NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

EMPLOYMENT OF ARBITRARY RIGID OBJECT MOTION AUTOFOCUS TO CONDUCT MOVING TARGET DETECTION IN MARITIME ENVIRONMENTS

by

Evan S. Campbell

June 2023

Thesis Advisor: David A. Garren
Second Reader: Ric Romero

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2023	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE
EMPLOYMENT OF ARBITRARY RIGID OBJECT MOTION AUTOFOCUS TO CONDUCT MOVING TARGET DETECTION IN MARITIME ENVIRONMENTS

5. FUNDING NUMBERS

6. AUTHOR(S) Evan S. Campbell

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE
A

13. ABSTRACT (maximum 200 words)

Synthetic aperture radar (SAR) imagery is a valuable tool used to conduct maritime surveillance. Ship movement induces azimuthal smearing in acquired radar images that can be exploited by autofocusing algorithms to perform moving target detection (MTD). Traditional autofocusing algorithms, such as phase gradient autofocus (PGA), assume linear movement of targets within a radar scene. The pitch and roll of surface vessels may be better modeled using arbitrary rigid object motion autofocus (AROMA), which employs a physical signal model designed to account for nonlinear translational and rotational target motion. AROMA uses this model to estimate and compensate for phase errors induced by non-linear target motion and generates an autofocused image as if the target were stationary. Analyzing autofocused imagery of a moving target compared to the original image results in a significant sharpness increase. Patches within an input radar image can be evaluated by sliding an interrogating window across the scene in a raster pattern. Autofocusing algorithms are used on each patch with the sharpness ratios of the pre- and post-autofocused images measured against a threshold to indicate the presence of a moving target. The detection and false alarm rates of the AROMA-based MTD algorithm are evaluated against traditional MTD autofocusing methods in both measured and simulated datasets to observe if the 3D modeling of AROMA exhibits improved detection performance over the traditional PGA.

14. SUBJECT TERMS
automated rigid object motion autofocus, AROMA, SAR, maritime, moving target detection

15. NUMBER OF PAGES
117

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

EMPLOYMENT OF ARBITRARY RIGID OBJECT MOTION AUTOFOCUS TO CONDUCT MOVING TARGET DETECTION IN MARITIME ENVIRONMENTS

Evan S. Campbell
Lieutenant, United States Navy
BS, Brigham Young University, 2014

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 2023

Approved by: David A. Garren
Advisor

Ric Romero
Second Reader

Douglas J. Fouts
Chair, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Synthetic aperture radar (SAR) imagery is a valuable tool used to conduct maritime surveillance. Ship movement induces azimuthal smearing in acquired radar images that can be exploited by autofocusing algorithms to perform moving target detection (MTD). Traditional autofocusing algorithms, such as phase gradient autofocus (PGA), assume linear movement of targets within a radar scene. The pitch and roll of surface vessels may be better modeled using arbitrary rigid object motion autofocus (AROMA), which employs a physical signal model designed to account for nonlinear translational and rotational target motion. AROMA uses this model to estimate and compensate for phase errors induced by non-linear target motion and generates an autofocused image as if the target were stationary. Analyzing autofocused imagery of a moving target compared to the original image results in a significant sharpness increase. Patches within an input radar image can be evaluated by sliding an interrogating window across the scene in a raster pattern. Autofocusing algorithms are used on each patch with the sharpness ratios of the pre- and post-autofocused images measured against a threshold to indicate the presence of a moving target. The detection and false alarm rates of the AROMA-based MTD algorithm are evaluated against traditional MTD autofocusing methods in both measured and simulated datasets to observe if the 3D modeling of AROMA exhibits improved detection performance over the traditional PGA.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AIS	automatic identification system
AROMA	arbitrary rigid object motion autofocus
ATR	automatic target recognition
CNN	convolutional neural network
COSMO-SkyMed	COnstellation of small Satellites for Mediterranean basin Observation
CR	cross range
CW	continuous wave
FFT^{-1}	inverse fast Fourier transform
FFT	fast Fourier transform
GHz	gigahertz
LFM	linear frequency modulation
ML	max likelihood
MTD	moving target detection
PGA	phase gradient autofocus
RG	down range
SAR	synthetic aperture radar
SHF	super high frequency
VMS	vessel monitoring system

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to Professor David Garren, my thesis advisor, who exhibited tremendous support toward my efforts throughout this process. I would also like to thank my second reader, Professor Ric Romero, whose instructions helped lay the foundation for my understanding of radar principles that made this project possible.

Lastly, I wish to thank my wife, Breanne, and my children, Leonidas, Adelaide, and Valora, for their constant understanding and patience during the long hours that went into this study.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Synthetic aperture radar (SAR) is an effective tool used to provide imagery over maritime environments as it can be employed regardless of weather conditions or time of day. SAR imagery provides valuable intelligence data in the form of traffic density estimates and tracking of vessels of interest [1]. Moving vessels present in a radar scene induce phase errors in recovered imagery data, which lead to azimuthal smearing in produced radar images [2]. This study assumes that the sea state within a radar scene induces sufficient pitch and roll in maritime vessels that noticeable smearing is recovered in the processed SAR image. Autofocusing algorithms have been developed to correct and compensate for phase errors in radar imagery to provide sharper autofocused images [2]. The sharpness present in an autofocused scene that has a moving target present is expected to be significantly higher than the original image, as the phase error compensation is expected to remove the observed smearing effects. This resulting sharpness improvement is likely to be greater in autofocused image patches with a moving target present versus image patches with only background clutter and no target induced smearing [3]. The higher sharpness improvement observed in autofocused patches with moving targets can be exploited to conduct moving target detection (MTD). MTD employs autofocusing algorithms that compensate for phase errors across individual image patches within a radar scene. The sharpness improvement of each autofocused patch is calculated and compared to a predetermined threshold. If the autofocused patch exhibits a sharpness improvement above the threshold, the MTD algorithm determines that a moving target was present in the interrogated patch [3]. This study looks to evaluate the performance of a novel autofocusing method, arbitrary rigid object motion autofocus (AROMA), against traditional autofocusing methods employed in MTD algorithms.

B. PROBLEM FRAMING

Maritime surveillance is valuable for several reasons, including monitoring of traffic density and tracking vessels of interest. Systems such as Vessel Monitoring System

(VMS) and Automatic Identification System (AIS) provide terrestrial and satellite updates of ship positional data but are not used by every seagoing platform [1]. VMS is mandatory for fishing vessels over 12 feet in length and AIS is required for commercial ships over 300 gross tons [1]. This allows for compliant maritime vessels to be monitored, with AIS alone globally tracking 160 000 ships on a typical day [1]. These two systems are reliant on the cooperative behavior of maritime vessels, in that they require platforms meeting the aforementioned size requirements to actively use the tracking systems. Ships that do not meet these size requirements can still impact maritime traffic density and must be accounted for using other means. Additionally, vessels may neglect to use VMS or AIS to escape surveillance and pose a challenge to intelligence efforts designed to track specific maritime targets of interest.

As cooperative systems may not always provide sufficient locational maritime data, other remote sensing methods can be employed [1]. SAR imagery is a remote sensing method that does not rely on the cooperative behavior of maritime targets that can be employed regardless of sunlight illumination or weather conditions due to its use of transmitted and received radar pulses [1]. SAR sensors placed on satellites offer additional benefits of wide area imaging and short revisit times over the same geographic area, due to these imaging satellites being employed in constellations consisting of numerous SAR platforms. This allows for SAR imagery to provide large amounts of measured data over maritime areas [1]. Due to the sheer amount of data and interrogation time requirements, it is difficult for imagery analysts to successfully examine and extract the most relevant and timely information that would be pertinent to maximize maritime situational awareness. It is desirable to provide a way to quicken detection efforts and autonomously interrogate radar imagery data to provide reliable locational updates of potential targets of interest within a vast maritime scene.

C. OBJECTIVE

This study looks to evaluate the use of autofocused based MTD algorithms to provide autonomous detection of maritime vessels of interest. Traditional autofocusing methods, such as phase gradient autofocus (PGA), assume that targets within a scene

exhibit stationary or constant velocity trajectories [4]. This may be effective to estimate and compensate for phase errors induced by linear movement of targets on land, but maritime targets on the sea do not move in a purely linear manner. Kelvin wakes and capillary waves due to the local sea state induce pitch and roll in seaborn vessels, resulting in nonlinear translational and rotational motion [1]. AROMA is a novel approach to autofocusing, intending to account for nonlinear motion of targets present in a radar scene when performing phase error estimation and correction [4]. AROMA employs a 3D physical signal model to track the movement of a target center relative to the position of the radar receiver during the SAR collection interval [4]. This study seeks to evaluate whether the AROMA capability of accounting for phase errors induced by nonlinear motion of maritime targets could produce improved detection performance over traditional autofocusing methods when evaluating measured imagery datasets from maritime environments.

The increased sharpness resulting from the autofocusing algorithms employed in this study is evaluated using intensity-based metrics. A common critique of intensity-based metrics is that their performance is degraded when subjected to complex noise environments [5]. High noise levels increase the magnitude of the observed background scatter within radar images and can result in MTD algorithms producing false alarms, an occurrence of the algorithm evaluating a moving target to be present in an image based on the improved sharpness of the autofocused background scatter when no target was present. This study injected various levels of noise into simulated radar scenes with moving targets present. MTD methods employing both AROMA and PGA were then employed on the simulated scenes to evaluate if the AROMA algorithm exhibited degraded detection performance compared to the traditional PGA.

A sliding window is a method of MTD used to interrogate smaller areas within a vast radar scene and evaluate for the presence of moving targets. An interrogating window is moved across the scene in a raster pattern with autofocusing performed on each radar image within the window [3]. These algorithms employ overlapping windows to ensure complete coverage of target scatterers present in the input image. These overlapping windows can result in redundant outputs of the same moving target that provide minimal

additional intelligence value. These redundant images can also provide partial coverage, where only a portion of the scattering centers from a singular target are present within an interrogating window. This study defines optimal coverage as when the entirety of target scattering centers, to include azimuthal smearing, are contained within the bounds of an interrogating window. This study seeks to evaluate the use of various metrics to identify windows providing optimal target coverage. These optimal windows will then be selected, and redundant partial outputs discarded using a consolidation algorithm. The proposed consolidation algorithm is designed to be used with AROMA and PGA sliding window MTD methods to produce singular, optimal outputs for each moving target present within an input radar image.

II. RADAR IMAGERY PRINCIPLES

Chapter II outlines the basic principles necessary to acquire a baseline understanding of synthetic aperture radar imagery.

A. SYNTHETIC APERTURE RADAR

SAR is a reliable method used to acquire imagery regardless of sunlight illumination and adverse weather conditions. SAR relies on a transmitter to emit electromagnetic energy that illuminates a target and then is reflected back to a radar receiver. The electromagnetic energy is transmitted at a specific frequency, or range of frequencies, and is split into separate radar pulses [2]. These pulses propagate away from a source in what is defined as the range (or down range) direction. The cross-range direction is defined as orthogonal to the range and is normally parallel to a transmitting radar platform. The geometry of the range and cross-range dimensions is depicted in Figure 1.

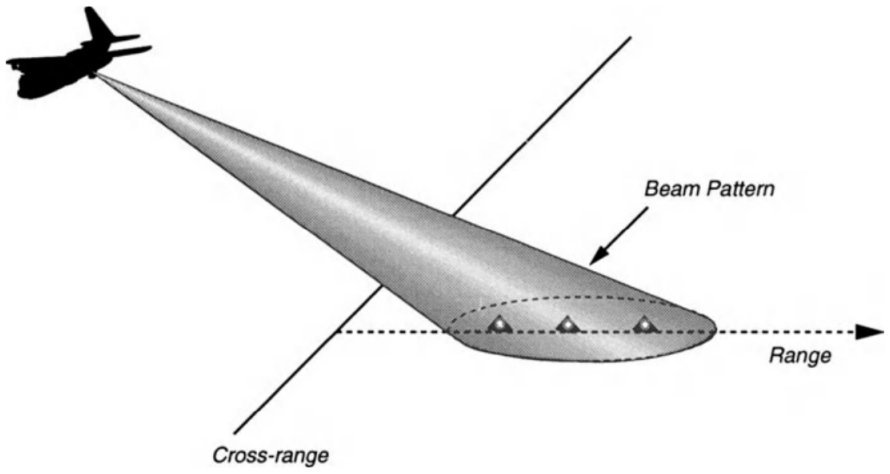


Figure 1. SAR Collection Geometry. Source: [2].

One way to transmit electromagnetic energy within a radar pulse is to employ a continuous burst waveform [2]. This continuous wave (CW) burst can be represented as a cosine function at the center frequency of the transmitted energy bound within an envelope

function. The pulse duration of the radar pulse is defined as τ_b with the time interval for the CW burst ranging from -$\tau_b/2$ to $\tau_b/2$ [2]. When the CW pulse reflects off a target, its return can be evaluated as:

$$\mathrm{r_b(t)} = \mathrm{A|g(u)|}cos\left(\omega_o\big(\mathrm{t}-\tau_0-\tau(\mathrm{u})\big)+\angle\mathrm{g(u)}\right)\mathrm{b}\big(\mathrm{t}-\tau_0-\tau(\mathrm{u})\big), \qquad (2.1)$$

where τ_0 represents the time for the radar return to reach the transmitting source after it reflects from the scene center with its return distance designated as u_0. The carrier radial frequency is given by ω_o. Time function $\tau(u)$ is used to represent the additional pulse return time from positions away from the center of the scene. A is the amplitude of the transmitted signal and $g(u)$ is the change in the waveform phase. The envelope function is represented as $b\big(t-\tau_0-\tau(u)\big)$ [2]. Figure 2 shows the scene collection geometry in terms of the distance variable u.

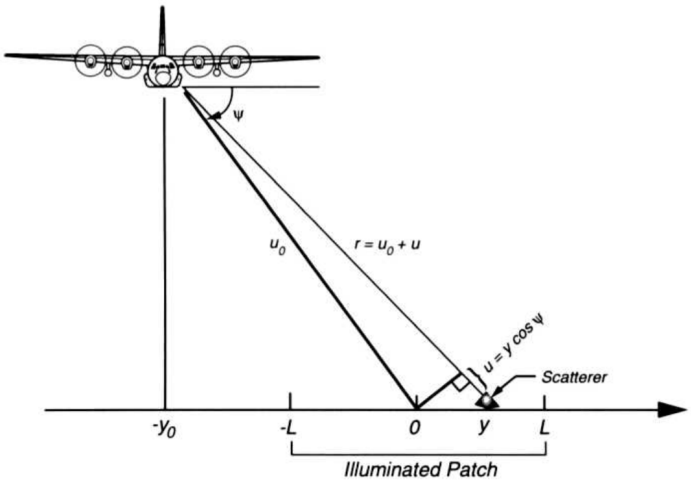


Figure 2. Scene Collection Geometry. Source: [2].

The illuminated scene ranges from distance -L to L and the specific ground range distance to a scatterer from the scene center is given by position y. Slant range r is defined as the actual distance between the radar platform and a scattering center in the ground plane and is defined as the distance u_0 plus an additional distance u. The additional distance u is

dependent on the ground range to a specific scattering center and the depression angle ψ between the radar platform and the target position in the ground plane [2]. This relationship is given in (2.2).

$$u = ycos(\psi) \tag{2.2}$$

The two-way propagation time for a signal to illuminate a target and reflect to the radar receiver is defined as τ_p. The CW duration, τ_b, is selected so that it is much smaller than τ_p to allow for the resolution of targets that are spatially close to each other in the ground plane. The effective duration of the CW burst is given by:

$$\tau_e = \frac{\int_{-\infty}^{\infty} f(t)dt}{f(0)}, \tag{2.3}$$

where *f(t)* is the pulse envelope waveform [2]. The total effective bandwidth of the CW burst is given by:

$$B_e = \frac{1}{2\pi}\frac{\int_{-\infty}^{\infty} F(\omega)d\omega}{F(0)}, \tag{2.4}$$

where $F(\omega)$ is the Fourier transform of the pulse envelope waveform. Both the effective duration and effective bandwidth are necessary to determine the range resolution in the slant plane. The range resolution is given in terms of τ_e and B_e in (2.5) and (2.6) respectively [2].

$$\delta_u = \frac{c\tau_e}{2} \tag{2.5}$$

$$\delta_u = \frac{c}{2B_e} \tag{2.6}$$

Radar processing mixes the return signal from (2.1) with in-phase and quadrature sinusoids and filters the resulting signal through a low pass filter. The final signal recovered after missing and filtering is given by:

$$\overline{r}_b(t) = a_o \int_{\frac{-r_p}{2}}^{\frac{r_p}{2}} g\left(\frac{c}{2}t\right) exp(-j\omega_0\tau) b(t-\tau)d\tau. \tag{2.7}$$

Equation 2.7 can be analyzed as a Fourier transform equation as seen in (2.8) [2].

$$R_b(\omega) = a_1 G\left[\frac{2}{c}(\omega + \omega_0)\right] B(\omega) \tag{2.8}$$

A spatial frequency variable U is defined to describe the spatial frequency offset and facilitate the calculation of cross-range resolution [2].

$$U = \frac{2}{c}\omega \tag{2.9}$$

The CW burst method uses long transmission times that result in small bandwidths which result in larger slant range resolutions as seen in (2.6). To acquire smaller resolution, another method of transmitting electromagnetic energy is employed termed Linear Frequency Modulation (LFM). During LFM, a range of frequencies are used during the radar transmission interval, termed a LFM chirp [2]. These frequencies either linearly increase, termed an upchirp, or linearly decrease, termed a downchirp. A LFM transmission signal is expressed as:

$$s(t) = exp[j(\omega_0 t + \alpha t^2)], \tag{2.10}$$

where α is the chirp rate or the rate of linear change in the signal frequency. The return echo of an LFM chirp is integrated from the near to the far slant range and is given by:

$$r_c(t) = A\boldsymbol{Re}\left\{\int_{-u_1}^{u_1} g(u) exp\left[j\left[\omega_0\big(t - \tau_o - \tau(u)\big) + \alpha\big(t - \tau_0 - \tau(u)\big)^2\right]\right] du\right\}, \tag{2.11}$$

where ***Re*** designates the real portion of the LFM return. The bandwidth of an LFM chirp waveform is a function of the chirp rate α and the chip duration τ_c, given in equation 2.12.

$$B_e = \frac{\alpha \tau_c}{\pi} \tag{2.12}$$

To estimate the final reflectivity $g(t)$ of the waveform return, a process called deramping is employed [2]. Deramping mixes the radar return signal $r_c(t)$ with the transmitted signal $s(t)$. The transmitted signal used in this process is represented by

delayed versions of in-phase and quadrature copies of the original $s(t)$ from (2.10). A low pass filter is then applied to the mixed signal to remove the terms at twice the original center frequency and a Fourier Transform is applied to the filter output [2].

The resulting spatial frequency in the slant range is given by:

$$U = \frac{2}{c}\omega = \frac{2}{c}\big(\omega_0 + 2\alpha(t - \tau_0)\big). \tag{2.13}$$

The down range ($\delta_y{}'$) and cross-range ($\delta_x{}'$) resolutions in the slant plane are given by (2.14) and (2.15) respectively.

$$\delta_y{}' = \frac{2\pi}{\Delta Y'} = \frac{c}{2B_c} \tag{2.14}$$

$$\delta_x{}' = \frac{2\pi}{\Delta X'} = \frac{\lambda}{2\Delta\theta} \tag{2.15}$$

The variable $\Delta Y'$ is the down range spatial extent and $\Delta X'$ is the cross-range spatial extent in the slant plane. The ground plane aperture angle is given by the term $\Delta\theta$ and is dependent on the diversity of viewing angles employed during the radar platform flight path as seen in Figure 3 [2].

The spatial frequency in the ground range is given by:

$$Y = \frac{2cos\psi}{c}\omega = \frac{2cos\psi}{c}\big(\omega_0 + 2\alpha(t - \tau_0)\big). \tag{2.16}$$

The down range (δ_y) and cross-range (δ_x) ground plane resolutions are given by (2.17) and (2.18) respectively.

$$\delta_y = \frac{2\pi}{\Delta Y} = \frac{c}{2B_c cos(\psi)} \tag{2.17}$$

$$\delta_x = \frac{2\pi}{\Delta X} = \frac{\lambda}{2\Delta\theta} \tag{2.18}$$

The variable ΔY is the down range spatial extent and ΔX is the cross-range spatial extent in the ground plane [2]. Optimal cross-range resolution can be achieved using a narrow radar beam. The radar beam width in the cross-range dimension is given by:

$$W_{cr} = R\frac{\lambda}{D}. \tag{2.19}$$

The range from the collection platform to the imaged radar scene is given by the term R. Standoff ranges between the radar platform and the area of interest can make it difficult to reduce R to achieve a sufficiently narrow cross-range beam. Waveforms that employ smaller wavelengths suffer attenuation effects from elements within the atmosphere and are not effective above the super high frequency (SHF) range, 3–30 GHz. The antenna dimensional size D cannot be sufficiently increased to achieve a desirably smaller cross-range beam as it would be difficult to place these large antennas on collection platforms. A solution to these restrictions is to employ a synthetic aperture to reduce the antenna beam width and achieve optimal cross-range resolution [2].

SAR relies on a principle termed aperture synthesis, a type of interferometry where data from a radar scene is collected from various viewing angles to reconstruct a reflectivity scene. The synthetic aperture is defined as the physical length of the flight path that the platform transits during the collection interval [2]. The antenna main beam remains focused on the patch center for the duration of the platforms collection leg as depicted in Figure 3.

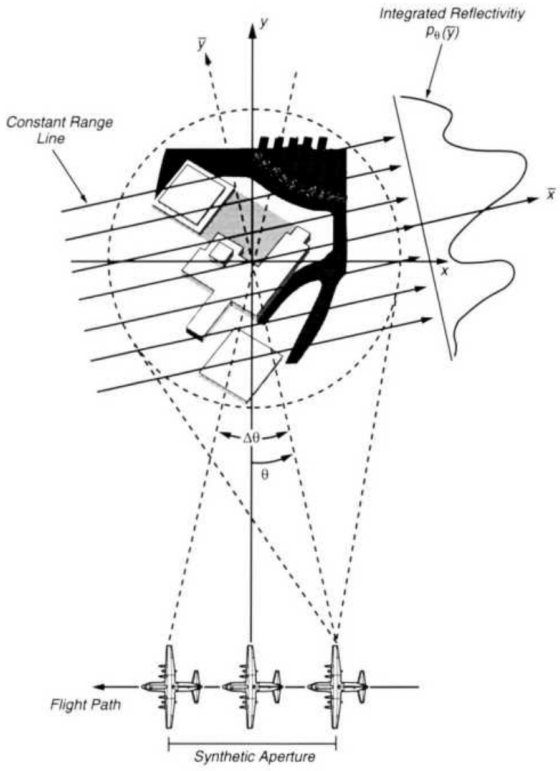


Figure 3. Synthetic Aperture Collection Leg. Source: [2].

SAR collection models fall into three categories: strip-map, scanning, and spotlight mode. Strip-map mode SAR entails the collecting platform aiming the antenna main bean orthogonal to the flight path without changing the beam focal point. Successive pulses are transmitted along the path at half antenna width spacing. Minimum transmission spacing is realized by adjusting the synthetic antenna size (flight path length) and the PRF of the radar signal [2]. The resolution of the recovered imagery is improved as the length of the flight path is increased. An example of strip-map mode SAR collection is depicted in Figure 4.

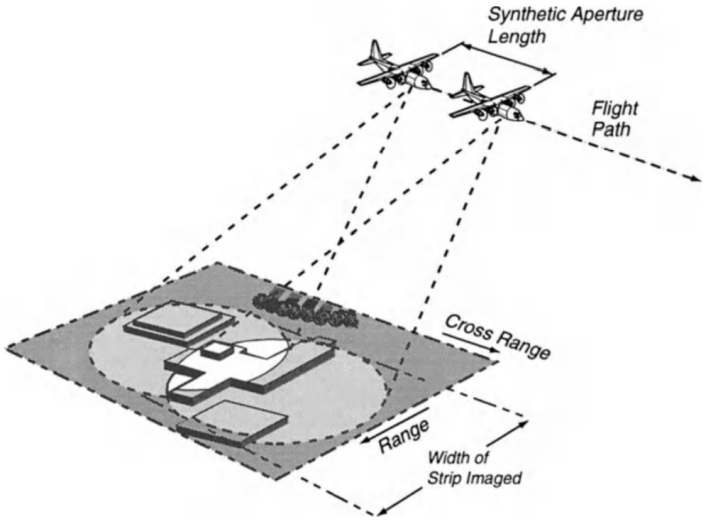


Figure 4. Strip-Map Mode SAR. Source: [2].

Scanning SAR mode is used to provide coverage over an area that is wider than the ambiguous range. The platform steers the antenna pattern to achieve numerous elevation angles. The increased number of elevation angles provides an increased coverage area but results in poor azimuth resolution [2]. An example of scanning mode SAR is depicted in Figure 5.

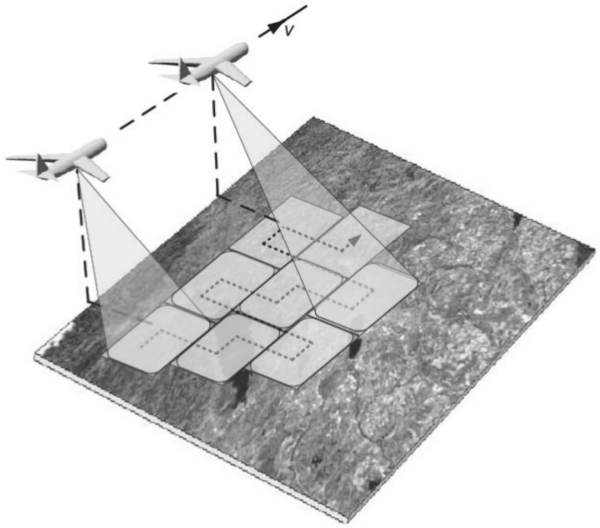


Figure 5. Scan Mode SAR. Source: [6].

Spotlight mode SAR involves slewing the radar beam so that the same coverage patch is illuminated for the duration of the flight path. By increasing the illumination time of a desired target area, the synthetic aperture length also increases resulting in better resolution of the coverage area [2]. Spotlight mode SAR is depicted in Figure 6.

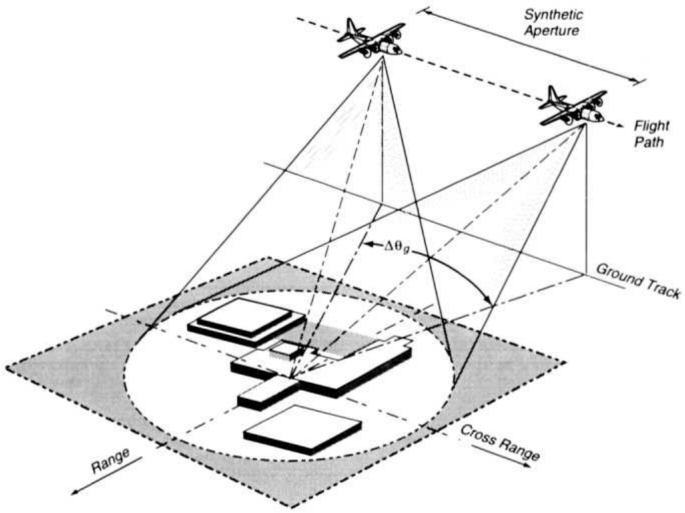


Figure 6. Spotlight Mode SAR. Source: [2].

The reflected scene from a spotlight mode SAR collection is recovered using tomographic framework and the projection slice theorem. Tomographic framework produces a layered final image through incorporating numerous thin slices obtained from multiple viewing angles [2]. The projection slice theorem is used to relate the values of the 1D Fourier transform from a projection function at a specified angle to the viewing angle scanned by the radar. Two dimensional Fourier transforms are converted to a polar raster space followed by an inverse Fourier transform used to acquire spatial-frequency data correlating to the final tomographic image [2]. Elevated targets present in a radar scene (buildings, trees, vessel masts) require the use of a 3D framework to reproduce the reflected scene. The employed 3D SAR collection geometry is depicted in Figure 7.

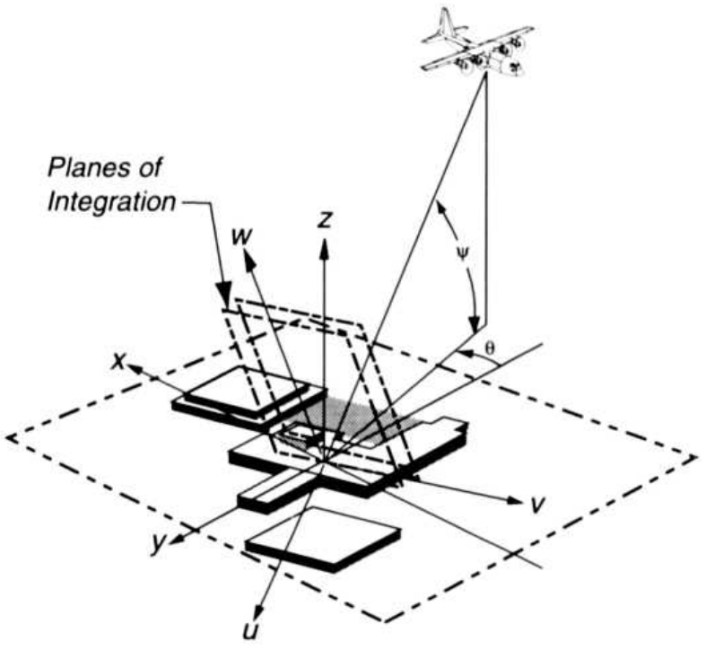


Figure 7. Three-Dimensional SAR Collection Geometry. Source: [2].

The recovered waveform phase data is converted to three dimensions, becoming $g(x, y, z)$, which is then integrated along a series of planes to produce a projection function with angular orientation in terms of azimuth θ and depression angle ψ [2]. The Fourier transform of $g(x, y, z)$ is then conducted for the three dimensions to extract the desired pulse data. The produced SAR image is a combination of the properties present in both the slant plane and ground plane. The polar data is converted to cartesian raster data to output the final image [2].

The maximum image dimensions depend on the spatial extent and the number of samples taken in both the range and cross-range dimensions as expressed by (2.20) and (2.21) respectively.

$$D_x = \frac{2\pi N_x}{\Delta X'} = \frac{\lambda N_x}{2\Delta\theta} \tag{2.20}$$

$$D_y = \frac{2\pi N_y}{\Delta Y'} = \frac{cN_y}{2B} \tag{2.21}$$

The variables N_x and N_y represent the number of samples taken in the down range and cross-range dimension respectfully [2].

To reconstruct an acceptable SAR image, the uncertainties in the distance from the SAR platform to the scene center between successive pulses must be within a fraction of the wavelength of the radar center frequency [2]. The standard method for estimating this slant range distance, u_0, is to employ navigation systems that incorporate accelerometers and other inertial measurement units onboard the radar platform. The distance u_0 does not always remain constant for the duration of the flight path as the platform may drift due to turbulence or other factors. Deviations in the expected slant range induce demodulation or phase errors to be present in individual radar pulses. For realistic SAR collection conditions, this error is modeled as a constant phase shift in each range-compressed pulse [2].

A model showing the range bins for uncorrupted phase data compared to corrupted phase data is given in Figure 8.

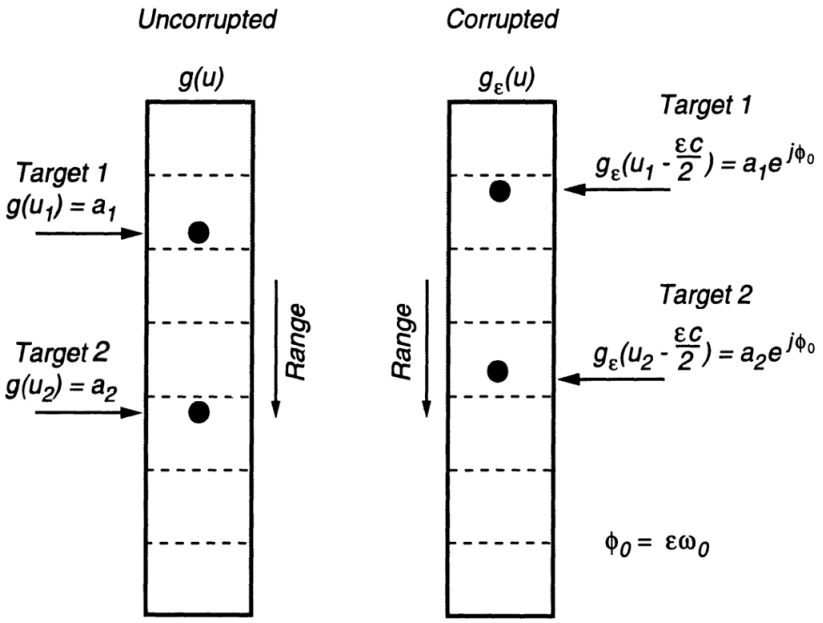


Figure 8. Uncorrupted and Corrupted Range-Compressed Pulse. Source: [2].

Two targets are present in Figure 8, with complex responses a_1 and a_2. The location of these targets in an uncorrupted environment is given by u_1 and u_2. The two conditions are related by range shift and a constant phase error term ϕ_0. The deviation in the slant range can be significant enough that a target can be presumed to reside in an incorrect range bin, as seen with Target 2 from Figure 8. This constant phase error term results in smearing observed in the cross-range dimension, also termed azimuthal smearing, of the produced SAR image [4]. The phase error due to deviations in slant range caused by platform motion is called high frequency error. High frequency error causes azimuthal smearing across the entirety of the range and cross range dimensions of the recovered radar image as observed in image b of Figure 9.

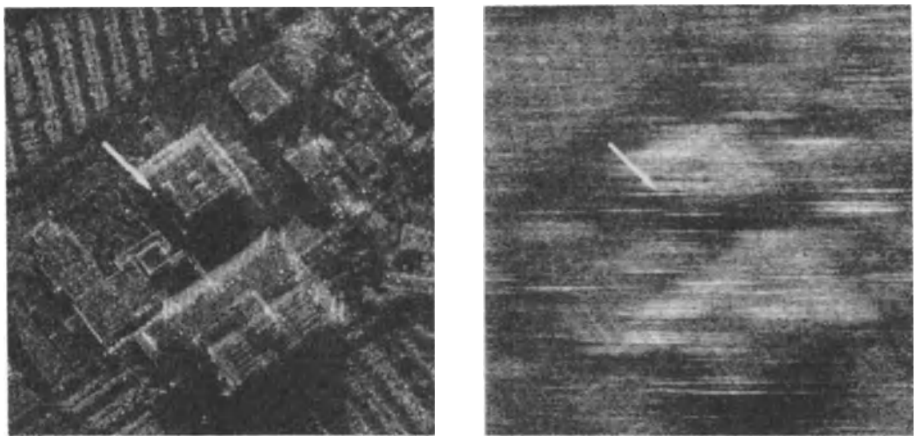


Figure 9. Uncorrupted and Corrupted Radar Scene. Source: [2].

Deviations in expected slant range can also be caused by target movement within the radar scene. If a scattering center is moving during the collection interval, its slant range will fluctuate from the expected value and will cause quadratic phase errors [4]. These phase errors induce azimuthal smearing from the recovered scattering centers within the target image and are the key point of interest in the autofocusing based MTD methods discussed in Chapter III.

B. AUTOFOCUSING METHODS

Autofocusing attempts to compensate for the phase errors induced by moving targets within a radar scene. The MTD algorithms evaluated in this study employed two different autofocusing methods: PGA and AROMA.

1. Phase Gradient Autofocus (PGA)

Phase Gradient Autofocus (PGA) is a popular method to estimate and correct phase errors in radar imagery that was introduced in 1989 [2]. PGA assumes that every target scatter present in a radar scene is effectively corrupted by the same phase error function. It then takes defocus information from multiple target scatters and generates a phase error estimate that can be corrected for across the complete radar image [2]. The PGA process is depicted in Figure 10.

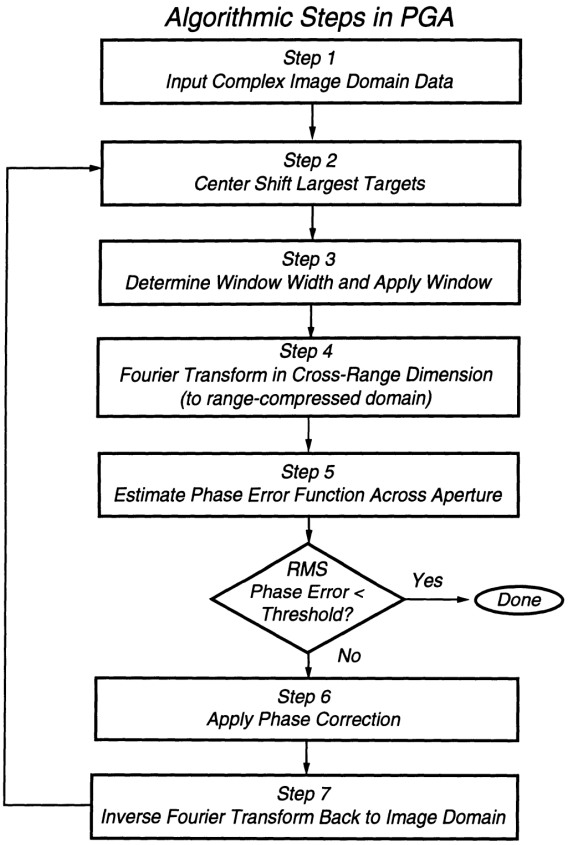


Figure 10. PGA Algorithm Steps. Source: [2].

The first step of PGA is to perform center shifting, a technique used to acquire average phase error estimates for multiple target smears within an image [2]. Strong target smears are desired as their elevated intensity above the background clutter of the image provide the best phase error estimation. The algorithm selects the strongest target scatter on each range line of the input image. Cells with the strongest target are then shifted to the center cross-range position of the range line, a process termed center shifting the image. Linear phase components that would have been present within the azimuth dimension of the range domain are removed with shifting of the range bins, leaving behind the desired phase error component for estimation [2].

After center shifting the input image, PGA uses a windowing process to select the center width of the image and eliminate the side boundaries. The rejected data in the image side boundaries represents noise to the phase error estimation scheme and its removal effectively increases the signal to noise ratio of the selected data in the image center. An optimal window size must be employed to acquire an effective phase error estimate [2]. A window that is too wide allows for noise away from the image center to be included whereas too thin of a window may fail to capture all the energy present within the center shifted target as the target scatter may be contained within multiple cross range bins within the same range line.

To determine appropriate window width, a method called non-coherent averaging is employed. As seen in Figure 11, the input image following center shifting (seen in the upper half of the figure) displays a region of blurring effects of equal width down the center of the image [2]. This central blurring effect is the same width across each range line because the strongest target scatters moved to the center are affected by the same phase error function.

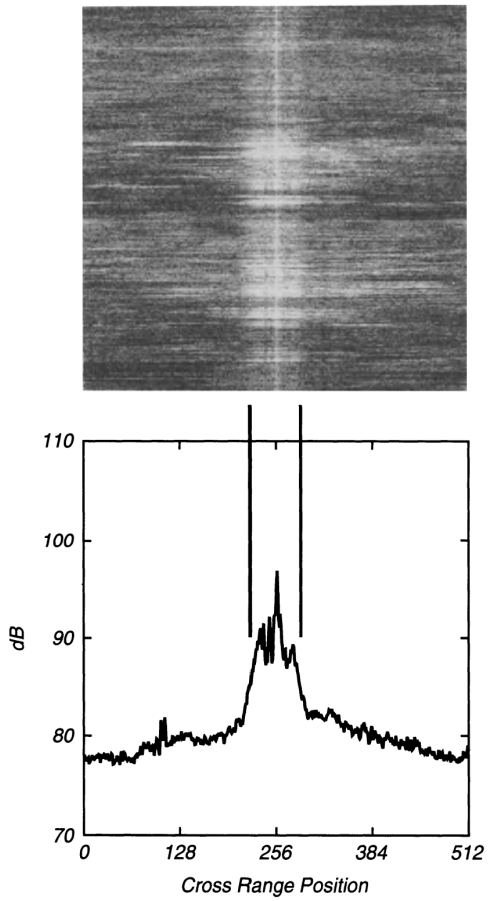


Figure 11. Center Shifted Image with Cross-Range Magnitude Sum. Source: [2].

Non-coherent averaging estimates the appropriate window width by summing the magnitude data along each cross-range bin. This magnitude data sum can be seen in the bottom half of Figure 11. This distribution of the magnitude sum is expected to have a max value at the image center, where the strongest target smears have been shifted [2]. There is also a noticeable plateau that exhibits the same width as the main blurring effect observed in the top half of Figure 11. To select the appropriate width to capture the desired blurring effect, a threshold dB level is chosen to select the cross-range bins that correlate to the plateau of the magnitude sum distribution [2].

Once the appropriate window size has been applied to the image, the center shifted targets are decompressed using a 1D discrete Fourier transform of each range line. PGA

assumes that the strong target at the center of each range line is a point reflector and any elements to the side are clutter which are modeled as independent identically distributed white Gaussian noise [2]. The Fourier transform causes the target signal to change to a constant while the background clutter remains Gaussian white noise. A maximum-likelihood (ML) estimate is then performed to estimate the phase error function affecting the center shifted target scatters. Adjacent pulses are evaluated to determine an estimate of their respective phase difference [2]. This ML phase difference estimator between adjacent pulses can be represented by:

$$\hat{\Delta\phi}(m) = \angle \sum_{k=1}^{N} \{\overline{g}^{*}(k, m-1)\overline{g}(k, m)\}, \tag{2.22}$$

where $\hat{\Delta\phi}(m)$ is the phase difference between adjacent pulses, k is the range line evaluated, and m is the specific pulse index from the radar. $\overline{g}(k, m)$ represents the complex imagery data from the circularly shifted image for the current pulse where $\overline{g}^{*}(k, m-1)$ represents the conjugate of the complex imagery data of the previously evaluated pulse index [2].

The phaser difference between adjacent pulses is then integrated across every pulse in the SAR data to compute the overall phase angle. This is represented in Figure 12.

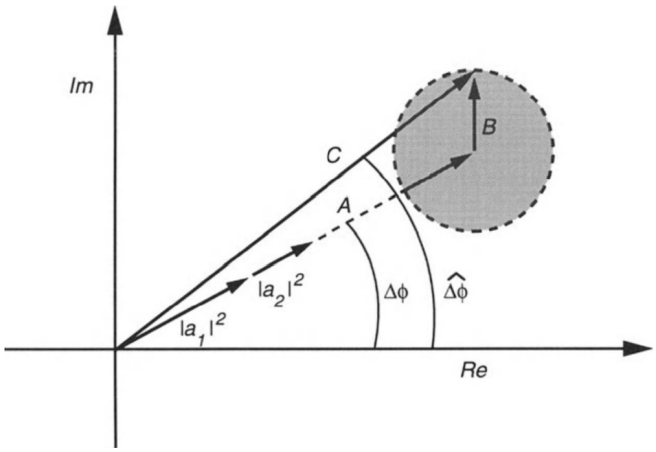


Figure 12. PGA Overall Phase Angle Vector. Source: [2].

The vector C represents the sum of the N smaller vectors computed from equation 2.22. These smaller vectors each have an angle, $\Delta\hat{\phi}(m)$, and a magnitude that will add coherently to create the sum vector. Deviations in each phase angle caused by white Gaussian noise will create a noise cloud represented by the circle with radius B. As the number of pulses evaluated increases the value of N in (2.22), the radius of this noise cloud becomes much smaller compared to the magnitude of the summed signal vector C. This results in a minor impact to the estimate of the overall phase angle $\Delta\phi(m)$ [2].

The overall phase angle obtained from the summation of phase angle differences for each pulse is the estimated phase error present in the initial radar image. The range compressed center shifted data from the 1D Fourier transform is multiplied by the complex conjugate of the phase error estimate $\Delta\phi(m)^*$ [2]. Multiplication by the phase error conjugate effectively eliminates the estimated phase error component within the center shifted dataset. The range-compressed data are now transformed back to the image domain through a 1D inverse Fourier transform. If the estimated phase error was equal to the true phase error present in the complex imagery data, the resulting radar imagery should appear uncorrupted with the azimuthal smearing eliminated [2].

PGA phase error estimation can be impacted by several factors. As a phase degraded target in the defocused input image is smeared, it is possible that the true peak of each target smear that is present in a range line is not selected in the center shifting process [2]. It is also possible that multiple strong targets are present on the same range line in close proximity, resulting in overlap of the target magnitudes when non-coherent averaging is performed. To account for these issues, PGA is performed in an iterative manner repeating the entire process on the output data from the ML phase error correction [2]. With each iteration conducted, center shifting becomes more accurate and smaller window widths can be employed. PGA ceases its iterative process when the estimated phase error angle $\Delta\phi(m)$ falls below a pre-determined threshold [2]. For the purposes of this study, 15 iterations for PGA were deemed sufficient to produce a negligible phase error estimate from the final output image.

2. Arbitrary Rigid Object Motion Autofocus (AROMA)

Arbitrary Rigid Object Motion Autofocus (AROMA), an algorithm proposed by Professor David Garren, builds on the principles of PGA, and attempts to generate a 3D physical signal model of the target to account for its nonlinear rotational and translational movement [4]. Using the 3D model, AROMA estimates and compensates for the phase error present in an input image to produce an autofocused output. The steps used in AROMA are outlined in Figure 13.

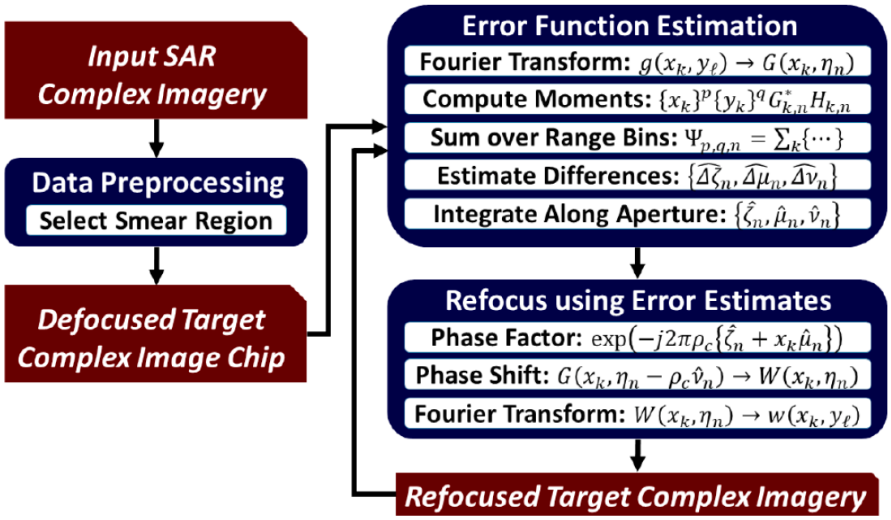


Figure 13. AROMA Algorithm Steps. Source: [4].

A baseline assumption for the AROMA algorithm is that the targets of interest within a radar scene move as a rigid object with arbitrary translation and rotation profiles relative to the collecting radar [4]. This is an accurate assumption for the maritime vessels in this study as the hull of the vessel acts as a rigid body changing its profile due to the pitch and roll induced by the sea state. A spherical coordinate system is defined to represent the target motion with the origin locked onto the target center. As the radar moves during collection, the radius, elevation angle, and azimuthal angle of the receiver relative to the target center will change [4]. The slant range coordinate for the target center relative to the receiver can be represented by:

$$s = xcos(\theta)cos(\phi) + ycos(\theta)sin(\phi) + r, \tag{2.23}$$

where s is the slant range coordinate, θ is the elevation angle of the receiver relative to the target center, ϕ is the receiver azimuth angle relative to the target, and r is the radial distance between the target and the receiver. The target is represented as a point scattering center with its arbitrary spatial location at $\{x, y, z\} = \{x, y, 0\}$ [4].

The next step generates a physical signal model based upon the rotational angles and translational distance between the radar receiver and the origin of the spherical coordinate system locked on to the target of interest [4]. A radar return recovered from the target scatter center is given by:

$$R_1 = \tilde{A_0} exp(j\omega_1) + N_1, \tag{2.24}$$

$$\omega_1 \equiv 2\pi\rho_c\{r_1 + x\cos(\theta_1)\cos(\phi_1) + y\cos(\theta_1)\sin(\phi_1)\}, \tag{2.25}$$

with $\tilde{A_0}$ representing the amplitude of the received signal and N_1 representing background noise. In (2.24) and (2.25), ω_1 denotes the recovered radar signal phase and ρ_c represents the spatial frequency. A second radar return recovered from the same target center is given by:

$$R_2 = \tilde{A_0} exp(j\omega_2) + N_2, \tag{2.26}$$

$$\omega_2 \equiv 2\pi\rho_c\{r_2 + xcos(\theta_2)cos(\phi_2) + ycos(\theta_2)sin(\phi_2)\}, \tag{2.27}$$

where subscript 2 designates the parameters corresponding to the later radar pulse with noise contributions between the two pulses assumed to be uncorrelated. The amplitude of successive radar returns is modelled to be constant as it is assumed to exhibit minimal changes between pulses. The model assesses that the recovered radar signal phase will change between radar returns due to the movement of the target scattering centers [4]. The difference in phase is given by (2.28).

$$\begin{aligned}\Delta w \equiv 2\pi\rho_c\{&r_2 - r_1 + xcos(\theta_2)cos(\phi_2) - xcos(\theta_1)cos(\phi_1) \\ &+ ycos(\theta_2)sin(\phi_2) - ycos(\theta_1)sin(\phi_1)\}\end{aligned} \tag{2.28}$$

The new expression for successive radar pulses R_1 and R_2 given by (2.29) and (2.30).

$$R_1 = A_0 + N_1 \tag{2.29}$$

$$R_2 = A_0 exp(j\Delta w) + N_2 \tag{2.30}$$

The next step accounts for the coordinates of the radar receiver in reference to a notional stationary target centered at the origin as well as the moving target of interest. Spherical coordinates are again used as the radial distance, radar elevation angle, and azimuth angle are now functions of slow time t. The stationary target maintains its position at $\{x, y, z\} = \{0,0,0\}$ whereas the radar moves based on its trajectory represented by $\{x, y, z\} = \{X_p(t), Y_p(t), Z_p(t)\}$ [4]. The radial distance, elevation angle, and azimuth angle of the radar relative to the stationary target are given by (2.31), (2.32), and (2.33) respectively.

$$r_0(t) \equiv \sqrt{\{X_p(t)^2 + Y_p(t)^2 + Z_p(t)^2\}} \tag{2.31}$$

$$\theta_0(t) \equiv arctan\left(\frac{Z_p(t)}{\sqrt{\{X_p(t)^2 + Y_p(t)^2\}}}\right) \tag{2.32}$$

$$\phi_0(t) \equiv arctan\left(\frac{Y_p(t)}{X_p(t)}\right) \tag{2.33}$$

The spherical coordinates of the radar in reference to the moving target of interest are given by $\{r(t), \theta(t), \phi(t)\}$. Using the relationship between a Cartesian coordinate system and spherical coordinates given by:

$$x = rcos(\theta)cos(\phi), \;\; y = rcos(\theta)sin(\phi), \;\; z = rsin(\theta). \tag{2.34}$$

AROMA defines the following variables to express the phase difference from (2.26) in linear form [4].

$$\mu(t) \equiv cos\big(\theta(t)\big)cos\big(\phi(t)\big) - cos\big(\theta_0(t)\big)cos\big(\phi_0(t)\big) \tag{2.35}$$

$$\nu(t) \equiv cos\big(\theta(t)\big)sin\big(\phi(t)\big) - cos\big(\theta_0(t)\big)sin\big(\phi_0(t)\big) \tag{2.36}$$

The difference seen in these variables between successive pulses is given by (2.37) and (2.38).

$$\Delta\mu(t) \equiv \mu_2(t) - \mu_1(t) = cos\big(\theta_2(t)\big)cos\big(\phi_2(t)\big) - cos\big(\theta_1(t)\big)cos\big(\phi_1(t)\big) \tag{2.37}$$

$$\Delta\nu(t) \equiv \nu_2(t) - \nu_1(t) = cos\big(\theta_2(t)\big)sin\big(\phi_2(t)\big) - cos\big(\theta_1(t)\big)sin\big(\phi_1(t)\big) \tag{2.38}$$

A third variable, ζ, is defined to represent the radial difference between radar position and the coordinate origin representing the target.

$$\zeta(t) \equiv r(t) - r_0(t) \tag{2.39}$$

$$\Delta\zeta(t) \equiv \zeta_2(t) - \zeta_1(t) = r_2(t) - r_1(t) \tag{2.40}$$

The physical signal model can then represent the phase difference from (2.26) using the new variables in a linear form.

$$\Delta\omega(t) = 2\pi\rho_c\{\Delta\zeta(t) + x\Delta\mu(t) + y\Delta\nu(t)\} \tag{2.41}$$

AROMA uses maximum-likelihood (ML) techniques similar to the PGA algorithm to estimate values of $\{\Delta\zeta(t), \Delta\mu(t), \Delta\nu(t)\}$ for successive radar pulses using (2.26), (2.27), and (2.28). The total difference functions $\{\zeta(t), \mu(t), \nu(t)\}$ are evaluated by integrating $\{\Delta\zeta(t), \Delta\mu(t), \Delta\nu(t)\}$ across all pulses present in the SAR data [4]. As with the PGA algorithm, AROMA selects the strongest scattering center present in each range line for phase error estimation. A radar down range position is denoted by x_k with k representing a specific range line. The cross-range position of the dominant scatter is denoted by $\tilde{y}(x_k)$. The final model representing the phase difference observed at each range line is given by:

$$\Delta\omega_k(t) = 2\pi\rho_c\big\{\Delta\zeta(t) + x_k\Delta\mu(t) + \tilde{y}(x_k)\Delta\nu(t)\big\}. \tag{2.42}$$

Estimation of phase error begins with analyzing the complex image function $g(x_k, y_l)$ with $x_k$representing the ground down-range coordinate and y_l representing the ground cross-range coordinate. A 1D discrete inverse Fourier transform is then performed on $g(x_k, y_l)$.

Many of the physical properties relating to the scattering centers within the scene are unknown with Gaussian probability density functions assigned to represent unknown differences in $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$ as follows [4]:

$$p(z_k|\Delta\zeta, \Delta\mu, \Delta\nu) = \frac{1}{\pi^2|Q_k|} exp\left(-{z_k}^{*T} {Q_k}^{-1} z_k\right). \tag{2.43}$$

Q_k represents the covariance matrix conditioned on the measurements of two successive pulses, G_k and H_k which are given by the vector:

$$z_k \equiv [G_k \ H_k]^T, \tag{2.44}$$

$$Q_k = E\{z_k {z_k}^{*T}\} = \begin{bmatrix} E\{G_k {G_k}^*\} & E\{G_k {H_k}^*\} \\ E\{H_k {G_k}^*\} & E\{H_k {H_k}^*\} \end{bmatrix}. \tag{2.45}$$

As dominant scattering centers are selected for each range line k, AROMA assumes that pulse amplitudes remain independent of additive noise with their respective variance represented as:

$${\sigma_A}^2 \equiv E\{|A_k|^2\}, \tag{2.46}$$

$${\sigma_N}^2 \equiv E\left\{\left|N_{1,2}\right|^2\right\} = E\left\{\left|N_{2,k}\right|^2\right\}. \tag{2.47}$$

The covariance matrix Q_k reduces to:

$$Q_k = \begin{bmatrix} {\sigma_A}^2 + {\sigma_N}^2 & {\sigma_A}^2 exp(-j\Delta\omega_k) \\ {\sigma_A}^2 exp(j\Delta\omega_k) & {\sigma_A}^2 + {\sigma_N}^2 \end{bmatrix}. \tag{2.48}$$

Two vectors are formed, G and H, comprised of elements recovered from the range lines of the range compressed image data [4]. A new conditional PDF is designated corresponding to all range lines simultaneously:

$$p(G, H|\Delta\zeta, \Delta\mu, \Delta\nu) = \frac{1}{\pi^{2K} {Q_0}^K} \prod_{k=1}^{K} exp\left(-\Lambda_k(\Delta\zeta, \Delta\mu, \Delta\nu)\right), \tag{2.49}$$

where K is the total number of range lines and Λ_k is the exponential argument from (2.41). The PDF logarithm is then maximized and joint ML optimization in multiple dimensions is performed. A new constant independent of $\{G\ H\}$ and $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$ is defined as:

$$\kappa_1 \equiv -2Kln(\pi) - Kln(Q_0). \tag{2.50}$$

The natural logarithm of $p(G, H|\Delta\zeta, \Delta\mu, \Delta\nu)$ can now be evaluated as:

$$ln(p(G, H|\Delta\zeta, \Delta\mu, \Delta\nu)) = \kappa_2 +$$

$$\frac{{\sigma_A}^2}{Q_0} \sum_{k=1}^{K} \{G_k {H_k}^* exp(j\Delta\omega(x_k, \Delta\zeta, \Delta\mu, \Delta\nu)) + {G_k}^* H_k exp(-j\Delta\omega(x_k, \Delta\zeta, \Delta\mu, \Delta\nu))\}. \tag{2.51}$$

Partial derivatives with regards to $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$ are calculated from the natural logarithm probability density function. The three partial derivative equations are then set equal to zero and the values of $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$ are solved for [4].

Linear approximation techniques employed in AROMA are assumed to be valid provided that the interval between radar pulses is small enough so that the rotation and translation of the target is sufficiently small between successive pulses [4]. With this minimal shift in movement, it is also assumed that the phase argument of $\Delta\omega_k(t)$ is small enough to apply the small phase argument approximation to the solved partial derivatives with respect to $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$. Again, if the interval between pulses yields a sufficiently high PRF, this linear phase approximation does not cause an issue. In cases when smaller radar PRF is used, the linear phase approximation in AROMA can result in greater error resulting in elevated target defocus [4].

AROMA then seeks to estimate the unknown increments observed in $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$ between successive radar pulse pairs for indexes n and n +1.

$$\Delta\zeta_n \equiv \zeta_{n+1} - \zeta_n, \Delta\mu_n \equiv \mu_{n+1} - \mu_n\ , \Delta\upsilon_n \equiv \upsilon_{n+1} - \upsilon_n \tag{2.52}$$

Standard linear techniques are used to estimate the values of $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$ with their respective estimates defined as $\{\hat{\Delta\zeta_n}, \hat{\Delta\mu_n}, \hat{\Delta\nu_n}\}$.

The cross-range spatial frequency observed at a pulse n is defined as η_n. Any linear phase variation in η_n can cause certain warping or offset effects within $\{\Delta\zeta, \Delta\mu, \Delta\nu\}$. To reduce the impact of these effects, the means of $\{\hat{\Delta\zeta_n}, \hat{\Delta\mu_n}, \hat{\Delta\nu_n}\}$ are removed [4]. Once the means are removed, AROMA integrates these functions along the synthetic aperture to give:

$$\hat{\zeta_{n+1}} = \hat{\zeta_n} + \hat{\Delta\zeta_n}, \tag{2.53}$$

$$\hat{\mu_{n+1}} = \hat{\mu_n} + \hat{\Delta\mu_n}, \tag{2.54}$$

$$\hat{\nu_{n+1}} = \hat{\nu_n} + \hat{\Delta\nu_n}. \tag{2.55}$$

The constant phase terms in $\{\hat{\Delta\zeta_n}, \hat{\Delta\mu_n}, \hat{\Delta\nu_n}\}$ induce changes only in the phase values of individual image pixels and do not yield undesired warping or shifting effects. As these phase terms do not affect the magnitude imagery, they are set equal to zero. The final phase error functions are estimated by accumulating results from previous AROMA iterations [4].

AROMA does not seem limited with respect to SAR collection conditions nor target translational and rotational motion. The linear approximation used in the algorithm limits the changes in rotation angle between successive radar pulses. It is possible that the algorithm does occlude a number of smaller target scatters present in the SAR collection scene. These occlusions are not expected to impact the accuracy of the error estimate if there are sufficient target scatters selected from the scene for evaluation [4].

As AROMA uses ML to give error estimates corresponding to specific values of n within the aperture index, the use of ML processing does not require averaging over a number of radar pulses. This allows AROMA to theoretically include sudden motions within the estimated error functions [4]. A possible issue regarding sudden target motions involves the increased level of target smearing present in the initial SAR imagery. The target scatter may be occluded if its level of smearing reduces its intensity below that of the background clutter present in the radar scene [4].

Once the total error function is estimated, AROMA refocuses the image data using the error estimate. Target focus compensation is obtained by multiplying the range-compressed, azimuth spread image data by the function $\beta(x_k, \eta_k)$ given by:

$$\beta(x_k, \eta_k) \equiv exp\left(-j2\pi\rho_c\left\{\hat{\zeta_n} + x_k\hat{u_n}\right\}\right). \tag{2.56}$$

The negative sign in the exponential argument achieves phase error compensation as the error estimate is subtracted from the phase values in the original data [4]. The compensated image data is defined as $B(x_k, \eta_k)$ and is given by:

$$B(x_k, \eta_k) = G(x_k, \eta_k)\beta(x_k, \eta_k). \tag{2.57}$$

This function represents the image data compensated for phase errors corresponding to $\hat{\zeta_n}$ and $\hat{u_n}$ estimates. Compensation of image data based on $\hat{v_n}$ error estimates is more complex as the correction is a function of y_ι. As y_ι is a Fourier transform coordinate of the cross-range spatial frequency η_n, AROMA applies a separate phase correction for each y_ι value [4]. The phase corrected data based on $\hat{v_n}$ error estimates is given by:

$$W(x_k, \eta_n) = \beta(x_k, \eta_n)\sum_{\iota'} g(x_k, y_{\iota'})exp\left(j2\pi y_{l'}\left\{\eta_n - \rho_c\hat{v_n}\right\}\right). \tag{2.59}$$

Simplifying (2.57) and using properties of the Fourier transform, the final expression for the refocused imagery data becomes:

$$W(x_k, \eta_n) = \beta(x_k, \eta_n)G\left(x_k, \eta_n - \rho_c\hat{v_n}\right). \tag{2.60}$$

A 1D Fourier transform is performed on the corrected range compressed data and image processing outputs an autofocused radar image [4]. The AROMA algorithm continues to run using the corrected imagery as the input for each new iteration until sufficient target focus is acquired. For the purposes of this study, AROMA was designated to conduct 15 iterations on each input image, matching the iteration count of PGA.

THIS PAGE INTENTIONALLY LEFT BLANK

III. MOVING TARGET DETECTION

Radar principles are a key contributor to the fundamentals of MTD. This chapter details how autofocused imagery can be evaluated using intensity-based metrics to reveal the presence of a moving target and how large swaths of a radar scene can be interrogated using a sliding window approach to conduct MTD.

A. INTENSITY BASED METRICS

Intensity based MTD algorithms evaluate the sharpness of an image patch before and after autofocusing. The increased sharpness of an autofocused image is computed using Muller-Buffington image sharpness metrics:

$$f_1 = \frac{\boldsymbol{S_1(after\ correction)}}{\boldsymbol{S_1(before\ correction)}} = \frac{\sum_{x,y}|g_{cor}(x,y)|^4}{\sum_{x,y}|g(x,y)|^4}. \tag{3.1}$$

The Muller-Buffington sharpness metric S_1 is defined as the summation of the intensity squared of the pixels of an image [3]. The feature metric f_1 denotes the sharpness ratio acquired by comparing the image sharpness before and after phase error correction by autofocusing. In [3], Fienup proposed that the sharpness ratio is an indicator of the presence of a moving target in the input image. Autofocusing algorithms that have successfully corrected for the phase error function induced from a moving target will output an autofocused image with significantly reduced azimuthal smearing present. This output image is much sharper than the original input and its corresponding f_1 value is expected to be higher than sharpness ratios from autofocused image patches with no initial smearing induced by moving targets [3].

Figure 14 shows an image patch with a moving target present with a calculated f_1 value of 5.365.

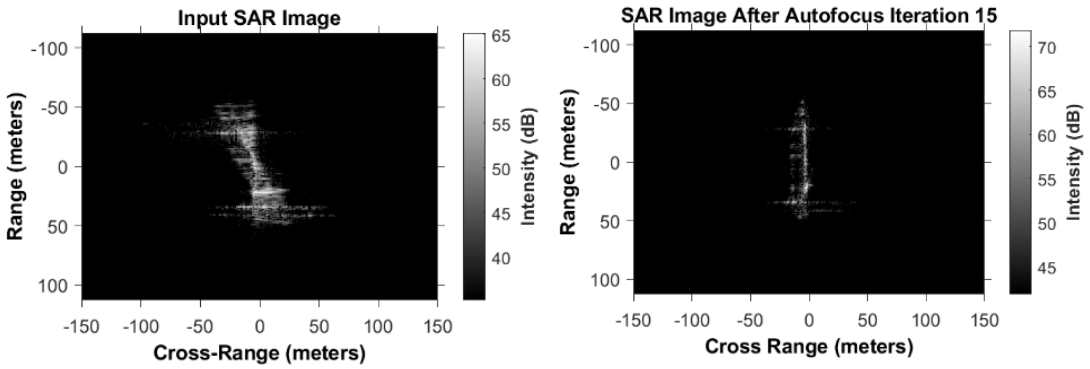


Figure 14. Moving Target Image Patch Pre/Post Autofocusing (f_1=5.365).

A sample $f_1$value of a radar scene with no target present is calculated as 0.8976 from the image patch depicted in Figure 15.

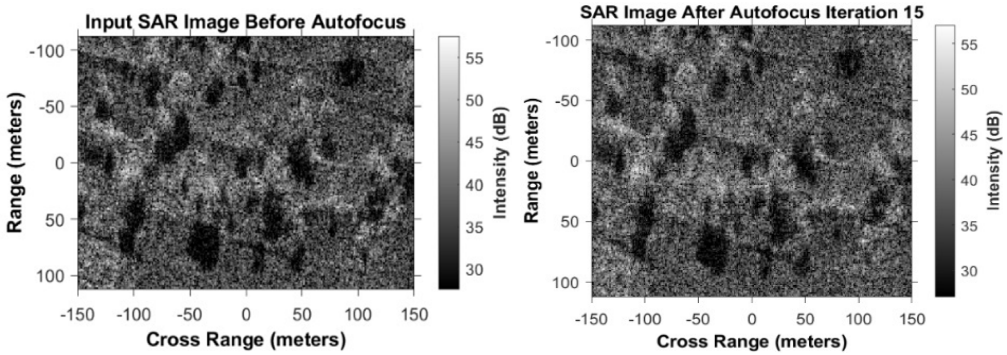


Figure 15. Image Patch with no Moving Target Pre/Post Autofocusing (f_1=0.8976).

The comparative f_1 value of the image patch with the moving target (5.365) is significantly larger than the f_1 value correlating to the image with no moving target present (0.8976). This relationship holds true in various maritime environments. Increasing sea states result in larger wave heights and more turbulent movement which cause increased levels of background scatter [1]. High levels of background scatter reduce the visibility of moving target smears and can result in decreased f_1 values compared to scenarios where the same smear is present in a low clutter background. An image patch depicting a target in turbulent sea states is shown in Figure 16 with an exhibited f_1 value of 1.986.

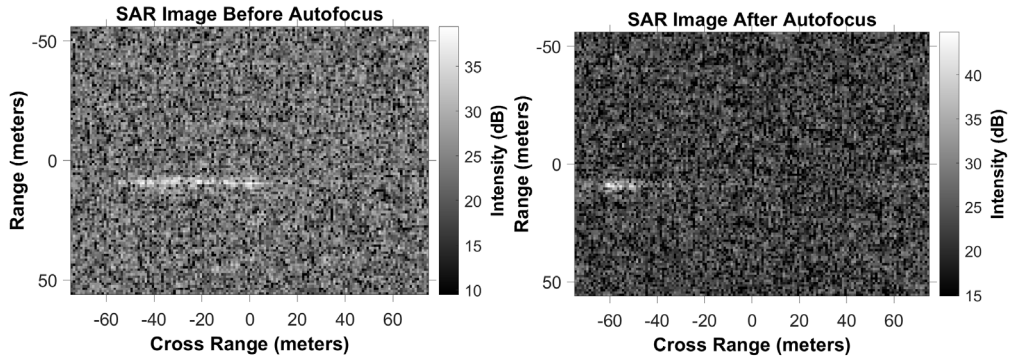


Figure 16. Image Patch with Moving Target Present in High Sea State Pre/ Post Autofocusing (f_1=1.986).

An image patch containing elevated background scatter due to a high sea state with no vessel present exhibits a calculated f_1 value of 0.9657, as seen in Figure 17.

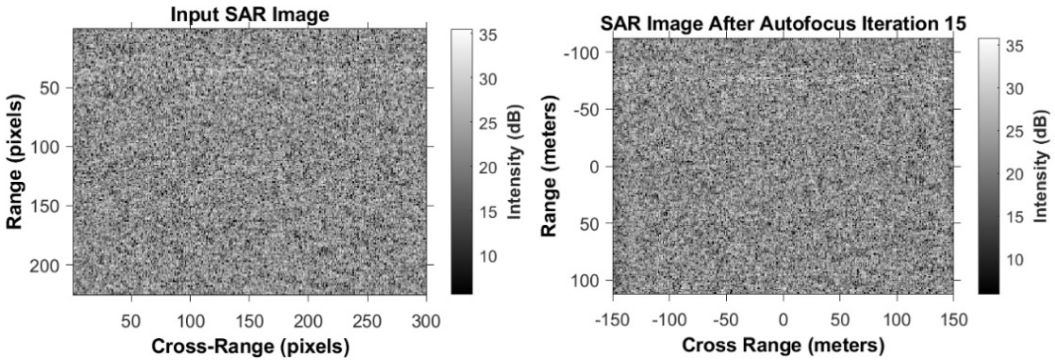


Figure 17. Background Clutter in Elevated Sea State Pre/Post Autofocusing (f_1=0.9657).

The f_1 value from the target in an elevated sea state in Figure 17 was lower than the f_1 value recovered from the moving target observed with minimal background clutter, 1.986 compared to 5.365. Despite the reduced autofocused sharpness ratio in the high clutter environment, the f_1 value was still significantly greater than the sharpness ratio calculated from the elevated sea state alone as seen in Figure 17, 1.986 compared to 0.9657. This shows that even in high clutter environments, images with a moving target exhibit slightly larger f_1 values than images with no moving target present.

A sharpness threshold, designated in this study as f_2, can be assigned to determine whether a moving target is present in an input image [3]. An image with an f_1 value above the threshold is assessed to have a moving target present whereas images with f_1 values below the threshold are assessed to contain only background clutter.

The complex noise backgrounds present in maritime imagery can prove problematic for autofocused-based MTD methods. The validity of intensity-based detection systems is dependent on the assumption that sufficient intensity contrast exists between a moving target and surrounding sea clutter [5]. As seen in Figure 16, high sea states reduce the observed sharpness ratio of an autofocused image patch and can mask the intensity of target scattering centers. It is possible that complex noise present in high resolution maritime SAR imagery can cause interrogated windows with targets present to fall below the sharpness threshold and be occluded from the MTD output [5]. The high level of background clutter in windows containing no target present can lead to a sufficiently improved sharpness ratio post autofocusing, such that the image patch passes the predetermined threshold and results in a false alarm output. This study seeks to evaluate the performance of different autofocus based MTD algorithms in various levels of background noise as detailed in Chapter IV.

B. SLIDING WINDOW ALGORITHM

A sliding window approach can be employed to iteratively interrogate an input image scene by sliding smaller image patches, termed windows, in a raster scan pattern across the input [3]. Autofocusing is performed on each window and a corresponding f_1 value is calculated. Any window exhibiting an f_1 value above the predetermined sharpness threshold is assessed to have a moving target present and is labeled as an output to the algorithm. This allows the MTD sliding window to interrogate large swaths of input radar images and only output windows that have detected moving targets. An overview of sliding window interrogation for a singular patch is presented in Figure 18.

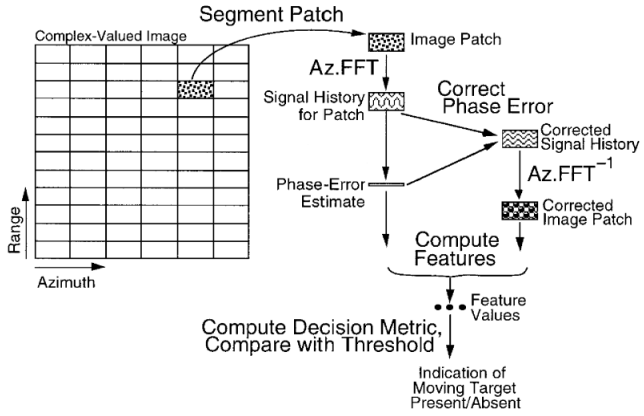


Figure 18. Sliding Window Algorithm Steps. Adapted from [3].

As shown in Figure 18, an individual image patch is selected and autofocusing is conducted. These autofocusing steps are detailed in Chapter II and are shown in Figure 18 to include a fast Fourier transform (FFT), phase error estimation, inverse fast Fourier transform (FFT^{-1}) and phase error correction. The feature values computed include the sharpness metric f_1 which is then compared to the threshold value f_2 to indicate the presence of a moving target in the interrogated window patch.

An appropriate interrogating window dimension is determined by the user to provide adequate coverage of anticipated targets of interest. Target size is dependent on the range and cross range resolutions from the provided radar imagery. If the range and cross-range resolutions are unknown, the sliding window can be designed to perform multiple iterations with different window dimensions to ensure complete coverage of target vessels of interest within the interrogating windows. The optimum interrogation window is roughly the size of the moving target of interest. Inclusion of excess background or the failure of a window to capture the entirety of the smears from a target scatterer causes degraded performance [3].

There may be instances where targets of varying sizes are present within the same radar scene. In this instance, it is appropriate to perform a sliding window interrogation multiple times, with each iteration employing a different window size [3]. The same approach can be applied when targets of the same size exhibit various velocity trajectories.

Faster moving targets induce a larger magnitude of azimuthal smearing and require larger interrogating window patches to provide full coverage. The outputs from multiple iterations of the sliding window algorithm are then combined to provide the total detections made from the input radar scene.

IV. EXPERIMENTAL DESIGN AND IMPLEMENTATION

A. SLIDING WINDOW ALGORITHM

A sliding window algorithm was developed for this study to iteratively interrogate image patches and evaluate for the presence of a moving target. This algorithm was generated using MATLAB and was initiated by extracting complex valued data taken from the in-phase and quadrature channels of input radar images.

Once complex data had been extracted, an appropriate window size would be determined by assigning range and cross range span parameters. The range and cross range span would be designated so that target scatters of interest would be captured with adequate coverage within the window. The adequate coverage would allow for the entire target of interest to fit within the window and thus allow autofocusing to be performed on the total scattering centers belonging to the given target. This yields a desired target scatter to background clutter ratio. Target scatters that took up too much or too little of the image patch result in suboptimal detection performance [3].

Range and cross range resolution is a factor in determining desired window size. For example, to provide coverage of a 100 m target using range and cross range resolutions of one m would require including at least 100 pixels in both the range and cross range dimension for the interrogating window. If resolution for an input image is unknown, a sliding window can be designed to run multiple times with different sized windows for each iteration to ensure coverage of desired target sizes. An example window with desired target coverage is given in Figure 19.

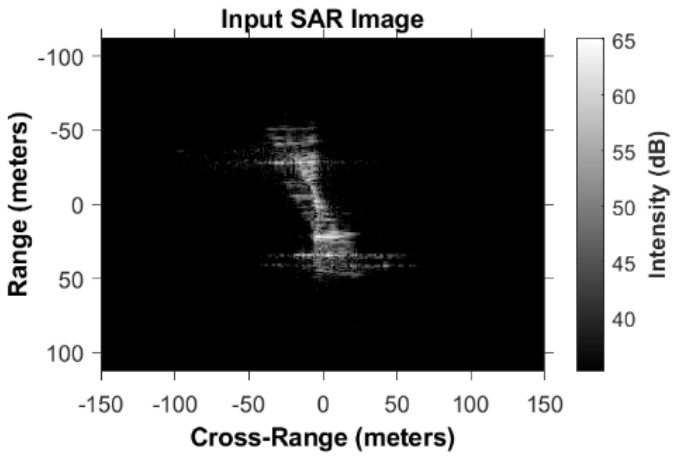


Figure 19. Desired Target Coverage.

The primary targets of interest from the measured datasets used in this study covered roughly 100 m in length (as seen in the range dimension in Figure 19) and the appropriate window size to allow for complete coverage was 224 pixels in range span and 300 pixels in cross range span. This window size allows for the 100 m target to be oriented in any direction and still allows coverage of both the desired target scatters and the background.

Once an appropriate window size is selected, the number of windows used in the cross range and range dimensions is also designated. The windows are designed to overlap with each other to ensure complete coverage of a target. If no overlap was employed, it would be possible to split a target scatter between two interrogating windows and yield only partial coverage of the target. The sliding window algorithm was designed to use displacements equal to half of the window width in both the range and cross range dimensions to ensure complete target coverage [3]. An example of the overlapping sliding window design is shown in Figure 20.

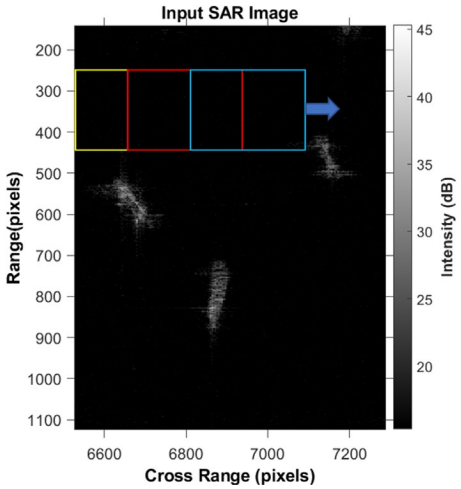


Figure 20. Overlapping Sliding Window Concept.

The MATLAB code can be configured to take the dimensions of the complete input image and automatically designate the number of range and cross range bins that will give maximum coverage.

The following equations govern the relationship between the initial range window and the additional range windows (X) needed to cover the entire range of an input image. The range span is denoted by r, the full image range dimension is denoted by R, the number of windows used in the range dimension is denoted by N_{rw}.

$$r + X\left(\frac{r}{2}\right) = R \tag{4.1}$$

$$N_{rw} = X + 1 \tag{4.2}$$

$$N_{rw} = 2\frac{R-r}{r} + 1 \tag{4.3}$$

The final N_{rw} variable will be rounded down to the nearest whole number to avoid computational error. This results in a minute area of the input image not receiving coverage by the sliding window algorithm. The user can manually design a window to provide coverage of the image border if desired.

Similarly, the number of cross range windows to use resulting in max coverage of an input image is given by:

$$N_{cw} = 2\frac{C-c}{c} + 1 \tag{4.4}$$

where C denotes the full image cross range dimension and c represents the cross-range span. The final number of cross range windows used, N_{cw}, is also rounded down to the nearest whole number to avoid computational error in MATLAB.

Once the number of range and cross range windows is designated, the sliding window algorithm interrogates each window patch individually and performs autofocusing. Two different sliding window algorithms were designed for this project, one employing the traditional PGA autofocusing method, and the other employing the novel AROMA method. The differences in these methods have been previously discussed in Chapter II.

Fifteen iterations of autofocusing are performed on each window patch with the f_1 (Muller-Buffington Sharpness metric) calculated for both the input unfocused window, and the autofocused window after 15 iterations of the designated autofocusing algorithm. The sharpness ratio is calculated using both sharpness metrics per (3.1) and is evaluated against a predetermined sharpness threshold designed to indicate the presence of a moving target.

The nominal sharpness threshold, f_2, used in the code was set to 2.0 [3]. Any interrogated window patch that had an f_1 value greater than the threshold was deemed to have captured a moving target and both the input radar image and the autofocused radar image would be output from the algorithm.

B. CONSOLIDATION

Overlapping window coverage employed in this algorithm is effective to ensure at least one of the windows provided complete coverage of a target of interest. This overlapping coverage, however, often results in multiple adjacent windows passing the sharpness threshold that output redundant images of the same target scatter. The redundant images can consist of complete and partial coverage of a target of interest as seen in Figure 21.

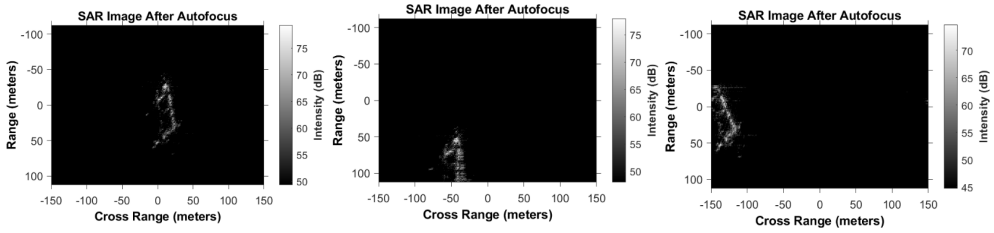


Figure 21. Complete and Partial Target Coverage.

Figure 21 shows output images from the same target scatter that passed the sharpness threshold in the sliding window algorithm. The first image is the most desirable as it contains complete coverage of the target of interest. The two partial outputs contain minimal intelligence value compared to the output with complete coverage.

A consolidation method was designed to account for the redundant images and output a single window that contained the most complete coverage of a target. Several metrics were considered to evaluate an image for completeness.

The intensity squared metric:

$$Intensity^2 = \sum \left| g_{cor}(x,y) \right|^4. \tag{4.5}$$

The intensity metric:

$$Intensity = \sum \left| g_{cor}(x,y) \right|^2. \tag{4.6}$$

The magnitude sum of the pixels of an image, termed in this study as the *sum*:

$$Sum = \sum \left| g_{cor}(x,y) \right|. \tag{4.7}$$

Images with partial and complete scatters from three different targets were evaluated from measured datasets. The intensity squared, intensity, and sum metrics were calculated for each image. The results of these calculations are given in Table 1.

Table 1. Metrics Used for Determination of Optimal Coverage.

		Intensity Squared $\left({}^{V^2}/_{m^2}\right)$	Intensity $({}^{V}/_{m})$	Sum $\left({}^{V^{0.5}}/_{m^{0.5}}\right)$
Target 1	Complete	7.2598×10^{14}	3.044×10^{8}	1.7499×10^{6}
	Partial 1	5.1054×10^{14}	2.7956×10^{8}	1.6686×10^{6}
	Partial 2	5.1681×10^{14}	1.812×10^{8}	1.4268×10^{6}
	Partial 3	1.2335×10^{14}	1.3169×10^{8}	1.374×10^{6}
Target 2	Complete 1	3.059×10^{16}	1.812×10^{9}	3.029×10^{6}
	Complete 2	3.215×10^{16}	1.899×10^{9}	3.037×10^{6}
	Partial 1	2.924×10^{16}	1.765×10^{9}	2.834×10^{6}
	Partial 2	1.745×10^{16}	1.4814×10^{9}	2.5694×10^{6}
	Partial 3	2.052×10^{16}	8.8769×10^{9}	1.836×10^{6}
	Partial 4	6.54×10^{15}	1.0168×10^{9}	2.4818×10^{6}
Target 3	Complete	7.833×10^{16}	1.299×10^{9}	2.3176×10^{6}
	Partial 1	6.112×10^{14}	4.0991×10^{8}	1.948×10^{6}
	Partial 2	2.269×10^{13}	9.910×10^{7}	1.4439×10^{6}
	Partial 3	9.1477×10^{16}	1.139×10^{9}	2.056×10^{6}

The highest intensity squared and intensity metrics nearly always correlated to the image with the most complete target of interest with minor discrepancies. In reference to target 1, the partial 2 image had less of the target scatter present than the partial 1 image, hence it is expected that the more complete image would have a higher value for its completeness metric. However, the intensity squared value for the partial 2 image was higher than the value for the partial 1 image. Similar results were observed with target 2 in partial images 2 and 3 with the lower coverage partial image (partial 3) resulting in a higher intensity squared and intensity metric value. The intensity squared and intensity metric did not accurately reflect a higher value for the more complete partial target during this trial and they were omitted as a completeness metric for this study.

Target 2 samples included two complete target images, ‘Complete 1’ centered in the middle of the image and ‘Complete 2’ with the target positioned off center. The off centered image did have a higher completeness metric value across all three metrics, but it did not impact the evaluated proficiency of the completeness metrics as the off centered complete target still had a higher metric value than all other partial images correlating to target 2.

The highest sum value correlated to the most complete image with no observed discrepancies and was selected as the completeness metric to consolidate redundant outputs from the sliding window.

The sliding window algorithm generates a matrix, referred to as the image index, with dimensions matching the number or windows employed in the range and cross range directions. When an interrogated window patch is above the preset f_2 threshold, the corresponding cell within the image matrix is set to 1. Interrogated patches with f_1 values below the preset threshold are set to 0. An additional matrix matching the dimensions of the image matrix is also generated called the summation index. When an image patch passes the f_2 threshold, the sum of its pixel magnitude is calculated and stored in its corresponding cell within the summation index. The image and summation indexes are then zero padded to allow for iterative row/column calculations within MATLAB.

The consolidation process proceeds by iteratively testing adjacent cells for the value of 1 down each column and row within the image index. When consecutive cells in a row with a value of 1 are found, their corresponding cells within the summation index are compared and the cell index with the higher summation value is accounted for in a new matrix termed the row resolved image index. The row resolved image index has the same dimensions as the zero padded image index and contains the position of cells that had higher summation values when adjacent ones were present on any given row in the original image index. A new column resolved image index is also generated using the same process to consolidate adjacent ones down a given column. To account for isolated windows with no adjacent 1s present, a matrix termed the single resolved image index is generated in which cells that had a one present from the image index with no adjacent ones in neighboring cells would have their row/column bin data stored.

Once row, column, and single resolved image indexes have been populated from the image index, these indexes are summed together. This new matrix is termed the "final resolved image index." The final resolved image index will have a 0 present in cells representing no positive detections, and either a 1 or 2 present in any cell that had a positive detection from the sliding window algorithm after consolidation.

The final process in consolidation will interrogate the final resolved image index along each row and column to look for adjacent cells with a value greater than 0. When two adjacent non-zero cells are found, the cell that has a lower summation value from its correlating position within the summation index will be set to zero within the final resolved image index. This leaves no adjacent cells with non-zero values present, effectively eliminating the probability of outputting redundant images of the same target down each row and column.

Of note, this method does not effectively eliminate redundant outputs of targets that are large enough to take three or more windows in each dimension to cover. This is why an appropriate window size is needed to allow coverage of desired targets of interest.

This consolidation method also fails to eliminate redundant target outputs if cells with a positive value are positioned diagonally from each other within the image index vice horizontally or vertically adjacent. Additional code can be added to select for the diagonal cell with the higher summation value comparable to the row and column selection previously mentioned. This option was not used in this study as most occurrences of singular diagonal cells passing the f_2 threshold correlated with target scatters that were in close proximity to each other. In this case, elimination of a diagonal cell results in one of the closely spaced target scatters being eliminated from the algorithm output field. Electing to not screen for diagonal cells in the consolidation algorithm leads to minimal redundant outputs in certain scenarios, but this is optimal over alternatively eliminating a closely spaced target entirely.

The consolidation algorithm is designed to output both the unfocused and autofocused image for each non-zero cell within the final resolved image index. In addition to these images, the algorithm also outputs the range and cross range pixel index that was

used for the window as well as the range window and cross range window index to facilitate mapping the location of the outputted target to the original image. The algorithm also outputs a detection counter to track the number of targets detected in the input image.

C. DATASETS

1. CSTARS Dataset

The measured datasets used in this study were acquired from the CSTARS Radar Imagery collection, obtained using COSMO-SkyMed (COnstellation of small Satellites for Mediterranean basin Observation) [7]. COSMO-SkyMed is an Italian earth observation system used for both civilian and defense purposes. The satellite constellation consists of X-band radar satellites within a low earth orbit. The CSTARS Imagery employed in this study was collected over maritime environments in vicinity of northern Haiti in 2010 [7].

A total of six radar images represented using in-phase and quadrature data were used in this study. Three images displayed significant maritime activity with numerous target scatterers of interest that were used to evaluate the ability of the sliding window algorithm to perform MTD in a maritime environment on measured datasets. Three images did not display observable maritime activity and were used to evaluate the false alarm rates of the AROMA and PGA based MTD methods. The measured datasets for this study are given in Table 2. The radar images are portrayed with the radar down-range (RG) oriented along the vertical axis and the radar cross-range (CR) oriented along the horizontal axis.

Table 2. CSTARS Measured Dataset Images.

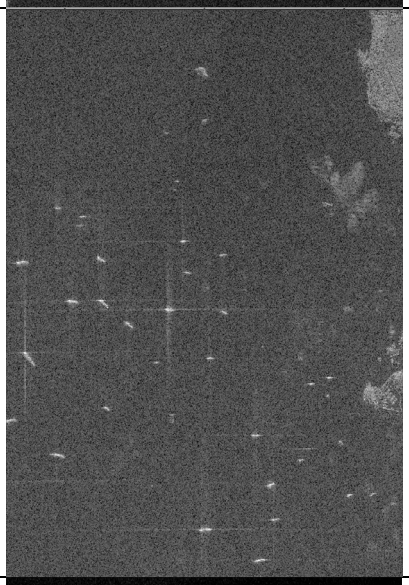
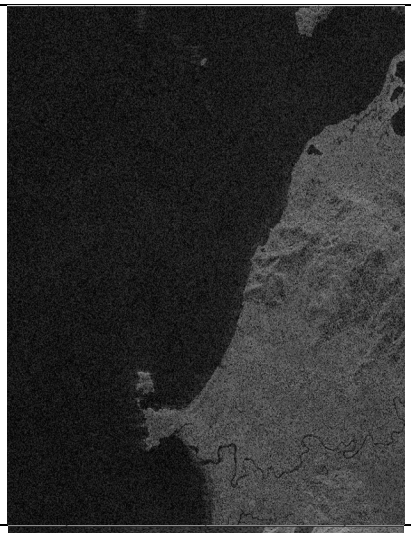
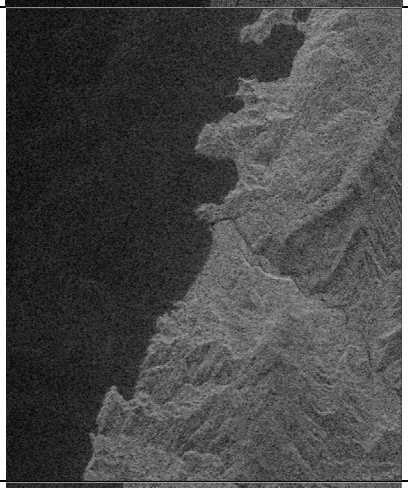
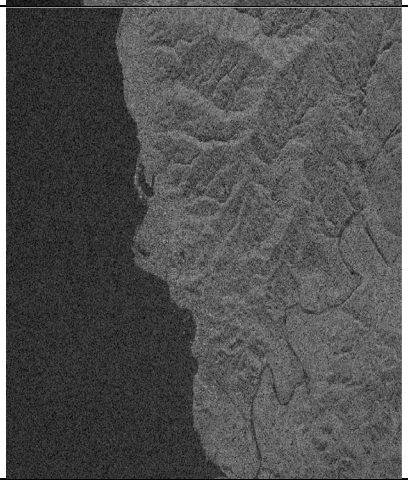
Image Index	Full Radar Input Image	Image Dimensions in pixels (RG × CR)
1		18 647 × 14 383
2		20 432 × 14 397
3		17 076 × 14 358

Image Index	Full Radar Input Image	Image Dimensions in pixels (RG × CR)
4		18 635 × 14 394
5		17 746 × 14 395
6		20 160 × 14 377

"COSMO-SkyMed™ Product - ASI [2010] processed under license from ASI – Agenzia Spaziale Italiana. All rights reserved."

2. Simulated Data

Simulated datasets for this study were generated using a SAR simulator with prescribed scattering centers in MATLAB. This code was written and provided by Professor David Garren for this study. A transmission waveform was simulated using preselected radar sensor parameters and a radar scene was generated by creating target centers and associated background noise. The radar collection parameters were designed so that the down range and cross range resolution would each be 1 m^2.

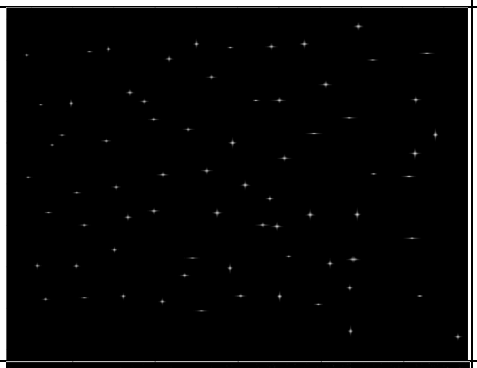
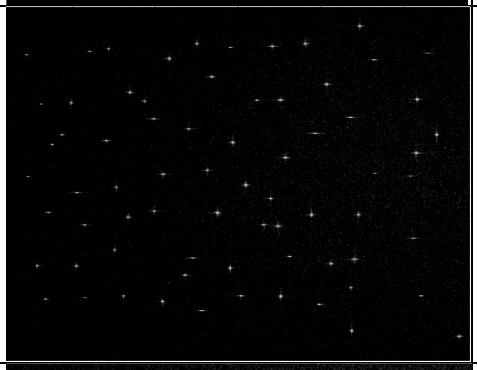
Each scattering center was designed to represent a 1 m × 1 m target and was positioned within the scene with 20–50 pixel spacing between the next closest scattering center. This adequate spacing allows for the detection algorithm to identify singular scattering centers as targets of interest. Directional velocity vectors were also assigned to the scattering centers. Low speed vectors were used (1 m/s or less) to simulate slow moving targets whose azimuthal smearing would not extend and overlap with other scattering centers.

A critique of intensity-based detection algorithms is that their performance becomes degraded when elevated noise is present in the input data [5]. A baseline noise magnitude of 0.0 was used in the simulated target scene to evaluate the AROMA and PGA based MTD algorithms performance in an optimal environment. To evaluate the AROMA based MTD algorithms performance in elevated noise conditions, noise magnitudes of 5.0, 10.0, and 20.0 were also injected into the target scene. The average scattering center magnitude used in the SAR simulator was 10.0. A calculation of SNR can be made by using:

$$SNR = \frac{Scattering\ Center\ Magnitude^2}{Noise\ Magnitude^2} \tag{4.8}$$

The resulting SNR for the four environments evaluated were infinite (no noise injected), 6 dB, 0 dB, and -6 dB for the heavy noise environment. The images generated using the SAR simulator are outlined in Table 3.

Table 3. Simulated Data Images.

Image Index	Full Radar Input Image	SNR	Image Dimensions in pixels (RG × CR)	Observed Targets
1		Infinite	427 × 560	71
2		6.0 dB	427 × 560	71
3		0.0 dB	427 × 560	71

4		-6.0 dB	427 × 560	71

V. ANALYSIS OF RESULTS

A. MEASURED DATASET RESULTS

1. Index 1

The sliding window algorithms designed in this study were evaluated for their detection performance against target scatters present in measured datasets. Index one of the measured datasets exhibited four targets scatters perceivable by visual inspection, each about 100 pixels in length. The interrogating window size selected for the algorithms was 224 pixels in the range dimension and 300 pixels in the cross-range dimension. The detection threshold value of 2.0 was initially used in the AROMA algorithm and the input image was tested using 40 range bins and 90 cross range bins, providing 4500 bins of total coverage. The selection of the input image that was evaluated is shown in Figure 22.

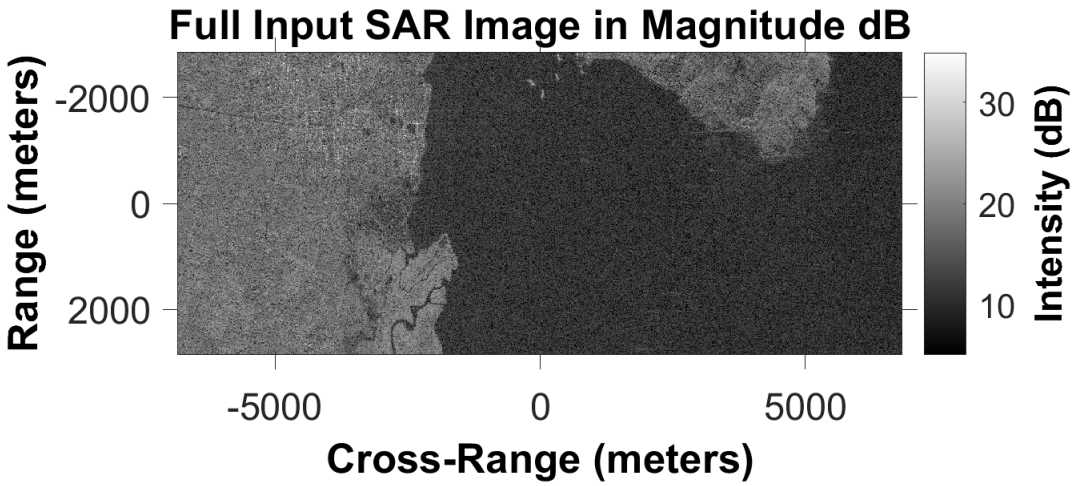


Figure 22. Image Index 1 of Measured Dataset.

The AROMA algorithm was successfully able to detect all four target images of interest using the detection threshold of 2.0. The consolidation algorithm successfully eliminated redundant outputs and there were no false alarms observed in the 4500 windows interrogated. The four target outputs of interest are shown in Figure 23.

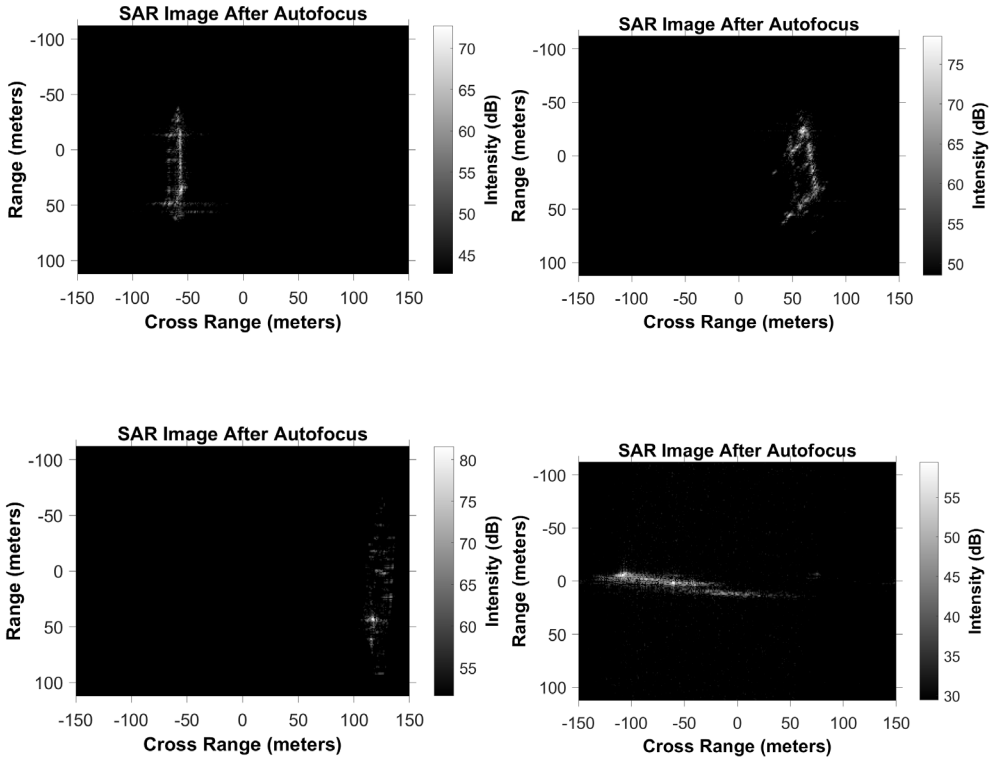


Figure 23. AROMA Detections from Measured Dataset Index 1.

The PGA algorithm was then evaluated for its ability to detect all four targets of interest. The same detection threshold, window size, and coverage area was employed, but one of the four target scatters was not detected.

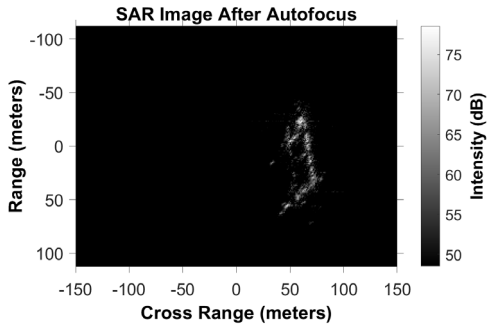


Figure 24. PGA Target below Threshold in Measured Dataset Index 1.

The target displayed in Figure 24 exhibited a sharpness ratio of 1.7203, not sufficiently high to pass the preset detection threshold of 2.0. To ensure that the PGA

algorithm detected all four targets of interest, the detection threshold was lowered to 1.7 and PGA was re-evaluated. With a lower detection threshold, the fourth target of interest was generated as an output image. All four detections of interest for the PGA MTD algorithm are given in Figure 25.

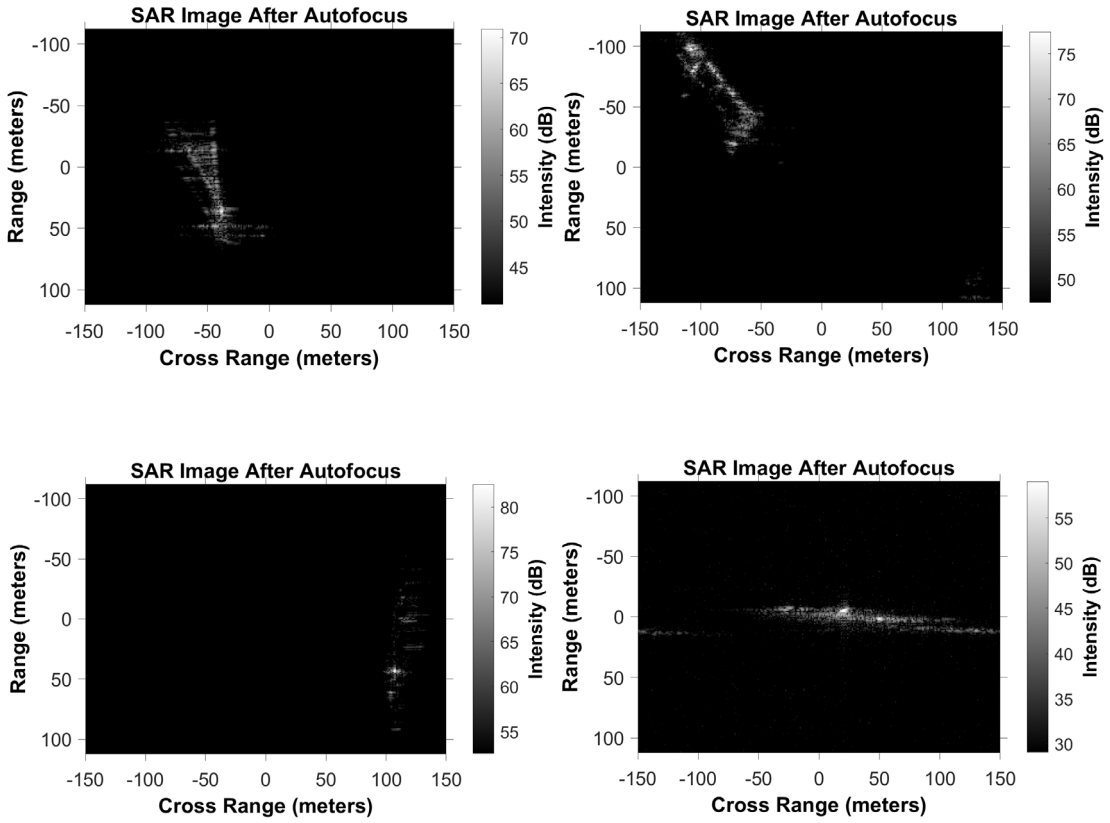


Figure 25. PGA Detections from Measured Dataset Index 1.

The lower detection threshold also resulted in an increased number of false alarms observed, with seven out of 4500 windows classified as false alarms yielding a false alarm rate of 0.156%. The main source of false alarms originated in image patches that included coastal areas, with both water and land background details present. Sample false alarm windows are shown in Figure 26.

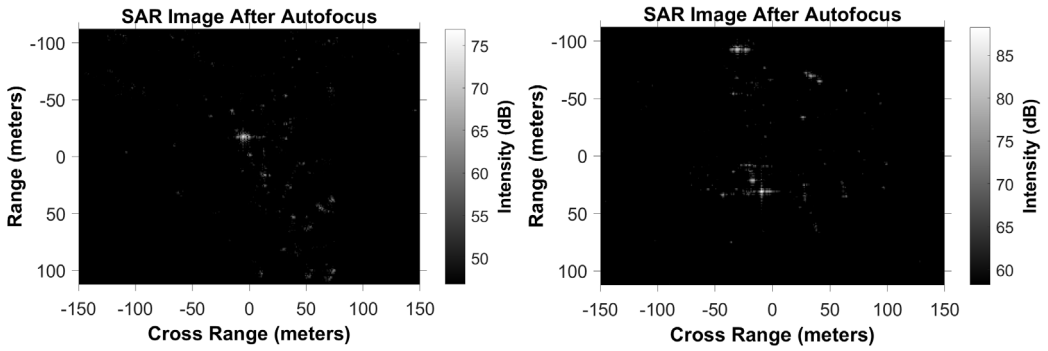


Figure 26. PGA False Alarms from Measured Dataset Index 1.

This study observed higher sharpness ratios provided by the AROMA algorithms compared to the ratios provided by the traditional PGA algorithm. This trend was noticed when evaluating Image Index 1 and was consistent for the remaining measured data images. When analyzing a singular target from Index 1, the sharpness ratio of the AROMA autofocused image compared to the original was 5.365. This same patch when autofocused by the PGA algorithm yielded a sharpness ratio of 1.720. The autofocused outputs of the same target using AROMA and PGA are shown in Figure 27.

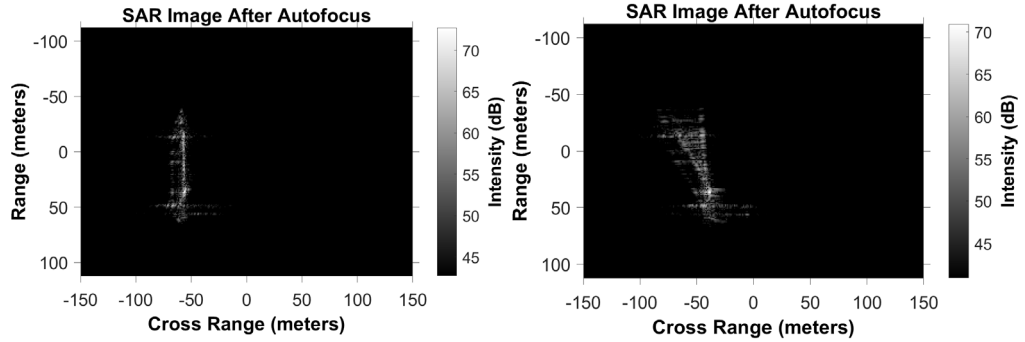


Figure 27. Sharpness Contrast of AROMA and PGA Autofocused Images.

The AROMA autofocused image on the left in Figure 27 exhibits less azimuthal smearing than the PGA autofocused image on the right. This visual observation and the increased sharpness ratio provide evidence that the AROMA algorithm was better able to compensate for the phase error induced by the large vessels of interest compared to the 1D PGA model.

Along with the four large 100-pixel targets of interest, PGA also generated 14 smaller scatterers as output images. Samples of these smaller target outputs are shown in Figure 28.

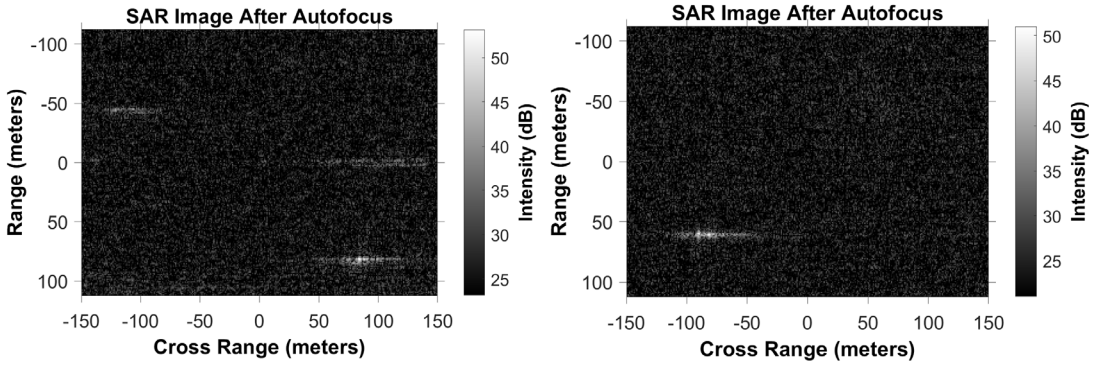


Figure 28. Small Craft Detected by PGA in Measured Dataset Index 1.

These smaller scatters were unable to be classified by inspection but potentially could have been buoys or smaller watercraft. These smaller targets were not the high value large targets of interest meant to be detected by the 224 × 300 window pixel dimensions, yet they could still serve as useful intelligence. In an age where small, unmanned surface vehicles pose a threat to naval operations, it may be beneficial to accurately detect their presence in an open ocean scene. The PGA MTD algorithm detected the desired large target scatters of interest and the smaller target scatters while employing the same window size and threshold. This raised the question of if the AROMA algorithm could also detect these smaller craft if a reduced window size or detection threshold are employed.

The AROMA MTD algorithm was evaluated on the first measured image index again with a reduced window size of 150 pixels in the range dimension and 200 windows in the cross-range dimension. It was discovered that this reduced window size also required a reduced threshold to output image patches with the smaller target scatters as detection outputs. A reduced threshold value of 1.3 was implemented and the coverage area was adjusted. The original 50 × 90 range and cross range bins resulted in a coverage area of 5712 × 13 650 pixels. The reduced window size would require 75 range bins and 135 cross range bins to yield coverage area of 5700 × 13 600 pixels.

Using the new reduced window, reduced detection threshold (1.3), and the new coverage area, AROMA was evaluated on its ability to detect smaller scatters of interest. The AROMA algorithm detected nine smaller target smears that were not recovered using the larger window size in previous trials. The observed false alarm rate was 0.3356% with most false alarms seen in image patches over the coast, where land and sea backgrounds were present. Sample recovered small target smears are shown in Figure 29.

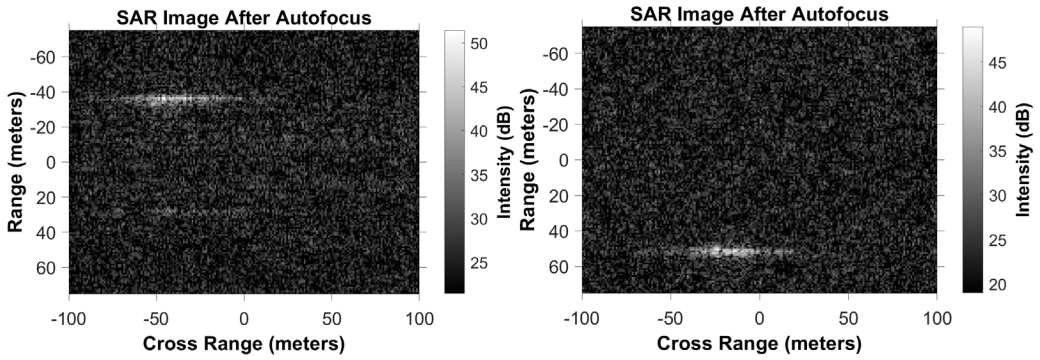


Figure 29. Small Craft Detected by AROMA in Measured Dataset Index 1.

To recover additional small target smears, the detection threshold was reduced again to a value of 1.1 and the same coverage area was interrogated using the AROMA algorithm. There were 28 additional small target smears recovered that were not present in the original 224 × 300 interrogating windows. The observed false alarm rate was 0.68% with most false alarms occurring in image patches over coastal areas.

The reduced window sizes and thresholds employed in the AROMA algorithm could detect small target smears while generating false alarm rates below 1%, still within acceptable levels for a detection algorithm. The objective of this study, however, is to evaluate the performance of the MTD algorithms against larger maritime targets of interest that possess intelligence value. The AROMA algorithm detected all four target smears correlating to targets of interest with zero false alarms compared to 0.156% false alarm rate for the PGA algorithm. The comparative performance of AROMA and PGA MTD algorithms against the measured dataset in Index 1 is summarized in Table 4.

2. Index 2

The second image index from the CSTARS dataset exhibited larger target scatters of interest that would not be adequately covered by the 224 × 300 window size previously employed. A larger window size was selected with 300 pixels used in the range dimension and 600 pixels in the cross-range dimension. An example input target image and autofocused output is shown in Figure 30.

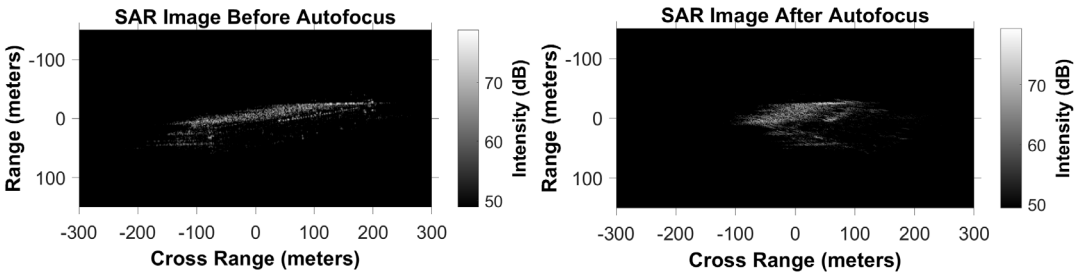


Figure 30. Large Maritime Targets from Measured Dataset Index 2.

Upon initial trials with the AROMA algorithm, the larger windows providing coverage of these target scatters provided much lower sharpness ratios, with the ratios for the above target image ranging from 1.14 to 1.37. It was evident that the detection threshold of 2.0 was too large to allow for effective detection of these larger target scatters and the threshold was lowered to 1.1.

A total of 80 range bins and 40 cross range bins were used to provide coverage over the 38 visible target scatters present in Index 2. The AROMA MTD algorithm was evaluated and produced a total of 50 output images. Sample positive detections are shown in Figure 31. Sample false alarm outputs are shown in Figure 32.

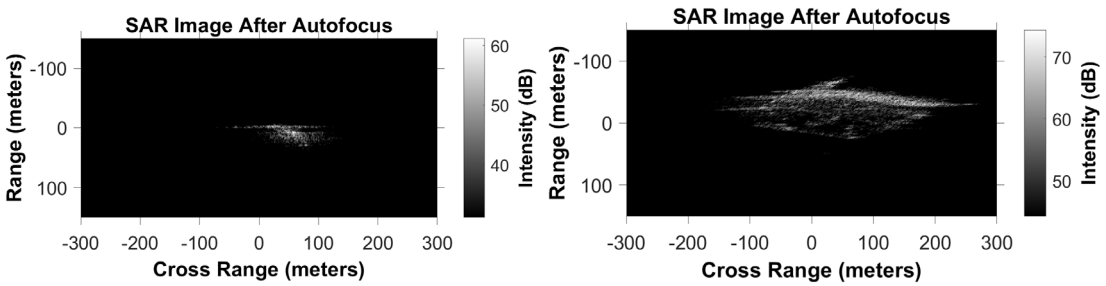


Figure 31. AROMA Detections from Measured Dataset Index 2.

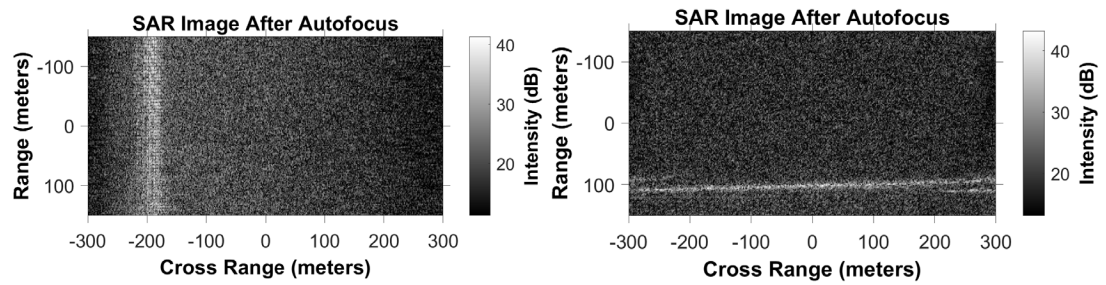


Figure 32. AROMA False Alarms from Measured Dataset Index 2.

A total of 5 output images were classified upon inspection, resulting in a false alarm rate of 0.3125%. The AROMA MTD algorithm was sucessfully able to detect all 38 target scatters of interest with seven cases of redundant scatters. From analyzing the Resolved Image Index, these seven sets of windows that passed the sharpness threshold were aligned diagonally in a way that they were not eliminated by the consolidation algorithm. When higher threshold values are employed (2.0), the off diagonal autofocused images of the same target do not pass this threshold and are not included in the final output windows.

The PGA algortihm was evaluated on Image Index 2 using the same increased window size (300 × 600 pixels), the same lowered threshold and number of interrogating windows to provide the same coverage area. PGA produced a total of 68 output images. Sample positive detections are shown in Figure 33. Sample false alarm outputs are shown in Figure 34.

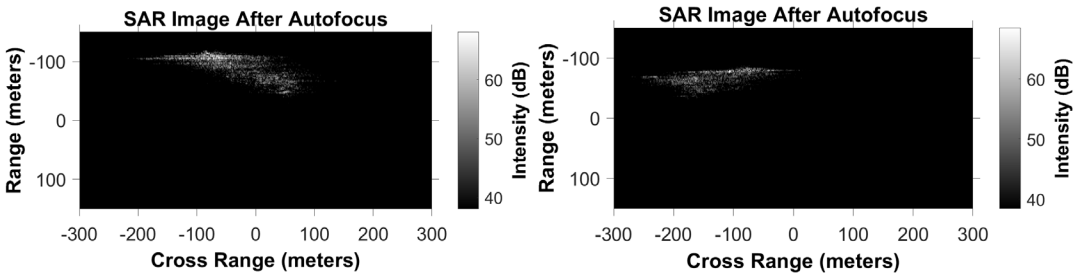


Figure 33. PGA Detections from Measured Dataset Index 2.

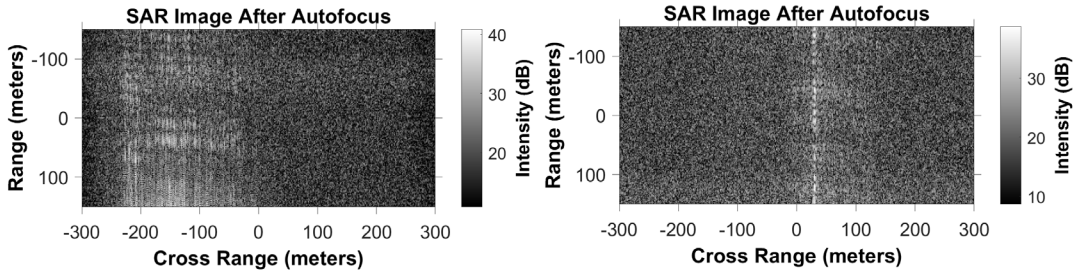


Figure 34. PGA False Alarms from Measured Dataset Index 2.

Of the 68 detections made by the PGA algorithm, 29 were classified as false alarms, resulting in a false alarm rate of 0.906%. PGA was able to successfully detect all 38 target scatters present in the input image with 5 cases of a redundant target output.

Both AROMA and PGA detected all targets of interest visible in Image Index 2, with AROMA producing two more redundant images than PGA. The false alarm rate for the PGA algorithm was within acceptable levels but was still 0.75% higher than the false alarm rate of the AROMA algorithm. The comparative performance of AROMA and PGA MTD algorithms against the measured dataset in Index 2 is summarized in Table 4.

3. Index 3

CSTARS Index 3 contained targets of interest that required larger window coverage of 300 pixels in the range dimension and 600 pixels in the cross-range dimension. Coverage of 33 visible targets of interest was acquired by using 106 interrogating range bins and 46 cross range bins. A lower detection threshold value of 1.1 was employed to evaluate the performance of the larger interrogating windows as was conducted in Index 2.

The AROMA algorithm generated 55 total output detections. Sample positive detections are shown in Figure 35. Sample false alarm outputs are shown in Figure 36.

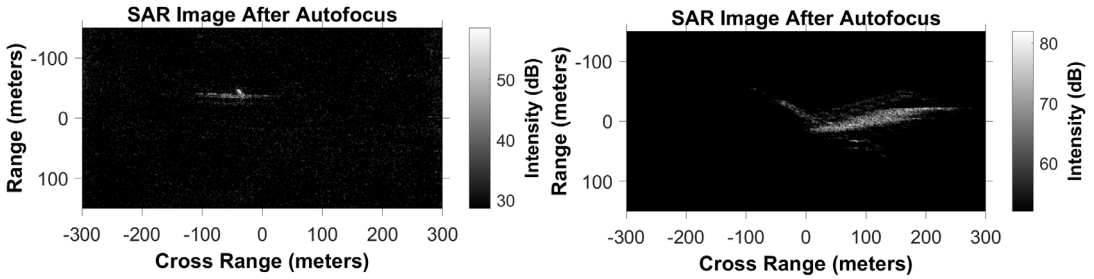


Figure 35. AROMA Detections from Measured Dataset Index 3.

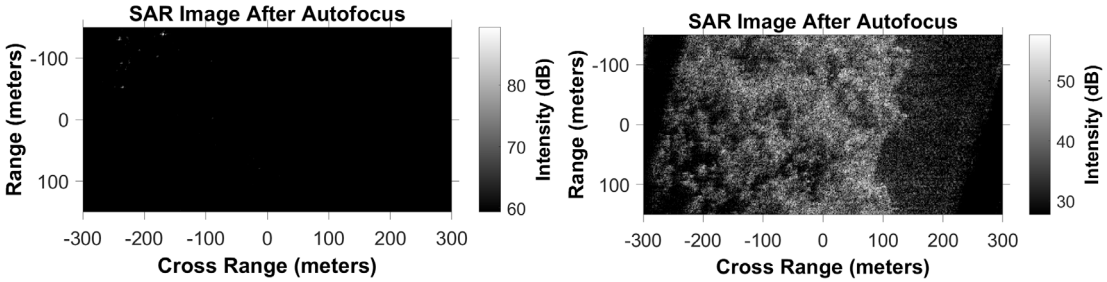


Figure 36. AROMA False Alarms from Measured Dataset Index 3.

Of the 55 output images, seven were classified as false alarms, resulting in a false alarm rate of 0.1436%. Six of the false alarms were output from windows interrogating coastal regions with both land and sea backgrounds present, similar to the false alarms observed from the PGA MTD evaluation of Index 1. Out of the 33 targets visible in the image, 31 were detected by the AROMA algorithm resulting in a detection rate of 93.94%. Upon inspection, the targets missing from the detection count would have required a detection threshold of less than 1.05 to be selected as outputs, a low threshold value that would increase the number of false alarms observed in vicinity of the coastal region of the input image.

The PGA MTD algorithm was evaluated using the same window size, detection threshold, and coverage area. PGA generated 57 total output images. Sample positive detections are shown in Figure 37. Sample false alarm outputs are shown in Figure 38.

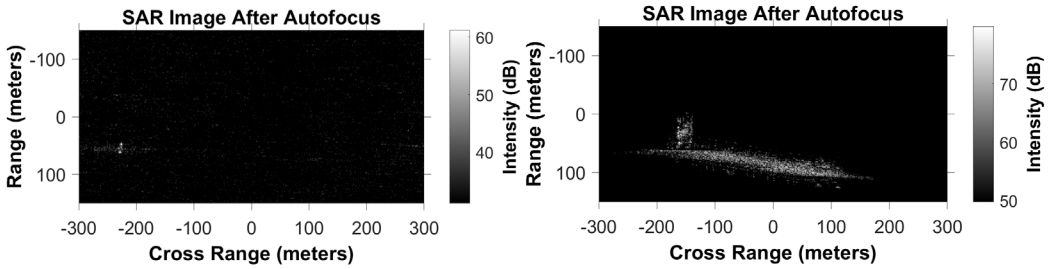


Figure 37. PGA Detections from Measured Dataset Index 3.

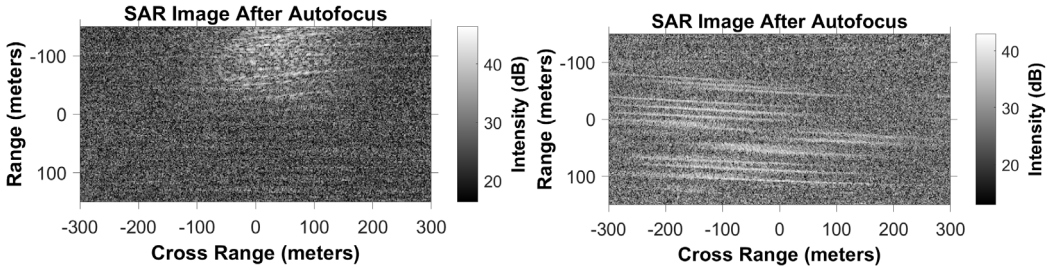


Figure 38. PGA False Alarms from Measured Dataset Index 3.

Of the 57 total detections, 10 were classified as false alarms resulting in a false alarm rate of 0.205%. This false alarm rate is within an acceptable level but is still 0.0614% higher than the false alarm rate observed from the AROMA algorithm. Only 29 of the 33 targets visible in the input image were detected by the PGA algorithm, resulting in a detection rate of 87.87%. There were 18 cases of redundant image outputs due to the diagonal orientation of output windows in the Image Index that were not eliminated by the consolidation algorithm.

The AROMA algorithm exhibited a higher detection rate than the PGA algorithm by 6.07%. The sub 90% detection rate for the PGA algorithm is not acceptable for adequate MTD performance. AROMA also exhibited a lower false alarm rate while both the PGA and AROMA false alarm rates were within acceptable levels. The comparative performance of AROMA and PGA MTD algorithms against the measured dataset in Index 3 is summarized in Table 4.

4. Indexes 4–6

Indexes 4 through 6 of the CSTARS dataset exhibited no target scatters when visually inspected, reference Table 4. These datasets were evaluated using the AROMA and PGA algorithms to test false alarm rates over a large maritime and coastal environment absent of targets of interest.

Index 4 was interrogated using the nominal window size of 224 pixels in the range dimension and 300 pixels in the cross-range dimension. A total of 2500 windows were employed, giving the same coverage area for both the AROMA and PGA trials. The AROMA algorithm was evaluated using a detection threshold of 2.0 and resulted in no positive detections as expected, and no false alarms. The PGA algorithm was evaluated using the adjusted threshold of 1.7 as was used in previous trails for optimum detection performance. One false alarm was generated by the PGA algorithm, resulting in a false alarm rate of 0.04%. This singular false alarm is shown in Figure 39.

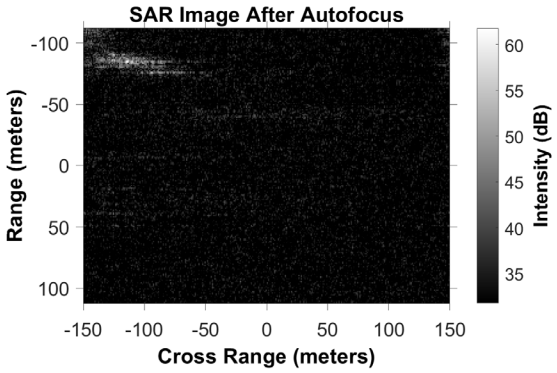


Figure 39. Singular PGA False Alarm from Measured Dataset Index 4.

Index 5 was interrogated using the standard window size of 224 pixels in the range dimension and 300 pixels in the cross-range dimension. A total of 50 range bins and 50 cross range bins were employed providing 2500 windows of coverage. The nominal detection threshold value of 2.0 was employed for the AROMA trial and a reduced threshold of 1.7 was employed for PGA. Both AROMA and PGA failed to generate an output upon interrogating Index 5, resulting in a 0% false alarm rate.

Index 6 was interrogated using the standard window size of 224 pixels in the range dimension and 300 pixels in the cross-range dimension. There were 50 range bins and 50 cross range bins employed to give 2500 windows of coverage. The standard detection values were employed, 2.0 for AROMA and 1.7 for PGA. The AROMA algorithm generated no false alarm outputs whereas the PGA trial generated five false alarms resulting in a false alarm rate of 0.25%. Four of the five false alarms exhibited a sharpness ratio above 2.0 with the highest value observed to be 7.19. These values would have still resulted in false alarms outputs if the nominal detection threshold value of 2.0 had been employed for the PGA trail. Sample false alarms are seen in Figure 40.

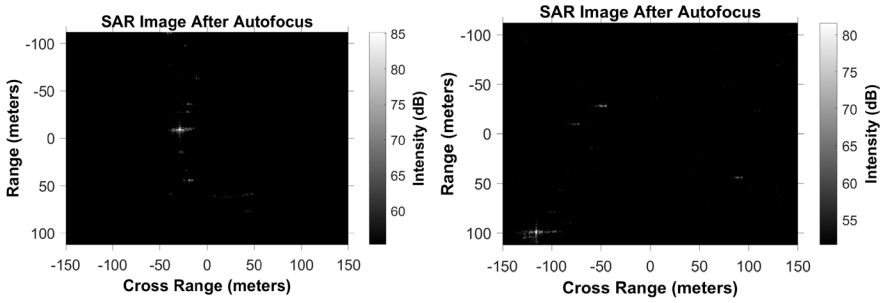


Figure 40. PGA False Alarms from Measured Dataset Index 6.

All the false alarms output by the PGA algorithm from Index 6 occurred in vicinity of a coastal area, consistent with high probability areas for false alarms observed in other trials. The comparative performance of AROMA and PGA MTD algorithms against the measured dataset in Indexes 4–6 is summarized in Table 4.

Table 4. Measured Data Performance for AROMA and PGA MTD Algorithms

Measured Data		
	AROMA	PGA
Index 1		
Window Size	224 × 300	224 × 300
Interrogating Windows	4500	4500
Detection Threshold	2	1.7
Target Scatters Present	4	4
Detection Count	4	4
Detection Rate	100.00%	100.00%
False Alarm Count	0	7
False Alarm Rate	0.00%	0.16%
Index 2		
Window Size	300 × 600	300 × 600
Interrogating Windows	3200	3200
Detection Threshold	1.1	1.1
Target Scatters Present	38	38
Detection Count	38	38
Detection Rate	100.00%	100.00%
False Alarm Count	5	29
False Alarm Rate	0.16%	0.91%
Index 3		
Window Size	300 × 600	300 × 600

Measured Data		
Interrogating Windows	4876	4876
Detection Threshold	1.1	1.1
Target Scatters Present	33	33
Detection Count	31	29
Detection Rate	93.94%	87.87%
False Alarm Count	7	10
False Alarm Rate	0.14%	0.21%
Index 4		
Window Size	224 × 300	224 × 300
Interrogating Windows	2500	2500
Detection Threshold	2	1.7
Target Scatters Present	0	0
False Alarm Count	0	1
False Alarm Rate	0.00%	0.04%
Index 5		
Window Size	224 × 300	224 × 300
Interrogating Windows	2500	2500
Detection Threshold	2	1.7
Target Scatters Present	0	0
False Alarm Count	0	0
False Alarm Rate	0.00%	0.00%
Index 6		

Measured Data		
Window Size	224 × 300	224 × 300
Interrogating Windows	2500	2500
Detection Threshold	2	1.7
Target Scatters Present	0	0
False Alarm Count	0	5
False Alarm Rate	0.00%	0.25%
Overall Detection Rate	97.33%	94.66%
Overall False Alarm Rate	0.00060%	0.00259%

B. SIMULATED DATASET RESULTS

1. Infinite SNR (No Noise)

The AROMA and PGA based MTD algorithms were tested using simulated input data with no injected background noise. Upon review of the sharpness ratios provided from the autofocused images of the simulated targets, it was determined that 2.0 was too large of a threshold value to use for accurate detection of moving target smears. A threshold value of 1.11 was used to screen for moving target smears against a noiseless background. The simulated input image contained 78 target scatters, see Figure 41.

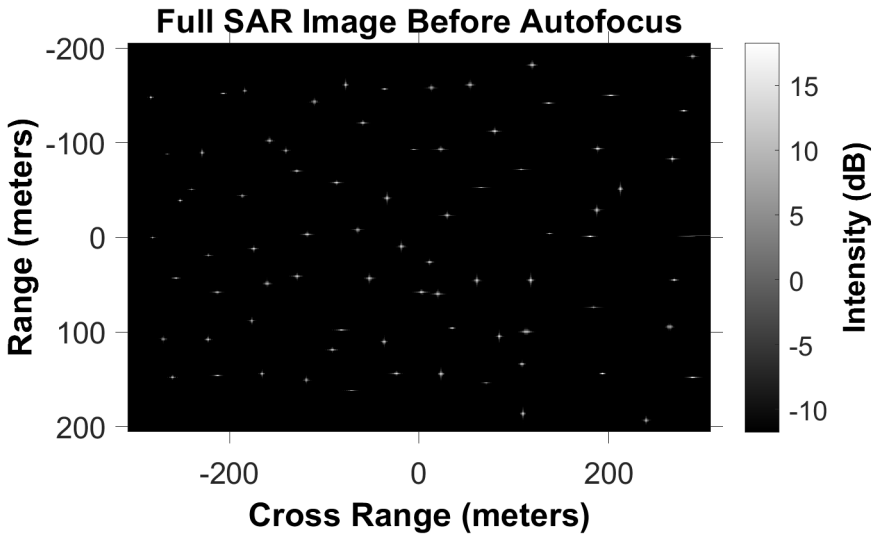


Figure 41. Input Image from Simulated Dataset, Infinite SNR.

The window size for the AROMA MTD trial was 20 pixels in the range span and 30 pixels in the cross-range span. A total of 40 range and 40 cross range windows were employed to provide coverage of the 78 targets. The AROMA MTD algorithm provided 105 total detections using a threshold value of 1.11. Sample positive detections are shown in Figure 42. Sample false alarm outputs are shown in Figure 43.

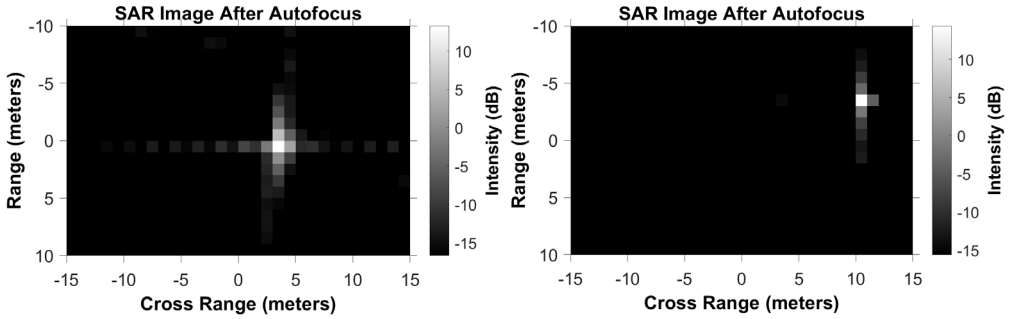


Figure 42. AROMA Detections in Noiseless Environment.

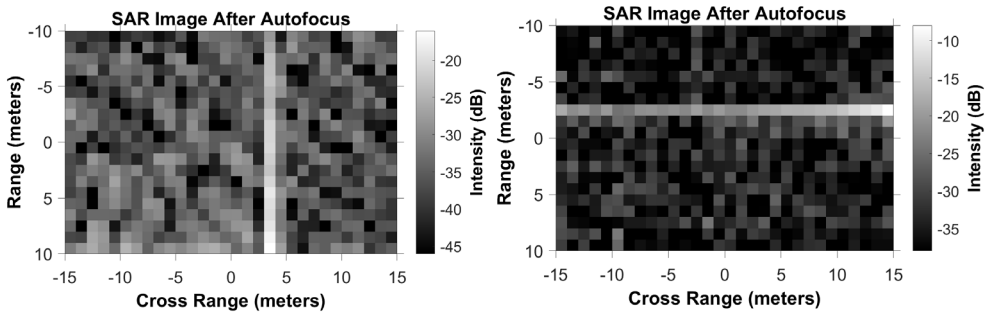


Figure 43. AROMA False Alarms in Noiseless Environment.

Of the 105 detections, 28 were visibly classified as false alarms, resulting in a false alarm rate of 1.75%. The 77 remaining outputs contained the desired autofocused images of target scatters. Two of the output windows captured multiple targets within the same patch due to the large size of the window and the proximity of the targets. A total of 76 of the 78 targets of interest were detected using the AROMA MTD algorithm, resulting in a detection rate of 97.436%. The windows containing the remaining two targets still passed the detection threshold but were not included in the final output after consolidation. These targets were directly adjacent to other target windows, causing the consolidation algorithm to only output one of the adjacent targets in both cases.

The PGA MTD algorithm was then run on the same input image employing the same window size and threshold conditions used in the AROMA trial. A total of 99 total detections were made using PGA. Sample positive detections are shown in Figure 44. Sample false alarm outputs are shown in Figure 45.

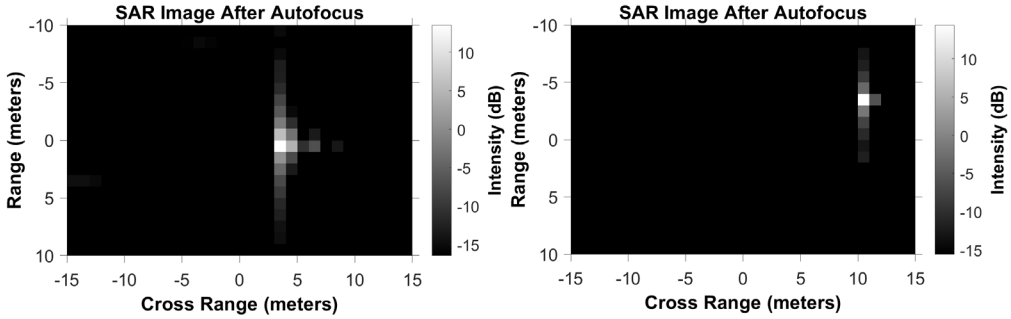


Figure 44. PGA Detections in Noiseless Environment.

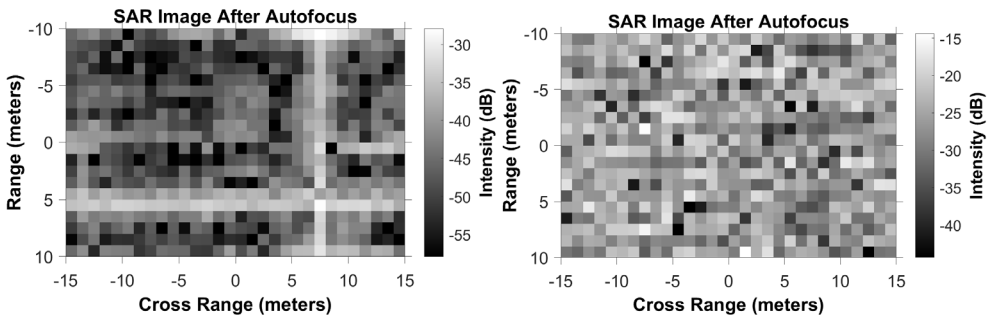


Figure 45. PGA False Alarms in Noiseless Environment.

Of the 99 detections, 30 resulted in false alarms, which yielded a false alarm rate of 1.875%, 0.125% higher than the false alarm rate of the AROMA algorithm when evaluated under the same conditions. The PGA algorithm yielded 69 output windows with positive detections of target scatters, resulting in a detection rate of 89.74%, 7.67% lower than the AROMA algorithm. The comparative performance of AROMA and PGA algorithms against a noiseless simulated scene is summarized in Table 5.

2. SNR of 6.0 dB

As noise was injected into the target images, an additional window size was designed to evaluate detection performance. There were two instances in the noiseless AROMA trial when a window containing a target passed the detection threshold but was eliminated by the consolidation algorithm because it was adjacent to a window containing a separate target scatter. Smaller window sizes would allow for additional window coverage of the same target and allow for more window spacing between adjacent scatters, improving the chance of avoiding elimination by the consolidation algorithm. The new smaller windows used 14 pixels in the range span and 20 pixels in the cross-range span. The smaller 14 × 20, and the original 20 × 30 window sizes were evaluated in a noisy environment.

The noise magnitude was set to 5.0 to generate a radar scene representing a maritime environment of relatively low sea state with minimal observable background clutter. The observed SNR in this scene was 6.0 dB. The AROMA algorithm was run on the input image using the original window size of 20 range pixels by 30 cross-range pixels.

The same number of windows in the range and cross range dimension were used as in the noiseless trial to include all 78 target scatters in the input image. The input SAR image is observed in Figure 46.

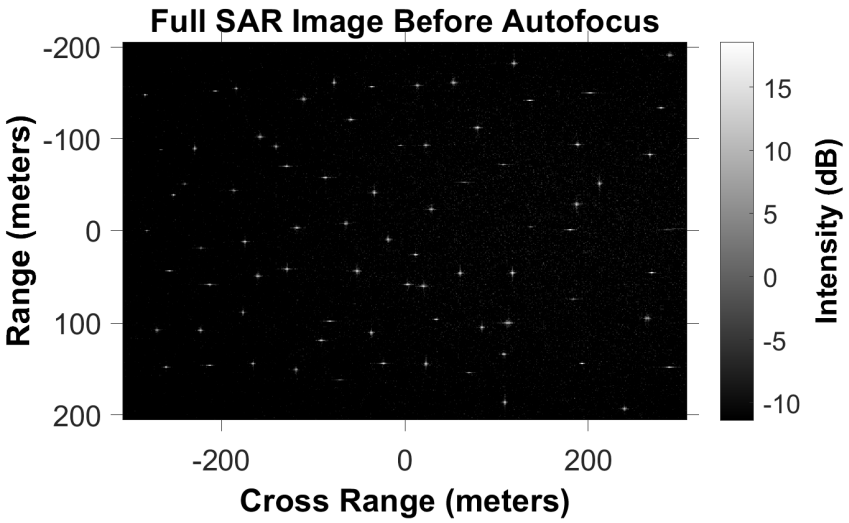


Figure 46. Input Image from Simulated Dataset, SNR of 6.0 dB.

The AROMA algorithm generated 89 total detections. Sample positive detections are shown in Figure 47. Sample false alarm outputs are shown in Figure 48.

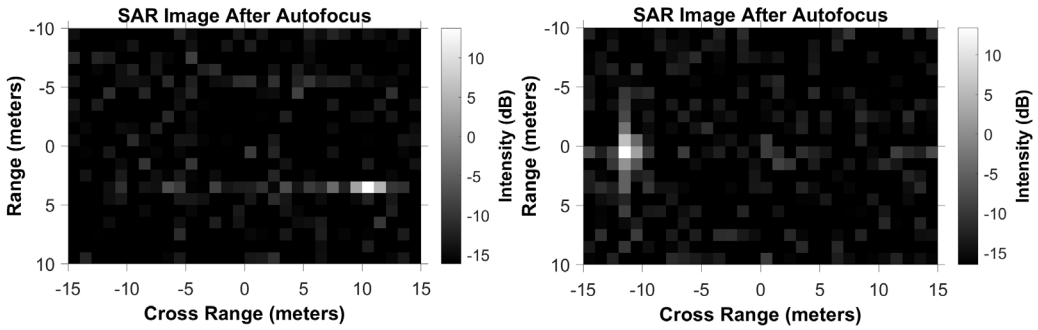


Figure 47. AROMA Detections with SNR of 6.0 dB.

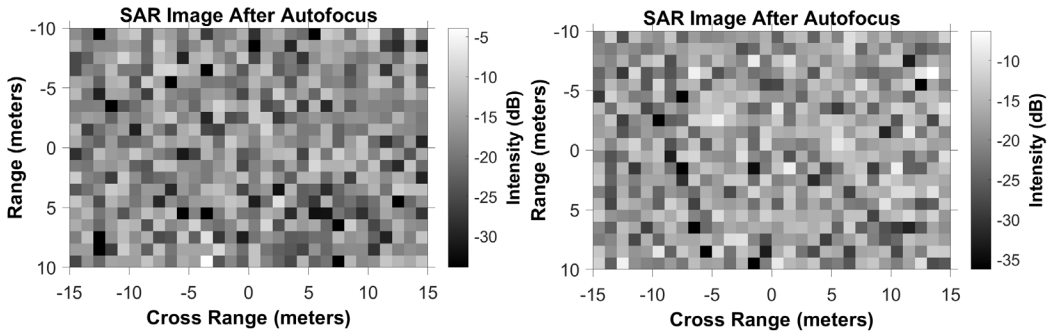


Figure 48. AROMA False Alarms with SNR of 6.0 dB.

Of the 89 total detections, two were visually classified as false alarms resulting in a false alarm rate of 0.125%. AROMA was able to detect 77 of the 78 target scatters, resulting in a detection rate of 98.72%. 10 redundant target images were output due to the diagonal position of windows passing the detection threshold.

The PGA MTD algorithm was conducted using the same threshold, window dimensions, and window coverage. PGA generated a total of 155 detections. Sample positive detections are shown in Figure 49. Sample false alarm outputs are shown in Figure 50.

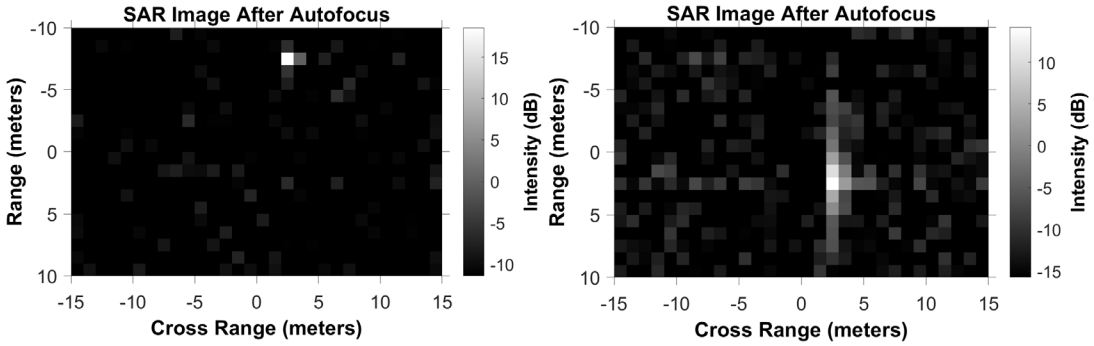


Figure 49. PGA Detections with SNR of 6.0 dB.

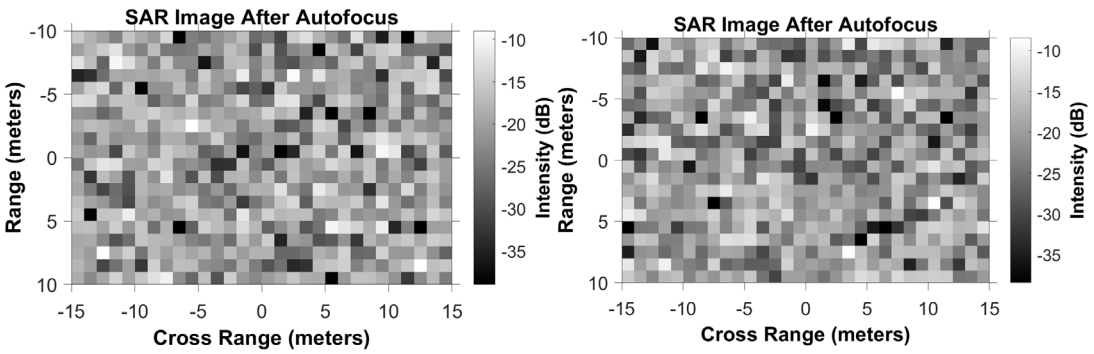


Figure 50. Figure 50. PGA False Alarms with SNR of 6.0 dB.

Of the 155 detections, 71 were visually classified as false alarms resulting in a false alarm rate of 4.43%, 4.3% higher than the false alarm rate of the AROMA algorithm. A total of 73 targets were detected by the algorithm out of the 78 present in the input image, resulting in a detection rate of 93.59%, 3.85% lower than the AROMA algorithm. There were 11 instances of redundant images of the same target scatter. The comparative performance of AROMA and PGA algorithms against a low noise simulated scene is summarized in Table 5.

The AROMA based sliding window algorithm was run on the input image using the smaller 14 × 20 window size. A total of 60 windows were used in the range direction and 55 windows used in the cross-range direction to achieve coverage of 71 simulated target scatters. The input target scene is shown in Figure 51.

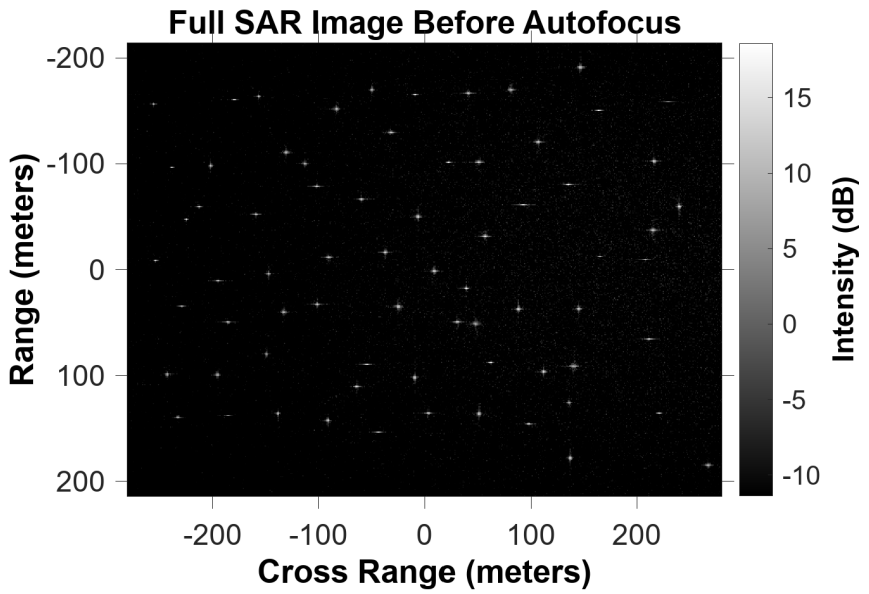


Figure 51. Input Image from Simulated Dataset Using Reduced Window Size, SNR of 6.0 dB.

The AROMA MTD algorithm made 100 total detections using the smaller interrogating windows. Sample positive detections are shown in Figure 52. Sample false alarm outputs are shown in Figure 53.

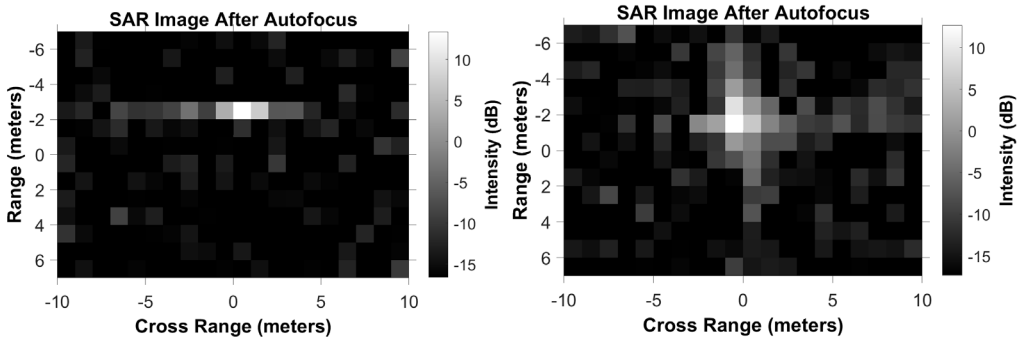


Figure 52. AROMA Detections Using Reduced Window Size, SNR of 6.0 dB.

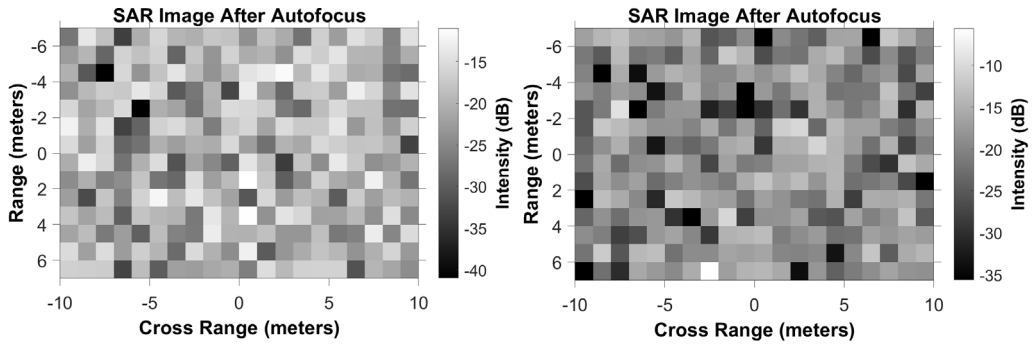


Figure 53. AROMA False Alarms Using Reduced Window Size, SNR of 6.0 dB.

Of the 100 detections made, 23 were classified as false alarms resulting in a false alarm rate of 0.697%. AROMA detected 69 of the available 71 target scatters, resulting in a detection rate of 97.183%. A total of eight instances of redundant target outputs were observed.

The PGA MTD algorithm was run with the same reduced window size. PGA generated a total of 342 detections. Sample positive detections are shown in Figure 54. Sample false alarm outputs are shown in Figure 55.

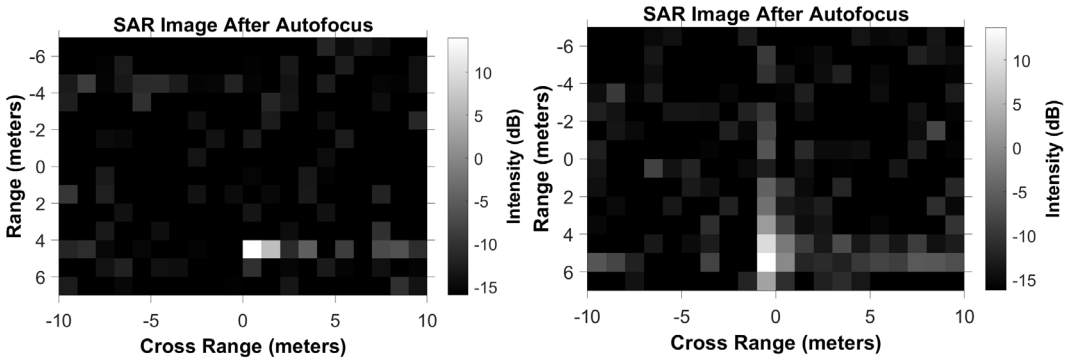


Figure 54. PGA Detections Using Reduced Window Size, SNR of 6.0 dB.

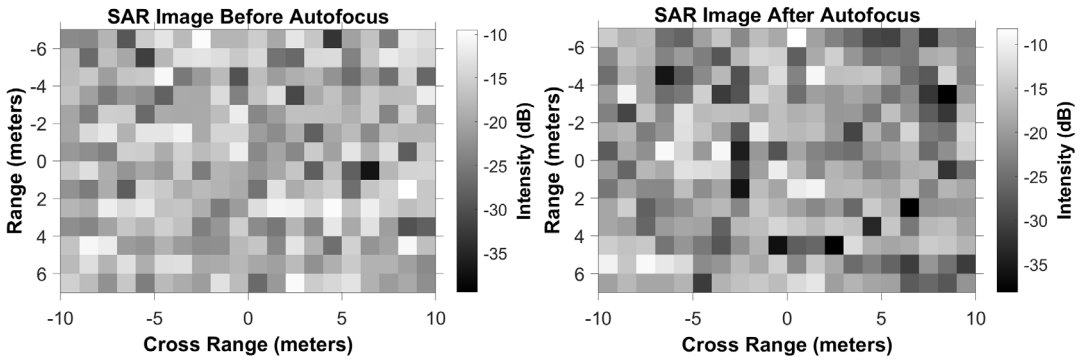


Figure 55. PGA False Alarms Using Reduced Window Size, SNR of 6.0 dB.

Using the same detection threshold as the AROMA trial, PGA produced significantly more false alarms. A total of 267 false alarms were observed, resulting in a false alarm rate of 8.09%. This percentage would have been much higher if the consolidation algorithm had not effectively reduced the number of output false alarm windows. A total of 69 targets of the 71 present in the input image were detected by the PGA algorithm with six redundant output images, resulting in a detection rate of 97.18%. This was the same detection rate as the AROMA algorithm under the same conditions but the false alarm rate for the PGA algorithm was greater than the AROMA algorithm by 7.39%. The false alarm rate was unacceptable for MTD standards with 3.5 false alarms present for every positive identification.

To lower the false alarm rate of the PGA algorithm to acceptable levels, the detection threshold was raised to 1.6 and an additional PGA MTD trial was conducted using the same window size and image coverage. A total of 106 detections were made with 33 detections identified as false alarms for a false alarm rate of 1.0%. Out of the 71 target scatters in the image, the PGA with the elevated threshold detected 68 targets with five redundant image outputs. The detection rate was 95.77% for the PGA algorithm with an elevated threshold, still lower than 97.18% detection rate of the AROMA algorithm with the original threshold value. The performance of AROMA and PGA algorithms with the reduced window size operating in a simulated low noise environment is summarized in Table 6.

3. SNR of 0.0 dB

The noise magnitude was increased to 10.0 to simulate slightly turbulent sea states yielding an observed SNR of 0.0 dB. The performance of the AROMA and PGA MTD algorithms was evaluated using a nominal window size of 20 × 30 pixels to give coverage of 78 target scatters as seen in Figure 56.

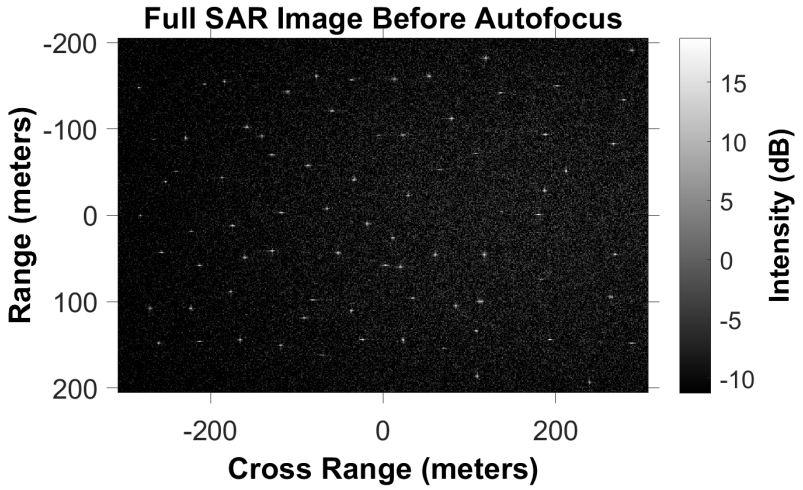


Figure 56. Input Image from Simulated Dataset, SNR of 0.0 dB.

The AROMA MTD algorithm recovered 88 total detections, one of which was identified as being a false alarm. Sample positive detections are shown in Figure 57. The singular false alarm output is shown in Figure 58.

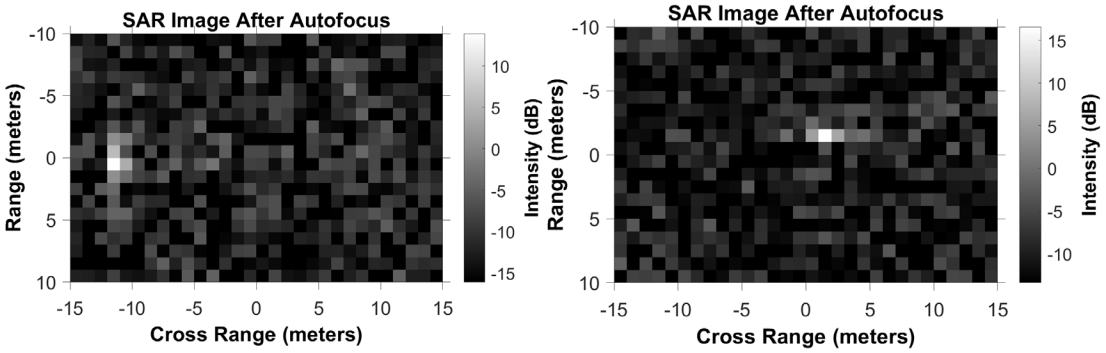


Figure 57. AROMA Detections with SNR of 0.0 dB.

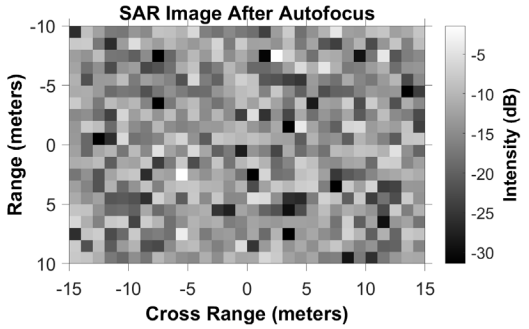


Figure 58. Figure 58. AROMA False Alarm with SNR of 0.0 dB.

The observed false alarm rate for the larger window size in a medium sea state was 0.0625%. Of the 78 target scatters present, 76 were accounted for using the detection algorithm resulting in a detection rate of 97.44%. As was the case when zero noise power was introduced, a target successfully identified by the AROMA MTD algorithm was eliminated via the consolidation algorithm, bringing the positive detection count from 77 to 76. The diagonal position of windows passing the detection threshold resulted in 11 redundant outputs from the consolidation algorithm.

The PGA algorithm evaluated using the same input image, window size, and coverage resulted in 166 total detections. Sample positive detections are shown in Figure 59. Sample false alarm outputs are shown in Figure 60.

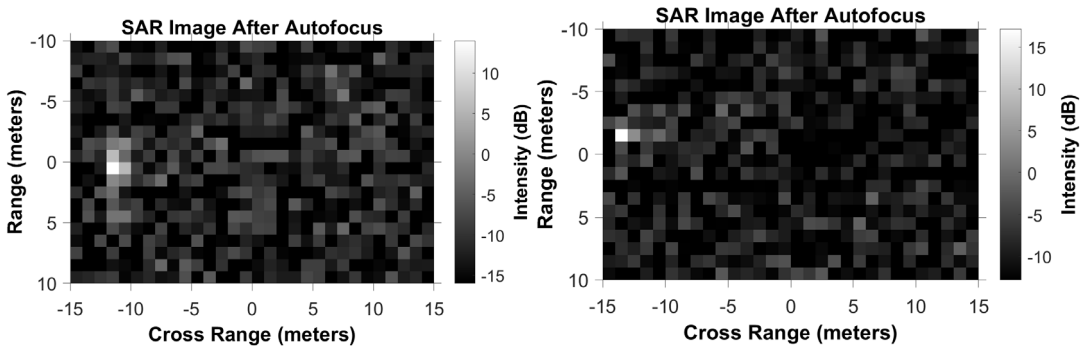


Figure 59. PGA Detections with SNR of 0.0 dB.

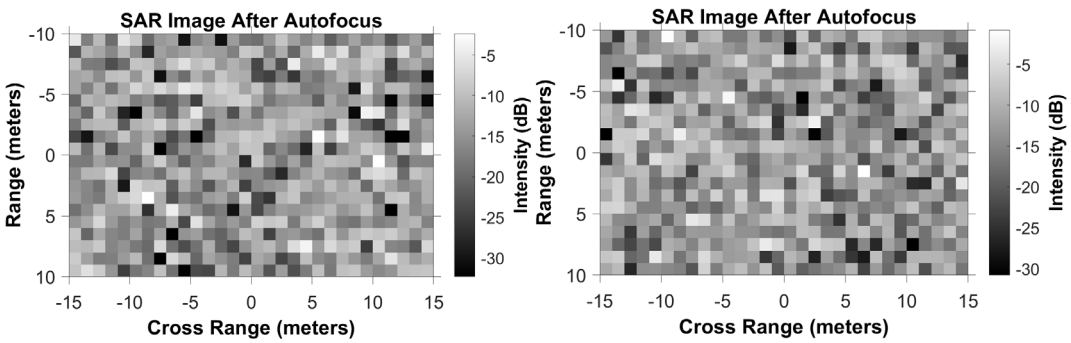


Figure 60. PGA False Alarms with SNR of 0.0 dB.

Of the 166 detections, 86 were visually identified to be false alarms. The resulting false alarm rate of 5.375% was 5.31% greater than the false alarm rate of the AROMA algorithm used in the same conditions with the same threshold. Of the 78 targets present in the input image, 72 were successfully identified resulting in a detection rate of 92.31%. A total of eight output windows displayed redundant target scatters.

As expected, higher noise magnitudes in the simulated environment resulted in lower detection rates for both the AROMA and PGA algorithms. The AROMA algorithm outperformed the PGA algorithm in terms of both false alarm rate and detection rate. The performance of both algorithms with the nominal window size operating in a simulated medium noise environment is summarized in Table 5.

Reduced window sizes were employed in the same medium noise environment for both the AROMA and PGA MTD algorithms. Window sizes of 14 × 20 pixels were employed to give coverage of 71 target scatters as seen in Figure 61.

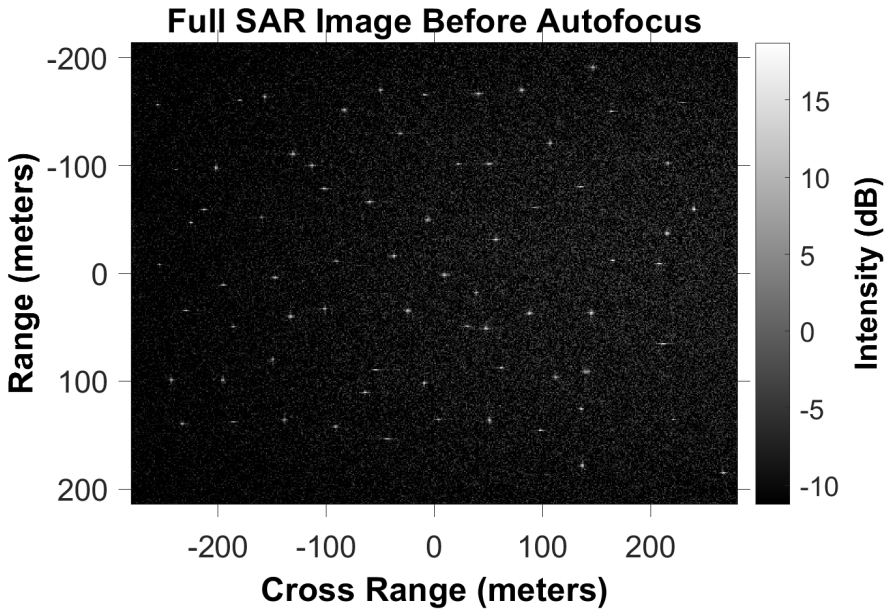


Figure 61. Input Image from Simulated Dataset Using Reduced Window Size, SNR of 0.0 dB.

The AROMA MTD algorithm employing the smaller window size produced 96 total detections. Sample positive detections are shown in Figure 62. Sample false alarm outputs are shown in Figure 63.

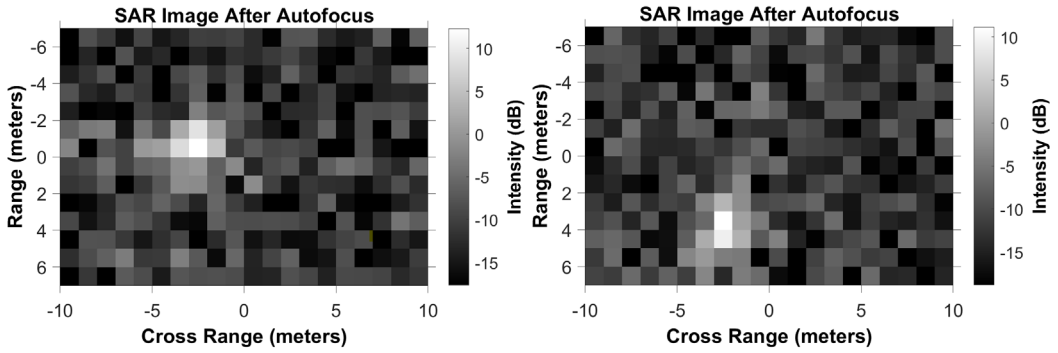


Figure 62. AROMA Detections Using Reduced Window Size, SNR of 0.0 dB.

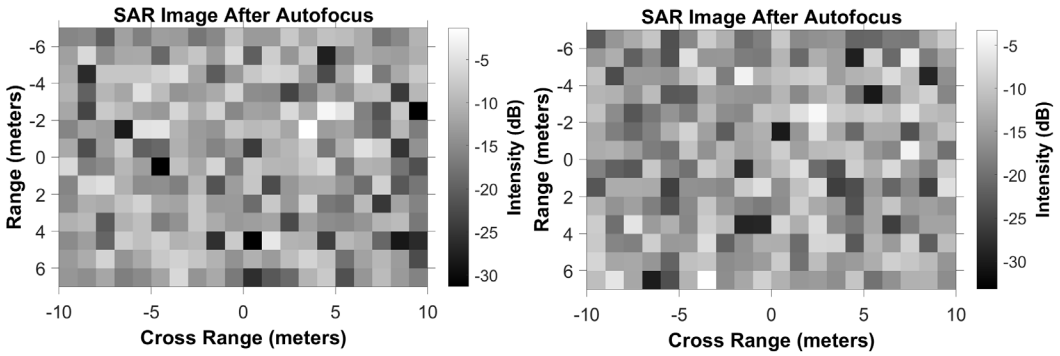


Figure 63. AROMA False Alarms Using Reduced Window Size, SNR of 0.0 dB.

Of the 96 detections, 18 were visually classified as false alarms resulting in a false alarm rate of 0.545%. A total of 66 target scatters were detected out of the 71 present in the input image, resulting in a detection rate of 92.9%. A total of 12 output images were observed to be redundant target detections.

The PGA algorithm when evaluated under the same noise conditions, window size, and coverage produced 338 outputs. Sample positive detections are shown in Figure 64. Sample false alarm outputs are shown in Figure 65.

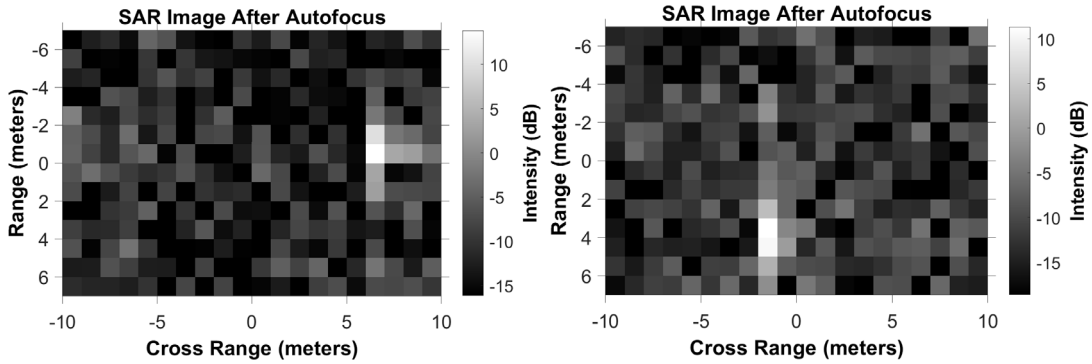


Figure 64. PGA Detections Using Reduced Window Size, SNR of 0.0 dB.

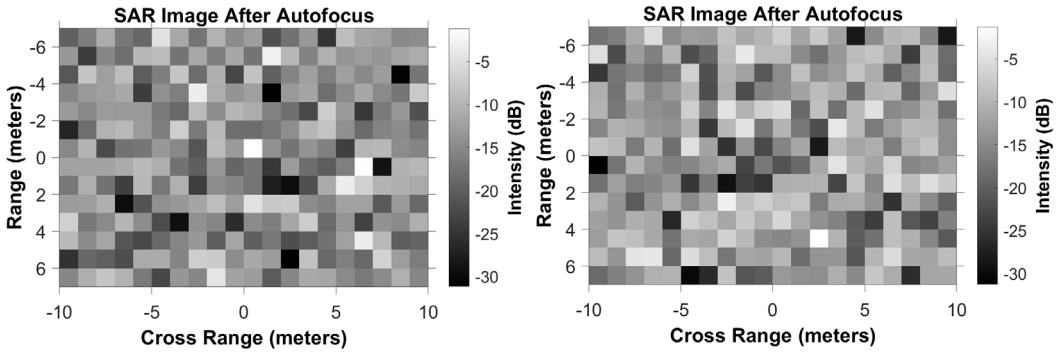


Figure 65. PGA False Alarms Using Reduced Window Size, SNR of 0.0 dB.

A total of 262 false alarms were identified, resulting in a false alarm rate of 7.94% Of the 71 targets present, 68 were detected resulting in a detection rate of 95.77%, which was higher than the AROMA rate by 2.87%. There were eight output images identified as redundant target scatters.

This was the singular case that the PGA detection rate outperformed the AROMA algorithm, although the PGA false alarm rate was still higher than the AROMA rate by 7.395%. The PGA performance was evaluated again using the same window size and noise environment but also raising the detection threshold to 1.6 to reduce the number of false alarms observed. When the PGA algorithm was evaluated with a higher threshold, 99 total detections were made. 26 of the 99 detections were classified as false alarms, resulting in a false alarm rate of 0.787%. Of the 71 targets present, 68 were detected resulting in a detection rate of 95.77%, matching the PGA detection rate when a lower detection threshold was employed. There were eight output images identified as redundant target scatters. The performance of AROMA and PGA algorithms with the reduced window size operating in a simulated medium noise environment is summarized in Table 6.

4. SNR of -6.0 dB

The noise magnitude was set to a value of 20.0 to resemble a turbulent sea state with a high level of background clutter yielding a SNR of -6.0 dB. The AROMA MTD algorithm was evaluated first in the heavy noise environment using the nominal window

size of 20 × 30 pixels. A total of 40 windows in both the range and the cross-range dimension were employed to give coverage of 78 target scatters as seen in Figure 66.

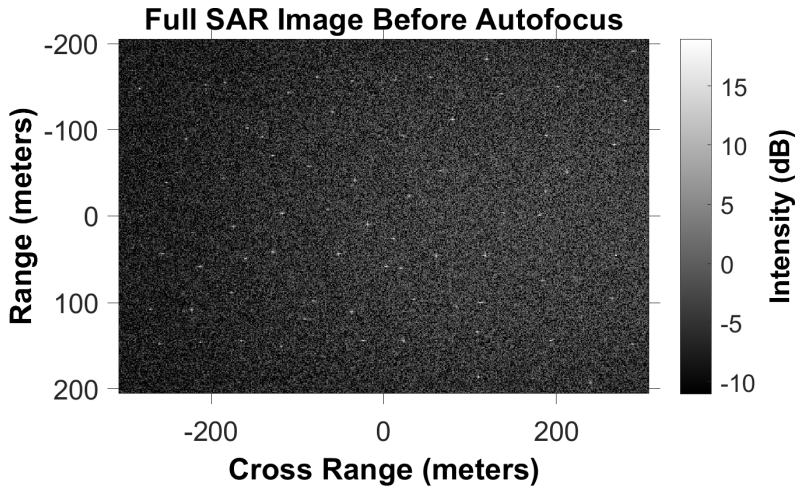


Figure 66. Input Image from Simulated Dataset, SNR of -6.0 dB.

The AROMA algorithm recovered 81 total detections, two of which were classified as false alarms. Sample positive detections are shown in Figure 67. Sample false alarm outputs are shown in Figure 68.

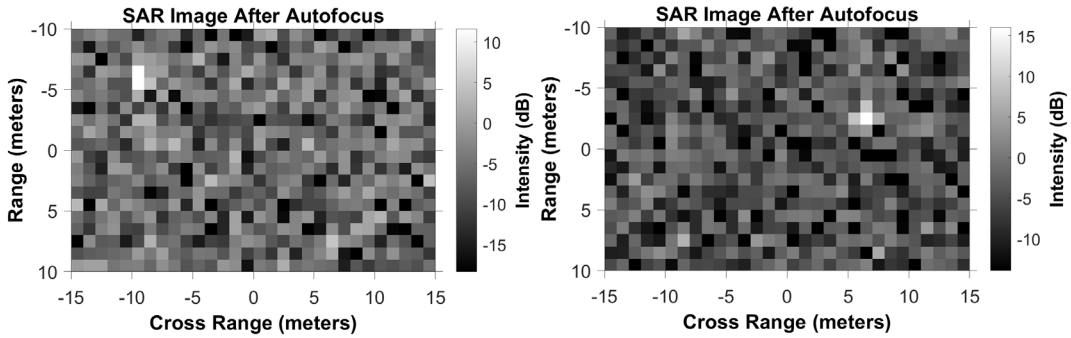


Figure 67. AROMA Detections with SNR of -6.0 dB.

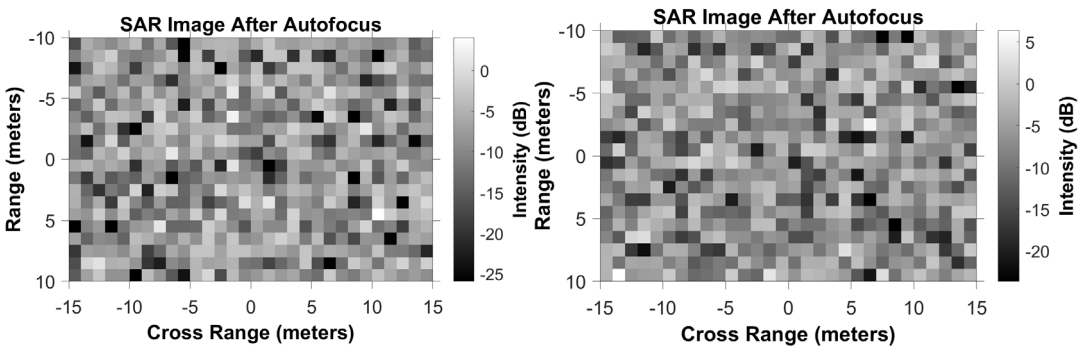


Figure 68. AROMA False Alarms with SNR of -6.0 dB.

The AROMA MTD algorithm recovered 88 total detections. The false alarm rate was measured to be 0.125%. Of the 78 target scatters present in the input image, 74 were detected by AROMA resulting in a detection rate of 94.87%. There were five outputs classified as redundant target images.

The PGA algorithm evaluated in the heavy noise environment using the same window size and coverage resulted in 176 total detections. Sample positive detections are shown in Figure 69. Sample false alarm outputs are shown in Figure 70.

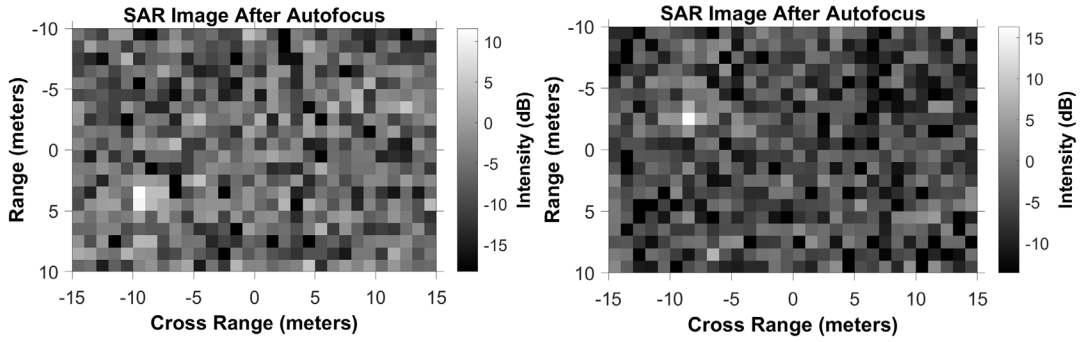


Figure 69. PGA Detections with SNR of -6.0 dB.

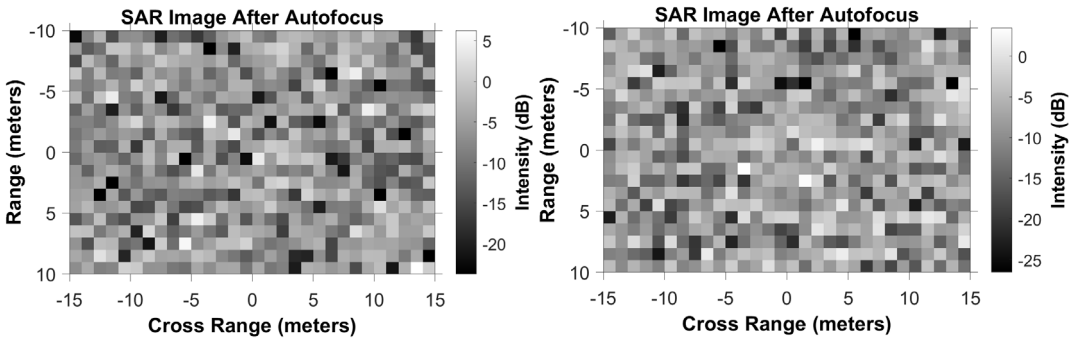


Figure 70. PGA False Alarms with SNR of -6.0 dB.

There were 92 outputs classified as false alarms, resulting in a false alarm rate of 5.75%. 74 of the available 78 targets were recovered by PGA, matching the detection rate of the AROMA algorithm using the same condition at 94.87%. As the environmental noise was increased from 10.0 to 20.0, the detection rate of the AROMA algorithm decreased as expected (97.44% to 94.87%). The PGA algorithm exhibited a higher detection rate in the heavy noise environment, contrary to expectations of intensity-based metrics. The PGA false alarm rate increased in the -6.0 SNR environment compared to the rate from the 0.0 dB SNR, from 5.375% to 5.75%. This supports the conclusion that elevated background scatter caused more autofocused windows to pass the detection threshold, whether the passing windows are false alarms or positive detections. AROMA and PGA displayed comparable detection rates in the heavy noise environment, whereas the PGA algorithm yielded a much higher false alarm rate (5.75% compared to 0.125%). The performance of both algorithms with the nominal window size operating in a simulated heavy noise environment is summarized in Table 5.

Both the AROMA and PGA MTD algorithms were evaluated in the simulated heavy noise environment using a reduced window size of 14 × 20 pixels. The algorithm employed 60 windows in the range dimension and 55 windows in the cross-range dimension, yielding coverage of 71 simulated target scatters as seen in Figure 71.

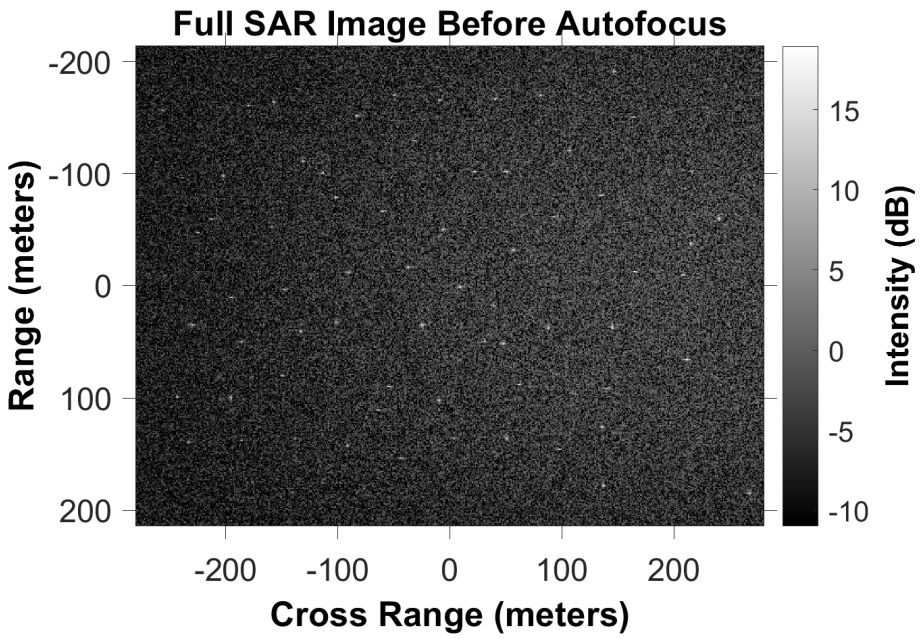


Figure 71. Input Image from Simulated Dataset Using Reduced Window Size, SNR of -6.0 dB.

The AROMA MTD algorithm produced 97 total detections when ran with the reduced window size. Sample positive detections are shown in Figure 72. Sample false alarm outputs are shown in Figure 73.

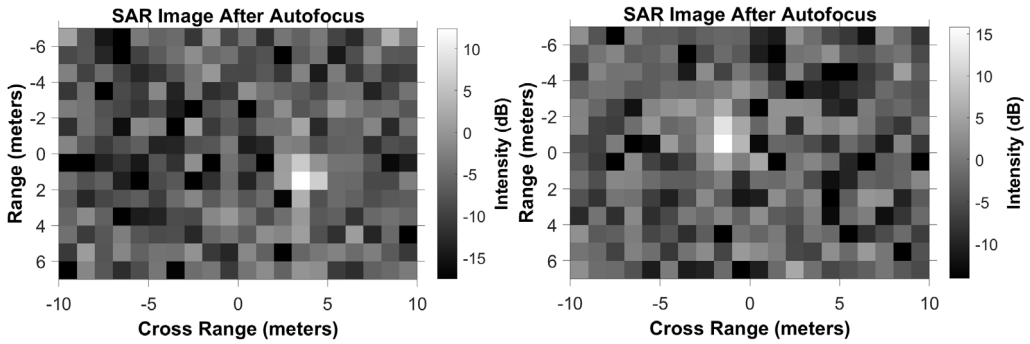


Figure 72. AROMA Detections Using Reduced Window Size, SNR of -6.0 dB.

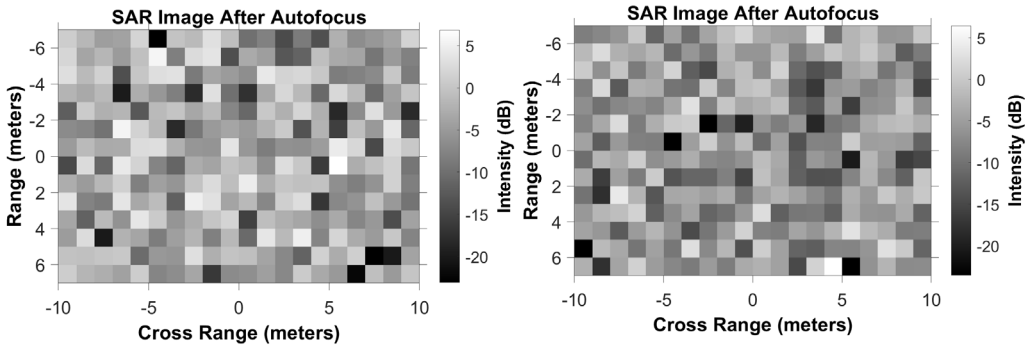


Figure 73. AROMA False Alarms Using Reduced Window Size, SNR of -6.0 dB.

A total of 25 output windows were classified as false alarms, resulting in a false alarm rate of 0.757%. Out of the 71 target scatters present in the input image, the AROMA algorithm detected 68, resulting in a detection rate of 95.77%. A total of four output windows were classified as redundant target images.

The PGA algorithm was evaluated using the same reduced window size, detection threshold, and window coverage. The PGA algorithm produced a total of 68 detections. Sample positive detections are shown in Figure 74. Sample false alarm outputs are shown in Figure 75.

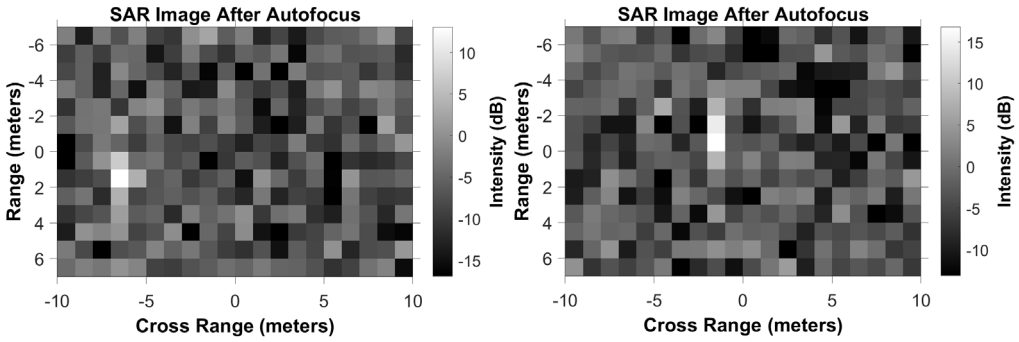


Figure 74. PGA Detections Using Reduced Window Size, SNR of -6.0 dB.

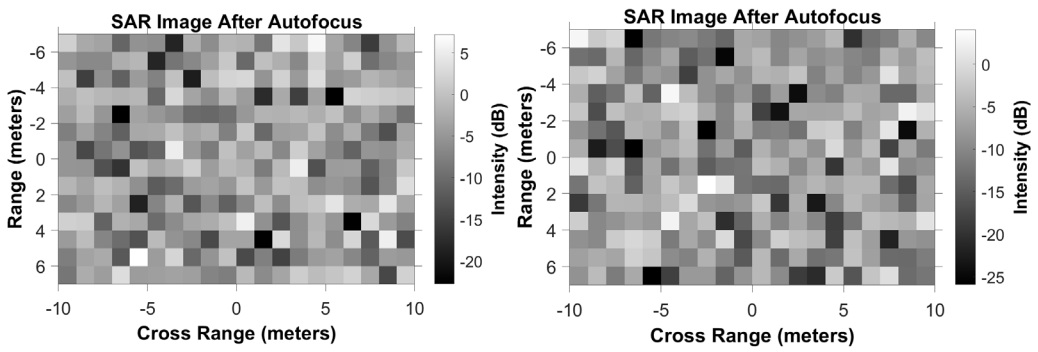


Figure 75. PGA False Alarms Using Reduced Window Size, SNR of -6.0 dB.

There were 312 output images classified as false alarms, resulting in a false alarm rate of 9.45%. The PGA algorithm correctly detected 68 of the 71 available targets, resulting in a detection rate of 95.77%. There were three instances of redundant output target images. The detection rate for PGA and AROMA using the detection threshold and reduced window size was equivalent whereas the false alarm rate for PGA far exceeded the rate of AROMA. The PGA false alarm rate was the highest value out of all evaluations performed on both measured and simulated datasets. This was expected as the high noise present in the -6.0 dB environment induced a higher level of background clutter, causing more autofocused images to pass the detection threshold value of 1.1.

To reduce the false alarm rate of the PGA algorithm to acceptable levels, the detection threshold was raised to 1.6. The detection performance was re-evaluated using the new threshold while keeping the same reduced window size and coverage. A total of 108 detections were observed from the algorithm with the elevated detection threshold. There were 35 output images visually classified as false alarms, resulting in a false alarm rate of 1.06%. This false alarm rate was much lower than the PGA algorithm ran with the original threshold (1.1), although it was still higher than the AROMA algorithm evaluated under the lower threshold. The detection rate for PGA evaluated using the elevated threshold fell to 90.14%, with only 64 of the 71 available target scatters correctly detected. As expected, raising the detection threshold limited the number of false alarms present, but also prevented several target scatters from passing the threshold resulting in a lower detection rate. The performance of the PGA and AROMA algorithms with reduced window size operating in a simulated heavy noise environment is summarized in Table 6.

Table 5. Simulated Data Performance for AROMA and PGA MTD Algorithms Using Nominal Window Size (20 × 30).

Simulated Data		
	AROMA	PGA
SNR = Infinite		
Detection Count	76/78	69/78
Detection Rate	97.44%	88.46%
False Alarm Count	28/1600	30/1600
False Alarm Rate	1.75%	1.88%
SNR = 6.0 dB		
Detection Count	77/78	73/78
Detection Rate	98.72%	93.59%
False Alarm Count	2/1600	71/1600
False Alarm Rate	0.13%	4.43%
SNR = 0.0 dB		
Detection Count	76/78	72/78
Detection Rate	97.44%	92.31%
False Alarm Count	1/1600	86/1600
False Alarm Rate	0.06%	5.38%
SNR = -6.0 dB		
Detection Count	74/78	74/78
Detection Rate	94.87%	94.87%
False Alarm Count	2/1600	92/1600
False Alarm Rate	0.13%	5.75%

Table 6. Simulated Data Results Using Reduced Window Size (14 × 20).

Simulated Data			
	AROMA	PGA	PGA Elevated Threshold
SNR = 6.0 dB			
False Alarm Count	23/3300	267/3300	33/3300
False Alarm Rate	0.70%	8.09%	1.00%
Detection Count	69/71	69/71	68/71
Detection Rate	97.18%	97.18%	95.77%
SNR = 0.0 dB			
False Alarm Count	18/3300	262/3300	26/3300
False Alarm Rate	0.55%	7.94%	0.79%
Detection Count	66/71	68/71	68/71
Detection Rate	92.96%	95.77%	95.77%
SNR = -6.0 dB			
False Alarm Count	25/3300	312/3300	35/3300
False Alarm Rate	0.76%	9.45%	1.06%
Detection Count	68/71	68/71	64/71
Detection Rate	95.77%	95.77%	90.14%

PGA exhibited improved performance as noise was injected into the scene. More windows passed the detection threshold, improving the chance that consolidation would render target scatters as the final output. This effect was noticeable in areas where target scatters were close together but in gaps where no target scatters were present, the high number of windows passing the threshold led to consolidation of numerous false alarms.

The singular trial resulted in PGA exhibiting a higher detection rate than the AROMA algorithm, when the SNR was set to 0.0 dB and a reduced window size was used.

VI. CONCLUSION AND FUTURE WORK

A. CONCLUSION

This study sought to evaluate if improved MTD performance can be acquired using the AROMA algorithm over traditional PGA. The 3D physical signal model of AROMA accounting for nonlinear translational and rotational target movement may be better equipped to model pitch and roll of surface vessels, as sharpness ratios from patches interrogated with AROMA were consistently greater than the same patches evaluated using PGA. The increase in sharpness of AROMA autofocused images was evident by visual inspection of output images (see Figure 27). With AROMA providing higher sharpness ratio values, a greater threshold could be designated to screen images for the presence of a moving target. The larger threshold value resulted in fewer nontarget windows exceeding the threshold, yielding a lower probability of false alarms for the AROMA algorithm.

When tested on measured datasets, the AROMA based sliding window algorithm consistently produced lower false alarm rates than the PGA algorithm. The higher threshold value used in AROMA did not sacrifice detection performance as AROMA matched or exceeded the detection rate of PGA in all measured data images. In indexes 2 and 3, larger targets were present that required the use of larger interrogation windows and a reduced threshold value. In these two cases, AROMA and PGA were tested using the same threshold with AROMA providing the higher overall detection rate and lower number of false alarms. Upon evaluating MTD performance on measured data using various detection thresholds and interrogating window sizes, AROMA produced a higher overall detection rate at 97.33% and a lower false alarm rate at 0.0006%.

Simulated datasets were generated to test AROMA and PGA based MTD algorithms on environments with various levels of injected background noise. A common critique of intensity-based metrics is that performance levels drop in the presence of elevated background noise as was confirmed with the trail of the traditional PGA method. The PGA algorithm produced an acceptable false alarm rate of 1.88% with no background noise present but, as noise was injected into the scene, the false alarm rate for PGA rose

with each increasing level of noise magnitude with the highest observed false alarm rate at an unacceptable 5.75%. The detection rate of the PGA algorithm in increasing levels of noise did not exhibit decreased performance, with the highest detection rate at 94.87% produced from the scene with the highest level of injected noise. The simulated PGA results showed that increasing levels of noise caused more interrogated windows to exceed the detection threshold value, resulting in more false alarms as expected.

The AROMA MTD algorithm did not exhibit as severely degraded performance in elevated noise environments as the PGA algorithm. The AROMA false alarm rate dropped with increasing levels of background noise until slightly rising to 0.13% when the observed SNR was its minimum level at -6.0 dB. These false alarm rates are far below the rates of the PGA algorithm and are well within acceptable standards for an MTD algorithm. The detection rate of the AROMA algorithm in increasing noise levels showed a steady decline but stayed within acceptable levels. The lowest AROMA detection rate matched the highest PGA detection rate of 94.87% when the SNR was -6.0 dB but, in all other scenarios, the AROMA detection rate remained 5–9% higher than the PGA algorithm. The use of reduced window sizes and adjusted thresholds resulted in more comparable performance between the two algorithms, yet AROMA still produced a higher overall detection rate and lower false alarm rate in these scenarios. This study proposed that the AROMA based MTD algorithm exhibits minimal performance degradation compared to traditional intensity-based methods when exposed to high levels of background noise.

The sliding window algorithm in this study employed a consolidation technique aimed to eliminate redundant target outputs and generate a single optimal coverage output for each target present within an image. This consolidation algorithm was effective when interrogating scenes with adequate spacing between target vessels. While the magnitude sum metric was effective at identifying windows that provided optimal coverage, large interrogating windows and overlap resulted in instances when multiple windows providing optimal coverage still passed the detection threshold. If these windows were diagonally adjacent, the consolidation algorithm would fail to eliminate the window with the lower magnitude sum value and redundant target outputs would result. The consolidation algorithm was designed to interrogate horizontally and vertically adjacent windows for the

presence of redundant targets. Through analyzing the provided measured datasets, most cases of diagonally adjacent windows passing the detection threshold occurred when two separate targets were positioned close together. This study elected to not interrogate diagonally adjacent windows using the consolidation algorithm as it was deemed more egregious to occlude a target altogether in the MTD output than the rare case of producing a redundant output.

This study concludes that the AROMA algorithm can be effectively employed in a sliding window MTD approach to autonomously provide optimal coverage of maritime targets. Upon testing measured datasets, AROMA produced higher detection rates and lower false alarms than traditional PGA methods. AROMA displayed minimal performance degradation when evaluated in simulated scenes with high levels of background noise, showing that it can be a viable option to conduct MTD in various maritime environments.

B. FUTURE WORK

The measured datasets provided by COSMO-SkyMed provided valuable imagery data over littoral areas in one geographic region. Future work aims to evaluate measured datasets from various geographic regions and sea states. High levels of background noise induced from heavy sea states are desired to test AROMA based MTD performance on measured data and not only in simulated high noise environments.

As AROMA provides autonomous detection of maritime targets of interest, it is desirable to also provide classification of these vessels. Convolutional neural networks (CNNs) and other methods of automatic target recognition (ATR) are subjects of future efforts to provide autonomous classification of vessels in a radar scene that are detected by current MTD algorithms [8]. This would alleviate the burden of imagery analysts required to examine and extract the most relevant and timely information from the vast amounts of available radar imagery and provide expedited maritime intelligence.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] G. Di Martino and A. Iodice, *Maritime Surveillance with Synthetic Aperture Radar*, Edison, NJ, USA: SciTech Publishing, 2020.

[2] C. V. Jakowatz Jr., D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and P. A. Thompson, *Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach*. New York, NY, USA: Springer Science Business Media, 1996.

[3] J. R Fienup, "Detecting moving targets in SAR imagery by focusing," *IEEE Transactions on Aerospace and Electronic Systems*, July 2001.

[4] D. A. Garren, "Theory of arbitrary rigid object motion autofocus for non-uniform target rotation and translation," *IET Radar, Sonar and Navigation*, vol. 14, no. 11, pp. 1803–1814, November 2020. https://ietresearch.onlinelibrary.wiley.com/doi/epdf/10.1049/iet-rsn.2020.0201.

[5] H. Lang, Y. Xi, and X. Zhang, "Ship detection in high-resolution SAR images by clustering spatially enhanced pixel descriptor," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, August 2019.

[6] C. Ozdemir, *Inverse Synthetic Aperture Radar Imaging with MATLAB Algorithms*. Hoboken, NJ, USA: John Wiley & Sons, 2012. Accessed: Jul. 12, 2021 [Online]. Available: http://ebookcentral.proquest.com/ lib/ebook-nps/detail.action?docID=818515

[7] e-GEOS Non-exclusive End-user License for Cosmo-SkyMed Products, 2009. "COSMO-SkyMed™ Product -ASI [2010] processed under license from ASI – Agenzia Spaziale Italiana. All rights reserved."

[8] U. K. Majumder, E. P. Blasch, and D. A. Garren, *Deep Learning for Radar and Communications Automatic Target Recognition*, NY, USA: Artech House, 2020.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California