Fast Scene Voxelization Revisited

Elmar Eisemann, Xavier Décoret

Fast Scene Voxelization and Applications

Elmar Eisemann*
ARTIS-GRAVIR/IMAG-INRIA†/ ENS‡

Xavier Décoret§
ARTIS-GRAVIR/IMAG-INRIA

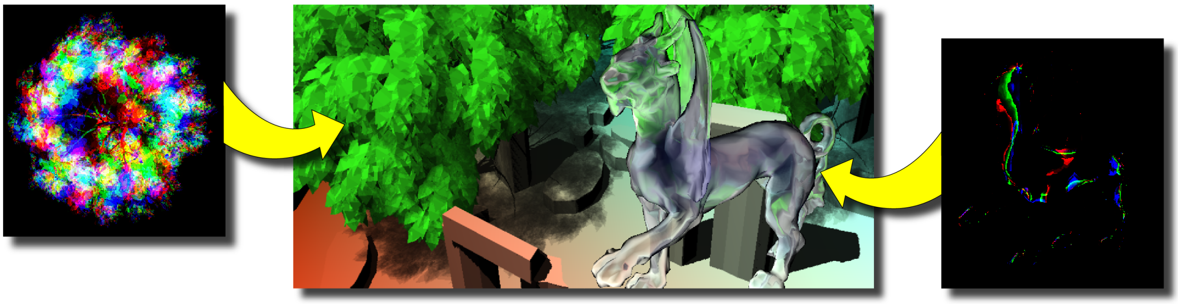

Figure 1: Slicemaps (small figures), can be used for several applications and a variety of effects. Distance is encoded in color, what leads to an efficient voxel representation of the scene. Transmittance shadow maps use this information to attenuate light passing through semi-transparent surfaces (shadows of the trees). We also show that the obtained volume estimation can be used for refraction (lion).

Abstract

This sketch paper presents an overview of ”Fast Scene Voxelization and Applications” published at the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games. It introduces *slicemaps* that correspond to a GPU friendly voxel representation of a scene. This voxelization is done at run-time in the order of milliseconds, even for complex and dynamic scenes containing more than 1M polygons. Creation and storage is performed on the graphics card avoiding unnecessary data transfer. Regular but also deformed grids are possible, in particular to better fit the scene geometry. Several applications are demonstrated: shadow calculation, refraction simulation and shadow volume culling/clamping.

1 Method

Voxelizing a scene means testing geometry for intersection with an englobing grid of cells. Our approach accomplishes this task efficiently with graphics hardware. Instead of interpreting RGBA as 4 unsigned byte values, we will consider it as a 32 bit vector, which itself corresponds to a 32 cell column in the voxelization. An image is thus nothing but a voxel grid, in which each cell is represented by one bit. We call this representation *slicemap*. The implementation is simple and allows the voxelization of millions of polygons in real-time. In the future, when 32 bit logical operations become available, 4 MRT[1] $\times 4$ channels $\times$ 32 bits $=$ 512 slices are obtained in a single pass. Our approach shows several similarities to the work by Dong et al.[Dong et al. 2004], who also store depth in bits. Some major differences exist. First we do not fuse several views (which reduces cracks, but limits resolution). Second voxel grids usually contain axis-aligned cubes, whereas our approach offers several ways to deform the grid (e.g. improving precision by local refinement based on two depth masks containing the furthest and closest fragment for each pixel). We also present a fast technique to obtain a filled voxelization without iterative flood fill operations and present all-new applications.

2 Applications

Transmittance Shadow Maps: Illumination is attenuated based on the number of blockers traversed by each light ray before impinging on the surface. This calculation can be performed quickly due to the way we represent voxels and leads to convincing effects for complicated objects like trees or hair at low cost.

Shadow Volume Culling and Clamping: Slicemaps offer the possibility to cull and clamp shadow volumes [Lloyd et al. 2004]. The aforementioned possibility to deform the voxel grid is key here.

Refraction: Given a watertight object, we obtain a filled voxel representation avoiding flood fill. We estimate the object's volume and use it for refraction and scattering effects.

3 Conclusion

The paper presents a GPU adapted voxel representation, applicable to any rasterizable object providing depth. The presented applications are not straightforward and present contributions on their own. Local refinement combines the advantage of exact shape information and approximated volume and results in a good overall estimation. Error analysis and conservativity remain future research.

References

Dong, Z., Chen, W., Bao, H., Zhang, H., and Peng, Q. 2004. Real-time voxelization for complex models. In *Proceedings of the 12th Pacific Conference on Computer Graphics and Applications*, 43–50.

Lloyd, B., Wendt, J., Govindaraju, N. K., and Manocha, D. 2004. Cc shadow volumes. In *Proceedings of the 2nd EG Symposium on Rendering*, Eurographics Association, Springer Computer Science, Eurographics.

*e-mail: Elmar.Eisemann@inrialpes.fr
†ARTIS is a team of the GRAVIR/IMAG laboratory, a joint effort of CNRS, INRIA, INPG and UJF
‡École Normale Supérieure Paris
§e-mail:Xavier.Decoret@inrialpes.fr
[1] Multiple Render Targets