. In the perspective model arrangements can be represented by a data structure known as a *vertical decomposition* and can be constructed using a three-dimensional sweep algorithm [20].

We consider first the algorithm in which the arrangement of potential critical curves is computed before any pruning is performed [18]. Each of these (well-behaved) O(n^3) curves can be computed in constant time. The arrangement induced by these curves is thus constructed in O(n^6 log n) time using the two-dimensional sweep algorithm referred to above. Since each such curve may be intersected by the other O(n^3) curves in a total of O(n^3) distinct points, and since there exists O(n^2) curves induced by EV-events and O(n^3) curves induced by EEE-events, the total number of EV-event curve segments thus induced is bounded by O(n^2n^3) = O(n^5), while the total number of EEE-event curve segments is bounded by O(n^3n^3) = O(n^6). For each of these segments, a determination is made as to whether there exists an occluding face for some feature involved in each event associated with that segment (if so, the segment is not an actual critical curve segment and, hence, is removed from consideration). Since this determination can be made so as to not affect the upper bound on the complexity of the entire algorithm (see the next paragraph), this algorithm therefore determines the VSP in O(n^6 log n) time.

As noted in [25] determining the existence of an occluding face for a feature would ordinarily require the inspection of each object face in turn, an O(n) time procedure that would cause the entire algorithm to execute in O(n^7) time. However, this difficulty is avoided by the maintenance of a structure (called the *augmented view of the transparent object*) consisting of a list of faces for each object edge *e* which are *visually adjacent* to *e* in the transparent image, and are also either *physically adjacent* to *e* in the object or *partially occluded* by *e* with respect to all viewpoints on the segment (there are O(1) such faces for each edge). Note that this list must be updated for every edge as each of the O(n^6) segments is considered in turn, however this step can be accomplished in tandem with the calculation of the views of the scene (section 3.5) and so as not to affect the upper bound on the complexity of the entire algorithm (we omit the details). For each feature involved in each event associated with the segment under consideration it can therefore be determined whether there exists a visually adjacent face which is *not* one of the faces in this list (note that the maintenance of the augmented view allows this to be accomplished in O(1) time for each feature, as opposed to O(n) time). If such a face exists, it is also an occluding face for that feature relative to all viewpoints on the segment. As stated above, if an occluding face can be found for any feature involved in each event associated with the segment, the segment must be pruned from the arrangement.

The algorithm of [18] does not yield an *output-sensitive* run time (one that is a function of the size of its output) because it requires the computation of the entire arrangement of *potential* critical curves (of which there will always be Θ(n^3) in the general case). This dominates the run-time complexity of all other steps in the algorithm. Output-sensitive approaches, where the run-time complexity of the algorithm depends directly on the complexity of the VSP itself, are achieved by first removing all potential critical regions which are not actual critical regions, and *then* computing the arrangement thus induced. The other two algorithms discussed in section 3.2 provide examples of this approach.

In the algorithm which computes event occlusion endpoints (EOE points), in which pruning precedes the computation of the arrangement [17], it has been noted that the O(n)

edges of the object may contribute at most O(n) such points to each of the O(n^3) potential critical curves (each computed in constant time), thus yielding O(n^4) EOE points in total (each such point is also computable in constant time). The sorting of these EOE points along each critical curve (which is required in order to determine those curve segments for which occlusion of the associated event occurs) and hence the total time to find the set of actual critical curves is therefore accomplished in O($n^4 \log n$) time. The resulting arrangement of m (well-behaved) curves (with O(m^2) vertices), then, again appealing to the two-dimensional sweep algorithm, may be constructed in O($m^2 \log m$) time. The entire algorithm therefore finds the VSP in O($n^4 \log n + m^2 \log m$) time, and is output-sensitive. As noted in [17], the O(n^4) EOE points induce O(n^4) actual critical curve segments, but these segments lie on O(n^3) well-behaved curves. Thus the VSP has at most O(n^6) vertices and the worst-case running time for this algorithm is again O($n^6 \log n$).

Plantinga and Dyer [25] claim that, with appropriate generalizations, all set-theoretic operations involved in computing the aspect space volume for an object can be accomplished in O(n^5) time in both the four and five-dimensional cases. The orthographic model version of the algorithm described in section 3.2 performs these operations and then projects the one-dimensional aspect space volumes associated with critical events into viewpoint space, yielding an arrangement of m (well-behaved) curve segments which may be constructed in O($m^2 \log m$) time. Thus the entire algorithm computes the VSP in O($n^5 + m^2 \log m$) time. Note that this result is also output-sensitive. As in the analysis of the previous algorithm, the m segments lie on O(n^3) well-behaved curves in the orthographic model, thus making the worst-case run-time complexity of this algorithm O($n^6 \log n$). Note that while the worst-case performances of the two output-sensitive orthographic model algorithms of [17] and [25] are identical, the algorithm of [17] is more efficient when $m^2 \log m < n^5$.

Finally, the perspective model version of this algorithm projects the two-dimensional aspect space volumes associated with critical events into viewpoint space. We assume that the resulting set of actual critical surface patches may be subdivided into a total of m well-behaved surface patches. Then, assuming a vertical decomposition representation of the resulting (three-dimensional) arrangement of these m well-behaved surfaces, and assuming use of the three-dimensional sweep algorithm referred to at the beginning of this section, this algorithm can compute the VSP in time O($n^5 + m\lambda_q(m) \log m + V\log m$) where V is the complexity of this vertical decomposition and $\lambda_q(m)$ is the maximum length of a particular class of Davenport-Schinzel sequences (section 4.1) (q is a constant depending on the nature of the well-behaved surfaces and their boundaries) [20]. The complexity of a vertical decomposition in $\mathbf{R}^3$ is nearly the same as the complexity of the arrangement it represents. Also $\lambda_q(m)$ is nearly linear in m. In the worst case the arrangement consists of $\Theta(n^3)$ surfaces implying V is nearly O(n^9), thus leading to a worst-case run-time complexity of nearly O($n^9 \log n$).

3.5 Stable Views and the Complexity of the Aspect Graph

At this point recall that the aspect graph, unlike the VSP, stores the entire set of stable opaque views of the scene. An analysis of the complexities of computing, storing, and retrieving these views for a non-convex object in the orthographic model is provided next. Analogous considerations apply for the perspective model, and for convex objects in either model, and are thus omitted here.

Since, in the orthographic model, the size of the VSP is $O(n^6)$, there are $O(n^6)$ opaque views to be computed. Each view is the projection of $O(n)$ object features onto the image plane and itself has worst-case complexity $\Theta(n^2)$ (since each T-junction arises due to the alignment along some extended sight line of object edge pairs – see section 2). A naïve approach would be to compute (and store) each view independently of all other views. However, this method is relatively costly and can be improved upon. A procedure to calculate all views without increasing the complexity of the algorithm to compute the VSP is provided in [18]. Recall that this algorithm, described in section 3.2, computed the arrangement induced by the set of potential critical curves before performing any pruning of these curves. This algorithm may be augmented to do pruning and calculate opaque views in tandem as follows. An initial opaque view is first calculated for some region in this arrangement of potential critical curves. Then each segment of this arrangement is analyzed (for purposes of pruning) in an order such that the opaque view of one region bordering the segment has already been computed while the opaque view of the other bordering region is yet to be determined (equivalently, one may think of traversing the dual graph with nodes for each full-dimensional cell in the arrangement and arcs connecting each adjacent cell – similar in concept to the aspect graph itself but instead dual to the arrangement of *potential* critical regions, along a *spanning path* which visits each node). This latter view is calculated based on the known view and the set of events that induced the segment being analyzed. A detailed catalog is provided in [18] explaining how to update the known view to produce the new view for every class of event that can occur (for example, there are three different classes of EEE-events depending on the geometry of the situation (Fig 14)). This calculation does not increase the complexity of the overall algorithm because it can be shown [18] that for EV-events the time to compute the new view is $O(n)$ (the catalog in [18] shows that at an EV-event it is possible for $O(n)$ edges to be adjacent to the vertex involved in the event (Fig 15) and that each of these can be updated in $O(1)$ time in the view), while for EEE-events the time to compute the new view is $O(1)$ (the catalog shows that at an EEE-event only $O(1)$ edges and T-junctions must be updated, and that each of these can be done in $O(1)$ time). Thus the total time spent updating views at all segments is $O(n^5 n) = O(n^6)$ for EV-events and $O(n^6)$ for EEE-events. The complexity of the entire algorithm for computing the VSP and all opaque views therefore remains $O(n^6 \log n)$.

A naïve approach to storing the set of views would be to maintain them all independently in the aspect graph, which would then possess a space complexity of $O(n^6 n^2) = O(n^8)$. When most views differ from their adjacent views in only a small number of ways (for example those views separated in the VSP by boundary segments associated with a small number of EEE-events only), the amount of redundant information stored would be large.

A data structure is proposed in [17] to eliminate such redundancies. However, it allows only the retrieval of the set of individual features existing in any particular view rather than the actual reproduction of that entire view. Here, the views are computed along a spanning path as in [18]. In addition, each visit to a segment is considered to be a step in time. Features existing in the new view that did not exist in the old view are thought of as having been created at that time and features existing in the old view that will not exist in the new view are thought of as having been deleted at that time. Thus, with every feature existing in any view there is associated a time interval, $[t_i, t_j]$, denoting that the feature was created in the view computed at time t_i, existed at all views computed from times t_i to t_{j-1}, and then was finally deleted at time t_j. Also, with each view is associated the time of its calculation. Thus to retrieve the view that was calculated at time t, it is only necessary to determine all features whose existence intervals contain t. A data structure to handle such queries efficiently is a priority search tree. A collection of k intervals may be stored in such a tree with size $O(k)$, and the tree may be built, assuming the intervals are already sorted by their left endpoints (as these time intervals are), in time $O(k)$. Since, as we have seen, the number of features created and deleted in total is $O(n^6)$ (at most $O(n)$ features are created or deleted at each of the $O(n^5)$ segments associated with EV-events and at most $O(1)$ features are created or deleted at each of the $O(n^6)$ segments associated with EEE-events), k is $O(n^6)$. Thus, the size of the tree is $O(n^6)$. Using this method, therefore, allows the complexity of the aspect graph to be maintained at $O(n^6)$, the same as that of its dual, the VSP.

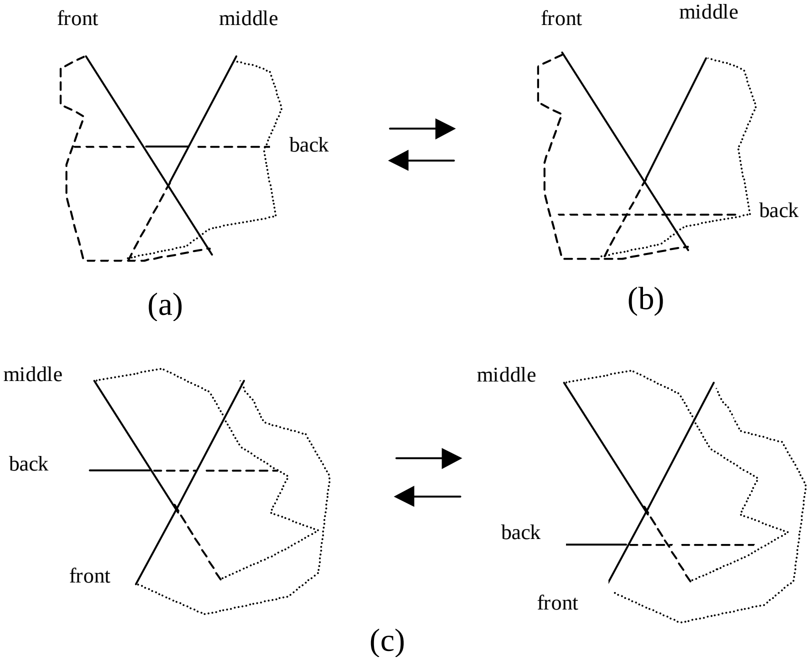

Fig 14. Three classes of EEE-events: (a to b) two T-junctions disappear, (b to a) two T-junctions appear and (c) a T-junction moves from one edge to another. (From [18])

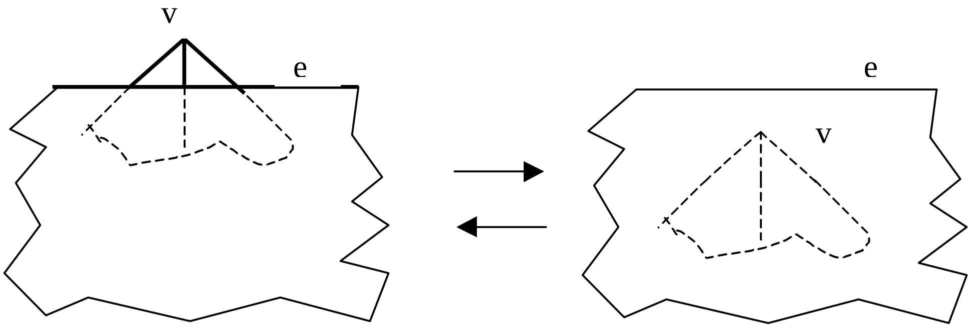

Fig 15. An EV-event requiring the updating of O(*n*) edges (adjacent to the vertex) in the view. (From [18])

One pays a bit of a penalty in retrieval time in exchange for the space savings allowed by the priority search tree. Whereas the naïve approach permits us to retrieve a view of size s in time $O(s)$ since the entire view is stored, the priority search tree has a retrieval time of $O(s + \log k) = O(s + \log n)$. Note however, that the worst-case running times of both approaches are identical when $s \geq \log n$.

3.6 Convex Polyhedra

We consider the special case of a scene consisting of a collection of k disjoint *convex* polyhedra with a total of n edges. Bounds on the complexity of the VSP of such a scene in both the orthographic and perspective models may be found in [7] and are, in fact, lower than those for a general polyhedral scene with n total edges. The upper bound analysis considered here proceeds by determining the number of distinct triples of edges (associated with EEE-events) that can induce actual critical regions. The number of such triples (naively $O(n^3)$) dominates the number of vertex-edge pairs (associated with EV-events) that induce actual critical regions and hence is an upper bound on the total number of actual critical regions of the VSP. Recall (section 3.2) that every such edge triple is associated with extended sight lines that intersect each edge, are tangent to the objects containing the edges (with the possible exception of the object containing the edge furthest from the viewpoint), and do not intersect any other object features anywhere between the viewpoint and the furthest edge. Actually, the bound that is sought is on the number of such edge triples with the additional property that each edge belongs to a distinct polyhedron for, asymptotically, this will dominate the number of edge triples arising from non-distinct polyhedra.

Towards this end, consider three distinct polyhedra P_i, P_j and P_l and an edge e_l of P_l. We parameterize the points on e_l by t, $0 \leq t \leq 1$. Suppose there exists a plane R such that, for all t, the rays emanating from $e_l(t)$ and tangent to P_i meet P_i before they intersect the plane, and suppose further that the same property holds for R with respect to P_j. (The plane R, if it exists, is a translate of some plane through e_l such that P_i and P_j both lie on the same side of it. If no plane with this property can be found then choose a plane through e_l slicing both P_i and P_j and perform the analysis that follows first on the portions of P_i and P_j in one of the half-spaces induced by it, and then on the portions of P_i and P_j in the other. In this case R will be a translate of this plane.) The intersection of the plane R with the set of rays emanating from $e_l(t)$ and tangent to P_i (the *silhouette* of P_i from $e_l(t)$), which we denote $Q_i(t)$, is a convex polygon. The polygon $Q_j(t)$ is defined similarly. We may consider the edges and vertices of $Q_i(t)$ and $Q_j(t)$ to be the projections, onto R, of corresponding edges and vertices on P_i and P_j, respectively (Fig 16). Polygon $Q_i(t)$ ($Q_j(t)$) will be *unbounded* if any face of P_i (respectively P_j) is coplanar with the plane through e_l parallel to R, or if P_i (respectively P_j) had to be sliced.

As t is allowed to vary the shapes and positions of $Q_i(t)$ and $Q_j(t)$ will be altered, but they will remain convex polygons. When a ray emanating from $e_l(t)$ is tangent to *both* P_i and P_j, say at edges e_i and e_j, respectively, then an intersection of the edges of $Q_i(t)$ and $Q_j(t)$ that are the projections of e_i and e_j occurs (and we say that the edge triple (e_i, e_j, e_l) gives rise to this intersection). Each such ray (now considered as emanating from a point at infinity) is thus tangent to P_i at e_i, is tangent to P_j at e_j, and intersects edge e_l (playing the role of the furthest of the three edges from the point at infinity). In the orthographic model, each of these rays may be lengthened to an extended sight line such that the resulting collection of extended sight lines includes all those with the properties delineated in section 3.2 (with respect to edges e_i, e_j and e_l). In the perspective model, this collection of rays constitutes a set of *maximal* extended sight lines, such that each extended sight line with the properties of section 3.2 (with respect to edges e_i, e_j and e_l) is contained within one of the maximal lines in this set. If the foregoing analysis is repeated for all edges e_l on P_l and the entire analysis repeated for all P_i, P_j and P_l in the scene, we see that, in either model, every extended sight line associated with an EEE-event can be associated with a ray possessing the properties mentioned above.

A bound on the number of intersections of distinct polygon edge pairs generated in the manner described (we emphasize that we are *not* interested in the total number of intersections among polygon pairs, but only the number of distinct edge pairs that have intersection) determines a bound on the number of distinct edge triples giving rise to these intersections. A consequence of the above analysis is that this, in turn, provides a bound on the number of such triples that induce all extended sight lines associated with EEE-events. This, finally, yields a bound on the number of actual critical regions in the VSP, from which a bound on its complexity can be directly obtained.

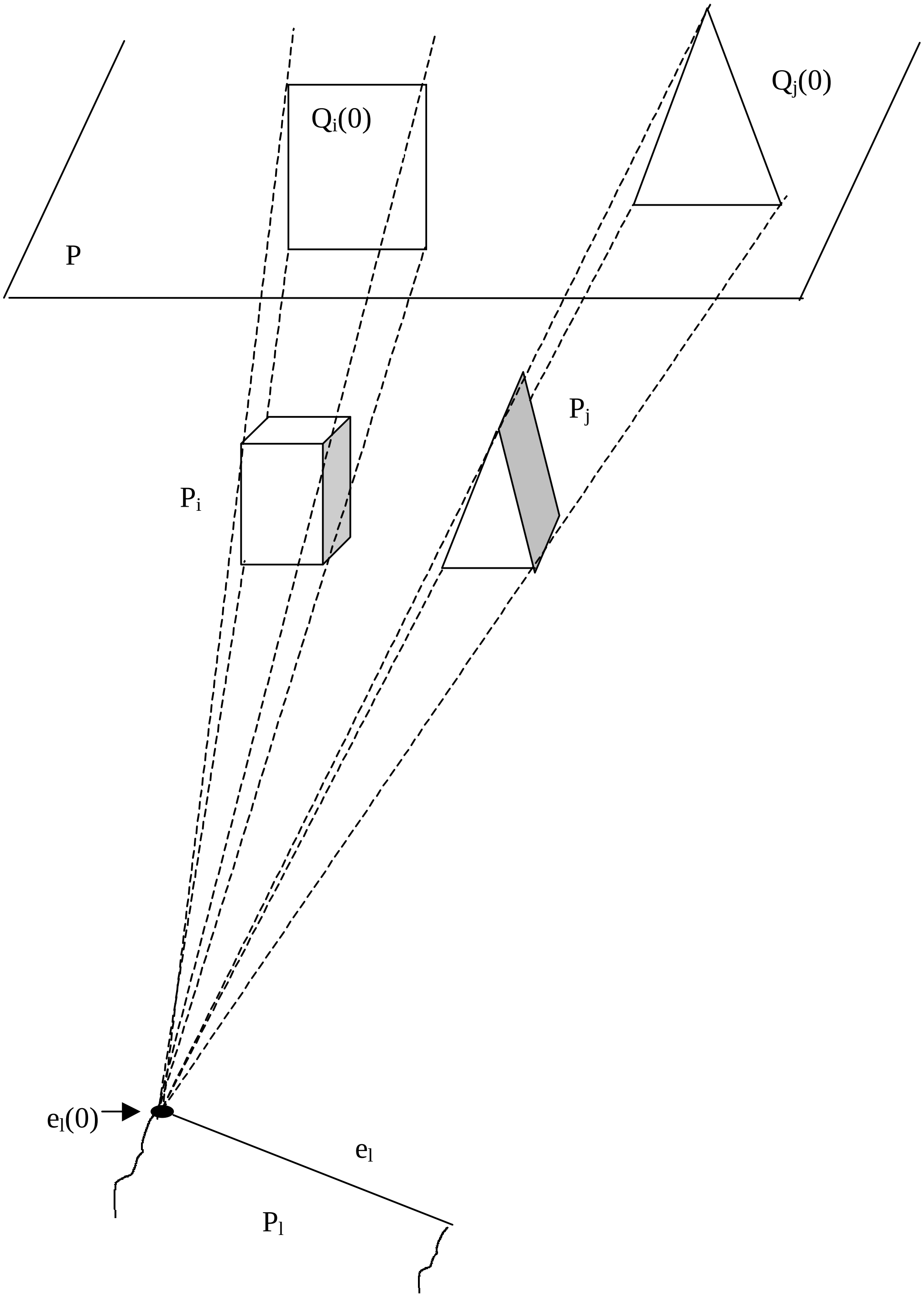

Fig 16. The silhouettes $Q_i(t)$ and $Q_j(t)$ (cast by P_i and P_j, respectively) at $t = 0$.

To find a bound on the number of intersections of distinct polygon edge pairs, we proceed as follows. Let $|P|$ denote the complexity (the total number of vertices, edges and faces) of polyhedron P. At $t = 0$, polygons $Q_i(0)$ and $Q_j(0)$, have (since their edges are projections of edges of P_i and P_j, respectively) $O(|P_i|)$ and $O(|P_j|)$ edges, respectively. Since each edge of a convex polygon will intersect a second convex polygon at most twice, $Q_i(0)$ and $Q_j(0)$ will intersect in a number of points equal to at most twice the number of edges of the polygon with the fewest edges. Thus they will intersect in $O(|P_i| + |P_j|)$ points.

As t increases, intersections among new pairs of edges arise in two ways only. Either an edge of one polygon will start to intersect a new edge in the other polygon as the shapes and positions of the polygons are altered, or else edges in one of the polygons are replaced by other edges (so that intersections occurring on the old edges are replaced by those occurring on the new).

In the first case an edge, say on $Q_i(t)$, will start to intersect a new edge, on $Q_j(t)$, just when one of its adjacent vertices intersects the new edge. This occurs when a ray, emanating from $e_l(t)$, is tangent to P_i at the vertex which projects to the vertex in $Q_i(t)$ and is simultaneously tangent to P_j at the edge which projects to the new edge in $Q_j(t)$. Now, consider a fixed vertex v in, say, P_i. As t varies, the collection of rays from $e_l(t)$ meeting v sweeps out a planar region within the affine hull of e_l and v. At most two of these rays will, in addition, be tangent to an edge of P_j (Fig 17). Thus the vertex corresponding to v in $Q_i(t)$ will intersect a new edge of $Q_j(t)$ for, at most, two distinct values of t. Any vertex of P_i or P_j is thus associated with at most two intersections of new edge pairs. The number of intersections of new edge pairs, arising from the intersections of a vertex of one polygon with a new edge of the other polygon is therefore $O(|P_i| + |P_j|)$.

The second case occurs at those values of t for which $e_l(t)$ becomes coplanar with a facet of either P_i or P_j. Assume, for the sake of argument, that this facet lies in P_i. Just before the co-planarity event occurs, the rays emanating from $e_l(t)$ and tangent to P_i at this facet will intersect some of the edges adjacent to this facet, and these edges will be projected into $Q_i(t)$ as a nearly straight polygonal chain. When the co-planarity event occurs, this chain will straighten and will be replaced by a different polygonal chain comprised of the projections of the remaining edges adjacent to the facet. Just after the co-planarity event, the new chain will cease to be straight. Since a line segment can intersect a convex polygon in at most two points, it is clear that, at the moment of co-planarity, at most two intersections with $Q_j(t)$ involving the edges of $Q_i(t)$ from the old chain will be replaced by at most two intersections involving edges from the new chain. Thus the chosen facet is associated with at most two intersections of new edge pairs, and this will be true for any facet in P_i or P_j. We therefore see that the number of intersections of new edge pairs arising from the replacement of one set of edges by a second set of edges in one of the polygons is $O(|P_i| + |P_j|)$.

We complete the analysis by noting that the initial $O(|P_i| + |P_j|)$ intersections (at $t = 0$) is therefore augmented by at most $O(|P_i| + |P_j|)$ additional intersections as t varies throughout its range. Thus, for fixed polyhedra P_i, P_j and P_l, and a fixed edge e_l on P_l, the number of edge triples (e_i, e_j, e_l) giving rise to intersections of $Q_i(t)$ and $Q_j(t)$ is bounded by $O(|P_i| + |P_j|)$. Repeating this analysis for all edges e_l on P_l yields a bound of $(|P_l| \cdot O(|P_i| + |P_j|))$. Then, summing these results over all polyhedra P_i, P_j and P_l yields

$$\Sigma_{i \neq j \neq l} (|P_l| \cdot O(|P_i| + |P_j|)) = \Sigma_{i \neq j \neq l} (|P_l| \cdot O(|P_j|)) = \Sigma_i \Sigma_j \Sigma_l O(|P_l| \cdot |P_j|) = O(n^2 k)$$

triples in total. Since this, as mentioned earlier, is a bound on the number of actual critical regions (which represents an improvement over the naïve bound because $k < n$) it has been shown that the complexity of the VSP in the orthographic model is $O((n^2k)^2) = O(n^4k^2)$, while in the perspective model it is $O((n^2k)^3) = O(n^6k^3)$.

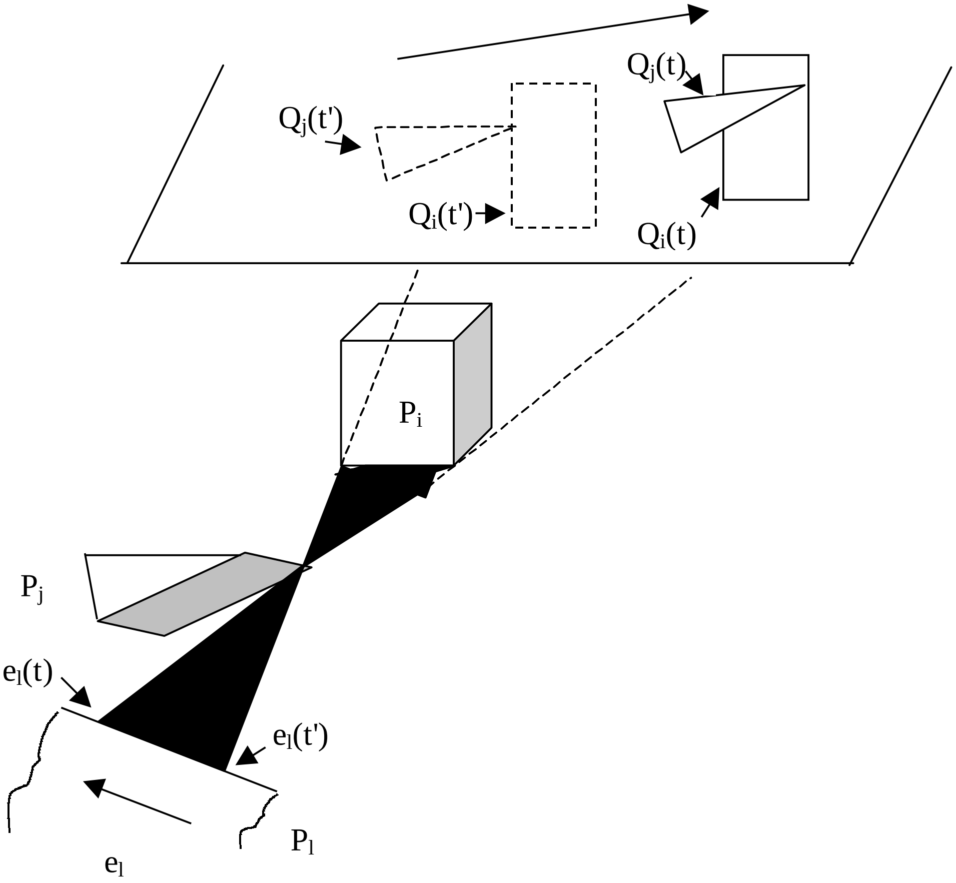

Fig 17. The planar region swept out by rays from *e_l* through *v*, as *t* varies.

Recently [4], a lower bound construction has been discovered which establishes that the above bounds are in fact tight in the worst case; $\Theta(n^4k^2)$ in the orthographic model and $\Theta(n^6k^3)$ in the perspective model. This construction consists of $\Theta(k)$ long, narrow, convex polyhedra ('needles') each with constant complexity lying above a single, nearly flat, convex polyhedron ('fan') with $\Theta(n)$ coplanar edges. The fan, in turn lies above a nearly flat convex polyhedron ('drum') with $\Theta(n)$ horizontal, parallel edges (Fig 18). The polyhedra are positioned so that their edges are all arbitrarily close to the hyperbolic paraboloid $y = xz$. Thus the needle edges lie arbitrarily close to the line $y = x$ in the horizontal plane $z = 1$. Likewise, the drum edges lie arbitrarily close to the line $y = -x$ in the horizontal plane $z = -1$. Finally, the fan edges lie arbitrarily close to the parabola $y = x^2$ in the plane $x = z$. Moreover, the vertices of the fan are positioned so that each fan edge crosses this parabola at exactly two points.

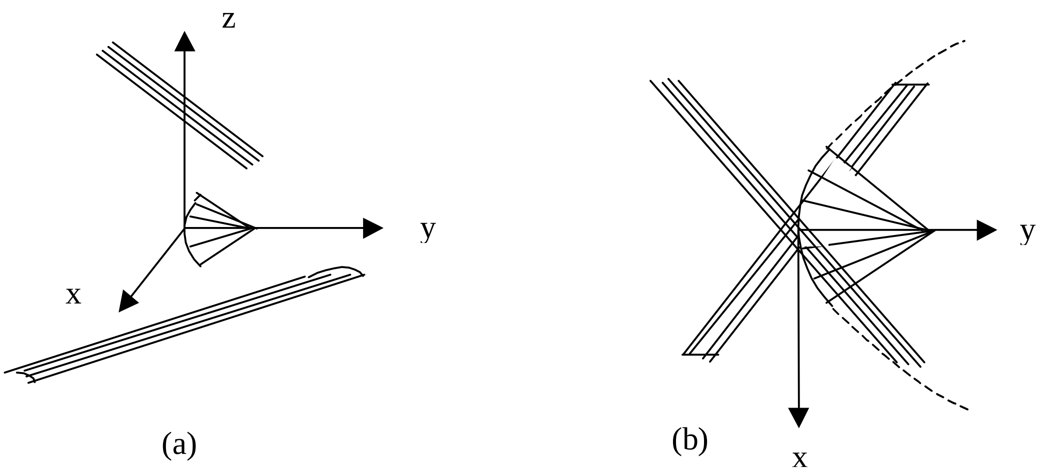

Fig 18. (a) A portion of the lower bound construction for $\Theta(k)$ convex polyhedra with $\Theta(n)$ total edges. (b) The same construction as viewed from the plane $z = +\infty$ near the z-axis. (From [4])

The configuration just described implies that the $\Omega(n^2k)$ critical surfaces associated with EEE-events induced by the collection of triples consisting of a needle edge, fan edge and drum edge all belong to a family of hyperbolic paraboloids which can be created by small perturbations of the original hyperbolic paraboloid $y = xz$. Also, it can be shown that none of these EEE-events are occluded in a small rectangular area of viewpoints near the positive z-axis on the sphere at infinity (orthographic model) or sufficiently far away from the scene (perspective model). Finally, it can be shown that these paraboloids do not intersect each other in this region and each extends from the left side to the right side of the rectangle (so that the rectangle is partitioned into $\Omega(n^2k)$ slices).

Therefore, by repeating the above construction and rotating it ninety degrees to the original, an $\Omega(n^2k)$ by $\Omega(n^2k)$ grid of critical regions is created in this rectangular region in the sphere at infinity, yielding a VSP of complexity $\Omega(n^4k^2)$ in the orthographic model. Furthermore, it is possible to place three instances of the above construction oriented orthogonally to each other and sufficiently far away from each other so as to create a $\Omega(n^2k)$ by $\Omega(n^2k)$ by $\Omega(n^2k)$ grid of critical regions in a small rectangular volume of $\mathbf{R}^3$ far from any of the constructions, yielding a VSP of complexity $\Omega(n^6k^3)$ in the perspective model.

3.7 Flight Paths

We now present several output-sensitive algorithms for computing the potential critical regions of a collection of polyhedral objects (with a total of n edges) when the viewpoint space consists of a straight-line flight path in $\mathbf{R}^3$ [9]. (The techniques of pruning, described in section 3.2, and view computation, in section 3.5, may equally well be applied here once the potential critical regions have been found, to determine the VSP and aspect graph, respectively.) Note that here the potential critical regions are those points where the potential critical surfaces of the perspective model (described in section 3.1) intersect the flight path. Since there are O(n^3) such surfaces, and since each will intersect a line in at most a constant number of points, there will be O(n^3) critical regions. In these algorithms the flight path f is parameterized by time t, $0 \leq t \leq 1$, so that a point $f(t_0)$ on f is the current viewpoint at time $t = t_0$.

The first algorithm proceeds by selecting an edge e of one of the polyhedral objects and sweeping a triangle with base e and apex $f(t)$ on f from $f(0)$ to $f(1)$. It discovers all values of t for which two other edges e' and e'' pierce the triangle such that these two pierce points are collinear with $f(t)$. When this event occurs e, e' and e'' are aligned along some line of sight from $f(t)$. This represents a possible EV-event when two of these edges are adjacent and a possible EEE-event when all three edges are non-adjacent (note that these events may in fact be occluded from the viewpoint $f(t)$, and so are associated with potential, rather than actual, critical regions). As the sweep proceeds, this algorithm maintains a priority queue to keep track of these collinearity events plus two other kinds of events; the meeting of an edge endpoint by the sweep triangle, or the departure of an edge from the sweep triangle through one of the sides of the triangle. There are O(n) events of the latter two types. Let k_e be the number of collinearity events. We know that k_e = O(n^2) because for each of the O(n^2) pairs of edges e' and e'' there are in $\mathbf{R}^3$ at most two lines intersecting the four segments e', e'', e and f (we emphasize, however, that only collinearities that actually occur are found via this method – a brute force method, running in Θ(n^2) time, comparing all edge pairs would necessarily be less efficient than this method, other than in the worst case). There are thus O(n^2) events in total for each edge e and all priority queue insertions and deletions may be handled in time O(log n) (each event itself can be processed by analytical means in O(1) time, we omit the details). This implies that the sweep for each edge e can be accomplished in O(($n + k_e$) log n) time. Repeating this procedure for each of the n edges yields an algorithm with running time O(($n^2 + k$) log n) where $k = \Sigma\, k_e$ is the total number of potential critical regions. The complexity of this algorithm is O(n^3 log n) in the worst case.

The second algorithm presented in [9] is also a sweep algorithm. It makes use of a *duality transform* [15] which, in this case, maps points (lines) in a *primal plane* to lines (respectively, points) in a *dual plane*. This mapping is one-to-one and is such that the *property of incidence* is maintained; that is, if a point p is incident to a line l in the primal plane, then, under the mapping, the dual point l* is incident to the dual line p*. An immediate consequence of this is that any three collinear points in the primal plane will map to three lines that intersect at a single point in the dual plane.

In this algorithm, we imagine a half-plane bounded by f and consider the set of *pierce points* at which it is intersected by object edges. As this half-plane proceeds to rotate around f pierce points will appear or disappear and will move about in the plane. Now we consider the duality transform which maps these pierce points in this primal half-plane to a set of lines in the dual plane. As the sweep progresses, the dual lines will appear, disappear and move about in accordance with the movement of their corresponding pierce points in the primal plane.

As the sweep proceeds, this algorithm maintains a priority queue to keep track of the appearance and disappearance of lines in the dual plane (there are O(n) such events) and the intersection of three dual lines at a single point (there are O(n^3) such events). By the property of incidence this latter type of event corresponds to the collinearity of three pierce points in the primal plane. Since f is a line *segment*, and not a full line, one in fact needs to consider only those collinearities corresponding to lines that pass through a point $f(t)$, for some $0 \leq t \leq 1$, in the primal plane. Such collinearities represent the alignment of three edges along some line of sight from $f(t)$. They occur, assuming f coincides with a portion of the positive z-axis, inside a horizontal strip in the dual plane. Thus this algorithm will also need to keep track of O(n^2) additional events consisting of intersections of dual line pairs at points along the boundary of this strip. Note that the total number of events processed during the course of this algorithm is O(n^3).

Now, at $t = 0$, the dual plane contains an arrangement of n lines, with complexity O(n^2). Each triangular cell in this arrangement is checked (in constant time) to determine if its three bordering edges will collapse to a single point at some future time. These O(n^2) possible future events can be inserted onto an initial priority queue in O(n^2) time.

The algorithm also maintains a standard incidence graph data structure for representing arrangements [15] to keep track of the dual line arrangement as the sweep proceeds. This graph must be updated when each event occurs. For each appearance or disappearance of a dual line, the portion of this graph which must be updated represents the *zone* of that line in the arrangement, a substructure of complexity O(n) [15]. The time spent in updating the graph for each appearance or disappearance event is thus O(n). Each of the O(n) newly created triangular cells of this zone needs be checked to see if it will collapse to a single point at some future time. Since each appearance or disappearance event yields O(n) possible future events, the time spent in updating the priority queue for each appearance or disappearance event is O(n log n). Therefore the total time spent in processing the O(n) appearance and disappearance events is O(n^2 + n^2 log n)) = O(n^2 log n).

For each collinearity event only a localized portion of the graph (of constant complexity) must be updated. In addition only a constant number of triangular cells are created or destroyed, and each new one needs to be checked for possible future collapse. Since each collinearity event yields O(1) possible future events, the time spent in updating the priority queue for each collinearity event is O(log n). Thus if there are k collinearity events encountered during the sweep then the total time spent in processing them is O(k log n).

Similarly, each event involving the strip boundary in the dual plane may be handled in time O(log n). Therefore the total time spent in processing these O(n^2) events is O(n^2 log n).

Thus, the entire algorithm possesses a run-time complexity of O((n^2 + k) log n). Since k = O(n^3), it runs in time O(n^3 log n) in the worst case.

The third algorithm of [9] makes use of a transform called a *skewed projection*, defined as follows. As usual, the flight path f is parameterized by t ($0 \leq t \leq 1$). An edge e (skew to f) is chosen and parameterized by u ($0 \leq u \leq 1$). Note that the collection of line segments with one end on e and the other end on f defines a tetrahedron T. For a point p in the interior of T there exists a unique segment in this collection that passes through p. The skewed projection $sp_e(p)$ maps p to the ordered pair (u_i, t_i) where $e(u_i)$ and $f(t_i)$ are the endpoints of this segment, lying on e and f, respectively. For a line segment e' lying wholly in the interior of T it can be shown that $sp_e(e')$ is either a line segment or a connected portion of a hyperbola in the space [0,1] x [0,1]. For a polygon P lying wholly in the interior of T it can be shown that $sp_e(P)$ is a *curved polygon* (a bounded, connected region in [0,1] x [0,1] with sides that are straight lines or portions of hyperbolae which are the skewed projections of the edges of P).

The algorithm proceeds as follows. Fix edge e and compute the skewed projection $sp_e(P)$ of each polygonal face P in the collection of polyhedra. This yields a collection of curved polygons in [0,1] x [0,1]. Each intersection of boundary curves of two polygons, say at point (u_0, t_0), represents a collinearity of the point $f(t_0)$ on f, the point $e(u_0)$ on e, and two points on two edges of the faces corresponding to the intersecting polygons. If any of e or the two other edges are adjacent this implies the occurrence of an EV-event at $f(t_0)$. Otherwise, an EEE-event occurs at $f(t_0)$. An output-sensitive randomized algorithm to compute all boundary intersections for a collection of O(n) curved polygons runs in expected time O(n log n + k_e) (where k_e is the total number of intersections) [24]. Repeating this algorithm for each edge e in the collection of polyhedra (and letting $k = \Sigma k_e$) yields an overall expected run-time complexity for this algorithm of O(n^2 log n + k). While this algorithm possesses a worst-case complexity of O(n^3), which is better than the first two algorithms, note that, unlike the others, its worst case complexity does not dominate the time required to output the potential critical regions as a *sorted* list along f. If sorting is desired, the resulting worst-case complexity is again O(n^3 log n), matching that of the previous two algorithms.

3.8 Articulated Assemblies

An *articulated assembly* [10] is an object with moving parts, consisting of appendages that move relative to a fixed base. Here we will consider an object with polyhedral parts. The opaque view of such an object depends not only on the viewpoint but also on the object's *configuration*, the position of each appendage relative to the base. The configuration is represented by the values assigned to each of k parameters representing the degrees of freedom of movement of the collection of parts. Thus a given configuration may be regarded as a point in a k-dimensional *configuration space*. It is possible to compute a partition of the (d+k)-dimensional space formed by the Cartesian product of configuration space and a d-dimensional viewpoint space to obtain maximally connected regions of points (in the product space) from which the corresponding opaque views of the object are isomorphic. This partition is termed the *direct visual potential* and is similar in concept to the VSP for rigid objects. The direct visual potential may be computed by algorithms similar to those presented in section 3.2 for determining the VSP of a rigid object. Here, however, the potential critical regions are hyper-surfaces in (d+k)-dimensional space rather than curves on $\mathbf{S}^2$ or surfaces in $\mathbf{R}^3$.

For any fixed configuration, an articulated assembly may be regarded as a rigid object, which induces a particular VSP in viewpoint space. One may, therefore, consider configuration space separately from viewpoint space and compute a partition of configuration space alone, which yields maximally connected regions of points (in configuration space) for which the induced *VSP's* are isomorphic. We will not define precisely what it means for two VSP's to be isomorphic (i.e. to have the same qualitative, topological structure) except to point out that such a definition would be along the lines of that provided in section 2, regarding isomorphisms between LISG's.

In order to discover the critical regions of this partition, it is necessary to understand how the topological structure of the induced VSP changes as the configuration point varies in configuration space, i.e. as the parts of the object move (this is similar, in the case of rigid objects, to understanding how the qualitative structure of the opaque view changes as the viewpoint varies in viewpoint space). The qualitative structure of the VSP changes as the configuration point varies whenever regions (of any dimension) in the VSP are either created or destroyed (termed *definition events*) or whenever two existing regions (of the same dimensionality) coincide exactly (termed *coincidence events*). Potential critical regions in the VSP are classified as *fixed* or *varying* depending on whether they change shape or position as the configuration point varies. The intersections (of all dimensions) of these regions may be classified as fixed or varying too. Both potential critical regions and their various intersections can undergo definition or coincidence events as the configuration point varies. For example, two intersecting, varying (portions of) planes induced by distinct EV-events may become parallel (thus the line of intersection is destroyed). As a second example, a varying (portion of a) plane will itself be destroyed when the vertex and edge of the EV-event inducing it become collinear (or will be created when the opposite occurs).

Just as potential critical regions in viewpoint space are induced by EV- and EEE-events, so potential critical regions in configuration space are induced by definition and coincidence events. Each such region consists of the hyper-surface in configuration space for which a specific definition or coincidence event occurs. Note that these surfaces are not necessarily linear or quadric. Such regions are obtainable through analytical methods (we omit the details) and the partition in configuration space may be determined. Also, pruning of these potential critical regions may be performed, and the VSP at each cell of the remaining arrangement calculated (again, we omit the analytical details). The resulting actual critical regions in configuration space along with the collection of VSP's is called the *hierarchical visual potential*.

Upper bounds on the complexities of the direct and hierarchical visual potential are as follows. (Bowyer et al [10], do not prove that potential critical regions in either case are well-behaved, but this is implicit in their analysis. Certainly it is not obvious that critical regions arising from definition or coincidence events possess these properties.) We assume a k-dimensional configuration space and a perspective model viewpoint space. An articulated assembly with n total edges induces $O(n^2)$ potential critical surface patches due to EV-events and $O(n^3)$ such patches due to EEE-events. In the (k+3)-dimensional Cartesian product space this $O(n^3)$ collection of patches induces (after pruning) an arrangement of complexity $O((n^3)^{k+3}) = O(n^{3k+9})$. Thus the complexity of the direct visual potential is $O(n^{3k+9})$. Note that when $k = 0$, as in the case of rigid bodies, this reduces to the familiar $O(n^9)$ result.

For the hierarchical visual potential, note that as the configuration point varies, any of the $O(n^9)$ regions of the VSP may coincide with any other, leading to $O((n^9)^2)$ coincidence events. In addition, there can be $O(n^9)$ definition events. Thus there will be $O(n^{18})$ potential critical surface patches in configuration space, yielding an arrangement of complexity $O(n^{18k})$. After pruning, each cell of the resulting arrangement is associated with a VSP of complexity $O(n^9)$. Therefore an upper bound on the complexity of the hierarchical visual potential is $O((n^{18k})(n^9)) = O(n^{18k+9})$. Unfortunately, the prohibitive complexity of the hierarchical visual potential reduces its usefulness in most practical applications.

4 Aspect Graphs and Polyhedral Terrains

This section begins with a short introduction to *Davenport-Schinzel sequences*, which often arise in complexity analyses involving polyhedral terrains. This is followed by an analysis of the complexity of the VSP induced by a terrain when the viewpoint space is a (straight-line) *vertical* flight path. An algorithm for computing the VSP in this special case is also presented. We next consider the complexity of the VSP in the case of an *arbitrary* (straight-line) flight path above a terrain, and sketch an algorithm for its computation. Following this, we consider the complexity of the VSP in the orthographic model. The results in the case of vertical flight paths along with a bound on the number of actual critical regions (an argument involving the calculation of the complexity of the *lower envelope* of a collection of three-dimensional surface patches) allows this analysis to be recast in the context of *sparse arrangements* (those with low *vertical stabbing number*). This section concludes with an analysis of the complexity of the VSP induced by a terrain in the perspective model, where the set of viewpoints consists of all points in $\mathbf{R}^3$ above the terrain.

A *polyhedral terrain* (or, simply, a *terrain*) is a piecewise-linear bivariate function whose domain of definition is a simply connected region of the xy-plane. In the exposition to follow the domain of definition will be the entire plane. Thus a terrain is a two-dimensional piecewise-linear surface intersected by any vertical line at exactly one point. Without loss of generality, we will always assume the terrain to be triangulated. If a terrain has n edges it also has $O(n)$ vertices and faces, because there is a one-to-one correspondence between the features of the terrain and its projection onto the xy-plane (this projection can be regarded as a triangulated graph embedded in the plane).

4.1 Davenport-Schinzel Sequences

Davenport–Schinzel (DS) sequences [27] play an important role in much of the analysis involving terrains. A *DS*(n, s) *sequence* is a sequence on the alphabet of n symbols such that no two adjacent symbols are identical and such that there is no subsequence of length $s + 2$ (of not necessarily adjacent symbols) contained within it which consists of an alternation of two distinct symbols $(\ldots, a, \ldots, b, \ldots, a, \ldots, b, \ldots)$. DS($n$, s) sequences are important in that they can be used to specify successions of contiguous portions of the lower (or upper) envelope of a collection of continuous, univariate functions whose graphs possess a bounded number of pairwise intersections. More precisely, given n continuous, univariate (real-valued) functions $f_1, f_2, \ldots, f_n$ each defined over the same interval I whose graphs intersect pairwise in at most s points (for example, a collection of polynomials of fixed degree), the indices of the functions which achieve the minimum (or maximum) value for all f_i as I is traversed from left to right form a DS(n, s) sequence. When the functions are each defined over distinct connected subintervals of I only, the indices obtained in this fashion yield a DS(n, s+2) sequence. An important property of these sequences is that for any fixed s, the maximum length of a DS(n, s) sequence, denoted by $\lambda_s(n)$, is o(n log* n), and so is nearly linear in n. The exact form of this bound depends on the choice of s.

Note that the number of points along I, termed *breakpoints*, at which the identity of the function achieving the minimum (or maximum) value changes from f_a to f_b (i.e. those points along I at which the graph of the current function on the lower (or upper) envelope f_a is intersected by the graph of another function f_b, which then becomes the current function on the lower (or upper) envelope) is therefore $O(\lambda_s(n))$ for functions defined over I, or $O(\lambda_{s+2}(n))$ for functions each of which is defined over a distinct connected subinterval of I. These bounds can be shown to be tight in the worst case.

Extensions of the theory of envelopes to several dimensions yield bounds on the number of vertices in the upper or lower envelope of the graphs of a collection of n continuous, well-behaved *multivariate* functions [27]. (We will be concerned exclusively with those vertices which occur at the intersection, on the envelopes, of the graphs of $k + 1$ k-variate continuous, well-behaved functions. There are, however, vertices of other types occurring on the envelopes of such a collection of functions.) These results, in the case of trivariate and 5-variate functions, are used, for example, in proofs for terrains yielding bounds on the complexity of the VSP in the orthographic (section 4.4) and perspective (section 4.5) models.

4.2 Vertical Flight Paths

An analysis of the complexity of the VSP for a terrain σ (with n edges) may be found in [12] for the special case where the viewpoint space consists of a vertical flight path in $\mathbf{R}^3$ above σ. As we have seen (section 3.7) a naïve bound on the complexity of the VSP for arbitrary straight-line flight paths is O(n^3). In this special case, however, this bound can be improved upon.

In this section, we denote by F^* the orthogonal projection of any feature F onto the xy-plane. We will assume that the flight path f coincides with the positive z-axis and that the origin lies on the terrain itself. Since f is vertical, all points $f(t)$, $0 \leq t \leq +\infty$, project to a single point f^* (i.e. the origin). It is therefore possible to obtain a partial order for all edges e of σ by defining $e_i < e_j$ whenever there exists a ray emanating from f^* that intersects e_i^* before intersecting e_j^*. Note that for this to be a partial order, it may be necessary, by cutting the edges into no more than $2n$ sub-edges, to ensure the existence of some angular orientation θ_i about f which is not crossed by any edge. Otherwise, a *cycle* (where edges $e_1, e_{2, \ldots,} e_k$ exist such that $e_1 < e_2$, $e_2 < e_3, \ldots, e_{k-1} < e_k$ and $e_k < e_1$) could potentially occur (Fig 19). (We note that the ability to assign a partial order in this way is peculiar to polyhedral terrains and that for more general polyhedral scenes such a partial order often cannot be found. We shall see that this basic property of terrains permits the development of a large body of unique results.) Once a partial order is obtained, it can be completed to a total order by a topological sort.

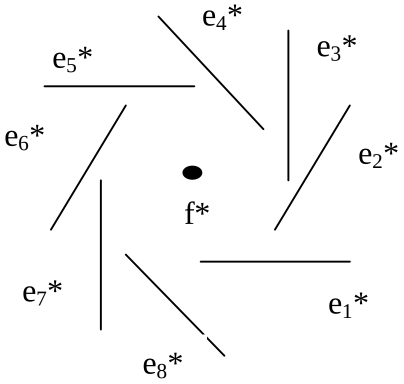

Fig 19. A cycle in the edge ordering. (From [12])

As in the case of k disjoint convex polyhedra (section 3.6) the analysis considered here seeks to bound the number of distinct edge triples associated with EEE-events that can induce actual critical regions. Every such edge triple is associated with extended sight lines (emanating from a viewpoint on f) that intersect each edge, are tangent to the terrain at those edges (with the possible exception of the edge furthest from the viewpoint) and do not intersect any other feature of the terrain anywhere between the viewpoint and the furthest edge. The analysis proceeds as follows. Select an edge e of σ. For each edge $e_i < e$ let $g_i(\theta)$ be the (partially defined, well-behaved, univariate) function whose value is the azimuth φ of the ray at orientation θ passing through the z-axis and e_i, and terminating at e, if such a ray exists (note that this ray need not lie wholly above σ). If, for a segment s, $I(s)$ denotes the angular interval of directions of all rays in the xy-plane emanating from f^* and meeting s^*, then g_i will be defined over the connected interval $I(e) \cap I(e_i)$. Also, define $\varphi_0(\theta)$ to be the (partially defined, well-behaved, univariate) function whose value is the azimuth φ of the ray at orientation θ passing through f^* and e (if such a ray exists). This function will be defined over the connected interval $I(e)$. Note that if one of the functions g_i (say g_j) attains the maximum value for all g_i and φ_0 at θ_0, then there exists a ray emanating from some point on f, tangent to σ at e_j and terminating at e without passing through any other feature of σ (Fig 20).

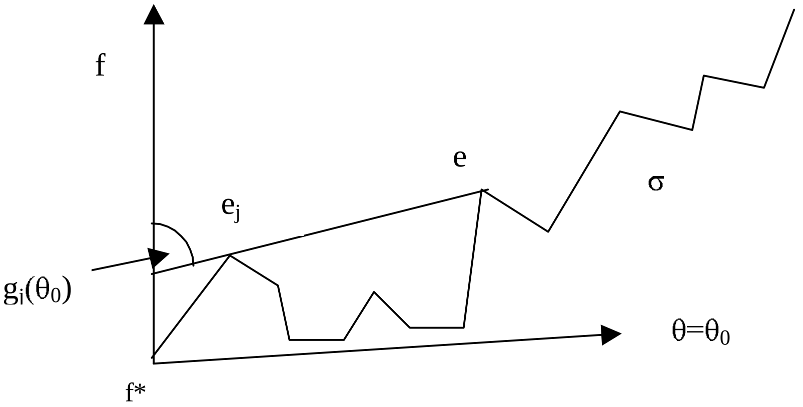

Fig 20. The ray with azimuth $g_j(\theta_0)$ passing through the positive *z*-axis, e_j and *e* and tangent to σ (shown in cross-section) at e_j. (From [12])

Furthermore for any edge $e_k < e, k \neq j$, if, at orientation θ_0, there exists a ray emanating from some point on the *z*-axis passing though e_k and terminating at e, such a ray will pass through some other feature of σ (in addition to e_k). Finally, note that if $\varphi_0(\theta_0)$ exists, any function g_k for which $g_k(\theta_0)$ is defined and $g_k(\theta_0) < \varphi_0(\theta_0)$ corresponds to a ray which emanates from the *negative* *z*-axis and passes through e_k and e. Thus if φ_0 should happen to attain the maximum value for all g_i and φ_0 at θ_0 then no ray with orientation θ_0 emanating from anywhere on the positive *z*-axis and terminating at e can be tangent to σ at some e_k less than e.

Now let $F_e(\theta) = \max \{ \max_{e_i < e} g_i(\theta), \varphi_0(\theta) \}$. $F_e(\theta)$ is the upper envelope of a set of partially defined, well-behaved, univariate functions each defined on a connected interval. Thus whenever $F_e(\theta_0) = g_j(\theta_0)$ for some j, there exists a line segment, at orientation θ_0, emanating from some point on f, tangent to σ at e_j and terminating at e without passing through any other feature of σ.

Therefore, the existence of a breakpoint on $F_e(\theta)$ at some value θ_0 involving functions g_a and g_b implies that there exists a viewpoint on f possessing an extended sight line at orientation θ_0 which intersects e_a, e_b, and e, is tangent to σ at both e_a and e_b and does not intersect any other feature of σ between the viewpoint and e. Edges e_a, e_b, and e thus constitute an edge triple for which there is an associated EEE-event inducing an actual critical region. (We note that points of intersection of these functions not occurring on the upper envelope $F_e(\theta)$ are associated with EEE-events in the transparent view which induce potential but not actual critical regions. Also, breakpoints on $F_e(\theta)$ involving one of the functions g_i and φ_0 do not correspond to EEE-events.) By bounding the number of breakpoints on $F_e(\theta)$ for each edge e in σ an upper bound on the complexity of the VSP may thus be obtained.

It is clear that $g_a(\theta)$ and $g_b(\theta)$ will not intersect more than twice in their common domain of definition (since the four line segments f, e_a, e_b, and e may not be collinear along more than two lines) while $g_a(\theta)$ and $\varphi_0(\theta)$ will not intersect more than once (since the edge e_a may not intersect the affine hull of f^* and e more than once). Also, it is possible, due to co-planarities among the various edges considered, for these functions to overlap along a continuous interval rather than a unique point. This does not affect the analysis, however, as the plane in which the edges lie intersects f in at most one point, and can therefore lead to the discovery of at most one triple. Such intersections may therefore be handled as if the functions intersected at a single point. As stated in section 4.1, the number of breakpoints occurring on the upper envelope of a set of n partially defined, well-behaved, univariate functions each defined on a continuous interval and intersecting pairwise at most s times is $\lambda_{s+2}(n)$. Thus, in this case, $s = 2$ and the total number of breakpoints on F_e is at most $\lambda_4(n) = \Theta(n \cdot 2^{\alpha(n)})$ where $\alpha(n)$ is the (slowly growing) inverse Ackermann function.

If the entire preceding argument is applied to each of the n edges e of σ (and noting that the number of actual critical regions induced by EV-events is $O(n^2)$) it is seen that the complexity of the VSP for a terrain with vertical flight path is therefore $O(n\lambda_4(n)) = O(n^2 \cdot 2^{\alpha(n)})$.

It is unknown whether the above bound is tight. However, it can be shown [12] that this complexity is at least $\Omega(n\lambda_3(n)) = \Omega(n^2\alpha(n))$, as follows. The *upper rim*, or *horizon*, of a collection of edges (with respect to a given viewpoint) consists of a set of segments each belonging to one of the edges such that no extended sight line emanating from the viewpoint and passing through one of these segments passes below any other edge in the collection (Fig 21). The complexity of the upper rim of a collection of n edges is equal to the complexity of the upper envelope of n line segments corresponding to those edges, and, since the existence of a such a collection having an upper envelope with complexity $\Theta(\lambda_3(n))$ can be demonstrated, is thus $\Omega(\lambda_3(n)) = \Omega(n\alpha(n))$ in the worst case.

A terrain may be constructed by positioning n edges in distinct vertical planes which are arbitrarily close to each other and far away from f, and by then dropping a steep wedge down from each of these edges to meet the xy-plane. The upper rim of these edges will contain $\Omega(n\alpha(n))$ breakpoints as seen from all viewpoints on some contiguous portion of f. In fact they will appear as vertices from these viewpoints due to the proximity of the wedges to each other. A hill with n horizontal edges is placed behind the wedges and can be positioned so that the (nearly-planar) actual critical regions induced by each pseudo-vertex and hill edge intersect f at distinct points (Fig 22). There will be $\Omega(n^2\alpha(n))$ such regions. Thus the complexity of the VSP in this example is $\Omega(n^2\alpha(n)) = \Omega(n\lambda_3(n))$.

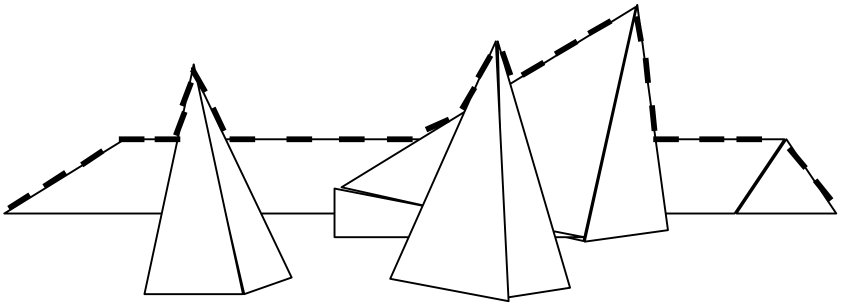

Fig 21. The upper rim (dashed lines) of a set of edges. (From [12])

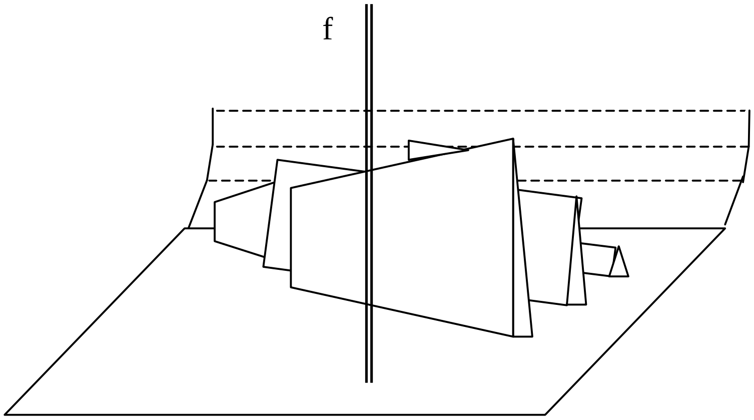

Fig 22. A terrain with *n* edges inducing a VSP with complexity $\Omega(n\lambda_3(n))$ when the viewpoint space is a vertical flight path. (From [12])

An algorithm for discovering all $O(n\lambda_4(n))$ actual critical regions on vertical flight path f is found in [9]. This algorithm makes use of a *skewed projection*, discussed in section 3.7. A special property of terrains under skewed projections is as follows. For a fixed edge e, the skewed projection $sp_e(\sigma)$ of a terrain σ (i.e. the collection of skewed projections of all faces of σ) is the entire area under a curve, rather than a general curved polygon. This can be seen by noting that if a line with endpoints $e(u_0)$ and $f(t_0)$ intersects σ, then all lines with endpoints $e(u_0)$ and $f(t)$, for $0 \leq t \leq t_0$, will also intersect σ (i.e. if $(u_0, t_0) \in sp_e(\sigma)$, then all points $(u_0, t) \in sp_e(\sigma)$, for $0 \leq t \leq t_0$) (Fig 23).

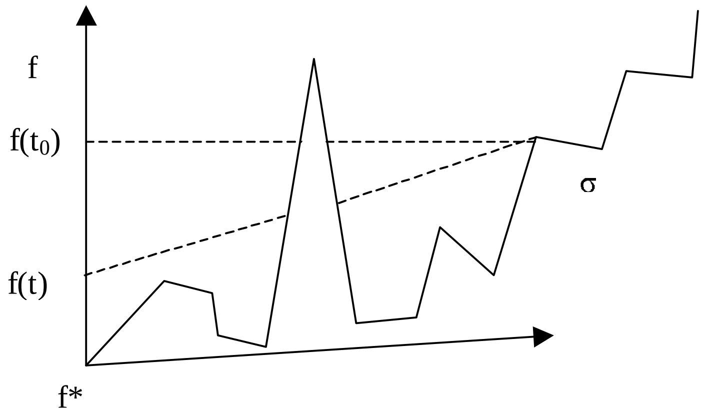

Fig 23. If a line segment with endpoints $e(u_0)$ and $f(t_0)$ passes through σ (shown in cross-section), so will a line segment with endpoints $e(u_0)$ and $f(t)$ for $0 \leq t \leq t_0$.

This implies that all actual critical regions can be discovered by determining, for each edge *e*, the *t* coordinate of all breakpoints of the upper envelope of the collection of O(*n*) curves which form the upper boundary of the skewed projection $sp_e(\sigma)$ (note that points of intersection of these curves occurring other than on the upper envelope correspond to potential critical regions which are not actual critical regions). Because each curve is either a line segment or a portion of a hyperbola (and therefore a partially defined, well-behaved, univariate function defined over a contiguous interval), there are $O(\lambda_4(n))$ such breakpoints in $sp_e(\sigma)$ for each edge *e*, and thus $O(n\lambda_4(n))$ breakpoints in total (agreeing with the previous result on the number of actual critical regions for a vertical flight path). This entire algorithm can be shown to run in $O(n\lambda_4(n) \log n)$ time (we omit the details).

4.3 Arbitrary Flight Paths

We begin this section with a construction demonstrating that the upper bound on the complexity of the VSP induced by a vertical flight path above a terrain, derived in the previous section, does not apply to arbitrary straight-line flight paths [12]. The construction consists of a terrain with two parallel rows of *n* tall, thin peaks in front of a hill with *n* faces (Fig 24).

The flight path *f* is chosen so that both rows of peaks lie between *f* and the hill. By choosing a hill edge and pairs of non-parallel edges from the two rows of peaks, it is not difficult to select $\Omega(n^3)$ triples of skew edges whose associated EEE-events induce actual critical regions that intersect *f* at $\Omega(n^3)$ distinct points. This implies that the number of actual critical regions induced by the arbitrary flight path *f* is indeed $\Omega(n^3)$.

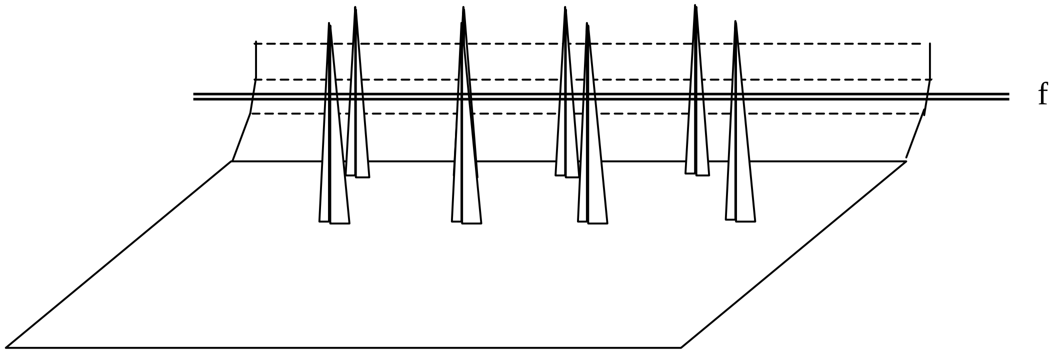

Fig 24. A terrain inducing $\Omega(n^3)$ actual critical regions on the arbitrary flight path *f*. (From [12])

This section now continues with a sketch of an output-sensitive algorithm [9] that can discover the $k = O(n^3)$ actual critical regions on an arbitrary straight-line flight path f above a terrain σ with n edges. We have seen that a total order can be established for the edges of σ with reference to a *vertical* flight path (section 4.2). Bern et al [9], incorrectly assert that a total order can also be established for the edges of σ, with reference to the *arbitrary* flight path f, such that any ray emanating from any point on f (having positive dot product with f in direction of increasing t) and passing through edges e_j and e_k will encounter e_j before e_k if and only if $e_j < e_k$. Although such an ordering cannot be established for f as a whole [8], it is possible to subdivide f into $O(n)$ sub-segments such that for each of these a total order can be established [2] (recall, however, that each total order arises from a topological sort on the corresponding underlying *partial* order established on the edges – this consideration will become important below). Then, since each ordering can be determined in $O(n \log n)$ time (as claimed in [9], we omit the details), the entire collection of orderings may be found in $O(n^2 \log n)$ time. Therefore the absence of a single total order does not increase the run-time complexity of the algorithm to be described.

We assume that f is parameterized by t, $0 \leq t \leq 1$. Let $S_t(i)$ denote the upper rim, with respect to a viewpoint $f(t)$, of the first i edges in the total order established for the segment of f containing $f(t)$. This upper rim has complexity $O(\lambda_3(i))$ (section 4.2). Let $E_t(i)$ be the list of terrain edges (of length $O(i)$) appearing on $S_t(i)$ and let $V_t(i)$ be the list of vertices (of length $O(\lambda_3(i))$) appearing on $S_t(i)$.

This algorithm maintains a priority queue of future events sorted by increasing t. These events are:

1) A future point $f(t)$ becomes collinear with an endpoint of edge e_i and a point on some edge in $E_t(i\text{-}1)$ (where the edge ordering is that in effect at $f(t)$).

2) A future point $f(t)$ becomes collinear with a point on edge e_i and a vertex in $V_t(i\text{-}1)$ (where the edge ordering is that in effect at $f(t)$).

3) A value of t is reached at which the edge ordering that is in effect changes.

Events of the first type are those EV-events visible in the opaque view for which the vertex lies behind the edge along an extended sight line emanating from $f(t)$, while events of the second type are EV-events visible in the opaque view for which the vertex lies in front of the edge along an extended sight line emanating from $f(t)$ (if the vertex is formed by adjacent edges in $E_t(i\text{-}1)$) or are EEE-events visible in the opaque view (if the vertex is formed by the alignment of non-adjacent edges in $E_t(i\text{-}1)$ along an extended sight line emanating from $f(t)$). Equivalently, all points $f(t)$ which are involved in one of these collinearity events are those possessing an extended sight line with the properties delineated in section 3.2 (see Fig 10).

The bound on the number of actual critical regions implies that there cannot exist more than $O(n^3)$ collinearity events in total (and, as we have seen, there are $O(n)$ events involving an edge ordering change). An endpoint of edge e_i may be involved in events of the first type with any of the $O(i)$ edges of $E_t(i\text{-}1)$ (and conversely each edge of $E_t(i\text{-}1)$ is involved in O(1) such events). The edge e_i may be involved in events of the second type with any of the $O(\lambda_3(i))$ vertices of $V_t(i\text{-}1)$ (and conversely each vertex of $V_t(i\text{-}1)$ is involved in O(1) such events). Therefore, for any fixed t, the n upper rims associated with $f(t)$ may generate as many as $O(n^2)$ events of the first type and $O(n\lambda_3(n))$ events of the second type.

The algorithm begins by computing all n upper rims with respect to $f(0)$ and the $O(n\lambda_3(n))$ future events generated by these rims. This can be accomplished using a standard hidden surface elimination algorithm so as not to dominate the time complexity that shall be exhibited for the rest of the algorithm (we omit the details). These initial events can be inserted into the priority queue in time $O(n\lambda_3(n) \log n)$ (alternatively, an initial priority queue consisting of $O(n\lambda_3(n))$ events could be built in $O(n\lambda_3(n))$ time).

The algorithm proceeds by examining all future events in chronological order until the list of unprocessed events has been exhausted. Assume there are k collinearity events in total. For each such event, either an edge or vertex that was not part of $S_t(i)$ just before the event becomes part of $S_t(i)$ just after the event, or an edge or vertex that was part of $S_t(i)$ just before the event is no longer a part of $S_t(i)$ after the event. In any case, two steps must be accomplished. First, all upper rims $S_t(j)$ for $j \geq i$ may require an adjustment consisting of the insertion or deletion of O(1) edges and vertices (for example, when a vertex is replaced by an edge in $S_t(i)$, that vertex can no longer appear in any $S_t(j)$

for $j \geq i$). Naively, this can be accomplished in $O(n\lambda_3(n))$ time when the n upper rims, each of size $O(\lambda_3(n))$, are stored in separate lists. This can be improved to $O(\log (n\lambda_3(n))) = O(\log n)$ by maintaining a similar list implementation for the n upper rims. Thus, for the entire collection of k collinearity events this can be accomplished in $O(k \log n)$ time.

Second, the $O(\lambda_3(n))$ events associated with the affected edge or vertex must be inserted into or deleted from the priority queue as appropriate (for example, when a vertex is replaced by an edge in $S_t(i)$ that vertex will no longer be involved in any events with any $S_u(j)$ for $u \geq t,\ j \geq i$). Thus each event may be processed in $O(\lambda_3(n) \log n)$ time. For the entire collection of k collinearity events, then, this can be accomplished in $O(k\lambda_3(n) \log n)$ time.

Another difficulty that arises is at the $O(n)$ points at which the total edge ordering changes. A change to the total edge ordering implies that a change has occurred in the underlying partial ordering (see section 4.2). What we would like is that the change in the partial ordering has a limited effect on the changes of the total ordering. For example, we would prefer that, at any point at which the total ordering changes, at most two adjacent edges in the total ordering (assume these are e_m and e_{m+1}) would swap places and no other edges would be affected. Thus only the silhouette containing the first m edges would need to be recomputed at this point (which could be accomplished so as not dominate the time complexity for the entire algorithm; we omit the details). The entire algorithm would therefore run in $O((n + k)\ \lambda_3(n) \log n)$ time in total. Unfortunately, it is by no means clear [2] that the underlying partial orderings associated with each of the total orderings (or the topological sorts used to produce the total orderings) will necessarily give rise to a collection of total orderings with the desired property. Only if this can be accomplished, would the asserted time bound for the algorithm be valid.

4.4 Orthographic Model

We now consider the complexity of the VSP in the orthographic model for a terrain σ with n edges [1], [7], [21], where the viewpoint space consists of all points on the sphere at infinity lying above the terrain. In the analysis presented in [1], the one-dimensional curves on $\mathbf{S}^2$ constituting the actual critical regions of the VSP are assumed to be in *general position*. This means that whenever two such curves intersect they do so at discrete points and that no three curves have a common point of intersection. The property of general position implies that the complexity of the VSP may be bounded by the number of its vertices. The analysis of [1], described next, obtains a bound for this number. As we shall see, this does not provide the best currently known bound for the complexity of the VSP in the orthographic model. However the analysis bears similarity to that used to obtain the best currently known bound for terrains in the perspective model (section 4.5), and it is instructive to present it here. It is also worth noting that a proof bounding the number of rays incident to four edges of a terrain, tangent to three of them, and lying wholly above the terrain (a seemingly unrelated construct), can also be cast in similar terms. This latter proof is sketched later on in this section.

Each vertex in the VSP may be associated with the two (EV- or EEE-) events corresponding to its incident actual critical regions. Thus the number of vertices such that at least one of the associated events is an EV-event is O(n^2n^3) = O(n^5). The naïve bound of O(n^3n^3) = O(n^6) for the number of vertices at which both events are EEE-events can be improved upon, as is now shown.

Select two edges e_1 and e_2 from σ. Form the collection of O(n) continuous, trivariate, real-valued, partial functions $F_i(s_1, s_2, \theta)$, one for each of the other edges e_i in σ, where s_1 parameterizes points on e_1, s_2 parameterizes points on e_2, and θ is the common angle of rotation about the z-axis of two rays r_1 and r_2, if they exist, which emerge from $e_1(s_1)$ and $e_2(s_2)$, respectively, and intersect e_i. The value of $F_i(s_1, s_2, \theta)$ is the *smaller* of the two azimuth angles formed by r_1 and r_2 if they both exist. If only one of these rays exists the value of $F_i(s_1, s_2, \theta)$ is the azimuth of that ray. If neither ray exists then $F_i(s_1, s_2, \theta)$ is undefined. It can be shown that the functions F_i are well-behaved. In particular, the graphs of each quadruple of these functions intersects in at most a constant number of points.

The existence of $\varphi = F_i(s_1, s_2, \theta)$ implies that one of the rays (with direction (θ, φ)) meets σ at e_i. If this ray is r_1 (r_2) then, if r_2 (r_1, respectively) exists, a ray r emanating from $e_2(s_2)$ ($e_1(s_1)$, respectively) in direction (θ, φ) will pass *above* σ at e_i (note that edge e_i intersects the vertical plane passing through r).

Let F be the lower envelope of this collection of functions, and assume that there exists an F_i such that $\varphi = F(s_1, s_2, \theta) = F_i(s_1, s_2, \theta)$ for some (s_1, s_2, θ). Then one of the rays r_1, r_2 (with direction (θ, φ)) meets σ at the edge from which it emanates, intersects σ at e_i and, furthermore, passes *above* all other edges of σ which intersect the vertical plane it defines. If this ray is r_1 (r_2) then a ray r emanating from $e_2(s_2)$ ($e_1(s_1)$, respectively) in direction (θ, φ) meets σ at the edge from which it emanates, and passes *above* all edges of σ which intersect the vertical plane it defines. Note that while this implies that each ray passes above the interiors of all *bounded* faces of σ intersecting the vertical plane it defines (since each is tangent to, or passes above, the edges intersecting this plane adjacent to such a face) this is not enough to imply the identical property for every *unbounded* face of σ (for a ray that passes above every edge of an unbounded face which intersects the vertical plane it defines may still penetrate into the region below that face). It may be necessary to introduce O(n) additional functions to ensure that, when $F = F_i$ for some (s_1, s_2, θ), each ray will pass entirely above the interiors of *all* faces of σ which intersect the vertical plane it defines. This, in turn, implies that, aside from any points of intersection with edges e_1, e_2 and e_i of σ, both rays lie wholly above σ.

We consider those vertices of F formed by the intersection of the graphs of *four* functions F_i. Each such envelope vertex may be associated with the existence of two rays, emanating from e_1 and e_2, tangent to σ at a total of *four* other edges (corresponding to the four intersecting function graphs), and passing above σ at all other points. The vertices in this collection for which exactly two of the four tangencies belong to each of the two rays form a subset of the set of vertices of F. If such a vertex occurs at (s_1, s_2, θ, φ) then (now considering the two rays as emanating *from* infinity and terminating at e_1 or e_2) (θ, φ) represents a viewing direction from which two rays emanate, are each tangent to σ at two edges and subsequent to this meet σ at a third edge further from the viewpoint. Thus each ray is an extended sight line associated with an EEE-event inducing an actual critical region (section 3.2) in the VSP. Moreover, these EEE-events are associated with distinct triples of edges. The viewpoint (θ, φ) is therefore a vertex in the VSP. We have thus reduced the problem of bounding the number of vertices in the VSP to one of determining a bound for the number of (a subset of) vertices of the lower envelope of a collection of O(n) continuous, trivariate, real-valued, well-behaved, partial functions (the introduction of additional functions due to unbounded faces may necessitate the removal of some points of quadruple intersection from consideration if they no longer appear on the lower envelope, however the analysis remains valid). A bound for this number is O($n^{3+\varepsilon}$) (see below), where $\varepsilon > 0$ may be selected as small as desired by an appropriate choice of the implied constant [27].

We note here that any point of quadruple intersection (for which the associated rays each intersect exactly two edges of σ) that does *not* lie on the lower envelope need not be considered in the above argument because at least one of the EEE-events corresponding to these rays will induce only a potential, but not actual, critical region. More precisely, given a point (s_1', s_2', θ', φ') at which the graphs of four functions intersect and given that the minimum value for all F_i at (s_1', s_2', θ') is achieved by some *other* function F_j, then one of these rays, emanating from e_1(s_1') or e_2(s_2') and passing through exactly two other edges, will, in addition, pass through some other feature of σ, thus causing the occlusion of the associated EEE-event from the viewpoint (θ', φ') (Fig 25).

Repeating the above argument for all O(n^2) pairs of edges e_1 and e_2 yields a bound of O($n^2 n^{3+\varepsilon}$) = O($n^{5+\varepsilon}$) on the number of vertices in the VSP. Hence the complexity of the VSP induced by σ in the orthographic model is O($n^{5+\varepsilon}$).

The foregoing argument made use of a result stating that the complexity of the lower envelope of O(n) well-behaved (section 3.1) d-dimensional surface patches in (d+1)-dimensional space is O($n^{d+\varepsilon}$) (in that argument $d = 3$). When $d = 2$ this result yields a complexity of O($n^{2+\varepsilon}$). However, as shown in [21], when the relative interiors of any triple of a collection of well-behaved two-dimensional surface patches are known to intersect in at most *two* points, this bound can be improved to O($n^2 \cdot 2^{c(\log n)^{1/2}}$) for some constant c depending on the degree and shape of the surfaces. This result is important for terrains in the orthographic model because, as also shown in [21], each extended sight line (a *ray* in this context) incident to a fixed terrain edge, tangent to three other edges, and otherwise lying wholly above the terrain (and therefore associated with a constant number of event occlusion endpoints, section 3.2) corresponds to a vertex in the lower envelope of a collection of two-dimensional surface patches (in a three-dimensional space) with the properties just stated. Thus an immediate bound of O($n \cdot n^2 \cdot 2^{c(\log n)^{1/2}}$) = O($n^3 \cdot 2^{c(\log n)^{1/2}}$) is obtained for the number of rays incident to a terrain edge, tangent to *three* other edges and otherwise lying wholly above the terrain. As we shall see shortly, this result yields an improved bound for the complexity of the VSP induced by a terrain in the orthographic case. First, however, we sketch the analysis of [21] showing how the bound on the number of rays incident to a fixed edge and tangent to three other edges is obtained. Many of the ideas presented here are similar to those found in the previous proof.

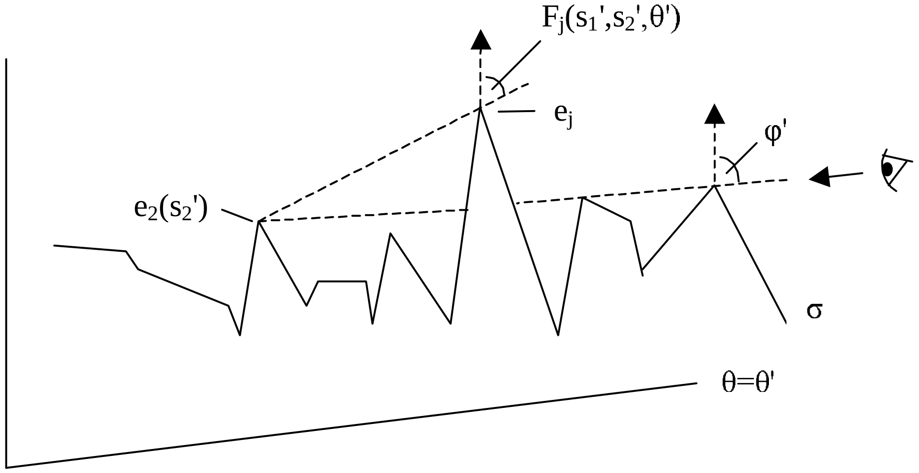

Fig 25. One of the rays associated with the intersection (*not* on the lower envelope) of the graphs of four functions F_i will pass below an edge, and thus penetrate a feature, of σ (shown in cross-section).

Select an edge e from σ and consider the collection of rays emanating from e. Each such ray r may be represented by three parameters (t, θ, φ) where t fixes the point $e(t)$ on e from which r emanates and where (θ, φ) is the direction of r. Each ray r thus corresponds to a point in a three-dimensional *dual* space. Note that the collection of rays intersecting e and any other edge e_i corresponds to a two-dimensional surface patch in this dual space. Each such surface patch defines a continuous, bivariate, real-valued, partial function $F_i(t, \theta)$, one for each of the other edges e_i, where t parameterizes points on e and θ is the angle of rotation about the z-axis of the ray, if it exists, emanating from $e(t)$ and intersecting e_i. The value of $F_i(t, \theta)$ is the azimuth of this ray if it exists. Otherwise, $F_i(t, \theta)$ is undefined. The functions F_i can be shown to be well-behaved. In addition, the intersection of the graphs of any three functions F_i corresponds to a ray passing through e plus a total of three other edges. Since in $\mathbf{R}^3$ there can be at most two such rays, we see that every triple of surface patches may intersect in at most two points.

Let F be the lower envelope of this collection of functions. If there exists an F_i such that $\varphi = F(t, \theta) = F_i(t, \theta)$ then a ray exists which emanates from $e(t)$, intersects e_i, and lies above the interiors of all bounded faces of σ which intersect the vertical plane it defines. Similar to the previous proof, functions may be added to the collection F_i to ensure that when $F = F_i$ for some (t, θ) this ray will in addition lie above the interiors of *all* (unbounded as well as bounded) faces of σ which intersect the vertical plane it defines.

Thus, aside from the two points of intersection with edges of σ, this ray will lie wholly above σ.

Those vertices of F formed by the intersection of the graphs of three functions F_i may therefore be associated with an extended sight line incident to an edge of σ, tangent to three other edges of σ and otherwise lying wholly above σ. As stated, the number of such vertices has been shown [21] to be $O(n^2 \cdot 2^{c(\log n)^{1/2}})$ where c is a constant depending on the nature of the functions F_i. (Points of intersection of the graphs of three functions F_i not appearing on the lower envelope may be ignored in this analysis, since they correspond to rays which penetrate some feature of σ other than one of the four edges, and thus do not correspond to EOE points). Applying the entire analysis to each of the $O(n)$ edges e of σ yields a bound of $O(n^3 \cdot 2^{c(\log n)^{1/2}})$ for the number of extended sight lines incident to an edge of σ, tangent to three other edges of σ and otherwise lying wholly above σ.

As promised, this result can be used to obtain an improved upper bound for the complexity of the VSP induced by a terrain in the orthographic model [7]. As discussed in section 3.2, the actual critical regions of the VSP consist of curves in $\mathbf{S}^2$ which have their endpoints at either EOE points or at endpoints of the potential critical regions of which they are a part. There are clearly $O(n^3)$ points of the latter type. In addition, since there are a constant number of EOE points associated with each extended sight line incident to an edge of σ, tangent to three other edges of σ and lying wholly above σ, and since there have been shown to be $O(n^3 \cdot 2^{c(\log n)^{1/2}})$ such extended sight lines, we see that the total number of actual critical region endpoints in the VSP is $O(n^3 \cdot 2^{c(\log n)^{1/2}})$. This immediately implies that there are at most this many actual critical regions in the VSP.

The analysis of [7] combines this result with that of section 4.2, where it was proved that the complexity of the VSP induced by a terrain for a vertical flight path is $O(n\lambda_4(n))$, to obtain the improved bound on the complexity of the VSP in the orthographic model. We observe that the result of section 4.2 may be restated in a slightly different form as follows: Any vertical line meets along its path at most $O(n\lambda_4(n))$ actual critical regions of the perspective model VSP. In particular, vertical lines *at infinity* also possess this property. Equivalently, each such line (with fixed θ and varying φ), corresponding to a *meridian* (line of longitude) on the sphere at infinity, thus meets at most $O(n\lambda_4(n))$ actual critical regions of the orthographic model VSP (since these arise as the intersections of actual critical regions in the perspective model VSP with the sphere at infinity). Since this is true for all meridians, this implies, in the terminology of [7], that the VSP in the orthographic model is therefore a *sparse* (two-dimensional) arrangement (of well-behaved curves) with low *vertical stabbing number*. A central result, proved in [7], is that the complexity of such arrangements is $O(NK)$, where N is the number of curves of the arrangement, and K is its maximum vertical stabbing number. A direct application of this result yields a bound on the complexity of the VSP in the orthographic model of $O((n^3 \cdot 2^{c(\log n)^{1/2}})\ (n\lambda_4(n))) = O(n^5 \cdot 2^{\alpha(n) + c\,(\log n)^{1/2}}) = O(n^5 \cdot 2^{c'\,(\log n)^{1/2}})$ for a constant c' slightly larger than c.

A worst-case lower bound on the complexity of the VSP in the orthographic model is $\Omega(n^5\alpha(n))$ [7] so that the given upper bound is nearly tight. A construction that achieves this bound consists, in part, of a collection of n arbitrarily close wedges (with $\Omega(n\alpha(n))$ breakpoints on its upper rim (section 4.2) placed in front of a hill with n faces (Fig 26). Because the wedges are so close, the breakpoints of the rim may be thought of as pseudo-vertices and the actual critical regions induced by each pseudo-vertex and hill edge are nearly horizontal planes and parallel to one another. Any meridian of (a particular region of) the sphere at infinity will intersect these regions at $\Omega(n^2\alpha(n))$ distinct viewpoints.

Also, a collection of n pyramids is placed arbitrarily close to a second hill with n faces and a set of n prisms with nearly vertical sides is placed in front of and far away from both of these. Because the pyramids are arbitrarily close to the hill, each alignment of a pyramid edge and a hill edge may be thought of as a pseudo-vertex (Fig 27). The prisms are arranged so that the actual critical regions induced by each pseudo-vertex and prism edge are nearly vertical planes and parallel to one another. Any line on the sphere at infinity parallel to the equator (in the same region containing the meridians mentioned above), will intersect these regions at $\Omega(n^3)$ distinct viewpoints (Fig 28).

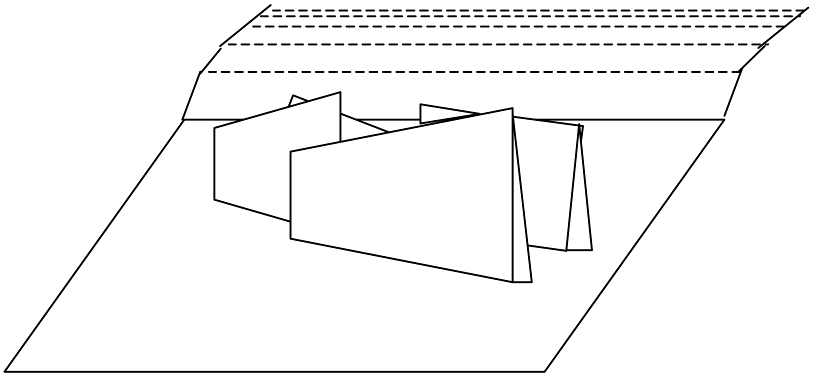

Fig 26. The first part of the construction. (From [7])

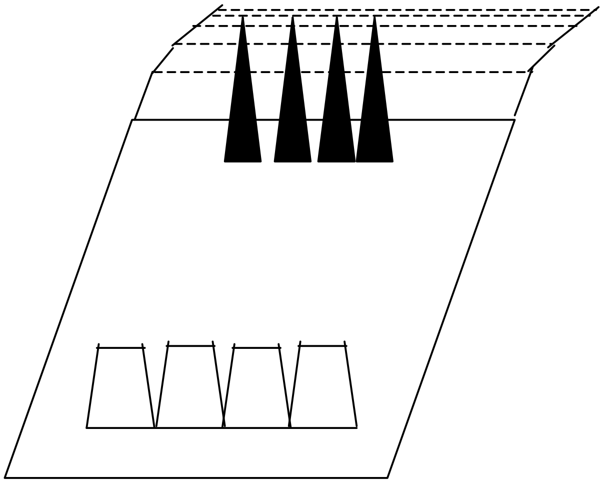

Fig 27. The second part of the construction. (From [7])

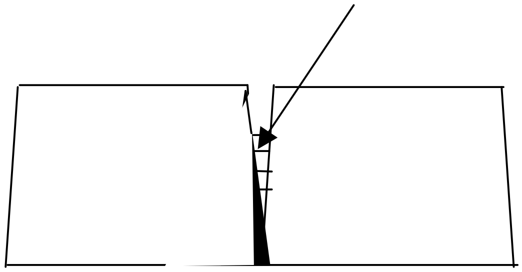

Fig 28. An alignment of a pseudo-vertex (formed by a hill edge and pyramid edge) and a prism edge as seen from a viewpoint in (some region of) the sphere at infinity. (From [7])

Thus the entire construction consists of a terrain with $O(n)$ edges inducing a grid of $\Omega(n^2\alpha(n))$ by $\Omega(n^3)$ actual critical regions in a particular region of the sphere at infinity (Fig 29). This immediately implies that the complexity of the VSP in the orthographic model for this terrain is $\Omega(n^5\alpha(n))$.

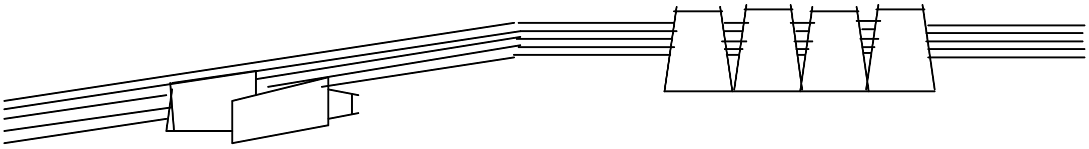

Fig 29. The complete construction. (From [7])

4.5 Perspective Model

The best known upper bound for the complexity of the VSP of a terrain σ with n edges in the perspective model arises from an analysis [1] similar to that presented at the beginning of section 4.4 for the case of the orthographic model. Because many of the requisite ideas are higher-dimensional analogs of those discussed earlier, we only provide a sketch of this analysis. Again, we consider only those viewpoints lying above σ, and again we assume that the actual critical regions of the VSP are in general position. This implies that whenever three of these surfaces intersect they do so at discrete points and that no four of them have common intersection. As before, this property allows the complexity of the VSP to be bounded by the number of its vertices. Here a vertex is formed by the intersection of *three* surfaces each associated with an EV- or EEE-event. There are $O(n^3n^3n^2) = O(n^8)$ vertices for which at least one of these is an EV-event. The naïve bound on the number of vertices associated with three EEE-events is $O(n^3n^3n^3) = O(n^9)$ and can be improved upon as is now shown.

Select three edges e_1, e_2 and e_3 from σ and form the collection of continuous, 5-variate, real-valued, partial functions $F_i(s_1, s_2, s_3, x, y)$, one for each of the other edges e_i in σ, where s_1, s_2 and s_3 parameterize points on e_1, e_2 and e_3, respectively, and the value of $F_i(s_1, s_2, s_3, x, y)$ is the maximum z-coordinate of the (at most) three points on the vertical line l passing through (x, y) and met by any of three line segments l_1, l_2, and l_3, if they exist, which emanate from $e_1(s_1)$, $e_2(s_2)$ and $e_3(s_3)$, respectively, each passing through e_i and terminating at l (if none of these segments exist the value of $F_i(s_1, s_2, s_3, x, y)$ is undefined). It can be shown that the functions F_i are well-behaved. In particular, the graphs of every collection of six of these functions intersect in at most a constant number of points.

The existence of $z = F_i(s_1, s_2, s_3, x, y)$ implies that one of these three line segments terminates at (x, y, z) and intersects σ at e_i. If either of the other two segments exist (emanating, say, from $e_k(s_k)$) then there exists an associated line segment (also emanating from $e_k(s_k)$) terminating at (x, y, z) and passing *above* σ at e_i.

Let F be the lower envelope of this collection of functions. If there exists an F_i such that $z = F(s_1, s_2, s_3, x, y) = F_i(s_1, s_2, s_3, x, y)$ for some (s_1, s_2, s_3, x, y) then a line segment exists which emanates from either $e_1(s_1)$, $e_2(s_2)$ or $e_3(s_3)$, intersects σ at e_i, terminates at (x, y, z) and lies above the interiors of all bounded faces of σ which intersect the vertical plane it defines. Furthermore, each segment emanating from one of the other two points and terminating at (x, y, z) lies above the interiors of all bounded faces of σ which intersect the vertical plane it defines. Similar to the analysis presented for the orthographic model, functions may be added to the collection F_i to ensure that when $F = F_i$ for some (s_1, s_2, s_3, x, y), each of the three segments will in addition lie above the interiors of *all* faces of σ which intersect the vertical plane it defines. Thus, aside from points of intersection with σ at edges e_1, e_2, e_3 and e_i, all three line segments will lie wholly above σ.

Those vertices of F formed by the intersection of the graphs of six functions F_i may therefore be associated with three line segments emanating from e_1, e_2 and e_3, respectively, tangent to σ at a total of six other edges, each passing above σ at all other points and terminating at a common point above the terrain. The vertices in this collection for which exactly two of the six tangencies belong to each of the three line segments may therefore be associated with three EEE-events inducing actual critical regions in the VSP. Moreover, these EEE-events are induced by three distinct triples of edges, and are visible from the common point of intersection of the three line segments (since these segments may be regarded as extended sight lines emanating *from* that point). Thus if such an envelope vertex occurs at (s_1, s_2, s_3, x, y) then the point (x, y, z) is a vertex in the VSP. Envelope theory tells us that a bound for the number of such envelope vertices is $O(n^{5+\varepsilon})$ where $\varepsilon > 0$ may be selected as small as desired by an appropriate choice of the implied constant [27]. (As in the orthographic model argument, points of intersection of the graphs of six functions F_i not appearing on the lower envelope may be disregarded in the above analysis.) Applying the foregoing argument to all triples of edges e_1, e_2 and e_3 yields a bound of $O(n^3 n^{5+\varepsilon}) = O(n^{8+\varepsilon})$ for the complexity of the VSP in the perspective model.

Although there exists a three-dimensional analog of the previously discussed (section 4.4) two-dimensional result for sparse arrangements with low vertical stabbing number [7], this is not helpful in finding an improved upper bound for the complexity of the VSP in the perspective model, as it was in the orthographic model. This result states that an arrangement of N well-behaved surface patches in $\mathbf{R}^3$ with maximum vertical stabbing number K, has complexity $O(N^2K)$. As we have seen (section 4.2), the perspective model VSP is an arrangement of surface patches with $K = O(n\lambda_4(n))$ [12] which are not necessarily well-behaved, so the three-dimensional result cannot be directly applied. However, each surface patch can be subdivided into a collection of patches which *are* well-behaved such that the three-dimensional result is indeed applicable to the entire collection of resulting patches. In fact, it can be shown (we omit the details) that, analogous to the orthographic case, the entire collection of well-behaved patches can be generated such that the total number of these patches is proportional to the total number of maximal extended sight lines (line *segments* in this case) emanating from some viewpoint in $\mathbf{R}^3$, lying wholly above the terrain, intersecting the terrain at four edges and tangent to the terrain at the first three edges encountered. Aronov and Halperin [3], unfortunately, have been able to construct an example showing that this number is $\Theta(n^4)$ in the worst case. Thus the application of the three-dimensional sparse arrangement result yields a VSP complexity of $O((n^4)^2 n\lambda_4(n))$ and does not improve the bounds previously established.

A worst-case lower bound on the complexity of the VSP in the perspective model is $\Omega(n^8\alpha(n))$ [7] so that the given upper bound is nearly tight. A construction that achieves this bound is an augmentation of the construction delineated previously for the orthographic case (section 4.4). A second set of pyramids and prisms is placed in front of a hill at a ninety-degree angle to the first set. The result of this is the construction of a terrain with $O(n)$ edges inducing a grid of $\Omega(n^2\alpha(n))$ by $\Omega(n^3)$ by $\Omega(n^3)$ nearly-planar actual critical regions in a particular region of $\mathbf{R}^3$ (note here that the grid is formed by a collection of two-dimensional surfaces in $\mathbf{R}^3$, whereas in the construction of section 4.4 the grid is formed by a collection of one-dimensional curves on $\mathbf{S}^2$). This immediately implies that the complexity of the VSP in the perspective model for this terrain is $\Omega(n^8\alpha(n))$.

5 Related Work

We conclude this survey with a sampling of research efforts that expand the aspect graph concept in several directions. One such effort deals with the complexities of aspect graphs induced by object silhouettes for several different kinds of viewpoint spaces. Another deals with the approximation of aspect graphs by simpler data structures. We also look at some structures that represent alternatives to aspect graphs. We discuss the computation of aspect graphs for non-polyhedral objects. Finally, we mention briefly an effort to construct the aspect graph using simpler algorithms than those discussed in this survey, and cite work on the aspect graph, and related structures, in two dimensions.

5.1 Silhouettes

The *silhouette* of a convex object with respect to a viewpoint is the projection (into the image plane or sphere) of the collection of object edges separating visible object faces from hidden ones. When the total number of edges in a scene is much greater than the number of silhouette edges visible from any particular viewpoint (this is often the case when, for example, non-polyhedral objects are approximated by large numbers of small polyhedral facets) the VSP induced by considering only silhouette edges associated with each viewpoint is often simpler than that induced by considering the entire collection of object edges from each viewpoint. In some applications it is often sufficient to determine the VSP for the silhouettes only.

The analysis in [32] examines several variations on this theme. Bounds on the complexity of the VSP induced by the arrangement of silhouettes of a collection of convex objects in a scene (the *silhouette arrangement*) are derived. Also, bounds on the complexity of the VSP induced by the silhouette arrangement with its hidden (with respect to the viewpoint) edge portions removed (the *silhouette map*) are derived. Finally, we consider all points on edges in the silhouette map for which there exists an extended sight line emanating from the viewpoint intersecting the edge point and then passing through the interior of some object face. The structure obtained when all such edge points are removed from the silhouette map is called the *union-of-silhouettes*. Bounds on the complexity of the VSP induced by the union-of-silhouettes are also determined. Note that essentially each variation considered arises as a modification to the definition of an *image* (see section 2). Previously an image consisted of the projections of *all* object vertices and edges onto the image plane or sphere (transparent image) or of the projections of *visible* (with respect to a given viewpoint) object vertices and edges or edge portions onto the image plane or sphere (opaque image). Now we will consider the projection of other selected subsets of object vertices and edges (with respect to a given viewpoint). We will redefine the subset of interest in each particular case.

Two different types of scenes are considered in [32]. The first consists of k distinct convex polyhedra with n edges in total (in this case the viewpoint space is an arbitrary straight-line flight path). The second consists of a terrain possessing k convex 'mountains', each with a base formed by a convex polygon in the xy-plane, and with n edges in total (here the viewpoint space is a vertical flight path). The silhouette of a convex mountain with respect to a viewpoint is the projection (into the image plane or sphere) of the collection of edges separating visible faces from hidden ones plus the collection of visible edges adjacent to the base.

Recall (section 3.6) that, in the case of convex polyhedra, the total number of actual critical regions was found to be $O(n^2k)$. It is shown in [32] that, for an arbitrary straight-line flight path f, this bound can be improved to $O(k^2n)$ when restricting consideration to the silhouette arrangement, silhouette map and union-of-silhouettes. For the first two of these cases, this bound is, in fact, tight.

The analysis, as in section 3.6, derives a bound on the number of distinct edge triples that may participate in EEE-events such that each edge belongs to a distinct polyhedron. Note however that here we are concerned exclusively with silhouette edges (in fact, to be more precise, silhouette arrangement edges). Recall that the complexity of a polyhedron P (the number of its vertices, edges and faces) is denoted by $|P|$. Assume, without loss of generality, that f coincides with the positive z-axis. Consider two polyhedra P_i and P_j. We imagine a half-plane with boundary f sweeping about f with longitude θ (denoted by $\gamma(\theta)$), for $0 \leq \theta \leq 2\pi$, and consider the slope of the outer lower bi-tangent line to P_i and P_j passing through f and coplanar with (and lying on) $\gamma(\theta)$. For each interval of θ for which the lower outer bi-tangent line is determined by exactly one edge of P_i and one edge of P_j, the function $\varphi_{ij}(\theta)$ representing this slope will be continuous and well-behaved. These intervals start or terminate at values of θ for which one of the edges of P_i (P_j) determining the bi-tangent line is replaced by another edge of P_i (P_j, respectively), or when the bi-tangent line becomes coplanar with a face of P_i or P_j (so that it is determined by three edges in total). The first event clearly occurs O($|P_i| + |P_j|$) times as the sweep proceeds. The second event is also seen to occur O($|P_i| + |P_j|$) times during the sweep by noting that when the bi-tangent line is coplanar with a face of P_i (P_j) there are at most two possible ways it may also be tangent to P_j (P_i, respectively). (Note that the planar region swept out by all rays emanating from f and coplanar with the face of one polyhedron intersects the other polyhedron, if at all, in a convex polygon, and that only at most two rays in this collection are tangent to that polygon (Fig 30)). This analysis shows that the domain of $\varphi_{ij}(\theta)$ consists of O($|P_i| + |P_j|$) intervals on each of which the function is well-behaved and continuous.

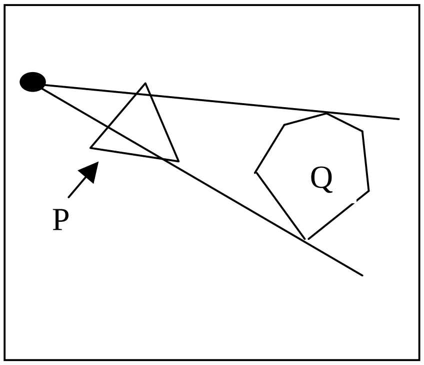

Fig 30. At most two rays originating on *f* and coplanar with a facet of *P* are tangent to *Q* (the diagram shows the slice of *P* and *Q* in the plane of the facet of *P*). (From [32])

One kind of EEE-event (we shall consider the other kinds shortly) involving three silhouette arrangement edges of distinct polyhedra P_i, P_j and P_l occurs when the lower outer bi-tangent line (passing through f) of P_i and P_j is collinear with the lower outer bi-tangent line of P_j and P_l (also passing through f). This can potentially occur only when these two lines have the same slope, or, in other words, for all values of θ for which $\varphi_{ij}(\theta) = \varphi_{jl}(\theta)$. Since $\varphi_{ij}(\theta)$ consists of $O(|P_i| + |P_j|)$ well-behaved, continuous curves and $\varphi_{jl}(\theta)$ consists of $O(|P_j| + |P_l|)$ well-behaved, continuous curves there can be at most $O(|P_i| + |P_j| + |P_l|)$ such points of intersection. Thus, for fixed polyhedra P_i, P_j and P_l the number of distinct edge triples of the silhouette arrangement participating in EEE-events associated with only the lower outer bi-tangents intersecting f is bounded by $O(|P_i| + |P_j| + |P_l|)$. Summing these results over all polyhedra P_i, P_j and P_l yields

$$\Sigma_{\,i\neq j\neq l}\, O(|P_i| + |P_j| + |P_l|) \;=\; \Sigma_{\,i\neq j\neq l}\, O(|P_l|) \;=\; \Sigma_{\,i}\,\Sigma_{\,j}\,\Sigma_{\,l}\, O(|P_l|) \;=\; O(k^2n)$$

triples in total. This entire foregoing analysis must be repeated for EEE-events involving all combinations of lower outer bi-tangents, upper outer bi-tangents and both inner tangents among triples of polyhedra (for example an EEE-event may also occur when the lower outer bi-tangent of P_i and P_j is collinear with the upper outer bi-tangent of P_j and P_l, etc.) There are sixteen such combinations and, since a similar analysis may be performed in each case, they each yield the same $O(k^2n)$ result. Thus the total number of critical surfaces overall is $O(k^2n)$. This, therefore, is the maximum complexity of the VSP induced by the silhouette arrangement of k convex polyhedra (with n total edges) for a straight-line flight path. Since the silhouette arrangement is a refinement of the silhouette map and union-of-silhouettes, the maximum complexity of the induced VSP in both of these cases is also $O(k^2n)$.

A lower bound construction is obtained [32] by placing k long and narrow convex polyhedra of negligible thickness ('needles') in a small volume around a viewpoint at which the silhouette arrangement (silhouette map, union-of-silhouettes) takes on its worst-case complexity. The needles are placed so that along a straight-line flight path in this volume each needle edge will align with each vertex in the silhouette arrangement (silhouette map, union-of-silhouettes) along some extended sight line (to produce an EV- or EEE-event depending on whether the vertex is the projection of an actual vertex in the scene or is a T-junction – see section 2) and so that each such alignment occurs at a distinct viewpoint. Since the worst-case complexities for the silhouette arrangement, silhouette map and union-of-silhouettes of k convex polyhedra with total complexity n are, respectively, $\Omega(kn)$ [20], $\Omega(kn)$ [32] and $\Omega(k^2 + n\alpha(k))$ [5] where $\alpha(k)$ is the inverse Ackermann function, the complexities of the induced VSPs are therefore, respectively, $\Omega(k^2n)$, $\Omega(k^2n)$ and $\Omega(k^3 + kn\alpha(k))$.

For the case of a general polyhedral terrain with n total edges and vertical flight path, recall (section 4.2) that, when considering *all* edges in the scene, worst-case lower and upper bounds on the number of actual critical regions were found to be $\Omega(n\lambda_3(n))$ and $O(n\lambda_4(n))$, respectively. The case of a terrain with k convex mountains (with n total edges) and vertical flight path, considered in [32], is just a special case of this, so those bounds apply here also. However, the analysis in [32] shows that the above $\Theta(k^2n)$

bounds for convex polyhedra also apply in this case for the silhouette arrangement and silhouette map and, in fact, can be further improved to $\Omega(kn)$ and $O(n\lambda_4(k))$ respectively for the union-of-silhouettes.

The argument for the upper bound for the union-of-silhouettes proceeds similarly to that for the case of convex polyhedra. For a vertical flight path f, a bound is sought on the number of distinct edge triples participating in EEE-events such that each edge belongs to a distinct mountain. This can be found by determining for each triple of mountains, M_i, M_j and M_l, the number of collinearities that can occur between the upper outer bi-tangent of M_i and M_j and the upper outer bi-tangent of M_j and M_l, both passing through f. If we denote the complexity of mountain M by $|M|$ then, similar to the previous argument, the function representing the slope of the upper outer bi-tangent of M_i and M_j, $\varphi_{ij}(\theta)$, will consist of $O(|M_i| + |M_j|)$ well-behaved, continuous curves in its domain of definition (in fact, here it is important to further specify that the curves intersect pairwise at most twice – see section 4.2 for a similar analysis with regard to the functions g_i).

Since we are dealing with a terrain and a vertical flight path, a partial order for all mountains M may be obtained by defining $M_i < M_j$ whenever any ray emanating from f and passing through both M_i and M_j intersects M_i before intersecting M_j (such a partial order is possible once cycles are eliminated - we have seen a comparable analysis in section 4.2 but omit the similar details here). The partial order may be completed to a total order by a topological sort. For a given mountain M_l, we consider the $O(k)$ slope functions φ_{il} (θ), for $0 < i < l$, and the $O(k)$ slope functions $\varphi_{lj}(\theta)$, for $l < j \leq k$. Note that, for a given θ, only the function in φ_{il} with *minimum* value is associated with an upper outer bi-tangent that does not pass through any of the mountains from M_1 to M_l (see section 4.2 for a similar analysis in a slightly different context). Likewise, for a given θ, only the function in φ_{lj} with *maximum* value is associated with an upper outer bi-tangent that does not pass through any of the mountains from M_l to M_k. Thus, a collinearity of two such bi-tangents can only be visible from f at those values of θ for which the lower envelope of the functions φ_{il} (θ) coincides with the upper envelope of the functions $\varphi_{lj}(\theta)$. It can be shown that the overlay of the upper and lower envelopes of these two $O(k)$ collections of functions, each function consisting of $O(|M_l| + |M_j|)$ $(0 < j \leq k,\ j \neq l)$ well-behaved, continuous curves in its domain of definition, where these curves intersect pairwise at most twice, has complexity $O((\lambda_4(k)/k)\ \Sigma_j\ (|M_l| + |M_j|))$. Repeating the entire argument for each M_l, $1 < l < k$, and noting that the number of edge triples associated with EEE-events involving *inner* tangents of mountain pairs (Fig 31) does not affect the upper bound (the details of this are omitted from [32]) yields a final bound on the number distinct edge triples associated with EEE-events of

$$\Sigma_l\ \ O((\lambda_4(k)/k)\ \Sigma_j\ (|M_l| + |M_j|)) \ = \ O((\lambda_4(k)/k)\ kn) \ = \ O(n\lambda_4(k)).$$

A lower bound construction may be obtained [32] by considering a terrain consisting of $\Theta(k)$ mountains ('peaks') in front of (with respect to all viewpoints on a vertical flight path f) a single mountain ('drum') with $\Theta(n)$ horizontal edges (Fig 32). As the viewpoint moves along f, each time it crosses a plane containing one of the facets of the drum, one drum edge adjacent to the facet will replace the other drum edge adjacent to the facet on the union-of-silhouettes (at the instant of crossing the two facets will be coplanar with the viewpoint). Each time this occurs there will be an alignment along an extended sight line from that viewpoint of the two drum edges and a point on each of the $\Theta(k)$ peak edges. Each such alignment corresponds to an EEE-event. Furthermore, each alignment will occur on the union-of-silhouettes as seen from that viewpoint. Since $\Theta(n)$ planes may be crossed there can, therefore, be as many as $\Theta(kn)$ EEE-events involving edges on the union-of-silhouettes. Thus the worst-case lower bound complexity of the induced VSP is $\Omega(kn)$. As we have seen, this bound is nearly tight.

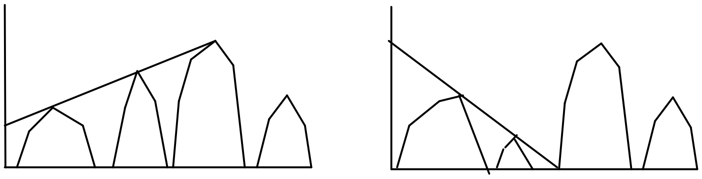

Fig 31. Two possible ways an EEE-event involving silhouette edges from three distinct mountains may occur. (From [32])

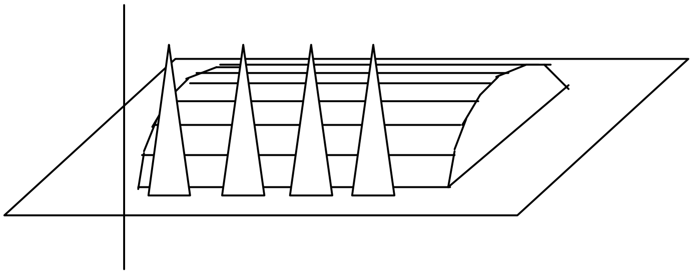

Fig 32. The lower bound construction for a terrain with $\Theta(k)$ mountains with total complexity $\Theta(n)$. (From [32])

Not addressed in [32] are new results pertaining to complexity bounds for the orthographic model and perspective model VSP induced by the silhouette arrangement, silhouette map and union-of-silhouettes of $\Theta(k)$ convex polyhedra with $\Theta(n)$ total complexity. The recent construction [4] demonstrating the lower bound worst-case complexity of the VSP induced by all edges in such a scene (section 3.6) may be modified to obtain a lower bound worst-case complexity for the union-of-silhouettes case. This is accomplished by replacing the drum in that construction (see Fig 18) with a second collection of $\Theta(k)$ needles. An analysis similar to the one outlined in section 3.6 shows that the induced VSP will have a worst-case lower bound complexity of $\Omega((nk^2)^2) = \Omega(n^2k^4)$ in the orthographic model and $\Omega((nk^2)^3) = \Omega(n^3k^6)$ in the perspective model. Since the silhouette arrangement and silhouette map are refinements of the union-of-silhouettes, the stated bounds apply equally well to these cases also.

Another new result concerning an upper bound on the orthographic model VSP can be determined by combining results pertaining to sparse arrangements of well-behaved surface patches (sections 4.4 and 4.5) with results specified above. Since, as shown above, no vertical line (in fact, no line in any direction and, in particular, no vertical line at infinity) will intersect more than $O(nk^2)$ critical regions and since the number of such critical regions is bounded by the number of critical regions induced by all edges in the scene (which, as we have seen, section 3.6, is $O(n^2k)$), we immediately derive an upper bound on the complexity of the orthographic model VSP of $O((n^2k)(nk^2)) = O(n^3k^3)$.

Since the surface patches of the VSP in the perspective model are not necessarily well-behaved, the preceding analysis concerning sparse arrangements does not apply. Trivially, however, the $O(n^6k^3)$ bound on the complexity of the perspective model VSP induced by all edges in the scene (recall section 3.6), serves as an upper bound for the complexity of the perspective model VSP induced by the silhouette arrangement, silhouette map and union-of-silhouettes for that scene.

5.2 Approximations to Aspect Graphs

The relatively prohibitive complexity of the aspect graph ($\Theta(n^6)$ and $\Theta(n^9)$ in the worst case for general non-convex polyhedra in the orthographic and perspective models, respectively, section 3.3) has motivated a number of researchers to seek approximations to aspect graphs. The goal of this research is to build simpler data structures with lower complexities while still providing sufficient utility for the problem domain at hand. In this section we examine several instances of these structures including finite-resolution aspect graphs, scale space aspect graphs and weighted aspect graphs.

5.2.1 Finite-Resolution Aspect Graphs

Up to now it has been assumed that any object feature visible from a given viewpoint is, in addition, *resolvable* from that viewpoint. This means that a viewer positioned at that viewpoint is able to distinguish the projection of that feature from the projection of all nearby features, regardless of how close the projections are to each other in the image plane or sphere. Such a viewer is said to possess *infinite spatial resolution capability*. Since, in fact, any real-world viewer has only finite spatial resolution capability, Shimshoni and Ponce [28] have suggested that the construction of an aspect graph taking this real-world limitation into account would yield a data structure of relatively low complexity that would still retain enough information to prove useful for the kinds of matching problems discussed in the introduction to this survey. Such an aspect graph would store finite-resolution opaque views at its nodes, rather than infinite-resolution opaque views.

An algorithm to construct a *finite-resolution aspect graph* for the case of non-convex polyhedra with n edges in the orthographic model is presented in [28]. A threshold parameter ε represents the minimum distance between feature projections in the (infinite-resolution transparent image) image plane that would allow them to be distinguished by a viewer with finite-resolution capability. Thus, for example, two vertices are resolvable when their projections in the infinite-resolution transparent image are greater than ε apart. In $\mathbf{S}^2$ this condition can be shown to occur for all viewpoints outside two diametrically opposite circular caps. The boundaries of these caps consist of viewpoints for which the projections of the two vertices are exactly ε apart. From viewpoints inside the caps these projections are less than ε apart and the two vertices are not resolvable. In general, any two features will not be resolvable in some two-dimensional area on $\mathbf{S}^2$.

The infinite-resolution transparent image consists of vertices, edges and T-junctions (section 2). In the context of finite resolution, potential critical events occur at those viewpoints for which any pair of image features are exactly ε apart. A complete catalog of all such events, and the potential critical curves they induce, is provided in [28].

Once the finite-resolution potential critical curves are calculated, standard algorithms (section 3.2) may be employed to perform region pruning (due to occlusion) and to compute the resulting arrangement in $\mathbf{S}^2$. At this point, in the infinite-resolution case, each remaining (actual) critical curve would separate regions with distinct infinite-resolution opaque views. However, in the finite-resolution case, it may still occur that some of the remaining critical curves separate cells that are associated with identical finite-resolution opaque views. A simple example showing why this can occur is as follows. Suppose that on one side of a potential critical curve vertices p and q are separated (in the infinite-resolution opaque view) by a distance greater than ε, but that p is within ε of the two edges adjacent to q. Suppose that on the other side of this potential critical curve p and q are separated by a distance less than ε. In the latter case, p and q are merged when calculating the finite-resolution opaque view. In the former case, p is merged with each of the edges adjacent to q when calculating the finite-resolution opaque view, which forces it to be merged with q also. In either case, the finite-resolution opaque view consists of a ‘thick vertex’ replacing both p and q and is identical on either side of the potential critical curve (Fig 33).

A supplementary pruning step is required to solve this problem. This step involves computing the infinite-resolution opaque view at each cell in the arrangement, and repeatedly merging *all* feature pairs in each view which are separated by a distance less than the ε threshold (a detailed catalog of merge steps and the order in which they are to be applied is supplied in [28]). This repeated merging of opaque image features may simplify the images to the point where they become identical in adjacent cells. When all merging steps are complete the final finite-resolution opaque view at each cell has been determined, and those potential critical regions found to separate cells with identical finite-resolution opaque views are removed from the arrangement.

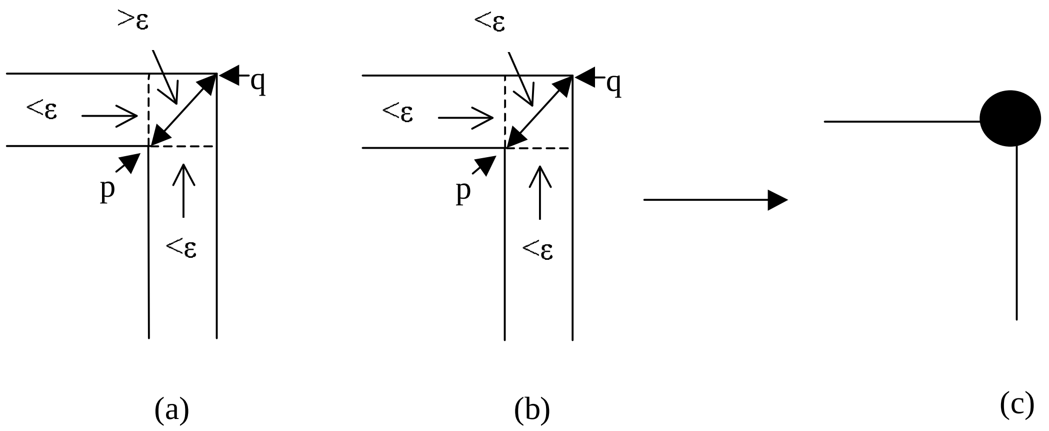

Fig 33. Both configurations (a) and (b) lead to an identical finite-resolution opaque view (c). (From [28])

There are O(n) vertices, O(n) edges and O(n^2) T-junctions (since each arises as the overlap of a pair of edge projections) in the infinite-resolution transparent image. Thus, in the context of finite resolution, there are O(n^4) potential critical events to be dealt with (and O(n^4) associated potential critical curves). These potential critical curves are not necessarily quadric. For example, such curves arising due to the interaction of two T-junctions are of degree six. However, it can be shown that they are well-behaved. Thus the arrangement of these curves has complexity O(n^8), which is worse than the infinite-resolution case (a worst-case lower bound on the complexity of this arrangement is not considered in [28]). Shimshoni and Ponce postulate, however, that in most instances the supplementary pruning step, described above, will be sufficiently effective so as to yield a final arrangement (and hence aspect graph) of complexity less than its infinite-resolution counterpart.

Since there are O(n^4) well-behaved potential critical curves, there exist O(n^5) event occlusion endpoints (EOE points, section 3.2). As in a previously discussed algorithm, these points are sorted (in O(n^5 log n) time) and, assuming a complexity of O(m) for the arrangement induced by these curves, this portion of the algorithm can be completed in O(n^5 log n + m log m) time. However, the time for the supplementary pruning step potentially requires the determination of distances between O(n^4) image feature pairs (in the infinite-resolution opaque image) for each of the O(m) cells in the arrangement, yielding a complexity of O(mn^4) for this step. Since m = O(n^8), the time for the supplementary pruning step dominates the rest of the algorithm in the worst case. The time complexity of this algorithm is inferior to that of its infinite-resolution counterpart. It is to be hoped that for most situations the simplicity of the resulting finite-resolution aspect graph relative to its infinite-resolution counterpart will justify the added cost associated with its computation.

5.2.2 Scale Space Aspect Graphs

In the previous discussion it seems intuitive that in general, as ε increases, the complexity of the resulting finite-resolution aspect graph should decrease. (Consider first, for example, the case where $\varepsilon = 0$; this implies that all object features are resolvable, and thus that the finite-resolution aspect graph is identical to the infinite-resolution aspect graph. Next consider the case where ε is so large that no two features are resolvable; the resulting finite-resolution aspect graph will consist of a single amorphous opaque view.) We may generalize this notion and consider a parameter that takes on a continuous range of values such that (ideally) the complexity of the associated aspect graph monotonically decreases as this parameter value increases. Such an analysis is presented in [16] where an analogy is obtained with the notion of *scale* in the domain of signal processing. Note, however, that any analogy between aspect graphs and signals is imperfect. In particular, it is difficult to find a parameter that reflects any real-world property such that this idealized monotonic behavior actually occurs.

In signal processing, scale allows for the 'smoothing out' of a signal in a controlled way as a parameter σ (the *scale parameter*) increases. For $\sigma = 0$ the original signal remains intact and as σ increases the signal is gradually smoothed out to a flat line. One may equate the 'importance' of a signal feature with the value of σ at which it disappears. With these considerations, the (idealized) notion of a *scale space aspect graph* may be formulated as follows [16]: for a given object, the scale space aspect graph induced at $\sigma = 0$ is the infinite-resolution aspect graph computed by standard algorithms (section 3.2). As σ increases, the induced scale space aspect graph monotonically decreases in complexity until, finally, the complexity is reduced to zero. In this way scale may be employed to obtain aspect graphs of any desired complexity merely by selecting a sufficiently large scale value.

The term *scale space* in the above terminology denotes the Cartesian product of viewpoint space and the (uni-dimensional) space defined by the scale parameter. Parallels between the notions of scale space and configuration space (section 3.8), in the case where there exists only a single degree of freedom, are drawn in [16]. Just as for configuration space, one may consider both the direct and hierarchical visual potentials induced by the scale space.

The threshold parameter described for the finite-resolution case (for the orthographic model), which sets a minimum distance between projected features in the image plane, provides a (nearly) suitable scale parameter. The parallel to this in the perspective model is angular distance between projected features along the image sphere. Other possible definitions, examined in [16], for the scale parameter include physical size of the viewer (in the infinite-resolution case the viewer is assumed to be a point located at a unique viewpoint) and degree of smoothing of the object under consideration (in the infinite-resolution case every indentation or extrusion of the object, no matter how miniscule, is equally significant in the aspect graph).

Using angular distance along the image sphere as the scale parameter, the scale space aspect graph is analyzed in detail in [16] for the case of non-convex polyhedra (polygons) in the perspective model in two dimensions. As for finite-resolution aspect graphs, potential critical regions arise at viewpoints from which pairs of image features are at a distance from each other equal to the scale parameter. Here, in addition, when the scale value is non-zero, potential critical regions associated with feature pairs arise at viewpoints far enough away from the object. The analysis details how these regions deform with changing scale value. It is shown that an appropriate choice of scale value must be made to obtain an aspect graph with complexity less than its infinite-resolution counterpart.

5.2.3 Weighted Aspect Graphs

Another approach to obtaining simpler aspect graphs is found in [6]. Here the *observation probability* (*weight*) of a point on an object feature is defined to be proportional to that fraction of viewpoint space from which it is visible. For example, in the orthographic model, a point in the relative interior of the face of a cube is visible from a hemisphere on $\mathbf{S}^2$. Thus its observation probability is one-half. The observation probability of an object feature is proportional to the fraction of viewpoint space from which any point on that feature is visible. Finally, the observation probability of any opaque view is proportional to the portion of viewpoint space from which all of its features are visible. Equivalently, this is the relative volume of its associated cell in the VSP. However, as detailed in [6], it is possible to compute the observation probability of an opaque view without first computing the VSP. An aspect graph node may be associated with each opaque view, and the observation probability of that view assigned to the node. We can consider those views associated with observation probabilities equal to or below a certain specified threshold as, in some sense, unimportant and remove the associated nodes from the aspect graph, yielding a data structure of greater simplicity.

We note that the analysis of [6] assumes a model of *homogeneous prior probabilities*. This implies that each viewpoint in the viewpoint space is equally likely to serve as the viewing location. Other models are possible, however. It is shown in [6] that, under the assumption of homogeneous prior probabilities, the perspective model is equivalent to the orthographic model since all cells with finite volume in $\mathbf{R}^3$ will be assigned an observation probability of zero. Only cells with infinite volume will have non-zero probability and, given an observation probability threshold of zero, will not be eliminated from consideration. But such cells are exactly those of the orthographic model.

5.3 Alternatives to Aspect Graphs

There exist various alternative structures to the aspect graph that answer the same kinds of visibility questions that are often handled by aspect graphs. The 3d visibility complex, for example, can be used to encode the view from any viewpoint in the perspective model. In contrast, other structures have been developed to answer more limited queries than those handled by aspect graphs but are advantageous in that they may be less costly to build or have smaller complexity. The uniform tessellation of the viewpoint space associated with a scene provides one example of this kind of alternative structure. It can be used to answer visibility queries of more limited types, for example those involving questions related to the notion of conservative visibility in a densely populated outdoor urban scene.

5.3.1 3D Visibility Complex

The set of all lines in $\mathbf{R}^3$ can be mapped to a four-dimensional dual space [13]. We associate a line having direction (θ, φ) with the plane oriented orthogonally to that line and passing through the origin. If the line intersects this plane at point (u, v) then the line maps to the dual point (θ, φ, u, v). The set of lines tangent to a single convex polyhedral object maps to a three-dimensional surface patch in this dual space.

Given a scene consisting of a collection of convex polyhedral objects, the analysis in [13] considers the arrangement of such three-dimensional surface patches in the dual space. The set of lines in $\mathbf{R}^3$ passing through a given perspective model viewpoint (each of which can be associated with a pair of extended sight lines, oriented in opposite directions, emanating from that viewpoint) map to a two-dimensional set of points in the dual space. The silhouette arrangement (section 5.1) induced by the scene with respect to that viewpoint, which is equivalent to the transparent image when considering object silhouette edges only, can be constructed by taking the intersection of this two-dimensional set with the arrangement of surface patches. This is because each point in the set that is also on one of the surface patches represents an extended sight line that is tangent to the polyhedron inducing that surface patch. The silhouette of any object with respect to the viewpoint may be determined by finding all such points of intersection with the object's corresponding surface patch. The silhouette arrangement is simply the union of these silhouettes.

Likewise, the entire transparent image induced by the scene with respect to a given viewpoint can be constructed by taking the intersection of the two-dimensional set with the set of four-dimensional volumes enclosed by the three-dimensional surface patches. This is because each point in the intersection represents an extended sight line that penetrates the interior of the polyhedron inducing the enclosing surface patch. The image of any object with respect to the viewpoint may be determined by finding all such points of intersection with the volume enclosed by the object's corresponding surface patch.

The determination of the silhouette map (section 5.1) induced by the scene with respect to a given viewpoint, which is equivalent to the opaque image when considering object silhouette edges only, requires a slight modification to the above scheme. Instead of mapping each line to a point in the dual space, one maps a set of collinear segments to each such point [13]. Consider the set of all line segments in $\mathbf{R}^3$ such that the relative interior of each segment lies entirely outside any of the objects in the scene and such that each segment endpoint meets the boundary of an object or else lies at infinity. Assuming there are n convex objects, there are potentially as many as n+1 segments collinear to any line (Fig 34). If this line maps to (θ, φ, u, v) in the dual space, then each of the collinear segments maps to that point also. Thus as many as n+1 distinct values may be associated with each point in the dual space under this mapping. Because of this, as many as n+1 branches may be encountered when, for example, traversing a one-dimensional slice in the dual space (Fig 35). (As a second example, this mapping appears as a collection of surfaces on a two-dimensional slice in the dual space (Fig 36).) From a slightly different perspective, this is referred to in [13] as the introduction of a fifth "pseudo-dimension" in the four-dimensional dual space. This dual mapping of line segments is known as the *3d visibility complex* induced by the scene.

Similar to the silhouette arrangement computation, the silhouette map with respect to a given viewpoint is determined by intersecting the 3d visibility complex with the two-dimensional dual set corresponding to the collection of extended sight lines associated with the viewpoint in $\mathbf{R}^3$. The difficulty here is that the opaque image must be found by following the correct branch at each branch point. Thus at each branch point the relative location of the viewpoint with respect to each object must be considered and branches corresponding to segments meeting occluded objects must be ignored.

In a similar fashion, the entire opaque image with respect to a viewpoint can be found by taking the intersection of the two-dimensional set with the volumes enclosed by the surfaces of the 3d visibility complex, taking care to follow the correct branch at each such intersection point.

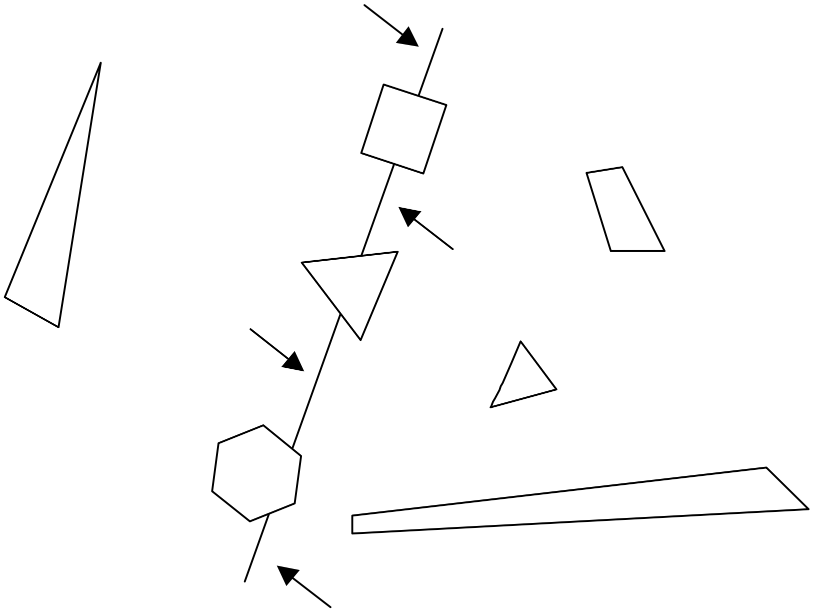

Fig 34. A set of collinear maximal line segments in $\mathbf{R}^3$.

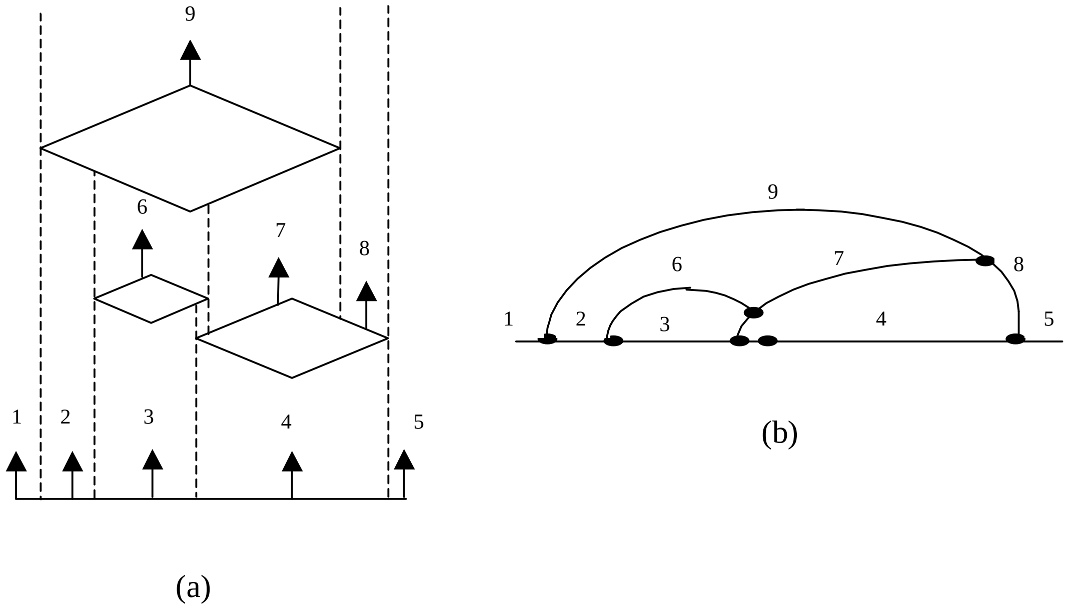

Fig 35. The mapping of segments on a one-dimensional slice in the dual space (b) for the given configuration of objects (a). The mapping is locally one-dimensional for all but a finite set of points (at which as many as *n*+1 branches may occur). (From [26])

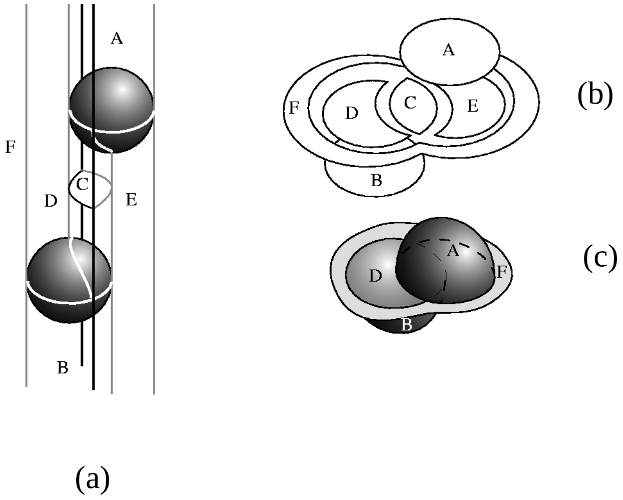

Fig 36. The mapping of segments on a two-dimensional slice in the dual space (c) for the given configuration of objects (a). In (b) the mapping is 'exploded' so that the various surfaces in (b) are more easily seen. (From [13])

5.3.2 Uniform Tessellation

Rather than partitioning viewpoint space into the (irregularly shaped) regions of the VSP it is sometimes sufficient to tessellate viewpoint space into uniform cells and to compute a representative view for each uniform cell rather than for each VSP region. Although it is obviously easier to partition the viewpoint space uniformly than to compute the VSP, the danger is that if the tessellation is not fine enough, certain views will be lost (for example, in those cells which admit more than one view). On the other hand, if the tessellation is too fine, work may be duplicated as identical views are computed for cells that would exist in the same region of the VSP. Nevertheless, this technique often provides a reasonable alternative to computing the VSP, and for that reason is widely used.

The analysis in [11] considers the special case of dense configurations of large numbers of randomly distributed objects, as are often found in outdoor urban scenes. Here, the perspective model viewpoint space is tessellated into uniform cells. In addition, for each uniform cell, the objects in the scene are classified into three categories. *Visible* objects are those for which at least a portion of the object is visible from some viewpoint within the cell. *Strongly occluded* objects are those which are completely occluded by a single object (called a *strong occluder*) from any viewpoint within the cell, and moreover the identity of the occluding object does not change as the viewpoint varies within the cell. *Weakly occluded* objects are those which are completely occluded from any viewpoint within the cell, but have no associated strong occluder. The union of visible and weakly occluded objects constitutes the *conservative visibility set* (it is worth noting here that this definition of conservative visibility is, in fact, but one of several to be found in the literature and that the conservative visibility set is often defined more loosely as *any* superset of the set of visible objects with respect to a given cell [31]). Instead of determining the opaque view associated with each cell, what is actually computed in the analysis of [11] is the conservative visibility set associated with each cell in the tessellation. It can be shown that in dense scenes of the type mentioned above, for any given cell, the number of weakly occluded objects is small in comparison to the number of strongly occluded objects. Thus the conservative visibility set provides a reasonable approximation to the set of visible objects from the cell. Cohen-Or et al [11], go on to describe an algorithm that finds the conservative visibility set for each cell and does so much more efficiently than any known algorithm to find the set of all visible objects for each cell. Thus if obtaining the conservative visibility set is sufficient within some problem domain, this algorithm provides a viable alternative to aspect graph related algorithms.

Some auxiliary results proved in [11] are also worth mentioning. It is shown that for densely occluded scenes with randomly distributed objects of identical size, the probability that an object is visible from a given cell decreases *exponentially* with the number of objects that can potentially lie between it and the cell. This latter number, in turn, increases directly with the number of objects in the scene and the distance of the object from the cell. A similar exponential relationship also holds for those objects in the conservative visibility set associated with the cell, although here it is shown that this relationship only holds when the size of the cell is less than the size of the object.

5.4 Miscellaneous Topics

In this survey we have restricted ourselves to scenes consisting of polyhedral objects. However a great deal of effort has been expended on the analysis of algorithms for computing aspect graphs induced by smooth objects with non-planar faces. A description that is representative of these efforts can be found in [22], which provides a brief discussion of algorithms for the general case and then supplies a more detailed analysis for the special case of a single solid of revolution in the orthographic model.

In the general case it is shown that there exist events corresponding to the EV- and EEE-events with which we are familiar, and that, in addition, there are other kinds of events which do not correspond to either of these (for example, the projections of two curved edges becoming tangent at a common point in the relative interior of both). It is also shown that the three steps of computing potential critical regions, finding the view at each region in the resulting arrangement and pruning the potential critical regions can each be reduced to finding the roots of a system of polynomial equations.

In the particular case of a single solid of revolution it is shown in detail that simplifications exist for the computations required at each of these three steps. A solid of revolution is a piecewise smooth surface generated by an algebraic curve rotated about the z-axis. It is shown that all views from any viewpoint on a line of latitude on $\mathbf{S}^2$ are identical. Thus all potential critical regions are also lines of latitude. Thus the VSP in this case is simpler than in the more general case.

In [23] an algorithm is described for finding the VSP of a collection of n disjoint spheres where the viewpoint space is a flight path along a geodesic in $\mathbf{S}^2$. The algorithm proceeds by tracking the movements of the centers of the spheres as the flight path is traversed.

A complete implementation of a simplified algorithm to compute the aspect graph for a convex polyhedron in the perspective model is detailed in [30]. The algorithm presented is not optimal in time or space (both are $O(n^4)$). However, this work is representative of others that describe alternative algorithms for computing VSPs which are easier to implement than those discussed in this survey.

Finally, we mention research efforts dealing with aspect graph related issues in two dimensions. A thorough treatment of aspect graphs induced by a single convex polygon in the perspective model in two dimensions can be found in [19]. Also, the mathematical theory behind a two-dimensional analog of the 3d visibility complex (section 5.3.1) is detailed in [26].

6 References

[1] Agarwal P.K. and Sharir M., “On the Number of Views of Polyhedral Terrains”, *Discrete and Computational Geometry*, vol. 12, pp. 177 – 182, 1994.

[2] Aronov B., Personal communication, 1999.

[3] Aronov B., Personal communication, 1999.

[4] Aronov B., Brönnimann H., Chiang Y-J., Halperin D. and Schiffenbauer R., “The Number of Views of Polyhedral Scenes”, to appear.

[5] Aronov B. and Sharir M., “The Common Exterior of Convex Polygons in the Plane”, *Computational Geometry Theory and Applications*, vol. 8, pp. 139 – 149, 1997.

[6] Ben-Arie J., “Probabilistic Models of Observed Features and Aspects with Application to Weighted Aspect Graphs”, *Pattern Recognition Letters*, vol. 11, pp. 421 – 427, 1990.

[7] de Berg M., Halperin D., Overmars M. and van Kreveld M., “Sparse Arrangements and the Number of Views of Polyhedral Scenes”, *International Journal of Computational Geometry and Applications*, vol. 7 no. 3, pp. 175 – 195, 1997.

[8] Bern M., Personal communication, 1999.

[9] Bern M., Dobkin D., Eppstein D. and Grossman R., “Visibility with a Moving Point of View”, *Algorithmica*, vol. 11, pp.360 – 378, 1994.

[10] Bowyer K., Sallam M., Eggert D. and Stewman J., “Computing the Generalized Aspect Graph for Objects with Moving Parts”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15 no. 6, pp. 605 – 610, 1993.

[11] Cohen-Or D., Fibich G., Halperin D. and Zadicario E., “Conservative Visibility and Strong Occlusion for Viewspace Partitioning of Densely Occluded Scenes”, *Eurographics ’98*, vol. 17 no. 3, 1998.

[12] Cole R. and Sharir M., “Visibility Problems for Polyhedral Terrains”, *Journal of Symbolic Computation*, vol. 7, pp. 11 – 30, 1989.

[13] Durand F., Drettakis G. and Puech C., “The 3D Visibility Complex: A New Approach to the Problems of Accurate Visibility”, *Proceedings of 7th Eurographics Workshop on rendering in Porto, Portugal (Rendering Techniques ’96, Springer Verlag)*, pp. 245 – 257, 1996.

[14] Durand F., Drettakis G. and Puech C. "The Visibility Skeleton: A Powerful and Efficient Multi-Purpose Global Visibility Tool", *Proceedings ACM SIGGRAPH 97,* pp. 89 – 100.

[15] Edelsbrunner H., *Algorithms in Combinatorial Geometry*, Springer-Verlag, 1987.

[16] Eggert D.W., Bowyer K.W., Dyer C.R., Christensen H.I. and Goldgof D.B., "The Scale Space Aspect Graph", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15 no. 11, pp. 1114 – 1130, 1993.

[17] Gigus Z., Canny J. and Seidel R., "Efficiently Computing and Representing Aspect Graphs of Polyhedral Objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13 no. 6, pp. 542 – 551, 1991.

[18] Gigus Z. and Malik J., "Computing the Aspect Graph for Line Drawings of Polyhedral Objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12 no. 2, pp. 113 – 122, 1990.

[19] Gualtieri J.A., Baugher S. and Werman M., "The Visual Potential: One Convex Polygon", *Computer Vision Graphics and Image Processing*, vol. 46, pp. 96 – 130, 1989.

[20] Halperin D., "Arrangements", in *Handbook of Discrete and Computational Geometry*, CRC Press LLC, Boca Raton, Florida, pp. 389 – 412, 1997.

[21] Halperin D. and Sharir M., "New Bounds for Lower Envelopes in Three Dimensions, with Applications to Visibility in Terrains", *Discrete and Computational Geometry*, vol. 12, pp. 313 – 326, 1994.

[22] Kriegman D.J. and Ponce J., "Computing Exact Aspect Graphs of Curved Objects: Solids of Revolution", *International Journal of Computer Vision*, vol. 5, pp. 119 – 135, 1990.

[23] Lenhof H.-P. and Smid M., "Maintaining the Visibility Map of Spheres While Moving the Viewpoint on a Circle at Infinity", *Algorithmica*, vol. 13, pp. 301 – 312, 1995.

[24] Mulmuley K., "A Fast Planar Partition Algorithm, I", *Proceedings 29th IEEE Foundations of Computer Science*, pp. 580 – 589, 1988.

[25] Plantinga H. and Dyer C.R., "Visibility, Occlusion, and the Aspect Graph", *International Journal of Computer Vision*, vol. 5 no.2, pp. 137 – 160, 1990.

[26] Pocchiola M. and Vegter G., "The Visibility Complex", *International Journal of Computational Geometry and Applications*, vol. 6 no. 3, pp. 279 – 308, 1996.

[27] Sharir M. and Agarwal P.K., *Davenport - Schinzel Sequences and Their Geometric Applications*, Cambridge University Press, 1995.

[28] Shimshoni I. and Ponce J., “Finite-Resolution Aspect Graphs of Polyhedral Objects”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19 no. 4, pp. 315 – 327, 1997.

[29] Snoeyink J., “The Number of Views of Axis-Parallel Objects”, *Algorithms Review*, vol. 2, pp. 27 – 32, 1991.

[30] Stewman J. and Bowyer K., “Direct Construction of the Perspective Projection Aspect Graph of Convex Polyhedra”, *Computer Vision Graphics Image Processing*, vol. 51, pp. 20 – 37, 1990.

[31] Teller S., Personal communication, 2000.

[32] Zhang L, Efrat A., Guibas L. and Hall-Holt O.A., “On Incremental Rendering of Silhouette Maps of a Polyhedral Scene”, *ACM-SIAM Symposium on Discrete Algorithms*, 2000.