

Shape Reconstruction from Raw Point Clouds using Depth Carving

Fabio Guggeri^{1,2}, Riccardo Scateni¹ and Renato Pajarola²

¹Dipartimento di Matematica e Informatica, University of Cagliari, Cagliari, Italy

²Department of Informatics, University of Zurich, Zurich, Switzerland

Abstract

Shape reconstruction from raw point sets is a hot research topic. Point sets are increasingly available as primary input source, since low-cost acquisition methods are largely accessible nowadays, and these sets are more noisy than used to be. Standard reconstruction methods rely on normals or signed distance functions, and thus many methods aim at estimating these features. Human vision can however easily discern between the inside and the outside of a dense cloud even without the support of fancy measures. We propose, here, a perceptual method for estimating an indicator function for the shape, inspired from image-based methods. The resulting function nicely approximates the shape, is robust to noise, and can be used for direct isosurface extraction or as an input for other accurate reconstruction methods.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

The continuously decreasing cost of shape acquisition techniques, from low-cost range scanners to image-based reconstruction algorithms, resulted in a growing availability of point datasets and, consequently, in an increased interest for robust shape reconstruction methods. Previous works in literature approach the problem in different ways, from fitting an implicit function to the points [KBH06] [CBC*01] to Voronoi or Delaunay based approaches [ACSTD07]. However, many datasets are noisy, incomplete and may lack normal information. Reconstruction of unoriented point sets is still an open research topic. Some works may focus on estimating the point normals [OF05] to apply a standard reconstruction method, or define a vertex-distance function to extract the surface as the zero-isosurface [MDGD*10] [HK06]. Anyway the final goal, and the main challenge, is to distinguish between what is *inside* or *outside* the shape.

In this work we describe an intuitive method for partitioning the space in the two regions, inside and outside the shape. We start from a set of unoriented points which sample the surface of the shape. We wanted to exploit the key idea that human vision is able to perceive the underlying shape in a dense cloud, thanks to spatial proximity and the perception

of depth given by stereopsis. From the human standpoint, what is perceived as *empty space* between the viewer and the points is considered to be outside space. The enclosing volume is, then, iteratively *carved* according to this perceptual paradigm. The resulting volume is the *Depth Hull* of the point cloud, which constitutes an accurate approximation of the underlying shape.

2. Background

The proposed approach is based on principles of Computer Vision that may result unfamiliar to a Computer Graphics audience. This section briefly introduces the basis for the Depth Hull reconstruction.

2.1. The Visual Hull

Introduced by Laurentini [Lau94], the *Visual Hull* (VH) of an object O is the maximal shape that, from any point of view, returns the same silhouette as the original one. As a complete computation is computationally expensive, it is common to refer to a *view-dependent* VH, that is the maximal shape, or *closest approximation* of O given a set of viewpoints $C \subseteq \mathbb{R}^3$, and is so defined: let $C \subseteq \mathbb{R}^3$ be a set

of points of view; the *visual hull* of an object O relative to C , $\mathcal{VH}(O, C)$, is defined as the subspace of \mathbb{R}^3 such that, for each point $p \in \mathcal{VH}(O, C)$, and each point $c \in C$, the projective ray starting at c and passing through p contains at least a point of O

$$\mathcal{VH}(O, C) = \{ p \mid \overline{cp} \cap O \neq \emptyset, \forall c \in C \}$$

that is, for each point $p \in \mathcal{VH}(O, C)$ and each viewpoint $c \in C$, the projective ray connecting p and c intersects the object O in at least one point. An example is shown in Figure 1 (left).

2.2. The Depth Hull

The *Depth Hull* (DH) is a generalization of the VH first defined by Bogomjakov and colleagues [BGM06] to approach the problem of reconstructing and rendering an image-based geometry from a set of depth cameras, also called Z-Cams. Such cameras are capable of returning depth information for each pixel in the view, and the *umbra* of a viewed scene is the portion of the visual cone behind the depth map, or the shadow cone generated by the object if a point light source is placed in the camera position. The DH is then defined as the intersection of the *umbrae* of the given reference Z-Cams (see Figure 1, right), and it is shown to be the best approximation of the geometry of an object when viewed from a reference set of viewpoints.

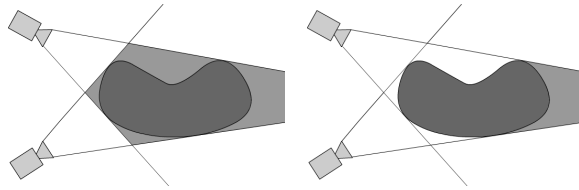


Figure 1: On the left: the *Visual Hull* (in light grey) of a shape (dark grey) given by two cameras is the closest approximation obtainable by an image-based approach. On the right: The *Depth Hull* (in light grey) of a shape (dark grey) given by two Z-Cams. By placing a third Z-Cam in the shadow zone, the shape is accurately reconstructed.

3. Indicator function estimation

When dealing with raw point sets, the main challenge is the definition of what portion of space is *inside* and what is *outside* of the represented shape. For algorithms like Poisson Reconstruction [KBH06] the presence of point normals is an indicator of the space gradient, while Radial Basis Function [CBC*01] uses a signed distance function given by the normal directions. The importance of signing can be found in other works specifically aiming at raw sets [MDGD*10], showing how the definition of inside and outside plays a

central role in the reconstruction problem. However, the human visual system is extremely fast and accurate in understanding the spatial relations of a dense enough point cloud. Spatial proximity is seen as a strong indication of an underlying surface. Many image-based reconstruction algorithms like Space Carving [KS00] are based on the same principle: the object is detected in an area of interest, and what falls outside such area is marked as *outside* also in the resulting reconstruction. This is a direct application of the Visual Hull definition.

It is then possible to reconstruct the Depth Hull of the point cloud by simulating a set of Z-Cams by checking the *depth buffer* of each rendered scene. The possibility of placing a virtually infinite number of Z-Cams around the object allows for a fast and accurate estimation of the *inside* of an object. We then define an indicator function for each voxel according to the *depth values* of each Z-image, given N Z-Cams $C_1 \dots C_N$ and the corresponding functions $D_i(x, y)$ returning the depth value of the i -th Z-image in the pixels x, y . Let \mathbf{M}_i be the transformation matrix from the object coordinates to the framework of the i -th Z-Cam, and, given a point \mathbf{p} in object-space, let $\mathbf{p}^i = \mathbf{M}_i \mathbf{p}$ be the transformed point in the i -th Z-Cam space. The function is defined as

$$X(\mathbf{p}) = \begin{cases} 1, & \text{if } \mathbf{p}_z^i > D_i(\mathbf{p}_x^i, \mathbf{p}_y^i) \forall i \in [1 \dots N] \\ 0, & \text{elsewhere} \end{cases}$$

where the subscript is just a coordinate selection $\mathbf{p}_x = \mathbf{e}_x \cdot \mathbf{p}$. In simpler terms, each point is projected on an image and if its Z-value is higher than the depth of the corresponding pixel, then the point is behind the *umbra* of such image. If the condition is satisfied for each camera, then it meets the Depth Hull definition, meaning that the point has to be considered *inside* the object. For a dense point cloud, the described function accurately reflects the Depth Hull of the shape and, consequently, a very good approximation of the underlying object.

4. The Depth Carving Algorithm

A direct computation for each voxel in a regular grid may be very expensive for high grid resolutions, as for N voxels and k viewpoints the complexity is $\mathcal{O}(N \times k)$. The usage of an adaptive octree is the best solution, but some time can be saved by aiming to process only *external* cells in the grid.

The **Depth Carving** algorithm we propose takes inspiration from the Space Carving [KS00] algorithm and other similar procedures [SBS02a] [SBS02b] used in image-based reconstruction: the sign computation is restricted to the external, *visible* voxels thanks to a *plane sweep* approach. The theoretical complexity is lowered to $\mathcal{O}(E \times k)$ where E denotes the *external* voxels in the shape.

Let G be a regular grid composed of $N_x \times N_y \times N_z$ voxels initialized with value 1, enclosed by the bounding box of the point cloud, and let $G_{x,y,z}$ be the voxel cell at coordinates

x, y, z . Let B be the set of *border voxel*, that is those voxels with at least one empty neighbor. Starting from an arbitrary point of view, each border voxels is *tested* by reprojecting its centroid onto the Z-Image in that direction: if the voxel is *in front* of the Z-Image (i.e., its reprojected depth value is less than the one stored in the image) its value is set to 0, the cell is removed and its 6-neighbors, if not already included in B , are added to the set. When every border voxel has been tested, the process is reiterated from a different point of view until each of all the viewpoints have been considered.



Figure 2: We obtain correct shape reconstruction from dense point clouds, independently from which kind of shape they sample (from left to right: Mug, Tooth, Armadillo, and Fandisk).

4.1. Implementation issues

As previously said, the Z-Cams can be simulated by synthetic cameras in a 3D visualization framework, using the GPU **depth buffer** to obtain the Z-Images. However, some care must be put into the implementation to overcome issues resulting from this approach. The point clouds may not be dense enough to obtain a smooth depth image (see Figure 3, left). This means that the empty space between the vertices will be *seen* by the camera and carved, resulting in an incorrect reconstruction. This issue can be addressed by lowering the projection resolution, so that during the rasterization no projective ray is able to traverse the space without intersecting a single vertex, or by performing a **morphological opening** on the Z-image: this step efficiently simulates a view-dependent Z-splating (shown in Figure 3, right) by smoothing the peaks in the Z depth function, so that small *holes* in the depth image are covered by the values of the

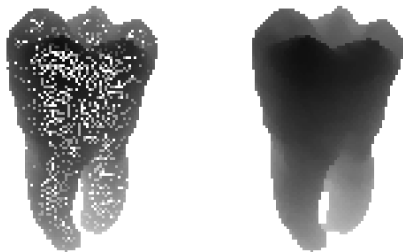


Figure 3: Missing depth data (left) can be recovered by a morphological opening of the depth image (right)

neighboring vertices, while maintaining an accurate contour of the shape.

5. Discussion and Results

In this section we show some results of our method and comparisons to a state-of-the-art algorithm. All the timings are obtained by single-threaded implementations on a 2.4GHz Intel CORE i5 processor. The meshes shown in Figure 2 are isosurfaces extracted after the sign estimation with a simple Marching Cube algorithm.

For dense clouds with no missing data, the reconstruction is accurate enough. Even if the resulting shapes are not visually pleasing, it must be considered that an isosurface taken from a binary dataset cannot guarantee smooth features. However, the method is intended as an aid for other algorithms as the Radial Basis Function or the Poisson, where a *signing* step is required. Our claim is that the proposed approach is suitable as a lightweight signing tool for unoriented point sets, thus extending the applicability of the cited methods without the need for normal estimation.

Noise robustness The approach is very robust to noise, as shown in Figure 4. Thanks to the multi-view approach, randomly scattered points cannot cast a consistent set of *umbrae*, and the space around them is then carved. The process may detect structured, dense outliers forming a solid shape, as the algorithm has no information on the target shape: everything that may resemble a solid object is reconstructed.

Timing The main advantage of this approach is the independence on the primitive numbers, as the vertices are a factor only in the time needed for rasterization. The whole approach is then fast and efficient. We compare to the signing step in Muller and colleagues [MDGD*10], a primitive-based algorithm, showing in Table 1 that our method is faster and more robust to changes in the sampling fineness.

5.1. Limitations

The main limitation of our method raises in presence of missing data. Wherever the cloud is not dense enough, the

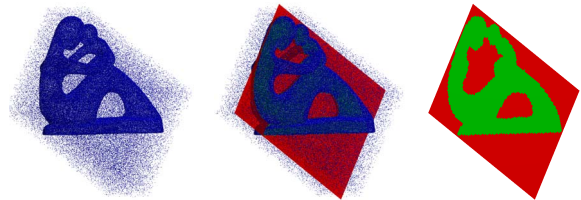


Figure 4: The algorithm is robust to white noise, as shown by the cross-section of the indicator function (right image). Internal cells are marked in green, external cells are marked in red.

Object	Vertices	Muller et al.	Depth Hull
Mug	798,877	42.878	15.126
Fertility	241,607	195.631	4.956
Raptor	49,995	311.416	1.065
Tooth	9,337	3.348	0.467

Table 1: Comparisons between our method and the sign guessing step proposed by Muller and colleagues, with an adaptive octree of level 8.

carving reaches the other side of the shape leaving an erroneous sign estimation (Figure 5). The algorithm, at the present state of development, cannot detect this absence and interprets the holes as strong concavities. Future improvements to this approach should take into account this issue in order to be applicable to all kinds of datasets. The dependence on parameters as the projection resolution or the size of the opening is unclear, and an automatic detection of the optimal parameters would be an interesting development for this approach.



Figure 5: Reconstruction from a holed model results in an excessive carving inside the shape. The depth image of the Bunny model shows missing data in its base (lighter colors denote farther points). On the right, the isosurface shows a set of cavities given by the depth carving.

6. Conclusion

We showed here how a perceptual approach can help in solving the signing challenge. The approach takes inspiration from image reconstruction and perception psychology to interpret a raw point set and correctly define an inside/outside function for the shape.

While still in a preliminary stage, the results are encouraging, suggesting that the approach could be integrated to other more complex methods for the analysis and understanding of unoriented point sets. Robustness to noise is accomplished thanks to the multi-view approach that discards scattered points. It is necessary to address the missing data issue: fu-

ture improvements should include a depth-based hole detection by studying the Z-Image gradient.

Acknowledgments

This work has been done while Fabio Guggeri was visiting the Department of Informatics of the University of Zurich.

References

- [ACSTD07] ALLIEZ P., COHEN-STEINER D., TONG Y., DESBRUN M.: Voronoi-based variational reconstruction of un-oriented point sets. In *Proceedings of the fifth Eurographics symposium on Geometry processing* (Aire-la-Ville, Switzerland, Switzerland, 2007), Eurographics Association, pp. 39–48. 1
- [BGM06] BOGOMJAKOV A., GOTSMANN C., MAGNOR M.: Free-viewpoint video from depth cameras. In *Proc. Vision, Modeling and Visualization (VMV) 2006* (Nov. 2006), pp. 89–96. 2
- [CBC*01] CARR J. C., BEATSON R. K., CHERRIE J. B., MITCHELL T. J., FRIGHT W. R., MCCALLUM B. C., EVANS T. R.: Reconstruction and representation of 3d objects with radial basis functions. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 67–76. 1, 2
- [HK06] HORNUNG A., KOBELT L.: Robust reconstruction of watertight 3d models from non-uniformly sampled point clouds without normal information. In *Proceedings of the fourth Eurographics symposium on Geometry processing* (Aire-la-Ville, Switzerland, Switzerland, 2006), SGP '06, Eurographics Association, pp. 41–50. 1
- [KBH06] KAZHDAN M., BOLITHO M., HOPPE H.: Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing* (Aire-la-Ville, Switzerland, Switzerland, 2006), SGP '06, Eurographics Association, pp. 61–70. 1, 2
- [KS00] KUTULAKOS K. N., SEITZ S. M.: A theory of shape by space carving. *Int. J. Comput. Vision* 38 (July 2000), 199–218. 2
- [Lau94] LAURENTINI A.: The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* 16, 2 (1994), 150–162. 1
- [MDGD*10] MULLEN P., DE GOES F., DESBRUN M., COHEN-STEINER D., ALLIEZ P.: Signing the unsigned: Robust surface reconstruction from raw pointsets. *Computer Graphics Forum* 29, 5 (2010), 1733–1741. 1, 2, 3
- [OF05] OUYANG D., FENG H.-Y.: On the normal vector estimation for point cloud data from smooth surfaces. *Comput. Aided Des.* 37 (September 2005), 1071–1079. 1
- [SBS02a] SAINZ M., BAGHERZADEH N., SUSIN A.: Carving 3D models from uncalibrated views. In *Proceedings IASTED International Conference Computer Graphics and Imaging (CGIM)* (2002), pp. 144–149. 2
- [SBS02b] SAINZ M., BAGHERZADEH N., SUSIN A.: Hardware accelerated voxel carving. In *Proceedings Ibero-American Symposium in Computer Graphics (SIACG)* (2002), pp. 289–297. 2