Selecting Effective Occluders for Visibility Culling

Vladlen Koltun and Daniel Cohen-Or

Department of Computer Science, Tel-Aviv University, Tel-Aviv, Israel

Abstract

This paper deals with the problem of identifying effective occluders for visibility culling. The solid-angle metric is commonly used for measuring the potential significance of occluders from a single viewpoint. In this paper, we show that it does not extend properly to from-region occlusion calculations. We propose to measure the effectiveness of an occluder by means of the size of its umbra. We first present an analytic object-space algorithm to accurately compute this measure. We then define an approximation which reflects the effectiveness of an occluder, and introduce a hardware-assisted algorithm to rapidly compute it.

1. Introduction

Recently from-region visibility culling algorithms have received considerable attention [2, 3, 5, 7, 10, 11]. The advantage of computing the visibility from a region rather than from a point is that it allows to compute the visibility information offline. This considerably speeds up the online rendering of complex scenes. Most visibility techniques are based on selecting a set of significant convex occluders, and finding objects that are occluded by this set [2, 4, 5, 6, 12]. This allows to avoid the cost of computing the occlusion due to insignificant complex objects, such as furniture in architectural models or lampposts in urban scenes. Taking such objects into account usually does not increase the occlusion, while drastically increasing the time spent on the culling process. In other words, most of the occlusion is typically caused by a relatively small group of significant (effective) occluders.

The selection of the effective occluders has to be based on some automatic criterion that reflects their potential significance. The correctness of this criterion is an important issue when visibility culling is concerned. If the criterion gives incorrect results, one may fail to classify significant objects as occluders, thereby losing their contribution to the occlusion. Moreover, it may spend unnecessary amounts of time calculating occlusion due to small, insignificant objects that were incorrectly classified as occluders.

In the case of from-point visibility, the common criterion for measuring the significance of an occluder is the solid angle (or some approximation of it) subtended by the occluder at the viewpoint [4, 6, 12]. So far, no suitable criterion has been proposed for measuring the significance of occluders from a region. Currently, from-region visibility techniques estimate the occlusion potential of an object from a region by estimat-

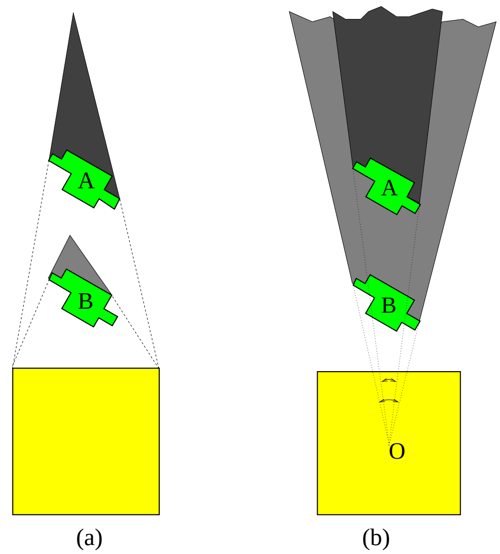

Figure 1: *The solid angle from a point in the viewcell is not a correct metric of the effectiveness of occluders. The solid angle of B from any point within the viewcell is larger than that of A (shown in (b)), while A is a better occluder (shown in (a)).*

ing the solid angle it subtends at one of the points within the region [2, 5, 7].

In the next section we show that such estimation gives correct results only when the object is larger than the viewing region. In other cases, this estimation might incorrectly classify significant occluders as insignificant and vice-versa. This is illustrated in Figure 1 where object A occludes a bigger part of the scene than object B does (the shadowed parts in (a)). Nevertheless, its solid angle from *any* point in the viewing region is smaller than that of B (the solid angle of an object from a point is proportional to the size of its umbra from that point, shown in (b)). In such cases, selecting the "good" occluders by the solid angle, erroneously treats B as a more significant occluder than A, resulting in decreased effectiveness of the culling.

The potential amount of occlusion caused by a given object is proportional to the volume of the shadow cast by the object (the *umbra* of the object), treating the viewcell as an area light-source [3]. The fact that the object's umbra is big means that a big part of the scene is occluded by the object from all points within the viewcell, implying that the object is a significant occluder. Thus, a suitable metric of the potential significance of an occluder from a region is the size of its umbra. In section 3 we describe an efficient algorithm for accurately computing this metric. It also functions correctly when dealing with degenerate viewing regions with zero volume, making it applicable for from-point occlusion culling as well.

In section 4 we introduce a criterion that estimates the potential from-region occlusion of a given object. We present a fast hardware assisted algorithm to compute this estimation. A fast and approximate estimation of the occlusion potential of an object is more effective for most applications. We conclude in section 5.

2. The solid angle metric

The solid angle measures the size of an object, as seen from a point in space. Its value is determined by the size $d\omega$ of the object's projection on the unit sphere centered around the view point (Figure 2). Informally, we can say that the solid angle grows with the visible size of the object, and diminishes with the object's distance.

All major occlusion culling techniques have used some approximation of the solid angle metric to select significant occluders. A popular approximation has been the area-angle metric, introduced by Coorg and Teller [4]. Let θ be the angle between the normal of the surface and the directional vector from the surface to the view-point (Figure 2). Let r be the distance to the view-point, and A the area of the surface. The area-angle of the surface from the view-point is

$$\frac{A * cos\theta}{r^2}.$$

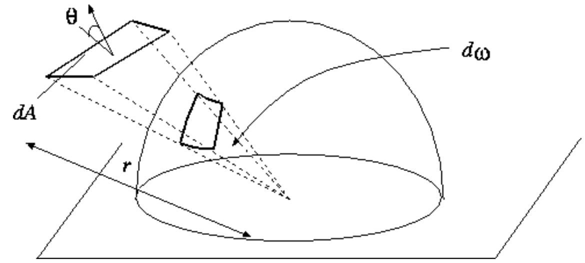

Figure 2: *The solid angle of the surface is proportional to the size $d\omega$ of its projection on the unit sphere.*

We need a new technique to generate a value that reflects the significance of occluders from a region, since the solid angle of an object from some point within the region is not necessarily correct to estimate this value. Note that while the solid angle of an object diminishes with the distance, the size of the object's umbra might get larger with distance. Consider, for instance, the case when the object is smaller than the viewing region, as shown in Figure 1. The object's umbra is finite, in contrast to the infinite from-point umbra. The solid angle metric and its approximations do not take into account the size and shape of the region, and may cause incorrect results in many practical situations.

3. The accurate method

In this section we propose an object-space algorithm that correctly measures the potential occlusion contribution of an object in both the from-region and from-point cases. The idea is to directly compute the volume of the shadow cast by the object on the scene, treating the viewcell as a volumetric light source. Objects casting a bigger shadow have more potential to occlude a bigger portion of the scene and are thus more effective occluders.

We start by discussing the method in 2D scenes. This is practically important for various visibility techniques of 2.5D scenes [7, 11]. In 2D, the object's shadow (umbra) is bounded on the sides by the two supporting lines of the object and the viewcell (Figure 3). The umbra is also bounded by the object itself (in the front), and either by the intersection point between the two supporting lines, or by the bounding rectangle of the scene in the back. Figure 3 illustrates these two cases. The reason for bounding the umbra by the bounding rectangle of the scene, is that we are only interested in the size of the shadow within the scene, where it can potentially occlude objects.

We compute the area of the polygonal representation of the umbra. This is done by finding the vertices A and B on which the supporting lines lie. If necessary, the lines are intersected with the scene's bounding rectangle. Once the polygon that represents the umbra is computed, its area is computed by the simple and robust algorithm given in [9].

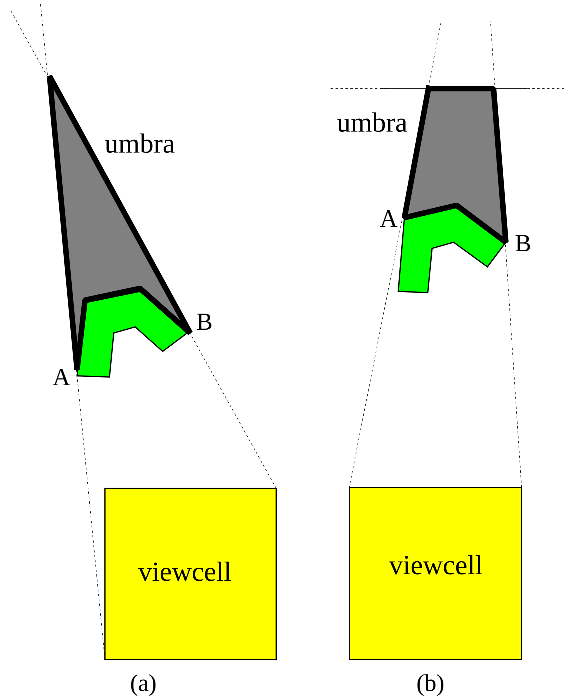

Figure 3: *The umbra in 2D is always bounded by two supporting lines. It is also bounded either by their intersection point (in (a)), or by the scene's boundary (in (b)).*

The extension of the above to 3D is much more involved since the from-region umbra of a non-convex polyhedron is not necessarily connected. Moreover, the problem of finding the exact umbra of a non-convex 3D object of arbitrary topology is asymptotically as hard as computing the exact visibility from a region. In the following we assume a convex viewcell and convex occluders. Most occlusion culling algorithms in 3D operate only on convex occluders [2, 3, 4, 6]. This is justified, among others, by the fact that occluders can be approximated by a set of simple convex objects [1].

Similarly to the 2D case, we compute the polyhedral representation of the occluder's umbra, and then compute its volume. Explicitly computing the polyhedral representation of the shadow of an object caused by a distributed (volumetric) light source is in fact the same calculation as in shadowing algorithms, for which an efficient algorithm is presented in [8]. After computing this shadow, its volume can be computed by the robust algorithm from [9], which is an extension of the algorithm used to compute the area of a 2D polygon.

4. A fast approximate metric in 3D

The algorithm presented in the previous section is an object-space analytic method, which produces an accurate measure. Here, we present a fast hardware assisted algorithm that computes an approximate measure.

We place two projection planes behind the occluder: one immediately behind it and the other parallel to and behind the first one at a small constant distance δ (Figure 4). To obtain our measure, we project the occluder from the viewcell onto both projection planes, and compute the area of the two projections. Similarly to [5], we define the projection of an object as the intersection of the object's umbra with the projection plane. A nice property of projection planes is that to compute the projection of an object one does not have to compute the object's umbra. Formally, the measure M is defined as

$$M = \frac{A2}{A1},$$

where $A1$ is the area of the front(closer) projection, and $A2$ is the area of the rear projection.

This ratio yields an estimation of the growth rate of the umbra, which reflects the umbra's size. It is larger than 1.0 for occluders whose umbra grows with distance. If we enlarge an occluder, this ratio becomes larger. If we take an object that is *bigger* than the viewcell and move it away from the viewcell, this ratio becomes smaller. Alternatively, if we take an object *smaller* than the viewcell and move it away, this ratio becomes larger, as does the occluder's umbra. Also, if we move any object away from the viewcell towards infinity, this ratio converges to 1.0. In the degenerate case where the volume of the viewcell is zero, this ratio is proportional to the area-angle of the occluder.

Note that the distance δ between the two projection planes effectively screens out very small and ineffective occluders. For such occluders, whose umbra is very small, the intersection of the umbra with the second projection plane is null (Figure 4 (d)). An important property of the presented method that distinguishes it from the area-angle metric is that it gives an estimation of occlusion for an object, rather than just a single polygon. Note that the effective occlusion of an object from a region can not be expressed as the sum of the occlusion of its parts.

The projection of a convex object onto a plane is a convex polygon. It can be computed analytically by computing the projections from each corner of the viewcell and then intersecting them on a 2D plane [8]. However, Durand et.al. [5] showed that the projection can be computed approximately in image-space by a hardware assisted method that utilizes the stencil buffer. The idea is to maintain a bitmap on the projection plane, and render the object onto this bitmap from each corner of the viewcell. The stencil buffer is used to identify which pixels were rendered upon from all corners of the viewcell. These are the pixels that belong to the projection of the object from the viewcell. The number of these pixels gives the approximate projection area.

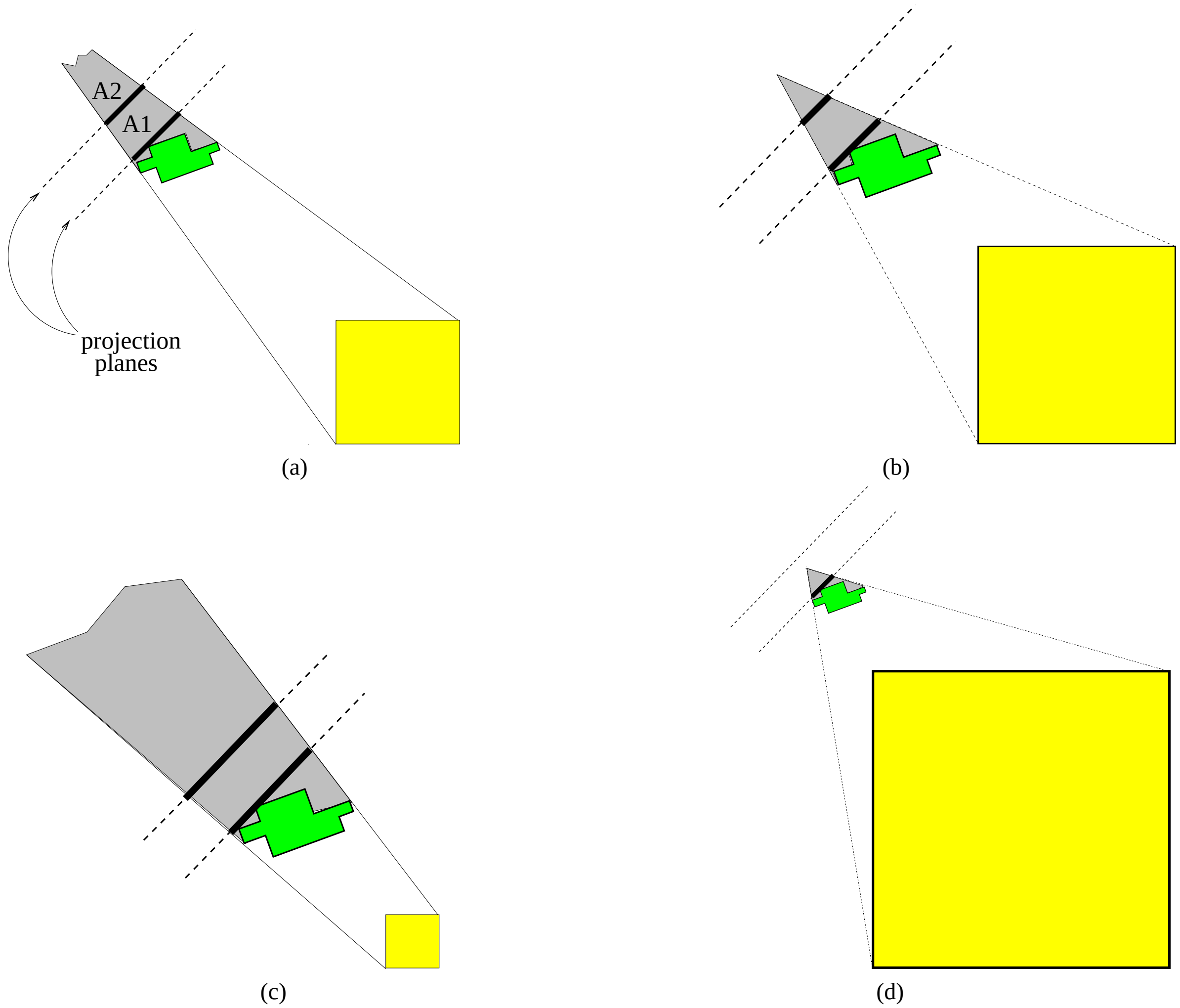

Figure 4: *A 2D illustration of the approximate metric. In (a), $0 < A2 < A1$, thus $0 < M < 1$. In (b), the object is closer and its umbra is smaller. Consequently, M is smaller than in (a). In (c), $A2 > A1$ and $M > 1$. In (d), $A2 = 0$ and thus $M = 0$.*

5. Discussion and Conclusion

In [7] we have described the process of constructing *virtual occluders*, which encapsulate the cumulative occlusion of a group of objects. To construct virtual occluders, a set of *seed* objects is selected. Then, a group of objects around each seed is detected, such that the occlusion caused by the group can be captured by a single convex virtual occluder. In context of the construction of virtual occluders, it is important to select the significant occluders as seed objects. Moreover, ordering the occluders by their effectiveness speeds up the occlusion calculation [7]. Of course, the described measure is applicable whenever one needs to select or order a subset of available objects, based on their potential occlusion.

We have described a novel criterion that measures the effectiveness of a given occluder, and an algorithm to compute it. We have shown that current methods may produce incorrect results when the visibility is computed from a region. The proposed criterion and algorithms are important, as they help to quickly identify the significant occluders in a scene. They can be used as a part of, and improve the effectiveness of many various visibility culling techniques.

6. Acknowledgements

This work was supported by a grant from the Israeli Ministry of Science and by a grant from the Israeli Academy of Sciences (center of excellence).

References

1. Carlos Andujar, Carlos Saona-Vazquez, and Isable Navazo. Lod visibility culling and occluder synthesis. *Computer Aided Design*, 1999. 3
2. Carlos Andujar, Carlos Saona-Vazquez, Isable Navazo, and Pere Brunet. Integrating occlusion culling and levels of details through hardly-visible sets. *Computer Graphics Forum*, 19(3), 2000. 1, 2, 3
3. Daniel Cohen-Or, Gadi Fibich, Dan Halperin, and Eyal Zadicario. Conservative visibility and strong occlusion for viewspace partitionin of densely occluded scenes. *Computer Graphics Forum*, 17(3):243–254, 1998. ISSN 1067-7055. 1, 2, 3
4. Satyan Coorg and Seth Teller. Real-time occlusion culling for models with large occluders. *1997 Symposium on Interactive 3D Graphics*, pages 83–90, April 1997. ISBN 0-89791-884-3. 1, 2, 3
5. Frédo Durand, George Drettakis, Joëlle Thollot, and Claude Puech. Conservative visibility preprocessing using extended projections. *To appear in the proceedings of SIGGRAPH 2000*, 2000. 1, 2, 3
6. T. Hudson, D. Manocha, J. Cohen, M. Lin, K. Hoff, and H. Zhang. Accelerated occlusion culling using shadow frusta. In *Proceedings of the 13th International Annual Symposium on Computational Geometry (SCG-97)*, pages 1–10, New York, June4–6 1997. ACM Press. 1, 3
7. Vladlen Koltun, Yiorgos Chrysanthou, and Daniel Cohen-Or. Virtual occluders: An efficient intermediate pvs representation. In *Proceeding of Eurographics Workshop on Rendering*, June 2000. 1, 2, 4
8. T. Nishita and E. Nakamae. Continuous tone representation of three-dimensional objects taking account of shadows and interreflection. *Computer Graphics (Proceedings of SIGGRAPH 85)*, 19(3):23–30. 3
9. Joseph O'Rourke. *Computational Geometry in C (2nd Ed.)*. Cambridge University Press, 1998. 2, 3
10. Gernot Schaufler, Xavier Decoret, Julie Dorsey, and Francois Sillion. Conservative volumetric visibility with occluder fusion. *To appear in the proceedings of SIGGRAPH 2000*, 2000. 1
11. Peter Wonka, Michael Wimmer, and Dieter Schmalstieg. Visibility preprocessing with occluder fusion for urban walkthroughs. In *Proceeding of Eurographics Workshop on Rendering*, June 2000. 1, 2
12. Hansong Zhang, Dinesh Manocha, Thomas Hudson, and Kenneth E. Hof III. Visibility culling using hierarchical occlusion maps. *Proceedings of SIGGRAPH 97*, pages 77–88, August 1997. ISBN 0-89791-896-7. Held in Los Angeles, California. 1