On Incremental Rendering of Silhouette Maps of a Polyhedral Scene

Alon Efrat* Leonidas J. Guibas† Olaf A. Hall-Holt†
Li Zhang†

Computer Science Department
Stanford University
Stanford, California 94305
{alon,guibas,olaf,lizhang}@cs.stanford.edu

Abstract

We consider the problem of incrementally rendering a polyhedral scene while the viewpoint is moving. In practical situations the number of geometric primitives to be rendered can be very large — hundreds of thousands or millions, but these may come from only a moderate number of objects that happen to have been finely tessellated. It is sometimes advantageous to render only the silhouettes of the objects, rather than the objects themselves, and then exploit coherence or other methods to optimize the rendering of single-object regions with uniform reflectance properties. Such an approach is also regularly used in the domain of *non-photorealistic rendering*, where the rendering of silhouette edges plays a key role. The hard part in efficiently implementing a kinetic approach to this problem is to realize when the rendered silhouette undergoes a *combinatorial change*.

In this paper, we obtain bounds on a number of combinatorial problems involving the complexity of these events for a collection of k objects, with a total of n edges. We assume that our objects are convex polytopes, and that the viewpoint is moving along a straight line, or along an algebraic curve. The resulting bounds will then depend on both the number of objects, and the total number of edges, in such a way that we can describe the advantages of focusing on the silhouettes of combinatorially large objects. We also study the special case when the scene is a *polyhedral terrain*, and present improved bounds for this case. In addition to event bounds, we also obtain algorithms that compute all the changes occurring during a linear motion, (both for general scenes and for terrains).

*Partially supported by a Rothschild Fellowship and by DARPA grant DAAE07–98–C–L027.

†Partially supported by NSF grant CCR–9623851 and by ARO MURI grant DAAH04–96–1–007

1 Introduction

There is an increasing demand in computer graphics applications for rendering large and complex environments involving scenes with many millions of polygons. The computational demands of such a task have to be addressed by both improved hardware and better algorithms. The very high complexity of these environments in terms of simple geometric primitives, such as triangles, is in part an artifact of the traditional rendering pipeline of current graphics systems, which are based on triangle scan-conversion as the basic primitive. In general the number of different objects present in a scene is much less than millions – and the high triangle count is due to the tessellation of more complex curved objects into polyhedral approximations that can be rendered by the hardware.

Triangle edges have to be handled properly in order to obtain high quality renderings of a scene. There is a vast literature in computer graphics on how to deal with edge problems such as jaggies, antialiasing, etc. Yet it is important to realize that not all edges are created equal. Edges in the rendered image separating two different objects are much more likely to be problematic than edges separating two polygons belonging to the same object. Across the former we will have depth discontinuities, a different reflectance function on the two sides, different colors, etc. Across the latter simple edges, interpolatory smoothing techniques work well to simulate the appearance of a smooth surface. The former edges are *silhouette* edges, and they are the topic studied in this paper.

We consider a small number of objects that have been tessellated into a much larger number of triangles. Given a point of view, each object has a *silhouette*, a collection of edges forming closed cycles that separate triangles visible from triangles invisible to the viewpoint. We focus on the geometric structure of these silhouettes and their arrangements. Such silhouette structures are important in efficient rendering. For example, we can calculate the shadow and perform occlusion culling by using the silhouette of the objects, or we can answer ray shooting queries efficiently if we know silhouette structures. In recent work [9], Gu etc. present an efficient image-based rendering method by using the object silhouette to clip the images rendered on a coarse geometry. If we can compute these silhouettes for the viewpoint, and also maintain them as the viewpoint moves around (incremental rendering), not only do we know the most important edges in the image to be rendered, but we can also facilitate many other rendering operations.

The hard part in efficiently implementing any algorithm for rendering moving objects is to realize when the image undergoes a *combinatorial change*, defined as a change where either the polytope edges that define the silhouette are changed or the topology of the silhouette structure is changed (e.g. when two silhouettes start or stop intersecting or when a hole in their union appears or disappears).

The input to the first type of problems we investigate is a set $\mathcal{S} = \{P_1 \ldots P_k\}$ of convex polytopes in 3-D that we have to render, and a viewpoint p. As mentioned, we make the realistic assumption that the the number of polytopes k is much smaller than the total number n of vertices of these polytopes. Let S be a plane representing the display surface. The *shadow* of an object is the perspective projection of the object on S from p. The *silhouette* of a polytope is the boundary of its shadow, which is a convex polygon. The *silhouette arrangement* is the arrangement of the silhouettes. The *silhouette map* is the arrangement on S with the hidden part removed. Formally, we assign a unique color (or ID) to each object. For any point q on the background sphere S, assign q the color of the first object hit by the ray starting from the perspective point and shooting to q. The boundary of the monochromatic regions is exactly the silhouette map. The *silhouettes-of-union*, (abbreviated *uo-silhouette*) is the boundary of the union of all the shadows on S.

We describe combinatorial bounds on these geometric structures, in each of the following three cases

- *Static view-point* — the viewpoint p is static.
- *Linear motion* — p moves along a straight line, and the goal is to bound the number of combinatorial changes.
- *Algebraic motion* — p moves along an algebraic curve, and the goal is to bound the number of changes each of the structures goes through.

In this paper we present these and other bounds using the following notation. Let $\beta_s(n) = \lambda_s(n)/n$, where $\lambda_s(n)$ is the maximum length of a Davenport-Schinzel sequence of order s with n letters. When s is constant, $\beta_s(n)$ is an extremely slowly growing function — in particular, $\beta_3(n) = \alpha(n)$, the functional inverse of Ackermann's function.

The bounds are summarized in Table 1. For the static viewpoint, the bounds given are for the combinatorial size of the respective structure. For the moving viewpoint columns, the bounds indicate the number of combinatorial changes in the relevant structure.

For terrains, we consider the measure in terms of the number of 'mountains' in a terrain. Roughly speaking, a mountain is a up-convex body with its base on the xy-plane. For a terrain with k mountains and n vertices and a vertically moving viewpoint, we are able to obtain a roughly $\Theta(kn)$ bound on the number of combinatorial changes in the silhouette map and uo-silhouette. When a terrain has n vertices, we can always decompose it into $O(n)$ mountains where each mountain is formed by taking the vertical prism under a triangle. Although in the worst case, k can be as large as $\Theta(n)$, in real applications, the number of mountains is usually small. This bound is then better than the roughly $\Theta(n^2)$ bound on the number of combinatorial changes of the visibility map for a vertically moving view point where the lower bound can be achieved by two mountains.

One key lemma we use to achieve the upper bounds is to bound the number of so called EEE events, i.e. the number of times when the shadow of three edges on the silhouettes come together. This is in turn obtained by bounding the number of lines that touches three convex polytopes and passes through the perspective point during the entire motion. We show that, for a linearly moving point, this number is linear in the total complexity of these three convex polytopes instead of quadratic, and thus obtain better upper bounds.

Based on the upper bound on the combinatorial changes, we can obtain algorithms that compute the all the changes occurring during a linear motion, (both for general scenes and for terrains) in time (respectively) $O(k^2 n \log n)$ and $O(kn\alpha(n) \log^2 n)$.

Related results: Similar problems were investigated both analytically (usually in the Computational Geometry community) and empirically (in the Computer Graphics community). Among the analytic results, several authors considered the problem of computing the visibility maps from a moving viewpoint [12, 10, 5]. In [4], de Berg, Halperin, Overmars and van Kreveld described a list of results regarding the complexity of the aspect graph for different scenarios, and its relations to complexity of arrangements. Barequet et al. [2] showed how to use the BAR-tree to obtain fast rendering of the silhouette of a (not necessarily convex) polytope.

structure	Static viewpoint	Linear motion	Algebraic motion
silhouette arrang.	$\Theta(kn)$	$\Theta(k^2 n)$	$\Theta(kn^2)$
silhouette map	$\Theta(kn)$	$\Theta(k^2 n)$	$\Theta(kn^2)$
union of silhouette	$\Theta(n\alpha(k) + k^2)$	$O(k^2 n)$ $\Omega(kn\alpha(k) + k^3)$	$O(kn^2)$ $\Omega(n^2\alpha(k) + k^2 n)$

Table 1. The bounds on the complexity of silhouette structures

Other works [11, 8, 9] in the graphics community also use shadows and silhouettes as a means to simplify the description of a complicated environment. Silhouettes are also useful in collision detection [6], and other applications.

2 Lower bounds

2.1 Silhouette structures from a static point

From a fixed viewpoint, the shadow of each polytope is a convex polygon on the background plane. For k convex polygons with n vertices, the tight bounds of $\Theta(kn)$ and $\Theta(n\alpha(k) + k^2)$ are known for, respectively, the complexity of their arrangement and the boundary of their union [1]. These bounds yield the tight bounds for silhouette arrangements and uo-silhouettes. For the silhouette map, we will construct an example to show complexity of $\Omega(kn)$ and thus obtain a tight bound of $\Theta(kn)$ for silhouette maps.

Figure 1 (i) shows an example of our construction of k fat convex polygons, and figure 1(ii) shows an enlargement of a neighborhood containing k polygon corners. The goal of the construction is to create n/k corner neighborhoods with silhouette map complexity k^2 each, yielding a total complexity of $(n/k) \times \Omega(k^2) = \Omega(kn)$. The corner of a polygon consists of the corner vertex, and two incident edges, which we will call the left and right edges. Let us focus on a particular corner neighborhood N (such as the one depicted in figure 1(ii)). To obtain the desired complexity, we need to ensure that the left edge of each polygon corner contributes a vertex to the silhouette map at the right edge of all preceding polygons in the depth order. Intuitively, we seek a set of nearly parallel left edges with increasing slope, such that each left edge lies to the right of the endpoints of the previous left edges (in the depth order).

To verify that this is possible, consider a sequence of tangents to the unit hyperbola in the first quadrant of the plane ($y = \sqrt{x^2 - 1}$). Let the tangent points be a_i, where they are given in order of increasing x-coordinate.

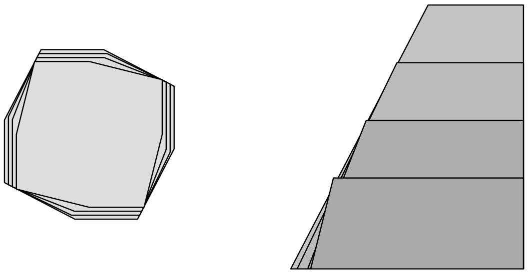

Figure 1. The lower bound construction of the silhouette map. Left: The global arrangement. Right: Close up of a corner.

For a tangent to the hyperbola at a_i, let b_i be the intersection of the tangent with the asymptote $y = x$. Let the b_i be our corner vertices, and let the left edge from b_i extend down toward a_i. Thus a left edge must pass below the corner vertices of all preceding polygon corners. We still need to construct the spacing among the a_i. First let all the right edges proceed to the right, with slope $0 < \alpha < 1$. Now let a_{i+1} be the intersection of the previous right edge with the hyperbola. That is, we make each right edge go up toward the hyperbola, and choose all tangent points except the first one to be the intersections of the right edges with the hyperbola. Thus we also ensure that any left edge will be above the tangent points corresponding to all previous left edges, and hence visibly intersecting all previous right edges.

To construct the k polygons from such a neighborhood, we place n/k rotated copies of the neighborhood near the vertices of a sufficiently large n/k-gon, and connect corresponding left and right edges. In order for this to work, we need to make sure that the corner angles are sufficiently large. In the neighborhood construction the position of the first tangent point and the parameter α were left unspecified. These together determine a lower bound on all the corner angles, which can be anything less than π, so we can always make them large enough.

In summary, we have that:

Theorem 2.1. *In the worst case, the silhouette arrangement, silhouette map, and uo-silhouette have complexity $\Theta(kn)$, $\Theta(kn)$, and $\Theta(n\alpha(k)+k^2)$, respectively.*

2.2 Lower bounds for a moving perspective point.

Now that we have tight bounds for a fixed viewpoint, we now consider the number of changes for a moving viewpoint. First, we will give a generic construction for lower bounds. Suppose that σ is the maximum complexity from a static perspective point p. There exists a small ball B around p such that for any point q inside B, the complexity of the structure from q has complexity σ. For linear motion, we can put k line segments inside B so that when we move the viewpoint p linearly, the shadow of those k line segments sweep over the silhouette structure. This way, we create $k\sigma$ changes. For algebraic motion, we take the classical example of a quadratic curve intersecting a convex n-gon n times. Then, in this way, we can create $n\sigma$ changes for algebraic motion. Combining this approach with the maximum possible complexity for a static point, we now have the following:

Theorem 2.2. *For linear motion, the silhouette arrangement, silhouette map, uo-silhouette can change $\Omega(k^2n)$, $\Omega(k^2n)$, $\Omega(kn\alpha(k)+k^3)$ times, respectively. For algebraic motion, the lower bounds are $\Omega(kn^2)$, $\Omega(kn^2)$, and $\Omega(n^2\alpha(k)+k^2n)$, respectively.*

3 Upper bounds

In the paper of de Berg et al. [4], they bound the number of different views to a scene of k convex objects. They derived an upper bound of $O(kn^2)$ on the number of surface patches that form a partition of the viewpoint space into cells with the same combinatorial structure of the visibility. Since a constant degree algebraic curve can intersect such a surface patch only a constant number of times, we can obtain the following upper bound by borrowing Theorem 5.2 of [4].

Theorem 3.1. *For a scene that consists of k convex polyhedra with total complexity n, there can be $O(kn^2)$ combinatorial changes to the silhouette-arrangement when the perspective point moves along a constant degree algebraic curve.*

However, when the point moves linearly, we may obtain better bounds on the number of changes of the silhouette arrangement.

Theorem 3.2. *For a scene that consists of k convex polyhedra in general position with total complexity n, there can be $O(k^2n)$ combinatorial changes to the silhouette-arrangement when the perspective point moves linearly. This bound also applies to the number of changes in the silhouette-map.*

For a convex polyhedron P, a line ℓ is said to be *tangent* to P if ℓ intersects P only on the boundary of P. When ℓ intersects P at exactly one point, it is called *strictly tangent* to P. The following simple fact is useful.

Fact 3.3. *A line ℓ is strictly tangent to P if and only if ℓ is strictly tangent to $P\cap\gamma$ for any plane γ that contains ℓ. In addition, this is the case if and only if there* ***exists*** *a plane γ' that contains ℓ so that ℓ is strictly tangent to $P\cap\gamma'$.*

The key step to prove Theorem 3.2 is to bound the number of lines that touch a given line and three convex polytopes. We will show that the number of these lines is linear in the total complexity of those three convex polytopes. We regard this lemma as having independent interest.

Lemma 3.4. *For any given line ℓ and three convex polyhedra P_1, P_2, and P_3 in general position, the number of lines that touch ℓ and are tangent to P_1, P_2, and P_3 is $O(|P_1|+|P_2|+|P_3|)$.*

In what follows, without loss of generality, we assume that the line ℓ is the z-axis. Consider the family of planes that pass through the z-axis. We parameterize them according to the angles they make with the x-axis: $\Gamma=\{\gamma(\theta):0\le\theta<\pi\}$. For a convex polyhedron P, denote by $P(\theta)$ the intersection between P and $\gamma(\theta)$. Clearly, $P(\theta)$, if not empty, is a convex polygon lying on $\gamma(\theta)$. For two convex polyhedra P,Q and any θ, $0\le\theta<\pi$ where $P(\theta)$ and $Q(\theta)$ are not empty, let us define the slope function $\phi_{P,Q}(\theta)$ (or $\phi(\theta)$ if P,Q are clear from the context) to be the slope of the lower outer bi-tangent between $P(\theta)$ and $Q(\theta)$. Let $\phi_{P,Q}$ denote the graph of the function $\phi_{P,Q}(\theta)$ as θ varies.

First, we observe that:

Lemma 3.5. *The graph of the function $\phi_{P,Q}$ consists of $O(|P|+|Q|)$ arcs, each of which is a constant degree rational function of $\tan(\theta)$.*

Proof: For any particular θ, the cross intersection $P(\theta)$ is a convex polygon. A vertex of this polygon is either a vertex v of P or $e\cap\gamma(\theta)$ for an edge e of P. For all the vertices created by the same edge, we think of them as a single vertex moving on a low degree rational curve as the plane rotates. If a bi-tangent is defined by the same pair of vertices, then its slope is just a rational function in terms of $\tan\theta$. When can a breakpoint be created? There are two possibilities: first, when either a previous vertex is deleted or a new one is created; second, when three vertices, two from one polygon and one from the other, are collinear and the line that passes through them is a bi-tangent line. Clearly the first type of event can happen at most $O(|P| + |Q|)$ times as it is the number of times when the plane sweeps over the vertices of P or Q.

For the second type of event, suppose that for some θ, $p_1, p_2 \in P(\theta)$ and $q \in Q(\theta)$ are collinear and the line ℓ' determined by p_1p_2q is a bi-tangent line to $P(\theta), Q(\theta)$. First of all, p_1, p_2 must be adjacent vertices on $P(\theta)$ by convexity. Further, by the general position assumption, ℓ' is strictly tangent to $Q(\theta)$ at q, which implies that ℓ' is strictly tangent to Q according to Fact 3.3.

Consider the edges $e_1, e_2 \in P$ that correspond to p_1, p_2, i.e. $p_1 = e_1 \cap \gamma(\theta)$ and $p_2 = e_2 \cap \gamma(\theta)$. We know that e_1, e_2 must be on the same face, say f (Figure 2). Consider the plane B that supports f. Observe that $B \cap Q$ is again a convex polygon. Suppose that B intersects ℓ at point r. We claim that rq is strictly tangent to $B \cap Q$. This simply follows from the fact that ℓ' is indeed strictly tangent to Q and by Fact 3.3. From any point, we can draw at most two tangent lines to another convex polygon. This is to say that for any face f of P, there can be at most two such points q on Q such that for any $p_1, p_2 \in f \cap \gamma(\theta)$, q is collinear with p_1, p_2 as a bi-tangent line. Therefore, the second type of event can happen at most $O(|P| + |Q|)$ times.

This concludes the proof of the lemma. $\square$

Building on Lemma 3.5, we now can prove Lemma 3.4.

Proof of Lemma 3.4. For P_1, P_2, P_3, we plot two functions $\phi_1 = \phi_{P_1,P_2}$ and $\phi_2 = \phi_{P_1,P_3}$. By Lemma 3.5, ϕ_1 consists of $O(|P_1|+|P_2|)$ low-degree rational arcs and ϕ_2 consists of $O(|P_1| + |P_3|)$ such arcs. The graphs ϕ_1 and ϕ_2 can intersect at $O(|P_1|+|P_2|+|P_3|)$ points. For a line ℓ' that touches ℓ, consider the plane γ that is determined by ℓ and ℓ'. For ℓ' to be tangent to P_1, P_2, P_3, ℓ' must be a common tangent to $P_1 \cap \gamma, P_2 \cap \gamma$, and $P_3 \cap \gamma$. This is related to an intersection point between ϕ_1 and ϕ_2. Therefore, in total this is bounded by $O(|P_1| + |P_2| + |P_3|)$. Of course, there are different types of tangents. But this is not a problem as there are constantly many combinations (16 combinations to be precise). We have therefore proved Lemma 3.4. $\square$

Next, we proceed to prove Theorem 3.2.

Proof of Theorem 3.2. The silhouette arrangement changes combinatorially only when the edges that define a silhouette change, or when the silhouettes of two polytopes start or stop intersecting, or when three silhouette edges intersect at a single point. These changes correspond to the following three types of events.

1. The **first type** occurs when the viewpoint crosses a plane supporting a facet.
2. The **second type** occurs when the viewpoint crosses a plane determined by a vertex and an edge from a different polytope.
3. The **third type** occurs when there is a ray from the viewpoint that touches three different polytopes.

The first type of events are bounded by $O(n)$ as there are $O(n)$ facets in total. When this type of event happens, what happens to the silhouette is that an edge is replaced by two other edges, or two edges become collinear and are replaced by a single edge. Each such event causes at most $O(k)$ changes to the silhouette arrangement. Therefore, this kind of event causes $O(kn)$ changes.

For an event of type two, suppose the vertex and polytope involved are v and P, respectively. Consider the double cone C formed by the union of all lines passing through v and P. The event happens only when the viewpoint on ℓ crosses the boundary of C, which can happen at most twice. This implies that the second type of event can happen at most $O(kn)$ times. Once such event happens, it can cause $O(1)$ changes to the silhouette as it makes a vertex cross an edge in the silhouette.

The third type is the hard case, when there is a line from the viewpoint that goes through the boundary of three polytopes. By Lemma 3.4, we know that this is bounded by

$$\sum_{i,j,k} O(|P_i| + |P_j| + |P_k|) = O(k^2 n)\,.$$

This concludes the proof of the theorem. $\square$

Notice that we actually up-bound the number of changes of the silhouette arrangement, and therefore the silhouette map and uo-silhouette. For the silhouette arrangement and silhouette map, these upper bounds match the lower bounds of Theorem 2.2. Unfortunately, there still is a gap remaining between the lower and upper bounds for the uo-silhouette.

4 Terrain with k mountains

As another application of Lemma 3.5, we may bound the number of changes of the silhouette for our special

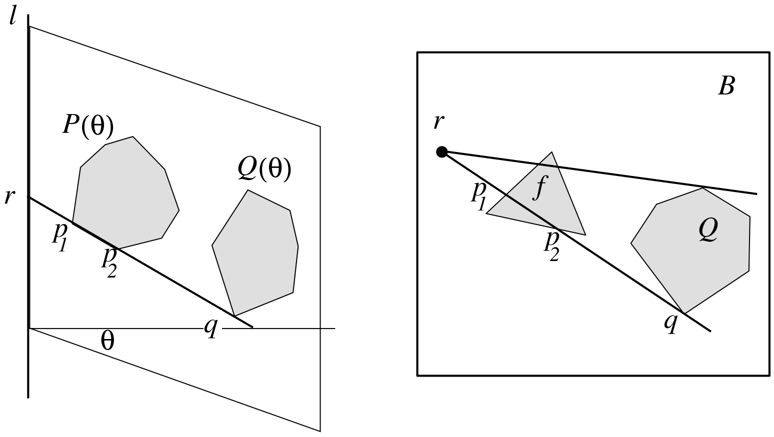

Figure 2. Combinatorial change to a bi-tangent

terrains, namely, terrains that consist of mountains. A convex object M is called a *mountain* if for any vertical line ℓ, $\ell \cap M = [0, a]$ for some $a \geq 0$. Or intuitively, a mountain is an upper-convex object whose base is on the xy-plane.

Cole and Sharir showed in [7] that for a viewpoint moving vertically in a terrain with n vertices, the visibility changes $O(n^2 2^{\alpha(n)})$ times, beating the naive bound of n^3. Having a small number of mountains does not help to reduce the lower bound there as the $\Omega(n^2)$ lower bound can be constructed by using *two* mountains. On the other hand, consider the silhouette arrangement from a vertically moving viewpoint. It is not difficult to construct an example similar to Figure 1 (i) to show that there can be $\Omega(k^2 n)$ changes to it, matching the upper bound given in Theorem 3.2. However, we may obtain a better bound on the number of combinatorial changes of the silhouette map (and uo-silhouette) for a viewpoint moving vertically.

Theorem 4.1. *The silhouette map and uo-silhouette changes $\Omega(kn)$ and $O(kn\beta_s(k))$ times for a point moving vertically in a terrain with k mountains and n vertices where s is a constant depending only on the degree of the motion.*

Proof: For the lower bound, consider the picture in which we have cylindrical mountain P with n sides. In front of the mountain, we have another k peaks(skinny tetrahedra). Then when the viewpoint moves vertically, each time it crosses a plane supporting a facet of P, it causes $\Omega(k)$ changes to the uo-silhouette and thus the silhouette map. In total, the changes are $\Omega(kn)$.(Figure 3)

For the upper bound, let us again consider the three types of event used previously. The first two cases are

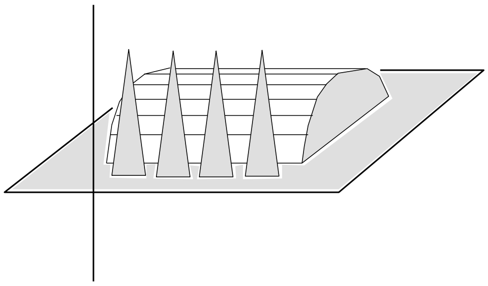

Figure 3. Lower bound construction for terrain.

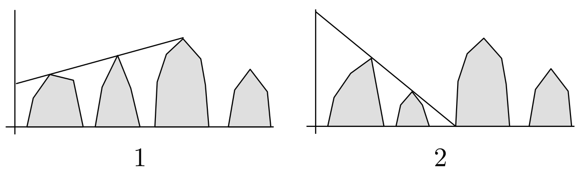

Figure 4. Lines that touch z-axis and three mountains.

bounded by $O(kn)$ as we have seen from the argument for general convex polytopes.

To bound the events of the third type, we now need to count the number of lines that touch the z-axis and three mountains and avoid all the other mountains. Again consider Γ, the family of the planes that pass through the z-axis. There are two cases where such a line can appear. One case is when there is a line that is tangent to three mountains from the same side and avoids all the other mountains; and the other case is when there is a line that touches the base of one mountain and is tangent to two other mountains. (Figure 4)

Like in [7], we define a partial ordering on the mountains so that $P \prec Q$ if there is a ray emanating from a point on the z-axis that intersects both P, Q and hits P first. Since the P_i's are mountains, this ordering is consistent. Among the three polyhedra involved in an event, we say the object that is furthest from the z-axis, or the greatest one under the partial ordering $\prec$, the *dominant* object.

Now, let us focus on one polytope, say P_1. We will bound the number of events in which P_1 is the dominant object. For the first case, such an event happens when there is a line passing through the z-axis and tangent to P_i, P_j, P_1 from above, where $P_i \prec P_j \prec P_1$. In addition, this line must not intersect any other object P for $P \prec P_1$.

For an object P_j that $P_j \prec P_1$, define a function $\phi_j(\theta)$ as the slope of the upper outer bi-tangent of $P_1(\theta)$ and $P_j(\theta)$, if both $P_1(\theta), P_j(\theta)$ are non-empty. Otherwise, $\phi_j(\theta)$ is undefined. Then, the above event happens when $\phi_i(\theta), \phi_j(\theta)$ have the same value and they are the highest among all the $\phi(\theta)$'s for $P \prec P_1$. In other words, we can charge each of these events to a unique break point on the upper envelope of the arrangement of the ϕ_j's. As we have shown in Lemma 3.5, the functions ϕ_j consist of $O(|P_1| + |P_j|)$ algebraic arcs. By the standard argument, the complexity is bounded by $\beta_s(k) \sum_j (|P_1| + |P_j|)$. Summing this up for all the polytopes, we have:

$$\sum_i \left(\beta_s(k) \sum_j (|P_i| + |P_j|)\right) = kn\beta_s(k)\,.$$

Similarly, we can bound the events in the second case by defining another function ξ_j as the slope of the line incident to the base of P_1 and tangent to P_j from above.

To summarize, the number of changes of the silhouette for a vertically moving point is $O(kn\beta_s(k))$ in a terrain with k mountains and n vertices. $\square$

5 Algorithms

We can apply the above combinatorial bounds for a linearly moving viewpoint, both in a general scene and for our terrains, to devise algorithms to maintain the different silhouette structures for a viewpoint that moves along an input line.

For the general scene that consists of k convex objects, imagine a plane γ that rotates around the z-axis from 0 to π. (Again, we assume that the line along which the viewpoint moves is the z-axis.) Consider the intersection of γ and the convex polytopes, which is a set of convex polygons. When γ rotates, those polygons deform and move. We wish to detect when a common tangent line to three convex polygons arises during the motion. This can be solved by tracking the bi-tangents of each pair of convex polygons. For each particular object, we maintain a list of tangents that touch it, sorted by their slopes. The task of detecting when an event happens reduces to detecting when the slopes of two bi-tangent lines become the same. This way, we can detect all the lines that touch three convex polyhedra and the z-axis. And it can be seen that the events that happen in our algorithm can be counted exactly as those counted by Theorem 3.2 and Lemma 3.5. The working space is $O(k^2 + n)$ and the processing time for each event is $O(\log n)$. We can apply similar algorithms to terrains. The only difference is that instead of maintaining the sorted list of all the tangents to an object, we maintain the lowest (or highest) one according to which side it is as described in the proof of Theorem 4.1. This can be done by the kinetic tournament data structure presented in [3].

Thus, we have that:

Theorem 5.1. *For k convex polyhedra with n vertices in total and a given line ℓ, in $O(k^2 n \log n)$ time, we can compute the partitioning of ℓ into intervals so that from the viewpoints in the same interval, the silhouette arrangement remains combinatorially invariant. When the polyhedra constitute a terrain with k mountains and ℓ is a vertical line, such a subdivision for silhouette maps can be computed in $O(kn\beta_s(k)\log^2 n)$ time.*

In [5], a method is presented to solve the problem of answering a ray shooting query for a ray emanating from a given line. Similarly, we can build such ray-shooting data structures once we compute the points at which a combinatorial change occurs in the silhouette map. Such ray shooting structures take roughly $O(k^2 n)$ and $O(kn)$ space and preprocessing time, for general polyhedral scenes and terrains, respectively. The details are omitted in this abstract.

6 Open Questions

The main open questions are:

1. The bound in Theorem 3.2 is not tight. The suspected tight bound is roughly $\Theta(k^3 + kn)$.

2. The corresponding bounds for algebraic motions are $\Omega(k^2 n + n^2)$ and $O(kn^2)$. Again, there is a gap of $O(k)$.

3. We have described an algorithm to compute all the changes for a viewpoint moving on a given line in about $O(k^2 n)$ time. Can we do it in an on-line manner, for example, in a kinetic data structures framework?

Acknowledgments

We thank Danny Halperin for suggesting the proof of Theorem 3.1 and T.M. Murali for helpful discussion. We also thank Tamara Munzner and an anonymous referee for pointing us to some relevant literature.

References

[1] B. Aronov and M. Sharir. The common exterior of convex polygons in the plane. *Comput. Geom. Theory Appl.* 8:139–149, 1997.

[2] G. Barequet, C. Duncan, M. Goodrich, S. Kumar, and M. Pop. Efficient prespective-accurate silhouette computation. *Proc. 15th Annu. ACM Sympos. Comput. Geom.*, pp. 417–418. 1999.

[3] J. Basch, L. J. Guibas, and J. Hershberger. Data structures for mobile data. *Proc. 8th ACM-SIAM Sympos. Discrete Algorithms*, pp. 747–756. 1997.

[4] M. de Berg, D. Halperin, M. Overmars, and M. van Kreveld. Sparse arrangements and the number of views of polyhedral scenes. *Internat. J. Comput. Geom. Appl.* 7:175–195, 1997.

[5] M. Bern, D. Dobkin, D. Eppstein, and R. Grossman. Visibility with a moving point of view. *Algorithmica* 11:360–378, 1994.

[6] W. Bouma and G. Vaněček Jr. Velocity-based collision detection. *Graphics Gems V*. Academic Press, 1995.

[7] R. Cole and M. Sharir. Visibility problems for polyhedral terrains. *J. Symbolic Comput.* 7:11–30, 1989.

[8] F. C. Crow. Shadow algorithms for computer graphics. *Comput. Graph.* 11(2):242–248, 1977.

[9] X. Gu, S. J. Gortler, H. Hoppe, L. McMillan, B. J. Brown, and A. D. Stone. Silhouette mapping. Tech. Rep. TR-1-99, Department of Computer Science, Harvard University, 1999.

[10] H. P. Lenhof and M. Smid. Maintaining the visibility map of spheres while moving the viewpoint on a circle at infinity. Report MPI-I-92-102, Max-Planck-Institut Inform., 1992.

[11] D. Luebke and C. Erikson. View-dependent simplification of arbitrary polygonal environments. *Proc. SIGGRAPH '97*, pp. 199–208. Computer Graphics Proceedings, Annual Conference Series, ACM Press, Aug. 1997.

[12] K. Mulmuley. Hidden surface removal with respect to a moving point. *Proc. 23rd Annu. ACM Sympos. Theory Comput.*, pp. 512–522. 1991.