

Alternative Way to Compute the Euler Number of a Binary Image

J. H. Sossa-Azuela^{*1}, E. B. Cuevas-Jiménez², D. Zaldivar-Navarro³

¹ Centro de Investigación en Computación-IPN
Av. Juan de Dios Bátiz, esquina con Miguel Othón de Mendizábal
Mexico City, C. P. 07738. MEXICO
*hsossa@cic.ipn.mx

^{2,3} Centro Universitario de Ciencias Exactas e Ingenierías (CUCEI)
Universidad de Guadalajara
Av. Revolución 1500
Col. Olímpica C.P. 44430
Guadalajara, Jalisco, MEXICO

ABSTRACT

In this paper an alternative way to compute the (E) Euler number of a binary image via information about its pixels is presented. The P perimeter of the objects in the image, their P_c contact perimeter and the T -type pixel are used to obtain this important invariant. This is the second time the Euler number is described in terms of the contact perimeter of the objects in an image but with new results. The first paper that reports computing the Euler number of a binary shape in terms of the P and P_c is in [E. Bribiesca, Computation of the Euler number using the contact perimeter, Computers and Mathematics with Applications 60:1364-137 (2010)]. Bribiesca's proposal is useful only for unit-width shapes. In this paper, we extend Bribiesca's method for non-unit-width shapes.

Keywords: Binary image characterization, perimeter, contact perimeter, Euler number, topological descriptor, topological invariant.

RESUMEN

En este trabajo se presenta un método alternativo para el cálculo del número de Euler (E) de una imagen binaria mediante información de sus píxeles. El perímetro P de los objetos en la imagen, sus perímetros de contacto P_c y el tipo T de la celda son utilizados para obtener este importante invariante. Esta es la segunda vez que el número de Euler es descrito en términos del perímetro de contacto de los objetos en una imagen. El primer trabajo que reporta el calcular el número de Euler de una forma binaria en términos de P y P_c es en [E. Bribiesca, Computation of the Euler number using the contact perimeter, Computers and Mathematics with Applications 60:1364-137 (2010)]. La propuesta de Bribiesca es útil sólo para formas de grosor unitario. En este trabajo extendemos la propuesta de Bribiesca para el caso de formas de cualquier grosor.

1. Introduction

A digital binary image is a two-dimensional array that has two gray levels in $\{0,1\}$. An image is composed of all the flat connected regions representing the projections of the objects onto the discrete plane. From the pixels of each of these objects, several geometrical and topological features can be obtained to characterize these objects.

Among the topological features, the Euler number or Euler characteristic of an image has proven to be an important feature in many image analyses and visual inspection applications. The Euler number has been used, for example, in the

recognition of industrial parts [1]. Mathematically, the Euler number of a binary image is given as

$$E = N - H \quad (1)$$

Several methods have been proposed to obtain the Euler number of a binary image. Refer for example to [2], [3], [4], [5], [6], [7], [8], [9] and [14]. In particular in [3], the Euler of a binary image is computed from its quadtree, while in [7], this feature is computed from the connectivity graph of the image. In [10], [11] and [12], hardware implementations to compute the Euler number of an image are proposed. In [10], for example, a

VLSI implementation is given, while in [11] and [12], a pipeline architecture and a co-processor are used, respectively.



Figure 1. Euler numbers of two simple shapes.

In [8], the Euler number of a binary image is computed in terms of the number of terminal points (points with just one neighbor) and the number of three-edge points (points with only three neighbors) of the skeletons of the regions inside the image. More in detail, if Tps is the number of terminal points of the n skeletons from n objects in the image, and $TEps$ is the number of three-edge points of these n skeletons, then

Several methods have been proposed to obtain the Euler number of a binary image. Refer for example to [2], [3], [4], [5], [6], [7], [8], [9] and [14]. In particular in [3], the Euler of a binary image is computed from its quadtree, while in [7], this feature is computed from the connectivity graph of the image. In [10], [11] and [12], hardware implementations to compute the Euler number of an image are proposed. In [10], for example, a VLSI implementation is given, while in [11] and [12], a pipeline architecture and a co-processor are used, respectively.

In [8], the Euler number of a binary image is computed in terms of the number of terminal points (points with just one neighbor) and the number of three-edge points (points with only three neighbors) of the skeletons of the regions inside the image. More in detail, if Tps is the number of terminal points of the n skeletons from n objects in the image, and $TEps$ is the number of three-edge points of these n skeletons, then

$$E = \frac{Tps - TEps}{2} \quad (2)$$

For the case of the objects shown in Figure 1: $Tps(\tilde{N}) = 4$ and $TEps(\tilde{N}) = 0$, thus $E(\tilde{N}) = 2$. Similarly: $Tps(B) = 0$ and $TEps(B) = 2$, thus $E(B) = -1$.

In short, in [14], the author reports a method to compute the Euler number of 2D and 3D unit-width shapes by means of the so-called contact perimeter. This idea of computing the Euler number of shapes from the skeleton of the shape was first used in [8].

In this paper we use also the shape contact perimeter to compute its Euler number. Instead of using the skeleton of the shape to compute the Euler number of a shape, we use all the pixels of the region of the shape. As a result, we derive a different formulation which provides the same exact value for the Euler number of a shape.

The rest of this work is organized as follows. In section 2, the proposal is fully described. In section 3, several examples to illustrate the functioning and efficiency of the method are given. Finally, Section 4 is dedicated to the conclusions of this investigation and directions for further research.

2. Perimeters

In this section the concepts of perimeter: P , and contact perimeter: P_c are given. These definitions are valid for triangular, rectangular and hexagonal cells. They will be necessary to derive in the following sections the expressions to compute the Euler feature of a binary shape.

In this section the concepts of perimeter: P , and contact perimeter: P_c are given. These definitions are valid for triangular, rectangular and hexagonal cells. They will be necessary to derive in the following sections the expressions to compute the Euler feature of a binary shape.

According to [13], the P perimeter of the shapes inside an image composed of pixels corresponds to the sum of lengths of the sides of each shape. For example, the perimeters of the shapes shown in Figure 2 (a), (c) and (e) are 14, 9, and 20, respectively. From the same reference, the P_c contact perimeter of a shape in an image composed of pixels corresponds to the sum of the

lengths of segments that are common to two pixels. For example, the contact perimeter of the shapes shown in Figure 2 (b), (d) and (f) are 11, 9 and 5, respectively.

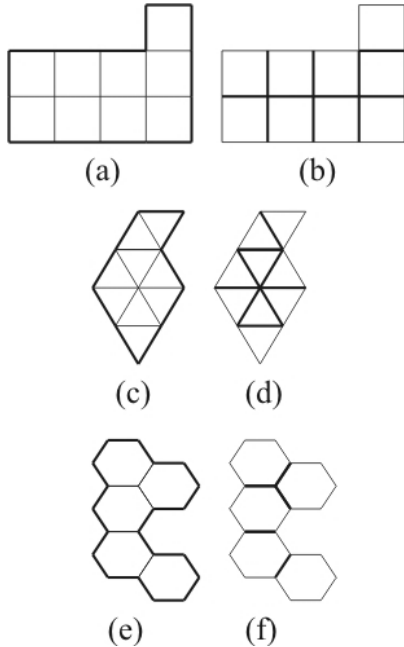


Figure 2. Perimeter and contact perimeter of three simple binary shapes.

Based on these two parameters, in [13], the authors show that the following relationship is held: For any S_n shape, composed of n pixels, it is held that

$$2P_c + P = nT \tag{3}$$

with P_c being again the contact perimeter, P being the perimeter of the shape and T being the number of sides of the pixel. For a triangular pixel, $T = 3$, for a square pixel, $T = 4$ and for a hexagonal pixel, $T = 6$.

Remark 1. Although the theory that we are going to present in the following paragraphs is valid for regions composed of square, triangular and hexagonal pixels, in this paper we are going to consider only square pixels (pixels for which $T = 4$).

Assumption 1. In what follows, as in [13], and in order to avoid the so-called structural problem of pixels (connectivity paradox), we suppose that the pixels of a S_n shape, composed of n pixels, are four-connected: pixels are only allowed to connect to other pixels only by their sides and not by their corners.

3. Bribiescas' proposal to compute the Euler number of a binary shape

In [14], E. Bribiesca presents a general equation to compute the number of H holes of unit-width shapes. He does not directly provide an equation to compute the Euler number of the unit-width shape. He first computes the number of H holes. He then applies Equation (1) to obtain the Euler number for the given shape.

The equation to compute the number of holes of a unit-width shape proposed in [14] is the following one:

$$H = \frac{(T-2)P_c - P}{T} + 1 \tag{4}$$

As mentioned in [14], this equation is only valid for unit-width shapes. In fact, if we apply Equation (4) to the three shapes of Figure 3, we get the following results:

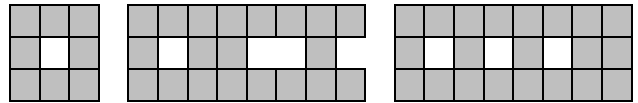


Figure 3. Three shapes, with one, two and three holes, respectively.

Shape 1: $P = 16, P_c = 8, NT = 0$, and $H = \frac{2 \cdot 8 - 16}{4} + 1 = 1$. Correct.

Shape 2: $P = 34, P_c = 23, NT = 2$, and $H = \frac{2 \cdot 23 - 34}{4} + 1 = 4$. Non correct.

Shape 3: $P = 34, P_c = 25, NT = 2$, and $H = \frac{2 \cdot 25 - 34}{4} + 1 = 5$. Non correct.

As we can see, only in the first case, Equation (4) provided the correct result. This is because only the first shape is unit-width shape. The other two are not.

In Figure 4, the parts where the two shapes are not unit-width have been marked dark grey.

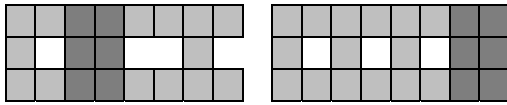


Figure 4. Three shapes, with one, two and three holes, respectively.

As can be seen, in both cases the shapes have a visible width of two pixels at these parts. To get the correct number of holes for these two shapes by means of equation (4), we would need first to thin them until getting two unit-width shapes. Suppose we do not want to do this; instead we would like to have an appropriate equation to compute the number of holes for any non-unit width shape. In the following sections we are going to derive an equation that allows computing the number of H holes of any non-unit width shape. Next, we are going to extend this equation to the case of shapes composed of several components. Finally, we are going to derive a general formula to compute the E Euler number of N binary shapes.

4. The concept of tetra pixel

As a first step to get an equation that allows computing the number of H holes of any non-unit-width shape, let us suppose that we take any unit-width shape and erode it with the four-structural element shown in Figure 5. The reference of the structural element is the one shown in gray. It is easy to show that when eroding any unit-width shape, all of its pixels will disappear at the end of the erosion process. No pixels will appear marked at the end of the erosion process.



Figure 5. Structural element.

Now let us take any non-unit-width shape and erode it with the same structural element. It is also easy to show that at the end of the erosion process at least one pixel of the shape will remain, at least by one pixel, due to the fact that the structural element fits.

From this, we can conclude that a shape is a unit-width shape if at the end of its erosion by a structural element, as the one shown in Figure 5, all of its pixels disappear. We can thus state the following definition:

Definition 1. Let S_n be a shape composed of n pixels. Let EE be the four-structural element shown in Figure 5. Let m be the number of pixels marked by eroding S_n with EE . Thus, if after eroding S_n with EE , $m = 0$, S_n is a unit-width shape; and if $m \neq 0$, S_n is a non-unit-width shape.

As an example, let us erode the three shapes of Figure 3. Figure 6 shows the erosion results. One can see that only the second and third shapes have not been eliminated. These remaining pixels will allow stating the following useful definition:



Figure 6. Only the second and third shapes have not been completely eroded.

Definition 2. Let S_n be a shape composed of n pixels. Let EE be the four-structural element shown in Figure 5. Let m be the number of pixels marked by the erosion process by eroding S_n with EE . Each of these m pixels is called a tetra pixel.

According to this definition, a S_n shape is a unit-width shape if it has no tetra pixels; on the contrary, if the shape has at least one tetra pixel, then it is a non-unit-width shape.

From now on, let NT be the number of tetra pixels of a S_n shape as the pixels marked by eroding with the EE of Figure 6.

5. Computing the number of holes and the Euler number of a binary image

From the material given in Sections 2, 3 and 4, It is possible to derive the set of relations that governs the connectivity of the pixels of the shapes of a binary image and that allows computing the number of H holes and the E Euler number of this image.

Let us begin with the equation to compute the number of H holes of a non-unit-width S_n shape. For this, we have the following result:

Proposition 1. The number of H holes of a binary S_n shape composed of n pixels is given as

$$H = \frac{(T-2)P_c - P}{T} + 1 - NT \tag{5}$$

Proof. We are going to proceed as in [14] by induction on the number of pixels of the S_n shape. For $n = 1$, S_1 consists of a single pixel. Thus, $H = 0$, $P_c = 0$, $P = T$, and $NT = 0$. We see that these values satisfy Equation (5).

As usual, when proceeding by induction, let us assume that Equation (5) is satisfied for S_n . Let H', P', P'_c, NT' be the number of the S_{n+1} shape after adding one pixel to S_n . Let ΔP_c be the increment to P_c after adding this pixel. Thus, $P'_c = P_c + \Delta P_c$, $P' = P + T - 2\Delta P_c$, and $NT' = NT$.

We have to show that (4) holds for S_{n+1} , this is:
 $H' = \frac{(T-2)P'_c - P'}{T} + 1 - NT'$

Let us proceed:

$$\begin{aligned} H' &= \frac{(T-2)(P_c + \Delta P_c) - (P + T - 2\Delta P_c)}{T} + 1 \\ &\quad - NT \\ &= \frac{(T-2)P_c - P}{T} + 1 \\ &\quad + \frac{(T-2)\Delta P_c - T + 2\Delta P_c}{T} - NT \\ &= H + (\Delta P_c - 1) - NT. \end{aligned}$$

Let us numerically verify the efficiency of Equation (5) by applying it to the three shapes of Figure 3.

For shape 1: $P = 16, P_c = 8, NT = 0$, and $H = \frac{2 \cdot 8 - 16}{4} + 1 - 0 = 1$.

For shape 2: $P = 34, P_c = 23, NT = 2$, and $H = \frac{2 \cdot 23 - 34}{4} + 1 - 2 = 2$.

For shape 3: $P = 34, P_c = 25, NT = 2$, and $H = \frac{2 \cdot 25 - 34}{4} + 1 - 2 = 3$.

In all three cases, as expected, the results are correct. Let us now extend Proposition 1 to the case of a shape or image composed of N components. For this we have the following result:

Proposition 2. The number of H holes of N binary shapes is given as

$$H = \frac{(T-2)P_c - P}{T} + N - NT \tag{6}$$

Proof. The proof can be done by induction again.

Finally, lets us derive the equation that allows directly computing the Euler number of any unit-width or non-unit-width or binary image composed of N components. For this we have the following result:

Proposition 3. The Euler number E of n binary images composed of squared pixels ($T = 4$), is always given as

$$E = \frac{P - (T-2)P_c}{T} + NT \tag{7}$$

Proof. From Equation (1) and Proposition 2:

$$\begin{aligned} E = n - H &= n - \left(\frac{(T-2)P_c - P}{T} - n + NT \right) \\ &= \frac{P - (T-2)P_c}{T} + NT \end{aligned}$$

By taking again each one of the three shapes of Figure 3, we have:

For shape 1: $P = 16, P_c = 8, NT = 0$, and $E = \frac{16 - 2 \cdot 8}{4} + 0 = 0$.

For shape 2: $P = 34, P_c = 23, NT = 2$, and $H = \frac{34 - 2 \cdot 23}{4} + 2 = -1$.

For shape 3: $P = 34, P_c = 25, NT = 2$, and $H = \frac{34 - 2 \cdot 25}{4} + 2 = -2$.

The reader can apply Equation (1) to the three shapes to verify that the results obtained by means of Equation (7) are correct.

6. Conclusions and present research

In this paper, an alternative way to compute the Euler number of a binary image composed of one or more possibly non-unit-width shapes is developed. It is based on information directly obtained from each pixel of each shape in the image: the perimeter, the contact perimeter and the number of the so-called tetra pixels. The proposal is useful for triangular, rectangular and hexagonal cells. A main feature of the proposal is its simplicity. Without loss of generality, only the case for squared pixels is presented. Due to the fact that each pixel can be processed independently, the proposal can be parallelizable.

Formulae for the case of triangular and hexagonal cells will be the subject of a future research. Another important case to consider in the future is the one related with 8-connected regions. Finally, another interesting extension of the proposal, of course, is for the case of three-dimensional cells (voxels). This will be also considered in a future investigation.

References

- [1] H. S. Yang and S. Sengupta, Intelligent shape recognition for complex industrial tasks, *IEEE Control Syst. Mag.* 23-29 (June 1988).
- [2] S. B. Gray, Local properties of binary images in two dimensions, *IEEE Trans. Comput.* 20(5), 551-561 (May 1971).
- [3] Ch. R. Dyer, Computing the Euler number of an image from its quadtree, *Comput. Vision, Graphics Image Process.* 13, 270-276, (1980).
- [4] H. Beri and W. Nef, Algorithms for the Euler characteristic and related additive functionals of digital objects, *Comput. Vision, Graphics Image Process.* 28, 166-175, (1984).
- [5] H. Beri, Computing the Euler characteristic and related additive functionals of digital objects from their beentree representation, *Comput. Vision, Graphics Image Process.* 40, 115-126 (1987).
- [6] M. H. Chen and P. F. Yan, A fast algorithm to calculate the Euler number for binary images, *Pattern Recognition Letters* 8(12), 295-297 (December 1988).
- [7] F. Chiavetta and V. Di Gesù, Parallel computation of the Euler number via connectivity graph, *Pattern Recognition Letters*, 14(11), 849-859 (November 1993).
- [8] J. L. Díaz de León S. and H. Sossa, On the computation of the Euler number of a binary object. *Pattern Recognition*, 29(3):471-476 (1996).
- [9] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, 1982.
- [10] S. Dey, B. B. Bhattacharya, M.K. Kundu, T. Acharya, A fast algorithm for computing the Euler number of an image and its VLSI implementation, in: *Proc. 13th International Conference on VLSI Design*, 330-335 (2000).
- [11] A. Bishnu, B. B. Bhattacharya, M. K. Kundu, C.A. Murthy, T. Acharya. A pipeline architecture for computing the Euler number of a binary image. *Journal of Systems Architecture: the EUROMICRO Journal.* 51(8):470-487 (2005).
- [12] S. Dey et al. "A co-processor for computing Euler number of a binary image using divide-and-conquer strategy", *Fundamenta Informaticae*, 76, pp.75-89 (2007).
- [13] E. Bribiesca, Measuring 2-D shape compactness using the contact perimeter, *Computers Math. Applic.* 33(11), 1-9 (1997).
- [14] E. Bribiesca, Computation of the Euler number using the contact perimeter, *Computers and Mathematics with Applications* 60:1364-137 (2010).

Acknowledgements

The authors thank SIP-IPN and CONACYT for the economical support under grants number 20111016 and 155014. We kindly thank the reviewers for their comments which contributed to the improvement of this paper.

Authors' Biographies**Humberto SOSSA-AZUELA**

He received the B.S. degree in electronics and communications engineering from Universidad de Guadalajara (University of Guadalajara), Mexico, in 1980, the M.Sc. degree in electrical engineering from CINVESTAV-IPN, Mexico in 1987, and the Ph.D. degree from Institut Nationale Polytechnique de Grenoble (National Polytechnic Institute of Grenoble), France, in 1992. From 1997 he has been with National Polytechnic Institute of Mexico, where he is currently a full-time professor in the Computing Research Center. From 1988, he is also a member of the Mexican National Research System (SNI). His research interests include image analysis, pattern recognition, neural networks applied to image retrieval, shape description and robot control.

**Erik CUEVAS-JÍMENEZ**

He received the B.S. degree with distinction in electronics and communications engineering from Universidad de Guadalajara, Mexico, in 1995, the M.Sc. degree in industrial electronics from ITESO, Mexico in 2000, and the Ph.D. degree from Freie Universität Berlin, Germany, in 2005. In 2001, he was awarded a scholarship from the German Service for Academic Interchange (DAAD) as full-time researcher. From 2006, he has been with Universidad de Guadalajara, where he is currently a full-time professor in the Department of Computer Science. From 2008, he has also been a member of the Mexican National Research System (SNI). His research interests include computer vision and artificial intelligence applications in medical image.

**Daniel ZALDIVAR-NAVARRO**

He received the B.S. degree with distinction in electronics and communications engineering from Universidad de Guadalajara, Mexico, in 1995, the M.Sc. degree in industrial electronics from ITESO, Mexico, in 2000, and the Ph.D. degree from Freie Universität Berlin, Germany, in 2005. In 2001, he was awarded a scholarship from the German Service for Academic Interchange (DAAD) as full-time researcher. Since 2006, he has been with Universidad de Guadalajara, where he is currently a professor in the Department of Computer Science. Since 2008, he has also been a member of the Mexican National Research System (SNI). His current research interests include biped robots design, humanoid walking control, and artificial vision.