Real-time Model Slicing in Arbitrary Direction using Octree

Hei-Ting Tamar Wong†
School of Creative Media
City University of Hong Kong
httwong2@cityu.edu.hk

Yaozhun Huang†
Department of Mechanical and Biomedical Engineering
City University of Hong Kong
yaozhuang5-c@my.cityu.edu.hk

Sze-Chun Tsang†
School of Creative Media
City University of Hong Kong
szectsang2@cityu.edu.hk

Miu-Ling Lam[1, 2, 3]
[1]School of Creative Media
[2]Centre for Robotics and Automation
City University of Hong Kong
[3]CityU Shenzhen Research Institute
miu.lam@cityu.edu.hk

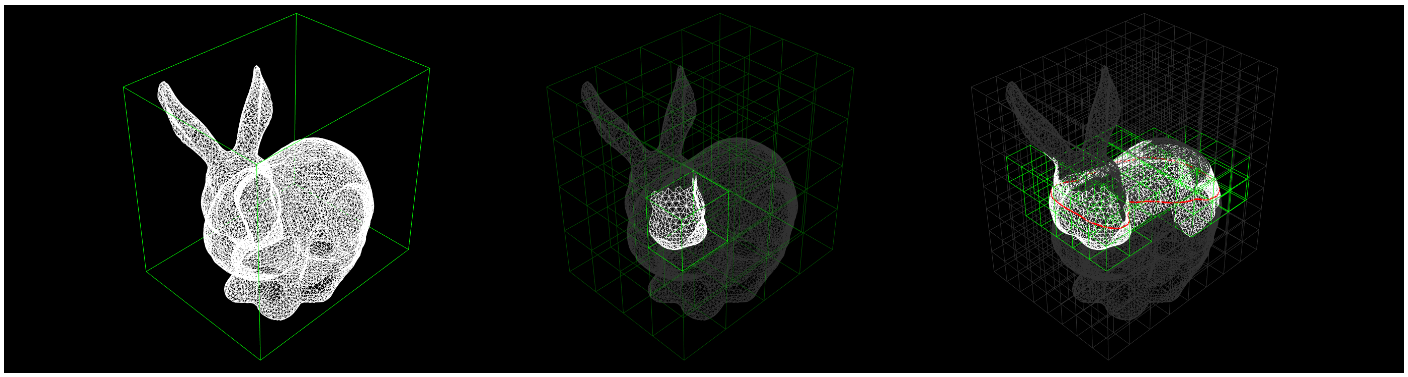

Figure 1: Octree data structure is used to reduce the computational time in finding the intersecting triangles and intersection curves of a 3D model with any arbitrary plane.

ABSTRACT

In applications such as computational swept volume light painting, real-time contour rendering is essential to ensure shape fidelity. However, there is, at yet, no real-time and scalable solution for slicing a model in arbitrary direction. We propose a new slicing method by organizing the triangular mesh into Octree data structure. The approach can significantly reduce the computational time and improve the performance of real-time rendering. The data structure is invariant to the slicing direction, thus constructing the Octree is a one-time, offline pre-process.

CCS CONCEPTS

• **Theory of computation → Data structures and algorithms for data management**; **Computing methodologies** → *Computational photography*

KEYWORDS

slicing, Octree, computational light painting

† These authors contributed equally to this work.

SIGGRAPH '17 Posters, July 30 - August 03, 2017, Los Angeles, CA, USA
ACM 978-1-4503-5015-0/17/07.
DOI: http://dx.doi.org/10.1145/3102163.3102185

ACM Reference format:

Hei-Ting Tamar Wong, Yaozhun Huang, Sze-Chun Tsang, and Miu-Ling Lam. 2017. Real-time Model Slicing in Arbitrary Direction using Octree. In *Proceedings of ACM SIGGRAPH, Los Angeles, California USA, July 2017 (SIGGRAPH'17)*, 2 pages.
DOI: 10.1145/3102163.3102185

1 INTRODUCTION

We have previously proposed a computational and swept volume approach to produce high-fidelity volumetric light painting [Huang et. al. 2016]. In particular, a flat screen is mounted to a 6-DOF robot arm and displays synchronized image sequence that represents the cross-section profiles of a virtual 3D object that is sliced at arbitrary positions and orientations. Real-time rendering is essential to ensure the synchronization between the robot pose and display contents, and, thus, the fidelity of the resultant long exposure. At every time instance, the intersection curves between the target 3D model and the plane of the flat display is computed and rendered in real-time. The process is highly related to "slicing" of 3D model which is an offline process in 3D printing.

Despite the fact that slicing algorithm has been widely explored, previous works only focus on slicing in one direction and the process is offline. Given any geometric model constructed by a set of unordered and unstructured triangles that approximates the surface of the model, the naivest method for slicing is to traverse all triangles for each slicing plane to obtain the intersection segments [Chalasani et. al. 1991]. Traversal approach wastes a great amount of time in dealing with unreachable meshes which are far from the slicing plane. Tata et. al. [1998] described an efficient algorithm for slicing the geometric model by pre-grouping the triangular facets in the *z*-direction. However, the method is only applicable when the slicing direction is fixed. Once the slicing direction changed, re-grouping of triangles is required. Rock et. al. [1991] proposed to utilize the topology information for improving the slicing efficiency. However, the method still relies on fixed slicing direction and finding the first intersection point between the model and slicing plane is time-consuming.

2 OUR APPROACH

We propose to utilize the Octree data structure to reduce the computational time during real-time model slicing. Given a geometric model, its axis-aligned bounding box (AABB) is identified (Fig. 1 *left*) and represents the root of the Octree structure (level-0). A hierarchical 8-ary tree is built by recursively subdividing the model into eight octants with *xy*, *yz*, *xz*-coordinate planes as the split directions. Each leaf node in the Octree contains a collection of triangles of the model that are geometrically enclosed by the node boundary (Fig.1 *middle*).

When an intersection query is received, the algorithm will check if the AABB of the model intersects with the slicing plane. If there is no intersection, the query will stop and return no triangle. If intersection is identified, the algorithm will conquer each node recursively. The recursion is terminated if the child node does not intersect with the slicing plane or it is a leaf node at the maximum tree level. As a result, a set of triangles is obtained at the conquered leaf nodes. The intersecting segments are then extracted by calculating the intersection line between the slicing plane and each triangle in the set. Due to the property of Octree, we can implement the method with a simple recursive function to access only the triangles which are closed to the slicing plane and omit the unreachable triangles, thus making the process very efficient. In addition, the slicing process is independent to the slicing direction which can ensure stable time cost and robust rendering performance.

We study the relationship between the choice of maximum tree level and the triangle number in the context of computational efficiency. In particular, we have simulated the average slicing time on a number of models of various triangle numbers (2K to 2.5M) using different Octree levels (0 to 6). If the Octree has too few levels, the model is subdivided too coarsely and too many triangles need to be considered in the leaf nodes. However, an Octree with many levels do not always guarantee faster computation because more time might need to spend on divide-and-conquer in unnecessarily high level of details; calculating the intersection between the slicing plane with a massive number of bounding boxes consumes a great amount of computing resources. Fig.2 indicates the average slicing time of four example models with various triangle numbers and demonstrates which number of tree levels has the optimal performance (red dots) on each model. The optimal tree level number increases with the number of triangles in the model.

Fig. 3 shows the results of computational light painting based on the proposed real-time rendering method. The photographs are unedited. These demonstrate that the proposed Octree-based model slicing method is computationally efficient and can produce high-fidelity 3D light painting imagery without any observable distortion. The proposed slicing or contour construction method is applicable for any applications that required real-timeliness and direction arbitrariness.

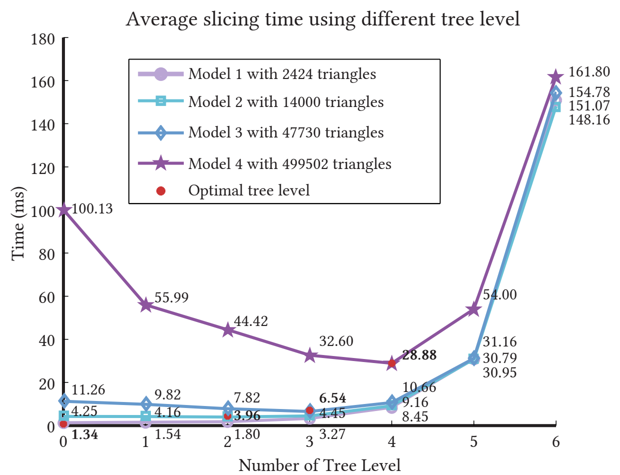

Figure 2: Different optimal Octree levels (red) yielded by models with different triangle numbers.

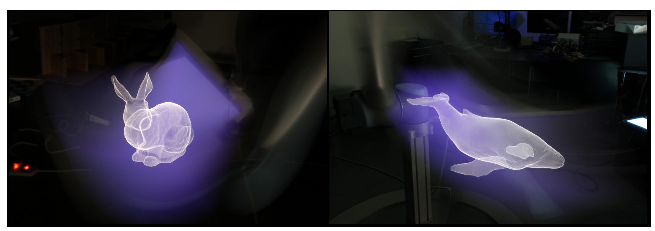

Figure 3: Real-time rendering result of light painting the Stanford bunny model and a humpback whale model.

ACKNOWLEDGMENTS

This work was partially supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11206216), National Natural Science Foundation of China (61502406) and Croucher Foundation (9500016).

REFERENCES

K. L. Chalasani, B. N. Grogan, A. Bagchi, C. C. Jara-Almonte, A. A. Ogale, and R. L. Dooley. 1991. An Algorithm to Slice 3D Shapes for Reconstruction in Prototyping Systems. In *Proceedings of the 1991 ASME Computers in Engineering Conference*, 209-216.

Yaozhun Huang, Sze-Chun Tsang, and Miu-Ling Lam. 2016. Computational swept volume light painting via robotic non-linear motion. ACM SIGGRAPH 2016 Posters, 27. DOI: https://doi.org/10.1145/2945078.2945105

Stephen J. Rock, and Michael J. Wozny. 1991. Utilizing Topological Information to Increase Scan Vector Generation Efficiency. In *Proceedings of Solid Freeform Fabrication Symposium*, 3-5.

Kamesh Tata, Georges Fadel, Amit Bagchi, and Nadim Aziz. 1998. Efficient slicing for layered manufacturing. *Rapid Prototyping Journal* 4, no. 4, 151-167. DOI: https://doi.org/10.1108/13552549810239003