A Recursive Approach to Identify the Objects in a 2D Image

Cezar Popescu

"Politehnica" University of Timişoara, Faculty of Automation and Computers,
Department of Automation and Applied Informatics.
Bv.V. Pârvan nr.2, Timişoara, Romania.
Phone: +40 0256 403246, e-mail: cpopescu@aut.utt.ro

Abstract: The goal of this paper is to present an original method, used to identify and number the objects in a 2D image. The objects are seen as sets of congruent pixels, separated from the background through a segmentation processing. The unique identification of each object is requested by the subsequent procedures of description and recognition.

Keywords: image processing and segmentation, object identification, description and recognition.

1 Segmentation, Identification, Description

The paper deals with objects that may be seen as bidimensional, or may be described accurately enough by a bidimensional view or projection. Second, we assume that the image has been preprocessed by some filtering method and that it has been segmented into distinct objects.

The approach on segmentation is based on finding discontinuities [5] [6]. On gray level images of some objects placed on a background we may assume that the histogram of the image is bimodal, i.e. the gray levels of the objects and of the background are centered on different mean values and have Gaussian distributions of the deviations, thus forming two "modes". A threshold segmentation is then the most common method and it is proved that an optimal segmentation may be obtained, in terms of minimizing the mean-square error [4]. Fig. 1 presents a gray level image of some objects on a background and the corresponding segmented image.

After the segmentation, the image becomes a conventional one, i.e. each pixel holds information not on its original color or gray level, but on its membership to the background or to the set of objects. At this moment, the objects are not yet uniquely identified.

The separation and identification of each object is a compulsory preprocessing before their description (in whatever terms) and recognition.

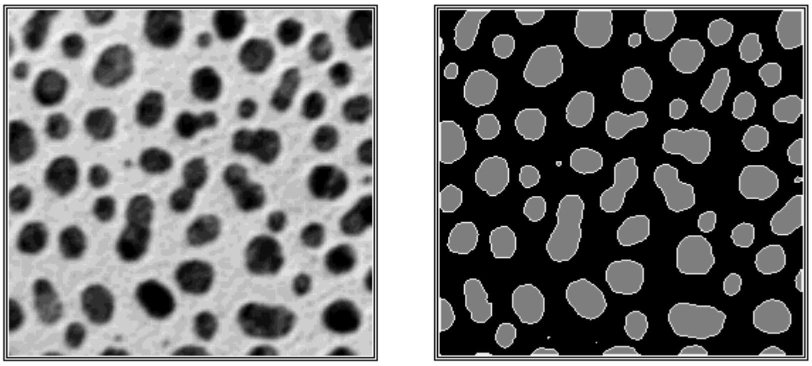

Figure 1

After the identification, objects may be descripted. This means that the information about the bidimensional shape of each object is somehow put in the form of a signature. This signature is, mainly, a set or an array of numbers which is computed so that it retains the important features of the shape but, at the same time, is short enough to be introduced in an automated classifier. This is the pourpose and meaning of the recognition step: to classify the signature of an object as belonging to either a nominated class of predefined objects, or to a default class of objects that are unrecognizable.

2 The Basis and Formulation of the Identification Problem

An object in a bidimensional image may be seen as a set, or an aggregate of connected pixels. Connectivity is defined in terms of neighbourhood in an orthogonal system of coordinates [2] [3]. Although other systems of coordinates were described and used, with some relative advantages, like the hexagonal shaped pixels [1], it is generally accepted that square shaped pixels are an optimal solution to represent an image. The neighbours of a pixel of coordinates (x,y) are defined as:

- 4-neighbours having the coordinates (x,y-1), (x-1,y), (x+1,y) and (x,y+1)

- d-neighbours (diagonal neighbours) having the coordinates (x-1,y-1), (x+1,y-1), (x-1,y+1) and (x+1,y+1)
- 8-neighbours, as the reunion of the two sets mentioned above

Fig. 2 presents a square area of nine pixel, consisting of the central one and its 8-neighbours.

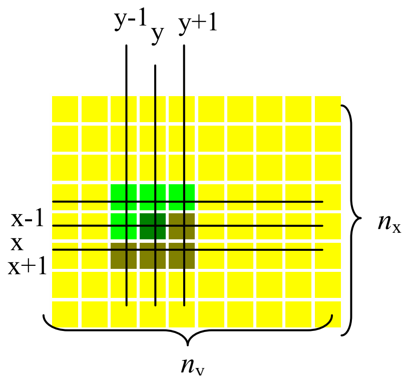

Figure 2

The coordinates of the current pixel of its already explored neighbour and of those not yet explored

As mentioned, after the segmentation phase, the objects are separated from the background, as a whole, but not one from each other. A straightforward method of individual identification and numbering is described below.

1. Declare an object counter and initialize it with zero.
2. Run through the image, line by line, left to right and top to bottom. When an object pixel is encountered, examine its 8-neighbours.

   2a. If none of them is numbered, then we have just found the first pixel of a new object. Increment the counter and mark the pixel with this value.

   2b. If some of the neighbours are already marked and numbered, then mark the current pixel with the same number.

Unfortunately, this simple algorithm is not complete. Fig. 3 shows two examples when a unique decision in case 2b cannot be made.

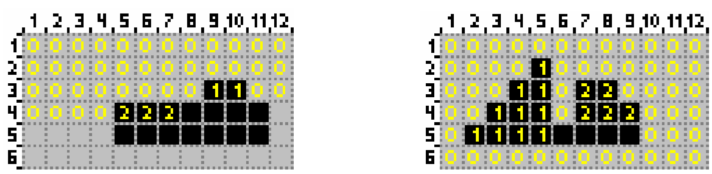

Figure 3

3 The Recursive Solution

The procedure starts from a source image, which is a conventional one, resulted after the process of segmentation, and stored with a byte per pixel. In this byte, one bit stores the information about the membership of the pixel to the set of objects, respectively to the background. Other bits may hold other information like, as an example, if the pixel is situated on the contour of the object.

In the destination image, which is also a conventional one, stored with two bytes per pixel, separate objects will be identified. Information about the object/background and contour membership still exists, but the objects will be separately identified and counted and an object index will be stored in each object pixel.

The method consists of the following steps:

1. An object counter is declared and initialized with zero. The current image coordinates are placed in the left upper corner.

2. The source image is browsed line by line, from left to right and from top to bottom until a pixel that belongs to an object is reached. Meanwhile, the pixels that belong to the background are marked with a zero in the destination image.

3. The object counter is incremented and the pixel in the current position is marked with this value in the destination image. A flag with the signification „end of object" is set to false. Then all the 8-neighbours of the current pixel are explored, looking for a pixel that, in the source image, is marked as belonging to the set of objects, but in the destination image is not yet marked. With the exception of the case when the object consist of a single pixel, such a neighbour will be found.

4. This neighbour is also marked, by calling a function that receives, as parameters, the position of the pixel and the current value of the object counter. Then, the function makes a new search through the 8-neighbours of this new pixel and it calls itself, recursively, from neighbour to neighbour, starting with the left 4-neighbour and continuing clockwise.

5. In the moment when the instantiation of the recursive function that was first called, at step 3, executes its return sequence, it means that the object has been browsed completely. Consequently the „end of object" flag is set to true and the sequential browsing of the image is continued from the point where it was interrupted at step 3. In conclusion, for each object, its first encountered pixel is marked in the process of sequential browsing, while all the others are marked recursively.

A special situation occurs at the margins of the image, where a complete 9 pixel neighbourhood cannot be defined. One solution would be to include code, both in the sequential procedure and in the recursive function, that tests this situation and treats these special cases accordingly. Although not difficult to implement, this solution will significantly increase the execution time.

Another solution is to create a border around the source image, enlarging it. All these pixels will be marked as background. The sequential browsing will be done only over the interior pixels. At the same time, the recursive procedure will stop naturally at these borders, since they belong to the background.

Conclusions

A recursive method for the identification and numbering in an image has been presented. The method is:

- consistent, in the sense it leaves no situation untreated.
- coherent, in the sense that it rezolves all possible conflicts.

The advantage of the method is that its consistency and coherency are easily proved, based on the recursivity itself. The method does not involve special treatment for any particular cases. It is, mathematically, elegant and simple from the point of view of the effort that is involved in conception and implementation as a software application.

The main disatvantage consists in a relatively long processor runtime, due to the multiple browsing and examination of the same pixel. Each time a recursive function call is implied, which leads to a specific call/return overhead. Although modern processors have RISC architectures and modern compilers have complex optimization schemes, these are efficient only for leaf functions, or for a certain call depth wich will not be sufficient for the recursive calls.

There is also, in theory, a possibility that, for very large images, the amount of RAM that was allocated for the stack may be outrun. This will not manifest as a fatal error since modern computers, like usual PCs have memory management systems and virtual memory is implemented. Still the time costs implied by the actual use of harddisk based virtual memory will be very high in this situation.

References

[1] Coster, M., Charmant, J-L. “Precis d’analyse d’images”, Edition CNRS, Paris, 1985

[2] Gonzalez, R.C., Woods, R.E. “Digital Image Processing”, Addison Wesley, 1992

[3] Gui, V., Lacrămă, D., Pescaru, D. “Prelucrarea Imaginilor”, Editura Politehnica, Timişoara, 1999

[4] Nedevschi, Sergiu. “Prelucrarea Imaginilor şi Recunoaşterea Formelor”, Editura Albastră, Cluj Napoca, 1997

[5] Popescu, C., "Descriptors for Object Recognition Based on Statistical Moments", PERIODICA POLITECHNICA, Transactions on AUTOMATIC CONTROL and COMPUTER SCIENCE,Vol.49 (63), 2004

[6] Popescu, C., "A Contour Based Descriptor for Object Recognition" SACI Transactions, Timisoara, 2004