

# Automatic and Semi-automatic Extraction of Curvilinear Features from SAR Images

Emre Akyılmaz<sup>\*a</sup>, O. Erman Okman<sup>a</sup>, Fatih Nar<sup>a</sup>, Mujdat Cetin<sup>b</sup>

<sup>a</sup>SDT Space and Defence Technologies, Galyum Blok No.2 ODTU Teknokent Ankara, Turkey;

<sup>b</sup>Faculty of Engineering and Natural Sciences, Sabanci University, Orhanli-Tuzla, Istanbul, Turkey

## ABSTRACT

Extraction of curvilinear features from synthetic aperture radar (SAR) images is important for automatic recognition of various targets, such as fences, surrounding the buildings. The bright pixels which constitute curvilinear features in SAR images are usually disrupted and also degraded by high amount of speckle noise which makes extraction of such curvilinear features very difficult. In this paper an approach for the extraction of curvilinear features from SAR images is presented. The proposed approach is based on searching the curvilinear features as an optimum unidirectional path crossing over the vertices of the features determined after a despeckling operation. The proposed method can be used in a semi-automatic mode if the user supplies the starting vertex or in an automatic mode otherwise. In the semi-automatic mode, the proposed method produces reasonably accurate real-time solutions for SAR images.

**Keywords:** SAR, curvilinear feature, curvilinear structure, edge detection, Dijkstra's algorithm, graph search

## 1. INTRODUCTION

SAR imaging sensors have the ability to work in all weather conditions during day and night which make the resulting images vital for remote sensing applications. Some of the extensively used image analysis tasks are water segmentation, land cover/use classification, change detection (CD), automatic target detection (ATD) and automatic target recognition (ATR) [17]. Because of the sensor capabilities, there is a high demand for these images which also results in rapid increase in the quantity of SAR images that need to be analyzed. However, SAR images suffer from speckle noise which makes it quite challenging to improve the speed and accuracy of the algorithms designed for the image analysis tasks.

SAR images exhibit important features for the detection and the recognition of man-made structures, e.g. buildings, which produce bright intensity values due to their reflectivity characteristics. On the other hand, some man-made structures, such as roads, may have smooth surfaces which results in lower reflectivity characteristics, than their environment. These reflectivity characteristics play a crucial role in the image analysis tasks. Accurate extraction of curvilinear features in remotely sensed imagery can provide useful information for recognition of man-made structures. In some cases, the specific algorithm which aims to find the curvilinear features may not easily detect the desired features of man-made structures, such as fences, as they appear as discontinuous bright scattered points in SAR images. Although there have been numerous attempts for the automatic extraction of curvilinear features from SAR images, the skill of an expert may still be necessary for a reliable result. Because of the discontinuity among bright scattered points and low signal to noise ratios (SNR) among them, the detection of curvilinear features needs to be performed with global approaches rather than local approaches. The skill of such an expert is especially useful for scanning large areas to detect or recognize objects [2].

Katartzis et al. considered extraction of linear features exploiting: (i) local approaches which use local operators, and (ii) global approaches which depend on additional knowledge about the features to be extracted [9]. Local operators evaluate geometric properties such as derivatives of intensity functions, which are useful in edge detection techniques or morphological operations. For example, many edge detection operators detect edges by evaluating the first order derivatives. As mentioned before, curvilinear features in SAR images contain discontinuous bright scattered points therefore calculation of the magnitude of the gradient may not produce the desired result due to such discontinuities. Additionally, edge detectors such as Canny [15] which are usually applied at the pixel level, perform operations on false edges that results in some waste of computational resource. On the other hand, global methods extract lines from SAR images by evaluation of possible curvilinear features globally, using tools such as Hough transforms, Markov Random

Field models, or minimum-cost search techniques [9][11]. Since local methods compute geometric image properties and deduce meaningful results in local areas, they are usually faster but less accurate compared to global methods.

In many applications, the local and global techniques are performed jointly to extract curvilinear features [4-7, 11, 13]. Since linearity is a desired property of many target types, these applications are useful for the extraction of specific curvilinear features such as road networks. Wiedemann and Hinz [5] proposed a method for the extraction of lines, in which they represent road segments by use of different image channels. In their proposed model, potential road segments which depend on the threshold values of radiometric contrast between lines and their surroundings are constructed by prior knowledge of the road networks. Road segments are extracted from various image channels and fused afterwards. After this fusion process of potential road segments, the seeds are selected for construction of the road network. Seeds are connected as an optimal path found by Dijkstra's algorithm [10]. Wessel et al. stated that the visibilities of roads may differ in each SAR image and as a solution they applied this method to SAR images [18]. In order to provide correct results from the road network extraction, the method proposed in [5] is modified to include internal evaluation of the extracted road segments which are classified for different road types. Dhinesh et al. presented a multi-resolution approach by which the speckles are firstly reduced in the aerial images, and then the extracted curvilinear features in different image resolutions are classified to find the desired networks [6]. Dimou et al. performed a fuzzy edge enhancement followed by an edge tracing on SAR images as an alternative edge detection approach [7].

Although these techniques are suitable for finding specific curvilinear features such as roads networks, they are not very well suited to extract curvilinear features appearing in structures like fences or pipelines which contain bright scattered points connected with weak edges. The detection of an edge between these discontinuous bright points is extremely difficult due to the low SNR.

In this paper, an approach for extracting curvilinear features from SAR images is proposed. This approach is centered on finding a unidirectional path which is composed of edges with relatively high SNR. Since the bright points are considered as possible parts of the curvilinear features, these bright points are examined individually without considering the discontinuity of linear segments. The proposed approach consists of three phases. In the first phase, potential graph vertices are found by filtering the regional maxima using a region based constant false alarm rate (RB-CFAR) algorithm [8] where regional maxima are found on a despeckled image obtained by feature preserving despeckling (FPD) [1]. In the second phase, Dijkstra's graph search algorithm is used to extract curvilinear features as the possible paths crossing over the vertices [10]. In the last phase, optimum path is extracted by back tracking the graph from end point to start point and accepted as curvilinear feature if smoothness condition is satisfied. Start and end points are determined around a seed point which is either provided by a user or detected automatically. This methodology is implemented for SAR images and the results demonstrate the effectiveness of such methodology in extracting curvilinear features.

In Section 2 of this paper, the proposed approach is described. The implementation details are provided in Section 3. Experimental results are presented in Section 4. Finally, the paper is concluded in Section 5.

## 2. METHOD

In SAR images, curvilinear features usually contain discontinuous edges and bright scattered points. Edge detection techniques can be used to find the properties of the image by using discrete derivatives or ratio of average (ROA) operators calculated on the image [19]. As the scattered points are not fully connected, edge detection techniques mostly fail to extract curvilinear edges. In addition, false edges which may not be actual parts of the curvilinear features may appear due to the speckle noise in SAR images [1].

In this study, the curvilinear features to be extracted are defined as an optimum path composed of a sequence of feasible bright points. The search for this path through the edges between the bright points is posed as a graph search problem. For this purpose, Dijkstra's graph search algorithm [10] is used.

The proposed method has the characteristics of both local and global methods. It is similar to the global methods in terms of minimizing a cost function that is applicable over a large part of the image. On the other hand, the graph is constructed locally in a dynamic manner in the sense that the construction proceeds from a partial state of the path to its completion. In this respect, the proposed approach contains some elements of local methods as well. We believe that the proposed approach combines useful characteristics of local and global methods in order to produce reasonably accurate results in a computationally efficient way. Even if the proposed method can be argued as a hybrid approach, it seems that it better fits to the family of global methods.

The proposed approach consists of three phases: vertex extraction, graph search, and path validity check phases which are shown in Figure 1.

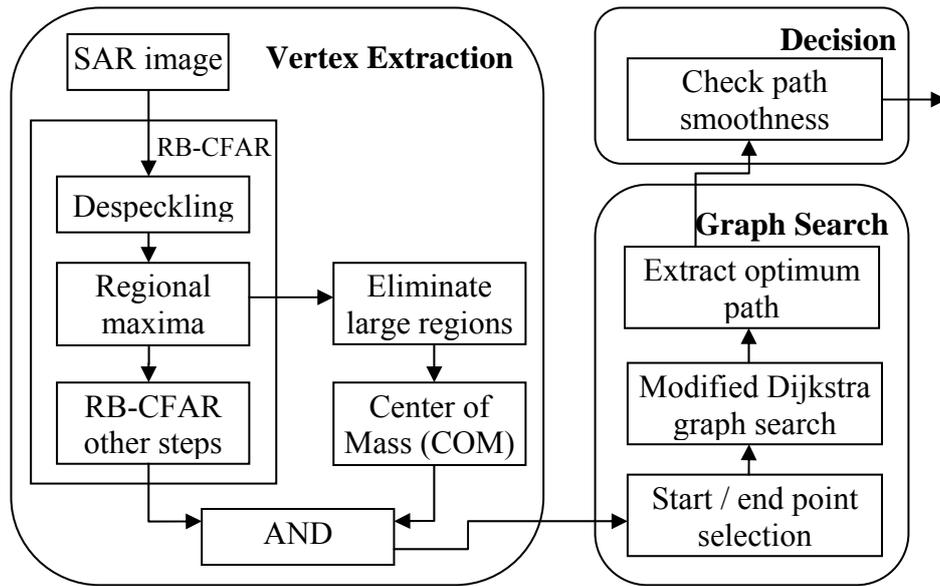


Figure 1. Block diagram of the proposed approach.

## 2.1 Vertex Extraction

The properties of SAR images such as image resolution, the level of speckle noise and clutter complexity, pose challenges for automatic target detection methods. In these methods, separation of an object with different characteristics from a background clutter is one of the most central problems [8]. However, background clutter contains speckle noise which is a spatially correlated noise. Such noise can be misinterpreted as being part of potential target objects, which may reduce the efficiency of target detection algorithms. In order to separate an object from background clutter in speckled SAR images, constant false alarm rate (CFAR) based methods can be used because of their speed and accuracy.

The main purpose of using CFAR based methods is to find the objects with different characteristic comparing to their clutter with a constant false alarm rate. Traditional CFAR methods use a center area, guard region and a clutter region to estimate clutter statistics. Therefore, in traditional CFAR based methods prior knowledge about the size and shape of the object must be known which may not be possible in every scenario. Traditional CFAR methods work successfully for targets with homogenous background clutter but detector performance degrades for targets with heterogeneous background clutter. However, the recently proposed RB-CFAR method determines the shape and size of the target and clutter region adaptively by employing a region growing method [8]. This region based approach leads to the detection of targets with different shapes and sizes. This flexibility with the shape of the targets also increases the detection performance for the targets with heterogeneous background clutter. The RB-CFAR method is used in this study since prior knowledge about target regions is not available in the problems of interest here and background clutter may be both homogenous and heterogeneous (Figure 2). Its use also provides a fast execution time since its steps are parallelizable. Furthermore, the RB-CFAR method and the proposed curvilinear feature extraction method both contain FPD and regional maxima steps in common, which helps to decrease the overall computation.

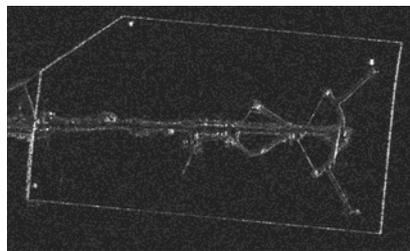


Figure 2. A SAR image containing fences.

As mentioned before, curvilinear features are extracted as the optimum path crossing over the possible vertices where vertices are bright point scatterers. Vertex extraction steps are:

- a) Despeckle the image using FPD
- b) Determine the regional maxima on the despeckled image
- c) Refine regional maxima by eliminating large areas
- d) Calculate the center of mass (COM) for each regional maximum
- e) Find target regions using the RB-CFAR method
- f) Apply an AND operator to COMs and RB-CFAR  $\rightarrow$  AND(d, e)

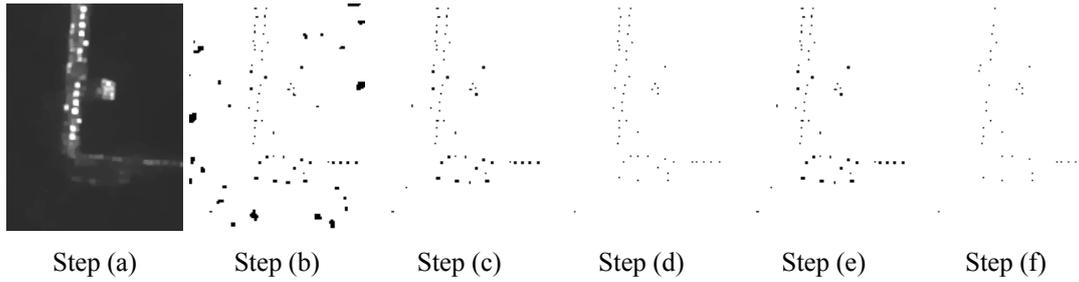


Figure 3. Steps for the vertex extraction.

In Figure 3, vertex extraction steps for the bottom left corner of the SAR image given in Figure 2 are shown. As seen in Figure 3, a sparse vertex representation is obtained from dense image pixels which allow the use of a graph search algorithm on obtained vertices (Figure 3(f)).

Brief explanation of each of the vertex extraction steps is given below:

a. Despeckle the image using FPD

Bright scattered points have greater reflectivity values compared to their neighbors in a despeckled image (Figure 3). Based on that observation, we first despeckle the images using FPD method (step(a)) and then find the regional maxima on the despeckled image (step(b)). In FPD the SAR image denoising problem is posed as an optimization problem:

$$\hat{f} = \arg \min_f J(f) \quad (1)$$

where the cost function  $J(f)$  is defined as:

$$J(f) = \|g - f\|_2^2 + \lambda_1^2 \|f\|_k^k + \lambda_2^2 \|Df\|_k^k \quad (2)$$

in which  $g$  is the observed, speckled SAR image,  $\|\cdot\|_k$  denotes the  $l_k$  norm,  $D$  is the 2-D derivative operator,  $f$  is the unknown noise-free reflectivity image of the scene, and  $\lambda_1$  and  $\lambda_2$  are scalar parameters. The first term of the objective function is the data fidelity term; the second and third terms are for enhancing point based and region based features, respectively. This cost function has been obtained as a simplified version of the cost function used in the feature-enhanced SAR imaging approach of [14]. FPD parameter  $k$  is set to 1 due to its good edge preserving characteristics,  $\lambda_1$  is set to zero since there is no need to enhance the point based features for our despeckling objectives, and  $\lambda_2$  is set to 10 which is tuned for 16 bit TerraSAR-X strip mode and TerraSAR-X high resolution spotlight mode SAR images.

b. Determine regional maxima on despeckled image

Relatively bright local regions in the scene are found using the regional maxima as we describe next. A regional maximum is defined as a group of connected pixels with the same intensity value that is greater than the intensities of pixels in the neighboring regions [16]. For defining regions and region boundaries either 4 or 8 connectedness can be used. In this study 8-connectedness is used to allow diagonally connected sub-regions within a region. In Figure 4, regional maxima are illustrated with bold numbers where numbers corresponds to reflectivity values in a SAR image.

.	.	1	3	3	2	1	7	7	2	1	.	.
.	.	1	3	1	2	1	8	8	9	1	.	.
.	.	1	1	1	6	6	7	7	8	1	.	.
.	.	4	1	3	7	4	6	6	5	1	.	.
.	.	4	1	7	7	7	7	7	4	1	.	.
.	.	4	2	5	7	7	7	5	5	2	.	.
.	.	3	2	2	2	3	2	1	3	2	.	.
.	.	3	2	3	2	2	2	4	2	2	.	.
.	.	3	3	3	3	4	4	4	4	2	.	.

Figure 4. Illustration of regional maxima.

c. Refine regional maxima by eliminating large areas

Since a curvilinear feature is expected to cover a relatively narrow strip in the SAR image, regional maxima that cover a wide area are not good candidates for being part of a curvilinear structure. Based on this observation, if a regional maximum covers a larger area than a pre-determined threshold, we eliminate it and do not consider it in further processing. By eliminating large regions, we are left with groups of bright points which may potentially be parts of curvilinear features.

d. Calculate the COM of each regional maximum

The algorithm we described above for the extracting the regional maxima produces regions which are formed and defined by a group of pixels. However, we need a single representative pixel for each region since it will act as a vertex for the corresponding scatterer. COMs are calculated and pixels other than COMs are removed from regional maxima for regions having more than one pixel in regional maxima. If the COM does not lie within the extracted region, than the pixel in the region with closest distance to the COM is used.

e. Find target regions using the RB-CFAR method

RB-CFAR method consists of a number of algorithmic steps starting with despeckling the image using FPD so that homogeneous regions are smoothed. In the second step, regions which are brighter as compared to their neighbors (regional maxima) in the despeckled image are found and declared as seed regions to initiate the search potential targets. Target region candidates are obtained by growing these seed regions using a region growing algorithm. A candidate target region may have an arbitrary, irregular shape hence the guard and clutter regions enveloping this candidate target region will have similar arbitrary shapes as well since they are constructed by dilating the candidate target region. Possible non-clutter pixels in the clutter regions are censored so that clutter statistics can be obtained in a more robust manner. Clutter statistics are calculated using the original image within the clutter region. Decision of possible target regions, either target or non-target, is made using the maximum intensity of that candidate target region using an adaptive threshold value which satisfies a CFAR constraint. In this study, we used the Rayleigh distribution and a probability of false alarm of  $P_{fa} = 10^{-2}$  for the RB-CFAR method to detect targets including potential scatterers. In FPD step of RB-CFAR, parameter  $k$  is set to 1,  $\lambda_1$  is set to zero, and  $\lambda_2$  is set to 10, so existing FPD result and regional maxima can be used in RB-CFAR to increase computational efficiency.

f. Apply an AND operator to COMs and RB-CFAR

The final vertex is obtained by ANDing the COMs and the RB-CFAR image. If RB-CFAR eliminated a target, then ANDing with the COMs image removes the scatterer from the COMs image. Remaining COMs are the vertices which will be used by the graph search algorithm. For example, there may be various regional maxima in a sea clutter after applying FPD but RB-CFAR eliminates most of them hence substantial amount of false vertices are eliminated.

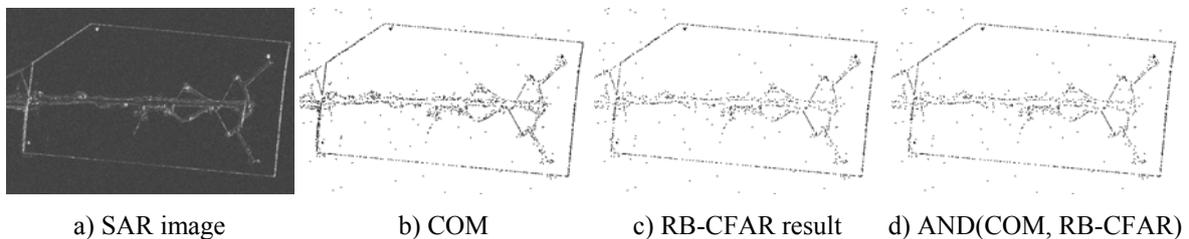


Figure 5. ANDING the COM and the RB-CFAR image (visually clear).

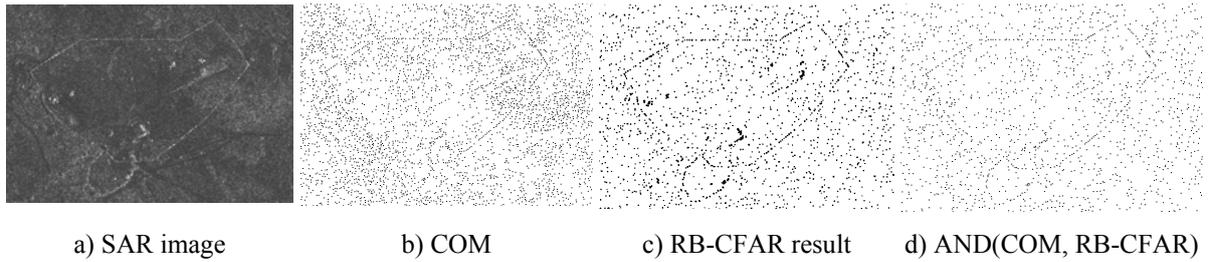


Figure 6. ANDing the COM and the RB-CFAR image (visually less clear).

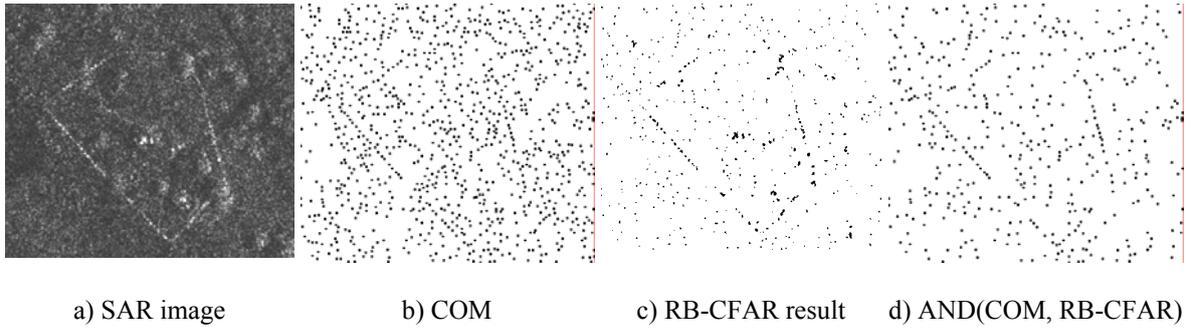


Figure 7. ANDing the COM and the RB-CFAR image (not completely visible).

In Figure 5, the image which has visually clear curvilinear feature is shown. Our vertex extraction produced successful results in terms of eliminating the false targets. In Figure 6, we consider an image with a less prominent curvilinear structure. In this case extraction of the curvilinear feature is more challenging as evidenced by our results. In Figure 7, the curvilinear features are not completely visible and exhibit an interrupted pattern so it is a very challenging problem to extract the curvilinear features from this type of image.

## 2.2 Graph Search

We depict the curvilinear features as the paths crossing over scatterers that satisfy edge distance and smoothness constraints. A cost function is developed to assess the quality of a path. Therefore curvilinear feature extraction in a region of interest is defined as the search of the optimum path using Dijkstra's graph search. Edges are defined only between the vertices within a predefined distance to each other. A Kd-tree [14] is constructed from the vertices obtained in the vertex extraction phase in order to find the vertex within a specified radius in  $O(\log n)$  operation where  $n$  is the number of vertices. Only the edges having an average reflectivity value greater than a predefined threshold value are used. The threshold value should be selected in a generous manner so that the real edges are not missed.

Graph search starts with a set of predefined start and end vertex. If only a seed point is provided, the choice of start and end vertices are automatically determined using the modified ROA operator.

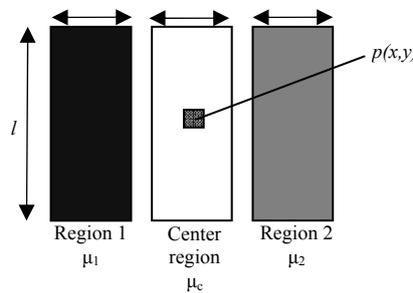


Figure 8. ROA window.

In ROA window there are a central region, two side regions and a guard region between them as shown in Figure 8. Length of window (l) may change, width of the center region is 5 pixels, width of the guard region is 3 pixels, and width of the side regions are 4 pixels. ROA response is calculated as

$$p(x, y) = \max \left\{ \tanh\left(\tau \frac{\mu_c + 1}{\mu_1 + 1}\right), \tanh\left(\tau \frac{\mu_c + 1}{\mu_2 + 1}\right) \right\} \quad (3)$$

where  $\tau$  is a scaling factor for tanh (hyperbolic tangent) function which is set to 0.25. Response value is normalized between 0 and 1 due to tanh function. ROA response is increased as  $\mu_c$  becomes higher comparing to either  $\mu_1$  or  $\mu_2$ .

In order to determine start and end points, ROA window is shifted up to 7 pixels around the seed point and centered on the COMs. At each COM, ROA window is rotated from  $0^\circ$  to  $180^\circ$  and the response  $p(x,y)$  is calculated. Seed point is refined as the location of the COM that has the highest response. Start and end points are placed in a distance  $S_r$  from the refined seed point in the direction of highest ROA response (Figure 9.b).

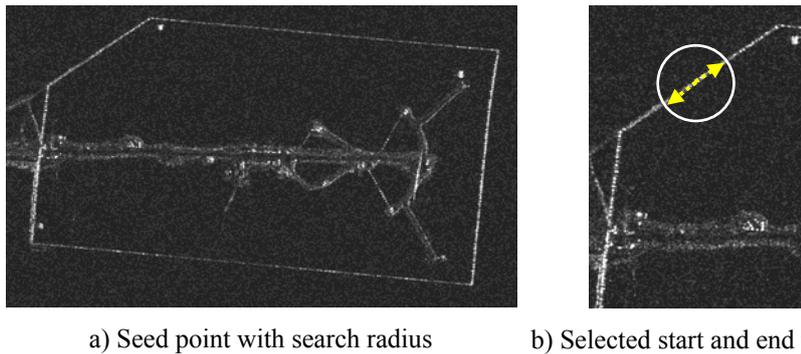


Figure 9. Automatic detection of start and end points using a seed point.

Highest ROA response in a region of interest (ROI) is calculated on COMs in order to detect seed point automatically. The result of the experiments show that the use of ROA operator is a reasonable but current approach needs to be improved to developed a full automatic seed point detection technique.

Dijkstra's graph search algorithm is slightly modified in order to extract curvilinear features from the extracted vertices in a dynamic manner using a normalized cost function. During the search of the optimum path, graph is dynamically constructed where its expansion is stopped when either algorithm is reached to the desired end point or no new edges are found to explore. Essentially, Dijkstra's algorithm uses only distance to define a cost function hence optimum path corresponds to the shortest one. In our method, length of the edge, smoothness of the edge direction, reflectivity value of scattered points and ROA response in edge direction define the cost function. If the calculated cost of a path is not normalized then Dijkstra's algorithm will tend to find a path with minimal number of edges to decrease the total cost. In order to avoid this problem, the cost associated with a path is normalized using total length of path so that Dijkstra's algorithm can find a path in any length. Additionally, dynamic graph exploration is proposed hence the employed graph search algorithm is called as modified Dijkstra's algorithm.

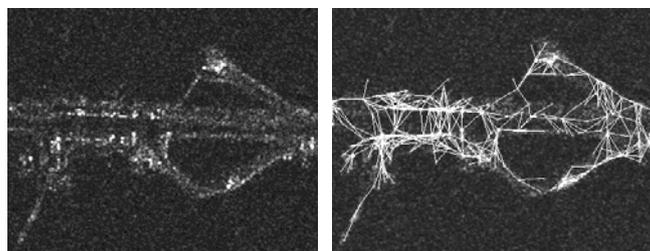


Figure 10. SAR image and graph of extracted vertices.

In Figure 10, graph for extracted vertices is shown. The edges between the two points belong to a curvilinear path where the one with minimum cost constitute the optimal path.

The cost function for a path is defined as:

$$J_{path} = \frac{\sum_{i=1}^n J_i}{\sum_{i=1}^n L_i} = \frac{\sum_{i=1}^n (w_L L_i^2 + w_S S_i + w_P P_i + w_R R_i)}{\sum_{i=1}^n L_i} \quad (4)$$

where  $n$  is the number of edges in the path,  $J_i$  is the edge cost of the  $i^{\text{th}}$  edge,  $L_i$  is the normalized length of the  $i^{\text{th}}$  edge,  $S_i$  is the directional smoothness of the  $i^{\text{th}}$  edge,  $P_i$  is the scattered power of the end point of the  $i^{\text{th}}$  edge,  $R_i$  is the average reflectivity of the  $i^{\text{th}}$  edge, and  $J_{path}$  is the normalized cost value for the whole path.  $w_L$ ,  $w_S$ ,  $w_P$ , and  $w_R$  represent the weights of the corresponding cost terms in edge cost  $J_i$ .

- Normalized length of the  $i^{\text{th}}$  edge ( $L_i$ )

The normalized edge length  $L_i$  is used to favor the exploration of vertices in closer proximity during the search for the optimal path.  $L_i$  is defined as

$$L_i = \frac{\|\vec{E}_i\|}{L_{\max}} \quad (5)$$

where  $L_{\max}$  is the radius of the search space in the Kd-tree,  $\vec{E}_i$  is the  $i^{\text{th}}$  edge direction,  $\vec{V}_{i-1}$  is the starting point (vertex) of  $\vec{E}_i$  therefore  $\vec{V}_i = \vec{V}_{i-1} + \vec{E}_i$ , and  $\|\vec{E}_i\|$  is the length of the edge  $\vec{E}_i$ . This term fosters the algorithm to choose closest bright scattered points as potential vertex candidates. As it can be seen from the expression when  $L_i$  starts to increase, the scatter distance value increases which results an increase in the cost function. Hence there is a preference to choose closer vertices.

- Directional smoothness of the  $i^{\text{th}}$  edge ( $S_i$ )

The directional smoothness term is used to indicate a preference for smooth turns of edges in the optimal path. Without this term, the appearance of zigzag behavior in extracted path is very common.

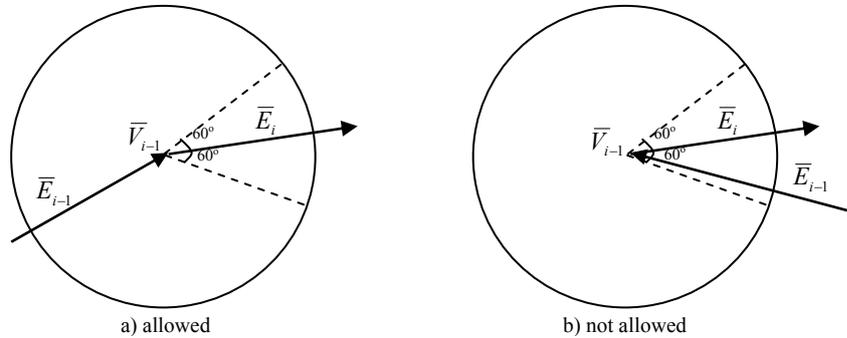


Figure 11. Illustration of directional smoothness.

Directional smoothness is calculated by taking the dot product of vectors  $E_i$  and  $E_{i-1}$ .

$$\cos \theta_i = \frac{\langle \vec{E}_i, \vec{E}_{i-1} \rangle}{\|\vec{E}_i\| \|\vec{E}_{i-1}\|} \quad (6)$$

where  $\theta_i$  is the angle between the  $i^{\text{th}}$  edge and the preceding edge. If  $\theta_i$  is zero then maximum smoothness is obtained, if  $|\theta_i|$  is  $\pi$  then maximum roughness is obtained. To prevent sharp turns, a thresholding on  $\theta_i$  is defined such that only edges with  $\cos \theta_i > \cos T_0$  are explored where  $T_0$  is  $120^\circ$ . By applying this thresholding operation on  $\theta_i$ , edges oriented up to  $60^\circ$  around the opposite direction are disallowed.

For allowed edges the directional smoothness is defined as

$$S_i = 1 - \frac{\cos \theta_i - \cos T_\theta}{1 - \cos T_\theta} \quad (7)$$

where  $S_i$  is zero for  $\theta_i = 0$ , and  $S_i$  is 1 for  $|\theta_i|=120^\circ$ . Hence,  $S_i$  produce smaller values for smooth directional changes which make it suitable for a cost function to be minimized. Thus, the optimal path exhibits a tendency to have smooth boundary curvature.

- Scattered power of the end point of the  $i^{\text{th}}$  edge ( $P_i$ )

The scattered power term forces the optimal path to pass through the bright pixels having higher reflectivity values. This term is defined as

$$P_i = 1 - I_i \quad (8)$$

where  $I_i$  is the normalized image reflectivity value at the end point ( $\bar{V}_i$ ) of the  $i^{\text{th}}$  edge.  $I_i$  is normalized between 0 and 1 where 0 corresponds to lowest reflectivity and 1 corresponds to highest reflectivity. When the end point reflectivity of the edge becomes 1  $P_i$  becomes 0 which makes it suitable for a cost function to be minimized. Thus, the optimal path exhibits a tendency to pass through points with high scattered power.

- ROA response of the  $i^{\text{th}}$  edge ( $R_i$ )

Average reflectivity of the edge and its ratio towards its neighbors shows the higher possibility of being a part of optimal path. ROA response of the  $i^{\text{th}}$  edge is defined as

$$R_i = 1 - p(x, y) \quad (9)$$

where  $p(x,y)$  is the ROA response of the edge  $E_i$  that is defined in equation 3.

### 2.3 Extraction of the Optimal Path

The Graph Search algorithm stops when the end point is reached, and afterwards the optimal path is extracted. To extract the optimal path, starting from the end point, each previously visited vertex is labeled until the start point has been reached. Average directional smoothness of the path is calculated using equation 11. If this average value is lower than a threshold then path smoothness constraint is satisfied, therefore the path is marked as an optimal path, otherwise it is ignored.

$$O_{path} = \begin{cases} 1 & S_{path} < T_s \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

$$S_{path} = \frac{\sum_{i=1}^n L_i S_i}{\sum_{i=1}^n L_i} \quad (11)$$

where  $n$  is the number of edges in the extracted path,  $S_{path}$  is the directional smoothness of the extracted path, and  $T_s$  is the path smoothness threshold. This formulation allows few sharp turns in combination with many smooth turns. For example, for a path having an equilateral triangle shape with sides having length 100 and containing 10 edges,  $S_{path}$  is 0.025 which is very close to zero meaning a very smooth path. In this study, the default value for  $T_s$  is chosen as 0.25 for automatic mode and 0.50 for semi-automatic mode. In the semi-automatic mode,  $T_s$  is selected more loosely compared to the automatic mode since a robust seed point is supplied by the user.

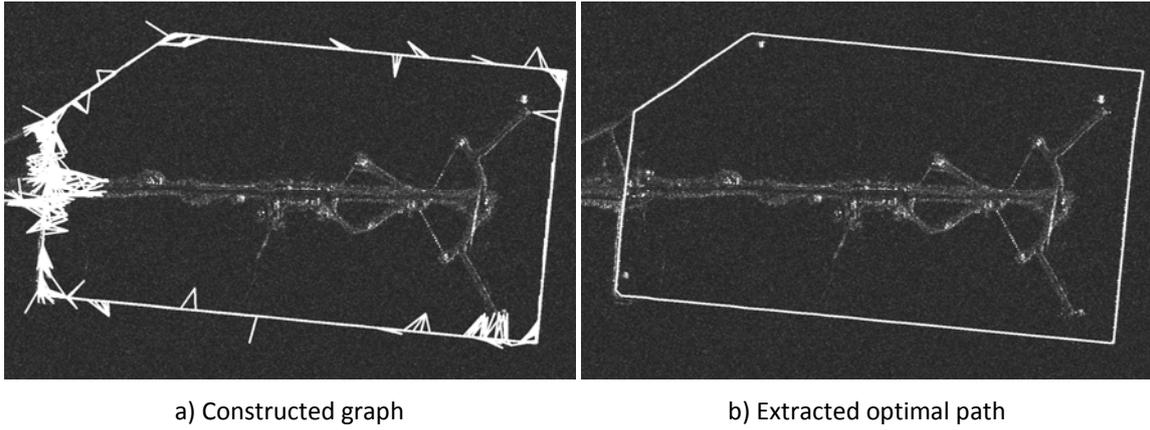


Figure 12. Extraction of the optimal path

In Figure 12, the dynamically constructed graph and the extracted optimal path are shown. If the dynamically constructed graph only explores the edges in the optimal path and its close proximity then it shows that the curvilinear feature is extracted in a computationally efficient manner. Such computational efficiency occurs for curvilinear features with homogenous background and very high SNR. One can also argue that exploration of the edges in the optimal path and its close proximity also shows the quality of the cost function employed; smaller number of explored edges means a better cost function definition and a better search methodology.

### 3. IMPLEMENTATION

The core of the proposed approach is a slightly modified version of Dijkstra's algorithm. Two modifications are, (i) use of a normalized cost function with terms other than distance and (ii) dynamical construction of the explored graph. However these modifications introduce some issues (Figure 13) that are not inherent to Dijkstra's algorithm, and that need to be resolved. During dynamic exploration of the graph, self-intersections may appear so if an edge will cause a self-intersection than this edge is not added to graph. Since this check is based on an integer line intersection algorithm it is quite fast. The normalized cost function may cause cyclic loops to be appearing during dynamic graph exploration which causes part of the graph, other than cyclic part itself, to be lost or infinite loops in optimum path extraction.

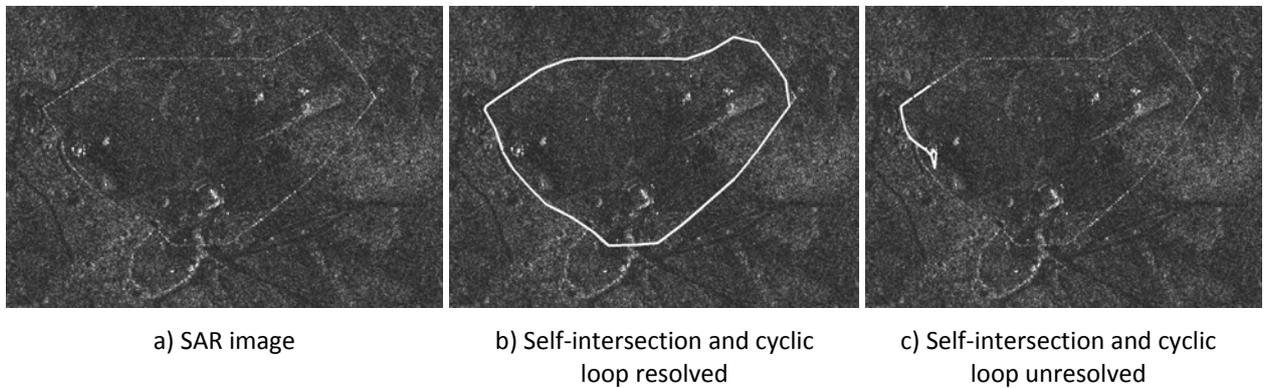


Figure 13. Self-intersection and cyclic loop issues.

In order to prevent cyclic loops, the path is traversed until either the seed point is reached or the end point of the edge candidate is reached. If the end point of the edge candidate is reached, then it means a cycle will be formed if this edge is added to the graph hence this new edge is not added to the graph. Dijkstra's algorithm have computational complexity of  $O(|E|+|V|\log|V|)$  where our modified Dijkstra's algorithm has a computational complexity of  $O(|E|+|V|\log|V|+M^2)$  where  $M$  is the length of the maximum path obtained during graph search. Our modified Dijkstra's algorithm has a new term with a considerable overhead yet the proposed method still runs in real-time range.

#### 4. EXPERIMENTAL RESULTS

$w_L$ ,  $w_S$ ,  $w_P$ , and  $w_R$  are the weights of the terms in cost function given by equation 4 . Therefore, these weights have great influence of how successful the curvilinear feature is extracted.  $w_P$  and  $w_R$  terms are guiding the search for optimum path while  $w_L$  and  $w_S$  act as regularization terms. If weights are not balanced curvilinear feature extraction can be inaccurate or completely wrong. Explored graph and extracted optimum path for different weights are shown in Figure 12. Best path is obtained in Figure 14.d while weights in Figure 14.b and Figure 14.c produces inaccurate results. If only  $w_L$  is used solution corresponds to shortest path (Figure 14.e), if only  $w_S$  is used solution corresponds to smoothest path (Figure 14.f), if only  $w_P$  or  $w_R$  is used solution is not regularized hence wrong (Figure 14.g, Figure 14.h).

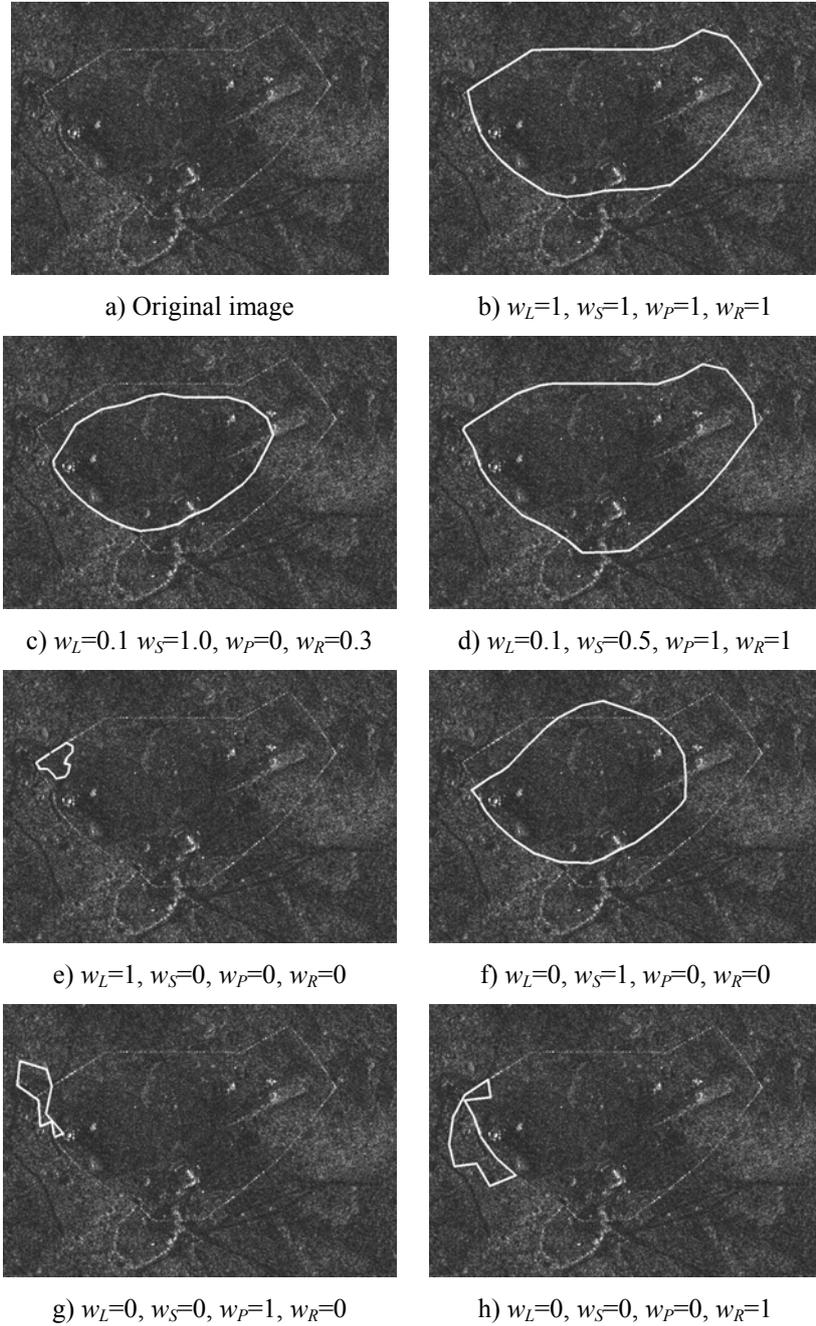


Figure 14. Experimental results.

## 5. CONCLUSION

In this paper, an approach for the extraction of curvilinear features from SAR images is presented. Curvilinear feature is defined as an optimum unidirectional path crossing over the vertices. Proposed approach provides accurate extraction of desired curvilinear features in SAR images with high amount of speckle noise. It can be used in a semi-automatic mode if the user supplies the starting vertex or in an automatic mode otherwise. In the semi-automatic mode, proposed method produces reasonably accurate solutions in near real-time for SAR images. In automatic mode, proposed method produces reasonably accurate solutions only in a region of interest enveloping the curvilinear feature.

## REFERENCES

- [1] Okman, E.O., Nar, F., Demirkesen, C. and Çetin, M., "Feature Preserving SAR Despeckling and its Parallel Implementation with Application to Railway Detection", EUSAR (2012).
- [2] McKeown, Jr.D.M, McGlone.,C., Cochran,S.D., Hsieh, Y.C., Roux,M. and Shufelt, J.A., "Feature Extraction and Object Recognition", Manual of Photogrammetry Addendum, American Society for Photogrammetry and Remote Sensing, Maryland (1996).
- [3] Guindon, B., "Application of spatial reasoning methods to the extraction of roads from high resolution satellite imagery", Proc. of International Geoscience and Remote Sensing Symposium, Seattle, Washington (1999).
- [4] Wessel, B. and Wiedemann, C. "Analysis of Automatic Road Extraction Results From Airborne SAR Imagery", ISPRS Archives Vol. XXXIV Part 3/W8, Munich (2003).
- [5] Wiedemann, C. and Hinz S., "Automatic Extraction and Evaluation of Road Networks from satellite imagery", IAPRS, Vol. 32, Part 3-2W5, "Automatic Extraction of GIS Objects from Digital Imagery", München, 1999 Wiedemann and Hinz (1999).
- [6] Dhinesh, M., Das, S. and Varghese, K., "Automatic Curvilinear Structure detection from Satellite Images using Multiresolution GMM", IJISE GA USA ISSN:1934-9955, vol.2 bo.1 (2008).
- [7] Dimou, A., Uzunoglou, N., Frangos, P., Jager, G. and Benz, U., "Linear Features detection in SAR images using Fuzzy Edge Detector (FED)", Space-Based Observation Technology, October (2000).
- [8] Nar, F., Demirkesen, C., Okman, O.E. and Çetin, M., "Region Based Target Detection Approach for Synthetic Aperture Radar Images and Its Parallel Implementation", SPIE, (2012).
- [9] Katartzis, A., Sahli, H., Pizurica, V. and Cornelis, J., "A Model-Based Approach to the Automatic Extraction of Linear Features from Airborne Images", IEEE Trans. on Geoscience and Remote Sensing, vol. 39 (9), pp. 2073-2079 (2001).
- [10] Dijkstra, E.W., "A Note on two problems in connection with graphs", Numerische Mathematik 1: 269-271, (1959).
- [11] Hellwich, O., Mayer, H. and Winkler, F., "Detection of Lines in Synthetic Aperture Radar (SAR) Scenes"
- [12] Bentley, J. L., "Multidimensional binary search trees used for associative searching", Magazine Communications of the ACM, New York / USA, Volume 18 Issue 9 (1975).
- [13] Wang, C. and Wang, R., "Line Extraction in SAR images", Image Processing and Pattern Recognition in Remote Sensing, SPIE Vol. 4898, SPIE (2003).
- [14] Çetin, M. and Karl, W.C., "Feature Enhanced Feature-Enhanced Synthetic Aperture Radar Image Formation Based on Nonquadratic Regularization", IEEE Trans. on Image Processing, vol 10, no. 4, pp. 623 – 631 (2001).
- [15] Canny, J., "A computational approach to edge detection", IEEE Trans. on Pattern Analysis Machine Intelligence vol. 8 no.11: 679 – 698. (1986).
- [16] Dougherty, E.R., Lotufo, R.A., "Hands-on Morphological Image Processing", SPIE Tutorial Texts in Optical Engineering (2003).
- [17] Yeang, C.P, Cho, C., Shapiro, J.H., "Toward target recognition from synthetic aperture radar imagery using electromagnetics-based signatures", Society of Photo-Optical Instrumentation Engineers (2003).
- [18] Wessel, B., Wiedemann, C., Ebner, H., "The role of context for road extraction from SAR imagery", IGARSS, vol 6, pp. 4025-4027 (2003).
- [19] Tupin, F., Maitre, H., Mangin, J. F., Nicholas, J. M., Pechersky, E., "Detection of Linear Features SAR Images: Application to Road Network Extraction", IEEE Trans. Geoscience and Remote Sensing, vol. 36, no. 2 (1998).