EXPLOITING HIGHER-ORDER STATISTICS IN RADAR DATA FOR SCATTERING CENTER ISOLATION

by

JEFFREY BRANDON HALL

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2016

Acknowledgements

Firstly, I want to thank God for sustaining me through the trials of graduate studies.

I want to extend special gratitude to my wife, Eryn, and children, Haile and Joshua. Without their patience and encouragement this work would not have been possible. I would like to dedicate this dissertation to my wonderful and patient family.

I also need to thank all of my colleagues at the Raytheon company for their support and encouragement during my PhD studies. In particular, I wish to express thanks to Dave Fittz, Chris Korman and Tony Kurth for their continual confidence. I am grateful for Raytheon's financial support for this work which was provided through the Education Assistance Program and the Advanced Study Program.

I am also very grateful to my supervising professor, Dr. Saibun Tjuatja. His knowledge and insight was crucial to my success in this endeavor. I appreciate his patience and encouragement throughout my Doctoral career.

I would also like to thank my dissertation committee for their time and guidance in the development of this work.

Abstract

EXPLOITING HIGHER-ORDER STATISTICS IN

RADAR DATA FOR

SCATTERING CENTER ISOLATION

Jeffrey Brandon Hall, PhD

The University of Texas at Arlington, 2016

Supervising Professor: Saibun Tjuatja

Embedded in each observation of a RADAR system is an amalgam of scattered signals. An important function of RADAR signal processing is to resolve or separate this mixture of target signals.

Imaging using RADAR data is a well-established area of research. Conventional RADAR imaging methods use a two-dimensional Fourier transform to back-project RADAR measurements to scattering sources [1]. The Fourier transform based imaging techniques have resolution limited by the bandwidth and spatial diversity of available data samples.

Super-resolution methods such as MUSIC overcome the resolution limitation by employing an alternate model for the measured data [2]. These techniques are capable of enhancing RADAR images through an increase in the resolvability of scattering produced at particular spatial locations.

This research focuses on enhancing RADAR imaging techniques through a concept of isolating and localizing scatterers from within a mixture. The isolation step is accomplished through the notion of separating signals based on their non-Gaussianity.

Finite sized radar targets have non-Gaussian probability densities which enables the use of this measure in the distinction of scatterers [3]. The second attractive feature of non-Gaussianity as a measure of distinction is that a mixture of sources with non-Gaussian probabilities tends to Gaussian as the number of independent sources increases.

The approach of this manuscript uses a kurtosis maximization algorithm to search for the most non-Gaussian components of a mixture. Once these elements are isolated, an attempt is made to generate an accurate localization of the individual scatterers through a composite maximum view of the components.

This research employs simulated and measured RADAR data. The measured RADAR data was captured on the turntable ISAR system at the Wave Scattering Research center of the University of Texas at Arlington.

Results show that there is a strong correlation between leptokurtic scattering data and the unique scattering centers in the radar field of view. The results of the testing, on synthetic target sets, yielded a 52% probability of detection of the known scatterers with a localization error of zero in most cases.

List of Illustrations

List of Tables

Chapter 1 Introduction

RADAR sensing is becoming ubiquitous in modern society. From monitoring weather to vehicle collision avoidance, and robot navigation to monitoring climate change, applications of radar remote sensing are expanding.

RADAR signal processing is a rich field of research. Scatter detection and interpretation has been an overarching theme of the research in this field for decades. It is the intent of this work to expand the domain of knowledge relative to information extraction from RADAR scatter.

Radar scattering localization is a fundamental area of interest in the field of remote sensing. Back-projection RADAR imaging is a common method for separating and analyzing scattering within radar data. The back-projection technique has an inherent irresolution due to Fourier basis which is used to process the data. The resolution of Fourier based techniques is limited by the bandwidth of the sampling system. Improvement in scattering localization is realized through super-resolution techniques, which transcend the diffraction limited Fourier based signal processing. The research of this dissertation seeks to test a concept of enhancing the resolution of radar imaging techniques through the use of statistical signal processing.

Research Goals

The key goal in this research is to test the hypothesis that the most non-Gaussian elements of a scattering mixture are correlated to the scattering centers in the observation field.

A secondary goal of this work is the development of an application to exploit the potential correlation with an objective to produce a method of RADAR imaging to detect and localize scattering centers within the radar data.

High Kurtosis and Scattering Center Correlation

The hypothesis that the scattering centers with a RADAR observation are correlated to the most non-Gaussian elements of the scattered data is rooted in the concept that a mixture of sources is more Gaussian than the individual sources making up the mixture. The fundamental basis for this concept is the Central Limit Theorem.

Kurtosis is a fourth-order statistical cumulant which provides a measure of Gaussianity. Normalized kurtosis has been employed as a cost function in several well-known blind source separation (BSS) techniques. This class of techniques is typically referred to as Independent Component Analysis (ICA) [4] [5] [6]. A well-known algorithm based on this class of techniques, robustICA, will be used to perform estimation of the highly kurtotic elements within the scattering mixture. This work presents a novel application of this signal processing technique as it relates to the physics underlying radar scattering data

Method to Exploit Scattering Kurtosis in RADAR Imaging

The novel method developed for this research uses the fourth-order statistics of the radar data to adapt beamforming weights such that the scattering elements which possess the highest non-Gaussianity are emphasized. This work develops an innovative application of the well-known Independent Component Analysis algorithm in the estimation of the optimal beamforming weights for the separation of high kurtosis elements of the RADAR scattering data.

The results shown in this work establish that there is a strong correlation between leptokurtic scattering data and the unique scattering centers in the radar observation. This manuscript describes the basis and technique of a novel method for

extraction of the leptokurtic elements in RADAR scattering data. This work also explores a RADAR imaging technique which exploits the relationship between the scattering centers and the leptokurtic elements within the radar data.

Prior Work

There is ongoing research in advanced RADAR imaging techniques and super-resolution imaging. In the following sections some of the recent contributions will be outlined.

State of the Art in RADAR Imaging

In a 2014 article, Ash et al. presented algorithms and models for anisotropic scattering. This research was aimed at removing sample space limitations imposed by the point scattering model. This new perspective allows for the estimation of scattering using up to a full circular aperture in k-space [7].

In their 2014 article, Çetin et al. highlight the prospect for sparsity based techniques in enhancing the capability of SAR systems. Application of the sparse model to SAR signal processing has led to a host of advanced image formation algorithms [8]

Multiple Input Multiple Output (MIMO) SAR is an emerging research area which shows promise to enhance the performance of SAR systems [9]. The MIMO concept expands the capability of digital beamforming on receive to include a set of beamformers which are matched to a set of unique separable transmit waveforms. Cristallini et al. published an article in 2011 which theorized a super-resolution affect, in the range dimension of a SAR image, through the use of a MIMO sensing scheme which employs a constellation of SAR platforms [10]. In [9] and [11] reference is made

to the prospect of application of the MIMO concept to enable single-pass interferometric and tomographic SAR as well as allusions to entirely new processing schemes enabled by the increase in degrees of freedom.

State of the Art in Super-resolution Imaging

In their 2012 paper Zhu et al., demonstrated the ability of the SL1MMER algorithm to produce super-resolution in the elevation dimension and as a result effectively enhance the detection of double bounce scattering. This paper illustrated the importance of resolving double bounce scattering in the application of urban infrastructure monitoring using tomographic SAR [12]. In their 2012 paper Fischer et al. demonstrated the resolution of a linear prediction algorithm applied to data from a W-band automotive radar in a MIMO configuration [13]. Also in 2012, Mitchell and Tjuatja reported a method to estimate the dimension of the signal subspace using a feature vector developed from polarimetric measurements [14].

Contribution of the Proposed Method

The research detailed in this work adds to the art in the area of RADAR imaging. The incorporation of higher-order statistical information in the signal processing chain shows the ability to isolate and localize scatterers. This work shows that there is a strong correlation between leptokurtic scattering data and the scattering centers in the radar observation. The leptokurtic statistics of the RADAR data can be used as a contrast during the formation of weights to focus the radar data in the direction of each leptokurtic source. Use of the higher-order statistics to focus sub-aperture radar observations, enables, precise localization of scatterers which enhances the interpretability of a RADAR image.

In this work, a technique, the Isolated Scatterer Method (ISM), is developed which shows some merit in extracting unique scattering centers, however, this technique is also shown to be much less effective in the isolation of closely spaced scatterers.

Chapter 2 RADAR Scattering and Signal Processing Fundamentals

RADAR Scattering Fundamentals

Radio detection and ranging (RADAR) is a remote sensing modality which relies on the use of scattered electromagnetic waves in the radio frequency (RF) domain to infer characteristics about a remote target. Central to this sensing modality is the scattering of the waves.

Scattering Physical Model

Scattering of electromagnetic (EM) radiation is produced when a propagating EM wave front impinges upon a scatterer. The incident field results in a scattered field which is redirected in direction and magnitude according to the geometry and material of the scatterer. The scattered fields are the input to RADAR signal processing algorithms. It is the purpose of the signal processing algorithms to infer details about the objects which have produced the scatter. This section gives an overview of the physics that lead to the scattered fields. The intent of this section is to illustrate the extent of the assumptions and approximations required for closed from solutions to the wave scattering problem and illustrate the combinatorial variety of independent scattering mechanisms, which could exist in a non-canonical target. This knowledge will also serve as a basis for the assumptions made in the signal modeling and processing.

The relationship between incident and scattered fields is given by equations (1) and (2). The superscripts denote the total (t), incident (i), and scattered (s) elements of the electric (E) and magnetic (H) fields. E^i and H^i denote the fields in the absence of any scatterers [15]. Using a physical optics approximation, which neglects the effects of

edges, it is possible to solve for the elements of the fields for a few canonical shapes and geometries.

$$E^t = E^i + E^s \quad (1)$$

$$H^t = H^i + H^s \quad (2)$$

Scattering from finite size objects

Analytical formulations for scattering from finite-sized objects exist for several canonical shapes [15]. These formulations serve as a basis of understanding for scattering mechanisms present in a RADAR image. A practical target set would contain a diversity of scattering mechanisms of these types.

Closed form solutions, which describe the scattering for several canonical target types, have been developed. However, due to the necessary simplifying assumptions, computational complexity and diversity of target shapes their practical utility is hindered.

Some examples of scattering mechanisms for which closed form solutions have been developed are a flat plate, a circular cylinder and a sphere. Flat plates are a common feature in man-made structures. Buildings, roadways and vehicles all present features that resemble flat plates. Circular cylinders occur in many instances of practical scattering problems. From cannon barrels to airplane fuselages to tree trunks, cylinders appear in many forms for realistic RADAR targets. The conducting sphere is also a common target of interest in practical RADAR applications. Because of its uniform cross-section over all aspect angles the conducting sphere is often used as a reference by which the scattering properties of other targets can be estimated. The

measured data taken on the turntable system used in this research has been calibrated using the return from a conducting sphere as a reference

Various simplifying assumptions have allowed for the analytical solution to a few canonical scattering cases. These analytical solutions provide information on the signal characteristics which are expected from targets of these types. In practice a target set will consist of many types and orientations of scatterers and a challenge to RADAR signal processing is the separation and interpretation of this variety of scatterers. Knowledge of the exact scattering solutions allows the conclusion that differing scattering mechanisms produce responses which are unique in the wavenumber domain. This enables the assumption that these scattering mechanisms can be separated based on processing of the RADAR data in this domain.

Scattering Centers

Knowledge of the physical scattering models allows for the development of signal processing to extract information about the scatterers. In practice, a RADAR senses a collection of scatterers of unknown types, distribution and orientation. The ambiguity of a practical RADAR targets requires a highly generalized model of the scattering. Figure 1 shows a simple illustration to illuminate the idea of multiple simultaneous scattering types in a single observation. When considering practical scattering it is typical to consider these targets as collections of scattering centers [16] [17] [18].

Modeling scattering has been a robust area of research for over a century. The physical models of EM scatter have largely been based on ray tracing techniques such as physical and geometric optics (PO) (GO) [19] [20] [21]. A shortcoming of these physical models is that they neglect any scattering which occurs in the shadowed

region or by the edges of the target. The Geometric Theory of Diffraction (GTD) is an extension of geometric optics which takes into account diffraction which occurs due to edges, corners and vertices [22]. This model is more complete in the sense that additional scattering information is available, however, it is computationally expensive due to the fact that the wave equation must be solved for a larger set of boundary conditions which represents the edges of the scattering material.

In this research, the ray tracing techniques of physical and geometric optics will be employed. This choice was made in an attempt to keep the scattering model as general as possible so that the methods developed are applicable to generic scattering problems.

The analytical modeling of sources in this research will employ a plane wave source incident on a scattering center model with point source characteristics resulting in a spherical wave-front from each infinitesimal source. Use of this model will allow for analysis of scattering from a pseudo-random field of scatterers with known characteristics.

Scattering Statistical Model

More modern research has focused on using the statistical properties of the received signal to enable the extraction of information about the scattering field. The statistical moments of the received signal are used to infer characteristics of the target ensemble. In practice it is oft proper to consider the collection of scattering centers from a statistical perspective. David Middleton authored a seminal series of papers on the "Statistical Theory of Reverberation and Similar First-Order Scattered Fields" in 1967 and 1972 [23] [24] [25] [26]. This work models the stochastic process as a Poisson sampling process in which the scattering statistics are influenced by the field of point

scatterers as well as properties of the sensor including the sampling scheme of the aperture.

The physical model of scattering has the advantage that it provides a detailed picture of the scattering mechanism. However, the main disadvantage is the inability of this specialized model to handle the complex geometries, waveforms, and higher-order statistics. In Middleton's quasi-phenomological model, the detailed boundary conditions are absorbed into an impulse response which captures the interaction of the incident field, the scatterer, and the governing geometry. This type of modeling is an intermediate position between the detailed physics of the scattering and the macroscopic random function of time which it ultimately processed.

This research focuses on exploiting the statistical information embedded in the ensemble of scattered fields to extract information about constituents of the target. Using a quasi-phenomological model the signal statistics are related to the scattering mechanisms of the target.

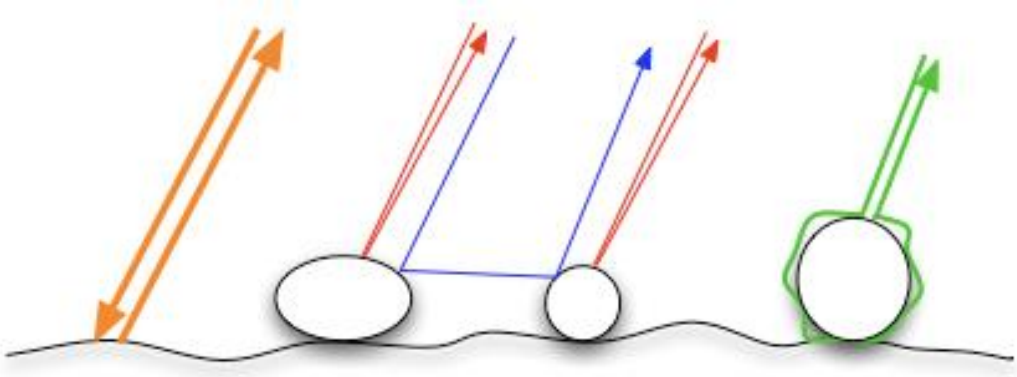

Figure 1: Practical scattering field example

Signal Model

To the first order a radar return is the complex superposition of scattering from all objects within the beam of the radar antenna. This can be expressed as the superposition of the various scatterers as shown in Equation (3) [27].

$$x_i = \sum_{j=1}^{M} a_{ij}\gamma_j + \epsilon_i \quad (3)$$

The coefficients a_{ij} represent the mixing coefficients for a particular scattering center γ_j for a given radar observation x_i with additive noise ϵ_i. In this study the x_i values are the radar measurements for a turntable rotation instance which corresponds to a look angle with index i. Each γ_j corresponds to an underlying component in the mixture which is a complex value in the frequency domain.

The resolution of the sensing system is typically large enough that the returns from multiple scattering centers are received simultaneously. Within this agglomeration there is an inherent irresolution. Depending on the target set, the irresolution prevents the separation of closely spaced scatterers or the separation of desired scatter from undesired scatter such as that from clutter. This dilemma can be visualized by considering the returns in Figure 1. If the scattering source of interest we considered to be the center object, it can be seen that the signal of interest is the reflection marked by the red arrow originating from this object. From this illustration it can be seen that there are multiple scattering elements superimposed into the observed mixture. Those elements include the return from the background, double bounce scattering, creeping waves, and scattering from nearby objects. These sources are respectively denoted in orange, blue, green, and red arrows. This simple illustration gives a sense of the complexity present in a practical radar return. This complexity is generally overcome by a simplification which treats each resolution cell, within a scattering scene, as a single

point source emanating a spherical wave front. This simplified signal model is the enabling assumption behind back-projection RADAR imaging techniques. The understanding contributed by the signal model shown in (3) allows the use of the novel method presented herein to isolate scatterers and extract additional information about the scattering field prior to the RADAR imaging.

RADAR Imaging Fundamentals

The separation of desired scattering from interference is a persistent area of research. Increasing the real resolution of a system comes at the cost of technological complexity which typically involves an increase in aperture size and bandwidth. In RADAR signal processing systems such as Synthetic Aperture RADAR (SAR) and Inverse Synthetic Aperture RADAR (ISAR) the amount of target information available is increased as the synthetic aperture size is increased. The coherence of a particular target signal over the collection parameters, such as angle and frequency, is the enabling factor in creating the increased resolution attributable to the synthetic aperture systems. Each element of the target scene has a phase and magnitude profile which varies as a function of the observation parameters. The synthetic aperture concept results in an increased resolution, typically in one dimension such as angle, and therefore affords greater success in separation of scattering.

When observed by the real aperture these target profiles are amalgamated into a complex measurement for each of the observations. SAR processing allows the phase history, attributable to a particular resolution cell, to be projected back to a perceived scatterer location and amplitude. This back-projection allows for the formation of a two or three-dimensional intensity image of a target scene. The resulting resolution of the target scene image is a function of the bandwidth, both spatial and

spectral, of the collection. Thus, the resolution comes at the expense of system complexity and longer dwell times.

Imaging Algorithms

RADAR imaging techniques fall into one of two broad categories, the first being those algorithms which are based on the Fourier transform relationship between the scattered fields in k-space and the scattering function of the object and the second being the class of super-resolution imaging methods based on alternate models of the scattering data.

Back-Projection Imaging Algorithms

For the first class of algorithms, Bojarski's identity gives the Fourier transform relationship between scattered field, $\overline{\mathrm{E_s}}(\mathrm{k_x}, \mathrm{k_y}, \mathrm{k_z})$ and the object reflectivity function, $\Gamma(\mathrm{x}, \mathrm{y}, \mathrm{z})$, which is related to the scattering properties of the object [1]. This relationship is shown in equation (4). The scattered electric field is a function of a variety of parameters. Some of these parameters, such as frequency and polarization, are set by the RADAR system. Others, such as the target shape and the material of the object being interrogated, are generally unknown. The scattered field can be measured and processed in order to estimate the target properties.

$$\overline{\mathrm{E_s}}(\mathrm{k_x}, \mathrm{k_y}, \mathrm{k_z}) = \iiint_{-\infty}^{\infty} \Gamma(\mathrm{x}, \mathrm{y}, \mathrm{z}) \exp\left(-\mathrm{j}(\mathrm{k_x x} + \mathrm{k_y y} + \mathrm{k_z z})\right) \mathrm{dxdydz} \qquad (4)$$

As shown in equation (5), the object reflectivity function can be recovered by inverse Fourier transform of the measured scattered field. This inversion allows for the reconstruction of the source location and magnitude which in turn leads to formation of 2 and 3-dimesional images of the scattering sources.

$$\bar{\Gamma}(x,y,z) = \iiint_{-\infty}^{\infty} \overline{E_s}(k_x,k_y,k_z)\exp\left(j\left(k_x x + k_y y + k_z z\right)\right) dk_x dk_y dk_z \quad (5)$$

The reflectivity function, $\bar{\Gamma}$, estimated by the back-projection technique is a composition of many scattering sources as shown in the signal model shown by equation (3). The mixture of returns in a resolution cell results in potential obscuration of the signal of interest. The presence of unwanted signals in the return creates a challenge in the extraction and interpretation of information from the signal of interest. This intent of this research is to explore the extraction of irresolvable target information and the application of this information to enhancing the scope of inference for RADAR images.

Super-Resolution Imaging Algorithms

Imaging based on the Fourier transform suffers from several performance limitations. The resolution of an image, which is based on the Fourier expansion, is limited by the size of the collection aperture. The resolution is inversely proportional to the spectral and spatial bandwidth over which the data is sampled. A second key limitation is diffraction which results in spectral leakage through the sidelobes of the Fourier impulse response [28] [29].

Super-Resolution systems are those which transcend the diffraction limitation imposed by standard optical processing techniques which are based on the Fourier transform. There are many forms of super-resolution techniques which are based on modern spectral estimation methods such as adaptive sidelobe reduction (ASR), space variant apodization (SVA), periodogram, minimum variance method (MVM), reduced rank minimum variance method (RRMV M), autoregressive linear prediction (ARLP), Pisarenko, eigenvector (EV) / multiple signal classification (MUSIC), Tuft-Kumaresan

ARLP (TKARPL) and parametric maximum likelihood (PML). All of these techniques - with the exception of ASR and SVA - come at a computational expense relative to the Fourier transform method. The majority of these methods employ a model which is based on the sample correlation matrix of the data. The treatment of the correlation matrix depends on the method, however, the primary objective of the techniques is to increase the signal-to-interference ratio (SIR) for the image.

The MUSIC approach enhances the SIR by separating the signal subspace from the interference subspace. This approach to signal processing has been explored since the 1970s. Schmidt popularized a general subspace approach in his PhD thesis [30] and a subsequent journal paper [2]. His method, called MUSIC, is a general approach to determine the parameters of multiple wavefronts, using signals sensed by an antenna array. The basic premise of the MUSIC algorithm is to use the orthogonality of the signal space eigenvectors to noise space eigenvectors to determine the direction and amplitude of the signal vectors using matrix algebra techniques. A key step in the application of the MUSIC algorithm is the determination of the Eigen decomposition of the correlation matrix [28].

The correlation matrix can be used to model the combination of the signal and noise subspaces as shown in equations (6) and (7). In equation (6) the $\sigma_0 \mathrm{N}$ term denotes white clutter and the vector $\mathbf{S}$ is the sum of L sinusoids originating from spatial locations denoted by a set of radial vectors $\mathrm{r_l}$.

$$\mathrm{X} = \sigma_0 \mathrm{N} + \sum_{\mathrm{l}=1}^{\mathrm{L}} \sigma_\mathrm{l} \mathrm{W}(\mathrm{r_l}) = \sigma_0 \mathrm{N} + \mathbf{S}\sigma \tag{6}$$

$$\mathbf{R} = \sigma_0^2 \mathbf{I} + \mathbf{S}\mathrm{E}(\sigma\sigma^{\mathrm{H}})\mathbf{S}^{\mathrm{H}} = \sigma_0^2 \mathbf{I} + \mathbf{SCS}^{\mathrm{H}} \tag{7}$$

The correlation matrix can now be modeled in terms of its Eigen decomposition. The Eigen space is then segregated into two classes. One class

containing the Eigen values and vectors which correspond to the L sinusoids in the signal subspace and the other class representing the interfering clutter. Formulaically this is expressed in (8). Equation (9) shows the equation for the inverted correlation matrix as this is the term which will be used to estimate the sources in the MUSIC.

$$\mathbf{R} = \sum_{\text{clutter}} \lambda_m V_m V_m^H + \sum_{\text{signal}} \lambda_m V_m V_m^H = \sum_{\text{clutter}} \sigma_0^2 V_m V_m^H + \sum_{\text{signal}} \lambda_m V_m V_m^H \quad (8)$$

$$\mathbf{R}^{-1} = \sum_{\text{clutter}} \sigma_0^{-2} V_m V_m^H + \sum_{\text{signal}} \lambda_m^{-1} V_m V_m^H \quad (9)$$

The next step in the MUSIC signal processing algorithm takes advantage of the signal to noise subspace orthogonality. Using this assumption, a spatial search through the noise subspace can be conducted in which the output produces sharp peaks at locations where a signal is estimated to originate. The objective function of this search takes the form of equation (10) where $W(r)$ denotes the complex exponential associated with of a 2D Fourier transform tuned to a spatial location r. Under the point scatterer assumption these unit sinusoids correspond to scattering from that location.

$$S_{MUSIC}(r) = \frac{1}{W^H(r)(\sum_{\text{clutter}} \sigma_0^{-2} V_m V_m^H)W(r)} \quad (10)$$

Blind Source Separation Fundamentals

Blind source separation (BSS) is a rich class within the inverse problem domain which seeks to ascertain the constituents of a set of mixtures given only observations of the mixtures. There are a number of mathematical methods aimed at accomplishing the estimation of mixture components.

Principal Component Analysis (PCA) is a popular BSS method which separates the components by mapping the data in such a way as to sequentially maximize the variance of each component. PCA is a member of a set of source

separation techniques which employ second order statistics in the process. This class of techniques is capable of extracting signals from a set of mixtures which are uncorrelated with one another [6].

Singular Value Decomposition (SVD) extracts components of a mixture by applying a sequence of rotation and scaling such that the resulting data form a generalized Eigenvalue Decomposition. The Eigenvalue Decomposition seeks the roots of the characteristic polynomial for the observation set (eigenvalue) and subsequently solves for an associated set of Eigenvectors which model the data in an orthogonal basis [31]. These techniques are based on linear independence according to second order statistics.

Statistical signal processing techniques such as MUSIC and ESPRIT perform signal separation through Eigen decomposition of the covariance matrix for a given observation set. These techniques therefore perform the separation based on the second order statistics of the dataset.

Independent Component Analysis (ICA) performs source separation based independence of the elements of the mixture. ICA is built upon the foundation of the central limit theorem which states that

> If a set of signals $\mathbf{s} = (s_1, s_2, \cdots, s_M)$ are independent with means $(\mu_1, \mu_2, \cdots, \mu_M)$and variances $(\sigma_1^2, \sigma_2^2, \cdots, \sigma_M^2)$ then, for a large number M of signals $\mathbf{s}$ the signal
>
> $$x = \sum_{j=1}^{M} s_j \qquad (11)$$
>
> has a pdf which is approximately Gaussian with mean $\sum_j \mu_j$and variance $\sum_j \sigma_j^2$.

This foundation enables the existence of a class of BSS algorithms which seek to separate sets of mixtures based on their non-Gaussianity. The premise of the method

is that the less Gaussian a signal then the lower the likelihood that it is mixture of sources. Given the assumption that the sources are independent of one another, the most non-Gaussian signals extracted from a Gaussian mixture of signals must be the source signals [6] [32].

Independent Vector Analysis (IVA) is an emerging technique for performing joint blind source separation (JBSS). IVA is an extension of ICA designed to comprehend source correlation among multiple datasets [33] [34]. This technique seeks restitution of latent sources as captured in manifold sets. The correlation of the underlying sources among the observation sets is taken into account which naturally lends the technique to subapertured data collection.

Chapter 3 Novel Approach to Scattering Separation

The objective of beamforming in radar data is to extract desired characteristics from the data and suppress those which are not germane to the task of developing information from the data. Radar signal processing, in general, is devoted to the extraction and identification target features. Beamforming of radar data emphasizes target features of interest and deemphasizes the noise, clutter and interference which is embedded in the radar data.

The novel method proposed here uses the higher-order statistics of the radar data to adapt beamforming weights such that the scattering elements which possess the highest non-Gaussianity are isolated. Two common algorithms for weight optimization aimed at isolating the highly kurtotic elements from within the data are JADE and robustICA [4] [5]. This work employs robustICA in the estimation of the optimal beamforming weights for the separation of high kurtosis elements of the RADAR scattering data.

Formulation and Application of the New Technique

Considering a scattering source located at $r_\gamma, \theta_\gamma, \phi_\gamma$ the signal model is

$$s_\gamma[k,\theta,\phi] = \frac{\Gamma_\gamma * e^{\mathbb{i}*(k*r_\gamma*\cos(\theta_\gamma-\theta)\sin(\phi_\gamma-\phi))} * e^{\mathbb{i}*(k*r_\gamma*\sin(\theta_\gamma-\theta)\sin(\phi_\gamma-\phi))}}{\left(4*\pi*r_\gamma*\cos(\theta_\gamma-\theta)\sin(\phi_\gamma-\phi)+r_\gamma*\sin(\theta_\gamma-\theta)\sin(\phi_\gamma-\phi)\right)^2}+\epsilon[k,\theta,\phi] \quad (12)$$

In the first order, radar data for a given set of observation parameters $[k,\theta,\phi]$, is a linear mixture of scattering sources at various locations. The signal mixture model can be written as in (13).

$$\mu[k,\theta,\phi] := \sum_{\gamma=1}^{R} s_\gamma[k,\theta,\phi] \quad (13)$$

In general, beamforming is the weighted superposition of multiple simultaneous observations of a signal. For a synthetic aperture radar system a single source and receiver which samples data over a discrete set of frequencies and look angles is used to acquire the data. To apply a beamforming to this set of data, it is necessary to synthetically group the observations to form set of subapertures which is commensurate with subsequent signal processing steps. The minimum required set of subapertures is dictated by the number of components which are being estimated. For instance, if it were desired to isolate N components from within the mixture, this would require at least N observations or subapertures of the observation set. The technique chosen to accomplish this preprocessing step is grouping of the data into subapertures which span a predefined portion of the observation space in terms of line-of-sight angle and frequency spectrum.

To form subapertures, the native k-space radar data is divided into an integer number of equal-sized partitions along the wave number and look angle dimensions of the data. In the event the number of subaperture divisions, in one or both of the dimensions, is not an integer, the remainder of the data in either dimension will be excluded from the subapertured data set. Assuming diversity in the azimuthal and wavenumber domains, Equations (14) and (15) show how the subapertures are formed from the original data. In these equations, the variable T signifies the full dimensional measurement data.

$$\mathrm{T}_{\mathrm{sa}}^{\mathrm{i}}(\mathrm{m},\mathrm{n}) = \mathrm{T}\left(\left\{\theta_1, \theta_2 \cdots \theta_{\bar{\bar{\theta}}_{\mathrm{P}}}\right\}, \left\{\mathrm{f}_1, \mathrm{f}_2 \cdots \mathrm{f}_{\bar{\bar{\mathrm{f}}}_{\mathrm{N}}}\right\}\right) \tag{14}$$

$$\text{where } \bar{\bar{\cdot}} \text{ denotes cardinality of the set}$$

$$\mathrm{T}_{\mathrm{sa}}^{\mathrm{i}} \in \mathrm{T}\ \forall\ \mathrm{i} \in \{1,2,\ldots,NP\} \mid \mathrm{T}_{\mathrm{sa}}^{\mathrm{i}} \cap \mathrm{T}_{\mathrm{sa}}^{\mathrm{j}\neq\mathrm{i}} = \{\} \tag{15}$$

The subaperture approach applied to the radar data containing signal mixtures is shown below. The full data plane for the radar observation is given by (16). Equation 15 depicts the subapertured data plane for $(NP-1)$ subaperture divisions

$$X=\begin{bmatrix}\mu[\mathrm{k}_1,\theta_1,\phi] & \mu[\mathrm{k}_1,\theta_2,\phi] & \dots & \mu[\mathrm{k}_1,\theta_M,\phi]\\ \mu[\mathrm{k}_2,\theta_1,\phi] & \mu[\mathrm{k}_2,\theta_2,\phi] & \dots & \mu[\mathrm{k}_2,\theta_M,\phi]\\ \vdots & \vdots & \vdots & \vdots\\ \mu[\mathrm{k}_L,\theta_1,\phi] & \mu[\mathrm{k}_L,\theta_2,\phi] & \dots & \mu[\mathrm{k}_L,\theta_M,\phi]\end{bmatrix} \tag{16}$$

$$\begin{bmatrix} X_1=\begin{bmatrix}\mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*1},\phi]\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*1},\phi]\\ \vdots & \vdots & \vdots & \vdots\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*1},\phi]\end{bmatrix}\\ X_2=\begin{bmatrix}\mu[\mathrm{k}_{\lfloor L/N\rfloor*1+1},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1+1},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1+1},\theta_{\lfloor M/P\rfloor*1},\phi]\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*1+2},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1+2},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1+2},\theta_{\lfloor M/P\rfloor*1},\phi]\\ \vdots & \vdots & \vdots & \vdots\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*2},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*2},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*2},\theta_{\lfloor M/P\rfloor*1},\phi]\end{bmatrix}\\ \vdots\\ X_N=\begin{bmatrix}\mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+1},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*1},\phi]\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+2},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*1},\phi]\\ \vdots & \vdots & \vdots & \vdots\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*N},\theta_{\lfloor M/P\rfloor*0+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*0+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*1},\phi]\end{bmatrix}\\ X_{N+1}=\begin{bmatrix}\mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*1+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*1+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+1},\theta_{\lfloor M/P\rfloor*2},\phi]\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*1+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*1+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*0+2},\theta_{\lfloor M/P\rfloor*2},\phi]\\ \vdots & \vdots & \vdots & \vdots\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*1+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*1+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*1},\theta_{\lfloor M/P\rfloor*2},\phi]\end{bmatrix}\\ \vdots\\ X_{NP}=\begin{bmatrix}\mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+1},\theta_{\lfloor M/P\rfloor*(P-1)+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+1},\theta_{\lfloor M/P\rfloor*(P-1)+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+1},\theta_{\lfloor M/P\rfloor*P},\phi]\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+2},\theta_{\lfloor M/P\rfloor*(P-1)+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+2},\theta_{\lfloor M/P\rfloor*(P-1)+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*(N-1)+2},\theta_{\lfloor M/P\rfloor*P},\phi]\\ \vdots & \vdots & \vdots & \vdots\\ \mu[\mathrm{k}_{\lfloor L/N\rfloor*N},\theta_{\lfloor M/P\rfloor*(P-1)+1},\phi] & \mu[\mathrm{k}_{\lfloor L/N\rfloor*N},\theta_{\lfloor M/P\rfloor*(P-1)+2},\phi] & \dots & \mu[\mathrm{k}_{\lfloor L/N\rfloor*N},\theta_{\lfloor M/P\rfloor*P},\phi]\end{bmatrix}\end{bmatrix} \tag{17}$$

The subapertures can then be placed in the block form represented by $\widehat{\mathbf{X}}$. The rows of this block matrix are considered separate observations of the signal mixture model. This synthetic set of observations can now be used to estimate weights which extract the most leptokurtic elements within the mixture.

$$\mathbf{X}=[\mathrm{vect}\{X_1\}\,\mathrm{vect}\{X_2\}\,\cdots\,\mathrm{vect}\{X_{NP}\}]^T \tag{18}$$

Once the sub-apertures are formed, an instantaneous mixture model can be written as in (19).

$$\boldsymbol{S}=\mathbf{WX} \tag{19}$$

The next step in the process is to estimate the location of the leptokurtic components of the subaperture observation set $\mathbf{X}$. This is performed through the use of an optimization routine.

To seek out the most non-Gaussian components of an observed mixture of sources, a common method used by several Independent Component Analysis algorithms, is to perform an optimization function with kurtosis as the cost function. RobustICA is the choice of routines used in this work because of its ability to handle complex sources, forego Principal Component Analysis and its computational efficiency [5] [35]. The formulation below is the robustICA approach to kurtosis maximization.

The kurtosis as function of the extraction weights, $\mathbf{w}$, is shown in (20)

$$\mathcal{K}(\mathbf{w}) = \frac{\mathrm{E}\{|\mathrm{y}|^4\} - 2\mathrm{E}^2\{|y|^2\} - |\mathrm{E}\{y^2\}|^2}{\mathrm{E}^2\{|y|^2\}} \tag{20}$$

RobustICA employs an exact line search using the kurtosis cost function shown in (20) and estimating the optimal step size as

$$\mu_{opt} = \arg\max_{\mu} |\mathcal{K}(\boldsymbol{w} + \mu\mathbf{g})| \tag{21}$$

Where the search direction $\mathbf{g}$ is given by the gradient of the kurtosis with respect to the weights.

$$\nabla_{\mathrm{w}}\mathcal{K}[\mathrm{w}] = \frac{4}{\mathrm{E}^2\{(|\mathrm{y}|)^2\}}\{\mathrm{E}\{(|\mathrm{y}|)^2\mathrm{y}^*\mathrm{x}\} - \mathrm{E}\{\mathrm{yx}\}\mathrm{E}\{\mathrm{y}^{*2}\}\} - \frac{(\mathrm{E}\{\mathrm{y}^4\} - (|\mathrm{E}\{\mathrm{y}^{*2}\}|)^2\mathrm{E}\{\mathrm{y}^*\mathrm{x}\})}{\mathrm{E}\{(|\mathrm{y}|)^2\}} \tag{22}$$

To reduce the computational expense of the exact line search, Zarzoso and Common make use of a polynomial representation of the kurtosis cost function. Derivation of the optimal step size polynomial is given in [5].

Deflationary orthogonalization is used to prevent the development of identical weights as the routine traverses the set of leptokurtic sources. This is accomplished by

forcing each new set of extracting weights to lie within the orthogonal subspace of previously extracted weights which are stored in a matrix $\mathbf{W_k} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{k-1}]$. The formulation for deflationary orthogonalization and the weight normalization constraint are shown in (23) and (24).

$$\mathbf{w}^+ \leftarrow \mathbf{w}^+ - \mathbf{W_k}\mathbf{W_k^H}\mathbf{w}^+ \tag{23}$$

$$\mathbf{w}^+ \leftarrow \frac{\mathbf{w}^+}{\|\mathbf{w}^+\|} \tag{24}$$

Once the matrix $\mathbf{W}$ has been formed, each of the leptokurtic sources can be estimated from the instantaneous linear mixture model (19). Once the desired sources of interest have been estimated, the next step is to map the data back to the native k-space domain to enable RADAR imaging

The extracted elements are transformed back into the native k-space domain through a remapping of the subapertures. The domain of the remapped observation matrix $\mathbf{x}$ formed from each extracted component matches the domain original data with the possible exception of any excluded data resulting from the remainder of the subaperture process.

Figure 2 is a block diagram illustration of the signal processing for this technique. This includes the division of the k-space domain into subapertures followed by the extraction of the highly kurtotic components and finally remapping of the data back into the native measurement domain

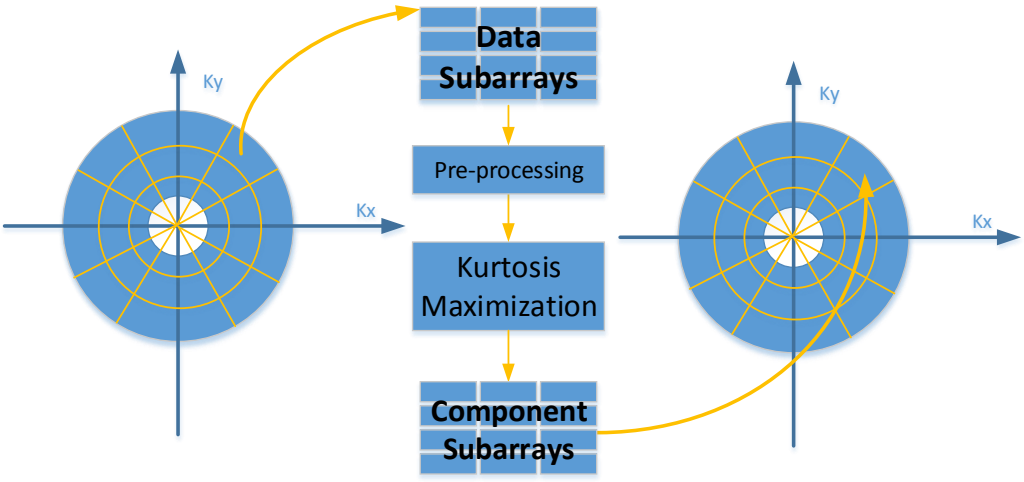

Figure 2: Signal Processing Flow Diagram

Each of the extracted components is remapped to the measurement domain and a RADAR image is formed using the discrete Fourier transform

$$\bar{\Gamma}_{\gamma}(x,y) = \sum_{\theta=-\pi/2}^{\pi/2} \sum_{k=-\infty}^{\infty} \hat{s}_{\gamma}[k,\theta,\phi] e^{j(k_x x + k_y y)} \tag{25}$$

Where $k_x = k \sin\theta \sin\phi$, $k_y = k \sin\theta \cos\phi$ and ϕ is a constant. This equation projects the estimated scattering sources $\hat{s}_{\gamma}$back to the location which is defined by the phase history of the extracted source. The sources have now been isolated through focusing of the subapertures using kurtosis as a contrast.

Taking advantage of the assumption that each highly kurtotic element focuses to an independent scattering center, an additional signal processing step of picking the maximal value in each of the high-kurtosis images is performed. This step is depicted in Equation (26). Compilation of the peaks for each of the extracted components allows a composite view of the scattering centers to be constructed.

$$\left(x_\gamma, y_\gamma\right) = \underset{[x,y]}{argmax} \left|\bar{\Gamma}_\gamma(x,y)\right| \tag{26}$$

The following chapter of this manuscript critically examines the application of this technique to synthetic and measured RADAR imaging data sets. Consideration will be given to the accuracy with which this technique can produce an estimate of the location of each scattering center within a scene. The subsequent chapters pose an examination of the application of the technique under varying noise and scatterer configurations in order to characterize the capability and limitation of this proposed technique.

Performance Assessment of the Scatter Separation Technique

A parametric study of the division into subapertures has been performed. Increasing numbers of subapertures in the frequency dimension were applied to scattering scenes containing a variety of discrete scattering centers. Increasing the number of subapertures in a given data dimension provides more degrees of freedom for the beamformer.

Using the simulated scattering data, the proposed technique was applied while using various levels of division along the frequency dimension of the RADAR data. The results of these various trials have been analyzed to produce an understanding of the performance with respect to estimated scattering center location error and the ability of the technique to extract all known scatterers. The data is analyzed within the back-projected image domain which is generated using Equation (25). The scattered electric field $\hat{s}_\gamma$ in this equation is the field produced by each of the individual extracted components in the estimated isolated scatterer matrix $\widehat{\boldsymbol{S}}$.

$$\bar{\Gamma}_{\gamma}(x,y) = \sum_{\theta=-\pi/2}^{\pi/2} \sum_{k=-\infty}^{\infty} \hat{s}_{\gamma}[k,\theta,\phi] e^{j(k_x x + k_y y)} \quad (27)$$

The next step in the signal processing involves taking the location of the peak magnitude position of the extracted component in the image domain. This step is based on the assumption that each $\bar{\Gamma}_{\gamma}$ represents an isolated mechanism within the field of scatterers. Therefore the maximum magnitude produce from the application of the direction search vectors, $e^{j(k_x x + k_y y)}$, is taken to represent the true location of each scatterer. The set of peaks developed from the isolated scatterers is combined to form a composite view of the positions for the set of extracted scatterers.

Two metrics were used to quantify the effectiveness of the technique at resolving individual scatterers. The first metric is a measure of the location error relative to the known theoretical scatterer locations. The formulation for computing the distance error is the standard Euclidean distance measure shown in Equation (28). This metric is hence forth referred to as Target Location Error (TLE).

$$\varepsilon_{dist} = \sqrt{(x_{extracted} - x_{selected})^2 + (y_{extracted} - y_{selected})^2}$$

Where

$$\{x_{selected} \mid x_{selected} \in Y\} = \underset{x_{known}}{\operatorname{argmin}}(|x_{extracted} - x_{known}|) \quad (28)$$

And

$$\{y_{selected} \mid y_{selected} \in Y\} = \underset{y_{known}}{\operatorname{argmin}}(|y_{extracted} - y_{known}|)$$

The second metric is a measure of the ability of the technique to extract the full set of known scatterers within the observation field. This metric is computed as a percentage based on the ratio of cardinality of the set of extracted scatterers to the cardinality of the set of known scatterers. This represents the probability of detection of the set of known scatterers within the scene.

Chapter 4 Results

Scattering Center Isolation Through Kurtosis Maximization

This section is dedicated to presenting evidence to support the conclusion that beamforming based on the fourth order statistical cumulant, kurtosis, is useful in extracting individual scattering elements within a scene of scatterers. An application based on the composite maximum pixel of each of the extracted components is presented as a RADAR imaging technique based on this novel method. This method will be referred to as the Isolated Scatterer Method (ISM).

The data presented here is both synthetic and measured. The synthetic data is generated assuming a turntable ISAR measurement system which mimics that of the measurement system used to collect the measured data. The data synthesis is parameterized such that the collection parameters and target scattering properties can be varied. The target set is assumed to be a field of point scatterers that can be placed at any range or angle relative to the turntable coordinate system. The center of the turntable is the reference for the range, and the first collection angle is the reference for the angular term of each scatterer's polar coordinate. The magnitudes of each of the scatterers is another parameter available through the model. Another important parameter is the magnitude of the additive white Gaussian noise imparted on the synthetic data. Using a combination of the scattering magnitudes and the noise level, several SNR cases were realized. The performance of this technique under various SNR conditions was explored.

The effectiveness of the application of this hypothesis was studied over a host of conditions as depicted in Table 1. This section of the manuscript will focus on two of the test conditions. Excellent performance was realized in the first of these test cases

and the second test case produced marginal performance. The intent of the selection of these cases is to illustrate views and metrics which were used in this research, as well as set some expectation for the practical limitations of this technique.

Table 1: Study Parameters

SNR	-30	-20	-10	0		
Scatterer Count	2	3	4	5	6	7
Excess Degrees of Freedom	0	1	2			

Test Case 1

For this test case excellent performance was realized. The metrics used to assess performance, which are introduced in this section, are 1) a measure of location error, and 2) percentage of scatterers detected.

Examination of this case will begin with a depiction of the performance attributable to typical Fourier based radar imaging. Figure 3 is an example of an ISAR image formed using the back-projection technique shown by equation (5). The synthetic data set used for this image was made up of three isotropic scatterers. The locations and orientations were randomly chosen using a uniform distribution $\mathcal{U}(0,1)$ for the range and a uniform distribution $\mathcal{U}(0,2\pi)$ for the angle. The theoretical scattering orientations are shown in Figure 3 using colored circles to show the location of the point scatterer. This example has an SNR of 0 dB for each scatterer in the complex sampled data.

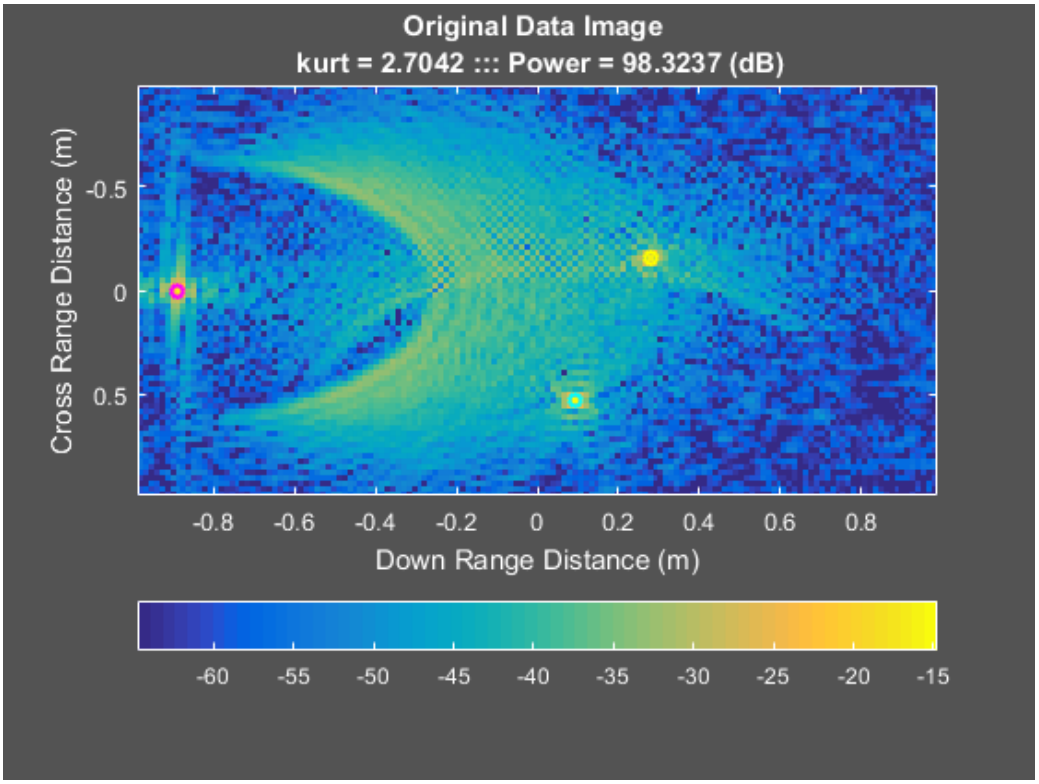

Figure 3: ISAR image of synthetic target of 3 scatterers

Examination of the ISAR image reveals three well defined regions of scattering near the locations of the synthesized scatterers. Also visible in the image are artifacts attributable to noise and sidelobes of the known scatterers. Noise in a radar observation is unavoidable and its mitigation has spurred decades of research. The sidelobes are another limitation of the diffraction-limited Fourier based imaging. Super-resolution techniques attempt to apply advanced signal processing to reduce the negative impact of noise and sidelobes.

The next set of images are those formed after application of the beamforming weights associated with kurtosis maximization. To create a multiplicity of unique beamforming weights, this processing technique requires three observations of the

target set. To satisfy this requirement, the original k-space data set was segmented into three subapertures in the frequency dimension of the data. These subapertures form sectors of annuli of equal widths which span the full angular dimension of the data set.

The choice of three subapertures is driven by the a-priori knowledge that there are three scattering elements contained in the scene. As part of the parametric study that accompanied this research a variety of partitioning in the frequency dimension was explored. The range of partitioning was dictated by the number of synthetic scatterers in the field. A set of subapertures equal in number to the count of synthetic scatterers is classified as a having zero “excess degrees of freedom.” The range of excess degrees of freedom in this research ranges from zero to two. Cases with excess degrees of freedom greater than zero are afforded more data snapshots, or subapertures, than the minimum required to localize all of the known scatterers.

The most desirable outcome for all cases in this study is precise localization of all of the known scatterers in the field which amounts to a location error of zero. This requires that the exact pixel location of the synthetic scattering location be emphasized such that it is the maximum in the scene. Metrics shown in this chapter will illustrate the degree to which all of the examined cases meet these criteria.

Figure 4 shows an ISAR image formed using the weighted complex data produced using the technique of this paper. It can be seen that the beamforming weights, which yield the highest kurtosis using this data, have emphasized the scatterer marked by the cyan circle. The second and third extracted components of high kurtosis are shown in Figure 5 and Figure 6 respectively. Figure 5 shows the emphasis of the scatterer noted in yellow, and Figure 6 the beamforming technique has favored the magenta scatterer.

In each of the images shown in Figure 4 through Figure 6, the black asterisks represent the location of the maximal pixel in the scene. It can be seen from these figures that the technique has resulted in the maximization of the magnitude at each of the exact locations of the known scatterers.

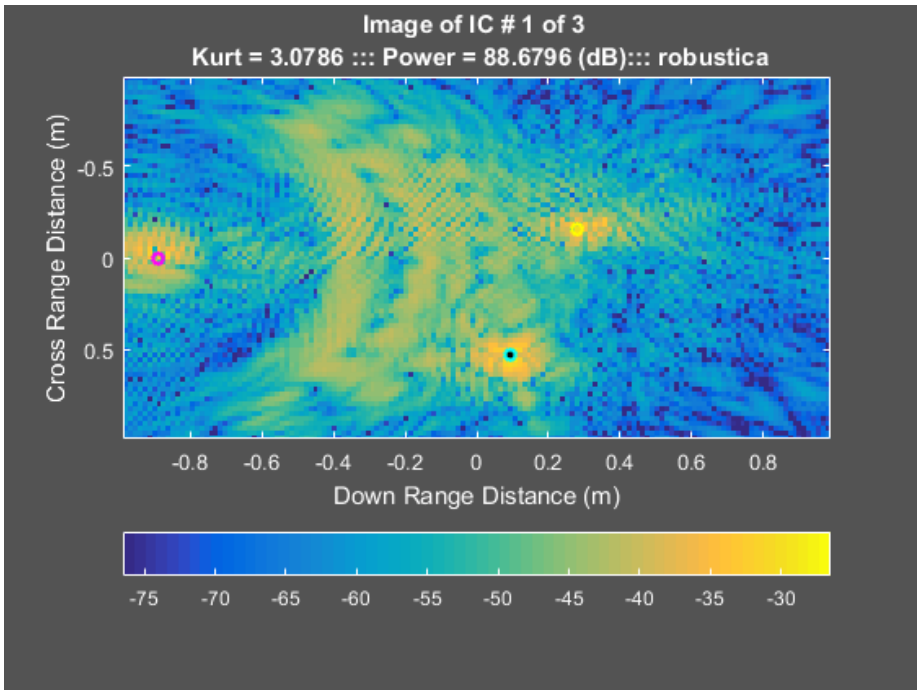

Figure 4: High Kurtosis Scattering Element #1 of 3

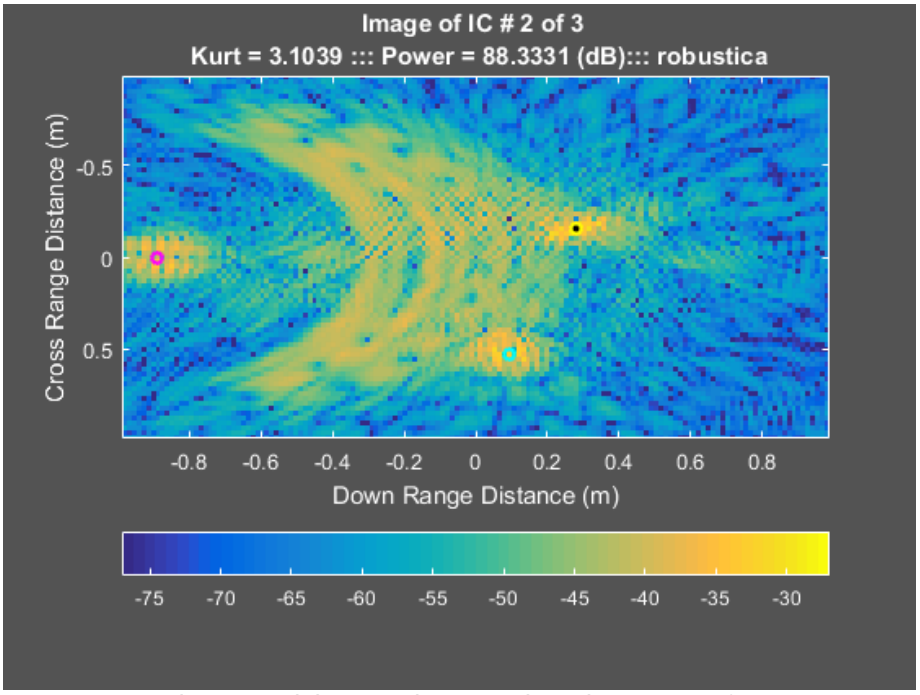

Figure 5: High Kurtosis Scattering Element #2 of 3

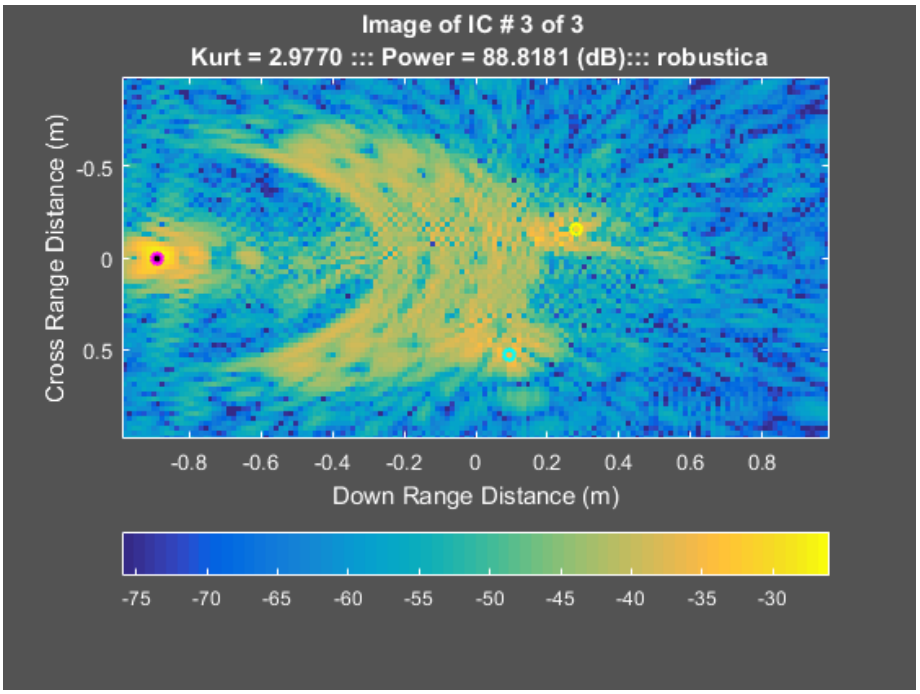

Figure 6: High Kurtosis Scattering Element #3 of 3

Another outcome produced by the application of this beamforming technique is the de-emphasis of the other two scatterers in the scene. This is more readily perceived using the graph in Figure 7. This graph shows the Signal-to-Interference Ratio (SIR) which represents the ratio of the power present in the pixel closest to the theoretical location of the scatterers to the power present in the rest of the pixels within the scene. The normalization reference for each of the scatterers is the power in the pixel for the original ISAR image. Examining location of the cyan and the magenta points for the second extracted component in this graph, it can be seen that the values are negative indicating a reduction in the power within the de-emphasized pixels. The nulling of the similar magnitude scatterers by the beamformer serves to increase the contrast of the emphasized pixel relative to the rest of the scene.

Another important observation about the image produced from this component is that the sidelobes of the emphasized scatterer also experience an increased gain. The elevation of the sidelobes is contrary to the goal of increasing the resolvability of the individual scattering elements. Therefore, in the development of the application for this theory, an additional processing step was applied. In this step all sidelobes are removed through extraction of only the highest magnitude pixel within the scene. The output of this processing step is shown in Figure 8. An alternate view of the output from this technique is shown in Figure 9 where the composite maximum image is overlaid on the original ISAR image. Examination of this view shows utility in drawing attention to the true scattering source locations. The resolution of the overlaid composite view of the maximums can be made arbitrarily small.

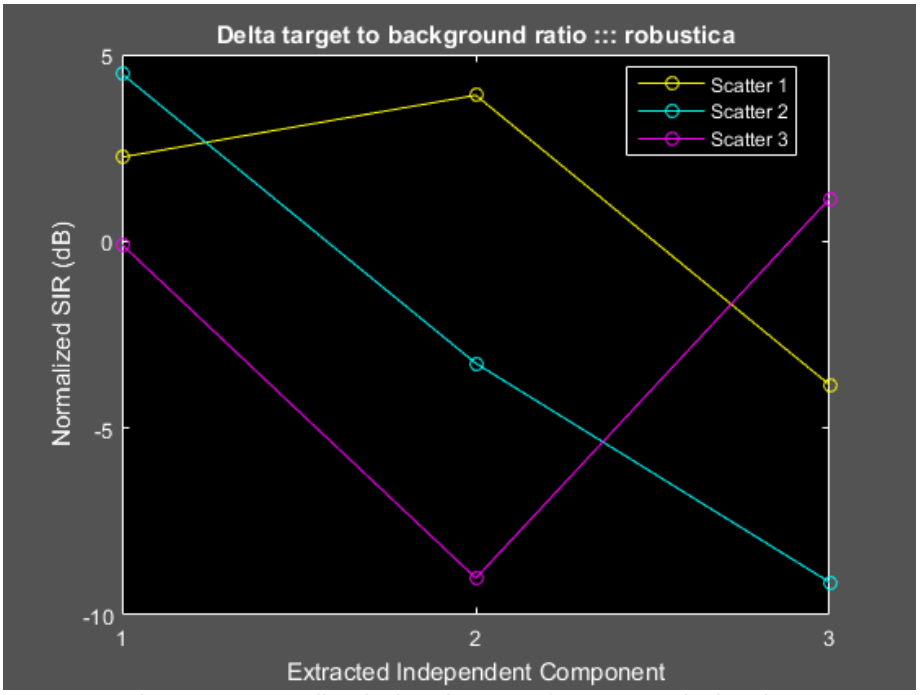

Figure 7: Normalized Signal-to-Interference Ratio (SIR)

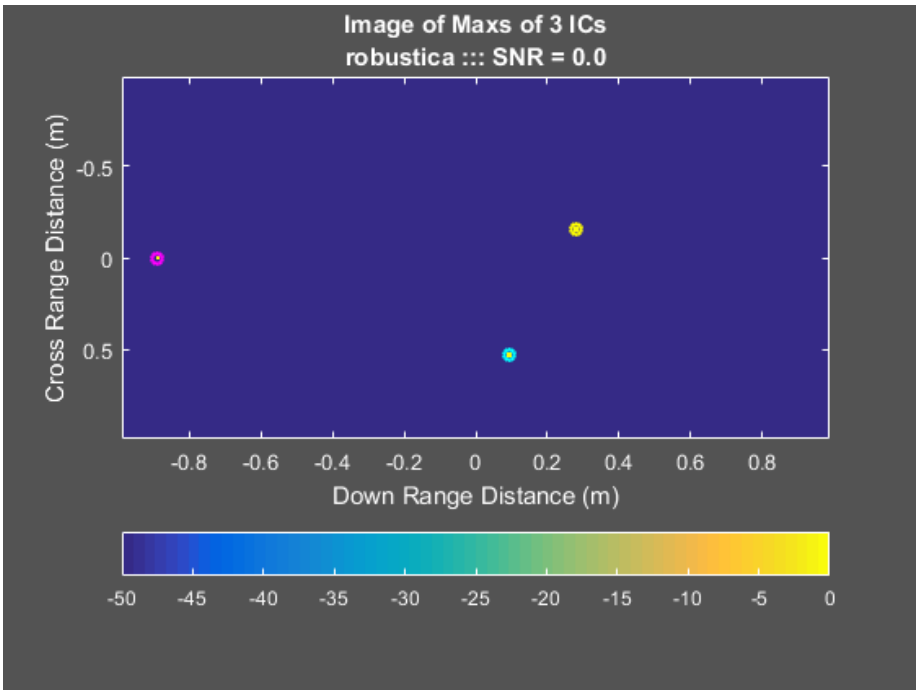

Figure 8: Composite Maximum Image of 3 Extracted Components

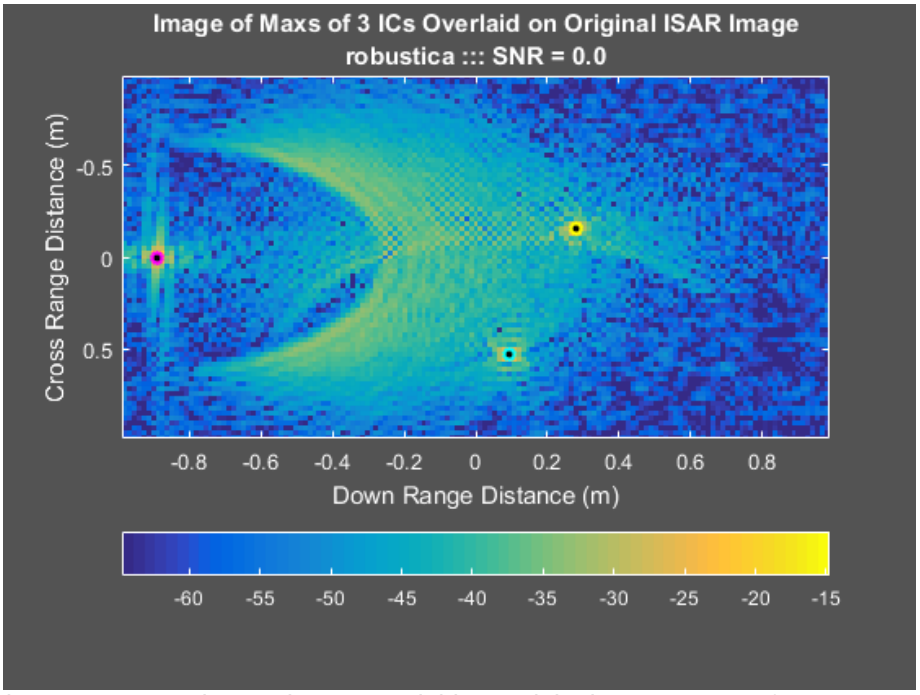

Figure 9: Composite Maximum Overlaid on Original ISAR Image of 3 Scatterers

Test Case 2

The following test case illustrates some of the noted deficiencies in the proposed approach. This case will examine a field of seven known scatterers. The SNR for this example has been set at -20 dB. This example will also have zero excess degrees of freedom which implies seven subaperture divisions.

This examination will begin with a view of the standard ISAR image formed from the complete dataset. Figure 10 reveals that the poor SNR has produced an image in which it is difficult to distinguish the areas scattering from the elevated background noise.

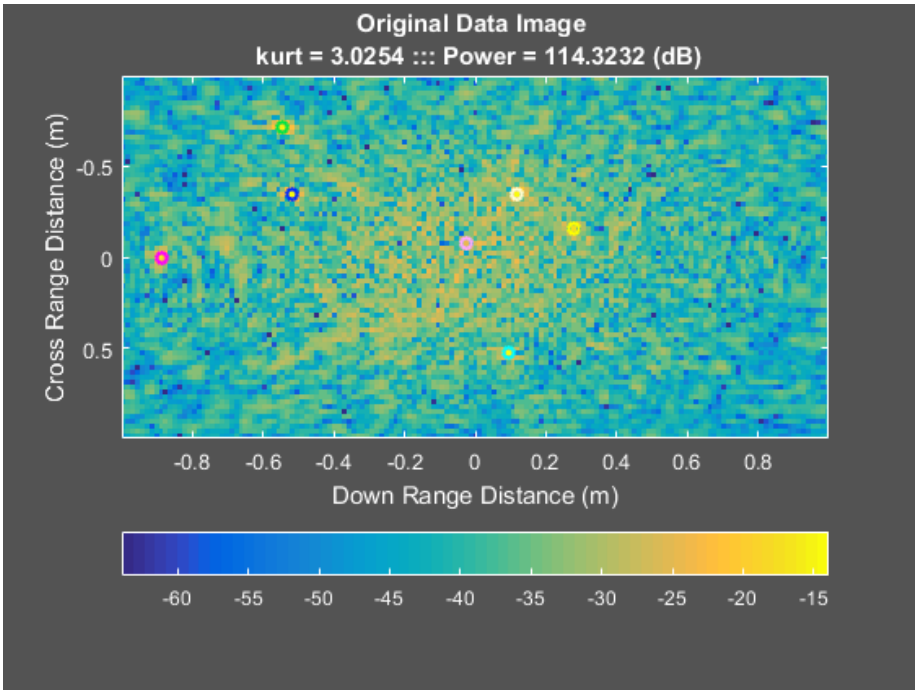

Figure 10: ISAR image of synthetic target of seven scatterers

The next 7 images (Figure 11 through Figure 17) show the images formed from the seven highest kurtosis elements of the scattering data. It can be seen from these images that the ability of the method to precisely locate the scatterers is poor. In this case only two of the seven scatterers is localized with zero location error. This amounts to a 29% probability of detection. If the results are considered with some allocation for target location error, then the probability of detection increases to 57%. The detected locations using the method described in this work have a set of location errors as depicted in Figure 20. Each of the detections is associated with the closest known scattering location. In the distance error plot the association is depicted by the color of the marker for the given detection. The distance error ranges from zero to 0.1 meter for this test case. The two scattering centers which, were exactly localized, are noted by

the green and white markers. There were duplicate detections associated scatterers marked with the green, white, and purple. Of the duplicate detections, some appear to be false alarms caused by the noise and sidelobes present in the image. Figure 13 and Figure 17 are examples of potential false alarms. Also shown below in Figure 18 is the composite maximum image for this trial. Figure 19 shows the composite maximum overlaid on the original ISAR image.

Examination of this test case shows some of the deficiencies in this approach. The performance of the proposed method is degraded in lower signal-to-noise ratio conditions. Another deficit of the method is the inability of the method to guarantee detection of all known scatterers. However, this novel method has been shown to produce useful information such as a non-zero probability of detection and precise localization even in low SNR conditions.

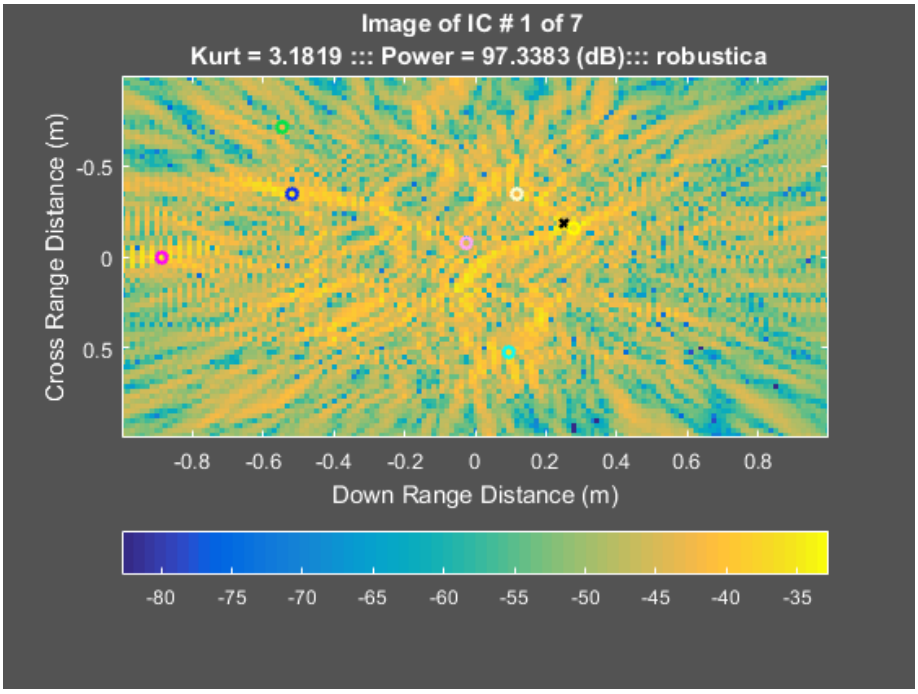

Figure 11: High Kurtosis Scattering Element #1 of 7

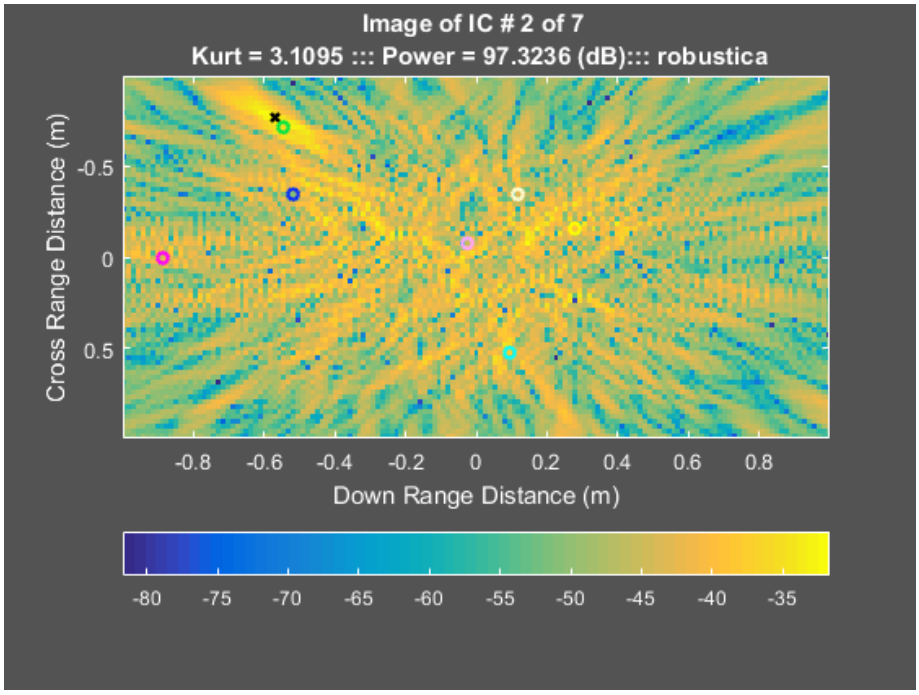

Figure 12: High Kurtosis Scattering Element #2 of 7

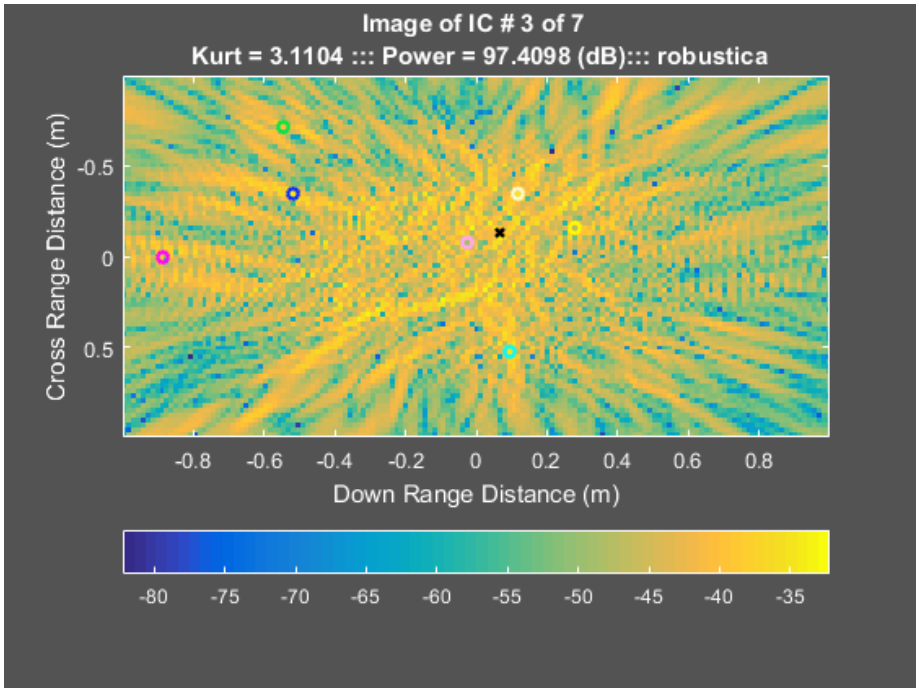

Figure 13: High Kurtosis Scattering Element #3 of 7

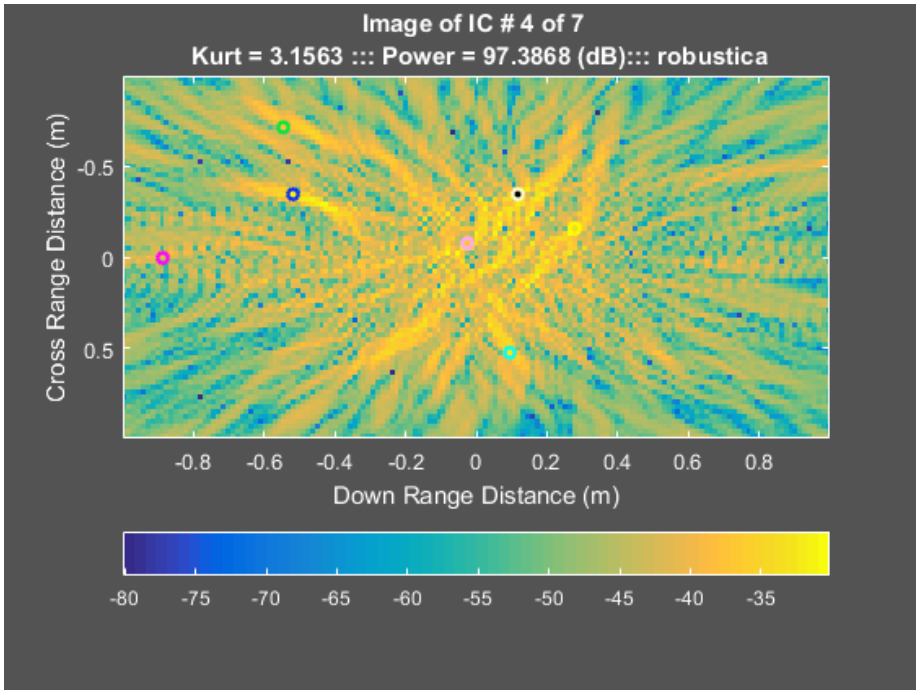

Figure 14: High Kurtosis Scattering Element #4 of 7

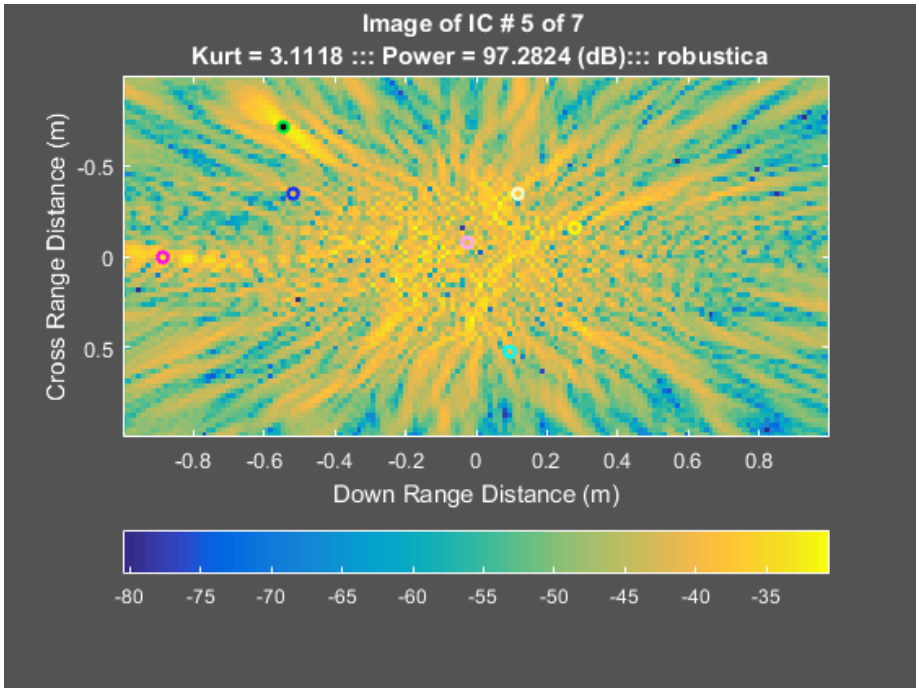

Figure 15: High Kurtosis Scattering Element #5 of 7

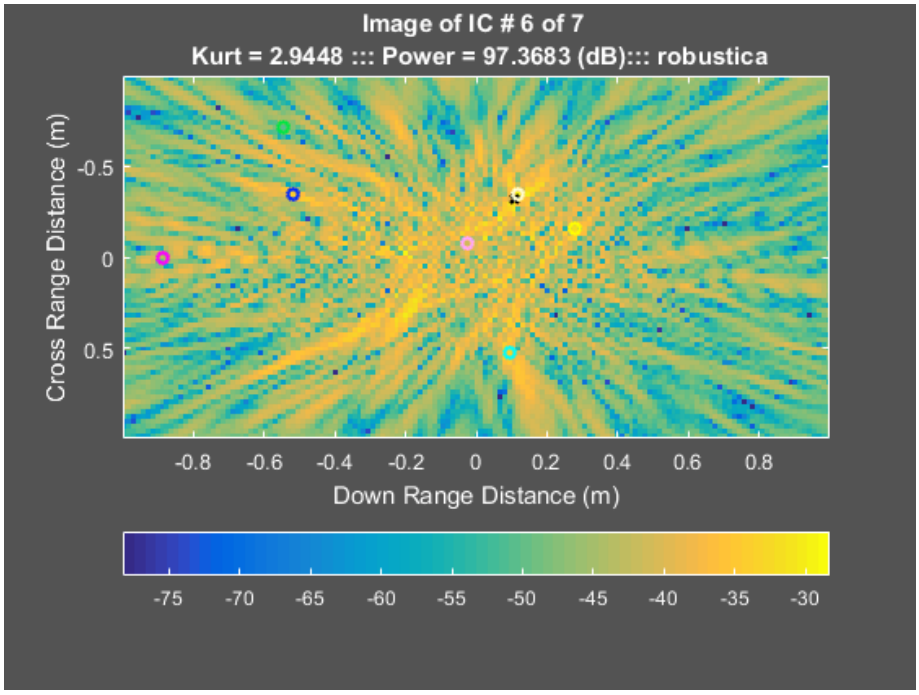

Figure 16: High Kurtosis Scattering Element #6 of 7

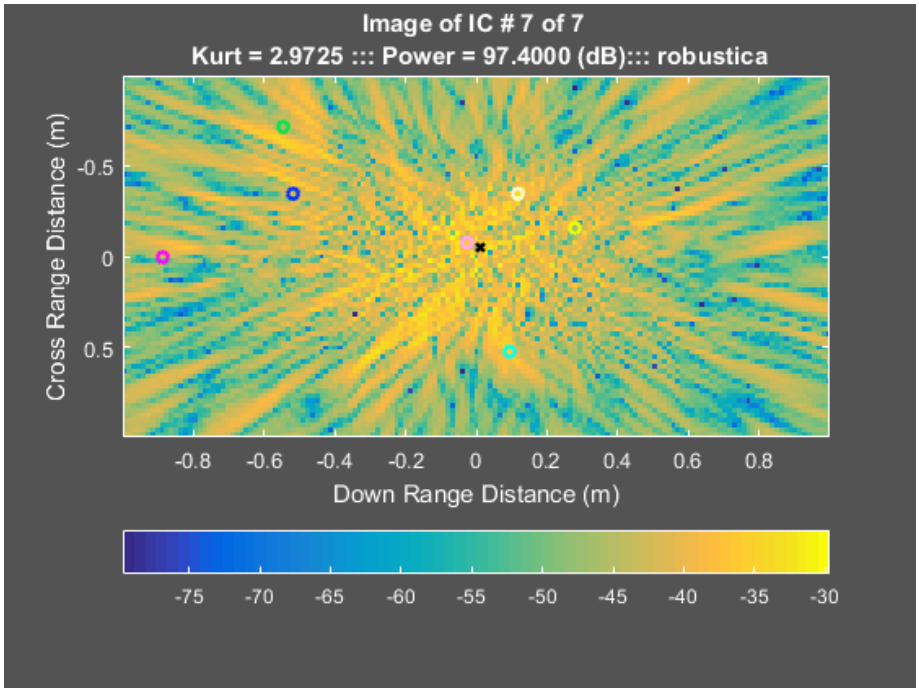

Figure 17: High Kurtosis Scattering Element #7 of 7

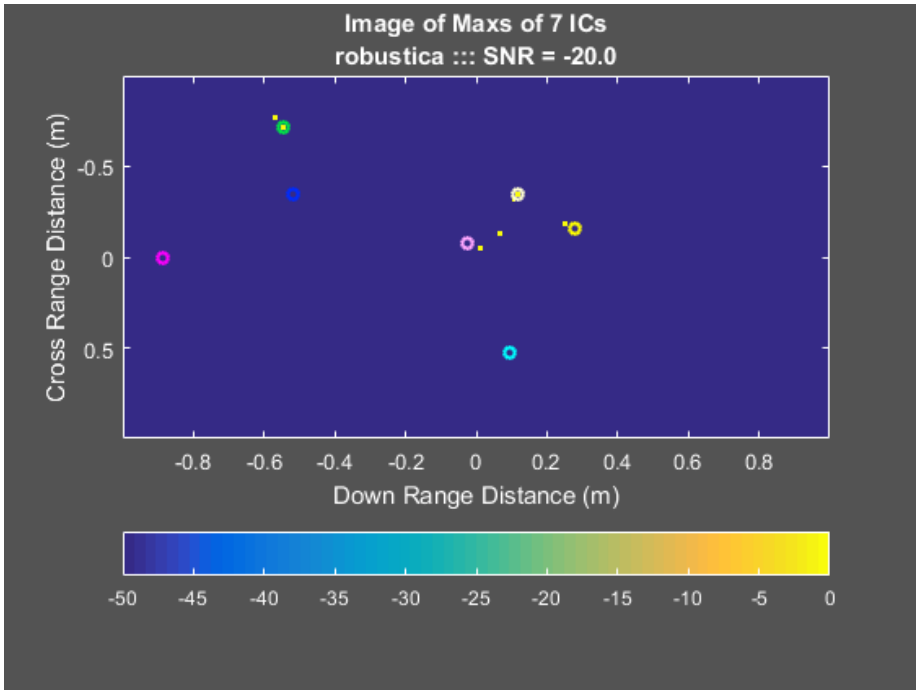

Figure 18: Composite Maximum Image of 7 Extracted Components

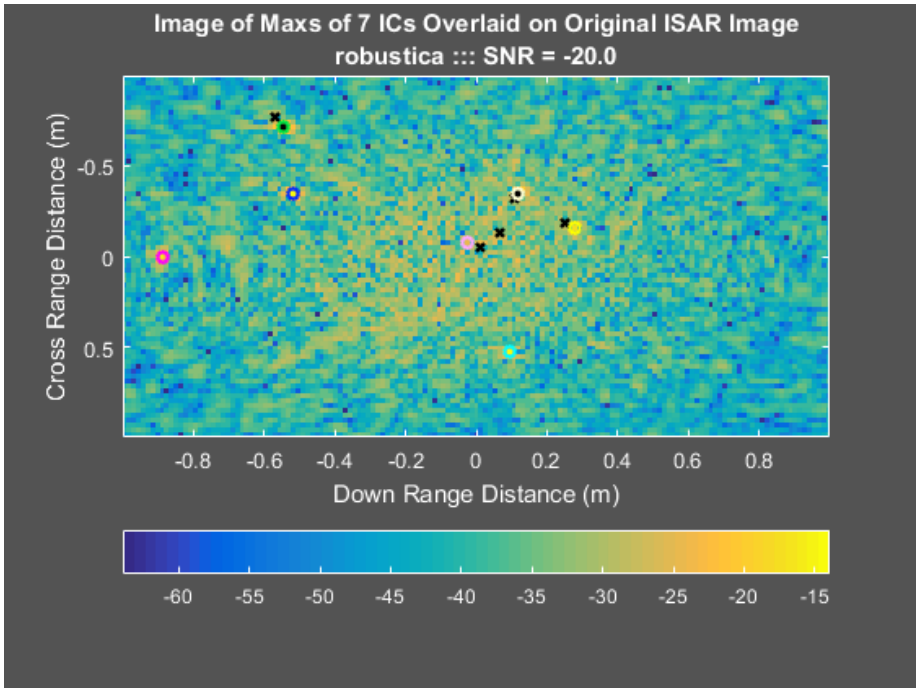

Figure 19: Composite Maximum Overlaid on Original ISAR Image of 7 Scatterers

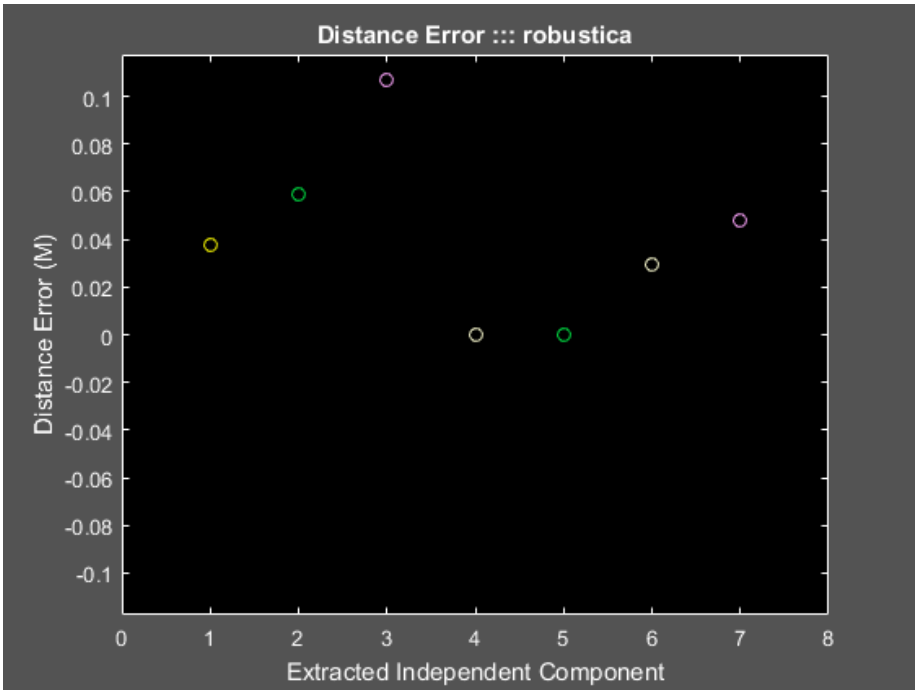

Figure 20: Distance Errors for 7 Detected Scattering Centers

This section has illustrated that the highly kurtotic elements of a scattering mixture have a strong correlation to the individual scattering elements within the mixture. This evidence points to an affirmative conclusion for the hypothesis proposed by the work. Also shown in this section was the application of the novel beamforming technique proposed by this work. The performance of the method was shown to be negatively impacted by low signal-to-noise ratio in the radar scattering data.

The distance error metric was introduced and the method of associating detections with known scattering locations was explained. The following sections will apply these metrics to all of the permutations of test cases described by the parameters in Table 1.

Parametric Study of Location Error

This section of the manuscript is dedicated to examination of the residual location error associated with the scattering centers detected through the application of a technique based on the hypothesis of this paper.

The method was tested over 72 test cases defined by all permutations of the parameters listed in Table 1. The performance of the technique, with regard to localization error, has been assessed. The analysis considered in this section is performed under a set of two additional conditions. These conditions are specified by the value of the limit applied to maximum allowable target location error (TLE). Beyond the parameterized limit chosen for each case, the association between a detected scatter and a known scatterer is considered invalid and is thus not considered a detection. The set of results below will consider two levels of maximum TLE, 0.075 m and 0.025m. The MATLAB for generating the parametric study results is included in Appendix A of this manuscript.

The first two figures below, Figure 21 and Figure 22, show how the target location error varies as a function of SNR for the 0.075 m maximum TLE condition. Figure 21 incudes a linear least squares model along with the 95% confidence bounds. Examination of this figure shows that there is a significant degree of inverse correlation between target location error and SNR. The next chart, Figure 22, shows a parameterized view of the same data. This chart shows how the relationship between TLE and SNR varies for each of the test conditions. This chart also shows that for the -30 dB SNR test case there are very few detections. Further examination of this will be discussed in the next section of this manuscript.

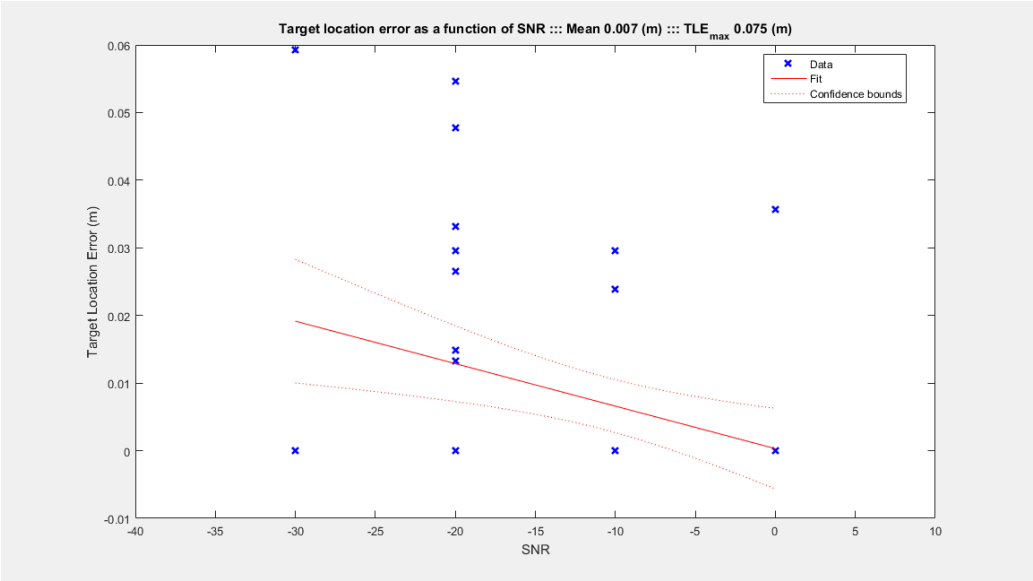

Figure 21: Target location error as a function of SNR (TLEmax = 0.075 m)

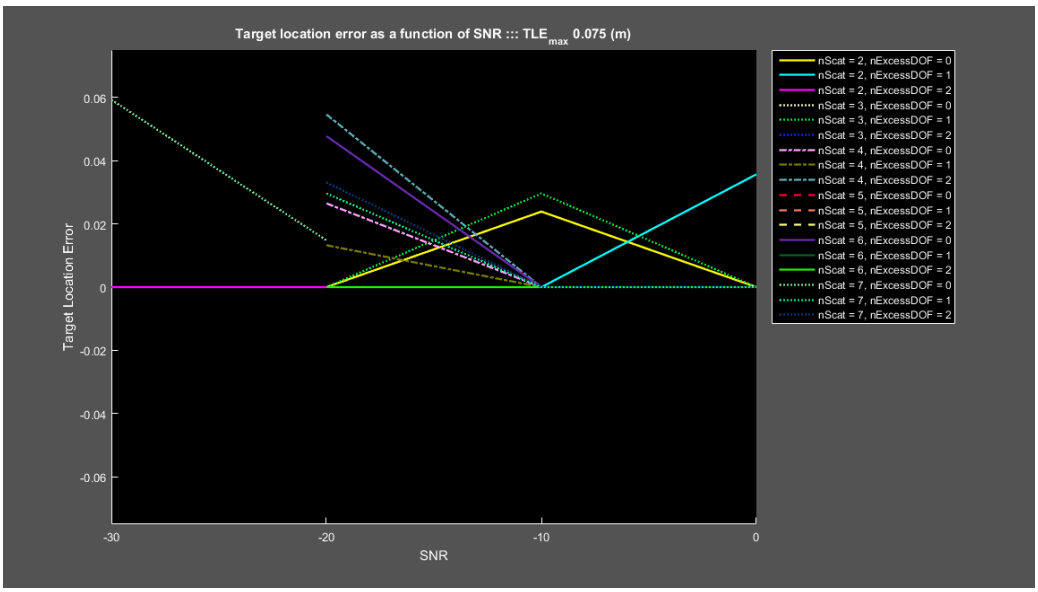

Figure 22:Target location error as a function of SNR (Parameterized) (TLEmax = 0.075 m)

The next two figures, Figure 23 and Figure 24, show the relationship between TLE and SNR given a maximum allowable TLE of 0.025 m. Examination of Figure 23 shows that there is some degree of inverse correlation between target location error and SNR, however the correlation in this case is not significant to the level of 95% confidence. This lack of significance can be determined by the fact that a horizontal line, which would represent a correlation of zero, can be drawn between the confidence bounds shown on the chart.

These figures appear to contain very few data points. The apparent lack of detections is due to the overlay of multiple data points with zero target location error. There are actually 29 data points on this plot. The conclusion that can be drawn from the observation of these charts is that the TLE, for a significant portion the data points in this study, is zero which indicates that 40% of the test cases produced precise localization of known scatterers. The probability of detection for the various test conditions is examined in the next section of this manuscript.

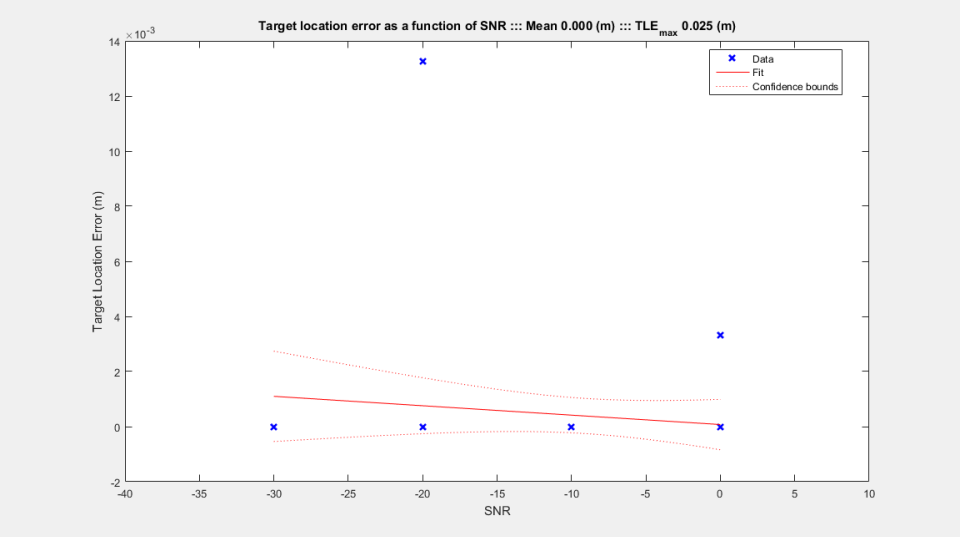

Figure 23:Target location error as a function of SNR (TLEmax = 0.025 m)

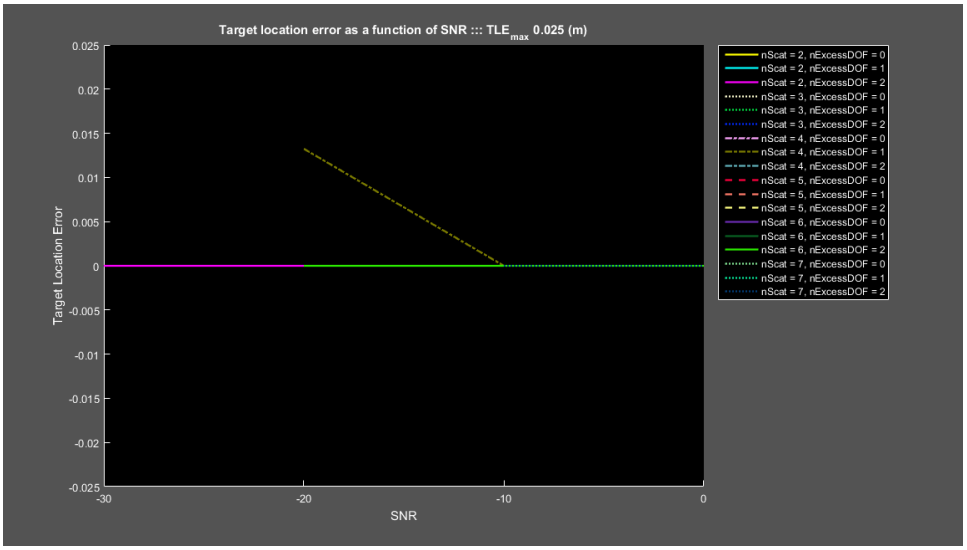

Figure 24: Target location error as a function of SNR (Parameterized) (TLEmax = 0.025 m)

The following set of plots depict the relationship between TLE and excess degrees of freedom. The technique put forth in this paper was applied to a variety of synthetic scattering targets. The synthetic data which was partitioned into subapertures which yielded excess degrees of freedom ranging from 0 to 2. Permutations of the various parameters of this study resulted a total of 72 test conditions. The 0.025 m and 0.075 m conditions are shown in Figure 25 and Figure 26 respectively. For both of the allowable TLE conditions there is no significant correlation between target location error and excess degrees of freedom. This is significant in the sense that overestimating the number of sources to be extracted does not have a significant impact on TLE performance.

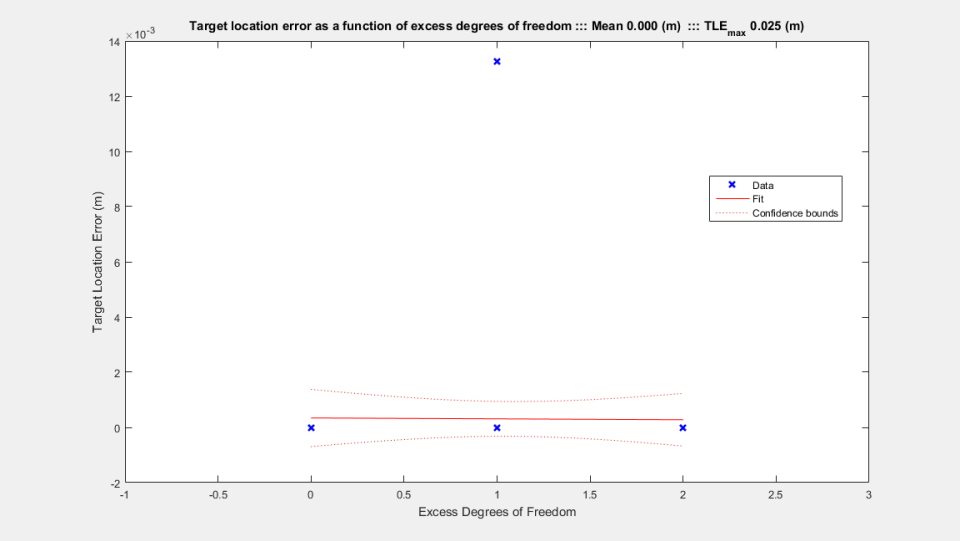

Figure 25: Target location error as a function of excess degrees of freedom (TLEmax = 0.025 m)

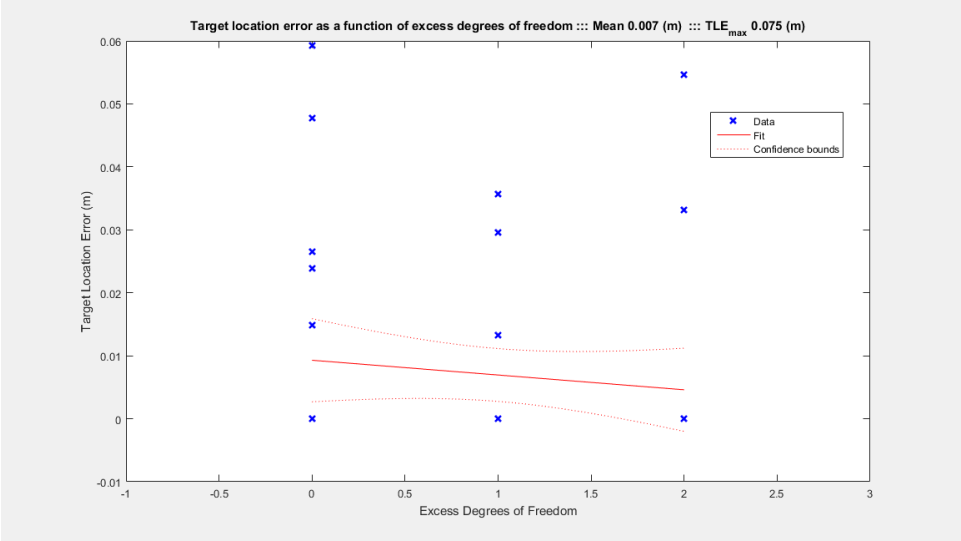

Figure 26: Target location error as a function of excess degrees of freedom (TLEmax = 0.075 m)

This section has presented the performance of the application, of a technique based on the hypothesis of this work, relative to target location error. It was shown that a larger allocation of maximum TLE results in a greater number of detections with a

larger spread in the location error. It was also shown that when using this technique there is a degree of inverse correlation between target location error and SNR. When the technique is used with a smaller allocation of maximum allowable TLE, the approach produces a detection with a target location error of zero in 58.3% of the test cases. This result provides additional evidence for the affirmative conclusion of the hypothesis that the high-kurtosis elements within a mixture of scattering data are correlated to the scattering centers within the scene.

Parametric Study of Probability of Detection

This section of the paper is directed at characterizing the ability of this technique to correctly extract known synthetic scatterers under a variety of signal-to-noise ratio conditions and excess degrees of freedom.

The signal processing technique of this paper is applied to a set of synthetic data which contains a varying number of scatterers ranging in number from two to seven. Each of these data sets are subjected to four noise levels of -30, -20, -10 and 0 dB. Under each of these noise conditions the data is partitioned into progressively greater numbers of subapertures which yield excess degrees of freedom ranging from 0 to 2. This study examined a total of 72 test conditions. For all of these various permutations, the probability of detection metric was computed. The probability of detection metric is computed using a hard limit for maximum target location error (TLE). Beyond this parameterized limit the association between the detected scatter and the known scatterer is assumed to be invalid and is thus not considered a detection. The set of results below will consider two levels of maximum TLE, 0.075 m and 0.025m. It will be shown that higher precision target localization leads to a lower probability of detection due to the tighter association window.

The first set of results are those for which a 0.075 m maximum allowable TLE was assumed. The results of this condition are shown in Figure 27 through Figure 28. The first two figures below depict the probability of detection as a function of SNR. These charts show that there is significant correlation between the probability of detection and SNR. The next pair of charts illustrate the relationship between probability of detection and excess degrees of freedom. From this set of plots, it is possible to conclude that probability of detection is not a strong function of the number of excess degrees of freedom. The lack of correlation between excess degrees of

freedom and probability of detection implies that overestimation of the number of sources to extract has little impact on the probability of detection performance.

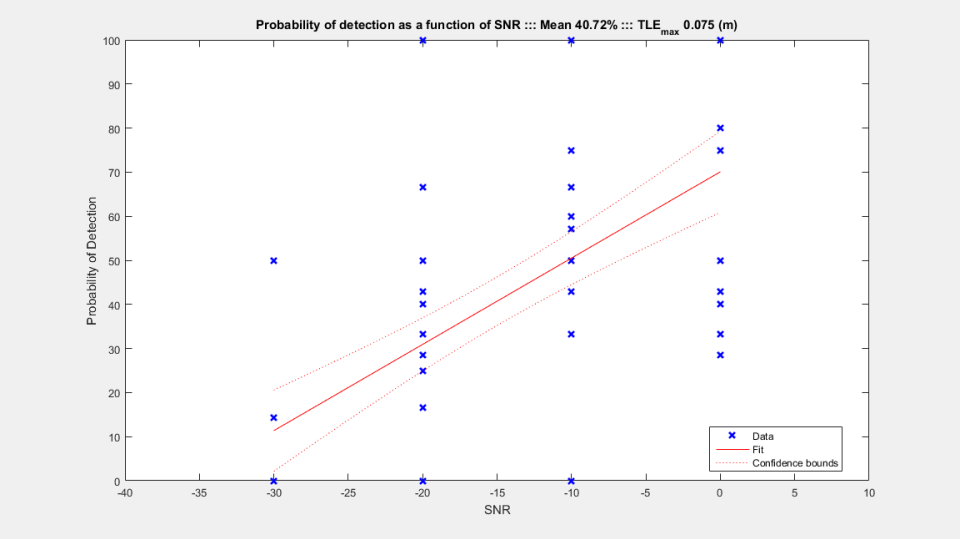

Figure 27: Probability of detection as a function of SNR (TLE_{max} = 0.075 m)

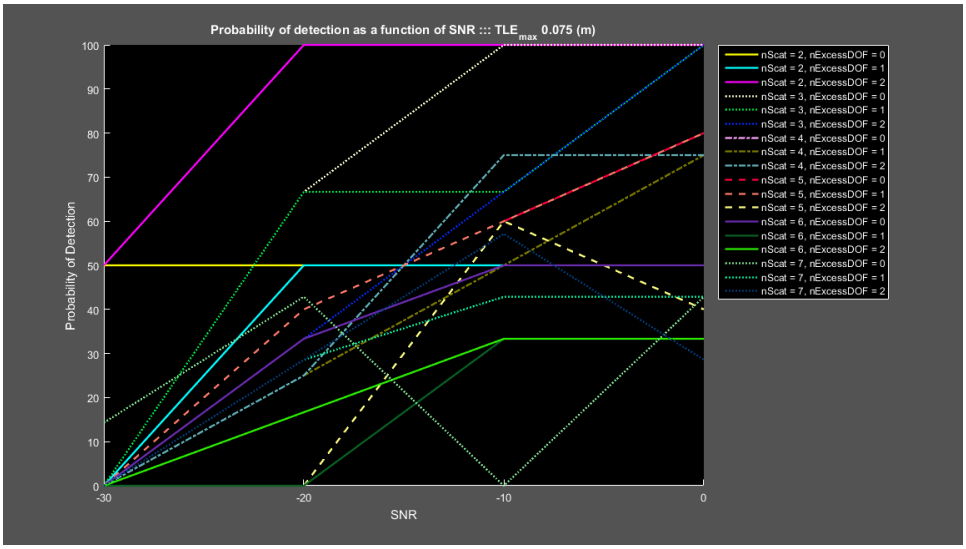

Figure 28: Probability of detection as a function of SNR (Parameterized) (TLE_{max} = 0.075 m)

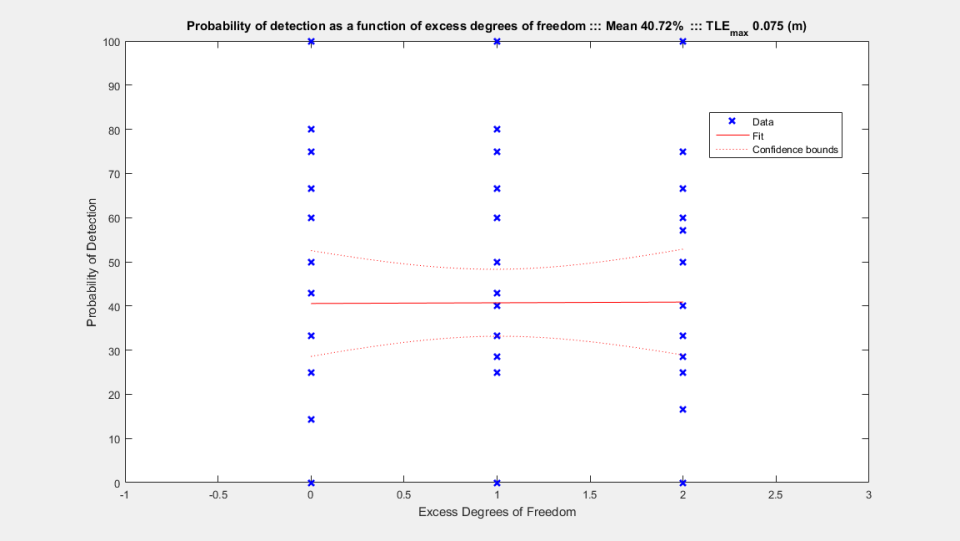

Figure 29: Probability of detection as a function of excess degrees of freedom (TLE_{max} = 0.075 m)

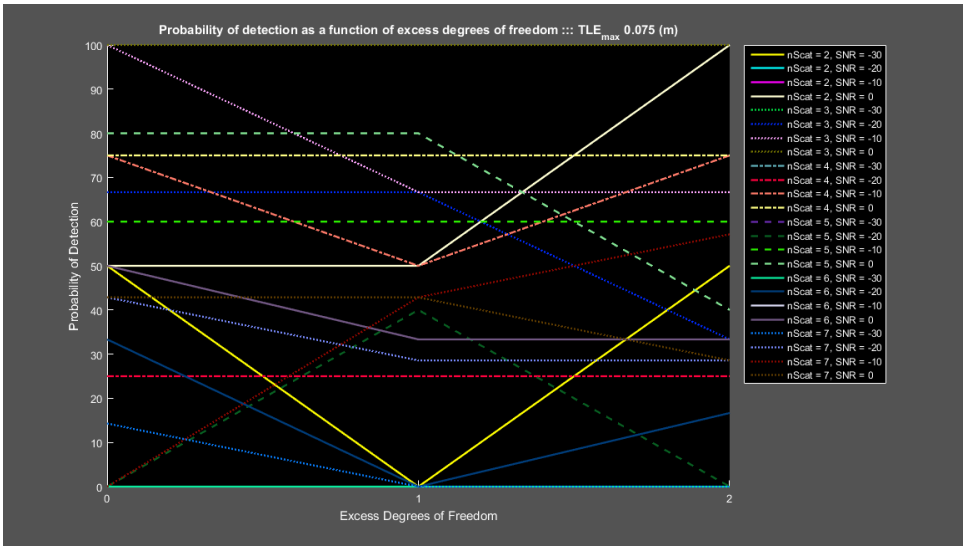

Figure 30: Probability of detection as a function of excess degrees of freedom (parameterized) (TLE_{max} = 0.075 m)

Figure 31 and Figure 32 show the probability of detection performance under the condition of a 0.025 m maximum allowable TLE. These charts maintain the

correlation properties of the previous case. The significant difference is the reduction in the average probability of detection from the previous case which has an association window which is 9 times larger. The condition of 0.075 m TLE_{max} produced an average probability of detection of 40.7% and the 0.025 m TLE_{max} condition yielded an average probability of detection of 26.5%.

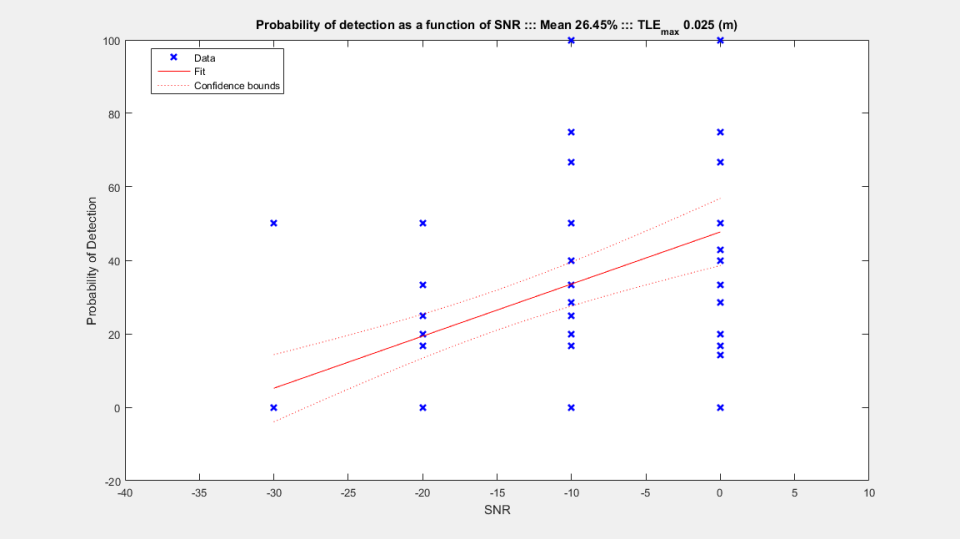

Figure 31: Probability of detection as a function of SNR (TLEmax = 0.025 m)

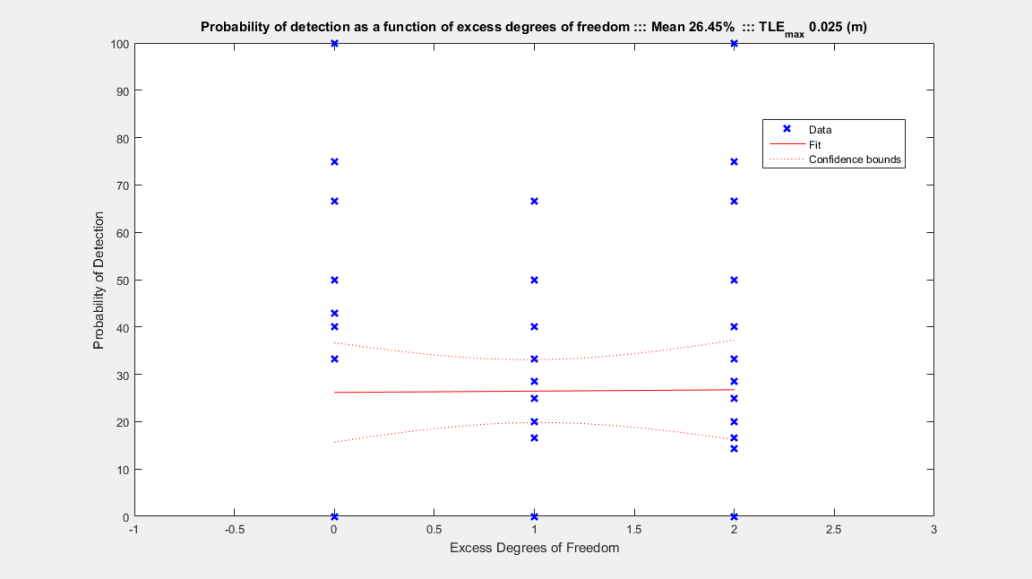

Figure 32: Probability of detection as a function of excess degrees of freedom (TLEmax = 0.025 m)

This portion of the paper has shown that the application of the hypothesis that: high-kurtosis elements of a radar scattering mixture are correlated to the scattering centers within the observation field. The application detailed in this paper was tested over 72 scenario parameter permutations as well as two performance requirement conditions which limited the allowable TLE for scatterer association to 0.075 m and 0.025 m. The performance of this novel technique was examined with regard to probability of detection. For this analysis probability of detection was defined as the ratio of detected scatterers with valid associations to the total number of known scatterers within the scene. Under these conditions and criteria, the larger allowable TLE produced an average probability of detection of 40.7% and the smaller condition yielded an average probability of detection of 26.5%. These results are colored by the very poor performance in the -30 dB SNR case. Exclusion of the results for this test

condition results in an average probability of detection of 52.2% and 33.4% for the larger and smaller maximum TLE condition respectively. These results are summarized in Table 2.

Table 2: Summary of Simulated Scattering Results

	TLEmax = 0.075		TLEmax = 0.025	
	all	exclude (SNR = -30)	all	exclude (SNR = -30)
Probability of Detecting No Scatterers	9.72%	12.96%	4.17%	5.56%
Probability of Detecting All Scatterers	26.39%	7.41%	40.28%	24.07%
Average Probability of Detection	40.72%	52.18%	26.45%	33.42%
Probability of Detection with zero TLE (inc. duplicate detections)	58.33%	74.07%	58.33%	74.07%

As shown in the previous section, the TLE associated with the valid detections exhibited very good performance with a majority of detections resulting in zero error. The probability of detection shown here could likely be improved as noted in the Future Works section of this manuscript. However, the association probability shown here has proven that there is a correlation between high kurtosis elements of radar scattering data and the scattering centers within the scene.

Performance Comparison to MUSIC for Closely Spaced Scatterers

This section of the manuscript is dedicated to examining the capability of the Isolated Scatterer Method to precisely localize closely spaced scatterers. A comparison is made to the performance of the MUSIC algorithm under the same test conditions.

This numerical experiment simulated five closely spaced scatterers under three noise conditions. The set of configurations for the numerical experiments is detailed in Table 3. The MATLAB used in the generation of these results can be found in Appendix B of this thesis.

Table 3: Closely Spaced Scatterer Experiment Conditions

Separation	1.25	2.5	3.75	cm
SNR	0	-10	-20	dB

A simulation was performed using a 4 GHz collection bandwidth and 32 angle samples over the range of -80 to 80 in the azimuthal direction. This results in a null-to-null resolution of approximately 4.15 cm. Given that ½ of the null-to-null beam width is 2.075 cm, the two smallest spacing conditions result in irresolution for the scatterers. These conditions require super-resolution to separate the scatterers.

The MUSIC algorithm used for this set of experiments was implemented according to the formulations in Chapter 2. For the results shown below, the MUSIC algorithm is given the precise number of scatterers. Additional test cases were completed the size of the signal subspace was allowed to be greater than the actual number of sources. The results from these test cases also support the conclusions below.

For this set of experiments, the Isolated Scatterer Method is implemented in the same manner as the previous set of experiments and is described in Chapter 3. For the results shown below, the excess degrees of freedom were constrained to zero.

The primary metric used in this comparison is the Probability of Detection (Pd) with zero Target Location Error (TLE). This metric measures the likelihood of either technique to produce precise localization for the present set of scatterers.

The three figures below give graphical examples of the output from this study. Each of these figures contains three images. The top center image is that of a standard ISAR image. The bottom left image is the composite maximum view produced using the Isolated Scatterer Method. The bottom right image is the image formed using the MUSIC algorithm. The three images all contain an overlay of 5 colored circles representing the actual location of the synthesized scatterers.

Figure 33 illustrates these three image types for a scatterer separation of 1.25 cm. The ISAR image clearly shows that the unique scattering centers are irresolvable using this technique. This view of the scattering shows a large region of scattering in and around the locations of the 5 point scatterers. The image formed using the ISM shows very poor performance of this technique. This results in 20% probability of detection with zero TLE. The MUSIC image shows fairly poor performance as well, also producing a 20% probability of zero TLE.

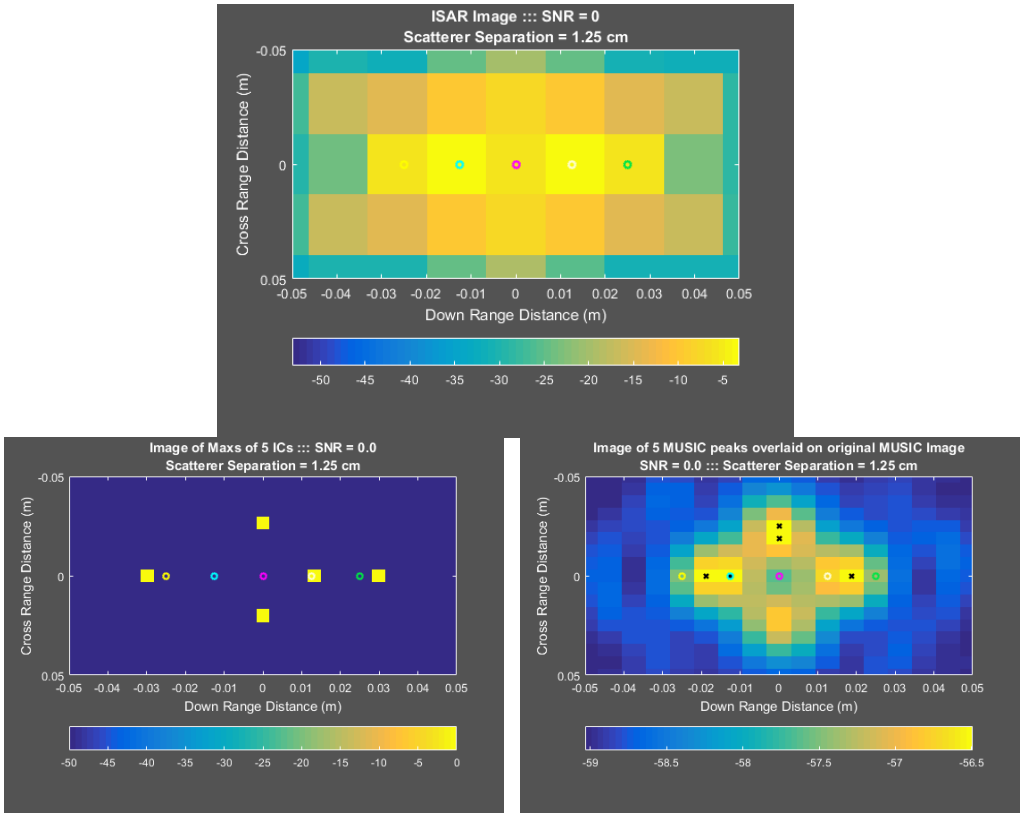

Figure 33: Closely Spaced Scatterer Results (1.25cm, 0 SNR)

Figure 34 shows a second test condition in which the scatterer are slightly further apart at a spacing of 2.5 cm. In this case, the scatterers in the ISAR image are still irresolvable. This can again be observed as a region of scattering around the actual point scatterers. The MUSIC algorithm has performed better given the wider spaced scatterer. The probability of detection with zero TLE is now 60% for the MUSIC algorithm. The ISM did not improve and again only achieved 20% for this set of conditions.

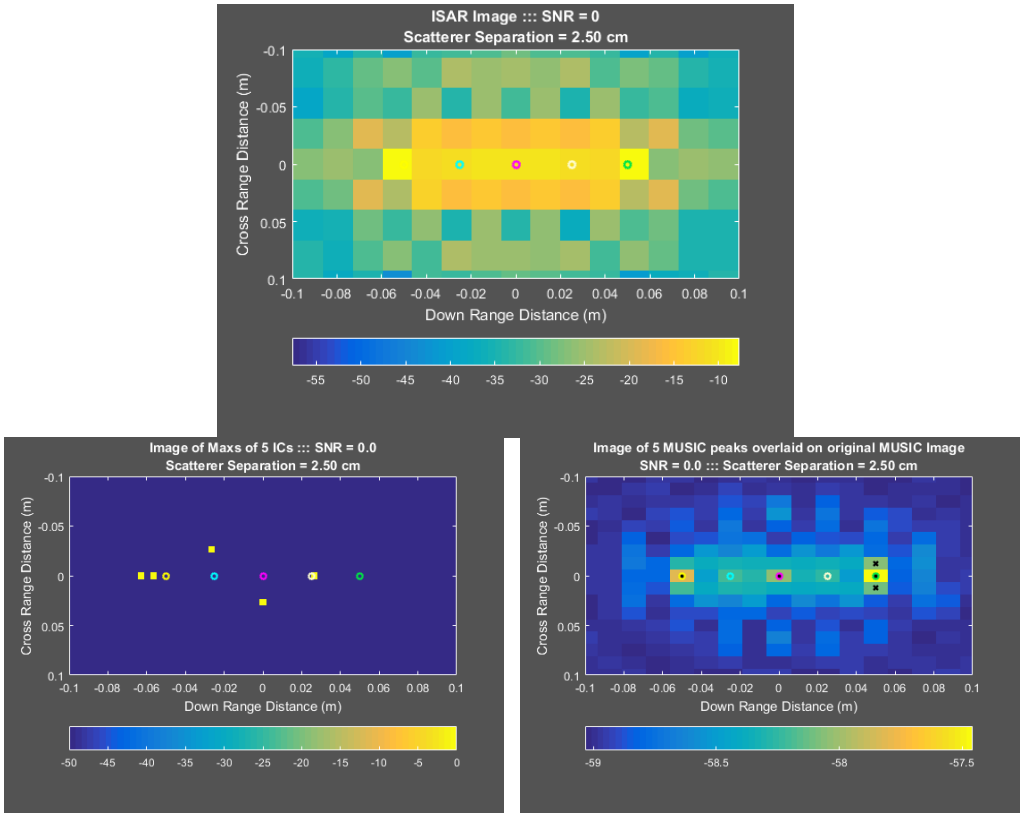

Figure 34: Closely Spaced Scatterer Results (2.50cm, 0 SNR)

For the last of the three examples, the scatterers were set to 3.75 cm. From the top image in Figure 35 it can be seen that the scatterers are now resolvable in the standard ISAR image. For this case, the MUSIC technique has achieved 100% probability of detection with zero TLE while the Isolated Scatterer Method had zero successful detections. This drop in performance and the general poor performance for ISM could be due to the technique focusing the subapertures on areas of interaction among the scatterers. The results show there is clearly an advantage to using MUSIC over ISM in the closely spaced scatterer condition.

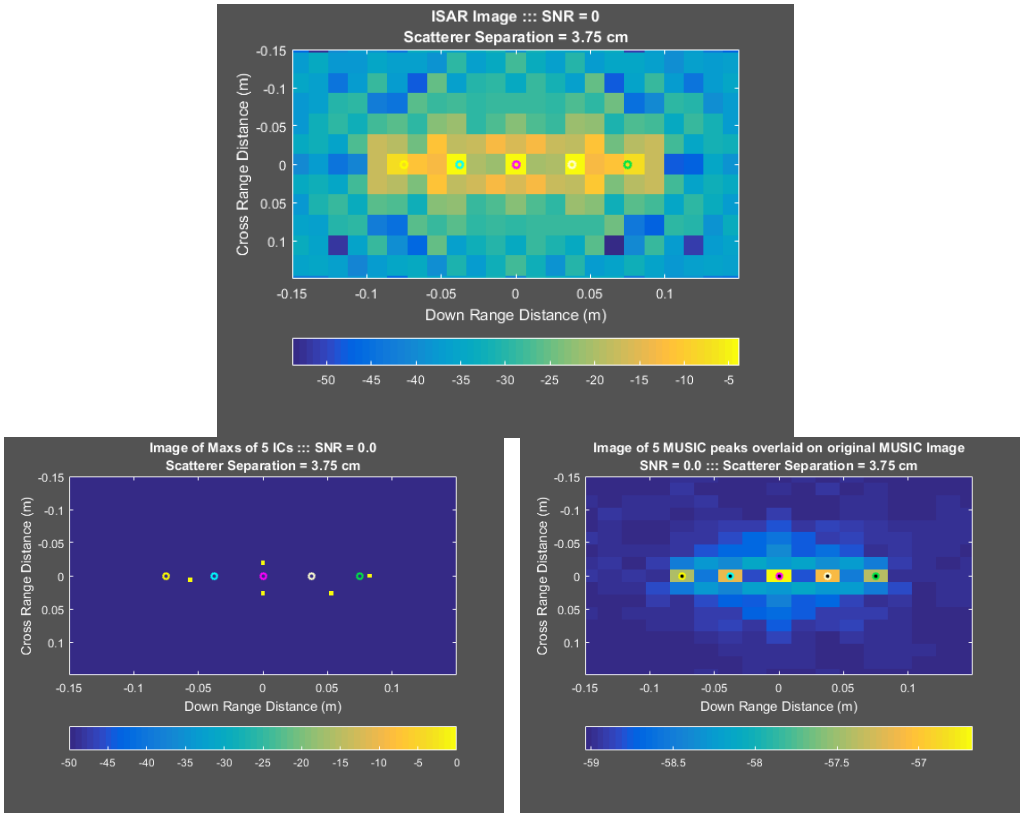

Figure 35: Closely Spaced Scatterer Results (3.75cm, 0 SNR)

Table 4 shows the full set of results for this study. It can be concluded that the Isolated Scatterer Method is not well suited for localization of closely spaced scatterers. The MUSIC algorithm outperforms in nearly all of the cases. For conditions requiring super-resolution the new technique does not show much promise.

Table 4: Results of Closely Spaced Scatterer Study

Probability of Zero TLE detection		SNR					
		0		-10		-20	
		MUSIC	ISM	MUSIC	ISM	MUSIC	ISM
Separation	1.25cm	20%	20%	20%	0%	20%	0%
	2.5cm	60%	20%	60%	0%	20%	40%
	3.75cm	100%	0%	100%	0%	60%	20%

Application of Method to Measured Data Sets

The focus of this section is the examination of the results produced when the technique is applied to several turntable ISAR measurement data sets. The data sets analyzed in this section were gathered using the turntable ISAR system at the Wave Scattering Research Laboratory of the University of Texas at Arlington. The target sets were interrogated using microwaves in 4-8 GHz frequency range with 801 points. The target sets are subjected to a single polarization which is either vertical or horizontal. The polarization used for each data set is noted in the description of the target set for each of the test cases. A single antenna elevation angle of 60 degrees was used and the azimuthal angle was stepped between 0 and 360 degrees in 5 degree increments. Figure 36 shows a side view of the measurement setup inside the anechoic chamber.

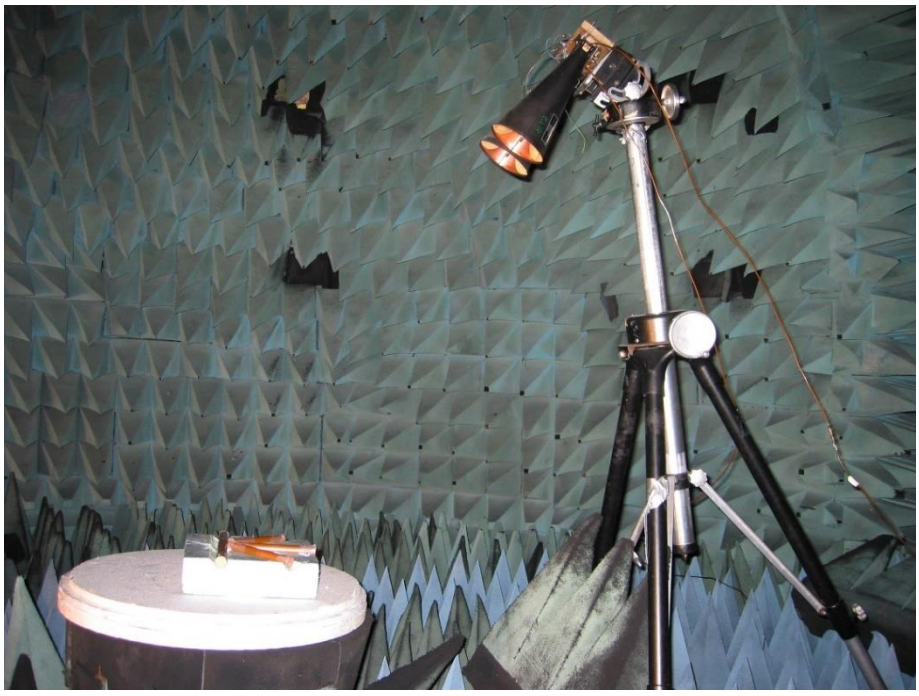

Figure 36: Photo of Measurement Set-up (Side View)

Measurement Data Set 1: Four cylinders in half-asterisk formation

This first data set is made up of four conductive cylinders in a “half-asterisk” formation. To maintain the configuration while performing the rotation of the turntable, the entire target set was placed on a template of nonconductive foam. This configuration is pictured in Figure 37. Vertical polarization was used to collect data for this target set.

Shown in Figure 38 is a standard ISAR image of the target set. This image has been augmented with red lines to give an approximate reference for the position of the target in the image. This figure shows a robust area of scattering around the longest cylinder in the configuration. Areas of scattering can also be seen in the regions near the ends of the three shorter cylinders. The hourglass shape of the scattering region around the longest cylinder can be speculated to be multiple bounce scattering caused by the interaction of the shorter cylinders with the longer one. The scattering, which is produced by a corner reflection, tends to be visible over a range of angles centered around the vertex of the corner. This observation will be relevant when considering the images formed from the high-kurtosis sources.

Figure 37: Photo of Target for Measurement Dataset 1

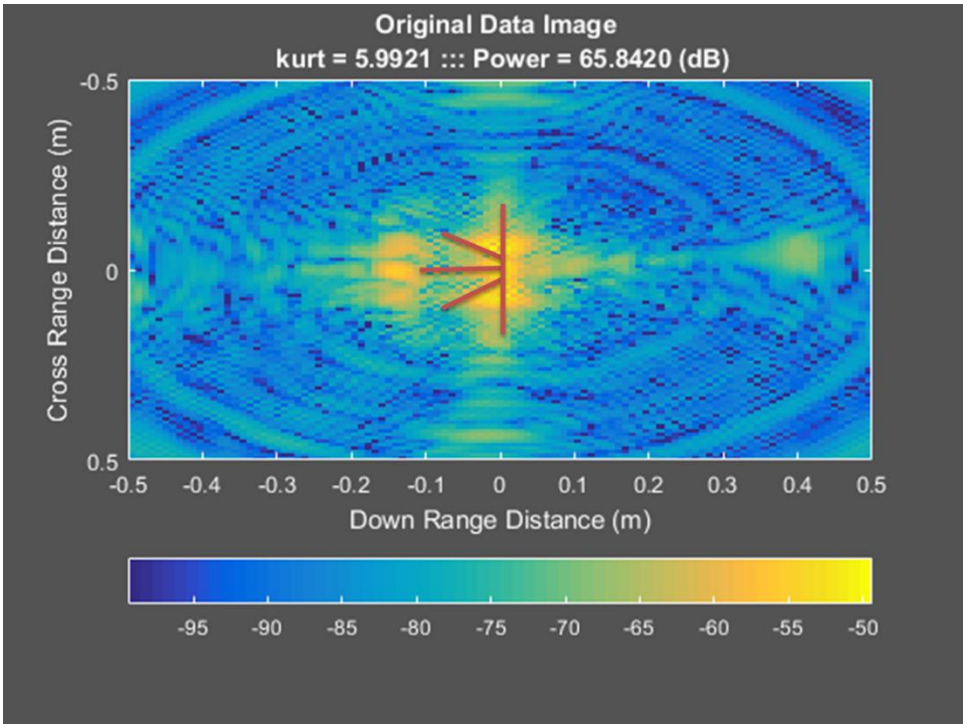

Figure 38: Standard ISAR Image of Measurement Dataset 1

The signal processing technique based on the hypothesis of this paper in applied with a range of degrees-of-freedom between three and ten. Below are the eight composite maximum images formed from these cases. These are shown in Figure 39 through Figure 46. The first four of these images show that the technique has produced scattering centers localized along the center of the longest cylinder. The last seven images of this show scattering centers that have been localized to positions which are associated with the middle cylinder of the shorter cylinder set.

The intersections of the cylinders form a geometry conducive to double bounce scattering that is typical of a dihedral corner reflector. This geometry presents a range of angles that could result in reflected electric field in the direction of the receiving antenna after being reflected from the surface of two adjacent cylinders. The last four images in this series show a clustering of estimated scattering centers in areas which are likely places for multiple-bounce scattering. The presence of double bounce scattering could be a possible explanation for the clustering of the estimated scattering centers in these images. Further study of the performance of this technique in under multiple bounce scenarios is a potential area of future research.

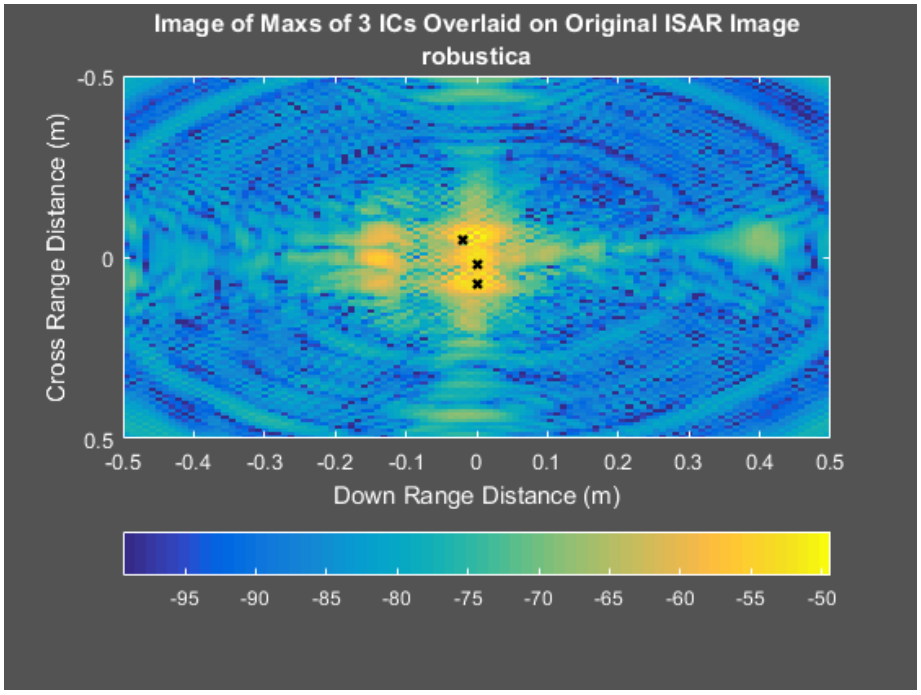

Figure 39: Measured data set 1 results (three degrees-of-freedom)

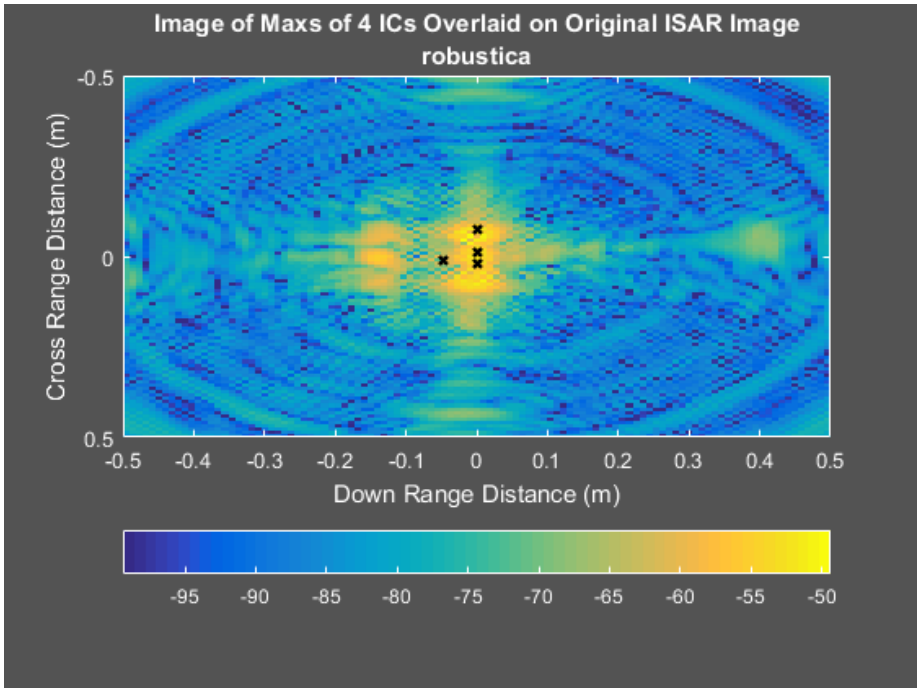

Figure 40: Measured data set 1 results (four degrees-of-freedom)

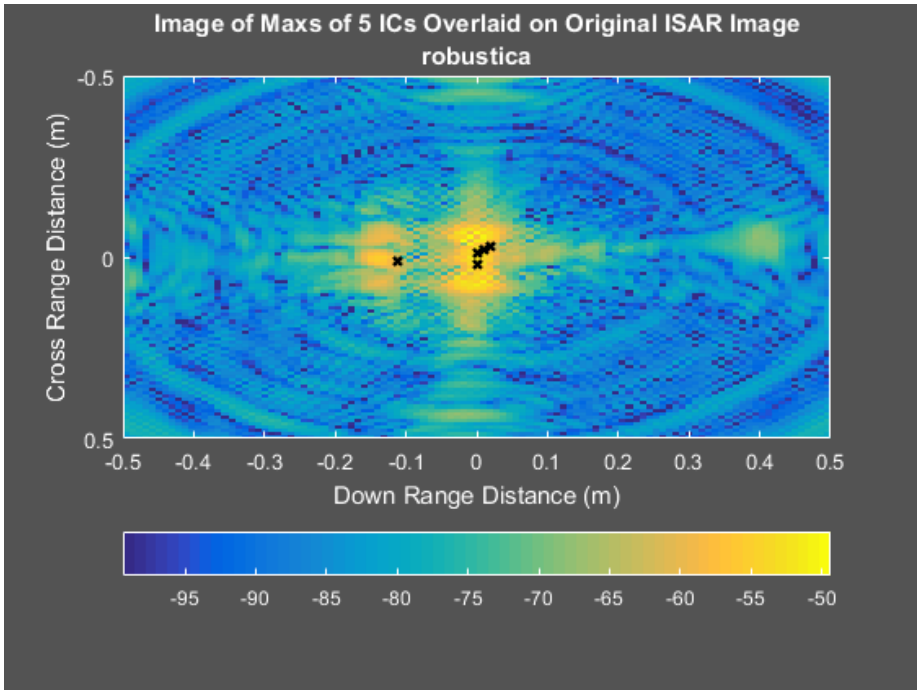

Figure 41: Measured data set 1 results (five degrees-of-freedom)

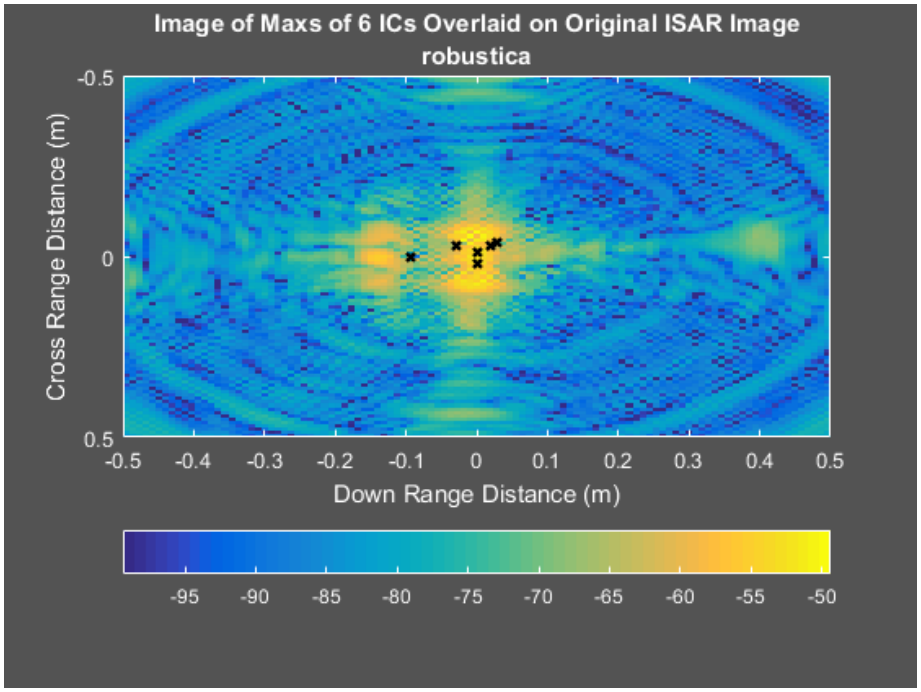

Figure 42: Measured data set 1 results (six degrees-of-freedom)

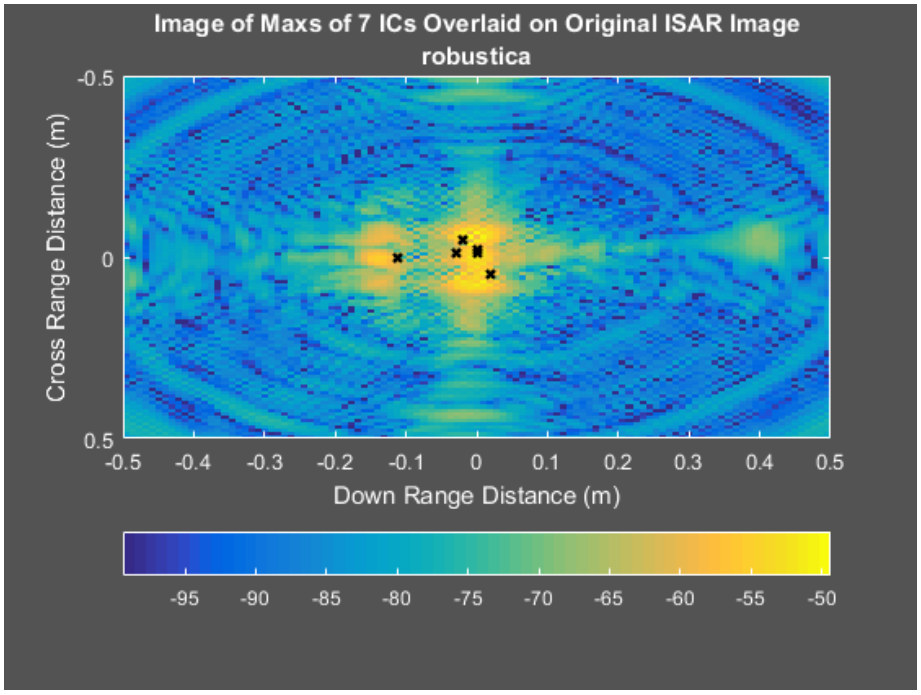

Figure 43: Measured data set 1 results (seven degrees-of-freedom)

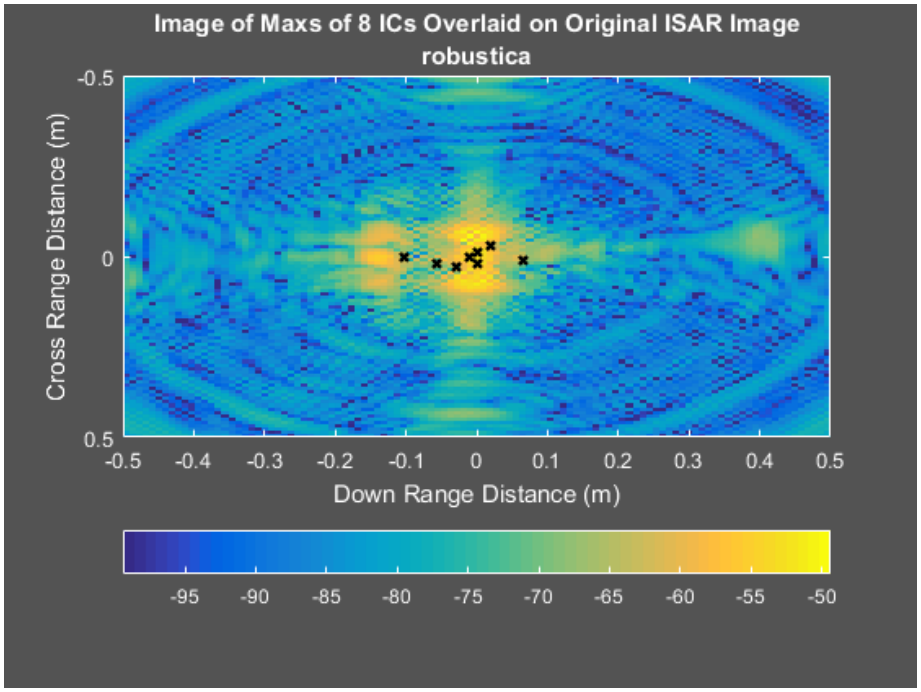

Figure 44: Measured data set 1 results (eight degrees-of-freedom)

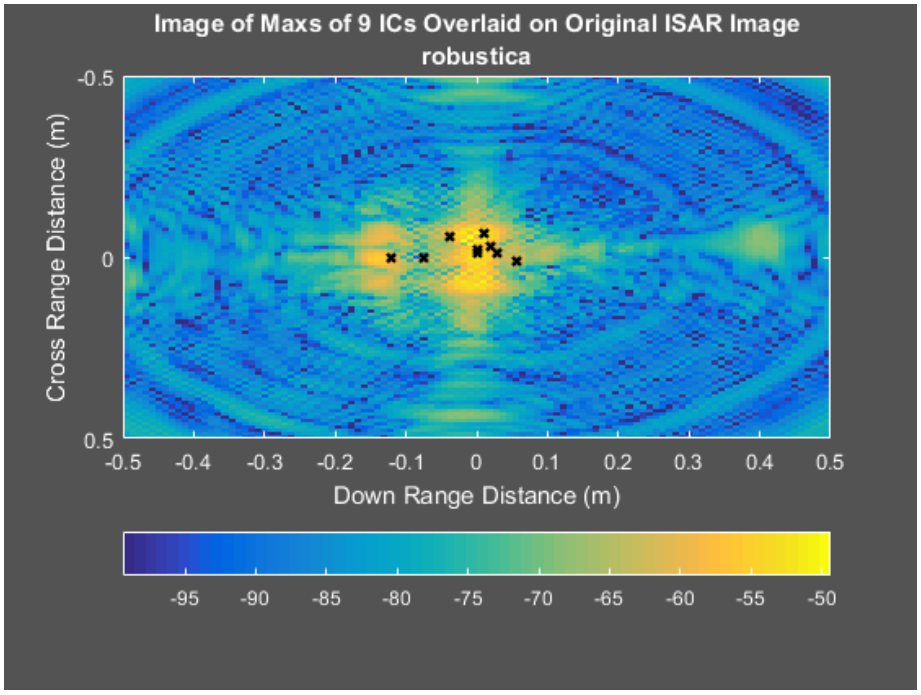

Figure 45: Measured data set 1 results (nine degrees-of-freedom)

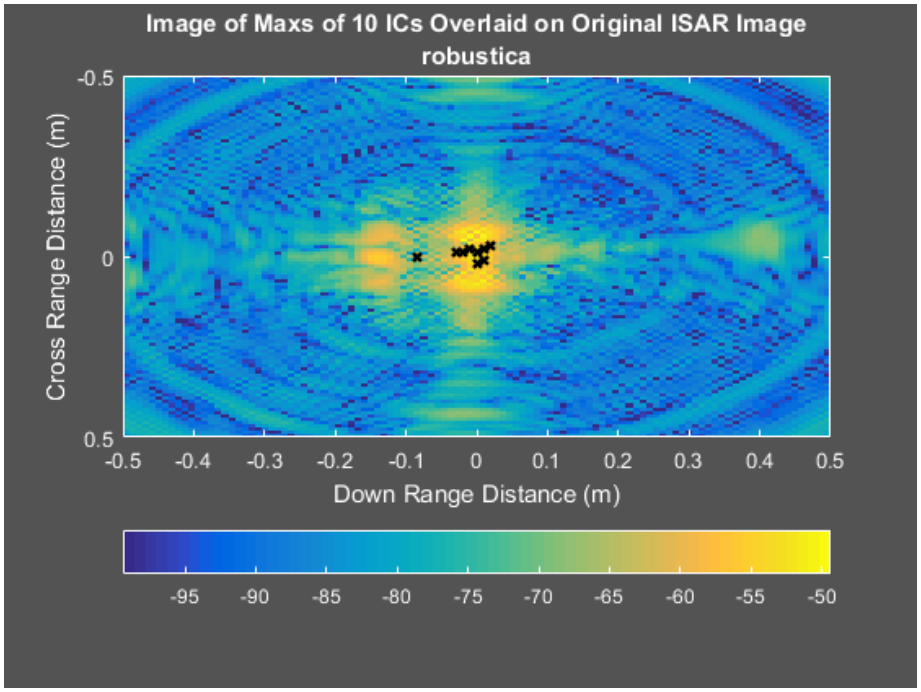

Figure 46: Measured data set 1 results (ten degrees-of-freedom)

The application of the concept of this thesis to the first measurement data set has shown possibility for extraction dimensional information for a target. The extracted scattering centers may yield detail on the length and width of the circular cylinders which comprise this complex target. It has also shown promise for the ability of the method to extract higher-order scattering such as double bounce. Future research should include measurement of a target set whose dimensions and position are more precisely known.

Measurement Data Set 2: Two cylinders in T formation

This data set was comprised of two conductive cylinders laid out in a “T” formation. The data for this target was also collected using a vertically polarized wave on transmit and receive. A photographic image of the target set is shown in Figure 47. Figure 48 is a radar image formed using the standard back projection technique. This image has been augmented with lines to mark the approximate location of the cylinders in the target.

The first four images, Figure 49 through Figure 52, show that the technique has produced estimated scatterers localized near the intersection of the two cylinders and in the region which would correspond to the broadside look angle for the top of the “T”.

Figure 53 includes detected scattering centers further down on the cylinder making up the vertical leg of the “T” shape. Two scattering centers appear to be on either side of the vertical cylinder. This ability to detect the width of an object would be a particularly useful application of this technique. This aspect could be an extension of this study for future research.

The addition of degrees of freedom in the next three images, Figure 54 through Figure 56, add to the clustering of detected scattering centers near the intersection of

the cylinders. Also noticeable in these images is the clustering along the position of the cylinder which forms the top of the “T”.

It is not clear why the clustering does not cover as much of the vertical cylinder as it does the horizontal cylinder yet there are several points which would be associated with the vertical cylinder.

The application of the technique to this measured data set appears to have produced some spatial information which could be used to augment interpretation of the standard ISAR image.

Figure 47: Photo of Target for Measurement Data Set 2

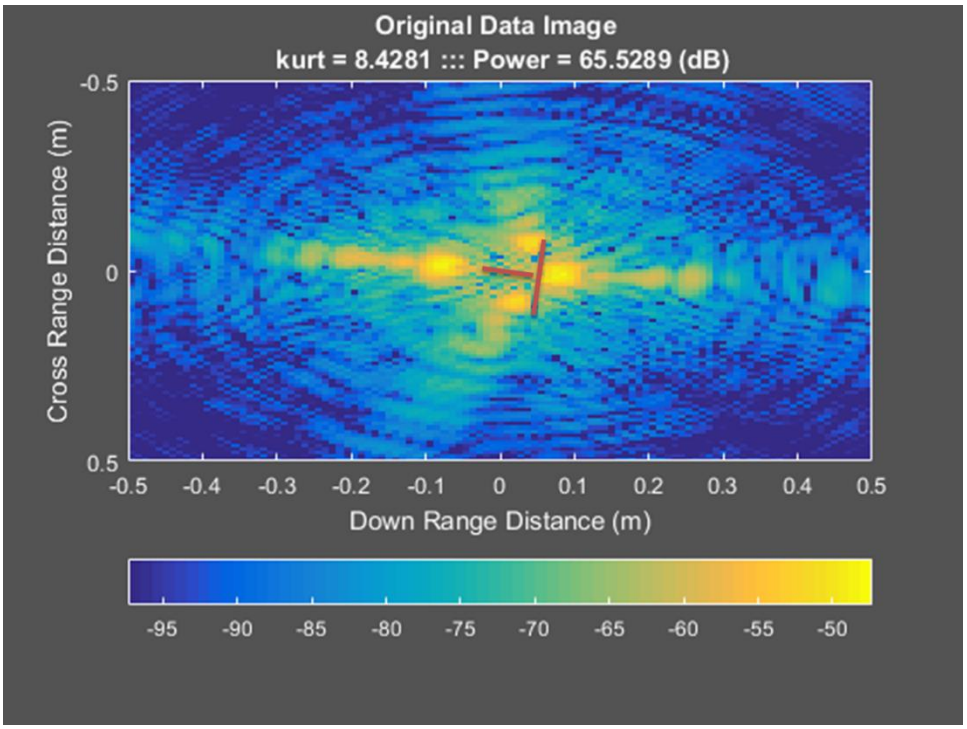

Figure 48: ISAR Image of Measurement Data Set 2

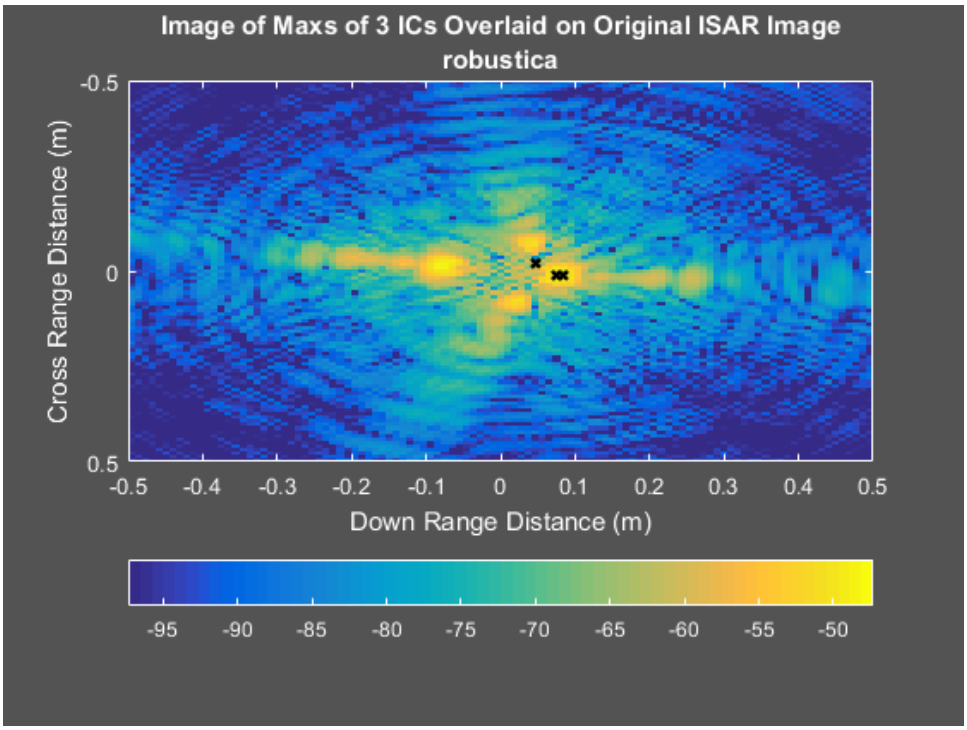

Figure 49: Measured data set 2 results (three degrees-of-freedom)

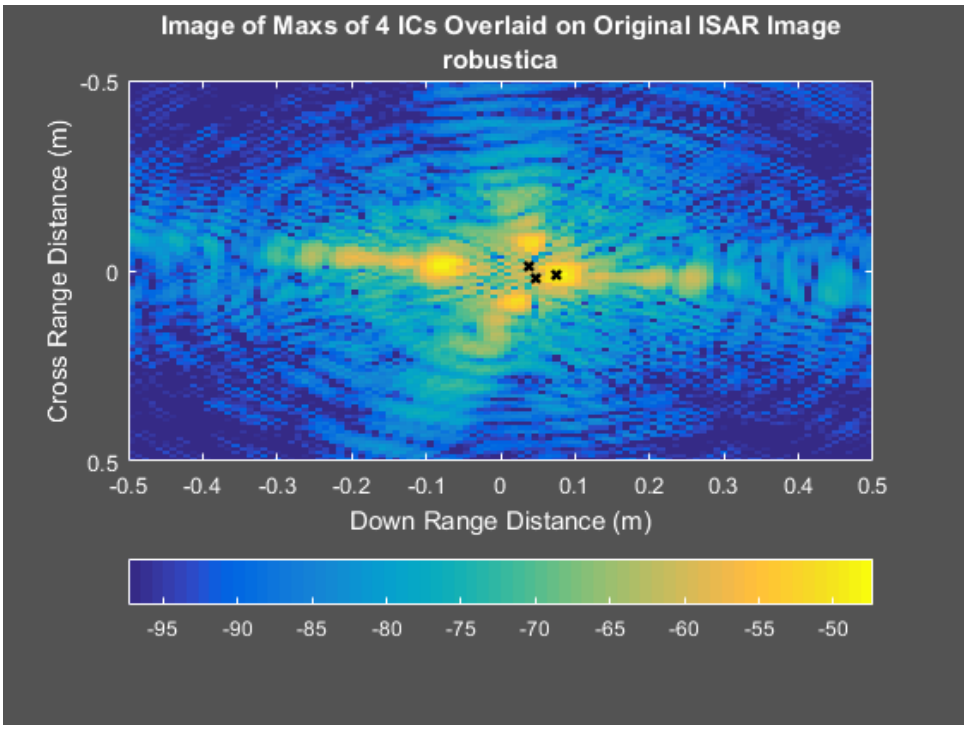

Figure 50: Measured data set 2 results (four degrees-of-freedom)

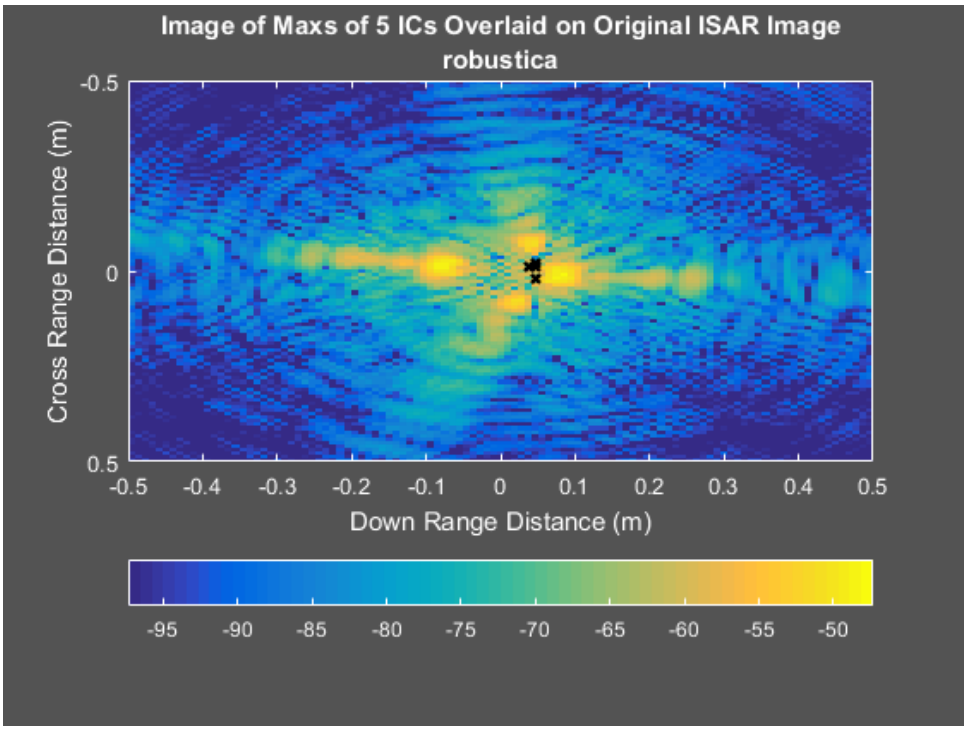

Figure 51: Measured data set 2 results (five degrees-of-freedom)

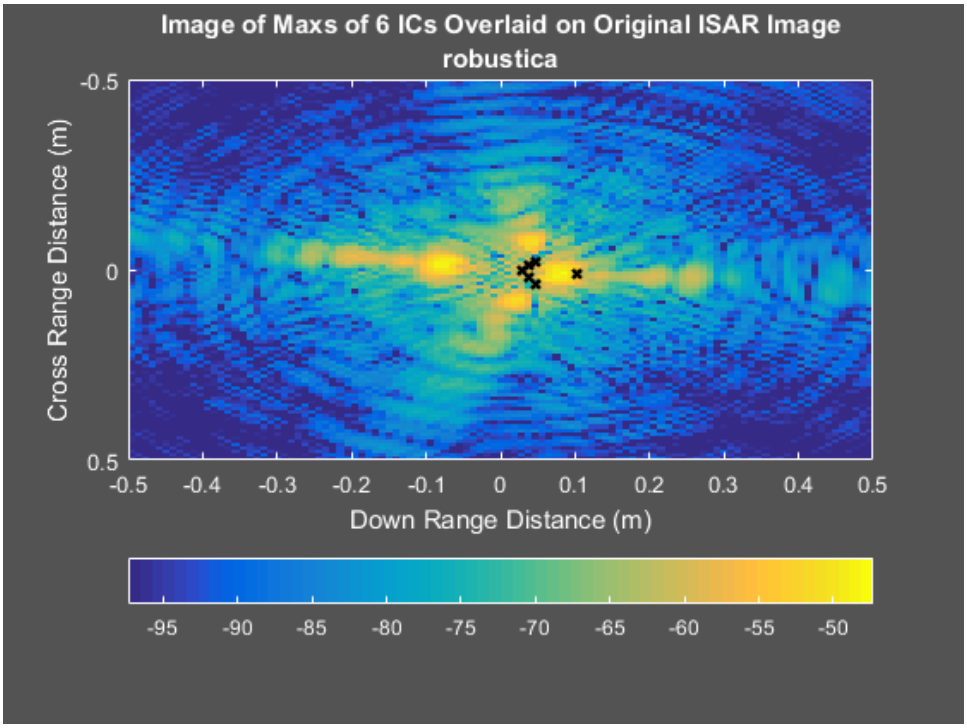

Figure 52: Measured data set 2 results (six degrees-of-freedom)

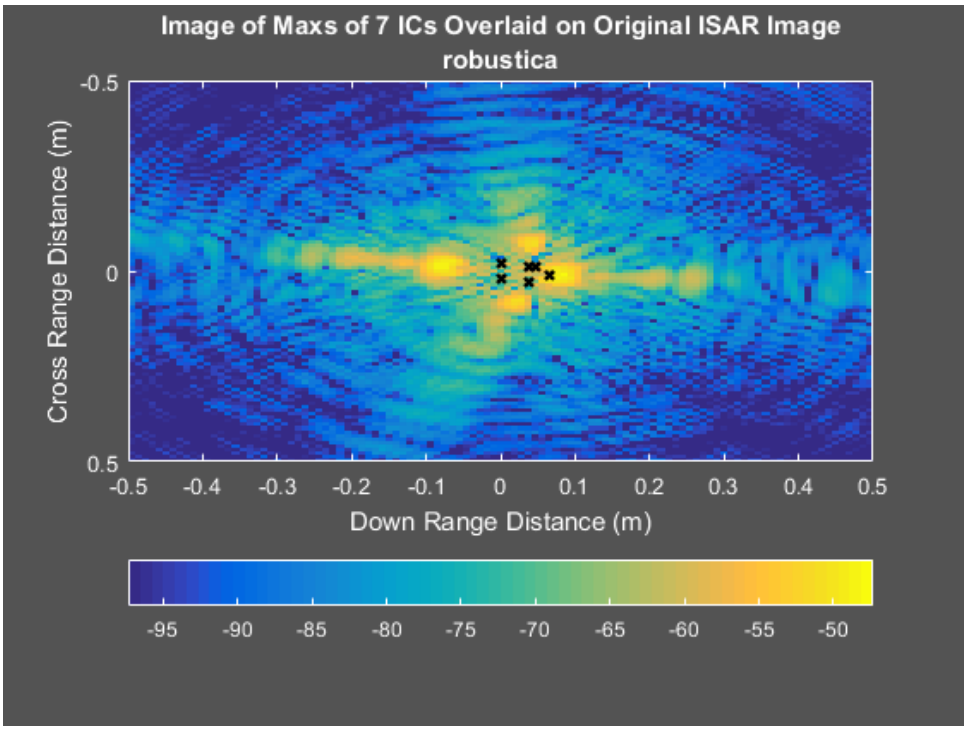

Figure 53: Measured data set 2 results (seven degrees-of-freedom)

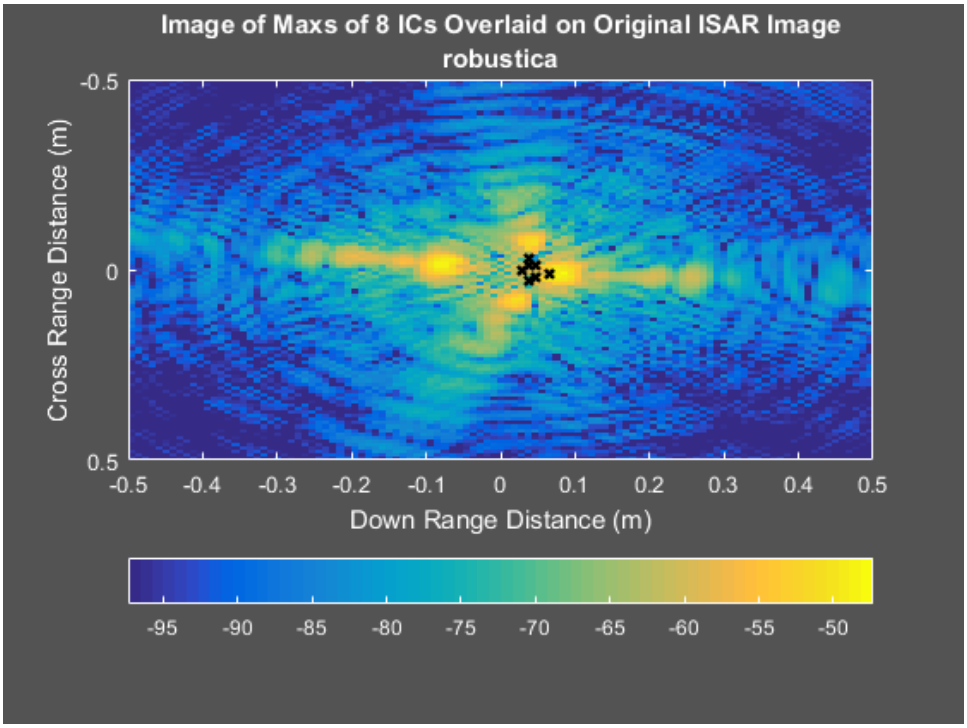

Figure 54: Measured data set 2 results (eight degrees-of-freedom)

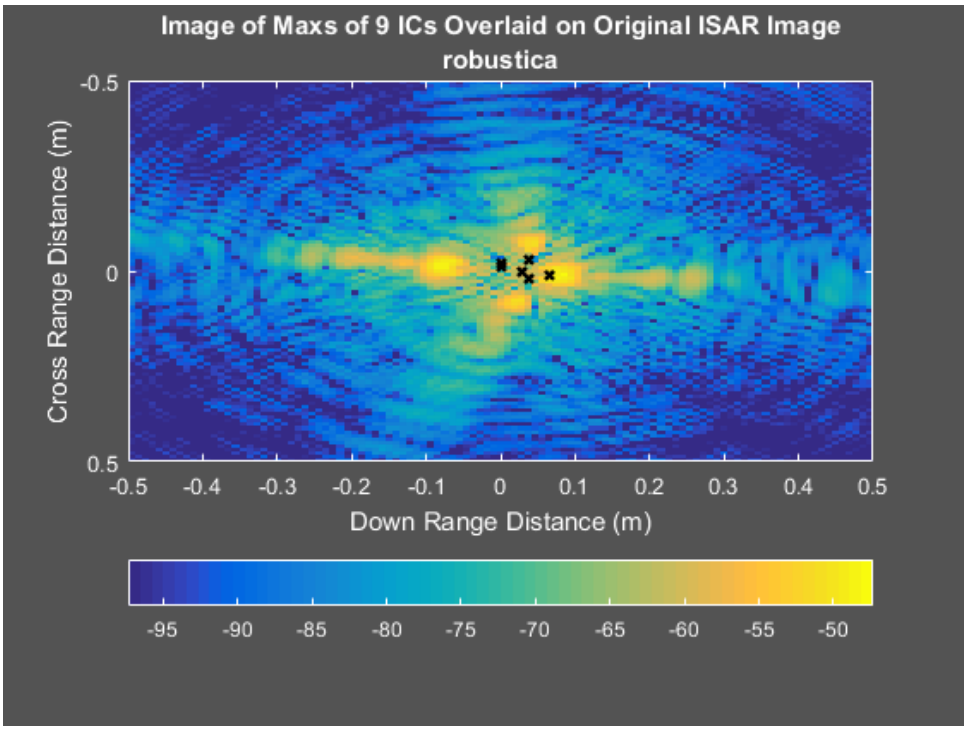

Figure 55: Measured data set 2 results (nine degrees-of-freedom)

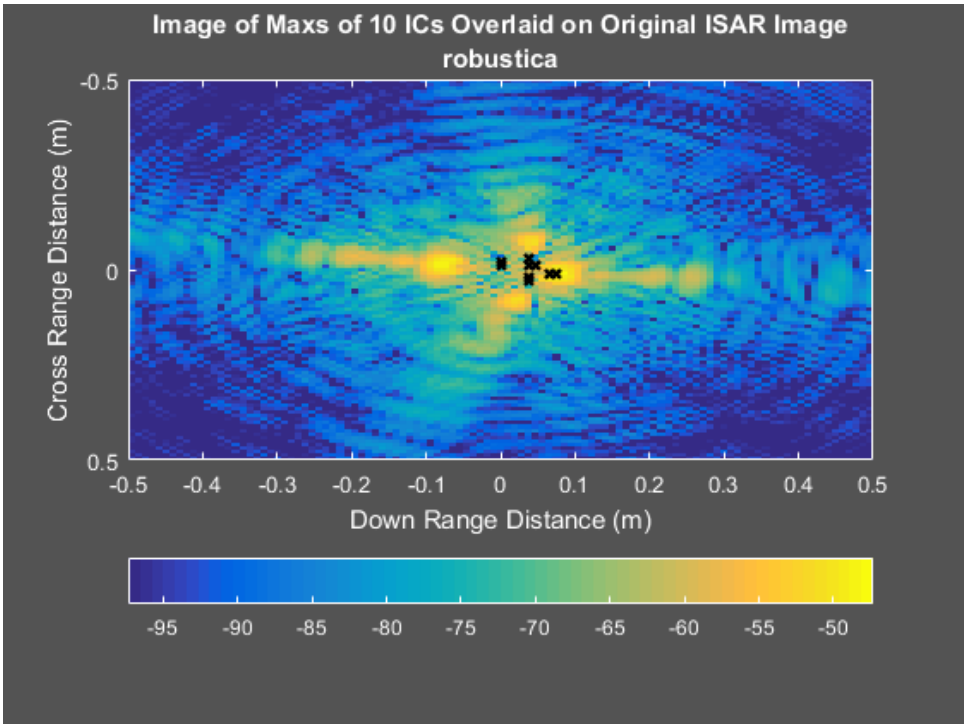

Figure 56: Measured data set 2 results (ten degrees-of-freedom)

Similar to the results of the first data set, the application of the Isolated Scatterer Method has shown possibility for extraction of dimensional and shape information from a complex target. It has also shown the potential for extracting double bounce scattering. Measurement of a more precisely dimensioned target is need to support this conjecture.

Measurement Data Set 3: Three cylinders in a triangle formation with elevated sphere

This data set was comprised of three conductive cylinders laid out in a triangle formation with a 4 in conductive sphere elevated above the triangle. Horizontal polarization was applied during the collection of scattering data for this target. A photographic image of the target set is shown in Figure 57. Figure 58 is a radar image formed using the standard back projection technique. Also shown on this image is a

notional view of the target set marked in magenta lines. This augmentation is meant to give a frame of reference for the scattering which is visualized in the radar image.

The ISAR image shows strong scattering at the intersection of the cylinders, at the broadside angle of the cylinders and at the location of the center of the sphere. There are significant sidelobe artifacts in this image. These artifacts make the recognition and interpretation of the true scattering centers difficult. The application of the new technique proposed by this paper aids in the interpretability of the radar image. Figure 59 through Figure 66 show illustrations of the results of the application of the new technique using degrees of freedom which vary from three to ten. The first four images in this set show that the technique has localized estimated scattering centers at the intersections of the cylinders and at the center of the sphere.

Figure 63 through Figure 66 reveal scattering centers that have been localized close to one another. As with the previous example, it is possible to hypothesize that these closely spaced scatterers may represent separate scattering centers corresponding to the two sides of the same cylinder. However, the multiplicity of scattering center localization which occurs in the region of the center sphere does not show enough separation to allow for the localized scattering centers to be from two sides of the sphere.

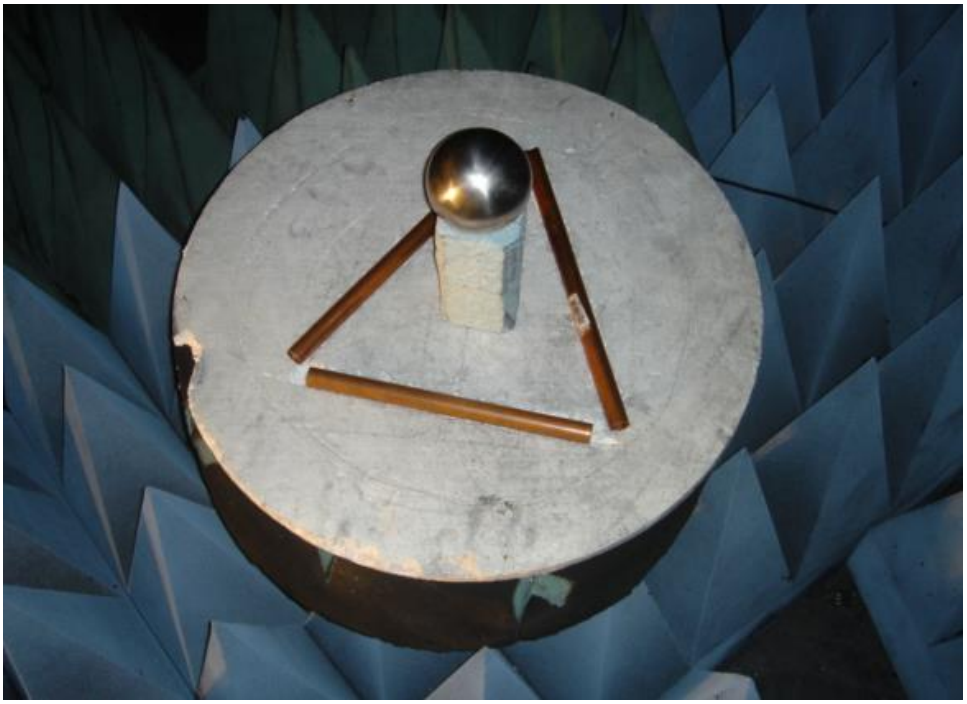

Figure 57: Photo of Target for Measurement Data Set 2

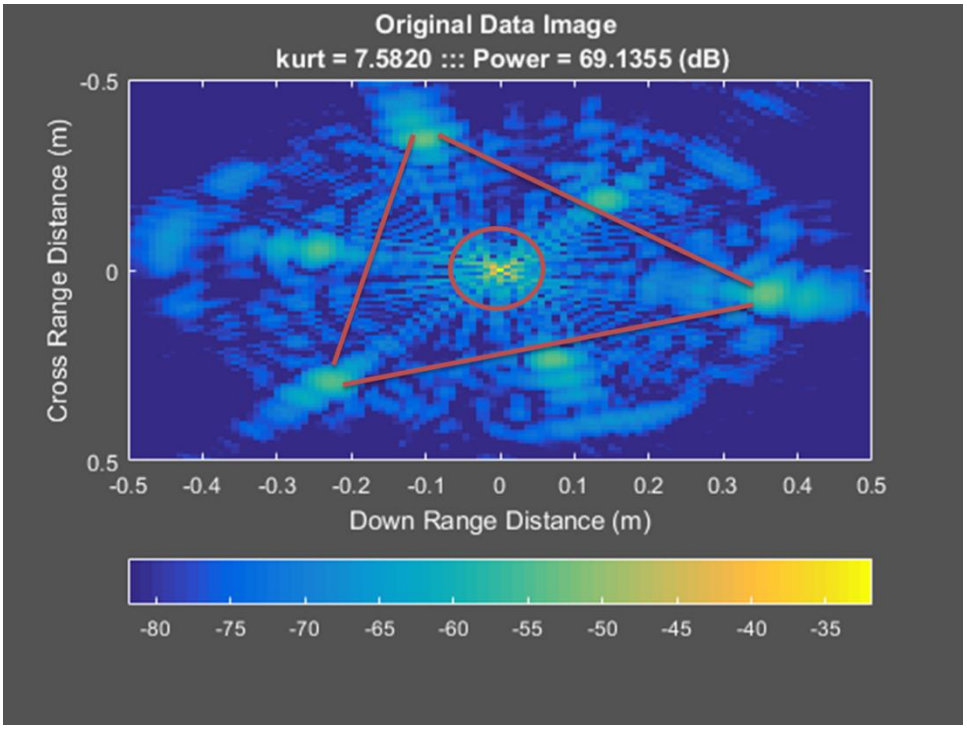

Figure 58: ISAR Image of Measurement Data Set 3

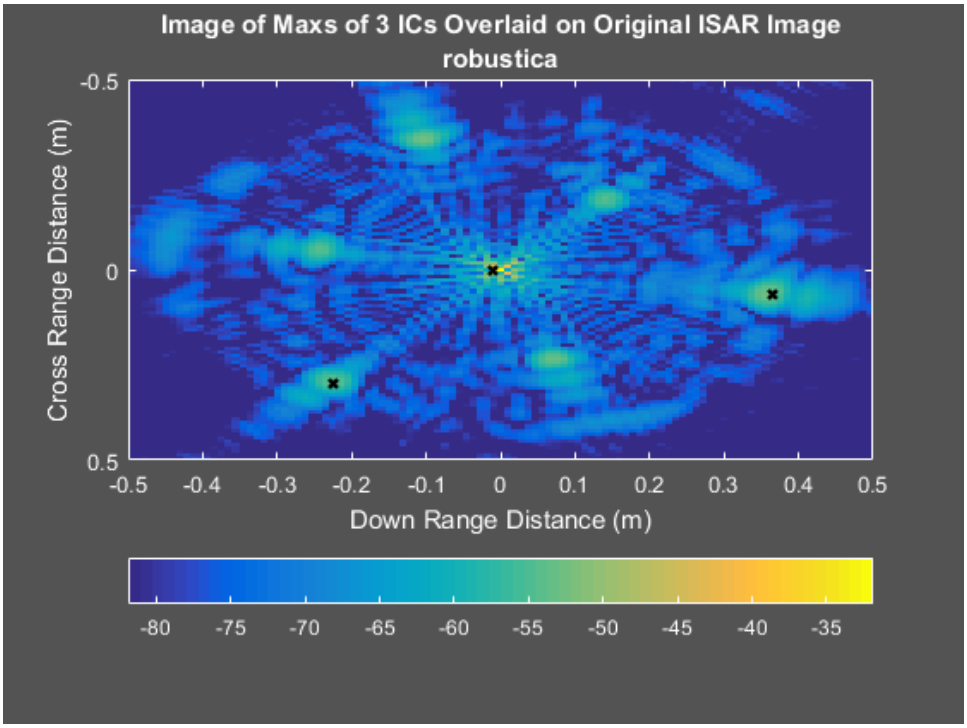

Figure 59: Measured data set 3 results (three degrees-of-freedom)

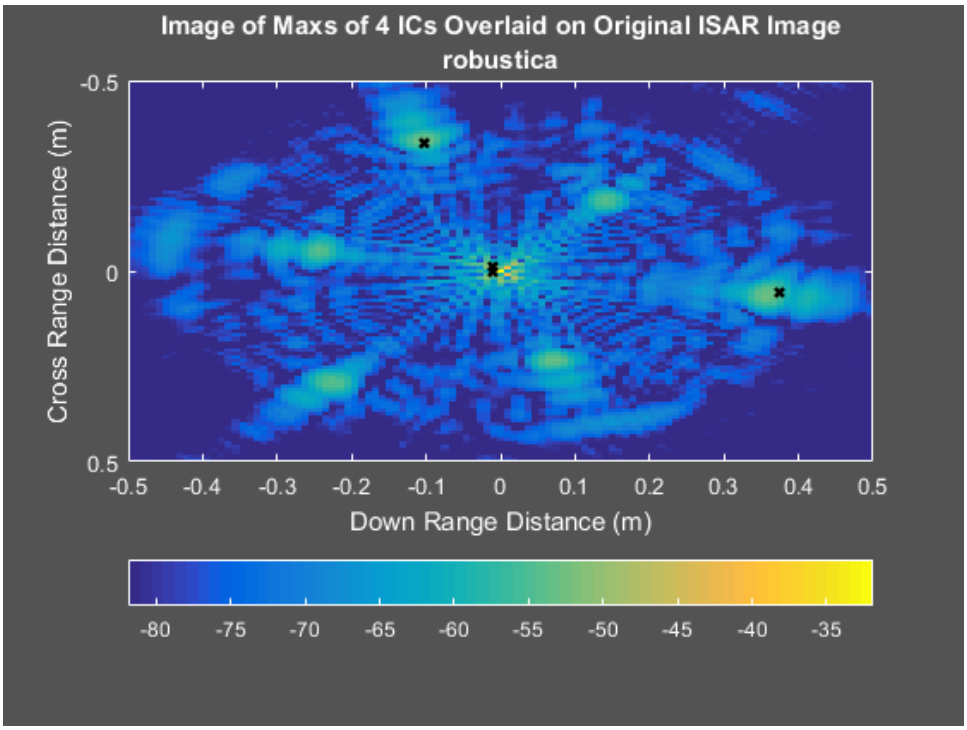

Figure 60: Measured data set 3 results (four degrees-of-freedom)

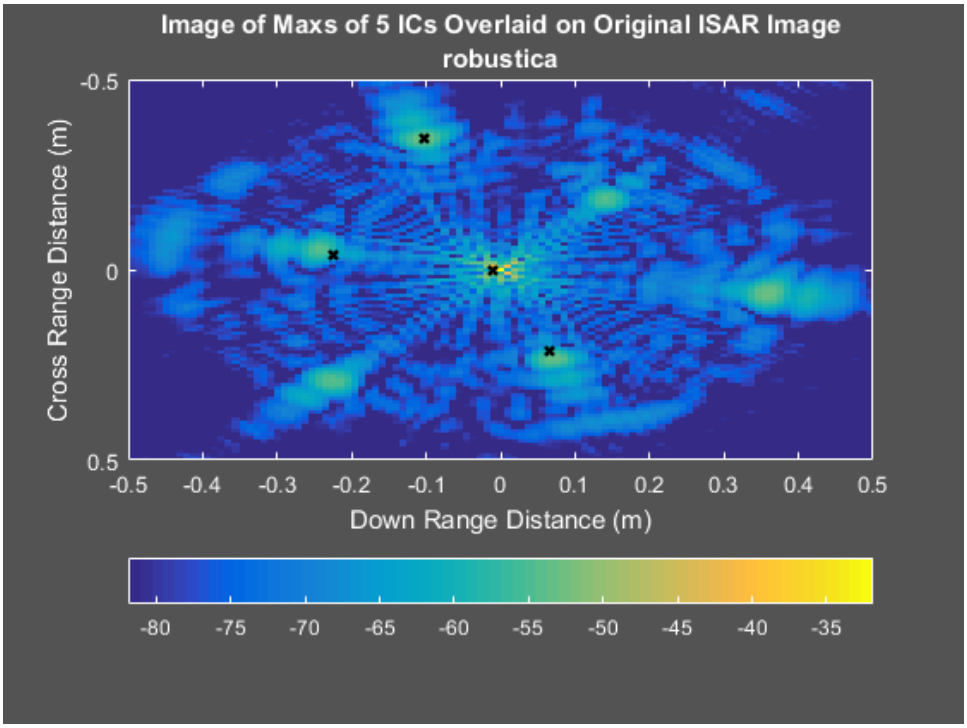

Figure 61: Measured data set 3 results (five degrees-of-freedom)

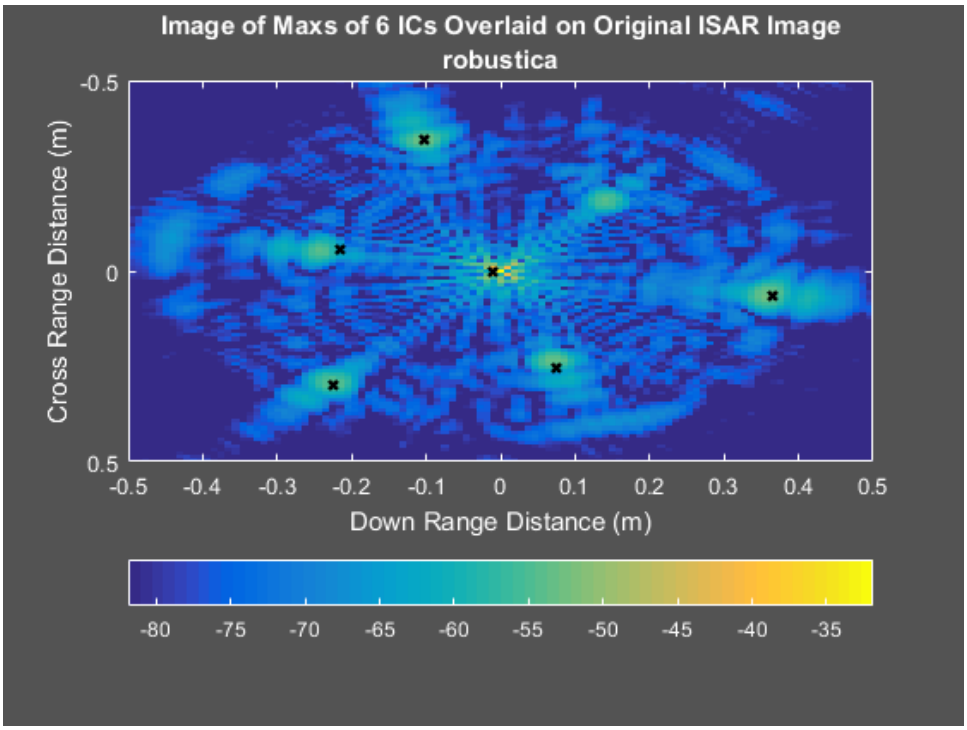

Figure 62: Measured data set 3 results (six degrees-of-freedom)

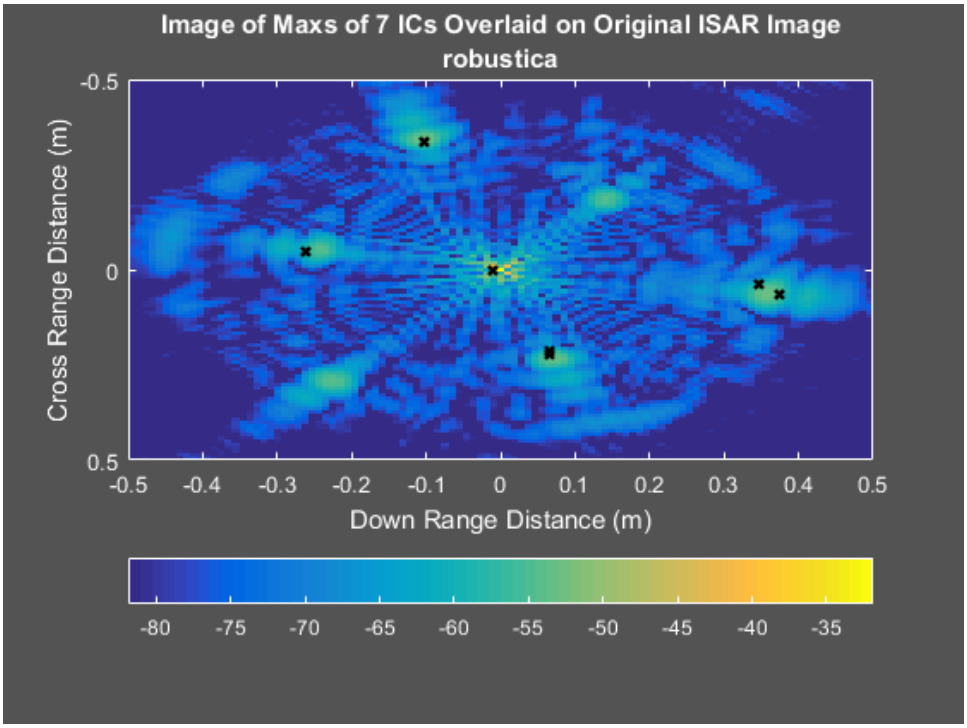

Figure 63: Measured data set 3 results (seven degrees-of-freedom)

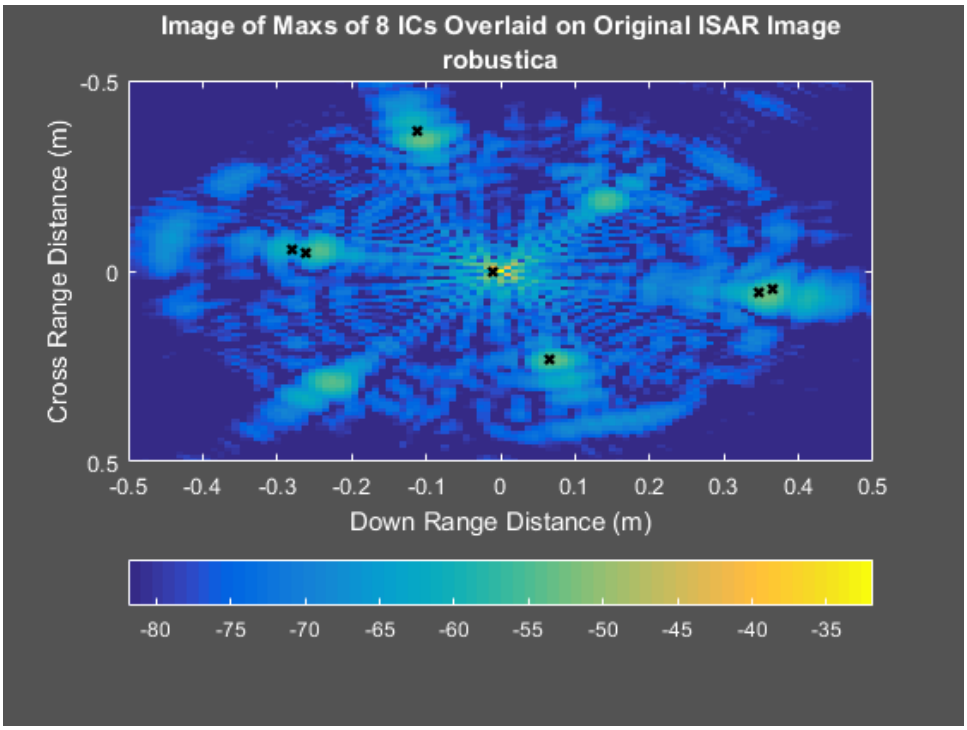

Figure 64: Measured data set 3 results (eight degrees-of-freedom)

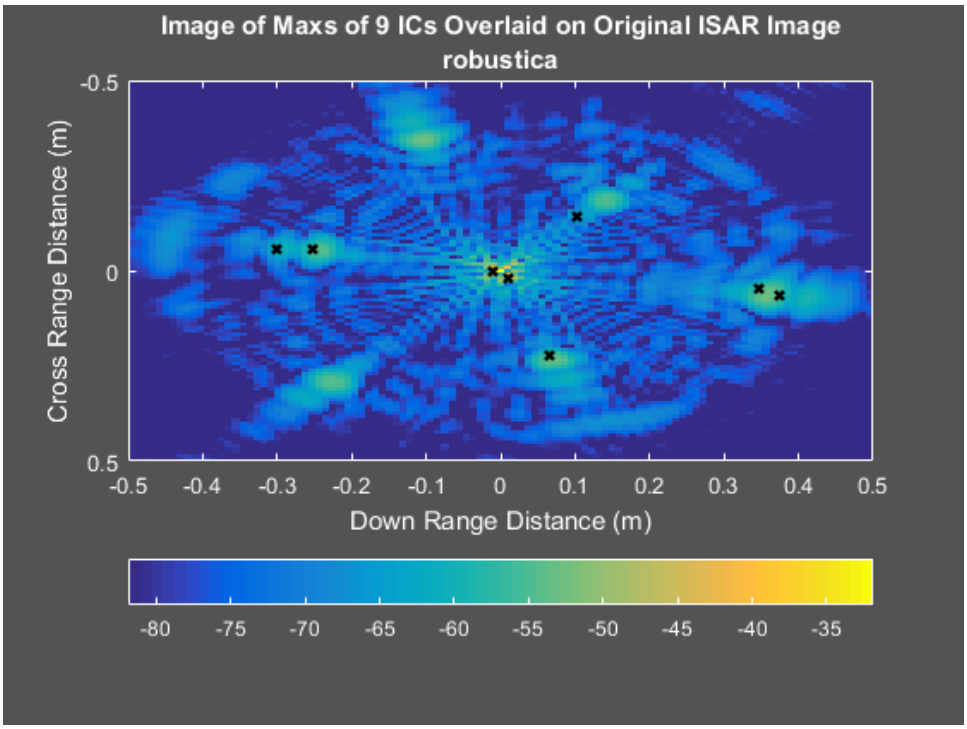

Figure 65: Measured data set 3 results (nine degrees-of-freedom)

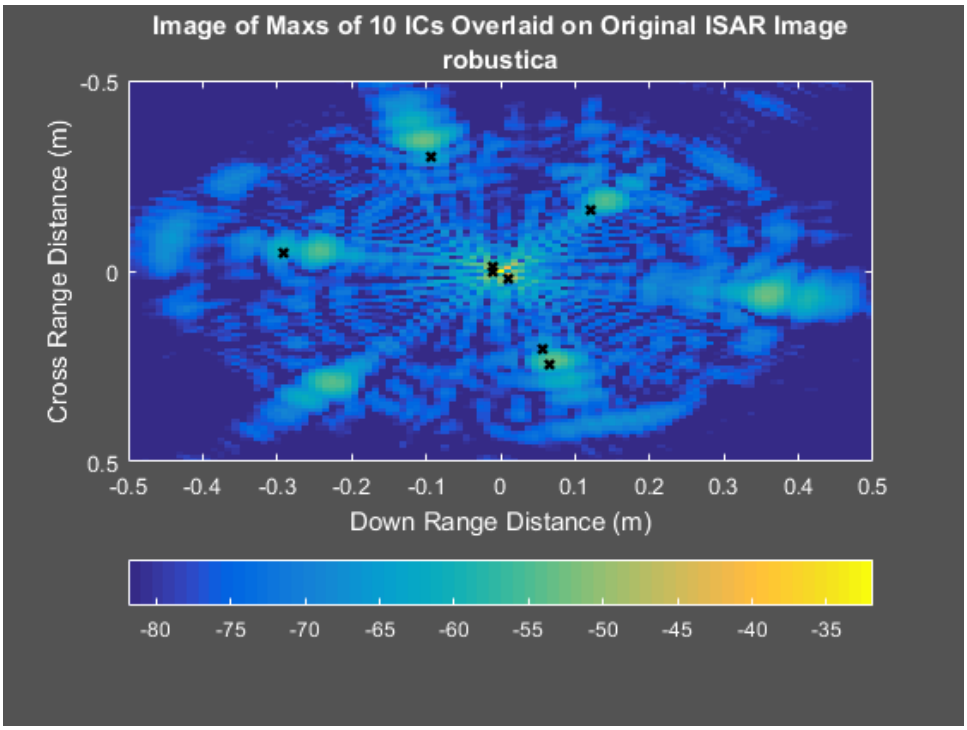

Figure 66: Measured data set 3 results (ten degrees-of-freedom)

The scattering centers detected by the new technique have been localized in the regions associated with the centers and intersections of the cylinders as well as the center sphere. In some cases, the results show what appears to be separated scattering centers for either side a cylinder. If this supposition were to be proven, this could yield valuable information with regard to the dimensions of a target. Further precision measurements are needed to confirm that the separated scattering centers are indeed at locations which would be commensurate with location of the diameter of the cylinder.

Additional application of this technique to measurements from existing RADAR systems would enable further testing of the performance. To reduce the number of estimated scatterers, it would be possible to apply this technique to a chipped region from a large SAR image. Once chipped the section of the SAR image would need to be transformed back to the k-space domain for application of this technique. As shown above, the original image can be augmented with the scattering centers estimated using this new technique.

Two particularly interesting chipping examples would be a motor vehicle or a building with door and window features. A key objective of applying this technique to these examples would be to determine if additional dimensional information can be gleaned using the Isolated Scatterer Method.

Chapter 5 Conclusions and Future Work

Conclusions

The research presented here has shown that there is a strong correlation between the high-kurtosis elements of radar scattering data and the scattering centers that make up the observed scene. The results shown in Chapter 4 reveal that through the use of this technique it is possible to isolate individual scattering elements, and from these isolated elements an accurate composite view of the scattering scene can be formed. The Isolated Scatterer Method (ISM) does show some practical limitations. The effectiveness of the technique has dependency on SNR. An activation threshold between -30 and -20 dB SNR was identified and localization performance improved as the SNR increased above -20 dB. Another known deficiency of the technique is due to phase dispersion across the subapertures. This prevents ideal focusing of the subapertured data. Also documented in Chapter 4, the performance of this technique when applied to closely spaced scatterers was notably poor. Comparison to the MUSIC algorithm revealed a clear disadvantage for the Isolated Scatterer Method for the closely spaced scatterer condition.

The application of the novel technique demonstrated in this paper produces a viable method for augmentation of SAR/ISAR images through precise localization of scattering centers. This novel technique takes advantage of the correlation tested in this work. Employing the assumption that leptokurtic elements of radar scattering data are correlated to scattering centers, it is reasonable to consider each of the isolated high-kurtosis elements as a single scattering center therefore the maximal point within the image domain for each of the isolated elements represents the true location of the scattering center. Through sequential selection of the maximal points it is possible to form a composite image of the scattering scene. Due to the fact that the components

have been isolated, the effect of interference, from sidelobes of nearby, strong scatterers, is reduced. This application of the newly developed signal processing technique has its limitation yet, shows practical utility.

The technique was tested on synthesized point target under 72 test conditions. The technique was also applied to three sets of measurement data collected on the turntable ISAR system in the anechoic chamber of the Wave Scattering Research Center at the University of Texas at Arlington.

The results of the testing, on synthetic target sets, yielded a 52% probability of detection of the known scatterers. Testing also showed that detected scatterers had a high probability of precise localization with 74% probability of detection with zero TLE. This result indicates an affirmative conclusion for the tested hypothesis.

Application of the technique, to the measured data sets in this thesis, offered no absolute positional comparison but is included to confirm the technique works on measured data with reasonable results.

This work has confirmed that the leptokurtic elements of a scattering mixture have some degree of correlation to the scattering centers in the target set. Also presented was a novel signal processing technique which exploits this correlation.

Future Works

Many aspects of this research warrant further exploration. The use of higher order statistics in radar signal processing is a valuable extension to the well-known statistical signal processing techniques based on the second order cumulant.

One area of future work should be exploring the application of this technique using Independent Vector Analysis (IVA) to extract the high-kurtosis elements. The optimization method afforded by this algorithm takes into account the correlation among the dataset manifold. An obstacle in the application of this technique is mapping of the output of the algorithm to the native k-space domain for imaging. Once this obstacle is overcome, the IVA algorithm should afford better performance over the ICA technique used here because the algorithm is able to exploit correlation between the datasets.

Another area which warrants expansion is the application of this technique using radar imaging methods other than the Fourier back-projection technique. Through the use of an imaging method such as MUSIC.

Further study of the subaperture partitioning particularly in the angle dimension is a possible avenue of additional research. Use of overlapped or non-contiguous subapertures may hold the potential to increase the probability of detection which would yield a more complete set of scattering centers.

Development of a calibration method to reduce the amount of phase dispersion across the sub-apertures is also a worthy extension of this research which could improve the performance of the Isolated Scatterer Method.

Observations stemming from the measurement data sets allude to the possibility of the technique to extract multiple bounce scattering. The expansion of the

modeled study to include multiple bounce scattering should be an avenue of further research.

An important avenue of further research is measurement of a target set in which the locations of the scattering elements are known precisely. High precision positional information for a measurement target could provide quantitative proof of the hypothesized feature extraction capability of this method.

Appendix A

MATLAB for Parametric Study

```
fclose all;clear all;clc;close all
%%RNG control
s = RandStream('mt19937ar','Seed',1);
RandStream.setGlobalStream(s);
%% set options
load('colorsDistinct');
set(0,'DefaultAxesColorOrder',colors)
visbility = 'on';
total = tic;
maxRange = 1; % Note larger ranges will require a greater number
of interpolation points
nAngSamps = 36;
baseAngRange = pi;
angRange = baseAngRange;
nFreqPoints = 801;
rAnt = 100; %(m)
performPCA = 0;
RCScomp = true;
randScat = true;
method = 'o';%{'o','r'}
plotdBlim = 50;
completedRunsLog = '';
%% Output Base Directory
%enter the base directory for the output
outputDirBase = '';
iHyper = 0;
%% loops
tTotal = tic;
nScatSet = 3;%2:7;
iSNRSet = 1:4;
for nScat = nScatSet
for nAngSA = 1%:nScat+2
nSASet = nScat:nScat+2;
for nFreqSA = nSASet;
nTotalSA = nAngSA*nFreqSA;
if and(nTotalSA >= (nScat),nTotalSA <= 2*(nScat))
for iSNR = iSNRSet
clear kurt power
for inIC = nScat:nTotalSA
skip = false;
snrS = [0 -20 -10];%[-10:2:10];
snr = snrS(iSNR);
nICextract = inIC;
algoChoice = 'robustICA';
for iBinsSep = 1:3;
nBinsSet = [0.5 1 2];
```

```
nBinsSep = nBinsSet(iBinsSep);
dateTimeStamp = datestr(now,30);
if ~RCScomp
if randScat
outputDir =
fullfile(outputDirBase,sprintf('MaxRange%0.2f\\%0.0fAngleSamples
R%0.0fRandomScatLocs\\%0.0fScatterers_%0.0fdBThresh0PhaseRCS\\%0
.0fSNR\\%0.0fICs\\%0.0fAngSA_%0.0fFreqSA\\%s',...
maxRange,nAngSamps,rAnt,nScat,plotdBlim,snr,nICextract,nAngSA,nF
reqSA,algoChoice));

else
outputDir =
fullfile(outputDirBase,sprintf('MaxRange%0.2f\\%0.0fAngleSamples
R%0.0fControlledScatLocs\\%0.1fbins\\%0.0fScatterers_%0.0fdBThre
sh0PhaseRCS\\%0.0fSNR\\%0.0fICs\\%0.0fAngSA_%0.0fFreqSA\\%s',...
maxRange,nAngSamps,rAnt,nBinsSep,nScat,plotdBlim,snr,nICextract,
nAngSA,nFreqSA,algoChoice));
end

else
if randScat
outputDir =
fullfile(outputDirBase,sprintf('MaxRange%0.2f\\%0.0fAngleSamples
R%0.0fRandomScatLocs\\%0.0fScatterers_%0.0fdBThresh0PhaseRCS\\%0
.0fSNR\\%0.0fICs\\%0.0fAngSA_%0.0fFreqSA\\%s',...
maxRange,nAngSamps,rAnt,nScat,plotdBlim,snr,nICextract,nAngSA,nF
reqSA,algoChoice));
else
outputDir =
fullfile(outputDirBase,sprintf('MaxRange%0.2f\\%0.0fAngleSamples
R%0.0fControlledScatLocs\\%0.1fbins\\%0.0fScatterers_%0.0fdBThre
sh0PhaseRCS\\%0.0fSNR\\%0.0fICs\\%0.0fAngSA_%0.0fFreqSA\\%s',...
maxRange,nAngSamps,rAnt,nBinsSep,nScat,plotdBlim,snr,nICextract,
nAngSA,nFreqSA,algoChoice));
end

end
resFile = fullfile(outputDir,'1_Results.pptx');
if exist(resFile,'file')
iHyper = iHyper + 1;
prevFile = resFile;
fprintf('File has been saved: <a
href="matlab:winopen(''%s'')">%s</a>\n',prevFile,prevFile);
skip = true;
else
skip = false;
mkdir(outputDir)
isOpen  = exportToPPTX();
if ~isempty(isOpen),
```

```
% If PowerPoint already started, then close first and then open
a new one
exportToPPTX('close');
end
% http://www.mathworks.com/matlabcentral/fileexchange/40277-
exporttopptx
exportToPPTX('new','Dimensions',[12 6], ...
'Title','Parametric Study Results', ...
'Author','JHall', ...
'Subject','AlgorithmResults', ...
'Comments','This file has been automatically generated by
exportToPPTX');
% add entry to log of completed runs
fidLogID = fopen(completedRunsLog,'a');
fprintf(fidLogID,'nScat= %0.0f rAnt= %0.1f SNR= %0.0f nAngSamps=
%0.0f nAngSA= %0.0f nFreqSamps= %0.0f nFreqSA= %0.0f nTotalSA=
%0.0f Dir= %s\n',...
nScat,rAnt,snr,nAngSamps,nAngSA,nFreqPoints,nFreqSA,nTotalSA,out
putDir);
fclose(fidLogID);
end
if ~skip
notesText = sprintf('%0.0f Scatterers\n%0.0f FreqSA\n%0.0f
AngleSA\nsnr = %0.0f\n %0.0f
ICs\n%s\n%s',nScat,nFreqSA,nAngSA,snr,
nICextract,algoChoice,outputDir);
clc
fprintf(1,'Output Dir: %s\n',outputDir);
diary(fullfile(outputDir,'Dairy.txt'))
%% Initial Set up and options
fprintf(1,'DTS: %s\n',dateTimeStamp);
fprintf(1,'filename: %s\n', mfilename);
%================================================================
===========
%% constants and data collection variables
%================================================================
===========
light_speed = 2.997925e008;
f_start = 4e9;
f_stop = 8e9;
nFreqPoints = 801;
phi = linspace(0,angRange-angRange/nAngSamps,nAngSamps);%(rad)
azimuth angle of the antenna/turntable
step = (f_stop - f_start) / (nFreqPoints);
f = linspace(f_start, f_stop, nFreqPoints);
lambda = light_speed ./ f;
%wavenumber
k = 2*pi ./ lambda;
depression = 45;
elevation = depression;
```

```
stamp = datestr(now);
fprintf(1,'Date Time Stamp: %s\n',stamp);
fprintf(1,'Elevation Angle: %0.1f\n',elevation);
fprintf(1,'Max Range: %0.1f\n',maxRange);
markerColor = [0 0 0];
markerSize = 25;
markerLineWidth = 1.75;
if randScat
zoomLims =[...
-.5 .5;
-.5 .5];
else
zoomLims =[...
0.4 0.6;
-0.1 0.1];
end
fprintf('Per Sample SNR = %0.2f\n',snr);
%% Interpolation
kx = (k' * (sind(90-elevation) .* cos(phi)));
ky = (k' * (sind(90-elevation) .* sin(phi)));
rangeRes = -light_speed / (4*(f_stop-f_start));
if angRange == 2*pi
rStep = light_speed./(2*f_stop).*(cosd(elevation));
rStepX = 0.5*pi./(max(kx(:)) - min(kx(:)));
rStepY = 0.5*pi./(max(ky(:)) - min(ky(:)));
elseif and(pi/2 < angRange,angRange <= pi)
rStepY = light_speed./(2*(f_stop-f_start)).*cosd(elevation);
rStepX = 0.5*rStepY;
else
rStepY = light_speed./(2*(f_stop-f_start)).*cosd(elevation);
rStepX = rStepY;
end
Nsamps = 2^11;
%Axes
xR = -((0:(Nsamps-1)) * rStepX - (Nsamps)/2 * rStepX);
yR = -((0:(Nsamps-1)) * rStepY - (Nsamps)/2 * rStepY);

if randScat
s = RandStream('mt19937ar','Seed',1);
RandStream.setGlobalStream(s);
aTall = (2*rand(1,100)-1) * pi;
aTnQ = aTall(1:nScat);
rTall = rand(1,100) * maxRange;
rTnQ = rTall(1:nScat);
rTnQ(rTnQ <= 1/(4*pi)) = 1/(4*pi);
xTnQ = (rTnQ.*cos(aTnQ));
yTnQ = (rTnQ.*sin(aTnQ));
else
%the two lines below are to set
%deterministic locations for two scaterers
```

```
angDeterm = zeros(nScat,1);
rDetMin = maxRange/2;
for iXX = 1:nScat
xTnQ(iXX) = (rDetMin + (iXX-
1)*nBinsSep*rStepX)*cos(angDeterm(iXX));
yTnQ(iXX) = (rDetMin + (iXX-1)*nBinsSep*0)*sin(angDeterm(iXX));
end
end
quantizeBins = 0;
if quantizeBins
xT = round(xTnQ / (rStepX)) *(rStepX);
yT = round(yTnQ / (rStepY)) *(rStepY);
else
xT = xTnQ;
yT = yTnQ;
end
rT = hypot(xT,yT);
aT = atan2(yT,xT);
[XR YR] = meshgrid(xR,yR);
yPixSize = abs(mean(diff(yR)));
xPixSize = abs(mean(diff(xR)));
xScat = zeros(1,nScat);
yScat = zeros(1,nScat);
for iScat = 1:nScat
% find closest points
[minDistX(iScat), xScat(iScat)]  = min(abs(xR - xT(iScat)));
[minDistY(iScat), yScat(iScat)]  = min(abs(yR - yT(iScat)));
minDist(iScat) = hypot(minDistX(iScat),minDistY(iScat));
end
pixMismatch =
(minDist./hypot(abs(mean(diff(xR))),abs(mean(diff(yR)))));
pixMismatch = pixMismatch ./ max(pixMismatch);
% Range Normalization
dTheory= (rT)./sind(elevation);
RangeNormalization = abs(1./(4 *pi*dTheory));
RCS = 1./((RangeNormalization));
RCS = RCS./max(RCS);
if ~RCScomp
RCS = ones(1,nScat);
end
%% Model Scattering
strFormat = [repmat('%0.4f ',1,numel(RCS)-1) '%0.4f (dB)\n'];
fprintf(1,sprintf('Scatter Magnitude
Theoretical:%s',strFormat),20*log10(RCS));
strFormat = ['Scatter Phase Angles Theoretical:' repmat('%0.4f
',1,numel(RCS)-1) '%0.4f (deg)\n'];
fprintf(strFormat,180/pi*angle(RCS));
fprintf(1,'Dynamic Range of Plots: %0.1f (dB)\n',plotdBlim);
if randScat
zoomLims =[...
```

```
-(max(rT)+.1) max(rT)+.1;
-(max(rT)+.1) max(rT)+.1];
else
zoomLims =[...
(max(rT)-.1) max(rT)+.1;
-0.1 0.1];
end
zoomLimsExpanded = 5 *zoomLims;
[target, scatFunction]  =
simulateTurntableISAR('nFreq',nFreqPoints,...
'fstart',f_start,'fstop',f_stop,'theta',pi/2-elevation *
pi/180,...
'nScatterers',nScat,'r', rT,'a',aT ,'RCS',RCS,'dsnr',-snr,...
'rAnt',rAnt,'phi',phi,'outputdir',
outputDir,'lambda',lambda,'k',k);
[r, c] = size(target);
targetWeighted = target;

%% K-Space Generation and polar to rectangular interpolation
%================================================================
===========
%Create K-Space Indexes
%================================================================
===========
[F, PHI] = meshgrid(f,phi);
theta = elevation*pi/180;
f_span = (f_stop - f_start);
rStept = light_speed.*cosd(elevation)./(f_span);
%% Input Data Characteristics
kurt.Input.Full = kurtosis(real(targetWeighted(:)));
power.Input = 20*log10(sum(abs(targetWeighted(:))));
%% 2D interpolation of full data set
tInterp = tic;
x = transpose(kx);
y = transpose(ky);
v = targetWeighted;
%rectangular grid
Xmin = min(x(:));
Xmax = max(x(:));
Ymin = min(y(:));
Ymax = max(y(:));
xi = linspace(Xmin, Xmax, Nsamps);
yi = linspace(Ymin, Ymax, Nsamps);
[XF, YF] = meshgrid(xi,yi);
interpType = 'linear';
extrapType = 'none';
FF =
scatteredInterpolant((x(:)),(y(:)),v(:),interpType,extrapType);
ZIF = FF((XF),(YF));
ZIF(isnan(ZIF)) = 0;
```

```
ZIF(isinf(ZIF)) = 0;
Inner = sqrt(XF.^2 + YF.^2) < min(k).* sin(pi/2-theta);
Outer = sqrt(XF.^2 + YF.^2) > max(k).* sin(pi/2-theta);
ZIF(Inner) = 0;
ZIF(Outer) = 0;
fprintf(1,'Interpolation Time = %0.2f\n',toc(tInterp));
%% ISAR Image of Original data
target_xformF = ifftshift(ifft2((ZIF)));
power.ZIF= sum(abs(ZIF(:)).^2);
power.xform = sum(squeeze(abs(target_xformF(:))).^2);
dataO = 20*log10(abs(target_xformF));
maxValO = max(dataO(:));
targetQuads = floor(aT/(pi/2));
fprintf(1,'RCS compensation: %0.0f',RCScomp);
magScatOrigdB = zeros(1,nScat);
angScatOrigRad = zeros(1,nScat);
scatter2BGOrigdB = zeros(1,nScat);
scatter2ClutterOrigdB = zeros(1,nScat);
cluserDimOver2 = 0;
extractedAOI =
zeros(nScat,2*cluserDimOver2+1,2*cluserDimOver2+1);
cluserDimOver2Expand = 5;
extractedAOIExpand =
zeros(nScat,2*cluserDimOver2Expand+1,2*cluserDimOver2Expand+1);
% findindices within zoomlims
logicalXlim = and(xR >= zoomLims(1),xR <= zoomLims(3));
logicalYlim = and(yR >= zoomLims(2),yR <= zoomLims(4));
valMaxOriginaldB = max((dataO(:)));
[rowLocMaxOriginal, columnLocMaxOriginal] = find((dataO) ==
valMaxOriginaldB);
max2BGOrigdB =
20*log10(abs(target_xformF(rowLocMaxOriginal,columnLocMaxOrigina
l))...
./ (sum(sum(abs(target_xformF(logicalXlim,logicalYlim)))) -
abs(target_xformF(rowLocMaxOriginal,columnLocMaxOriginal))));
formString = repmat(' %0.4f',1,nScat-1);
formStringInteger = repmat(' %0.0f',1,nScat-1);
formString = strcat(formString,' %0.4f\n');
formStringInteger = strcat(formStringInteger,' %0.4f\n');
fprintf(1,sprintf('\nOriginal Scatter X location:
%s',formString),xT);
fprintf(1,sprintf('Original Scatter Y location:
%s',formString),yT);
for iScat = 1:nScat
[minDistX(iScat), xScat(iScat)]  = min(abs(xR - xT(iScat)));
[minDistY(iScat), yScat(iScat)]  = min(abs(yR - yT(iScat)));
minDist(iScat) = hypot(minDistX(iScat),minDistY(iScat));
extractedAOIYExpand(iScat,:) = yScat(iScat)-
cluserDimOver2Expand:yScat(iScat)+cluserDimOver2Expand;
```

```
extractedAOIXExpand(iScat,:) = xScat(iScat)-
cluserDimOver2Expand:xScat(iScat)+cluserDimOver2Expand;
extractedAOIExpand(iScat,:,:) =
target_xformF(extractedAOIYExpand(iScat,:),...
extractedAOIXExpand(iScat,:));
[magScatOrigdB(iScat), locScatOrig(iScat)] =
max(20*log10(abs(extractedAOIExpand(iScat,:))));
[locScatOrigY(iScat) locScatOrigX(iScat)] =
find(squeeze(20*log10(abs(extractedAOIExpand(iScat,:,:)))) ==
magScatOrigdB(iScat));
scatter2BGOrigdB(iScat) =
20*log10(max(abs(extractedAOIExpand(iScat,:)))...
./ (sum(sum(abs(target_xformF(logicalXlim,logicalYlim)))) -
max(abs(extractedAOIExpand(iScat,:)))));
angScatOrigRad(iScat) =
mean(angle(extractedAOIExpand(iScat,locScatOrig(iScat))));
locScatOrigXM(iScat) =
xR(extractedAOIXExpand(iScat,locScatOrigX(iScat)));
locScatOrigYM(iScat) =
yR(extractedAOIYExpand(iScat,locScatOrigY(iScat)));
end
fprintf(1,sprintf('Adjusted Scatter X location:
%s',formString),xT);
fprintf(1,sprintf('Adjusted Scatter Y location:
%s',formString),yT);
max2ScatRatioOriginal = valMaxOriginaldB - magScatOrigdB;
formString = repmat(' %0.4f',1,nScat-1);
formString = strcat(formString,' %0.4f\n');
fprintf(1,sprintf('\nScatters to BG ratio
Original:%s',formString),scatter2BGOrigdB(:));
fprintf(1,'Max to BG ratio Original: %0.4f\n',max2BGOrigdB);
fprintf(1,'Max Value Original: %0.4f\n',valMaxOriginaldB);
fprintf(1,'Max Location Original: Row %0.0f, Col %0.0f\n',...
rowLocMaxOriginal,columnLocMaxOriginal);
fprintf(1,sprintf('Scatters Magnitude
Original:%s',formString),magScatOrigdB(:));
fprintf(1,sprintf('Scatters Phase Angle
Original%s',formString),angScatOrigRad(:));
h = figure('Name','ISAR
Image','NumberTitle','off','visible',visbility);
whitebg(h,'k')
imagesc(xR,yR,data0);shading interp
title(sprintf('Original Data Image \n kurt = %0.4f ::: Power =
%0.4f (dB)',kurt.Input.Full,power.Input))
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
caxis([maxVal0-plotdBlim maxVal0]);
colorbar('location','SouthOutside')
hold on
```

```
for iMark = 1:nScat
scatter(xT(iMark),yT(iMark),'SizeData',markerSize,...
'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))
if regexp(scatFunction,'wAngular')
line([xT(iMark),xT(iMark)+maxRange/10*cos(scatterBeamDirs(iMark)
-scatterBeamWidths(iMark)/2)],...
    [yT(iMark),...
    yT(iMark)+maxRange/10*sin(scatterBeamDirs(iMark)-
scatterBeamWidths(iMark)/2)],...
    'LineWidth',markerLineWidth,'color',1-colors(iMark,:))
line([xT(iMark),xT(iMark)+maxRange/10*cos(scatterBeamDirs(iMark)
+scatterBeamWidths(iMark)/2)],...
    [yT(iMark),...

yT(iMark)+maxRange/10*sin(scatterBeamDirs(iMark)+scatterBeamWidt
hs(iMark)/2)],...
    'LineWidth',markerLineWidth,'color',1-colors(iMark,:))
end
end
hold off
set(h,'visible','on')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
set(h,'InvertHardcopy','off')
saveas(h,fullfile(outputDir,'ImageOrigISAR.fig'));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',h,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(h)
%% Create Subapertures
entryCount = 0;
outputFlag = true;
for nAnglesPerSA = floor(size(target,1)/nAngSA)%1:8
fprintf(1,'\nNumber of Sensors per SA: %0.1f\n',nAnglesPerSA);
fprintf(1,'Number of Angle SAs: %0.1f\n',nAngSA);
nFreqPerSA = floor(c/nFreqSA);
fprintf(1,'Number of Freq  per SAs: %0.1f\n',nFreqPerSA);
fprintf(1,'Number of Freq SAs: %0.1f\n',nFreqSA);
iMaxFreq =  nFreqSA* nFreqPerSA;
fprintf(1,'BW of Freq SAs: %0.0f MHz\n',step * nFreqPerSA /
1E6);
fprintf(1,'Total Number of SAs: %0.1f\n',nTotalSA);
iMaxAng =  nAngSA* nAnglesPerSA;
targetSubSet = zeros(nTotalSA,nAnglesPerSA*nFreqPerSA);
nAngsOut = nAnglesPerSA * nAngSA;
nFreqsOut = nFreqPerSA * nFreqSA;
SACount = 0;
entryCount = entryCount+1;
iFreq = zeros(nFreqSA,nFreqPerSA);
```

```
for iterAng = 1:nAngSA
for iF = 1:nFreqSA
    fIX = 1:nFreqSA;
    iterFreq = fIX(iF);
    SACount = SACount + 1;
    % Continuous sampling
    iAng = nAnglesPerSA*(iterAng - 1)+1:nAnglesPerSA*iterAng;
    iFreq(iF,:) = nFreqPerSA*(iterFreq -
1)+1:nFreqPerSA*iterFreq;
    targetSubSetTemp = target(iAng,iFreq(iF,:));
    targetSubSet(SACount,:) = targetSubSetTemp(:);
    kurt.Input.SA(SACount) =
kurtosis(real(targetSubSetTemp(:)));
end
end
%% RobustICA
tol = 1e-2;      % termination threshold parameter
max_it = 1e5;    % maximum number of iterations per independent
component
if outputFlag
outputFlag = false;
%negative kurtSign extracts subgaussian components,
%positive kurtSign extracts superGaussian components,
%zero does not extract components
kurtSign = zeros(1, nTotalSA);
kurtSign(1:nICextract) = 1;
stringFormat = strcat('kurtSign:  ', repmat('%0.0f
',1,nTotalSA), ' \n');
fprintf(1,stringFormat,kurtSign);
end

if performPCA
if entryCount ==1
    fprintf(1,'Perform PCA is on\n');
end
else
if entryCount ==1
    fprintf(1,'Perform PCA is off\n');
end
end
if entryCount ==1
fprintf(1,'Method is %s\n',method);
end
% http://www.i3s.unice.fr/~zarzoso/robustica.html
[algo, shat, A, iter, W] = robustica((targetSubSet), kurtSign,
tol,...
max_it, performPCA, method, 0, [], 0,entryCount);
%% Remap subapertures
for nIC=nICextract
fprintf(1,'Number of "ICs": %0.1f\n',nIC);
```

```
reCompose = zeros(nICextract, iMaxAng,iMaxFreq);
reComposeWeighted = zeros(nIC, iMaxAng,iMaxFreq);
Yprime = zeros(nTotalSA,nFreqPerSA * nAnglesPerSA);
%% Remap data to k space
for iIC = 1:nICextract
    temp = zeros(size(shat));
    temp(iIC,:) = shat(iIC,:);
    Yprime = temp;
    SACount=0;
    for iterAng = 1:nAngSA
        for  iF = 1:nFreqSA
            iterFreq = fIX(iF);
            SACount = SACount + 1;
            % Continuous sampling
            iAng = nAnglesPerSA*(iterAng -
1)+1:nAnglesPerSA*iterAng;
            iFreq(iF,:) = nFreqPerSA*(iterFreq -
1)+1:nFreqPerSA*iterFreq;
            reCompose(iIC,iAng,iFreq(iF,:)) =  ...

reshape(Yprime(SACount,:),nAnglesPerSA,nFreqPerSA);
        end
    end
    kurt.Output(iIC) =
kurtosis(real(reCompose(iIC,reCompose(iIC,:)~=0)));
    power.Output(iIC) = 20*log10(sum(abs(reCompose(iIC,:))));
    reComposeWeighted(iIC,:,:) = squeeze(reCompose(iIC,:,:));
end
sumICsFreqDomain = squeeze(sum(real(reComposeWeighted),1));
kurt.SumICs = kurtosis(sumICsFreqDomain(:));
power.SumICs = 20*log10(sum(abs(reComposeWeighted(:)))) -
10*log10(size(reComposeWeighted,1));

fprintf(1,'Kurtosis of input data: %0.4f\n',kurt.Input.Full);
stringFormat = strcat('Kurtosis of output data:  ',
repmat('%0.4f ',1,nTotalSA), ' \n');
fprintf(1,stringFormat,kurt.Output);
fprintf(1,'Kurtosis of sum of the output data: %0.4f
(dB)\n',kurt.SumICs);

fprintf(1,'Power of input data: %0.4f (dB)\n',power.Input);
stringFormat = strcat('Power of output data:  ', repmat('%0.4f
',1,nTotalSA), ' (dB)\n');
fprintf(1,stringFormat,power.Output);
fprintf(1,'Power of sum of the output data: %0.4f
(dB)\n',power.SumICs);

%% Plot ICs
upsampleRatio = 1;
upsampleRatioFFT = 2;
```

```
xRuS = -((0:(Nsamps*upsampleRatio-1)) * rStepX -
(Nsamps*upsampleRatio)/2 * rStepX);
yRuS = -((0:(Nsamps*upsampleRatio-1)) * rStepY -
(Nsamps*upsampleRatio)/2 * rStepY);
target_xformFIC =
zeros(nIC,Nsamps*upsampleRatioFFT,Nsamps*upsampleRatioFFT);
scatter2BGICsdB = zeros(nIC,nScat);
Scat2Back = zeros(nIC,nScat);
magScatICsdB = zeros(nIC,nScat);
angScatICsRad = zeros(nIC,nScat);
Scat2BackNorm = zeros(nIC,nScat);
scatter2ClutterICsdB = zeros(nIC,nScat);
scatter2ClutterICsdBNorm = zeros(nIC,nScat);
extractedAOIICs =
zeros(nIC,nScat,2*cluserDimOver2+1,2*cluserDimOver2+1);

clear ikurtSort
[valkurtSort, ikurtSort] = sort(kurt.Output,'descend');
max2ScatRatioICs = zeros(nIC,nScat);
max2BGICsdB = zeros(nIC,1);
rowLocMaxIC = zeros(nIC,1);
columnLocMaxIC = zeros(nIC,1);
valMaxICdB = zeros(nIC,1);
diffScat = zeros(nScat,nIC,nAngsOut,nFreqsOut);

distMeasPixX = zeros(nIC,nScat);
distMeasPixY = zeros(nIC,nScat);
distMeasPixXY = zeros(nIC,nScat);
distMeasMeterXY = zeros(nIC,nScat);
mindistMeter = zeros(nIC,1);
closestScatIndex = zeros(nIC,1);

for iIC = 1:nIC
    iSelect = iIC;
    rangeRes = -light_speed / (4*(f_stop-f_start));
    rangeResProj = -light_speed / (4*(f_stop-f_start)) /
cos((theta - pi/2) * 180/pi);
    %% 2D interpolation of ICs
    x =
transpose(kx(1:size(reComposeWeighted,3),1:size(reComposeWeighte
d,2)));
    y =
transpose(ky(1:size(reComposeWeighted,3),1:size(reComposeWeighte
d,2)));
    v = squeeze(reComposeWeighted(iSelect,:,:));
    %rectangular grid
    xiuS = linspace(min(xi),max(xi),upsampleRatio*numel(xi));
    yiuS = linspace(min(yi),max(yi),upsampleRatio*numel(yi));
    [XF, YF] = meshgrid(xiuS,yiuS);
    interpType = 'linear';
```

```
    extrapType = 'none';
    FF =
scatteredInterpolant(x(:),y(:),v(:),interpType,extrapType);
    ZIF = FF(XF,YF);
    ZIF(isnan(ZIF)) = 0;
    ZIF(isinf(ZIF)) = 0;
    Inner = sqrt(XF.^2 + YF.^2) < min(k).* sin(theta);
    Outer = sqrt(XF.^2 + YF.^2) > max(k).* sin(theta);
    ZIF(Inner) = 0;
    ZIF(Outer) = 0;
    %%
    upsampleRatioFFT = 2;
    target_xformFIC =
zeros(nIC,Nsamps*upsampleRatioFFT,Nsamps*upsampleRatioFFT);
    nfftColIC = upsampleRatioFFT*size(ZIF,2);
    nfftRowIC = upsampleRatioFFT*size(ZIF,1);
    target_xformFIC(iIC,:,:) =
ifftshift(ifft2(ifftshift(ZIF),nfftRowIC,nfftColIC));
    data = 20*log10(abs(squeeze(target_xformFIC(iIC,:,:))));
    maxVal = max(data(:));
    %find peaks in the IC data
    valMaxICdB(iIC) = max(data(:));
    [rowLocMaxIC(iIC), columnLocMaxIC(iIC)] = find(data ==
valMaxICdB(iIC));
    xRuS = -((0:(Nsamps*upsampleRatioFFT-1)) *
rStepX/upsampleRatioFFT...
        - ((Nsamps-1)*upsampleRatioFFT)/2 *
rStepX/upsampleRatioFFT);
    yRuS = -((0:(Nsamps*upsampleRatioFFT-1)) *
rStepY/upsampleRatioFFT...
        - ((Nsamps)*upsampleRatio*upsampleRatioFFT)/2 *
rStepY/upsampleRatioFFT);
    % check power levels
    power.reComposeWeighted(iIC) =
sum(abs(reComposeWeighted(iSelect,:)).^2);
    power.ZIFIC(iIC)= sum(abs(ZIF(:)).^2);
    power.RecomposeScaled(iIC) =
sum(abs(reComposeWeighted(iSelect,:)).^2)/(numel(Yprime(:))
/(Nsamps^2));
    power.xformIC(iIC) =
sum(squeeze(abs(target_xformFIC(iIC,:))).^2);
    %% peal picking
    valMaxICdB(iIC) = max(data(:));
    [rowLocMaxIC(iIC), columnLocMaxIC(iIC)] = find(data ==
valMaxICdB(iIC));
    upsampleRatioFFT = 2;
    xRuS = -((0:(Nsamps*upsampleRatioFFT-1)) *
rStepX/upsampleRatioFFT...
        - (Nsamps*upsampleRatioFFT)/2 *
rStepX/upsampleRatioFFT);
```

```
    yRuS = -((0:(Nsamps*upsampleRatioFFT-1)) *
rStepY/upsampleRatioFFT -
(Nsamps*upsampleRatio*upsampleRatioFFT)/2 * rStepY/2);
    hIC = figure('Name','ISAR Image of
ICs','NumberTitle','off','visible','on');
    whitebg(hIC,'k')
    imagesc(xRuS,yRuS,data);shading interp
    xlabel('Down Range Distance (m)');
    ylabel('Cross Range Distance (m)')
    colormap('parula')
    caxis([maxVal-plotdBlim
maxVal]);colorbar('location','SouthOutside')
    hold on
    scatter(xRuS(columnLocMaxIC(iIC)),
yRuS(rowLocMaxIC(iIC)),ones(size(rowLocMaxIC(iIC))),'SizeData',m
arkerSize,...

'Marker','x','LineWidth',markerLineWidth,'MarkerEdgeColor',[0 0
0],'MarkerFaceColor',[0 0 0]);
    for iMark = 1:nScat
        scatter(xT(iMark),yT(iMark),'SizeData',markerSize,...

'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark+1,:))
    end
    hold off
    title(sprintf('Image of IC # %0.0f of %0.0f \n Kurt = %0.4f
::: Power = %0.4f (dB)::: %s',...

iIC,nIC,kurt.Output(iSelect),power.Output(iSelect),algo),...
        'interpreter','none')
    set(gca,'XLim',zoomLims(1,:))
    set(gca,'YLim',zoomLims(2,:))
    set(hIC,'InvertHardcopy','off')

saveas(hIC,fullfile(outputDir,sprintf('ImageIC%0.0fOF%0.0f.png',
iIC,nIC)));
    slideNum = exportToPPTX('addslide'); %#ok<NASGU>
    exportToPPTX('addpicture',hIC,'Scale','maxfixed');
    exportToPPTX('addnote',notesText);
    delete(hIC)
    %% examine miss distance
    for iScat = 1:numel(xT)
        extractedAOIICs(iIC,iScat,:,:) = target_xformFIC(iIC,...
            yScat(iScat)-
cluserDimOver2:yScat(iScat)+cluserDimOver2,...
            xScat(iScat)-
cluserDimOver2:xScat(iScat)+cluserDimOver2);
        [magScatICsdB(iIC,iScat), locScatICs(iIC,iScat)] =
max(20*log10(abs(extractedAOIICs(iIC,iScat,:))));
```

```
        scatter2BGICsdB(iIC,iScat) =
20*log10(max(abs(extractedAOIICs(iIC,iScat,:))).*  power.ZIF / power.ZIFIC(iIC) ...
            ./
(sum(sum(abs(target_xformFIC(iIC,logicalXlim,logicalYlim)))) -
max(abs(extractedAOIICs(iIC,iScat,:))));
        angScatICsRad(iIC,iScat) =
(angle(extractedAOIICs(iIC,iScat,locScatICs(iIC,iScat))));
    end
    Scat2Back(iIC,:) = scatter2BGICsdB(iIC,:);
    Scat2BackNorm(iIC,:) = scatter2BGICsdB(iIC,:)-
scatter2BGOrigdB;
    max2BGICsdB(iIC) =
20*log10(abs(target_xformFIC(iIC,rowLocMaxIC(iIC),
columnLocMaxIC(iIC))).*  power.ZIF / power.ZIFIC(iIC) ...
        ./
(sum(sum(abs(target_xformFIC(iIC,logicalXlim,logicalYlim)))) -
abs(target_xformFIC(iIC,rowLocMaxIC(iIC),
columnLocMaxIC(iIC))));
    max2ScatRatioICs(iIC,:) = valMaxICdB(iIC) -
magScatICsdB(iIC,:);
    formString = repmat(' %0.4f',1,nScat-1);
    formString = strcat(formString,' %0.4f\n');
    fprintf(1,sprintf('\nScatters to BG ratio
IC%%0.0f:%s',formString),iIC,scatter2BGICsdB(iIC,:));
    fprintf(1,'Max to BG ratio IC%0.0f:
%0.4f\n',iIC,max2BGICsdB(iIC));
    fprintf(1,'Max Value IC%0.0f: %0.4f\n',iIC,valMaxICdB(iIC));
    fprintf(1,'Max Location IC%0.0f: Row %0.0f, Col %0.0f\n',...
        iIC,rowLocMaxIC(iIC),columnLocMaxIC(iIC));
    fprintf(1,sprintf('Scatters Magnitude (dB)
IC%%0.0f:%s',formString),iIC,magScatICsdB(iIC,:));
    fprintf(1,sprintf('Scatters Phase Angle (Rad)
IC%%0.0f:%s',formString),iIC,angScatICsRad(iIC,:));
    % gauge distance
    % between maximum pixel
    % and the known
    % locations of the
    % scatterers in the
    % scene
    distMeasPixX(iIC,:) =
bsxfun(@minus,columnLocMaxIC(iIC),xScat);
    distMeasPixY(iIC,:) = bsxfun(@minus,rowLocMaxIC(iIC),yScat);
    distMeasPixXY(iIC,:) =
hypot(distMeasPixX(iIC,:),distMeasPixY(iIC,:));
    distMeasMeterXY(iIC,:) =
hypot(bsxfun(@minus,xRuS(columnLocMaxIC(iIC)),xR(xScat)),...
        bsxfun(@minus,xRuS(rowLocMaxIC(iIC)),xR(yScat)));
end
```

```
for iIC = 1:nIC
    [mindistMeter(iIC), closestScatIndex(iIC)] =
min(distMeasMeterXY(iIC,:));
    fprintf(1,sprintf('\nMinimum Miss Distance (M)
IC%%0.0f:%s',formString),iIC,mindistMeter(iIC,:));
    fprintf(1,sprintf('Closest Known Scatterer
IC%%0.0f:%s',formStringInteger),iIC,closestScatIndex(iIC,:));
end

% Plot Image of the max of all ICs
dataMaxs = ones(Nsamps,Nsamps) * 20*log10(eps);
for iMax = 1:numel(rowLocMaxIC)
    dataMaxs(rowLocMaxIC(iMax), columnLocMaxIC(iMax)) = 0;
end
hICm = figure('Name','ISAR Image of max of
ICs','NumberTitle','off','visible',visbility);
whitebg(hICm,'k')
imagesc(xR,yR,dataMaxs);shading interp
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
caxis([-plotdBlim 0]);colorbar('location','SouthOutside')
hold on
for iMark = 1:nScat
    scatter(xT(iMark),yT(iMark),'SizeData',markerSize,...

'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))
end
hold off
title(sprintf('Image of Maxs of %0.0f ICs\n%s ::: SNR =
%0.1f',...
    nIC,algo,snr),...
    'interpreter','none')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
set(hICm,'InvertHardcopy','off')
saveas(hICm,fullfile(outputDir,sprintf('ImageMaxsICs%0.0f.fig',n
IC)));
slideNum = exportToPPTX('addslide'); %#ok<NASGU>
exportToPPTX('addpicture',hICm,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(hICm)

% Overlay Image of the max of
% all ICs on original ISAR
hICmO = figure('Name','ISAR Image of max of ICs
Overlay','NumberTitle','off','visible',visbility);
whitebg(hICmO,'k')
```

```
imagesc(xR,yR,dataO);shading interp
hold on
scatter(xRuS(columnLocMaxIC),
yRuS(rowLocMaxIC),ones(size(rowLocMaxIC)),'SizeData',markerSize,
...

'Marker','x','LineWidth',markerLineWidth,'MarkerEdgeColor',[0 0
0],'MarkerFaceColor',[0 0 0]);
hold off
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
caxis([maxValO-plotdBlim
maxValO]);colorbar('location','SouthOutside')
hold on
for iMark = 1:nScat
    scatter(xT(iMark),yT(iMark),'SizeData',markerSize,...

'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))

end
hold off
title(sprintf('Image of Maxs of %0.0f ICs Overlaid on Original
ISAR Image\n%s ::: SNR = %0.1f',...
    nIC,algo,snr),...
    'interpreter','none')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
set(hICmO,'InvertHardcopy','off')
saveas(hICmO,fullfile(outputDir,sprintf('ImageMaxsICs%0.0fOverla
y.fig',nIC)));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',hICmO,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(hICmO)
end
end
hMissD=figure('visible','on');
plot(1,mindistMeter(1),'Marker','o','Color',colors(closestScatIn
dex(1),:))
hold on
for iIC = 2:nIC
plot(iIC,mindistMeter(iIC),'Marker','o','Color',colors(closestSc
atIndex(iIC),:))
end
hold off
maxMissError =
max(abs(min(mindistMeter)),abs(max(mindistMeter)))*1.1;
axis([0 nIC+1 -maxMissError maxMissError+eps])
```

```
set(gca,'XTick',0:nIC+1)
titleText = sprintf('Distance Error ::: %s',algo);
title(titleText,'interpreter','none');xlabel('Extracted
Independent Component');
ylabel('Distance Error (M)');
whitebg(hMissD,'k')
set(hMissD,'InvertHardcopy','off');
saveas(hMissD,fullfile(outputDir,sprintf('MissD.png')));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',hMissD,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(hMissD)
fprintf(1,'\npower of original kspace =
%0.4f\n',10*log10(power.ZIF));
fprintf(1,'power of sum of IC kspace =
%0.4f\n',10*log10(sum(power.ZIFIC)));
fprintf(1,'Input power = %0.4f (dB)\n',power.Input);
fprintf(1,'power of sum of IC kspace = %0.4f
(dB)\n',10*log10(sum(10.^(power.Output./10))));
% count the number of misses
nMisses = sum(max2ScatRatioICs~=0,2);
fprintf(1,'Total Processing Time: %0.4f\n',toc(tTotal));
newFile = exportToPPTX('save',fullfile(outputDir,'1_Results'));
exportToPPTX('close')
iHyper = iHyper +1;
hyperLinks{iHyper} = sprintf('<a
href="matlab:winopen(''%s'')">%s</a>\n',newFile,newFile);
fprintf('File has been saved: <a
href="matlab:winopen(''%s'')">%s</a>\n',newFile,newFile);

%% Save selected variables
save(fullfile(outputDir,'Variables'),'angScatICsRad','angScatOri
gRad','hyperLinks',...
'kurt','magScatICsdB','magScatOrigdB','max2BGICsdB','max2BGOrigd
B','max2ScatRatioICs',...
'max2ScatRatioOriginal','notesText','outputDir','Scat2Back','Sca
t2BackNorm',...
'scatter2BGICsdB', 'scatter2BGOrigdB', 'scatter2ClutterICsdB',
'scatter2ClutterICsdBNorm',...
'valMaxICdB','valMaxOriginaldB','power',...
'rowLocMaxIC', 'columnLocMaxIC','xScat','yScat',...
'distMeasMeterXY', 'distMeasPixX', 'xPixSize',
'distMeasPixY','locScatOrigXM','locScatOrigYM',...
'yPixSize','mindistMeter','closestScatIndex',...
'nMisses','xT','yT','rT','aT','W','A')

end
end
end
end
```

```
end
end
end
end
diary('off')
toc(tTotal)


function [target, scatFunction]  =
simulateTurntableISAR(varargin)
% 'nScatterers'
% 'r' %  vector of scatterer location relative to center of
turntable
% 'a' % vector of scatterer angle relative 0
% 'RCS' % vector of scattering magnitudes for scatterers
% 'rAnt' % antenna location relative to center of the turntable
% 'aAnt' % antenna angle relative to zero
% 'theta' % depression angle of the antenna
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%
% Default values for optional parameters
% All
nScatterers       = 3;
theta             = 120*pi/180;         %(deg2rad) depression
angle of the antenna
r                 = [.2 .4 .1];         %(m)
a                 = [0 pi pi/2];        %(rad)
RCS               = [0 0 0];            %(dB)
rAnt              = 50;                 %(m) radius to antenna
aAnt              = 1;                  %(rad) angle to antenna
phi               = linspace(0,2*pi-2*pi/72,72);%(rad) azimuth
angle of the antenna/turntable
addNoise          = 1;
dSNR              = 10;                %(dB)
RMSnoisePhase     = 0 * pi/180;        %(deg2rad)
fStart            = 4e9;                %(Hz)
fStop             = 20e9;               %(Hz)
nFreq             = 256;
% Read the optional parameters
scatFunction = mfilename('fullpath');
fprintf(1,'Scattering Model: %s\n',scatFunction);
if (rem(length(varargin),2)==1)
error('Optional parameters should always go by pairs');
else
for i=1:2:(length(varargin)-1)
if ~ischar (varargin{i}),
error (['Unknown type of optional parameter name (parameter' ...
' names must be strings).']);
end
% change the value of parameter
```

```
switch lower (varargin{i})
case 'nscatterers'
nScatterers = varargin{i+1};
case 'r'
r = varargin{i+1};
case 'a'
a = varargin{i+1};
case 'rcs'
RCSini = varargin{i+1};
case 'rant'
rAnt = varargin{i+1};
case 'aant'
aAnt = varargin{i+1};
case 'theta'
theta = varargin{i+1};
case 'phi'
phi = varargin{i+1};
case 'addnoise'
addNoise = varargin{i+1};
case 'dsnr'
RMSnoiseAmpdB = varargin{i+1};
case 'rmsnoisephase'
dSNR = varargin{i+1};
case 'fstart'
fStart = varargin{i+1};
case 'fstop'
fStop = varargin{i+1};
case 'nfreq'
nFreq = varargin{i+1};
case 'outputdir'
OutputDir = varargin{i+1};
case 'k'
k = varargin{i+1};
case 'lambda'
lambda = varargin{i+1};
otherwise
% Hmmm, something wrong with the parameter string
error(['Unrecognized parameter: ''' varargin{i} '''']);
end;
end;
end


fid = fopen(fullfile(OutputDir,'Setup.txt'),'w+');

strFormat = ['r:\t\t' repmat('%0.4f ',1,numel(r)-1) '%0.4f
(m)\n'];
fprintf(fid, strFormat,r) ;
strFormat = ['a:\t\t' repmat('%0.4f ',1,numel(a)-1) '%0.4f
(rad)\n'];
```

```
fprintf(fid, strFormat,a) ;
strFormat = ['RCS:\t\t' repmat('%0.4f ',1,numel(RCSini)-1)
'%0.4f (dB)\n'];
fprintf(fid, strFormat,20*log10(RCSini)) ;
strFormat = ['scatterphaseangles:\t\t' repmat('%0.4f
',1,numel(RCSini)-1) '%0.4f (deg)\n'];
fprintf(fid, strFormat,180*pi*angle(RCSini)) ;
strFormat = ['rAnt:\t\t' repmat('%0.4f ',1,numel(rAnt)-1) '%0.4f
(m)\n'];
fprintf(fid, strFormat,rAnt) ;
strFormat = ['aAnt:\t\t' repmat('%0.4f ',1,numel(aAnt)-1) '%0.4f
(rad)\n'];
fprintf(fid, strFormat,aAnt) ;
strFormat = ['theta:\t' repmat('%0.4f ',1,numel(theta)-1) '%0.4f
(rad)\n'];
fprintf(fid, strFormat,theta) ;
strFormat = ['phi:\t\t' repmat('%0.4f ',1,numel(phi)-1) '%0.4f
(rad)\n'];
fprintf(fid, strFormat,phi) ;
strFormat = ['addNoise:\t' repmat('%0.4f ',1,numel(addNoise)-1)
'%0.4f (ul)\n'];
fprintf(fid, strFormat,addNoise) ;
strFormat = ['dSNR:\t\t' repmat('%0.4f ',1,numel(dSNR)-1) '%0.4f
(dB)\n'];
fprintf(fid, strFormat,RMSnoiseAmpdB) ;
strFormat = ['fStart:\t' repmat('%0.4f ',1,numel(fStart)-1)
'%0.4f (GHz)\n'];
fprintf(fid, strFormat,fStart/1E9) ;
strFormat = ['fStop:\t' repmat('%0.4f ',1,numel(fStop)-1) '%0.4f
(GHz)\n'];
fprintf(fid, strFormat,fStop/1E9) ;
strFormat = ['nFreq:\t' repmat('%0.4f ',1,numel(nFreq)-1) '%0.4f
(ul)\n'];
fprintf(fid, strFormat,nFreq) ;
fprintf(fid, 'OutputDir: %s\n',OutputDir) ;
fclose(fid);
%% Measurment Specifics
nPhi = size(phi,2);
%% Table/Scatterer Geometry
fVoltage = 1; %0 for power 1 for voltage
if fVoltage
cal = transpose(exp(-1j*k*rAnt/cos(theta)) .* ...
1./(4*pi*rAnt/cos(theta)));
else
cal = transpose(exp(-1j*k*rAnt/cos(theta)) .* ...
1.^2./(4*pi*(rAnt/cos(theta)).^2));
end
randPhase = exp(1j*rand(1,nScatterers) * 2* pi);
for iX = 1:nPhi
d= sqrt(rAnt^2 + r.^2 - 2 .* rAnt .* r .* cos(a - phi(iX)))...
```

```
./sin(theta);
RCS = RCSini;
if addNoise
if fVoltage
noiseamp = mean(abs(cal)) * 10^(RMSnoiseAmpdB/20)/sqrt(2) ; %
(the sqrt(2) accounts for real+imaginary powers)
else
noiseamp = mean(abs(cal)) * 10^(RMSnoiseAmpdB/10)/sqrt(2) ;
end
N = noiseamp*(randn(nFreq,1)+1j*randn (nFreq,1));
if fVoltage
targetRaw(:,iX) =
sum(bsxfun(@times,randPhase,bsxfun(@times,RCSini,...
bsxfun(@times,exp(-1j*transpose(k)*(d)),...
1./(4*pi*abs(d))))),2)...
+ N;
else
targetRaw(:,iX) =  sum(bsxfun(@times,RCSini,...
exp(-1j*transpose(k)*d) .* (transpose(lambda).^2 *
(1./(4*pi*d.^2)))),2)...
+ N;
end
realizedSNR(iX) = 20*log10(max(abs(sum(bsxfun(@times,RCSini,...
bsxfun(@times,exp(-1j*transpose(k)*(d)),...
1./(4*pi*abs(d)))),2)))...
./mean(abs(N))/nScatterers);
else
if fVoltage
targetRaw(:,iX) =  sum(bsxfun(@times,RCSini,...
exp(-1j*transpose(k)*d) .* ((transpose(lambda) *
(1./(4*pi*d))))),2);
else
targetRaw(:,iX) =  sum(bsxfun(@times,RCSini,...
exp(-1j*transpose(k)*d) .* ((transpose(lambda.^2) *
(1./(4*pi*d.^2))))),2);
end
end
target(:,iX) = (targetRaw(:,iX)) ./ (cal);
end
fprintf(1,'Realized SNR:
%0.2f\n',20*log10(mean(10.^(realizedSNR/20))));

target = (transpose(target));
```

Appendix B

MATLAB for ISM / MUSIC Comparison

```
% MUSIC Versus IsolatedScattererPeakPicking
clear all; close all; clc

%% RNG control
s = RandStream('mt19937ar','Seed',1);
RandStream.setGlobalStream(s);
%% target
%target parameters
nScats = 3;
separSet =[0.025 .050 0.075]/2;
for iSep =1:numel(separSet)
separ = separSet(iSep);
maxDist = separ * (nScats-1)/2;
xscat=linspace(-maxDist,maxDist,nScats);
yscat=zeros(1,nScats);%maxDist*(rand(1,nScats)-1);%
ascat=ones(1,nScats);
zscat=zeros(1,nScats);
snrS = [0 -10 -20];
%% Plot Options
visbility = 'on';
load('colorsDistinct');
set(0,'DefaultAxesColorOrder',colors)
zoomLims = [-maxDist*2 maxDist*2;-maxDist*2 maxDist*2];
markerColor = [0 0 0];
markerSize = 25;
markerLineWidth = 1.75;
%% Output Base Directory
%enter the base directory for the output
outputDirBase = '';
iHyper = 0;
%% loops
for nAngSA = 1
nSASet = nScats:nScats+2;
for nFreqSA = nSASet;
nTotalSA = nAngSA*nFreqSA;
for iAlgo = 2
for iSNR = 1:numel(snrS)
SNR = snrS(iSNR);
for inIC = nTotalSA;%nScats:nTotalSA
RandStream.setGlobalStream(s);
%% Data collection parameters
minfre=4e9;
maxfre=8e9;
minphi=-80 * pi/180;
maxphi=80 * pi/180;
nPhi=32;
```

```
nFreq = 801;
theta = 45 * pi/180;

phi = linspace(minphi,maxphi,nPhi);
freq = linspace(minfre,maxfre,nFreq);
nphi = length(phi);
nfreq = length(freq);
SOL=3e8;
Beta=(2*pi*freq/SOL)*sin(theta);
lambda=SOL./(2*pi*freq);
[F, PHI] = meshgrid(freq,phi);
f_span = (maxfre-minfre);
fstep = f_span / (nFreq);
Res.SR = SOL / (2*f_span);
Res.projected = Res.SR/cos(theta);


%% Configure output
outputDir =
fullfile(outputDirBase,sprintf('Sep%0.2fcm\\%0.0fScatterers\\%0.
0fSNR\\%0.0fICs\\%0.0fAngSA_%0.0fFreqSA',...
separ*1E2,nScats,SNR,inIC,nAngSA,nFreqSA));
%power point
resFile = fullfile(outputDir,'1_Results.pptx');
if exist(resFile,'file')
iHyper = iHyper + 1;
prevFile = resFile;
fprintf('File has been saved: <a
href="matlab:winopen(''%s'')">%s</a>\n',prevFile,prevFile);
skip = true;

else
skip = false;
mkdir(outputDir)
isOpen  = exportToPPTX();
if ~isempty(isOpen),
% If PowerPoint already started, then close first and then open
a new one
exportToPPTX('close');
end
% http://www.mathworks.com/matlabcentral/fileexchange/40277-
exporttopptx
exportToPPTX('new','Dimensions',[12 6], ...
'Title','MUSIC Compare Results', ...
'Author','JHall', ...
'Subject','AlgorithmResults', ...
'Comments','This file has been automatically generated by
exportToPPTX');
end
if ~skip
```

```
notesText = sprintf('%0.0f Scatterers\n%0.0f FreqSA\n%0.0f
AngleSA\nsnr = %0.0f\n %0.0f ICs\n%s',nScats,nFreqSA,nAngSA,SNR,
inIC,outputDir);
clc
dateTimeStamp = datestr(now,30);
fprintf(1,'Output Dir: %s\n',outputDir);
diary(fullfile(outputDir,'Dairy.txt'))
fprintf(1,'DTS: %s\n',dateTimeStamp);
fprintf(1,'filename: %s\n', mfilename);
fprintf('Per Sample SNR = %0.2f\n',SNR);
%% Model Data Collection
[target, kxspace, kyspace] = ...
simulateISAR(-xscat, -yscat, zscat, ascat, phi(:),freq(:),
SOL,theta,nfreq,nphi,nScats,SNR);
%                           figure;imagesc(angle(target));
%% Resolutions and axes
rStepY = SOL./(2*f_span).*cos(theta);
rStepX = 0.5*rStepY;
Nsamps = 2^10;
Res.projectedInterp = Res.projected * nFreq/Nsamps;
xR = -((0:(Nsamps-1)) * rStepX - (Nsamps)/2 * rStepX);
yR = -((0:(Nsamps-1)) * rStepY - (Nsamps)/2 * rStepY);
rStept = SOL.*cosd(theta)./(f_span);
tRange = (nFreq) * rStept;
xRt = linspace(0,-tRange,nFreq);
xRt = xRt - mean(xRt);
[Range, Angle] = meshgrid(phi*180/pi,xRt);
%% ISAR Image
tInterp=tic;
x = (kxspace);
y = (kyspace);
v = target;
%rectangular grid
Xmin = min(x(:));
Xmax = max(x(:));
Ymin = min(y(:));
Ymax = max(y(:));
xi = linspace(Xmin, Xmax, Nsamps);
yi = linspace(Ymin, Ymax, Nsamps);
[XF, YF] = meshgrid(xi,yi);
interpType = 'linear';
extrapType = 'none';
FF =
scatteredInterpolant((x(:)),(y(:)),v(:),interpType,extrapType);
ZIF = FF((XF),(YF));%
ZIF(isnan(ZIF)) = 0;
ZIF(isinf(ZIF)) = 0;
Inner = sqrt(XF.^2 + YF.^2) < min(Beta);
Outer = sqrt(XF.^2 + YF.^2) > max(Beta);
ZIF(Inner) = 0;
```

```
ZIF(Outer) = 0;
fprintf(1,'Interpolation Time = %0.2f\n',toc(tInterp));
target_xformF = ifftshift(ifft2((ZIF)));
dataO = 20*log10(abs(target_xformF));
maxValO = max(dataO(:));
power.ZIF= sum(abs(ZIF(:)).^2);

%annalyze original position error
magScatOrigdB = zeros(1,nScats);
angScatOrigRad = zeros(1,nScats);
scatter2BGOrigdB = zeros(1,nScats);
scatter2ClutterOrigdB = zeros(1,nScats);
cluserDimOver2 = 0;
extractedAOI =
zeros(nScats,2*cluserDimOver2+1,2*cluserDimOver2+1);
%generate expanded region to find where scatterers occur in
original ISAR image
cluserDimOver2Expand = 5;
extractedAOIExpand =
zeros(nScats,2*cluserDimOver2Expand+1,2*cluserDimOver2Expand+1);
% findindices within zoomlims
logicalXlim = and(xR >= zoomLims(1),xR <= zoomLims(3));
logicalYlim = and(yR >= zoomLims(2),yR <= zoomLims(4));

valMaxOriginaldB = max((dataO(:)));
[rowLocMaxOriginal, columnLocMaxOriginal] = find((dataO) ==
valMaxOriginaldB);
max2BGOrigdB =
20*log10(abs(target_xformF(rowLocMaxOriginal,columnLocMaxOrigina
l))...
./ (sum(sum(abs(target_xformF(logicalXlim,logicalYlim)))) -
abs(target_xformF(rowLocMaxOriginal,columnLocMaxOriginal))));

formString = repmat(' %0.4f',1,nScats-1);
formStringInteger = repmat(' %0.0f',1,nScats-1);
formString = strcat(formString,' %0.4f\n');
formStringInteger = strcat(formStringInteger,' %0.4f\n');
fprintf(1,sprintf('\nOriginal Scatter X location:
%s',formString),xscat);
fprintf(1,sprintf('Original Scatter Y location:
%s',formString),yscat);
for iScat = 1:nScats

% find closest points
[minDistX(iScat), xScat(iScat)]  = min(abs(xR - xscat(iScat)));
[minDistY(iScat), yScat(iScat)]  = min(abs(yR - yscat(iScat)));
minDist(iScat) = hypot(minDistX(iScat),minDistY(iScat));
```

```
extractedAOIYExpand(iScat,:) = yScat(iScat)-
cluserDimOver2Expand:yScat(iScat)+cluserDimOver2Expand;
extractedAOIXExpand(iScat,:) = xScat(iScat)-
cluserDimOver2Expand:xScat(iScat)+cluserDimOver2Expand;
extractedAOIExpand(iScat,:,:) =
target_xformF(extractedAOIYExpand(iScat,:),...
extractedAOIXExpand(iScat,:));

[magScatOrigdB(iScat), locScatOrig(iScat)] =
max(20*log10(abs(extractedAOIExpand(iScat,:))));
[locScatOrigY(iScat) locScatOrigX(iScat)] =
find(squeeze(20*log10(abs(extractedAOIExpand(iScat,:,:)))) ==
magScatOrigdB(iScat));
scatter2BGOrigdB(iScat) =
20*log10(max(abs(extractedAOIExpand(iScat,:))).../ (sum(sum(abs(target_xformF(logicalXlim,logicalYlim)))) -
max(abs(extractedAOIExpand(iScat,:)))));
angScatOrigRad(iScat) =
mean(angle(extractedAOIExpand(iScat,locScatOrig(iScat))));


locScatOrigXM(iScat) =
xR(extractedAOIXExpand(iScat,locScatOrigX(iScat)));
locScatOrigYM(iScat) =
yR(extractedAOIYExpand(iScat,locScatOrigY(iScat)));

end
max2ScatRatioOriginal = valMaxOriginaldB - magScatOrigdB;
%% make ISAR Image
h = figure('Name','ISAR
Image','NumberTitle','off','visible','on');
whitebg(h,'k')
imagesc(xR,yR,data0);shading interp
title(sprintf('ISAR Image ::: SNR = %0.0f\nScatterer Separation
= %0.2f cm',SNR,separ*100))
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
plotdBlim = 50;
caxis([maxVal0-plotdBlim maxVal0]);
colorbar('location','SouthOutside')
hold on
for iMark = 1:nScats
scatter(xscat(iMark),yscat(iMark),'SizeData',markerSize,...
'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))

end
hold off
```

```
set(h,'visible','on')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
saveas(h,fullfile(outputDir,'ImageOrigISAR.fig'));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',h,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(h)
%% MUSIC
% decimate the data to 27 x 32
targetD=target(1:29:nfreq,:);
kyspaceD = kyspace(1:29:nfreq,:);
kxspaceD = kxspace(1:29:nfreq,:);
[~,REslab]=corrmtx(targetD(:),length(targetD(:))-1,'modified');
Res.Music = maxDist/4;
xvec=([zoomLims(1,1):Res.Music:zoomLims(1,2)]);
yvec=([zoomLims(2,1):Res.Music:zoomLims(2,2)]);
tmusic = tic;
[M10]=MUSIC2D_JH(REslab, nTotalSA, yvec, xvec, kyspaceD(:),
kxspaceD(:));
fprintf(1,'MUSIC Time = %0.2f\n',toc(tmusic));
dataM = 20*log10(abs(M10));
maxValM = max(dataM(:));
%% make MUSIC image
h = figure('Name','MUSIC
Image','NumberTitle','off','visible','on');
whitebg(h,'k')
imagesc(xvec,yvec,dataM);shading interp
title(sprintf('MUSIC Image ::: Signal Subspace Dim = %0.0f :::
SNR = %0.0f\nScatterer Separation = %0.2f
cm',nTotalSA,SNR,separ*100))
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
colorbar('location','SouthOutside')
hold on
for iMark = 1:nScats
scatter(xscat(iMark),yscat(iMark),'SizeData',markerSize,...
'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))
end
hold off
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
saveas(h,fullfile(outputDir,'ImageMUSIC.fig'));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',h,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(h)
%% Record peaks of MUSIC image
```

```
% find closest points
for iScat = 1:nScats
[minDistX_MUSIC(iScat), xScat_MUSIC(iScat)]  = min(abs(xvec -
xscat(iScat)));
[minDistY_MUSIC(iScat), yScat_MUSIC(iScat)]  = min(abs(yvec -
yscat(iScat)));
minDist_MUSIC(iScat) =
hypot(minDistX_MUSIC(iScat),minDistY_MUSIC(iScat));
end
tempO = dataM;
yPixSizeMUSIC = Res.Music;
xPixSizeMUSIC = Res.Music;
rowLocMax_MUSIC = zeros(nTotalSA,1);
columnLocMax_MUSIC = zeros(nTotalSA,1);
distMeasPixX_MUSIC = zeros(nTotalSA,nScats);
distMeasPixY_MUSIC = zeros(nTotalSA,nScats);
distMeasPixXY = zeros(nTotalSA,nScats);
distMeasPixXY_MUSIC = zeros(nTotalSA,nScats);
mindistMeter_MUSIC = zeros(nTotalSA,1);
closestScatIndex_MUSIC = zeros(nTotalSA,1);
distMeasMeterXY_MUSIC = zeros(nTotalSA,nScats);
for iIc = 1:nTotalSA
[tempMag tempIx] = nanmax(tempO(:));
[rowLocMax_MUSIC(iIc),columnLocMax_MUSIC(iIc)] = find(tempO ==
tempMag);
tempO(tempIx) = nan;
distMeasPixX_MUSIC(iIc,:) =
bsxfun(@minus,columnLocMax_MUSIC(iIc),xScat_MUSIC);
distMeasPixY_MUSIC(iIc,:) =
bsxfun(@minus,rowLocMax_MUSIC(iIc),yScat_MUSIC);
distMeasPixXY_MUSIC(iIc,:) =
hypot(distMeasPixX_MUSIC(iIc,:),distMeasPixY_MUSIC(iIc,:));
distMeasMeterXY_MUSIC(iIc,:) =
hypot(distMeasPixX_MUSIC(iIc,:)*xPixSizeMUSIC,...
distMeasPixY_MUSIC(iIc,:)*yPixSizeMUSIC);
[mindistMeter_MUSIC(iIc), closestScatIndex_MUSIC(iIc)] =
min(distMeasMeterXY_MUSIC(iIc,:));
end
%% Image overlay of MUSIC peaks
hMmO = figure('Name','ISAR Image of max of ICs
Overlay','NumberTitle','off','visible',visbility);
whitebg(hMmO,'k')
imagesc(xvec,yvec,dataM);shading interp
hold on
scatter(xvec(columnLocMax_MUSIC),
yvec(rowLocMax_MUSIC),ones(size(rowLocMax_MUSIC)),'SizeData',mar
kerSize,...
'Marker','x','LineWidth',markerLineWidth,'MarkerEdgeColor',[0 0
0],'MarkerFaceColor',[0 0 0]);
hold off
```

```
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
plotdBlimMUSIC = 20;
colorbar('location','SouthOutside')
hold on
for iMark = 1:nScats
scatter(xscat(iMark),yscat(iMark),'SizeData',markerSize,...
'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))
end
hold off
title(sprintf('Image of %0.0f MUSIC peaks overlaid on original
MUSIC Image\nSNR = %0.1f ::: Scatterer Separation = %0.2f
cm',...
inIC,SNR,separ*100), 'interpreter','none')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
set(hMmO,'InvertHardcopy','off')
saveas(hMmO,fullfile(outputDir,sprintf('ImageMaxs%0.0fMUSICOverl
ay.fig',inIC)));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',hMmO,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(hMmO)
%% Make Subapertures
nAnglesPerSA = floor(size(target,2)/nAngSA);
nFreqPerSA = floor(size(target,1)/nFreqSA);
fprintf(1,'\nNumber of Sensors per SA: %0.1f\n',nAnglesPerSA);
fprintf(1,'Number of Angle SAs: %0.1f\n',nAngSA);
fprintf(1,'Number of Freq  per SAs: %0.1f\n',nFreqPerSA);
fprintf(1,'Number of Freq SAs: %0.1f\n',nFreqSA);
iMaxFreq =  nFreqSA* nFreqPerSA;
fprintf(1,'BW of Freq SAs: %0.0f MHz\n',fstep * nFreqPerSA /
1E6);
fprintf(1,'Total Number of SAs: %0.1f\n',nTotalSA);
iMaxAng =  nAngSA* nAnglesPerSA;
clear targetSubSet targetSubSetTemp
SACount = 0;
for iterAng = 1:nAngSA
for iF = 1:nFreqSA
fIX = 1:nFreqSA;
iterFreq = fIX(iF);
SACount = SACount + 1;
iAng = nAnglesPerSA*(iterAng - 1)+1:nAnglesPerSA*iterAng;
iFreq = nFreqPerSA*(iterFreq - 1)+1:nFreqPerSA*iterFreq;
avgAngle(SACount) = mean(phi(iAng));
avgBeta(SACount) = mean(Beta(iFreq));
targetSubSetTemp = (target(iFreq,iAng));
targetSubSet(SACount,:) = targetSubSetTemp(:);
```

```
end
end
%% RobustICA
tol = 1e-2;      % termination threshold parameter
max_it = 1e5;    % maximum number of iterations per independent
component
kurtSign = zeros(1, nTotalSA);
kurtSign(1:nTotalSA) = 1;
performPCA = 0;
algoChoice = 'robustICA';
method = 'o';%{'o','r'}
entryCount = 1;
% robustICA -> X : observed signals (one row per signal, one
column per sample)
% http://www.i3s.unice.fr/~zarzoso/robustica.html
[algo, shat, A, iter, W] = robustica(targetSubSet, kurtSign,
tol,...
max_it, performPCA, method, 0, [], 0,entryCount);
for nIC=inIC%nIC = nTotalSA%nAnglesPerSA-1; % This allows
examination over various nIC
%% Remap data to k space
fprintf(1,'Number of "ICs": %0.1f\n',nIC);
reCompose = zeros(nIC, iMaxAng, iMaxFreq);
Yprime = zeros(nTotalSA,nFreqPerSA * nAnglesPerSA);
for iIC = 1:inIC
temp = zeros(size(shat));
temp(iIC,:) = shat(iIC,:);
Yprime = temp;
nAngsOut = nAnglesPerSA * nAngSA;
nFreqsOut = nFreqPerSA * nFreqSA;
SACount=0;
% look for highest weight subaperture
[val loc] = max(abs(W),[],1);
for iterAng = 1:nAngSA
    for  iF = 1:nFreqSA
        iterFreq = fIX(iF);
        SACount = SACount + 1;
        % Continuous sampling
        iAng = nAnglesPerSA*(iterAng -
1)+1:nAnglesPerSA*iterAng;
       iFreq = nFreqPerSA*(loc(iIC) - 1)+1:nFreqPerSA*loc(iIC);
        reCompose(iIC,iAng,iFreq) =  ...

transpose(reshape(Yprime(iIC,:),nFreqPerSA,nAnglesPerSA));
    end
end

kurt.Output(iIC) =
kurtosis(real(reCompose(iIC,reCompose(iIC,:)~=0)));
power.Output(iIC) = 20*log10(sum(abs(reCompose(iIC,:))));
```

```
end
reComposeWeighted = permute(reCompose,[1 3 2]);
%% output some stats
sumICsFreqDomain = squeeze(sum(real(reComposeWeighted),1));
kurt.SumICs = kurtosis(sumICsFreqDomain(:));
power.SumICs = 20*log10(sum(abs(reComposeWeighted(:)))) -
10*log10(size(reComposeWeighted,1));
                              stringFormat = strcat('Kurtosis of
output data:  ', repmat('%0.4f ',1,nTotalSA), ' \n');
fprintf(1,stringFormat,kurt.Output);
fprintf(1,'Kurtosis of sum of the output data: %0.4f
(dB)\n',kurt.SumICs);
                              stringFormat = strcat('Power of
output data:  ', repmat('%0.4f ',1,nTotalSA), ' (dB)\n');
fprintf(1,stringFormat,power.Output);
fprintf(1,'Power of sum of the output data: %0.4f
(dB)\n',power.SumICs);

%% Plot ICs
upsampleRatio = 1;
upsampleRatioFFT = 1;
cluserDimOver2 = 0;

xRuS = -((0:(Nsamps*upsampleRatio-1)) * rStepX -
(Nsamps*upsampleRatio)/2 * rStepX);
yRuS = -((0:(Nsamps*upsampleRatio-1)) * rStepY -
(Nsamps*upsampleRatio)/2 * rStepY);
target_xformFIC =
zeros(nIC,Nsamps*upsampleRatioFFT*upsampleRatio,Nsamps*upsampleR
atioFFT*upsampleRatio);
scatter2BGICsdB = zeros(nIC,nScats);
Scat2Back = zeros(nIC,nScats);
magScatICsdB = zeros(nIC,nScats);
angScatICsRad = zeros(nIC,nScats);
Scat2BackNorm = zeros(nIC,nScats);
scatter2ClutterICsdB = zeros(nIC,nScats);
scatter2ClutterICsdBNorm = zeros(nIC,nScats);
extractedAOIICs =
zeros(nIC,nScats,2*cluserDimOver2+1,2*cluserDimOver2+1);

clear ikurtSort
[valkurtSort, ikurtSort] = sort(kurt.Output,'descend');
max2ScatRatioICs = zeros(nIC,nScats);
max2BGICsdB = zeros(nIC,1);
rowLocMaxIC = zeros(nIC,1);
columnLocMaxIC = zeros(nIC,1);
valMaxICdB = zeros(nIC,1);
diffScat = zeros(nScats,nIC,nAngsOut,nFreqsOut);
```

```
distMeasPixX = zeros(nIC,nScats);
distMeasPixY = zeros(nIC,nScats);
distMeasPixXY = zeros(nIC,nScats);
distMeasMeterXY = zeros(nIC,nScats);
mindistMeter = zeros(nIC,1);
closestScatIndex = zeros(nIC,1);
%%
for iIC = 1:nIC
iSelect = iIC;
%% 2D interpolation of ICs
upsampleRatio = 1;
iAng = nAnglesPerSA*(iterAng - 1)+1:nAnglesPerSA*iterAng;
x =
(kxspace(1:size(reComposeWeighted,2),1:size(reComposeWeighted,3)
));
y =
(kyspace(1:size(reComposeWeighted,2),1:size(reComposeWeighted,3)
));
v = squeeze(reComposeWeighted(iSelect,:,:));
%rectangular grid
xiuS = linspace(min(xi),max(xi),upsampleRatio*numel(xi));
yiuS = linspace(min(yi),max(yi),upsampleRatio*numel(yi));
[XF, YF] = meshgrid(xiuS,yiuS);
interpType = 'linear';
extrapType = 'none';
FF = scatteredInterpolant(x(:),y(:),v(:),interpType,extrapType);
ZIF = FF(XF,YF);
ZIF(isnan(ZIF)) = 0;
ZIF(isinf(ZIF)) = 0;
Inner = sqrt(XF.^2 + YF.^2) < min(Beta);
Outer = sqrt(XF.^2 + YF.^2) > max(Beta);
ZIF(Inner) = 0;
ZIF(Outer) = 0;
%% upsample image of ICs
upsampleRatioFFT = 4;
target_xformFIC =
zeros(nIC,Nsamps*upsampleRatio*upsampleRatioFFT,Nsamps*upsampleR
atio*upsampleRatioFFT);
Res.projectedInterpUS = Res.projectedInterp *
1/upsampleRatioFFT;
nfftColIC = upsampleRatio*upsampleRatioFFT*Nsamps;%*size(ZIF,2);
nfftRowIC = upsampleRatio*upsampleRatioFFT*Nsamps;%*size(ZIF,1);
target_xformFIC(iIC,:,:) =
ifftshift(ifft2(ifftshift(ZIF),nfftRowIC,nfftColIC));
data = 20*log10(abs(squeeze(target_xformFIC(iIC,:,:))));
maxVal = max(data(:));
%find peaks in the IC data
valMaxICdB(iIC) = max(data(:));
```

```
[rowLocMaxIC(iIC), columnLocMaxIC(iIC)] = find(data ==
valMaxICdB(iIC));
xRuS = -(0:(Nsamps*upsampleRatio*upsampleRatioFFT-1)) *
rStepX/(upsampleRatio*upsampleRatioFFT)...
    + Nsamps/2 * rStepX;
yRuS = -(0:(Nsamps*upsampleRatio*upsampleRatioFFT-1)) *
rStepY/(upsampleRatio*upsampleRatioFFT)...
    + Nsamps/2 * rStepY;
% check power levels
power.reComposeWeighted(iIC) =
sum(abs(reComposeWeighted(iSelect,:)).^2);
power.ZIFIC(iIC)= sum(abs(ZIF(:)).^2);
power.RecomposeScaled(iIC) =
sum(abs(reComposeWeighted(iSelect,:)).^2)/(numel(Yprime(:))
/(Nsamps^2));
power.xformIC(iIC) =
sum(squeeze(abs(target_xformFIC(iIC,:))).^2);
%find peaks in the IC data
valMaxICdB(iIC) = max(data(:));
[rowLocMaxIC(iIC), columnLocMaxIC(iIC)] = find(data ==
valMaxICdB(iIC));
end

end
indexOfNonZero = reComposeWeighted(iSelect,:)~=0;
%% examine miss distance
%find peak pixel location
scatter2BGOrigdB(iScat) =
20*log10(max(abs(extractedAOIExpand(iScat,:)))...
./ (sum(sum(abs(target_xformF(logicalXlim,logicalYlim)))) -
max(abs(extractedAOIExpand(iScat,:)))));
% findindices within zoomlims
logicalXlim = and(xR >= zoomLims(1),xR <= zoomLims(3));
logicalYlim = and(yR >= zoomLims(2),yR <= zoomLims(4));
%find the magnitude of the scatterers in the ICs
for iScat = 1:numel(xscat)
[minDistX(iScat), xScat(iScat)]  = min(abs(xR - xscat(iScat)));
[minDistY(iScat), yScat(iScat)]  = min(abs(yR - yscat(iScat)));
cluserDimOver2Expand = 5;
extractedAOIExpand =
zeros(nScats,2*cluserDimOver2Expand+1,2*cluserDimOver2Expand+1);
extractedAOIYExpand(iScat,:) = yScat(iScat)-
cluserDimOver2Expand:yScat(iScat)+cluserDimOver2Expand;
extractedAOIXExpand(iScat,:) = xScat(iScat)-
cluserDimOver2Expand:xScat(iScat)+cluserDimOver2Expand;
extractedAOIExpand(iScat,:,:) =
target_xformF(extractedAOIYExpand(iScat,:),...
extractedAOIXExpand(iScat,:));

extractedAOIICs(iIC,iScat,:,:) = target_xformFIC(iIC,...
```

```
yScat(iScat)-cluserDimOver2:yScat(iScat)+cluserDimOver2,...
xScat(iScat)-cluserDimOver2:xScat(iScat)+cluserDimOver2);

[magScatICsdB(iIC,iScat), locScatICs(iIC,iScat)] =
max(20*log10(abs(extractedAOIICs(iIC,iScat,:))));
scatter2BGICsdB(iIC,iScat) =
20*log10(max(abs(extractedAOIICs(iIC,iScat,:)))...
.*  power.ZIF / power.ZIFIC(iIC) ...
./ (sum(sum(abs(target_xformFIC(iIC,logicalXlim,logicalYlim))))
- max(abs(extractedAOIICs(iIC,iScat,:)))));
angScatICsRad(iIC,iScat) =
(angle(extractedAOIICs(iIC,iScat,locScatICs(iIC,iScat))));
end
Scat2Back(iIC,:) = scatter2BGICsdB(iIC,:);
Scat2BackNorm(iIC,:) = scatter2BGICsdB(iIC,:)- scatter2BGOrigdB;


max2BGICsdB(iIC) =
20*log10(abs(target_xformFIC(iIC,rowLocMaxIC(iIC),
columnLocMaxIC(iIC)))...
.*  power.ZIF / power.ZIFIC(iIC) ...
./ (sum(sum(abs(target_xformFIC(iIC,logicalXlim,logicalYlim))))
- abs(target_xformFIC(iIC,rowLocMaxIC(iIC),
columnLocMaxIC(iIC)))));

max2ScatRatioICs(iIC,:) = valMaxICdB(iIC) - magScatICsdB(iIC,:);

formString = repmat(' %0.4f',1,nScats-1);
formString = strcat(formString,' %0.4f\n');
fprintf(1,sprintf('\nScatters to BG ratio
IC%%0.0f:%s',formString),iIC,scatter2BGICsdB(iIC,:));
fprintf(1,'Max to BG ratio IC%0.0f:
%0.4f\n',iIC,max2BGICsdB(iIC));
fprintf(1,'Max Value IC%0.0f: %0.4f\n',iIC,valMaxICdB(iIC));
fprintf(1,'Max Location IC%0.0f: Row %0.0f, Col %0.0f\n',...
iIC,rowLocMaxIC(iIC),columnLocMaxIC(iIC));
fprintf(1,sprintf('Scatters Magnitude (dB)
IC%%0.0f:%s',formString),iIC,magScatICsdB(iIC,:));
fprintf(1,sprintf('Scatters Phase Angle (Rad)
IC%%0.0f:%s',formString),iIC,angScatICsRad(iIC,:));

for iIC = 1:nIC
% gauge distance
% between maximum pixel
% and the known
% locations of the
% scatterers in the
% scene
distMeasPixX(iIC,:) = bsxfun(@minus,columnLocMaxIC(iIC),xScat);
```

```
distMeasPixY(iIC,:) = bsxfun(@minus,rowLocMaxIC(iIC),yScat);
distMeasPixXY(iIC,:) =
hypot(distMeasPixX(iIC,:),distMeasPixY(iIC,:));
distMeasMeterXY(iIC,:) =
hypot(bsxfun(@minus,xRuS(columnLocMaxIC(iIC)),xR(xScat)),...
bsxfun(@minus,xRuS(rowLocMaxIC(iIC)),xR(yScat)));
[mindistMeter(iIC), closestScatIndex(iIC)] =
min(distMeasMeterXY(iIC,:));
fprintf(1,sprintf('\nMinimum Miss Distance (M)
IC%%0.0f:%s',formString),iIC,mindistMeter(iIC,:));
fprintf(1,sprintf('Closest Known Scatterer
IC%%0.0f:%s',formStringInteger),iIC,closestScatIndex(iIC,:));
end
fprintf(1,'\n');
% Plot Image of the max of all ICs
dataMaxs = ones(Nsamps*upsampleRatioFFT,Nsamps*upsampleRatioFFT)
* 20*log10(eps);
for iMax = 1:numel(rowLocMaxIC)
dataMaxs(rowLocMaxIC(iMax), columnLocMaxIC(iMax)) = 0;
end
hICm = figure('Name','ISAR Image of max of
ICs','NumberTitle','off','visible',visbility);
whitebg(hICm,'k')
imagesc(xRuS,yRuS,dataMaxs);shading interp
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
caxis([-plotdBlim 0]);colorbar('location','SouthOutside')
hold on
for iMark = 1:nScats
scatter(xscat(iMark),yscat(iMark),'SizeData',markerSize,...
'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))

end
hold off
title(sprintf('Image of Maxs of %0.0f ICs ::: SNR =
%0.1f\nScatterer Separation = %0.2f cm',nIC,SNR,separ*100),
'interpreter','none')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
set(hICm,'InvertHardcopy','off')
saveas(hICm,fullfile(outputDir,sprintf('ImageMaxsICs%0.0f.fig',n
IC)));
slideNum = exportToPPTX('addslide'); %#ok<NASGU>
exportToPPTX('addpicture',hICm,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(hICm)
% Overlay Image of the max of
% all ICs on original ISAR
```

```
hICmO = figure('Name','ISAR Image of max of ICs
Overlay','NumberTitle','off','visible',visbility);
whitebg(hICmO,'k')
imagesc(xR,yR,dataO);shading interp
hold on
scatter(xRuS(columnLocMaxIC),
yRuS(rowLocMaxIC),ones(size(rowLocMaxIC)),'SizeData',markerSize,
...
'Marker','x','LineWidth',markerLineWidth,'MarkerEdgeColor',[0 0
0],'MarkerFaceColor',[0 0 0]);
hold off
xlabel('Down Range Distance (m)');
ylabel('Cross Range Distance (m)')
colormap('parula')
caxis([maxValO-plotdBlim
maxValO]);colorbar('location','SouthOutside')
hold on
for iMark = 1:nScats
scatter(xscat(iMark),yscat(iMark),'SizeData',markerSize,...
'Marker','o','LineWidth',markerLineWidth,'MarkerEdgeColor',1-
colors(iMark,:))
end
hold off
title(sprintf('Image of Maxs of %0.0f ICs Overlaid on Original
ISAR Image ::: SNR = %0.1f\nScatterer Separation = %0.2f
cm',nIC,SNR,separ*100), 'interpreter','none')
set(gca,'XLim',zoomLims(1,:))
set(gca,'YLim',zoomLims(2,:))
set(hICmO,'InvertHardcopy','off')
saveas(hICmO,fullfile(outputDir,sprintf('ImageMaxsICs%0.0fOverla
y.fig',nIC)));
slideNum = exportToPPTX('addslide');
exportToPPTX('addpicture',hICmO,'Scale','maxfixed');
exportToPPTX('addnote',notesText);
delete(hICmO)

%% close power point
newFile = exportToPPTX('save',fullfile(outputDir,'1_Results'));
exportToPPTX('close')
iHyper = iHyper +1;
hyperLinks{iHyper} = sprintf('<a
href="matlab:winopen(''%s'')">%s</a>\n',newFile,newFile);
fprintf('File has been saved: <a
href="matlab:winopen(''%s'')">%s</a>\n',newFile,newFile);
%% Output Variables

save(fullfile(outputDir,'Variables'),'angScatICsRad','angScatOri
gRad','hyperLinks',...
'kurt','magScatICsdB','magScatOrigdB','max2BGICsdB','max2BGOrigd
B','max2ScatRatioICs',...
```

```
'max2ScatRatioOriginal','notesText','outputDir','Scat2Back','Sca
t2BackNorm',...
'scatter2BGICsdB', 'scatter2BGOrigdB', 'scatter2ClutterICsdB',
'scatter2ClutterICsdBNorm',...
'valMaxICdB','valMaxOriginaldB','power',...
'rowLocMaxIC',
'columnLocMaxIC','xScat','yScat','rowLocMax_MUSIC','columnLocMax
_MUSIC',...
'distMeasMeterXY', 'distMeasPixX',
'distMeasPixY','locScatOrigXM','locScatOrigYM',...
'distMeasPixXY_MUSIC', 'distMeasPixX_MUSIC',
'distMeasPixY_MUSIC',...
'mindistMeter','closestScatIndex','mindistMeter_MUSIC','closestS
catIndex_MUSIC',...
'xscat','yscat','W','A','avgBeta','avgAngle')
end
end
end
end
end
end
end

function [Es, kxspace, kyspace] = simulateISAR(xscat, yscat,
zscat, ascat,...
phi, freq, SOL,theta,nfreq,nphi,nScats,SNR)
kxspace= (2*pi.*freq./SOL) * sin(theta)*sin(phi') ;
kyspace= (2*pi.*freq./SOL) * sin(theta)*cos(phi') ;
if isinf(SNR)
Es= zeros(nfreq,nphi);
else
noiseamp = 10^(-SNR/20)/sqrt(2) ; % (the sqrt(2) accounts for
real+imaginary powers)
Es = noiseamp*(randn(nfreq,nphi)+1j*randn (nfreq,nphi));
end
for iphi=1:nphi
for ifreq=1:nfreq
k=[kxspace(ifreq,iphi) kyspace(ifreq,iphi) 0];
for m=1:nScats
r=[xscat(m) yscat(m) zscat(m)];
Es(ifreq,iphi)=Es(ifreq,iphi)+ascat(m)*exp(-2j*k*r.');
end
end
end


function [ M Mflip ] = MUSIC2D_JH( R, scats, xvec, yvec,
kxspace, kyspace )
[~,~,v]=svd(R);
Vn=v(:,scats+1:end);
```

```
[~,P]=size(Vn);
den=zeros(length(xvec),length(yvec));
for x=1:length(xvec)
for y=1:length(yvec)
r=[xvec(x) yvec(y)];
ab = exp(-2j*(kxspace'*r(1)+ kyspace'*r(2)));
b=ab(:);
for m=1:P
den(x,y)=den(x,y)+(abs(conj(b')*Vn(:,m)))^2;
end
end
end
M=1./den;
Mflip =rot90(flipud(M.'),1);
end
```

[1] N. N. Bojarski, "A survey of physical optics inverse scattering identity," *IEEE Transactions on Antennas & Propagation,* Vols. AP-30, no. 5, pp. 980-989, September 1982.

[2] R. Schmidt, "Multiple emitter location and signal parameter estimation," *Antennas and Propagation, IEEE Transactions on,* vol. 34, no. 3, pp. 276-280, Mar 1986.

[3] F. T. Ulaby and M. C. Dobson, Handbook of Radar Scattering Statistics for Terrain, Norwood: Artech House, 1989.

[4] J. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *Radar and Signal Processing, IEE Proceedings F,* vol. 140, no. 6, pp. 362-370, Dec 1993.

[5] V. Zarzoso and P. Common, "Robust Independent Component Analysis by Iterative Maximization of the Kurtosis Contrast With Algebraic Optimal Step Size," *IEEE Transactions on Neural Networks,* vol. 21, no. 2, pp. 248-261, Feb. 2010.

[6] J. V. Stone, Independent Component Analysis: A Tutorial Introduction, Cambridge, MA: The MIT Press, 2004.

[7] J. N. Ash, E. Ertin, L. C. Potter and E. G. Zelnio, "Wide-Angle Synthetic Aperture Radar Imaging: Models and algorithms for anisotropic scattering," *IEEE Signal Processing Magazine,* pp. 16-26, July 2014.

[8] M. Çetin, I. Stojanović, N. Ö. Önhon, K. R. Varshney, S. Samadi, W. C. Karl and A. S. Willsky, "Sparsity-Driven Synthetic Aperture Radar Imaging: Reconstruction, autofocusing, moving targets, and compressed sensing," *IEEE Signal Processing Magazine,* pp. 27-40, July 2014.

[9] G. Krieger, "MIMO-SAR: Opportunities and Pitfalls," *IEEE Transactions on Geosecience and Remote Sensing,* vol. 52, no. 5, pp. 2628-2645, May 2014.

[10] D. Cristallini, D. Pastina and P. Lombardo, "Exploiting MIMO SAR Potentialities With Efficient Cross-Track Constellation Configurations for Improved Range Resolution," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 49, no. 1, pp. 38-52, Jan. 2011.

[11] M. Younis, G. Kreiger and A. Moreira, "MIMO SAR techniques and trades," in *Proceedings of the 10th European Radar Conference,* Nuremberg, Germany, 2013.

[12] X. X. Zhu and R. Bamler, "Demonstration of Super-Resolution for Tomographic SAR Imaging in Urban Environment," *IEEE Transactions*

on Geoscience and Remote Sensing, vol. 50, no. 8, pp. 3150-3157, Aug. 2012.

[13] C. Fischer, M. Andres, H.-L. Bloecher, J. Dickmann and W. Menzel, "Adaptive super-resolution with a synthetic aperture antenna," in *European Radar Conference (EuRAD)*, Amsterdam, The Netherlands, 2012.

[14] J. Mitchell and S. Tjuatja, "Super-resolution ISAR imaging using polarimetric techniques for subspace dimensionality," in *IEEE Geoscience and Remote Sensing Symposium (IGARRSS)*, Munich, Germany, 2012.

[15] C. A. Balanis, Advanced Engineering Electromagnetics, John Wiley & Sons, Inc., 1989.

[16] S. K. Chaudhuri and W.-M. Boerner, "A polarimetric model for the recovery of the high-frequency scattering centers from bistatic-monostatic scattering matrix data," *IEEE Transactions on Antennas and Propagation*, vol. 1, pp. 87-93, 1987.

[17] L. C. Potter and R. L. Moses, "Attributed scattering centers for SAR ATR," *IEEE Transactions on Image Processing*, vol. 1, pp. 79-91, 1997.

[18] J.-S. Lee, T. L. Ainsworth and Y. Wang, "Generalized Polarimetric Model-Based Decompositions Using Incoherent Scattering Models," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 5, pp. 2474-2491, May 2014.

[19] S. A. Schuster, An Introduction to the Theory of Optics, London: Edward Arnold, 1904.

[20] R. W. Wood, Physical Optics, New York: The MacMillan Company, 1911.

[21] M. Born and E. Wolf, Principles of Optics, 7th ed., Cambridge: Cambridge University Press, 1999.

[22] J. B. Keller, "Geometrical Theory of Diffraction," *Journal of the Optical Society of America*, vol. 52, no. 2, pp. 116-130, Feb. 1962.

[23] D. Middleton, "A statistical theory of reverberation and similar first-order scattered fields--I: Waveforms and the general process," *IEEE Transactions on Information Theory*, Vols. IT-13, no. 3, pp. 372-392, 1967.

[24] D. Middleton, "A statistical theory of reverberation and similar first-order scattered fields--II: Moments, spectra and special distributions," *IEEE Transactions on Information Theory*, Vols. IT-13, no. 3, pp. 393-414, 1967.

[25] D. Middleton, "A statistical theory of reverberation and similar first-order scattered fields--III: Waveforms and fields," *IEEE Transactions on Information Theory,* Vols. IT-18, no. 1, pp. 35-67, 1972.

[26] D. Middleton, "A statistical theory of reverberation and similar first-order scattered fields--IV: Statistical models," *IEEE Transactions on Information Theory,* Vols. IT-18, no. 1, pp. 68-90, 1972.

[27] M. McCoy, V. Cevher, Q. Dinh, A. Asaei and L. Baldassarre, "Convexity in Source Separation : Models, geometry, and algorithms," *Signal Processing Magazine, IEEE,* Vols. 31, no.3, pp. 87,95, May 2014.

[28] S. R. DeGraaf, "SAR imaging via modern 2-D spectral estimation methods," *IEEE Transactions on Image Processing,* vol. 7, no. 5, pp. 729-760, May 1998.

[29] S. M. Kay and S. L. Marple, Jr., "Spectrum analysis—A modern perspective," *Proceedings of IEEE,* vol. 69, no. 11, pp. 1380-1419, NOV. 1981.

[30] R. O. Schmidt, "A Signal Subspace Approach to Multiple Emitter Location and Spectral Estimation," 1981.

[31] G. E. Shilov, Linear Algebra, R. A. Silverman, Ed., Mineola, NY: Dover Publications, 1977.

[32] A. Hyvärinen, J. Karhunen and E. Oja, Independent Component Analysis, New York: John Wiley & Sons, Inc., 2001.

[33] M. Anderson, T. Adah and X.-L. Li, "Joint Blind Source Separation With Multivariate Gaussian Model: Algorithms and Performance Analysis," *IEEE Transactions on Signal Processing,* vol. 60, no. 4, pp. 1672-1683, April 2012.

[34] T. Adali, M. Anderson and G.-S. Fu, "Diversity in Independent Component and Vector Analyses: Identifiability, algorithms, and applications in medical imaging," *IEEE Signal Processing Magazine,* pp. 18-33, May 2014.

[35] V. Zarzoso and P. Common, "Comparative Speed Analysis of FastICA," in *Proc. 7th Int. Conf. Ind. Compon. Anal. Signal Separat.*, London, 2007.

[36] J. Hall and S. Tjuatja, "Analysis of ISAR imaging using combined MUSIC and ICA algorithms," in *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*, Vancouver, Jul. 2011.

[37] J.-S. Lee and E. Pottier, Polarimetric Radar Imaging: From Basics to Applications, Boca Raton, FL: CRC Press, 2009.

[38] E. F. Knott, J. Schaeffer and M. Tuley, Radar Cross Section, SciTech Publishing, 2004.

[39] J. Hall and S. Tjuatja, "MUSIC and ICA algorithms applied to full polarimetric ISAR imaging," in *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*, July 2012.

[40] M. Novey and T. Adali, "Using complex-valued ICA to efficiently combine radar polarimetric data for target detection," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 19-24 April 2009.

[41] N. Besic, G. Vasile, J. Chanussot and S. Stankovic, "Polarimetric Incoherent Target Decomposition by Means of Independent Component Analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1236-1247, March 2015.

[42] S. C. Douglas, "Fixed-point algorithms for the blind separation of arbitrary complex-valued non-Gaussian signal mixtures," *EURASIP Journal on Applied Signal Processing* , Jan. 2007.

[43] Z. Ding and T. Nguyen, "Stationary points of a kurtosis maximization algorithm for blind signal separation and antenna beamforming," *IEEE Transactions on Signal Processing*, vol. 48, no. 6, pp. 1587-1596, June 2000.

[44] E. Gonen and J. Mendel, "Applications of cumulants to array processing. III. Blind beamforming for coherent signals," *IEEE Transactions on Signal Processing*, vol. 45, no. 9, pp. 2252-2264, Sept. 1997.

[45] M. Dogan and J. Mendel, "Applications of cumulants to array processing .I. Aperture extension and array calibration," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1200-1216, May 1995.

[46] M. Dogan and J. Mendel, "Applications of cumulants to array processing. II. Non-Gaussian noise suppression," *IEEE Transactions on Signal Processing*, vol. 43, no. 7, pp. 1663-1676, Jul 1995.

[47] P. Chevalier, L. Albera, A. Ferreol and P. Comon, "On the virtual array concept for higher order array processing," *IEEE Transactions on Signal Processing*, vol. 53, no. 4, pp. 1254-1271, April 2005.

[48] P. Chevalier and A. Ferreol, "On the virtual array concept for the fourth-order direction finding problem," *IEEE Transactions on Signal Processing*, vol. 47, no. 9, pp. 2592-2595, Sept. 1999.

[49] M. Skolnik, Ed., Radar Handbook, Third ed., McGraw-Hill Companies, 2008.

Publications Supporting this Work

J. Hall and S. Tjuatja, “Analysis of ISAR imaging using combined MUSIC and ICA algorithms,” Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International, 2011.

J. Hall and S. Tjuatja, “MUSIC and ICA algorithms applied to full polarimetric ISAR imaging,” Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International, 2012.

J. Hall and S. Tjuatja, “On the application of complex ICA to Radar data in the frequency domain,” Geoscience and Remote Sensing Symposium (IGARSS), 2014 IEEE International, 2014.

Biographical Information

Jeffrey Hall earned a Bachelor of Science in Electrical Engineering at the University of Texas at San Antonio in 2002. He then went to work for Lockheed Martin Corporation and continued work toward a Master's degree in Electrical Engineering. This degree was awarded to him in 2009 by the University of Texas at Arlington. Also in 2009, Jeffrey went to work for the Raytheon Corporation. During the subsequent years, Jeffrey worked on completion of a PhD degree in Electrical Engineering. Jeffrey completed this degree program in 2016 at the University of Texas at Arlington.

Jeffrey's areas of interest are radar systems engineering, design and signal processing. His experience spans a broad range in the radar domain. He has experience in antenna and RF design, radar calibration, radar signal processing, and radar integration and test. His research has focused on physics based scattering center extraction and adaptive signal processing.