Linear Objects Detection on SAR Images

Oleg Yu. Ivanov

Ural Federal University, Yekaterinburg, Mira st., 19, Russia,
ol_iv@list.ru

Abstract. Allocation of linear structures on images is required to solve a large number of thematic problems of the Earth remote sensing. Roads and railways, power lines, pipelines, and borders of natural areas "land-sea", "forest-field") are examples of such structures. It is known that traditional algorithms for linear targets and structures detection are not effective, because radar images have special distinctive features (*i.e.* speckle-noise), which complicate the target detection problems. In the paper neural network algorithm based on the Hough transform and application of Kohonen neural networks is suggested and researched. At the first stage, a radar image is transformed into a Hough plane, where linear targets give peak responses; then a Kohonen neural network is used to find these peaks. It is shown that the neural gas algorithm for the network weights adjustment is more suitable than the "Winner takes all" rule. Also, an exponential weight calibration function for better convergence is offered. Examples of real radar space images processing obtained by the RADARSAT-1 are given. Results of processing show that the suggested algorithm is suitable for linear targets detection on the radar images.

Keywords: Linear objects detection algorithm, synthetic aperture radar, Hough transformation, Kohonen neural network

1 Introduction

Nowadays satellite imagery becomes increasingly popular due to the wide spectrum of solved problems. Successful operation with a variety of orbital remote monitoring systems allow one to obtain quality images of the Earth surface in different bands of the electromagnetic spectrum. Among the latter, the synthetic aperture radar (SAR) technology plays an important role and it can carry out imagery regardless of weather conditions and natural lighting of surface [1, 6, 7].

There are several algorithms for automatic linear structures detection on the remote sensing data, and functional analysis (Fourier analysis, Gabor wavelets algorithm) and parametric analysis algorithms [2] are most commonly used. The problem becomes more complicated by the fact that coherent radar images (obtained by SAR) have their own very specific characteristics, and the most important among them is their distinctive spotting, so-called the speckle-noise [1]. Speckle significantly reduces effectiveness of the mentioned algorithms. In this paper, we consider the algorithm of linear structures detection based on the Hough transformation with subsequent analysis with on the basis of Kohonen

neural network. Application of the neural networks gives one a possibility of parallel processing and provides high noise immunity.

2 Classification algorithm

The proposed algorithm consists of several steps. At first, the Hough transformation is performed on a fragment of the original image $I(m, n)$, which transforms a pixel coordinates (m, n) into a Hough space $A(\rho, \theta)$

$$\rho = m \cos \theta + n \sin \theta.$$

where m and n are the point pixel Cartesian coordinates, and ρ and θ are the point polar coordinates.

As a result of this transformation, each line of the original image will correspond to a point in the Hough space (Fig. 1) For a binary image, the transformation is performed for non-zero pixels only. For a gray imagery, the transformation result values in the Hough plane are multiplied on the pixel value [4]. The optimal sampling interval of the parameter ρ is equal to one, and the parameter θ sampling each interval should not exceed 0.02π radians. Choice of the origin at the center of original image also helps to reduce errors in subsequent calculations.

Further, the image in the Hough plane is converted into an array of training vectors for a neural network. The easiest way is to match the value pair (ρ, θ) on the Hough plane with vectors $\boldsymbol{x}$, whose number for this cell will be proportional or equal to the $A(\rho, \theta)$ value for this cell. To improve convergence of the learning algorithm, it is better to take such number of vectors that will be proportional to $(A(\rho, \theta))^q$, $q = 1.5 \dots 2.0$ (Fig. 2, 3) [4].

The next step consists in normalization of vectors; this is performed by the formulas
$$x_1 = 0.7(2N_{\text{ang}}\theta_i - 1)/(N_{\text{ang}}/N_{\text{norm}}),$$
$$x_2 = 0.7(2N_{\text{norm}}\rho_i - 1),$$
$$x_3^2 = 1 - x_1^2 - x_2^2,$$

where N_{norm} and N_{ang} are the sizes of the Hough plane (in pixels). As a result, the vectors become three-dimensional.

Typically, in self-organizing networks each neuron is connected to all components of the input vector using synaptic connections. The weights of the neurons synaptic connections form a vector $\boldsymbol{w}$, which is needed to be initialized before the training. To reduce the number of "dead" neurons, it is better to use the uniform or random initialization. The number of neurons should be not less than the number of lines you want to detect. Usually, it is enough to take a few dozen, since too many neurons lead to unnecessary increase of computational costs and increase possibility of detecting false objects.

The next step is the self-organization (learning) of the competitive type neural network (Kohonen network). There are several algorithms for learning the self-organizing neural networks, such as the WTA algorithm (Winner Takes All),
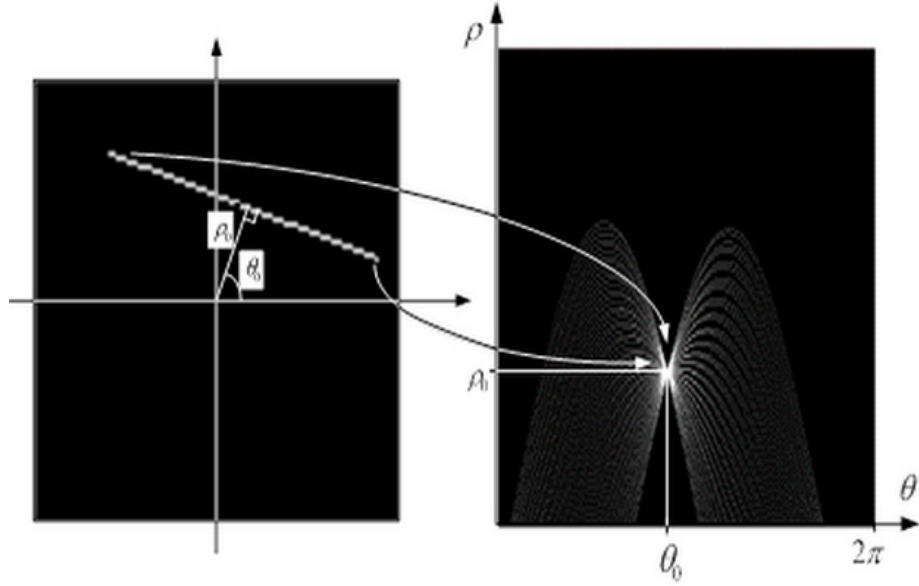
Fig. 1. Hough transformation for a linear object on a binary image

the WTM algorithm (Winner Takes Most), the neural gas algorithm, and others. These algorithms differ in the rate of convergences, in efficient use of neurons, and in implementation complexity of calculations at each iteration. In this work, the choice was made in favor of the neural gas algorithm with coordinate-wise metric ("Manhattan") [6]. The Manhattan metric has the form

$$d(\boldsymbol{x}, \boldsymbol{w}_i) = \sum_{i=1}^{N} |\boldsymbol{x} - \boldsymbol{w}_i|,$$

and requires a minimum computational cost among the other metrics. With this metrics, and the neural gas one algorithm has the best convergence among self-organization algorithms.

The weights adjustment in the neural gas algorithm is performed with the coefficient $G(i, \boldsymbol{x})$

$$G(i, \boldsymbol{x}) = \ exp\left(-\frac{m(i)}{\lambda}\right),$$

where $m(i)$ is the order of the neuron after sorting by the Manhattan distance. During the neural network training, the vectors $\boldsymbol{w}$ are sequentially supplied to the input. To improve the convergence, they are introduced to the network in a random order. Further, taking into account the chosen metric, the neurons sorting and neuron weight adjustment procedures are performed. Next, the process repeats until predetermined number of cycles is completed (Fig. 4). After that,
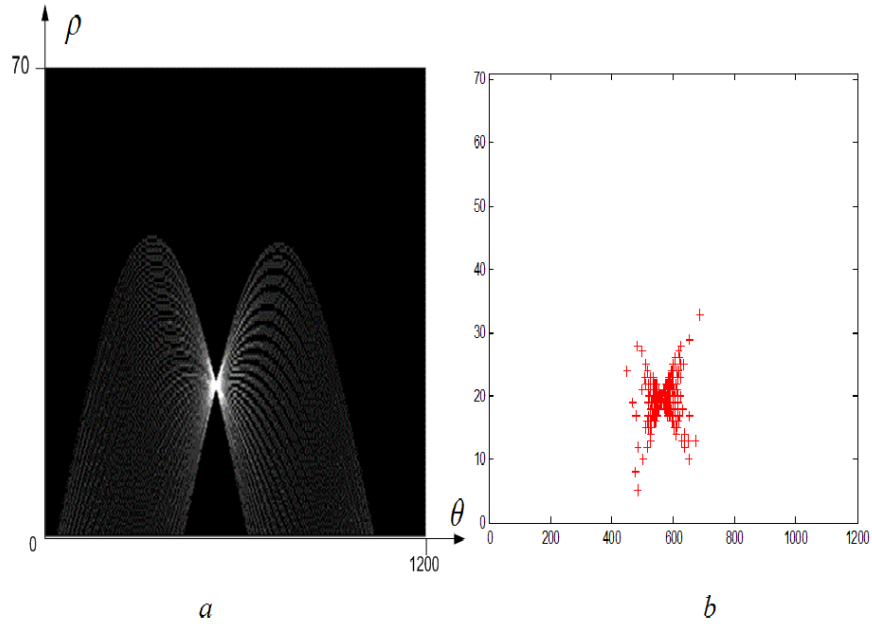
Fig. 2. Transformation of Hough plane into training vectors array: (a) Hough plane; (b) training vectors array

the weights fine adjustment procedure with an exponential calibration coefficient [6, 7] is recommended to be used

$$F(i, \boldsymbol{x}) = \ exp\,(-ad)\,,$$

where d is the Manhattan distance and a is an experimentally adjusted coefficient (it may decrease during network training).

As a result of the network self-organization and the weights fine adjustment, the most neurons group near the center of training vectors clusters, but some of them become "dead" or "wandering" (*i.e.*, they do not find any vectors cluster). Formed neurons groups are combined, and the "dead" and "wandering" neurons are discarded that leads to reduction of a number of detected lines.

The peculiarity of this problem is that the neural network works only in training mode. After the end of the training process, the weights of neurons are determined; then they are denormalized and descaled that result coordinates of desired straight lines. The inverse Hough transformation allows one to display these linear structures in the spatial coordinates.

The final processing stage is the weight statistical analysis of the pixels on the selected lines [5, 6]. It is used, at first, to find the borders of a linear structure segment if it passes not through the full image, and, secondly, to eliminate the
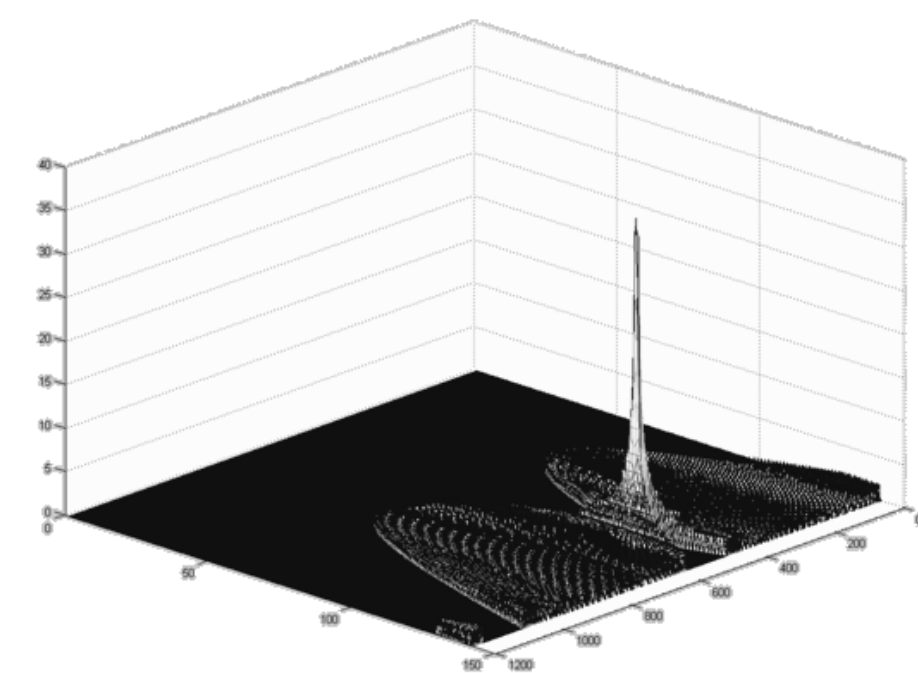
Fig. 3. Hough plane

remaining false lines. The first problem may be solved by the correlation analysis algorithms, and the second one is performed by the image brightness distribution analysis [5].

3 Experimental results

The developed algorithm has been tested both on models and on real radar images. Figure 5a shows an image obtained by the synthesized aperture radar RADARSAT-1 (spatial resolution is 8 m). In the picture, there are images of the four forest belts of different lengths, as well as, a bright fragment in the lower right corner. In addition, the fragment has a horizontal stripe of a medium brightness that is the image defect.

The resulting linear structures are presented on Fig. 5d. It is seen that all lines are detected correctly. A line truncation in the lower right corner takes place due to the image brightness decrease there.

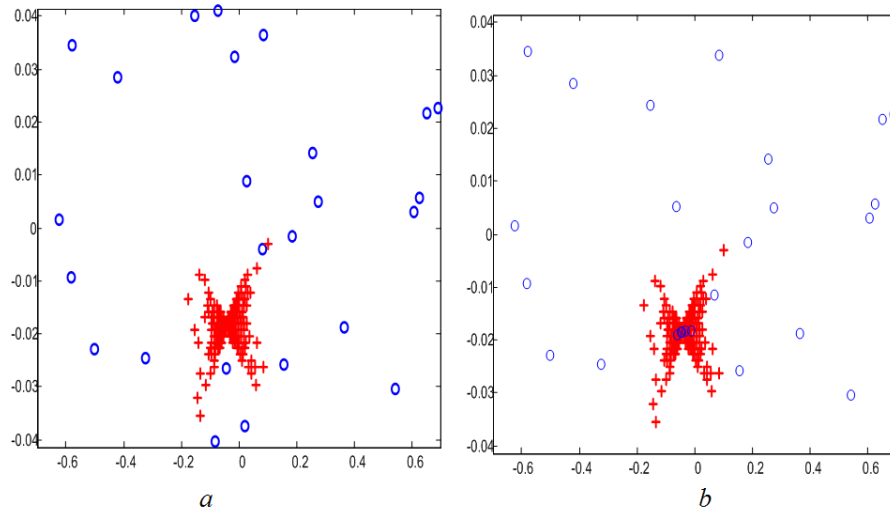Figure 6 demonstrates the RADARSAT-1 imagery processing results.

Fig. 4. Network training: (a) the beginning of the process (random weights initialization); (b) near the end of the process
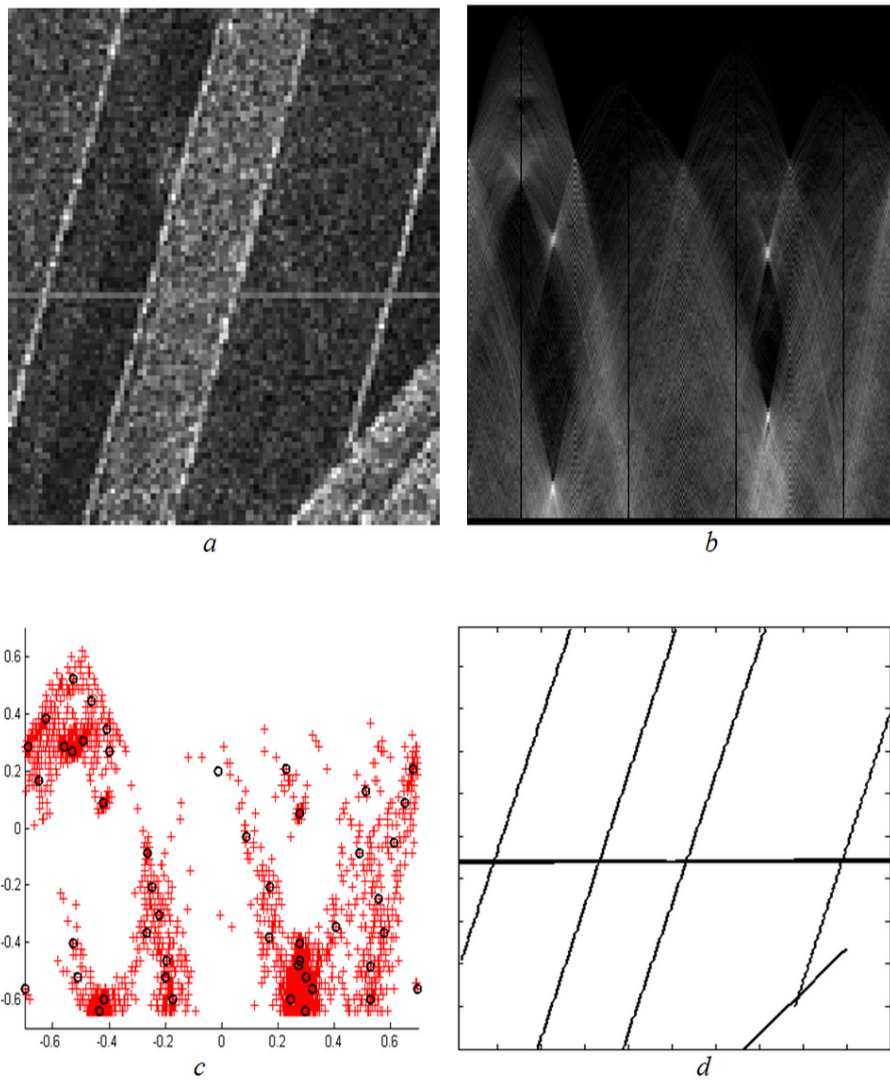

Fig. 5. RADARSAT-1 imagery processing results: (a) original scene; (b) Hough plane; (c) training vectors and neurons; (d) linear elements detecting process
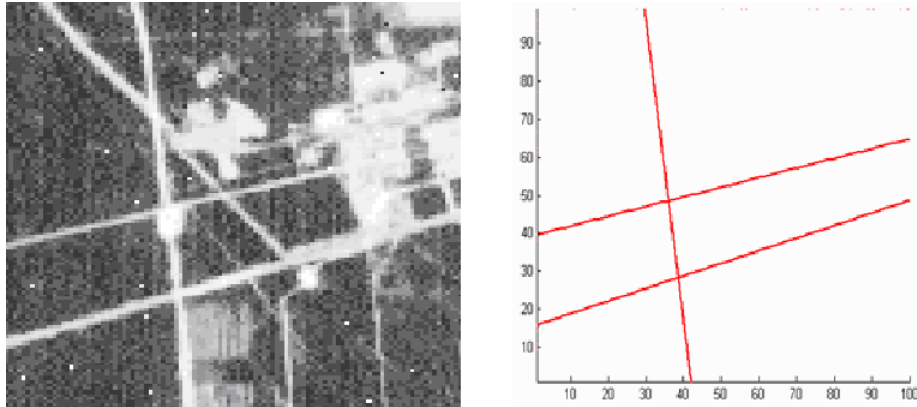
Fig. 6. LANDSAT-7 imagery processing results: (a) original scene; (b) linear elements detecting process

4 Conclusion

The proposed algorithm can significantly improve linear elements detection efficiency on radar images compared with traditional algorithms. It becomes possible due to the application of a neural network, which has enough low sensitivity to purity of the input data. Application of the neural network algorithm automates the process of linear structures detection due to the exclusion of thresholds selection from the algorithm.

5 Acknowledgment

References

1. Kondratenkov, G. S., Frolov A. Yu.: Radiovideniye. Radiolocatsionnye sistemy distansionnogo zondirovaniya Zemli [Radiovision. Radar systems for remote sensing of the Earth] (in Russian). Radiotekhnika, Moscow (2005)
2. Ivanov, O. Yu., Kobernichenko, V. G., Neronsky L. B.: Bystriy algoritm tsifrovogo sintezirovaniya aperturi [Fast digital aperture synthesis algorithm] (in Russian). Radiotekhnika, 1(1), 23–29 (1994)
3. Dorosinsky, L. G.: The research of the distributed objects on radar image recognition algorithms. Proc. of the CriMiCo 2013 23rd International Crimean Conference Microwave and Telecommunication Technology, Crimea. 1216–1218 (2013)

4. Pratt, W. K. Digital image processing. Mir, Moscow (1982)
5. Ivanov, O. Yu., Korkunov P. V., Sosnovsky, A. V.: Algoritm vydeleniya lineinykh struktrur na izobrazheniyakh, osnovanniy na preobrazovanii haffa [an algorithm for linear structures detection on the imagery based on hough transformation] (in Russian). Proc. of VII nauchnoprakticheskaya konferentsiya "Sviaz-Prom 2010", , Ekaterinburg. 321-326, (2010)
6. Ossovsky, S.: Neural networks for information processing. Finansy i statistika, Moscow (2004)
7. Duda, O., Hart, E.: Pattern classification and scene analysis. Mir, Moscow (1976)