A POLYHEDRAL APPROACH TO COMPUTE THE GENUS OF A VOLUME DATASET

D. Ayala, E. Vergés and I. Cruz
Department de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain

Keywords: Binary Volume Models, Orthogonal Pseudo-polyhedra, Euler Characteristic, Genus.

Abstract: Topological characteristics are fundamental in many areas. In this paper we present a method that computes the Euler characteristic and the genus of a volume dataset. The followed approach, based on the analogy between binary volume datasets and orthogonal pseudo-polyhedra (OPP), computes the mentioned values using two models that are suitable for representing OPP: the Extreme Vertices Model (EVM) and the Ordered Union of Disjoint Boxes (OUBD). We show the results of these methods for phantom models as well as real datasets, and compare the efficiency of the presented methods with those based on the classic voxel model.

1 INTRODUCTION

The measurement of the topological characteristics of an object such as its number of connected components and cavities or its genus is a useful tool in many applications as image analysis. These attributes can be computed as a way to detect the relevant features of a model. Alternatively, they can be used to prove that homotopy is preserved in several kinds of processes, as skeletonization or simplification. In the Bio-CAD field, the genus is related to the connectivity which is used to measure the strength of bones (osteoporosis) or the quality of biomaterials designed to repair them.

In this paper we present a method for computing the Euler characteristic (EC) and the genus of a binary volume dataset. The presented approach considers, first, the non-manifold orthogonal polyhedron (OP) analog to the binary volume model and, then, a manifold OP homotopic to the non-manifold one. After that, the classic method that computes the EC for polyhedra is applied to the special case of OP. Finally, the genus is obtained from the EC and the shells of the model obtained with the application of a connected component labeling (CCL) process. OP are represented with two alternative models, the Extreme Vertices Model (EVM) and the Ordered Union of Disjoint Boxes (OUBD). EVM is a very concise B-Rep model while OUBD is a decomposition model.

In this paper, we contrast the results and performance of the presented methods with those that compute the same parameters using the voxel model. We use several phantom models and real datasets for these tests. The main contributions are the following. First, the implementation of a method that computes the connectivity for binary datasets as well as for OP (without voxelizing them), which is faster than using the reference voxel method. Second, it also extends the functionality of the EVM and OUBD models, which are useful for many operations such as Boolean, morphological, CCL, or visualization (Aguilera, 1998; Rodríguez and Ayala, 2003).

The paper is arranged as follows. Next section reviews related work. Section 3 introduces the EVM and OUBD models. Section 4 presents the connectivity computation method and Section 5 discusses the achieved results. Finally Section 6 concludes this paper and points out future work lines.

2 BACKGROUND AND RELATED WORK

2.1 Binary Volume Models and Orthogonal Polyhedra

A binary volume model is a union of voxels with associated values restricted to 0 for background voxels and 1 for foreground voxels. In them, three kinds of adjacency relations are defined between voxels: 6, 18 and 26-adjacency. Two voxels are 6-adjacent if they share a face, 18-adjacent if they share an edge or a face, and 26-adjacent if they share at least a vertex.
An adjacency pair \((m, n)\) defines the adjacency applied to a binary volume dataset, meaning that foreground voxels are \(m\)-adjacent and background voxels are \(n\)-adjacent (Kong and Rosenfeld, 1989). Using the same adjacency relations for foreground and background voxels results in paradoxes that can be avoided by restricting these pairs to \((6, 26)\) and \((26, 6)\) (Kong and Rosenfeld, 1989; Lachaud and Montanvert, 2000).

Latecki (Latecki, 1997) defines the term of well-composed. A binary volume model is well-composed if the configurations shown in Figure 1 (left and middle), modulo reflections and rotations, are avoided. He also relates the concepts of well-composedness with manifoldness. Figure 1 (right) shows the non-manifold configuration for 2D images.

However, \((6, 26)\) and \((26, 6)\) binary volume models are non-manifold as 26-adjacency causes non-manifold configurations on its boundary.

A polyhedron is a subset of \(\mathbb{R}^3\) that is bounded, closed, regular and semianalytic (Requicha, 1980). A 2-manifold polyhedron can be informally described as a collection of planar faces such that each edge is shared only by two faces and each vertex is the apex of only one cone of faces (Rogni and Requicha, 1991). A pseudo-polyhedron is an extension of the concept of polyhedron with a non-manifold boundary (Tang and Woo, 1991; Rossignac and Cardoza, 1999). Informally, a pseudo-polyhedron can be defined as a polyhedron with non-manifold edges and vertices. A non-manifold edge is adjacent to more than two faces and a non-manifold vertex is the apex of two cones of faces. These two cases are those corresponding to the non-manifold configurations of Figure 1 (left and middle).

The approach followed in this paper uses an orthogonal pseudo-polyhedron instead of a voxel model to represent a binary volume model as a map between these two representations can be established (Khachan et al., 2000). Moreover, as our goal is to compute the Euler characteristic and the genus using an expression suitable for manifold polyhedra, we will convert the orthogonal pseudo-polyhedron into a manifold one. The basic method to do this consists in duplicating non-manifold edges and vertices (Floriani et al., 2003). The matchmaker method (Rogni and Cardoza, 1999) decomposes a non-manifold polyhedron into manifold parts by replicating and bending the non-manifold edges and, then, replicating and moving slightly apart the remaining non-manifold vertices. This method minimizes the number of duplications. Operations of cutting and stitching (Guéziec et al., 2001) are applied to polygonal surfaces to remove non-manifold edges and vertices, for compression purposes. Non-manifold decomposition has also been extended to general \(d\)-dimensional simplicial complexes (Floriani et al., 2003). The authors obtain manifold parts for \(d \leq 2\) and pseudomanifolds parts that require a further decomposition for \(d \geq 3\). In (Attene et al., 2009) the authors justify that tetrahedra models not only have to be well-shaped but also manifold models in order to implement efficient methods for Boolean operations or model simplification. They present an approach for volume tetrahedral models based on operations that repair locally first singular edges and then singular vertices.

### 2.2 Euler Characteristic and Genus

The Euler characteristic, \(\chi\), of a binary volume model can be expressed in terms of the Betti numbers (Massey, 1991):

\[
\chi = \beta_0 - \beta_1 + \beta_2
\]  

Figure 1: Non-manifold 3D (left and middle) and 2D (right) configurations.

Figure 2: Manifold vertices in orthogonal polyhedra: two configurations for V3 and one for V4 and V6.
\( \beta_i \), being the \( i \text{th} \) Betti number. \( \beta_0 \) and \( \beta_2 \) are associated to the number of connected components and cavities while \( \beta_1 \) is the connectivity which is related to the genus. \( \chi \) can be computed from a voxel model with the following expression (Odgaard and Gundersen, 1993):

\[
\chi = n_0 - n_1 + n_2 - n_3
\]

(2)

\( n_0, n_1, n_2 \) and \( n_3 \) standing respectively for the number of vertices, edges, faces and voxels. \( \beta_0 \) and \( \beta_2 \) are usually computed using a CCL algorithm and \( \chi \) and the connectivity \( \beta_1 \) with expressions 2 and 1, respectively. Toriwaki et al. present two different strategies to compute expression 2 and allow several adjacency pairs (Toriwaki and Yonekura, 2002).

Moreover, in the solid modeling area, \( \chi \) can be computed from a polyhedron with the following expression (Mäntylä, 1988):

\[
\chi = V - E + F - R = 2(S - g)
\]

(3)

\( V, E, F \) and \( R \) being the number of vertices, edges, faces and internal rings of faces, respectively, of a polyhedron. \( S \) is the number of shells (number of sets of connected faces) and \( g \) is the genus.

For the special case of triangulated surfaces, \( \chi \) can be expressed as:

\[
\chi = V - E + T
\]

(4)

\( V, E \) and \( T \) being the number of vertices, edges and triangles of the mesh.

A triangle mesh can be extracted from a voxel model to approximate a given isosurface separating interior and exterior voxels. This mesh can be obtained with the Marching Cubes algorithm (Lorensen and Cline, 1987) or one of its derived methods that try to solve the flaws of cracks and ambiguities of the initial proposal (Bourke, 1994).

Expression 4 is combined with Expression 1 to compute the connectivity of a triangular isosurface corresponding to the Laplacian of the electron density field of adenine (Konkle et al., 2003). In this case, \( \beta_0 \) and \( \beta_2 \) are computed with a CCL method applied to triangular surfaces. An incremental method is used to compute the Betti numbers for any dimension as well as for \( \alpha \)shapes (Delfinado and Edelsbrunner, 1993).

The Euler characteristic and genus are used to determine if two objects have the same topology. Informally, two objects are homotopic, i.e., have the same topology, if they have the same number of connected components, tunnels and cavities (Kong and Rosenfeld, 1989). Homotopy is a desirable property in many methods and applications such as thinning. The concepts of simple voxel and simple deformations are also used to define topology preservation for skeletonization and to devise the corresponding thinning methods, based on the deletion of simple voxels (Rosenfeld et al., 1998; Bertrand and Malandain, 1994; Borgefors et al., 1999). In isosurface extraction, topology-preservation is sometimes a desirable property that can be evaluated by computing the Euler characteristic (Schaefer et al., 2007). Sarioz et al. (Sarioz et al., 2004) compute the Betti numbers \( \beta_0, \beta_1 \) and \( \beta_2 \) with a fast incremental approach. They use these triples, computed for the binary volumes obtained from every possible threshold value of a gray-scale 3D image, to classify a large collection of datasets.

In the Bio-CAD field, the connectivity is related to biomechanical properties and is used to measure the strength of bone or other materials. The method based on expressions 1 and 2 is used to evaluate the osteoporosis degree of mouse femur (Martín-Badosa et al., 2003) or human vertebrae (Odgaard and Gundersen, 1993) or to evaluate hydraulic properties of sintered glass (Vogel et al., 2005).

There are methods that compute the connectivity from a curve skeleton previously computed. Pothuau et al. (Pothuau et al., 2002; Pothuau et al., 2004) apply a 3D-line skeleton graph analysis (3D-LSGA) to MR images of human vertebrae. The number of loops of this graph is related to \( \beta_1 \). Moreover, there is a relationship between the coordination number of this graph (average number of edges in a vertex) and the connectivity (Ioannidis and Chatzis, 2000). Some other approaches are based on the computation of a surface skeleton. A plate/rod model is derived, based on the surface/linear regions of the surface skeleton (Staub and Müller, 2006; Peyrin et al., 2007) and the connectivity is measured as the relative amount of plate and rod elements.

### 2.3 Alternative Models

Binary volume models are mostly represented with the classic voxel model. However, in the volume analysis and visualization field, several alternative models have been devised for specific purposes.

Hierarchical decomposition models as octrees and kd-trees have been used for Boolean operations (Samet, 1990), CCL (Dillencourt et al., 1992), thinning (Quadros et al., 2004) and also to extract and simplify isosurfaces (Andújar et al., 2002; Vanderhyde and Szymczak, 2008; Greß and Klein, 2004).

Other models aim to store only surface voxels to gain storage and computational efficiency. The semi-boundary representation allows direct access to surface voxels and performs fast visualization and manipulation operations (Grevera et al., 2000). There
are methods for morphological operations and CCL that use this representation (Thurfjell et al., 1995). The slice-based binary shell representation also stores only surface voxels and is applied to render binary volumes (Kim et al., 2001).

3 EVM AND OUDB REPRESENTATIONS

In this work, the Euler characteristic and the genus are computed using two alternative representations. This section introduces the EVM and OUDB models. For a deeper study of these models and their properties, see (Aguilera, 1998) and (Rodríguez et al., 2004).

The Extreme Vertices Model (EVM) is an implicit B-Rep model for orthogonal pseudo-polyhedra that stores only a subset of its vertices, named extreme vertices (EV). These vertices are the ending vertices of maximal uninterrupted segments (see Figure 3) which are stored lexicographically.

A cut, \( C_i \), is the set of extreme vertices lying on the same orthogonal plane (see Figure 3, above, left). An orthogonal polyhedron can also be seen as a sequence of orthogonal prisms represented by its section, \( S_i \) (see Figure 3, above, right). Cuts and sections are orthogonal polygons embedded in 3D space. For each main direction, sections can be computed from cuts and cuts from sections with the two following expressions:

\[
S_i = S_{i-1} \oplus C_i, \quad S_0 = \emptyset \quad \text{and} \quad C_i = S_{i-1} \oplus S_i,
\]

\( i = 1..n \). The symbol \( \oplus \) stands for the XOR Boolean operation.

These operations are actually performed on the projections of cuts and sections onto the main plane parallel to them. So, from now on, we denote as \( C_i \) and \( S_i \) the projection of the \( i \)th cut and the \( i \)th section respectively onto the main plane parallel to them.

Moreover, there is a property concerning orthogonal polygons and polyhedra represented with EVM that states that the XOR Boolean operation is reduced to the application of an XOR at the vertices’ level (0-D XOR). \( EVM(A \oplus B) = EVM(A) \oplus EVM(B) \). Therefore the computation of sections from cuts and cuts from sections is a simple XOR of its vertices.

The Ordered Union of Disjoint Boxes (OUDB) is a decomposition model that is derived from EVM by splitting the aforementioned orthogonal prisms into boxes (see Figure 3, below). OUDB is axis-aligned as octrees and bintrees but the partition is done along the object geometry like the BSP model. In the presented approach, to compute the genus, we will use a CCL method developed for the OUDB model (Rodríguez and Ayala, 2003).

The improved performance achieved with these alternative representations relies on the fact that the corresponding algorithms are basically traversals of extreme vertices or boxes instead of traversals of voxels and that, in most cases, the number of extreme vertices and boxes is substantially smaller than the number of foreground voxels.

Many operations have been developed for these models and some of them are used in the presented approach as morphological erosion and dilation (Rodríguez and Ayala, 2003). These operations are recursive in the dimension and are actually performed on the 1D sections of the 2D sections of the aforementioned prisms.

4 COMPUTATION OF THE EULER CHARACTERISTIC AND GENUS

The approach followed in this paper to compute the Euler characteristic and the genus is based on expression 3, suitable for manifold polyhedra. As the continuous analog of a (26, 6) or (6, 26) binary volume model is an orthogonal pseudo-polyhedron, we will convert it into a manifold one with an offset operation.

A positive offset is defined as (Rossignac and Requicha, 1986):

\[
P' = P \oplus d = \{ p \mid \exists q \in P, \text{dist}(p, q) \leq d \}\]  

(5)
dist can be computed in several metrics. In this paper, we use the chessboard metric, \(L_\infty\), to guarantee the closure of orthogonal polyhedra under offset operations.

A negative offset is defined as the complement of the expanded complement. Positive and negative offsets are related respectively to dilation and erosion (Gonzalez and Woods, 1992) and to Minkowski sum and decomposition (Ghosh and Haralick, 1996).

Then, our method is supported by the following proposition:

**Proposition 1** Let \(P\) be an orthogonal pseudo-polyhedron that is the continuous analog of a \((26, 6)\) binary volume model and let \(P'\) be the positive offset of \(P\) with distance \(d < 0.5s\), \(P' = P \oplus d\), and \(s\) being the voxel’s size of the underlying binary volume model. Then, the two following properties hold:

- \(P\) and \(P'\) are homotopy-equivalent
- \(P'\) is a 2-manifold

This proposition is now demonstrated informally. With the positive offset, we duplicate all the non-manifold edges and vertices topologically as well as geometrically and this is the basic process to convert a non-manifold to a manifold. However, a non-restricted positive offset could result in new connections or disconnections of the boundary of \(P\) and, therefore, the resulting offset would have a different number of connected components, cavities or tunnels than the original object. Even worse, new non-manifold elements could appear in other regions. Both problems are avoided if the offset distance is less than half the size of the voxel of the underlying binary volume model.

An analogous proposition can be stated for \((6, 26)\) binary volume models, in which case a negative offset is applied.

Figure 4 (above) shows the result of applying a positive offset to two orthogonal pseudo-polyhedra that include the two possible non-manifold configurations (edge and vertex). Figure 4 (below) is analogous for a negative offset.

Figure 5 shows a 2D example with an orthogonal polygon and the underlying binary model. The initial polygon (in brown) has 2 connected components and 1 hole. Above, we apply a positive offset with \(d = 0.5s\) and the resulting polygon has only one connected component and no holes. Moreover, with this offset a non-manifold vertex has disappeared but a new one has been created. With \(d < 0.5s\) (below) the resulting polygon is homotopy-equivalent to the initial one, the non-manifold vertex has been removed and no new non-manifold vertices have appeared.

The proposed method uses alternatively the EVM and OUDB representations of the polyhedron (EVM-Rep and OUDB-Rep) that have already been computed. The whole process consists of the following steps, that are explained below:

**Input:** EVM-Rep and OUDB-Rep of the polyhedron

**Output:** Euler characteristic \(\chi\) and genus \(g\)

1. Compute the number of connected components from OUDB-Rep, \(n_1\)
2. Compute the complement of OUDB-Rep
3. Compute the number of connected components of the complement minus one, \(n_2\)
4. Compute the number of shells, \(S = n_1 + n_2\)
5. Apply a positive offset with \(d < 0.5s\) to the EVM-Rep
6. For each cut, $C_i$, $i = 1..n$, of the dilated EVM-Rep in each direction $X$, $Y$, $Z$ do:

(a) Compute the number of faces, $F_i$, (number of connected components) from the OUDB-Rep of $C_i$

(b) Compute the number of rings, $R_i$, (number of connected components of the complement of the OUDB-Rep of $C_i$, minus one)

(c) Detect and count $V4_i$ and $V6_i$ vertices

7. Compute the total number of faces ($F$), rings ($R$) and vertices $V4$ and $V6$: $F = \sum_{i=1}^{n} F_i$, $R = \sum_{i=1}^{n} R_i$, $V4 = \sum_{i=1}^{n} V4_i$, and $V6 = \sum_{i=1}^{n} V6_i$

8. $V = EV + V4 + V6$

9. $E = (3EV + 4V4 + 6V6)/2$

10. $\chi = V - E + F - R$

11. $g = S - \chi/2$

Steps 1, 2 and 3, use a CCL algorithm (Rodríguez et al., 2004) to compute the number of connected components (step 1) and the number of cavities (steps 2 and 3). This algorithm follows a classic two-pass strategy like the algorithms based on voxel models but it traverses a set of boxes instead of voxels. As this method can deal with pseudo-manifold polyhedra, it can be applied to the initial object as well as to the dilated object.

Step 5 applies a positive offset to the EVM-Rep in order to guarantee that the resulting polyhedron is a two-manifold. The applied method performs 1-dimensional offsets consecutively in the three main directions. The details of the algorithm can be found in (Rodríguez and Ayala, 2003).

The following steps compute the number of the remaining elements in expression 3. We denote by $F$, $E$ and $R$, the number of faces, edges, and internal rings of faces and by $V3$, $V4$, $V6$ the number of vertices with 3, 4 and 6 faces converging at them, respectively.

EVM-Rep stores only extreme vertices that, in the case of manifold polyhedra, coincide with vertices $V3$. Therefore, $V3$ is obtained directly from the EVM-Rep. To compute $F$, $R$, $V4$ and $V6$, we perform a traversal of all the cuts of the EVM-Rep. Every cut is a 2D orthogonal polygon that is also EVM and OUDB represented. The faces and rings of a cut (in 2D) play the same role as the number of connected components and cavities (in 3D) and so they are computed in the same way, i.e., by applying the same CCL algorithm. Therefore, step 6 (a) and (b) are 2D versions of steps 1 to 3.

Step 6 (c) is performed taking into account the following considerations. Although we are working with a two-manifold 3D polyhedron, its faces may present the non-manifold 2D pattern shown in Figure 1 (right): the two coplanar faces incident to a V4 vertex present this pattern as well as the three pairs of two coplanar faces of a V6 vertex (see Figure 2). To compute $V4$ and $V6$ we rely on these facts. When processing each cut, 2D non-manifold vertices are detected and stored in a list together with the number of times that have been found. $V4$ vertices are those detected once while $V6$ are those detected three times.

Detection of 2D non-manifold vertices is performed with EVM-Rep, i.e., with cuts and sections in 1D. When a 1D cut of a polygon contains a vertex of the previous 1D section, this vertex is non-manifold. This property is actually evaluated for the projections of cuts and sections onto the main axis parallel to them. Figure 6 shows an example of a polygon with 8 cuts and 7 sections. Cut $C_7$ contains a vertex of section $S_5$ (their projection onto the $Y$ axis), therefore the marked vertex is non-manifold.

![Figure 6: 2D example with a non-manifold vertex.](image)

Step 7 indicates that all these elements computed for each cut are added in order to compute $F$, $R$, $V4$ and $V6$ for the whole object. Step 8 computes the total number of vertices ($V$) and step 9 the total number of edges ($E$) from the number of edges that each class of vertices has: $E = (3V3 + 4V4 + 6V6)/2$. Finally, steps 10 and 11 compute the Euler characteristic and the genus, respectively, using expression 3.

5 RESULTS

We have measured the $\chi$ and genus of several test datasets by means of three methods: the methods applied to voxel models and to triangle meshes (see Section 2.2) and the method applied to EVM/OUDB that is presented in this paper (see Section 4). These three methods produce the same results, and in this section we compare their execution times.
Table 1: For each tested dataset, the size of the voxel model, number and percentage of internal voxels, number of extreme vertices in the EVM-Rep, ratio between extreme vertices and internal voxels, and number of triangles of the triangle mesh isosurface extracted with MC.

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Int. V.</th>
<th>% Int. V.</th>
<th>EV</th>
<th>(\frac{EV}{intv})</th>
<th>Triangles</th>
<th>(\frac{triangles}{EV})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) stag beetle</td>
<td>411x371x247</td>
<td>1737343</td>
<td>4.6</td>
<td>132184</td>
<td>7.6</td>
<td>567972</td>
<td>23.3</td>
</tr>
<tr>
<td>(b) aneurysm</td>
<td>233x215x240</td>
<td>69743</td>
<td>0.6</td>
<td>50318</td>
<td>72.1</td>
<td>175064</td>
<td>28.7</td>
</tr>
<tr>
<td>(c) daisy</td>
<td>110x113x121</td>
<td>565823</td>
<td>37.6</td>
<td>27570</td>
<td>4.9</td>
<td>125442</td>
<td>22.0</td>
</tr>
<tr>
<td>(d) brain</td>
<td>2300x250x214</td>
<td>435477</td>
<td>40.6</td>
<td>240584</td>
<td>5.5</td>
<td>1000222</td>
<td>29.1</td>
</tr>
<tr>
<td>(e) engine</td>
<td>1391x197x108</td>
<td>901818</td>
<td>30.5</td>
<td>101114</td>
<td>11.2</td>
<td>663900</td>
<td>15.2</td>
</tr>
<tr>
<td>(f) scaffold</td>
<td>132x132x132</td>
<td>367632</td>
<td>16.0</td>
<td>89568</td>
<td>24.4</td>
<td>492480</td>
<td>18.2</td>
</tr>
<tr>
<td>(g) colon</td>
<td>512x512x426</td>
<td>7262520</td>
<td>65.0</td>
<td>1653726</td>
<td>2.3</td>
<td>7862654</td>
<td>21.0</td>
</tr>
<tr>
<td>(h) xmas</td>
<td>459x499x480</td>
<td>801013</td>
<td>0.7</td>
<td>241476</td>
<td>30.1</td>
<td>1574984</td>
<td>15.3</td>
</tr>
<tr>
<td>(i) ball</td>
<td>100x100x100</td>
<td>357869</td>
<td>35.8</td>
<td>32348</td>
<td>9.0</td>
<td>135640</td>
<td>24.2</td>
</tr>
<tr>
<td>(j) rock</td>
<td>313x211x501</td>
<td>4575762</td>
<td>13.1</td>
<td>821012</td>
<td>17.9</td>
<td>607500</td>
<td>13.5</td>
</tr>
<tr>
<td>(k) bunny</td>
<td>432x331x334</td>
<td>288577</td>
<td>5.8</td>
<td>437938</td>
<td>15.2</td>
<td>2053328</td>
<td>21.3</td>
</tr>
<tr>
<td>(l) chest</td>
<td>314x231x240</td>
<td>1249644</td>
<td>7.2</td>
<td>259604</td>
<td>20.8</td>
<td>1211292</td>
<td>21.4</td>
</tr>
<tr>
<td>(m) fertility</td>
<td>232x91x177</td>
<td>862965</td>
<td>23.1</td>
<td>47118</td>
<td>5.5</td>
<td>263200</td>
<td>17.9</td>
</tr>
<tr>
<td>(n) disk brake</td>
<td>511x512x73</td>
<td>2854762</td>
<td>13.5</td>
<td>153498</td>
<td>5.9</td>
<td>1150196</td>
<td>13.3</td>
</tr>
<tr>
<td>(o) neptune</td>
<td>308x512x227</td>
<td>175427</td>
<td>4.9</td>
<td>116680</td>
<td>6.7</td>
<td>570788</td>
<td>20.4</td>
</tr>
<tr>
<td>(p) vase</td>
<td>139x255x139</td>
<td>825336</td>
<td>16.8</td>
<td>101552</td>
<td>12.3</td>
<td>474812</td>
<td>21.4</td>
</tr>
<tr>
<td>(q) venus</td>
<td>148x148x112</td>
<td>3214277</td>
<td>28.7</td>
<td>109184</td>
<td>3.4</td>
<td>559354</td>
<td>19.5</td>
</tr>
<tr>
<td>(r) atenea</td>
<td>350x195x512</td>
<td>11758227</td>
<td>33.6</td>
<td>179400</td>
<td>1.5</td>
<td>989998</td>
<td>18.1</td>
</tr>
<tr>
<td>(s) ramses</td>
<td>347x999x523</td>
<td>45089181</td>
<td>24.9</td>
<td>426330</td>
<td>0.9</td>
<td>2954748</td>
<td>14.4</td>
</tr>
<tr>
<td>(t) hand</td>
<td>511x358x175</td>
<td>1262770</td>
<td>3.9</td>
<td>142828</td>
<td>11.3</td>
<td>654412</td>
<td>21.8</td>
</tr>
</tbody>
</table>
Table 2: For each dataset, its number of positive and negative shells, genus and EC and the computation times in seconds for
the methods using the voxel model, the OUDB/EVM-Reps and the triangle mesh.

<table>
<thead>
<tr>
<th>Name</th>
<th>C⁺</th>
<th>C⁻</th>
<th>genus</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MVoxels</td>
<td>EVM</td>
<td>MVoxels</td>
<td>EVM</td>
</tr>
<tr>
<td>(a)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stagbeetle</td>
<td>17</td>
<td>114</td>
<td>226</td>
<td>-190</td>
</tr>
<tr>
<td>(b)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>aneurysm</td>
<td>406</td>
<td>12</td>
<td>146</td>
<td>544</td>
</tr>
<tr>
<td>(c)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>daisy</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>(d)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>brain</td>
<td>1</td>
<td>542</td>
<td>516</td>
<td>54</td>
</tr>
<tr>
<td>(e)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>engine</td>
<td>9</td>
<td>194</td>
<td>130</td>
<td>146</td>
</tr>
<tr>
<td>(f)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>scaffold</td>
<td>4</td>
<td>0</td>
<td>256</td>
<td>504</td>
</tr>
<tr>
<td>(g)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>colon</td>
<td>11820</td>
<td>11818</td>
<td>91378</td>
<td>-898</td>
</tr>
<tr>
<td>(h)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xmas</td>
<td>1324</td>
<td>40</td>
<td>171</td>
<td>2386</td>
</tr>
<tr>
<td>(i)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ball</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>(j)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rock</td>
<td>2502</td>
<td>26</td>
<td>647</td>
<td>3762</td>
</tr>
<tr>
<td>(k)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bunny</td>
<td>1</td>
<td>155</td>
<td>14</td>
<td>284</td>
</tr>
<tr>
<td>(l)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>chest</td>
<td>1</td>
<td>106</td>
<td>246</td>
<td>-278</td>
</tr>
<tr>
<td>(m)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fertility</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>-6</td>
</tr>
<tr>
<td>(n)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>disk brake</td>
<td>1</td>
<td>0</td>
<td>11</td>
<td>-2</td>
</tr>
<tr>
<td>(o)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>neptune</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>-6</td>
</tr>
<tr>
<td>(p)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vase</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>(q)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>venus</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>(r)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>atenea</td>
<td>1</td>
<td>30</td>
<td>31</td>
<td>0</td>
</tr>
<tr>
<td>(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ramoses</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>-2</td>
</tr>
<tr>
<td>(t)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>hand</td>
<td>1</td>
<td>0</td>
<td>5</td>
<td>-8</td>
</tr>
</tbody>
</table>

The presented algorithms have been written in C++ and tested on a PC Intel Core 2 CPU E6600 at 2.4 GHz with 3.2 GB and running Linux. All reported execution times are in seconds.

We have measured the connectivity in a selection of datasets that encompass a broad range of shape features and occupancy ratios. These datasets where obtained from the following public volume repositories: Aim@Shape, Stanford, S. Roettger, volvis, TU Wien or own collection.

The volume models are usually provided as grayscale voxel models that record the density at every sampling point in a 3D image. Then, a threshold is applied to segment the voxels into two groups: background and foreground voxels.

To obtain a more fair comparison between the tried methods, the initial voxel models have been reduced to match their corresponding minimum bounding box. These resulting datasets present nonmanifold configurations and still contain many isolated cavities and disconnected components.

Figure 7 shows rendered views of the test datasets. These datasets present a diverse topology: some have few components (e.g. fertility) but others have many parts (e.g. xmas) or rock; some have few tunnels (e.g. bunny) and others many of them (e.g. scaffold); some contain visible cavities (e.g. colon) but others do not (e.g. ramoses); some present a simple geometry with few faces (e.g. vase) whilst others are more intricate (e.g. chest).

We consider that the EVM/OUDB representations are available and ignore the cost of conversion from and to a voxel model (which is less than 10 seconds for most datasets and less than 120 s. for the larger ones. Similar times appear for the conversion to triangle meshes using Marching Cubes). Nevertheless, these EVM/OUDB conversion algorithms have also been published (Aguilera and Ayala, 2001; Rodríguez et al., 2004).

Table 1 shows the characteristics of the datasets: the size of the voxel model, the number and percentage of internal voxels, the number of extreme vertices of the corresponding EVM-Rep, the ratio between internal voxels and extreme vertices, the number of triangles of a triangular surface mesh obtained via Marching Cubes and the ratio between triangles and extreme vertices. Note how the equivalent representation of a binary model as an EVM usually provides a large reduction of the number of elements.

Table 2 summarizes the measures of the number of positive and negative shells, genus and EC and the time needed to compute them using the three referenced methods, with the best times highlighted.

The time to compute the $\chi$ value is negligible when equation 4 is applied to a triangle mesh, so this value is not shown in Table 2. Observe that the proposed method is faster than the voxel model reference method, in some cases up to an order of magnitude. The difference is smaller if compared with the triangle mesh method: the computation is simpler in that
case but it has to process a larger number of elements, because a triangle mesh extracted with MC generates a huge number of small triangles.

Also note how the ramses statue, with a boxy shape, has a very compact encoding as an EVM-Rep. On the contrary, the aneurysm dataset produces a quite large EVM, as its internal volume is small and has many thin features. In this case, the execution time for the EVM method is worse. The contrary happens with the xmas tree, which has a very inefficient voxel model representation as it is almost empty. In a nutshell, the proposed method behaves better with very large and compact (not sparse) datasets, such as the set of scanned statues.

6 CONCLUSIONS AND FUTURE WORK

In this paper we have presented a method to compute the Euler characteristic and the genus of a volume dataset using two alternative models, EVM-Rep and OUDB-Rep, and we have analyzed their performance compared to the reference methods applied to voxel models and triangle meshes.

We have tested several public volume datasets. We conclude that the connectivity computation in general is better in the new presented approach with these test datasets. This improvement is much noticeable in the complete computation than in the $g$ sub-procedure. The divergence in execution times can be traced to the size of the datasets but above all to the intricacy of the represented objects: the method for voxel models is almost constant time in relation to the number of voxels, the triangle mesh method mainly depends on the number of triangles and the proposed method depends on the number of extreme vertices of the dataset, with these two latter numbers directly proportional to the surface tortuosity of the model.

In the biomedical field, there are many other structural parameters worth to be studied as they describe the properties of a biomaterial. We have already developed methods (Vergés et al., 2008) for obtaining a pore network. We are now studying other structural parameters such anisotropy.

We also intend to evaluate temporal sequences and gray-scale datasets representing real samples with voxel as well as EVM and OUDB models.

ACKNOWLEDGEMENTS

This work has been partly funded by the CICYT project TIN2008-02903 of the Spanish government.

REFERENCES


