01 Jan 2004

BIUP3: Boundary Topological Invariant of 3D Objects Through Front Propagation at a Constant Speed

Xiaoqing Frank Liu

Missouri University of Science and Technology, fliu@mst.edu

BIUP^3: Boundary Topological Invariant of 3D Objects Through Front Propagation at a Constant Speed

Franck Xia
Department of Computer Science; University of Missouri-Rolla
Rolla, MO 65409, USA; E-mail: xiaf@umr.edu

Abstract: Topological features constitute the highest abstraction in object representation. Euler characteristic is one of the most widely used topological invariants. The computation of the Euler characteristic is mainly based on three well-known mathematical formulae, which calculate either on the boundary of object or on the whole object. However, as digital objects are often non-manifolds, none of the known formulae can correctly compute the genus of digital surfaces. In this paper, we show that a new topological surface invariant of 3D digital objects, called BIUP^3, can be obtained through a special homeomorphic transform: front propagation at a constant speed. BIUP^3 overcomes the theoretic weakness of the Euler characteristic and *it applies to both manifolds and non-manifolds*. The computation of BIUP^3 can be done efficiently through a virtual front propagation, leaving the images unaffected.

Keywords: Topological invariant, digital topology, boundary, front propagation, topological boundary invariant.

1. Introduction

One of the key issues in computer imagery is to find an appropriate representation of objects to be processed. Mathematically, the objects are fully determined by their boundaries, and topological features of boundaries constitute the highest abstraction of the objects [1]. Euler characteristic is one of the most frequently used topological invariants in various fields involving computer images. Although the Euler characteristic has been widely used for decades, it has its own limitations: None of known mathematical theorems about the Euler characteristic could provide a satisfactory solution for describing topological features, i.e. genus, of digital surfaces. The reason is that digital images are often non-manifolds and there is no known mathematical theorem that can determine the topological invariants, such as genus, of boundaries of non-manifolds. So it would be interesting to discover new and better topological invariants of boundaries.

However, new topological invariants are very difficult to come by. The Euler characteristic has been known for centuries. Fortunately, there is a good point to start: In [2], a 2D topological invariant called perimeter increment under dilation has been proven. By this invariant, the contours of *closed* objects have a topological property under dilation with the unit disc: the external contours increase and internal contours decrease, and the increment in both cases is the perimeter of the unit disc. This result can be generalized to R^2 via a continuous front propagation and a topological invariant, called boundary invariant under propagation (BIUP^2), can be obtained. In this paper, we show that BIUP^2 can further be generalized to 3D. The obtained new 3D invariant, called BIUP^3, characterizes the topological property of surfaces of digital objects and overcomes the weakness of the Euler characteristic. In the sequel, we first discuss the limitations of known formulae about the Euler characteristic. The notion and properties of BIUP^3 are introduced in section 3. Section 4 briefly describes an algorithm for computing BIUP^3. We present in the end our result with a brief analysis on the strength of BIUP^3.

2. Limitations of the Euler Characteristic

For any connected component S, the Euler characteristic can be defined either on the whole component S (denoted by $\chi(S)$) or over its boundary ∂S ($\chi(\partial S)$). When defining $\chi(S)$ over the whole component via v – e + f - t, where v is the number of vertices, e the number of edges, f of faces and t of tetrahedrals, by Euler-Poincaré formula $\chi(S)$ = (no. of connected components)) + (no. of cavities) – (no. of handles) [1]. When S contains cavities, the boundary of S has one external surface and one or more internal surfaces. In the Euler-Poincaré formula, handles are on the boundary and hence could be on either the external surface or on the internal surfaces of the cavities. As $\chi(S)$ is intrinsically defined on the whole component, it cannot tell where a handle locates, i.e. on the external or an internal surface. Thus $\chi(A)$ cannot fully differentiate topologically distinct objects.

Alternatively, the Euler characteristic can also be defined over ∂S, i.e. $\chi(\partial S) = v - e + f$. For any manifold S, when calculating over its surface, it is well-known that $\chi(\partial S) = 2(1 - g)$, g being the genus of the surface [3]. However, this formula does not hold when S is non-manifold[§]. In Fig. 1.a, the object has no handle and hence g = 0. However, as v = 14, e = 23, and f = 12, $\chi(\partial S) = 3 \neq 2(1 - g)$. It is easy to verify that the same problem exists with all the other non-manifold objects in Fig. 1. Obviously, $\chi(\partial S)$ calculated in this way is no longer a topological invariant. It is worth mentioning that non-manifolds contain topological singularities and they are very difficult to analyze mathematically. Yet non-manifolds are everywhere in computer images [4].

[§] An object C is a manifold if $\forall x \in C$, there exists a neighborhood N(x) of x such that N(x) is homeomorphic to the unit ball.

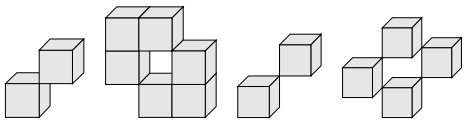

Figure 1: Non-manifold digital objects

Gauss-Bonnet Theorem in differential geometry connects the total Gaussian curvature κ_G with the Euler Characteristic: $\oint_{\partial S} \kappa_G(\sigma).d\sigma = 2\pi.\chi(\partial S)$ [5]. As κ_G is defined on regular surfaces whereas digital surfaces are intrinsically discrete and irregular, it is impossible to accurately calculate κ_G based on its analytical form in digital images [6], let alone the total Gaussian curvature $\oint_{\partial S} \kappa_G(\sigma).d\sigma$. It has been suggested that when defining angle deficit of vertex v_i of ∂S (denoted by $ad(v_i)$) to be 2π minus the sum of face angles incident to v_i, then $(1/2\pi)\Sigma_i ad(v_i) = 2(1-g)$ [7]. However, one can verify that this formula is again invalid when objects are non-manifolds (Fig. 1.c or 1.d).

So there is no theoretically valid method for calculating the topological feature, i.e. genus, of surfaces. Thus there is a theoretic and practical interest for us to explore new and better invariants that can topologically distinguish the surfaces of digital objects.

3. BIUP: Boundary Invariant under Propagation in R^3

3.1 Definition of BIUP^2

The boundary topological invariant we propose is a generalization of our previous work on 2D global boundary invariant [2]. The mathematical idea of our approach has something in common with the level set methods which investigate the properties of boundaries or fronts under propagation in the normal direction [8]. In order to obtain topological invariant, we choose a constant propagation velocity v and focus on the integral property of the propagating front. For any connected component S in R^2 with a regular boundary/front B(S), we consider the first order derivative of the total curve length of B(S) in the normal direction. Let x be $(x_1(s),x_2(s))$, the front of the connected component at time t be B(x,t), the perimeter of the front at t be $P(t) = \oint_{B(t)} \sqrt{(\frac{\partial B(x,t)}{\partial x_1}\frac{\partial x_1}{\partial s})^2 + (\frac{\partial B(x,t)}{\partial x_2}\frac{\partial x_2}{\partial s})^2}\,ds$, and that of the front at $t+\Delta t$ after propagation be $P(t+\Delta t) = \oint_{B(t+\Delta t)} \sqrt{(\frac{\partial B(x,t+\Delta t)}{\partial x_1}\frac{\partial x_1}{\partial s})^2 + (\frac{\partial B(x,t+\Delta t)}{\partial x_2}\frac{\partial x_2}{\partial s})^2}\,ds$. We define boundary invariant under propagation in R^2 (BIUP^2) as $\lim_{\Delta t\to 0} \frac{P(t+\Delta t)-P(t)}{v\Delta t}$ and discover that BIUP^2 $= \pm 2\pi$, + for external and – for internal boundary.

3.2 Definition of BIUP^3

For the 3D case, let us denote A(t), A(t+Δt), and A(t+2Δt) the area of propagating boundary B at t, t+Δt, and t+2Δt (Fig. 2). For various shapes, our experiments show that $\lim_{\Delta t\to 0} \frac{[A(t+2\Delta t)-A(t+\Delta t)]-[A(t+\Delta t)-A(t)]}{(v\Delta t)^2}$ has the same invariant property as $\lim_{\Delta t\to 0} \frac{P(t+\Delta t)-P(t)}{v\Delta t}$ in R^2.

Definition: $\lim_{\Delta t\to 0} \frac{[A(t+2\Delta t)-A(t+\Delta t)]-[A(t+\Delta t)-A(t)]}{(v\Delta t)^2}$ is called boundary invariant under propagation in R^3 (**BIUP^3**).

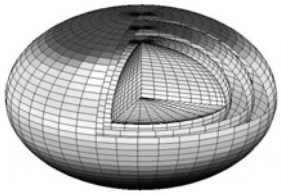
Figure 2: Front propagation in R^3

3.3 Computing BIUP^3 in Z^3

3.3.1 Localization of Propagation

For digital images in Z^3, we choose to work with the cellular or cubic representation [1] with which each voxel is represented by the unit cube. An explicit simulation of front propagation can then be done based on a discrete version of Huygens' principle that we propose here: placing the center of a small cube on the surface and rolling it over the surface. The envelope of the rolling cubes forms A(t+Δt).

We propose an efficient way for computing BIUP^3 which does not need any front propagation. In fact, the front propagation can be done locally and virtually: First divide the whole boundary into elementary boundary patches. For each patch, we virtually propagate the patch and calculate the second order increment of the patch. Accumulate then the increments patch by patch to obtain [A(t+2Δt) – A(t+Δt)] – [A(t+Δt) – A(t)], and finally compute the derivative by definition.

Definition: Let each boundary surfel (surface of the unit cube) of B(x,t) be divided into four equal area squares called ¼ boundary surfels. The *elementary boundary patch* around vertex v_i on B(x,t) (denoted as $\Delta B(v_i,t)$) is the set of all the ¼ boundary surfels adjacent to v_i.

The white regions in Fig. 3 illustrate three elementary boundary patches around the central vertices.

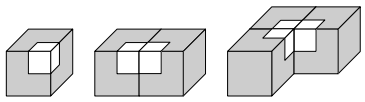

Figure 3: Elementary boundary patches

Proposition: $\cup_i \Delta B(v_i,t) = B(x,t)$. ■

Denote I(A) as the interior of set A.

Proposition: $\forall i \neq j$, $I(\Delta B(v_i,t)) \cap I(\Delta B(v_j,t)) = \varnothing$. ■

The above two propositions ensure that the whole boundary can be covered by all the elementary boundary patches without any redundancy. Denote the area of $\Delta B(v_i,t)$ by $\Delta A(v_i,t)$ and $\Delta A(v_i,t+2\Delta t) + \Delta A(v_i,t) - 2\Delta A(v_i,t+\Delta t)$ by $\partial^2 \Delta A(v_i,t)$.

Proposition: $\Sigma_i \Delta A(v_i,t) = A(t)$. ■

Proposition: $\Sigma_i \partial^2 \Delta A(v_i,t) = [A(t+2\Delta t) - A(t+\Delta t)] - [A(t+\Delta t) - A(t)]$. ■

Fig. 4 illustrates all the possible $\Delta B(v_i,t)$ under 6-, 18-, and 26-connectedness [1], all the others can be obtained by symmetry or rotation. The black dot in each pattern represents the vertex v_i.

3.3.2 Virtual Front Propagation

Let the distance of propagation be $\delta = v\Delta t$. The key for computing $\partial^2 \Delta A(v_i,t)$ is to determine $\Delta A(v_i,t+\Delta t)$ which depends not only on $\Delta B(v_i,t)$ but also on δ. Once $\Delta A(v_i,t+\Delta t)$ in known, substituting any δ in $\Delta A(v_i,t+\Delta t)$ by 2δ, we get $\Delta A(v_i,t+2\Delta t)$.

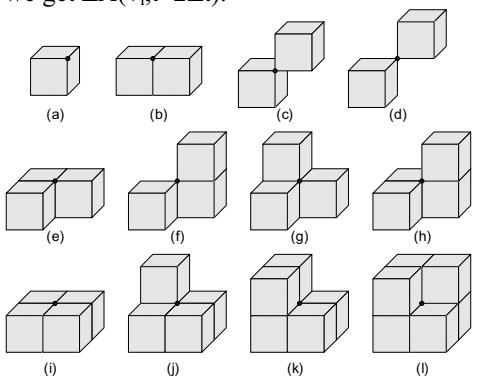

Figure 4: Shapes of $\Delta B(v_i,t)$ under various connectedness

In Fig. 5, dark areas represent $\Delta B(v_i,t)$s and the white ones the corresponding propagated boundary patches $\Delta B(v_i,t+\Delta t)$s obtained by propagation from $\Delta B(v_i,t)$s in the direction of normal depicted by the arrows.

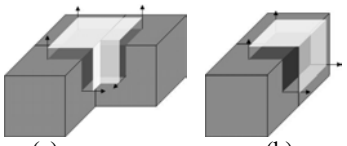

Figure 5: Computing propagation locally

Table 1 provides the value of $\partial^2 \Delta A(v_i,t)$ for each patch in Fig. 4. For example, the value of the patch centered at the black dot in Fig.4.a is $6\delta^2$in column #2 of the Table.

$\Delta B(v_i,t)$	a	b	c	d	e	f	g	h	i	j	k	l
$\partial^2 \Delta A(v_i,t)$	$6\delta^2$	0	$-12\delta^2$	$-36\delta^2$	$-6\delta^2$	$-18\delta^2$	$-12\delta^2$	$-12\delta^2$	0	$-6\delta^2$	0	$6\delta^2$

Table 1: The Values of $\partial^2 \Delta A(v_i,t)$ for all the possible $\Delta B(v_i,t)$s.

3.3.3 Property of BIUP^3

Proposition: Any digital object homeomorphic to a solid ball can be deformed homeomorphically and repeatedly to a unit cube (Fig. 6.a) by removing/adding one cube at a time.

Proposition: Any digital object homeomorphic to a torus can be deformed homeomorphically and repeatedly to a digital ring (Fig. 6.b) by removing/adding one cube at a time.

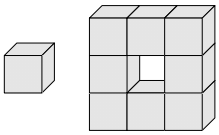

Figure 6: Basic patterns

Let BIUP$^3(\partial 1)$ be the BIUP^3 on the surface of the unit cube.

Theorem: BIUP$^3(\partial S)$ = BIUP$^3(\partial 1).(1\text{-}g)$[§].

Due to page limitation, we provide only a sketch of our proof done by induction. We illustrate only two cases: objects homeomorphic to a solid ball and a torus using 6-connectedness [1]. Figure 6 depicts the basic shapes of these two object types. For the basic step, it is easy to verify that for Fig.6.a, $\Sigma_i \partial^2 \Delta A(v_i,t) = 8 \times 6\delta^2$, hence BIUP$^3(\partial 1) = \lim_{\Delta t \to 0} \Sigma_i \partial^2 \Delta A(v_i,t)/\delta^2 = 48$. For Fig.6.b BIUP$^3(\partial S) = 0$ = BIUP$^3(\partial 1).(1\text{-}1)$. The induction step of the proof is combinatorial. Any object S homeomorphic to a solid ball can be deformed homeomorphically and repeatedly to the unit cube by removing one cube at a time. Based all the patterns shown in Fig. 4, and for any fixed propagation distance δ, each time when we remove

[§] Here ∂S should be understood as one surface of ∂S.

any cube from any of the patterns in Fig. 4, the sum of all the elementary boundary patches $\Sigma_i \partial^2 \Delta A(v_i,t)$ around the cube remains unchanged before and after removing the cube. As the rest of B(S) is not affected by the removed cube, $[A(t+2\Delta t) - A(t+\Delta t)] - [A(t+\Delta t) - A(t)]$ remains constant when the size of S is reduced by one. Since $BIUP^3(\partial 1) = 48$ for Fig. 6.a, $BIUP^3(\partial S) = 48$ for any 6-connected object homeomorphic to the solid ball. ■

4. Algorithm for Computing $BIUP^3$

For computing BIUP3, our algorithm reads 3D raster images and traces the boundary of any connected components in the images using an existing algorithm [9]. During the boundary tracing, we save for each vertex the information about its own elementary boundary patch such as the number of surfels and the number of neighboring cubes. The computation of $\partial^2 \Delta A(v_i,t)$ is done through a look-up table implemented in the algorithm. Currently, the algorithm works for both 6 and 18 connectedness. Part of our algorithm for computing $BIUP^3$ is shown below (not including the standard surface tracing)

```
Procedure BIUP3( input )
  while (input is not empty)
    remove face f from input;
    for each vertex vx on f
     V(vx) = V(vx) + 1
    end of for
  end of while

  for each vx in V
    switch V(vx)
     case 12
       V(vx) = 24
     case 9
       if no of cubes shared by vx == 3
         V(vx) = -6
       else V(vx) = 18
     case 8
       V(vx) = 0
     case 7
       if no of cubes shared by vx == 3
         V(vx) = -18
       else V(vx) = 6
     case 6
       if no of cubes shared by vx == 3
         V(vx) = 12
       else V(vx) = -12
     otherwise
       V(vx) = 6 * (4 - mv(x, y, z))
    end of switch
  end of for
  return sum(V(vx))
end of Procedure BIUP3
```

5. Result and Analysis

We have implemented our algorithm with Matlab and tests of our method with various 3D images have confirmed the theorem. For instance, the object S in Fig. 7 has 33 holes. $BIUP^3(\partial S) = -1536 = 48(1-33) = BIUP^3(\partial 1).(1-g)$.

In terms of computation effort, as the algorithm works only on the surfaces of object S, its complexity is O(N), N being the number of surface points on S. In comparison, the computation of $\chi(S)$ requires O(|S|), |S| being the number of points in S. Obviously, N < |S| significantly.

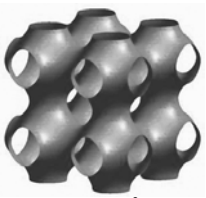

Figure 7: $BIUP^3 = -1536$

$BIUP^3$ correctly works on any kind of images, including non-manifolds for which $\chi(\partial S)$ is no longer a topological invariant. $BIUP^3$ also enables us to compute the genus of surfaces, one by one, hence a topological representation of objects based on surfaces can be obtained. This is not possible with the Euler characteristic $\chi(S)$.

6. Conclusion

From the history of mathematics, we know that discovering global topological invariants is not an easy task. The one we use frequently in computer science, Euler characteristic, has been with us for centuries. In this paper, we show how the basic concept of level set methods, i.e. successive propagation of front, can be combined with the fundamental notion of homeomorphic transform in topology for discovering a new topological invariant, i.e. $BIUP^3$, which can distinguish different types of surfaces: It determines the number of handles on surfaces, and it works for either manifolds or non-manifolds. We are currently working on how to efficiently build a topological representation of 3D objects. Our investigation indicates that a generalization of BIUP to higher dimensions could lead to similar topological invariants.

Reference:

1. T.Y. Kong & A. Rosenfeld, Digital topology: introduction and survey, CVGIP, 48, 1989, 357-393
2. F. Xia, Invariant Property of Contour: PIUD for Closed Sets and VPIUD for Arbitrary Images, 9th Scandinavian Conf. on Image Analysis, 1995, vol. 1, 103-112
3. H. Edelsbrunner, Geometry and Topology for Mesh Generation, Cambridge Uuiv. Press, 2001
4. A. Guéziec, F. Bossen, G. Taubin, & C. Silva, Efficient compression of non-manifold polygonal meshes, Visualization'99, 1999, 73-80
5. M. De Carmo, Differential Geometry of Curves and Surfaces, Prentice Hall, 1976
6. Trucco E. & Fisher R.B., Experiments in curvature-based representation of object from range data, IEEE T-PAMI, 17(2), 1995, 177-182
7. L. Alboul & R. van Damme, Polyhedral metrics in surface reconstruction, Mathematics of Surfaces VI, 1996, 171-200
8. S. Osher & R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer, 2003
9. J.K. Udupa & V.G. Ajjanagadde, Boundary and object labeling in 3-dimensional images, CVGIP, 51, 1990, 355-369