BABEŞ-BOLYAI University of Cluj-Napoca
Faculty of Matematics and Computer Science
Proceedings of the Symposium **"Colocviul Academic Clujean de INFORMATICĂ"**, p. 79–82
Cluj-Napoca, 2003 june 24^{th}

A Method for Contour Continuity Reconstruction

Radu Lupşa[1]

September 2003

Abstract

This paper presents a methot that can be used either as a part of a contour detection algorithm or as a post-processing step for interconnecting the contour fragments detected by a contour-detecting algorithm.

Keywords: edge detection, image segmentation

1 Introduction

Contour detection is the first step in almost every image-analisys system. A large class of image segmentation methods relies on contour detection. Therefore, it is important to have good contour detection algorithms.

Contours are points where image luminosity changes suddenly. Therefore, computing the luminosity gradient is the first step in virtually any edge-detecting algorithm, preceeded sometimes by some image de-noising or some similar image enhancing step.

As the derivative operands, such as the gradient, are very sensitive to input noise, a noise-reducing scheme is almost allways necesar as a pre-processing step in edge detection. The noise-reduction is often either a convolution with a gaussian or a median filter.

However, noise-reducing schemes often have the drawback of blurrying or otherwise significantly altering the image; in extreme cases the de-noised image can be more difficultly understood by a human than the original, noisy, image.

[1] "Babeş-Bolyai" University, Faculty of Mathematics and Computer Science, Department of Computer Science, rlupsa@cs.ubbcluj.ro

Next, the gradient magnitude in a point is interpreted as a strength measure of the edge at that point. Simply thresholding the edge strength, and declaring edge points the points with edge strength measure above the threshold (and with laplacian equal to zero, in order to get zero-width edges) would miss a lot of edges where the luminosity difference between the two sides is too small, and will detect a lot of spurious edges because of the noise. That's why a more complicated algorithm must be used in order to declare that a point is an edge point or not. A simple yet good edge detector is the Canny edge detector [1], where the edge detection is activated at an edge strength above a given upper threshold, and once activated the algorithm follows the edge as long as its strength does not drop below the lower threshold.

Unfortunately, a common problem with most edge detection algorithms [4] is that one or a few pixels where the gradient magnitude is very small are enough to interrupt an edge. On the other hand, humans are able to see an edge even if it has small regions where it is not visible. We have no problem in following a dotted line, for instance.

Another approach to the edge detection is the active contours method (also known as "snakes"). It starts with a circle or another closed curve, placed somewhere (almost) randomly on the image, and then the curve evolves in an iterative process of maximizing some integral quantity and tries to fit on the maximum edge strength. A lot of improuvments have been added to the original algorithm proposed by Terzopoulos et. all [5] for instance for allowing topology changes; however, one must know what he looks for in an image in order to use an active contour method in order to find it.

2 Edge amplification mask

The main ideea behind the algorithm is to detect lines on the maxima (or, more precisely, on the ridges) of the gradient magnitude; and then to prolonge the edges on the (imaginary) lines passing by the edges detected before.

In order to do so, the edge detecting algorithm performs the following steps:

1. Compute, at each pixel, the gradient and the gradient norm; the latter we will call the *edge strength*.

2. Filter the edge strength by performing a convolution with a mask. The convolution will strengthen the points that are already alligned with points with large values for the edge strength.

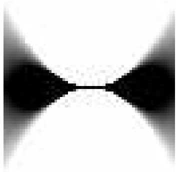

Figure 1: The mask; high values are depicted in dark, low values are depicted in white

3. Finally, apply a skeletonization step on the previous step result, and declare as edges the ridges of those edge strengths.

The fast version of the algorithm computes, in each point, the integral of the product of the edge strength and the mask value with the mask rotated according to the gradient direction in that point, so that continuation edges are searched only along the direction orthogonal to the gradient. A potentially more accurate, yet slower, method is to find the mask orientation that gives the maximum integral value. The latter method is able to retrieve the continuity of a dotted line, where the gradient computed direction at each dot is completely unreliable.

The mask is depicted in figure 2 It is created with the following function:

$$f(x,y) = \begin{cases} \frac{ax^2-y}{a^2x^4} & , \ |y| \le x^2 \\ 0 & , \ |y| > x^2 \end{cases}$$

with $a = 1$.

3 Conclusion and future work

The algorithm described above works with good results. It priviledgiates long edges even with weak contrast over noise, texture or other local luminosity variations.

A promising line for improuvment is to find an adaptative scheme for the size of the mask and for its parameter a.

References

[1] J. Canny A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6), p. 679–698, November 1986*

[2] J. Malik, S. Belongie, Th. Leung, J. Shi Contour and Texture Analysis for Image Segmentation *Kluwer Academic Publishers, 2000*

[3] J. Malik, S. Belongie, Th. Leung, J. Shi Textons, Contours and Regions: Cue Integration in Image Segmentation *IEEE International Conference on Computer Vision, September 1999*

[4] Michael D. Heath A robust visual method for assesing the relative performance of edge detection algorithms *PhD thesis, Univ. of South Florida, December 1996*

[5] M. Kaas, A. Witkin, D. Terzopoulos Snakes: Active contour models *International Journal of Computer Vision, vol. 1, no. 4, p. 321–331, 1987*