A FAST, ACCURATE METHOD TO SEGMENT AND RETRIEVE OBJECT CONTOURS IN REAL IMAGES

Giancarlo Iannizzotto , Lorenzo Vita

Facoltà di Ingegneria - Università di Messina
Contrada Papardo, Salita Sperone
MESSINA - ITALY
phone: +39 90 6763012
e-mail: ianni@iit.unict.it, lvita@iit.unict.it

ABSTRACT

1. INTRODUCTION

By definition, an object should be represented in a 2-D image by a well-defined area, i.e. an area completely surrounded by a closed contour. Often, however, this characteristic property is very difficult to be exploited by segmentation algorithms, as contours in natural images are confused and not continuous: they come out from contour extraction procedures as dotted lines, not regular and full of holes, however not closed lines. This is a crucial point, as it leads to a loss of exact cognition of the boundaries between objects.

Most image analysis algorithms require a compact, easy-to-handle representation of the shapes of extracted areas; of course, the preference goes to geometrical models, Superquadrics being an example [1], which can easily vary their shape in such a way to adapt it to the representation of real objects. As the models in question are defined by a set of parameters, they usually adopt function minimization techniques which quantify the residual error in the operation of adapting the shape of the model to that of the object; the sought parameters are, obviously, those which minimize these functions. Since those procedures are generally very expensive in terms of elaboration time, it is important to use geometrical models which permit fast recovery of parameters.

2. OVERVIEW

In order to retrieve contour points, a lot of algorithms and methodologies have been developed during the last years [2] [3]. Gradient-based segmentation exploits the feature of border points of having high values of brightness gradient. Canny [4] developed an optimal methodology to extract border points from images, and although since 1986 many other methods have been developed, Canny's one seems to be yet the most used. Problems arise, however, after the edge point detection, due to the necessity of tracking the points so extracted: points need to be chained toghether, in order to obtain closed, continuous lines, and the chaining process is not trivial. We need to distinguish gaps between consecutive points belonging to the same border from gaps between points not belonging to the same chain; moreover, we need to distinguish real border points from false border points, which may be very frequent and are often originated by noise, texture, and shadows. For these reasons, gradient-based segmentation methods are not so much used, even though they are generally more accurate than the other methods, assuring no false dismissals of border points.

3. OUR SOLUTION

In order to overcome the problems just mentioned, an interesting way has been chosen by Malladi, Sethian and Vemuri [5]: they suggested to start with a geometric model which adapts itself to the object contours iteratively, moving and changing its shape according to the equation of a propagating front (the work started from modelling physics phaenomena such as solid/liquid interfaces). The idea is very good, but the algorithm seems to be rather slow and very expensive in terms of computation: at every cicle, for each front we have to compute an interpolating spline which includes all points of the front, until the front fits completely the border being surrounded, thus terminating the procedure. The direction and the speed of movement are calculated, for every point, starting from the analytical formula of the interpolating curve.

The behaviour of propagating fronts in this approach looks much more like amoebas behaviour just after having surrounded something to eat: it stretches, shrinks, changes shape until it fits the contours of the object just surrounded. Moreover, in presence of objects with holes, it is able to generate something like a son, which fit exactly the internal contours of the holes.

Our choice was to restart from the beginning, keeping only the main idea of emulating in some way the behaviour of a hungry amoeba. Our algorithm emulates a cellular membrane which moves and changes its shape point by point, without interpolating the entire front at every cicle, but mantaining certain constraints related to the underlying nature of the membrane (finite resilience, finite deformability, continuity, etc.). The result is a membrane which, starting from the outer edges of the image, shrinks until it fits perfectly the contours of the contained objects, splitting in two every time it narrows until obstruction and laying down a son internal to the contour being surrounded every time it stops moving (i.e. every time it fits perfectly the external contour of the object), in order to retrieve internal contours.

The direction and the speed of the movement are calculated point by point using a special mask which keeps track of 8-neighboroughs points. This procedure is very fast, since it involves only linear combinations of integer values. The gaps, created between points when points move, are fitted using a very fast linear interpolation. The line is thus kept continuous and closed.

Every time the line meets itself, i.e. a point of the line meets another point of the same line, the closed contour (the amoeba) must split. The algorithm for splitting is rather complex, but the underlaying idea is that the points where the line meets itself are points between two different objects, so the line has to split in two different closed lines.

The points of a line keep on moving until they find some border point (i.e. points which belong to the border of some object). When a line stops moving, it has surrounded completely the outer border of some object. In order to retrieve the internal borders (if there are internal borders), the amoeba lays a son internal to the object, which starts moving in the same manner of the parent. In Figure 1 some experimental results show the good behaviour of our algorithm. It should be noted how the line fits exactly the borders of the objects, and how it distinguishes separate contours, even when there are, in the images, two or more very near objects. Moreover, it can be noted that internal contours are always detected.

The chains resulting from our algorithm are subsequently fitted with Q-Splines [6] [7], a particular family of splines which have the characteristic of being very fast to fit (not interpolated, but least-squares fitted. The adopted algorithm assures a good-performing fitting), so obtaining smooth and accurate contours.

4. CONCLUSIONS

We developed a fast method to segment 2-D grey-level images , obtaining closed-line, continuous, smooth and analytical representations of object contours. The results are very good, even in presence of noisy or low-contrast images, and the shape of objects are retrieved with very good accuracy.

5. REFERENCES

[1] F. Solina and R. Bajcsi. Recovery of parametric models from range images: the case for superquadrics with global deformation. *IEEE Trans. on PAMI*, 12(2), Feb. 1990.

[2] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Addison-Wesley.

[3] Dana H. Ballard and C.M. Brown. *Computer Vision.* Prentice-Hall, 1982.

[4] J. F. Canny. Finding edges and lines in images. Technical Report Tech. Rep. 720, MIT Artificial Intelligence Laboratory, 1983.

[5] R. Malladi, J. A. Sethian, and B.C. Vemuri. A fast level set based alghoritm for topology-independent Shape modelling. University of Berkeley TechReport.

[6] Myron Flickner, Jim Hafner, Eduardo Rodríguez, and Jorge Sanz. Fast least-squares curve fitting using quasi-orthogonal splines. IBM techreport RJ 9819.

[7] Myron Flickner, James Hafner, Eduardo J. Rodríguez, and Jorge L. C. Sanz. Periodic quasi-orthogonal spline bases and applications to least-squares curve fitting in digital images. IBM techreport RJ 9420.

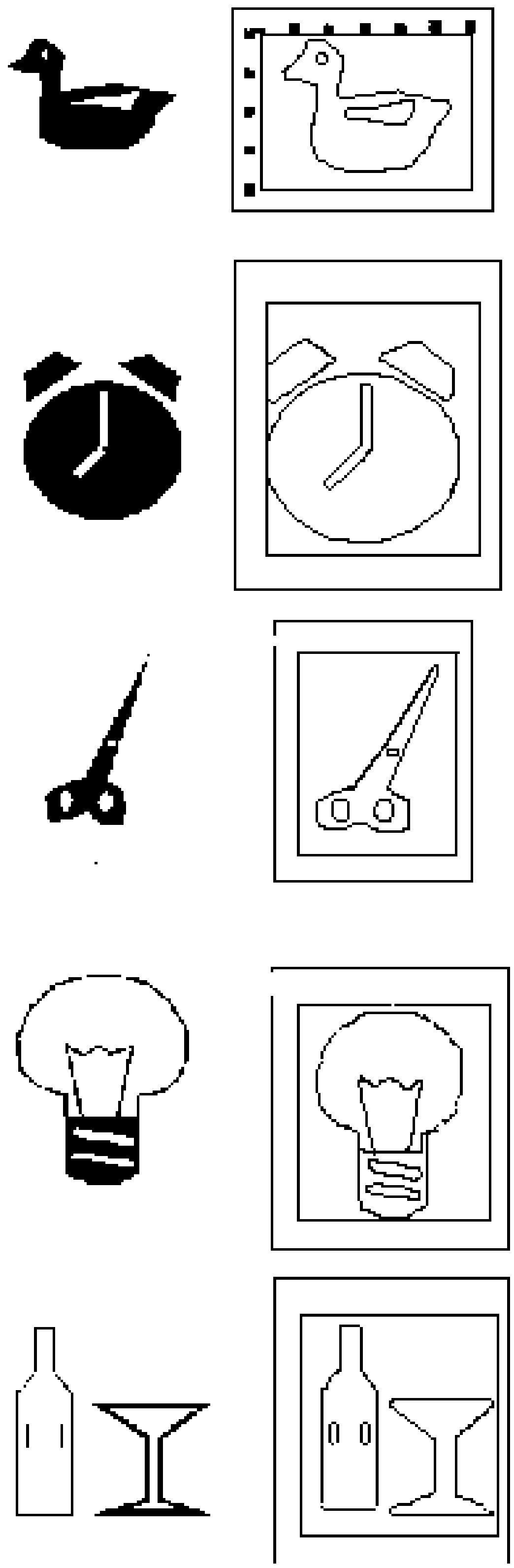

Figure 1: Some experimental results: on the left the starting images, on the right the retrieved contours.