Polygonization of volumetric reconstructions from silhouettes

Anselmo Antunes Montenegro
Departamento de Informática e Ciência da Computação – IME – UERJ
anselmo@ime.uerj.br
Luiz Velho, Paulo C. P. Carvalho and Jonas Sossai Jr.
IMPA–Instituto Nacional de Matemática Pura e Aplicada
{lvelho, pcezar, jsossai }@visgraf.impa.br

Abstract

In this work we propose a method for the polygonization of octree-based reconstructions by dual contouring. *Dual contouring is an adaptive method for determining contiguous polygonal meshes from signed octrees. It determines the positioning of the vertices of the mesh by minimizing a quadratic error expressed in terms of hermitian data. In order to apply dual contouring on volumetric reconstruction from silhouettes we devised a method that is able to determine the discrete topology of the contour in relation to the octree cells, as well as the hermitian data corresponding to the intersections and normals of conic volumes whose intersection approximates a structure known as* Visual Hull. *Due to the discrete and extremely noisy nature of the data used in the reconstruction we had to devise a different criterion for mesh simplification that applies topological consistency tests only when the geometric error measure is beyond a given tolerance. We present results of the application of the proposed method in the extraction of a mesh corresponding to the surface of objects of a real scene.*

1 Introduction

Computer Graphics and related areas have recently directed a special interest towards the problem of shape reconstruction and determination of the reflectance properties of surfaces in real world scenes. The subarea that deals with such problems is commonly known as 3-D Photography.

There are many different approaches to 3-D Photography, some requiring more sophisticated equipment, as those based on laser scanning for range image determination, and some which rely solely on images taken from the scene.

3-D Photography based on laser scanning yields very precise reconstruction. Besides requiring special equipment, it has a somewhat complex pipeline: range image extraction, registration, structuring and texture mapping.

These techniques are usually employed when high quality models must be obtained as, for example, in the reconstruction of masterpieces for art analysis and archeological data. Unfortunately the complexity and the level of resources required by such methods forbids its use by the common application user.

An alternative to laser scanning methods is active stereo based on *structured light* [13]. The main idea of such methods is to project light with some known structure onto the scene. By doing this it is possible to compute 3D coordinates of the object's surface by triangulation without the severe ambiguity problems present in conventional stereo. Recent advances in such methods are the combination of photometric stereo [14] and use of spatial-temporal coherence to enhance the quality of the results [2, 20].

Although more affordable than laser scanning, stereo reconstruction based on structured light requires good quality projectors, which makes the final apparatus inadequate to be used when the access to the site is difficult, or when there is no appropriate power source.

In certain cases, one would like to reconstruct the object of interest by a collection of images taken from a set of spatially distributed cameras or a simple hand-held camera. Among the most popular methods used in these cases are the methods based on *volumetric reconstruction*, commonly known as *Space Carving* or *Voxel Coloring*.

Volumetric based methods rely on enumerated spatial-occupancy representations or adaptive representations based on hierarchical spatial data structures, such as octrees [11]. These representation schemes, although adequate for the underlying reconstruction mechanism, are neither compact, nor appropriate for use in processes like rendering, animation, and augmented reality.

An alternative to overcome this drawback is to avoid the determination of explicit structures corresponding to the volume of occupancy of the scene. An example of such method is the *Image Based Visual Hull* [10] proposed by Matusik which works directly in image space and is capable of rendering an approximated textured reconstruction of an object from a set of images and silhouette contours.

Matusik's method is very efficient and can be used in augmented and virtual reality real time applications. On the other hand it is not appropriate for modelling. In this case we would rather compute mesh-based representations from the volumetric representations.

In this paper, we address the problem of reconstructing mesh representations of objects from a collection of silhouette images. The method we propose is based on the polygonization of an octree representing an approximation to the volume of occupancy of the objects.

The polygonization of signed octrees can be obtained by a method known as *Dual Contouring* which is an adaptive method for determining contiguous polygonal meshes from signed octrees [5]. It determines the positioning of the vertices of the mesh by minimizing a quadratic error function expressed in terms of hermitian data (pairs of points and its normals). The method is also capable of producing a topologically safe simplification of the computed mesh.

Differently from the work in [5], the surfaces we wish to polygonize are not known. We only know a set of images of the silhouettes of objects in a scene which enables us to determine a structure known as *Visual Hull* [8].This structure is the maximal shape that reproduces the silhouettes of the shape to be reconstructed when projected according to the original camera parameters associated to the input images.

One of the main contributions of this paper is a method for the polygonization of octree-based representations of Visual Hulls. We devised a mecanism to determine signs of the vertices of the octree cells representing the topology of the contour associated to the Visual Hull in relation to each cell. Besides, we also developed a method for computing hermitian data associated to the approximated *Visual Hull*.

Another contribution is a new simplification criterion that deals with problems that arise in the *dual contouring* of volumetric data obtained from silhouettes, as the presence of geometrical and topological noise. The latter affects considerably the capacity of simplification of the original dual contouring method. In order to solve this we proposed a simplification criterion that uses a relaxation of the topological test when the geometric error is sufficiently small.

This paper is organized as follows. In section 2, we describe the fundamental ideas related to volumetric reconstruction from silhouettes and polygonization of volumetric data. In section 3, we present an overview of our method for extracting mesh surfaces from silhouette data, which is based on *octree carving* and *dual contouring*. In section 4, we describe in details each step of our method: the use of octree carving to determine an approximation to the Visual Hull of the scene; the computation of approximated hermitian data associated to the approximated Visual Hull; and the application of the dual contouring based on the computed hermitian data. In section 5, we present and analyse results. Finally, in section 6 we draw some conclusions and present final comments about this work.

2 Background

As mentioned above, our method uses a dual contouring polygonization method, applied to hermitian data extracted from volumetric reconstructions based on silhouettes. In this section we review silhouette-based volumetric reconstruction and *dual contouring*.

2.1 Volumetric reconstruction from silhouettes

Silhouette-based reconstruction methods aim to approximate a geometric structure known as *Visual Hull*. Basically, a set of images of the object to be reconstructed is segmented in foreground and background regions. The centers of projection of the cameras associated to each image and the corresponding regions belonging to the interior of the foreground determine conic volumes whose intersection approximates the *Visual Hull*.

In order to overcome the difficult computation required by methods based on geometric intersection, volumetric reconstruction methods were proposed. Instead of computing volume intersections, such methods classify the cells in a volumetric representation of the space as *interior*, *exterior* and *boundary* by projecting them onto the silhouette images.

Volumetric methods became very popular as they proved to be quite simple and efficient. There are several different ways to to compute the *Visual Hull* of an object from silhouettes. Here we exemplify only the two most commonly used: *voxel-based* and *octree-based methods*.

2.1.1 Voxel based reconstruction

Examples of the first volumetric reconstruction methods that used voxel-based representations are [3] and [9]. Later these methods were modified to consider also photometric constraints which brought forth methods known as Voxel Coloring [7, 16].

2.1.2 Octree based reconstruction

Voxel-based reconstruction has the inconvenient that it requires too much memory space. In order to deal with this drawback, several methods working on octree representations were proposed. The very first methods worked only with a limited number of images and with images produced by orthographic projections [1, 18]. There were others as, for example,[15], that were a little more generic, but still with strong limitations. Perhaps the most successful octree-based reconstruction method is the one proposed by Szeliski which reconstructs the shape of an object using a sequence of classification steps in increasing levels of resolution[17].

2.2 Polygonization of volumetric data

Polygonization of volumetric data is a well investigated subject in Computer Graphics and Geometric Modelling. The most traditional methods are the space-partitioning methods such as *Marching Cubes* [19] and its variants.

Another approach is based on dual methods, such as *Surface Nets* [4], which generate one vertex positioned near, or on the contour, for each cube that intersects it. In such methods, the vertices belonging to four cubes sharing an edge with sign change can be connected to form a quadrilateral. Meshes produced by dual methods tend to have better aspect ratio than those produced by *Marching Cubes*. An example of hybrid method is the *Extended Marching Cubes* (EMC) [6].

The algorithms described above are formulated for uniform grids, which waste memory space when representing cubes with homogeneous signs. Some methods were proposed to generate contours from octrees. Nevertheless, they restrict the octree to have neighboring leaves that differ by at most one level. They usually need some crack repair to ensure a closed contour. In the next subsection we describe a method that works with octrees without such drawbacks.

2.2.1 Dual Contouring

Ju et al [5] proposed an adaptive method that is able to compute a contiguous polygonal mesh from a signed octree.

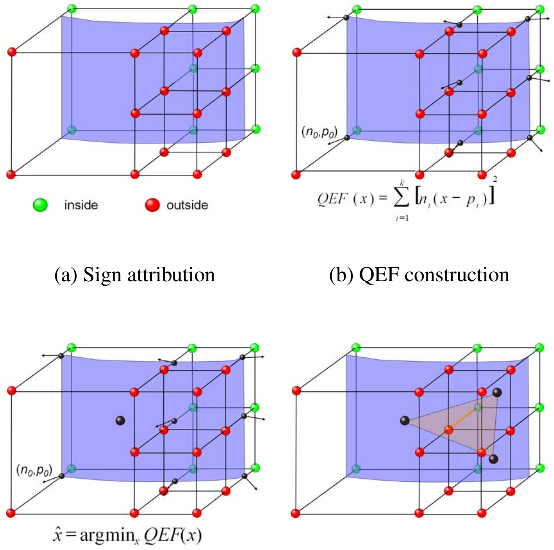
The method consists of three basic steps:

1. Generate a signed octree whose homogeneous leaves are maximally collapsed.

2. Construct a *quadratic error function* (QEF) for each heterogeneous leaf and simplify the octree.

3. Recursively generate polygons for the simplified octree.

The construction of the signed octree depends on the original description of the data. For an implicit function, it can be easily computed by determining the signs of the function applied to the vertices of each octree cell (Figure 1(a)). In the problem we consider here, there is no such description because we only know a collection of images of the objects. Hence, we had to devise an appropriate scheme for sign determination that we will describe further in the text.



(a) Sign attribution (b) QEF construction

(c) QEF minimization (d) Polygon generation

Figure 1. Dual Contouring

In *Dual Contouring*, a quadratic error function (*QEF*)

$$E(x) = \sum (n_i.(x - p_i))^2 \tag{1}$$

is specified for each leaf cell where p_i and n_i correspond, respectively, to the intersections of the edges of the cells with the contour and the unit normals to the contour at each intersection (Figure 1(b)). The pairs (n_i, p_i) determine the hermitian data used by the algorithm.

The minimization of the *QEF* in 1 computes the vertex inside a cell intercepted by the contour that is closer to it according to the quadratic error(Figure 1(c)). The quadratic error function in equation 1 can be expanded as follows:

$$E(x) = (Ax - b)^T(Ax - b) \tag{2}$$
$$= x^T A^T x - 2x^T A^T b - b^T b \tag{3}$$

where A is a matrix whose rows are the the normals n_i and b is a vector whose entries are $n_i.p_i$.

Ju et al devised a particular representation based on the *QR* decomposition so that robust minimization computation could be done when using float number representation. In our implementation we used double float number representation.

The *Dual Contouring* method builds a polygonal mesh from a simplified octree. The simplified octree is obtained by collapsing the octree leaves according to the error measured by the *QEFs*. The quadratic error on an coarse cell is computed as the sum of the quadratic errors in its children cells. If such error is smaller than a given tolerance, then the finer leaf cells are considered candidates to be collapsed.

Besides the quadratic error evaluation, a topological test is applied to the group of candidate leaf cells in order to finally decide whether they can be collapsed or not. This test checks if the resulting cell is topologically consistent with the topology of its children cells. Ju et al proposed the following test:

- Test whether the dual contour for the coarse cell is a manifold. If not, stop.
- Test whether the dual contour for each individual children cell is a manifold. If not, stop.
- Test whether the fine contour is topologically equivalent to the coarse contour on each of the sub-faces of the coarse cell. If not, stop; otherwise safely collapse.

Finally, polygons are created by connecting vertices in the cells that share a common edge with sign change (Figure 1(d)). In order to detect the cells that share a common edge Ju et al proposed a recursive algorithm that searches for *minimum edges* which are edges that are not adjacent to any edge in a level of refinement higher than its own level.

Each generated polygon is typically a quadrilateral, but it can be also a triangle if the edge is adjacent to cells in different levels of resolution. For more details of the method see the original work in [5].

3 Method Overview

The method we propose in this work can be summarized in the following steps:

1. Acquisition of calibrated foreground and background images.
2. Silhouette determination.
3. Octree carving and octree sign determination.
4. Dual Contouring.

First of all, we capture a set of images of the objects to be reconstructed and a set of background images of the scene. Here we suppose that the images are taken from previously calibrated cameras. Otherwise, both set of images must contain a calibration pattern so that the intrinsic and extrinsic parameters of the cameras, associated with each image, can be determined later. This is the case when a hand-held system is used as in [12]. In such case, a background model for each image is obtained by a warping process so that the objects of interest can be separated from the rest of the scene.

The silhouettes are determined by classifying pixels in foreground and background according to the background model previously determined. They are used to compute the volume of occupancy of the objects of interest by an octree carving algorithm.

In the next two sections we describe in more details the octree carving step and the dual contouring step.

3.1 Octree carving and octree sign attribution

The octree carving method we use here is similar to the method proposed by Szeliski in [17]. The algorithm determines the volume of occupancy of the objects by applying a classification method onto the bounding volume by iterative refinement. The space is represented by an octree and each cell is classified in the following labels:

- black - cell is completely inside the object,
- white - cell is completely out of the object.
- grey - undefined cell.

First the scene is represented by a bounding volume which can be the octree root cell or a group of subdivided cells. Initially all these cells are active and black. For each silhouette image, the cells are projected onto the support of a silhouette image and classified according to the rules described in table 1:

old color ⇒ result ⇓	black	grey	white
in	black	grey	white
undefined	grey	grey	white
out	white	white	white

Table 1. Update scheme of the cell labels.

The classification is based on *half-distance maps* [4] computed for each silhouette image. Later we will show that this approximation impacts the way signs are assigned to the vertices of the octree.

When all cells are classified with respect to each silhouette image, the algorithm is applied recursively onto those cells that were considered undefined, which are subdivided and considered active and labeled as black. The process stops when there are no undefined cells, or a certain level of refinement was reached.

3.1.1 Sign attribution

The classification of the octree carving can be used to define an initial sign attribution for the vertices on the leaf cells of

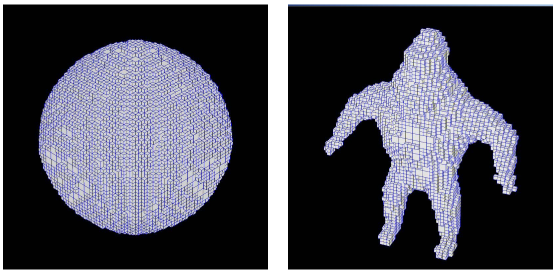


Figure 2. Octree carving of two syntetic models.

the octree. We first assign positive signs to the vertices of the leaf cells classified as *interior* cells and negative signs to the vertices of the leaf cells that are *exterior* cells, that is, out of the volume of occupancy. Those cells that are *undefined* have their vertices classified one by one by projecting them onto the silhouette images.

Here we use a hashing scheme so that the classification of the vertices of the leaf cells determine the signs of the vertices of all the cells of the octree. Finally, we apply a recursive algorithm to reclassify each internal(non-leaf) cell, with respect to the volume of occupancy, based on the signs of its children. This is necessary because the octree carving classification may differ from the classification based on the signs, because the first was done based on a half distance map which produces a conservative result.

4 Dual Contouring of Visual Hulls

The dual contouring is the kernel of the method and is responsible for determining the polygonal mesh that approximates the objects to be reconstructed. Some particular characteristics of the problem of shape reconstruction from silhouettes led us to adapt the original dual contouring method. The crucial difference is that the surface that characterizes the contour is not known and, consequently, we cannot attribute precise signs to the vertices of the cells neither determine exact hermitian data.

In fact, in our approach, we compute a polygonal surface that approximates the *Visual Hull* of the scene based on the silhouette images. In the next subsections we show the details of each step necessary for the polygonization of the approximated *Visual Hull* by *dual contouring*.

4.1 Hermitian data from silhouettes

In the original application of dual contouring, the determination of hermitian data is relatively simple because data is typically represented through implicit functions. In the problem of 3D reconstruction from silhouettes we only have knowledge of a set of silhouettes of the objects we try to reconstruct. Because of such difference in nature, we had to devise a way to estimate hermitian data for each leaf cell of the octree.

Our approach is to use the geometric information inferred from the silhouette images and their respective centers of projection. The silhouette images and the centers of projection of the respective cameras define conic volumes whose intersection approximates the *Visual Hull*, which is the shape supposed to be reconstructed in our approach.

For each leaf cell, and for each edge with sign change e, we project e in each silhouette image of the scene. If a projected edge pe crosses the silhouette in a given silhouette image, at a point p', we compute its 3D intersection p, with the corresponding conic volume V, and the 3D unit normal n at p.

The intersection point p of e with V is obtained by computing the intersection of the side line s and the line r on e. It is possible to express s in terms of the center of projection c and the direction d given by the difference between the 3D point p'' and c, where p'' is the unprojection of p' onto the far plane of the frustum of the camera.

The unit normal n at p is perpendicular both to the sideline s and to the perpendicular cross section of V at p. Hence it can be computed by the cross product between the 3D tangent vector t at p and the direction vector d of the sideline.

In order to compute the 3D tangent vector t we compute the 2D tangent vector t' at p' which can be estimated from a map of gradients of an approximated euclidian distance map defined on the silhouette image. The 3D tangent vector is obtained by unprojecting t' onto the far plane of the camera's frustum. Finally, the 3D normal is computed by $d \times t$. This description is shown in figure 3.

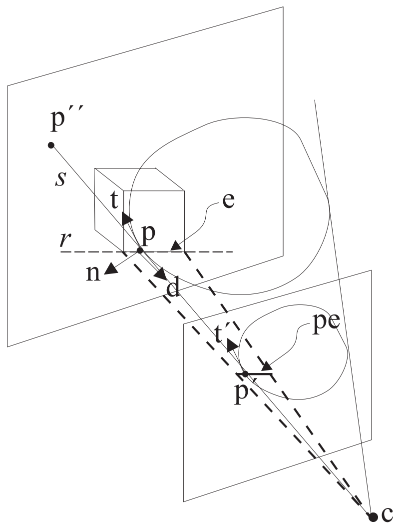


Figure 3. Computation of hermite data.

4.2 Dual Contouring from silhouettes

The polygonization of the approximated *Visual Hull* can be accomplished by computing the dual contour on the signed octree with hermitian data associated to the leaf cells. First of all, it is necessary to define precisely what is the quadratic error function that is used to deal with the fact that the hermitian data is just an approximation, differently from the hermitian data defined by implicit functions.

4.2.1 Error quadratic function minimization

One of the greatest problems in the use of dual contouring for mesh extraction from silhouette images is that the hermitian data are only approximated. Moreover, there are errors introduced by discretization inherent to image data. Another problem is that we do not have only one intersection-normal pair for each edge with sign change, differently from well behaved implicit functions. In fact, we may have as many intersection-normal pairs for an edge with sign change as the number of silhouette images.

These problems make the minimization process sensitive to errors produced by noise in the hermitian data. Initially, we used equation 1 as the *QEF* but this led to many problems. Vertices were positioned out of its cells due to instability on the associated linear system, even when double precision numeric representation is used.

As in [5] we added an additional term that forces the minimization process to approach the center of mass of the cell, when the set of hermitian data is composed by almost colinear normals. The resulting QEF is given by $E(x) = \sum (n_i.(x-p_i))^2 + \sum (x-p_i)^2$

4.2.2 Simplification

The simplification process we describe here is a modification of the procedure described in [5] to deal with the specific characteristic of the data we are using in the reconstruction.

In the original article, Ju et al did not mention any particular way to describe the tolerance value to evaluate and compare the $QEFs$. In order to evaluate the method described here, we defined the tolerance value as a percentage of an error value which is a function of the dimensions of the bounding volume of the reconstruction space. This value describes a bound to the quadratic error in the root cell considering the maximum number of edge intersections η, which induces the number of hermitian data considered equal to the product of the total number of edges by the number of silhouette images.

When $\|x - p_i\| \leq 1$ we have

$$E'(x) = \sum_{i=1}^{\eta} (n_i.(x-p_i))^2 + \sum_{i=1}^{\eta} (x-p_i)^2 \leq 2.\eta \qquad (4)$$

For a cell with arbitrary size we have

$$\frac{E(x)}{\|x-p_i\|^2} = \sum_{i=1}^{\eta} (n_i.\frac{(x-p_i)}{\|x-p_i\|})^2 + \sum_{i=1}^{\eta} (\frac{x-pi}{\|x-p_i\|})^2 \leq 2.\eta, \qquad (5)$$

and consequently,

$$E(x) \leq 2.n.\|x-p_i\|^2 \leq 2.\eta.D, \qquad (6)$$

where D is the diameter of the root cell.

The error value $k = 2.\eta.D$ is used only as a reference in our tests. We must bear in mind that k is not a bounding value to the sum of $QEFs$, but it is large enough for our simplification purposes and it varies accordingly to the data size.

The topological aspect of dual contouring simplification has many problems with octrees built from silhouettes due to what we call here *topological noise*. The topological noise here consist of small discrete topological substructures, that arise due to noisy data but also, and maybe mainly, due to the intrinsic discrete nature of the topological representation. These small substructures bid a greater simplification of the mesh because of topological inconsistencies in very small local regions.

This can be illustrated by figure 4. In all examples, the simplification considered only topological inconsistencies and no geometric error. We can see that the original method in the right column could not go further in the simplification process because of topological inconsistencies in the essay to collapse finer cells.

In order to solve this problem we modified the tests in the simplification stage so that the topological test is taken into consideration only when the geometric error is sufficiently large. This means that local topological inconsistencies are accepted when $E(x) < c.k$. Hence, in the beginning, small structures can have their topology modified, but in later stages the global topology of the object is kept constant. The results are illustrated in the right column of figure 4 where this relaxation strategy was used. We can see that we can obtain a mesh with much fewer triangles because we do not consider the presence of the topological inconsistencies in small features.

5 Results

In this section we present the results of the reconstruction of a mesh associated to a real scene captured with a hand-held camera. The results were obtained by using a PentiumIV 1.6 computer with 512Mb memory and a Pentax camera with 5.2 megapixels. We used eight 640x480 calibrated images of the scene and eight 640x480 calibrated images of the background. Calibration was done by inserting a calibration pattern in the scene (Figure 5). For each image a background was estimated by warping the set of background images as in [12].

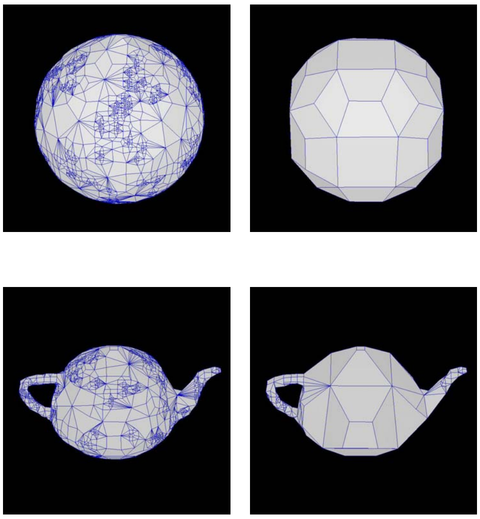


Figure 4. Left - orginal simplification strategy. Right - use of topological tolerance based on the quadratic error function.

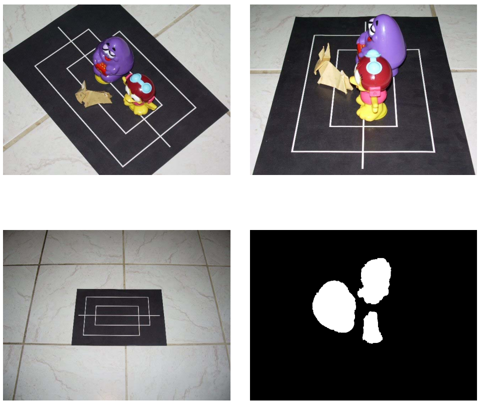


Figure 5. Top - two samples of the input set. Bottom - background sample and silhouette image.

Figure 6 shows the results obtained by dual contouring the octree-based reconstruction with no simplification. The mesh has 38718 polygons and was computed in 61s from an octree of maximum depth equal to seven. The extremely noisy nature of the data can be seen in figure 6.

The results of the simplification process are shown in figure 7 which represents a view of the reconstructed object with different levels of simplification. The left column represents the original simplification strategy proposed in [5] and the right column represents the results obtained by using the modified criteria proposed in this work. The error tolerances are measured as a percentage of k (the error value derived previously). In the modified criteria we used a constant factor $c = 0.05$, which means that the topological test is not considered for errors smaller than 5% of k. We can observe that the modified criteria produces meshes with much fewer polygons than the original criteria with the same value of quadratic error. It can also be noticed that in the original method the topological noise almost dominates the geometric error as the number of triangles is almost constant for all values of tolerance.

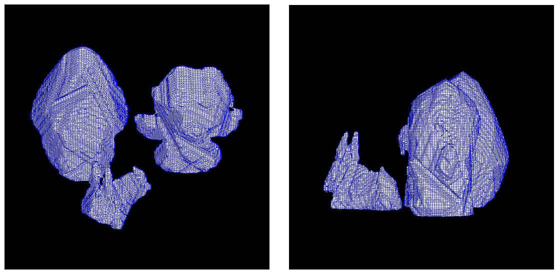


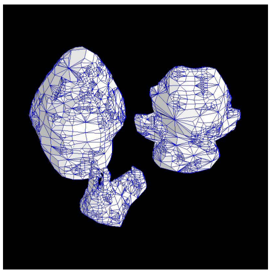
Figure 6. Two views of a reconstruction with no simplification.

6 Conclusion

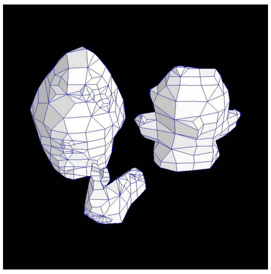
We presented in this paper a method for the polygonization of octree-based reconstructions from silhouettes. The results have shown that the method is a promising way to obtain low-resolution reconstructions of objects that can be combined very well with view-dependent texturing.

We also have investigated the use of dual contouring for extremely noisy data. We had to deal with topological problems in the simplification stage not usually found in the dual contouring of implicit surfaces. This fact indicates that in the case of discrete data, the relation between topology and geometry is complex. We solved such problems by considering topology not independently of geometry, by adopting a simplification procedure that combines both topological criteria and geometrical error.

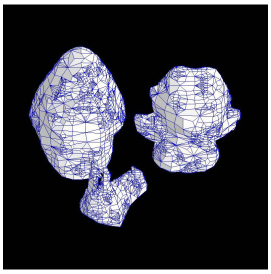
Future work must investigate more precisely this new kind of geometric object that arises from the polygonization of volumetric structures obtained from silhouettes. The relationship between topology and geometry in the case of discrete reconstructions must also receive special attention. Another future work is the design of multi-resolution structures for the meshes obtained by dual contouring and its use in view-dependent level of detail.



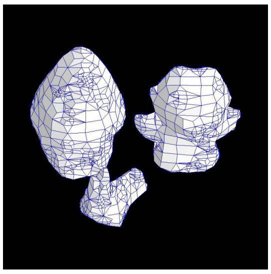
(a) 7434 polygons, error tolerance = 0.01, t = 59s



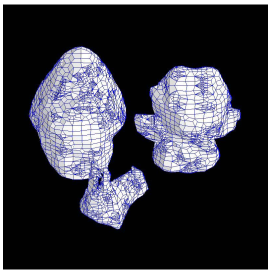
(b) 668 polygons, error tolerance = 0.01, t = 69s



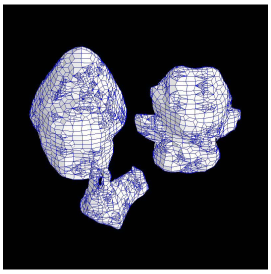
(c) 7545 polygons, error tolerance = 0.001, t = 71s



(d) 2447 polygons, error tolerance = 0.001, t = 70s



(e) 8007 polygons, error tolerance = 0.0001, t = 60s



(f) 7914 polygons, error tolerance = 0.0001, t = 71s

Figure 7. Results

7 Acknowledgements

This work was was partially funded by CNPq and FAPERJ. Anselmo Montenegro is grateful to FAPERJ for the financial support under grant E-26/152.827/2004.

References

[1] C. Chien and J. Aggarwal. Volume/Surface octrees for the representation of three-dimensional objects. *Computer Vision,Graphics and Image Processing*, 36:100–113, 1986.

[2] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence - PAMI*, 27:296–302, 2005.

[3] T. Fromherz and M. Bichsel. Shape from contours as intital step in shape form multiple cues. In *ISPRS Commission III Symposium on Spatial Information Form Digital Photogrammetry and Computer Vision*, pages 240–256, 1994.

[4] S. F. F. Gibson. Using distance maps for accurate surface reconstruction in sampled volumes. In *Volume Visualization Symposium - IEEE*, pages 23–30, 1998.

[5] T. Ju, F. Losasso, S. Schaefer, and J. Warren. Dual contouring of hermite data. In *SIGGRAPH 2002*, 2002.

[6] L. P. Kobbelt, M. Botsch, U. Schwanecke, and H. P. Seidel. Feature-sensitive surface extraction form volume data. In *SIGGRAPH 2001*, pages 56–57, 2001.

[7] K. Kutulakos. Approximate n-view stereo. In *European Conference on Computer Vision*, pages 67–83, Jun/Jul 2000.

[8] A. Laurentini. The visual hull concept for silhouette based image understanding. *IEEE PAMI*, 16:150–162, 1994.

[9] L. Massone, P. Morasso, and R. Zaccaria. Shape from occluding contours. In *Spie Conf. On Intellingent Robots and Computer Vision*, pages 114–120, Nov. 1985.

[10] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *SIGGRAPH 2000, Computer Graphics Proceedings*, 2000.

[11] A. A. Montenegro. *Reconstrução de Cenas a Partir de Imagens Através de Escultura Do Espaço Por Refinamento Adaptativo*. PhD thesis, PUC-Rio, September 2003.

[12] A. A. Montenegro, P. Carvalho, L. Velho, and M. Gattass. Space carving with a hand-held camera. In *XVII Brazilian Symposium on Computer Graphics and Image Processing*, Curitiba, 2004.

[13] E. Mouadibb, J. Battle, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem: A survey. *Pattern Recognition*, 31:963–982, 1998.

[14] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi. Efficiently combining positions and normals for precise 3d geometry. In *ACM Transaction on Graphics - SIGGRAPH*, pages 536–543, Los Angeles, USA, 2005.

[15] M. Potmesil. Generating octree models of 3d objects from their silhouettes in a sequence of images. *Computer Vision, Graphics and Image Processing*, 40, 1987.

[16] S. Seitz and C. Dyer. Photorealistic scene reconstruction by voxel coloring. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1067–1073, 1997.

[17] R. Szeliski. Rapid octree construction form image sequences. *Computer Vision, Graphics and Image Processing: Image Understanding*, 58:23–32, 1993.

[18] J. Veenstra and N. Ahuja. Efficient octree generation form silhouettes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 537–542, Miami, Florida, 1986.

[19] L. W. and C. H. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH 1987*, 1987.

[20] L. Zhang, B. Curless, and S. Seitz. Spacetime stereo: Shape recovery for dynamic scenes. *Computer Vision and Pattern Recognition*, 2003.