Fast Algorithm for Maneuvering Target Detection in SAR Imagery Based on Gridding and Fusion of Texture Features

YUAN Zhan, HE You, CAI Fuqing
Research Institute of Information Fusion Technology, Naval Aeronautical and Astronautical University, 128 Erma Road, Yantai 264001, China

© Wuhan University and Springer-Verlag Berlin Heidelberg 2011

Abstract  Designing detection algorithms with high efficiency for Synthetic Aperture Radar (SAR) imagery is essential for the operator SAR Automatic Target Recognition (ATR) system. This work abandons the detection strategy of visiting every pixel in SAR imagery as done in many traditional detection algorithms, and introduces the gridding and fusion idea of different texture features to realize fast target detection. It first grids the original SAR imagery, yielding a set of grids to be classified into clutter grids and target grids, and then calculates the texture features in each grid. By fusing the calculation results, the target grids containing potential maneuvering targets are determined. The dual threshold segmentation technique is imposed on target grids to obtain the regions of interest. The fused texture features, including local statistics features and Gray-Level Co-occurrence Matrix (GLCM), are investigated. The efficiency and superiority of our proposed algorithm were tested and verified by comparing with existing fast detection algorithms using real SAR data. The results obtained from the experiments indicate the promising practical application value of our study.

Keywords  synthetic aperture radar imagery; target detection; texture feature; gridding; gray-level co-occurrence matrix; fusion

CLC number  P237

Introduction

Synthetic Aperture Radar (SAR) is a kind of active microwave imaging sensor and can generate high-resolution radar images. It has been widely used in the fields of remote sensing, mapping and reconnaissance in recent years because of its all-weather, all-day ability to provide ground images. Especially in the field of military surveillance, SAR increasingly plays an important role. Therefore, a task which is of particular importance is that of Automatic Target Recognition (ATR) for SAR imagery.

A typical SAR ATR system involves three processing stages: locating potential targets (detection), eliminating man-made clutter and natural clutter false alarms (discrimination) and classifying remaining detections by target type (classification). With the increasing amounts of image data collected by SAR sensors, designing high-efficiency ATR algorithms to satisfy the real-time requirements of operator SAR ATR systems is very important.

Target detection, as the first stage of ATR, aims to extract the Regions of Interest (ROIs) that most possi-
bly contain the potential targets in SAR imagery. There are two crucial factors we should take into consideration when designing detection algorithms: detection probability and efficiency. To guarantee that all targets of interest are detected out, the detection algorithms should possess high detection probability. To accelerate the speed of ATR, the detection algorithms must exhibit high detection efficiency. In the past, many detection algorithms for SAR imagery have been presented, for example, the constant false alarm rate (CFAR) detectors\cite{2-4} based on contrast or feature brightness; the multiresolution detector\cite{5} using the difference in fluctuation behavior of phase and amplitude from man-made objects and natural clutter under different resolution; the two-feature EF/CFAR detector\cite{6} utilizing both the contrast and extended fractal features. Up to now, target detection for SAR imagery is still an active area of research.\cite{7, 8} Although much effort has been made to improve the detection performance, it is a hard yet very meaningful task to detect targets in a large scene image with high detection probability and efficiency at the same time.

Analyzing the detection methods, we find that most of them use this strategy to detect targets of interest: visiting every pixel in SAR imagery. In general, the clutter pixels in a SAR image dominate a majority part of the image and the pixels of target of interest are just a small amount. So visiting every pixel in original image would necessarily consume a lot of time. If we can quickly locate the possible regions containing targets of interest, and then extract ROIs based on these regions, the detection time can be reduced a lot. Motivated by this idea, in this paper, we propose a fast detection scheme based on the idea of gridding. The original SAR image is divided into a series of grids which stand for clutter regions or target regions. Considering the different texture presented by natural clutter and cultural target, we utilize texture features to identify these clutter regions and target regions. By fusing the identifying results of every texture feature, the final target regions are obtained. Finally, we apply the dual threshold segmentation technique to the target regions to obtain the ROIs. In contrast to CFAR detector, this scheme need not assume any statistical models for background clutter or calculate the test statistics, so its computational complex is much less than that of CFAR detector, and the runtime of detection is sped up a lot.

1 Texture features and SAR image texture

1.1 Texture features

Many methods have been presented for the computation of image texture features. Kandaswamy et al.\cite{9} summarized some of them, for example, the Gray-Level Co-occurrence Matrix (GLCM), Markov Random Fields (MRFS), Gabor wavelets, and tree-structured wavelets. Moreover, local statistics features are also used to analyze image texture for their computational efficiency. To test different texture features' ability to identify the target regions fast, in this study, we investigate the local statistics features and features derived from GLCM. These texture features are described briefly as follows.

1) Local statistics features

The local statistics features provide information related to the gray level distribution of the image. Let $I$ be the random variable representing the gray levels in the region of interest. The local statistics histogram $H(I)$ is defined as

$$H(I) = \frac{\text{number of pixels with gray level } I}{\text{total number of pixels in the region}}$$  (1)

Based on the histogram $H(I)$, we define:

Moments:

$$m_i = E(I^i) = \sum_{I=0}^{N-1} I^i H(I)$$  (2)

Central moments:

$$\mu_i = E[(I - E(I))^i] = \sum_{I=0}^{N-1} (I - m_i)^i H(I)$$  (3)

With different $i$ in Eq. (3), we obtain:

- Skewness ($SKE$),
  $$\mu_3 = E[(I - E(I))^3] = \sum_{I=0}^{N-1} (I - m_i)^3 H(I)$$ is a measure of the degree of histogram asymmetry around the mean;

- Kurtosis ($KUR$),
  $$\mu_4 = E[(I - E(I))^4] = \sum_{I=0}^{N-1} (I - m_i)^4 H(I)$$ is a measure of the histogram sharpness.

2) Features derived from GLCM
The features derived from GLCM provide information about the relative positions of the various gray levels within the image region. Let \( d \) be the relative distance measured in pixel numbers and \( \alpha \) be the relative orientation \((\alpha = 0^\circ, 45^\circ, 90^\circ, 135^\circ)\). The co-occurrence matrix is defined as a \( N_g \times N_g \) matrix, and its \((i,j)\)th elements, \( P_{i,j} \), is defined as

\[
P_{i,j} = \frac{p_{i,j}^{d,\alpha}}{\sum_{i,j} R} 
\]

where \( p_{i,j}^{d,\alpha} \) is the frequency of pixel pairs, at relative position \((d, \alpha)\), with gray-level values \( i \) and \( j \), respectively; \( R \) is the total number of possible pairs.

Many texture features such as angular second moment, correlation, and variance have been derived from the GLCM. Three representative ones are:

- **Contrast (CON):**
  \[
  CON = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} |i - j|^2 P_{i,j} 
  \]

- **Difference Average (DA):**
  \[
  DA = \sum_{i=0}^{N_g-1} i P_i (i) 
  \]

- **Difference Variance (DV):**
  \[
  DV = \sum_{i=0}^{N_g-1} (i - DA)^2 P_i (i) 
  \]

where \( P_i (k) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} P_{i,j} \).

Calculation of GLCM is well known for being time consuming, and many fast algorithms to calculate GLCM have been presented. Bastos et al.\[10\] proposed a method based on an indexed list for fast computing GLCM.

1.2 Analysis of SAR image texture

The texture of an image region depends on the distribution of gray levels of pixels in this region. Concretely, it depends on the spatial relation between primitive texture elements, their scale, and/or orientation.\[9\] Texture contains important information about the structural arrangement of object surfaces and their relationship to their neighboring surfaces. It is not only a valuable feature to discriminate different land types,\[9\] but also an important property in detecting targets in SAR images,\[11\] due to its spatial, scale and orientation properties.

The texture is also fit to classify objects in SAR imagery. As we know, SAR has the ability to generate high-resolution images of earth or sea ground. For maneuvering targets such as howitzers, armored vehicles and tanks, they are no longer a single point and occupy several pixels in SAR imagery. Each such pixel reflects the Radar Cross Section (RCS) or backscattering coefficient of a part of the target, and the variation of target pixel’s RCS reflects the spatial fluctuation properties of the RCS in terms of the entire target. This fluctuation reveals the spatial configuration and its intrinsic variation of the surface of a maneuvering target.

There is no doubt that the surface of a maneuvering target is very different from that of the natural clutter such as forest, farmland and so on. Firstly, maneuvering targets are usually composed of metallic materials whose dielectric constant is different from that of natural clutter. Secondly, a maneuvering target’s surface consists of some regular dihedrals and trihedrals such as corners, edges and flat plates which can lead to greater fluctuation of RCS compared with the natural clutter. Combined with these factors, maneuvering targets will present unique texture features that distinguish it from the natural clutter. Hence, it seems promising that texture features can be utilized to detect targets we are interested in.

2 The proposed fast detection algorithm

2.1 Grid the original SAR image

One disadvantage of the CFAR detection scheme is that it screens every pixel in the image to judge whether the Cell Under Test (CUT) is a target pixel or not based on a local test statistic. This detection strategy is very time-consuming when the original image is very large and the computation of test statistic is very complex, for example, the test statistic of \( K \) distribution.\[3\] In practice, the clutter pixels are a major part of the image and the amount of target pixels is very small. When detecting targets of interest, we only need to find out the regions that most possibly contain the targets from the image, and then de-
tect targets based on those regions in order to improve the efficiency of detection. On the other hand, the texture features are usually calculated based on an image area. To utilize the texture features to detect targets, we have to partition the original SAR image into a series of small areas based on which the texture features are calculated. Gridding the image is an approach to realize the partition of image, which segments the image into a series of grids with fixed size (as shown in Fig.2). Each grid represents a clutter region which does not contain targets, or a target region which contains potential targets, depending on the texture inside it. In contrast to segmentation-based detection techniques,[12] gridding the image is not to obtain a homogenous area with similar back-scattering behavior.

2.2 Identify clutter regions and target regions

For each grid, if it contains maneuvering targets, it will present unique texture features which are different from those presented when it does not contain maneuvering targets. Consequently, we can judge whether the grid contains the maneuvering targets or not according to its texture features. We see this judging process as a classification problem and design certain discriminative criterion to classify the grid into two types: target region and clutter region. We design a discriminative criterion for local statistics features and GLCM features.

\[
\begin{align*}
  \text{if } F_{grid_i}(\cdot) > T_r, \text{ then } & grid_i \in \omega_r, \\
  \text{if } F_{grid_i}(\cdot) < T_r, \text{ then } & grid_i \in \omega_c. 
\end{align*}
\]

where \(grid_i\) is the \(i\)th grid of the image; \(F(\cdot)\) represents the operator that computes the texture feature of \(grid_i\); \(\omega_r\) and \(\omega_c\) represent the target class and the clutter class, respectively; \(T_r\) is a threshold.

The main point of difficulty when implementing the discriminative criterion above is how to determine the threshold. To be sure, the threshold is a critical factor in classifying the grids correctly. Fortunately, in these experiments, we find that the GLCM texture features of the target region are greater to an extent than those of the clutter region (as shown in Fig. 3). Therefore, there is no difficulty determining the threshold to distinguish a target region from a clutter region from GLCM texture features. There is no remarkable distinction, however, between target region and clutter region in terms of kurtosis and skewness features (as shown in Figs. 4-5), which makes it difficult to determine the threshold for these two features. In a practical system, we can compute the typical clutter regions’ texture features and store them in a look-up table, and then select an appropriate threshold based on the detection environment of the time. Note that one texture feature alone above can be used to identify the region.

2.3 Fuse identifying results of every texture features

In general, only using one texture feature to identify target regions may produce large amount of false alarms due to the inappropriate selection of threshold. But by fusing every texture feature’s identifying result, i.e. (8), we can keep the number of false alarms to the minimum. Considering that calculation of \(DV\) and \(DA\) is very time-consuming, we select the \(SKE\), \(KUR\) and \(CON\) and fuse their identifying results.

Let \(T_S\), \(T_K\), and \(T_C\) be the thresholds of \(SKE\), \(KUR\) and \(CON\), respectively. For every grid, whether it belongs to the target region depends on the following fusion rule

\[
\begin{align*}
  \text{if } & F_S(grid_i) > T_S \text{ and } F_K(grid_i) > T_K \text{ and } F_C(grid_i) > T_C, \text{ then } grid_i \in \omega_r, \\
  \text{else } & grid_i \in \omega_c. 
\end{align*}
\]

where \(F_S(\cdot), F_K(\cdot), \text{ and } F_C(\cdot)\) represent the operators that compute \(SKE\), \(KUR\) and \(CON\) of \(grid_i\), respectively.

2.4 Extract ROIs using dual threshold segmentation technique

Once we have obtained the target regions, we can use various detection algorithms such as the CFAR detection algorithm to extract ROIs in the target regions. To obtain the fine sharp features of original targets which are very helpful for target pose estimation,[13] we used the image segmentation technique to extract the ROIs. In target regions, the target pixels are usually brighter greatly than the clutter pixels; and these bright target pixels always collect together, therefore forming a shine region, whereas the bright clutter pixels are often isolated points and
randomly distributed throughout the whole target region, which makes it possible to exploit the threshold segmentation technique for extracting the ROIs.

Owing to the coherent imaging principle, a SAR image is affected severely by speckle noise. Therefore in the target regions, there exist some clutter pixels whose intensity may be very high. But, only using a single threshold to detect the target may lead to detecting out these clutter pixels. Considering that bright target pixels always collect together, we use the dual threshold segmentation technique to detect target pixels. Firstly, we set a high threshold \( T_1 \). Pixels with a gray level that is higher than \( T_1 \) are denoted as central pixels. This step guarantees that the brightest target pixels are detected out. Secondly, we set a threshold \( T_2 \) \( (T_2 < T_1) \) and a distance threshold \( d_0 \). Pixels whose gray level is greater than \( T_2 \) and where distance from any one central pixel is smaller than \( d_0 \) are denoted as target pixels. Consequently, the clutter pixels with high gray level and long distance from central target pixels would not be detected out as target pixels. The procedures of determining two thresholds \( T_1 \) and \( T_2 \) are as follows.

Let \( I \) be the random variable representing image pixel, and \( p(I) \) be the clutter statistical model. \( \alpha_i, i = 1, 2 \) are two given a priori values standing for the proportions of clutter pixels in whole image, and thus \( 1 - \alpha_i, i = 1, 2 \) represent a pixel’s confidence probability of belonging to target pixels. The thresholds \( T_1 \) and \( T_2 \) satisfy the following formula

\[
p(I > T_i) = 1 - \alpha_i, i = 1, 2
\]

Solve (9) and \( T_i, i = 1, 2 \) can be obtained. For Gaussian statistical model, we have

\[
p(I > T_1) = \int_{T_1}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(I - \mu)^2}{2\sigma^2})dI
\]

\[
= 1 - \alpha_i, i = 1, 2
\]

\[
T_1 = \sqrt{2\sigma} \text{erfcinv}(2(1 - \alpha_1)) + \mu, i = 1, 2
\]

where \text{erfcinv}(\cdot) is the error function; \( \mu \) and \( \sigma^2 \) are the mean and variance of Gaussian statistical model, respectively.

Fig. 1 presents the flowchart of our proposed fast detection algorithm.

### 3 Experiment results and discussion

For these experiments, we selected two high-resolution SAR images I and II (as shown in Fig.2 and Fig.8, respectively) to establish the detection performance of the proposed detection scheme against two existing CFAR-based fast detection algorithms proposed by Gao\textsuperscript{[14]} and Salowe\textsuperscript{[15]}, respectively. The false alarm is defined as a detected out clutter grid in this study. Two images are imaged at X-band and VV polarization with resolution of 0.5 m×0.5 m and 3 m×3 m, size of 580×436 and 700×260, respectively. Six grid sizes 10×10, 15×15, 20×20, 25×25, 30×30, 40×40 are used to evaluate the detection probability and runtime of our proposed algorithm.

The parameters to be used in Gao’s algorithm and Salowe’s algorithm are as follows.

- The background clutter area size and guard area size are 8×8 and 10×10, respectively, and the false alarm rate is set to \( P_{fa} = 10^{-4} \) for two algorithms. The threshold used by Salowe’s algorithm is determined according to Eqs.(10)-(12).

All experiments are operated on a 1.69-GHz AMD D CPU with 2-GB RAM.

- Experiment with image I

In this experiment, the targets to be detected in
image I are marked by white ellipse. First of all, we grid the original SAR image, and a series of grids are produced as shown in Fig.2. Then, we calculate the texture features in every grid. Figs. 3-5 show the values of three texture features in every grid in the case of 30×30. Based on the distinctness of texture features of clutter and maneuvering targets, an appropriate threshold for every texture feature can be determined from their corresponding figures. Take the texture feature $CON$ for example, in the simulation; we found there exists a grid whose $CON$ value is quite larger than that of any other grids regardless of the size of the grid. Hence, a reasonable threshold for $CON$ when the grid size is 30×30 ought to be around 2000.

Fig.6 is the extracted target grid by our proposed algorithm where the targets to be detected are located. Fig.7 is the obtained ROIs by imposing dual threshold segmentation technique on target grid, i.e. Fig.6. As far as the detection probability is concerned, Fig.12 gives the number of false alarms of our presented algorithm with different grid sizes when all target grids are detected out. The maximum number of false alarms is three, and with the grid size increasing, the number of false alarms gradually decreases to zero.

Table 1 gives the comparison of runtime of three fast detection algorithms. As we can see from the table, our proposed algorithm runs faster than Gao’s algorithm and Salowe’s algorithm regardless of the grid size. Especially when the grid size is 30×30 or 40×40, the proposed algorithm achieves high detection probability and run speed at the same time.
Table 1  Detection time used by three fast
detection algorithms with image I

| Detection scheme       | Grid size | Runtime (s) |
|------------------------|-----------|-------------|
| 10×10                  | 10.017    |
| 15×15                  | 5.041     |
| The proposed           | 20×20     | 3.158       |
| detection algorithm    | 25×25     | 2.378       |
|                        | 30×30     | 1.886       |
|                        | 40×40     | 1.277       |
| Gao’s algorithm        |           | 21.815      |
| Salowe’s algorithm     |           | 31.897      |

- Experiment with image II

In this experiment, the white rectangle in Fig.8 indicates the targets to be detected. Fig.9 shows the detection results of our proposed algorithm with grid size 20×20, from which we can see all targets of interest are detected out and five clutter grids are yielded. Fig.10 and Fig.11 give the detection results of Gao’s algorithm and Salowe’s algorithm, respectively. It is clear that besides the true targets of interest, many false alarms are detected out which are mainly from the trees with comparable reflection intensity to the true targets.

Table 2  Detection time used by three fast
detection algorithms with image II

| Detection scheme       | Grid size | Runtime (s) |
|------------------------|-----------|-------------|
| 10×10                  | 7.274     |
| 15×15                  | 3.467     |
| The proposed           | 20×20     | 2.326       |
| detection algorithm    | 25×25     | 1.942       |
|                        | 30×30     | 1.280       |
|                        | 40×40     | 0.959       |
| Gao’s algorithm        |           | 16.374      |
| Salowe’s algorithm     |           | 18.279      |

Fig. 11  Detection results by Salowe’s algorithm

Fig.12 gives the detection results with other grid size cases. Unlike experiment I, in this simulation, the number of false alarms changes greatly as the grid size changes. This is partly because the clutter environment confronted by the proposed algorithm is more complex. However, as long as the grid size is selected appropriately, a high detection probability can also be achieved.

Table 2 indicates the runtime of three algorithms, and from the table, the same conclusions demonstrated in experiment I can be obtained.

The two experiments indicate that the larger the grid size, the less the time used by five texture features. But the grid size can’t be set too large, or more clutter pixels will be incorporated into a target grid, which leads to little difference of texture feature be-
tween clutter grids and target grids, and consequently results in production of more false alarms, as shown in Fig. 12.

4 Conclusion

In this paper, we presented a fast detection scheme to extract ROIs in high-resolution SAR imagery. The appeal of this scheme is that it abandons the strategy of searching for objects in the entire image, as done by many detection algorithms, so the detection efficiency is improved a lot. Moreover, the presented algorithm has the characteristics of low computational complexity, high detection probability and easy realization, which makes it a promising detection technique for engineering applications in SAR ATR systems. We also have shown the feasibility of incorporating texture features into target detection for SAR imagery, as presented by Ohanian et al. [11] with theory analysis and simulation validation. Furthermore, our scheme is different from that proposed by Gao, et al. [16] who use a global instead of a local CFAR threshold to fast locate the targets. Although the experiments with limited SAR data verified our scheme, we need to test this scheme with more real SAR data collected under various conditions.

As we can see, the efficiency of the proposed scheme is very sensitive to the grid size. It is apparent that the grid size plays an important role in reducing runtime and correctly identifying the target region. Note that besides the local statistics features and GLCM features, there exist many other texture features. Therefore, we can fuse more texture features in the algorithm to achieve better performance.

References

[1] Novak L M, Halversen S D, Owirka G J, et al. (1997) Effects of polarization and resolution on SAR ATR[J]. IEEE Transactions on Aerospace and Electronic Systems, 33(1): 102-116

[2] Biscoglie M D, Galdi C (2005) CFAR detection of extended objects in high-resolution SAR images[J]. IEEE Transactions on Geoscience and Remote Sensing, 43(4): 833-843

[3] Kuttikkad S, Chellappa R (1994) Non-Gaussian CFAR techniques for target detection in high-resolution SAR images[C]. Proceedings of the IEEE International Conference on Image Processing, New York

[4] Banerjee A, Burlina P, Chellappa R (1999) Adaptive target detection in foliage-penetrating SAR images using alpha-stable models[J]. IEEE Transactions on Image Processing, 8(12): 1823-1831

[5] Subotic N S, Collins L M, Gorman J D, et al. (1995) Multiresolution target detection in SAR imagery[C]. Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal, New York

[6] Kaplan L M (2001) Improved SAR target detection via extended fractal features[J]. IEEE Transactions on Aerospace and Electronic Systems, 37(2): 436-451

[7] Marques R R, de Medeiros F N S, Ushizima D M (2009) Target detection in SAR images based on a level set approach[J]. IEEE Transactions on Systems, Man, and Cybernetics, 32(2): 214-222

[8] Wang Changcheng, Liao Mingsheng (2009) A target detection method on multi-aperture SAR images[J]. Geomatics and Information Science of Wuhan University, 34(1): 32-35

[9] Kandaswamy U, Adjeroh D A, Lee M C (2005) Efficient texture analysis of SAR imagery[J]. IEEE Transactions on Geoscience and Remote Sensing, 43(9): 2075-2083

[10] Bastos L deO, Liatsis P, Conci A (2008) Automatic texture segmentation based on k-means clustering and efficient calculation of co-occurrence features[C]. Proceedings of the 15th IEEE International Conference on Systems, Signals, and Image Processing, New York

[11] Ohanian P P, Dubes R C (1992) Performance evaluation for four classes of texture features[J]. Pattern Recognition, 25(8): 819-833

[12] Lombardo P, Sciotti M (2001) Segmentation-based technique for ship detection in SAR images[J]. IEE Proceeding Radar, Sonar & Navigation, 148(3): 147-159

[13] Sun Y J, Liu Z P, Todorovic S, Li J (2007) Synthetic aperture radar automatic target recognition using adaptive boosting[J]. IEEE Transactions on Aerospace and Electronic Systems, 43(1): 112-125

[14] Gao Gui, Zhang Jun, Zhou Diefei, et al. (2008) The automatic acquirement of targets’ region-of-interest from SAR for the application of Earth observation[J]. Advance in Earth Science, 23(1): 39-46

[15] Salowe J S (1996) Very fast SAR detection[C]. Proceedings of SPIE-The International Society for Optical Engineering, Orlando

[16] Gao Gui, Kuang Gangyao, Zhang Qi, et al. (2007) Fast detecting and locating groups of targets in high-resolution SAR images[J]. Pattern Recognition, 40: 1378-1384