DIGITAL TOPOLOGICAL AND MATRIX STRUCTURED IMAGE PROCESSING *

Shao Juliang Li Deren

Department of Photogrammetry and Remote Sensing
Wuhan Technical University of Surverying and Mapping,
39 Loyu Road, Wuhan 430070, P. R. China

17th ISPRS Congress, Comm. III, Washington D. C.

ABSTRACT

Digital topology deals with the topological properties of digital image and provides a sound mathematical basis for image processing operations such as image thinning, border following and connected component labelling. Matrix structure is also a consistent mathematical framework for image processing. This paper reviews the concepts of these two fields and suggests some image processing operations such as image thinning, border following, region growing and discrete Fourier transform by integrating these two methods. In this integration the digital topology of imagery is considered as constraint condition and the matrix structure of imagery is used as the parallel representation method. This investigation would be valuable for image matching and image understanding.

KEY WORDS: Digital Topology; Matrix Structure; Image Processing; Algorithm;

1. INTRODUCTION

1.1 Digital topology

Digital topology is to study the topological properties of digital image arrays. Its results provide a sound mathematical basis for image processing operations such as image thinning, border following, and region growing. Most people [Kong & Rosenfeld 1989, Arcelli 1979, and Tsao & Fu 1982] paid attention to the properties of the digital topology with two—and three—dimensional binary image arrays, but not with gray—scale image arrays. However, some tasks such as region growing, image understanding and pattern recognition, etc. relate to the digital topology with the gray—scale image arrays. We review therefore some basic concepts about the digital topology, and extend the connectivity of the binary image to that of gray value image.

Let (i, j) be a point of an given image. It then has four horizontal and vertical neighbors described as the following:

$$(i-1,j),(i,j+1),(i,j-1),(i+1,j)$$

Such points are called to be 4 — ADJACENT. Moreover, (i, j) has four diagonal neighbors, i. e.

$$(i-1,j-1),(i-1,j+1),$$
$$(i+1,j-1),(i+1,j+1)$$

These points together with four 4—adjacent points are called to be 8—ADJAECNT.

A PATH is a sequence $(p_i \mid 0 \leqslant i \leqslant n)$, and p_i is adjacent to p_{i+1}. A set of pixels is said to be CONNECTED if there is a path between any two pixels.

Here, we set up a theorem related to the gray—scale image.

Theorem 1: A set of pixels posses connectivity, and is called region, when a pixel is extended to be such a set of pixels according to the following steps:

A. Select a starting point as region $0(R_0)$, and give a

* This paper is the early research to interpret the man—made objects from the aerial photograghs.

threshold t.

B. Solve the average v of region 0.

C. Computer the difference between the gray value of any neighbor in region 0 and v. If the absolute difference is less than t, this point is considered as the domain of region 1.

D. If new region is no longer extended, loop stop. Otherwise, repeat like from step A to Step C.

Suppose p_0 be a starting point, and p_{1i} be one of the new growed pixels. According to the step C, p_{1i} and p_0 form a path. p_{1i} and p_{1j} then establish a path through p_0, i. e. any two points in the region 1 can form a path. Similarly, p_{ki} is a new growed pixel from region k—1 to region k. Suppose region k—1 be connected, a path is formed from p_{ki} to any pixel of region k—1. So, new paths are formed from p_{ki} to p_{kj} through pixels of region k — 1, i. e. any two points in the region k can form a path . According to the definition of connectivity, such a set of pixels posses connectivity and is called to be connected region.

In fact, the above procedure is region growing. We say here that the connectivity of digital topology constrains region growing.

1. 2 Matrix Structure

Matrix is well known by people who are engaged in each different kind of the field. But, using matrix as representation and calculation structure in image processing is just at the first step [Dougherty & Giardina 1987]. A digital image is similar to a matrix or array of numbers. As a result, it is a useful tool for standarizing representation of the pictures and the parallel algorithms. In this paper, we introduce a mathematical structure, the "bound matrix", for the representation of digital images. This structure is used in conjunction with block diagrams to serve as a concise expression of digital image processing operation.

Definition 1. A array—type structure consisting of m by n entities:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}_{r,t}$$

where,

1) each a_{pq} is a real number or a $*$ (star)

2) $1 \leqslant p \leqslant m, 1 \leqslant q \leqslant n$

3) r and t are integers

is called a BOUND MATRIX, or an m x n bound matrix, and the stars denote values that are not known. The location of the a_{11} entry is (r, t). The direction from a_{11} to a_{12} makes the value of q increased, and the direction from a_{11} to a_{21} makes the value of p decreased. This $*$ (star) representation here is helpful to visualize all values outside a bound matrix to be stars. Unusually, the origin is changable and available in the location of any entry.

In many disciplines, such as computer architecture and communications, block diagram is a useful device. Therein, they serve as a language for describing and performing operations with and between images and bound matries. When this bound matrix is used for representing the gray value distribution on the two — dimensional plane, it is called gray — matrix structure. The procedure of above region growing can be denoted by this gray— matrix structure.

2. IMAGE THINNING AND MATRIX REPRESENTATION

2. 1 Image Thinning

Image thinning is a common preprocessing operation in computer aided design, automated cartography, facsimile transmission, and pattern recognition.

Many algorithms of image thinning do not preserve topology. This is not permissive in the fields needing to utilize the topological properties. Therefore, Acrelli proposed thinning algorithms should be obedient to the following criterian for North :

$$W \cup (-S) \cup E \cap (-W) \cup NW \cup (-N)$$
$$\cap (-N) \cup NE \cup (-E) \cap (-E) \cup SE \cup (-S)$$

$\cap$ (- S) $\cup$ SE $\cup$ (- W) = 0 (1)

where, " $\cap$ ", " $\cup$ " and "- " denote boolean disjunction, conjunction, and negation, respectively; w=0 or 1 according as the west neighbor of p is a signal or background, and similarly for the other compass directions. In fact, N≡1, since this rule applies only to north border points. If "east", "south", and "west" substitute for "north", it is applicable to delete points in parallel from each side in turn (e. g. ,in the N,S,E,W).

2. 2 Matrix Representation

Before we deal with the representation method for equation (1), we expand some basic concepts [Dougherty & Giardina 1987] used for equation (1) with their block diagrams. Suppose f and g be the input images .

$$[ADD(f,g,\cdots)](i,j) = \begin{cases} f(i,j)+g(i,j)+\cdots. & \text{if all inputs are defined at}(i,j) \\ *, & \text{elsewhere} \end{cases}$$

Here, we have extended the above concept from two images to infinite number images. This entension will efficiently cut down the computational complexity for more than 2 images. This situation is similar to most of the matrix operations. The block diagram for ADD is

$$\begin{matrix} f \to \\ g \to \\ \to \end{matrix} \boxed{ADD} \to ADD(f,g)$$

$$[MULT(f,g)](i,j) = \begin{cases} f(i,j) \times g(i,j) \times \cdots, & \text{if all inputs are defined at}(i,j) \\ *, & \text{if either input is undefined at}(i,j) \end{cases}$$

The block diagram corresponding to multiplication is

$$\begin{matrix} f \to \\ g \to \\ \to \end{matrix} \boxed{MULT} \to MULT \quad (f,g,\cdots)$$

$$[SUB(f)](i,j) = \quad -f(i,j)$$

the negation of $*$ = $*$. The block diagram for SUB is defined as

$$f \to \boxed{SUB} \to SUB\ (f)$$

$$TRAN(f;i,j) = (a_{pq})_{r+i,t+j}$$

The block diagram for TRAN is given by

$$\begin{matrix} f=(a_{pq})_{rt} \to \\ i \to \\ j \to \end{matrix} \boxed{TRAN} \to TRAN(f,i,j) = (a_{pq})_{r+i,t+j}$$

$$[EXTADD(f,g,\ldots)](i,j) = \begin{cases} f(i,j)+g(i,j)+\cdots, & \text{as long as any image is defined at } (i,j)\text{, add here.} \\ *, & \text{if all inputs are undefined at } (i,j) \end{cases}$$

We can here absorb these concepts into boolean operations.

$$[BN(f)](i,j) = \begin{cases} 1 & \text{if the pixel is undefined at}(i,j) \\ 0 & \text{if the pixel is defined at } (i,j) \end{cases}$$

$$[AND(f,g,\cdots)](i,j) = \begin{cases} 1, & \text{if all inputs are 1 at}(i,j) \\ 0, & \text{elsewhere} \end{cases}$$

$$[OR(f,g,\cdots)](i,j) = \begin{cases} 1, & \text{if at least 1 image is ture at}(i,j) \\ 0, & \text{elsewhere} \end{cases}$$

For equation (1) N can be gotten by translating the origin from (0,0) to (0,+1), and similarly for the other compass directions. (- N) can be obtained from N by negation BN, and similarly for W,E,and S. " $\cap$ "and " $\cup$ " can be denoted by OR and AND, respectively. For a non — isolated and non — end point, it is solved by employing EXTADD after translating for 8 compass directions. Obviously,if the accumulative value of the neighbors whose are 1 is more than 2, the point is non — isolated and non — end. The current point is a signal, as long as the value of this point is 1. Hence, we can use matrix structure to represent the image parallel thinning for the north border algorithm as provided in figure 1.

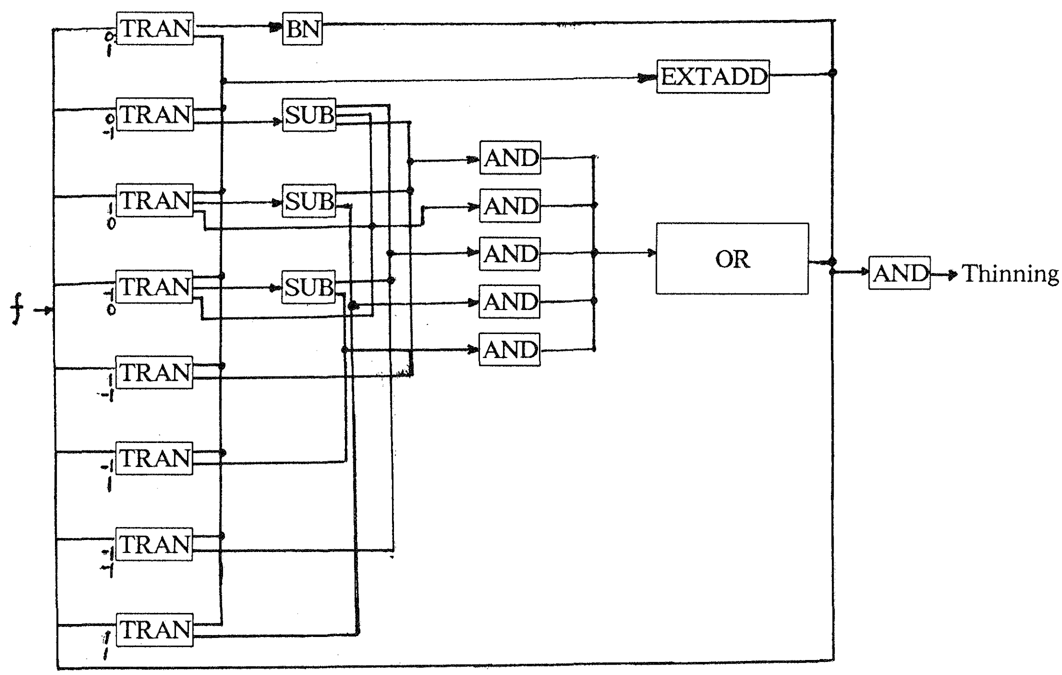


Fig. 1 Block Diagram for Image Thinning

The experimental result for this methodology is showed in figure 2.

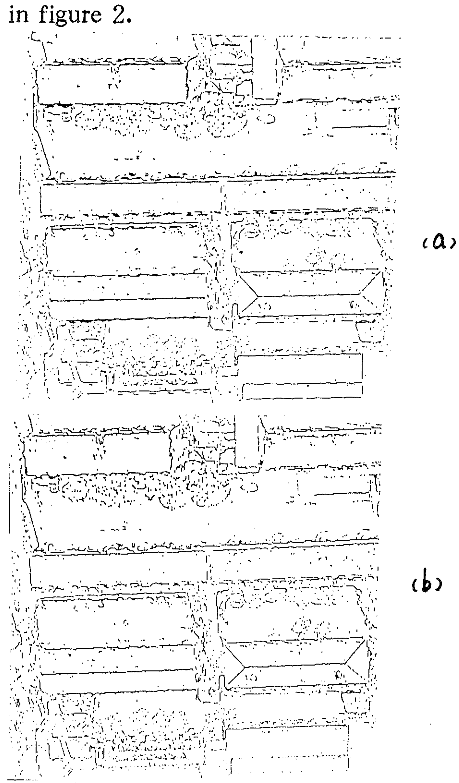


Fig. 2 a). The Original Image.
b). The Thin—line Image after Performing Image Thinning

3. BORDER FOLLOWING AND MAIRIX REPRESENTATION

Border following is one of the underlying techniques in the processing of digitized binary images, and its techniques have also found applications in various other related problems, i. e. pattern recognition, image analysis and image data compression. In general, the borders of a subset S of an image are considered as the set of points adjacent to S in —S which is the complement of S. The traditional approach to find borders is that at first find a starting point , decide the direction of the border in the 8 neighbors , and follow the trend. Moreover, the intersection points must be put in the store, so that those points can be found in next loop. Especially, when the borders in the image posses higher rate to whole image, this method will be clumsy. As a result, we develop a new parallel algorithm:

(1). For a digital image f, translate f for a pixel distance in E, S, W, and N directions, respectively, and then obtain a f_{TRANi} ($i = 1, \ldots, 4$).
(2). For f_{TRANi}, take the logical negation operation, and then get f_{BNi} ($i=1, \ldots, 4$).
(3). Holding the logical disjunction opreation to f_{BNi}, obtain f_{BN}.
(4). Operating f and f_{BN} with logical conjuntion , finally get the borders.

Obviously, this method is very efficient to be performed in parallel array processors. Its diagram is also simple(fig. 3).

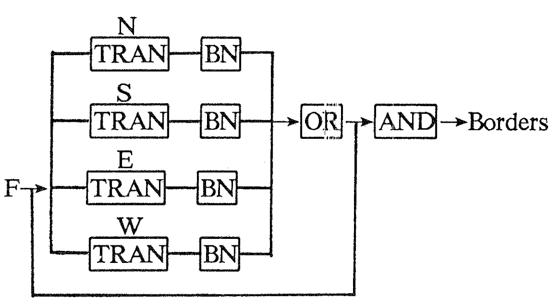
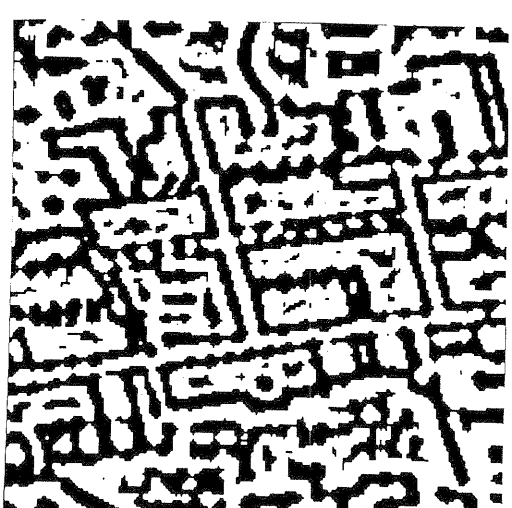
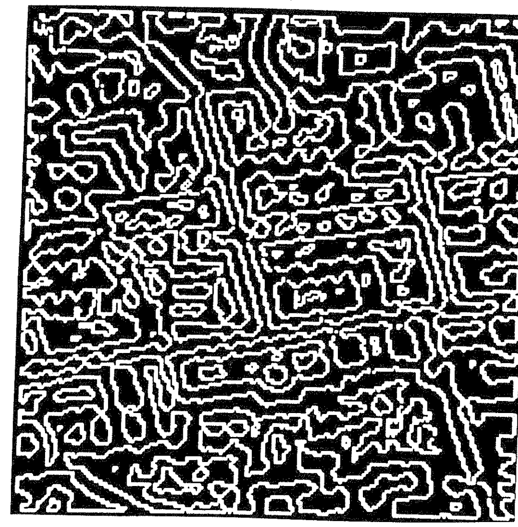


Fig. 3 Block Diagram for Border Following

Fig. 4 is provided as a experimental result for border following.



(a)



(b)

Fig. 4 a). The Original Binary Image.
b). The Border Image Found by Border Following

4. REGION GROWING AND MATRIX REPRESENTATION

A common image processing task is to separate out a particular region of the overall image on the basis of gray value or texture. The image is segmented into a feature region and a background region. The steps about region growing have been described previously. Meanwhile, the proceduce is proved to be connectivity — preserving. Several new operations should be introduced, before presenting the block diagram of GROW.

$$[THRESH(f_i,t)](i,j) = \begin{cases} 1, & \text{if the grav value} \geqslant t \text{ at}(i,j) \\ 0, & \text{otherwise} \end{cases}$$

$$[ABS(f)](i,j) = |f(i,j)|$$

Unless $f(i,j)$ is undefined, in which case $[ABS(f)](i,j) = *$.

$$[DIV(f)](i,j) = \begin{cases} \dfrac{1}{f(i,j)}, & \text{if } f(i,j) \text{ is real and not} \\ *, & \text{elsewhere} \end{cases}$$

[SCALAR(r;f)](i,j) = r × f(i,j)

unless f(i,j) = *, in which case SCALAR(r;f) is also undefined at (i,j).

The implementation of GROW, which is somewhat involved, is presented in Figure 5.

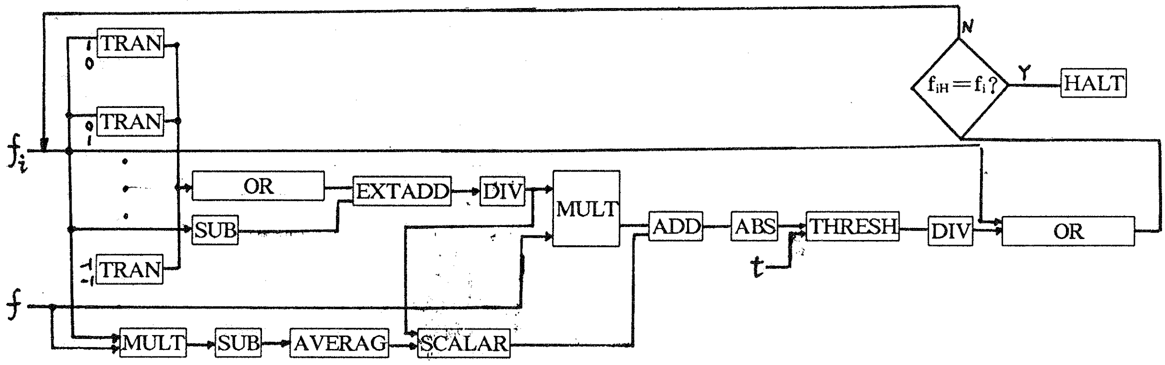


Fig. 5 Block Diagram for Region Growing

In order to test whether the algorithm is feasible, a experiment is performed. As a result, it is imparted in figure 6.

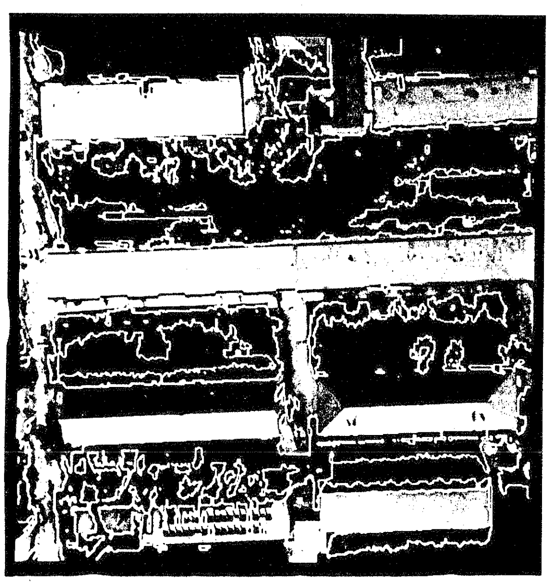


Fig. 6 The Region Outlined by Highlighted Curves Resulted from Region Growing

5. DISCRETE FOURIER TRANSFORM AND MATRIX REPRESENTATION

The discrete Fourier transform is a very useful vehicle in the digital signal processing and the digital image processing, and its algorithm has widely been discussed [Rosenfeld & Kak 1982, Wang 1990]. Give the m by n bound matrix

$$f = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,n-1) \\ f(1,0) & f(1,1) & \cdots & f(1,n-1) \\ \cdots & \cdots & \cdots & \cdots \\ f(m-1,0) & f(m-1,1) & \cdots & f(m-1,n-1) \end{bmatrix}_{U,V}$$

The discrete Fourier transform (DFT) of f is the image

$$F = \begin{bmatrix} F(0,0) & F(0,1) & \cdots & F(0,n-1) \\ F(1,0) & F(1,1) & \cdots & F(1,n-1) \\ \cdots & \cdots & \cdots & \cdots \\ F(m-1,0) & F(m-1,1) & \cdots & F(m-1,n-1) \end{bmatrix}_{U,V}$$

where the gray value F(p,q) in F is given by

$$F(p,q) = \sum_{r=0}^{m-1}\sum_{s=0}^{n-1} a_{pr} f(r,s) b_{sq}$$

This expression equals to F =A f B, where matrax A conists of

$$a_{pr} = \frac{e^{(\frac{-2\pi i r p}{m})}}{m}, \quad 0 \leqslant p,r \leqslant m-1$$

and matrix B consists of

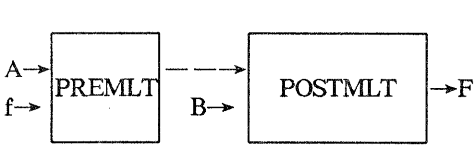
$$b_{sq} = \frac{e^{(\frac{-2\pi i s q}{n})}}{n}, \quad 0 \leqslant s,q \leqslant n-1$$

In fact,

$$A = \frac{1}{m}\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{(-2\pi i)/m} & e^{(-4\pi i)/m} & \cdots & e^{[-2\pi i(m-1)]/m} \\ 1 & e^{(-4\pi i)/m} & e^{(-8\pi i)/m} & \cdots & e^{[-4\pi i(m-1)]/m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & e^{[-2\pi i(m-1)]/m} & e^{[-4\pi i(m-1)]/m} & \cdots & e^{[-2\pi i(m-1)^2]/m} \end{bmatrix}$$

The matrix B is similarly described.

According to the calculating procedure described previously, the block diagram for the discrete Fourier transform is denoted by figure 7.



where. POSTMLT(f,B) performs

$$e_{pq} = \sum_{i=1}^{n} f_{pi} b_{iq}$$

if any f_{pi} is a star, so also is e_{pq}.
PREMLT(A,f) performs

$$d_{pq} = \sum_{i=1}^{m} A_{pi} f_{iq}$$

if any f_{iq} equals to a star, the d_{pq} is a star.

The destinations that we introduce DFT are:

(1). Since DFT can be indicated by matrix, other transforms can also be represented by matrix.

(2). Though current FFT is faster than the matrix method used in serial machines, we say it is very effeicient to be run in parallel array processors.

6. CONCLUSIONS

Digital topology and Matrix structure provide two sound mathematical foundations for image processing. Our thrust has been in the direction of recognition and decision. The goal of this investigation is to introduce underlying digital techniques that lead to the formation of quantitative knowledge parameters and to quantitative decision techniques, the both being central to the development of image matching and image understanding, since image matching and image understanding must employ the topological properties to obtain the satisfying results reliably and effectively and Matrix structure is a very useful vehicle for dealling with the parallel and formative algorithms. Of course, the real parallel algorithms to be preformed must have the enough parallel processors as its premise. Nevertheless, it is optimistic for us to see to popularize the parallel processors.

REFERENCES

Arcelli C. , 1979, A Condition for Digital Point Removal, Signal Process . 1, 283—285.

Dougherty E. and Giardina C. , 1987, Matrix Structured Image Processing, Prentice — Hall, Ins.

Hall R. W. , 1989, Fast Parallel Thinning Algorithms ; Parallel Speed and Connectivity Preservation, Commn. ACM 32, 124—131.

Herman G. T. , 1990, On Topology as Applied To Image Analysis, CVGIP 52, No. 3, 409—415.

Kong Y. and Rosenfeld A. , 1989, Digital

Topology; Introduction and Survey, CVGIP 48, No. 3, 358—393.
Lunscher W. H. H. J. and Beddoes M. P., 1987, Fast Binary — Image Boundary Extraction, CVGIP 38, 229—257.
Pavlids T., 1982, An Asynchronous Thinning Algorithm, CVGIP 20, 133—157.
Ronse C., 1986, A Topological Characterization of Thinning, Theoret. Comput. Sci. 43, 31—41.
Rosenfeld A. and Kak A. C., 1982, Digital Picture Processing, 2nd ed., vol. 2, Academic Press, New York.
Rosenfeld A., 1979, Picture Languages, Chap. 2, Academic Press. New York.
Rosenfeld A., 1974 Adjacency in Digital Picture, Inform. and Control 26, 24—33.
Suzak S. and ABE K., 1985, Topological Structural Analysis of Digigized Binary Images by Border Following, CVGIP 30., 32—46.
Tsao Y. F. and Fu K. S., 1982, A General Schame for Constructing Skeletion Models, Informa. Sc., 27, 53—87.
Wang ZhiZhuo, 1990, Principles of Photogrammetry (with Remote Sensing), WUTSM and Publishing House Of Surveying and Mapping.