Concavity Detection
Using a Binary Mask-Based Approach

Jean-Michel JOLION
Laboratoire Reconnaissance de Formes et Vision, Bât. 403 / INSA
F-69621 Villeurbanne cedex, France
tel : 33 4 72 43 87 59, fax : 33 4 72 43 80 97
email : jolion@rfv.insa-lyon.fr

Abstract: We present in this paper a study of concavity detection using binary edge mask information, and consistency links between them. A model for a corner is proposed as a particular arrangement of edge masks. In a second step, we merge these corner in order to build concavities. Enclosures are obtained as a particular case of concavities. Examples are shown in the context of suburban area and more particularly in house images. We also discuss, about the limits of this approach in terms of information provided by the edge masks and propose some improvements in order to extend our concavity definition.

1. Introduction

The notion of concavity is of importance in image analysis. Biederman, in his theory of human image understanding, Recognition-by-components [Bie-85], assumes that an image of an object is segmented at regions of deep concavity into an arrangement of simple convex generalized cone primitives, such as cylinders, bricks, wedges, and cones. In this approach, concavities are discontinuities in minima of negative curvature. Many other works have been proposed in order to model [Ros-82], detect [Phi-87, Liu-90], or measure [Ros-85] concavities. In this paper, we present a detection technique for concavities using the information provided by binary edge masks [Can-88]. In this approach, we try to avoid numeric decision wherever possible. Using the edge mask labels, we propose a symbolic detection of the concavities. We build high level models by successively relaxing the constraints on the model.

2. Symbolic Pixel Labeling for Curvilinear Feature Detection

Traditionally, curvilinear feature detection has been performed by algorithms that , given an input image, use a local operator to produce a numeric value for each pixel that indicates the likelihood that a line passes through that point. Other algorithms generate line segments using mathematical operators such as the Hough transform [Ill-88]. Still other algorithms find candidate starting points and sequentially follow the curvilinear feature until no more evidence can be found for its continuation (see [Fis-81]). These algorithms may have some control parameters such as thresholds that can be manipulated to improve the output to some extend. There are two problems with this approach when it is used as part of a scene analysis program [Can-88]:

(1) The scene analysis program controls the curvilinear feature extraction algorithm by means of numerical values and thus becomes an "algorithm expert" instead of a "curvilinear feature expert".
(2) The mathematical local operators provide no detailed justification for their decision to include or exclude a given point from the set of curvilinear features. At best, it may give some numeric measure of confidence in its decision, but this is only a summary of separate reasons for labeling the point as part of a curvilinear feature.

The system that we are working on addresses these problems in the following ways:
(1) *Information preservation.* We try to keep all the individual sources of information, rather than replacing them by summary numeric measures, so that later stages can review the evidence used in making early decisions, even if the decision was to ignore the data. This also permits giving the user more detailed explanations of the chain of reasoning.
(2) *Symbolic instead of numeric reasoning.* We try to avoid numeric decision making (such as thresholding confidence measures) wherever possible. The input data, of course, is numeric so some of the reasoning will involve numeric comparisons, arithmetic, etc. Symbolic labels, however, are used at a very early stage to make the reasoning process more explicit.

The problem domain for our curvilinear feature detection system is road network in aerial photographs. However, if we view the gray levels if the image as the third dimension of the image, our system finds ridges (or valleys). The thickness of a curvilinear feature is related to, but not completely dependent on, the sharpness of the peak (or valley); thus as long as a curvilinear feature wider than one pixel has a ridge (or valley), we can detect it. We further restricted our problem domain by not allowing roads to have high curvature, such as hairpin turns. The output of our system is a set of curvilinear feature fragments for an expert system to reason about.

The first step of our algorithm is to extract the edge mask. For each pixel p, we generate a set of masks that describe the pattern of gray levels in the 3x3 neighborhood centered at p (see [Net-87]). Each mask is characterized by the threshold interval that produces it. The size of this interval is called the robustness of the mask (larger the interval, more confident the mask). In our application, we are not interested in all the binary masks that can be associated to a given 3x3 neighborhood. For a complete analysis of such binary masks see [Dup-97]. We focus on binary edge like masks whose list is given in Figure 1.

Considering each mask to be an hypothesis about p, we want to construct consistent interpretations from the hypotheses. Noticing that the masks centered at two adjacent pixels (in the eight-connected sense of adjacency) correspond to two overlapping 3 x 3 neighborhoods, we define mask consistency in the following way:

(1) *Joint robustness greater than O.* There must exist at least one gray level that can serve as a threshold value for both masks. This means that there is a non-zero overlap in the ranges of threshold values (robustness) for the two masks.
(2) *Identical overlap.* The two binary masks must have the same black and white pattern in the region where the two 3 x 3 neighborhoods overlap.
(3) *Geometrical consistency.* The two masks must make geometric sense in the given problem domain. This means that no domain assumptions are contradicted by the combined mask pair.

The first two conditions guarantee that the gray level values of the two masks are compatible. The third condition guarantees that the two masks make geometric sense in the problem domain. Two consistent masks are said to have a consistency link between them, and the transitive closure of these consistency links is used to build connected components representing consistent hypotheses.

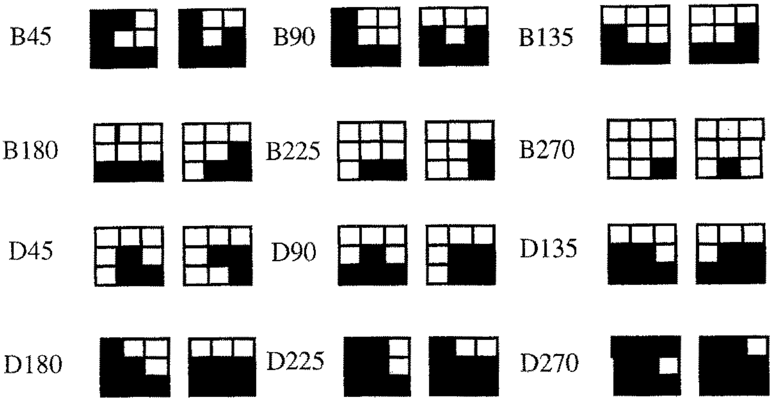

Figure 1 : The classification of edge-like masks

3. Corner Detection

3.1 Definitions and Hypotheses

In the digital case, concavity is not a locally detectable property [Min-88]. However, using edge mask labels, it is possible to define a concept of corner. In the following, we only use the dark masks (masks used when the edge is composed of dark pixels). Equivalent results can be obtained with bright masks. We first propose some definitions and hypotheses we use in this paper.

The edge operator provides us with a set of connected components. Let $C = [(p_i,m_i), i=1,n]$. C is a connected component (or part of) if forall i in [1,n-1], it exists a consistency link, as defined before, between the pixels p_i and p_{i+1} with the masks m_i and m_{i+1}. We note $L(m_i)$ the label of mask m_i.

Hypothesis : A corner is a connected component or a sub-part of a connected component.

This hypothesis is of importance and we will discuss later in this paper (§4) the problem of corner between two different connected components.

3.2 Sharp Corner

The notion of corner is related to the notion of angle and especially to the ninety degree value. An angle can be defined by three points. So, we can defined the first class of corners, called *sharp corner* as:

Let $C = [(p_1,m_1),(p_2,m_2),(p_3,m_3)]$. C is a sharp corner iff:

$$L(m_2) = D90 \textit{ and } L(m_1) \neq L(m_2) \textit{ and } L(m_3) \neq L(m_2)$$

Figure 2 shows an example of sharp corner.

3.3 Smooth Corner

In real data, due to noise and variation of object orientations, real corners are rarely sharp corners. So, we build a new class of corners, called *smooth corner*, as :

Let $C = [(p_1,m_1),(p_2,m_2),(p_3,m_3),(p_4,m_4)]$. C is a smooth corner iff:

$$L(m_2) = L(m_3) \textit{ and } L(m_2) \text{ in } \{D90, D135, D225\} \textit{ and } L(m_1) \neq L(m_2) \textit{ and}$$

$L(m_3) \neq L(m_4)$ *and* $L(m_1)$ in {D180, D135, D225}
and $L(m_4)$ in {D180, D135, D225}

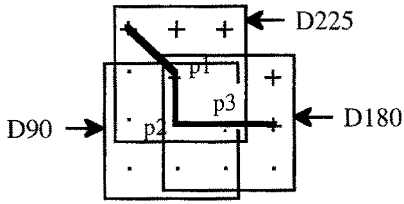

Figure 2 : Sharp Corner (+ and . stand for the pixel locations)

Figure 3 shows an example of smooth corner. As a particular case of this class, we can find two neighbor D90 masks which is not allowed as two sharp corners because of the second and third conditions (see Figure 4).

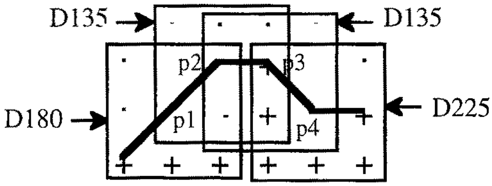

Figure 3 : Smooth Corner (+ and . stand for the pixel locations)

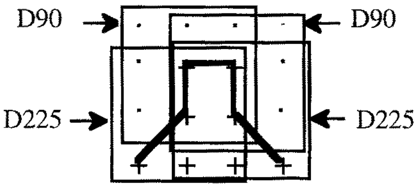

Figure 4 : Double D90 mask (+ and . stand for the pixel locations)

3.4 Large Corners

In order to take into account the different situations which can occurred in real image, we extend the notion of smooth corner to *large corner* using the following definition:
Let $C = [(p_s,m_s),(p_1,m_1),\ldots,(p_n,m_n),(p_e,m_e)]$. C is a large corner iff:
Any part C' of C is not a large, smooth or sharp corner

$L(m_s)$ in {D180, D135, D225} *and* $L(m_e)$ in {D180, D135, D225} *and*
$L(m_1) = L(m_n)$ *and* $L(m_1)$ in {D90, D135, D225} *and* $L(m_1) \neq L(m_s)$
and $L(m_n) \neq L(m_e)$ *and* $[(p_2,m_2),\ldots,(p_{n-1},m_{n-1})]$ is a straight line.

Using the edge mask labels, a straight line can be defined as follows :
$[(p_1,m_1),\ldots,(p_k,m_k)]$ is a straight line iff
for all i in [1,k], L(m–i) in {D180, D135, D225} it doesn't exist (i,j) such that
$L(m_i) = L(m_j)$ and $L(m_i) \neq D180$
for all a in [i+1,j-1], $L(m_a) = D180$

Figure 5 is an example of large corner. We call p_s, the starting pixel, p_e, the ending pixel, p_1 and p_n the curvature pixels, $p_2, \ldots, p_{n-1}$, the internal pixels. (n+2) is the length of the corner. The smooth corner is a particular case of this definition for n=2.

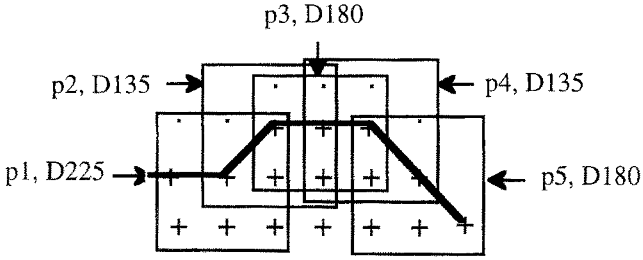

Figure 5 : Large Corner (+ and . stand for the pixel locations)

3.5 Results and Confidence Measure

Figure 6 shows the cul-de-sac image and Figure 7 the result of the edge operator for the house in the mid left. Figures 8 and 9 summarize all the sharp, smooth and large corners extracted. Several corners (sharp or smooth) appear in the same area. Some pixels are used in different corners. This is the result of large number of segments extracted by the mask-based edge operator. From a human point of view, there is only one corner in a corner of a house. In order to find the best corner, a first solution may be to select the best edge segments. As Netanyahu and Rosenfeld shown in the case of mask volume reduction [Net-87], this is not a good solution.

On the contrary, in the mask-based approach, we want to delay as long as possible the feature detection decisions in order to base them on a maximal amount of knowledge. So, we have to keep all the solutions. However, we can provide the higher level with a confidence measure which characterizes the match of a given set of pixels with a given set of masks. We use the confidence measures defined by Netanyahu and Rosenfeld in [Net-87]. For each mask, they compute robustness, contrast, Fisher distance, homogeneity of the foreground, and the foreground road similarity. The first three seem to be the most appropriate evaluation of the goodness of the mask. We extend theses notions from the pixel level to the corner level.

Joint robustness greater than zero is needed for consistency link between two pixels. But this constraint is relaxed for a set of pixels (we are not looking for absolute thresholding effect). So, this measure is not of interest in the corner context. On the other hand, we can use the initial definition of contrast (difference between mean gray level from both sides of the curvilinear feature) and Fisher distance. At the corner level, we have several masks. We have more points than at the pixel level. So, the estimations of means and standard deviations (required for Fisher distance only) are more reliable. But, there is no way to compute these parameters directly from those of the individual masks. Detail of the contrast measure is shown on Figures 8 and 9.

4. From Corner to Concavity

4.1 Definition

Corners, as we defined them before, are local features (except for high values of length, but in this case, is it still corners ?). Concavity is a more global concept. We propose to go from local to global by a merging process. In order to define this higher level model, we need to relax the constraints. Indeed, we cannot keep local constraints in a global context because of noise. So, we decide to remove the preliminary hypothesis on the inclusion of the pattern in a connected component. Thus, we will no longer use the masks. We propose the following definition of a concavity : "*A concavity is a set of corners (sharp, smooth or large) which have a non null intersection composed of connected pixels*".

4.2 A model for concavity

Let $P = [p_1 \dots p_n]$ and $Q = [q_1 \dots q_m]$ be two different corners. The first constraint in the merging process is : $P \cap Q \neq \emptyset$. Due to different index notations, $P \cap Q$ must contain one of the following components:

1) $[p_1 \cdots p_k] = [q_k \cdots q_1] = I$
2) $[p_1 \cdots p_k] = [q_{m-k+1} \cdots q_m] = I$
3) $[p_{n-k+1} \cdots p_n] = [q_1 \dots q_k] = I$
4) $[p_{n-k+1} \cdots p_n] = [q_m \cdots q_{m-k+1}] = I$

Remark : In the following, we will only consider the second type of intersection. Equivalent results can be obtained with the others.

The second constraint is : $P \cap Q$ is composed by connected pixels. So, we must have $P \cap Q = I$ which is equivalent to (considering the other pixels) : $[p_{k+1} \cdots p_n] \cap [q_1 \dots q_{m-k}] = \emptyset$. An example of a concavity is show in Figure 10. In this case, the new concavity is $[q_1 \dots q_m, p_{k+1} \cdots p_n]$. This process is iterated until no such pair of corners or concavities could be merged.

Enclosure is a particular case of concavity. These one cannot be found yet if we allow only one intersection between two concavities. So, we build a new set of constraints (again for the second type of intersection): P and Q form an enclosure iff

it exists k such that $[p_1 \dots p_k] = [q_{m-k+1} \cdots q_m]$
it exists j such that $[p_{n-j+1} \cdots p_n] = [q_{j+1} \cdots q_{m-k}]$
$[p_{k+1} \cdots p_{n-j}] \cap [q_{j+1} \cdots q_{m-k}] = \emptyset$
$k+j < \text{Min}(n,m)$

If the second condition is not verified (*i.e.* j=0 in the third), P and Q form a concavity.

4.3 Results

Figure 11 shows the enclosures and maximal concavities found in the cul-de-sac image. Many large concavities belong to the houses but only one to the end of the end of the road. Absence of concavities, especially in the right down part of the picture, may be explained in different ways. First, it is very difficult to find concavities because there is not enough edge segments in these places. Moreover, they are too small. In that case, the absence of concavities is due to the absence of corners. This situation may also be the result of noise or of particular orientations which are characterized by highly broken lines. Another way to find more concavities is to more relax the constraints. Indeed, we can remove the need of a non null intersection between two corners and allow a gap between them. Then, we obtain a new model called *large concavity*. The gaps must be small (1 or 2 pixels). And, as in many cases,

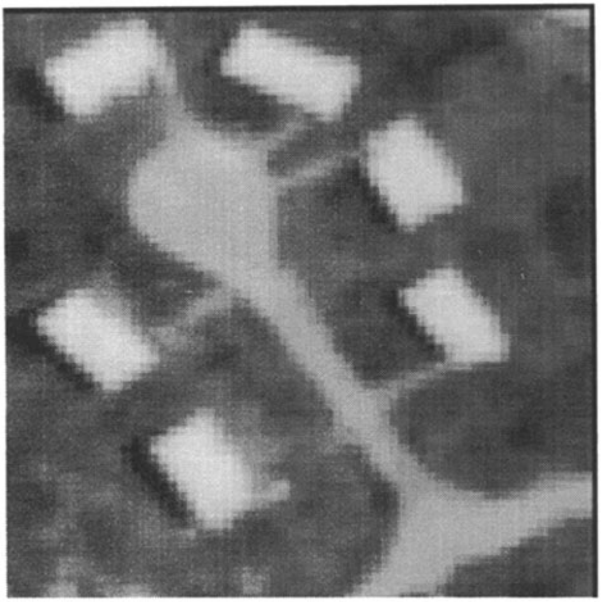

Figure 6 : The Cul-de-sac image

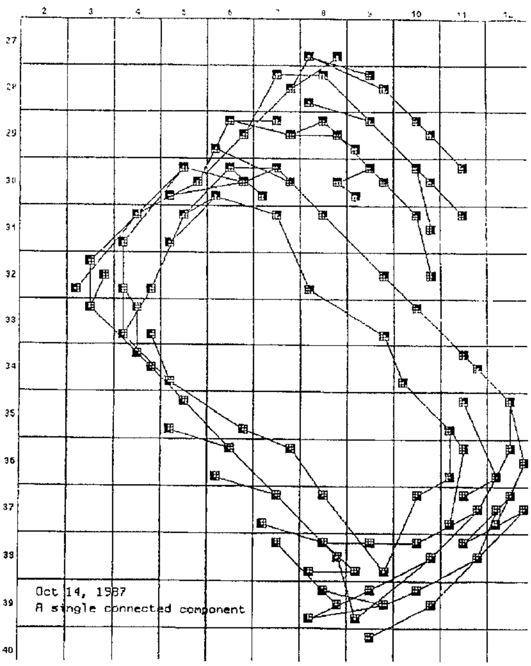

Figure 7 : Result of the edge operator: masks and consistency links

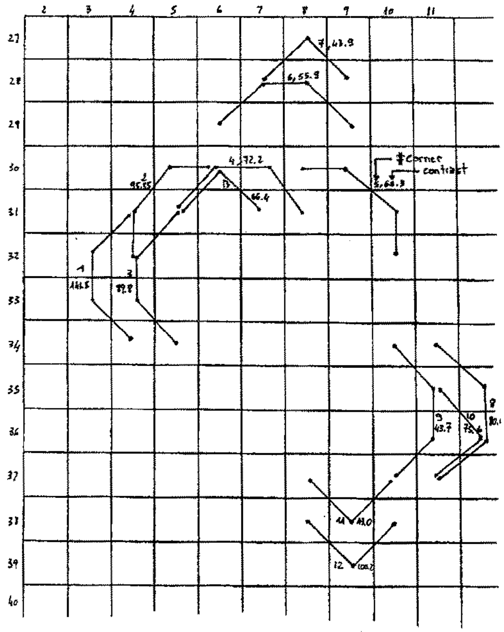

Figure 8 : Sharp and Smooth Corners

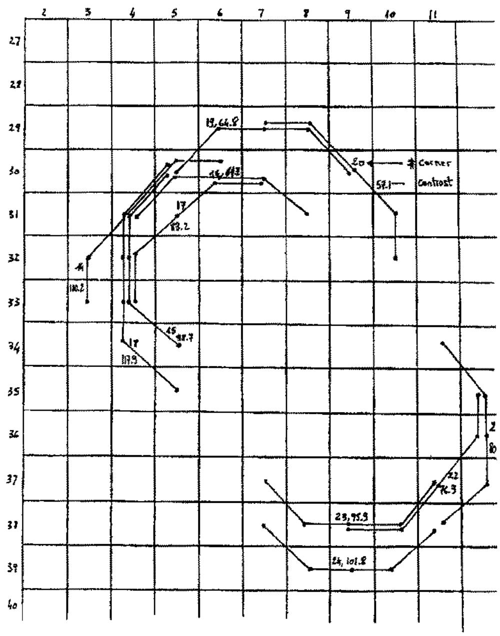

Figure 9 : Large Corners (length in {5,6})

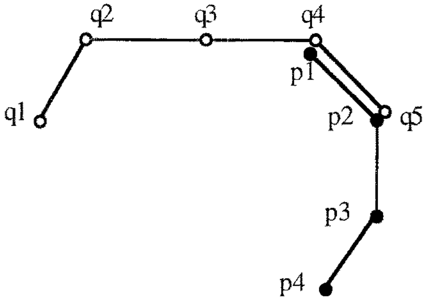

Figure 10 : From Corner to Concavity

we have to add a constraint (less strong than the need of common pixels) on consistent orientations between the two concavities we plan to merge.

5. Discussion

The notion of relaxing constraints, in order to build high level models, by replacing a constraint by a strongless one, is used in all the process we have described from the pixels to the large concavities. Table 1 shows this evolution.

Model	Added constraint	Removed constraint
pixel		
	robustness > 0	
edge mask		
	consistency link	robustness
edge segment		
	ninety degree angle	
sharp corner		
	straight line between two identical masks	two identical neighbor masks
large corner		
	common points between two corners	inclusion in a connected component
concavity		
	consistent orientation	common points between two concavities
large concavities		

Table 1 : From pixel to large concavity

The models and algorithms we used give pretty good results for houses but the quality of the detection highly depends on the quality of the edge detection. Especially, we need long segments whatever are the orientation of the objects. The work of Mraghni [Mra-97] on mask-based edge detection will be of great help to improve these results.

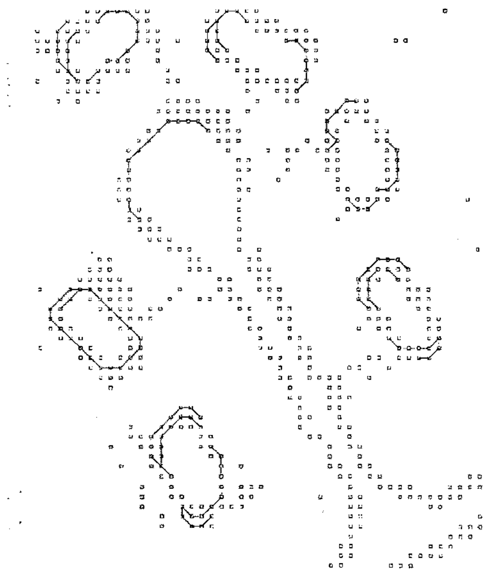

Figure 11 : Enclosures and Maximal Concavities

References

Bie-85 Biederman (I.), Human Image Understanding: Recent Research and a Theory, *CVGIP*, 32, 1985, 29-73.

Can-88 Canning (J.), Kim (J.J.), Netanyahu (N.S.), Rosenfeld (A.), Symbolic pixel labeling for curvilinear feature detection, *Pattern Recognition Letters*, 8, 1988, 299-308.

Dup-97 Duperthuy (C.), Jolion (J.M.), Toward a generalized primal sketch, *8th Int. Workshop on Theoretical Foundations of Computer Vision*, Dagstuhl, Germany, 1996, Springer Verlag, Advances in Computing, 1997, 109-118.

Fis-81 Fischler (M.A.), Tenenbaum (J.M.), Wolf (H.C.), Detection of Roads and Linear Structures in Low-Resolution Aerial Imagery Using a Multisource Knowledge Integration Technique, *Comp. Graph. and Image Proc.*, 15, 1981, 201-223.

Ill-88 Illignworth (J.), Kittler (J.), A Survey on Hough Transform, *Comp. Vision Graph. and Image Proc.*, 44, 1988, 87-116.

Liu-90 Liu (H.C.), Srinath (M.D.), Corner detection from chain-code, *Pattern Recognition*, 23(1/2), 1990, 51-68.

Min-88 Minsky (M.), Paperts (S.), *Perceptrons*, second edition, MIT Press, Cambridge, 1988.

Mra-97 Mraghni (M.C.), Détection de chaînes de contours dans une image numérique par approche symbolique et par grammaire de formes, *Ph.D. Thesis*, Univ. of Tours, France, 1997.

Net-87 Netanyahu (N.S.), Rosenfeld (A.), Mask matching for linear feature detection, *Center for automation research technical report CS-TR-1759*, 1987.

Phi-87 Phillips (T.Y.), Rosenfeld (A.), A method for curve partitioning using Arc-Chord distance, *Pattern Recognition Letters*, 5(4), 1987, 285-288.

Ros-82 Rosenfeld (A.), Kak (A.C.), *Digital Picture Processing*, second edition, Academic Press, 2(11), 1982.

Ros-85 Rosenfeld (A.), Measuring the sizes of concavities, *Pattern Recognition Letters*, 3(3), 1985, 71-75.