A Framework for n-dimension Visibility Calculation

Lilian Aveneau, Sylvain Charneau, Laurent Fuchs, Frédéric Mora

A Framework for n-dimensional Visibility Computations

L. Aveneau, **S. Charneau**, **L. Fuchs** and **F. Mora**

University of Poitiers,
XLIM/SIC, CNRS, France
lilian.aveneau@xlim.fr [presenter, corresponding]

Abstract

Visibility computation is a fundamental task in computer graphics, as in many other scientific domains. While it is well understood in two dimensions, this does not remain true in high dimensional spaces. Using Grassmann Algebra, we propose a framework for solving visibility problems in any n-dimensional spaces, for $n \geq 2$.

Our presentation recalls the problem statement, in two and three dimensions. Then, we formalize the space of n-dimensional lines. Finally, we show how this leads to a global framework for visibility computations, giving an example of use with exact soft shadows.

1 Problem statement

1.1 About Visibility

Visibility is used in many cases in computer graphics. The simplest one is the visibility between 2 given points, with a Boolean-valued function answering the question: "Are the two points visible each other?". A classical solution uses a visibility ray, which works in any dimension where such a ray approach is available [1].

Some applications use or need a global visibility information on continuous set of points (*eg.* two polytopes as their convex hulls). For examples, it can serve to optimize the previous predicate, or to compute soft shadows. In such cases, a ray approach is clearly unusable alone, due to sampling problems between the two point sets. We need to study all the *potential* discontinuities in visibility changes using the set of occluders between the two sets: If a discontinuity occurs, obviously it happens at an occluder boundary. More precisely, since lines are subspaces of dimension 2, then discontinuities in visibility appear at occluder ridges – or (n-2)-face, or subfacet (vertexes in 2d, edges in 3d).

So, the visibility problem is expressed in an *occlusion* space. How to represent such a space, and then to make calculation? Intuitively, since visibility is a line problem, the occlusion space is the space of n-dimensional lines.

1.2 The dimension problem

To make computations in line space seems trivial, thinking in dimension 2. Using a dual representation of the usual point space, the problem stays in dimension 2 (in fact a projective plane). Then the polytope ridges are the polygon vertexes. So the study of discontinuities consists to lines and points studied in a plane, as with the visibility complex [2].

In dimension 3, this does not remain true: The line space dimension is greater than 3, and so similarities with 2D approaches are very hard to kept. Then, many authors propose other approaches to by-pass this difficulty. A recent one uses the Plücker space where sets of lines become polytopes, and the occlusions are computed by subtracting polytopes of occluded lines to the set of visible ones [3, 4]. According to [5, 6], the visibility representation is made with BSP trees, where an internal node contains the ridge responsible for an occlusion, *i.e.* an occluder edge.

While this approach works correctly, it raises many new problems with respect to a 2D one. More generally, defining algorithms based on a particular dimension does not allow to reuse or extend previous works. A n-dimensional solution, enclosing and generalizing all the previous ones, should help to this kind of considerations. Moreover, Geometric Algebras provide the level of abstraction to achieve successfully such a generalization [7].

1.3 Toward a global visibility framework

We propose here a global visibility framework. It allows to think about visibility problems and solutions with a single approach, where a solution stays correct whatever the dimension of point space is. It is based on a space of lines defined in any dimension, using the exterior product of the Grassmann Algebra [7]. While this is a classical definition of lines in mathematics, this approach is uncommon in computer graphics. Using this new formalism, we prove a major theorem about the set of lines stabbing two convex polytopes. Next, we propose a generalization of Mora's work, into a n-dimensional visibility framework. As a result, we use it to compute very high quality soft shadows.

2 Line spaces

To compute visibility between object, we denote $\mathfrak{G}_n$ the n-dimension geometrical space of the geometric objects. It is embedded into the projective space $\mathbb{P}^n$. As $\mathbb{P}^n$ is built from $\mathbb{R}^{n+1}$, linear manifolds of $\mathbb{P}^n$ can be represented by elements of $\bigwedge(\mathbb{R}^{n+1})$.

2.1 Plücker line

Classical Plücker lines are defined in a 3D projective space. Using homogeneous notation, a point P is denoted by 4 coordinates using the vector: $p = (x, y, z, w)^T$. Without loss of generality, a Plücker line between two homogeneous points A and B is defined by using the Grassmann exterior product, as $\Pi_{AB} = a \wedge b$, resulting to the well-known Plücker coordinates. It is interesting to notice that this classical definition of Plücker lines works with any geometric space $\mathfrak{G}_n$ for $n \geq 2$. Indeed, a line is always the set of points that are linearly dependant to two distinct points. So, we formulate the following definition.

Definition 1 *A n-dimensional line, passing through two projective points A and B of $\mathfrak{G}_n$ with respective 1-vector coordinates a and b, is represented with the exterior product $a \wedge b$.*

2.2 From line to line space

The elements of $\bigwedge^k(\mathbb{R}^{n+1})$, for $k \leq n$, are homogeneous: If $\mathcal{K} \in \bigwedge^k(\mathbb{R}^{n+1})$ represents a subspace of $\mathbb{R}^{n+1}$, then $\mathcal{K}' = \lambda K$, for $\lambda \in \mathbb{R}^*$, represents the same subspace. Hence, $\mathbb{P}^{\binom{n+1}{k}-1} = \mathbb{P}(\bigwedge^k(\mathbb{R}^{n+1}))$ is the space of the 1-subspaces of $\bigwedge^k(\mathbb{R}^{n+1})$. Each point of $\mathbb{P}^{\binom{n+1}{k}-1}$ represents a unique linear manifold of $\mathfrak{G}_n$. This leads to the following definition of the line space $\mathfrak{D}_n$ of $\mathfrak{G}_n$.

Definition 2 *The space of lines of $\mathfrak{G}_n$, denoted by $\mathfrak{D}_n$, is the projective space $\mathbb{P}(\bigwedge^2(\mathbb{R}^{n+1}))$.*

From this definition, the line space is a projective space of dimension $\binom{n+1}{2}$. In dimension 2, the line space is of dimension $\binom{3}{2} = 3$, while in dimension 3, the classical Plücker space is of dimension 6. This is directly related to the dimension problem, as presented in section 1.2.
From previous works on visibility computations, we know that 3-dimensional Plücker lines do not fill all the space line $\mathfrak{D}_3$. They are all located on the Grassmannian $G^{\mathbb{R}}(2, 4)$, the set of all

2-subspaces of $\mathbb{R}^4$. It is also the set of all decomposable 2-vectors. In dimension n the set of real Plücker lines are on the Grassmannian $G^{\mathbb{R}}(2, n+1)$.
In our visibility framework, we need a mechanism to classify lines with respect to occluders. It is used to characterize incidence and relative orientation of lines to projective linear manifolds. This classification is directly based on properties of the exterior product (antisymmetric and incidence properties). Then, two linear manifolds represented by the multivectors A and B are incident if and only if $A \wedge B = 0$.
Thus Grassmann Algebra provides a formal justification to look at the occluder ridges for a global visibility computation.

2.3 The minimality theorem

This formalization of the visibility problem with the Grassmann Algebra leads to a better understanding and has allowed us to demonstrate the following theorem in dimension n.

Theorem 1 *The set of lines stabbing two convex polytopes A and B in $\mathfrak{G}_n$ is the intersection of the Grassmannian with the convex hull of the lines going through one vertex of A and one vertex of B iff the support planes of A and B do not intersect in A or B.*

This theorem is fundamental for our visibility framework. It is particularly used to optimize computations in any dimension.

3 Visibility framework

3.1 Framework overview

The Plücker line space and the minimality theorem allow us to propose a framework dedicated to global visibility computation in n dimensions. Notice that all the geometry (occluders and polytopes) are convex polytopes. We present this framework, emphasizing the visibility computations between couple of polytopes. The overview is:

1. Calculation of the set of visible lines (minimality theorem).
2. Subtract all occluders, one after one (using classification).
3. Grassmannian intersection, for maintaining a valid BSP structure.

3.2 Visibility representation

Visibility is stored in a BSP, where each inner node contains a ridge, and leaves are Boolean (fully visible or fully invisible). This is valid in any dimension [8]. Notice that while such a representation is already used by Mora in [5], it is here generalized and extended to dimension n.

3.3 Use case: Soft shadows

We conclude our presentation with a use-case example: Rendering with soft shadows [6]. Visibility computation can be seen as a black box that can be plugged on any rendering engine. The figure 3.3 provides an illustration and shows the contribution of our framework compared to discretized approaches.

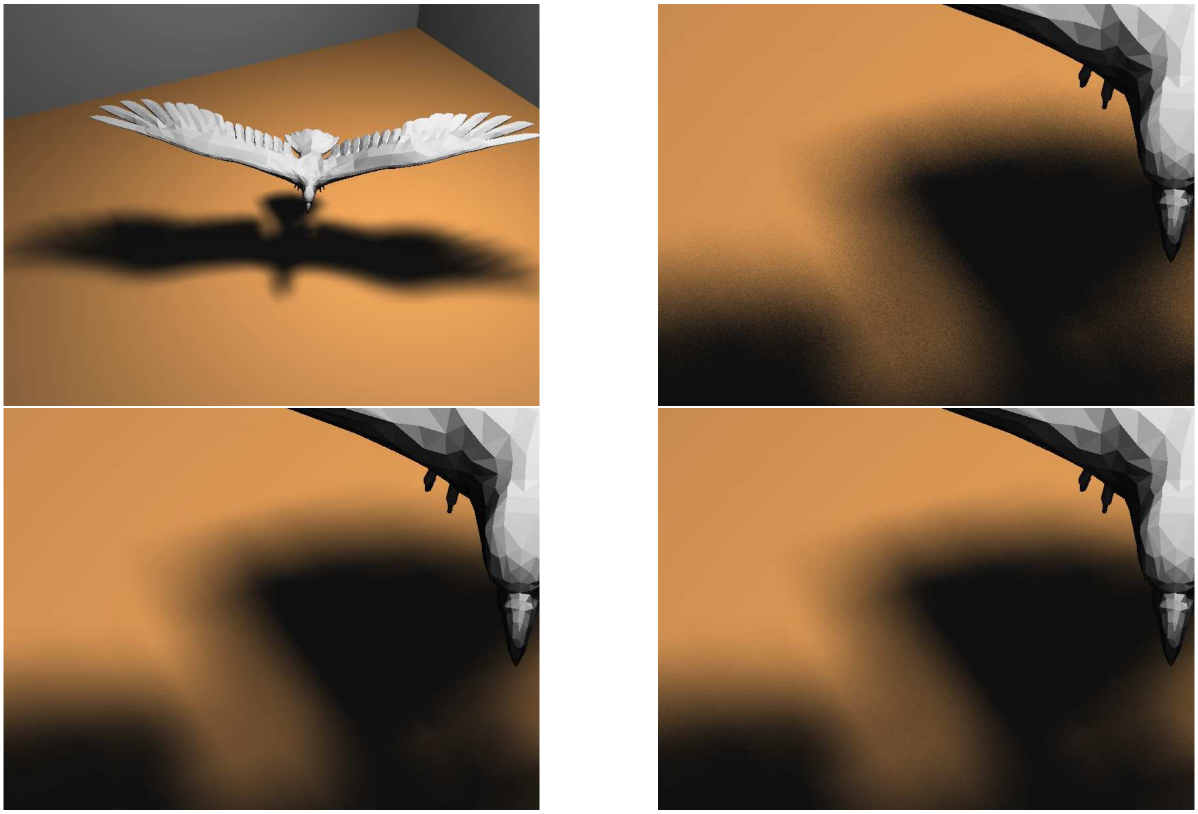

Figure 1: Top left image illustrates a final result on soft shadows computation; Bottom left proposes a zoom on shadows, accurately computed with our framework; Right images propose the same zoomed view, but using a classical sampling approach, with 32 samples per pixels on top image and 128 on bottom one.

References

[1] T. Möller, and B. Trumbore, Fast, Minimum Storage Ray-triangle Intersection. In *Journal of graphics tools*, volume 2(1), pages 21–28, 1997.

[2] M. Pocchiola and G. Vegter, The visibility complex. In *Int. J. Comput. Geometry Appl.*, volume 6(3), pages 279-308, 1996.

[3] S. Nirenstein, E. H. Blake, and J. E. Gain, Exact From-Region Visibility Culling. In *Rendering Techniques*, pages 191–202, 2002.

[4] D. Haumont, O. Makinen, and S. Nirenstein, A Low Dimensional Framework for Exact Polygon-to-Polygon Occlusion Queries. In *Rendering Techniques*, pages 211–222, 2005.

[5] F. Mora, L. Aveneau, and M. Mériaux, Coherent and exact polygon-to-polygon visibility. In *WSCG'05*, pages 87–94, 2005.

[6] F. Mora, and L. Aveneau, Fast and Exact Direct Illumination. In *CGI'05*, pages 87–94, 2005.

[7] L. Dorst, D. Fontijne, and S. Mann. *Geometric Algebra for Computer Science*, Morgan Kaufmann Publishers, 2007.

[8] B. Naylor, J. Amanatides, and W. Thibault, Merging BSP trees yields polyhedral set operations. In *SIGGRAPH '90*, pages 115–124, 1990.