Coastline Detection and Tracing in SAR Images

JONG-SEN LEE, MEMBER, IEEE, AND IGOR JURKEVICH

Abstract—**Images of coastlines generated by Synthetic Aperture Radars (SAR's) suffer from a number of deficiencies which arise from the presence of the speckle effect and the strong signal return from a wind-roughened, wave-modulated sea. The frequent lack of contrast caused by these effects makes coastline detection difficult by most conventional procedures such as gray-level thresholding or segmentation by edge magnitude. This paper describes an algorithm for the global detection of coastlines based on a sequence of basic image-processing procedures and a new edge tracing algorithm. The application of the proposed procedure to Seasat SAR and SIR-B (Shuttle imaging Radar B) images demonstrates that with only a modest computational burden it produces a good visual match between the detected coastline and the coastline of original image. Additionally, the separation of land from water achieved by this algorithm permits clean pseudocoloring of coastal area images.**

I. Introduction

COASTLINE detection in SAR images is an important step toward the computer scene description of coastal areas. Knowledge of a coastline's orientation, position, and outline is essential in activities such as autonomous navigation, the verification of a radar platform's attitude and position, the geolocation of targets (e.g., ships), geographic mapping, etc. Coastline detection in SAR images is not, however, as simple a procedure as it is, for instance, in photographic or Landsat Thematic Mapper (TM) images. In the latter case, especially in Band-4 images, a simple act of thresholding followed by edge detection will effectively extract the coastline from the image. The difficulties in SAR images are associated with the nature of the signal return from the ocean and land areas. The return from the wind-roughened and wave-modulated water can frequently equal or exceed the return from a nearby land area, resulting in an inadequate contrast for unambiguous separation. Furthermore, the presence of the speckle effect, generated by the coherent signal-scattering within SAR resolution elements, complicates the detection problem. As a consequence, the coastline in many SAR images is so indistinct that even experienced observers have difficulties discerning it without the aid of a topographic map or other geographic knowledge.

Problems associated with coastline detection in SAR images have been considered in studies such as [1]. In the latter, digital contour maps are used to simulate a SAR image, and then the coastline is defined by matching the simulated SAR image with the one under investigation. The purpose of this paper is to describe an alternative method for detecting coastlines with reasonable accuracy.

Coastline detection is in the class of boundary detection problems. Similar problems have been encountered in areas such as the determination of chest and heart boundaries in radiographs and cineangiograms [2] as well as object recognition in various remotely sensed images. The algorithms developed in these studies are problem-dependent. Specific knowledge concerning the boundary is used to form rules that guide the grouping of pixels into boundaries. We have examined many SAR images containing coastlines and observe that the ocean areas, in general, are more homogeneous in the gray level than the land areas. The difficulty in using edge maps in defining a boundary is that the strong edges are not continuous. The boundary tracing by edges is an untractable programming problem which, even in a crude approximation, is very computationally intensive [3]. The local edge tracing in coastline detection must be guided by the global information about the coastline; that is, information extending over the whole image. Hence the first step is to obtain a rough separation between the land and water. Refinement is then made in the neighborhood of the roughly defined boundaries.

The development of the coastline-detection algorithm in this paper is covered in the following sections. A Seasat SAR image (512 × 512 pixels) of the Chesapeake Bay area (Fig. 1(a)) processed at four looks is used to illustrate the algorithm development. In section II a simple speckle-smoothing algorithm [4] is applied to the image to reduce speckle without blurring the major edge features. A global ocean- and land-detection algorithm is described in Section III. Section IV is devoted to the rough coastline detection and the description of a contour-tracing algorithm. The refinement of the detected coastline is carried out in Section V. The application to pseudocoloring of some typical coastal SAR images is discussed in Section VI. Section VII is devoted to remarks and discussions.

II. Pre-Processing by Speckle Reduction

It is well known that speckle appearing in SAR images impedes the interpretation of the image either by a human operator or a computer. It has been shown [5], [7] that the speckle distribution is described quite accurately by a multiplicative noise model. The basic relation of this model is given by

$$z_{i,j} = x_{i,j}\, v_{i,j}$$

Manuscript revised March 2, 1990.

The authors are with the Center for Advanced Space Sensing, Naval Research Laboratory, Washington, DC 20375.

IEEE Log Number 9035819.

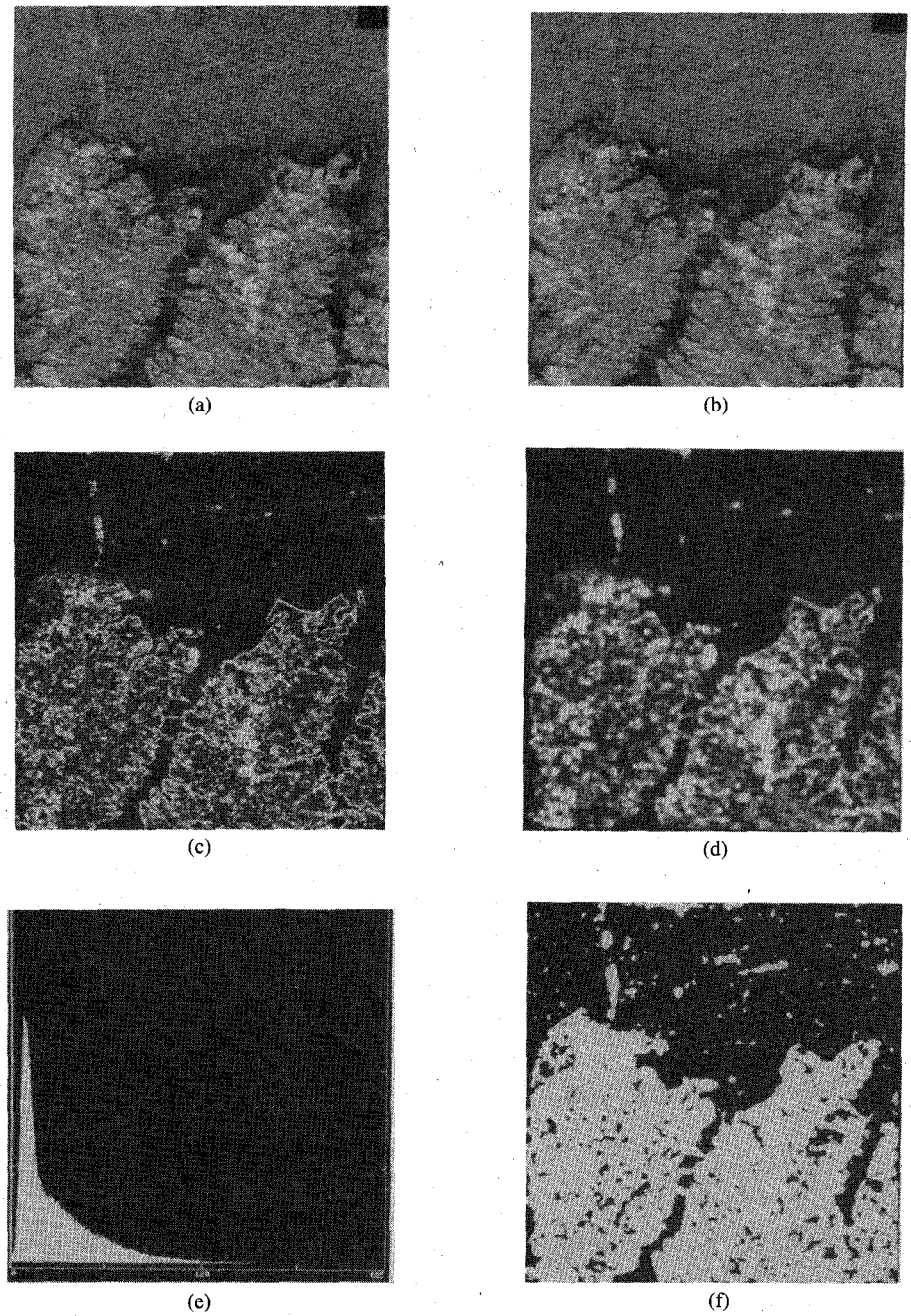


Fig. 1. Seasat SAR image of the Cheasapeake Bay is used to illustrate the procedure of coastline detection. (a) Seasat SAR original. (b) Sigma filter smoothed (twice). (c) Edge map of (b). (d) 5 × 5 mean filter (twice). (e) Histogram of (d). (f) Threshold applied to (e).

and

$$v_{i,j} \sim (1, \sigma^2)$$

where $z_{i,j}$ is the gray level of the observed SAR pixel, $x_{i,j}$ is its ideal or noise-free counterpart, and $v_{i,j}$ is the noise, characterized by a distribution with mean = 1 and variance σ^2. For instance, for 1-look-amplitude SAR images, $\sigma = 0.52$, and for 4-look-amplitude images, $\sigma = 0.25$. Several algorithms [4]–[6] based on the multiplicative noise model of the speckle effect have been developed to smooth the speckle noise without degrading the sharpness of the major edges in the image. The sigma filter, described in detail in [4], is selected for the present study. A filtered pixel value is represented by an average of those pixels within the range of two standard deviations (2σ) of the center pixel. For SAR images (multiplicative noise), the 2σ range is given by

$$(z_{i,j} - 2\sigma z_{i,j},\ z_{i,j} + 2\sigma z_{i,j}).$$

The result of smoothing a typical image twice with a (5 × 5) window is shown in Fig. 1(b). The speckle has been reduced considerably, while coastline edges remain unaffected. It is to be noted that the sigma filter was applied iteratively. The reasons why this can be done are discussed in detail in [12].

It is apparent that it is impossible to segment either Fig. 1(a) or (b) by grey-level thresholding, since many land areas have lower or equal grey levels than some of the ocean areas. To proceed, a Sobel edge operator [9] was then applied to the data to generate the preliminary edge map shown in Fig. 1(c).

III. Sea and Land Separation

To link the coastline edges displayed in Fig. 1(c) is not a simple task. As can be seen, gaps exist in the coastline, and some strong edges would mislead a tracing into the inland area. For this reason a simple procedure is implemented which generates an approximation to the land boundary. The edge map of Fig. 1(c) is quite discontinuous despite the fact that speckle has been smoothed. To fill the gaps between neighboring pixels, the edge map is dilated by applying a (5×5) mean filter to the data of Fig. 1(c). Because it is found that a single pass through the filter produces inadequate broadening, the data is subjected to this treatment one more time. This step produces the image shown in Fig. 1(d). The histogram of the latter image is then computed and displayed in Fig. 1(e). Assume at this point that at least 20% of the total number of pixels belong to the land or the sea. Since the sea area is more homogeneous than the land area, the corresponding sea-area pixels will peak around the lower gray levels. Furthermore, the averaging operation performed on Fig. 1(c) makes the sea pixels in Fig. 1(d) normally distributed. The mean and standard deviation of the truncated normal distribution are estimated according to the procedure in [8]. A threshold is then set at (mean + 2x(standard deviation)) and applied to the data of Fig. 1(d). The result of this operation is shown in Fig. 1(f).

In the next section a newly devised edge-tracing algorithm is described.

IV. Preliminary Coastline Tracing

At this point the coastline can be traced by either of the following two methods:

i) The first approach starts with a coastline pixel (i.e., the boundary pixel in Fig. 1(f)) and applies the techniques of countour-following such as those described by Duda and Hart [10] or Rosenfeld [11].

ii) The second approach applies an edge operator to the binarized version of Fig. 1(f) and then applies an edge-tracing algorithm.

The second approach is chosen for our work because it is felt that it is somewhat easier to implement in a computer program.

There exists a rich literature on various edge-following techniques. For instance, the end-point method can be applied to the problem with low curvature edges. The Hough transform is suitable for the detection of straight edges or edges of a particular shape. Under appropriate conditions even dynamic programming techniques can be used [3].

Data in Fig. 1(f) are operated upon by the Robert's edge operator and the result is shown in Fig. 2(a). The reason for applying Robert's operator [9] is that the edges generated are 1-pixel wide, and that will make edge tracing more precise. Since the edge map is obtained from the thresholded image, the coastline in the edge map will be continuous. The following procedure is then used to trace the coastline:

Step 1. Selection of a Starting Coastline Pixel

The starting pixel for tracing is selected from the data comprising the threshold binary image shown in Fig. 1(f). Scanning the image row-wise (or column-wise), we select the starting coastline pixel by the following criteria: (i) The pixel must be a boundary pixel which divides a continuous sequence of land (bright) pixels and a continuous sequence of sea (dark) pixels. For this application, the run of the continuous dark and bright pixels is chosen to be at least 50 pixels; and (ii) from a pixel satisfying the above criterion, a cluster of bright pixels from the eight-connected neighbors is calculated to evaluate the size of the potential land area. The minimum number of pixels in the cluster is established from consideration of the false land area to be avoided and the size of a closed contour (e.g., an island) we are interested in tracing. In the present application, the size of the cluster must be greater than 2500 pixels for the pixel to be considered a starting coastline pixel. Generally, the above criteria take into account the pixel spacing and size of the area enclosed by the contour to be traced.

Step 2. Definition of the Tracing Direction

From the starting pixel, find the 8-connected neighboring edge points and use one of the 8-connected pixels and the center pixel to define the starting tracing direction. Eight tracing directions are allowed, as indicated in Fig. 3. The starting pixel and edge pixel (current pixel) defining the tracing direction are written onto the output coastline image and their coordinates are stored in a large array for later use. To avoid the problem of endless tracing in closed loops, the previous starting pixel is deleted (in Fig 2(a)).

Step 3. Determination of the Next Coastline Pixel

The next coastline pixel is chosen from the current pixel in the tracing direction. Only pixels associated with the direction index, defined in Step 2, are considered as candidates. They are shown as square boxes in Fig. 4, where the pixel marked with an "x" is the current pixel and (0, 1, 2, 3, 4, 5, 6, 7,) are direction indices. Should there be more than one candidate pixel available (i.e., more than one edge pixel in the directional mask), the edge pixel closer to the search direction, as indicated by the directional index, is chosen first. The procedure is carried one step further, taking the candidate pixel as the new current pixel to determine if a new candidate pixel exists. In other words, we look one step ahead to determine whether to

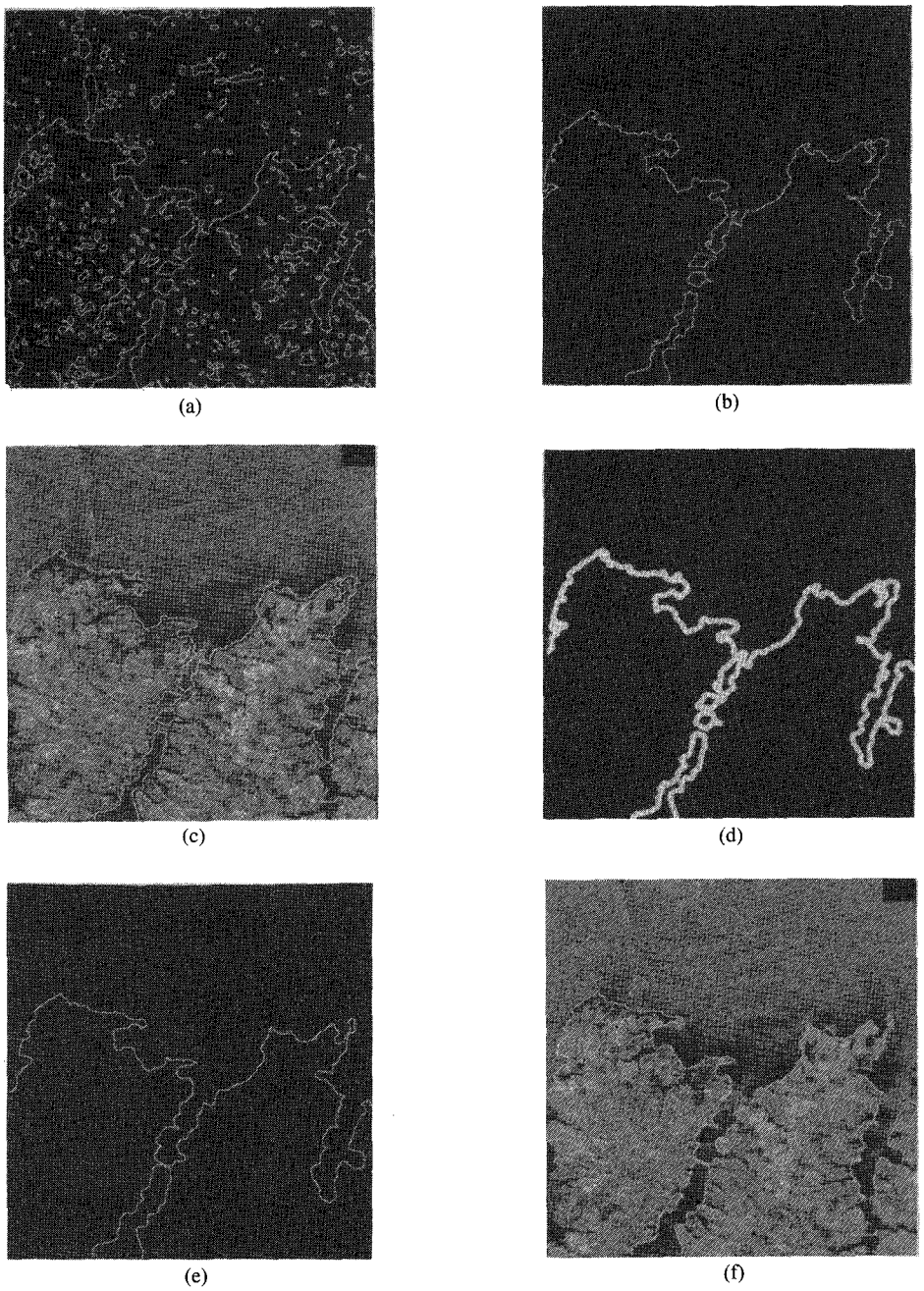


Fig. 2. Coastline tracing and refinement. (a) Robert's edge map of Fig. 1(f). (b) Coastline traced from (a). (c) Coastline of (b) overlaying Fig. 1(a). (d) Mean filter applied twice to (c) and thresholded. (e) Refined coastline detected. (f) Coastline of (e) overlaid on Fig. 1(a).

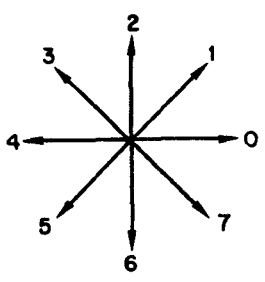


Fig. 3. Direction index.

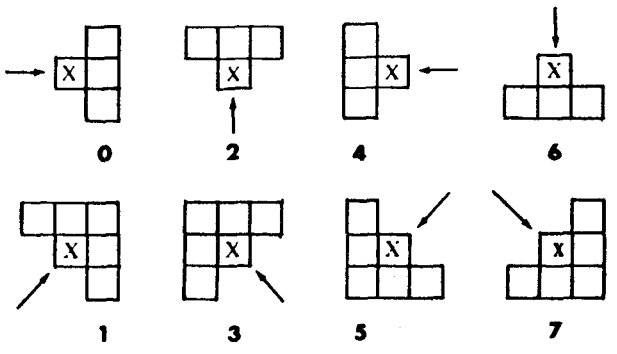


Fig. 4. Directional masks.

accept the candidate pixel as the coastline pixel. The coastline pixel and other candidate pixels in the masks are then deleted in Fig. 2(a) to avoid endless tracing in closed loops.

It should be noted that due to the contrast and resolution limitations of these SAR images, one can encounter various coastline features that cannot be handled with the assumed directional mask. Typically, multiple small loops are in this category. As the algorithm proceeds along such a feature, a point is reached where no new candidate pixel is available due to the deletion of traced edge pixels, and the algorithm stalls. This problem is overcome by keeping a record of the traced coastline pixels and then backtracing the coastline from the stalled pixel, one pixel at a time. Since the previously traced coastline pixel and other candidate pixels are eliminated in Fig. 2(a), pixels in a (5 × 5) window centered at the backtraced pixel are considered as candidates (or new starting pixels) and are processed as in Step 2. Pixels in the center (3 × 3) neighborhood of the (5 × 5) window are tested first. Then the remaining pixels are checked, provided that no coastline pixel was found in the 3 × 3 neighborhood. The process continues until a new coastline pixel is found.

Step 4. Coastline Tracing in the Opposite Direction

The tracing ends when the procedure encounters the image boundary. In such a case the procedure returns to Step 2 to start the tracing in the other direction. If the tracing terminates at the original starting pixel, an island or lake has been detected.

Step 5. Tracing of the Lake and Waterway Inlet Boundaries

Large lakes or waterway inlets can also be traced if necessary. The result of such computations is demonstrated in Fig. 2(b), where one inlet is traced.

To gain an appreciation of the precision of the algorithm, as formulated to this point, we overlay the coastline with the original image and show the result in Fig. 2(c). The detected coastline pixels are, on average, six-to-eight pixels away from the original image coastline pixels. This mismatch occurs primarily due to the application of a (5 × 5) window mean filter to generate Fig. 1(d) and the fact that it is the data from the latter figure that is used to obtain the preliminary coastline trace shown in Fig. 2(b). A refinement of the procedure, designed to take care of this problem, is presented in the next section. However, for the purpose of geolocation the procedure is sufficiently accurate even without this added step.

V. Refinement

This section deals with the visually observed discrepancy between the detected coastline and the coastline in the original image. A simple algorithm is introduced which effectively decreases this error. The approximately defined coastline of Fig. 2(c) is dilated by applying a (5 × 5) mean filter twice and then thresholding the result to produce Fig. 2(d). The coastline is retraced with the algorithm of Section IV. However, only the inside edges are traced. For closed features (e.g., the lakes, interior to the coastline) the outside edge is traced. The result of this operation in shown in Fig. 2(e) and its overlay with the original image in Fig. 2(f). The refined coastline matches that of the original image to within a pixel or two. The inlet in the right lower corner is not detected because of the presence of a bridge and the fact that this body of water is too narrow. Further refinements are possible due to the fact that radar backscattering from inlets and lakes is lower and more homogeneous than from the surrounding land areas.

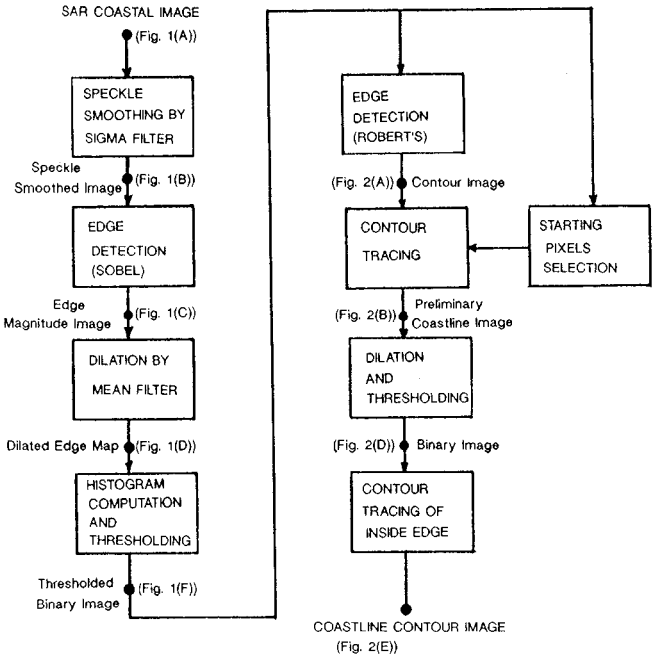


Fig. 5. Block diagram of coastline detection and training.

For clarity, a block diagram outlining the algorithm is given in Fig. 5.

Another example of coastline detection by this algorithm is given in Fig. 6. Fig. 6(a) shows a SIR-B image of Martha's Vineyard, Massachusetts. Fig. 6(b) shows the preliminary coastline detection, and Fig. 6(c) shows the results of the refinement described in Fig. 6(d). A reasonably good fit, based on visual examination, substantiates the development of this algorithm.

VI. Pseudocoloring of Coastal SAR Images

An interesting application of the SAR coastline-detection algorithm is the pseudocoloring of the coastal SAR images. The difficulties in assigning colors to gray levels in SAR images arise from the overlap of gray levels between the land and sea pixels for the reason discussed in Section I. The detection of the coastline makes it possible to code the land area and sea area separately. This can easily be accomplished by generating, for instance, a mask for land areas using Fig. 2(e) and an area fill interior to the boundary. The application of this mask to Fig. 1(a) produces, in effect, two images. One contains mostly land

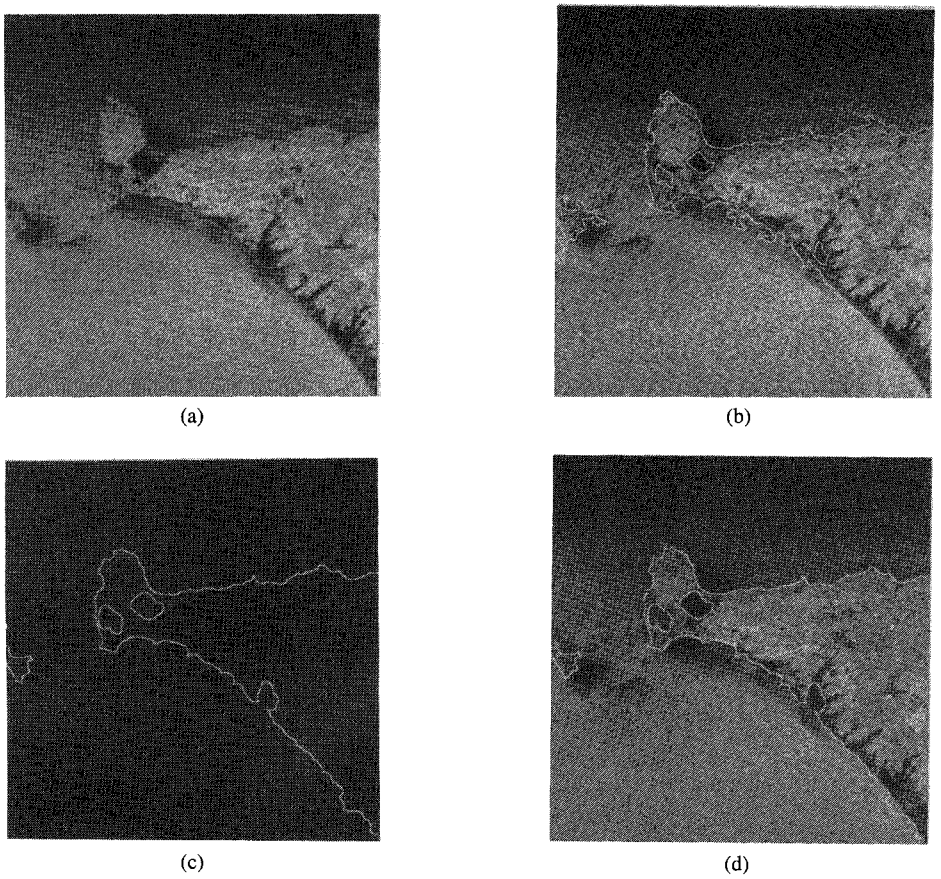


Fig. 6. Coastline detection of a SIR-B image (Martha's Vineyard, MA). (a) SIR-B original. (b) Coastline detection (preliminary). (c) Refined coastline detection. (d) Coastline overlaid on original.

pixels and the other, mostly water pixels. The two images are pseudocolored separately using different color assignments and then combining the images to produce the image shown in Fig. 7. Conventionally, the sea pseudocolor pattern has the blue hue, and the land pseudocolor pattern has the variation from green, brown, to white. The pseudocolor patterns have to be adjusted visually to achieve the best result. The coastline and inlets shown in Fig. 7 are in fact better defined because the pseudocolors were assigned interactively.

VII. Remarks and Discussion

i) In our judgement, the refined coastline is accurate enough for the applications stated in Section I, such as autonomous navigation and geolocation. For geographic mapping, however, further processing is required. This involves tracing narrow inlets, small lakes, bridges, etc. The complexity of the algorithm would be increased greatly. Further refinement can be achieved by searching the pixel neighborhood in the direction normal to the detected coastline. The edge, the gray-level variation, and even the texture should be considered as attributes in determining the true water–land boundaries. If narrow inlets are to be detected, algorithms should be developed to trace them. In essence, the refinement should be guided by the results shown in Fig. 2(e).

ii) The refinement of Section V not only improves the definition of the detected coastline, but also eliminates small fluctuations which yield false detection. In this manner the small off-shore islands and peninsulas are better defined, as demonstrated in the Martha's Vineyard image (left–middle of Fig. 6(c)).

iii) The success of this algorithm depends on the homogeneity of the ocean area in an SAR image, as compared with that of the land area. To a lesser degree it also depends on the contrast between the water and land areas appearing in the image. The crucial step which governs the success or failure of this algorithm is the thresholding of the dilated-edge image as illustrated by Fig. 1(f). In our experiments, coastlines in approximately 75% of the attempted cases were successfully traced. A more complete evaluation of the performance of this algorithm, both in terms of its precision and computational complexity, was outside the scope of this study.

iv) The algorithm developed in this paper uses a chain of many basic image-processing operations, such as the mean filter, Sobel edge operator, thresholding, and histogram computation, to solve a complicated image-segmentation problem. With the currently available image-processing hardware such as VICOM, I^2S, etc., these operations can be performed in real time. The sigma filter and contour-tracing scheme can likewise be efficiently implemented. Hence this algorithm has the potential for near real-time applications.

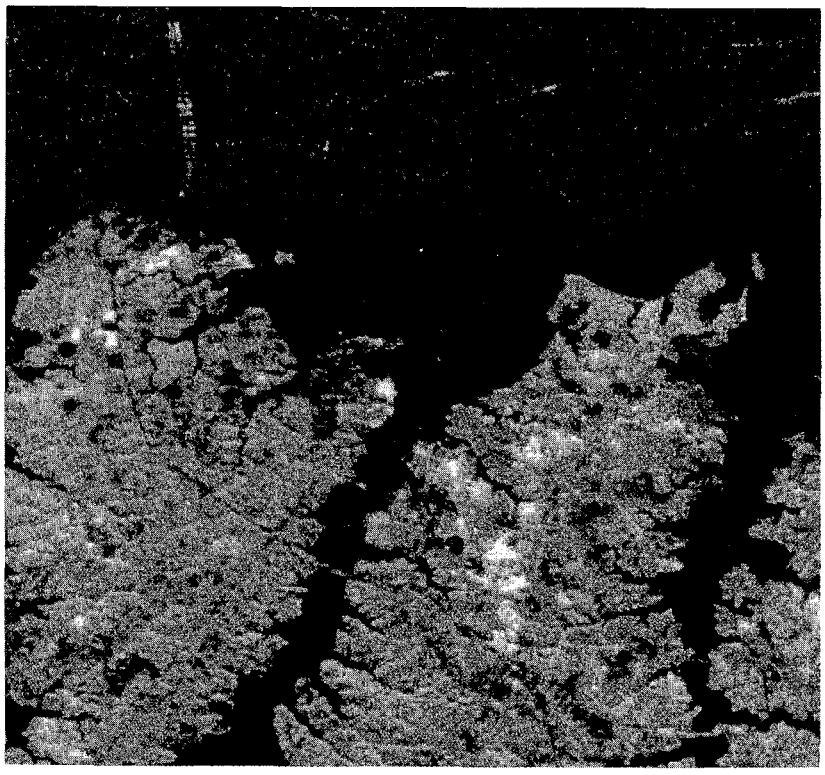


Fig. 7. Seasat SAR Chesapeake Bay Area pseudo-colored by applying coastline detection algorithm (512 × 512 pixels).

VIII. Conclusion

An algorithm has been developed for coastline detection in SAR images. To validate its performance it was applied to typical Seasat SAR and SIR-B images. The algorithm performs reasonably well for geolocation and autonomous navigation. Further improvement in accuracy for the purpose of geographic mapping would require additional processing guided by the global information on the detected coastline. The algorithm is conceptually simple and computationaly efficient and has the potential for achieving real-time digital-processing performance.

References

[1] K. Eldhuset, "Automatic ship and ship wake detection in spaceborne SAR images from coastal regions," in *Proc. IGARSS '88* (Edinburgh, Scotland), Aug. 1988, pp. 1529–1533.

[2] C. Chow and T. Kaneko, "Automatic boundary detection of the left ventricle from cineangiogram," *Comp. Biomed. Res.*, vol. 5, pp. 388–410, 1972.

[3] D. Ballard and C. Brown, *Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[4] J. S. Lee, "A simple speckle smoothing algorithm for Synthetic Aperture Radar images," *IEEE Trans. Syst., Man. Cybern.*, vol. SMC-13, pp. 85–89, 1983.

[5] J. S. Lee, "Speckle analysis and smoothing of Synthetic Aperture Radar images," *Computer Graph. Image Process.*, vol. 17, pp. 24–32, 1981.

[6] V. S. Frost, J. A. Stiles, K. S. Shanmugan, and J. C. Holtzman, "A model of radar images and its application to adaptive digital filtering of multiplicative noise," *IEEE Trans. Patten Anal. Machine Intell.*, vol. PAMI-4, pp. 157–166, Mar. 1982.

[7] J. S. Lee, "Speckle suppression and analysis for Synthetic Aperture Radar images," *Opt. Eng.*, vol. 25, no. 5, pp. 636–643, May 1986.

[8] J. S. Lee and M. Yang, "Threshold selection using estimates from a truncated normal distribution," *IEEE Trans. Syst., Man, Cybern.*, vol. 19, pp. 422–429, Mar. 1989.

[9] W. K. Pratt, *Digital Image Processing*. New York: Wiley, 1978.

[10] R. O. Duda and P. E. Hart, *Pattern Recognition and Scene Analysis*. New York: Wiley, 1973.

[11] A. Rosenfeld, *Picture Processing by Computer*. New York: Academic, 1968.

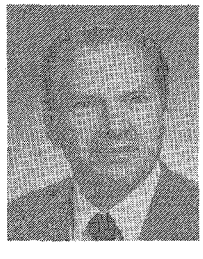
[12] J. S. Lee, "Digital image smoothing and the sigma filter," *Comput. Vision, Graph., Image Process.*, vol. 24, pp. 255–269, 1983.

*

Jong-Sen Lee (S'66–M'69) received the B.S. degree in electrical engineering from the National Cheng Kung University, Tainan, Taiwan, in 1963, and the Ph.D. degree from Harvard University, Cambridge, MA, in 1969.

Since then, he has been with the U.S. Naval Research Laboratory, Washington, DC, where he is a Senior Research Engineer in the Center for Advanced Space Sensing. He is also the Principal Investigator for the remotely sensed signatures program. His research interests cover a wide spectrum of areas: He has been working in the area of control theory, operations research, and radiative transfer. His current interest is in the area of SAR speckle analysis and modeling, digital image processing, and pattern recognition.

*



Igor Jurkevich received the B.S. degree in mathematics from Denison University, Granville, OH, and the M.S. degree in physics, as well as the Ph.D. degree in astronomy, from the University of Pennsylvania, Philadelphia.

He held previous positions with RCA, in the Airborne Fire Control Department, and the Space Sciences Laboratory of the General Electrical Company. His interests at that time were in the areas of radar and electro-optical fire control, celestial mechanics, space-flight dynamics, and satellite payload integration. He was also a Research Associate at the Flower and Cook Observatory and an Adjunct Professor of Astronomy at the University of Pennsylvania. He is presently a Supervisory Operations Research Analyst in the Center for Advanced Space Sensing, Naval Research Laboratory (NRL), Washington, DC. Since 1972, when he came to the NRL, his activities have been in the area of operations research, digital image processing, and synthetic aperture radar signal processing.