A Contour Based Descriptor for Object Recognition

Cezar Popescu

"Politehnica" University of Timişoara, Faculty of Automation and Computers, Department of Automation and Industrial Informatics, Bd.V. Pârvan nr.2, 300223 Timişoara, Romania
Phone: +40 0256 403246, E-Mail: cpopescu@aut.utt.ro

Abstract – The paper deals with the extraction of features for the recognition of bidimensional objects. A set of invariants that is based on the contours of the objects is proposed. The invariance to rotation is founded on the properties of the Fourier Transform, applied to a circular signature of the contour. The specific method of creating the signature allows concave objects to be described. Interior contours are also accepted and there is no restriction with regard to the topological aspects of the object.

Keywords: contour, circular signature, Fourier transform, feature extraction.

1 Image Preprocessing and Segmentation

Feature extraction is one of the most delicate stages in object description and recognition, mainly because of the large variety of the objects and of the features of these objects and of the physical magnitudes that can be associated with these features.

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 extraction of the contour of an object may be prior or next to segmentation. The first technique is based on gradient and Laplacian filters. Given a gray level image, described as bidimensional discrete function:

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \ldots & f(0,Y-1) \\ f(1,0) & f(1,1) & \ldots & f(1,(Y-1) \\ \ldots & & & \\ f(X-1,0) & f(X-1,1) & \ldots & f(X-1,Y-1) \end{bmatrix} \tag{1}$$

the gradient of the image is defined as:

$$\mathbf{G}[f(x,y)]=\begin{bmatrix} G_x \\ G_y \end{bmatrix}=\begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} \tag{2}$$

and is a vector indicating the direction in which the change in the gray level is maximum. The magnitude of the gradient is computed as:

$$G[f(x,y)] = \sqrt{G_x^2 + G_y^2} \tag{3}$$

The Laplacian is the second derivative of the image function in a certain point and is given by:

$$L[f(x,y)] = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{4}$$

This approach on segmentation is based on finding discontinuities. The gradient detects edges while the Laplacian detects isolated points and lines. Used together, the two operators can detect both edges in an image and on which side of the edge a certain pixel is located [1].

Gradient and Laplacian operators are usually computed through spatial masks. Edges detected this way are discontinuous of variable width. A process of thinning and connecting is then necessary in order to form continuous, closed boundaries. More robust, but also more computing time consuming methods have also been developed. A "watershed" approach of the gray level profile of an image gives a more accurate evaluation of the direction of the gradient than small size spatial masks [2].

A second approach of segmentation is based on finding similarities in the image. These similarities usually refer to color, texture or gray level. Region growing by pixel aggregation, region splitting and merging and threshold segmentation are the most common methods.

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 [1].

A robust, iterative method of finding the value of the threshold is described in [3]. The method is based on the k-means algorithm [4] and consists in three steps:

a) The mass center of the gray level histogram of the image, μ is computed and is chosen as a first approximation of the threshold T_0.

b) Based on this threshold, the histogram is separated in two and the centers of mass of both partial histograms, μ_1 and μ_2 are computed.

c) The new threshold is then:

$$T = \frac{\mu_1 + \mu_2}{2} \tag{5}$$

The second and third steps are iterated until:

$$|T - T_0| \le \varepsilon \tag{6}$$

As the gray levels are quantized and represented as integers, for computational reasons, ε=1 is a normal choice and the algorithm will converge in less than ten iterations.

2 Contour Description

2.1 Contour Definition and Detection.

The notion of contour is defined in terms of basic relationships between pixels, i.e. neighborhood and connectivity. A pixel *p* at coordinates (x,y) has four horizontal and vertical neighbors, at coordinates (x,y-1), (x-1,y), (x+1,y) and (x,y+1), called 4-neighbours and noted $N_4(p)$. A set of four diagonal neighbors, $N_D(p)$, may also be defined, at coordinates (x-1,y-1), (x+1,y-1), (x-1,y+1), (x+1,y+1). The reunion of these pixels forms the 8-neighbors, $N_8(p)$.

A set of pixels is connected if they are adjacent in some sense of the above and if they share some criterion of similarity. After segmentation, this means that they should belong to the same object. Three types of connectivity may be defined:

- 4-connectivity. Two pixels, *p* and *q* are 4-connected if *q* is in the set $N_4(p)$.

- 8-connectivity. Two pixels, *p* and *q* are 8-connected if *q* is in the set $N_8(p)$.

- Mixed connectivity. Two pixels, *p* and *q* are m-connected if *q* is in the set $N_4(p)$ or, else, *q* is in the set $N_D(p)$ and $N_4(p) \cap N_4(q)$ is empty. Mixed connectivity is introduced in order to eliminate multiple path connections that may appear because the way the 8-connectivity is defined.

Before contour detection the image has to be segmented and each pixel must be marked as belonging to an object or to the background. Then all the contour pixels are detected in only one single pass over the image. The algorithm is nonrecursive and it does not imply a procedure of trekking each contour, pixel by pixel. Each pixel is labeled as belonging to the contour of a certain object if:

a) It belongs to an object.

b) It has at least one 4-neighbour that belongs to the background.

The algorithm produces thinned closed contours by linking pixels that are 8-connected, either to a 4-neighbour, if it exists, or to a diagonal neighbor.

2.2 Contour Description by Chain Codes.

Chain codes are used to represent a contour through a connected sequence of vectors of specified direction and length. The representation is based on 4- or 8-connectivity of the segments, the direction of each segment being represented by a code described by Freedman (Fig.1).

Since the images are sampled on an orthogonal grid, with the same step on both axes, a first approach of the problem would be to follow the contour, pixel by pixel, generating a code for each segment that links two connected pixels [5].This direct approach is not practical because, on one hand, the chain generated this way will be too long and, on the other hand, the noise that appears along the contour, due to imperfect segmentation, will lead to a code that will not reflect the real shape properties of the object.

One solution is to resample the contour applying a grid step that is larger than the sampling step. When following the contour pixel by pixel, a new node in the chain will be added only when the new grid is crossed.

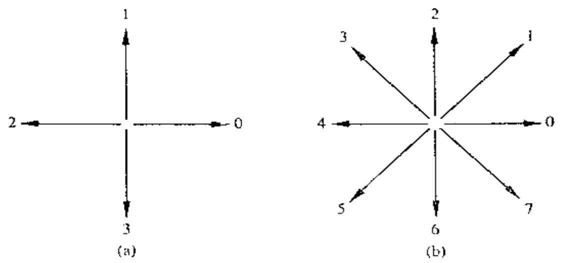

Fig.1

The chain code depends on the starting point. The normalization is done by means of a direct procedure. Given a chain that has been generated starting from an arbitrary point, the code is treated as a circular sequence of numbers representing directions and the starting point is redefined so that the magnitude of the integer number represented by the sequence is minimum. The invariance to rotation is obtained through a differential coding. The invariance to scaling is obtained by modifying the step of the coding grid.

These normalizations are efficient only if the contours themselves are invariant to rotation and scaling, which only seldom happens. In the contrary case, the errors will increase if the sampling resolution of the image is lower.

2.3 Contour Description by Signatures.

A signature is a way of representing the contour of a bidimensional object by means of a one-dimensional function. We assume that this function will be easier to describe. Several means of computing a signature have been developed. The most common is to compute the distance from the centroid of the object to its contour as a function of the angle of the vector that leads from the centroid to the contour.

A signature that is computed this way, in an orthogonal coordinate system, is dependent on the position (translation and rotation) of the object and also on its size. The invariance to translation is easily achieved by translating the origin of the coordinate system into the centroid of the object, thus making the coordinate system relative to the object. The invariance to size is also straightforward if we rescale the signature by normalizing the largest sample to the unit.

For the invariance with respect to rotation a solution that is suggested in [1] uses code chains as in 2.2 in order to obtain a reference point on the contour. However, more difficulties appear in an attempt to do a practical implementation of the method, because the chain is a vector of a variable cardinal that has to be superimposed on the signature. When the signature is then sampled, the number of samples and the angle between samples will also result variable.

2.4 Fourier Transform Based Rotation Invariance.

Let *f(x)* be a continuous function and *F(u)* its Fourier transform:

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx \tag{7}$$

If the function is discretized, then the discrete transform is given by:

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-j\frac{1}{N}2\pi ux} \tag{8}$$

The Fourier transform has a set of interesting and useful properties, among whom the property of translation is expressed by the Fourier pairs:

$$\begin{aligned} & f(x) e^{j2\pi u_0 x / N} \Leftrightarrow F(u - u_0) \quad and \\ & f(x - x_0) \Leftrightarrow F(u) e^{-j2\pi u x_0 / N} \end{aligned} \tag{10}$$

If we express *F(u)* in the exponential form:

$$F(u) = |F(u)| e^{j\Phi(u)} \tag{11}$$

we may note that only the phase angle *Φ(u)* changes with the translation of *f(x)*, while the amplitude spectrum remains unchanged.

Now, let's compute the Fourier transform of the circular signature of an object. If the object is rotated with an angle φ, this equals a translation of the circular signature with the same angle. It then follows that *the amplitude spectrum of the Fourier transform of the circular signature of the contour of an object is an invariant descriptor with respect to the object's translation, rotation and size*.

3 Implementation and Results.

3.1 Algorithm Implementation

In comparison with the chain codes or other methods of achieving the rotation invariance, the above method has several advantages. First of all, it does not need a second, larger grid, which would be equivalent to a subsampling of the image. Second, there is no need to follow the contour in order to compute the descriptor. All the algorithms are straightforward and imply no recursivity. Third, no assumption has to be made about the topological aspects of the shape.

The image is passed through the following procedures during its processing:

a) The color image is converted to gray level. A Gaussian or uniform smoothing filter is applied in order to prepare the image for the segmentation. The image is converted from a standard bitmap format to an internal representation, with one byte per pixel.

b) A threshold segmentation is applied, according to the algorithm described in §1. The image is converted to an internal format with two bytes per pixel. The first byte holds information about the pixel belonging to an object or to the background. A bit is set accordingly to 1 or to 0. The second byte is reserved for object identification.

c) During a single pass over the image, setting another bit in the first byte of their representation marks all contour pixels.

d) All the objects are uniquely identified and numbered, in the second byte of each pixel. The procedure cannot always be done in one single pass over the image. Concave objects may require additional passes. Still, the original algorithm developed for this procedure is nonrecursive and, therefore, very efficient.

e) A circular signature of the contour is computed for each object. Recursivity is avoided again. At the same time topologically complex objects, with concavities and holes, are allowed and are treated in the same manner as simple convex shapes. In order to do this:
   - The centroid is computed.
   - The 2π rad circle is divided into a fixed number of sectors. Then a passage over the object is done and each contour pixel is assigned to the appropriate sector. The Euclidian distance to the centroid is computed.
   - If several pixels are assigned to the same sector, a mean value for the distance is computed.
   - If, due to the small size of an object, some sectors are void, the gaps are filled by interpolation.

f) The signature is scaled assigning the unit distance from the centroid to the largest value.

g) The amplitude specter of the Fourier transform of the signature, F(u), is computed.

The circle is divided into 32 sectors so, due to symmetry, 16 values of the amplitude specter will result. The zero order component of F(u) represents the mean value, that is, the centroid. Since the signature is computed in relative coordinates this value is zero. From the rest of the components only a part of them are included in the signature. The rejection of the higher order components is equivalent with a low pass filter. For small objects the sampling itself is a source of significant noise, especially with respect to rotation.

3.2 Experimental Results

The objects database was created with care to preserve the real-world characteristics of the images but, at the same time, to ensure that they are reproducible. Several sets of characters, in Arial, Times and other fonts, with sizes of 12, 10, 8,6 and 4 points were laser printed on normal paper and scanned at a high resolution in arbitrary positions. Fig. 2 shows a sample of a fragment of an image.

Fig.2

Realistic details of texture, color, shade and imperfections are clearly visible. The characters in the picture are in fonts Arial 12 and 10, scanned at 600 dots per inch.

Fig.3 presents a screen of the application. The image in the background is the original while the image in the front is processed for object segmentation and numbering, contour extraction and signature computation. The window in the lower right side shows a binary coded zoom of a portion of the letter "T", 10 points in size. The window on the upper right side shows a set of measurements of the "T" object, the circular signature and its Fourier amplitude specter in linear and logarithmic representation. The pop-up window shows the results of an object recognition based on a minimum distance classificator.

Fig. 4 presents the signatures of the letter "B" in normal position (left) and rotated with approx. 45 degrees (right). The same data are presented in Fig.5 for the letter "R". The results show clear differences between the signatures, although the shapes seem similar. High frequency noise of significant magnitude is present, as expected, in the rotated images.

Fig.3

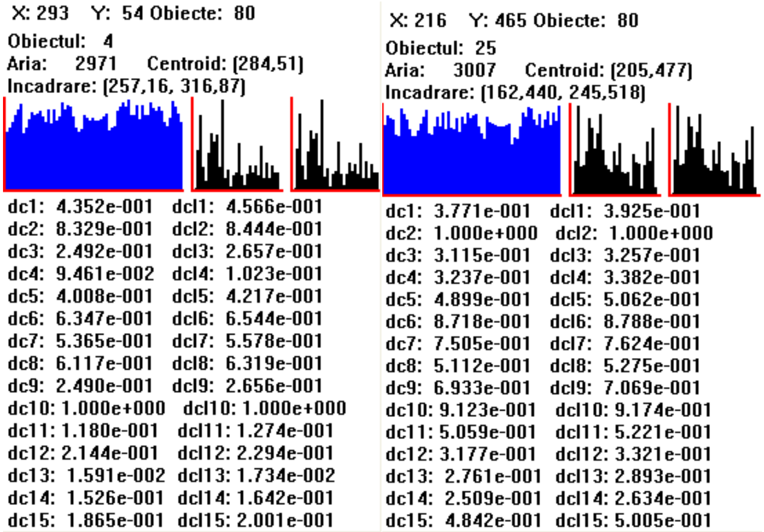

Fig.4

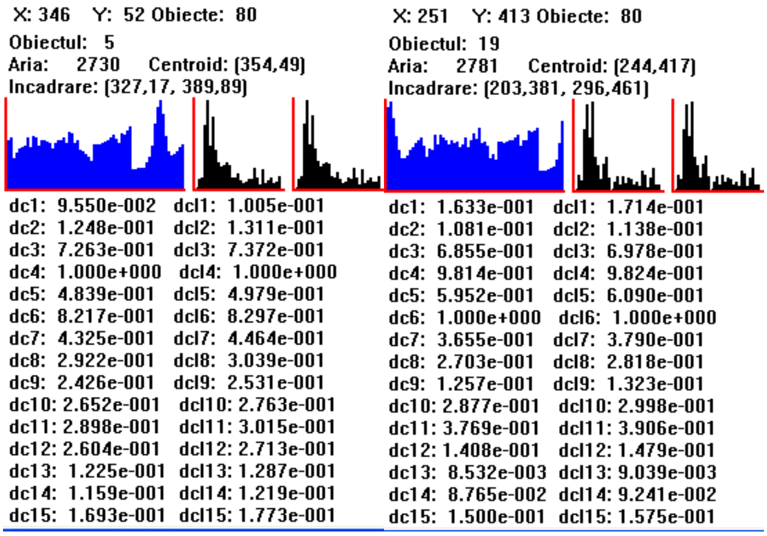

Fig.5

Conclusions

The paper presents a method and an algorithm for extracting features for object recognition from bidimensional objects, based on their contours.

It is assumed that the objects can be really described by means of a bidimensional projection and that they are separable, i.e. there are no superpositions.

The preprocessing of the image consists in conversion to gray level, low pass noise filtering using spatial masks and segmentation. The method of segmentation is not critical or of main importance. The images used in the application consist of dark objects on a white background and have bimodal histograms. That allowed the operation of segmentation to be performed with a robust algorithm. Once separated, objects were identified and numbered with a fast, nonrecursive procedure. Contours were extracted in a single pass operation, based on the relations of connectivity between pixels.

A classic method of using the circular signature as a starting point for a shape descriptor was improved in its theoretical aspects and made reliable by using a property of the Fourier transform to obtain a descriptor that:

- is invariant to rotation
- is insensitive to any origin along the contour

In the implementation aspects an algorithm was developed that is:

- fast, generating the signature in one single pass, without recursive trekking along the contour
- gives descriptors of fixed size, no matter how many pixels belong to the contour
- requires no restrictive assumptions about the topology of the object, in terms of concavity or holes.

The images that were used for the validation and testing of the method share both conditions of reproductibility and natural environment acquisition. The objects shown by the images have all the characteristics of *real*, not simulated or digitally created, artifacts, like industrial parts or other similar objects.

The results show that the method is reliable and may be used for object recognition.

Future work is required to obtain a large set of quantitative results, in different conditions, in order to evaluate this method in comparison with other ones.

References

[1] Gonzalez, R.C., Woods, R.E. Digital "Image Processing", Addison Wesley, 1992.

[2] Gauch, J.M. "Image Segmentation and Analysis via Multiscale Gradient Watershed Hierarchies", IEEE Transactions on Image Processing, Vol. 8, No. 1, pp69-79, January 1999.

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

[4] Duda, R.O., Hart, P.E., Storck, D.G. "Pattern Classification", John Wiley & Sons, 2001

[5] Zingaretti, P., Gasparroni, M., Vecci, L. "Fast Chain Coding of Region Boundaries". IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 20, No. 4, April 1998, pp 407-415.