AUTOMATED EXTRACTION OF AIRPORT RUNWAY PATTERNS FROM RADAR IMAGERY

Mr. Richard A. Hevenor, Electronics Engineer and
Dr. Pi-Fuay Chen, Electronics Engineer
U.S. Army Topographic Engineering Center
Fort Belvoir, Virginia 22060-5546
United States of America
ISPRS Technical Commission III

ABSTRACT:

A method is presented to extract linear terrain features from synthetic aperture radar imagery. An input radar image is smoothed with an edge preserving smoothing operation. Edge detection is performed using a Sobel operator, and both the magnitude and the directional images are computed. The edges are strengthened using several iterations of a relaxation operation in which both the magnitude image and the directional image are updated with each iteration. The output of the relaxation operation is a binary edge image, which is thinned. A connected components routine is run in which two passes through the image are used to provide a unique label for each connected component. The connected components related only to the runway pattern are extracted by computing certain properties of each component. A border-following algorithm is used to follow only the outermost borders and give each of the pixels on an outermost border a maximum brightness value. A tracking algorithm is used to change the binary image array into a set of Freeman chain codes, which serve as the input to a line-forming routine that uses a standard polygon approximation algorithm. Experimental results on a real synthetic aperture radar image are presented.

KEY WORDS: Image Analysis, Image Interpretation, Image Processing, Machine Vision, SAR

INTRODUCTION

The problem of extracting airport runway patterns from optical photography has been the subject of study for several years (Nevatia and Babu, 1980). In the referenced work, the emphasis was on low level vision computations, and little effort was made to isolate the connected components of the airfield. Also, very little work has been done in extracting airport runway patterns from synthetic aperture radar imagery. Since airports represent a potential military target, they should be extracted from radar imagery as quickly as possible.

The purpose of this paper is to present a systematic procedure consisting of a number of image processing algorithms that allow one to go from an original radar image containing an airfield to a binary image consisting only of components that are related to the airfield. All connected components in the image that relate to other terrain features will be eliminated. No attempt was made to make the procedure robust or general. The main objective was to see if the components of an airfield could be isolated for the sample image used. This image was a 512 by 512 pixel image that contained an airfield located near Elizabeth City, North Carolina (see Figure 1). The radar system used to obtain this image was the UPD-4 system which is an X-band radar with HH-polarization. The radar image was digitized with 8-bits. The image processing algorithms were written in the LISP programming language and executed on a Symbolics 3670 LISP machine. The same algorithms were later recoded in the C programming language and implemented on a SUN 3/180 microcomputer system for ease of transfer to a development laboratory. The rest of this paper will discuss the procedure used to extract the airfield from the radar image. Each of the algorithms used in this procedure will be briefly discussed. Finally, the results obtained after applying all the algorithms will be presented.

METHODOLOGY

The procedure for extracting the airfield from the image shown in Figure 1 is given in Figure 2. In this procedure, processing routines are applied that begin with edge preserving smoothing and end with polygon approximation. Some of the processing routines are elementary, such as edge detection; however, some of them are very complicated, such as relaxation for edge reinforcement and border following. The purpose of edge preserving smoothing is to eliminate noise and at the same time preserve edges so that they are not blurred. The edge detection algorithm will enhance edges and compute the direction of the edge at each pixel. The direction of each edge is needed as an input to the next algorithm which is relaxation for edge reinforcement. The purpose of this routine is to enhance edges using the contextual information of the surrounding neighborhood. The output of the relaxation calculations is a binary edge image in which many of the strong edges are too thick. A thinning operation is required in order to thin most edges down to the thickness of 1 pixel in width. The connected components algorithm provides a unique label (number) for each pixel in a given component of 1-pixels. Use is made of the definition of 8-connectivity for 1-pixels and 4-connectivity for 0-pixels. These definitions were also used for the thinning operation. Eleven region property calculations were performed to isolate the connected components that belong only to the airfield. A border-following algorithm is used to determine and uniquely label all of the 1-pixels that exist between a given connected component of 1-pixels and a connected component of 0-pixels. Two border-following algorithms were implemented for this work. The first algorithm finds and labels all border pixels for outer borders and hole borders. The second algorithm finds just the outermost borders. The difference between outer borders and hole borders will be explained in detail later. After border-following has been completed, lists are generated, each of which consists of an 8-direction chain together with the coordinates of the initial point in the chain. These lists are referred to as Freeman chain codes and are used as input data to a polygon approximation routine. The purpose of polygon approximation is to approximate irregular curves with a consecutive series of line segments. The particular polygon approximation routine used will be discussed later. The following sections of this paper discuss in more detail each of the algorithms mentioned above.

The final section will present the result of applying the algorithms to the radar image in Figure 1.

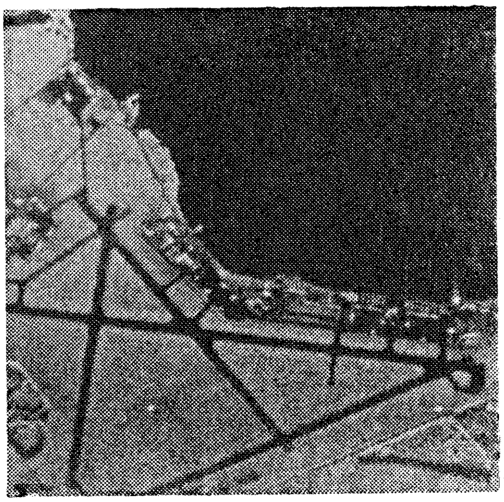

Figure 1. Original Radar Image of an Airfield.

PROCEDURE
EDGE PRESERVING SMOOTHING -> EDGE DETECTION (MAGNITUDE AND DIRECTION) -> RELAXATION FOR EDGE REINFORCEMENT -> THINNING -> CONNECTED COMPONENTS -> REGION PROPERTY CALCULATIONS -> EXTRACT CONNECTED COMPONENTS OF THE AIRFIELD -> BORDER FOLLOWING -> ELIMINATE PIXELS NOT ON THE OUTERMOST BORDERS -> GENERATE FREEMAN CHAIN CODES -> POLYGON APPROXIMATION

Figure 2. Procedure Used for Extracting the Airfield

Edge Preserving Smoothing

The purpose of an edge preserving smoothing algorithm is to eliminate noise and to preserve edges from degradation. The variation of the gray tone in a neighborhood around each pixel is used to determine the direction that is most homogeneous. Smoothing is then performed in this direction. The particular approach to edge preserving smoothing used in this research consisted of analyzing the gray tone variations within each 5- by 5-pixel area in the image. For each 5- by 5-pixel area, nine geometric figures are formed using the center pixel. Four of the geometric figures are pentagons. Four of the geometric figures are hexagons. One of the geometric figures is a square. Each of the four pentagon figures is formed by using the center pixel and one of the outermost edges of the 5- by 5-pixel area. Each of the hexagon figures is formed by using the center pixel and one of the outermost corners of the 5- by 5-pixel area. The 3- by 3-pixel square is formed using the center pixel and its first nearest neighbors. The pixels associated with each of the geometric figures are used to compute the mean and variance of the gray tone for each figure. The pentagon and hexagon figures each have 7 pixels associated with them. The square has 9 pixels associated with it. A list of nine means and nine variances is generated from all of the computations involving the nine geometric figures. The gray tone value of the center pixel is replaced by the particular mean gray value that is associated with the smallest variance. The theory behind this edge preserving technique was developed by Nagao and Matsuyama (Nagao and Matsuyama, 1980). The algorithm can also be used in an iterative manner, that is, the output of one smoothing operation can be used as the input to another.

Edge Detection (Magnitude and Direction)

After edge preserving smoothing has been performed, an edge detection operator is used to enhance edges and to compute the direction of each edge. The edge detection operator used was the Sobel operator. This operator consists of two 3 by 3 masks. The masks are applied to each pixel to calculate a magnitude image and a directional image. The magnitude image is the edge enhanced image. The directional image contains the direction of the edge at each pixel. The direction of an edge is defined as the angle between the edge and the x-axis. The x-axis extends along the top row of the image with the origin at the pixel in the upper left-hand corner. The y-axis extends downward along the first column of the image. The magnitude image is computed by taking the square root of the sum of the squares of the result of applying the two masks at each pixel. The directional image is calculated by taking the inverse tangent of the ratio of the results of applying the two masks. Because the direction of an edge has a 180 degree ambiguity, a convention must be established to eliminate this ambiguity and establish a fixed direction for each edge. The convention used in this research was that the edge direction was taken in such a way that the darker side is always on the left when facing in the direction of the edge.

Relaxation for Edge Reinforcement

The result of applying the Sobel edge operator yields an image in which some edges are defined very well, some edges are poorly defined, and some edges have holes in them. In addition, some large responses are obtained where there are no edges. These errors occur because of noise in the original image and also because the Sobel edge detector is not perfect. The purpose of the relaxation calculations is to enhance edges by increasing the gray tone value of the pixels that are really on edges, and to decrease the gray tone value of the pixels that are not on edges. Initially, the magnitude and direction of the edge at each pixel are obtained from the edge detection operation. The magnitude at each pixel location is divided by the maximum of the magnitudes over the entire image in order to define the probability of an edge at each pixel. The location of each pixel will be designated by the quantity (i,j), where i represents the row dimension and j represents the column dimension. The relaxation process for edge reinforcement consists of defining a new edge probability and a new edge angle at each (i,j) in terms of the old ones at (i,j) and its neighbors. The neighbors used in this research were the first and second nearest neighbors. This definition of neighbors is not a restriction on the basic technique. As explained by Schachter, et al. (Schachter, et al., 1977) the number of neighbors used in the relaxation calculations is arbitrary. However, if more neighbors are used, the result will be longer computation times. The calculation of the new edge probability and the new edge angle uses the compatibility coefficients Ree, Ren, Rne, Rnn. These coefficients are used to determine the relationship between a pixel (i,j) and one of its neighbors at (u.v). For example, if an edge exists at (i,j) and an edge exists at (u,v), one must have a quantitative relationship which explains how an edge at (u,v) will affect the edge at (i,j). In this example the relation is given by the compatibility coefficient Ree. If an edge exists at (i,j) and no edge exists at (u,v), the relation is given by the compatibility coefficient Ren. Ree is known as the edge/edge coefficient and depends upon the edge angles at the two points, on the angle between a line joining the point (i,j) with (u,v), and the x-axis, and on the distance between (i,j) and (u,v). Collinear edges will reinforce one another, while edges at different angles will weaken one another. Similarly, edge points are weakened by points containing no edges that are collinear with them, and the no-edge points are strengthened by edges alongside them. The definitions of the compatibility coefficients, the method of computing the edge probability, and the method of computing the new edge angle are the same as those given by Schachter, et al. (Schachter, et al., 1977). The details of the derivation of the relaxation equations are given in the paper by Schachter, et al. (Schachter, et al., 1977) and also in the report by Hevenor and Chen (Hevenor and Chen, 1990).

Thinning

After several iterations of relaxation, the image is essentially binary. However, some of the strong edges are now too thick, and a thinning routine must be used in order to obtain linelike patterns. We assume that the image consists of only two gray values, represented by 0 and 1. A frame around the image consisting of the first row, the first column, the last row, and the last column will be assumed to contain only 0 pixels. Consider a pixel and its eight neighbors. If the center pixel has the value of 1, then a 1 in any of the 8 neighboring positions will be considered connected to the center pixel. This is known as 8-connectivity. We will assume 8-connectivity exists for 1-pixels and 4-connectivity exists for 0-pixels. A thinning routine was used which makes use of computing the eight connectivity number developed by Yokoi, Toriwaki, and Fukumura (Yokoi, et al., 1975) to analyze the topological properties of a binary image. The details of the thinning routine are presented in the report by Hevenor and Chen (Hevenor and Chen, 1990).

Connected Components

The purpose of the connected components routine is to provide a unique label for each component of 1-pixels in the binary image that has been thinned. Each label is a number that is assigned to every pixel in a given connected component. This labeling operation can be performed by scanning the entire binary image with a 3 by 3 array and considering the following pattern:

C	B	E
D	A	

If we scan along the image from left to right and from top to bottom, and if pixel A is the pixel presently being considered and it has a value of 1, then a label must be assigned to A. If the pixels at D, C, B and E are all 0, then A is given a new label. If pixels C, B and E are all 0 and D = 1, then A is given the label of D. Each possible construction of 0's and 1's for the pixels D, C, B and E must be considered when providing a label for A. If two or more pixels in the set D, C, B and E are equal to 1 and they all have the same label, then A is also given the same label. The real difficulty comes when two or more of the pixels D, C, B and E have different labels. This can occur when two or more separate components, which were originally assigned different labels are found to be connected at or near pixel A. For these cases the pixel A is given the label of any one of the pixels D, C, B or E, which has a value of 1. An equivalence list consisting of pairs of equivalent labels is formed. After the binary image has been completely scanned, the equivalence list is restructured in such a way as to contain a number of lists. Each of these lists contains all of the equivalent labels associated with a particular connected component. A new label is then given to each of the new lists and is assigned to each of the appropriate pixels.

Region Property Calculations

The purpose of region property calculation is to determine if the computation of certain quantities can be used to isolate the particular connected components that belong only to the airfield runway pattern. Computations are performed on each connected component. Eleven region properties were used in this research and are:

1. Area
2. Centroid
3. Orientation of the axis of least inertia
4. Maximum moment of inertia
5. Minimum moment of inertia
6. Elongation
7. Measure of region spread
8. Scatter matrix
9. Eigenvalues of the scatter matrix
10. Perimeter
11. Compactness

Border Following

Border following determines and uniquely labels all 1-pixels that exist between a given connected component of 1-pixels and a connected component of 0-pixels. A border point is defined as a 1-pixel that has at least one 0-pixel in its 4-neighborhood. A border point can also be described as a 1-pixel located on the boundary between a connected component of 1-pixels and a connected component of 0-pixels. The frame of a binary image consists of the first row, the last row, the first column and the last column. It will be assumed that all the pixels on the frame have a gray value of 0. There are two types of borders that can exist in binary images of interest, outer borders and hole borders. An outer border is defined as the set of 1-pixels located between a connected component of

1-pixels and a connected component of 0-pixels which surrounds it directly. A hole border is defined as the set of 1-pixels located between a connected component of 0-pixels and a connected component of 1-pixels that surrounds it directly. For both outer borders and hole borders, the border itself is defined by a set of 1-pixels. The 0-pixels are never used to define a border, except in the case of the frame of the image that is considered a special hole border. For any given connected component of 1-pixels there is one, and only one, outer border and it is unique. There are two algorithms that were implemented for border following. The theory behind both of these algorithms was developed by Suzuki and Abe (Suzuki and Abe, 1985) and the details are given there.

Generate Freeman Chain Codes

Once the border pixels have been found, the binary image must be converted into a format that can be used to form line segments. The Freeman chain code is used to represent a boundary with a sequence of numbers, each of which indicates the change in direction from one border pixel to the next. An 8-direction chain was used. Each number corresponds to a particular change in direction from one pixel to the next. For example, the number 0 represents a change in direction of 0 degree from one pixel to the next. The angular change is measured from the horizontal to a line formed by joining the current pixel in the chain to the next pixel in the chain. The number 4 represents a change of 180 degrees from one pixel to the next, etc. One entire chain code is represented by a list whose first two elements are the (i,j) coordinates of the initial point in the chain, followed by a sequence of numbers each of which comes from the set {0, 1, 2, 3, 4, 5, 6, 7}. Each of these numbers represents the change in direction from one pixel to the next. The entire binary image will consist of many such chains. To form the chain codes, a tracking routine was developed that examines the nearest neighbors of a pixel in the chain to find the next pixel in the chain. An index array, which is as big as the original image and is initially set to contain all zeros, is used to indicate if a 1 has already been included in a chain or not. The input binary image is scanned by starting with the pixel in the upper left-hand corner of the image. When a 1 is found at the location (i,j) and the contents of the index array at (i,j) is 0, a new chain is started. As each pixel in the chain is found, the appropriate number from the set {0, 1, 2, 3, 4, 5, 6, 7} is placed into the chain list and a 1 is placed into the corresponding location of the index array. When the scan reaches the pixel in the lower right-hand corner, the algorithm terminates.

Polygon Approximation

An arbitrary, two-dimensional, open or closed digital curve is represented by a Freeman chain code and consists of an ordered set C of N pixels. The purpose of polygon approximation is to replace each digital curve with a consecutive series of line segments, each of which are within a certain error of the original curve. There are many polygon approximation techniques, and the theory behind the one used in this research was developed by Ramer (Ramer, 1972). The pixels in the set C can be considered as the vertices P_i of a polygon if the curve is closed. If the curve is not closed, the set C can be considered as the vertices of a consecutive series of line segments. In either case, the task of the polygon approximation algorithm is to provide a reduced set C' with a smaller number of edges N' and whose vertices P'_k coincide with those of the original set C.

RESULTS

The result of applying the above mentioned algorithms to the radar image shown in Figure 1 is given below in Figure 3. The region properties of area, orientation of the axis of least inertia, and the measure of region spread were the only properties required to obtain this final result. It can be seen that the runway patterns have clearly been delineated with line segments and the other components in the image have been eliminated. The image is virtually noise free and presents a good extraction of the runway pattern shown in Figure 1.

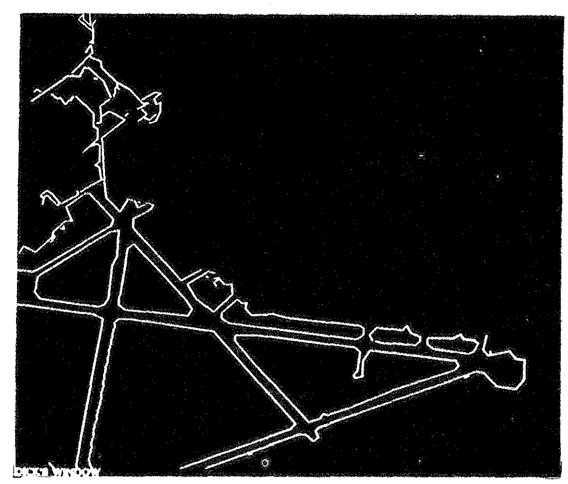

Figure 3. Polygon Approximation.

CONCLUSIONS

1. The approach taken to extract airfields from radar imagery is valid.

2. Although the method of edge reinforcement using relaxation is computationally intensive, it produces excellent results.

3. The components of the airfield were extracted using the three region property calculations of area, orientation of the axis of least inertia, and the measure of region spread.

REFERENCES

+ References from JOURNALS:

a) Nevatia R. and Babu K.R., 1980. Linear Feature Extraction and Description. Computer Graphics and Image Processing, Vol. 13, pp. 257-269.

b) Ramer, U., 1972. An Iterative Procedure for Polygonal Approximation of Plane Curves. Computer Graphics and Image Processing, No. 1, pp. 244-256.

c) Schachter, B.J., Lev, A., Zucker, S.W., and Rosenfeld, A., 1977. An Application of

Relaxation Methods to Edge Reinforcement. IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-7, No. 11, pp. 813-816.

d) Suzuki, S., and Abe, K., 1985. Topological Structural Analysis of Digitized Binary Images by Border Following. Computer Vision, Graphics and Image Processing, No. 30, pp. 32-46.

e) Yokoi, S., Toriwaki, J., and Fukumura, T., 1975. An Analysis of Topological Properties of Digitized Binary Pictures Using Local Features. Computer Graphics and Image Processing, No. 4, pp. 63-73.

f) Zhou, Y.T., Venkateswar, V., and Chellappa, R., 1989. Edge Detection and Linear Feature Extraction Using a 2-D Random Field Model. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No. 1, pp. 84-95.

+ References from BOOKS:
Nagao M. and Matsuyama T., 1980. A Structural Analysis of Complex Aerial Photographs. Plenum Press, New York, pp. 32-52.

+ References from GRAY LITERATURES:
Hevenor, R.A., and Chen, P.F., 1990. Automated Extraction of Airport Runway Patterns from Radar Imagery. Research Note ETL-0567, U.S. Army Engineer Topographic Laboratories, Fort Belvoir, VA-USA.