Rapid Surface and Volume Mesh Generation from Depth-Augmented Visual Hulls

James Gregson *
University of British Columbia

Zheng Wang †
University of British Columbia

Figure 1: *Left to right: Input scan data. Single-scan depth map. All-scans silhouette map. Marching cubes reconstruction using Depth-Augmented Visual Hull.*

1 Introduction

Reconstructing scanned geometry is an important operation in geometry processing. Volumetric algorithms reconstruct the object volume by transforming range images into global coordinates and using scanline algorithms to build a scalar field that can be isocontoured to obtain the surface [Curless and Levoy 1996].

We take a slightly different volumetric approach by generating depth-maps for each range image. These divide the scanned frustum into an inside and outside half-space. Inside/outside tests may then be performed on the depth maps in image space using shadow-mapping techniques [Williams 1978]. To accommodate imperfections in the depths maps we use additional silhouette maps to help resolve ambiguously classified points. We call the combination Depth-Augmented Visual Hulls.

2 Our Approach

We assume as input a set of range scans consisting of two-dimensional arrays of points with implicit connectivity (quadrilaterals in our case) along with the per-scan transformations required to align the scans. We cull poor quadrilaterals in a preprocessing step based on ratios of edge lengths and quadrilateral areas. This helps remove quadrilaterals from areas that were scanned at a glancing angle, areas with specular reflections and areas where self-occlusions occurred.

We then build a set of depth maps–one per scan–of the scanned geometry by rendering it from the point of view of the scanner head. Points within the scanned frustum can then be classified as 'inside' or 'outside' by transformation into the scanner coordinate frame and comparison of the point depth value with the stored depth value. If the depth-maps were a perfect representation of the underlying geometry, only points classified as inside for all scans would be inside the object. Any meshing algorithm that can function with only a boolean predicate could then be used to generate a surface [Lorensen and Cline 1987] or volume [Labelle and Shewchuk 2007] mesh.

Invariably the scans are not perfect. Furthermore the culling process removes portions of the depth map. This causes points projecting near the object silhouette to be mis-classified as outside. To accommodate this we augment the depth-maps with silhouette maps. Each scan is then capable of classifying a point as strictly inside, strictly outside, or ambiguously positioned. This last classification is used for points within the silhouette, but with depth-map values beyond the object extents. The binary inside/outside predicate for all scans must then be modified to only classify a point as inside the object if it is an inside point in at least one scan, and not an outside point in any other. This essentially uses the depth information where available, and falls back to a visual hull for scans where data is incomplete.

The resulting algorithm is straightforward. Every candidate point is classified as inside or outside using the modified boolean predicate. Cells whose corners are entirely inside may be written to disk, while those with a mixture of inside and outside vertices have their surface-crossing edges added to a list for further processing. The surface intersection for those edges are then determined by binary search using the binary predicate. The resulting edge-intersections can then be stored for use in marching cubes or isosurface stuffing to generate a water-tight mesh.

Although our preliminary implementation is serial, the algorithm was designed to be implemented in parallel on the GPU. All operations are GPU and cache-friendly. Even so, our unoptimized implementation is fast; the 109,000 triangle reconstruction of the Bunny took 14 seconds on a mid-range laptop. Due to simple operation and coalesced memory access we expect a GPU version currently in development to be significantly faster.

References

CURLESS, B., AND LEVOY, M. 1996. A volumetric method for building complex models from range images. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, 303–312.

LABELLE, F., AND SHEWCHUK, J. R. 2007. Isosurface stuffing: fast tetrahedral meshes with good dihedral angles. *ACM Trans. Graph. 26*, 3, 57.

LORENSEN, W. E., AND CLINE, H. E. 1987. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph. 21*, 4, 163–169.

WILLIAMS, L. 1978. Casting curved shadows on curved surfaces. In *In Computer Graphics (SIGGRAPH 78 Proceedings*, 270–274.

*e-mail: jgregson@cs.ubc.ca
†e-mail: zhwang@cs.ubc.ca

Copyright is held by the author / owner(s).
SIGGRAPH 2010, Los Angeles, California, July 25 – 29, 2010.
ISBN 978-1-4503-0210-4/10/0007