

An Improved Method for Detection of Stationary Targets in High Clutter SAR Images

John Weatherwax

Abstract—In a series of papers [1]–[6] L. Novak and others at MIT Lincoln Laboratory developed an Automatic Target Recognition (ATR) system for identifying stationary ground targets appearing in Synthetic Aperture Array (SAR) RADAR images. The first step of that processing involves object detection and is done with an image filtering operation aimed at estimating statistics of the clutter surrounding each pixel within the SAR image. The original detector proposed by Novak et. al. works very well, when targets are separated by a reasonable distance or the background clutter returns are not too severe. In this paper, we present an alternative detector algorithm that has better detection performance with densely spaced targets or targets in high clutter environments. Following a description of our algorithm we comment on efficient computational implementations, and finally present comparison studies showing the stated improved performance.

I. INTRODUCTION

IN a series of papers [1]–[6] L. Novak and others at MIT Lincoln Laboratory developed an Automatic Target Recognition (ATR) system for identifying stationary ground targets appearing in Synthetic Aperture Array (SAR) RADAR images. It was envisioned that this software would enable very large regions of terrain to be searched automatically. Here we briefly present a summary of enough of the signal processing chain to enable the reader to understand the improvements developed in this paper. Please see the series of papers written by Novak and others for a more complete characterization.

The ATR signal processing chain developed by Novak and others is as follows, see figure 1 for a flow diagram. A SAR images is collected and passed into the system. A two parameter CFAR detection algorithm [7] is used to locate pixels that are significantly bright compared to their surrounding clutter. This CFAR image is thresholded and each connected component in the resulting binary image is considered a potential target. For each potential target a few simple features are calculated and “obvious” non-targets are mitigated (discarded). Obvious non-targets include ones that are much too large or small to represent a typical target. For the potential targets that remain, a set of nine features is calculated. This feature vector is compared with known clutter and target mean feature vectors using a Gaussian quadratic classifier to further eliminate clutter detections. Finally, any remaining detections are compared to template targets from

Manuscript received January 1, 2005; revised December 30, 2005. This work was sponsored by the U.S. Government under Air Force Contract F19628-00-C-0002. Options, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

J. Weatherwax is with Lincoln Laboratory, Massachusetts Institute of Technology.

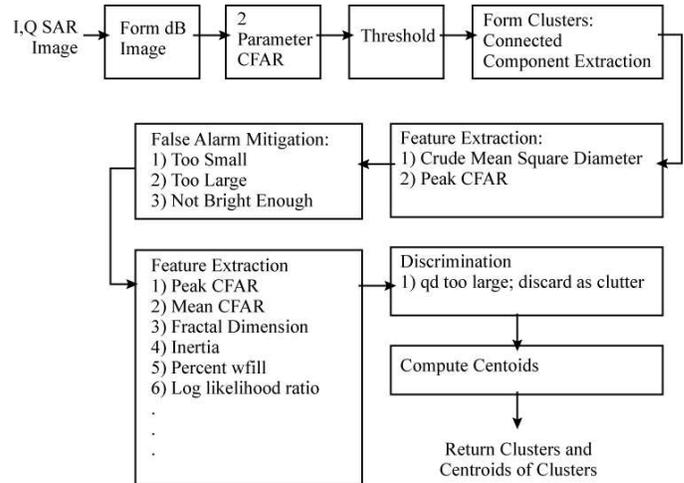


Fig. 1. High level overview of the SAR image processing proposed in [4]. As this paper is concerned only with the processing that happens in the box labeled “2 Parameter CFAR”, only that calculation will be discussed in any detail. Please see the original articles for more information on the other components.

a target image database to further classify each detection by target type. A nice summary of much of Novak’s work in this area is given in the recent book [8].

An intricate and important part of the entire processing chain just described is the CFAR detector, represented in figure 1 as the “2 Parameter CFAR” box. In this paper, we will discuss a CFAR detector algorithm that gives better performance for densely spaced targets and for targets in heavy clutter regions than that originally proposed by Novak et. al. Towards this end, the remainder of this paper is as follows. In section II we describe the original Novak CFAR detector and a median CFAR detector developed to overcome the difficulties previously mentioned. To the authors knowledge this median CFAR approach is novel and is one of the contribution from this paper. Next, in section III we describe novel modifications to each CFAR algorithm that improve computational efficiency without sacrificing detection performance. In section IV, we demonstrate the improved performance on some sample Moving and Stationary Target Acquisition and Recognition (MSTAR) SAR images. Finally, in section V we conclude.

II. CFAR DESCRIPTIONS

In this section we first review the baseline CFAR algorithm presented by Novak in [1] and then present a modified CFAR algorithm based on the original version but using order statistics to compute clutter statistics.

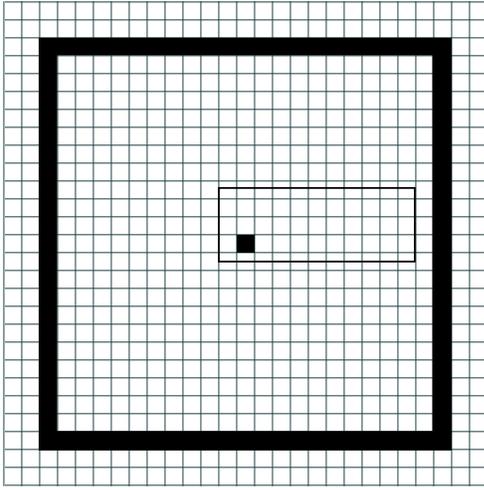


Fig. 2. The original CFAR stencil [1]. The border pixels are chosen to be far enough from the center so that a target of interest will not intersect them. The highlighted pixel at the center is the pixel that the clutter estimate is computed for.

A complex SAR image ($C_{i,j}$) is first transformed into a log-detection image ($D_{i,j}$) with elements defined by

$$D_{i,j} = 10 \text{Log}_{10}(C_{i,j}C_{i,j}^*), \quad (1)$$

where the star represents complex conjugation. This operation is performed for every pixel in the input SAR image, or $1 \leq i \leq M$ and $1 \leq j \leq N$, for a SAR image of dimension $M \times N$. This log detection image D is then further processed with a moving average stencil. For each log-detection pixel $D_{i,j}$ in the image, the algorithm consists of computing estimates of the local-clutter statistics from pixels *around* the pixel of interest. The local clutter statistics required at pixel location (i, j) are the mean $\hat{\mu}_C$ and the standard deviation $\hat{\sigma}_C$ of the shaded border pixels in figure 2. The formulas for these estimates are obtained using standard statistical formulas for the mean and standard deviation [9]. For the mean this equation is

$$\hat{\mu}_C = \frac{1}{N_{\text{Stencil}}} \sum_{(i,j) \in \text{Stencil}} D_{i,j}, \quad (2)$$

while for the standard deviation this equation is

$$\hat{\sigma}_C^2 = \frac{1}{N_{\text{Stencil}}} \sum_{(i,j) \in \text{Stencil}} (D_{i,j} - \hat{\mu}_C)^2. \quad (3)$$

Here the sums are taken over each (i, j) pixel that is a member of the stencil and N_{Stencil} is the number of pixels in the CFAR stencil. A typical stencil used for this filtering is shown in figure 2. The size is chosen large enough so that a target of interest, centered at pixel (i, j) will not intersect its own clutter ring. In addition the clutter statistics $\hat{\mu}_C$ and $\hat{\sigma}_C$ are computed for every pixel in the input image¹.

With these two definitions the CFAR image at pixel (i, j) is computed using the following equation

$$\text{CFAR}_{i,j} = \frac{D_{i,j} - \hat{\mu}_C}{\hat{\sigma}_C}. \quad (4)$$

¹They are not be written with (i, j) indices to avoid the resulting cumbersome notation.

This equation then physically represents the number of standard deviations each pixel has from a locally computed clutter background. A true target located at (i, j) should be much brighter than the surrounding clutter and have correspondingly large values for $\text{CFAR}_{i,j}$. To apply this process to *every* pixel in the SAR image boundary condition specification will be necessary. A great number of different methods can be used for this purpose. As the exact method used is not important for this paper this issue will not be discussed further.

Were the full ATR processing described in figure 1 to continue, this CFAR image would now be thresholded and the connected components that remain in the resulting binary image represent the location of potential targets. For this paper, we will only be concerned with the production of a CFAR image and it will not be necessary to discuss further processing.

During development, it was noted that over bodies of water or regions of very uniform clutter where $\hat{\sigma}_C \ll 1$, the following simple modification to equation 4 resulted in an algorithm that performed better with respect to false alarms

$$\text{CFAR}_{i,j} = \frac{D_{i,j} - \hat{\mu}_C}{\max(\hat{\sigma}_C, \sigma_{\text{Floor}})}. \quad (5)$$

Here σ_{Floor} is an empirically derived constant.

The CFAR algorithm given by equations 2, 3, and 5 suffers from the problem that when targets are closely spaced, so that they intersect each others clutter stencils, the neighboring targets RADAR return will incorrectly corrupt the statistical features computed in equations 2 and 3. This corruption will result in a clutter mean value that is too large and the resulting failure to detect closely spaced targets. In addition, targets that are adjacent to objects that posses strong RADAR returns i.e. trees or cultural clutter will also fail to be detected. Examples of this type of behavior will be presented in section IV.

To correct for these difficulties, an alternate algorithm was developed. The alternate algorithm uses the same stencil as in figure 2, but performs the statistical calculation using rank order statistics. In this case the clutter mean is computed with the following expression,

$$\hat{\mu}_C = \text{Median}_{(i,j) \in \text{Stencil}} D_{i,j}. \quad (6)$$

In the next subsection we present the method used to calculate $\hat{\sigma}_C$.

A. Calculation of the Clutter Standard Deviation $\hat{\sigma}_C$

The estimate of the clutter standard deviation will be presented in this subsection. For all pixel values in the stencil region we first construct a cumulative distribution function (CDF). This is a function F such that [9]

$$F(x) = \text{Pr}\{X \leq x\}. \quad (7)$$

Where Pr stands for probability. This function is only defined for pixel values in the clutter ring, an example of which is given in figure 2. In words, we require that the probability a pixel value in the clutter ring is less than x be given by $F(x)$. A typical example CDF for a stencil given by figure 2 is given in figure 3. With the notion of a CDF we can define empirical

percentile pixel values. In this paper we will only work with two percentiles x_l and x_r . For a fraction f with $0 \leq f \leq 1$, the value x_f is defined using the CDF F as

$$F(x_f) = f. \quad (8)$$

Here f in the above equation can symbolically represents l and r . Thus, $x_{0.5}$ represents the median value of the clutter. Using our empirically obtained CDF we can compute fractional values x_f . To relate the calculated fractional values with a standard deviation estimate we assume a Gaussian distribution of clutter giving the following analytical expression for the CDF [9]

$$F(x) = \frac{1}{2} \left[1 + \operatorname{erf} \left(\frac{x - \tilde{\mu}}{\sqrt{2}\hat{\sigma}_C} \right) \right]. \quad (9)$$

Where erf is the error function, defined by

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-u^2} du. \quad (10)$$

With this background the algorithm used to compute $\hat{\sigma}_C$ is then given by the following steps. First, a constant numerical parameter q ($0 < q < 1$) is chosen for the calculation of two empirical symmetrical fractional values (x_l and x_r) given by

$$x_l = x_{\frac{q}{2}} \quad \text{and} \quad x_r = x_{1-\frac{q}{2}}. \quad (11)$$

With this specification, the amount of probability between the values x_l and x_r is $1 - q$. Mathematically, this is represented by

$$F(x_r) - F(x_l) = 1 - q. \quad (12)$$

Upon inserting equation 9 we obtain

$$\frac{1}{2} \operatorname{erf} \left(\frac{x_r - \tilde{\mu}}{\sqrt{2}\hat{\sigma}_C} \right) - \frac{1}{2} \operatorname{erf} \left(\frac{x_l - \tilde{\mu}}{\sqrt{2}\hat{\sigma}_C} \right) = 1 - q. \quad (13)$$

Using the oddness property of the error function

$$\operatorname{erf}(-x) = -\operatorname{erf}(x), \quad (14)$$

inside the second term on the left hand side of the above equation we obtain

$$\frac{1}{2} \operatorname{erf} \left(\frac{x_r - \tilde{\mu}}{\sqrt{2}\hat{\sigma}_C} \right) + \frac{1}{2} \operatorname{erf} \left(\frac{\tilde{\mu} - x_l}{\sqrt{2}\hat{\sigma}_C} \right) = 1 - q. \quad (15)$$

Now remembering that $x_l < \tilde{\mu} < x_r$, and the symmetry with which we choose x_l and x_r we note that

$$\tilde{\mu} - x_l = x_r - \tilde{\mu} = \frac{1}{2}(x_r - x_l), \quad (16)$$

and therefore we can substitute $\frac{1}{2}(x_r - x_l)$ inside each argument in equation 15 and remove all dependence on the unknown mean $\tilde{\mu}$. When this is done and the expression simplifies and the solution for $\hat{\sigma}_C$ is

$$\hat{\sigma}_C = \frac{x_r - x_l}{2\sqrt{2} \operatorname{erf}^{-1}(1 - q)}. \quad (17)$$

Here we have used erf^{-1} to represent the inverse of the function defined in equation 10.

At this point, with equations 6 and 17 we are then able to use equation 5 to produce a CFAR image from the input SAR image.

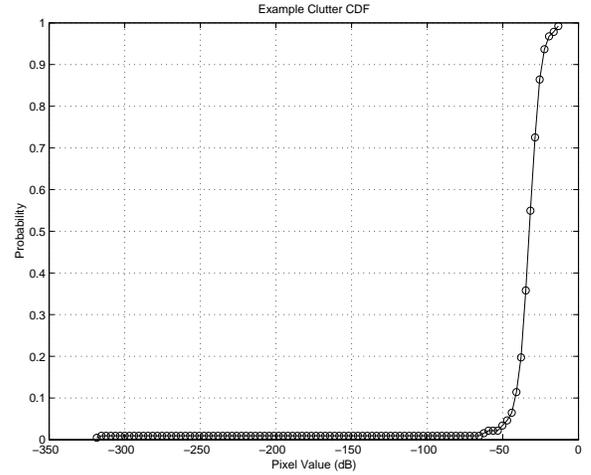


Fig. 3. A typical empirical cumulative distribution function (CDF) for the magnitude of clutter returns surrounding a target.

The two methods described above were implemented and tested. The detection results obtained were very similar to those that will be presented in section IV and we delay a presentation of any results until then.

It must be mentioned that each method just presented are not very computationally efficient. In particular, the median filter is found to be *extremely* slow due to the required computationally intensive sorting. In the next section we describe modifications to both algorithms aimed at decreasing the processing time required to produce a CFAR image.

III. ALGORITHMIC SPEED UPS

In this section we present computationally efficient algorithms that greatly decrease the amount of processing time required for the production of a CFAR image. Towards this end, in the first subsection we describe modifications applied to the original Novak stencil. In the second subsection we present modification to the median filtering approach. In the following section we present detection results obtained with each algorithm.

A. Algorithm Improvements to the Novak Stencil

In this subsection we present speed based algorithm improvements to the original Novak stencil. Towards this end we first consider the modified stencil shown in figure 4. This modified stencil is different than the previous stencil of figure 2 in two ways. The first, is that the center pixel in the original stencil has now been replaced by a “block” of pixels. In addition, the clutter ring that consisted of a single pixel border is replaced by a rectangular “block” ring of pixels. With these two modifications the CFAR algorithm works much as before. We present the modified algorithm in two steps, the first involves calculating *global* statistics and the second involves calculating *local* statistics.

We begin the modified Novak algorithm with a calculation of the *global* second order statistics of the entire fullscene SAR image. The global mean $\hat{\mu}_G$ and standard deviation $\hat{\sigma}_G$

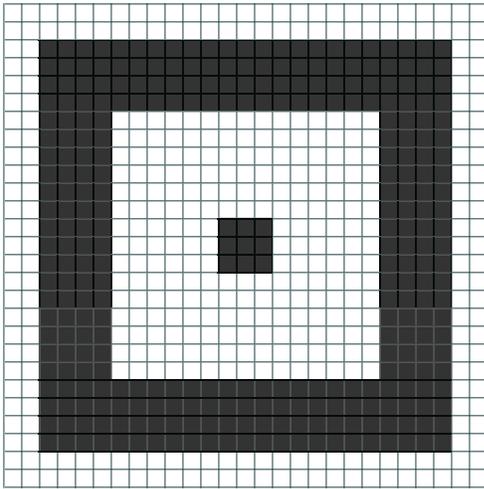


Fig. 4. The modified block Novak CFAR stencil. The highlighted pixels in the central block are *all* updated by clutter statistics estimations performed over all highlighted pixels in the rectangular border. We note that the number of pixels chosen in the central region and the block border are just demonstrative. In a practical systems the number of pixels in each structure would be much larger.

are calculated using the standard statistical formulas for the mean and standard deviation of a set of data [9]

$$\hat{\mu}_G = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N D_{i,j} \quad (18)$$

and

$$\hat{\sigma}_G^2 = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (D_{i,j} - \hat{\mu}_G)^2 \quad (19)$$

Here M represents the number of rows in the input SAR image and N the number of columns. These global statistics will be compared with local statistics to determine the statistic used in the CFAR equation 5.

The determination of the *local* clutter statistics is done with the outer clutter ring shown in figure 4. The relations used for computing these statistics are given by equations 2 and 3 from section II. Repeated here for convenience these equations are

$$\hat{\mu}_L = \frac{1}{N_{\text{Stencil}}} \sum_{(i,j) \in \text{Stencil}} D_{i,j}, \quad (20)$$

and

$$\hat{\sigma}_L^2 = \frac{1}{N_{\text{Stencil}}} \sum_{(i,j) \in \text{Stencil}} (D_{i,j} - \hat{\mu}_L)^2. \quad (21)$$

Once estimates of the clutter mean and standard deviation are computed, these numbers are used as representative estimates **for all pixels in the central block**. Thus, this algorithm results in regions all of which have the *same* local clutter statistics. This difference, in how many pixels we are computing the local statistics *for*, leads to improved computational algorithm performance. As in the previous algorithm, the number of pixels between the central region and the clutter ring should be large enough so that the largest target of interest located at the center of the stencil will not intersect its own clutter ring.

Given the local estimates of the clutter statistics, $\hat{\mu}_L$ and $\hat{\sigma}_L$, the final estimate of the clutter mean and standard deviation

values for the central block of pixels in figure 4 is computed with the following two equations

$$\hat{\mu}_C = \max(\hat{\mu}_L, \hat{\mu}_G), \quad (22)$$

and

$$\hat{\sigma}_C = \min(\max(\hat{\sigma}_L, \sigma_{\text{Floor}}), \hat{\sigma}_G) \quad (23)$$

The “max” in equation 23 insures that the local standard deviation is not too small, while the “min” insures that the true local distribution is used. A typical ordering for the three standard deviations in the above equations is given by

$$\sigma_{\text{Floor}} \leq \hat{\sigma}_L \leq \hat{\sigma}_G. \quad (24)$$

Finally, using $\hat{\mu}_C$ and $\hat{\sigma}_C$, equation 4 is used to calculate the CFAR value for *every* pixel in the central block.

The main computational benefit this method has is that it has turned single pixel operations into block pixel operations. In the new algorithm, the input SAR image is now tiled with much larger computational blocks, rather than single pixels. In addition, local clutter estimates will be made using the ring around the center block and applied to every pixel in the center block. Thus many pixels will be processed with the same clutter mean and standard deviation estimate. This statistical calculation happens only *once* for each central block and not for each pixel individually.

The algorithm just presented, when tested, did speed up the CFAR calculation but retained the problem of not detecting targets in high clutter environments. In addition, this CFAR algorithm introduces some algorithmic dependence, namely the method yields artificial boundaries in the CFAR image as each stencil is moved to a new set of pixels. These boundaries are certainly not physical and may detract detection performance. In the next section a modified median detection filter is presented that corrects both of these deficiencies *and* is computationally efficient.

B. Algorithm Improvements to the Median Stencil

In this subsection we present modifications to the median CFAR algorithm presented in section II that results in a computationally more efficient algorithm with improved performance in high clutter environments. Many of the ideas used in this algorithm are borrowed from the work on the modified Novak stencil presented in the previous subsection.

As a first observation, we cannot directly apply the results from the previous subsection to the median filter. The addition of more pixels in the clutter border in figure 4 will drastically slow down the median calculation. For computational efficiency the number of pixels sorted in the median calculation has to be even smaller than previously. This is accomplished with downsampling and the exact method used will be presented in the algorithm description below.

The first step in this modified median CFAR calculation is the computation of a mean and a standard deviation image from the original SAR image. These are images where each pixel represents the mean and standard deviation of the pixels in a box surrounding the pixel of interest. This is most easily performed using a stencil with an odd number of pixels along a side, say $2d + 1$. See figure 5 for examples of typical

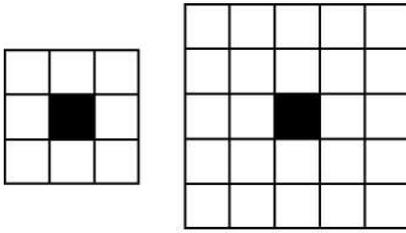


Fig. 5. Typical stencil forms used in computing mean and standard deviation images. The center pixel (in black) is overwritten with the average (respectively standard deviation) of all the pixels in the box. The specific stencil size is application dependent.

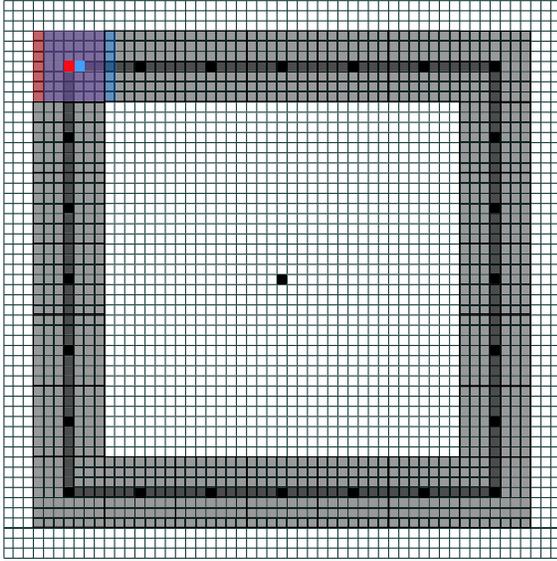


Fig. 6. The block median stencil used for computational efficiency. In the upper left corner of that figure two filters have been overlaid. The meaning is that the average of all pixels in that stencil have been placed at their corresponding center location.

stencil forms used in this processing with $d = 1$ and $d = 2$ respectively.

As this is an application of a moving average filter standard algorithms can be used [10] for processing. However, during our work on this algorithm a more computationally efficient method was developed and is described in [11]. As this processing is not critical to the remaining algorithm it will not be further described.

Once a mean and standard deviation image has been produced by local filtering or the algorithm described in [11], local clutter statistics ($\hat{\mu}_C$ and $\hat{\sigma}_C$) are computed for each pixel in the CFAR image using a filter like that shown in figure 2. Specifically, for a given pixel in the original SAR image a single pixel border is constructed around the corresponding pixel in the mean and standard deviation images. A stencil of this type is shown in figure 6 as a dark black line, surrounding the central black pixel this computation will be associated with. In addition, the pixels in the original SAR image that are averaged to compute the values on this single pixel border are shown in gray.

Rather than compute the median of all points along this ring, we compute the median of all points spaced by a $2d + 1$

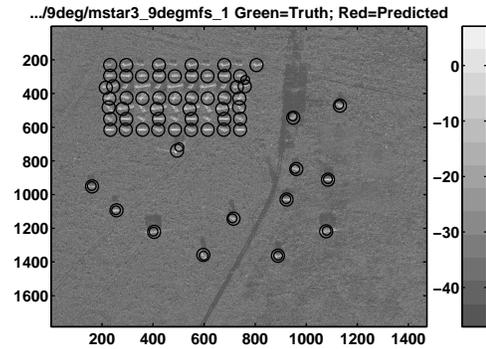


Fig. 7. Detection of dense ground targets using Novak's original *mean* based CFAR. Notice how many internal targets in the blockade are not detected.

pixels (or more) from each other. These pixels are shown in figure 6 in black along this single pixel border. When this median is performed in the *mean* image this median value will be assigned to $\hat{\mu}_C$. Similarly, when performed in the standard deviation image this median value is assigned to $\hat{\sigma}_C$. This downsampling of pixels used in computing the median is the greatest source of increased computational efficiency. In the next section we will present a comparison of the detection results obtained when using each of the methods discussed in this section.

IV. ALGORITHM COMPARISONS

In this section we present some simple comparisons between the various CFAR algorithms presented in this paper. Specifically, comparisons are made between the computationally improved Novak CFAR filter presented in subsection III-A, and a computationally improved median CFAR filtering presented in subsection III-B.

As a representative example of the type of results obtained when using each algorithm, we consider SAR images taken from the Moving and Stationary Target Acquisition and Recognition (MSTAR) data collection. In that collection, dense target environments were *not* considered, and as such no significant algorithmic differences were noted. To facilitate a comparison of detection performance in a dense environment modifications to the original images were performed to artificially increase target densities. This modification consists of adding a section of densely spaced targets in the upper left corner of a given image. The remaining sections of the image are not modified.

In figure 7 we present the modified image and the detection results obtained with the original Novak mean CFAR routine. The specific image chosen was taken from the third MSTAR collection set and represents spotlight SAR taken at a 9 degree depression angle. In figure 8 we present a magnification of the dense target region shown in figure 7. Here we can more clearly see the detection performance. Note the significant loss of target detections in the interior of the target cluster.

In figure 9 we diagram the Novak stencil used for the detection of a target in the center of a image. Note the contamination that other targets will cause to the stencil statistics. The neighboring targets are the reason for the missing detection from such an obvious target.

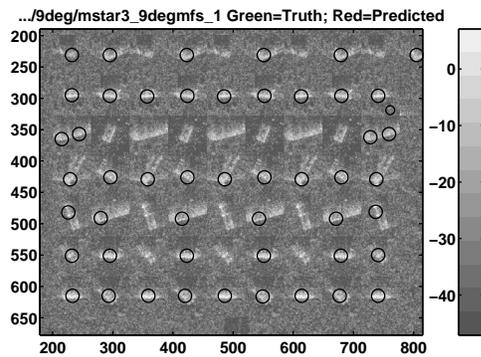


Fig. 8. Detection performance in a high density target environment using the original Novak a *mean* based CFAR. This figure is a close up of the high density region. Notice the significant loss of detections.

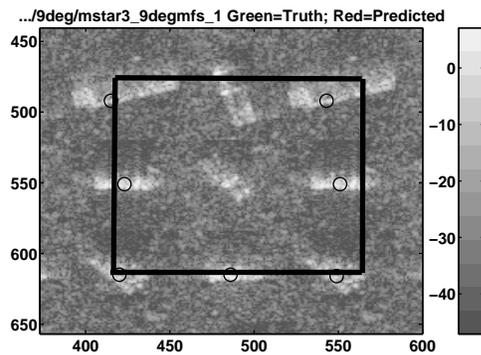


Fig. 9. Original Novak single pixel detection method. Note how neighboring targets can incorrectly influence the CFAR result at the center. In fact in this case the target is *not* detected.

In figure 10 we present the results of the median CFAR in the same SAR image as in figure 7. Here one can see the significant improvement that occurs because of the median filter.

In figure 11 we present a magnification of the dense target region shown in figure 10. Here we can more clearly see the detection performance. Note the extremely accurate target detections.

It has also been found that that the median CFAR filter presented above improved detection performance in strong clutter environments. There, the same effects of target contamination hold when non-target scatters overlap the detection stencil. The properties of the median filter work quite well at eliminating the spurious effects due to these abnormally large valued pixels.

V. CONCLUSION

In this paper we have presented several, modification to the original Novak [1] CFAR detection stencil that improve performance in various ways. First a rank ordering statistics approach was proposed for detection, that increases the probability of detection in dense target environments. Next a modification to the original stencil that improved computational efficiency was presented. Finally, the median computation was improved with subsampling. It is hoped that these algorithmic

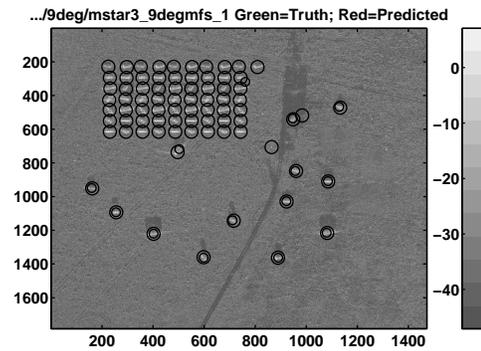


Fig. 10. Detection in medium target density using a *median* based CFAR. Here the internal targets *are* correctly detected.

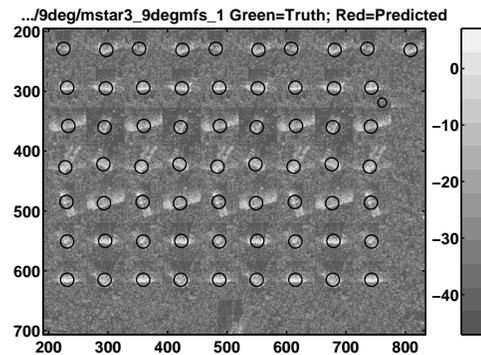


Fig. 11. Detection in medium target density using a *median* based CFAR. Close up of high density region. Again notice that the internal targets are correctly detected.

improvements will help the automatic target recognition community in providing a more robust algorithm.

ACKNOWLEDGMENT

The Authors Would Like To Thank Dr. John Kay For His Helpful Comments And Suggestions While This Research Was Carried Out.

REFERENCES

- [1] L. M. Novak, M. C. Burl, and W. W. Irving, "Optimal polarimetric processing for enhanced target detection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, pp. 234–243, 1993.
- [2] L. M. N. et al., "Effects of polarization and resolution on SAR ATR," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, pp. 102–116, 1997.
- [3] W. W. Irving, L. M. Novak, and A. S. Willsky, "A multiresolution approach to discrimination in SAR imagery," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, pp. 1157–1169, 1997.
- [4] L. M. N. et al., "The automatic target recognition system in SAIP," *MIT Lincoln Laboratory Journal*, vol. 10, pp. 187–202, 1997.
- [5] J. C. Principe, "Target prescreening based on a quadratic gamma discriminator," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, pp. 706–715, 1997.
- [6] L. M. Novak, G. J. Owirka, and A. L. Weaver, "Automatic target recognition using enhanced resolution SAR data," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, pp. 157–175, 1999.
- [7] H. L. V. Trees, *Detection, Estimation, and Modulation Theory, Part I*. Wiley-Interscience, 2001.
- [8] R. J. Sullivan, *Microwave Radar: Imaging and Advanced Processing*. Artech House, 2000.
- [9] M. R. Spiegel, J. Schiller, and R. A. Srinivasan, *Schaum's Outline: Probability and Statistics*. New York, USA: McGraw-Hill, 2000.

- [10] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. New Jersey, USA: Prentice Hall, 2001.
- [11] J. L. Weatherwax, "A fast method for computing local image statistics with rectangular stencils," *IEEE Transactions on Image Processing*, vol. ??, p. ??, 2005.

PLACE
PHOTO
HERE

John L. Weatherwax received a B.S. with honors in mathematics and a S.B. with honors in physics from the University of Missouri–Columbia in 1996. In September 1996 he attended Massachusetts Institute of Technology, Cambridge Ma. with a National Science Foundation Fellowship. In 2001 he graduated from M.I.T. receiving a doctorate in mathematics in the area of non-linear hyperbolic systems. From 2001 on, he has been a member of the technical staff working at M.I.T. Lincoln Laboratory, Lexington Ma. Currently he is in the ballistic missile division

working on discrimination algorithms.