Optimized fuzzy cellular automata for synthetic aperture radar image edge detection

Article *in* Journal of Electronic Imaging · February 2018

DOI: 10.1117/1.JEI.27.1.013030

4 authors, including:

Gholamreza Akbarizadeh
Shahid Chamran University of Ahvaz
118 PUBLICATIONS **2,812** CITATIONS
SEE PROFILE

A. Kosarian
Shahid Chamran University of Ahvaz
46 PUBLICATIONS **414** CITATIONS
SEE PROFILE

Kazem Rangzan
Shahid Chamran University of Ahvaz
74 PUBLICATIONS **534** CITATIONS
SEE PROFILE

Journal of Electronic Imaging

JElectronicImaging.org

Optimized fuzzy cellular automata for synthetic aperture radar image edge detection

Mohammad Farbod
Gholamreza Akbarizadeh
Abdolnabi Kosarian
Kazem Rangzan

SPIE. IS&T imaging.org

Mohammad Farbod, Gholamreza Akbarizadeh, Abdolnabi Kosarian, Kazem Rangzan, "Optimized fuzzy cellular automata for synthetic aperture radar image edge detection," *J. Electron. Imaging* **27**(1), 013030 (2018), doi: 10.1117/1.JEI.27.1.013030.

Optimized fuzzy cellular automata for synthetic aperture radar image edge detection

Mohammad Farbod,[a] Gholamreza Akbarizadeh,[a,*] Abdolnabi Kosarian,[a] and Kazem Rangzan[b]
[a]Department of Electrical Engineering, Faculty of Engineering, Shahid Chamran University of Ahvaz, Ahvaz, Iran
[b]Department of Remote Sensing and GIS, Faculty of Earth Sciences, Shahid Chamran University of Ahvaz, Ahvaz, Iran

Abstract. The development of coastline detection has been the subject of several reports. An optimized fuzzy cellular automata (FCA) algorithm for SAR image edge detection, which combines newly defined cellular automata (CAs) and fuzzy rules, is proposed. An extended Moore neighborhood is used for cellular automaton. Twelve custom masks were defined to study the edge angles of the radius two neighborhood of the main pixel. Comparing these edge angles with the radius, one neighborhood of the main pixel is useful when deciding whether or not a pixel is an edge pixel. The model was tested on two sets of images. The first dataset contained optical and simulated SAR images and the second contained an Envisat ASAR image and a ScanSAR image. A 3×3 Lee filter (as a preprocessing phase) was applied to each subimage containing coastlines, and the subimages were then processed using an FCA edge detector. The results were compared with those from a Sobel edge detector, Roberts edge detector, wavelet transform edge detector, and classic CA model. The results showed that the proposed method is more appropriate for edge detection of SAR images when compared with classic methods. The proposed method and wavelet transform edge detector showed good continuity, but the proposed method dealt better with speckle noise effects. [DOI: 10.1117/1.JEI.27.1.013030]

Keywords: fuzzy cellular automata; edge detection; machine learning; synthetic aperture radar.

Paper 170461 received Jun. 7, 2017; accepted for publication Feb. 9, 2018; published online Feb. 27, 2018.

1 Introduction

Synthetic aperture radar (SAR) is a method of simulating a very long antenna by combining echoes received by radar. A synthetic aperture is created by moving a real aperture or antenna to a series of positions along the flight path.[1] SAR images are used in target recognition and tracking, development of military systems,[2] coastline extraction,[3] and other useful applications. Because SAR is an active sensor and uses microwave bands,[4] it offers day/night imaging[5] and can penetrate cloud cover.[6]

One advantage of SAR imaging compared with optical imaging is that it operates independently of weather conditions and daylight.[7] Another advantage is its very high resolution.[8] However, due to the nature of the SAR image formation process, that is, fundamentally different from optical, SAR image information and contents embed a huge amount of noise which is named speckle noise. The speckle phenomenon for SAR imagery can be described as multiplicative noise, with standard deviation equal to pixel reflectivity value. So, one of the important components in SAR images, i.e., the edges, are extremely degraded. Edges provide the most basic and important details of images and contain important information for recognizing objects in images.[9] The Sobel, Canny, and other classic methods have been typically used for edge detection and image segmentation. These algorithms have disadvantages that can make edge detection slightly inaccurate. Selecting an appropriate filtering scale and operator is one of these problems. Another is that the neighborhood of each pixel is not involved in the edge detection.[9] In these methods, the gradient of each pixel is compared to a threshold; if the value is more than the threshold, the pixel is considered to be an edge pixel. The threshold value is usually defined manually, which increases the probability of error.[10] Use of the fuzzy cellular automata (FCA) algorithm is recommended to optimize edge detection.[11] This algorithm offers better filter scaling and involves the neighborhood of each pixel in edge detection.

Detailed structures in SAR images are inherently contaminated by speckle noise,[5] which can make SAR image processing a challenge. Speckle noise is also a major issue in SAR image analysis and can adversely affect coastline images.[3] Development of an efficient and effective automatic technique for SAR image segmentation is, thus, essential.[12] Various models have been proposed for SAR image segmentation, including threshold algorithms,[13] spatial density thresholding,[14] wavelets,[15] support vector machines,[16,17] and neural networks.[18]

Recently, SAR image applications have increased vastly. Edge detection is one of the primary tasks in image processing; thus, SAR image edge detection has been an important part of studying SAR images.[19] There are various applications of SAR image edge detections, so every single method may have good results for some applications and bad results for other applications. Recently, applications such as civilian vehicle edge detection,[20] edge and line extraction of SAR images,[21] constant false alarm rate edge detection,[22] building detection in urban areas,[23] coastline extraction,[24]

*Address all correspondence to: Gholamreza Akbarizadeh, E-mail: g.akbari@scu.ac.ir

1017-9909/2018/$25.00 © 2018 SPIE and IS&T

statistical edge detection in urban areas,[25] and many other applications have been studied.

As mentioned before, every method for SAR image edge detection is useful for a specific application; thus, the method should be optimized for the purpose of the study.

While the problem of edge detection in SAR imagery has been around for nearly three decades, there is still significant research effort in this field. Detailed analysis of FCA for automatic edge detection of SAR imagery has not been examined thus far, although an FCA model can be competitive in terms of accuracy for image edge detection.

SAR sensors installed on ERS satellites have been imaging the ocean surfaces since 1991. Obviously, the safety of shipping lanes makes ocean study more important.[26]

The boundary between land and ocean mass is named as coastline. Studying the coastline characteristics is the basis of measuring land and water resources.[27] Also, coastline detection is crucial in studying the descriptions of coastal areas, and automated navigation, geographical evolution studies, and cartography[28] require information about the coastline.[29]

In former studies on coastline, digital contour maps were used to simulate SAR images. By comparing the simulated SAR image with a real SAR image, the coastline was determined.[30]

The FCA algorithm is a combination of a cellular automata (CAs) model and fuzzy rule theory. In this algorithm, the cell state of the CA is changed to a fuzzy state and evolution rules into fuzzy rules.[31] The proposed model was applied to three sets of images. The first dataset contained optical and simulated SAR images, the second contained the ground truth of SAR images, which are noiseless, and the third dataset contained real SAR images that were primarily coastline images. The results were then compared with the results of classic edge detection algorithms and newer methods, such as the wavelet transform.[24]

This paper presents a method to extract edges from SAR images and detect coastlines inspired by human vision perception. The human vision system can detect edges, especially coastlines, based on neighborhood characteristics despite much noise in images. Since the edges in SAR images embed a huge amount of noise, the proposed method uses the edge neighborhood characteristics for edge detection in SAR images to conform to the human vision. The proposed method reduces the speckle effect very well and extracts useful edges such as coastlines in order to ease further analysis. The proposed algorithm is composed of efficient fuzzy rules of optimized cellular automata (OCA) that it is suitable for edge detection of SAR images in the presence of speckle noise. It is the first time that effective fuzzy rules following the proposed approach are developed, which are very efficient for "coastline detection" in SAR images. A classic CA edge detector with extended Moore neighborhood is the main idea of the algorithm. Twelve masks are defined that use the radius one and two of the main pixel's neighborhood. These masks are used to study the possibility of the main pixel to be a part of an edge (in the neighborhood of the pixel) or not. Fuzzy rules are defined to enhance the results of edge detection.

Section 2 describes the principles behind the proposed approach and the fuzzy rules previously applied to optical images. Section 3 describes the proposed method. Section 4 provides the experimental results and a detailed description of each step of coastline detection. Section 5 studies the experimental aspects and the conclusion follows in Sec. 6.

2 Background

2.1 *Cellular Automata*

CAs are discrete dynamic systems, in which behavior is based purely on local communication.[32] In CA, space is defined as a lattice composed of cells. Time progresses discretely and its rules are global. At each step of computation, each cell can obtain a new status by considering its neighbors. CA rules determine the effect of the neighboring cells on the CA cell.[33]

CA is a 7-tuple $\{Q, d, v, \in, \Delta, \delta, \lambda\}$ for which Q is a set of states, in which each cell can adapt and d specifies the dimensions of the cell space. If $d = 2$, there will be a 2-D CA. Array ν (for each cell in the CA) specifies $(k + 1)$ neighbors directly communicating with the cell, in which $\in$ is the input script and Δ is the output script. The symbol Δ is the transfer function in the form of $\delta: (Q\times \in^n)^{k+1} \rightarrow Q_{(t+1)}$. Accordingly, the state of each cell depends on the state and input memory of all neighboring cells in the current step. The variable n denotes the number of input and output memories of each cell, and λ is the relation converter and is a finite subset of $(Q\times \in^n)^{k+1} \times \Delta^n$. The converter specifies the output memory of the cell according to the state and input memory of its neighboring cells. Each cell writes the CA in the same memories from which it reads (as a result, $\in = \Delta$).

Different types of neighborhoods can be used in CA, depending on the purpose of the study. The most common neighborhoods are the Von Neumann, Moore, Smith, and Cole[10] These are radius one neighborhoods. In image processing, the pixels nearest to the main pixel are radius one neighbors. Figure 1 shows different types of neighborhoods used in CA. As shown, the Von Neumann neighborhood defines vertical and horizontal neighbors, the Moore

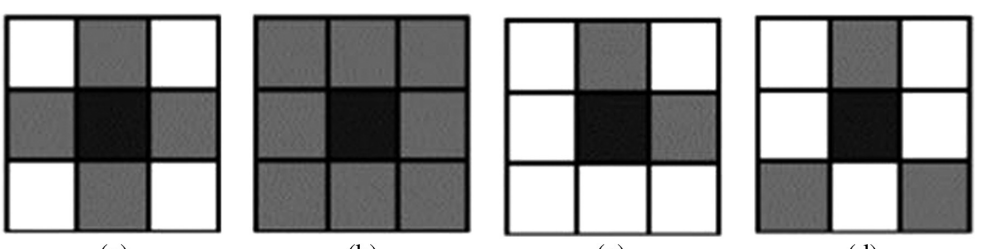

Fig. 1 Types of neighborhoods used in CA: (a) Von Neumann, (b) Moore, (c) Smith, and (d) Cole.

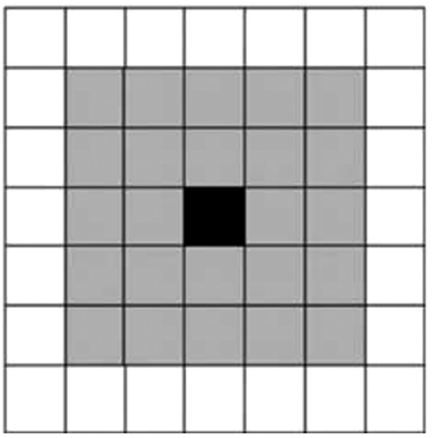

Fig. 2 Extended Moore neighborhood.

neighborhood defines all eight neighbors, the Smith neighborhood defines two neighbors, and the Cole neighborhood defines three neighbors.

Custom models for CA can be defined for an extended Moore used for radius one and two main pixel neighbors.[34] Figure 2 shows the extended Moore neighborhood. Many other custom models can be defined for different situations. In edge detection, the extended Moore model is most appropriate because it can be used to define more neighbors of the main pixel. An edge is a continuous line (except at the end of the edge), and the neighbors of an edge have a high possibility of being an edge pixel.

2.2 *Fuzzy Cellular Automata*

A CA is proposed that uses fuzzy logic to process noncrisp data, in which fuzzy and noncrisp values are used in the cells and fuzzy values are used instead of crisp values. The threshold in a CA is crisp, which makes the results poor. Many discontinuities appear in the edges because of the crisp values. Fuzzy rules enhance the results and improve the continuity of the edge, but the disadvantage is that a few noisy spots may become bolder.

3 Proposed Method

In this paper, a three-step algorithm is proposed for edge detection of SAR images. In the first step, a CA mask with an extended Moore neighborhood is applied to the SAR image. This step compares the difference between the gray levels of the main pixel and neighboring pixels to decide whether or not the pixel should be an edge pixel. In the second step, 12 masks are defined for the radius one and two main pixel neighborhoods. These masks determine the possibility of the main pixel being part of an edge (in the neighborhood of the pixel). In step three, fuzzy rules are defined to enhance the results of edge detection.

3.1 *Step One: Gray-Level Comparison*

In the first step, the radius one and two neighborhoods of the pixels are determined. The difference between the pixel gray level and all radius one and two neighbors is computed as

$$O_i = \max_{j=1:8} |P_9 - P_j|, \tag{1}$$

where P is a defined mask, in which P_9 is its center cell. P_j shows the gray level of pixel j. In this step, pixel O_i is studied. Mask P is applied to all of the second radius neighborhoods of O_1 (O_2 to O_8) as shown in Fig. 4(b).

If the maximum difference between gray levels is more than the defined threshold, T_1, the pixel is marked as an edge pixel. If the difference is less than threshold, T_2, the pixel is marked as a nonedge pixel. If the difference is between T_1 and T_2, the pixel radius two neighborhood is determined. If the neighbors of the pixel are known to be edge pixels (step two) and the edge angle of the main pixel and the neighbor pixel are the same, then the main pixel is marked as an edge pixel; otherwise, it is marked as a nonedge pixel. There is a challenge to select the correct values for T_1 and T_2.

SAR images have many different features with different specifications. There are various algorithms for feature extraction in SAR images. In coastline detection, the important feature to study is the boundary between the land and the water. Two threshold values should be extracted for the optimized fuzzy cellular automata (OFCA) method: the low threshold and the high threshold. There are various ways to estimate the best thresholds. One way is manually changing the threshold levels to reach the best results. The other way is to use a feature extraction method for SAR images to find the best values for the thresholds. Recently, various methods were used for feature extraction, such as maximum likelihood (ML). Tirandaz and Akbarizadeh[2] introduced a new algorithm for extracting SAR image boundaries. They developed an estimation algorithm named kurtosis curvelet energy (KCE) to estimate the best values for the parameters and proved that the algorithm is even stronger than ML for SAR images. The KCE algorithm computes two parameters named "α" and "β" which are the features of boundaries in SAR images. So, these two parameters are set as thresholds to give to the OFCA method as input thresholds (T_1 and T_2). The two main parameters α and β of KCE in the OFCA method are shown as below[2]

$$\alpha = \sqrt{\frac{\Gamma\left(\frac{7}{\beta}\right) \times \mathrm{m}^{(4)}_{|C_\phi|^2}}{\Gamma\left(\frac{9}{\beta}\right) \times \mathrm{m}^{(3)}_{|C_\phi|^2}}}, \tag{2}$$

$$\beta = F^{-1}(\text{Kurtosis}) = F^{-1}\left\{\frac{\mathrm{K}^{(4)}_{|C_\phi|^2}}{[\mathrm{K}^{(2)}_{|C_\phi|^2}]^2}\right\}, \tag{3}$$

where $\Gamma(.)$ is the gamma function, $m^{(3)}_{|C_\phi|^2}$ and $m^{(4)}_{|C_\phi|^2}$ are the third- and the fourth-order moments of random variable $|C_\phi|^2$, respectively, F^{-1} is the inverse function of the kurtosis of random variable $|C_\phi|^2$, and $\mathrm{K}^{(2)}_{|C_\phi|^2}$ and $\mathrm{K}^{(4)}_{|C_\phi|^2}$ are the second- and the fourth-order cumulant of random variable $|C_\phi|^2$, respectively.[2] In this paper, we used KCE algorithm for exactly estimating these two parameters.[2]

Figure 3 shows the Moore and the extended Moore neighborhoods that are defined to determine the pixel edge possibility.

3.2 *Step Two: Angle Comparison*

In this step, 12 masks are defined to determine the possibility of a pixel being an edge pixel. Four degrees of an edge

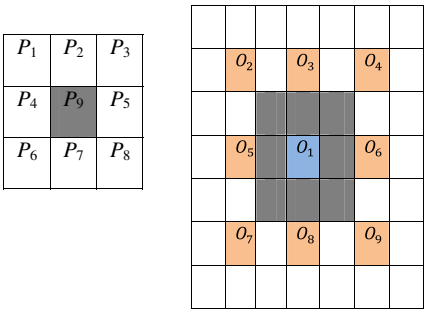

Fig. 3 (a) Moore neighborhood for radius one neighbors and (b) extended Moore neighborhood for radius two neighbors.

(0 deg, 45 deg, 90 deg, and 135 deg) have been defined and are shown in Fig. 4.

In step one, if the difference between the gray levels of the main pixel and the neighborhood is 50 to 130 (between $T_1 = 50$ and $T_2 = 130$), step two will be applied, in which the radius two neighbors of the main pixel [O_2 to O_9 in Fig. 3(b)] are determined. If the angle of the edge of the radius two neighbors is the same as for the radius one neighbor, the possibility of the main pixel being an edge pixel increases; otherwise, it is probably a noisy pixel that is a nonedge pixel.

Figure 5 shows the edge possibility of neighboring pixels. Neighbor O_2 in Fig. 3(b) is to the top left of the main pixel. If the main pixel radius one neighbors have the same pixels as a 135-deg mask, the possibility of the main pixel being an edge pixel increases because it has the same edge pattern as the top-left neighbor [Fig. 5(a)]. If not, it is probably a noisy pixel [Fig. 5(b)] and is marked as a nonedge pixel. Step one is denoted as OCA in this paper.

The pseudocode of the OCA is shown in Table 1.

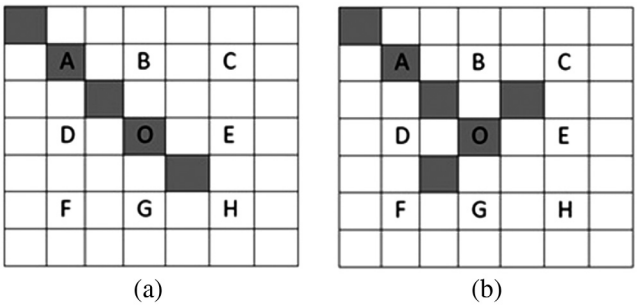

Fig. 5 Edge possibility of neighbor pixels: (a) edge pixel and (b) nonedge pixel.

Table 1 Pseudocode of OCA algorithm.

Step	Operation
1	Compute the difference in gray levels of main pixel and radius one and two neighborhood pixels [Eq. (1)].
2	Determine edge possibility of main pixel by comparing neighbor and main pixel edge angles (Fig. 6).
3	If $O_1 > 130$, then pixel is marked as an edge pixel. If $130 > O_1 > 50$ and one radius two neighbors and main pixel mask have the same angle, then pixel is marked as an edge pixel; otherwise, it is marked as a nonedge pixel.

3.3 *Applying Fuzzy Rules*

There are disadvantages to using a crisp threshold in CA because the edge loses its continuity in some parts. To solve this problem, the proposed fuzzy inference system is used for better edge detection.

The threshold in step one is now defined using fuzzy rules, then step two is applied to decide whether the pixel

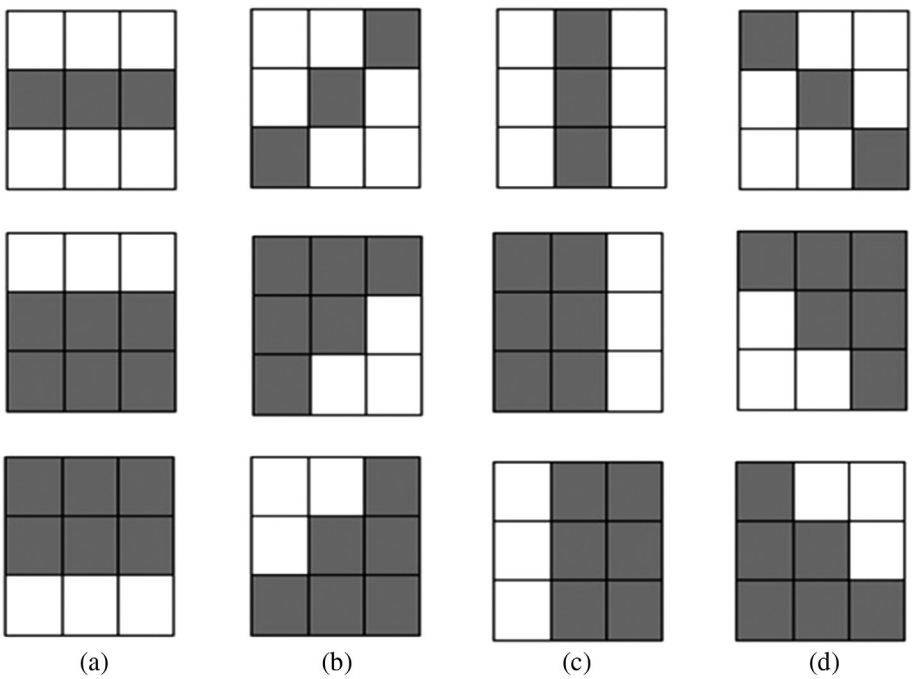

Fig. 4 Defined masks: (a) 0-deg masks, (b) 45-deg masks, (c) 90-deg masks, and (d) 135-deg masks.

is an edge or a nonedge pixel. The fuzzy system inference used in this paper is based on Mamdani.[35] The "AND" operator is the Zade min operator. The "OR" operator is the Zade max operator. For defuzzification, the centroid method is applied. The fuzzy inputs are the differences between the gray levels of the main pixel and the pixel's neighbors (O_1 to O_9).

To fuzzify the inputs, three linguistic variables are defined, which are constant for all inputs. The first linguistic variable is denoted as "low" and is a Z function [Eq. (4)] with parameters $a = 20$ and $b = 100$. The second linguistic variable is denoted as "mid" and is a Gauss2MF function [Eq. (5)] with parameters $\sigma_1 = 20$, $c_1 = 67.77$, $\sigma_2 = 10$, and $c_2 = 87.77$. The third linguistic variable is denoted as "high" and is a Gauss2MF function [Eq. (5)] with parameters $\sigma_1 = 15$, $c_1 = 100$, $\sigma_2 = 20$, and $c_2 = 30$. The Z and Gauss2MF functions are as follows:

$$Z(x;a,b) = \begin{cases} 1, & x \le a \\ 1 - 2\left(\frac{x-a}{b-a}\right)^2, & a \le x \le \frac{a+b}{2} \\ 2\left(\frac{x-a}{b-a}\right)^2, & \frac{a+b}{2} \le x \le b \\ 0, & b \le x \end{cases}, \tag{4}$$

$$\text{Gauss2MF}(x;\sigma_1,c_1,\sigma_2,c_2) = e^{\frac{-(x-c_1)^2}{2\sigma_1^2}} + e^{\frac{-(x-c_2)^2}{2\sigma_2^2}}, \tag{5}$$

where $\text{Guass2MF}(x;\sigma_1,c_1,\sigma_2,c_2)$ is a combination of two $(x;\sigma,c)$ parameters. The first function, specified by σ_1 and c_1, determines the shape of the left-most curve. The second function, specified by σ_2 and c_2, determines the shape of the right-most curve. Whenever $c_1 < c_2$, the Gauss2MF function reaches a maximum value of 1; otherwise, the maximum value is <1.

Table 2 Pseudocode of OFCA algorithm.

Step	Operation
1	Compute the difference in gray levels of main pixel and radius one and two neighborhood pixels [Eq. (1)].
2	Determine edge possibility of main pixel by comparing neighbor and main pixel edge angles (Fig. 6).
3	Use fuzzy inference system (Sec. 3.3).
4	If fuzzy output is >0 and one radius two neighbor and the main pixel mask have the same angle, then the pixel is marked as an edge pixel; otherwise, the pixel is marked as a nonedge pixel.

The output of the fuzzy system is the possibility of being an edge pixel and is denoted as "edge." Three linguistic variables are defined for the output. The first linguistic variable is denoted as "EL" and is a Z function [Eq. (4)] with parameters $a = 0.1$ and $b = 0.4$. The second linguistic variable is denoted as "EM" and is a Gauss2MF function [Eq. (5)] with parameters $\sigma_1 = 0.1$, $c_1 = 0.3$, $\sigma_2 = 0.1$, and $c_2 = 0.5$. The third linguistic variable is denoted as "EH" and is a Gauss2MF function [Eq. (5)] with parameters $\sigma_1 = 0.1$, $c_1 = 0.7$, $\sigma_2 = 0.2$, and $c_2 = 1$. The fuzzy rules are defined using AND and OR operations as below:

1. If O_1 is high, then edge is EH (1).
2. If O_1 is not high, O_2 is high, O_3 is high, O_4 is high, O_5 is high, O_6 is high, O_7 is high, O_8 is high, or O_9 is high, then edge is EH (0.75).
3. If O_1 is low, O_2 is high, O_3 is high, O_4 is high, O_5 is high, O_6 is high, O_7 is high, O_8 is high, and O_9 is high, then edge is EM (0.75).
4. If O_1 is low, O_2 is high, O_3 is high, O_4 is high, O_5 is high, O_6 is mid, O_7 is high, O_8 is high, and O_9 is high, then edge is EH (0.75).
5. If O_1 is mid, O_2 is high, O_3 is high, O_4 is high, O_5 is high, O_6 is mid, O_7 is mid, O_8 is mid, and O_9 is mid, then edge is EM (0.5).
6. If O_1 is mid, O_2 is mid, O_3 is mid, O_4 is mid, O_5 is mid, O_6 is high, O_7 is high, O_8 is high, and O_9 is high, then edge is EM (0.5).
7. If O_1 is high, O_2 is low, O_3 is low, O_4 is low, O_5 is low, O_6 is low, O_7 is low, O_8 is low, and O_9 is low, then edge is EL (1).
8. If O_1 is high, O_2 is mid, O_3 is mid, O_4 is mid, O_5 is mid, O_6 is mid, O_7 is mid, O_8 is mid, and O_9 is mid, then edge is EM (1).

The proposed method is called OFCA. The pseudocode of the OFCA algorithm is as shown in Table 2.

Figure 6 shows a block diagram of the whole steps of the proposed method. Each step was described in the previous sections.

To briefly review the algorithm, the steps of the proposed method are shown in the block diagram of Fig. 6. Initially, the input image (A) is scanned by the system. After that, the extended Moore neighborhood mask is applied to the image (A') in MATLAB® to analyze the state of each pixel (edge or nonedge). The outputs of this step (O_i and P_i) are the difference between the gray levels of the main pixel and the neighboring pixels of radius one (P_i) and radius two (O_i). After computing the mentioned gray-level differences, the result of this step (T) is compared with the defined thresholds

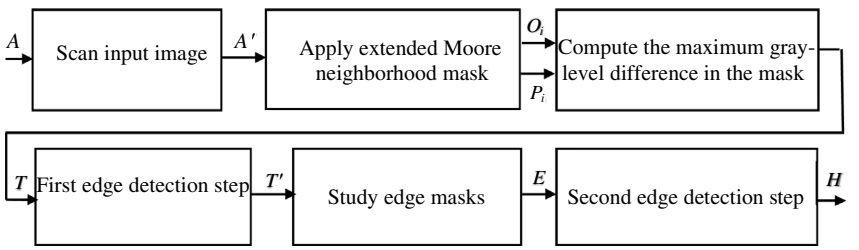

Fig. 6 Block diagram of the proposed method.

(T_1 and T_2) to study the first step of edge detection. If the value is among the defined thresholds (T'), the edge masks (Sec. 3.2) are studied to find edge angles similar to the edge masks in the neighborhood of the main pixel. In the final step, if the edge angles are similar (E), the fuzzy system decides whether to mark the pixel as an edge or nonedge. If the output of the fuzzy system (H) was >0, the pixel is marked as an edge pixel; otherwise, it is a nonedge pixel.

4 Results and Discussion

The proposed approach was applied to two datasets. The first contained optical and simulated SAR images and the second contained an Envisat ASAR and a ScanSAR image. The datasets were processed using the proposed FCA edge detector, and the results were compared with those of classic edge detection methods, such as Sobel,[36] Roberts,[37] CA,[38] Prewitt,[39] Canny–Deriche,[40] and an edge-enhancement algorithm based on the wavelet transform for automatic edge detection in SAR images.[24] The CA works on the gray level of the pixels, so intensity SAR images were used. A 3×3 Lee filter was applied to the images before processing to reduce the noise effects.

4.1 *Edge Detection of Optical Images and Simulated Synthetic Aperture Radar Images*

The ground truth of the Ku-band SAR image of China Lake airport in California was examined. The ground truth is a noise-free image that gives similar edge detection results to optical images. The classic methods produced better and smoother results because the algorithm was optimized for noisy SAR images and lost some details in the optical images. The results are shown in Fig. 7. As seen, the OFCA showed better continuity (especially at the top-left part of the image) than the OCA.

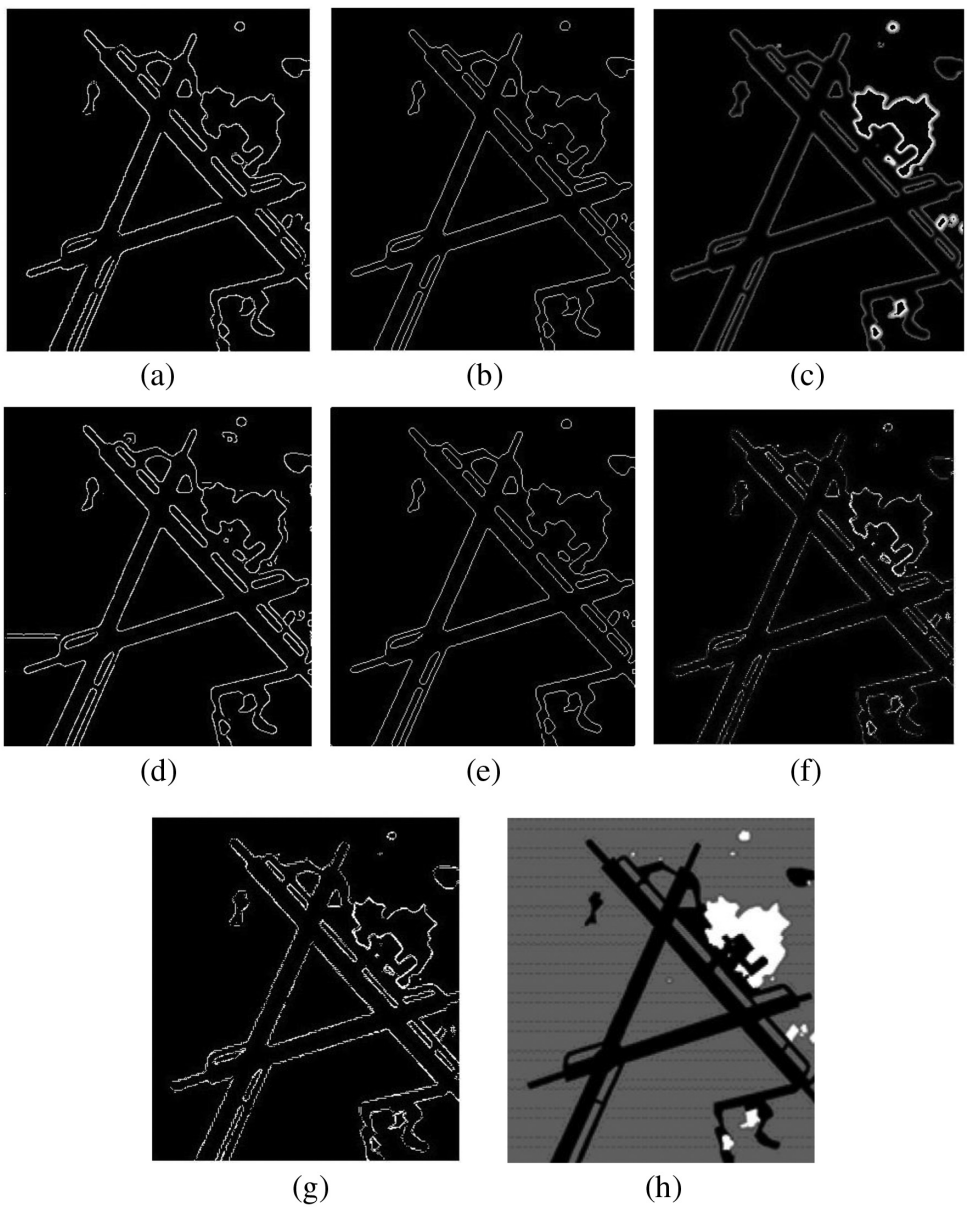

Fig. 7 China Lake airport edge detection results: (a) Sobel, (b) Roberts, (c) classic CA, (d) Canny, (e) Prewitt, (f) OCA, (g) OFCA, and (h) ground truth of the real SAR image.

There are several distributions for simulated SAR images. The product of the average of three multiplying speckle noise, Gaussian noise, and the salt-and-pepper noise in the ground truth is one of the methods to create a three-look simulated SAR image.[41] Averaging three gamma-distributed realizations is another way to generate three-look simulated SAR images.[12]

Figure 8 is a simulated SAR image. A two-look noisy image was generated by averaging two independent realizations of speckle. In other words, the two-look simulated SAR image was generated by averaging the product of two random speckles in the ground truth image. The Lee filter in conjunction with the Sobel and Roberts methods produced good results, but the classic CA struggled. Classic CA depends only on the neighbor pixels, and the speckle noise had a deleterious effect on the results. The OFCA method also provided good results.

Table 3 shows the numerical error rates of each method for Fig. 8.

Recall is the true edge pixels, which have been detected by each method. For example, if the ground truth image has 10 edge pixels and the method has detected 8 of them correctly, the recall would be 0.8.

Precision is the percent of correct edge pixels detected by each method. For example, if a method detects 10 pixels as edge pixels and the correct amount of edges is 8, the precision would be 0.8.

As can be seen from Table 3, Roberts, Canny, classic CA, and OFCA have good recall values. The Canny and classic CA have marked noise pixels as edge pixels, so their precision value is very low. The OFCA method deals very well with the noise and has higher precision, but loses the continuity in some parts, so the recall is not perfect.

A more complicated simulated SAR coastline image was created using random $+1/-1$ displacement along

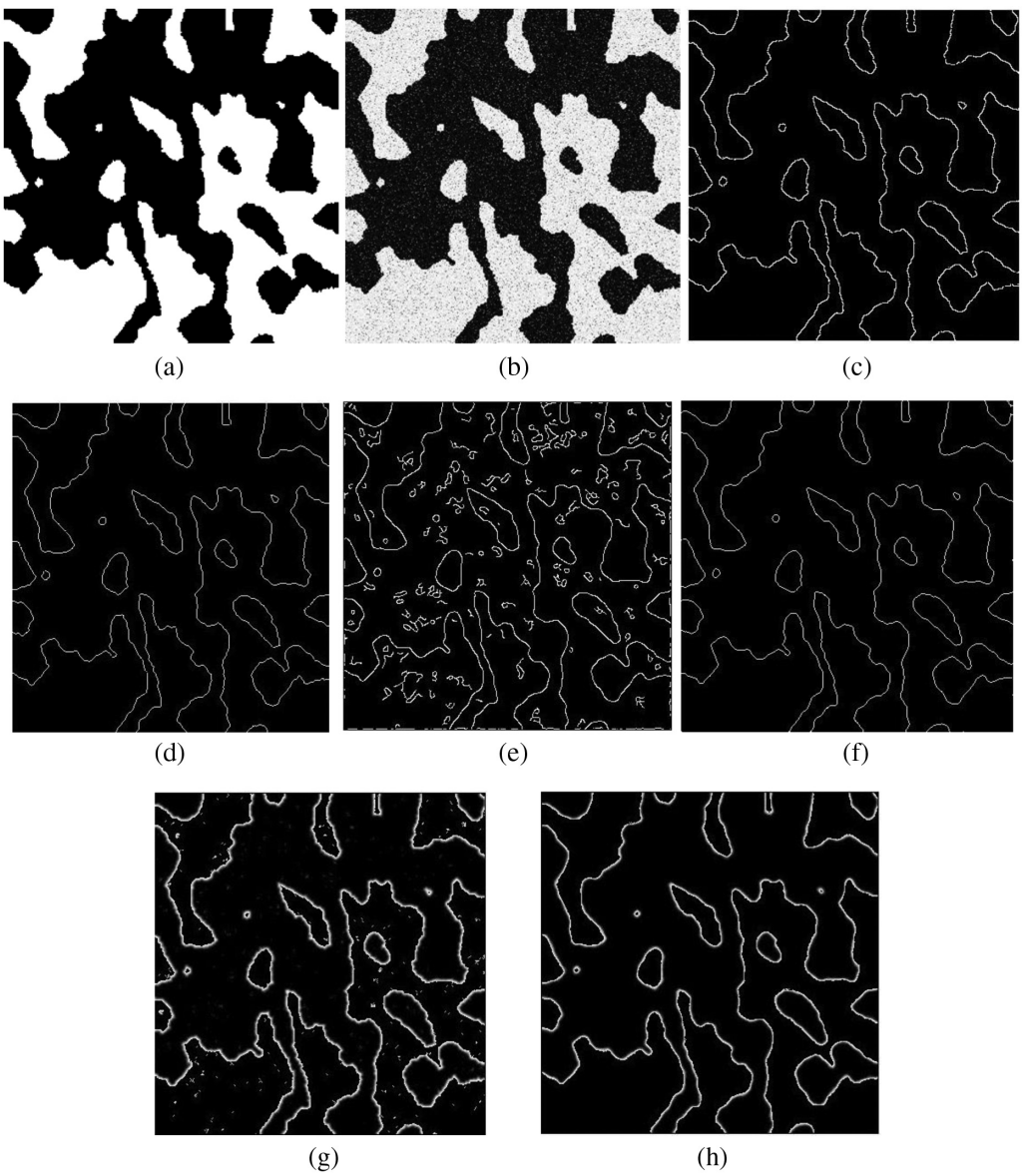

Fig. 8 Simulated SAR image: (a) noiseless image, (b) noisy image, (c) Sobel edge detection, (d) Roberts edge detection, (e) Canny edge detection, (f) Prewitt edge detection, (g) classic CA, and (h) OFCA edge detection.

Table 3 Numerical error rates for Fig. 8.

Edge detection method	PSNR (dB)	MSE	Recall	Precision	*F*-measure
Sobel	36.9590999	13.20	0.5462	0.5787	0.5620
Roberts	36.9545731	13.21	0.7151	0.6967	0.7058
Canny	31.3086381	48.49	0.6820	0.3808	0.4888
Prewitt	36.5605792	14.47	0.5484	0.5805	0.5640
Classic CA	34.1242292	25.35	0.7236	0.4246	0.5352
OFCA	34.1705915	25.09	0.6364	0.7858	0.7032

the vertical direction.[24] This image is shown in Fig. 9. The OCA algorithm removed the noise effects very well but showed discrete parts [Fig. 9(g)]. The OFCA shows very good continuity but added a few noise spots [Fig. 9(h)]. In comparison with the classic methods, OFCA produced much better results. In comparison with the wavelet transform,[24] which is shown in Fig. 9(h), OFCA produced similar results.

Table 4 shows the numerical error rates of each method for Fig. 9.

As can be seen from Table 4, the classic methods have medium recall but very low precision because they detect many noise pixels as edge pixels. The wavelet transform has excellent recall but a low precision. The wavelet transform cannot detect the noise pixels well enough. The OFCA method has a high recall (lower than wavelet due to losing continuity in some areas) and a very high precision. The OFCA method eliminates noise pixels better than the other methods.

4.2 *Edge Detection of Real Synthetic Aperture Radar Images*

The proposed model was also tested on a dataset containing Envisat-ASAR images (ENVISAT ASAR in IM mode acquired on July 24, 2003) as shown in Fig. 10. The purpose was to extract the coastline of the image. Classic methods struggled with this section because of the high noise of the image. They extracted the edges well but could not adequately remove the noise. OCA reduced the noise well but lost continuity in some parts. OFCA enhanced the edge (coastline) well but added a few noisy spots (Fig. 10).

The results were compared with an edge-enhancement algorithm based on the wavelet transform.[24] The wavelet transform showed slightly better continuity but contained random pale noisy effects. OFCA removed the noisy effects much better.

Table 5 shows the numerical error rates of each method for Fig. 10.

As can be seen from Table 5, the classic methods have medium recall and very low precision. The Canny method detects many noise pixels as edge pixels, thus it has the least precision value. The wavelet transform has the best recall due to its perfect continuity, but the noise pixels that are detected as edge pixels result in a lower precision

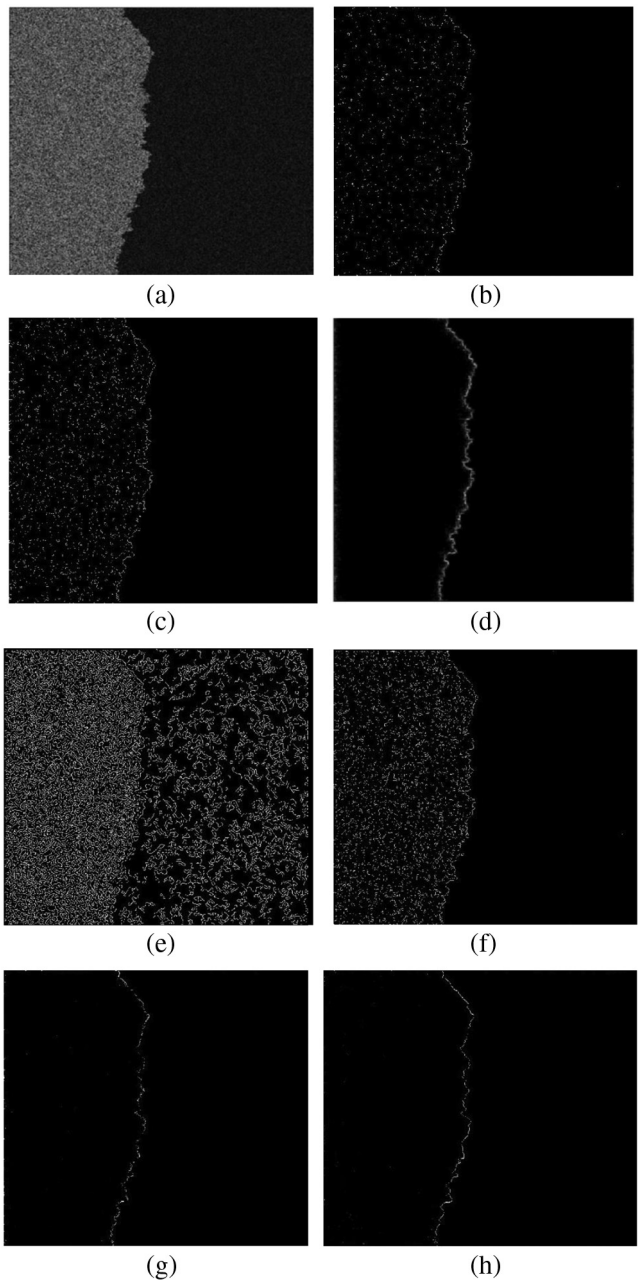

Fig. 9 Simulated SAR image: (a) original image, (b) Roberts edge detection, (c) Sobel edge detection, (d) wavelet transform edge detection, (e) Canny edge detection, (f) Prewitt edge detection, (g) OCA edge detection, and (h) OFCA edge detection.

for the method. The OFCA method has a good recall value and a very high precision value.

The other real SAR image employed was the ScanSAR scene collected on October 23, 2014, which includes part of the Alboran Sea from San Roque, Cádiz, Spain, to Benalmádena, Málaga, Spain (36°25′56.90″N). The image contains a wavy sea and has HH polarization. Again, OFCA produced the best results in comparison with the other methods (Fig. 11). The waves are natural edges in the images. As seen, all methods extracted the waves as edges. The classic methods produced confused results. OCA lacked some continuity, but OFCA produced good results. In some cases, the purpose was to extract only the

Table 4 Numerical error rates for Fig. 9.

Edge detection method	PSNR (dB)	MSE	Recall	Precision	*F*-measure
Sobel	34.7723706	21.84	0.5270	0.0738	0.1295
Roberts	37.4876480	11.69	0.5140	0.0792	0.1373
Wavelet transform	37.2442259	12.36	0.9640	0.4893	0.6491
Canny	28.0612780	102.41	0.5855	0.03318	0.0628
Prewitt	33.7510340	27.63	0.5284	0.0751	0.1316
OFCA	53.3167163	0.31	0.7353	0.9945	0.8455

coastline, not all edges (such as waves). For this, a CA Moore neighborhood mask was applied to the image. After the OFCA algorithm was applied, the mask determined all pixels again. Any pixel that contained no white pixels in the radius one neighborhood was removed. Afterward, all discrete pixels were removed and only continuous pixels (such as coastline pixels) remained.

Table 6 shows the numerical error rates of each method for Fig. 11.

As can be seen from Table 6, the classic methods have medium recall and very low precision. OCA and OFCA methods have higher recall values, but they have low precision values. The OFCA (customized for coastline detection) loses continuity in some areas and has a lower recall value, but eliminates all of the noncoastline pixels and has a high precision.

5 Experimental Aspects

For studying experimental aspects of the research, a ground truth of each image was produced to compare the results of

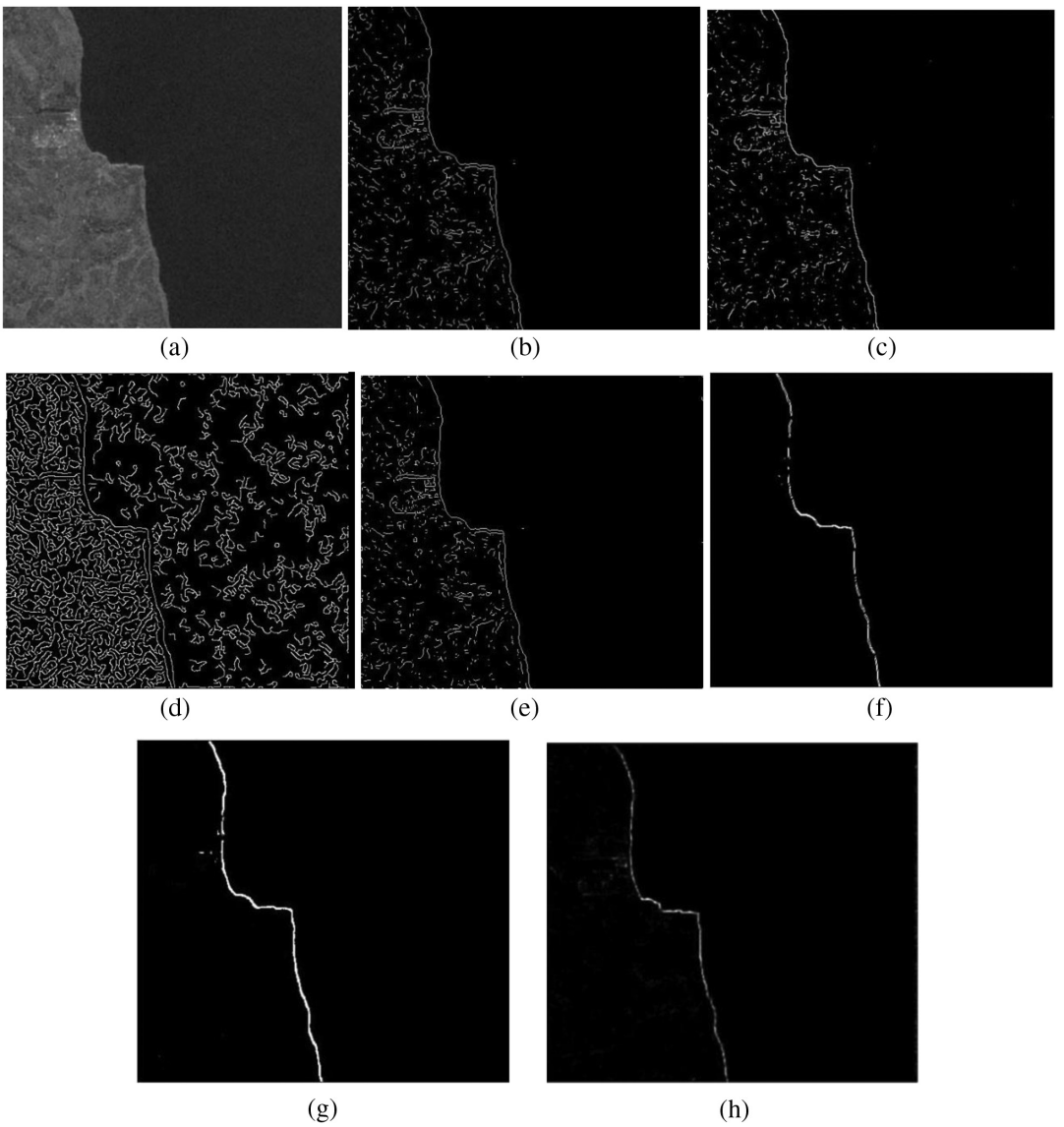

Fig. 10 Real SAR image coastline extraction: (a) Envisat-ASAR image, (b) Sobel edge detection, (c) Roberts edge detection, (d) Canny edge detection, (e) Prewitt edge detection, (f) OCA edge detection, (g) the proposed OFCA edge detection, and (h) wavelet transform edge detection.

Table 5 Numerical error rates for Fig. 10.

Edge detection method	PSNR (dB)	MSE	Recall	Precision	*F*-measure
Sobel	36.4638850	14.79	0.5208	0.1316	0.2101
Roberts	36.9472342	13.24	0.5430	0.1440	0.2276
Canny	28.9506500	83.45	0.5780	0.04108	0.0767
Prewitt	36.7301868	13.91	0.5170	0.1356	0.2149
Wavelet transform	41.6772752	4.45	0.8930	0.6548	0.7555
OFCA	56.6206513	0.14	0.7736	0.8413	0.8060

the method to the ideal result. Peak signal-to-noise ratio (PSNR) and mean squared error (MSE) of the results were calculated and were shown in Tables 3–6.

Precision is the number of correct results divided by the number of all returned results.

Recall is the number of correct results divided by the number of results that should have been returned.

A measure that combines precision and recall is the harmonic mean of precision and recall named as traditional

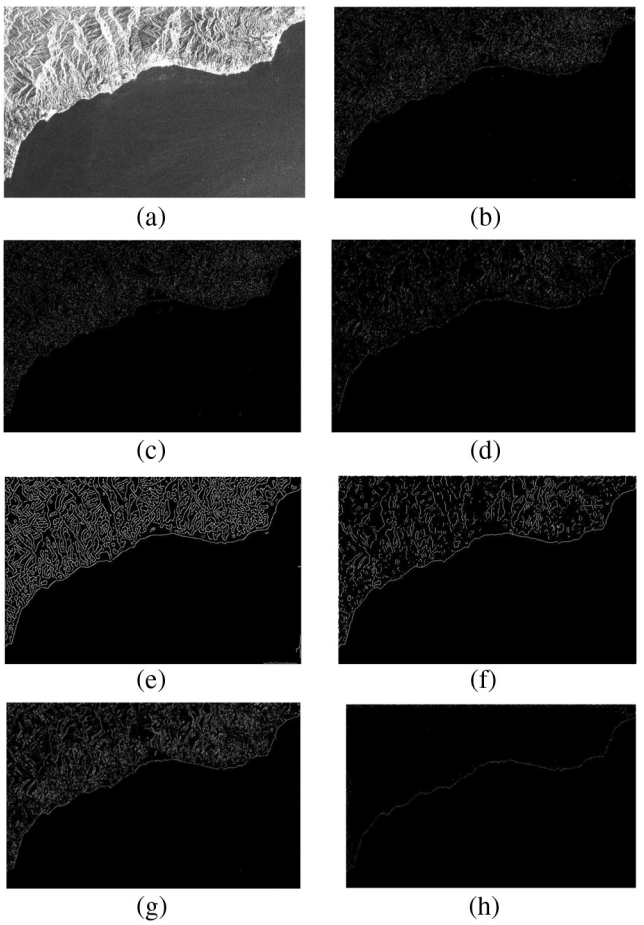

Fig. 11 (a) Alboran Sea, (b) Sobel edge detection, (c) Roberts edge detection, (d) OCA edge detection, (e) Canny edge detection, (f) Prewitt edge detection, (g) OFCA edge detection, and (h) the proposed OFCA for coastline detection.

Table 6 Numerical error rates for Fig. 11.

Edge detection method	PSNR (dB)	MSE	Recall	Precision	*F*-measure
Sobel	33.2637973	30.91	0.5213	0.0934	0.1585
Roberts	33.2620859	30.92	0.5588	0.1024	0.1731
Canny	30.0230693	65.19	0.5809	0.07921	0.1394
Prewitt	32.3606710	38.06	0.5203	0.0946	0.1601
OCA	35.6292953	17.93	0.6325	0.0977	0.1693
OFCA	32.4416196	37.35	0.6648	0.0936	0.1641
OFCA for coastline	46.3039639	1.53	0.6487	0.7880	0.7115

Table 7 Execution time of OFCA method.

Figures	Image resolution	Execution time (s)
8	372×372	337
9	775×707	1154
10	464×426	403
11	623×395	526

F-measure or balanced *F*-score. The traditional *F*-measure or balanced *F*-score is defined as below

$$F = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{6}$$

The numeric values of precision, recall, and *F*-measure are also shown in Tables 3–6.

All of the processes were done with the MATLAB® software. The characteristics of system hardware are as below

- Installed memory (Ram): 6 GB.
- Processor: Intel® Core™ i7-2630QM CPU @ 2 GHz.

The time of the process is mostly based on the resolution of the input images. For each pixel, the first and second radius neighborhoods are studied in each step of the method. The fuzzy inference system makes the process a little more complicated but gives better edge detection results. Table 7 shows the relation between the image's resolution and the process time.

6 Conclusion

This paper proposed an algorithm called OFCA for edge detection of SAR images. A custom CA mask was defined for edge detection. The result was an edge detection method. Fuzzy rules were applied to the system to enhance the edge detection results, and the results were compared with both classic edge detection and newer methods. The results show that the proposed OFCA classifier, as an automated

algorithm for coastline detection, performed better in situations where other models experienced poor accuracy.

Acknowledgments

The work described in this paper was supported by the Shahid Chamran University of Ahvaz (SCU), Ahvaz, Iran, as an MSc thesis under Grant No. 96/3/02/16670. The authors would like to thank the SCU for financial support. Also, the authors would like to thank the European Space Agency for providing ENVISAT images. Dr. Akbarizadeh reports grants from SCU, during the conduct of the study.

References

1. F. Del Frate, F. Pacifici, and D. Solimini, "Monitoring urban land cover in Rome, Italy, and its changes by single-polarization multitemporal SAR images," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **1**(2), 87–97 (2008).
2. Z. Tirandaz and G. Akbarizadeh, "A two-phase algorithm based on kurtosis curvelet energy and unsupervised spectral regression for segmentation of SAR images," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **9**(3), 1244–1264 (2016).
3. M. Modava and G. Akbarizadeh, "Coastline extraction from SAR images using spatial fuzzy clustering and the active contour method," *Int. J. Remote Sens.* **38**(2), 355–370 (2017).
4. F. Del Frate et al., "A novel multi-band SAR data technique for fully automatic oil spill detection in the ocean," *Proc. SPIE* **8891**, 889105 (2013).
5. A. Taravat, D. Latini, and F. Del Frate, "Fully automatic dark-spot detection from SAR imagery with the combination of nonadaptive Weibull multiplicative model and pulse-coupled neural networks," *IEEE Trans. Geosci. Remote Sens.* **52**(5), 2427–2435 (2014).
6. K. P. Luojus et al., "Accuracy assessment of SAR data-based snow-covered area estimation method," *IEEE Trans. Geosci. Remote Sens.* **44**(2), 277–287 (2006).
7. L. Gan et al., "Unsupervised SAR image segmentation based on triplet Markov fields with graph cuts," *IEEE Geosci. Remote Sens. Lett.* **11**(4), 853–857 (2014).
8. H. Zhang et al., "Synthetic aperture radar image segmentation by modified student's t-mixture model," *IEEE Trans. Geosci. Remote Sens.* **52**(7), 4391–4403 (2014).
9. K. Zhang, Z. Li, and X. Zhao, "Edge detection of images based on fuzzy cellular automata," in *Eighth ACIS Int. Conf. on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, pp. 289–294 (2007).
10. D. K. Patel and A. S. A. More, "Edge detection technique by fuzzy logic and cellular learning automata using fuzzy image processing," in *Int. Conf. on IEEE Computer Communication and Informatics (ICCCI '13)*, pp. 1–6 (2013).
11. S. Sinaie et al., "A hybrid edge detection method based on fuzzy set theory and cellular learning automata," in *Int. Conf. on Computational Science and Its Applications (ICCSA '09)*, pp. 208–214 (2009).
12. G. Akbarizadeh, "A new statistical-based kurtosis wavelet energy feature for texture recognition of SAR images," *IEEE Trans. Geosci. Remote Sens.* **50**(11), 4358–4368 (2012).
13. F. Del Frate et al., "Neural networks for oil spill detection using ERS-SAR data," *IEEE Trans. Geosci. Remote Sens.* **38**(5), 2282–2287 (2000).
14. L. Yu and J. Li, "Oil spill detection from SAR intensity imagery using a marked point process," *Remote Sens. Environ.* **114**(7), 1590–1601 (2010).
15. C. Brekke and A. H. S. Solberg, "Oil spill detection by satellite remote sensing," *Remote Sens. Environ.* **95**(1), 1–13 (2005).
16. J. Wan and Y. Cheng, "Remote sensing monitoring of Gulf of Mexico oil spill using ENVISAT ASAR images," in *21st Int. Conf. on Geoinformatics (GEOINFORMATICS '13)*, pp. 1–5 (2013).
17. M. Marghany, H. Mazlan, and A. P. Cracknell, "Fractal dimension algorithm for detecting oil spills using RADARSAT-1 SAR," in *Int. Conf. on Computational Science and Its Applications*, pp. 1054–1062, Springer, Berlin, Heidelberg (2007).
18. A. Taravat and N. Oppelt, "Adaptive Weibull multiplicative model and multilayer perceptron neural networks for dark-spot detection from SAR imagery," *Sensors* **14**(12), 22798–22810 (2014).
19. P. L. Shui and D. Cheng, "Edge detector of SAR images using Gaussian-gamma-shaped bi-windows," *IEEE Geosci. Remote Sens. Lett.* **9**(5), 846–850 (2012).
20. N. Pena et al., "Edge detection of real synthetic aperture radar images through filtered back projection," in *Int. Conf. on Systems and Informatics (ICSAI '12)*, pp. 1910–1913 (2012).
21. D. Borghys, V. Lacroix, and C. Perneel, "Edge and line detection in polarimetric SAR images," in *Proc. 16th Int. Conf. on Pattern Recognition*, Vol. 2 (2002).
22. J. Schou et al., "CFAR edge detector for polarimetric SAR images," *IEEE Trans. Geosci. Remote Sens.* **41**(1), 20–32 (2003).
23. A. Ferro, D. Brunner, and L. Bruzzonem, "Automatic detection and reconstruction of building radar footprints from single VHR SAR images," *IEEE Trans. Geosci. Remote Sens.* **51**(2), 935–952 (2013).
24. M. T. Alonso et al., "Edge enhancement algorithm based on the wavelet transform for automatic edge detection in SAR images," *IEEE Trans. Geosci. Remote Sens.* **49**(1), 222–235 (2011).
25. F. Baselice and G. Ferraioli, "Statistical edge detection in urban areas exploiting SAR complex data," *IEEE Geosci. Remote Sens. Lett.* **9**(2), 185–189 (2012).
26. A. Niedermeier, E. Romaneessen, and S. Lehner, "Detection of coastlines in SAR images using wavelet methods," *IEEE Trans. Geosci. Remote Sens.* **38**(5), 2270–2281 (2000).
27. H. Liu and K. C. Jezek, "Automated extraction of coastline from satellite imagery by integrating Canny edge detection and locally adaptive thresholding methods," *Int. J. Remote Sens.* **25**(5), 937–958 (2004).
28. S. Dellepiane, R. De Laurentiis, and F. Giordano, "Coastline extraction from SAR images and a method for the evaluation of the coastline precision," *Pattern Recognit. Lett.* **25**(13), 1461–1470 (2004).
29. X. Descombes et al., "Coastline detection by a Markovian segmentation on SAR images," *Signal Process.* **55**(1), 123–132 (1996).
30. J. S. Lee and I. Jurkevich, "Coastline detection and tracing in SAR images," *IEEE Trans. Geosci. Remote Sens.* **28**(4), 662–668 (1990).
31. S. Derrode and G. Mercier, "Unsupervised multiscale oil slick segmentation from SAR images using a vector HMC model," *Pattern Recognit.* **40**(3), 1135–1147 (2007).
32. G. Cattaneo et al., "Cellular automata in fuzzy backgrounds," *Phys. D* **105**(1–3), 105–120 (1997).
33. P. Maji and P. P. Chaudhuri, "FMACA: a fuzzy cellular automata based pattern classifier," in *Int. Conf. on Database Systems for Advanced Applications*, pp. 494–505, Springer, Berlin, Heidelberg (2004).
34. D. Nayak, S. K. Sahu, and J. Mohammed, "A cellular automata based optimal edge detection technique using twenty-five neighborhood model." arXiv preprint arXiv, 1402.1348 (2014).
35. Y. Wang and Y. Chen, "A comparison of Mamdani and Sugeno fuzzy inference systems for traffic flow prediction," *J. Comput.* **9**(1), 12–21 (2014).
36. J. Kittler, "On the accuracy of the Sobel edge detector," *Image Vision Comput.* **1**(1), 37–42 (1983).
37. D. Marr and E. Hildreth, "Theory of edge detection," *Proc. R. Soc. B: Biol. Sci.* **207**(1167), 187–217 (1980).
38. J. S. Huang and H. D. Tseng, "Statistical theory of edge detection," *Comput. Vision Graphics Image Process.* **43**(3), 337–346 (1988).
39. J. Prewitt and M. L. Mendelsohn, "The analysis of cell images," *Ann. N. Y. Acad. Sci.* **128**(1), 1035–1053 (1966).
40. R. Deriche, "Using Canny's criteria to derive a recursively implemented optimal edge detector," *Int. J. Comput. Vision* **1**(2), 167–187 (1987).
41. M. Rahmani and G. Akbarizadeh, "Unsupervised feature learning based on sparse coding and spectral clustering for segmentation of synthetic aperture radar images," *IET Comput. Vision* **9**(5), 629–638 (2015).

Mohammad Farbod received his BS degree in electronics engineering from the Shahid Chamran University of Ahvaz (SCU), Ahvaz, Iran, in 2013, and his MS degree in electronics engineering from the SCU, Ahvaz, Iran, in 2016. His research interests include synthetic aperture radar (SAR) image analysis and understanding, machine vision, image processing, and machine learning.

Gholamreza Akbarizadeh received his BS degree from Khajeh-Nassir Tousi University of Technology, Tehran, Iran, in 2003, and his MS and PhD degrees from Iran University of Science and Technology, Tehran, Iran, in 2005 and 2011, respectively, all in electronics engineering. His research interests include machine vision, image processing, remote sensing analysis, and geographic information system (GIS) techniques for earth surface mapping.

Abdolnabi Kosarian received his MS degree in electronic engineering from the University of Tehran, Iran, in 1987, and his PhD in electronic engineering from the University of Surrey, Surrey, UK, in 1998. Currently, he is an associate professor of electronic engineering at SCU, Iran. His research interests include semiconductor devices, electronic circuit design, and solar cells fabrication.

Kazem Rangzan received his BS degree in geology from Muslim Aligarh University, India, in 1984. He received his MS and PhD degrees from Muslim Aligarh University, India, in 1990 and 1993, respectively, all in applied remote sensing geology. His research interests include SAR image processing, machine learning, remote sensing, and GIS.