Voxel DAGs and Multiresolution Hierarchies: From Large-Scale Scenes to Pre-computed Shadows

Ulf Assarsson[1], Markus Billeter[2], Dan Dolonius[1], Elmar Eisemann[2]
Alberto Jaspe[3], Leonardo Scandolo[2], Erik Sintorn[1]

[1]Chalmers University of Technology, Sweden
[2]Delft University of Technology, The Netherlands
[3] CRS4 Visual Computing, Italy

Abstract

In this tutorial, we discuss voxel DAGs and multiresolution hierarchies, which are representations that can encode large volumes of data very efficiently. Despite a significant compression ration, an advantage of these structures is that their content can be efficiently accessed in real-time. This property enables various applications. We begin the tutorial by introducing the concepts of sparsity and of coherency in voxel structures, and explain how a directed acyclic graph (DAG) can be used to represent voxel geometry in a form that exploits both aspects, while remaining usable in its compressed from for e.g. ray casting. In this context, we also discuss extensions that cover the time domain or consider an advanced encoding strategies exploiting symmetries and entropy. We then move on to voxel attributes, such as colors, and explain how to integrate such information with the voxel DAGs. We will provide implementation details and present methods for efficiently constructing the DAGs and also cover how to efficiently access the data structures with e.g. GPU-based ray tracers. The course will be rounded of with a segment on applications. We highlight a few examples and show their results. Pre-computed shadows are a special application, which will be covered in detail. In this context, we also explain how some of previous ideas contribute to multi-resolution hierarchies, which gives an outlook on the potential generality of the presented solutions.

CCS Concepts
•***Computing methodologies*** $\rightarrow$ *Ray tracing; Visibility; Volumetric Models;*

Course Schedule

Opening	Elmar Eisemann	5m
Introduction	Ulf Assarsson & Elmar Eisemann	25m
Advanced DAG Encodings	Alberto Jaspe	20m
Attribute Compression	Dan Dolonius & Ulf Assarsson	30m
DAG Construction	Erik Sintorn	20m
DAG Ray-tracing	Markus Billeter	20m
Applications/Demos	(all)	10m
Compressed Shadow Volumes	Erik Sintorn	20m
Compressed Shadow Maps	Leonardo Scandolo	20m
Conclusion and Q&A	(all)	10m

1. Overview

We begin the course by a brief welcome, introducing the schedule and briefly outlining the content and purpose of the course. The course consists of seven presentations, plus a short demo session where we highlight applications and results. Each presentation is further outlined in Section 2. Presentations are either 20 or 30 minutes. Finally, we conclude the course with a brief Q&A. The table below lists the individual parts with their main speakers. The opening, break and conclusion are shown in italic.

We expect that all participants of the Eurographics conference will be able to follow this course, as it will introduce all necessary concepts.

2. Outline

The course consists of three parts; the first part introduces the methods, the second details the implementations, while the third part revolves around applications with a focus on precomputed shadows.

2.1. Introduction

Voxel DAGs and Multiresolution Hierarchies We begin by introducing binary voxel grids as a representation for geometry and discuss two of the important properties: sparsity and coherency. On the GPU for large-scale volumes sparse voxel octrees (SVOs) have proven very useful to enable a sparse encoding [GMIG08,CNLE09, CNSE10,LK10]. SVOs achieve their compression by avoiding storing information in empty space. Coherency can be exploited to further compress SVOs by finding similarities and compressing these. In practice, identical subtrees in an SVO are identified, and merged by pointing to a single common instance. This transforms the SVO

DOI: 10.2312/egt.20181028

Figure 1: *Voxel DAGs and Multiresolution Hierarchies enable many real-time applications.*

into a directed acyclic graph, a DAG [KSA13]; however, this DAG is still traversable much like any other tree-like structure. We will present results that highlight the utility of this approach.

We show that this structure extends to animated geometries [KRB*16]. In essence, each frame can be transformed into a separate DAG, which is further compressed by exploiting inter-frame coherency by merging common subtrees across frames.

Advanced DAG encodings In the second part, we explain how a more efficient lossless compression of geometry can be achieved. We show how to merge subtrees that are identical through similarity transforms, and by exploiting the skewed distribution of references to shared nodes to store child pointers using a variable bit-rate encoding [JMG16, JMG17, DKB*16, KRB*16]. We will show high-resolution voxelized representations of real-world scenes. We demonstrate very differing characteristics via large CAD models, 3D scans, and typical gaming scenes. For example, using such an encoding, a real-time GPU in-core visualization of the full Boeing 777 at sub-millimetric precision, including shadows, becomes possible.

Attribute Compression The common subtree merging discussed in the previous parts will perform poorly if voxel attributes, such as colors and normals, are included. Instead, recent papers have suggested methods by which the voxel attributes can be compressed separately from the geometric information [DKB*16, DSKA17]. In these methods, it is important to maintain a connection from each voxel to its attributes without negatively impacting the compressability of the DAG. The attributes must also be accessible at real-time framerates. In this section, we will describe two recent methods by which the memory footprint of voxel attributes can be reduced to as little as 5-10% while maintaining voxel connections with negligible overhead in the compressed geometry DAG.

2.2. Implementation Details

Practical DAG Construction Here, we show how to construct the DAG data structures efficiently in practice. First, how DAGs are constructed from an existing SVO by sorting each level bottom up and removing redundant nodes while updating the parent pointers. Next, we introduce an acceleration, where identical subtrees are identified by calculating a hash value for each sub-tree, so that the common subtree merging can be performed top-down, greatly reducing the required amount of work [KSDA16]. Finally, at very high resolutions, the original SVO may be too large to fit into memory, and we will discuss how sub-regions of the voxel grid can be compressed separately and merged to produce the final compressed DAG.

Practical DAG Ray-tracing We will briefly talk about our methods and algorithms for ray-tracing the DAGs on the GPU. We start by presenting a basic GPU DAG ray-tracer, and then introduce a number of optimizations. We will discuss different trade-offs that can be made, and briefly touch on their impact in terms of performance and storage.

2.3. Applications

Demos We start this session by showing several possible applications, such as global illumination, collision detection, point-cloud compression, free-viewpoint video, large-scale visualization and more. A particular focus lies on precomputed shadows.

Compact Voxelized Shadow Volumes So far we have discussed common subtree merging for compressing voxel data representing surfaces. In this section we will describe how we can similarly compress precomputed binary light-visibility (i.e, shadow) stored in a voxel grid [SKOA14]. Aligning the voxel grid with the frustum of the light source increases compression rates and regions inside objects (which will not be queried for shadow evaluations) can be exploited. The resulting compression is very efficient, while being accessible in real-time.

Shadow Map Compression using Multiresolution Hierarchies Shadow maps are a very efficient 2D representation for visibility information. While DAGs operate in 3D, similar compression techniques can be transferred to shadow maps as well. We will explain how to compactly represent shadow maps using multiresolution hierarchies (MH) [SBE16]. MHs exploit the fact that for shadow computations the result can still be accurate if the depth value remains within a certain valid depth interval per texel. These intervals can be fused, resulting in a strong compression without any quality sacrifice. We will show how to encode MHs using quadtrees in order to query them in real-time, and we will explain how to achieve further compression rates by finding and merging similar subtrees within this quadtree. The resulting structure allows extremely high resolution shadow maps (e.g. 1 million pixels squared) to be compressed losslessly with compression ratios in the order of $1:10000$, while enabling real-time queries and hierarchical filtering.

The work was partially funded at CRS4 by project TDM and Sardinia Regional Authorities, the NWO VIDI NextView at TU Delft, Swiss National Science Foundation Project 174321, Advanced Postdoc Mobility, and the Swedish Research Council (Grant 2014-4559).

References

[CNLE09] Crassin C., Neyret F., Lefebvre S., Eisemann E.: GigaVoxels : Ray-guided streaming for efficient and detailed voxel rendering. In *ACM SIGGRAPH Symposium on 2009 Interactive 3D Graphics and Games (I3D)* (feb 2009). URL: http://graphics.tudelft.nl/Publications-new/2009/CNLE09a. 1

[CNSE10] Crassin C., Neyret F., Sainz M., Eisemann E.: *GPU Pro*. AK Peters, 2010, inbook X.3 Efficient Rendering of Highly Detailed Volumetric Scenes with GigaVoxels, pp. 643–676. URL: http://graphics.tudelft.nl/Publications-new/2010/CNSE10a. 1

[DKB*16] Dado B., Kol T. R., Bauszat P., Thiery J.-M., Eisemann E.: Geometry and attribute compression for voxel scenes. *Computer Graphics Forum (Proc. Eurographics) 35*, 2 (2016). URL: http://graphics.tudelft.nl/Publications-new/2016/DKBTE16. 2

[DSKA17] Dolonius D., Sintorn E., Kämpe V., Assarsson U.: Compressing color data for voxelized surface geometry. In *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2017). URL: http://www.cse.chalmers.se/~uffe/dolonius2017i3d.pdf, doi:10.1145/3023368.3023381. 2

[GMIG08] Gobbetti E., Marton F., Iglesias GuitiÃąn J.: A single-pass gpu ray casting framework for interactive out-of-core rendering of massive volumetric datasets. *Visual Computer 24*, 7-9 (2008). URL: http://publications.crs4.it/pubdocs/2008/GMG08a. 1

[JMG16] Jaspe A., Marton F., Gobbetti E.: SSVDAGs: Symmetry-aware sparse voxel DAGs. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2016), pp. 7–14. URL: http://www.crs4.it/vic/data/papers/i3d2016-symmetry-dags.pdf, doi:10.1145/2856400.2856420. 2

[JMG17] Jaspe A., Marton F., Gobetti E.: Symmetry-aware sparse voxel DAGs (SSVDAGs) for compression-domain tracing of high-resolution geometric scenes. *Journal of Computer Graphics Techniques (JCGT) 6*, 2 (May 2017). URL: http://jcgt.org/published/0006/02/01/. 2

[KRB*16] Kämpe V., Rasmuson S., Billeter M., Sintorn E., Assarsson U.: Exploiting coherence in time-varying voxel data. In *Proceedings of the 2016 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2016). URL: http://www.cse.chalmers.se/~uffe/exploiting_coherence_in_time-varying_voxel_data.pdf. 2

[KSA13] Kämpe V., Sintorn E., Assarsson U.: High resolution sparse voxel DAGs. *ACM Transactions on Graphics 32*, 4 (2013). URL: http://www.cse.chalmers.se/~uffe/HighResolutionSparseVoxelDAGs.pdf. 2

[KSDA16] Kämpe V., Sintorn E., Dolonius D., Assarsson U.: Fast, memory-efficient construction of voxelized shadows. *IEEE Transactions on Visualization and Computer Graphics 22*, 10 (2016). extension. URL: http://www.cse.chalmers.se/~uffe/Fast%20memory%20efficient%20construction%20of%20voxelized%20shadows%20TVCG.pdf, doi:10.1109/TVCG.2016.2539955. 2

[LK10] Laine S., Karras T.: Efficient sparse voxel octrees. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2010). URL: http://research.nvidia.com/publication/efficient-sparse-voxel-octrees, doi:10.1145/1730804.1730814. 1

[SBE16] Scandolo L., Bauszat P., Eisemann E.: Compressed multiresolution hierarchies for high-quality precomputed shadows. *Computer Graphics Forum (Proc. Eurographics) 35*, 2 (May 2016). URL: http://graphics.tudelft.nl/Publications-new/2016/SBE16. 2

[SKOA14] Sintorn E., Kämpe V., Olsson O., Assarsson U.: Compact precomputed voxelized shadows. *ACM Transactions on Graphics 33*, 4 (2014). URL: http://www.cse.chalmers.se/~uffe/compactPrecomputedVoxelizedShadows.pdf. 2