Automatic stereo image matching using edge detection technique

Dr. Rehab H. Alwan,Mr. Mohamed A. Naji
Space Research center,Baghdad-IRAQ

Commission III WG / 4
KEYWORDS: Stereo image, edge detection,edge matching,edge linking

ABSTRACT

An edge matching technique has been used in this work where an algorithm was developed for detecting & coding the edge points which depends on the direction of the edge points the coding is followed by edge thining , edge linking & isolated points removing .Then points tracing process is performed to form the straight lines .

Lines matching operation is performed to group the lines in corresponding lines pairs .Features that are used in the matching process are ,line length ,line orientation ends point coordinates & line location .

1. Introduction

Edge-based matching is the process in which two representatives (edge) of the same object are pared together .Any edge or its representation on one image has to be compared and evaluated against all the edges on the other image .The matching is basically a selection process in which edges are pared according to some measures of similarity .

In order to make the edges in a form suitable for computer processing (especially stereo matching) it is usually of interest to use edge representations and descriptions .A lot of representations and approximations were proposed ,among these are :line representation (McIntosh,1988) ,polygon approximation (Greenfelot,1989) ,

ψ-s- representation (toth C.1992), Hough representation and so many others. Special interest will be focused here on the line approximation and matching.

2. Lines Exrtaction

Line approximation techniques have been studied since the early days of edge representations and approximations, and are described in text books (Gonzales R. and Winlz P., 1987) (Ballard .1982) .

Lines are commonly defined as collection of local edges that are contiguous in the image. Thus most of the algorithms rely on a two steps process for line extraction: detection of edges using one of the edge detection operators and then approximate these edges to lines.

2.1. Point Detection

The problem of detecting isolated points in an image applies in noise removal and particle analysis. The basic mask used for detecting isolated points in an image is shown in Fig.1.

-1	-1	-1
-1	8	-1
-1	-1	-1

Figure 1 : The mask used for detection isolated point

The center of this mask is moved from pixel to pixel and convolved with the previous mask.

In an area of constant gray level the result of the operation given above would be zero. On the other hand, if the mask is centered at an isolated point intensity is greater ((or less)) than the background then the result would be greater ((less)) than zero.

2.2. Edge Detection

Change or discontinuities in image attribute such as luminance and texture are fundamentally important primitive features of an image since they often provide an indication of the physical extent of objects within the image (Pratt .1978).

The purpose of edge detection is to generate a binary valued image from a detailed one containing the boundaries of the scenes or objects within the original image.

An edge be enhanced as the boundary between two regions with relatively distinct gray level properties. Basically, the idea underlying most edge detection techniques is the computation of local derivative operator.

The first derivative of an edge modeled in this manner is zero in all regions of constant gray-level, while the second derivative is zero in all locations except at the onset and termination of gray-level transiton. Based on these remarks, it is evident that the magnitude of the first derivative can be used to detect the presence of an edge, while the sign of the second derivative can be used to determine whether an edge pixel lies on the dark(e.g.background) or light (object) side of the edge.

The common methods used to calculate the derivatives are the gradient and Laplacian operations.

Several linear window operators were proposed to simplify the computations and save the time.

According to the following numbering :

$$\begin{bmatrix} a1 & a2 & a3 \\ a8 & a0 & a4 \\ a7 & a6 & a5 \end{bmatrix} \qquad (1)$$

where a_i , i= 1,2,.....,8 is the pixel gray-level within the window that is centered by (a0) which represents f(z,y). The image field g(x,y) can be found by using one of the following operators :

1) $g(x,y) = \text{Max}\,[|a_0 - a_2|\ ,\ |a_0 - a_4|\ ,\ |a_0 - a_6|\ ,\ |a_0 - a_8|]$ (2)

2) $g(x,y) = \frac{1}{4}\ (a_2 + a_4 + a_6 + a_8 - 4\,a_0)$ (3)

3) $g(x,y) = a_0 + \frac{1}{8} \sum_{i=1}^{8} a_i$ (4)

4) $g(x,y) = \text{Max}\,[|a_0 - a_2| + |a_0 - a_4| + |a_0 - a_6| + |a_0 - a_8|]$ (5)

Directional edge enhancement can be performed by convolving original image array with the compass gradient mask .

Another operator known as walls operators (Pratt ,1978), is used. According to this scheme an edge exists if the magnitude of logarithm of the image luminance at a pixel exceeds the magnitude of logarithm luminance of its four nearest neighbors by fixed threshold value;

$$g(x,y) = \log [\ f(x,y)\ ] - \frac{1}{4}\ [\ \log a_1 + \log a_3 + \log a_5 + \log a_7\ ] \qquad (6)$$

or

$$g(x,y) = \frac{1}{4}\ \log \left[\frac{f^4(x,y)}{a_1\, a_3\, a_5\, a_7}\right] \qquad (7)$$

2.2.1. Thresholding Techniques ((Binarization))

In digital image processing, thresholding is a well-known technique for image segmentation. Because of its wide application, quite a number of thresholding methods have been proposed over the years.

Let g(x,y) be an image to be segmented and T be a threshold. The result of thresholding an image function g(x,y) at gray level T is a binary image function e(x,y), such that :

$$e(x,y) = \begin{cases} b_0 & \text{if } g(x,y) < T \\ b_1 & \text{if } g(x,y) \geq T \end{cases} \qquad (8)$$

where {b 0 , b 1} is a pair of binary gray level.

In general, a threshold method is one that determines the value of T based on a certain criterion ,(Sahoo, 1988). Threshold selection is one of the key issues in image segmentation ((e.g. edge detection)). In threshold selection the following two points should be considered :

1- If the threshold level is set too high, it will not permit detection of low amplitude structural image elements.

2- If it is set too low then it will cause noise to be falsely detected as an image edge

2.2.2 Edge Thinning

One of the algorithms was developed by Zhang and Suen ,(Gonzalez,1987), for thinning binary image. In this algorithm it points have value(0). The method consists of successive process of two basic steps applied to the contour points of the given region, where a contour point is any pixel with value (1) and having one 8-neighbor valued (0) with reference to the 8-neighborhood definition shown in (Figure 2), the first step flag a contour point P1 for deletion if the following conditions are satisfied:

a) $2 \leq N(P_1) \leq 6$

b) $S(P_1) = 1$ (9)

c) $P_2.P_4.P_6 = 0$

d) $P_4.P_6.P_8 = 0$

where N(P1) is the number of non - zero neighbors of P1, that is

$N(P_1) = P_2+P_3+P_4+P_5+P_6+P_7+P_8+P_9$ (10)

and S(P1) is the number of (0-1) transitions in the order sequence of p2,p3,.......,p8,p9. for example, N(p1)= 4 and S(p1) = 3 in Figure 3 .

P_9	P_2	P_3
P_8	P_1	P_4
P_7	P_6	P_5

Figure 2 Pixels arrangement within a window

0	0	1
1	P_1	0
1	0	1

Figure 3 Illustration of conditions a and b in equation (9) In this case N(p1) = 4 and S(p1) = 3.

In the second step, condtions (a) and (b) remain the same but conditions (c) and (d) are changed to,

(c') $P_2.P_4.P_8 = 0$

(d')$P_2.P_6.P_8 = 0$ (11)

step (1) is applied to every broader pixel in the binary under consideration.

If one or more of the conditions (a) through (d) are vio- tep (1) is applied to every broat is not changed. If all condition are satisfied the point flagged for deletion.

2.2.3. Edge Linking

One of the simplest approaches for linking edge points is

to analyze the characteristics of pixels in a small neighborhood (e.g.3x3 or 5x5) about every point (x,y) in an image that has undergone an edge-detection process. All points that are similar are linked, thus forming the boundary of pixels that share some common properties.
The two principle properties used for establishing similarity of edge pixel in such analysis are :
1- The strength of the response of the gradient operator used to produce the edge pixel, and
2- The direction of the gradient.
The first property is given by the value of Gmag[f(x,y)] . Thus it is said that an edge pixel with coordinates (x,y) and in the predefined neighborhood of (x,y), is similar in magnitude to the pixel at (x,y) if

$$| G_{mag} [f(x,y)] - G_{mag} [f(x',y')] | \leq T \qquad (12)$$

Also it is said that an edge pixel at (x,y) in the predefined neighborhood of (x,y) has an angle similar to the pixel at (x,y) if

$$| \alpha (x,y) - \alpha (x',y') | \leq A \qquad (13)$$

where A is an angle threshold. The value of Gmag(.) and α (.) are calculated using the following equations .

$$G_{mag}[f(x,y)] = [G_X^2 + G_y^2]^{1/2} \qquad (14)$$

$$\alpha = \tan^{-1} (G_y / G_X) \qquad (15)$$

Based on the concepts given above, a point in the predefined neighborhood of (x,y) is linked to the pixel at (x,y) if both the magnitude and direction criteria are satisfied. This process is repeated for every location in the image.

3. Developed method

A developed method for extracting straight line segments is presented here. This method starts first with edge detection process. Figure 3 shows the results of different edge detectors with automatic thresholding technique. The Sobel operator is used since it provides clear edges and is not very much affected by pulsive noise.
Figure 5 shows edge maps using the thresholding technique mentioned above.
The automatic thresholding technique is done by dividing the gradient image g(x,y) into subimages with specific size, say (zxz). For each sub-image the average intensity (av) is calculated and each pixel within this subimage is compared with this average. An edge element is said to be present if the value of the pixel intensity is greater than the calculated average multiplied by a tuning factor(β). The above process can be implemented as follows:

$$av(x,y) = \frac{1}{z^2} \sum_{i=0}^{z-1} \sum_{j=0}^{z-1} g(x+i, y+j) \qquad (16)$$

$$e(x+i,y+j) = \begin{cases} 1 & \text{if } g(x+i,y+j) \geq \beta.av(x,y) \\ 0 & \text{otherwise} \end{cases} \qquad (17)$$

where x,y = o,z,2z .....(N-z),
N is the image dimensions , and
β is the tuning factor.
The value of β dictates the resultant edges. In order to obtain a large number of edges the value of b is set less than (1) and if only the strong edges are required the value of (β) is set greater than (1). Normally the value of β is almost always equal to (1).
The resultant edge image is coded to reduce the storage space required and speed up the processing. The edge element number corresponds to each direction. The value of this integer depends on the direction of the edge and the number of quantization levels.
The coded image is obtained according to the following criteria :-

$$e(x,y) = \begin{cases} 0 & \text{if } (G_{mag} < T) \\ 1 & \text{if } (G_{mag} > T) \text{ and } (0 \leq \alpha_{mod\ 180} < 45) \\ 2 & \text{if } (G_{mag} > T) \text{ and } (45 \leq \alpha_{mod\ 180} < 90) \\ 3 & \text{if } (G_{mag} > T) \text{ and } (90 \leq \alpha_{mod\ 180} < 135) \\ 4 & \text{if } (G_{mag} > T) \text{ and } (135 \leq \alpha_{mod\ 180} < 180) \end{cases} \qquad (18)$$

From equation (18) it is clear that (0) represents the background and the rest of the integers represents the edge elements.
In order to obtain an edge image containing this edges and only the dominant edges. One can merge the two edge images properties by multiplying them with each other ((considering that (0) represents background and (1) represent edge element)) as shown in Figure (6) .
Sometimes this process is used instead if using the thinning process.
While the use of thinning process is more effective way to obtain thin edges.
The thinning algorithm that have been discussed in section(2.2.2) are applied on the edges obtained form the edge detection and the results are shown in Figure 7.
The technique developed by (Nevatia and Babu, 1980) is used here. This method was chosen because it provides a thin and connected edges . A few minor modifi-

cations were made to this method to reduce the memory requirements and reduce the processing time. These modifications are done by using the developed coded edge images.
The thinning algorithm is followed by a linking characteristics of the implemented by analyzing the characteristics if the pixel neighborhoods, this analyses depends on the edge code.
The next step is removing the separate points by using the point detection technique discussed in section 2.1.
The point is removed if the result of convolving the image with the mask given in (Figure 1) is equal to (8).
The resultant edge image obtained above, which contains a thin linked coded edges, undergoes a tracing process, in which the edge elements are traced according to their orientation to form a straight line.
For each starting point ((a point which has no predecessor)) the tracing is carried out by seeking for the successor of this point, the seeking process is repeated from the current position until a point with no successor is reached. This point is recorded as an end point.
The obtained lines are identified by their two end points. Finally short lines are filtered out to reduce the possible corresponding ambiguity.

4. Straight Lines Matching

The matching algorithm finds corresponding pairs of lines from a pair of images, based on a match finction. This function combines several desripitive parameters-such as , length, orientation, light edge which may be defined as the average of lightest 10% of the gradient magnitude around the edge, (Mclutosh,1988), and end points coordinates of each line which are weighted according to their relative importance. Matched lines are those which have the largest match function values and correspond each other in both directions.
The match finction measures the similarity of the parameters attached to each line and is calculated for the *ith* parameter as followes:

$$M_{lr} = \sum_i v_i \; w_i \qquad (19)$$

1= 1,2,......,N,r = 1,2,....,M
where N is the number of lines in image L and M is the number of lines in image R,vi is the similarity value of the *ith* parameter and wi is the weight corresponding to the *ith* parameter, the sum of all the weights is (1)

$$\sum_i w_i = 1 \qquad (20)$$

Asimilarity measurement for all parameters ((but the angle and disparity parameters)) is defined as the ratio of the minimum of the parameter values for the two lines divided by the maximum to the two parameter. The similarity value v is defined as :

$$v_i = \frac{\text{Min [param}_i\text{ (line), param}_i\text{ (line r)]}}{\text{Max[param}_i\text{ (line), param}_i\text{ (line r)]}} \qquad (21)$$

where $0 \leq v_i \leq 1$

An absolute difference in values does not work as well due to the difference of measurement. The angle and disparity parameters must be compared in a slightly different manner because only the absolute difference between the two parameter values is relevant. This difference is compared to the expected maximum difference.

After similarity values v_i are calculated, a two similarity measures of each line from image (L) to each line from image (R). Each element in the array is the sum of the similarity values for each parameter multiplied by its corresponding weight value.
The similarity parameters that were used here in the implemented matching algorithm are : length, midpoint coordinates and orientation. The result are shown in Figure (8).

CONCLUSION

In this paper, we have described automatic system for stereo image matching, a system that uses edge based method.
The prime objective of this paper was to test the feasibility of using edges to match areal stereo pair , we developed an algorithm for extracting straight lines which starts with edge detection followed by edge linking, edge thinning, isolated points removing and points traeing to form the straight fines. Match function is calculated using line length orientation & end points coordinates to determine the similarity measure between the lines.

REFERENCES

- Ballord D.and Brown C.,1982. Computer vision, printice- Hall,Inc.
- Gonzalez R. and Wiulz P., 1987. Digital image processing, Addision-Wesley Pup.Con.Inc.
- Gireenfeld J.S. and Schenk A.F., 1989. Experiments with edge based stereo Matching, Photag. Eng. and Remate Sensing Vol.55 PP.171-180.
- Mcintosh J.H. and mAtch K.M., 1988. Matchinstraight lines, computer vision, graph and image processing Vol.40.PP.386-408.
- Nevatia R. and Babu K., 1980. Linears Feature Extraction and description,Conp. graph and image proc. Vol.13 PP.257-269.
- Pratt W.,1978 Degital image processing John wiely Sns Inc.

- Sahoo P., saftani S., Wong A. and chen y., 1988. Asurvey of thresholding techniques Computer Vision. graph and image processing,Vol 44 PP.233-260.
-Toth C. and schenk T.,1992 Fedurebased matching for outomaticreyistrat ITC journal PP.40-46.

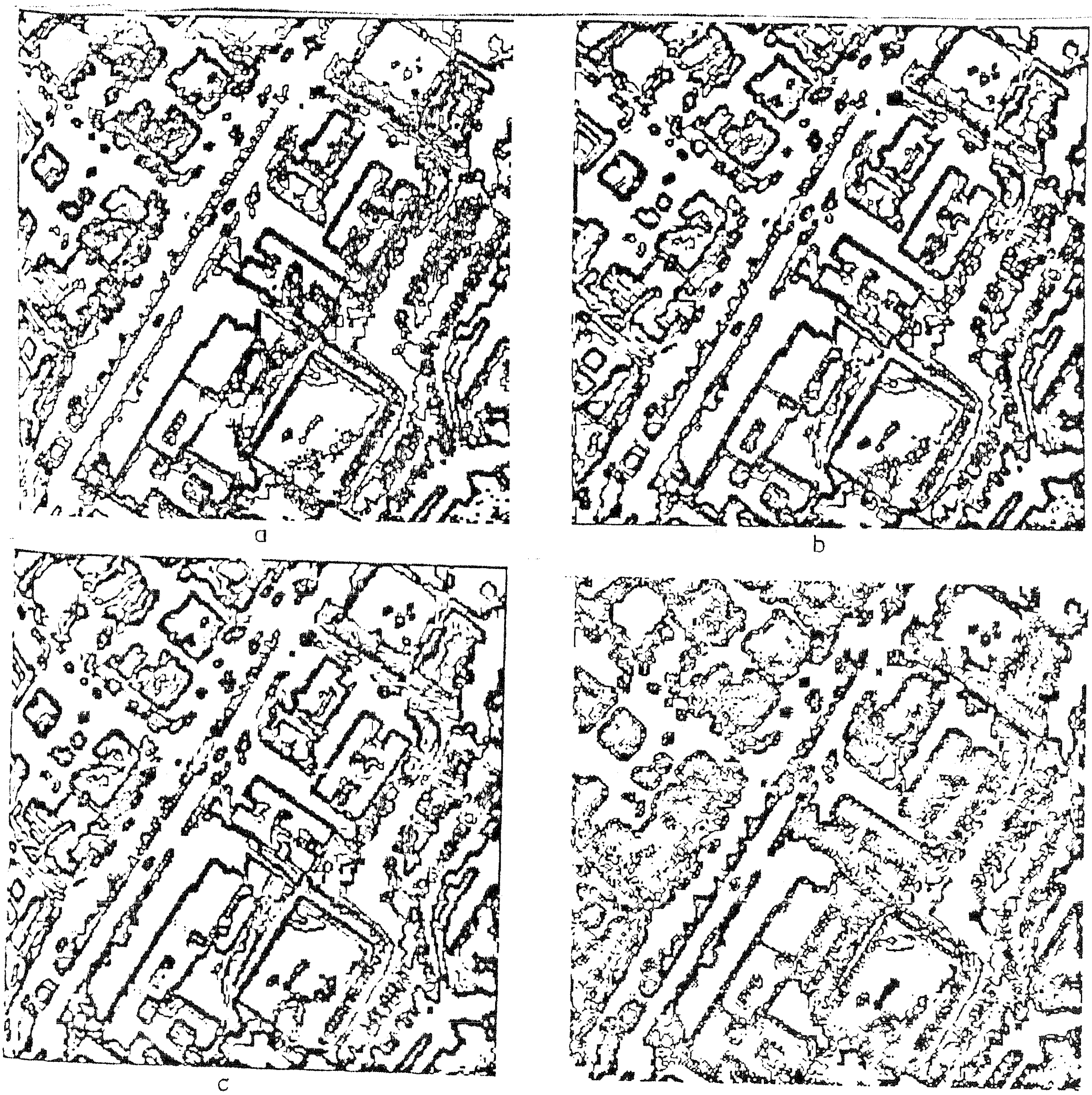
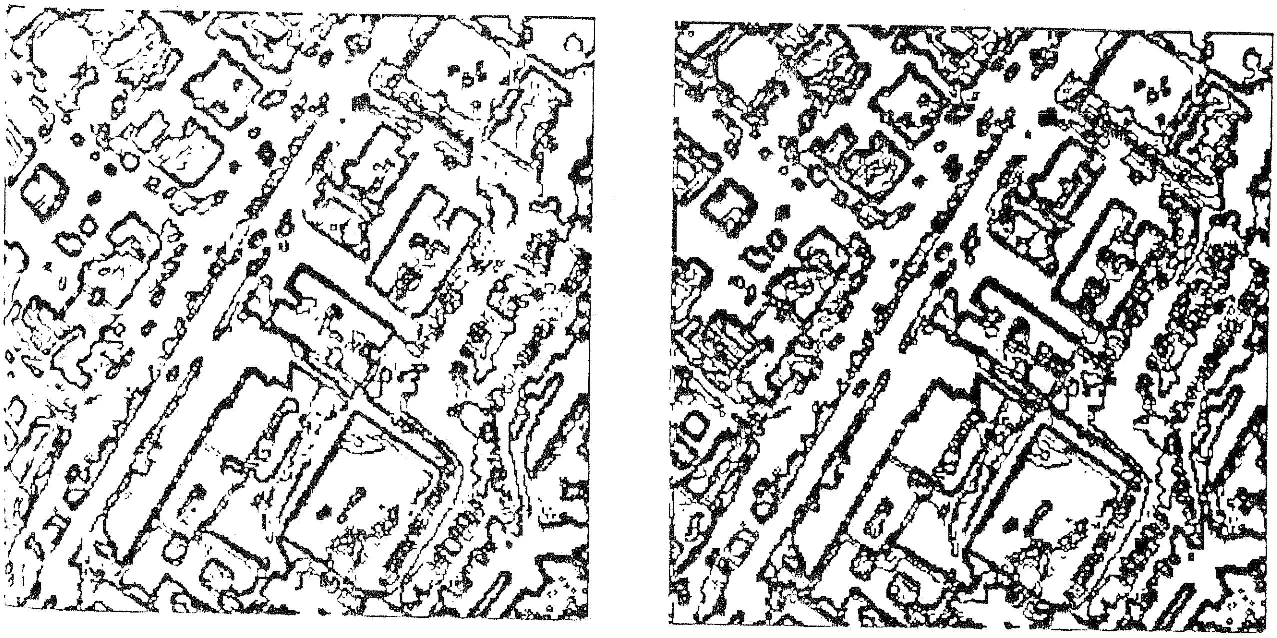
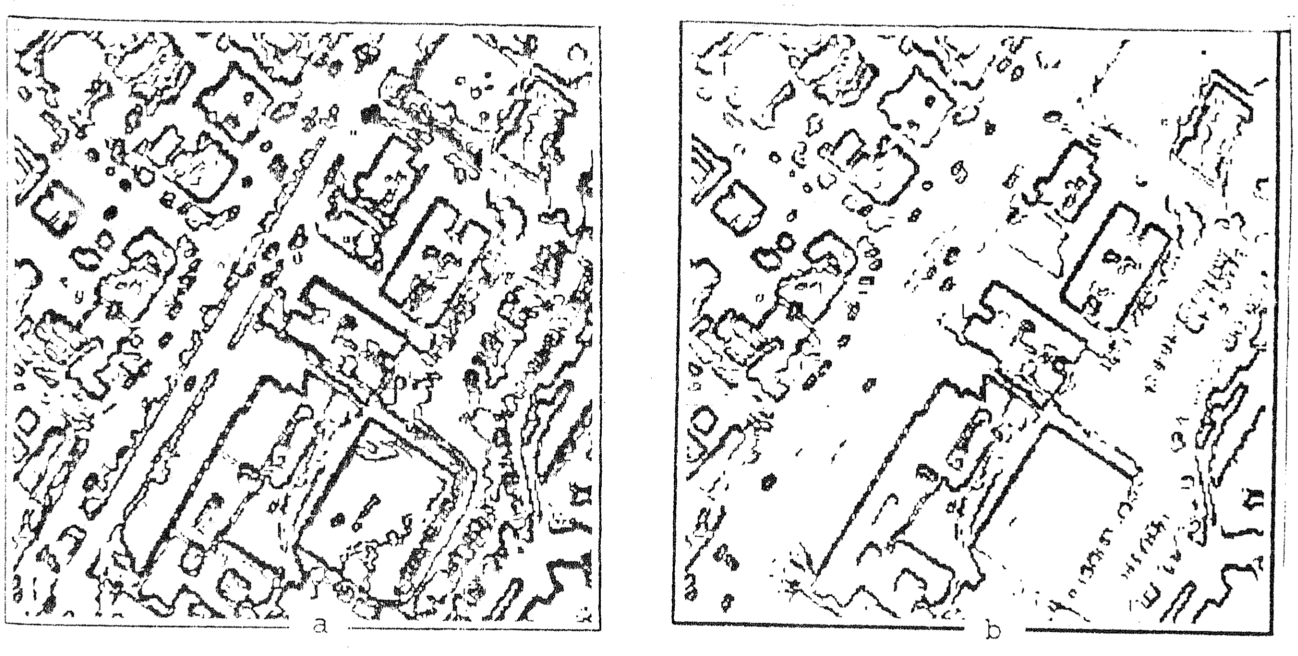


Figure (4) edge map :
a) Using Roberts operator
b) Using Sobel
c) Using Prewitt operator
d) Using Wallis operator



a b

Figure (5)
a) Automatic threshold edge map β = 1.2.
b) Automatic threshold edge map β = 0.8.



a b

Figure (6)
Multiplying of automatically threshold and globally threshold edge imges with each other

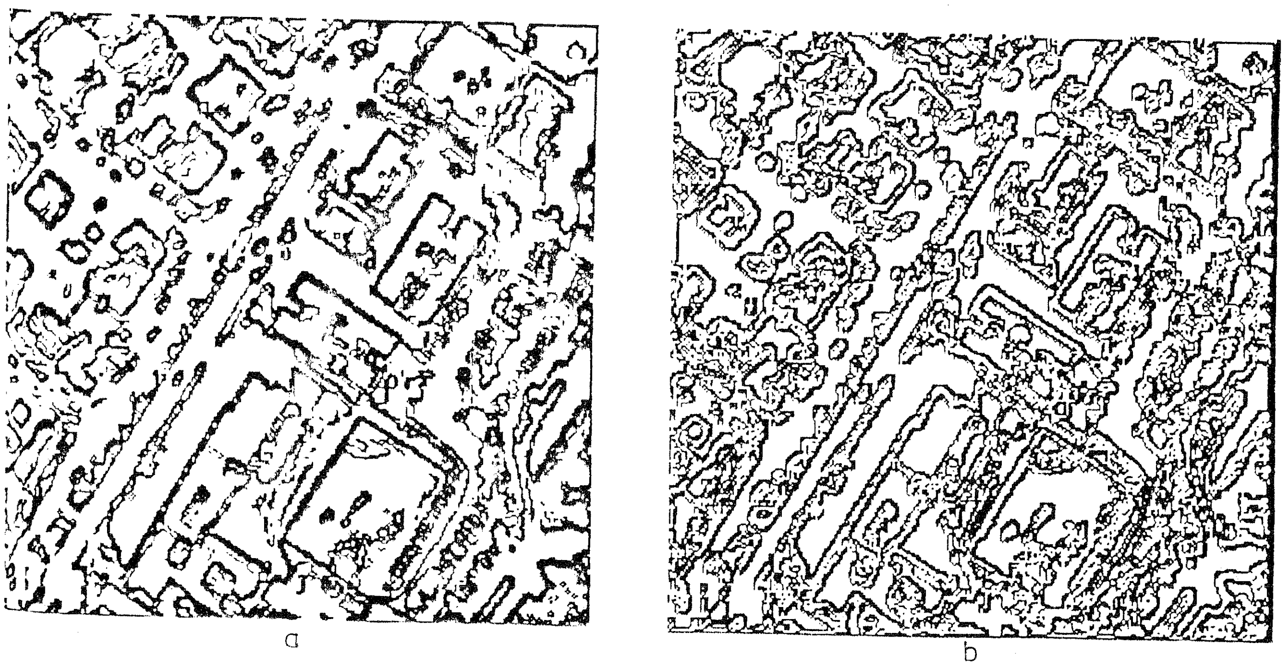


Figure 7 :
Edge thinning results
a) original edge image.
b) pointer to be removed in first step in Zhang's method.

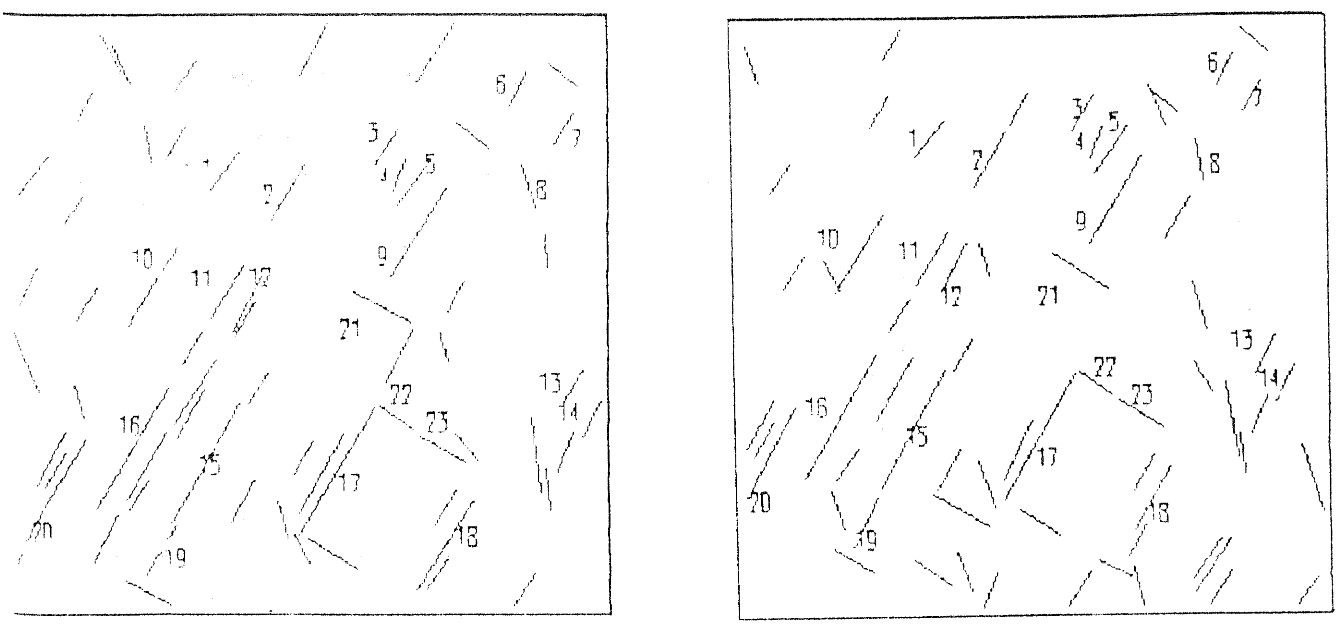


Figure 8 :
Corresponding lines obtained from applying the developed algorithm for matching straight lines segment .