

## Research Article

# An Approach to the Computation of the Euler Number by means of the Vertex Chain Code

Ernesto Bribiesca <sup>1</sup>, Ulf-Dietrich Braumann <sup>2,3,4,5</sup>, Angel Carrillo-Bermejo,<sup>6</sup>  
and Humberto Sossa-Azuela <sup>7</sup>

<sup>1</sup>Departamento de Ciencias de la Computación, Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, Apdo. 20-726, Mexico, D.F. 01000, Mexico

<sup>2</sup>Interdisziplinäres Zentrum für Bioinformatik, Universität Leipzig, Leipzig, Germany

<sup>3</sup>Institut für Medizinische Informatik, Statistik und Epidemiologie, Universität Leipzig, Leipzig, Germany

<sup>4</sup>Fakultät Ingenieurwissenschaften, Hochschule für Technik, Wirtschaft und Kultur Leipzig, Leipzig, Germany

<sup>5</sup>Fraunhofer-Institut für Zelltherapie und Immunologie, Leipzig, Germany

<sup>6</sup>Posgrado en Ciencia e Ingeniería de la Computación, Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, Apdo. Postal 20-726, Mexico, D.F. 01000, Mexico

<sup>7</sup>Centro de Investigación en Computación, Instituto Politécnico Nacional, Av. Juan deDios Bátiz s/n, Mexico, D.F., Mexico

Correspondence should be addressed to Ernesto Bribiesca; [bribiesca@iimas.unam.mx](mailto:bribiesca@iimas.unam.mx)

Received 9 May 2020; Accepted 22 June 2020; Published 16 July 2020

Academic Editor: Nadia A. Chuzhanova

Copyright © 2020 Ernesto Bribiesca et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present an approach to compute the number of holes in binary images using the Vertex Chain Code (VCC); the VCC was developed for representing and analyzing 2D shapes composed of cells. Using this code, it is possible to relate the outer to inner vertices of any 2D shape and to find interesting properties. Now, in this paper, we describe more properties of the VCC, such as the computation of the connected regions in a hole, the analysis of complementary chains, the computation of the number of holes in a binary shape or image, the computation of the Euler number, and the detection of convex and concave shapes. Finally, in order to illustrate the capabilities of proposed methods, we present the computation of topological properties of examples of objects of the real world.

## 1. Introduction

This work focuses on image representation by means of the Vertex Chain Code (VCC) [1], in order to obtain different properties of binary shapes, such as the computation of the connected regions in a hole and the computation of the Euler number. Also, the study of the geometrical and topological properties of binary shapes is an important topic in computer vision and pattern recognition. The Euler number or genus is a basic topological property of shapes. The Euler number is a fundamental feature in many applications: in character recognition, in graph theory, in the computation of porosity of objects, in biological imaging, in the recognition of industrial parts [2], and so forth.

Mathematically speaking, the Euler number  $E$  or genus is defined as the number of connected regions  $C$  minus the number of holes  $H$ , i.e.,  $E = C - H$ . Many authors have proposed different methods to compute this important topological descriptor, for example, in Refs. [2–13].

In order to compute the Euler number, some authors have used the VCC [1]. For instance, Wulandhari and Haron [14] proposed an interesting approach to compute the Euler number by means of the chain elements. They analyzed shapes with only one hole and obtained two VCC chains of the outer and inner boundaries of the analyzed shape. Afterwards, they add all chain elements and compare with the number of elements of both chains multiplied by two. Recently, Sossa-Azuela et al. [15] presented an interesting

approach to compute the Euler number by means of the VCC. They count the number of both “1” and “3” vertices and compute the number of holes by a simple formula. In Ref. [16], the authors propose two equations based on the pixel geometry and connectivity properties, which can be used to efficiently compute the Euler number of a binary digital image with either thick or thin boundaries. Despite computing this feature, the authors’ technique extracts the underlying topological information provided by the shape pixels of the given image. The correctness of computing the Euler number using the new equations is also established theoretically. The performance of the proposed method is compared against other available alternatives. Experimental results on a large image database demonstrate that the authors’ technique for computing the Euler number outperforms the earlier approaches significantly in terms of the number of basic arithmetic operations needed per pixel. Both equations are specialized only for 4-connectivity cases.

In order to present our approach to the computation of the Euler number, we use some concepts of the VCC. The VCC of a 2D shape composed of side-connected pixels is based on the number of pixel vertices which are in touch with the bounding contour of the shape. Among the main properties of the VCC, we can mention the following: (1) The VCC elements represent real values not symbols such as other chain codes, are part of the shape, and indicate the number of pixel vertices of the contour nodes, for instance, a chain which has no elements “3” represents a convex shape. (2) Using the VCC, it is possible to represent shapes composed of triangular, rectangular, i.e., squared (pixels), and hexagonal cells. (3) The VCC is invariant under translation and rotation and optionally may be invariant under starting point and mirroring transformation. (4) In order to define a chain element, most chain codes use information of the previous ones (mainly, the chain codes which are invariant under rotation). Thus, if a chain element is missing, next chain elements are affected. On the other hand, VCC is more robust to missing data because each chain element does not depend on a previous one.

In the content of this paper, Definition 9 and Theorem 8 are based on Ref. [1], while Theorems 13 and 14 are based on Ref. [15]. On the other hand, Definitions 1, 2, 10, and 11, Theorems 3, 4, 5, 7, 15, and 17, and Corollaries 6, 12, and 16 are the contribution of this paper.

In this work, we present the computation of the connected regions in a hole, the analysis of complementary chains, the computation of the number of holes in a binary shape or image, the computation of the Euler number, and the detection of convex and concave shapes. This paper is organized as follows. In Section 2, we present some concepts and definitions. Section 3 describes shapes without holes. Section 4 gives an analysis for shapes with holes. In Section 5, we describe the computation of the Euler number via the VCC. Section 6 presents some results of the computation of topological features of objects of the real world and a comparison of the proposed techniques with other previous methods. Section 7 shows computational comparative analyses. Finally, in Section 8, we give some conclusions.

## 2. Concepts and Definitions

In order to introduce our approach to the computation of the number of holes in binary images via the VCC, a number of concepts and definitions are presented below:

- (i) *Shape* is referred to as shape-of-object, and an object is considered to be a geometric entity composed of side-connected pixels [17], i.e., pixels with four-connectivity. There are two ways of connecting pixels: *four-connectivity* and *eight-connectivity*. In the content of this paper, we only consider pixels with four-connectivity
- (ii) *Binary shape* is a shape in which each pixel takes either the value zero or the value one [18]. Generally speaking, in the content of this paper, shape is referred as binary shape
- (iii) The *length* of all the sides of pixels is considered equal to one
- (iv) A *chain*  $A$  is an ordered sequence of elements and is represented by  $A = a_1 a_2 \cdots a_n = \{a_i : 1 \leq i \leq n\}$ , where  $n$  indicates the number of chain elements
- (v) An *element*  $a_i$  of a chain indicates the number of pixel vertices, which are in touch with the bounding contour of the shape in that element position

An important consideration is the assumption that an *entity* has been isolated from a digital image of the real world. This is called the *shape* and is defined as a result of previous processing. Shapes are composed of side-connected pixels, i.e., pixels with four-connectivity. In order to obtain the VCC of a 2D shape, Figure 1 illustrates an example of a discrete 2D shape. Figure 1(a) shows the abovementioned example of a shape composed of pixels. Figure 1(b) presents the chain elements and its chain of the shape illustrated in (a). The origin of the chain is represented by a circle in the abovementioned shape, and the arrow indicates the direction of the encoding. Note that there are only three different numbers of pixel vertices for the bounding contour: “1,” “2,” and “3.”

## 3. Shapes without Holes

In this subsection, we present an analysis for 2D shapes without holes by means of the VCC.

*Definition 1.* The outer vertices  $V_1$ ,  $V_2$ , and  $V_3$  of a shape correspond to the number of vertices “1,” “2,” and “3” (respectively) of the chain which represents the bounding contour of the shape.

The numbers of the outer vertices of the shape shown in Figure 1(b) are  $V_1 = 11$ ,  $V_2 = 6$ , and  $V_3 = 7$ .

*Definition 2.* The inner vertices  $V_4$  of a shape correspond to the number of vertices “4” with the largest number of pixel vertices. These vertices do not belong to the vertices of the

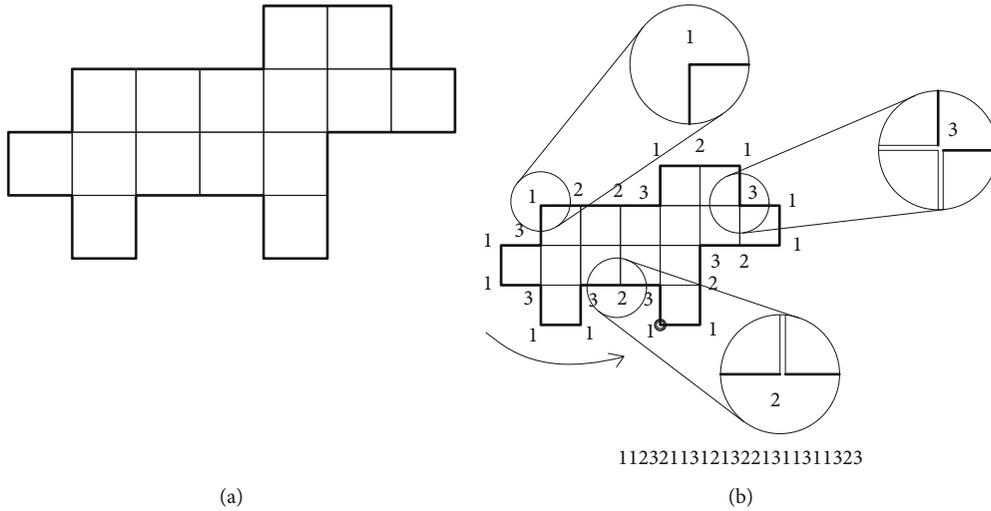


FIGURE 1: The VCC: (a) an example of a discrete 2D shape composed of side-connected pixels; (b) the chain elements defined by means of the VCC and its chain.

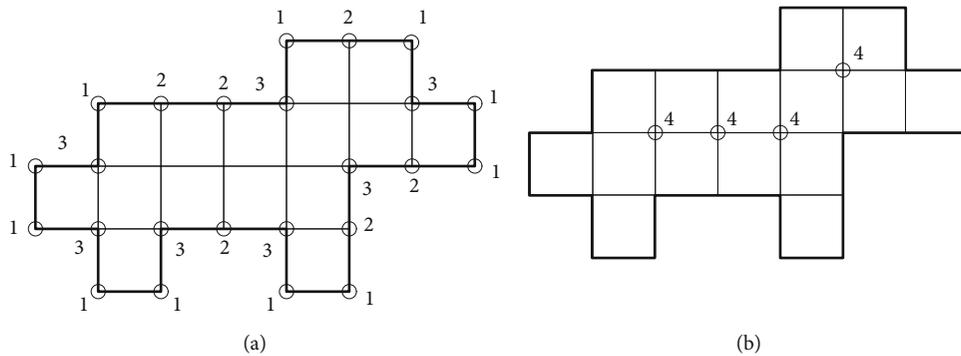


FIGURE 2: Vertices: (a) outer vertices of the shape illustrated in Figure 1; (b) inner vertices of the shape presented in Figure 1.

bounding contour; they belong to the interior of the shape and always have four neighbor pixels.

The number of the inner vertices of the shape illustrated in Figure 1(b) is equal to 4, i.e.,  $V_4 = 4$ . Figure 2 shows the outer (Figure 2(a)) and inner vertices (Figure 2(b)) of the shape presented in Figure 1.

**Theorem 3.** For any discrete shape composed of  $m$  pixels. The following equation is satisfied:

$$V_1 + 2V_2 + 3V_3 + 4V_4 = 4m, \tag{1}$$

*Proof.* There are four possible cases of vertices: the outer vertices “1,” “2,” and “3” and the inner vertices “4” (see Figure 2). For the base case, when the shape has only one pixel we have  $m = 1$ ,  $V_1 = 4$ ,  $V_2 = 0$ ,  $V_3 = 0$ , and  $V_4 = 0$ . By direct calculation, using Equation (1), we have

$$4 = 4. \tag{2}$$

For the next pixel added and depending on the four possible cases, Equation (1) is satisfied.

For example, substituting the values of the outer and inner vertices of the shape shown in Figure 1(a), where the number of pixels  $m$  is equal to 15, Equation (1) is satisfied. Using Equation (1), the number of inner vertices  $V_4$  is defined as follows:

$$V_4 = \frac{4m - (V_1 + 2V_2 + 3V_3)}{4}. \tag{3}$$

**Theorem 4.** The perimeter  $P$  of a shape without holes is equal to the sum of the numbers of its outer vertices, i.e.,

$$P = V_1 + V_2 + V_3. \tag{4}$$

*Proof.* Trivial.

For example, the perimeter of the shape shown in Figure 1(a) is equal to 24.

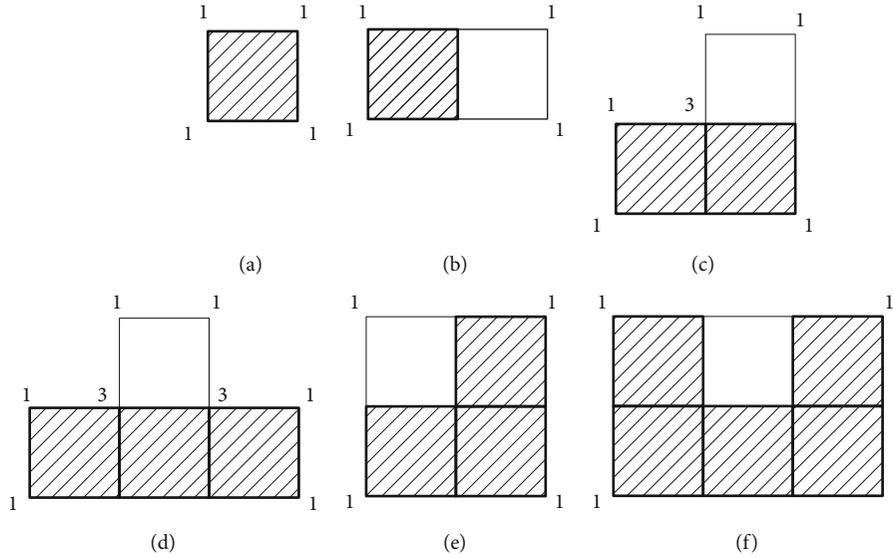


FIGURE 3: Proof of Theorem 7: (a) the basic case; (b–f) all possible cases.

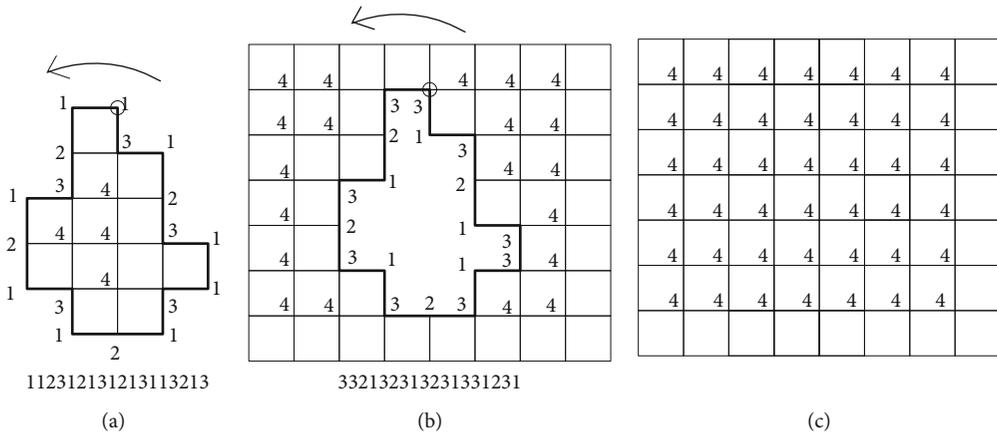


FIGURE 4: Complementary chains: (a) an example of a 2D shape and its outer vertices; (b) the complementary chain of the shape shown in (a). Now, this shape corresponds to a hole; (c) the set  $\vee$ , i.e.,  $G \cup G' = \vee$ .

**Theorem 5.** For any discrete shape composed of  $m$  pixels. If  $V3 = 0$ , then the shape is convex.

*Proof.* Trivial.

A convex shape is a shape which lies entirely in one-half plane of any line containing one of its sides; otherwise, it is concave [19].

**Corollary 6.** For any discrete shape composed of  $m$  pixels. If  $V3 \neq 0$ , then the shape is concave.

*Proof.* Trivial.

For example, using Corollary 6, the shape shown in Figure 1 corresponds to a concave shape.

**Theorem 7.** For any discrete shape (without holes) composed of pixels, the number of vertices  $V1$  is equal to the number of vertices  $V3$  plus 4, i.e.,

$$V1 = V3 + 4. \tag{5}$$

*Proof.* Any chain must have at least four elements “1”; this is due to the fact that only closed shapes are considered and they are composed of pixels. For the base case, when the shape has only one pixel, we have  $V1 = 4$  and  $V3 = 0$ . Therefore, Equation (5) is satisfied. For the next steps, we have to add more pixels. There are only five possible ways to add pixels for generating no hole in the analyzed shape: Figure 3(a) shows the most primitive shape only composed of one pixel. Figure 3(b) presents the first case; the added pixels did not change the variables  $V1$  and  $V3$ . Figure 3(c) illustrates the second case, in which  $V1$  was incremented by

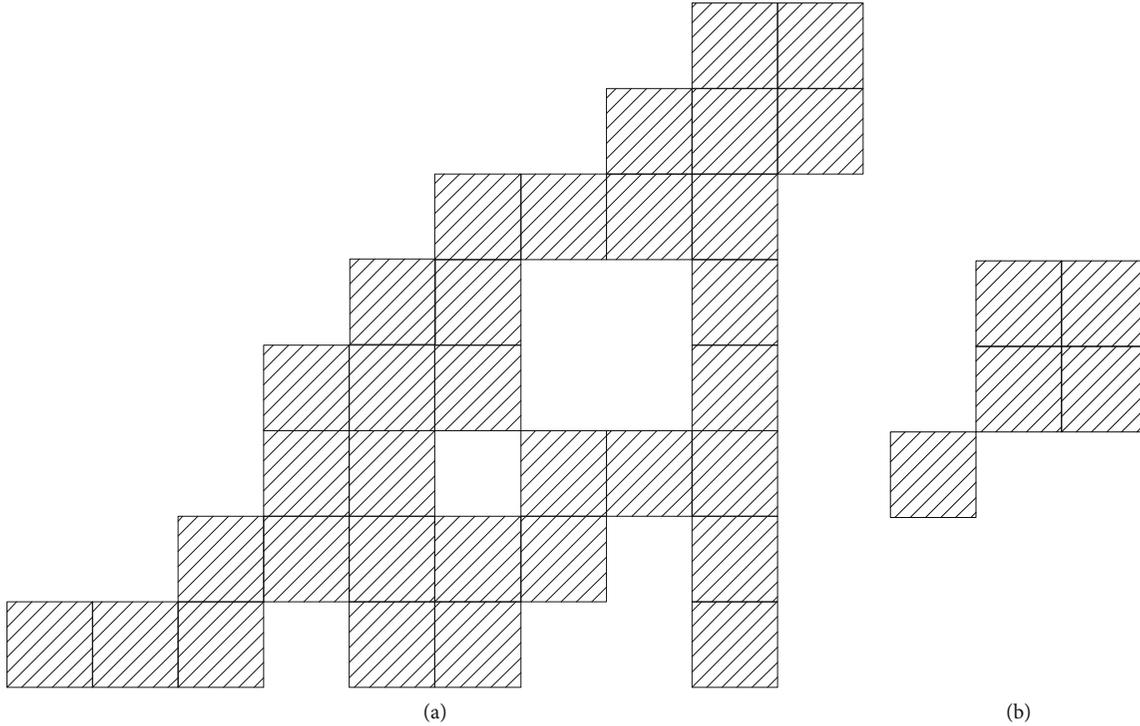


FIGURE 5: An invalid case: (a) an example of a 2D shape; (b) the inner complement (hole) of the shape shown in (a) which has no four-connectivity.

1 and V3 was incremented by 1, too. Figure 3(d) shows the third case, where both variables V1 and V3 were incremented by 2. Figure 3(e) presents the fourth case, where V1 and V3 were decremented by 1. Finally, Figure 3(f) illustrates the fifth case, where both variables V1 and V3 were decremented by 2. Thus, Equation (5) is satisfied for all cases, which we know is true.

Yip and Klette [20] presented angle counts for isothetic polygons and polyhedra. Thus, any isothetic simple polygon can be partitioned into a finite number of isothetic rectangles. In our case, our approach only considers discrete shapes composed of pixels.

We present an important theorem, which relates the sum of the chain elements to chain length (number of chain elements) for shapes (without holes) composed of pixels. This theorem was presented in [1]; however, in order to describe our proposed approach, this theorem should be considered.

**Theorem 8.** *For any shape (without holes) composed of pixels. The sum of the chain elements of the shape is equal to  $2n - 4$ , i.e.,*

$$\sum_{i=1}^n a_i = 2n - 4. \tag{6}$$

The proof of this theorem is not included.

#### 4. Shapes with Holes

*Definition 9.* The complement of a chain of a shape composed of pixels is another chain (named complementary chain) whose elements “1” are replaced by “3” and vice versa.

Figure 4(a) shows a shape and its chain. Figure 4(b) illustrates the complementary chain of the chain presented in (a). Note that now the shape of the complementary chain is a hole preserving the same contour.

*Definition 10.* The complementary chain of a shape corresponds to a hole, in which the contour of the original object is preserved.

For example, see Figure 4.

*Definition 11.* Let the set  $G$  be the binary image  $I(r, c)$ ,  $G'$  the complement of set  $G$ , and  $I'(r, c)$  the complement of image  $I(r, c)$ , respectively. Then,  $G \cup G' = \vee$ .

Figure 4 represents binary images as sets. Thus, if the binary image shown in Figure 4(a) corresponds to the set  $G$ , then the binary image described in Figure 4(b) corresponds to the complement of set  $G$ , represented by  $G'$ . Finally, Figure 4(c) represents  $G \cup G' = \vee$  (in set theory).

**Corollary 12.** *For any hole of a discrete shape composed of pixels, the number of vertices V3 is equal to the number of vertices V1 plus 4, i.e.,*

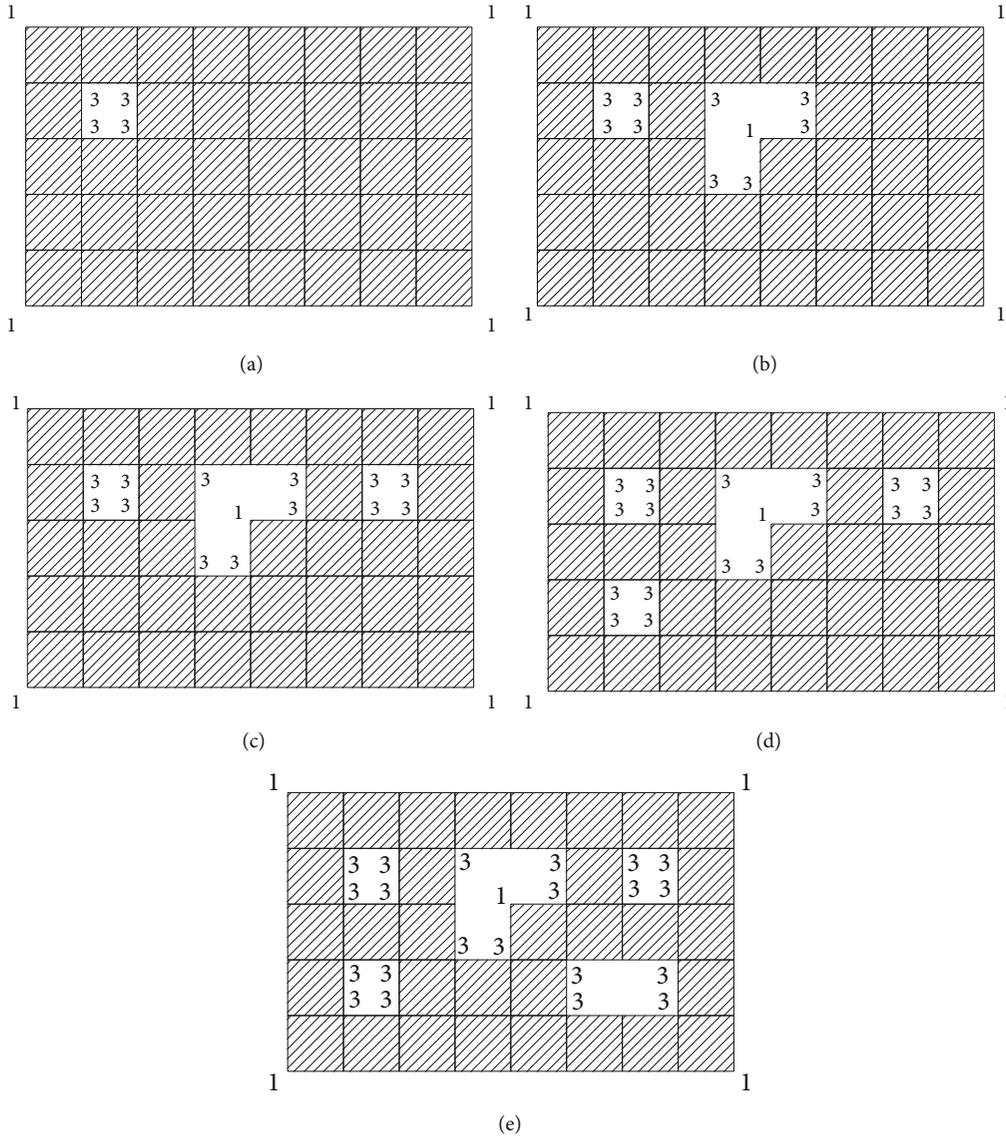


FIGURE 6: Proof of Theorem 13: (a) a binary shape with one hole; (b–e) the shape shown in (a) with two holes (b), with three holes (c), with four holes (d), and with five holes (e), respectively.

$$V3 = V1 + 4. \tag{7}$$

*Proof.* Based on Theorem 7, Equation (5), and the concept of the complement of a shape, Corollary 12 is proved.

Klette and Rosenfeld [21] described the “proper hole” which corresponds to a hole with four-connectivity. On the other hand, they also defined the “improper hole” with eight-connectivity. Thus, we recommend that all the shapes and their inner complements (holes) should consider side-connected (“proper holes”). Figure 5 shows an invalid case of a hole. Figure 5(a) illustrates an example of a shape with an “improper hole.” Figure 5(b) shows the “improper hole,” this is due to the fact that the “improper hole” has no four-connectivity.

### 5. How to Compute the Euler Number via the VCC

Based on the concepts of the number of the outer vertices  $V1$  and  $V3$ , Equation (5), and the approach proposed in [15], we have the following theorem.

**Theorem 13.** *The number of holes  $H$  of any shape composed of side-connected pixels is defined as follows:*

$$H = \frac{V3 - V1}{4} + 1. \tag{8}$$

*Proof.* For the base case, when the shape has only one hole, we have  $V1 = 4$  and  $V3 = 4$  (see Figure 6). Therefore, using Equation (8),  $H = 1$ . For the next holes in the same binary

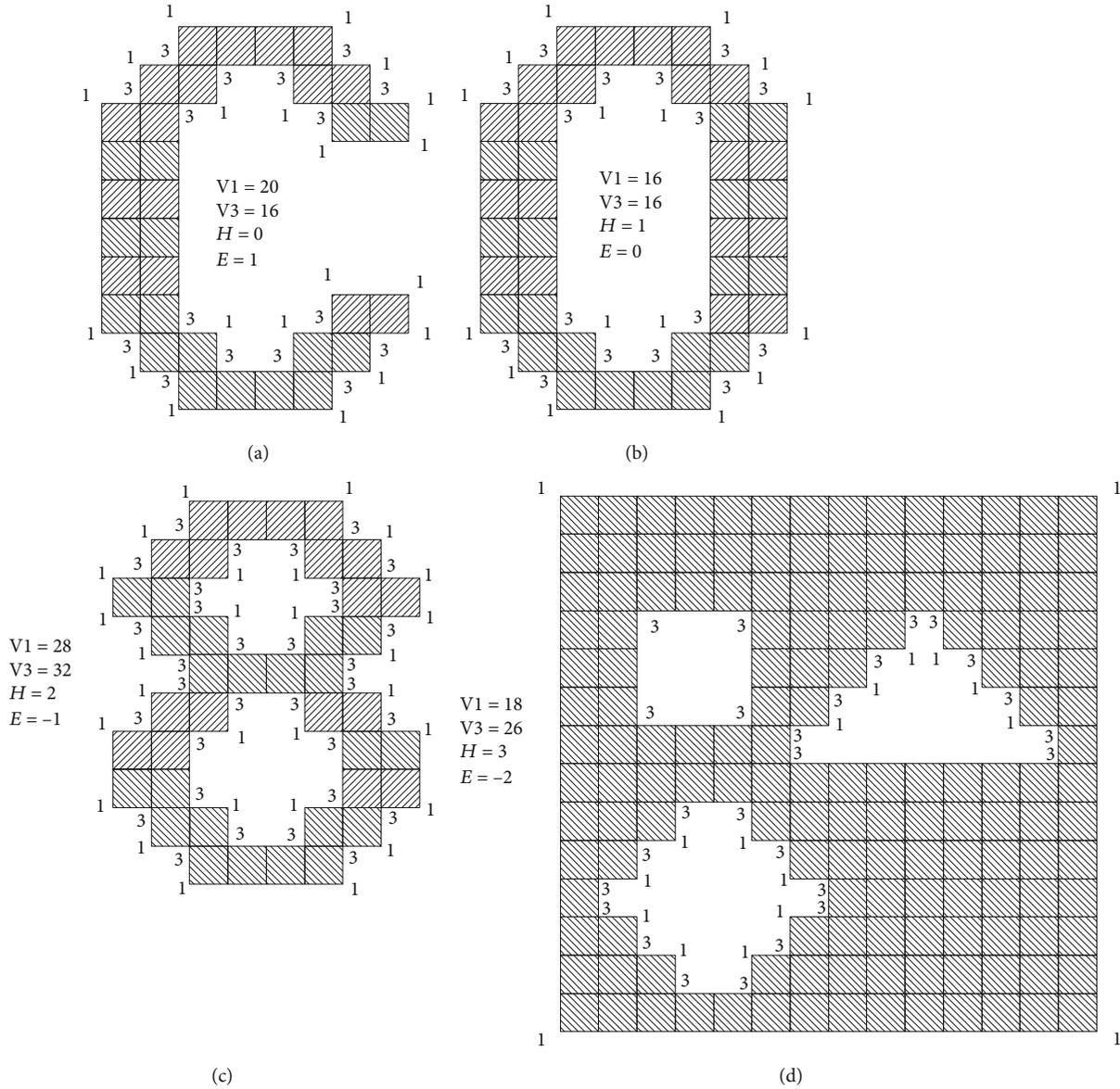


FIGURE 7: Examples of the computation of the number of holes and the Euler number: (a) an example of a 2D shape, where  $H = 0$  and  $E = 1$ ; (b) an example of a binary shape with one hole, where  $H = 1$  and  $E = 0$ ; (c) an example of a shape, where  $H = 2$  and  $E = -1$ ; (d) an example of a binary shape with three holes, i.e.,  $H = 3$  and  $E = -2$ .

shape, according to Equation (7),  $V3 - V1$  will be incremented by 4 for each new hole in the shape. Figure 6(b) presents the same shape shown in (a) now with two holes, (c) with three holes, (d) with four holes, and (f) with five holes, respectively. In each new step,  $V3 - V1$  is incremented by 4, which we know is true.

**Theorem 14.** *The number of holes  $H$  of any binary image composed of side-connected pixels is defined as follows:*

$$H = \frac{V3 - V1}{4} + C, \tag{9}$$

where  $C$  corresponds to the number of connected regions.

*Proof.* The proof of this theorem is similar like the proof of Theorem 13, just considering the variable  $C$ .

Substituting  $H$  from Equation (9) in the Euler-number equation  $E = C - H$ , we have.

$$E = \frac{V3 - V1}{4}. \tag{10}$$

For example, for the shape shown in Figure 1(b),  $V1 = 11$  and  $V3 = 7$ , in this case,  $C = 1$  because we only have a connected region. Substituting these numerical values in Equation (9),  $H = 0$  which is true; this shape has no hole. Therefore, the Euler number of the abovementioned shape is equal to one, i.e.,  $E = 1$ . Figure 7 illustrates four examples of the computation of the Euler number. Figure 7(a) presents an example of a binary shape without holes, where  $V1 = 20$ ,

$V_3 = 16$ , and  $C = 1$ ; therefore,  $H = 0$  and  $E = 1$ . Figure 7(b) shows an example where  $V_1 = 16$ ,  $V_3 = 16$ , and  $C = 1$ ; therefore,  $H = 1$  and  $E = 0$ . Figure 7(c) presents another example, where  $V_1 = 28$ ,  $V_3 = 32$ ,  $C = 1$ ,  $H = 2$ , and  $E = -1$ . Finally, for the shape shown in Figure 7(d),  $V_1 = 18$ ,  $V_3 = 26$ ,  $C = 1$ ,  $H = 3$ , and  $E = -2$ , respectively. Note that the computation of the Euler number of the abovementioned examples was computed in an easy way by means of the number of vertices  $V_1$  and  $V_3$ .

Now, if we consider the four binary shapes of the abovementioned examples shown in Figure 7 as a unique binary image composed of four connected regions, we have  $C = 4$ ,  $V_1 = 82$ , and  $V_3 = 90$ . Therefore,  $H = 6$  and  $E = -2$ .

**Theorem 15.** For any binary shape, if the number of vertices  $V_1$  is equal to the number of vertices  $V_3$ , i.e.,  $V_1 = V_3$ , then this binary shape only has one hole, i.e.,  $H = 1$ .

*Proof.* This is proved by direct calculation using Equation (8).

**Corollary 16.** For any binary image, if the number of vertices  $V_1$  is equal to the number of vertices  $V_3$ , i.e.,  $V_1 = V_3$ , then the Euler number is equal to zero, i.e.,  $E = 0$

*Proof.* This is proved by direct calculation using Equation (10).

### 5.1. How to Compute the Number of Connected Regions without Holes in a Hole

**Theorem 17.** The number of connected regions without holes  $C$  in any "proper hole" is defined as follows:

$$C = \frac{V_1 - V_3}{4} + 1. \quad (11)$$

*Proof.* Refer to proof of Theorem 14. Figure 8 illustrates an example of connected regions in a hole. The hole shown in Figure 8 has three connected regions, where  $V_1 = 31$  and  $V_3 = 23$ . Using Equation (11), we obtain that the number of connected regions (without holes) is equal to three, i.e.,  $C = 3$ .

## 6. Results

In order to illustrate the capabilities of the proposed method, we present two different applications: the first one, the computation of the number of holes of a slide of a piece of polyurethane; and the second one, two examples which show the Euler number computation applied to two images of single segments from tumorous glands of cribriform cases of prostate cancer (PCa).

**6.1. Computation of the Number of Holes in Polyurethanes.** Polyurethanes are polymers composed of units of organic chains joined by carbamate links. Flexible polyurethane foams are used as cushioning for a great variety of commercial products, such as bedding, furniture, packaging, carpet underlay, and in the automotive industry. In this kind of materials, it is important to know the number of

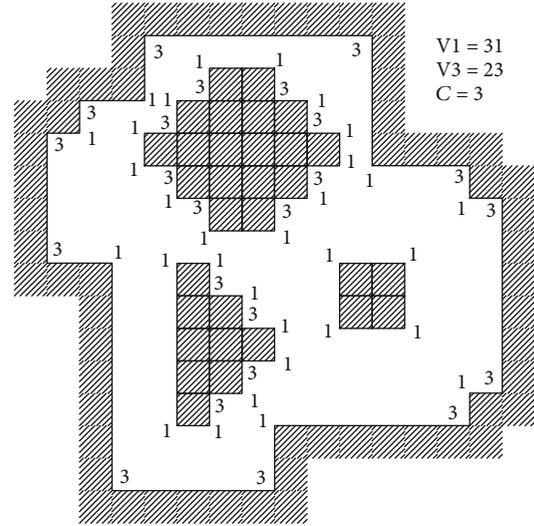


FIGURE 8: Examples of connected regions in a hole.

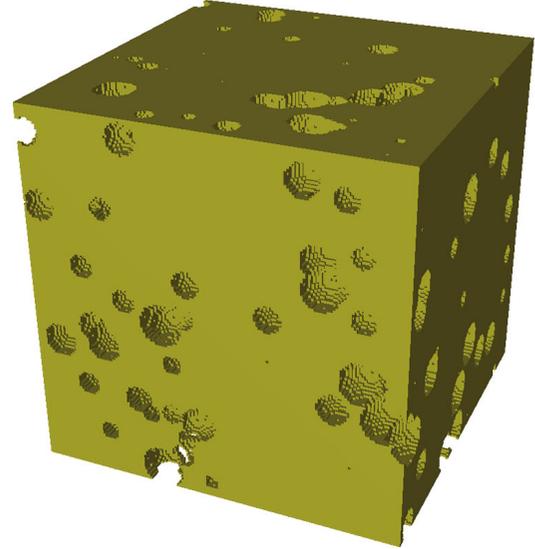


FIGURE 9: An example of a piece of polyurethane.

holes and the hole sizes. Thus, using the proposed method, we will compute the numbers of holes of a slide of a piece of polyurethane. This is due to the fact that the proposed method only works in the 2D domain. In order to have a digital representation of this kind of objects, a previous process was developed. Thus, Figure 9 shows an example of a piece of polyurethane. Figure 10 illustrates the voxels of the piece of polyurethane shown in Figure 9; the 3D array of voxels is equal to  $150 \times 150 \times 150$ . Figure 11 shows the bubbles in the polyurethane presented in Figure 10; note that these bubbles have different sizes. Finally, Figure 12 shows a slide of the object illustrated in Figure 10; notice that this slide corresponds to the first slide of the front of the object. Thus, the slide shown in Figure 12 is considered a binary image. Using the proposed method, the number of holes was computed and is equal to

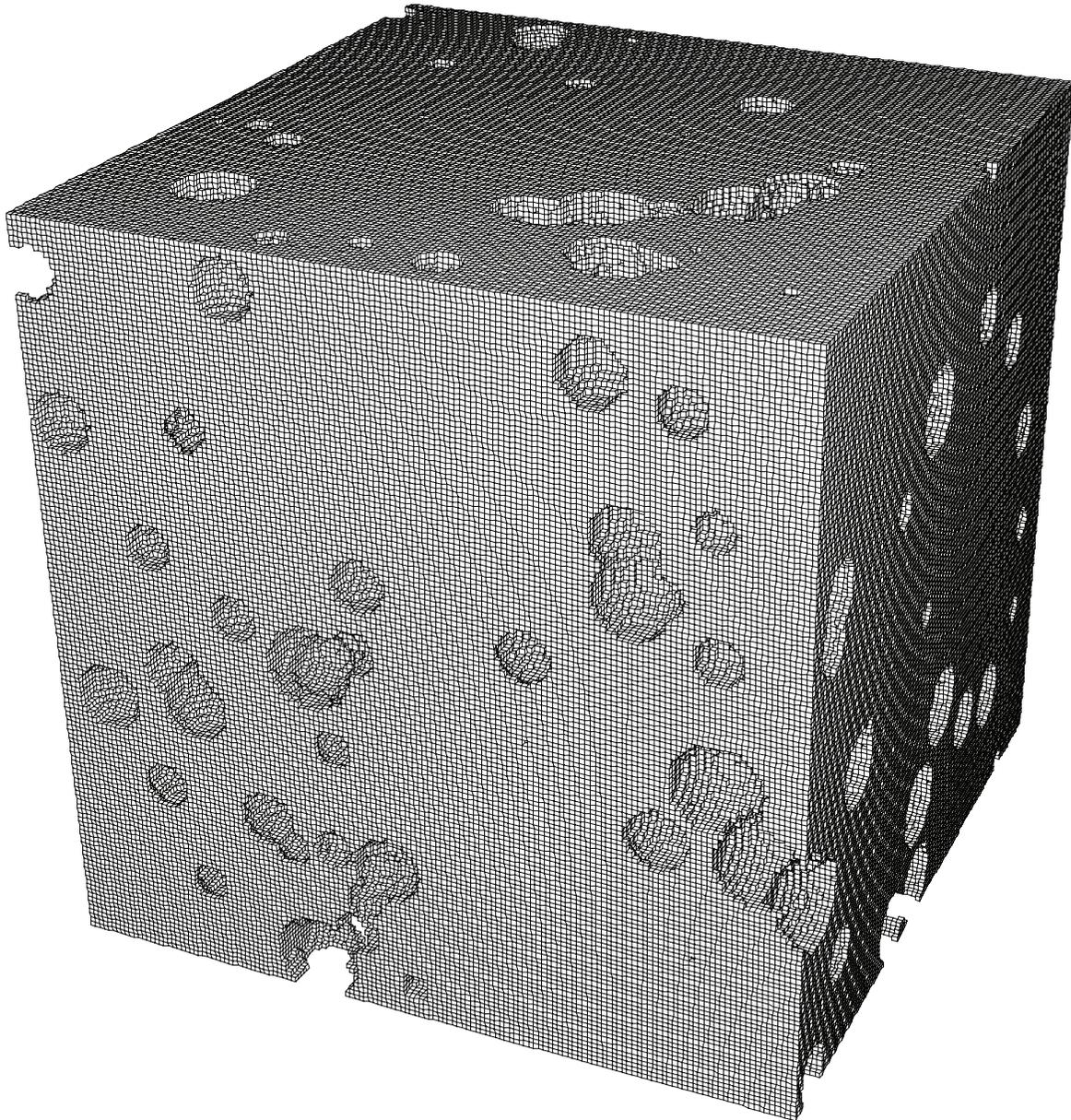


FIGURE 10: The voxels of the piece of polyurethane shown in Figure 9.

22, i.e.,  $H = 22$ . Therefore, the Euler number is equal to  $-21$ , i.e.,  $E = -21$ .

*6.2. Computation of the Euler Number of Images of Single Segments from Tumorous Glands.* The examples in Figures 13–16 illustrate the Euler number computation applied to two images of single segments from tumorous glands of cribriform (sieve-like) cases of prostate cancer (PCa). A reference pathologist has assessed the underlying tumor with Gleason grade 4 (and has explicitly noted such cribriform morphology). In order to assess a degree of this sieve-like form by a computerized algorithm, it will be useful to determine the number of holes as a measure of “cribriformity.” The figures illustrate how this is being done in practice.

*6.3. Comparison of the Proposed Method with Others.* Many authors have proposed different methods and algorithms to compute the Euler number; different techniques and representations have been used. For example, using bintrees, Bieri [7] defines the computation of the Euler characteristic of digital objects from their bintree representation. Based on quadtrees, Dyer [6] presents the computation of the Euler number of an image from its quadtree. Other methods have been proposed by means of graphs, such as Chiavetta and Di Gesù [8] who presented a method of parallel computation of the Euler number via connectivity graph. Bishnu et al. [10] define a pipeline architecture for computing the Euler number of a binary image. Other methods are based on the skeleton representation; for example, Díaz-De-León and Sossa-Azuela [12] present a

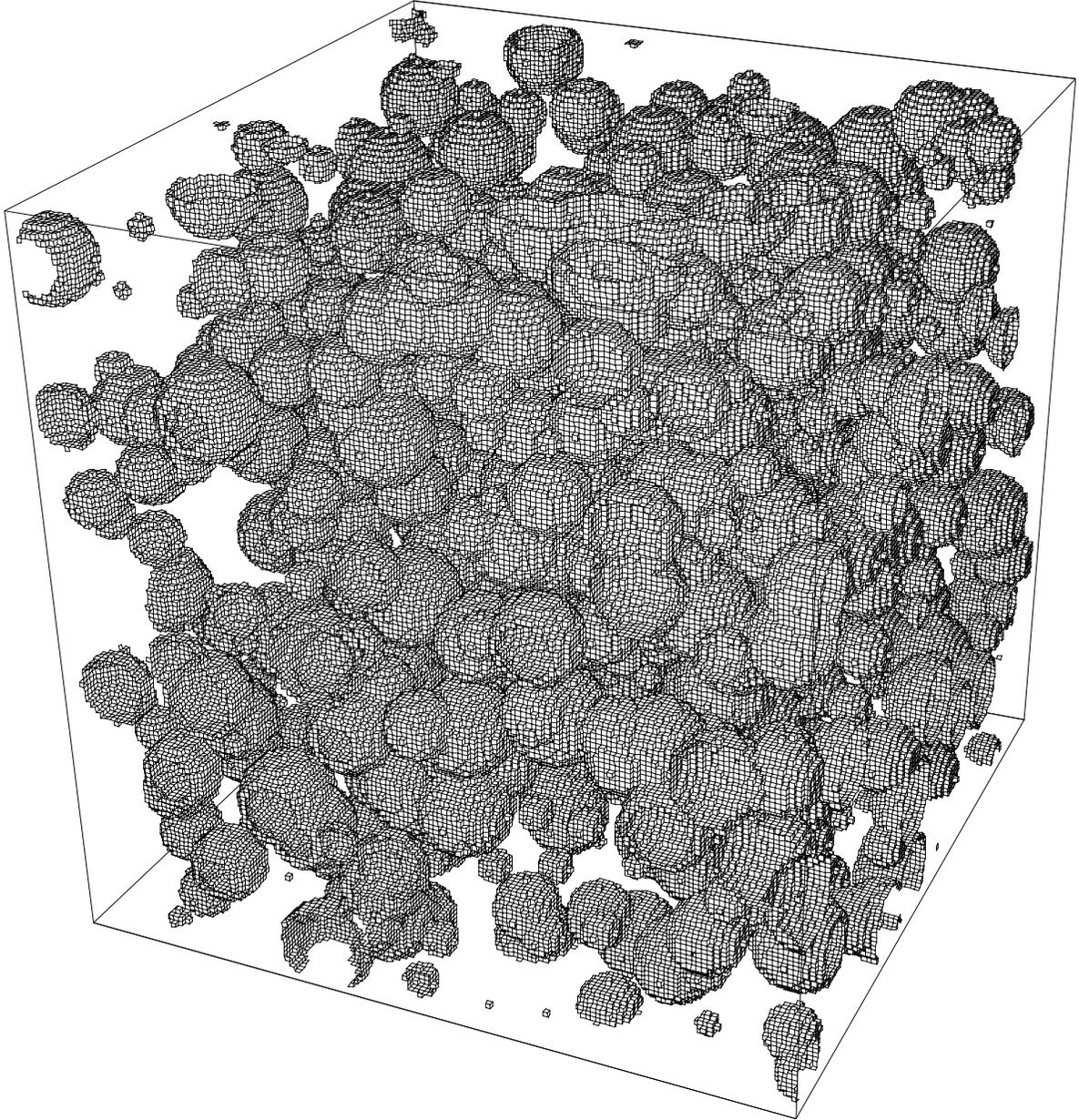


FIGURE 11: The bubbles in the polyurethane shown in Figure 10.

method to obtain the Euler number of a binary object via its skeleton; the number of terminal points and the number of three-edge points in the graph are used to compute this invariant. In [13], the concept of the contact perimeter [22] was used to compute the Euler number. Thus, using the contact perimeter  $P_c$ , the number of holes  $H$  for any unit-width shape was defined by the following equation:

$$H = \frac{2P_c - P}{4} + 1, \quad (12)$$

where  $P$  is the perimeter of the shape. Note the similarity with Equations (8) and (12).

Some authors have used the VCC to compute the Euler number. The aforementioned method presented in [14] is a fast algorithm to compute the Euler number. However, this method only considers shapes with one hole. Sossa-Azuola et al. [15] count the number of vertices “1” and “3” of the VCC and compute the number of holes by a simple formula. We based, in part, our approach on this technique.

## 7. Computational Comparative Analyses

In this section, we compare our proposed method with two of the methods reported in the literature. In particular, we compare our proposal against the method used by the commercial tool of MATLAB, which employs the

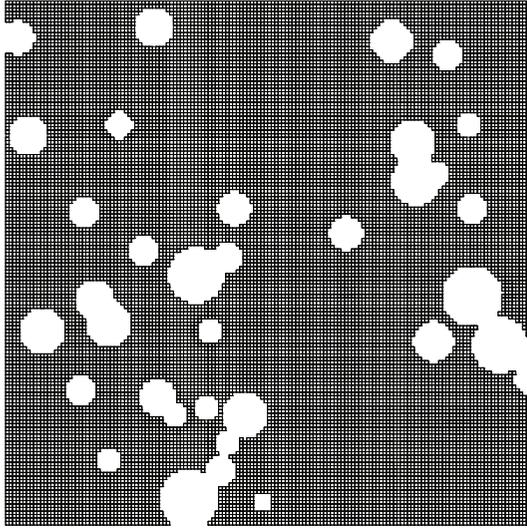


FIGURE 12: The front slide of the example of polyurethane shown in Figure 10.

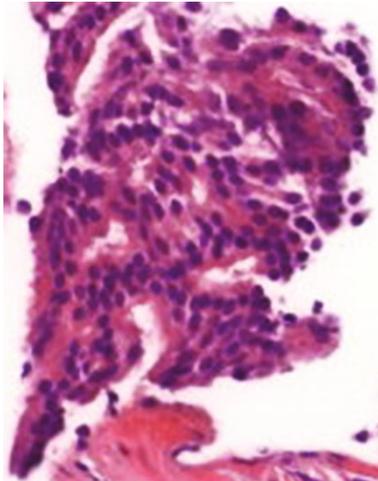


FIGURE 13: Example 1:  $195 \mu\text{m} \times 249 \mu\text{m}$ .

original algorithm introduced by Gray in [23], and the method described in [16].

The experiments were carried out on a Acer laptop with Intel, Core i7, 7th Generation, i7-7700HQ, 2.80 GHz, Quad-Core, and RAM 16GB; all the processes were computed on CPU, running under Windows 10 operating system; Figure 16 was used as the comparison example. Table 1 presents the complexity and execution time of the abovementioned methods. Note that the shortest execution time corresponds to the method proposed here. Furthermore, using the VCC notation, various shape features may be computed directly from this representation in an easy way.

### 8. Conclusions

We have found out more properties of the VCC. An important property of the VCC is that its chain elements represent real values not symbols such as other chain codes; they are

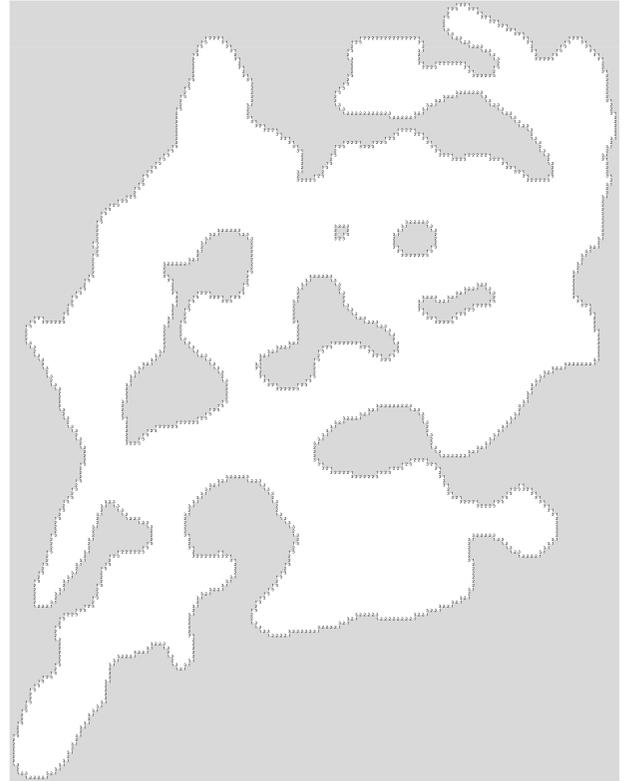


FIGURE 14: Example 1:  $195 \mu\text{m} \times 249 \mu\text{m}$ . From all VCCs, we here obtain  $V3 = 554$  and  $V1 = 538$ ; i.e., the number of connected regions is  $C = 1$ , the number of holes is being computed as  $H = (V3 - V1)/4 + C = (554 - 538)/4 + 1 = 5$ , and likewise the Euler number is  $E = (V1 - V3)/4 = (538 - 554)/4 = -4$ .

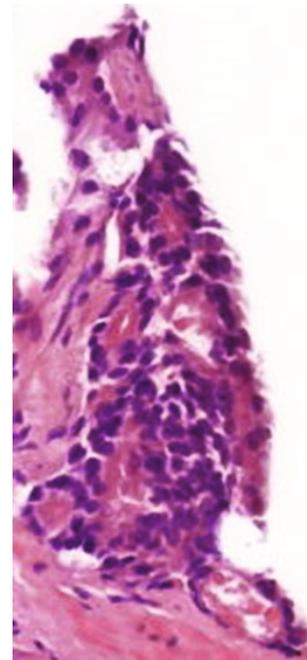


FIGURE 15: Example 2:  $173 \mu\text{m} \times 378 \mu\text{m}$ .

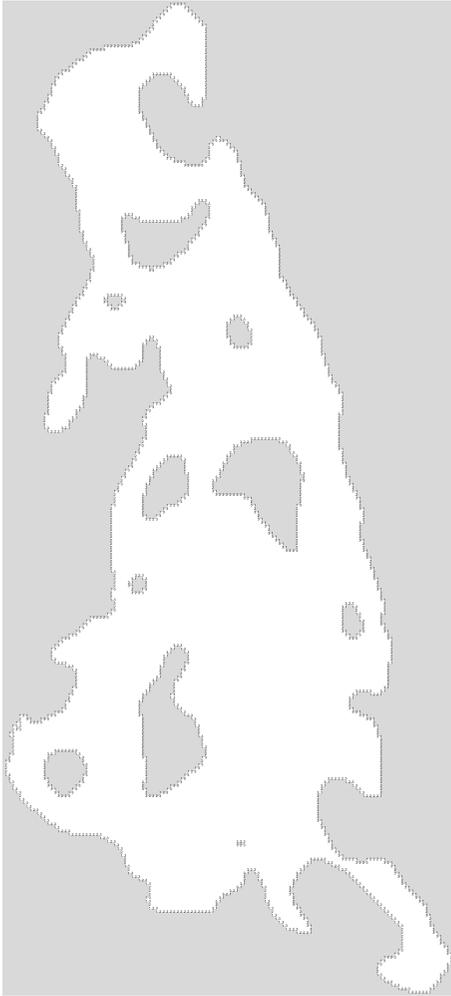


FIGURE 16: Example 2:  $173 \mu\text{m} \times 378 \mu\text{m}$ . From all VCCs, we here obtain  $V3 = 565$  and  $V1 = 529$ ; i.e., the number of connected regions is  $C = 1$ , the number of holes is being computed as  $H = (V3 - V1)/4 + C = (565 - 529)/4 + 1 = 10$ , and likewise the Euler number is  $E = (V1 - V3)/4 = (529 - 565)/4 = -9$ .

TABLE 1: Comparison of abovementioned methods.

Solution time (seconds)			
Method	Equation (8)	[23]	[16]
Full process	0.3359	0.0348	0.1035
Complexity	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Execution time	$1.5790e-4$	$1.6310e-4$	$1.7223e-4$

part of the shape and indicate the number of cell vertices of the shape nodes. Thus, the VCC descriptor of any shape has all its geometrical information. We have concluded that it is possible to obtain a part of topological information of a shape by means of its VCC description.

A number of definitions, corollaries, and theorems were presented, which allowed us to find out interesting properties, such as the computation of the connected regions in a hole, the analysis of complementary chains, the computation of the number of holes in a binary image or shape, the com-

putation of the Euler number, and the detection of convex and concave shapes. Finally, in order to prove the abovementioned concepts, we presented some examples of images of the real world.

Suggestion for further work would be to extend the VCC and the abovementioned concepts from 2D to 3D domain.

## Data Availability

Data used in preparation of this paper correspond to the images of single segments from tumorous glands of cribriform (sieve-like) cases of prostate cancer (PCa). A reference pathologist has assessed the underlying tumor with Gleason grade 4 (and has explicitly noted such cribriform morphology). A measure of “cribriformity” may be obtained by means of the determination of the number of holes using the proposed method. The data availability of the abovementioned images were supplied by Dr.-Ing. Ulf-Dietrich Braumann from University of Leipzig, braumann@uni-leipzig.de.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research work was, in part, supported by Institute of Research in Applied Mathematics and Systems from National Autonomous University of México and National Research System from National Council for Science and Technology.

## References

- [1] E. Bribiesca, “A new chain code,” *Pattern Recognition*, vol. 32, no. 2, pp. 235–251, 1999.
- [2] H. S. Yang and S. Sengupta, “Intelligent shape recognition for complex industrial tasks,” *IEEE Control Systems Magazine*, vol. 8, no. 3, pp. 23–30, 1988.
- [3] J. Bogaert, P. Van Hecke, and R. Ceulemans, “The Euler number as an index of spatial integrity of landscapes: evaluation and proposed improvement,” *Environmental Management*, Springer, New York, 2002.
- [4] M. N. DeMers, *Fundamentals of Geographic Information Systems*, John Wiley & Sons, New York, 1997.
- [5] I. Pitas, *Digital Image Processing Algorithms*, Prentice Hall, New York, 1993.
- [6] C. R. Dyer, “Computing the Euler number of an image from its quadtree,” *Computer Graphics and Image Processing*, vol. 13, no. 3, pp. 270–276, 1980.
- [7] H. Bieri, “Computing the Euler characteristic and related additive functionals of digital objects from their bintree representation,” *Computer Vision, Graphics, and Image Processing*, vol. 40, no. 1, pp. 115–126, 1987.
- [8] F. Chiavetta and V. Di Gesù, “Parallel computation of the Euler number via connectivity graph,” *Pattern Recognition Letters*, vol. 14, no. 11, pp. 849–859, 1993.
- [9] M. H. Chen and P. F. Yan, “A fast algorithm to calculate the Euler number for binary images,” *Pattern Recognition Letters*, vol. 8, no. 5, pp. 295–297, 1988.

- [10] A. Bishnu, B. B. Bhattacharya, M. K. Kundu, C. A. Murthy, and T. Acharya, "A pipeline architecture for computing the Euler number of a binary image," *Journal of Systems Architecture*, vol. 51, no. 8, pp. 470–487, 2005.
- [11] D. Ziou and M. Allili, "Generating cubical complexes from image data and computation of the Euler number," *Pattern Recognition*, vol. 35, no. 12, pp. 2833–2839, 2002.
- [12] J. L. Díaz-De-León and J. H. Sossa-Azuela, "On the computation of the Euler number of a binary object," *Pattern Recognition*, vol. 29, no. 3, pp. 471–476, 1996.
- [13] E. Bribiesca, "Computation of the Euler number using the contact perimeter," *Computers and Mathematics with Applications*, vol. 60, no. 5, pp. 1364–1373, 2010.
- [14] L. A. Wulandhari and H. Haron, "Characteristic of rectangular Vertex Chain Code for shapes with hole," in *2009 International Conference on Information Management and Engineering*, pp. 648–650, Kuala Lumpur, Malaysia, 2009.
- [15] J. H. Sossa-Azuela, R. Santiago-Montero, M. Pérez-Cisneros, and E. Rubio-Espino, "Computing the Euler number of a binary image based on a vertex codification," *Journal of Applied Research and Technology*, vol. 11, no. 3, pp. 360–370, 2013.
- [16] J. H. S. Azuela, R. Santiago, A. López, A. P. Ayala, E. V. Cuevas Jimenez, and E. R. Espino, "Alternative formulations to compute the binary shape Euler number," *IET Computer Vision*, vol. 8, no. 3, pp. 171–181, 2014.
- [17] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Pearson, 330 Hudson Street, New York, NY 10013, 4th edition, 2017.
- [18] R. M. Haralick and L. G. Shapiro, "Glossary of computer vision terms," *Pattern Recognition*, vol. 24, no. 1, pp. 69–93, 1991.
- [19] W. Karush, *Webster's New World Dictionary of Mathematics*, Webster's New World, Simon & Schuster, Inc., New York, 1989.
- [20] B. Yip and R. Klette, "Angle counts for isothetic polygons and polyhedra," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1275–1278, 2003.
- [21] R. Klette and A. Rosenfeld, *Digital Geometry*, Morgan Kaufmann Publishers, Elsevier, San Francisco CA 94111, U.S.A., 2004.
- [22] E. Bribiesca, "Measuring 2D shape compactness using the contact perimeter," *Computers & Mathematics with Applications*, vol. 33, no. 11, pp. 1–9, 1997.
- [23] S. B. Gray, "Local properties of binary images in two dimensions," *IEEE Transactions on Computers*, vol. C-20, no. 5, pp. 551–561, 1971.