# REGION BOUNDARIES ON A TRIANGULAR GRID

Charles T. Zahn Stanford Linear Accelerator Center\* Stanford University Stanford, California 94305

# Abstract

A simple graph model is developed for binary digital pictures on a triangular grid leading to consistent and intuitive definitions of connectivity and region boundaries as well as fast memory-efficient algorithms for computing boundaries and the "insidedness" tree. Boundary encodings are extremely compact and can be smoothed using a discrete implementation of minimumperimeter polygon methods. Attempts to generalize the model to nontriangular grids explain the well-known "anomoly" associated with connectivity on the square grid.

### Introduction

Much picture processing and pattern recognition research has been concerned with inputs encoded on a square grid. Following Rosenfeld<sup>1</sup> (p. 2), we call these binary digital pictures. The present paper concerns several problems encountered in transforming the grid format into a more structured format suitable for describing and/or recognizing the shapes embodied in the picture. The first problem is that of defining a concept of connectivity among points of the digital picture so that the resulting connected components correspond well with one's intuitive sense of connectivity and separateness. This topic is treated at some length by Rosenfeld.<sup>3</sup> Having defined connectivity there is the problem of how to construct an efficient algorithm for labeling the separate connected components (or equivalently detecting what and where they are). Montanari<sup>11</sup> among others has suggested a method for this problem. Each connected component is separated from adjacent connected components of opposite color by boundary curves whose precise definition and computation is the third problem. Some have preferred to construct border curves through extremal points of a connected component but we feel boundaries which fall between adjacent pairs of opposite colored points have more intuitive appeal. Rosenfeld^3 and  ${\rm Zahn}^{10}$  represent different approaches to boundary curve construction, the latter being the forerunner of the approach developed here. Finally, it is of some interest to know which connected components are enclosed by which others, and this information is nicely represented by an "insidedness tree" whose vertices may be components or boundary curves. Algorithms for constructing this tree are given by Montanari<sup>4</sup> and Buneman? See Rosenfeld<sup>1</sup> (pp.135-139) or<sup>2</sup> (p. 161) for brief discussions of these problems and excellent bibliographies.

All methods cited above concern digital pictures on a square grid. In this paper we develop a theory and algorithms for the above problems assuming digital pictures on a triangular grid as exemplified in Fig. 1. The triangular grid is also known as "hexagonal"<sup>6</sup> or "rhombic".<sup>7</sup> Attempts to arrive at a consistent definition of connectivity on the square grid have encountered an "anomoly" which forces 8-connectivity for one color and 4-connectivity for the other.<sup>3</sup> This has the effect of treating two connected shapes differently even if they differ only in color. The anomoly runs somewhat deeper and has been independently acknowledged by a number of authors. Golay<sup>6</sup> has emphasized the isotropic nature of the triangular grid and developed paralled pic-

Work supported by the U.S. Atomic Energy Commission under contract No. AT(04-3)-515 and the National Science Foundation grant No. GJ687 ture processing operators for this grid. Gray<sup>8</sup> also shows an appreciation for the elegance of this grid. In earlier work,<sup>9</sup> we recognized several important ways in which the triangular grid was at an advantage but, until now, we had not carried these hints to their logical conclusion.





In the following sections we develop a simple graph model for binary digital pictures on a triangular grid leading to consistent and intuitive definitions of connectivity and region boundaries and fast memory-efficient algorithms for computing boundaries and the insidedness tree. The boundary encodings can be smoothed (see Zahn<sup>17</sup>) using a discrete implementation of the minimum perimeter polygon methods of Montanari,<sup>4</sup>,11 and Sklansky et al.<sup>12</sup> The resulting data-structure represents connected regions (components) by their boundary curves so that shape comparisons can employ techniques developed for curves (e.g., <sup>13</sup>,<sup>14</sup>,<sup>16</sup> to cite a few).

### Triangular Picture Graphs

A triangular grid is an infinite planar graph each of whose faces is an equilateral triangle and each of whose vertices is incident to exactly six edges. A triangular picture graph is a triangular grid whose vertices are labeled 'black' or 'white' so that the set of black vertices is bounded. Figure 1 depicts a portion of a triangular picture graph as seen through a rectangular window. Although in practice all such pictures will be viewed through a bounded window, it is convenient for the theory which follows to consider infinite triangular picture graphs.

The <u>connectivity graph</u> C(T) of a triangular picture graph T is the subgraph of T consisting of all vertices of T and just those edges (<u>level edges</u>) whose end vertices have the same label. The heavy edges in Fig. 2 identify the connectivity graph for Fig. 1. The graph C(T) can be decomposed into connected components  $\{C_K\}$  each bearing a label 'black' or 'white' inherited from its vertices. For each  $C_K$  let the <u>connectivity</u> <u>region</u>  $R_K$  be defined as a region of the <u>plane contain</u>

1

(To be presented at the Second International Joint Conference on Pattern Recognition, Lyngby-Copenhagen, Denmark, August 13-15, 1974.)

ing all the edges in  $C_K$ . If a <u>level</u> face is one with only level edges, then  $R_K$  shall contain all level faces one or more of whose edges is a level edge of  $C_K$ . The level faces are shaded in Fig. 2.





A contrast edge of T is one with differently labeled end vertices and a contrast face is one with at least one contrast edge. The contrast edges are light in Fig. 2. It is easily seen that a contrast face has exactly two contrast edges. Each contrast face contains a unique  $\underline{contour} \ \underline{segment}$  which is a directed line segment joining the midpoints of the two contrast edges in such a way that the black vertex or vertices of the face lie to the right of the directed line. The contour segments for all (unshaded) contrast faces are depicted in Fig. 2. Some motivation for the term 'contour' is appropriate at this point. It is possible to construct a continuous piecewise-linear function  $F_{\rm TP}$  defined on the plane so that  $F_{T}(\mathbf{v}) = 0$  at white vertices. This is accomplished very simply by defining  $\ensuremath{\mathbb{F}_{\mathrm{T}}}$  on each separate triangular face of T to be the unique linear function whose values at the three vertices of the face are as prescribed by the vertex labels. The contour set of  $F_{T}$  at value 1/2 (i.e., points p with  $F_{T}(p)=1/2$ ) is then a family of mutually nonintersecting simple closed curves which separate black and white areas of the plane. The union of all contour segments defined above constitutes the contour set for FT.

We would like to define a graph based on T which corresponds to the contour curves of FT. It is convenient to use the dual graph for this purpose. Every planar graph G has a dual graph D(G) constructed by making a vertex in D(G) for each face of G and connecting two vertices of D(G) by an edge if the corresponding faces of G share an edge in G. There is thus a one-to-one correspondence between the edge sets of G and D(G) and between the face set of G and the vertex set of D(G). Now let the dual graph D(T) of a triangular picture graph T inherit labeling structure from T as follows: Appropriate vertices of D(T) will be designated as contrast vertices and labeled with the contour segment from the corresponding contrast face in T. Furthermore, edges of D(T) will be directed from vertex  $v_1$  to  $v_2$  if the corresponding edge in T is a contrast edge which has black on the right when crossing it from face  $f_1$  to face  $f_2$  (the correspondents of v, and vo respectively). The boundary graph B(T) is the subgraph of the labeled D(T) consisting of contrast vertices and directed edges. The appropriateness of

this definition is demonstrated by the following theorem which shows the consistency between the connectivity graph C(T) and the boundary graph B(T).

### Theorem 1

If  $C(T) = \cup C_K$  is the connectivity graph of the triangular picture graph T with connected components  $C_K$  defining connectivity regions  $R_K$  and  $B(T) = \bigcup B_1$  is the boundary graph of T with connected components  $B_1$  then

(a) Each  $B_1$  is a directed circuit and the set of contour segments labeling vertices of  $B_1$  forms a simple closed curve  $\gamma_1$ . The edge directions in  $B_1$  are compatible with the contour segment directions in the sense that when edge  $(v_1, v_2)$  belongs to  $B_1$  and  $s_K = p_K, q_K)$  is the contour segment label on vertex  $v_K$ , then  $q_1 = p_2$ .

(b) The curves  $\{\gamma_{i}\}$  form a mutually nonintersecting family of simple closed curves which partitions the remainder of the plane into connected regions  $\rho_{j}$  each containing as a subset exactly one of the connectivity regions  $R_{K}.$ 

# Proof

(a) A vertex v of B(T) corresponds to a contrast face f in T and f has exactly two contrast edges el and e2. When viewed from the center of f, one of these edges is a black-white directed edge (b, w) and the other a (w, b) pair. Suppose  $e_1$  is the (b, w) and  $e_2$  the (w, b). Then  $c_1$  corresponds to an edge of B(T)directed into v and e to an edge directed out from v. Hence any vertex of B(T) has exactly one in-directed and one out-directed edge; hence, the Bi are directed circuits. If directed edge  $(v_1, v_2)$  belongs to B(T)then T contains two adjacent contrast faces fl and f2 with a common contrast edge  $e_{12}$  whose labeling is (w,b) as viewed from  $f_1$  and (b, w) viewed from  $f_2$ . If  $s_K =$  $(p_K, q_K)$  is the contour segment for  $f_K$ , then clearly  $q_1$  and  $p_2$  are both the midpoint of  $e_{12}$ . Each edge of q1 and p, are both the muppent of 12. B(T) indicates that two contour segments from adjacent and since each B, is faces of T have a common endpoint and since each B; is a directed circuit the set of contour segments labeling its vertices must form a closed curve  $\gamma_i$ . The  $\gamma_i$  are simple because a contrast face has one unique contour segment.

(b) It is clear that the family of closed curves  $\{\gamma_i\}$  partitions the remainder of the plane into a set ( $\gamma_1$ ) participants the remainder of the plane line a set of maximal connected regions  $\rho_1$ . Each connectivity region  $R_K$  is entirely contained in one region  $\rho_1(K)$  since otherwise a curve  $\gamma_1$  would intersect  $R_K$  and this is impossible since  $R_K$  is made up entirely of level edges and faces. If two vertices  $v_1$  and  $v_2$  are both in the region  $\rho_1$  then they can be connected by a curve function  $\rho_1$  then they can be connected by a curve function  $P_1$  then they can be connected by a curve function. In the region  $p_j$  call only can be considered  $p_j$  is  $\delta_{12}$  lying entirely in  $p_j$ . It is an easy exercise to replace  $\delta_{12}$  by a curve  $\delta_{12}$  which lies entirely within edges of T as well as being in  $p_j$ . The curve  $\delta_{12}'$  is obtained by replacing each face-crossing subcurve of  $\delta_{12}$  by a portion of the face boundary. In the case of a contrast face the subcurve lies within a triangular or trapezoidal subface but it can still be pulled over to the boundary of the face in an obvious way. The curve  $\delta_{12}$  can be transformed to  $\Delta_{12}$  which consists of a sequence of edges of T simply by eliminating redundant loops in  $\delta_{12}$ . The curve  $\Delta_{12}$  joins  $v_1$  to  $v_2$  inside  $\rho_j$  and consists of edges of T. Now  $\Delta_{12}$  cannot contain a contrast edge for then a point on some  $\gamma_i$  would be inside  $\rho_j$ . Hence,  $\Delta_{12}$  is a level path in T and so  $v_1$ and v2 belong to the same RK. This means that two distinct  $R_{K}$  cannot be in the same region  $\rho_{j}$  . We have thus shown that there is a natural one-to-one correspondence between the connected regions  $\rho_{\rm K}$  determined by the

curves  $\{\gamma_1\}$  and the connectivity regions  $R_K$  determined by the connected components  $C_K$  of  $C(\mathbb{T})$  .

### Endproof

Theorem 1 shows that the regions  $\rho_j$  and  $R_K$  come in natural pairs  $(R_K \subseteq \rho_K)$  so that there is no reason to distinguish the indexing symbols. The connection between  $\rho_K$  and  $R_K$  is actually stronger than what we have stated above. In fact, the part of region  $\rho_K$  not in  $R_K$  is restricted to narrow bands near the boundary curves for  $\rho_K$ , where narrow means no wider than one half the length of an edge of T. The difference in area between  $R_K$  and  $\rho_K$  is thus approximately a linear function of the boundary perimeter of  $\rho_K$ . With these remarks as justification, we shall now restrict our attention to the connected regions  $\rho_K$  and no longer treat the connectivity regions  $R_K$  directly.

There is some more structure relating the curves  $\{\gamma_1\}$  and the regions  $\{\rho_K\}$  which can be captured very naturally in a tree-structure. Each  $\rho_{\rm K}$  except  $\rho_{\infty}$  (the sole unbounded region) has a unique outer boundary curve  $\gamma_1(K)$  so that there is a one-to-one correspondence between the bounded  $\rho_K$  and the curves of  $\gamma_i$ . Henceforth, we use the same indexing symbol and assume  $\gamma_{\rm K}$  is the outer boundary curve for region  $\rho_{\rm K}$ . In addition to its outer boundary, each  $\rho_{\rm K}$  (even  $\rho_{\rm CO}$ ) has zero or more inner boundary curves separating  $\rho_{\rm K}$  from its holes. We define the insidedness tree I(T) for a triangular picture graph T as a directed tree rooted at  $\rho_\infty$  having vertices  $\rho_{\rm K}$  and edges corresponding to boundary curves  $\gamma_{\rm K}$ . The pair ( $\rho_{\rm K}$ ,  $\rho_{\ell}$ ) is a directed edge of I(T) if the outer boundary curve  $\gamma_{\rm K}$  for region  $\rho_{\rm K}$  is also one of the inner boundary curves for  $\rho_{\ell}$ . In this case the edge ( $\rho_{\rm K}, \rho_{\ell}$ ) carries a label  $\gamma_{\rm K}$ . An equivalent way to phrase the condition is that region  $\rho_{\rm X}$  is a subset of one of the holes in region  $\rho_\ell$ . Region labels alternate in color as one passes up or down the tree and clockwise curves are just above black regions and vice versa for counter clockwise curves. The importance of this simple relationship is that in the next section we will show how to compute the boundary curves  $\gamma_{\rm K}$  for a picture T without directly computing the connectivity graph C(T)and so all information about the shape of a region  $\boldsymbol{\rho}_{K}$ and the label (b or w) of region R<sub>K</sub> must be inferred from the curves  $\{\gamma_K\}$ .

# Computing the Tree of Boundary Curves

Let a <u>TV-scan</u> triangular picture graph T\* be a black-white labeling of a partial triangular grid formed by staggered rows of vertices in a rectangular window with the restriction that all vertex labels are white along the outer border adjacent to the unbounded region of the plane. Figure 1 is a TV-scan triangular picture graph. There is no loss of generality here since T\* can always be extended to an (infinite) triangular picture graph T in the obvious way and any T can be transformed to a T\* without loss of information because the set of black vertices is bounded. The only difference between T and T\* for the previous theory is that  $C_{\rm co}$  and  $R_{\rm co}$  are infinite in T while they are connected to the outer border in T\*. Concerning curves  $\{\gamma_{\rm K}\}$  and  $\rho_{\rm co}$  there is no difference at all.

Now we describe a method for computing the boundary curves  $\{\gamma_K\}$  and the insidedness tree  $I(\mathbb{T}^*)$  for a TV-scan triangular picture graph T\*. Let  $r_1$  denote the ith row of vertices in T\* reading from top to bottom with i in the range [0,n]; we shall usually think of each row  $r_i$  as a sequence of edges of T\*. The sequence of triangular faces between  $r_{i-1}$  and  $r_i$  (ordered from left to right) we call the corridor  $\underline{C}_i$ . The corridor sequence  $\underline{\sigma}_i$  is obtained from  $C_i$  by replacing each face by its "undirected" contour segment value  $(/, \backslash, --)$ 

eliminating entries representing level faces. Figure 3 depicts a portion of corridor  $C_6$  from Fig. 2 and demonstrates pictorially much of what we attempt to verbalize in the next few paragraphs. Let the <u>skew sequence</u>  $\underline{S_i}$  be obtained from oi by deleting all flat (---) entries. Then  $\underline{S_i} = (\underline{s_i}, 1, \underline{s_i}, 2, \cdots, \underline{s_i}, 2\underline{m_i})$  can be broken into  $\underline{m_i}$  adjacent pairs where each pair along with the (possibly zero) flat segments in between constitute a connected piece of contour intersecting the corridor. Within the skew sequence some segments touch the top row  $r_{i-1}$  and the others touch the bottom row  $r_i$  and we call these two subsequences the <u>top sequence  $\tau_i$ </u> and the bottom ( $\underline{S_{odd}}$ ) or ( $\underline{S_{even}}$ ). This is because each connected subsequence ( $\underline{s_i}, 2\underline{K}, 1, \cdots, \underline{s_i}, 2\underline{K}$ ) of the full corridor sequence of touches the adjacent rows at the leftmost endpoint of segment  $\underline{s_i}, 2\underline{K}$ . The sequences  $\tau_i$  and  $\beta_i$  are indexed from left to right as subsequences of  $\underline{S_i}$ .





The following theorem shows how to reconstruct the directions for contour segments and how to relate connected subsequences from adjacent corridors. It provides the backbone of an algorithm for constructing the boundary curves  $\{\gamma_K\}$  from the corridor sequences of a picture graph T\*.

#### Theorem 2

A contour segment is directed upward (/ or  $\$ ) if it has an odd index in  $\tau_i$  or  $\beta_i$  and downward (/ or  $\$ ) if it has an even index in  $\tau_i$  or  $\beta_i$ . If  $C_i$  and  $C_{i+1}$  are adjacent corridors with  $\beta_i = (b_1, b_2, \dots, b_p)$  and  $\tau_{i+1} = (t_i, t_2, \dots, t_q)$  then p=q and each pair of segments  $(b_j, t_j)$ intersects at the midpoint of a contrast edge along row  $r_i$ .

### Proof

Consider the row  $r_i$  between adjacent corridors  $C_i$ and  $C_{i+1}$  as a sequence of edges of T\*. Since the leftmost and rightmost vertices of  $r_i$  are on the border of T\* they are labeled w. Eliminating the level edges of  $r_i$  we obtain a left-right sequence  $(e_1, e_2, \ldots, e_{2u})$  of contrast edges such that  $e_j$  is labeled (w,b) or (b,w) according as j is odd or even. Now the bottom sequence  $\beta_i$  for corridor  $C_i$  contains exactly those contour segments of  $\sigma_i$  which touch (i.e., have an endpoint on) row  $r_i$ . Each such segment  $b_i$  must therefore have an endpoint which is the midpoint of some contrast edge in row  $r_i$ . On the other hand, each contrast edge  $e_j$  on row  $r_i$  is adjacent to a contrast face in corridor  $C_i$  and therefore its midpoint  $m_j$  is the endpoint of some segment  $b_j$  in the bottom sequence  $\beta_i$ . We have shown that p=2uand that  $m_j$ , the midpoint of contrast edge  $e_j$ , is an endpoint of segment  $b_j$  in the bottom sequence for corridor  $C_i$ . An analogous argument leads to the conclusion that q=2u and  $m_j$  is an endpoint of segment  $t_j$  in the top sequence for corridor  $C_{i+1}$ . The segment directions for  $b_j$  and  $t_j$  depend on the labeling of contrast edge  $e_j$  and since these labelings alternate (w,b), (b,w),... the first part of the theorem follows immediately.

# Endproof

Figure 3 indicates that once the correct segment direction has been selected for the initial segment in each connected subsequence (i.e., the segments which have odd index in the skew sequence) then the remainder of the segments inherit directions in the obvious way.

We now outline algorithms to construct the boundary curves  $\{\gamma_K\}$  in a TV-scan triangular picture graph T\* and to link those curves into the insidedness tree I(T\*).

### Algorithm B

(1) Pass over the picture T\* in TV-scan order generating a corridor sequence for each pair of adjacent horizontal rows and inserting end-of-corridor codes. Add an end-of-picture code as the final symbol. Call this the <u>full</u> <u>sequence</u> of contour segments.

(2)Process the full sequence from left to right parsing it into corridor sequences and further into connected subsequences by identifying consecutive (odd, even) pairs in each skew sequence. For each segment of the skew sequence determine its direction and assign it to the top or bottom sequence for the corridor. Segments assigned to the top sequence are linked to the appropriate segment in the bottom sequence constructed during the processing of the previous corridor. Segments assigned to the bottom sequence are placed in a list for use by the next corridor. The direction of the initial segment of each connected subsequence is propagated through the remainder of the subsequence and appropriate links are made between adjacent segments.

(3) Pass once through the full sequence looking for the next untraced contour segment. When such a segment s is encountered, then trace through the linked sequence of segments (marking them as traced) starting at s and returning to s. Give this new curve a name and place the name in a list of curves with a reference to segment s as the top of the curve. The initial segment s will always be of type / and if it is linked along the curve direction to the next segment (---) in the full sequence, then the curve encloses a black region; otherwise, it encloses a white region. This information is recorded with the curve name and reference to s. Then the search for an untraced segment resumes directly after s in the full sequence. This is repeated until the end-of-picture code is encountered.

Algorithm B determines the geometry (except position) of each boundary curve  $\gamma_K$  in the picture  $T^*$  and also identifies the label of the immediately enclosed region  $\rho_K$ . The following algorithm constructs a tree which is almost identical to  $I(T^*)$ ; the difference is that the tree computed has vertices corresponding to boundary curves  $\gamma_K$  and is rooted at a fictitious curve at  $\infty$  called  $\gamma_{\infty}$ . As a simple consequence of the one-to-one correspondence  $\{\gamma_K\} \approx \{\rho_K\}$  this difference is

of no consequence. It does seem more natural to construct I(T\*) this way since the  $\{\gamma_K\}$  are the objects which have been computed by algorithm B.

# Algorithm I

(1) For each curve on the list of boundary curves, obtain its top segment s and then find the segment previous to s in the same corridor sequence. If s is the initial segment in a corridor sequence, then link its curve  $\gamma(s)$  to the fictitious curve at  $\infty$  called  $\gamma_{\infty}$ . If segment t precedes s in the same corridor sequence, then link  $\gamma(s)$  to the curve  $\gamma(t)$  containing segment t. The linkage symbolizes the fact that  $\gamma(s)$  and  $\gamma(t)$  can be connected by a curve which does not intersect any other boundary curves. If  $\gamma(s)$  and  $\gamma(t)$  enclose regions with identical labels, then the link implies that  $\gamma(s)$  and  $\gamma(t)$  are siblings in the insidedness tree (i.e.,  $\gamma(s)$  and  $\gamma(t)$  have a common immediately enclosing curve). Otherwise  $\gamma(s)$  will be a child of  $\gamma(t)$ .

(2) When each curve has been given a link it remains only to transform sibling links into the appropriate child link. This can be done by tracing sequences of sibling links until a child link is found and then letting all the intermediate siblings inherit this child link to their common parent. Since all the links point backwards in the full (TV-scan) sequence there will always be an end to sibling links. The child links thus determined define the insidedness tree of boundary curves.

The regions  $\rho_K$  are implicit in I(T\*) in the sense that each vertex with its children represent the outer and inner boundary curves for some region  $\rho_K$ . The degree of multiple connectivity of  $\rho_K$  is given by the number of such boundary curves.

If the position of regions is not important compared to their shape, size, orientation and degree of multiple connectivity, the algorithm B suffices. However, if the position of curves is required the following modifications to algorithm B will do the job.

(1a) Proceed as in step 1 of algorithm B with some additional information encoded into the corridor sequences. Before the first corridor sequence we enter the index of that corridor (i.e., the first corridor containing a contrast face). Subsequently, a corridor index is inserted before a new corridor sequence if and only if the previous corridor sequence contained no bottom segments. Within each corridor every adjacent pair of the form (/ -) will have inserted after it the horizontal position within the corridor of the segment /.

(3a) Proceed as in step 3 of algorithm B except keep track of the corridor index by incrementing it for each corridor sequence which has a bottom sequence and resetting it from the encoded index otherwise. The top of each curve is a pair of the form (/ -) so the position can be determined from the current corridor index and the encoded horizontal position. The reader is referred to Zahn<sup>17</sup> for a discussion of data structures and shortcuts for the implementation of Algorithms B and I.

### Smoothing Boundary Curves

When discrete grid systems are used to represent curves in the plane there occurs the distasteful phenomenon known as <u>quantization error</u>, the most undesirable effect of which is that perfectly straight lines are represented by zigzag polygons. In an attempt to undo this mischief several authors<sup>15,4</sup>, <sup>12</sup>, <sup>14</sup> have proposed methods for smoothing digitized curves. In particular, Montanarill and Sklansky et al<sup>12</sup> define a <u>minimum perim-</u> <u>eter polygon</u> which has the same digitization as a given curve. In the terminology of Montenari<sup>11</sup> the triangular grid is a <u>complete convex digitization scheme</u> (CCDS) and the <u>normal digitization</u> of a boundary curve is the sequence of triangular contrast faces corresponding to the segments of  $\gamma_K$ . The minimum perimeter polygon (MPP) is then the shortest polygon which lies entirely in the same faces and encounters them in the same cyclic order; intuitively visualize a rubber band woven through the contrast faces of  $\gamma_K$ . Although Montanari<sup>4</sup> gives a general method for computing the MPP of a digitization, it involves the solution of a nonlinear programming problem. For the special case of the triangular grid, we have obtained some simple rules (see Zahn<sup>17</sup>) which can be used to generate a good approximation to the MPP of a digitized boundary curve.

### General Picture Graphs

We can attempt to extend the previous ideas to a more general class of picture graphs as follows. Let a planar picture graph (Fig. 4) be a binary vertexlabeled planar graph and let the connectivity graph, level and contrast edges etc. be as for the triangular grid. Let a planar picture graph be called convex if each face is convex. Each contrast face is bounded by an even number of contrast edges, but for non-triangular faces there can be four or more such contrast edges and the pairing up to form segments of boundary is no longer unique. We can choose to join each pair of contrast edges bounding a sequence of level black edges around the face. This choice produces the solid arrows in Fig. 4. Pairing across white edge sequences produces the dashed arrows. Either choice gives a disjoint family of simple closed boundary curves, but neither is compatible with the connectivity graph in the sense of Theorem 1. There is a white vertex in Fig. 4 which is isolated in the connectivity graph but is not separated from the other white vertices by any solid boundary segments. The solid boundaries are compatible with an asymmetric definition of connectivity in which black vertices must share an edge, but white need only share a common face. The need for an asymmetric connectivity on the square grid<sup>3</sup> is a special case of this phenomenon; 8-connected is equivalent to face-connected across the diagonal of a square.



FIG. 4 A convex planar picture graph

These considerations indicate that the previous theory can only be extended to planar picture graphs with triangular faces. The triangular grid is the only regular triangular planar tesselation and has the additional nice property of being relatively isotropic.

# References

- 1. A. Rosenfeld, <u>Picture Processing by Computer</u> (Academic Press, New York, 1969).
- A. Rosenfeld, "Picture Processing by Computer," <u>Computing Surveys</u>, Vol.1, No. 3, Sept. 1969.
- A. Rosenfeld, "Connectivity in Digital Pictures," JACM, Vol. 17, No. 1, Jan. 1970.
- U. Montanari, "A note on Minimal Length Polygonal Approximation to a Digitized Contour," CACM, Vol. 13, No. 1, Jan. 1970.
- O. Buneman, "A Grammar for the Topological Analysis of Plane Figures," <u>Machine Intelligence 5</u> (edited by B. Metzer and D. Michie), American Elsevier, New York, 1970, pp. 383-393.
- M. Golay, "Hexagonal Parallel Pattern Transformations," IEEE Trans. Comp., Vol. C-18, No. 8, Aug. 1969.
- B. McCormick, "The Illinois Pattern Recognition Computer--ILLIAC III," IEEE Trans. Comp., Vol. EC-12, Dec. 1963.
- S. Gray, "Local Properties of Binary Images in Two Dimensions," IEEE Trans. Comp., Vol. C-20, No.5 May 1971.
- C. Zahn, "Two-Dimensional Pattern Description and Recognition via Curvaturepoints," Report No. SLAC-70, Stanford University, Dec. 1966.
- C. Zahn, "A Formal Description for Two-Dimensional Patterns," Proc.Int. Joint Conf. Art.Intelligence, Washington, D.C., May 1969 (also SIAC PUB 538).
- 11. U. Montanari, "On Limit Properties in Digitization Schemes," JACM, Vol. 17, No. 2, April 1970.
- J. Sklansky et al., "Minimum-Perimeter Polygons of Digitized Silhouettes," IEEE Trans.Comp., Vol. C-21, No. 3, March 1972.
- H. Freeman, "On the Digital Computer Classification of Geometrical Line Patterns," Proc.Nat. Elect.Conf., Oct. 1962.
- 14. C. Zahn and R. Roskies, "Fourier Descriptors for Plane Closed Curves," IEEE Trans.Comp.,Vol.C-21, No. 3, March 1972.
- H. Freeman and M. Glass, "On the Quantization of Line Drawing Data," IEEE Trans., Vol.SSC-5, No.1, 1969.
- U. Montanari, "Continuous Skeletons from Digitized Images," JACM, Vol.16, No. 4, Oct. 1969.
- 17. C. Zahn, "Region Boundaries on a Triangular Grid," SLAC Report 149 and Stanford Computer Sci. Dept. Report 72-277, May 1972.