



---

CONVEXITY DETECTION BY LINE SCANNINGLA DETECTION DE CONVEXITE PAR BALAYAGE DE LIGNE

Magdy Gaafar

Vision Lab, Univ. of Compiègne, BP 233, Royallieu, 60206, France

---

**RESUME**

En utilisant notre formulation originale de la convexité discrète, nous concevons un algorithme séquentiel pour sa détection. Cet algorithme est conçu pour recevoir et traiter une image lue ligne par ligne. Cet aspect permet de la traiter en temps réel et/ou avec économie de mémoire.

La convexité d'un objet ou d'une région est une propriété importante pour sa caractérisation. La déterminer dans le cas d'une image discrétisée et avec efficacité de calcul doit sensiblement enrichir nos possibilités de segmenter, de décrire et d'interpréter une scène donnée.

**SUMMARY**

Abstract: Convexity is an important pictorial property involved in separating and characterising objects or regions. Line scanning based algorithms are efficiently organised computation structures for image analysis. Starting from our former sequential algorithm based on contour following we describe an original line scan algorithm for convexity detection.



CONVEXITY DETECTION BY LINE SCANNING  
LA DETECTION DE CONVEXITE PAR BALAYAGE DE LIGNE

Magdy Gaafar

1.Introduction

Convexity is a fundamental geometric property. In image analysis it often plays an important role to characterise regions or correspondig objects. It is also used in the elimination of some ambiguities in region separation and segmentation.

In relatively recent work a number of image analysis algorithms took a line scanning approach(1)(4). Besides its elegance, this approach is economic in terms of necessary memory and efficient in terms of the necessary computation. It lends itself easily to specialised hardware architecture with input usually based on electronic or mechanical scanning to generate sequential time signals of the pictorial field(line by line).

Whether a given property could be computed or detected using such an algorithm is an interesting question. We do not try to answer this general question here, but we show how to proceed to reformulate the convexity detection problem creating a line scan algorithm accordingly. Beside satisfying our practical needs this, result is of important theoretic impact in showing its existence.

In what follows we will first summarise the main results of the work in(3) describing a contour following algorithm for convexity detection and deriving the main conditions determinig it. Next we will derive the corresponding formulation and structuring to reach the same goal using line scans. This algorithm inspects the contour segments at two levels of details -gross and fine- within the same scan. We will present these as 'macro' and 'micro' constrains. We also derive a simple first test which is necessary but insufficient for convexity. This gives rise to a simple algorithm based on comparing the successive discrete cellular breadths of region. The complexity is then discussed with the conclusion.

2.Discrete Convexity

To understand the algorithm some acquaintance with the theoretical results concerning discrete convexity are necessary. We will sufficiently summarise the main formulae here, for proof and detail refer to (3).

We defined a segment as of type:

Octant I if the angle "a" of the tangent to any of its points

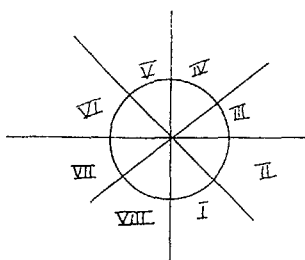


Fig 1. The Octant Types

is between 0 and 45°,

Octant II if "a" is between 45 and 90° ,..., Octant VIII if "a" is between 315 and 360°,

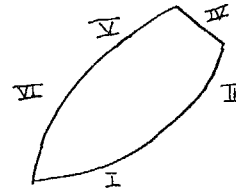
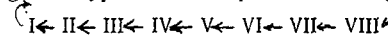


Fig 2. Example

It was shown that: (i) on following the closed contour of a convex region the angle of the tangent at successive points is a monotonic function. Consequently there is a fixed order of segment types in the sequence of clockwise contouring:



Each type of segment (extended to its extremities) occurs only once for a complete turn of the closed contour. The order is circularly symmetric i.e. VIII precedes I. The first octant met could be any one depending on the starting point. According to the shape of the contour some octant types could be completely absent. Thus on following a convex contour we could get the sequence III, IV, VI, VII, VIII, II or V, VI, I, II, IV; ... etc (see example fig 2). This could be represented by a ring state diagram.

(ii) On a more detailed level a segment which is constituted of horizontal or vertical block becomes our inspected entity. The successive lengths of these blocks expressed in number of cells (pixels) m respects certain constrains.

Fig(3) shows the structure of a segment of octant I type. In this case the constrains are :

- a)  $m_{k+1}$  cannot be  $\gg m_k + 2$
- b) if  $m_{k+1} = m_k + 1$  then  $m_{k+2}$  cannot be  $= m_{k+1} + 1$
- c) if  $m_{k+1} = m_k + 1$ , and  $m_{k+i+1} = m_{k+i}$  ( $i=1, 2, \dots, n$ ), then  $m_{k+n+2}$  cannot be  $= 1 + m_{k+n+1}$
- d) It can also be shown using straight lines formulation that

$$\text{if } m_{k+1} = m_k + 1 \text{ and } m_{k+i+1} = m_{k+i} \text{ (} i=1, 2, \dots, n_1 \text{),}$$

$$\text{followed by } m_{k+n_1+2} = m_k, \text{ and}$$

$$m_{k+n_1+2+j} = m_{k+n_1+1+j} \text{ (} j=1, 2, \dots, n_2 \text{)}$$

$$\text{then } n_2 \text{ must be } \leq n_1 + 1 \text{ for convexity.}$$

In other words, recursively analogous constrains hold for the number of repetition of successive blocks of equal length (we will call count) for sequences separated by a constrained single block. Similarly

- e) if  $m_{k+1} = m_k - 1$  and  $m_{k+i+1} = m_{k+i}$  ( $i=1, 2, \dots, n_1$ ) followed by  $m_{k+n_1+2} = m_k$

$$\text{then } m_{k+n_1+3} \text{ cannot equal } m_k, \text{ and if}$$

$$m_{k+n_1+2+j} = m_{k+n_1+1+j} \text{ (} j=1, 2, \dots, n_2 \text{)}$$

$$\text{then } n_2 \text{ must be } \geq n_1 + 1 .$$



CONVEXITY DETECTION BY LINE SCANNING  
 LA DETECTION DE CONVEXITE PAR BALAYAGE DE LIGNE

Magdy Gaafar

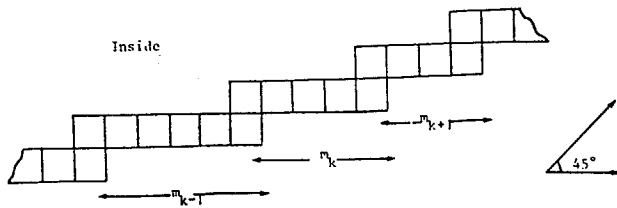


Fig 3. The Block structure

3.Line Scan Macro States

Let us consider a picture with only one region without holes.(The multi-region case only needs some extra pointers' management).If we scan this picture vertically from the top downwards then at any moment we are considering a horizontal strip. If the region is convex then,except for the first and the last,a strip crossing the region coincides with its contour at exactly two locations;each probably consisting of several adjacent cells due to area discretisation.The fact that there are two locations is because,for convexity of a region, any straight line  $P_1QP_2$  with  $P_1$  and  $P_2$  inside the region,  $Q$  must also be inside.Thus any sufficiently long straight line crossing has three segments two outside and one inside the region intersecting its contour once to enter it and once to exit.The constrained sequence order of octant types could be now represented by the shown state diagram(fig 4).Due to fixed scanning direction this relatively simpler state diagram replaces the formerly mentioned ring state diagram.

The subsegment of the digital contour occurring at the current line could be seen as the input to an automaton.We have two intersections at a line giving two subsegments thus two inputs feeding two automata the "Left" and the "Right".At any current line the overall state of the convexity detection system is given by both states of these automata.Since they share the initial input and the final one, and since they have similar initial state and terminal state we could schematically combine their state diagrams to get an interesting conceptual representation of the line scan situation.This situation is represented by both a left state and a right state.The

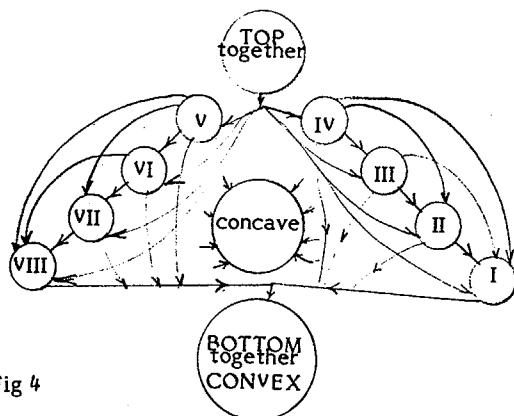


Fig 4

case where we have more than two inputs for the same region gives trivial concavity detection(not shown).

4.The Micro-states

Each of the formerly mentioned macroscopic state represents a current octant type I,II,..VIII .The sequence of octant types as implied by the Bistate diagram although necessary is not sufficient to guarantee convexity. We can miss what is referred to as smooth or gentle concavities.Finer constrains are still required relating the subsegments within each octant type.These are the constrains summarised (for octant I) in section 2 above.

Thus for each Macro-state octant type corresponds a set of micro-states with a lower level automaton analysing the constrains on the subsegments.This automaton examines the orientation and length of successive subsegment ,compares it referring to its current micro-state,updating its micro-state consequently.This comparison and transition reflects the required sequence constrains.It also checks whether it is receiving input belonging to the same octant type, or if not,generating a Macro-state transition.

The input vocabulary set for a micro-automaton consists of the lengths of subsegments of a given type .The automaton or its corresponding algorithm compares the current length wrt the former length and the relations (a to e)of section 2 decides the new state. This is either a normally updated micro-state represented (if no transition to concave state after simple violation) by coding the new length within the new state (case a),or a constrained state(b,c),or a constrained state with higher level counts of the occurrences of successive equal lengths(c,d,e).For octant I we use the flow-chart (Fig 5) which could be seen as a state diagram if the results of the tests( $>2, >1, 1, 0, <1, <2$ ) on the current difference in length are seen as vocabulary symbols.

When we are examining a horizontal subsegment with its corresponding micro-automaton I,IV,V,or VIII length information is found on the same line ;but for the other automata dealing with vertical subsegments, a buffering stage is necessary to constitute the subsegment from the successive single cells having the same abscissa. The buffering stage is also used in both cases to detect macro-state transition and activate the corresponding micro-automaton.It also detects the lateral extrema which(similar to longitudinal extrema i.e bottom and top) could take arbitrary length due to a sensitivity to the exact position of the discretising cellular array ,thus avoiding the useless or misleading test there.In the special case of unity length an extra transition constrain of overlapping intersection at the extremum is necessary.

5.The Differential Run Test

For a discrete binary image runs are the alternating chains of only ones or only zeros in a line. Run length is the number



CONVEXITY DETECTION BY LINE SCANNING  
LA DETECTION DE CONVEXITE PAR BALAYAGE DE LIGNE

Magdy Gaafar

of cells (eg 1's) in the run. If object regions have 1's in their cells and background 0's; and if there is one region (without holes) in such an image, then if the region is convex we have one 1-run (corresponding to a strip of the object) per line. It is interesting to study how the length of this run varies from line to line. Consider a convex region R in the real plane. The distance D between the two points of intersection with a horizontal line  $y=y_1$  is given by:

$$D(y_1) = x_R(y_1) - x_L(y_1)$$

the rate of change wrt y is

$$dD/dy = dx_R/dy - dx_L/dy = g_R - g_L$$

where g denotes the gradient at a given point giving the direction of the tangent to the contour at that point wrt the y-axis.  $g_R$  and  $g_L$  are the gradients at the point of right and left intersection respectively. We will call their difference the run gradient.

To satisfy the monotonic change condition of the tangent for a convex contour (3),  $g_R$  has to be monotonically decreasing and  $g_L$  monotonically increasing as we move in the y direction downwards (fig 6). This means that the difference  $g_R - g_L$  is monotonically decreasing. It could have discontinuities or stay zero as in the case of parallelograms but never increase. This means that for a convex region the run gradient starts positive to create the region, monotonically decreases, and ends negative to close the contour of the region. It is easy to see that the inverse is not true: that a monotonically decreasing run gradient ( $g_R - g_L$ ) does not imply proper monotonicity for the individual  $g_R$  and  $g_L$ . We can only profit from this property to carry out an efficient simple first test which is necessarily true but insufficient to decide convexity.

For a region in a discrete plane the run lengths are integers. The gradients are represented by the lengths of the stair steps. Monotonicity does not imply a monotonic change in these lengths since we can have some occasional sequence-constrained overshoots as described for straight lines (6), (3). These overshoots have a limited value of 1 (cell). Then with a tolerance of 1 from the right intersection and 1 from the left we have a total tolerance of 2. We can construct our test as follows:

On inspecting the region Runs on the successively scanned lines

a) Check that with a tolerance of 1 the Run Length increases or stays fixed until one transition point where it keeps decreasing or fixed until the end.

b) Check that successive Difference in Run Lengths decrease with an occasional tolerance of 2; i.e.  $(D_{i+1} - D_i) - (D_i - D_{i-1}) \leq 2$

By occasional we mean not (equal to two) twice in a row.

It is interesting to see the details of implementation especially how we determine D when an intersection consists of more than one cell. For details refer to (8).

### 6. Conclusion

We have programmed the above procedures and applied them to a number of computer generated discrete ellipses with or without notches. We always got very rapid and correct results on a microprocessor. It is interesting to note that the simplified Differential Run test when also applied on column scanning eliminate together with the line scan a big number of concave cases.

It is interesting to estimate or have a feeling of the complexity of required computation by discussing the number of required states for the different corresponding automata. It is obvious that for the Bi-automaton we need 4+4 Macro-states. For a micro-automaton the states number depends on the numeric range of lengths and counts. Both are related to the curvature of the contour and the alignment of low curvatures relative to the cellular array. Thus contours with long straight lines need more counts. If the lines are nearly parallel to the cellular array the lengths of segments are of higher values -but fewer of them -implying anyway more states. For the Differential Run the numbers range depend on the second derivative of the curve in the scanned direction.

### References

- (1) A.K. AGRAWALA, A.V. KULKARNI, "A Sequential Approach to the extraction of Shape Features", CGIP, vol 6, pp538-557.
- (2) S.M. SELKOW, "One Pass Complexity Of Digital Pictures Properties", JACM, vol 19, No 2
- (3) M. GAAFAR, "Convexity Verification, Block Chords, and Digital Straight Lines", CGIP, vol 6, pp361-370.
- (4) F. VEILLON, "One Pass Computation Of Morphological and Geometric Properties of Objects In Digital Pictures", Signal Processing 1, pp175-189.
- (5) A. ROSENFELD, "Arcs and Curves in Digital Pictures", JACM, vol 20, pp81-87.
- (6) A. ROSENFELD, "Digital Straight Line Segments", IEEE Trans Computers, C23, 1264-1269.
- (7) M. MINSKY, S. PAPERI, "Perceptrons: An Introduction To Computational Geometry", MIT Press, 1969.
- (8) M. GAAFAR, "La Detection de Convexité par Balayage de ligne", Rapport interne UTC, May 1984.

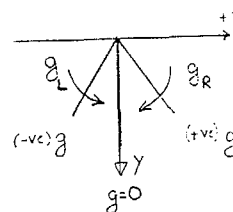
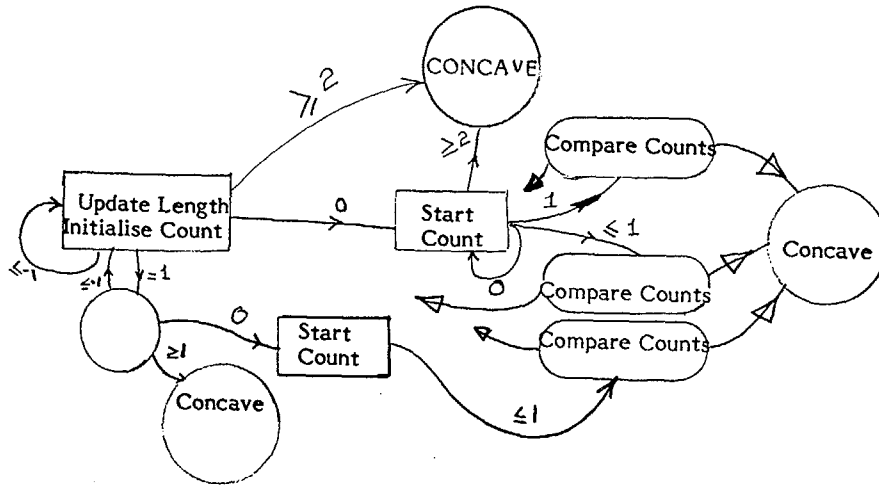


Fig 6.

CONVEXITY DETECTION BY LINE SCANNING  
LA DETECTION DE CONVEXITE PAR BALAYAGE DE LIGNE

Magdy Gaafar



(Fig 5)