Philips Research Laboratory
Av. Van Becelaere 2 - Box 8
B-1170 Brussels - Belgium

Report R454

Digital processing of binary images on a square grid, I :

Elementary topology and geometry

C. RONSE

June 1981

Abstract

This report is the first one in a general study of binary images digitized on a square grid. We investigate here problems like the choice of a grid, its isotropy, distances, connected components, simple closed paths, borders, edges, the concept of surrounding, etc...

FOREWORD

"Digital processing of binary images on a square grid" is a general study, with both theoretical and practical aspects, of two-tone (i.e. black and white) graphical and alpha-numerical data (like engineering drawings) which are digitized on a square grid through a raster scan.

It will be issued in several reports, numbered from I to VII. We give here the subdivision of that work.

Reports I, II, III : Part I : Topological and geometrical features.

I : Chapter I : Elementary topology and geometry

II : Chapter II : The skeletonization of a figure

III : Chapter III: The extraction of geometrical and topological features of a figure.

Reports IV, V, VI : Part II : Coding and noise cleaning

There are mainly 3 topics

- Run-Length Coding
- Vector Coding and Element Coding
- Noise Cleaning

VII : Part III : Applications :

- Optical Character Recognition
- Digital processing of engineering drawings
- Cybernetics
- Biomedical applications

Important note : In order to simplify the reading of this report, the proofs of the results are presented in the appendixes.

<u>Erratum</u>

Page 1.36 : Theorem 3

Delete point (iv) and replace it by the following :

(iv) Consider configurations of the following form up to a rotation :

(a) $\begin{matrix} x_0 & p & y_0 \\ x_1 & p' & y_1 \end{matrix}$ where $p,p' \in P$

(b) $\begin{matrix} q & y \\ x & q' \end{matrix}$ where $q,q' \in P$ and $k=8$

Then in (a), one of $\{x_0,x_1\}\setminus P$ and $\{y_0,y_1\}\setminus P$ is in $I(P)$ and the other is in $O(P)$, and the same holds in (b) for x and y.

Basic definitions and notations

The following concepts will often be used in one work. We give here their definition and their abbrevation.

- Tessellation : Decomposition of the Euclidean plane into isometric polygons. If these polygons have n sides, then one speaks of an n-tessellation or an n-gonal tessellation. If these polygons are regular, then the tessellation is regular. There are only three regular tessellations : the triangular, square and hexagonal ones.
- World : It is the Euclidean plane which contains the objects which must be represented on a grid, the world objects. World points have real coordinates (x,y), which are called the world coordinates; the x-axis is oriented towards the bottom and the y-axis is oriented towards the right.
- Grid : A simply connected portion of the plane (usually a rectangle or the whole plane) decomposed by a regular tessellation. The grid is identified with the set of polygons of that tessellation.
- Pel : Also called pixel (a contraction of "picture element"). This word designates a polygon in the tessellation of the grid. As said above, the grid is identified with the set of its pels. Each pel can be assigned two coordinates, its grid coordinates.
- Square grid : Grid arising from a square tessellation. Its pels can be grouped in rows and columns. The rows are counted from top to bottom, and the columns from left to right. The pel at the intersection of row i and column j has grid coordinates (i,j). It is practical to suppose that the center of the pel with grid coordinates (i,j) has world coordinates (i,j).

There are two types of square grids :

- the finite square grid, which covers a rectangle. We will write M for the number of rows and N for the number of columns. The rows are numbered 0,...,M-1, and the columns 0,...,N-1.

- the infinite square grid, which covers the whole plane. The rows and columns are numbered by rational integers.

. Grid topology : On a grid G we define a symmetrical and non reflexive relation $\sim$; for two pels p and p' such that $p \sim p'$, we say that p and p' are neighbours or adjacent; the relation $\sim$ is called the neighbourhood or adjacency relation. There are two choices for the adjacency relation, which coincide in the case of the hexagonal grid :

1°) $p \sim p'$ iff $p \neq p'$ and p and p' have an edge in common

2°) $p \sim p'$ iff $p \neq p'$ and p and p' have an edge or a vertex in common.

In the first case, we speak of the restricted adjacency, in the second one of the extended adjacency (or neighbourhood).

The restricted and extended adjacency relations are characterized by the number k of pels adjacent to a given pel p. The adjacency relation is then called the k-adjacency relation.

The ordered pair $(G,\sim)$ is called the grid topology.

. Dual Grid : Given a grid G, let G^* be the set of centers of the pels of G. For any v, $v' \in G'$ corresponding to p, $p' \in G$, join v and v' by a straight segment if $v \sim v'$. Then the resulting configuration is called the dual grid.

When $\sim$ is the restricted adjacency relation, the dual grid forms a tessellation, the dual tessellation of G.

The dual grid can be considered as a graph.

. Grid Representation : It is the function which associates a grid object (i.e. a set of pels) to a world object (i.e. a figure on the real plane). It is a 2-dimensional digitization. The pels in the grid object are coded as black (or 1), and the remaining pels are coded as white (or 0).

. Raster scan : It is the operation by which a two-tone sheet of paper is digitized on a memory. It is the physical correspondent of the grid representation.

. Figure : The set of black pels on a grid.

. Background : The set of white pels on a grid.

. Frame : On a finite grid it is the set of pels which are on the edge of the grid. For example in a square grid with M rows and N columns, it contains the pels having grid coordinates (i,j), where i=0 or M-1, or j=0 or N-1.

. Frame assumption (FA) : It states the following :

- For a finite grid, that the frame is contained in the background
- For an infinite grid, that the figure is finite.

. Restricted frame assumption (RFA) : It states that either the image or its complementary (interverting the figure and the background) satisfies the frame assumption.

. Run length code (RLC) : Given a figure F, one scans the intersection of F with each row (or column) in increasing order. On each row i, there is a succession of black and white runs of pels. A RLC codes F by coding the lengths of successive black and white runs on each successive row (or column). When it scans the rows, it is a horizontal RLC (HRLC); when it scans the columns, it is a vertical RLC (VRLC).

. Black run coding (BRC) : It is similar to RLC, but instead of coding the length of the runs, it codes the position of the first and last pel on each black run. It can be horizontal or vertical (HBRC and VBRC).

. White run coding (WRC) : The same as above, but one considers white runs instead of black runs (It can be a HWRC or a VWRC).

. Vector coding (VC) : This type of coding does not code exactly a figure, but an approximation of it, which depends on some parameters. It is derived from the BRC and so it can be horizontal or vertical (HVC and VVC). In HVC, the figure is described as a union of world trapezes having bases of the form [$(i_0,j_0),(i_0,j_1)$] and [$(i_1,j_2),(i_1,j_3)$], where [j_0,j_1] is a black run on row i_0 and [j_2,j_3] is a black run on row i_1. The approximation of the figure can then be reconstructed by a grid representation.

. Bidirectional vector coding (BVC) : It is a combination of HVC and VVC. It uses HVC for the parts of the figure which are relatively vertical, VVC for those which are relatively horizontal, and any of the two for other parts of the figure.

PART I : TOPOLOGICAL AND GEOMETRICAL FEATURES

Introduction

There is a wealth of scattered litterature on digital two-tone images. Most papers are concerned either with facsimile, telecopy and data compression, or with methods for the extraction of some geometrical and topological features (in relation to character recognition). A few papers [1,5,7,8,9,10] deal with the processing of engineering drawings. In particular, [1,5,7,8] treat the interpretation and editing of handsketched engineering drawings. There are a few theoretical studies of the square grid (see chapter 9 of [12] and [6,11,13] for example).

A deep theoretical study of grid images is needed in order to lay the concepts, algorithms and techniques used in that field on a firm base. It can also help engineers to solve practical problems, like the processing of engineering drawings.

In this way, the problems can be divided in two types :

(i) What are the topological and geometrical features to be considered ? How must they be defined ? How can they be extracted ?

(ii) How to encode and decode the informations contained in a grid picture ? How to minimize or correct the transmission errors ?

These two types of problems are studied in Parts I and II respectively of "Digital processing of binary images on a square grid".

It is necessary, when speaking about features, to specify in which sense they are to be considered. We will use here some concepts introduced in [3]. Features (or attributes) can be divided in three types :

- Physical features : Those which exist on the grid image. For example, there is no physical closure at the top of the letter 0 displayed in Figure 0-1 (which is similar to a facsimile output enlarged 40 times).

- Perceptual features : Those which are visible to the human eye. This supposes that the grid image represents a raster image having a known resolution (for example 8 pels per mm). For example the opening at the top of Figure 0-1 is not visible in the normal scale. So there is a perceptual closure there.

- Functional features : Those which are perceived by the mind because of the context. For example, consider Figure 0-2. It represents a hand-sketched logical circuit. It is visible to the human eye that the interconnections are not formed with straight segments. However one understands that they represent straight segments. So the straightness of these segments is a functional feature.

Examples of perceptual and functional attributes are given in [2,3,4,14]. As noted in these papers, functional attributes depend on what we call :

- The "alphabetic context". Suppose that we have a set of objects (an "alphabet"). Then an object can be recognized not only by its own properties, but by what distinguishes it from the other objects of the set. Let us give two examples :

(i) If we take as set the Roman alphabet from which we delete the letter Q, then if one writes Q, it will be recognized as 0. Conversely, if we take as set the Roman alphabet from which we delete the letter 0, then if one writes 0, it will be recognized as Q. Thus the absence or presence of the dash which distinguishes 0 from Q will be functionally irrelevant.

(ii) If we take as set of objects the set of horizontal or vertical lines, then a line making an angle of 5° with the horizontal will be recognized as horizontal.

- The "syntactic" context. Objects are recognized by their syntactic relations with other objects. For example, the symbol 0 will be read as the letter "Oh" when placed among letters, and as the numerals "zero" when placed among numerals. Other examples are found in [3, 14]. We reprint them in Figure 0-3.

In facsimile, telecopy, photocopy and digital reproduction of art graphics, only perceptual attributes must be taken in account.

In the processing and editing of handsketched engineering drawings, functional features must be extracted, and they are replaced by the corresponding physical features when editing the result.

It must be stressed that there exists no "absolute" physical attributes in the physical world. No machine can draw an "absolutely" straight line.

We will show in Part II that the vector coding method of [9,10] is sufficiently supple for allowing diverse interpretations of functional features of grid pictures.

This work (Part I) is divided in three chapters.

In Chapter I, we study elementary topological and geometrical features of grid images. These features are physical, not functional.

In Chapter II, we study the problem of "line thinning", in other word of the extraction of the skeleton.

In Chapter III, we study other features of grid images, in particular functional features.

REFERENCES

[1] N.G. Altman, "Automatic digitization of engineering drawings", Electro 78, Electronic Show and Convention, Session 23/3, Boston, May 1978.

[2] B. Blesser, T. Kuklinski, R. Shillman, "Empirical tests for feature selection based on a psychological theory of character recognition", Pattern Recognition, Vol. 8, pp. 77-85, 1976.

[3] B. Blesser, R. Shillman, T. Kuklinski, C. Cox, M. Eden, J. Ventura, "A theoretical approach for character recognition based on phenomenological attributes", 1st International Joint Conference on Pattern Recognition, pp. 33-40, 1973.

[4] B. Blesser, R. Shillman, T. Kuklinski, C. Cox, M. Eden, J. Ventura, "A theoretical approach for character recognition based on phenomenological attributes", International Journal of Man-Machine Studies 6, pp. 701-714, 1974.

[5] R.L.T. Cederberg, "Automatic interpretation of hand-drawn electrical schematics", International Conference on Computer Aided Design and Manufacture of Electronic Components, Circuits and Systems, pp. 118-122, July 1979.

[6] S.B. Gray, "Local properties of binary images in two dimensions", IEEE Transactions on Computers, Vol. C-20,n° 5, pp. 551-561, May 1971.

[7] M. Ishii, M. Iwasaki, T. Yamashita, "An automatic input system of logic circuit diagram for CAD", International Conference on Computer Aided Design and Manufacture of Electronic Components, Circuits and Systems, pp. 123-126, July 1979.

[8] W.C. Lin, J.H. Pun, "Machine recognition and plotting of hand-sketched line figures", IEEE Transactions on Systems, Man and Cybernetics, Vol. SCM-8, n° 1, pp. 52-57, Jan. 1978.

[9] W. Pferd, K. Ramachandran, "Computer aided automatic digitizing of engineering drawings", Proceedings of the IEEE Computer Software Applications Conference (Chicago), pp. 630-635, Nov. 1978.

[10] K. Ramachandran, "Coding method for vector representation of engineering drawings", Proceedings of the IEEE, vol. 68, n° 7, pp. 813-817, July 80.

[11] A. Rosenfeld, "Connectivity in digital pictures", Journal of the ACM, vol. 17, n° 1, pp. 146-160, January 1970.

[12] A. Rosenfeld, A.C. Kak, "Digital picture processing", Academic Press, 1976.

[13] A. Rosenfeld, J.L. Pfaltz, "Distance functions on digital pictures", Pattern Recognition, Vol. 1, pp. 33-61, 1968.

[14] R. Shillman, T. Kuklinski, B. Blesser, "Experimental methodologies for character recognition based on phenomenological attributes", 2nd International Joint Conference on Pattern Recognition, pp. 195-201, 1974.

Chapter I. Elementary topology and geometry

§ I. The digitization on a grid and its isotropy

A. What is a grid ?

The raster scan is the operation by which a two-tone image on a sheet of paper is digitized. The surface of that sheet is then divided in picture elements (also called pels or pixels) which form a regular geometric pattern, called a regular tessellation.

A tessellation is a decomposition of a surface on the Euclidean plane into isometric (i.e. metrically equal) polygons. If these polygons are regular, then the tessellation is called regular. These polygons are called cells.

There are three regular tessellations of the plane, the hexagonal, square and triangular ones, portions of which are shown in Figure 1-1.

If a regular tessellation of the plane forms a partition of it, in other words if the cells do not intersect each other on their edges, then it is clear that a cell may not contain its whole border, but only a part of it. A possible choice is given in Figure 1-2, where the vertices which belong to the border of a cell are displayed as blobs, and edges which belong to that border are indicated in full lines (those which do not belong to the border are written as dotted lines).

On the other hand, if one supposes that each cell contains its whole border, then this determines two relations of adjacency (or neighbourhood) between the cells :

- The restricted adjacency relation : two cells are adjacent (or neighbours) if they are distinct and have an edge in common.
- The extended adjacency relation : two cells are adjacent (or neighbours) if they are distinct and have an edge or vertex in common.

The adjacency relation is usually denoted by the symbol $\sim$.

If one represents each cell by a point (located at the center of that cell) and if we interconnect points corresponding to adjacent cells by an edge, then one gets a graph, the adjacency graph. The restricted and extended adjacency graphs are shown in Figures 1-3 and 1-4 respectively. It can be seen that the restricted adjacency graph forms a tessellation, the dual tessellation. The triangular and hexagonal tessellations are dual, the square one is self-dual.

We define the neighbourhood of a cell as the set of cells which are its neighbours. Thus there is a restricted and an extended neighbourhood of that cell.

Let k be the size of the neighbourhood of a cell. The following table gives the values of k in each case :

TABLE 1 : neighbourhood size

Tessellation/Adjacency	Restricted	Extended
Hexagonal	6	6
Square	4	8
Triangular	3	12

It is clear that the restricted and extended adjacency relations are characterized by k. Therefore one speaks of the k-adjacency relation, and the neighbourhood of a cell is called the k-neighbourhood.

We now call a grid a simply connected portion of the plane (in most cases a rectangle, sometimes the whole plane) decomposed in cells by a regular tessellation.

It is a hexagonal, square or triangular grid, according to whether the tessellation is hexagonal, square or triangular. The cells of the grid are called pels or pixels (a contraction for picture elements), because they correspond to the picture elements of the raster scan. The grid is identified with the set of its pels. Every pel can have two colours : black (which is coded as 1) and white (which is coded as 0).

We say that the grid is finite if it can be enclosed in a finite square (or equivalently, if it contains a finite number of pels).

In the case of the square grid, we will generally make a further assumption : if it is finite, then it covers a rectangle, and if it is infinite, then it covers the whole plane.

Now the pels can be assigned integer coordinates, the grid coordinates.

For example, in a square grid, the pels can be grouped in rows and columns. Each row (or column) can be given a number n (where n is an integer). The pel at the intersection of row i and column j has grid coordinates (i,j). When the square grid is finite, we suppose that the rows are numbered 0,...,M-1, and the columns 0,...,N-1. When it is infinite, i and j range over the set of rational integers.

Given a grid G, we call the dual grid the graph $(G^*,\sim^*)$, where G^* is the set of centers of pels of G and $\sim^*$ is a set of edges, joining pairs of elements of G^* corresponding to pairs of adjacent pels in G.

A finite square grid with M=5 and N=4, and its dual grid are shown in Figure 1-5. The plain edges stand for the 4-adjacency, while the dotted edges stand for the 8-adjacency between pels that are not 4-adjacent.

Let us compare the advantages and disadvantages of hexagonal, square and triangular grids respectively.

- In the hexagonal grid, the restricted and extended adjacencies are equivalent, and this removes the duality that we will always encounter in the rest of this chapter when dealing with the topological properties of the square grid. However, a natural system of grid coordinates must have an horizontal axis and a second axis making an angle of 60° or 120° with the first one (because cells cannot be grouped in columns, but only in rows and in "diagonals" forming an angle of 60° or 120° with the horizontal). Thus it priviledges angles of 60° and is practical for digitizing figures whose shape approaches the circle (like a hexagon) [11]. Other properties can be found in [2,3,4,7,10,13,21].
- In the square grid, the adjacency numbers 4 and 8 are powers of 2, which is useful for binary coding of the neighbourhood. It is practical for digitizing rectangular figures (like sheets of paper) and its natural system of grid coordinates has a vertical and a horizontal axis, which is standard. It priviledges angles of 90° and is thus the usual choice for digitizing engineering graphics.
- In the triangular grid, angles of 60° are priviledged. However its types of adjacency are unpractical.

Let us now study the operation by which a continuous two-tone image is digitized on a grid. The objects which must be represented on the grid are subsets of an Euclidean plane, which we call the <u>world</u>. These objects are <u>world objects</u>, they are sets of <u>world points</u> having world coordinates. It is better, especially when one uses a square grid, to choose the y-axis horizontal and oriented towards the right, and the x-axis vertical and oriented towards the bottom, because such a choice is consistent with the matrix notation and with the grid coordinates.

Then one can choose as unit of length the size of each pel, and the square grid can be taken in such a way that the center of a pel having grid coordinates (i,j) has world coordinates (i,j).

It happens that in the digitization through a raster scan, the vertical and horizontal definitions are different. For example in the MBLE telecopier, the horizontal definition is 8 pels/mm, while the vertical definition is 3,85 pels/mm in the standard mode and 7,7 pels/mm in the high definition mode (this follows the CCITT standards) [22]. In this case the grid forms a non-regular tessellation, with rectangular cells. We call it a rectangular grid. We can then suppose that the size (in world length) of a pel is h horizontally and v vertically. Then the grid can be placed in such a way that if a pel has grid coordinates (i,j), then its center has world coordinates (vi, hj).

The rectangular grid is often used instead of the square grid. The two choices are equivalent on the topological point of view, but not on the geometrical point of view, because vertical and horizontal arrays of a certain number of pels have not the same length.

B. The digitization of an image on a grid.

To the physical operation of the raster scan corresponds a mathematical operation called the grid representation. It associates to a world object (i.e. some type of figure on the Euclidean plane) a grid object (i.e. a set of pels, which will be coded as black).

What are the world objects to be represented on the grid ? We subdivide them according to their dimension. An object can be :

- 0-dimensional : a finite set of points (x,y).
- 1-dimensional : a finite union of line segments of the type :

$$\begin{cases} x = f(t) \\ y = g(t) \end{cases} \qquad 0 \leqslant t \leqslant 1, \tag{1}$$

where f and g are of type C_1 (*) (or analytic) and (f,g) is injective.

- <u>2-dimensional</u> : a finite union of surface segments of the type :

$$\begin{cases} x = f(t_0,t_1) & 0 \leqslant t_0 \leqslant 1, \\ y = g(t_0,t_1) & 0 \leqslant t_1 \leqslant 1, \end{cases} \qquad (2)$$

where f and g are of type C_1 (or analytic) and (f,g) is injective.

We will now describe the possible grid representations for these elements when the grid is square. We suppose that the pels are pairwise disjoint as world objects, and so that each pel contains the part of its border displayed in Figure 1-2. We assume also that if a pel has grid coordinates (i,j), then its center has world coordinates (i,j). Thus that pel as a world object is defined by the equations :

$$\begin{aligned} i - \tfrac{1}{2} \leqslant x < i + \tfrac{1}{2} . \\ j - \tfrac{1}{2} \leqslant y < j + \tfrac{1}{2} . \end{aligned} \qquad (3)$$

Now a world point (x,y) is represented by the pel (i,j) if it belongs to that pel (as a world object), in other words if it satisfies (3).

For a world line-segment, there are two types of grid representations, which are called in [6] <u>square-box quantization</u> and <u>grid-intersect quantization</u>. The latter is the most known and widely used grid representation.

In square-box quantization, the grid representation of the line segment is the set of pels containing points of that segment. Thus it is the set of pels having grid coordinates (i,j) such that there exist some x and y satisfying both (1) and (3).

(*) A function is of class C_1 if it is derivable and its derivatives are continuous.

In grid-intersect quantization, one looks at the intersections of the line segment with the edges of the dual grid, and for every such intersection point, we take the pel which contain it. In other words, it is the set of pels having grid coordinates (i,j) such that there exist some x and y satisfying both (1) and the following :

$$x = i \text{ and } j - \frac{1}{2} \leqslant x < j + \frac{1}{2}$$
$$\text{or} \qquad (4)$$
$$y = j \text{ and } i - \frac{1}{2} \leqslant y < i + \frac{1}{2}$$

These two representations are illustrated in Figure 1-6.

Moreover, if one scans the line by making t go continuously from 0 to 1, then the grid representation of the line segment forms a sequence of 8-adjacent pels. In the case of the square-box quantization, they are 4-adjacent, except when the line segment goes down through the upper left corner of a pel, an event which has probability 0 for random tracings. On the other hand, in the grid-intersect quantization, successive pels in that sequence may often be 8-adjacent but not 4-adjacent (according to [6], this may happen with a probability of 0,41 for random tracings).

We will not consider other possible methods for the grid representation of world line segments.

For world surfaces, we can also define the square-box quantization and the grid-intersect quantization.

In square-box quantization, the grid representation of a surface segment is the set of pels containing points of that segment. Hence it is the set of pels having grid coordinates (i,j) such that there exist some x and y satisfying both (2) and (3).

In the grid-intersection quantization, it is the set of pels having grid coordinates (i,j) such that there exist some x and y satisfying both (2) and (4).

Other grid representations of surfaces, as the thresholding of the black area on each pel, will not be considered.

For a rectangular grid, the same as above can be done, but we must replace x and y by x/v and y/h respectively.

Similar grid representations can be defined for the hexagonal and triangular grid.

C. Isotropy

When an image is digitally processed through certain techniques, then certain geometrical patterns become priviledged (for example an array of rectangles in Figure 6b of page 123 of [19] , and vertical lines in Figure 13 of page 92 of [22]). This is a restriction on the isotropy of the processing, because an isometry which does not preserve the array changes the result of the processing.

We will give here a more formal definition of isotropy. Let $\approx$ be a relation of "similarity" on processed images, where $A \approx B$ means something like :

"B has the same number of connected components as A",

"B is perceptually equal to A",

"B is equal to A", etc...

Let $\mathcal{P}$ be a processing which can be applied to any image I covering the whole plane, provided that the black (or non-white in the case of images having more than two tones) part of I can be enclosed in a finite frame. We suppose that for any finite image I, the processed image $I\mathcal{P}$ is the portion of $J\mathcal{P}$ enclosed in the frame of I, where J is the infinite image consisting of I surrounded by white.

Let I be such an infinite image whose non-white part can be enclosed in a finite frame.

Let π be an isometry of the Euclidean plane. For any object X, write $X\pi$ for the image of X by π. Then we define :

$$IP^{\pi} = I\pi \; P\pi^{-1} \tag{5}$$

We say that P is isotropic for π with respect to $\approx$ if for any such image I we have :

$$P^{\pi}(I) \approx P(I) \tag{6}$$

The set of such isometries π is the isotropy set of P with respect to $\approx$.

The following fact can easily be shown :

The isotropy set of P with respect to $\approx$ is a group if and only if $\approx$ is an equivalence relation on the set of processed images $P(I)$.

It is then called the isotropy group of P with respect to $\approx$ and write it $\Gamma(P,\approx)$.

Now the strongest equivalence relation is the equality = (physical, not perceptual or functional). We will then write "isotropy" for "isotropy with respect to =", and we will write $\Gamma(P)$ for $\Gamma(P,=)$.

Let us go back to the grid representation. It is clear that the isotropy group of the two grid representations defined above (square box quantization and grid-intersect quantization) is the symmetry group of the infinite grid (i.e. the group of all isometries preserving that grid).

In the square grid, this group is the product of the group of integer translations

$$\tau(a,b) : (i,j) \rightarrow (i+a,\ j+b), \qquad (7)$$

where a,b are rational integers, and the group of the 8 symmetries of the square (identity, 2 rotations of $\pm\ \pi/2$, central symmetry, 2 median symmetries and 2 diagonal symmetries).

Under the action of that group, all rows and columns of pels are in the same orbit. Thus none of them has a priviledged role, and the two directions are equivalent.

On the other hand, in the rectangular grid, the isotropy group is the product of the group of integer translations and the group of the 4 symmetries of the rectangle (identity, central symmetry, 2 median symmetries). Here all rows are equivalent, all columns are equivalent, but the horizontal and vertical directions are not equivalent. This may induce on the processed image a texture of vertical or horizontal lines (cfr. Figure 13 page 92 in [22]).

In the hexagonal and triangular grids, the isotropy group is the product of the group of integer translations and of the group of the 12 symmetries of the regular hexagon. Here the horizontal direction is equivalent to the two directions making an angle of 60° at 120° with it.

Clearly the isotropy group of the square grid is the most suitable for the processing of engineering or technical drawings. In the rest of this study, we will restrict ourselves to the square (or sometimes rectangular) grid.

§ II. The choice of the topology on a square grid

Let G be a square grid. As explained above, the pels of G can be grouped in rows and columns, and the pel at the intersection of row i and column j has grid coordinates (i,j). If G is finite, then i and j range over {0,...,M-1} and {0,...,N-1}, respectively. If G is infinite, then i and j range over the set of rational integers.

An image on G is a repartition of the pels of G in two tones (black and white). It can be seen as a map $G \to \{0,1\}$, where 0 generally corresponds to white and 1 to black.

If G is finite, then we call the frame of G the set of pels which are on the edge of G, in other words whose grid coordinates (i,j) satisfy the following condition :

$$i.(M-1-i).j.(N-1-j) = 0 \qquad (8)$$

The frame of G will be written FG.

Given an image on G, the set of black pels is called the figure and is written F, while the set of white pels is called the background and is written B. (Note that in [19] another meaning is given to that word).

The following assumption, called the frame assumption (abbrevition: FA), is often made on images (see for example [16]) :

- For a finite grid, it states that the frame belongs to the background (i.e. $FG \subseteq B$).
- For an infinite grid, it states that the figure is finite.

This requirement introduces a dissymmetry between the black and white pels of the image.

In § I we defined the two basic adjacency relations on the square grid, characterized by the neighbourhood size k of a pel outside the frame, and called k-adjacency (k = 4 or 8).

In the following, k will always designate one of the two numbers 4 and 8, and k' will designate the other one (k' = 12-k).

One might assume that the same k-adjacency can be chosen for the figure and the background. But this causes problems. Consider for example the two images displayed in Figure 1-7. If we use 4-adjacency on both the figure and the background, then in (a) a disconnected line disconnects the background, while in (b) two simply connected surfaces, which are not connected together, disconnect the background. On the other hand, if we use 8-adjacency for both the figure and the background, then in (a) a connected line does not disconnect the background, while in (b) a connected figure has a hole which is connected to the exterior. Both cases run against one intuition of the topology of figures on a plane. The contradiction can be lifted by using k-adjacency on the figure and k'-adjacency on the background (but the type of adjacency between pels of the figure and neighbouring pels of the background remains unspecified, which is not an important problem; we will not encounter it in that study). A more formal version of that argument can be found in [4,9,16], using the Euler number of a triangulated figure [8].

This choice for the adjacency relations on the figure and the background is classical. We will now study other possible choices. The basic idea is to transform the square tessellation into an equivalent one using distorted squares.

Suppose that whenever a black pel has an edge in common with a white one, we clip the corners of the black pel which are on this edge, as in Figure 1-8. Then the ambiguity arising from two black pels and two white ones having a corner in common (as in Figure 1-7) is removed. If one considers as adjacent two pels having an edge in common, then the following adjacency relations must be considered (see Figure 1-8) :

(a) Two 4-adjacent pels are adjacent.

(b) Two 8-adjacent pels p_1 and p_2 are adjacent if and only if one of the following holds, where p_3 and p_4 are the two pels which are 4-adjacent to both p_1 and p_2 :

(i) p_1 and p_2 are both white and p_3 and p_4 are not both white.

(ii) p_1 and p_2 have different colours and p_3 and p_4 are both black.

Of course, one can clip the corners of white pels and then one gets a dual adjacency relation.

Now it is possible to define a 6-adjacency for a square grid in order to avoid the ambiguity encountered above. In Figure 1-9 we show the configuration formed by the centers of pels in a square and a hexagonal grid. We see that they are similar up to a rotation of 30° of the x-axis. So by rotating the x-axis by 30° in the hexagonal grid, the configuration formed by the centers of the pels becomes the same as in the square grid. But then the hexagonal grid becomes the grid of Figure 1-10 (a), in which pels have the same position as in the square grid, but where we have a 6-adjacency relation. Here pels which have a vertex in common have an edge in common, and so the ambiguity seen above is avoided. Two topologically equivalent grids are shown in Figure 1-10(b) and (c). However, the isotropy group of a square grid having that 6-adjacency relation is smaller than the original one. In particular, a diagonal from top left to bottom right is not equivalent to one from top right to bottom left; the first one is connected, and the other one is not.

In page 3 of [19], another type of 6-adjacency relation is given. It is based on the shifting of odd rows in Figure 1-9 (b). We illustrate this adjacency relation on Figure 1-11. Here the isotropy group becomes even more restricted.

In the rest of this work we will use the classical choice for the adjacency relation : k-adjacency for the figure and k'-adjacency for the background ($\{k,k'\} = \{4,8\}$).

§ III. Paths and distances

For k=4 or 8, we call a k-path of length n a sequence $(x_0,\ldots,x_n)$ of pels such that for s=0,...,n-1, $x_s \neq x_{s+1}$ and x_s is k-adjacent to x_{s+1}. If $x_n = x_0$, then it is a closed k-path.

Note that a 4-path is an 8-path.

Let us now define distances on a grid.

When dealing with distances in the world, one usually uses the Euclidean distance, which is invariant under the group of isometries. But this choice is not evident for a square grid, because that grid is not invariant under the group of all isometries, but only under its group of isotropy. We will say that a real-valued nonnegative function d defined on the ordered pairs of pels is a distance if it satisfies the following 3 conditions for all pels x, y and z :

$d(x,y) = 0$ if and only if $x=y$.

$d(x,y) = d(y,x)$ (symmetry).

$d(x,z) \leqslant d(x,y) + d(y,z)$ (triangularity). (9)

Let x and y be two pels having respective grid coordinates (x_0,x_1) and (y_0,y_1). The most obvious example is the Euclidean distance d_e defined by :

$$d_e(x,y) = ((y_0-x_0)^2 + (y_1-x_1)^2)^{1/2} \quad (10)$$

Two other wellknown distances are the city block (or Manhattan) distance d_4 given by :

$$d_4(x,y) = |y_0-x_0| + |y_1-x_1|, \quad (11)$$

and the chessboard distance d_8 given by :

$$d_8(x,y) = \max\{|y_0-x_0|, |y_1-x_1|\}. \quad (12)$$

They are illustrated in Figure 1-12.

These two distances are integer-valued. Moreover, they are <u>regular</u>. One says (see [19]) that a distance function d is <u>regular</u> if and only if it satisfies the following 2 conditions :

- d is integer-values
- for any two pels x and y, if $d(x,y) \geqslant 2$, then there is some pel z such that $x \neq z \neq y$ and

$$d(x,y) = d(x,z) + d(z,y). \tag{13}$$

If d is regular, then the following holds :

- If $d(x,y) = k$, then there exist a sequence $x_0 = x, x_1, \ldots, x_k = y$ of pels such that for any $i=0,1,\ldots,k-1$, $d(x_i,x_{i+1}) = 1$. (14)

By taking the graph whose vertices are the pels and whose edges are pairs of pels at distance 1, then d is equivalent to the distance defined on that graph (the distance between 2 vertices is the length of the shortest edge-sequence joining them). The converse is also true : if d is a distance on the grid defined from a graph, then d is regular (this result is well-known).

Moreover, the following holds :

$$d_k(x,y) = 1 \text{ if and only if x and y are k-adjacent (k=4 or 8).} \tag{15}$$

It follows then from (14) that for any two pels x and y, $d_k(x,y)$ is the length of the shortest k-path from x to y.

It is easily seen that d_4 and d_8 are invariant under the isotropy group of the grid. This is natural, because d_4 and d_8 can be defined in terms of 4- and 8-adjacencies, which are isotropic concepts. There are other isotropic regular distances (for example the distance based on the movements of the horse in chess, where $d(x,y) = 1$ if and only if $\{|x_0-y_0|, |x_1-y_1|\} = \{1,2\}$).

Other distances can be found in [21].

Given a pel x, two sets Y and Z of pels, and a distance d, we define :

$$d(x,Y) = \min\{d(x,y) | y \in Y\} = d(Y,x). \quad (16)$$

$$\begin{aligned} d(Y,Z) &= \min\{d(y,z) | y \in Y, z \in Z\}. \\ &= \min\{d(y,Z) | y \in Y\}. \\ &= \min\{d(Y,z) | z \in Z\}. \end{aligned} \quad (17)$$

Given also a property P (like = r, > m, etc..), we can define

$$x^{dP} = \{p \in G | d(x,p)P\}. \quad (18)$$

$$Y^{dP} = \{p \in G | d(Y,p)P\}. \quad (19)$$

For example $x^{d_4 > 4} = \{p \in G | d_4(x,p) > 4\}$, $y^{d_8|60} = \{p \in G| d_8(Y,p)|60\}$.

In particular, we write

$$N_d^r(U) = U^{d=r} \quad (20)$$

and

$$S_d^r(U) = U^{d \leqslant r}, \quad (21)$$

where U is a pel x or a set Y. For $d = d_k$, write :

$$N_k^r(U) = N_{d_k}^r(U) \quad (22)$$

and

$$S_k^r(U) = S_{d_k}^r(U), \quad (23)$$

where $k = 4$ or 8.

Clearly $N_k^1(x)$ is the k-neighbourhood of x for any pel x. When r=1, we omit it and write $N_d(U)$, $S_d(U)$, $N_k(U)$ and $S_k(U)$.

Let us now give methods for computing the sets $S_d^r(U)$ for a regular distance d.

The first one comes from [21] :

Let d be a regular distance. For any image $\phi : G \to \{0,1\}$, define the image $f(\phi)$ as follows :

For any pel x,

$$f(\phi)(x) = \min\{\phi(y) \mid y \in S_d(x)\} \tag{24}$$

Let U be a set of pels. Then the sets $S_d^r(U)$ are computed as follows :

Let ϕ be the image having U as background. Define

$$\begin{aligned} \phi_0 &= \phi \\ \phi_{m+1} &= f(\phi_m) \text{ for } m=0,1,2,\ldots \end{aligned} \tag{25}$$

Then $S_d^r(U)$ is the background of ϕ_r for r=0,1,2,... In other words, for any pel x we have :

$$\begin{aligned} \phi_r(x) &= 1 \text{ if } r < d(x,U). \\ &= 0 \text{ if } r \geqslant d(x,U). \end{aligned} \tag{26}$$

The proof of this fact is elementary. It can be found in [21]. Now let us describe the second method, which comes from [19]. We assume again that d is regular and that U is the background of ϕ. We define the functions $\psi_m : G \to \{0,1,2,\ldots\}$ as follows :

For any pel x,

$$\psi_0(x) = \phi(x).$$

$$\psi_{m+1}(x) = \phi(x) + \min\{\psi_m(y) \mid y \in S_d(x)\} \qquad (27)$$

$$\text{for } m=0,1,2,\ldots$$

Then for any pel x we have :

$$\psi_r(x) = r \text{ if } r < d(x,U)$$

$$= d(x,U) \text{ if } r \geqslant d(x,U). \qquad (28)$$

In other words, $S_d^r(U)$ is the set of pels x such that $\psi_r(x) = \psi_{r+1}(x)$.

Again the proof is elementary.

In Appendix 1, we examine a sequential algorithm for the determination of the set of all d(x,U), $x \in G$, when d is regular.

To end this section, we must stress the importance of regular distances, which are associated to adjacency relations on the grid. As we restricted ourselves to the 4- and 8-adjacency relations, we will from now on restrict ourselves to the corresponding distances, in other words d_4 and d_8.

§ IV. Connected components

A. Introduction

Given a subset S of G and $x,y \in S$, then we say that y is k-connected to x in S if there is a k-path from y to x consisting entirely of pels of S. Clearly, it is an equivalence relation (i.e. it is reflexive, symmetrical and transitive).

Now this equivalence partitions S in non-void and pairwise disjoint subsets $S_1,\dots,S_c$ $(c \geq 1)$, such that for any $x,y \in S$, x and y are k-connected if and only if they belong to the same S_i $(i=1,\dots,c)$. These equivalence classes are called the k-connected components of S.

We say that S is k-connected if S has only one k-connected component.

Note that "4-connected" implies "8-connected" and that an 8-connected component of S is a union of 4-connected components of S.

Following [16], we will say that a figure F is simply k-connected if F is k-connected and B is k'-connected. If F is k-connected and B has m k'-connected components (m=1,2,...) then we will say that F is m-ply k-connected or that m is the order of k-connectivity of F.

B. The detection of connected components

In this subsection, we will give two algorithms for the detection of the k-connected components of a figure. Let us introduce the first one.

Suppose that the figure F is defined in the following way : For any row i, the elements of F on that line form runs which are separated by runs in B. A run is an interval of the form

$$[r, r+s] = \{r, r+1, \dots, r+s\}, \qquad (29)$$

where $s \geq 0$. The length of the run $[r,r+s]$ is $s+1$. The run $[r,r+s]$ on row i consists of the pels $(i,r),\ldots,(i,r+s)$.

It is clear that every run is included in a k-connected component of F. Thus the k-connected components of F are unions of runs. Now 2 runs $[a,a']$ and $[b,b']$ on two successive rows are 4-connected if and only if :

$$a \leq b' \text{ and } b \leq a'. \tag{30}$$

They are 8-connected if and only if :

$$a \leq b'+1 \text{ and } b \leq a'+1. \tag{31}$$

Now the procedure is the following. We label the runs $R_1,\ldots,R_t$ in their order of occurrence when the rows are scanned successively from left to right and from top to bottom. We consider the k-connected components of $R_1 \cup \ldots \cup R_n$ for $n=1,\ldots,t$ by induction. The k-connected component of R_m in $R_1 \cup \ldots \cup R_n$ $(1 \leq m \leq n \leq t)$ has label $C_n(m)$. The algorithm, expressed in pidgin ALGOL, is the following :

ALGORITHM 1

```
begin
for n ← 1 until t do
      begin
      if n > 1 then W_n ← {m ∈ {1,...,n-1}|R_m is k-connected to R_n}
                    V_n ← {C_{n-1}(m)|m ∈ W_n};
      for m ← 1 until n do
          if m = n or (n > 1 and) C_{n-1}(m) ∈ V_n
          then C_n(m) ← n else C_n(m) ← C_{n-1}(m)
      end;
Y ← {C_t(m)|m=1,...,t};
```

<u>procedure</u> ORDER(X) :

<u>for</u> u ← 1 <u>until</u> $|X|$ <u>do</u> $x_u \leftarrow \min(X \setminus \{x_i | i < u\})$

<u>return</u> $f : X \rightarrow \{1,\ldots,|X|\} : x_u \rightarrow u$;

g ← ORDER(Y);

<u>for</u> v ← 1 <u>until</u> t <u>do</u> $C(v) \leftarrow g(C_t(v))$;

<u>comment</u> The run R_v belongs to the C(m)-th k-connected component of F

<u>end</u>

This algorithm is presented in a loose way in [15,19], where it is called "tracking".

We will now present another algorithm for the determination of the k-connected components of a figure. In [15,19], it is called "propagation". It consists in starting with a pel, and constructing a chain of k-neighbours in F until no additional pel is obtained, which means that a k-connected component is found. In the following algorithm, the connected components are labelled C_n $(n \geqslant 1)$:

<u>ALGORITHM 2</u>

<u>procedure</u> PROP(x) :

C ← {x}

D ← {x}

<u>repeat</u> $C \leftarrow N_k(C) \setminus D$

$D \leftarrow D \cup C$ <u>until</u> $C = \emptyset$

<u>return</u> D

<u>begin</u>

$E \leftarrow \emptyset$;

1 ← n;

<u>comment</u> E is the set of pels which have been considered in the algorithm, and n is the number labelling the connected component which is considered;

<u>repeat</u> $x \leftarrow$ <u>any from</u> $F \setminus E$

$C_n \leftarrow PROP(x)$

$E \leftarrow E \cup C_n$

$n \leftarrow n+1$ <u>until</u> $E = F$;

<u>comment</u> The sets C_n are the k-connected components of F

<u>end</u>

One question arises : Which one of the two algorithms is the fastest and the easiest to implement ? Let us first remark that some operations can be made sequentially or in parallel; in Algorithm 1 it is the determination of W_n and the computing of $C_n(m)$; in Algorithm 2, it is the computing of $N_k(C)$ and then $N_k(C) \setminus D$ in the procedure at the beginning. It seems to us that the parallel implementation is easier for Algorithm 1, but then Algorithm 2 has a higher speed when the connected components of F have many holes. In the sequential implementation, Algorithm 2 needs at least as many steps as the size of F, which is generally much slower than Algorithm 1, especially when the connected components of F have few holes.

Thus it seems that Algorithm 1 is better, especially for figures with few holes.

C. <u>The detection of connected components of bounded size</u>

Let us now consider a related problem. How can one detect "small" connected components of a figure ? For example in an engineering drawing, small connected components represent alphanumerics, while larger ones represent gates and interconnections.

Now a connected component is "small" if it can be enclosed in an $U \times V$ rectangle for some "small" U and V.

One way to solve the problem is to use one of the two preceding algorithms, and to select the connected components which can be embedded in an U X V rectangle.

We will propose here three algorithms which solve directly the problem. Let us consider the first method.

We divide the grid into $Y \times Z$ rectangles where :

$$Y \geqslant 2U + 1$$

$$\text{and} \qquad Z \geqslant 2V + 1. \qquad (32)$$

These rectangles can be labelled (I,J), where I=0,...,M/Y-1 and J=0,...,N/Z-1. If Y does not divide M or Z does not divide N, then we extend the grid in order to correct that.The rectangle (I,J) contains the pels (i,j) such that :

$$YI \leqslant i \leqslant YI+I-1$$

$$ZJ \leqslant j \leqslant ZJ+J-1 \qquad (33)$$

This construction is illustrated in Figure 1-13.

On each rectangle (I,J), we look at the k-connected components of the restriction of F to that rectangle. For any set W of pels, we define :

$$i_0(W) = \min\{i \mid (i,j) \in W \text{ for some } j\}.$$

$$i_1(W) = \max\{i \mid (i,j) \in W \text{ for some } j\}.$$

$$j_0(W) = \min\{j \mid (i,j) \in W \text{ for some } i\}.$$

$$j_1(W) = \max\{j \mid (i,j) \in W \text{ for some } i\}. \qquad (34)$$

Let X be a k-connected component of the restriction of F to (I,J). Then X can be embedded in a U × V-rectangle if and only if :

$$i_1(X) - i_0(X) \leqslant U-1$$
$$\text{and} \qquad j_1(X) - j_0(X) \leqslant V-1 \qquad (35)$$

We label the edges of (I,J) e_{j0}, e_{j1}, e_{i0} and e_{i1}, and the corners of (I,J) c_{00}, c_{01}, c_{10} and c_{11}, as shown in Figure 1-14.

Suppose that X satisfies (35). Let e(X) be the set of edges of (I,J) which are not contained in the edge of the grid and which are touched by X. Then one of the following holds :

(i) $e(X) = \emptyset$

(ii) $e(X) = \{e_{ra}\}$, where r is i or j and a = 0,1

(iii) $e(X) = \{e_{ia}, e_{jb}\}$, where $a,b \in \{0,1\}$.

If (i) holds, then X is a k-connected component of F and so we select it.

If (ii) or (iii) holds, then let R be the Y × Z rectangle such that :

$$i_0(R) = \left\lfloor \tfrac{1}{2}(i_0(X)+i_1(X)-Y+1) \right\rfloor$$
$$\text{and} \qquad j_0(R) = \left\lfloor \tfrac{1}{2}(j_0(X)+j_1(X)-Z+1) \right\rfloor . \qquad (36)$$

It follows that for any pel (i,j) in X,

$$i_0(R) \leqslant i_0(X)-U$$
$$i_1(R) \geqslant i_1(X)+U$$
$$j_0(R) \leqslant j_0(X)-V$$
$$\text{and} \qquad j_1(R) \geqslant j_0(X)+V \qquad (37)$$

because of (32). Thus $X \subseteq R$.

If R goes beyond the frame of the grid, then we clip it and get another rectangle R'. Consider the connected components of the restriction of F to R'. Let X' be the one that contains X. Then one of the following holds :

(a) X' cannot be embedded in a U × V rectangle, and so we reject X and X'

(b) X' can be embedded in a U × V rectangle and by (37) X' does not touch the edges of R (and it can touch an edge of R' only if that edge is contained in the edge of the grid). Then X' is a k-connected component of F and so we select it.

Note that in the determination of the rectangle R, we can choose instead of (36) a more general formula :

$$i_0(R) = \lfloor i - \tfrac{1}{2}(Y-1) \rfloor$$

$$\text{and} \qquad j_0(R) = \lfloor j - \tfrac{1}{2}(Z-1 \rfloor , \qquad (38)$$

where i and j are any two integers satisfying :

$$i_0(X) \leqslant i \leqslant i_1(X).$$

$$j_0(X) \leqslant j \leqslant j_1(X). \qquad (39)$$

Now our method can be applied on all rectangles (I,J) sequentially or in parallel. One of the drawbacks of the parallel algorithm is that a suitable k-connected component can be found several times. For example in Figure 1-13, X_1 is found two times (for (I,J) = (1,1) and (1,2)), while X_2 is found four times (for (I,J) = (2,1),(2,2),(3,1) and (3,2)).

Let us describe a sequential algorithm. The rectangles (I,J) are examined in the lexicographical order (i.e. from left to right and from top to bottom). Given a k-connected component X of the restriction of F to (I,J) such that X satisfies (35), one of the following holds :

(i) $e(X) = \emptyset$

(ii) $e(X) = \{e_{r0}\}$, where r is i or j

(iii) $e(X) = \{e_{i0}, e_{j0}\}$

(iv) None of the preceding holds.

If (i) holds, then X is a k-connected component of F. If (ii) or (iii) holds, then either X is a k-connected component of F or some pel (i,j) of X is adjacent to a pel in (I,J-1), (I-1,J) or (I-1,J-1). But in this case (i,j) belongs to a rectangle R for (I,J-1), (I-1,J) or (I-1,J-1), and so we know that either (i,j) belongs to a rejected set or that X is contained in a selected component.

We give below the corresponding algorithm. For any set $X \subseteq G$ we write R(X) for the rectangle defined by (38), and we set :

$$R_{U,V}(X) = 1 \quad \text{if (35) holds.}$$
$$= 0 \quad \text{otherwise.} \qquad (40)$$

We write also C(X) for the set of k-connected components of the restriction of F to X. Moreover, if $Y \subseteq X \cap F$ and $Y \neq \emptyset$, then we write D(Y,X) for the k-connected component of the restriction of F to X which contains Y; D(Y,X) can be found by the procedure PROP(Y) of Algorithm 2 restricted to X. Finally we will write R for the union of rejected components and S for the set of selected components. The notation $\cup S$ means the union of all sets which belong to S. The rectangles (I,J) are ordered in the lexicographical order, and we can write $(I,J)\phi = (I',J')$, where (I',J') is the follower of (I,J), in other words where $I'\frac{N}{Z}+J' = I\frac{N}{Z}+J+1$.

ALGORITHM 3

```
begin
S ← ∅;
R ← ∅;
for (I,J) ← (0,0) step ϕ until (M/Y -1, N/Z -1) do
    begin
    n ← |C ((I,J))|;
    {X_1,...,X_n} ← C((I,J));
    for 1 ← s until n do
        begin
        if R_{U,V}(X_s)=0 or X_s ∩ R≠∅ then R ← R ∪ X_s else
        if X_s ⊄ ∪S then
        if e(X_s) ⊆ {e_{i0},e_{j0}} then S ← S ∪ {X_s} else
        R_s ← R(X_s);
        X'_s ← D(X_s,R_s);
        if R_{U,V}(X'_s) = 0 then R ← R ∪ X'_s else S ← S ∪ {X'_s}
        end
    end;
comment  S is the set of all k-connected components X of F such that
R_{U,V}(X) = 1
end
```

We will now give another algorithm based on black runs on rows. It is similar to Algorithm 1. A run R = [r,r+s] was defined in (29). We will write :

$$b(R) = r$$

$$e(R) = r+s$$

$$r(R) = \text{the row in which } R \text{ is.} \qquad (41)$$

As in Algorithm 1, we label the runs $R_1,\ldots,R_t$ in their order of occurrence when the rows are scanned successively from left to right and from top to bottom. Here $C_n(m)$ will be the label of the k-connected component of R_m in $R_1 \cup \ldots \cup R_n$ $(1 \leq m \leq n \leq t)$. The numbers $b_n(m)$, $e_n(m)$, $r_n(m)$ and $r'_n(m)$ will be respectively the minimum of the b(R)'s, the maximum of the e(R)'s, the minimum and the maximum of the r(R)'s for the runs R belonging to that component. Note that if one deletes from this algorithm the operations concerning these numbers, then one gets Algorithm 1.

ALGORITHM 4

<u>begin</u>

<u>for</u> $n \leftarrow 1$ <u>until</u> t <u>do</u>

    <u>begin</u>

    <u>if</u> $n > 1$ <u>then</u> $W_n \leftarrow \{m \in \{1,\ldots,n-1\} | R_m$ is k-connected to $R_n\}$

            $V_n \leftarrow \{C_{n-1}(m) | m \in W_n\}$;

    <u>for</u> $m \leftarrow 1$ <u>until</u> n <u>do</u>

    <u>if</u> m=n or (n > 1 and) $C_{n-1}(m) \in V_n$

    <u>then</u> $C_n(m) \leftarrow n$

        $b_n(m) \leftarrow \min\{b(R_n), b_{n-1}(u) | u \in W_n\}$

        $e_n(m) \leftarrow \max\{e(R_n), e_{n-1}(u) | u \in W_n\}$

        $r_n(m) \leftarrow \min\{r(R_n), r_{n-1}(u) | u \in W_n\}$

        $r'_n(m) \leftarrow \max\{r'(R_n), r'_{n-1}(u) | u \in W_n\}$

    <u>else</u> $C_n(m) \leftarrow C_{n-1}(m)$

        $b_n(m) \leftarrow b_{n-1}(m)$

        $e_n(m) \leftarrow e_{n-1}(m)$

        $r_n(m) \leftarrow r_{n-1}(m)$

        $r'_n(m) \leftarrow r'_{n-1}(m)$

    <u>end</u>;

$Y \leftarrow \{C_t(m) \mid m=1,\ldots,t\}$;

procedure ORDER(X) : see Algorithm 1;

$g \leftarrow$ ORDER Y;

for $v \leftarrow 1$ until t do $C(v) \leftarrow g(C_t(v))$;

for $z \leftarrow 1$ until y do

  $m(z) \leftarrow$ any m from $\{1,\ldots,t\}$ such that $C(v) = z$

  $I_0(z) \leftarrow r_t(m(z))$

  $I_1(z) \leftarrow r'_t(m(z))$

  $J_0(z) \leftarrow b_t(m(z))$

  $J_1(z) \leftarrow e_t(m(z))$

  if $I_1(z)-I_0(z) \leqslant U-1$ and $J_1(z)-J_0(z) \leqslant V-1$

   then write z and $\{m \mid C(m) = z\}$;

comment The run R_v belongs to the component $C(m) = X$, and we have $I_0(m) = i_0(X)$, $I_1(m) = i_1(X)$, $J_0(m) = j_0(X)$, $J_1(m) = j_1(X)$ (see (34)), and we write thus the components which satisfy (35)

end

The third algorithm is based on a simple idea. We suppose - but this is not crucial - that $FG \subseteq B$. Let $X \subseteq F$. Then the following two are equivalent :

(i) X is a k-connected component of F and $R_{U,V}(X) = 1$

(ii) There is an $(U+2)\times(V+2)$-rectangle R such that X is a k-connected component of the restriction of F to R and $X \cap FR = \emptyset$ (where FR is the frame of R).

Then we have only to move that rectangle on the grid in order to find the required connected components.

For $v=0,\ldots,(M-U-1)(N-V-1)-1$, write R_v for the $(U+2)\times(V+2)$-rectangle such that $i_0(R_V)(N-V-1)+j_0(R_V) = v$. Write also $E(X)$ for the set of k-connected components of the restriction of F to X which do not intersect the frame of R_V. Then the algorithm is the following :

ALGORITHM 5

```
begin
S ← ∅;
for v ← 0 until (M-U-1)(N-V-1)-1 do
    begin
    n ← |E(R_v)|;
    {X_1,...,X_n} ← E(R_v);
    for 1 ← s until n do
    if X_s ∉ S then S ← S ∪ {X_n}
    end;
comment   S is the set of all k-connected components X of F such that
R_U,V(X) = 1
end
```

Of the three algorithms, it seems that Algorithm 5 is the fastest. We have no indication as to whether Algorithm 3 is faster than Algorithm 4 or not.

§ V. Simple closed paths

In this section we show that for certain closed k-paths, called simple closed k-paths, the background has exactly two k'-connected components, provided that the frame assumption holds. These two components, called the inside and the outside, are characterized by several properties which we study here.

A sequence $P = (x_0,\dots,x_{n-1})$ of pels is a simple closed k-path of length n if the following three conditions hold for any $r,s \in \{0,\dots,n-1\}$:

(i) $x_r = x_s$ if and only $r = s$ (42)

(ii) $x_r \in N_k(x_s)$ if and only if $r \equiv s \pm 1 \pmod{n}$ (43)

(iii) If $k = 4$, then $n \geqslant 5$; if $k = 8$, then $n \geqslant 4$. (44)

For k = 4, our definition is the same as the one in [16] (for k=8, no explicit version of (iii) is defined in that paper).

Condition (i) means that the path never crosses itself. Condition (ii) means that two portions of the path may not touch each other. Condition (iii) garantees the existence of a hole inside the path; it rules out the following 3 parts :

a) The segment $x_0\ x_1$

b) The triangle $x_0\ x_1$ (k=8)

x_2

c) The square $x_0\ x_1$ (k=4)

$x_3\ x_2$

Before going further, let us write $\oplus$ and $\ominus$ for the addition and substraction modulo n.

Let B be the background of the path $P = (x_0,\dots,x_{n-1})$.

Given (i) and (ii), condition (iii) is equivalent to the following :

(iv) For any $r=0,\dots,n-1$, $S_8(x_r) \cap B$ has two k'-connected components. (45)

Indeed, it is easily checked that condition (iii) implies that $S_8(x_r)$ is (up to a symmetry of the square) one of the diagrams of Figure 1-15. Then it is easily checked that each of these diagrams satisfies (iv).

Now it is obvious that in the 3 ruled out paths (the segment, triangle and square) x_0 does not satisfy (iii). Therefore (iii) and (iv) are equivalent.

Now we will give some properties of closed paths in general.

Let $P = (x_0,\ldots,x_{n-1})$ be a closed k-path (in other words a k-path with $x_0 \in N_k(x_{n-1})$). For any pel y in the background of P, consider the world half-lines $L_v(y)$ (v=0,...,n-1) originating from the center of y and passing through the center of x_v, and the (oriented) angles $\alpha_v(y)$ between $L_v(y)$ and $L_{v\oplus 1}(y)$ (see Figure 1-16). Then define :

$$\alpha_P(y) = \frac{1}{2\pi} \sum_{v=0}^{n-1} \alpha_v(y). \tag{46}$$

Clearly $\alpha_P(y)$ is the number of times that P turns around y and it is therefore an integer. For example in Figure 1-16, $\alpha_P(y) = -1$.

Note that if we consider the inverse path $P' = (x'_0,\ldots,x'_{n-1})$, where $x'_i = x_{n-1-i}$ (i=0,1,...,n-1), then we have

$$\alpha_{P'}(y) = -\alpha_P(y) \tag{47}$$

On the other hand, if we take the path $P^t = (x_t, x_{t\oplus 1},\ldots,x_{t\oplus(n-1)})$, where t = 0,1,...,n-1, then

$$\alpha_{P^t}(y) = \alpha_P(y) \tag{48}$$

The following result gives a comparison between the different values of $\alpha_P(y)$, where $y \in B$:

Proposition 1. Let $y_0 = (i_0, j_0)$ and $y_1 = (i_1, j_1)$ be distinct pels in the background B of a closed path P. Let R be the rectangle spanned by (i'_0, j'_0), (i'_0, j'_1), (i'_1, j'_0) and (i'_1, j'_1), where for v=0,1, we have :

$$\begin{aligned} i'_v &= i_v \text{ if } i_0 \neq i_1 \\ &= i_v + (-1)^v \text{ if } i_0 = i_1. \\ j'_v &= j_v \text{ if } j_0 \neq j_1 \\ &= j_v + (-1)^v \text{ if } j_0 = j_1. \end{aligned}$$

(This is illustrated in Figure 1-17). Let $R_1 = \{(i,j) \in R \mid (i-i_0)(j_1-j_0) > (j-j_0)(i_1-i_0)\}$ and $R_0 = R \setminus R_1$ (R_1 consists of the pels of R whose centers lie strictly on the right side of the oriented line directed from the center of y_0 to the center of y_1) (see Figure 1-18). Then for any closed path $P = (x_0, \ldots, x_{n-1})$ we have :

$$\alpha_P(y_0) - \alpha_P(y_1) = |\{s \in \{0,\ldots,n-1\} \mid x_s \in R_1 \text{ and } x_{s\oplus 1} \in R_0\}| - |\{t \in \{0,\ldots,n-1\} \mid x_t \in R_0 \text{ and } x_{t\oplus 1} \in R_1\}| \tag{49}$$

The proofs of this proposition and of the other results of this section can be found in Appendix 2.

This result has immediate consequences :

Corollary 2. Let P be a k-path. Then the following holds for any $y, y' \in B$:

(i) If $y' \in N_4(y)$, then $\alpha_P(y) = \alpha_P(y')$

(ii) If $y' \in N_8(y)$, let y and y' belong (up to a rotation) to a 2×2 square $\begin{matrix} z' & y' \\ y & z \end{matrix}$. Then $\alpha_P(y) = \alpha_P(y')$, except if $\{z, z'\} \subseteq P$; then :

$$\alpha_P(y)-\alpha_P(y') = |\{u \in \{0,\ldots,n-1\} | x_u=z \text{ and } x_{u\oplus 1}=z'\}|$$

$$- |\{v \in \{0,\ldots,n-1\} | x_v=z' \text{ and } x_{v\oplus 1}=z\}|$$

(iii) If y and y' are k'-connected in B, then $\alpha_P(y) = \alpha_P(y')$.

It follows thus that the function α_P can be useful to study the connected components of the background of a closed k-path. We apply it to simple closed paths. In [16] it is proved that the background of a simple closed k-path can be divided into two nonempty parts, called the inside and the outside, such that any k'-path from a pel of the inside to a pel of the outside must intersect P. The author makes the frame assumption (that $FG \subseteq B$); we will make it.

The inside and outside of P are defined as follows :

If x = (i,j) is a pel in the background B, let H(x) be the set of pels of the form (i+u,j), where u=0,...,M-1-i. Then the intersection of H(x) with P is the union of mutually E-disconnected runs of the form [i+u+1, i+u+v] (v=1,...,M-1-u-i). The elements of such a run are successive pels of P, say $x_{r\oplus 1},\ldots,x_{r\oplus v}$. Now x_r and $x_{r\oplus v\oplus 1}$ must be pels of the form $(i+u+1-\delta, j+\varepsilon)$ and $(i+u+v+\delta', j+\varepsilon')$, where $\delta, \delta' \in \{0,1\}$ and $\varepsilon, \varepsilon' \in \{+1,-1\}$. If $\varepsilon = \varepsilon'$, then we say that H(x) touches P in that run. Otherwise we say that H(x) crosses P in that run. These two situations are illustrated in Figure 1-19. Now we say that x is inside P if H(x) crosses P an odd number of times and that x is outside P if H(x) crosses P an even number of times.

Write I(P) for the inside of P (i.e. the set of pels inside P) and O(P) for the outside of P (i.e. the set of pels outside P).

We have obtained the following result with the use of Proposition 1 and Corollary 2 :

Theorem 3. Let $P = (x_0, \ldots, x_{n-1})$ be a simple closed k-path. Let B be the background of P, I(P) the inside of P and O(P) the outside of P. Then we have the following :

(i) $I(P) \neq \emptyset \neq O(P)$, I(P) and O(P) are the two k'-connected components of B.

(ii) $FG \subseteq O(P)$

(iii) If $Q = \{x_r, \ldots, x_{r \oplus s}\}$ $(0 \leqslant s \leqslant n-1)$, then $S_8(Q) \cap B$ has two (nonempty) k'-connected components, namely $I(P) \cap S_8(Q)$ and $O(P) \cap S_8(Q)$.

(iv) For any pel x_r of P, for any pels $y, z \in S_8(x_r)/P$ such that y and z are symmetrical with respect to x_r, then one of them belongs to I(P) and the other to O(P).

(v) For any $y \in B$,

$$\alpha(y) = 0 \quad \text{if} \quad y \in O(P),$$
$$= \varepsilon \quad \text{if} \quad y \in I(P),$$

where $\varepsilon = 1$ if P leaves O(P) on its left, and $\varepsilon = -1$ if P leaves O(P) on its right.

Some parts of this result are also proved in [17].

It follows from this theorem that in the definition of the inside and the outside, we can choose for a pel $x = (i,j)$ the set H(x) to be the set of pels of the form $(i-u,j)$, $(i,j+u)$ or $(i,j-u)$ $(u \geqslant 0)$, without altering that definition.

Note that if we do not make the frame assumption, then O(P) may be k'-disconnected.

Note that for a k-connected figure F, we can define O(F) as the k'-connected component of B containing FG (we make of course the frame assumption), and I(F) as $B \setminus O(F)$. But then I(F) can be empty or k'-disconnected.

We now end this section with a sufficient condition for a closed k-path to contain a simple closed k-path.

<u>Proposition 4</u>. Let $P = (x_0,\ldots,x_{n-1})$ be a closed k-path. Suppose that $x_1 \notin S_k\ (x_{n-1})$, $N_k(x_0) \cap P = \{x_1,x_{n-1}\}$ and that for k=4 $\{x_0\} = N_4(x_1) \cap N_4(x_{n-1})$. Then P contains a simple closed path P' such that x_{n-1}, x_0 and x_1 are successive pels in P'. Moreover, $S_8(x_0) \cap B$ has two k'-connected components, one contained in I(P'), and the other in O(P').

§ VI. Edge and border - The edge-following algorithm

Given a figure F with background B, we will define edges and borders between neighbouring connected components of F and of B.

We will show that unless we use 4-connectedness for both F and B, the edge between a connected component X of F and a neighbouring connected component Y of B forms a single cycle. We give a procedure for the following of that edge.

We say that two sets X and Y are k-neighbouring if $d_k(X,Y) = 1$. Now if X is a union of 4-connected components of F (this is the case when X is a k-connected component) and Y is likewise a union 4-connected components of B, then $d_8(X,Y) = 1$ if and only if $d_4(X,Y) = 1$. Indeed, the second implies the first, while if the first holds, then for $x \in X$ and $y \in Y$ such that $d_8(x,y)=1$, then either $d_4(x,y) = 1$ or there is some $z \in N_4(x) \cap N_4(y)$; if $z \in F$, then $z \in X$ and so $d_4(X,Y) = d_4(z,y) = 1$, while if $z \in B$, then $z \in Y$ and so $d_4(X,Y) = d_4(x,z) = 1$. In this case, we simply say that X and Y are neighbouring.

Let $X \subseteq F$ and $Y \subseteq B$ be neighbouring sets. We make the following definitions :

The k-border of X to Y is the set :

$$\delta_k(X,Y) = X \cap N_k(Y) \tag{50}$$

Similarly we define the k-border of Y to X : $\delta_k(Y,X)$.

The k-border of X is the k-border of X to B, and we write it $\delta_k(X)$. The k-border of Y, written $\delta_k(Y)$, is the k-border of Y to F.

For a figure F, the borders $\delta_4(F)$, $\delta_8(F)$, $\delta_4(B)$ and $\delta_8(B)$ are shown in Figure 1-20.

Now the edge between X and Y can be defined in two ways, as a world object or as a grid object.

As a world object, the edge $\varepsilon(X,Y)$ between X and Y is the line separating the world surface represented by X and the world surface represented by Y. It is illustrated in Figure 1-21. We have :

$$\varepsilon(X,Y) = \varepsilon(Y,X) \tag{51}$$

and

$$\varepsilon(F) = \varepsilon(B). \tag{52}$$

Now this edge can be oriented. In $\varepsilon^{+}(X,Y)$, it is oriented with X on the left and in $\varepsilon^{-}(X,Y)$, it is oriented with X on the right (see Figure 1-21). Then we have :

$$\varepsilon^{+}(X,Y) = \varepsilon^{-}(Y,X)$$

and

$$\varepsilon^{-}(X,Y) = \varepsilon^{+}(Y,X), \tag{53}$$

$$\varepsilon^{+}(F) = \varepsilon^{-}(B)$$

and

$$\varepsilon^{-}(F) = \varepsilon^{+}(B). \tag{54}$$

Now we can define the edge as a grid object. We use the same notation as for the preceding definition :

$$\varepsilon(X,Y) = \{\{x,y\} \mid x \in X,\ y \in Y,\ d_4(x,y) = 1\} \tag{55}$$

$$\varepsilon^{+}(X,Y) = \{(x,y) \mid x \in X,\ y \in Y,\ d_4(x,y) = 1\} \tag{56}$$

$$\varepsilon^{-}(X,Y) = \{(y,x) \mid x \in X,\ y \in Y,\ d_4(x,y) = 1\} \tag{57}$$

Again properties (51), (52), (53) and (54) hold. In fact, there is a correspondance between this definition and the preceding one, because the pair $\{x,y\}$ corresponds to the edge between the pels x and y, while the ordered pair (x,y) corresponds to that edge oriented with x on the left (see Figure 1-22).

In [16], an edge of a figure F is defined as an ordered pair (x,y), where $x \in F$ and $y \in B$, in other words as an element of $\varepsilon^{+}(X,Y)$. We will call an <u>edge element</u>, a <u>positive edge element</u> and a <u>negative edge element</u> between X and Y an element of $\varepsilon(X,Y)$, $\varepsilon^{+}(X,Y)$ and $\varepsilon^{-}(X,Y)$ respectively.

It can be seen from Figure 1-21 that edges between connected components of F and connected components of B form in general single cycles. In fact, we will give an algorithm for finding cycles of edges, and we will show that under certain conditions edges form a single cycle.

We assume that we have a figure F such that either $FG \subseteq F$ or $FG \subseteq B$.

We first deal with edges between 4-connected components of F and neighbouring 8-connected components of B. Following [16], we define the mapping :

$$E_F^{+} : \varepsilon^{+}(F) \to \varepsilon^{+}(F) : (x,y) \to (x,y)\, E_F^{+} = (x^{+},y^{+}) \qquad (58)$$

where (x^{+},y^{+}) is defined as follows :

The pels x and y belong (up to a rotation) to a square $\begin{matrix} a & b \\ x & y \end{matrix}$ in G; then x^{+} and y^{+} are given in the table below as a function of the value (0 or 1) assigned to the pels a and b :

<u>TABLE 2</u>

a	b	x^{+}	y^{+}
0	*	x	a
1	0	a	b
1	1	b	y

(59)

This mapping is illustrated in Figure 1-23. Let us make a few remarks :
(a) It is well-defined; in other words, for any $(x,y) \in \varepsilon^{+}(F)$, a and b exist, because we may not have $\{x,y\} \subseteq FG$ (as we assumed that $FG \subseteq B$ or $FG \subseteq F$).

(b) It is one to one. In Figure 1-24, we indicate in function of the values taken by two pels c and d forming with x and y a square $\begin{smallmatrix} x & y \\ c & d \end{smallmatrix}$, the only possible choice for $(x,y)\ (E_F^+)^{-1} = (x^-, y^-)$.

(c) (x^+, y^+) is the immediate successor of (x,y) in $\varepsilon^+(F)$. Hence by (b) E_F^+ induces cycles on $\varepsilon^+(F)$.

(d) x and x^+ belong to the same 4-connected component of F, while y and y^+ belong to the same 8-connected component of B. This can easily be seen from Figure 1-23. Hence if X is a 4-connected component of F and Y is a neighbouring 8-connected component of B, then E_F^+ fixes the set $\varepsilon^+(X,Y)$, and by (b) it induces cycles on it. Hence $\varepsilon^+(X,Y)$ is a union of cycles.

Now we define a similar mapping for $\varepsilon^{-1}(F)$:

$$E_F^- = \varepsilon^-(F) \to \varepsilon^-(F) \ : \ (y,x) \to (y^-, x^-),$$

where $$(x^-, y^-) = (x,y)(E_F^+)^{-1} \qquad (60)$$

The pels y and x belong (up to a rotation) to a square $\begin{smallmatrix} d & c \\ y & x \end{smallmatrix}$. Then y^- and x^- are given in the table below as a function assigned to the pels c and d :

TABLE 3

d	c	y^-	x^-
*	0	c	x
0	1	d	c
1	1	y	d

(61)

This mapping is illustrated in Figure 1-25. We see that it is a mirror-symmetric image of $\varepsilon^+(F)$.

Lastly, we can define the operators E_B^+ and E_B^-, which are similar to ε_F^+ and ε_F^-, but where the roles of F and B are interchanged. They are the following :

$$E_B^+ : \varepsilon^+(B) \to \varepsilon^+(B) : (y,x) \to (y^+,x^+), \tag{62}$$

where (y^+,x^+) is defined as follows :

The pels x and y belong (up to a rotation) to a square $\begin{matrix} u & v \\ y & x \end{matrix}$ in G; then x^+ and y^+ are given in the following table as a function of the value assigned to the pels a and b :

TABLE 4

u	v	y^+	x^+
1	*	y	u
0	1	u	v
0	0	v	x

(63)

This mapping is illustrated in Figure 1-26. Similarly, we define E_B^- as follows

$$E_B^- : \varepsilon^-(B) \to \varepsilon^-(B) : (x,y) \to (x^-,y^-),$$

where $$(y^-,x^-) = (y,x)(E_B^+)^{-1} \tag{64}$$

If the pels x and y belong (up to a rotation) to the square $\begin{matrix} t & s \\ x & y \end{matrix}$, then x^- and y^- are given by the following table :

TABLE 5

t	s	x^-	y^-
*	1	s	y
1	0	t	s
0	0	x	t

(65)

This mapping is illustrated in Figure 1-27. It can also be found in [16].

The remarks (a), (b), (c) and (d) made about E_F^+ hold also for E_F^-, E_B^+ and E_B^-, but in the two latter ones, 4 and 8 must be interchanged. Concretely, the cycles induced by the 4 operators are as follows :

Write X_k for a k-connected component of F and $Y_{k'}$ for a neighbouring k'-connected component of B. Then :

- E_F^+ induces cycles on $\varepsilon^+(X_4^4,Y_8) = \varepsilon^-(Y_8,X_4)$.
- E_F^- induces cycles on $\varepsilon^-(X_4,Y_8) = \varepsilon^+(Y_8,X_4)$.
- E_B^+ induces cycles on $\varepsilon^-(X_8,Y_4) = \varepsilon^+(Y_4,X_8)$.
- E_B^- induces cycles on $\varepsilon^+(X_8,Y_4) = \varepsilon^-(Y_4,X_8)$. (66)

Note that E_F^+, E_F^-, E_B^+ and E_B^- induce also cycles on $\varepsilon(F)$, on non oriented edges.

What we have to show is that $\varepsilon(X_k,Y_{k'})$ has in fact only one such cycle.

We still assume that either $FG \subseteq B$ or $FG \subseteq F$. We call it the <u>restricted frame assumption</u> (RFA).

The following result (Theorem 5) is extremely important. In [16] it is proven for simply 4-connected figures satisfying the frame assumption; it is also argued that the result is true for any figure, but the argument is false, since it does not respect the frame assumption.

<u>Theorem 5</u>. (Suppose that the figure F satisfies the RFA). Let X_4 be a 4-connected component of F and let Y_8 be a neighbouring 8-connected component of B. Let X_8 be an 8-connected component of F and let Y_4 be a neighbouring 4-connected component of B. Then $\varepsilon^+(X_4,Y_8)$ (or $\varepsilon^-(X_4,Y_8)$ or $\varepsilon(X_4,Y_8)$) and $\varepsilon^+(X_8,Y_4)$ (or $\varepsilon^-(X_8,Y_4)$ or $\varepsilon(X_8,Y_4)$) contain only one cycle each.

The proof of this theorem and of the other results of that section can be found in Appendix 3.

From this theorem we deduce the <u>edge-following algorithm</u> : suppose that we search $\varepsilon^{\pm}(X_k, Y_{k'})$. From (66) we choose the appropriate mapping E among E_F^+, E_F^-, E_B^+ and E_B^-. We take two 4-adjacent pels $x \in X_k$ and $y \in Y_{k'}$, forming the edge element ε_0. For i=0,1,2,..., we compute $\varepsilon_{i+1} = (\varepsilon_i)E$ until we get $\varepsilon_n = \varepsilon_0$. Then the set $\{\varepsilon_0, \ldots, \varepsilon_{n-1}\}$ is the required edge.

Now what can one say about $\varepsilon(X_k, Y_k)$, where X_k is a k-connected component of F and Y is a neighbouring k-connected component ?

If k=4, then $\varepsilon(X_k, Y_k)$ does not necessarily form a cycle or a union of cycles, it may also be disconnected (see Figure 1-28).

For k=8 we get the following result.

<u>Proposition 6</u>. (Suppose that F respects the RFA). Let X_8 be an 8-connected component of F and let Y_8 be a neighbouring 8-connected component of B. Then $\varepsilon^+(X_8, Y_8)$ (or $\varepsilon^-(X_8, Y_8)$ or $\varepsilon(X_8, Y_8)$) forms a single cycle.

<u>Note</u>. The result still holds if we relax the hypothesis as follows : X_8 is 8-connected and a union of 4-connected components of F and Y_8 is an 8-connected component of B (or the reverse).

Let us now apply these two results to borders :

<u>Proposition 7</u>. (Suppose that F respects the RFA). Let $k, k_1, k_2 \in \{4,8\}$, with $(k_1, k_2) \neq (4,4)$. If X is a k_1-connected component of F and Y is a neighbouring k_2-connected component of B, then the border $\delta_k(X,Y)$ is a closed 8-path. Moreover, for $(k_1, k_2) = 4,8)$, $\delta_8(X,Y)$ is a closed 4-path.

We end this section with the following :

<u>Proposition 8</u>. (Suppose that F satisfies the RFA). Let X be a k-connected component of F and let $Y_1, \ldots, Y_m$ be the neighbouring k'-connected components of $\bar{X} = G \setminus X$. Then :

(i) B has m k'-connected components $Z_1,\ldots,Z_m$ neighbouring X, where $Z_i \subseteq Y_i \cap B$ $(i=1,\ldots,m)$; Y_i is the k'-connected component of $\bar{X}$ containing Z_i $(i=1,\ldots,m)$.

(ii) For any $i=1,\ldots,m$, $\varepsilon(X,Y_i) = \varepsilon(X,Z_i)$, $\delta_4(Y_i,X) = \delta_4(Z_i,X)$, $\delta_4(X,Y_i) = \delta_4(X,Z_i)$ and $\delta_8(X,Y_i) = \delta_8(X,Z_i)$.