SANDIA REPORT

SAND2000-1421
Unlimited Release
Printed June 2000

Open-Loop Adaptive Filtering for Speckle Reduction in Synthetic Aperture Radar Images

Judd A. Rohwer

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865)576-8401
Facsimile: (865)576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.doe.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800)553-6847
Facsimile: (703)605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/ordering.htm

SAND2000-1421
Unlimited Release
Printed June 2000

Open-Loop Adaptive Filtering for Speckle Reduction in Synthetic Aperture Radar Images

Judd A. Rohwer
Telemetry and Instrumentation Department
Sandia National Laboratories
P.O Box 5800
Albuquerque, New Mexico 87185-0986

Abstract

The Two-Dimensional Adaptive Correlation Enhancer Algorithm (2DACE) is an open-loop adaptive filtering technique that can be applied to Synthetic Aperture Radar (SAR) images for the purpose of reducing speckle. This report includes the development of the 2DACE algorithm and the optimum filter parameters for this specific task. The unique implementation of 2DACE with a data amplitude pre-compression operation was proven to effectively reduce speckle, enhance fine features, and maintain image resolution.

ACKNOWLEDGMENTS

I would like to thank Neeraj Magotra, who served as my thesis advisor at University of New Mexico, for the guidance he provided and the suggestions and tips he offered. I would also like to thank Armin Doerry of the Synthetic Aperture Radar Department 2345 for the SAR images and many helpful discussions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

This work presents research into the application of the Two-Dimensional Adaptive Correlation Enhancer Algorithm (2DACE), an open-loop adaptive filtering technique, to reduce speckle in Synthetic Aperture Radar images. Open-loop filters are application specific and the variable parameters must be optimized for the task. This project presents a unique implementation involving data amplitude compression to manage the gain control problems inherent with the open-loop architecture.

The objective was to determine if 2DACE could effectively reduce speckle without degrading the target resolution. The effects of varying filter parameters and development of the optimal parameter combination are presented. The filter was tested using two-dimensional Gaussian and Taylor Point Spread Functions as test images. White Gaussian noise and actual speckle from a Synthetic Aperture Radar image were added to the test images to determine the efficacy of 2DACE.

The performance of the 2DACE algorithm is quantified through signal-to-noise ratio calculations and resolution measurements. The SNR is calculated with the variance of the clean ideal image divided by the least squares error calculation of the filtered image and the clear image. As the error between the filtered image and the clear image decreases the SNR increases. Measurements of the -3dB and -14dB pulse widths of the two-dimensional Taylor Point Spread Function are documented to quantify the effects of the 2DACE filter and the parameters on the resolution.

Because of the wide dynamic range of the pixels in the SAR images, nonlinear data compression techniques were applied before and after the image processing. The compression techniques were required to control the open-loop gain of the filter, improve the resolution, and restore the contrast of the image.

As a final test the 2DACE filtering algorithm was applied to SAR images obtained from the Synthetic Aperture Radar Department 2345 at Sandia National Laboratories. A subjective comparison between the original speckled image and the filtered image proves the expected performance of the algorithm.

NOMENCLATURE

ACE	-	Adaptive Correlation Enhancer
APC	-	antenna phase center
AWGN	-	additive white Gaussian noise
FFT	-	Fast Fourier Transform
FIR	-	finite impulse response
FM	-	frequency modulation
GPS	-	Global Positioning System
IF	-	intermediate frequency
IMU	-	inertial measurement unit
LFM	-	linear frequency modulation
LPF	-	low pass filter
PCR	-	pulse compression ratio
PDF	-	probability density function
PRF	-	pulse repetition frequency
PSF	-	point spread function
SAR	-	Synthetic Aperture Radar
SNR	-	signal-to-noise ratio
2DACE	-	Two-Dimensional Adaptive Correlation Enhancer
2DLPF	-	two-dimensional low pass filter

CHAPTER 1

INTRODUCTION

Synthetic Aperture Radar (SAR) is a high-resolution radar system using coherent processing of backscattering Doppler histories to obtain small angular resolution [1]. SAR is capable of generating images in day or night, with or without cloud cover, in any type of weather condition. Typical applications of SAR are found in surveillance from air and space, navigation, missile guidance, and mapping [2,3,4].

In 1951 Carl Wiley of the Goodyear Aircraft Corporation [1] introduced the idea that a moving radar system could generate pulses, delayed in time. A pulse to pulse comparison of the returned signals could be used to produce radar images with greater resolution than a standard radar system [2]. In 1953 the first SAR system was tested at the University of Illinois [1].

SAR images are commonly used for detection, classification, and identification. The utility of the images is directly dependent upon the quality of the image. Image quality is a general term used to describe the appearance of an image. A wide range of factors combine to improve or degrade the image quality, such as contrast, clarity, sharpness of the edges, signal-to-noise ratio, distortion, warping of targets at the image boundaries, and the amount of speckle. The key attributes of SAR images are detectability of scatterers, spatial resolution, positional accuracy, and image intensity [2]. In order to meet the required specifications on the attributes all of the factors must be controlled.

Image quality is achieved through system design, real time signal processing, and post-processing techniques.

The radar system, radar waveforms, and signal processing affect image quality. The radiation pattern of the antenna affects the transmitting and receiving characteristics of the SAR system. The impulse response of the radar is characterized by the mainlobe and sidelobe specifications. Sidelobe minimization in the radar signal, storage process, and filter design is required for high quality images [4]. The sidelobes can be reduced by properly selecting window functions used in the signal processing system [5]. Overall the image quality is related to both the system design and signal processing techniques employed.

In general an object in a SAR image is very rough in relation to the wavelength of the radar signal. When coherent radiation is reflected from a rough surface the reflected signal has different phases due to the surface properties. The combination of the coherent nature of the signal and dephased returns produces the granular pattern of intensity. This granular pattern, known as speckle, is undesirable in SAR images because it reduces the resolution and detracts from the fine details.

Speckle is a granular looking byproduct of coherent imaging systems that adversely affects the image quality and target resolution. Objects viewed in highly coherent light acquire a granular appearance and the granularity has no apparent relationship to the properties of the object. The irregular pattern is best described by the methods of probability and statistics.

The area of speckle reduction in coherent information processing has been a constant focus of research for decades. In fact, some of the most important work in speckle was performed in the 1970s and 1980s [6]. Speckle reduction is not only relevant to SAR but is also important in the fields of acoustic imaging, laser imaging, optical information processing, holography, holographic interferometry, and laser microscopy [7]. These fields apply to many areas such as surveillance, mapping, medical technologies, and others.

The literature on speckle reduction in SAR images goes back to the 1970s and 1980s. As computing power increases new techniques are constantly developed to better reduce the speckle while enhancing the images. Real time signal processing and post-processing algorithms exist for improving the image quality by reducing the speckle content in the image. A few techniques used to remove speckle are: multilook SAR systems [3], low pass filters, adaptive one dimensional filtering techniques [8], adaptive two-dimensional filters [9,10], wavelet transforms and thresholding [11], and neural networks [12].

Various drawbacks are associated with speckle reduction in SAR images. SAR system design is complicated and expensive. The advantage of multilook systems is that a number of independent images of the same scene can combined to reduce the speckle. Multilook SAR systems are capable of generating very clear images [3], but they are difficult to design and build. Also it is not always possible or practical to obtain independent images of the same scene. The post-processing algorithms usually reduce the resolution, smooth out fine details, and/or are computationally complex. The two-dimensional low pass

filter (2DLPF) increases the pulse widths of the point targets, effectively reducing the resolution. In addition the 2DLPF smoothes out the fine features. Other post-processing algorithms, such as the two-dimensional adaptive least mean square algorithm [10], are computationally complex and have a slow rate of convergence. Research based on the wavelet transforms with applications to speckle reduction is widely documented. Wavelet decomposition codes are readily available on the internet and through software providers. The results of initial tests using wavelet decomposition and thresholding were unacceptable.

This work presents research into the application of the Two-Dimensional Adaptive Correlation Enhancer Algorithm (2DACE) [13], an open-loop adaptive filtering technique, to reduce speckle in SAR images. Open-loop filters are application specific and the key filter parameters must be optimized for the task. This report presents a unique post-processing approach that combines nonlinear data amplitude compression and the 2DACE algorithm. The nonlinear compression is applied prior to the 2DACE to manage the gain control problems inherent with the open-loop architecture. This process maintains the initial image resolution. The various parameters in the 2DACE filter development are documented and the optimum settings are developed through testing with synthetic images and SAR images.

Implementing the 2DACE filter algorithm to best filter the SAR images required finding the optimum filter parameters and developing pre and post-processing routines. Tuning the filter design to achieve the best balance between reducing speckle and maintaining image resolution involved filtering

synthetic data and actual SAR images with many variations of the filter parameters. The key parameters affecting the speckle reduction and resolution are the scaling option, the adaptation constant, β, and the filter lag, L. The filter size is $(2L+1)\times(2L+1)$ where the filter lag L is the maximum lag of the estimated autocorrelation function.

The 2DACE algorithm with the open-loop architecture is designed to rapidly converge to the two-dimensional autocorrelation function of the input image, an $m \times n$ matrix. Under ideal circumstances the two-dimensional filter will converge to a matched filter. Since the images are not stationary the filter coefficients are constantly updated with the time constant determined by the adaptation constant β.

The post-processing speckle reduction routine developed has a series of six operations. Refer to Figure 1. First data amplitude compression is applied to the image to reduce the dynamic range. The compression process controls the gain and maintains the image resolution. The second step is data padding of the input image with reflected pixel values. This allows the filter coefficients to converge to the statistics of the image before the convolution is performed with pixels on the border of the image. The third step is to filter the compressed and padded image with the 2DACE algorithm. The fourth step is to crop the image to remove the data padding and the fifth step is to restore the dynamic range of the image. Restoring the dynamic range is an expansion process that is effectively the inverse process of the initial data amplitude compression. Step six is nonlinear data compression routine. This final data compression routine is

optional and is applied after the 2DACE filtering to improve the contrast of the image. The data compression reduces the magnitude of the high intensity pixels and increases the magnitude of the low intensity pixels.

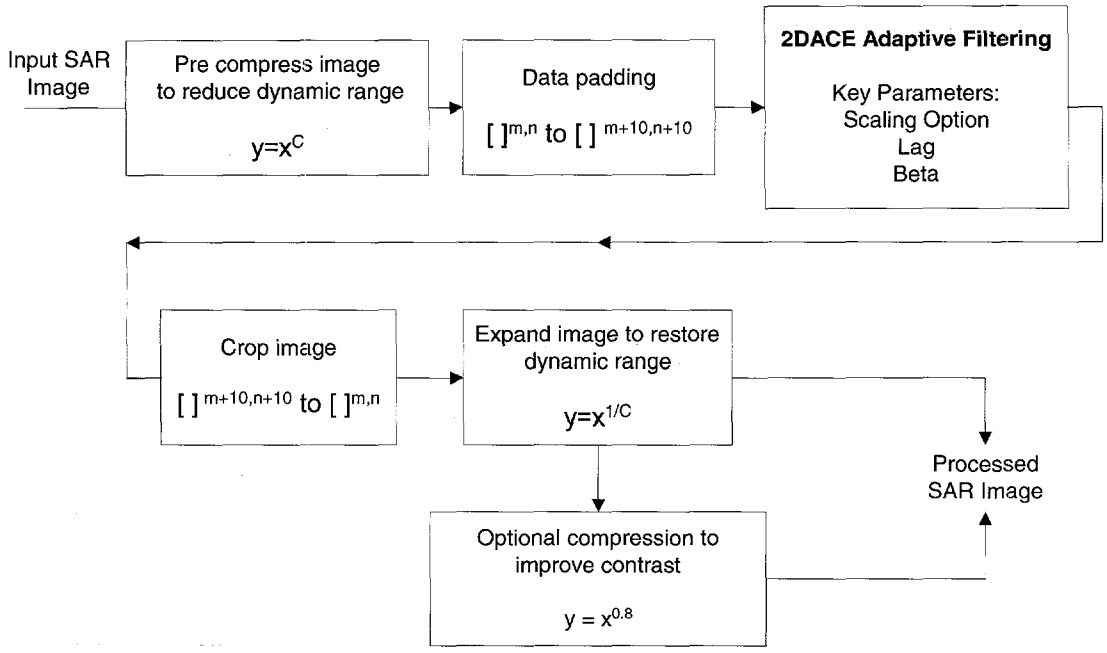


Figure 1. Block diagram showing the series of operations involved in speckle reduction with the 2DACE algorithm.

A two-dimensional low pass filter was used as the benchmark to test the performance of the 2DACE algorithm. The 3x3 LPF was easy to implement, provided a satisfactory degree of speckle reduction, and has been routinely used at Sandia National Laboratories as a quick means to improve the image quality. The drawback to the 2DLPF was the degradation in resolution and the blurring of the SAR image. Results presented in this report prove that the 2DACE has superior performance over the 2DLPF.

Chapter 2 presents a short background on Synthetic Aperture Radar. Chapter 3 includes the characteristics and statistics of speckle. Chapter 4 and 5 contain the analysis of the Adaptive Correlation Enhancer algorithm, the filter implementation, and detailed test results. SAR images processed with the 2DACE filter are included in Appendix A and the filtering algorithms can be found in Appendix C.

CHAPTER 2

BACKGROUND OF SYNTHETIC APERTURE RADAR

Synthetic Aperture Radar uses coherent processing of backscattering Doppler histories to obtain images with fine angular resolution [1]. Many subsystems within a SAR are used to form the final radar image. Platform motion sensing, microwave pulse generation, and image processing are used together and influence the range resolution and azimuth (or cross range) resolution. SAR is used for surveillance from air and space, navigation, missile guidance, mapping, and can generate images in any type of weather condition [2,3,4].

In 1951 Carl Wiley of the Goodyear Aircraft Corporation [1] introduced the idea that a pulse to pulse comparison of the returned signals from a moving radar system could be used to produce high quality radar images. The term Synthetic Aperture Radar was used to describe the process of image formation from a moving platform. The synthetic aperture produces images with a resolution that would otherwise require a much larger antenna aperture. Even though the synthetic aperture only improves the azimuth resolution comparable range resolution can be achieved by increasing the radar signal bandwidth. The moving platforms required to generate the synthetic aperture are commonly mounted on aircraft.

SAR was first tested in 1953 [1] at the University of Illinois. The first SAR was a stripmap mode system where the antenna is pointing in a fixed position relative to the flight line. The synthetic aperture image is formed by a

two-dimensional mapping of received signal energy. The intensity of each individual pixel is the energy in the signal received from the corresponding location illuminated by the radar. The brightness of the pixel is related to the degree of an object's radar reflectivity.

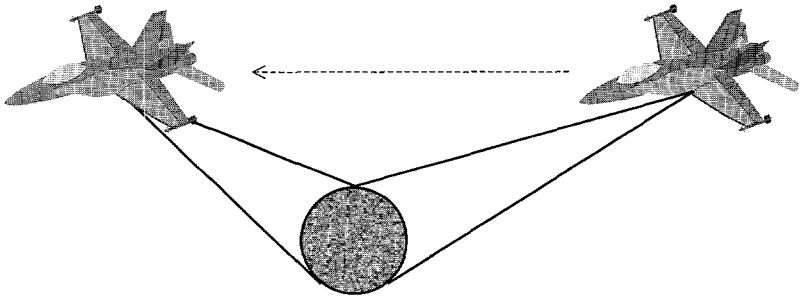
2.1 Spotlight Mode Versus Stripmap Mode

At Sandia National Laboratories the SAR spotlight mode technique is most commonly used. Spotlight mode focuses the antenna on a specific target area and the radar continuously illuminates the target area for a period of time. The flight path determines the length of the aperture.

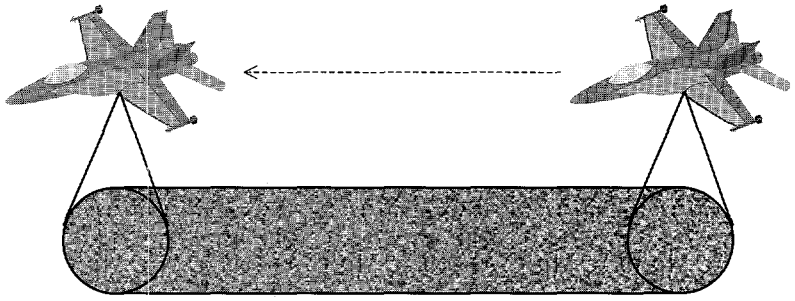
Spotlight mode has flexibility and two main advantages over stripmap mode. Spotlight mode achieves a finer resolution while using the same physical antenna aperture. In addition with spotlight mode multiple images of the same target or even multiple images of different targets can be processed from different viewing angles [2].

Stripmap mode is used to image a long strip of terrain and the resolution is limited to the azimuth beamwidth of the radar antenna. The azimuth resolution increases with the length of the synthetic aperture. The azimuth beamwidth determines how long a target is illuminated by the radar. Refer to Figure 2 for a block diagram comparison of Stripmap mode and Spotlight mode.

The capability of spotlight mode to process an image from multiple viewing angles is advantageous because scattering properties can vary with the angle of observation. Multiple viewing angles can improve target detection and identification and produce smoother, more interpretable images [2,4].



Spotlight Mode



Stripmap Mode

Figure 2. This diagram is a comparison of two SAR imaging techniques. Spotlight Mode focuses the antenna on a specific target area. In stripmap mode the antenna is focused in a fixed direction and is used to image a long strip of terrain.

2.2 Synthetic Aperture Radar Processing

The SAR system transmits pulses at time intervals as the radar platform moves along its flight path. The platform may deviate from the flight path horizontally or vertically. If these deviations are measured accurately then the received data can be appropriately phase compensated. SAR transmits and receives signals over an angular interval that is inversely proportional to the desired azimuth resolution. The time interval is called the coherent aperture time. The radar illuminated target reflects energy from each transmitted pulse and the SAR receives the reflected signals during a specific time delay interval. The interval is called the range gate.

Essentially, SAR image formation requires the following steps [2].

- Transmit and receive a wide bandwidth signal with a deterministic phase relationship from pulse to pulse.
- Measure and adjust for vertical and horizontal movements of the radar platform relative to the target.
- Format the data based on radar system parameters and data collection specifics.
- Compress the data in range and azimuth to achieve the desired resolution.

The radar pulse specifications and motion compensation are required in order to maintain the phase coherence from pulse to pulse during the coherent aperture time. SAR images are formed in a number of ways depending on the size of the image and the radar frequencies. To achieve the final image resolution the data is typically compressed with a two-dimensional Fast Fourier Transform (FFT). The coherent pulses are transmitted as a periodic pulse train with pulse repetition interval, **T**. The reciprocal of the pulse repetition interval is called the pulse repetition frequency (PRF). SAR usually has a constant PRF.

2.3 Basic Synthetic Aperture Radar Components

A SAR system has three basic components: the SAR sensor, the motion sensor, and the image processor [2]. The image processor uses data from the SAR sensor and motion sensor to generate the image.

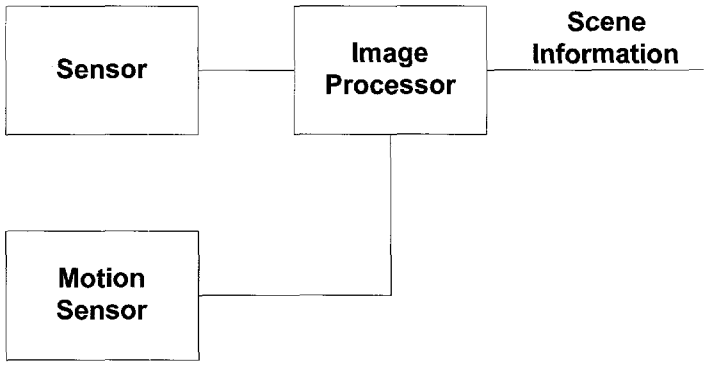


Figure 3. This is a basic SAR block diagram showing the connections between the sensor, motion sensor and image processor.

2.3.1 SAR Sensor

The SAR Sensor, shown in Figure 4, includes the electronics that generate, transmit, and receive the radar pulses. This includes the antenna, transmitter, receiver, transmission channel, data recording electronics, antenna steering, and timing electronics. Most SAR systems are monostatic, the same antenna is used for both transmitting and receiving the radar pulses [2]. The transmitter generates lowpass and bandpass coherent pulses and the receiver detects and filters the reflected pulses. The received signal is then converted to digital data with an A/D converter and stored or processed in real time. The timing electronics generate a stable clock signal and a coherent phase reference for the transmitted signals. The transmitter generates a wide bandwidth, high frequency signal from a digital waveform stored in the system memory. This generated signal is then filtered with a bandpass filter, mixed with the carrier signal, amplified, and then transmitted via the antenna.

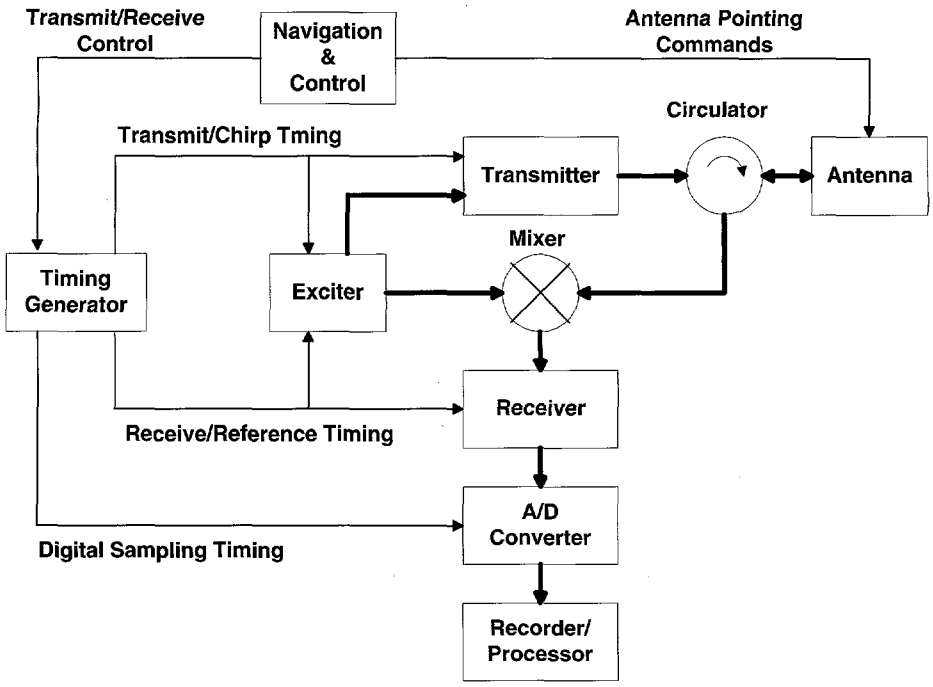


Figure 4. A detailed view of the SAR sensor [2] showing the connections from the antenna to: the A/D converter and data recorder, the navigation and control module, transmitter and receiver.

The backscattered signals detected by the antenna within the specified range gate are sent to the receiver for processing. Within the receiver the RF carrier frequency is removed along with any modulation generated in the transmitter. The A/D sampling rate must satisfy the Nyquist criteria in both the azimuth and range. In the azimuth the sampling rate is equivalent to the pulse repetition frequency. In the range the sampling rate must be greater than the signal bandwidth when using in phase and quadrature channels. Downsampling can be used to reduce the amount of data processed and stored in memory.

Azimuth ambiguities can be created by the edges of the main lobes or side lobes of the azimuth antenna pattern. This causes the pulse repetition frequency to not adequately sample the Doppler frequency of the returned data.

An antenna with low side lobes is the primary method to reduce or limit the energy from range and azimuth PRF regions. Range ambiguities occur when received signals are detected outside of the specified range gate of a transmitted pulse.

Popular frequencies used in SAR transmitters are VHF (30-300 MHz), X-Band (8-12 GHz), and up to KA-Band (27-40 GHz) [14]. Frequency selection is dependent upon the application and propagation factors.

2.3.2 Motion Sensor

Motion sensors are required for high-resolution SAR images generated on airborne systems; motion sensors are not required on satellite systems [2]. The motion sensor subsystem generates data used to estimate position changes of the SAR antenna. The motion sensor tracks the changes in the antenna phase center (APC). An inertial measurement unit (IMU) and a global positioning system (GPS) navigator are used to track the short duration and long duration changes in the APC. The IMU detects the short duration changes in motion, which are changes in the antenna position with duration less than the synthetic aperture time. Long duration changes are changes with periods greater than the aperture time. The position of the antenna is estimated by measuring the position vectors from the GPS antenna system to the IMU and then to the APC. Changes in the position vectors correspond to long duration changes in the APC.

The motion sensor is used to point the SAR antenna and specify the range gate – the expected transit time of the radar signal. The motion sensor also selects reference signals that are used in the receiver to remove corrupted

data from the received signal. The image processor uses motion sensor data to calculate the antenna position estimates which are then used to remove the effects of movements of the radar platform.

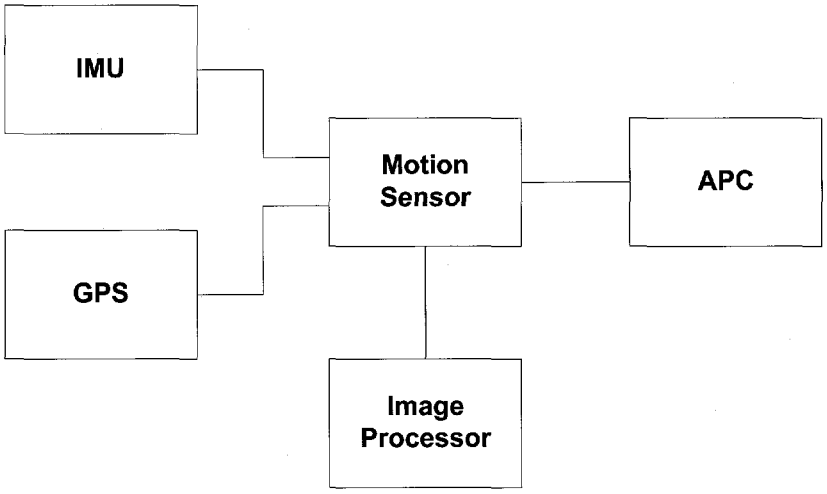


Figure 5. SAR motion sensor block diagram. The IMU and GPS operate independently and feed data to the motion sensor. The motion sensor controls the APC and sends the motion data to the image processor.

2.3.3 Image Processor

The image processing subsystem includes the hardware and software used to process the received signal and the motion sensor outputs. The image processing includes the azimuth and range filtering required to limit the scene size. Downsampling is performed in the image processor along with any required motion compensation due to non-ideal radar platform movements. The image processor also performs data formatting, compression with a two-dimensional FFT, and compensation for effects in the transmission channel and/or unanticipated radar signal defects.

The image processor can also be programmed to perform post-processing of the SAR image. The post-processing can include image enhancement, target

detection, target recognition, moving target identification, and image comparison [2]. Image enhancement includes speckle reduction – the topic of this thesis.

2.4 SAR Image Formation

Processing the Radar Return and forming the SAR image uses three sets of information: Positioning information, angular resolution, and intensity of radar return. Positioning information is used to locate the sources of the radar returns in two dimensions. A three-dimensional image can be formed with the addition of data and processing techniques [4]. Angular resolution is the ability of the system to discriminate between the independent scatterers. Each scatterer within a target scene is identified by the intensity of the radar return. The intensity is the characteristic used to form the image. Intensity varies across the target according to the reflectivity coefficients of the illuminated objects.

SAR systems can either transmit continuous waveforms or transmit pulses [2]. Most SARs transmit in the pulse mode with a fixed pulse repetition frequency (PRF) or by varying the amplitude or frequency of the pulse. Frequency modulation is used in many SAR systems to modulate a constant amplitude pulse. Discrete changes in phase or frequency are also used to vary the radar pulse.

2.4.1 Range Resolution

The distance between the radar system and the radar scatterer is the time delay between transmitting the radar pulse and receiving the return pulse. Range resolution is determined by the radar system's capability to distinguish

time differences between echoes from two adjacent targets. Therefore the range resolution directly depends on the radar bandwidth [4].

For a constant frequency waveform the range resolution ρ_r is given in (1-1) [2] where c is the speed of light and T_p is the pulse duration.

$$\rho_r = \frac{cT_p}{2} \qquad (1-1)$$

Conversely, the ability to detect the radar pulse in noise is related to the pulse width. For constant frequency pulse systems the range resolution is related to the pulse width. The shorter the pulse width the finer the resolution, but the probability of detection decreases with a decrease in pulse width [2].

As higher range resolution is desired there is an increase in difficulty in transmitting the required power level in the continuous wave pulse [4]. Power can be defined as $\int_T |x(t)|^2 dt$. As T decreases the average power also decreases. Modern SAR systems use linear frequency modulation (LFM) signals in order to increase range resolution and the power level in the pulse. The linear frequency modulation signal is easy to generate and is commonly referred to as a chirp.

The linear FM pulse has a pulse length of T_p and a frequency that varies with a chirp rate of γ. The bandwidth of the chirp is $\beta = \gamma T_p$ which has a compressed time duration of $\frac{1}{\beta}$. This time duration defines the range resolution given in equation (1-2).

$$\rho_r = \frac{cK_r}{2\beta} \qquad (1-2).$$

K_r is a variable that compensates for changes in the main lobe due to techniques used to reduce the sidelobe levels, such as antenna design. The pulse compression ratio (PCR) is defined as the pulse length T_p multiplied by the bandwidth $\beta = \gamma \cdot T_p$, $PCR = \gamma \cdot T_p^{\,2}$. The PCR represents the improvement in resolution obtained via pulse compression.

The pulse compression technique allows a transformation related to the chirp rate and distance of each individual scatterer. The received signal is mixed with a time delayed replica of the transmitted signal and converted into a frequency relative to the dechirp signal. This process is referred to as range dechirping. The time delay is equal to travel time to the center of the target scene and back to the antenna. The mixing of the received signal reduces the bandwidth of the return signals relative to the to the swath distance of the independent targets. A near range target would have a higher received frequency than a long-range target. The bandwidth of the dechirped signal is called the intermediate frequency (IF) and is calculated as shown in equation (1-3) where Δr is the swath range size.

$$\beta_{IF} = \frac{2\gamma\Delta r}{c} \qquad (1-3)$$

The swath range Δr produces a time delay between the near range target and the far range target. The time delay is the transmit - to - receive time $\frac{2\Delta r}{c}$. The

frequency of the dechirped signal from each scatterer is related to the time delay between the signal and the reference function within the SAR system.

2.4.2 Azimuth Resolution

The unique design of SAR is the generation of the synthetic aperture to reduce the synthetic beamwidth and increase the azimuth resolution. The synthetic aperture is constructed by moving radar with a defined beamwidth. As the radar travels in a straight line the beamwidths overlap in time and therefore can be used to construct the narrower synthetic beam.

The azimuth resolution is the ability to recognize two independent targets that are adjacent to one another. If the two targets are farther apart than the radar's azimuth beamwidth angle at the target scene then the two targets can be independent resolved. If two adjacent targets are both illuminated at the same time then they cannot be independently resolved.

The azimuth resolution for a synthetic aperture of length L [2] is defined as

$$\rho_a = \frac{\lambda_c k_a R_t}{2L\sin(\alpha_{dc})} \qquad (1-4)$$

R_t is the range to the target, λ_c is the center frequency of the carrier, k_a is the main lobe widening factor ~ 1.2 [4]. α_{dc} is the Doppler cone angle, the angle that the antenna is pointed relative to the platform's velocity.

The SAR processor builds a synthetic array by adjusting the phase of the complex received signals and adding the data samples. It receives data along the path length L - which is defined as the synthetic aperture length. The

coherent summation of data samples and the synthetic aperture generate a beamwidth that is significantly smaller than the actual antenna beamwidth.

The azimuth resolution ρ_a is dependent on the range to the target scene. In spotlight mode the azimuth resolution is not limited by the antenna aperture, D, of the radar system. Conversely, in stripmap mode the azimuth resolution is limited by $\frac{D}{2}$. An additional advantage of spotlight mode operation is that a smaller antenna beamwidth with high gain can be used. The high gain antenna improves the signal-to-noise ratio. The one drawback to spotlight mode is the smaller scene size compared to stripmap mode. Stripmap mode operation is independent of range because as the range increases the synthetic aperture length also increases. For spotlight mode the Doppler cone angle, α_{dc}, varies with the range and azimuth of the targets.

2.5 SAR Imaging

The SAR image provides a visualization of the complex radar reflectivity of a target scene [2]. The data can be expressed as a complex image $Ae^{j\phi}$. The physical features of the earth's surface cause changes in the phase ϕ and the amplitude A of the radar pulse. The SAR receives and separates the reflected waves' in phase $A\cos\phi$ and quadrature phase $A\sin\phi$ components. Thus the complex image $Ae^{j\phi}$ is formed with complex data $A\cos\phi + jA\sin\phi$. SAR images can be displayed in a variety of ways [2]: the real part $A\cos\phi$, imaginary part (quadrature) $A\sin\phi$, the amplitude A, the intensity $I = A^2$, or the log of the

intensity $\log I$. The most common method for displaying the SAR Images is the intensity output.

Coherent imaging systems, such as Synthetic Aperture Radars, exhibit noiselike characteristics classified as "Speckle" [3]. Speckle is a real electromagnetic phenomenon with noiselike characteristics and is a result of the scattering properties in the target scene [6]. The next chapter discusses the generation and statistics of speckle.

CHAPTER 3

SPECKLE – CHARACTERISTICS AND STATISTICS

Generally speaking, objects viewed in highly coherent light, acquire a granular appearance and the granularity has no apparent relationship to the properties of the object. The irregular pattern is best described by the methods of probability and statistics. Characterization of speckle has been a research topic of many years, actually some of the most important work in speckle was performed in the 1970s and 1980s [6].

In general an object is very rough in relation to the wavelength of the radiating source. When coherent radiation is reflected from a rough surface the returned signal has different phases due to the surface properties. The combination of the coherent nature of the signal and dephased returns produces the granular pattern known as speckle. Speckle is undesirable in SAR images because it detracts from the fine details and degrades the image quality by reducing the resolution. Figure 6 is a three dimensional plot of speckle from a SAR image. A chapter by J.W. Goodman in the publication 'Topics in Applied Physics: Laser Speckle and Related Phenomena' [6] documents the statistics and other details of speckle. The following sections are mostly based on this reference.

3.1 Speckle as a Random Walk

Speckle and its statistical properties have been associated with the Random Walk Process since the 1960s [15]. Using the random walk statistical process the electromagnetic wave is of the form $A(x,y,z)\exp(j2\pi vt)$ [16] where

A is the amplitude at point (x,y,z) and v is the frequency. The amplitude of the phasor can be rewritten as $A(x,y,z) = |A(x,y,z)| \exp(j2\pi vt)$ and the intensity of the wave is $I(x,y,z) = |A(x,y,z)|^2$. The intensity can also be written as $I = \mathrm{Re}[A^2] + \mathrm{Im}[A^2]$ with $\theta = \tan^{-1} \frac{\mathrm{Im}(A)}{\mathrm{Re}(A)}$. For the random walk process the complex fields have zero means, equal variances, and are uncorrelated. Using the central limit theorem for an arbitrarily large number of samples, N, the real and imaginary parts are asymptotically Gaussian.

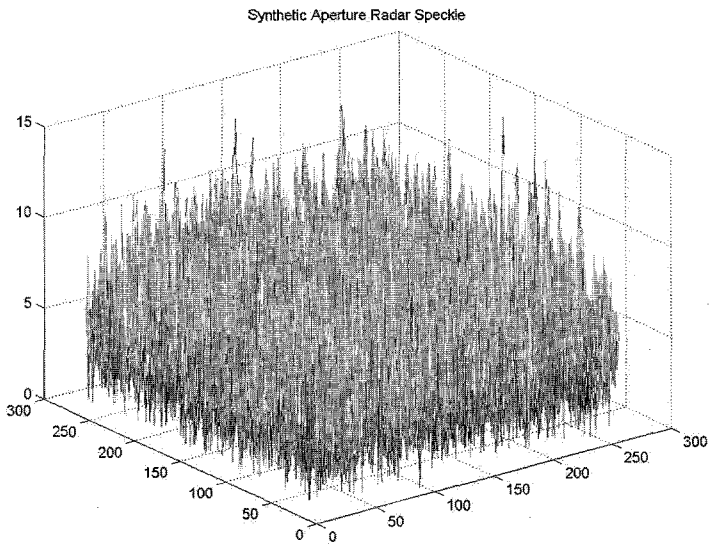


Figure 6. Mesh plot of speckle, 256x256 pixels. The speckle was cropped from a field in the SAR image F2_335493_308.

3.2 Joint and Marginal Distributions

From the equations in Section 3.1 the joint probability distribution function (PDF) of speckle can be calculated via the transformation method [6]. The Joint PDF is shown in equation (3-1) and equation (3-2) is the variance of speckle

where $|a_k|$ is the magnitude of the k^{th} elementary phasor from the phasor amplitude $A(x,y,z)$.

$$P_{I,\theta}(i,\theta) = \left\{ \begin{array}{ll} \frac{1}{4\pi\sigma^2}\exp\left(\frac{-I}{2\sigma^2}\right) & \text{for} \quad I \geq 0 \quad -\pi \leq \theta < \pi \\ 0 & \text{otherwise} \end{array} \right\} \qquad (3-1)$$

$$\sigma^2 = \underset{N \to \infty}{\text{Lim}} \frac{1}{N}\sum_{k=1}^{N}\left\langle \frac{|a_k|^2}{2} \right\rangle \qquad (3-2)$$

The marginal density functions, for intensity and phase respectively, are then

$$P_I(i) = \frac{1}{2\sigma^2}\exp\left(\frac{-I}{2\sigma^2}\right) \quad I \geq 0, \quad P_\theta(\theta) = \frac{1}{2\pi} \quad -\pi \leq \theta < \pi \qquad (3-3).$$

The intensity and phase are statistically independent. $P_I(i)$ has a negative exponential distribution and $P_\theta(\theta)$ has a uniform distribution.

3.3 Intensity of Speckle

In SAR images intensity is of primary interest. The n^{th} moment of the intensity is given by

$$\langle I^n \rangle = n!\langle 2\sigma^2 \rangle^n = n!\langle I \rangle^n \qquad (3-4).$$

The 1st and 2nd moments, the mean and variance, are of particular interest and can be calculated from (3-4). The mean value, n=1, is

$$\langle I \rangle = 2\sigma^2 \qquad (3-5)$$

The variance, n=2, is

$$\langle I^2 \rangle = 2\langle I \rangle^2, \sigma_I^2 = \langle I^2 \rangle - \langle I \rangle^2 = \langle I \rangle^2 \qquad (3-6).$$

Since $\sigma_I^2 = E\{I^2\} - \mu_I^2$ the standard deviation σ_I is equal to the mean intensity μ_I.

The first order statistics of the sum of amplitudes of speckle returns is a circularly complex Gaussian random variable. Since each speckle return is a circularly complex random variable the amplitude of the sum of the speckle returns is $A = \sum_{k=1}^{N} A_k$. $Re[A_k]$ and $Im[A_k]$ are Gaussian random variables therefore the sum $Re[A_k] + Im[A_k]$ is also a Gaussian random variable. From this argument $Re[A_k]$ and $Im[A_k]$ are also zero mean. The total intensity of a sum of speckle returns is $I = |A|^2$ which is defined as a negative exponential distribution, just like the independent speckle returns. The same process can be followed with a sum of intensity speckle patterns. $I = \sum_{k=1}^{N} I_k$ where $I = |A|^2$ and $I_k = |A_k|^2$. The correlation between N intensity components is

$$C_{KL} = \frac{\langle I_K I_L \rangle - \langle I_K \rangle \langle I_L \rangle}{\left[\langle (I_K - \langle I_K \rangle)^2 \rangle \langle (I_L - \langle I_L \rangle)^2 \rangle \right]^{\frac{1}{2}}} \qquad (3-8).$$

The intensity correlations are a result of correlations between the amplitudes of speckle pattern

$$U_{KL} = \frac{\langle A_k A_e^* \rangle}{\left[\langle |A_k|^2 \rangle \langle |A_e|^2 \rangle \right]^{\frac{1}{2}}} \qquad (3-9).$$

*denotes circular Gaussian statistics resulting in the limit of a large number of independent, scattered contributions.

For a single look SAR system the intensity of the image is the sum of the squares of the in phase (real) and quadrature (imaginary) components of the radar return. The components are assumed to be zero mean with Gaussian distributions. The intensity is $I = Re[A^2] + Im[A^2]$. The intensity has a Chi Squared distribution with two degrees of freedom, i.e. a Rayleigh distribution [16] For multilook SAR systems the intensity is the average of the square roots of the intensities from each look. If N is defined as the number of looks and $A_i = \sqrt{I_i}$, then $Z = \frac{1}{N}\sum_{i=1}^{N} A_i, Z = \frac{1}{N}\sum_{i=1}^{N}\sqrt{I_i}$ where Z is the gray level of the final pixel value and $\sqrt{I_i}$ has a Chi distribution with two degrees of freedom.

3.4 Statistics

The first order statistics of speckle are measured from a single observation point. The single observation point fully describes the brightness/intensity of the speckle. The second order statistics give a measure of the coarseness of the spatial properties of speckle.

The second order statistics are the autocorrelation and its frequency representation, the power spectral density. The speckle is observed at a distant point from the rough surface on which the radiation was scattered. The auto correlation is calculated at this distant point from the intensity function $I(x,y) = |A(x,y)|^2$ where (x,y) denotes the x,y plane. The autocorrelation function is given by

$$R_{II}\{I(x_1,y_1), I(x_2,y_2)\} = \langle I(x_1,y_1), I(x_2,y_2)\rangle \qquad (3-10).$$

The ensemble average is computed by averaging over the ensemble of rough surfaces. The average width of the speckle is directly related to the width of the autocorrelation function.

The autocorrelation is calculated assuming that the reflecting surface is rough compared to the wavelength of the incident wave. The amplitude of the reflected field $A(x,y)$ is a circularly complex Gaussian random variable at each point (x,y). The autocorrelation of the intensity can be related to the electromagnetic fields reflected. This is given by

$$R_{AA}\{A(x_1,y_1),A(x_2,y_2)\}=\langle A(x_1,y_1),A^*(x_2,y_2)\rangle \qquad (3-11).$$

R_{AA} is called the mutual intensity of the field. R_{II} is related to R_{AA} by

$$R_{II}=\langle I(x_1,y_1),I(x_2,y_2)\rangle+|R_{AA}\{A(x_1,y_1),A(x_2,y_2)\}|^2 \qquad (3-12).$$

$R_{AA}(x,y;x,y)$ is defined as $\langle I(x,y)\rangle$. Therefore R_{II} is calculated by finding R_{AA}.

The autocorrelation is based on three assumptions: 1) for the calculation we are only concerned with the modulus of R_{AA}, 2) the microstructure of the scattering surface is very fine and is unresolvable from the distant observation point, 3) the scattering region is uniform and square - LxL meters. With these three assumptions the autocorrelation reduces to

$$R_{II}(\Delta x,\Delta y)=\langle I\rangle^2\left[1+\sin c^2\left(\frac{L\Delta x}{\lambda z}\right)\sin c^2\left(\frac{L\Delta y}{\lambda z}\right)\right] \qquad (3-13).$$

Therefore the average width of the speckle is the value of Δx where $\sin c^2\left(\frac{L\Delta x}{\lambda z}\right)$ is equal to zero.

The power spectral density (PSD) of the intensity of the speckle $I(x,y)$ is the Fourier transform of the autocorrelation function $R_{II}(\Delta x, \Delta y)$. For the case of a uniform and square scattering spot the PSD is

$$P_{II}(v_x, v_y) = \langle I \rangle^2 \left[\delta(v_x, v_y) + \left(\frac{\lambda z}{L}\right)^2 \Lambda\left(\frac{\lambda z}{L} v_x\right) \Lambda\left(\frac{\lambda z}{L} v_y\right) \right] \qquad (3-14)$$

Λ is the triangle function and $\begin{cases} \Lambda = 1 - |x| & \text{for } |x| \le 1 \\ 0 & \text{otherwise} \end{cases}$. Speckle has no frequency components above $\frac{L}{\lambda z}$ in both the v_x and v_y directions.

3.5 Multiplicative Nature of Speckle

SAR uses coherent radiation to produce images. The coherent processing inherently produces speckle in the images. It has been repeatedly documented that speckle is multiplicative in nature [14,17,18,19]. The random multiplicative noise increases with the average gray level of the image [17]. As shown in Section 3.2 speckle has a negative exponential distribution. The standard deviation is equal to the mean, thus the multiplicative nature of speckle.

Not all researchers agree that speckle is multiplicative under all conditions. In [20] it is shown that the multiplicative model fails when the object contains fine details which cannot be resolved by the imaging system. Reference [21] models speckle as a physical process of the coherent image formation. The model includes noise in the form of signal dependent effects and shows that the speckle is spatially correlated.

3.5.1 Statistics of Multiplicative Noise

For multiplicative noise $z_{ij} = x_{ij} v_{ij}$, where x_{ij} is the ij pixel without noise, v_{ij} is the noise contribution to the ij pixel, and z_{ij} is the noisy pixel. v_{ij} has a mean, μ_v, of one and a standard deviation, σ_v, which is independent of x_{ij}. Since the signal x and the noise v are independent the mean of the noisy pixels is $\mu_z = \mu_x$

$$E\{z\} = E\{xv\} = E\{x\}E\{v\} = E\{x\} \qquad (3-15)$$

The variance is

$$\begin{aligned} E\{(z-\mu_z)^2\} = E\{(xv-\mu_x\mu_v)^2\} &= E\{x^2v^2\} - 2\mu_x\mu_v E\{xv\} + (\mu_x\mu_v)^2 \\ &= E\{x^2\}E\{v^2\} - 2\mu_x^2\mu_v^2 + (\mu_x\mu_v)^2 \\ &= E\{x^2\}E\{v^2\} - \mu_x^2\mu_v^2 \end{aligned} \qquad (3-16)$$

For areas of constant signal, i.e. over relatively flat areas of the SAR image $E\{x^2\} = \mu_x^2$. Therefore the variance becomes

$$\begin{aligned} E\{(z-\mu_z)^2\} &= \mu_x^2 E\{v^2\} - \mu_x^2\mu_v^2 \\ &= \mu_x^2\left(E\{v^2\} - \mu_v^2\right) \\ &= \mu_x^2\sigma_v^2 \end{aligned} \qquad (3-17)$$

Equation (3-17) shows that the variance is the mean of x squared times the variance of the noise. The standard deviation $\sigma_v = \frac{\sigma_z}{\mu_x} = \frac{\sigma_z}{\mu_z}$ where σ_z is the standard deviation of the noisy signal and μ_z is its mean. The marginal PDF of the speckle intensity has a negative exponential distribution as given in (3-3). Therefore $\sigma_v = 1$ and $\sigma_z = \mu_z$. This says that the standard deviation of the noise is equal to the mean, which is 1.

CHAPTER 4

ADAPTIVE CORRELATION ENHANCER ALGORITHM

This report presents a unique implementation of the Adaptive Correlation Enhancer algorithm (ACE), an open-loop adaptive filter. The two-dimensional version of ACE was applied to speckled SAR images with minimal degradation in resolution. This was accomplished by using a unique approach. Data amplitude compression was applied prior to filtering to control the open-loop gain and maintain the image resolution. In addition a nonlinear data amplitude compression technique was applied after the filtering process to balance the contrast of the image.

The one-dimensional ACE algorithm was first reported in [22] where it was applied to seismic data. The ACE algorithm updates the filter coefficients of a finite impulse response (FIR) filter and under steady state conditions it converges to the autocorrelation of the input data. The statistical properties of the ACE algorithm were first studied in [23] and are included in Section 4.1. 2DACE is the two-dimensional implementation of ACE. It is important to understand the development of the ACE algorithm in order to fully understand and apply the 2DACE algorithm.

4.1 Adaptive Correlation Enhancer Algorithm

The Adaptive Correlation Enhancer is an open-loop architecture for updating FIR filter weights. The FIR filter size is proportional to the lag of the update equation described below. The ACE filter coefficient update equation is given in (4-1).

$$w[n] = \beta w[n-1] + (1-\beta)\frac{\boldsymbol{x}[n]x[n]}{Lp[n]} \qquad (4-1)$$

$w[n]$ is the filter coefficient vector of size $(2L+1)x1$ and is given in equation (4-2)

$$w[n] = \begin{bmatrix} w_o[n] \\ w_1[n] \\ M \\ w_{2L}[n] \end{bmatrix} \qquad (4-2)$$

$\boldsymbol{x}[n]$ is the input data vector with matrix dimensions of $(2L+1)\times 1$.

$$\boldsymbol{x}[n] = \begin{bmatrix} x[n-L] \\ M \\ x[n] \\ M \\ x[n+L] \end{bmatrix} \qquad (4-3)$$

This notation shows that the ACE algorithm is a noncausal filter. β is referred to as the smoothing parameter or the adaptation constant. β is adjusted by the user and determines the rate of convergence of the filter. The output signal vector is the convolution of the input vector and the filter coefficient vector.

$$y[n] = x[n] * w[n] \qquad (4-4)$$

$p[n]$ is a measurement of the difference between the present input signal power $x^2[n]$ and the most recent recursive measurement $p[n-1]$. As in the update coefficient equation the past value is weighted with the value of β and the present signal power is weighted with $(1-\beta)$.

$$p[n] = \beta p[n-1] + (1-\beta)x^2[n] \qquad (4-5)$$

The filter update equation is rewritten in equation (4-6) where $k = \frac{(1-\beta)}{Lp[n]}$.

$$w[n] = \beta w[n-1] + k\boldsymbol{x}[n]x[n] \qquad (4-6)$$

For a stationary input $x[n]$, the above equation can be reduce to equation (4-7).

$$w[n] = \beta w[n-1] + k\boldsymbol{x}[n]x[n] \;\rightarrow\; w[n] = \frac{k}{(1-\beta)}\boldsymbol{x}[n]x[n] \qquad (4-7)$$

The steady state ACE impulse response is calculated by taking the expectation of both sides of equation (4-7)

$$E\{w[n]\} = \frac{k}{(1-\beta)}E\{\boldsymbol{x}[n]x[n]\} \qquad (4-8).$$

The expectation of $\boldsymbol{x}[n]x[n]$ is the autocorrelation of $x[n]$

$$E\{\boldsymbol{x}[n]x[n]\} = \begin{bmatrix} E\{x[n]x[n-L]\} \\ M \\ E\{x[n]x[n]\} \\ M \\ E\{x[n]x[n+L]\} \end{bmatrix} = \begin{bmatrix} R_{xx}(L) \\ M \\ R_{xx}(0) \\ M \\ R_{xx}(L) \end{bmatrix} \qquad (4-9).$$

By inspecting equation (4-9) it is clear that the steady state filter coefficients are clearly proportional to the autocorrelation of the input vector,

$$w[n] = \frac{k}{(1-\beta)}R_{xx}[n] \qquad (4-10).$$

4.1.1 Stability of ACE Algorithm

The stability of the ACE algorithm is governed by the adaptation constant, β. The Z transform of equation (4-1) is shown in equation (4-11).

$$W[z] = \frac{k}{(1-\beta z^{-1})}\boldsymbol{X}[z] * X[z] \qquad (4-11).$$

The convolution operator is $*$ and the Z transform of $x[n]$ is $X[z]$.

$$\boldsymbol{X}[z]=\begin{bmatrix} z^{-L}X[z] \\ M \\ X[z] \\ M \\ z^{+L}X[z] \end{bmatrix} \qquad (4-12)$$

The Z transform of the input vector $x[n]$ shows that L is the maximum lag of the system. The input vector can be shifted by L and the input vector becomes

$$\boldsymbol{X}'[z]=\begin{bmatrix} z^{-2L} \\ M \\ z^{-L} \\ M \\ z^{0} \end{bmatrix} X[z] \qquad (4-13).$$

$\boldsymbol{X}'[z]$ is the causal implementation of the ACE algorithm.

The adaptation constant, β, determines the region of convergence for the algorithm. The term $\frac{k}{(1-\beta z^{-1})}$ in the Z transform equation (4-11) shows that the region of convergence (ROC) is $|z|>\beta$ for the system to be a right handed, causal sequence, and $\beta<1$ for the system to be stable.

4.1.2 Effects of the Adaptation Constant

The smaller the magnitude of β, the less "memory" the recursive system has. This can be seen by inspecting equation (4-1).

$w[n]=\beta w[n-1]+(1-\beta)\frac{\boldsymbol{x}[n]x[n]}{Lp[n]}$. For $\beta<0.5$, $w[n]$ has more of a dependence on the current input data and less of a dependence on previous histories of the filtering coefficients.

The inverse Z transform of $W[z]$ is shown in equation (4-14).

$$w[n] = k\beta^n \boldsymbol{x}[n]x[n] \qquad (4-14).$$

The filter coefficient vector $w[n]$ is a function of the impulse response $h[n] = k\beta^n$ and the input data $\boldsymbol{x}[n] \cdot x[n]$. The time constant, η, of the algorithm is $\frac{1}{\ln\beta}$ and is approximated by $\frac{1}{1-\beta}$. As shown in equation (4-1) along with $\eta = \frac{1}{1-\beta}$, β controls the rate of adaptation of the filter coefficients. For small β there is a smaller contribution of the past coefficients and thus the system will adapt faster to the present input data. As with all adaptive algorithms the mean square error also increases as the adaptation rate decreases. Therefore to minimize the error due to gradient noise the smallest β is not necessarily the best.

The behavior of the ACE algorithm as a function of β can be shown with the variance of the filter coefficients. The variance of $w[n]$ is calculated in [22], [23] and is shown in equation (4-15).

$$\sigma_{wi} = \begin{cases} \dfrac{2k^2\sigma^2\left(\sigma^2 + A^2\right)}{(1-\beta)^2} + \dfrac{A^4k^2}{8\left(1 - 2\beta\cos(2\omega_o) + \beta^2\right)} & , \quad i = L \\ \dfrac{k^2\sigma^2\left(\sigma^2 + A^2\right)}{(1-\beta)^2} + \dfrac{A^4k^2}{8\left(1 - 2\beta\cos(2\omega_o) + \beta^2\right)} & , \quad i \neq L \end{cases} \qquad (4-15)$$

As β increases the variance of the filter coefficients decreases. As β increases the adaptation rate slows down and the algorithm uses more samples to obtain the estimate of the expected filter coefficients.

4.1.3 Gain of ACE Algorithm

An extensive analysis of the one dimensional ACE algorithm performance for a narrowband signal in additive broadband noise is documented in [23]. The analysis shows that the filter gain approximately equals the filter lag, L. The gain is defined as $G = \frac{SNR_{out}}{SNR_{in}}$.

4.2 Two-Dimensional ACE – 2DACE

The two-dimensional implementation of the Adaptive Correlation Enhancer was first presented in [13]. The research presented by C.D. Knittle and N. Magotra used the 2DACE algorithm to enhance features in varying contrast and to extract linear features in noisy images. One application of the 2DACE algorithm is to enhance narrowband signals in broadband noise. This was demonstrated by using the 2DACE on SAR images, along with a low pass filter, to detect and enhance linear perturbations caused by surface ships on the ocean. The horizontal lowpass filter was used to filter the speckle in the SAR image. Then the 2DACE filter was used to enhance the wakes of the surface ships. This work proves that the 2DACE algorithm has superior performance over a lowpass filter when applied to a speckle-corrupted image.

4.2.1 Analysis of 2DACE Algorithm

The analysis of the 2DACE algorithm directly follows the analysis of the ACE algorithm in Section 4.1. The 2DACE analysis is documented in [24] and repeated here for completeness.

The two-dimensional filter weight coefficients are

$$w[m,n] = \beta w[m,n-1] + (1-\beta)\frac{\boldsymbol{x}[m,n]x[m,n]}{2L^2 p[m,n]} \qquad (4-16).$$

This notation is for a horizontally scanning filter with dimension $(2L+1)\times(2L+1)$, where L is the lag of the filter. The 2DACE algorithm implemented in MATLAB scans across the columns in the image. The software implementation is shown in equation (4-17).

$$w_{j,k}[m,n] = \beta \cdot w_{j,k}[m,n-1] + (1-\beta)\frac{x[(m-L)+j,(n-L)+k]x[m,n]}{2L^2 p[m,n]} \quad 0 \le j,k \le 2L \quad (4-17)$$

One option is to transpose the input image and then the horizontally scanning filter scans down the rows. The filtered image is then transposed back to the initial $[m,n]$ matrix. When the last column in a row is reached the filter is moved to the first column in the next row. This can be referred to as left to right filtering. The filter coefficients $w[m,n]$ are written in matrix form as

$$w[m,n] = \begin{bmatrix} w_{0,0}[m,n] & \Lambda & w_{0,2L}[m,n] \\ w_{1,0}[m,n] & \Lambda & w_{1,2L}[m,n] \\ M & O & M \\ w_{2L,0}[m,n] & \Lambda & w_{2L,2L}[m,n] \end{bmatrix} \qquad (4-18)$$

The subscript $2L,2L$ refers to the element in the filter weight matrix and m,n refers to the index of the input image matrix $x[m,n]$. The input matrix to the filter is of the form shown in equation (4-19) where m,n is the element index.

$$\boldsymbol{x}[m,n] = \begin{bmatrix} x[m-L,n-L] & \Lambda & x[m-L,n+L] \\ x[m-L+1,n-L] & \Lambda & x[m-L+1,n+L] \\ M & O & M \\ x[m+L,n-L] & \Lambda & x[m+L,n+L] \end{bmatrix} \qquad (4-19)$$

β is referred to the smoothing parameter or adaptation constant, and L is the maximum lag of two-dimensional filter.

In equation (4-16) the present value of the input data is divided by a recursive estimate of the input signal power, $p[n,m]$.

$$p[m,n] = \beta p[m,n-1] + (1-\beta)x^2[m,n] \qquad (4-20)$$

Defining $k = \frac{(1-\beta)}{2L^2 p[m,n]}$ allows equation (4-16) to be written in the form show below in equation (4-21).

$$w[m,n] = \beta w[m,n-1] + k\boldsymbol{x}[m,n]x[m,n] \qquad (4-21)$$

The expectation of the two-dimensional filter weight coefficients is developed in equations (4-22) through (4-24).

$$E\{w[m,n]\} = E\{\beta w[m,n-1] + k\boldsymbol{x}[m,n]x[m,n]\} \qquad (4-22)$$

$$E\{w[m,n]\} = \beta E\{w[m,n-1] + k\boldsymbol{x}[m,n]x[m,n]\} \qquad (4-23)$$

$$E\{w[m,n]\} = \frac{k}{1-\beta} E\{\boldsymbol{x}[m,n]x[m,n]\} \qquad (4-24)$$

The right hand side of equation (4-24) is $\frac{k}{1-\beta}$ times the two-dimensional autocorrelation of the input data as shown in equation (4-25).

$$E\{\boldsymbol{x}[m,n]x[m,n]\} = \begin{bmatrix} R_{xx}[-L,-L] & \Lambda & R_{xx}[-L,L] \\ R_{xx}[-L+1,-L] & \Lambda & R_{xx}[-L+1,L] \\ M & O & M \\ R_{xx}[L-1,-L] & \Lambda & R_{xx}[L-1,L] \\ R_{xx}[L,-L] & \Lambda & R_{xx}[L,L] \end{bmatrix} \qquad (4-25)$$

$Rxx(i,j)$ is the two-dimensional autocorrelation of the input data where $|i,j| \le L$

4.2.2 Stability of 2DACE Algorithm

As with the ACE algorithm the stability of the 2DACE algorithm is apparent after taking the Z transform of equation (4-21). The two-dimensional convolution is denoted by $*$ and the Z transform of the input matrix, $x[m,n]$, is $X[z_1,z_2]$. The Z transform is shown below in equations (4-26) and (4-27).

$$W[z_1,z_2]=\frac{k}{\left(1-\beta z_1^{-1}\right)}\boldsymbol{X}[z_1,z_2]*X[z_1,z_2] \qquad (4-26)$$

$$\boldsymbol{X}[z_1,z_2]=\begin{bmatrix} z_1^{-L}z_2^{-L} & \Lambda & z_1^{-L}z_2^{L} \\ z_1^{-L+1}z_2^{-L} & \Lambda & z_1^{-L+1}z_2^{L} \\ M & O & M \\ z_1^{L}z_2^{-L} & \Lambda & z_1^{L}z_2^{L} \end{bmatrix} X[z_1,z_2] \qquad (4-27)$$

In equation (4-26) it is seen that the filter is stable and causal for $|z_1|>\beta$ as long as $\beta<1$.

4.2.3 Adaptation Constant

Within the Z transform equation of the filter coefficients the term $\frac{1}{\left(1-\beta z^{-1}\right)}$ can be approximated by the geometric series with ratio β in the time domain. The time constant of the 2DACE algorithm is defined as $\tau\approx\frac{1}{1-\beta}$. By fitting τ to the geometric series and showing that $\beta=e^{\frac{-1}{\tau}}$ it is evident that β controls the rate of adaptation. As $\beta\rightarrow 1$ τ rapidly increases. β is also referred to the smoothing parameter. This is a good explanation for the effect that β has on the input data.

For larger values of β the processed SAR image has significant smearing, caused by the smoothing effect of β.

4.2.4 Gain and Matched Filter of the 2DACE Algorithm

The two-dimensional filter gain is derived in [24] as $G_{j2d} = \frac{SNR_{in}}{SNR_{out}} \approx 2L^2$.

For the input signal in equation (4-28) the matched filter can be derived as follows.

$$x[m,n] = A\cos(\pi(\gamma_{0m} + \alpha_{on})) + \upsilon[m,n] \qquad (4-28)$$

$\upsilon[m,n]$ is a white noise process with variance σ^2 and γ_0, α_0 are vertical and horizontal frequencies. The impulse response of the 2DACE filter with the given input is shown in equation (4-29).

$$h_{m,n}(i,j) = \frac{k}{1-\beta}\left[\frac{A^2}{2}\cos(\pi\gamma_{o,i-L} + \pi\alpha_{o,j-L} + \sigma^2\delta(i-L, j-L))\right] \qquad (4-29)$$

The impulse response $h_{m,n}(i,j)$ resembles a matched filter excluding the impulse term $\sigma^2\delta(i-L, j-L)$.

CHAPTER 5

IMPLEMENTATION OF THE 2DACE ALGORITHM

The Two-Dimensional Adaptive Correlation Enhancer Algorithm is implemented by convolving the input image, a (j,k) matrix, with the 2D filter coefficients, $w[m,n]$. Each input pixel, element of the (j,k) matrix, is multiplied with each filter coefficient from $w[m,n]$ and then summed to produce $y[j,k]$ as given by

$$y[j,k] = \sum_{m=j-L}^{j+L} \sum_{n=k-L}^{k+L} x[j,k] \cdot w[m,n] \qquad (5-1).$$

Refer to Section 4.2 equation (4-16) for the filter update equation. The filter coefficients, $w[m,n]$, are updated at every pixel with the current statistics. Refer to Appendix C for a complete listing of the MATLAB code that performs the 2DACE filtering on an input image.

5.1 2DACE Filter Parameters

Implementing the 2DACE filter algorithm to best filter the SAR images required finding the optimum filter parameters and developing pre-processing and post-processing routines. Tuning the filter design to achieve the best balance between reducing speckle and maintaining image resolution involved filtering synthetic data and actual SAR images with many variations of the filter parameters. The parameters affecting the speckle reduction and resolution are 1) normalizing and unnormalizing routines 2) scaling option 3) adaptation constant, β 4) filter lag, L.

5.2 2DACE Tests with a 2D Gaussian Image

The initial investigation of the filter parameters used a two-dimensional Gaussian image embedded in white Gaussian noise of variance 0.02. The test image was 256×256 pixels in size and the SNR was -0.5862 dB. Refer to Figure 7 for the 2D Gaussian image and to Figure 8 for the 2D Gaussian image embedded in white Gaussian noise. Each parameter was individually varied while the other parameters were held constant. The output image was visually examined for degradation of the image and reduction in speckle. The subjective visual test was very important because a filtered image with high SNR could have poor visual characteristics due to target warping or DC shifts.

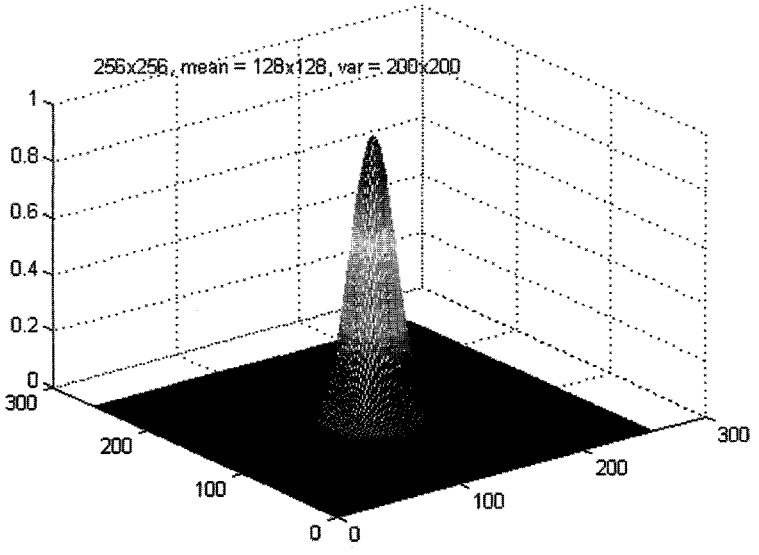


Figure 7. Mesh plot of a two-dimensional Gaussian image. The image is 256x256 pixels with $\mu_x, \mu_y = 128, \sigma_x^2, \sigma_y^2 = 200$, and a maximum amplitude of 1.

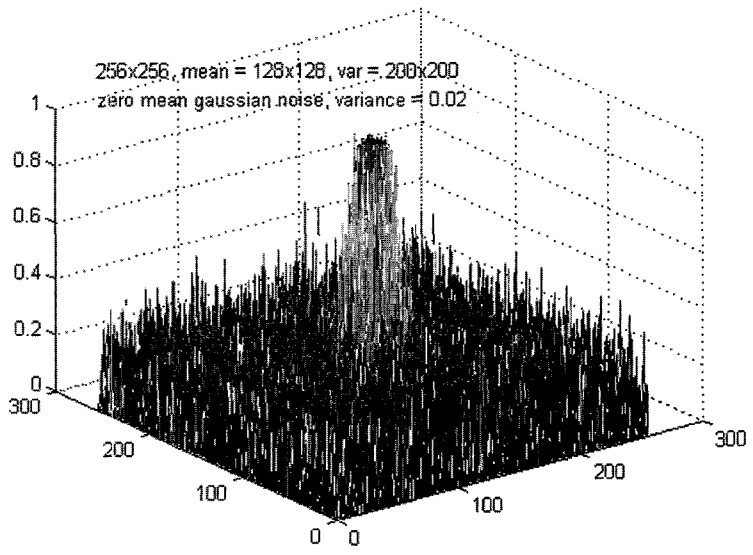


Figure 8. Mesh plot of a two-dimensional Gaussian image with zero mean AGWN. The image is 256x256 pixels with $\mu_x, \mu_y = 128, \sigma_x^2, \sigma_y^2 = 200$, and a maximum amplitude of 1. The variance of the AWGN is 0.02.

The signal-to-noise ratio was used as a method for quantitatively measuring the speckle reduction. The method used for calculating the SNR is the least square method. This definition of the SNR gives a measure of the difference between the two images. Equations (5-2) and (5-3) use the variance of the original, noise free image and the least mean square criterion [25] of the original image and the noisy image.

$$\mathrm{SNR}_{\mathrm{dB}} = 10\log_{10}\frac{\sigma^2}{\sigma_{\mathrm{ls}}^2} \qquad (5-2)$$

$$\sigma_{\mathrm{ls}}^2 = \frac{1}{\mathrm{MN}}\sum_{m=1}^{M}\sum_{n=1}^{N}\left|x(m,n) - x'(m,n)\right|^2 \qquad (5-3)$$

Equation (5-3) calculates the difference between the original image pixels $x(m,n)$ and the noisy image pixels $x'(m,n)$. The difference σ_{ls}^2 is the least square error.

For a noisy image the least square error, σ_{ls}^2, is large compared to the variance of the original image, σ^2, and the SNR_{dB} is negative. For filtered images σ_{ls}^2 is small compared to σ^2 and the SNR_{dB} is positive. This is a relative measurement in the sense that it is dependent on the original, noise free image. It gives a measurement used to determine the amount of noise removed by the filtering process along with the any changes in the image that increase the least square error.

The purpose of initial test series with the Gaussian image was to select the best normalizing/unnormalizing routine and to gain an understanding in how the filter parameters affect the filtering process. The degradation in resolution of the test image was investigated in a second series of tests with a two-dimensional Taylor Point Spread Function (PSF).

The results of the parameters variations in the 2D Gaussian test series are documented in the next four sections.

5.2.1 Normalizing and Unnormalizing Routines.

For the initial filter tests three normalizing and four unnormalizing routines were tested. For the normalizing and unnormalizing tests the lag was fixed to 1 and the adaptation constant β was fixed at 0.5. Two different scaling options were used and are documented in the next section.

The normalizing routines are:

1. Calculate the mean and dynamic range of the image. Subtract the mean from the image and scale the dynamic range to +/-1.

2. Calculate the image mean and subtract it from the image. Calculate the dynamic range of the zero mean image and then scale it to +/- 1.
3. No normalizing routine

The unnormalizing routines are

1. Factor the input mean and dynamic range into the output image.
2. Apply a DC shift to the filtered image so the minimum value is zero.
3. Apply a DC shift to the filtered image so the maximum pixel value equals a fixed value.
4. No normalizing routine.

Refer to Table 1 for a tabulation of the filter parameters and the calculated SNRs from each filtering process.

5.2.2 2DACE Scaling Options

Three different scaling options are built into the MATLAB implementation of the 2DACE algorithm. Refer to the 2D filter coefficient equation (4-21,5-4).

$$w[m,n] = \beta w[m,n-1] + k\boldsymbol{x}[m,n]x[m,n] \qquad (5-4),.$$

Three variations in k are investigated and are listed in equations (5-5), (5-6) and (5-7).

$$\text{Scaling Option 1} \quad k = (1-\beta) \qquad (5-5)$$

$$\text{Scaling Option 2} \quad k = \frac{(1-\beta)}{2L^2} \qquad (5-6)$$

$$\text{Scaling Option 3} \quad k = \frac{(1-\beta)}{2L^2 p[m,n]} \qquad (5-7)$$

In equations (5-6) and (5-7) L is the lag of the filter and $p[m,n]$ is the recursive measurement of the signal power defined in equation (4-20).

Equations (5-5) and (5-6) show that the 2D filter coefficients, $w[m,n]$,are recursively computed with the past values of the filter weights and the input power scaled by a constant. Comparing the normalized data, Scaling Option 1 produces the same results as Scaling Option 2. These two equations have a constant ratio, i.e. equation (5-5) multiplied by $2L^2$ is equal to equation (5-6). Because the two equations are scalar multiples of each other only test results of equation (5-6), Scaling Option 2, are documented in this thesis.

Equation (5-7), Scaling Option 3, shows that the 2D filter coefficients are recursively updated with past values of the filter weights and the input power scaled by $k = C/P$, where $C = \frac{(1-\beta)}{2L^2}$ and $P = p[m,n]$ is the recursively computed input signal power of equation (4-20).

Scaling Option 3 is the scaling factor documented in the analysis of the 2DACE algorithm in Section 4.2. Scaling Option 2 is simply the constant portion of Scaling Option 3. Refer to Table 1 below where the results of 2D Gaussian tests show the performance of the scaling options and normalizing/unnormalizing routines.

Table 1 2DACE Scaling and Normalizing Tests

SNR_{dB} in = -0.5862 dB

Filtering Options			SNR Calculations				
Scaling	Norm	Unnorm	SNR_{dB} out	SNR_{dB} gain	SNR_{dB} cout	SNR_{dB} tgain	SNR_{dB} cgain
2	1	1	1.5391	2.1253	-12.3833	-11.7971	-13.9224
3	1	1	0.9984	1.5846			
2	1	2	4.3955	4.9817	-3.9947	-3.4085	-8.3902
3	1	2	-7.9529	-7.3667			
2	1	3	4.3995	4.9817	-3.9947	-3.4085	-8.3902
3	1	3	-7.9529	-7.3667			
2	2	1	-12.7338	-12.1476	-17.7308	-17.1446	-4.9970
3	2	1	-14.0725	-13.4863			
2	2	2	4.3995	4.9817	-3.9947	-3.4085	-8.3902
3	2	2	-7.9529	-7.3667			
2	2	3	4.3995	4.9817	-3.9947	-3.4085	-8.3902
3	2	3	-7.9529	-7.3667			
2	3	4	4.8744	5.4606	4.3540	4.9402	-0.5204
3	3	4	3.4899	3.9761			
Conv3			3.6119	4.1918			

Table 1 lists the scaling options, normalizing and unnormalizing routines, and the associated SNRs. Five SNR measurements are included in the table. The input SNR, SNR_{dB} in, is the ratio of the clear image variance, σ^2, and the least mean square variance, σ_{ls}^2, and has a value of -0.5862dB. Refer to equations (5-2) and (5-3). SNR_{dB} out is the ratio of σ^2 to the filtered σ_{ls}^2 and SNR_{dB} gain is SNR_{dB} out - SNR_{dB} in. A second set of SNR measurements is applied only to filtering operation that incorporate Scaling Option 2. As described in Section 5.3.2 Scaling Option 2 requires a nonlinear compression operation on the filtered data. SNR_{dB} cout is the ratio of σ^2 to the σ_{ls}^2 of the filtered and compressed image. SNR_{dB} tgain is the total gain in the signal-to-noise ratio between the input image and the filtered and compressed image, SNR_{dB} cout - SNR_{dB} in. SNR_{dB} cgain is the gain between the filtered image and the compressed

image. Since the noise floor is raised by the compression process $\mathrm{SNR_{dB}}$ cgain is always less than 1.

A benchmark, conv3, was created by filtering the noisy Gaussian image with a 3×3 lowpass (LPF) filter. The 2DLPF was chosen as the benchmark because it is a standard, easy to implement, speckle reduction method used at Sandia National Laboratories. The 2DLPF is shown in equation (5-8).

$$\mathrm{LPF_{3\times3}} = \begin{bmatrix} 0.3333 & 0.3333 & 0.3333 \\ 0.3333 & 0.3333 & 0.3333 \\ 0.3333 & 0.3333 & 0.3333 \end{bmatrix} \qquad (5-8)$$

Each element in the lowpass filter has a value of 0.3333. A 2D convolution with the input $(\mathrm{m,n})$ matrix and the 2DLPF produces the filtered output image.

From the results it is clear that the normalizing and unnormalizing routines introduce errors in the filtered image. Inspection of the filtered images revealed that the normalizing routines subtract out the fine detail that is very important in SAR images. In addition the unnormalizing routines introduce DC shifts that distort the contrast.

Scaling Option 2 has good performance before the required compression routine. The average gain SNR measurements are close to 5dB. After the compression routine the SNR is reduced to –8.4dB, which is significantly lower than the input SNR which is –0.5862dB. For the case when the normalizing or unnormalizing routines are not included in the image processing the gain in SNR is 5.5dB for Scaling Option 2 and 4.0dB for Scaling Option 3. Scaling Option 3 has the best performance, but the SNR is less than the benchmark lowpass

filtering which has a gain in SNR of 4.2. In Section 5.4 data is presented that shows the 2DACE filtering algorithm outperforms the 3×3 LPF operation.

The tests also reveal that Scaling Option 2 suppresses the fine details in the SAR image. Scaling Option 3 retains the fine images but it is very sensitive to the choice of β. The data in Section 5.2.3 show the effects of β on the filtering process. To see the effects of Scaling Option 2 and Scaling Option 3 refer to Figure 9 and Figure 10 at the end of this section. At first glance the filtered images are very similar. Figure 9 has a small DC offset but the Gaussian image is very similar to the input image. Figure 10 has a lower SNR and the Gaussian image has some discontinuities. In Section 5.4 the results of the resolution tests are discussed. Scaling Option 3 has the best performance when resolution is the judging criteria.

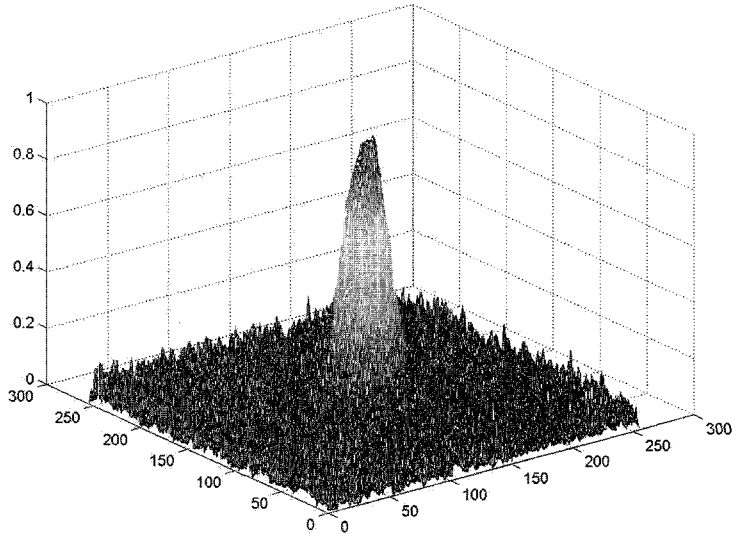


Figure 9. Mesh plot of the 2D Gaussian image filtered with 2DACE , Scaling Option = 2, Lag = 1, Beta = 0.50. The filtered image is normalized to 1.

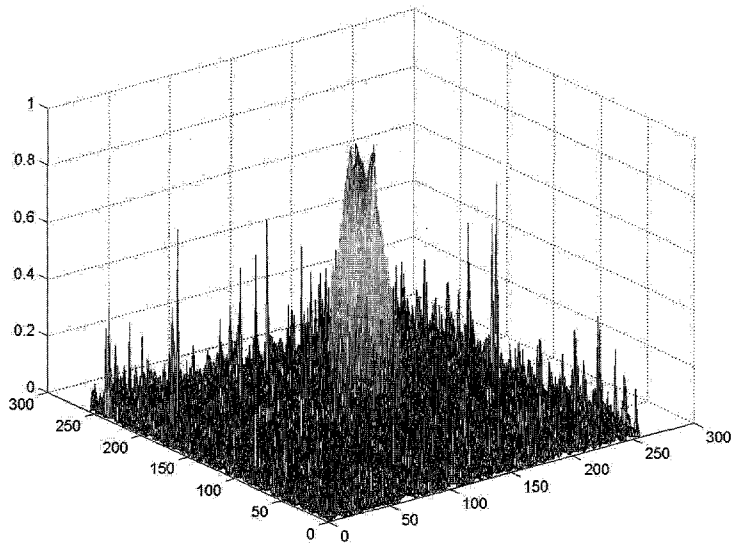


Figure 10. Mesh plot of the 2D Gaussian image filtered with 2DACE , Scaling Option = 3, Lag = 1, Beta = 0.50. The filtered image is normalized to 1.

In summary the results in Table 1 show that normalizing and unnormalizing routines degrade the quality of the image. Without the use of the normalizing routines Scaling Option 2 had a gain in SNR is 5.5dB and Scaling Option 3 had a gain of 4.0dB. Scaling Option 3 had a lower SNR than the benchmark lowpass filte which had a gain in SNR of 4.2. This was due to selection of the filter parameters. In Section 5.4 the superior performance of Scaling Option 3 is presented.

5.2.3 Adaptation Constant, β

The adaptation constant, β, is also referred to as the smoothing parameter. Tests show that for Scaling Option 2 as β increases the objects in the images are smoothed out and distorted. A very low value of β has the best performance when used with Scaling Option 2. For values of β, ~ 0.2 to 0.3, the

objects in the images are slighted warped or smoothed out. For large values of β, ~0.8 to 0.9, there is lateral distortion of the objects in the images. Figure 11 shows the 2D Gaussian test image filtered with Scaling Option 2 and β = 0.95. This image shows the significant warping of the image in the direction that the filter is shifted.

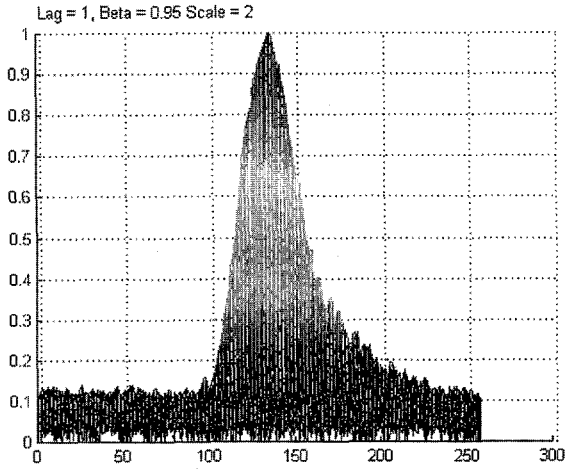


Figure 11. Side view of a 2D Gaussian image filtered with 2DACE, Scaling Option = 2, Lag = 1 Beta = 0.95. The plot shows the significant warping of the image in the direction that the filter is shifted

Many different filtering approaches were tried in an attempt to reduce the distortion when high values of β were used in the 2DACE algorithm. The approaches consisted of:

1. Shifting the filter from left to right, right to left, i.e. wrapping the filter around at the end of each row.
2. Scanning left to right, right to left, top to bottom then bottom to top.
3. Filtering small blocks of data.

None of these techniques reduced the distortion caused by the filtering with Scaling Option 2. As β is increased the distortion increases and the fine details and edges are smoothed out.

When Scaling Option 3 is used in the filtering algorithm the optimal β is between 0.5 and 0.8. Section 5.4 documents the resolution measurements and the optimum settings are defined. Table 2 lists the SNR measurements for different combinations of β and the filter lag, L.

Table 2 shows that as the filter lag was increased from 1 to 2 the SNR increased slightly. The drawback is that the resolution decreased. Therefore the best performance of the 2DACE filter is when the lag is set to 1, a 3×3 filter. The best values of β in terms of SNR are discussed below.

The results in Table 2 show that for Scaling Option 2 the SNR decreases as β increases. The input SNR is -0.5862 dB. The maximum gain in SNR for Scaling Option 2 is achieved with $\beta = 0.30$ and is 5.45dB before compression and 5.0dB after compression. For Scaling Option 3 the SNR increases as β increases. The distortion is also noticeable with Scaling Option 3 as β is increased past 0.85. The results show that the 2DACE filter significantly outperforms the 2DLPF used as a benchmark. For the 2DACE filter with Scaling Option 3, filter lag = 1, and $\beta = 0.75$ the gain in SNR is 10.1dB and the same measurement for the 2DLPF is 4.2dB.

Table 2 Filter Options and SNR Measurements

SNR_{dB} in = -0.5862 dB

Filtering Options			SNR Calculations				
Scaling	Lag	Beta	SNR_{dB} out	SNR_{dB} gain	SNR_{dB} cout	SNR_{dB} tgain	SNR_{dB} cgain
2	1	0.30	4.8582	5.4444	4.5404	5.1266	-0.3178
3	1	0.30	2.8615	2.9477			
2	1	0.50	4.8744	5.4606	4.3540	4.9402	-0.5204
3	1	0.50	3.3899	3.9761			
2	1	0.75	4.8546	5.4409	3.9080	4.4942	-0.9466
3	1	0.75	9.4761	10.0623			
2	1	0.95	4.5551	5.1413	0.9779	1.5641	-3.5772
3	1	0.95	10.4741	11.0603			
Conv3			3.6119	4.1918			
2	2	0.30	4.9612	5.5474	4.7370	5.3233	-0.2241
3	2	0.30	2.6233	3.2096			
2	2	0.50	4.9938	5.5800	4.5141	5.1003	-0.4797
3	2	0.50	3.1483	3.7346			
2	2	0.75	5.0115	5.5977	4.0341	4.6203	-0.9774
3	2	0.75	10.4135	10.9997			
2	2	0.95	4.6285	5.2147	1.0928	1.6790	-3.5357
3	2	0.95	10.6076	11.1939			
2	3	0.3	4.9723	5.5585	4.7979	5.3841	-0.1744
3	3	0.3	3.0817	3.6679			

5.2.4 Filter Lag, L

The effect of the filter lag, L, is similar to the adaptation constant β. The filter size is $2L+1$. Table 2 shows that there is a very small increase in the SNR as the lag is increased. As the filter size is increased the resolution is decreased, as shown in Section 5.4. This decrease in resolution is due to the increase in size of the filter weight matrix $w[m,n]$, where $(m,n)=(2L+1,2L+1)$. As the dimensions of weight matrix increase the two-dimensional convolution smoothes out the sharp edges of objects which reduces the resolution. In addition the convolution of the input data with the weight matrix uses more surrounding pixels

in the statistics and thus the fine details are lost. The reduction in resolution far outweighs the minor increase in SNR.

Another major effect of the filter lag is computational time. The 2DACE algorithm implementation in MATLAB has not been coded for the optimal computational time. This is not the focus of the thesis. For a standard 900×1280 SAR image the computational time is ~ 55 minutes when processed on a Pentium II 400 MHz processor. The computational time varies slightly, ~10%, depending on the scaling option selected. As the lag increases the computational time significantly increases. For a lag of 3, filter size of 7, the computational time is over 4 hours.

5.3 Pre-processing and Post-Processing

Pre-processing of the input data and post-processing of the filtered data are required. Data padding ensures that no pixels of the input image are destroyed by the filter start up. The filtered image should be compressed to improve the contrast. The intensity of the pixels corresponding to areas of low reflectivity is restored to a detectable level and the very high intensity areas are suppressed.

5.3.1 Data Padding

In the 2DACE algorithm the initial filter weights are set to zero. A finite number of operations are required to fill the filter coefficients with values based on the statistics of the image. The filter coefficient size is $(2L+1)\times(2L+1)$. Data from the surrounding pixels are used in the update equation to create the filter coefficient matrix. Therefore the filter lag, L, determines the size of the filter

coefficient matrix and the number of neighboring pixels used in the calculation. For example consider a filter lag of 1, if the current pixel value is at $x[j,i]$ the filter coefficient matrix will be calculated using pixels in the region $x[j-1,i-1]$ to $x[j+1,i+1]$. This shows that for pixels in the rows and columns that form the border of the image the filter coefficient update equation requires data outside of the image boundary.

The filter start up time requires padding the input image with reflected data around the borders. The data padding routine is designed to pad the borders with ten rows and columns. The routine reflects the data points symmetrically across the border. Data padding allows the filter coefficients to converge to the statistics of the image before the convolution is performed with pixels that are on the image's border. After the image is filtered the padded borders are cropped from the filtered image. Refer to Appendix D for the data padding routine.

5.3.2 Data Compression

Data compression is required when Scaling Option 2 is selected. The filter weights are updated with equation (5-9)

$$w[m,n] = \beta w[m,n-1] + \frac{(1-\beta)}{2L^2} x[m,n]x[m,n] \qquad (5-9).$$

The convolution of the input data with the filter weights is shown in equation (5-10)

$$y[m,n] = \sum_{j=m-L}^{m+L} \sum_{k=n-L}^{n+L} x[m,n]w[j,k] \qquad (5-10)$$

This equation shows that $y[m,n]$ is a function of the input data raised to the power of 3. Thus for values of $x[m,n]>1$ the output is much greater that the input. For values of $x[m,n]<1$ the output is much smaller than the input. The compression routine is required to reduce the dynamic range and enhance the fine features with low intensities.

The simple compression routine is the cube root of the filtered image, $y[m,n]_C = y[m,n]^{\frac{1}{3}}$. This restores the relation of $y[m,n]$ to the first power of $x[m,n]$.

Data compression is optional when Scaling Option 3 is selected. Compression is useful when the low intensity pixels of the filtered image need to be enhanced while the high intensity pixels need to be suppressed. The 2DACE algorithm with Scaling Option 2 produces filtered SAR images that are dark compared to the input image. Nonlinear compression is applied to the filtered image to enhances the low intensity pixels. The compression technique is of the form $y = x^C$ where x is the input data, c is the compression constant, and y is the output data. To find the best performance of the compression technique c was varied from 0.3 to 0.9. The contrast of the input image is matched when $c = 0.8$. With the optimum c the compression equation becomes $y = x^{0.8}$. Figure 12 shows the relation ship between input and output image pixel values for the compression equation $y = x^{0.8}$.

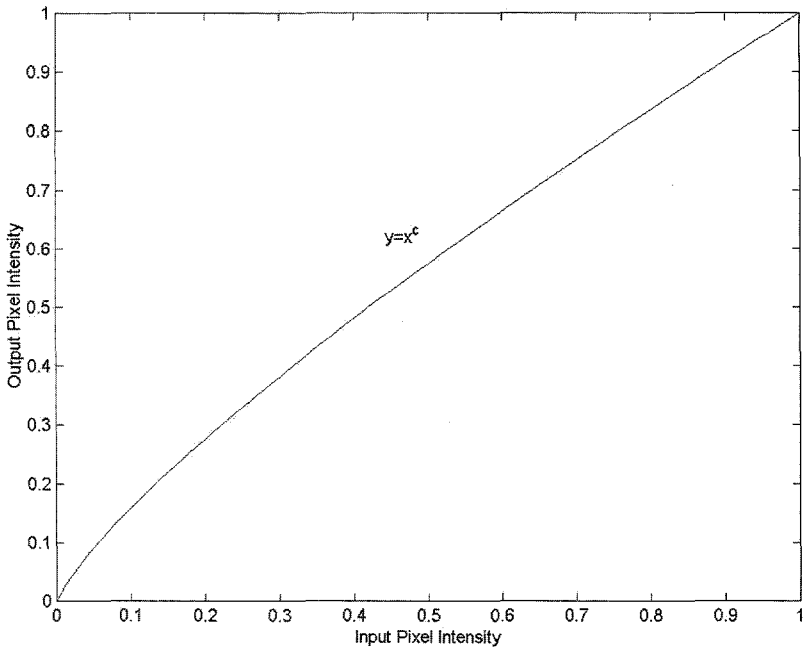


Figure 12. This plot shows the input to output mapping of the pixel amplitudes. The data amplitude compression $y = x^C$ is shown with $c = 0.8$.

5.4 Resolution Measurements of a 2D Taylor PSF and a SAR Point Target

Typically resolution is the most important characteristic of a SAR image. Every type of filtering involves a convolution process, therefore filters need to be carefully designed so the output data maintains the required characteristics. Maintaining the resolution while reducing speckle is the main focus of this thesis. This involves finding the 2DACE filter parameters that reduce the greatest amount of speckle yet do not adversely affect the resolution. The speckle reduction is quantified in Section 5.2. Table 2 shows that the greatest SNR improvement is with Scaling Option 3, and an adaptation constant, β, greater than 0.5. For β equal to 0.75 the increase in SNR is 10.1db. The degradation in resolution is quantified in this section.

To measure resolution a two-dimensional Taylor Point Spread Function (PSF) is used as a synthetic point target. Resolution is defined as the -3dB and -14dB widths of the normalized point target. The -3dB and -14dB widths are interpolated and the resolution units are pixels. Higher resolution means that smaller details can be resolved. Therefore if resolution is degraded the -3dB and -14dB measurements of the PSF are wider.

The synthetic target is generated with a zero padded one-dimensional Taylor window. A window size of 50 is used for generating the synthetic target. The FFT of the Taylor window produces the Taylor PSF. The $[50\times1]$ Taylor PSF is multiplied with its transpose, $[50\times1]\times[1\times50]=[50\times50]$, to create the two-dimensional $[50\times50]$ Taylor PSF. The MATLAB m file that creates the 2D Taylor PSF is included in Appendix E. Refer to Figure 13 for the 2D Taylor PSF.

The Taylor PSF is embedded in white Gaussian noise and tested with combinations of filter lags, adaptation constants, and the two scaling options. Additional tests are performed with the Taylor PSF embedded in a field of speckle taken from an actual SAR image. Finally to prove in the filter parameters a point target within a SAR image is tested. Refer to Figure 14 for the 2D Taylor PSF embedded in speckle and to Figure 15 for the SAR point target. The resolution measurements for the series of tests are documented in Appendix B.

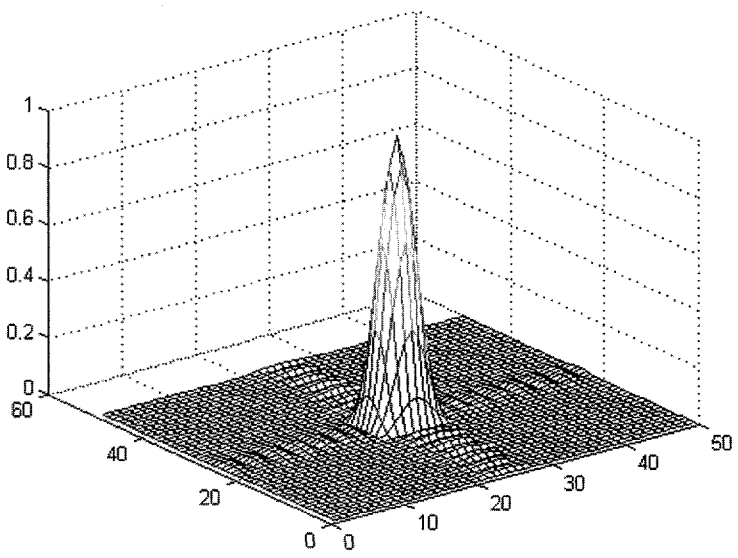


Figure 13. A 50x50 mesh plot of the 2D Taylor PSF generated from a 1D Taylor PSF. The 1D Taylor PSF is generated from the FFT of a $[50\times1]$ Taylor window. The $[50\times1]$ Taylor PSF is multiplied with its transpose, $[50\times1]\times[1\times50]=[50\times50]$, to create the two-dimensional $[50\times50]$ Taylor PSF. The maximum amplitude is 1.

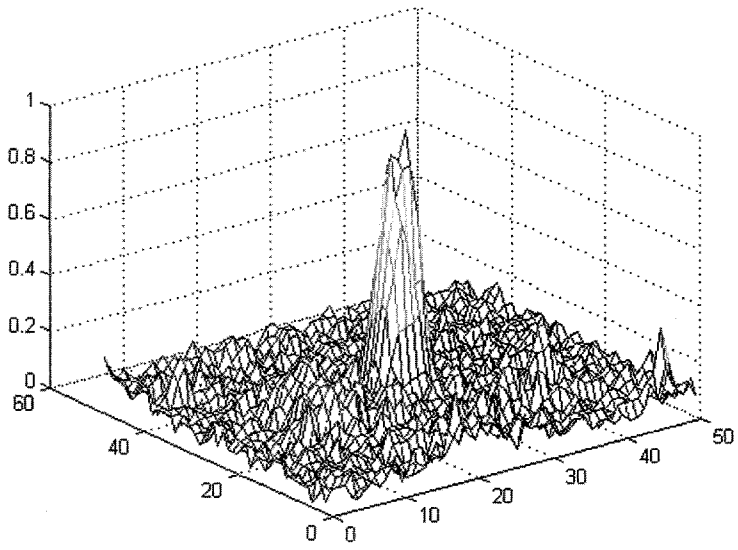


Figure 14. A 50x50 mesh plot of 2D Taylor PSF and SAR speckle. The speckle was cropped from the SAR image F2_335493_308.

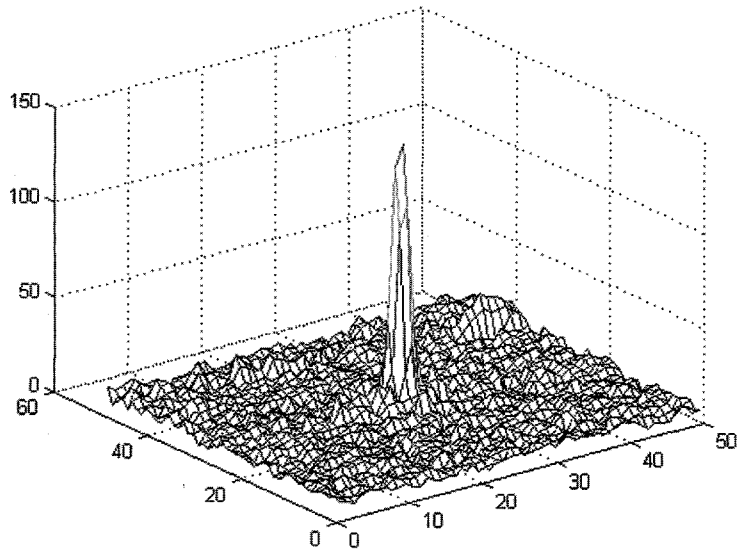


Figure 15. A 50x50 mesh plot of a SAR point target cropped from the SAR image F1_501442_217.

The -3dB and -14dB measurements are compared to the original image and the noisy image. The data, Appendix B Table 7, shows that the resolution is decreased for all filtering tests with Scaling Option 2. The -3dB and -14dB PSF measurements are greater in the filtered image than the -3dB and -14dB measurements in the clear and noisy images. The major increase in pulse width is in the horizontal direction. As the adaptation constant, β, increases from 0.3 to 0.9 the horizontal –14dB resolution measurements increase from 9.3 to 21.4 pixels. The equivalent resolution for the clear image is 7.2 pixels. The output image of the 3×3 2DLPF has narrower -3dB and -14dB PSF measurements then the 2DACE algorithm with Scaling Option 2. The –14dB horizontal resolution measurement is 8.7 pixels.

The drawback of a high β and Scaling Option 3 is that minor shadowing occurs in the direction that the filter is shifted, for all the tests documented this would be the vertical direction. No shadowing occurs in the horizontal direction. For a β value of 0.72 the shadowing is minimized and the resolution is very close to the input image. The results show that the horizontal resolution is better than the vertical resolution. Depending on the image characteristics the image can be rotated so the resolutions of the targets are maintained.

The 2DACE filtering algorithm with Scaling Option 3 has the best performance. The resolution of the filtered image closely matches the input image resolution when the filter lag is set to 1, and the adaptation constant, β, is between 0.60 and 0.95. The test results show that the vertical -3dB and -14dB PSF measurements increase with a smaller β and tend to decrease as β approaches 0.95. Table 3 includes the resolution measurements of the 2D Taylor PSF embedded in speckle for increments of β from 0.60 to 0.90.

Referring to the results, Table 3, the resolution of the filtered image is very close to the input image for values of β between 0.68 to 0.90. For β equal to 0.72 the percent difference in –3dB pixel width, when compared to the clear, noise free image, is less than 1.0% in the horizontal and vertical directions. The percent difference in –14dB pixel width is less than 1.3% in the horizontal direction and 7.1% in the vertical direction. When compared to the pulse widths of the noisy Taylor PSF the 2DACE algorithm with Scaling Option 3 reduces the pulse width and actually improves the resolution. The resolution increases with β (the pulse width decreases) as does the SNR.

The data also proves that the 2DACE algorithm outperforms the 3×3 *2DLPF. The percent difference in the –3dB pixel width for* 3×3 *2DLPF image and the clear image is 18.6% in the horizontal direction and 8.6% in the vertical direction. The percent difference in –14dB pixel width is close to 20% in the horizontal and vertical directions. Compared to the 2DACE data shown above it is evident that the 2DACE outperforms the* 3×3 *2DLPF.*

Table 3 Filter Tuning Measurements using 2D Taylor PSF

Taylor PSF with Speckle

Lag = 1, Scale = 3

	Horizontal Resolution Calculations (units = pixels)			**Vertical Resolution Calculations (units = pixels)**			
	3dB Resolution	14dB Resolution	Mean	3dB Resolution	14dB Resolution	Mean	LS SNR_{dB}
Clear image	3.4891	7.2331	0.1066	3.4891	7.2331	0.1066	
Noisy image	4.0911	8.0243	0.1812	3.7141	7.9859	0.1797	
Convolution with 3x3	4.1372	8.6717	0.2009	3.7889	8.7265	0.2107	
Beta							
0.60	3.6976	7.2471	0.1471	3.5987	8.3744	0.1676	3.4270
0.63	3.6509	7.2105	0.1434	3.5531	8.2452	0.1637	3.7788
0.66	3.5912	7.1804	0.1401	3.5125	8.0981	0.1601	4.1072
0.67	3.5761	7.1718	0.1391	3.5003	8.0443	0.1590	4.2110
0.68	3.5616	7.1640	0.1381	3.4889	7.9876	0.1578	4.3120
0.69	3.5309	7.1570	0.1372	3.4784	7.9280	0.1568	4.4100
0.72	3.4577	7.1412	0.1348	3.4522	7.7441	0.1538	4.6874
0.75	3.4109	7.1352	0.1330	3.4356	7.6190	0.1511	4.9423
0.76	3.4009	7.1359	0.1326	3.4325	7.5804	0.1503	5.0233
0.77	3.3936	7.1383	0.1323	3.4306	7.5435	0.1496	5.1027
0.78	3.3890	7.1424	0.1321	3.4301	7.5084	0.1490	5.1809
0.79	3.3872	7.1486	0.1320	3.4309	7.4752	0.1483	5.2581
0.80	3.3880	7.1569	0.1319	3.4332	7.4438	0.1478	5.3345
0.81	3.3915	7.1676	0.1320	3.4369	7.4145	0.1473	5.4104
0.84	3.4182	7.2138	0.1327	3.4571	7.3402	0.1463	5.6325
0.87	3.4530	7.2744	0.1338	3.4945	7.2932	0.1462	5.8172
0.90	3.4825	7.3802	0.1349	3.5740	7.2973	0.1482	5.7961

Figure 16 is a cross section of the input 2D Taylor PSF embedded in SAR speckle. Figure 17 is a cross section of the filtered Taylor PSF with $\beta = 0.72$, $L = 1$, and Scaling Option 3. The $-3dB$ and $-14dB$ resolution for each figure is documented in Table 3. The cross section in Figure 17 shows that for $\beta = 0.72$ the filter rebounds from zero in 8 samples and the resolution is very close to the resolution of the input image.

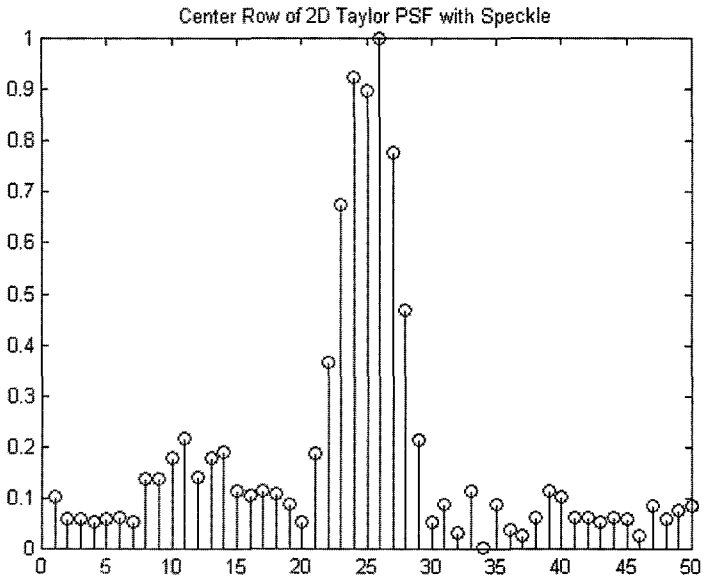


Figure 16. A cross section of the 2D Taylor PSF and SAR speckle.

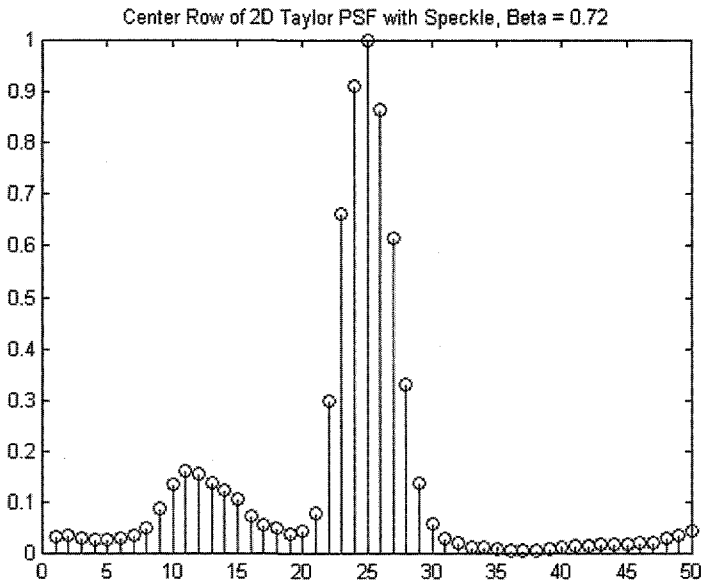


Figure 17. Cross section of the 2D Taylor PSF and SAR speckle filtered with 2DACE, Scaling Option = 3, Lag = 1, Beta = 0.72. The filtered image is normalized to 1.

Resolution measurements of the SAR point target support the filter parameter selections of $\beta = 0.72$, $L = 1$, and Scaling Option 3. The data shows that the horizontal and vertical resolutions of the filtered point target are 0.5 to 1.0 pixels greater than the input image, 11% to 33%. This decrease in resolution is still better than the decrease shown with the 3×3 2DLPF, which has a maximum increase in pixel width of 50%. Table 9 in Appendix B lists the fine tuning resolution data of the filtered SAR point target. The data shows that the horizontal resolution decreased as β increased, but the vertical resolution increased as β increased. The results don't match the results of the 2D Taylor PSF test. There is no change in resolution when the 2DACE filter is applied to the 2D Taylor PSFs.

The change is resolution of the SAR point target is due to the large dynamic range. The SAR point target has a peak amplitude of 150, which is very large compared to the speckle mean. Therefore the convolution process widens out the point target as it transitions from the base to the peak. To adjust for this problem the image is compressed before the 2DACE is applied, then the image is uncompressed after the filtering. Test results show that the compression reduces the dynamic range of the input image and as a result the resolution is not affected by the filtering process.

The data in Table 4 shows that with a compression of $y = x^c$, where c is the pre-compression factor, the resolution of the point target is equal before and after filtering. The resolution results are best for c = 0.10. For β equal to 0.72 and c = 0.10 the percent difference in –3dB resolution, when compared to SAR point target, is -2.2% in the horizontal direction and +6.1% in the vertical direction. The percent difference in –14dB resolution is less than -13% in the horizontal direction and -1.6% in the vertical direction. Referring to Figures 22 and 23 in Appendix A, it is apparent that the resolution of the point targets is unaffected by the filtering process.

Another benefit of the pre-compression is that the shadowing is removed. As shown in Figure 17 the filter generates a shadow in the direction that the filter is shifted. Since the dynamic range between point targets and the image mean is reduced, the filter won't clamp to a low value when transitioning from the peak.

Inspection of numerous filtered SAR images shows that the pre-compression factor, c, should be adjusted for each SAR image. Figure 23

shows the results when the pre-compression factor is 0.10. For other images the best value is closer to 0.25. Images with areas of very low pixel values tend to have black spots generated by the filtering process. This is the one drawback of the pre-compression operation. By increasing c, reducing the compression, the black spots are removed.

Table 4 SAR Point Target Resolution Measurements

Lag = 1, Scale = 3, Beta = 0.72

	Horizontal Resolution Calculations (units = pixels)		Vertical Resolution Calculations (units = pixels)	
	3dB	14dB	3dB	14dB
Speckle image	1.8776	4.5169	1.7609	4.3047
Convolution with 3x3	2.5805	6.7794	2.5037	6.3342
Pre-Compression Factor				
0.10	1.8364	4.7931	1.5536	4.2378
0.15	1.9486	4.9166	1.5771	4.2395
0.20	2.0787	5.0388	1.6161	4.2491
0.25	2.2239	5.1515	1.6713	4.3162
0.30	2.3456	5.2588	1.7485	4.4217
0.35	2.4626	5.3504	1.8033	4.5097
0.40	2.5780	5.4358	1.8540	4.5970

Comparing the SNR measurements in Table 2 with the resolution measurements in Table 4 shows the combination of filter parameters that maintains the resolution of the input image also has a very high SNR gain. This is ideal in the sense that the greatest amount of speckle is removed while the resolution of the input image is maintained.

For the data presented in Chapter 5 the optimum filter parameters are Scaling Option 3, $\beta = 0.72$, and $L = 1$. When tested with a 2D Taylor PSF this combination has an increase in SNR of 10.1dB and virtually no change in the resolution.

5.5 Comparison of Filtered SAR Images

A SAR image and three filtered images are shown on the next page. This section subjectively compares the different filtered images with the speckled SAR image. The four images are 650×550 and are cropped from the initial 900×1280 SAR image. Refer to Appendix A for the full-scale 900×1280 images. Figure 18 is the F1_501422_217 image from Sandia National Laboratories. Figure 19 is the SAR image filtered with the 3×3 LPF. Figure 20 shows the image filtered with the 2DACE algorithm, Scaling Option = 3, Lag = 1, and $\beta = 0.72$. Figure 21 is the SAR image filtered with the 2DACE algorithm, Scaling Option = 2, Lag = 1, and $\beta = 0.30$.

In Figure 18 the speckle can clearly be seen, especially around the high intensity targets. The granularity pattern of the speckle can also be seen in the areas of flat contrast. In Figure 19 there are noticeable effects of the LPF. The high intensity objects are brighter and blurrier than the original image. The fine details are removed with the filtering process. The image in Figure 20 is the output of the 2DACE filter with the optimal parameter settings. The speckle is removed from the entire image. The high intensity objects are not blurred, the objects are more defined than in the original image, and the fine details in the areas of flat contrast are more pronounced. Figure 20 has much better resolution than the original speckled SAR image in Figure 18. Figure 21 shows that Scaling Option 2 reduces the speckle but the fine features are lost in filtering process. In addition the high intensity objects are brighter and the edges are not as sharp.

Figure 18. F1_501422_217

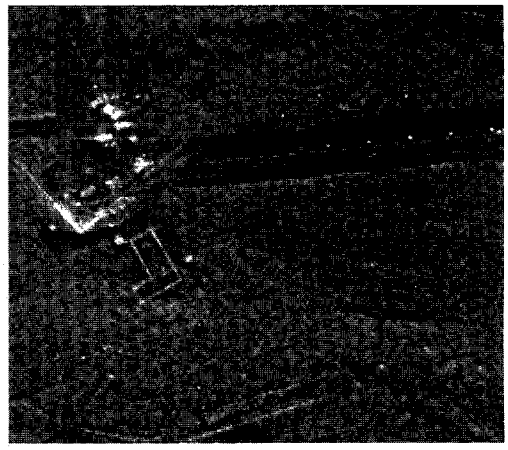


Figure 20. F1_501422_217 and Scaling Option 3

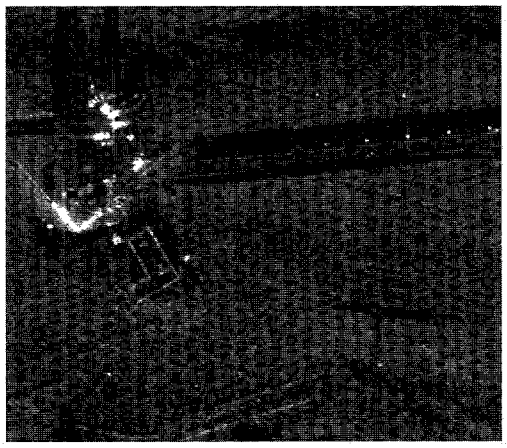


Figure 19. F1_501422_217 and 3x3 LPF

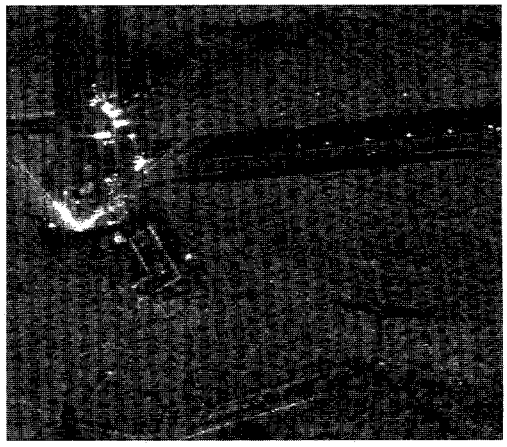


Figure 21. F1_501422_217 and Scaling Option 2

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

This research presented a unique implementation if the 2DACE algorithm involving nonlinear compression to manage the gain control problems inherent with the open-loop architecture. Data amplitude compression was applied prior to the 2DACE algorithm to control the gain and maintain the image resolution. In addition nonlinear data compression was applied after the 2DACE filtering to restore the contrast of the image.

2DACE was implemented in MATLAB and proven to effectively reduce speckle in SAR images without adversely affecting the resolution. The 2DACE filter algorithm was applied to 2D Gaussian and 2D Taylor PSF images to test the filter's performance. Finally SAR images were filtered with the 2DACE algorithm and compared with the speckled image and an image filtered with a benchmark 3×3 low pass filter.

Because the performance of open-loop adaptive algorithms is quite often application specific the parameters must be optimized for the task. The 2DACE algorithm has three parameters that affect the filtering process. The scaling option, filter lag, and adaptation constant work together to adaptively filter uncorrelated data from the image. The data presented in this report showed the effects of varying filter parameters and the optimal parameter combination was developed.

Three different scaling options in the filter update equation were incorporated in the MATLAB code. Scaling Option 1 and Scaling Option 2 are both scalars and have the same effect in the 2DACE algorithm. Scaling Option 3 includes a recursive signal power measurement. All tests were performed with Scaling Options 2 and 3.

The key factors in determining the performance of the filter were the gain in the signal-to-noise ratio and resolution measurements. The SNR was calculated with the variance of the clear, ideal image divided by the least square error calculation. As the error between the filtered image and the clear image decreased the SNR increased. Measurements of the -3dB and -14dB pulse widths of the Taylor Point Spread Function are documented to quantify the effects of the 2DACE filter and the parameters on the resolution.

A two-dimensional Gaussian image with additive white Gaussian noise was the initial test image used to determine the best normalizing and unnormalizing routine and demonstrate the effects of the scaling options. For these tests the filter lag was set to 1 and the adaptation constant was 0.5. The results of the tests show that the normalizing and unnormalizing routines adversely effect the image. The SNR decreased with the normalizing routines, likewise the least squares error increased. The SNR increased for both Scaling Option 2 and Scaling Option 3 when the normalizing or unnormalizing routines were removed from the filter implementation. Both scaling options had better SNRs than the benchmark 3×3 low pass filter.

A second series of tests was performed to determine the effects of varying the filter lag and the adaptation constant. The tests showed that a filter lag of 1 was the best selection in terms of maintaining image features and resolution. The amount of speckle filtered increases as the filter lag increases, as indicated by the SNR, but the image quality decreased in terms of the resolution. The reduction in resolution far outweighs the minor increase in SNR. In terms of the gain in SNR the tests also showed the best value for the adaptation constant, β, was different for each scaling option. For Scaling Option 2 the best β is 0.3. For Scaling Option 3 the best β was around 0.72. Both scaling options again outperform the benchmark 3×3 LPF. Scaling Option 3 had an SNR gain that was nearly double the SNR gain of Scaling Option 2.

A two-dimensional Taylor PSF with additive white Gaussian noise and SAR speckle was used to determine the 2DACE effects on point target resolution. The results show that for Scaling Option 2 the resolution decreased as β increased. Thus for Scaling Option 2the best value of β was 0.3. For Scaling Option 3 the resolution increased as β increased. Fine tuning measurements documented in Table 3 show the small changes in resolution as β was varied from 0.60 to 0.90. The resolution of the filtered image nearly equals the resolution of the Taylor PSF when β was 0.80. As β increased shadowing behind the horizontal dimension of the object increased. Therefore a compromise between resolution and shadowing was made with β = 0.72.

Test performed with a SAR point target show that the 2DACE filtering process with Scaling Option 3 did not affect the resolution in the horizontal or

vertical dimensions, as long as the image was compressed before the 2DACE filtering process. In addition, the shadowing was not produced around the point targets when the dynamic range was reduced with the pre-compression process. Tables 7, 9,10, and 11 include the resolution measurements of the filtered SAR point target made for various values of β and the pre-compression factor.

The results in this report show that there is latitude in selecting the scaling option and the adaptation constant. For Scaling Option 3 the adaptation constant β can be varied between 0.6 and 0.9, but the value of 0.72 is proven to be the best choice. For Scaling Option 2 β should by $\leq$ 0.3. Both scaling options produce results that improve the image quality.

The 2DACE filter with the optimum parameters was applied to SAR images obtained from Sandia National Laboratories. The filtered images showed that the 2DACE filter with Scaling Option 3, filter lag 1, and β = 0.72 removed the speckle, restored the fine features, and maintained the resolution of the point targets. The 2DACE filter with Scaling Option 2, filter lag 1, and β = 0.3 removed the speckle and only had a minor effect on the resolution. The two drawbacks of Scaling Option 2 are that the edges of large, high intensity objects were smoothed out and the very fine details in the low intensity areas were lost. Comparison of the images filtered with the 2DACE and the image filtered with the 3×3 LPF shows that the 2DACE has superior performance. It is easy to see this in a visual test of the images and all the results in this report point to this conclusion as well.

6.2 Future Work

Future work can be focused on automating the pre-compression routine. The data showed that the pre-compression factor c in the equation $y = x^c$ had a direct influence on the image resolution. Depending on the image the pre-compression factor generated unwanted black spots, discontinuities. By Adjusting c the pre-compression routine could be changed so the black spots are not generated. Therefore user involvement is required for obtaining the best results.

The future work will focus on automating the selection of the pre-compression factor. One method of automation would be to incorporate correlation mapping and image segmentation. This process would apply the pre-compression to independent areas of the image. The pre-compression factor would be selected based on the statistics of the selected area. This would result in optimum compression for maximum resolution in areas with features such as point targets and large objects. For areas without many features the pre-compression would adapt to a larger factor to reduce the probability of generating discontinuities in the image.

APPENDICES

Appendix A – SAR Images

The following eight pages are full scale SAR images. The images are 900 pixels by 1280 pixels. Figure 22 is the SAR image F1_501422_217 obtained from Sandia National Laboratories, Department 2345. Figure 23 is the filtered image with the optimal 2DACE parameters: pre-compression 0.10, Scaling Option 3, Lag = 1, $\beta = 0.72$. Figure 24 is the SAR image filtered with the 2DACE parameters Scaling Option 2, Lag = 1, $\beta = 0.30$. Figure 25 is the SAR image filtered with the benchmark 3×3 LPF. Refer to Section 5.5 for a comparison of these images. Figure 26 is the SAR image F2_335493_308 and Figure 27 is the image filtered with the following 2DACE parameters: pre-compression 0.25, Scaling Option 3, Lag = 1, $\beta = 0.72$. Figure 28 is the SAR image F1_171463_893 and Figure 29 is the image filtered with the following 2DACE parameters: pre-compression 0.33, Scaling Option 3, Lag = 1, $\beta = 0.72$. The pre-compression values used for processing Figures 26 and 28 were increased from 0.1 to reduce the discontinuities (black spots) generated from the pre-compression process.

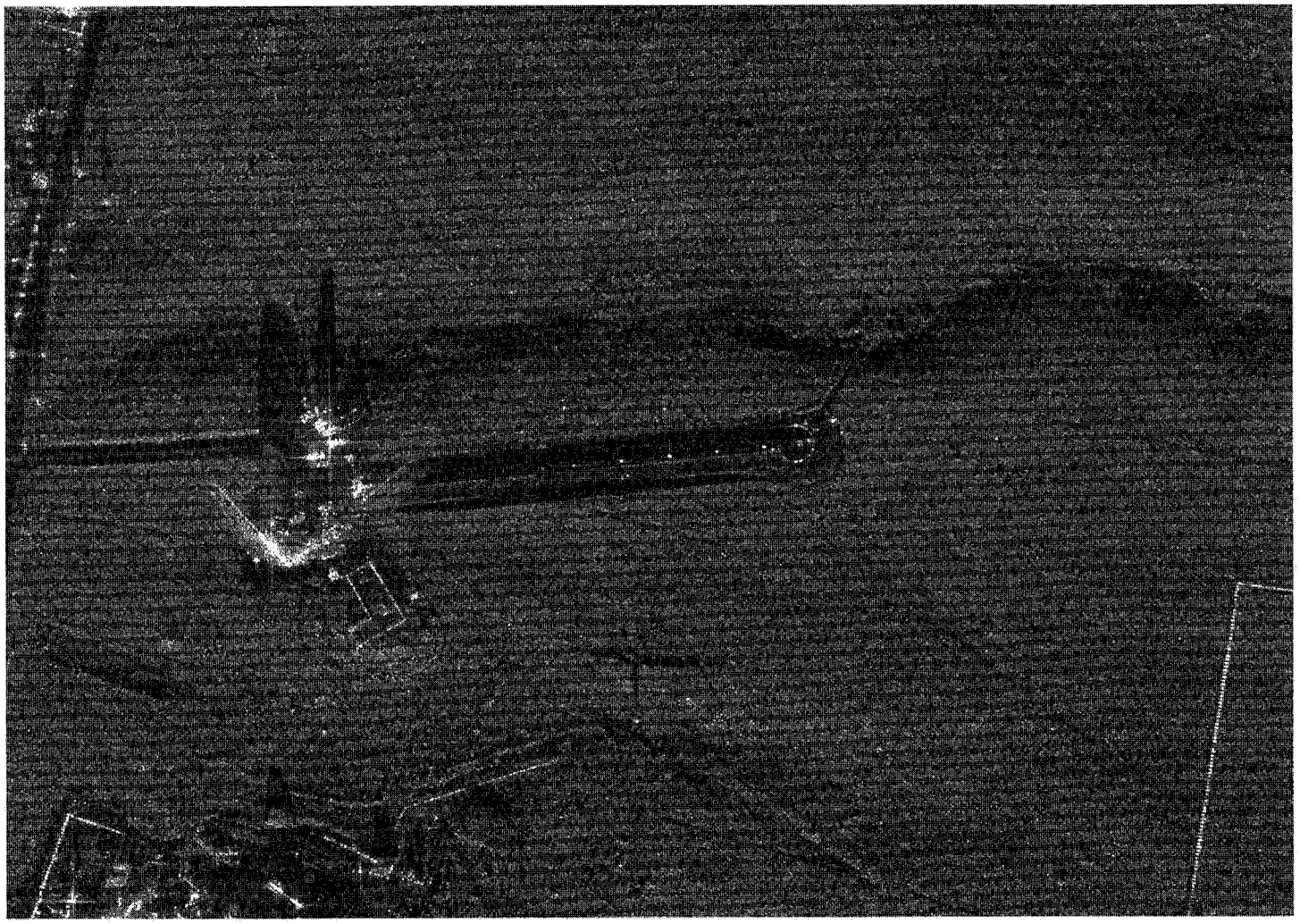


Figure 22 SAR Image F1 501422 217

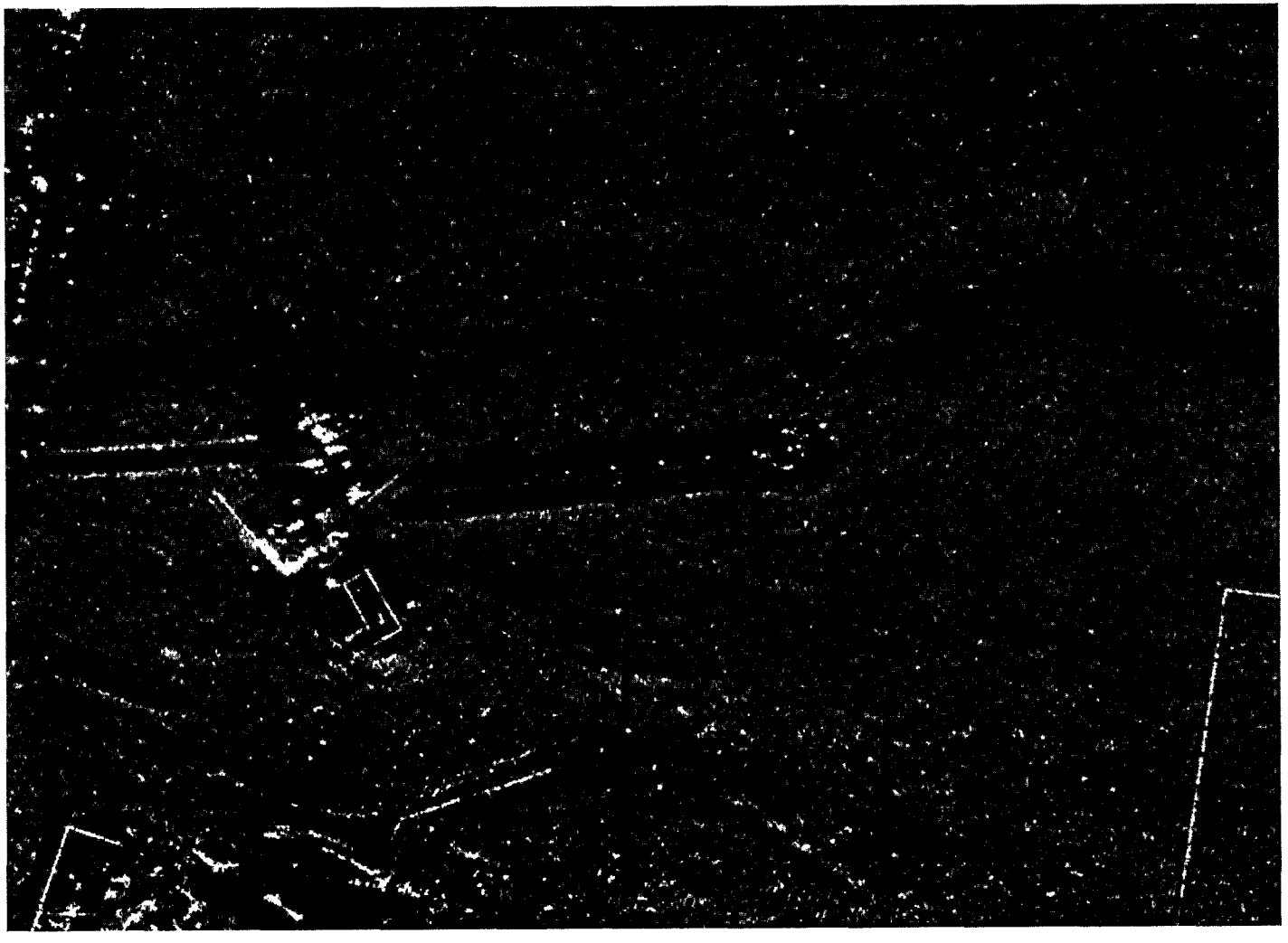


Figure 23 SAR Image F1 501422 217 2DACE Scaling Option 3, Lag = 1, Beta = 0.72 Pre Compress 0.10

Figure 24 SAR Image F1 501422 217 2DACE Scaling Option 2, Lag = 1, Beta = 0.30

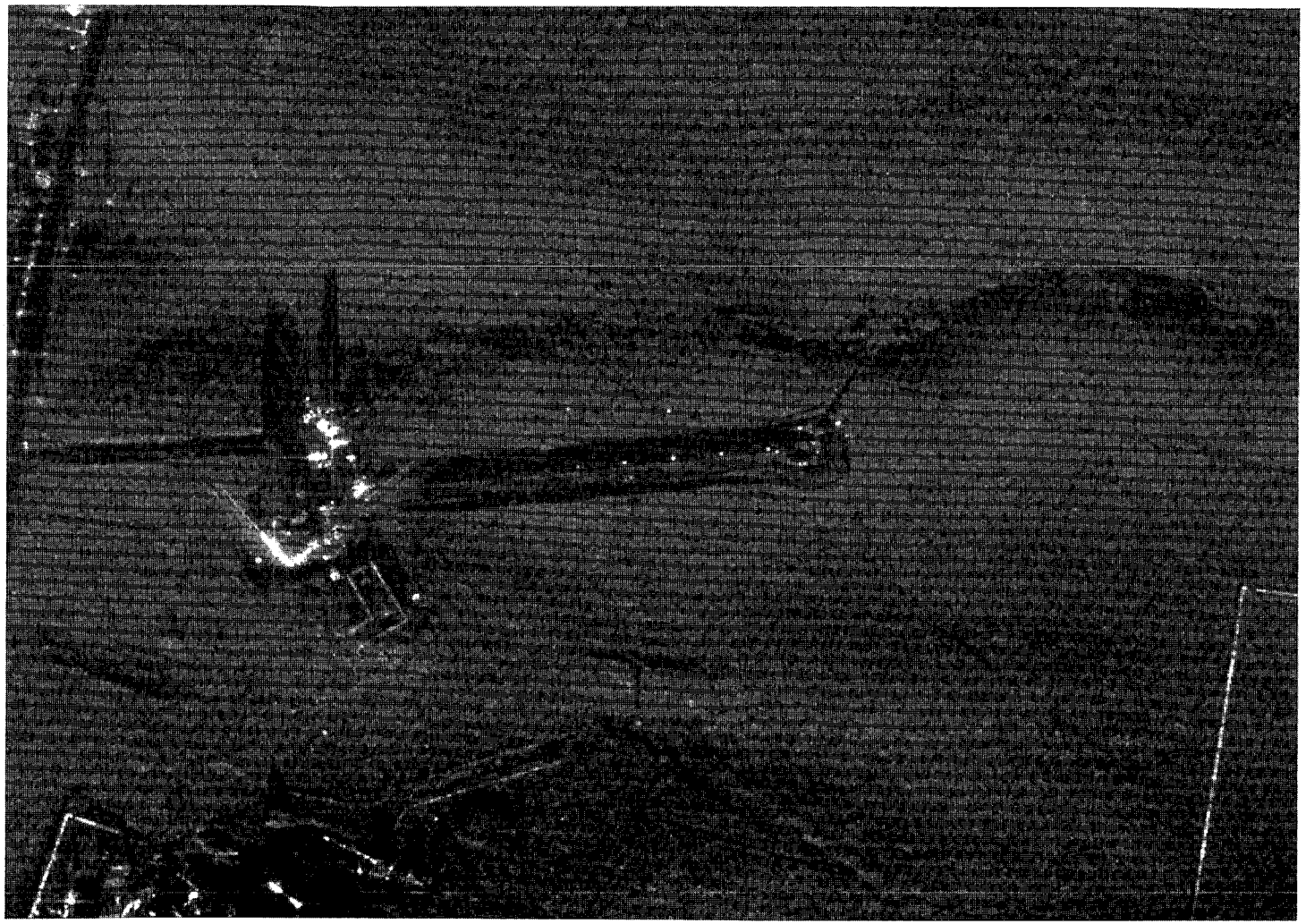


Figure 25 SAR Image F1 501422 217 3x3 LPF

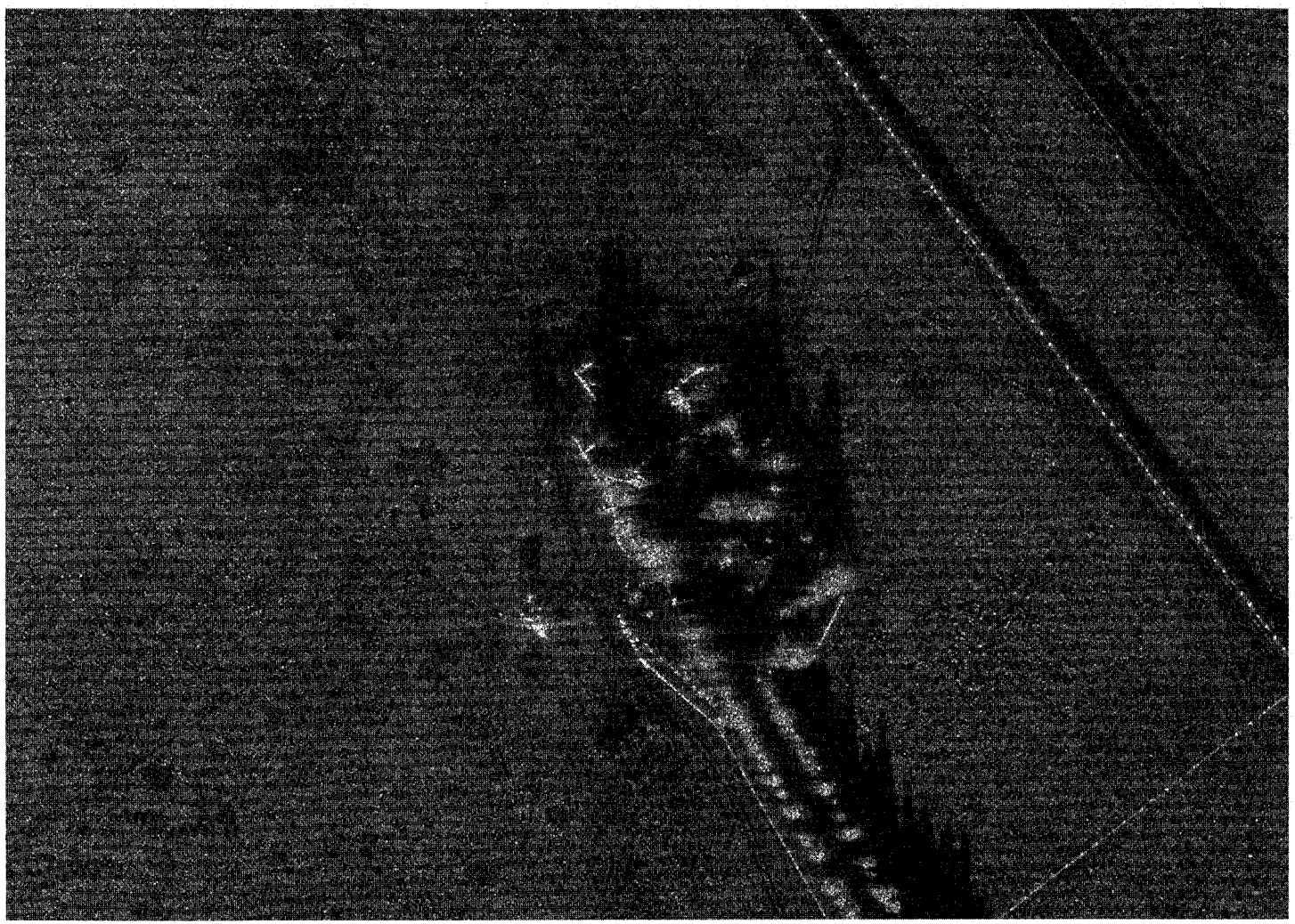


Figure 26 SAR Image F2 3335493 308

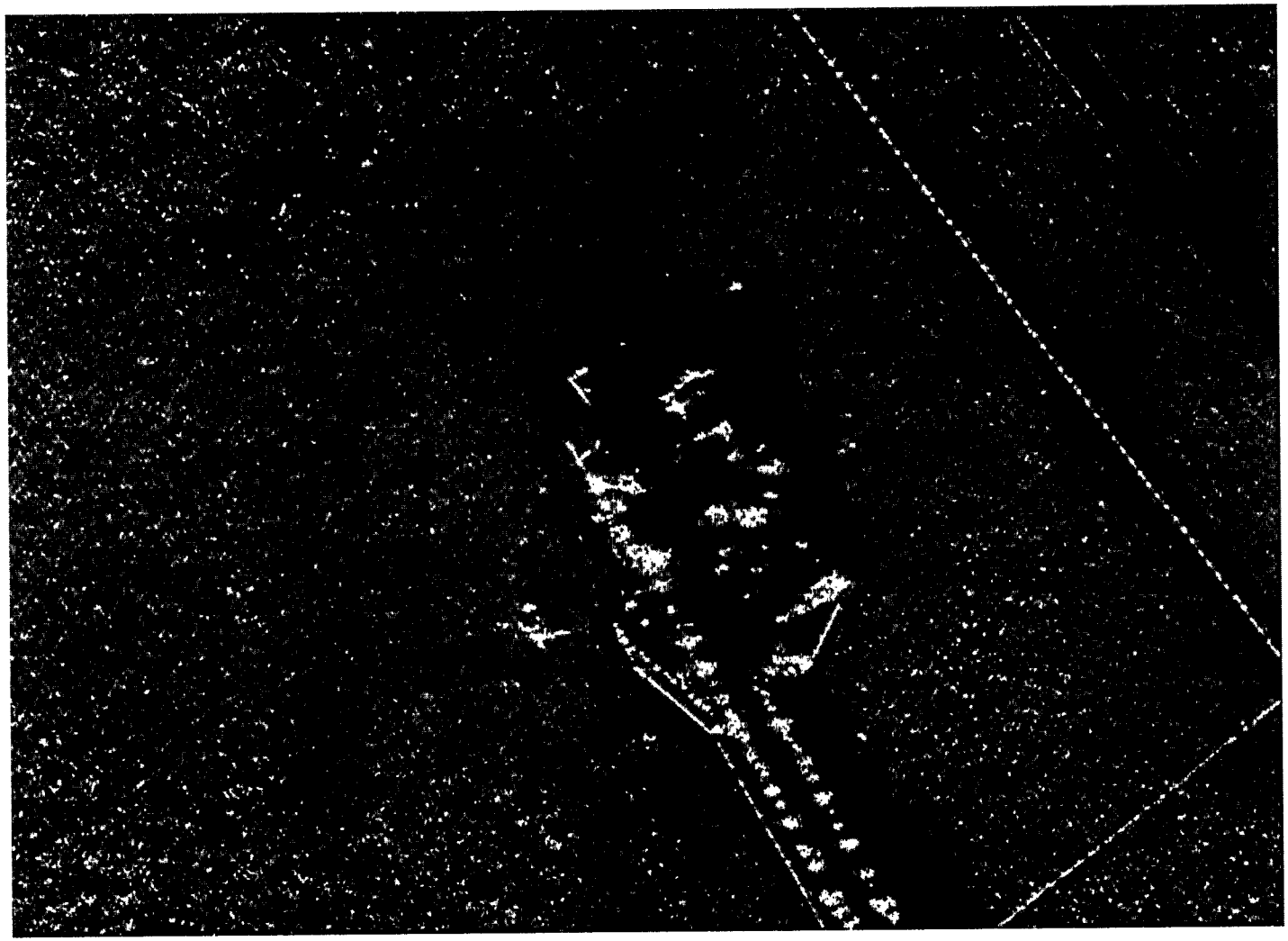


Figure 27 SAR Image F2 335493 308 2DACE Scaling Option 3, Lag = 1, Beta = 0.72 Pre Compress 0.25

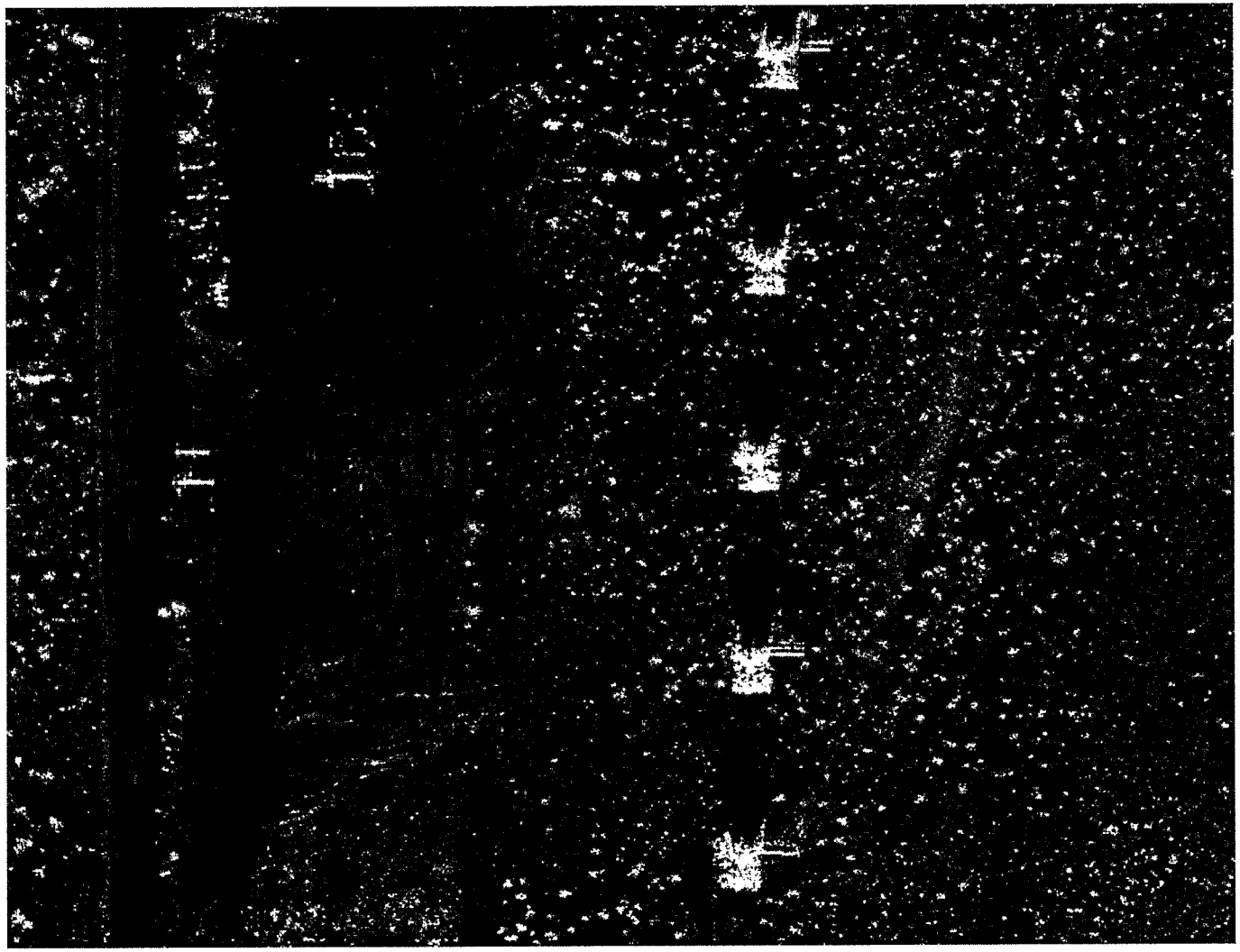


Figure 28 SAR Image F1 171463 893

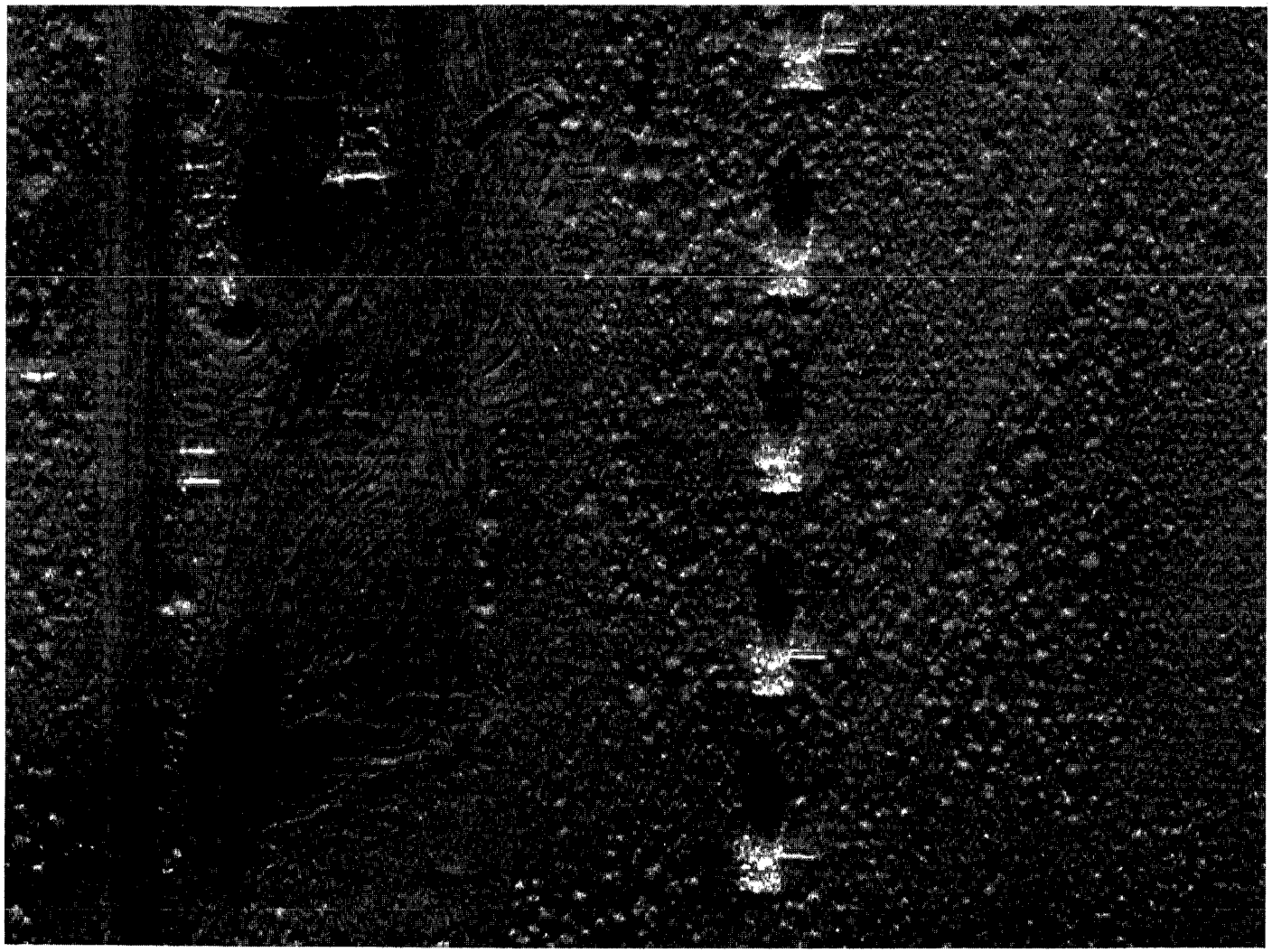


Figure 28 SAR Image F1 171463 893 2DACE Scaling Option 3, Lag = 1, Beta = 0.72 Pre Compress 0.33

Appendix B – Resolution Measurements

Tables 5, 6, and 7 contain the resolution measurement data for the 2D Taylor PSFs and the SAR point target. The -3dB and -14dB measurements are included for each combination of parameter setting. The means of the center row of the test images are included for reference. Tables 8 and 9 list the resolution measurement data used to find the optimal parameter settings.

Table 5 Resolution Measurements - Taylor PSF with WGN Noise

Filtering Options			Horizontal Resolution Calculations (units = pixels)			Vertical Resolution Calculations (units = pixels)		
			3dB Resolution	14dB Resolution	Mean	3dB Resolution	14dB Resolution	Mean
Clear image			3.4891	7.2331	0.1066	3.4891	7.2331	0.1066
Speckle image			4.1970	8.4216	0.1881	4.5863	7.6670	0.1548
Convolution with 3x3			3.6820	7.6517	0.1625	4.0077	8.8010	0.1665
Lag	**Beta**	**Scaling**						
1	0.30	2	4.1439	8.7049	0.1794	3.8978	9.3407	0.1597
1	0.50	2	4.2327	9.0146	0.1923	3.8073	9.3642	0.1607
1	0.75	2	4.4199	9.4829	0.2325	3.7234	9.4296	0.1686
1	0.95	2	4.7201	31.5649	0.3699	3.8171	9.9593	0.2091
2	0.30	2	4.7613	9.5757	0.2022	4.5538	10.5854	0.1828
2	0.50	2	4.9074	10.1360	0.2152	4.5040	10.6431	0.1846
2	0.75	2	5.1924	21.7958	0.2573	4.4601	10.8178	0.1937
2	0.95	2	5.5852	34.3476	0.4049	4.5931	11.7830	0.2371
1	0.30	3	3.7667	6.9457	0.0391	2.5475	8.8872	0.0516
1	0.50	3	3.5290	6.6624	0.0438	1.3160	8.0660	0.0633
1	0.75	3	3.0580	6.5124	0.0685	1.5961	7.3723	0.0938
1	0.95	3	3.1854	6.6089	0.1208	3.2739	7.4253	0.1123
2	0.30	3	2.1394	7.4380	0.0276	1.9708	9.8934	0.0387
2	0.50	3	3.2240	7.3494	0.0278	1.2895	9.0430	0.0446
2	0.75	3	3.4845	7.3361	0.0510	1.4881	8.3109	0.0810
2	0.95	3	3.5544	7.6024	0.1378	4.5086	8.5097	0.1422
3	0.3	2	5.5698	12.0407	0.2330	5.2449	11.9218	0.2056
3	0.3	3	1.8151	8.0418	0.0195	1.0856	11.3374	0.0287

Table 6 Resolution Measurements - Taylor PSF with Speckle

Filtering Options			Horizontal Resolution Calculations (units = pixels)			Vertical Resolution Calculations (units = pixels)		
			3dB Resolution	14dB Resolution	Mean	3dB Resolution	14dB Resolution	Mean
Clear image			3.4891	7.2331	0.1066	3.4891	7.2331	0.1066
Speckle image			4.0911	8.0243	0.1812	3.7141	7.9859	0.1797
Convolution with 3x3			4.1372	8.6717	0.2009	3.7889	8.7265	0.2107
Lag	**Beta**	**Scaling**						
1	0.30	2	4.4654	9.3991	0.2136	4.1817	8.5093	0.2143
1	0.50	2	4.5386	10.8460	0.2267	4.2150	8.7287	0.2174
1	0.75	2	4.7570	17.6212	0.2731	4.2854	9.4713	0.2343
1	0.95	2	5.1991	-	0.4244	4.5434	-	0.3001
2	0.30	2	5.0462	10.3513	0.2409	4.7332	10.0603	0.2438
2	0.50	2	5.1982	12.0245	0.2546	4.7746	10.2465	0.2473
2	0.75	2	5.5255	21.3603	0.3014	4.8634	11.5295	0.2646
2	0.95	2	6.1866	-	0.4611	5.2086	-	0.3367
1	0.30	3	4.2832	8.0440	0.1836	4.0755	9.5599	0.1989
1	0.50	3	3.7881	7.3634	0.1567	3.7703	8.8097	0.1761
1	0.75	3	3.4109	7.1352	0.1330	3.4356	7.6190	0.1511
1	0.95	3	3.3969	7.4311	0.1384	3.8165	7.5888	0.1613
2	0.30	3	3.3291	8.7854	0.1836	8.5324	11.9187	0.2056
2	0.50	3	3.5449	8.1256	0.1648	7.5317	11.0091	0.1949
2	0.75	3	3.6579	7.8403	0.1514	4.8742	9.4715	0.1857
2	0.95	3	3.7772	8.6830	0.1557	4.4832	9.0427	0.2002
3	0.3	2	3.6734	7.0171	0.1057	3.3351	6.6243	0.0909
3	0.3	3	2.6954	9.1062	0.1702	1.4289	13.0390	0.1976

Table 7 Resolution Measurements - SAR Point Target with Speckle

Filtering Options			Horizontal Resolution Calculations (units = pixels)			Vertical Resolution Calculations (units = pixels)		
			3dB Resolution	14dB Resolution	Mean	3dB Resolution	14dB Resolution	Mean
Speckle image			1.8776	4.5169	0.1290	1.7609	4.3047	0.1180
Convolution with 3x3			2.5805	6.7794	0.1793	2.5037	6.3342	0.1656
Lag	**Beta**	**Scaling**						
1	0.30	2	2.6376	8.4675	0.1858	2.4067	5.7127	0.1576
1	0.50	2	2.7415	10.3738	0.2038	2.4158	5.9161	0.1630
1	0.75	2	3.0827	19.5990	0.2568	2.4441	6.5362	0.1807
1	0.95	2	3.8210	-	0.4227	2.6039	18.9104	0.2511
2	0.30	2	3.1162	12.1286	0.2198	2.8661	8.4895	0.1920
2	0.50	2	3.2863	15.3025	0.2380	2.8928	8.9613	0.1993
2	0.75	2	3.6718	24.3536	0.2938	2.9688	12.4061	0.2213
2	0.95	2	4.7336	-	0.4675	3.3240	-	0.3048
1	0.30	3	1.2838	4.5011	0.0863	4.7095	6.7778	0.1081
1	0.50	3	1.5116	4.5209	0.0889	4.3175	6.1141	0.1185
1	0.75	3	2.0265	5.0329	0.0980	3.4258	5.7675	0.1296
1	0.95	3	2.2946	6.7559	0.1046	2.0877	5.0939	0.1175
2	0.30	3	1.8308	5.8470	0.0569	0.09001	7.1859	0.0841
2	0.50	3	2.0099	6.1999	0.0699	1.0186	7.2924	0.1131
2	0.75	3	2.3523	7.1100	0.0988	1.8771	7.5889	0.1586
2	0.95	3	3.3429	10.0175	0.1323	2.7264	7.4887	0.1798
3	0.3	2	3.4262	16.7145	0.2551	2.9755	10.6637	0.2263
3	0.3	3	2.2641	6.2233	0.0651	1.3538	4.0521	0.1017

Table 8 Filter Tuning Measurements - Taylor PSF with Speckle

Lag = 1, Scale = 3

	Horizontal Resolution Calculations (units = pixels)			Vertical Resolution Calculations (units = pixels)			
	3dB Resolution	14dB Resolution	Mean	3dB Resolution	14dB Resolution	Mean	LS SNR
Clear image	3.4891	7.2331	0.1066	3.4891	7.2331	0.1066	
Noisy image	4.0911	8.0243	0.1812	3.7141	7.9859	0.1797	
Convolution with 3x3	4.1372	8.6717	0.2009	3.7889	8.7265	0.2107	
Beta							
0.60	3.6976	7.2471	0.1471	3.5987	8.3744	0.1676	3.4270
0.63	3.6509	7.2105	0.1434	3.5531	8.2452	0.1637	3.7788
0.66	3.5912	7.1804	0.1401	3.5125	8.0981	0.1601	4.1072
0.67	3.5761	7.1718	0.1391	3.5003	8.0443	0.1590	4.2110
0.68	3.5616	7.1640	0.1381	3.4889	7.9876	0.1578	4.3120
0.69	3.5309	7.1570	0.1372	3.4784	7.9280	0.1568	4.4100
0.72	3.4577	7.1412	0.1348	3.4522	7.7441	0.1538	4.6874
0.75	3.4109	7.1352	0.1330	3.4356	7.6190	0.1511	4.9423
0.76	3.4009	7.1359	0.1326	3.4325	7.5804	0.1503	5.0233
0.77	3.3936	7.1383	0.1323	3.4306	7.5435	0.1496	5.1027
0.78	3.3890	7.1424	0.1321	3.4301	7.5084	0.1490	5.1809
0.79	3.3872	7.1486	0.1320	3.4309	7.4752	0.1483	5.2581
0.80	3.3880	7.1569	0.1319	3.4332	7.4438	0.1478	5.3345
0.81	3.3915	7.1676	0.1320	3.4369	7.4145	0.1473	5.4104
0.84	3.4182	7.2138	0.1327	3.4571	7.3402	0.1463	5.6325
0.87	3.4530	7.2744	0.1338	3.4945	7.2932	0.1462	5.8172
0.90	3.4825	7.3802	0.1349	3.5740	7.2973	0.1482	5.7961

Cross sections show that for beta = 0.72 the filter rebounds in about 8 samples and the resolution is about equal to the input image.

Table 9 Filter Tuning Measurements – SAR Point Target

Lag = 1, Scale = 3

	Horizontal Resolution Calculations (units = pixels)			Vertical Resolution Calculations (units = pixels)		
Beta	3dB Resolution	14dB Resolution	Mean	3dB Resolution	14dB Resolution	Mean
Speckle image	1.8776	4.5169	0.1290	1.7609	4.3047	0.1180
Convolution with 3x3	2.5805	6.7794	0.1793	2.5037	6.3342	0.1656
0.75	2.0265	5.0329	0.0980	3.4258	5.7675	0.1296
0.78	2.0996	5.1953	0.1009	3.3456	5.7251	0.1310
0.81	2.1776	5.3688	0.1048	3.2761	5.6783	0.1324
0.84	2.2623	5.5534	0.1075	3.2186	5.6297	0.1308
0.85	2.2925	5.6176	0.1088	3.2035	5.6143	0.1304
0.86	2.3089	5.6693	0.1103	3.1918	5.6002	0.1304
0.87	2.3094	5.7070	0.1122	3.1852	5.5885	0.1306
0.88	2.3097	5.7450	0.1133	3.1596	5.5700	0.1301
0.89	2.3096	5.7829	0.1138	3.1172	5.5456	0.1290
0.90	2.3090	5.8477	0.1139	3.0635	5.5176	0.1278
0.92	2.3058	6.1472	0.1120	2.8913	5.4399	0.1244
0.95	2.2946	6.7559	0.1046	2.0877	5.0939	0.1175

Table 10 Filter Tuning Measurements – Pre-Compression Results

Lag = 1, Scale = 3, compression = 0.25

	Horizontal Resolution Calculations (units = pixels)				Vertical Resolution Calculations (units = pixels)			
	3dB	14dB	3dB c	14dB c	3dB	14dB	3dB c	14dB c
Speckle image	1.8776	4.5169			1.7609	4.3047		
Convolution with 3x3	2.5805	6.7794			2.5037	6.3342		
Beta								
0.50	2.4069	5.7267	2.7216	8.5065	1.8336	4.2131	1.9958	5.0143
0.66	2.2504	5.3400	2.4982	6.3733	1.6919	4.0491	1.9273	5.3094
0.69	2.2363	5.2481	2.4776	6.1903	1.6707	4.1611	1.9438	5.4220
0.72	2.2239	5.1515	2.4598	6.0304	1.6713	4.3162	1.9728	5.5546
0.75	2.2089	5.0536	2.4426	5.8813	1.6922	4.4957	2.0154	5.7079
0.78	2.1841	4.9588	2.4229	5.6561	1.7310	4.6987	2.0592	5.9250
0.81	2.1498	4.8719	2.3965	5.6789	1.7855	4.9250	2.1066	6.3058
0.84	2.1004	4.7967	2.3582	5.6105	1.8534	5.1755	2.1611	6.7322
0.87	2.0314	4.7487	2.3019	5.5624	1.9164	5.5374	2.2223	7.2881
0.90	1.9473	4.7197	2.2089	5.5631	1.9859	6.1500	2.2888	7.9495
0.93	1.8863	4.8317	2.1202	5.8726	2.0664	6.9135	2.3574	8.9777

Table 11 Filter Tuning Measurements – Pre-Compression Variable Tests

Lag = 1, Scale = 3, Beta = 0.72

	Horizontal Resolution Calculations (units = pixels)				Vertical Resolution Calculations (units = pixels)			
	3dB	14dB	3dB c	14dB c	3dB	14dB	3dB c	14dB c
Speckle image	1.8776	4.5169			1.7609	4.3047		
Convolution with 3x3	2.5805	6.7794			2.5037	6.3342		
Pre-Compression Factor								
0.10	1.8364	4.7931	2.1150	5.7476	1.5536	4.2378	1.8584	5.3848
0.15	1.9486	4.9166	2.2375	5.8017	1.5771	4.2395	1.8880	5.4380
0.20	2.0787	5.0388	2.3488	5.8676	1.6161	4.2491	1.9370	5.4978
0.25	2.2239	5.1515	2.4598	6.0304	1.6713	4.3162	1.9728	5.5546
0.30	2.3456	5.2588	2.5715	6.1971	1.7485	4.4217	2.0126	5.6154
0.35	2.4626	5.3504	2.6806	6.3346	1.8033	4.5097	2.0518	5.6644
0.40	2.5780	5.4358	2.7904	6.4716	1.8540	4.5970	2.0938	5.7161

Appendix C – 2DACE Filter Algorithm - MATLAB Code

```
%Filter_2DACE.m

program_name = 'Filter_2DACE';

%The input image is transposed before filtering
%then transposed back after the filtering process.

%JAR 5/23/00

%Main routine for the Adaptive Correlation Enhancer Algorithm.
%The Scaling Option is 3, beta is 0.72 and the lag is 1.
%Prior to filtering the image is compressed.
%After filtering the image is uncompressed.

%The output consists of two images and a structure of filter options.
%The image filtered_image is the image filtered with 2DACE.
%The image "filtered_image_c is the filtered image compressed with y =
x^(0.8),
%which improves the contrast on some images.

%The initial image is padded around the borders by reflections of the
pixels.
%The number of padded columns and rows is 10.

image = input('Enter the name of the image    ');
input_name = input('Enter the name of the image... again   ', 's');

%start clock

t0 = clock;

%make sure that image is a double array

image = double(image);

image = image';

%normalize the image to 1

max_im = max(max(image));

image_norm = image/max_im;

%set compression constant

compress_constant = 0.15;
uncompress_constant = 1/compress_constant;

%compress image

image = image_norm.^(compress_constant);

max_im = max(max(image));

image_norm = image/max_im;
```

```
image = image_norm;

dimension_i = size(image);
rows_i = dimension_i(1,1);
columns_i = dimension_i(1,2);

pad = 10;
lag = 1;

filt_size = 2*lag+1
%filt_size = 1

scale = 3;
beta = 0.72;

%pad image with reflected data

[pad_image] = data_pad(image,pad);

%calculate dimensions of padded image

dimension = size(pad_image);
rows = dimension(1,1);
columns = dimension(1,2);

%weighted average adaptive correlation enhancer function for scale = 2

[filt_im] =
filter_auto(pad_image,rows,columns,filt_size,scale,beta,lag);

%crop images to original size

crop_c = columns_i-1;
crop_r = rows_i-1;
crop_s = pad + 1;

filt_im = imcrop(filt_im,[crop_s crop_s crop_c crop_r]);

filt_im = filt_im';

%uncompress image

max_im = max(max(filt_im));

filt_im_norm = filt_im/max_im;


filtered_image = filt_im_norm.^(uncompress_constant);

%adjust image to max level of 256.

max_pixel = max(max(filtered_image));
norm_filt_im = filtered_image/max_pixel;

filtered_image = norm_filt_im*256;

%adjust constrast
```

```
max_im = max(max(filtered_image));

filt_im_norm = filtered_image/max_im;

filtered_image_c = filt_im_norm.^(0.8);

max_pixel = max(max(filtered_image_c));
norm_filt_im = filtered_image_c/max_pixel;

filtered_image_c = norm_filt_im*256;


%calculate elapsed time and date

cpu_time = etime(clock,t0)/60;

date_today = date;

%structure of filter options

filter_options =
struct('program',program_name,'date',date_today,'input_image_name',...

input_name,'computational_time',cpu_time,'pad_length',pad,'lag',lag,'sc
ale',scale,'beta',beta,...
   'compression_constant',compress_constant);


clear program_name date_today input_name cpu_time pad lag beta scale
image_norm
clear max_image min_image dimension_i rows_i columns_i dimension rows
columns
clear filt_im t0 crop_c crop_r crop_s pad_image image filt_size
max_pixel
clear norm_filt_im max_im
```

```
%filter_auto.m

%Adaptive Correlation Enhancer function
%
%This filter runs left to right and then repeats on incremented row.
%Used with Filter_2DACE

function [corr_image] =
filter_auto(norm1_image,rows,columns,filt_size,scale,beta,lag)

variance = 0;
autocorrelation = 0;
sum = 0;
w = 0;
row_pick = 0;
start = 1;

%Zero the weight matrix

for k = 1:filt_size,

   for l = 1:filt_size,

   w(k,l) = 0;

   end
end

%Beginning of the filtering loop
%Compute the autocorrelation of the weight matrix

%Start by finding the weight matrix average

for m = 1:rows,

   row_pick = xor(start,row_pick);

   if row_pick == 1
      nstart = 1;
      cnt = 1;
      nstop = columns;
   else
      nstart = 1;
      cnt = 1;
      nstop = columns;
   end

   for n = nstart:cnt:nstop,

      corr_image(m,n) = 0;

      idwri = m-lag;

      for k = 1:filt_size,

         idwci = n-lag;

         for l = 1:filt_size,
```

```
            if ((idwri < 1)|(idwri > rows)|(idwci < 1)|(idwci >
columns));

            else

               update_data = norm1_image(idwri,idwci);

               corr_image(m,n) = corr_image(m,n) + w(k,l)*update_data;
            end

         idwci = idwci + 1;

         end

      idwri = idwri + 1;

      end


      switch scale

         case 1,
            cf = 1.0 - beta;
         case 2,
            cf = ((1.0 - beta)/(2*lag*lag));
         otherwise,

            variance = beta*variance + ((1.0 -
beta)*norm1_image(m,n)*norm1_image(m,n));

            if variance ~= 0

               cf = (1.0 - beta)/(2.0*lag*lag*variance);

            else

               cf = (1.0 - beta)/(2.0*lag*lag);

            end
         end

      %Update the weight matrix

      idwri = m-lag;

      for k = 1:filt_size,

         idwci = n-lag;

         for l = 1:filt_size,

%update_data is used to check for out of image bounds.  If the point is
outside
%the input image a value of zero is assigned to that point

            if ((idwri < 1)|(idwri > rows)|(idwci < 1)|(idwci >
columns))
```

```
                update_data = 0;
            else
                update_data = norm1_image(idwri,idwci);
            end
            w(k,l) = ((beta*w(k,l))+(cf*update_data*norm1_image(m,n)));
            idwci = idwci + 1;
        end
    idwri = idwri + 1;
    end
  end
end
```

Appendix D – Data Padding M File

```
%data_pad.m


%this function pads the image with reflected data around the border.
%the number of rows and columns of reflected data is equal to the
%variable 'lag'.

function [pad_image] = data_pad(image,lag)

dimension = size(image);

rows = dimension(1,1);
columns = dimension(1,2);

N = lag+columns;

column_I = eye(N);

row_matrix1 = imcrop(column_I, [lag+1 1 N N]);

pad1_t = row_matrix1*image';

pad1 = pad1_t';

N = lag+rows;

row_I = eye(N);

row_matrix2 = imcrop(row_I, [lag+1 1 N N]);

pad2 = row_matrix2*pad1;

dimension2 = size(pad2);

rows1 = dimension2(1,1);
columns1 = dimension2(1,2);

N = lag+rows1;

row_I = eye(N);

row_matrix3 = imcrop(row_I, [1 1 N-(lag+1) N]);

pad3 = row_matrix3*pad2;

N = lag+columns1;

column_I2 = eye(N);

row_matrix1 = imcrop(column_I2, [1 1 N-(lag+1) N]);

pad4_t = row_matrix1*pad3';

pad_image = pad4_t';
```

```
dimension = size(pad_image);

rows = dimension(1,1);
columns = dimension(1,2);

for m = 1:lag,

   for n = 1:columns,

      pad_image(m,n) = pad_image(2*lag-m+1,n);

   end
end

k = -1;

for m = rows-lag+1:rows,

   k = k+2;

   for n = 1:columns,

      pad_image(m,n) = pad_image(m-k,n);

   end
end

for n = 1:lag,

   for m = 1:rows,

      pad_image(m,n) = pad_image(m,2*lag-n+1);

   end
end

k = -1;

for n = columns-lag+1:columns,

   k = k+2;

   for m = 1:rows,

      pad_image(m,n) = pad_image(m,n-k);

   end
end
```

Appendix E – Two-Dimensional Taylor PSF M File

```
%taylor_psf.m

%this function creats a 2D Taylor PSF.

%image = input('enter the name of the image    ');

N = input('enter the length of the Taylor Window    ');

lag = input('enter the lag    ');

%function [pad2] =  vector_pad(image,lag)

wi=taylor(-35,4,N);

dimension = size(wi);

rows = dimension(1,1);
columns = dimension(1,2);

%zero pad the vector

vector_l = columns + (2*lag);

pad_v = zeros(1,vector_l);

for x = 1:vector_l,

   if x <= lag

      pad_v(x) = 0;

   elseif x > columns + lag

      pad_v(x) = 0;

   else

      pad_v(x) = wi(x-lag);

   end

end

%take fft and abs

t_psf = abs(fft(pad_v));

points = length(t_psf);

i = 1:points;

for x = points-5:points,
```

```
   t_psf(x) = 0;

end

%reflect Taylor FFT to generate PSF

t_psf2 = zeros(1,2*points);

for i = 1:points,

   t_psf2(i) = t_psf(points-(i-1));

end
j = points;
for i = 2*points-1:-1:points,

   t_psf2(i) = t_psf(j);
   j = j-1;

end

%Create 2D Taylor PSF

taylor_matrix = t_psf2'*t_psf2;

%Scale 2D Taylor PSF to 256

taylor_max = max(max(taylor_matrix));

taylor_256 = (taylor_matrix/taylor_max)*256;

%crop image to 50 x 50

dimension_256 = size(taylor_256);
rows_256 = dimension_256(1,1);
columns_256 = dimension_256(1,2);

center_r = rows_256/2;
center_c = columns_256/2;

first_r = center_r - 24;
first_c = center_c - 24;

taylor_test = imcrop(taylor_256,[first_r first_c 49 49]);



i = 1:points;

figure(1),
subplot(211);
stem(pad_v);
title('Taylor Window length = 50 Zero Pad = 50 each side');
subplot(212);
stem(t_psf);
title('Taylor PSF');

figure(2),
```

```
stem(t_psf2);
title('Taylor PSF Reflected');

figure(3),
imagesc(taylor_256);
title('Taylor PSF 2D Matrix');

figure(4),
imagesc(taylor_test);
%mesh(taylor_test);
title('Taylor PSF 2D Matrix 50 x 50');

dimension = size(taylor_256);
rows = dimension(1,1);
columns = dimension(1,2);

index = floor(rows/2);

center_row = taylor_256(index,:);

figure(5)
stem(center_row);
title('center row of Taylor PSF 2D Matrix');
```

References

1. J.J. Kovaly, Synthetic Aperture Radar, Artech House, Inc., Dedham, MA, 1976.
2. W.G. Carrara, R.S. Goodman, and R.M. Majewski, Spotlight Synthetic Aperture Radar, Signal Processing Algorithms, Artech House, Inc., Norwood, MA, 1995.
3. C.V Jakowatz Jr, D.E., Wahl, P.E., Eichel, D.C., Ghiglia, and P.A., Thompson, Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach, Kluwer Academic Publishers, Boston/London/Dordrecht, 1996.
4. S.A. Hovanessian, Introduction to Synthetic Array and Imaging Radars, Artech House, Inc. Dedham, MA, 1980.
5. A.W. Doerry, "Lynx SAR: Issues, Analysis, and Design Part 1", *Sandia Report,* SAND97-2383/1, October 1997.
6. J.W. Goodman, "Statistical Properties of Laser Speckle Patterns," Topics in Applied Physics: Laser Speckle and Related Phenomena, ed. J.C. Dainty, Springer-Verlay, 1984.
7. T. IWAI and T. Asakura, "Speckle Reduction in Coherent Information Processing," *Proceedings of the IEEE,* Vol. 84, No. 5, 756-781, May 1996.
8. V.S. Frost, J.A Stiles, K.S. Shanmugan, and J.C. Holtzman, "A Modal for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAMI-4, No 2, 157-166, March 1982.

9. J.S. Lee, "Digital Image Enhancement and Noise Filtering by Use of Local Statistics," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAMI-2, No 2, 165-168, March 1980.

10. M.H. Hadhoud and D.W. Thomas, "The Two-Dimensional Adaptive LMS (TDLMS) Algorithm," *IEEE Transactions on Circuits and Systems,* Vol. 35, No. 5, 485-494, May 1988.

11. E. Hervet, R. Fjortoft, P. Marthon, and A. Lopes, "Comparison of Wavelet-based and Statistical Speckle Filters," *SPIE,* Vol. 3497, 43-54, September 1998.

12. D. Blacknell, C.J. Oliver, and M. Warner, "Speckle Reduction of SAR Images Using Neural Networks," *SPIE,* Vol. 2584, 179-187, 1995.

13. C. Knittle and N. Magotra, "Two-Dimensional Adaptive Enhancement and Detection of Linear Features in Noisy Images," *25th Asilomar Conference,* November 1991.

14. S.R. Saunders, Antennas and Propagation for Wireless Communication Systems, John Wiley & Sons, Ltd., New York, NY, 1999.

15. J.W. Goodman, "A Random Walk Through the Field of Speckle," *Optical Engineering,* Vol. 25, No. 5, 610-612, May 1986.

16. J.W. Goodman, "Some Fundamental Properties of Speckle," *J. Opt. Soc. Am.,* Vol. 66, No. 11, 1145-1150, November 1976.

17. J.S. Lee, "Speckle Suppression and Analysis for Synthetic Aperture Radar Images," *Optical Engineering,* Vol. 25, No. 5, 636-643, May 1986.

18. Y.H. Lu, S.Y. Tan, T.S, Yeo, W.E, Ng, I. Lim, and C.B. Zhang, "Adaptive Filtering Algorithms for SAR Speckle Resolution," *IEEE,* 67-69, 1996.

19. H.H. Arsenault and G. April, "Properties of Speckle Integrated with a Finite Aperture and Logarithmically Transformed," *J. Opt. Soc. Am.*, Vol. 66, No. 11, 1160-1163, November 1976.

20. M. Tur, K.C. Chin, J.W. Goodman, "When is Speckle Noise Multiplicative?," *Applied Optics,* Vol. 21, No. 7, 1157-1159, April 1982.

21. D.T. Kuan, A.A. Sawchuk, T.C. Strand, and P. Chavel, "Adaptive Restoration of Images with Speckle," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* Vol. ASSP-35, No. 3. 373-383, March 1987.

22. N. Magotra, N. Ahmed, and E Chael, "Single Station (thee component) seismic data enhancement," *20th IEEE Asilomar Conference on Circuits, Systems, and Computers,* 409-413, November 1986.

23. C. Knittle and N. Magotra, "Analysis of the Adaptive Correlation Enhancer," *Digital Signal Processing 2*, 88-97, 1992.

24. C. Knittle and N. Magotra, "Image Enhancement Using the Adaptive Correlation Enhancer," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing,* Vol. 39, No. 12, 862-865, December 1992.

25. A.K. Jain, Fundamentals of Digital Image Processing, Prentice Hall, Englewood Cliffs, NJ, 1989.

Distribution

1	MS0529	Bruce C. Walker, 2308
1	MS0537	Bobby G. Rush, 2314
1	MS0537	Robert M. Axline, 2344
1	MS0537	Douglas L. Bickel, 2344
1	MS0537	Thomas J. Cordaro, 2344
1	MS0529	Armin W. Doerry, 2345
1	MS0519	Brett L. Remund, 2348
1	MS0507	Carolyne M. Hart, 2600
1	MS0986	David E. Ryerson, 2660
1	MS0986	Daniel E. Gallegos, 2664
1	MS0986	Stephen C. Roehrig, 2664
1	MS0986	Michael E. Partridge, 2665
10	MS0986	Judd A. Rohwer, 2665
1	MS0986	Tedd A. Rohwer, 2665
1	MS9018	Central Technical Files, 8940-2
2	MS0899	Technical Library, 9616
1	MS0612	Review & Approval Desk, 9612 for DOE/OSTI