UNIVERSITY OF OKLAHOMA
GRADUATE COLLEGE

ADVANCES IN SYNTHETIC APERTURE RADAR FROM A WAVENUMBER PERSPECTIVE

A DISSERTATION
SUBMITTED TO THE GRADUATE FACULTY
in partial fulfillment of the requirements for the
Degree of
DOCTOR OF PHILOSOPHY

By

WILLIAM GRIFFIN DOWER
Norman, Oklahoma
2017

ADVANCES IN SYNTHETIC APERTURE RADAR FROM A WAVENUMBER PERSPECTIVE

A DISSERTATION APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

Dr. Mark Yeary, Chair

Dr. Jeffrey Basara

Dr. Caleb Fulton

Dr. Nathan Goodman

Dr. Brian Rigling

Dr. Hjalti Sigmarsson

This dissertation is dedicated to Maverick and Winnie.

Acknowledgments

I'd like to thank my Mom and Dad, Bill and Karen Dower, for their support of my education through all these years. Both of my parents have encouraged me in very kind and thoughtful ways. They've been there for me when things have gone well and when they have not. I'm so fortunate to have such great parents. My grandmother Lynne Dower, great aunt Barbara Alexander, and great grand parents Leo and Pearl Alexander always stressed the importance of my education while growing up and have a profound influence on my life today.

I'd like to thank my PhD advisor, Dr. Mark Yeary, who inspired me to finish my PhD at OU after a prior attempt where I failed to do so. There was a point in my life where I believed I would not reach this one. Dr. Yeary helped me every step of the way, when I had doubts or concerns we would make a chart or a timeline that would help me see how to accomplish this goal. There were countless conversations about which direction my research would go or different paths I should pursue. I know that completing this degree will have a profound impact on my life and I will always be grateful to Dr. Yeary.

I would also like to thank the members of my committee who offered very thoughtful advice about my research and guidance on non-research related topics as well. Working in the Radar Innovations Lab (RIL) offered many great conversations about radar technology from each member. Although Dr.

Rigling is at Wright State University we emailed back and forth about his research as well. Over a tenth of the citations in this document have his name on them. I am very grateful for what I learned from each member. These members are Dr. Jeffrey Basara, Dr. Caleb Fulton, Dr. Nathan Goodman, Dr. Brian Rigling, and Dr. Hjalti Sigmarsson. Also I'd like to thank Dr. Jorge Salazar-Cerreño, Dr. Jessica Ruyle and Simon Duthoit for their contributions to the bistatic SAR testing in the anechoic chamber at the RIL. They offered their time, knowledge, and resources which helped me complete these tests. Also Dr. Michele Eodice and the staff at OU's Writing Center helped me edit so much of this document. Dr. Eodice created a STEM writing group, which I was a member of. Without her and the staff at the Writing Center this would have been a much more difficult endeavor.

There are also many people from elementary school to the universities that I attended and the places I worked who had unique influences on my education and played a significant role in my life that I would like to thank. Dr. Raymond Samaniego and Bryan Anderson mentored me while working and always encouraged me to continue to pursue a PhD. Dr. Shannon Blunt, my Masters thesis advisor, took me to my first IEEE Radar conference and introduced me to academic research in radars. I can remember many hours spent in his office learning about new radar technology. I'd like to thank Dr. Robert Fraga or "Coach", who coached me and so many students at Baker University through difficult math classes and onto Engineering degrees at places like the University of Kansas. My first grade teacher Ms. Shelley Walker who made learning fun and enjoyable. And my Mom, an educator, who always talked about and exemplified life long learning.

Table of Contents

List of Tables

List of Figures

List Of Abbreviations

ACP	Aperture Center Point
A/D	Analog to Digital
APC	Adaptive Pulse Compression
BPA	Backprojection Algorithm
CPI	Coherent Pulse Interval
CUDA	Compute Unified Device Architecture
dB	Decibels
DBS	Doppler Beam Sharpening
DF	Decimation Factor
ESA	European Space Agency
EM	Electro Magnetic
FBP	Fast Back Projection
FDWBPA	Fast Decimated Wavenumber Back Projection Algorithm
FFBPA	Fast Factorized Back Projection Algorithm
FFT	Fast Fourier Transform
GPU	Graphical Processing Unit
I	In-phase
IF	Intermediate Frequency
IFFT	Inverse Fast Fourier Transform

IFSAR	Introfermetic SAR
IPR	Impulse Response
ISLR	Integrated SideLobe Ratio
ISAR	Inverse Synthetic Aperture Radar
LO	Local Oscillator
LOS	Line Of Sight
LFM	Linear Frequency Modulated
NED	North East Down
NVIDA	Nvidia Corporation
OU	The University of Oklahoma
PFA	Polar Format Algorithm
PGA	Phase Gradient Autofocus
PRI	Pulse Repetition Interval
PSR	Peak to Sidelobe Ratio
Q	Quadrature
RAM	Random Access Memory
RCS	Radar Cross Section
RIL	Radar Innovations Lab
RVP	Residual Video Phase
RX	Receiver
SAR	Synthetic Aperture Radar
SRP	Scene Reference Point
STEM	Science Technology Engineering Math
TDC	Time Domain Correlation
TX	Transmitter

Abstract

This dissertation examines the wavenumber domain of Synthetic Aperture Radar (SAR) images. This domain is the inverse Fourier transform domain of a SAR image. The dissertation begins with the radar receiver's signal model and develops equations describing the wavenumber domain of a SAR image produced by a generalized bistatic and monostatic SAR system. Then, closed form expressions for bistatic synthetic aperture radar spatial resolution of a generalized system from the wavenumber domain are developed. These spatial resolution equations have not previously appeared in the literature. From these equations, significant resolution is found in both range and cross-range forecasting a forward-scatter bistatic SAR image when the elevation angles of each bistatic platform are significantly different. Next, wavenumber and time domain image formation algorithms are discussed. Developed within this dissertation is a wavenumber preprocessing method that increases the speed of the Back Projection Algorithm (BPA). This preprocessing method takes advantage of deramped SAR radar returns and their polar wavenumber format. This new algorithm is called the Fast Decimated Wavenumber Back Projection Algorithm (FDWBPA). Matlab functions are included to implement this algorithm, simulate bistatic SAR images and process the data from anechoic chamber tests demonstrating forward scatter resolution.

Chapter 1

Introduction

Prior to the widespread application of the back projection algorithm (BPA), it was necessary to understand Synthetic Aperture Radar (SAR) from a wavenumber or k-space perspective before image formation. Initially, SAR systems would record radar returns in a frequency domain by stretch processing the received signal in the range dimension while each coherently received radar return would have a small linear phase shift due to the radar's motion. Initially these radar returns were recorded on photographic film and the Fourier properties of optical lens were exploited to process the radar data. A light source shined through the film and optical lens would then produce a SAR image projected onto a flat surface [1]. The initial radar image processing method, Doppler Beam Sharpening (DBS), used these optical lens to demonstrate that radar imaging was possible in 1952 and 1953 [2]. Later, the integrated circuit led to faster processors that could handle the large amount of Fast Fourier Transforms (FFTs) needed to process the radar return data into a digital SAR image. To use these optical lens or FFTs to produce a SAR image, an understanding of a SAR image's Fourier properties in the wavenumber or k-space domain was necessary.

Today, processors cannot only handle the large amount of FFTs needed to produce a SAR image using a Fourier based algorithm, but can also process the radar return data through computationally intensive time domain inversion algorithms like the BPA or time domain correlation (TDC) algorithms. Today's SAR researchers can find themselves working solely with these algorithms due to their straight forward approach to image formation and the algorithm's ability to produce very high quality images. Soumekh describes the issues for engineers working solely in the time domain as:

> Unfortunately, the spectral shape and properties of the spotlight SAR signal are *transparent* to the users of the TDC and back-projection SAR image formation methods. Due to this fact these methods have provided a simplistic shortcut for those who do not fully understand SAR signal theory, its Nyquist sampling rates in measurement (t, u) domain and the reconstruction (x, y) domain, and the preprocessing of the measured SAR data which could enhance the quality of reconstructed SAR images [3, pp. 359–360].

An understanding of a SAR image's Fourier properties in the frequency or k-space domain is needed for BPA and TDC algorithms which will be demonstrated in this dissertation.

Preceding the author's dissertation research at the University of Oklahoma (OU), he worked in industry on SAR systems. Two different experiences motivated the research in this dissertation. First, a research and development project involved the development of a three-dimensional (3D) monostatic SAR mode using the BPA, where the author was a member of a team of engineers. Initially, the team had difficulty understanding how resolution would be achieved in each of the three dimensions of the mode. Thinking of the SAR mode for a k-space perspective helped aid the understanding. After this project the author wanted a deeper understanding of SAR k-spaces.

The second project involved mission planning for a bistatic SAR system.

Finding a closed form equation for both range and cross-range resolution for this study was difficult. Some equations involved determining the k-space a priori, while other equations are for specific configurations of a bistatic system. When comparing these bistatic resolution equations to monostatic SAR resolution equations, there are well-known generalized closed form range and cross-range equations for monostatic SAR. These well known monostatic SAR resolution equations give engineers a quantitative understanding of the system specifications necessary to generate resolution. From these industry experiences, the author decided to develop an understanding of bistatic SAR k-spaces to aid in uncovering a new understanding of SAR. From this understanding the author also developed a preprocessing method to decrease the computational complexity of the BPA while the received SAR signal is in the k-space domain.

Two significant developments were made in the author's research. First, the author developed generalized closed form, range and cross-range, bistatic SAR resolution equations which have not previously appeared in the literature. In [4], the authors present a time-domain gradient method to arrive at a bistatic range resolution equation with a similar form as the dissertation author but doesn't present it as a generalized equation. The author of this dissertation derived generalized closed form bistatic ground range and cross range resolution equations using the bistatic k-space. Similar to [4], the dissertation author suspended the idea of the bistatic plane where many authors have defined the bistatic angle. Instead the bistatic angle was defined within the ground plane where the final image would exist. By defining the bistatic angle in the ground plane, radar engineers are able to gain an understanding of the effects of the elevation angles of the bistatic transmitter and receiver for both range and cross-range resolution. These equations led to a convenient rule of thumb for determining the bandwidth, azimuth extent, and transmit frequency if the azimuth extent is equal for the transmitter and receiver which is always the case for monostatic SAR. These observations were demonstrated in simulation and within the large anechoic chamber at the University of Oklahoma, as well.

The second development within this disertation used an understanding of k-spaces to create a fast BPA to process SAR images. In [5], the authors developed a preprocessing method to interpolate and combine neighboring PRIs in the time domain. Having developed an understanding of k-spaces in the previous bistatic work, the author realized that an interpolation step could be removed if the data was processed differently. For high resolution SAR images that have a reasonable swath width, stretch processing is a common method to minimize intermediate frequency (IF) bandwidth usage. The SAR data saved from the analog to digital (A/D) converter is in a polar formatted k-space domain. Neighboring stretch processed, k-space domain, uncompressed radar returns are summed to create a similar fast back projection algorithm as the one described in [5]. This preprocessor is also more efficient because it avoids an algorithmically intensive interpolation step and works on the native data format found in many high resolution SAR systems. This algorithm and it's results are detailed within this dissertation, as well.

This dissertation is split into eight chapters, including this introduction chapter. The second chapter outlines the receive signal model and defines the k-space. This is done for monostatic and bistatic SAR and by taking the derivative of the phase of the receive signal. Also a stretch processing signal model is developed and the k-space is found from this model as well. The third chapter discusses SAR resolution and develops generalized closed form bistatic ground range and cross-range resolution equations. Chapter 4 outlines the bistatic polar format algorithm developed in [6] which was used to process bistatic SAR images in later chapters. This chapter also describes the fast BPA algorithm described in the previous paragraph. Chapter 5 contains simulated SAR results and imagery while Chapter 6 contains actual SAR images created from the anechoic chamber at OU and processed data used to demonstrate the results from the fast BPA algorithm. Chapter 7 presents future considerations for SAR systems. The last section in Chapter 7 presents an idea for research from the author's Masters dissertation at the University of Kansas and SAR cross range processing. Then the dissertation summarizes its key finding and ends with concluding remarks in Chapter 8.

Chapter 2

SAR Signal Model

Analysis of the signal model shows how a SAR k-space is formed from a Synthetic Aperture Radar (SAR) point target return model. In the literature, many derivations for SAR receive signal models are often derived in terms of ranges to specific locations within the scene or imaged area. These ranges are dependent on the two-way propagation time to these locations. A signal model which is applicable to both monostatic and bistatic systems can be derived in terms of the time delays within the signal model. After the signal model is derived, monostatic or bistatic ranges may be substituted where appropriate.

2.1 Generalized Bistatic SAR System

A bistatic Synthetic Aperture Radar (SAR) consists of two radar systems, a transmitter and receiver. A bistatic synthetic aperture is created by one platform transporting the transmitter and a second platform transporting the receiver which changes the bistatic system's position at each transmitted and received radar pulse. One of the two bistatic systems could also remain stationary while moving the other bistatic system to create the synthetic aperture. A means of coordination between the transmitter and receiver is assumed for each pulse.

Generalized bistatic radar systems have defined the bistatic angle, β, between the transmitter and receiver in a two-dimensional plane shown in Figure 2.1 (a). In literature pertaining to bistatic SAR, this two-dimensional plane has been adopted as the ground plane within a three-dimensional geometry depicted in Figure 2.1 (b). Previous expressions of bistatic range resolution have used range based time-domain methods to determine the spatial resolution in the two-dimensional plane [7], expressed using the ovals of Cassini. The authors of [4], used a time domain gradient method described in [8] to determine range resolution of a bistatic cell and [4] found similar results that are presented in this paper but not a closed form expression of the range resolution. These expressions for spatial resolution can be determined from the wavenumber domain known as k-space. The wavenumber domain or k-space, is the spatial frequency domain of the image, which means that it is simply the domain or space that is present after a SAR image is Fourier transformed. This method has been used to determine monostatic SAR resolution [9, 10], and the derived expressions for bistatic spatial resolution in this document become the same well-known monostatic spatial resolution equations when the terms are set to monostatic parameters.

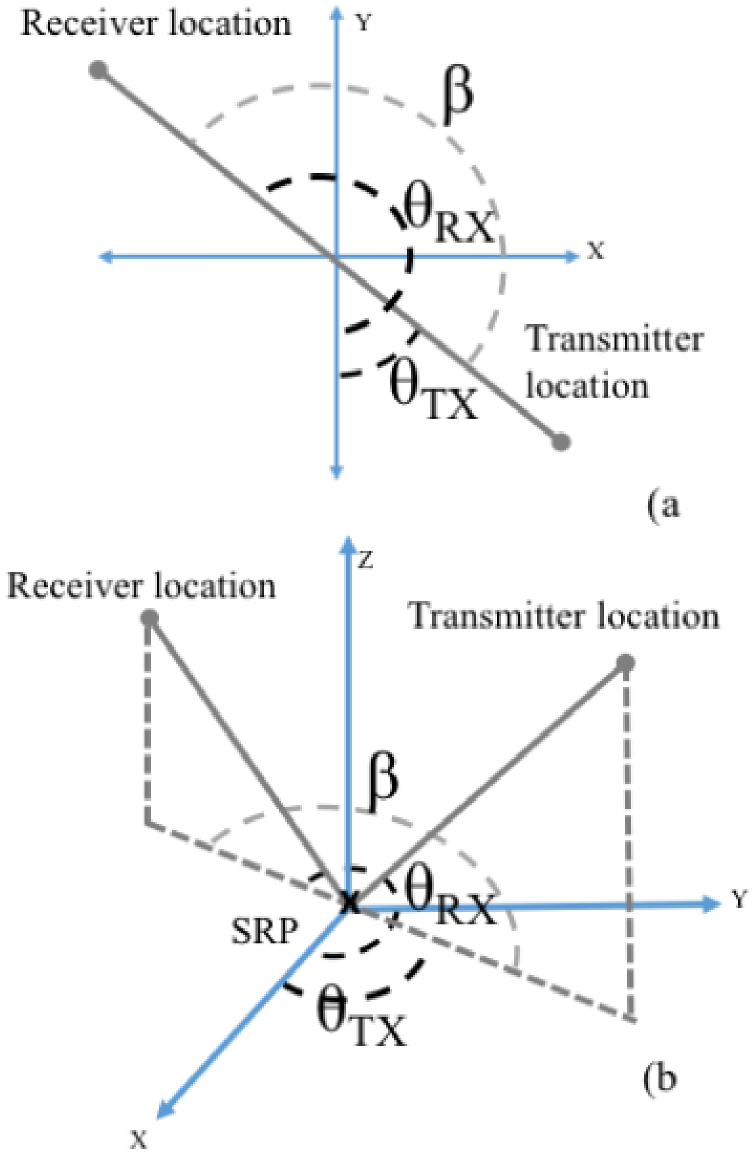

Figure 2.1: *Definition of the bistatic angle, β, and azimuth angles, Θ_{TX} and Θ_{RX}, with respect to the Scene Reference Point (SRP). (a) shows the x-y plane, (b) shows an isometric view.*

An interesting outcome from these derivations, in Chapter 3, is significant range and cross-range resolution in the ground plane of the forward-scatter bistatic region described in [7,11]. The forward-scatter region in this paper is defined as collection geometries with a bistatic angle $\beta = 180° \pm 20°$. Since this derivation is derived in a generalized three-dimensional geometry and then projected into a two-dimensional ground plane the different elevation angles of the transmitter and receiver enable range resolution in the ground plane when the two angles are unequal. As the difference between the two bistatic platforms' elevation angles increase, finer resolution is found in the ground plane. The elevation angles are shown in Figure 2.2.

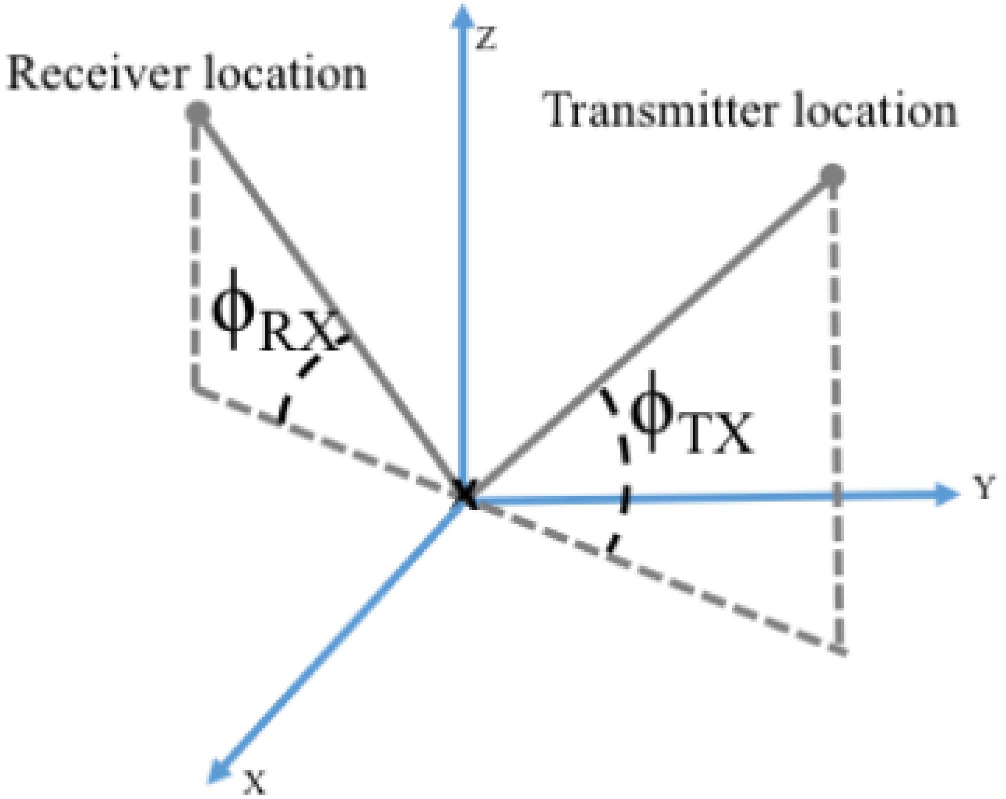

Figure 2.2: *Definition of the transmitter and receiver's elevation angles, ϕ_{TX} and ϕ_{RX}, respectively which are related to the Scene Reference Point (SRP).*

2.2 Generalized Bistatic SAR Received Signal Model

In general, the imaged scene is centered at the Scene Reference Point (SRP). The receiver will have information about the location of both platforms. In Figure 2.3, the transmitter and receiver are located at $\mathbf{p_t} = [x_t, y_t, z_t]^\intercal$ and $\mathbf{p_r} = [x_r, y_r, z_r]^\intercal$, respectively. Position information is used to derive timing information for each pulse transmitted and received forming the collection aperture. The receiver will also know the transmitted waveform and when it was transmitted. Targets within the scene exist and reflect the transmitted electromagnetic waves to the receiver as shown in Figure 2.3.

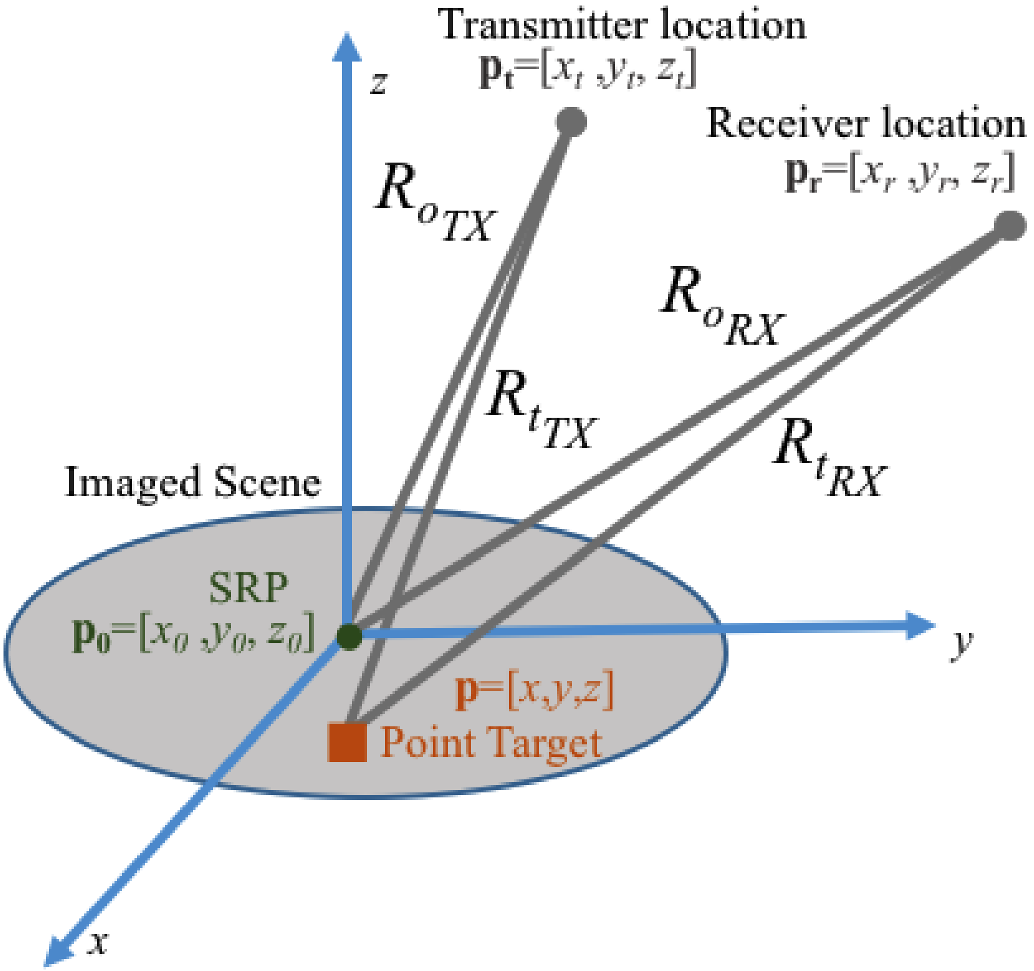

Figure 2.3: *Two bistatic radar systems imaging a scene containing the Scene Reference Point (SRP) and a target.*

The received signal model, $s(n,t)$, is given in Equation (2.1) where n is the discrete slow time index for each pulse, and t is the continuous fast time variable. The term n is indexed from $[1 : N_p]$ where N_p is the number of pulses transmitted during a collection aperture, and $t = 0$ occurs when the leading edge of the transmitted pulse leaves the transmitter.

$$s(n,t) = \int\int\int m(t - t_d(n,x,y,z)) f(x,y,z)\ dx\ dy\ dz \tag{2.1}$$

The transmitted waveform is $m(t)$, and $f(x,y,z)$ is the complex scattering amplitude from a target at the three dimensional location, $\mathbf{p} = [x,y,z]^\intercal$. The term $t_d(n,x,y,z)$ is the bistatic delay from the position of the scatterer, and the positions of the transmitter and receiver. Then the received data is motion compensated to the SRP at the location $\mathbf{p_0} = [x_0,y_0,z_0]^\intercal$ by determining the bistatic delay from the SRP and the transmitter and receiver, $t_0(n,x_0,y_0,z_0)$. The mathematical expressions for t_d and t_0 are given in Equations (2.2) and (2.3), while the ranges are depicted in Figure 2.3.

$$t_d(n) = \frac{R_{t_{TX}}(n) + R_{t_{RX}}(n)}{c} \tag{2.2}$$

$$t_0(n) = \frac{R_{o_{TX}}(n) + R_{o_{RX}}(n)}{c} \tag{2.3}$$

Where c is the speed of the electromagnetic wave (i.e. the speed of light). The received signal model is motion compensated to the SRP by subtracting t_0 and introducing a new time variable $t' = t - t_0(n,x_0,y_0,z_0)$. During each pulse repetition interval (PRI), the term $t' = 0$ occurs at the instance when the leading edge of the transmitted pulse reaches the SRP. Consequently, the received signal model becomes

$$s(n,t') = \int\int\int m(t' - (t_d(n,x,y,z) - t_0(n,x_0,y_0,z_0))) f(x,y,z)\ dx\ dy\ dz\ \ . \tag{2.4}$$

Taking the Fourier transform in the t' domain yields the following form of the received signal model.

$$S(n,w) = M(w)\int\int\int e^{-jw(t_d(n,x,y,z)-t_0(n,x_0,y_0,z_0))} f(x,y,z)\ dx\ dy\ dz \tag{2.5}$$

Where the term w is the angular frequency in units of radians per second. The two delay terms in the exponential can be simplified in terms of the range to the SRP as

$$t_d(n,x,y,z) - t_0(n,x_0,y_0,z_0) = \frac{R_{t_{TX}}(n) + R_{t_{RX}}(n)}{c} - \frac{R_{o_{TX}}(n) + R_{o_{RX}}(n)}{c} \tag{2.6}$$

and then rearranged as

$$t_d(n,x,y,z) - t_0(n,x_0,y_0,z_0) = \frac{R_{t_{TX}}(n) - R_{o_{TX}}(n)}{c} + \frac{R_{t_{RX}}(n) - R_{o_{RX}}(n)}{c} \quad . \tag{2.7}$$

The differential range for the transmitter is $\Delta R_{TX}(n) = R_{t_{TX}}(n) - R_{o_{TX}}(n)$. The equation, $\Delta R_{RX}(n) = R_{t_{RX}}(n) - R_{o_{RX}}(n)$ is the definition for the receiver's differential range, hence,

$$t_d(n,x,y,z) - t_0(n,x_0,y_0,z_0) = \frac{\Delta R_{TX}(n)}{c} + \frac{\Delta R_{RX}(n)}{c} \quad . \tag{2.8}$$

An approximation of the differential range to an arbitrary bistatic system is used in the following derivation of the signal model. The subscript a and A will refer to an arbitrary bistatic platform which is either the transmitter or receiver located at $\mathbf{p_a} = [x_a, y_a, z_a]^\intercal$. The TX or RX subscript for the transmitter and receiver subscripts respectively will be replaced with A and the r and t subscript will be replaced with a. The term ΔR_A is the difference between the range from an arbitrary bistatic radar system to the target and the range to the SRP from the radar, which leads to

$$\Delta R_A \begin{pmatrix} \mathbf{p} \\ \mathbf{p_0} \end{pmatrix} = R_{t_A} - R_{o_A} = |\mathbf{p_a} - \mathbf{p}| - |\mathbf{p_a} - \mathbf{p_0}| \quad , \tag{2.9}$$

where the dependence on n has been suppressed and $| \bullet |$ is the magnitude of $\bullet$.

In [12,13], a first order Taylor approximation is used to approximate ΔR_A for both the transmitter and receiver in terms of the Cartesian position of the target at $\mathbf{p} = [x, y, z]^\intercal$ and the SRP, $\mathbf{p_0} = [x_0, y_0, z_0]^\intercal$. The angular positions of the transmitter or receiver are located at $\mathbf{p_a} = [x_a, y_a, z_a]^\intercal$. In Equation (2.10), a first order multi-dimensional Taylor approximation is used to approximate the differential range when the target occupies the same location as the SRP leading to $\mathbf{p_0} = \mathbf{p} = [0, 0, 0]^\intercal$. In this equation $\overset{\rightrightarrows}{\nabla}$ is defined as the gradient in

$$\Delta R_A(\mathbf{0}_{6\times 1}) \approx \overset{\rightrightarrows}{\nabla} \left(\Delta R_A \begin{pmatrix} \mathbf{p} \\ \mathbf{p_0} \end{pmatrix} \right)^\intercal \Bigg|_{\begin{pmatrix} \mathbf{p} \\ \mathbf{p_0} \end{pmatrix} = \mathbf{0_{6\times 1}}} \cdot \begin{pmatrix} \mathbf{p} \\ \mathbf{p_0} \end{pmatrix} \quad . \tag{2.10}$$

The approximation now becomes

$$\Delta R_A \approx \frac{x_a(x - x_0) + y_a(y - y_0) + z_a(z - z_0)}{\sqrt{x_a^2 + y_a^2 + z_a^2}} \quad . \tag{2.11}$$

The SRP is located at $\mathbf{p_0} = [x_0, y_0, z_0]^\intercal = [0, 0, 0]^\intercal$. Then the following four geometric equations create the conversion in terms of the angle to the arbitrary bistatic radar system, $R_{o_A} = \sqrt{x_a^2 + y_a^2 + z_a^2}$, $x_a = R_{o_A} \cos\theta_A \cos\phi_A$, $y_a = R_{o_A} \sin\theta_A \cos\phi_A$ and $z_a = R_{o_A} \sin\phi_A$. Hence, the expression for ΔR_A is given by an approximation, which is also found in [12,13] as

$$\Delta R_A \approx x\cos(\theta_A)\cos(\phi_A) + y\sin(\theta_A)\cos(\phi_A) + z\sin(\phi_A) \quad . \tag{2.12}$$

Each angle can vary as the pulse number, n, changes. Figure 2.4 illustrates a transmitter, and a receiver as well as their relationship to the angles within Equation (2.12).

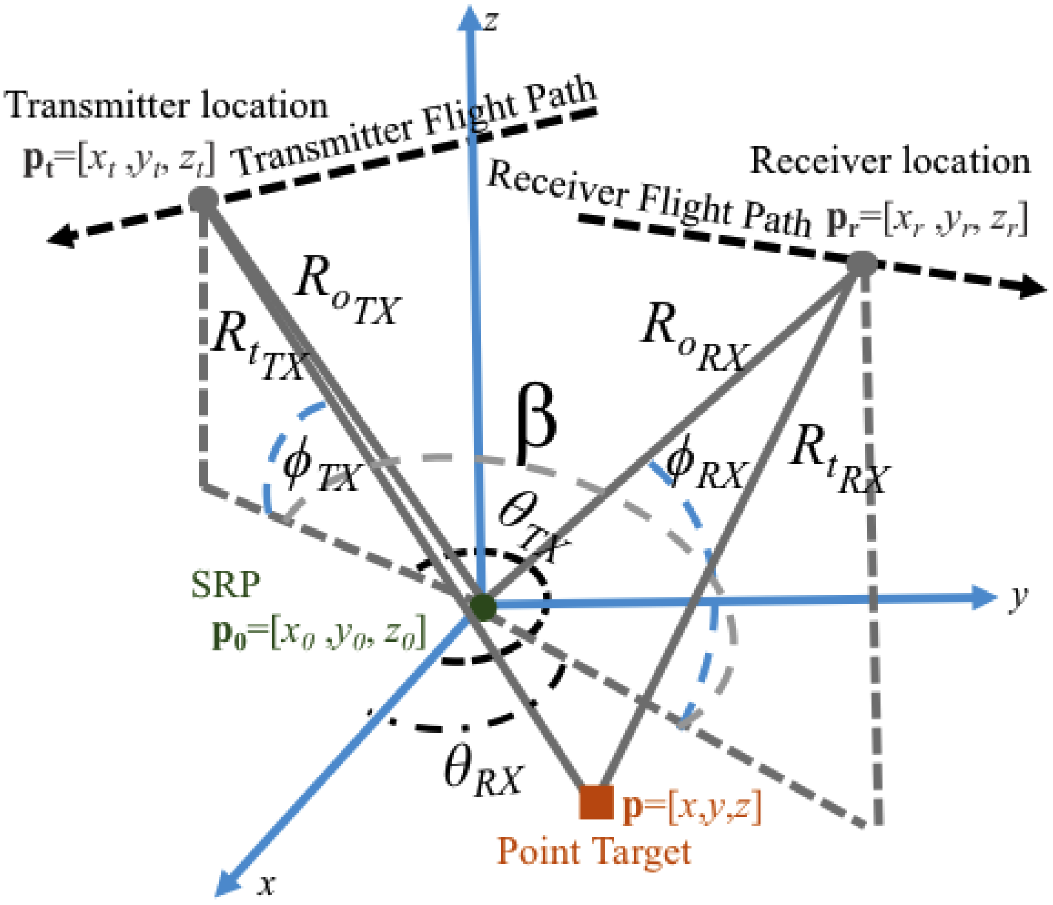

Figure 2.4: *Angular locations for a single PRI with respect to the Scene Reference Point (SRP). See Figures 2.1 and 2.2 for less complex depictions of the angles.*

At this point, returning to Equation (2.5) substitution of Equation (2.8),

$$S(n, w) = M(w) \int\int\int e^{-jw\left(\frac{\Delta R_{TX}(n)}{c} + \frac{\Delta R_{RX}(n)}{c}\right)} f(x, y, z)\, dx\, dy\, dz \tag{2.13}$$

is found, which is a bistatic signal model based on ΔR_{RX} and ΔR_{TX}. From this signal model, the bistatic phase of the signal can be isolated within

$$S(n,w) = M(w) \int\int\int e^{-j\Phi(n,w)} f(x,y,z)\ dx\ dy\ dz \quad . \tag{2.14}$$

Based on the ΔR approximation, the phase of the received signal now becomes

$$\Phi(n,w) = w\bigg(\frac{\Delta R_{TX}(n)}{c} + \frac{\Delta R_{RX}(n)}{c}\bigg) \quad . \tag{2.15}$$

Including the two-way bistatic phase change in the term k, where $k = \frac{2w}{c} = \frac{4\pi f}{c} = \frac{4\pi}{\lambda}$ [9] incorporates this two-way delay, the bistatic phase becomes

$$\Phi(n,k) = k\bigg(\frac{\Delta R_{TX}(n) + \Delta R_{RX}(n)}{2}\bigg) \quad . \tag{2.16}$$

Then, expanding the bistatic phase to include the ΔR approximation, the phase is represented in terms of the target location and transmitter and receiver angles to the SRP in

$$\begin{aligned}\Phi(n,k) = \frac{k}{2}\Big(&x(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))+ \\ &y(\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))+ \\ &z(\sin(\phi_{TX}) + \sin(\phi_{RX}))\Big) \quad .\end{aligned} \tag{2.17}$$

At this point in the signal model development it is important for the reader to be aware of the difference between $\Phi(n,k)$, the phase of the received signal model, and the elevation angles of the transmitter ($\phi_{TX}(n)$) and receiver ($\phi_{RX}(n)$) which also have a dependence on the slow time variable n which has been suppressed.

2.3 K-space defined within the Generalized Bistatic Receive Signal Model

In [14], the k-space is defined as a vector, $\mathbf{k} = [k_x, k_y, k_z]^T = w\alpha$, where $\alpha = [\frac{\partial t}{\partial x}, \frac{\partial t}{\partial y}, \frac{\partial t}{\partial z}]$ is the slowness vector pointing in the direction of the propagating wave. Now, the speed of light can be defined as $c = \frac{1}{|\alpha|}$ and the angular frequency becomes $w = \frac{\partial \Phi(n,k)}{\partial t}$. Using the chain rule, $\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial x}$, these definitions lead to the k-space being defined as,

$$\mathbf{k} = w\alpha = \left[\frac{\partial \Phi(n,k)}{\partial x}, \frac{\partial \Phi(n,k)}{\partial y}, \frac{\partial \Phi(n,k)}{\partial z}\right] \ . \tag{2.18}$$

Taking the derivative of the phase with respect to the Cartesian location of the target will allow the determination of the k-space or wavenumber respectively for the k_x, k_y, and k_z, k-space dimensions.

$$\frac{\partial \Phi(n,k)}{\partial x} = k_x = k\frac{\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX})}{2} \tag{2.19}$$

$$\frac{\partial \Phi(n,k)}{\partial y} = k_y = k\frac{\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX})}{2} \tag{2.20}$$

$$\frac{\partial \Phi(n,k)}{\partial z} = k_z = k\frac{\sin(\phi_{TX}) + \sin(\phi_{RX})}{2} \tag{2.21}$$

Equations (2.19) to (2.21) define the three dimensional k-space or wavenumber manifold of the bistatic received signal. The k-space method for bistatic SAR was introduced in [15,16]. The term k is the instantaneous wavenumber of the transmitted waveform. It is important to note that these k-space relationships described in Equations (2.19) to (2.21) in a Cartesian space are approximations due to the ΔR_A approximation defined in Equation (2.12) [12]. The use of this approximation aids in the development of closed form range and cross-range resolution equations which will be shown in Chapter 3. Similar forms of Equations (2.19) to (2.21) are used in the development of a bistatic polar format algorithm and bistatic flight paths found in [6,13,17]. Discussion about the bistatic polar format algorithm can also be found in Section 4.1.

2.4 Monostatic Receive Signal Model using Stretch Processing

Stretch processing requires a SAR system to transmit a Linear Frequency Modulated (LFM) waveform at a center frequency, f_c. Most high range resolution radar data is stretch processed to conserve Intermediate Frequency (IF) bandwidth unless the receive swath width is extremely large. Stretch processing requires demodulation with a chirp waveform using the transmitter's chirp rate, γ, to deramp the receive LFM waveforms reflected from the point scatterers within the imaged area [18].

Polar formatted deramped receive data, in the wavenumber domain, is the native format of most high resolution SAR phase histories. This data is recorded at the analog-to-digital (A/D) converter. Hence, the motivation for a stretched processed signal model in this paper. Other SAR signal models like the signal models found in [3, 6, 19] may be substituted for this signal model as well and some of these models are bistatic signal models. The signal model presented in this section is for a monstatic system and further details can be found in [9]. The algorithms described in Chapter 4 operate on the data while it is in spatial frequency, wavenumber, or k-space domain.

Deramp or dechirp on receive demodulation is a step within stretch processing and will be employed in this derivation. The ultimate goal of stretch processing is pulse compressed radar data in the range domain while requiring a smaller IF bandwidth for the radar's receiver chain and decreasing the A/D converter's sampling rate requirement. In Section 4.6, the deramped SAR received signals are preprocessed to reduce computations of the back projection algorithm (BPA).

The transmit signal's bandwidth is defined by a product of the chirp rate, γ, and the pulse width, T_p. The LFM waveform spans the frequencies from $f_c - \frac{T_p\gamma}{2}$ to $f_c + \frac{T_p\gamma}{2}$. In [9], the LFM waveform transmitted from a radar system is represented as

$$m_{tx}(t) = rect\left(\frac{\hat{t}}{T_p}\right)e^{j(\phi_t + 2\pi f_c t + \pi\gamma\hat{t}^2)} \quad . \tag{2.22}$$

Therefore, $\hat{t}$ is time referenced at the beginning of the Pulse Repetition Interval (PRI), T, such that $\hat{t} = t - nT$. At $t = 0$, the leading edge of the first transmit pulse of the pulse train leaves the antenna. The term n is the discrete index for each PRI and the first PRI is indexed to $n = 0$. There are N_p PRIs in the Coherent Pulse Interval (CPI) that form the synthetic aperture. The term ϕ_{tx} is the initial phase of the transmitter's local oscillator. The rectangular function, $rect()$, is

$$rect(u) = \begin{cases} 1 & \text{if } \mid u \mid \leq \frac{1}{2} \\ 0 & \text{if } \mid u \mid > \frac{1}{2} \end{cases} \quad . \tag{2.23}$$

After a pulse is transmitted, the waveform is reflected from a point target within the scene which generates,

$$s_r(n,t) = a_t rect(\frac{\hat{t} - t_d}{T_p}) e^{j(\phi_t + 2\pi f_c (t - td) + \pi\gamma(\hat{t} - td)^2)} \quad , \tag{2.24}$$

which is the signal prior to the first mixer in the receiver chain [9]. Where t_d, which varies as n increases, is the two-way time delay from the transmitter to a scatterer within the scene and then back to the receiver. For a monostatic radar, t_d is defined by $t_d(n) = \frac{2R_t(n)}{c}$ where $R_t(n)$ is the range to the point target at the PRI index n. The term, c, is the speed of the electromagnetic wave. The term, a_t, is the square root of the Radar Cross Section (RCS) of the target. At the receiver, the receive signal is mixed with the Local Oscillator (LO).

The reference chirp at the receiver's first LO is designed to have the instantaneous frequency of f_c when the scene center's return arrives at the first LO. Another name for the scene center is the Scene Reference Point (SRP) illustrated in Figure 2.5. Given an appropriately sized receive swath width, this method can decrease the needed IF bandwidth. The computational complexity of pulse compression is reduced to a Fast Fourier Transform (FFT) instead of convolution with a matched filter. This method will provide an analytic method to understand the receive signal's k-space.

The reference signal used at the first LO is

$$m_{ref}(n,t) = e^{-j(\phi_r + 2\pi f_c(\hat{t} - t_o) + \pi\gamma(\hat{t} - to)^2)} \quad . \tag{2.25}$$

For monostatic radar systems, the term t_o is the two-way delay to the SRP within the SAR image and a function of n. The term n increases as the radar system travels along the synthetic aperture due to range changing as n increases through the CPI. The term $t_o(n) = \frac{2R_o(n)}{c}$ where $R_o(n)$ is the range to the SRP. Figure 2.5 illustrates the relationships between the radar and the

SRP. The term ϕ_r is the initial phase of the receiver's local oscillator.

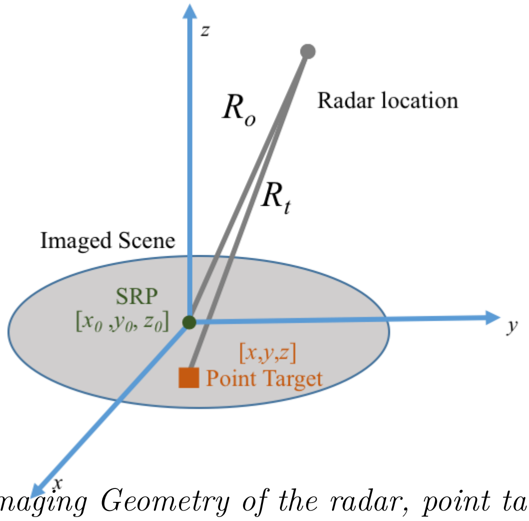

Figure 2.5: *Imaging Geometry of the radar, point target and scene.*

Mixing the receive signal model, Equation (2.24), with the reference signal, Equation (2.25), at the first LO in the receiver chain produces equation (2.26). This is the IF signal model that is used to analytically understand the k-space. The signal at the output of the mixer is

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t}-t_d}{T_p})e^{j(\Delta\phi - 2\pi(f_c+\gamma(\hat{t}-t_o))(t_d-t_o)+j\pi\gamma(t_d-to)^2)} \quad . \tag{2.26}$$

For the radar system, $t_d(n) = \frac{2R_t(n)}{c}$ where $R_t(n)$ is the range to the point target indicated in Figure 2.5. The $\Delta\phi$ term in this equation is the difference between the initial phases of the transmitter and the receiver's LO, such that $\Delta\phi = \phi_t - \phi_r$. For the monostatic case, this term becomes zero since $\phi_t = \phi_r$. Substituting for the monostatic delays

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t} - \frac{2R_t(n)}{c}}{T_p}) e^{-j(2\pi(f_c+\gamma(\hat{t}-\frac{2R_o(n)}{c}))(\frac{2R_t(n)}{c}-\frac{2R_o(n)}{c})-2\pi\gamma(\frac{2R_t(n)}{c}-\frac{2R_o(n)}{c})^2)} \tag{2.27}$$

is now the monostatic IF signal model.

This model can be simplified further by rearranging some terms and substituting $\Delta R(n) = R_t(n) - R_o(n)$, resulting in

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t} - \frac{2R_t(n)}{c}}{T_p}) e^{-j\frac{4\pi}{c}(f_c+\gamma(\hat{t}-\frac{2R_o(n)}{c}))\Delta R(n)+j\frac{4\pi\gamma}{c^2}\Delta R(n)^2} \quad . \tag{2.28}$$

Within this monostatic signal model, the RVP (Residual Video Phase) containing the $\Delta R(n)^2$ term is omitted [9]. For small scene sizes with respect to center frequency, this term can cause a negligible amount of phase distortion within the final image. If this distortion becomes non-negligible, it can be mitigated by corrections within the image formation algorithms due its dependence on range to the scene center [9, 13]. With further analysis of the monostatic receive signal model becomes

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t} - \frac{2R_t(n)}{c}}{T_p}) e^{-j\frac{4\pi}{c}(f_c+\gamma(\hat{t}-\frac{2R_o(n)}{c}))\Delta R(n)} \quad . \tag{2.29}$$

At this point, two important aspects about the model should be noted. First, the data within the model are motion compensated to the SRP which is at the range $R_o(n)$. Second, since the data has been deramped on receive, the fast-time diminsion of the data, $\hat{t}$, is now in the spatial-frequency dimension instead of the spatial-time dimension and can be pulse compressed using an FFT. The monostatic IF receive signal model can now be split into two separate parts. The monostatic IF signal,

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t} - \frac{2R_t(n)}{c}}{T_p})e^{-j\Phi(n,\hat{t})} \quad , \tag{2.30}$$

and its phase,

$$\Phi(n,\hat{t}) = \frac{4\pi}{c}(f_c + \gamma(\hat{t} - \frac{2R_o(n)}{c}))\Delta R(n) \quad . \tag{2.31}$$

2.5 K-space defined within the Monostatic Receive Signal Model

The wavenumber, k, is the spatial frequency dimension of the SAR image and is defined in [9] as

$$k = \frac{4\pi f}{c} = \frac{4\pi}{\lambda}\frac{[2\ cycles]}{[meter]} \quad . \tag{2.32}$$

The term f in Equation (2.32) represents the instantaneous frequency of the waveform, not a specific center frequency. The wavenumber's units in the numerator is two cycles to account for the two way propagation of the signal. Within Equation (2.31), the waveform's instantaneous frequency is substituted creating

$$\Phi(n,\hat{t}) = \frac{4\pi}{\lambda_c}(1 + \frac{\gamma}{f_c}(\hat{t} - \frac{2R_o(n)}{c}))\Delta R(n) = k_c(1 + \frac{\gamma}{f_c}(\hat{t} - \frac{2R_o(n)}{c}))\Delta R(n) \quad . \tag{2.33}$$

The terms k_c and λ_c, are the cener wavenumber and center wavelength of the transmitted waveform. The other term to examine is the $\Delta R(n)$ term which can be approximated using a first order Taylor Series expansion given that the target’s distance from the SRP is significantly smaller than the range of the radar system to the SRP [9].

$$\Delta R(n) = R_t(n) - R_o(n) \approx \cos(\phi)[x\cos(\theta) + y\sin(\theta) + z\cot(\phi)] \qquad (2.34)$$

The angles within Equation (2.34) have their dependence on n suppressed. Each angle varies as the pulse number, n, increases, which is illustrated in Figure 2.6.

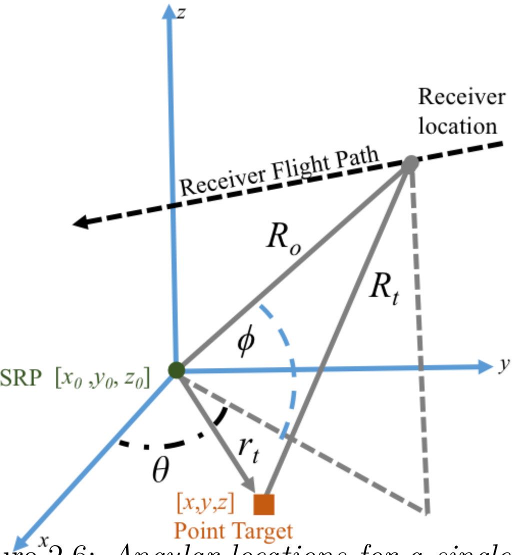

Figure 2.6: *Angular locations for a single PRI*

Further simplification leads to

$$\Delta R(n) \approx x\cos(\theta)\cos(\phi) + y\sin(\theta)\cos(\phi) + z\sin(\phi) \quad . \qquad (2.35)$$

Now, this approximation can then be placed into the phase equation to show the individual components of the k-space as

$$\Phi(n, \hat{t}) = k_c(1 + \frac{\gamma}{f_c}(\hat{t} - \frac{2R_o(n)}{c}))[x\cos(\theta)\cos(\phi) + y\sin(\theta)\cos(\phi) + z\sin(\phi)] \quad . \qquad (2.36)$$

Then the inner product of the $\boldsymbol{r_t}(n)$ vector with the $\boldsymbol{k}(n,\hat{t})$ vector creates the phase of the signal model. Where $\boldsymbol{r_t} = [x,\ y,\ z]$.

$$\Phi(n,\hat{t}) = \boldsymbol{k}(n,\hat{t}) \bullet \boldsymbol{r_t}(n) \tag{2.37}$$

$$k_r(n,\hat{t}) = k_c\Big(1 + \frac{\gamma}{f_c}\Big(\hat{t} - \frac{2R_o(n)}{c}\Big)\Big) \tag{2.38}$$

The k-space in a monostatic range return is given as

$$\boldsymbol{k}(n,\hat{t}) = k_r(n,\hat{t})[\cos(\theta(n))\cos(\phi(n)),\ \sin(\theta(n))\cos(\phi(n)),\ \sin(\phi(n))]\ . \tag{2.39}$$

Then, Equations (2.40) to (2.43) define the three dimensional k-space of the monostatic receive signal depicted in Figure 2.7. Where $\hat{t}$ ranges between $[\frac{2R_t(n)}{c} - \frac{T_p}{2},\ \frac{2R_t(n)}{c} + \frac{T_p}{2}]$ due to the $rect()$ function in Equation (2.30) for the signal model.

$$\boldsymbol{k}(n,\hat{t}) = [k_x(n,\hat{t}),\ k_y(n,\hat{t}),\ k_z(n,\hat{t})] \tag{2.40}$$

$$k_x(n,\hat{t}) = k_r(n,\hat{t})\cos(\theta(n))\cos(\phi(n)) \tag{2.41}$$

$$k_y(n,\hat{t}) = k_r(n,\hat{t})\sin(\theta(n))\cos(\phi(n)) \tag{2.42}$$

$$k_z(n,\hat{t}) = k_r(n,\hat{t})\sin(\phi(n)) \tag{2.43}$$

Figure 2.7 (a) illustrates a single radar return collected by the radar system in Figure 2.6 traversing a flight path.

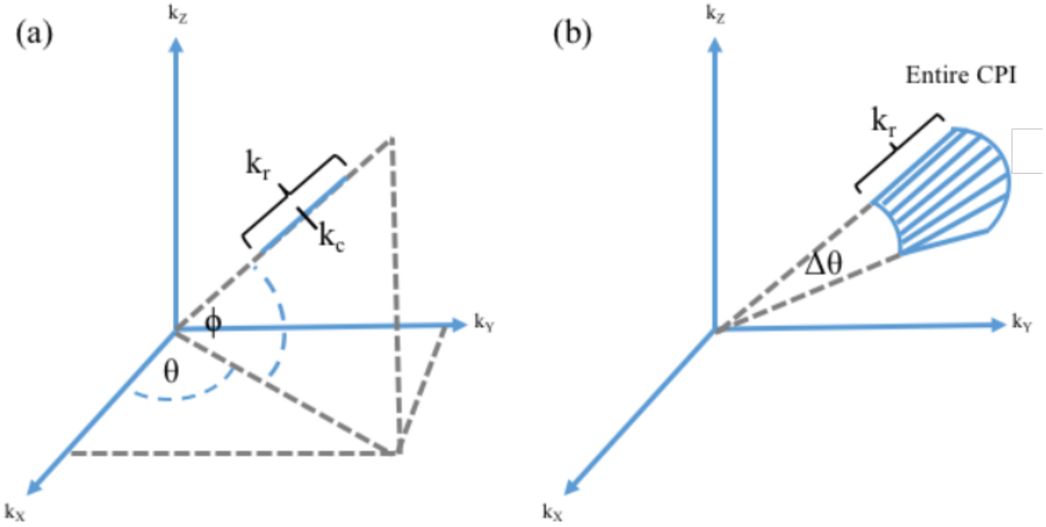

Figure 2.7: *K-space representations of the signal. a) a single PRI within the k-space b) All the PRIs over a CPI transformed into a k-space.*

This return will have the same angular locations in k-space as it did in the SRP referenced Cartesian space where it was collected. The angular equivalence between the spatial and k-space domains is understood through the Fourier Slice Theorem which is discussed in Section 4.3. All the returns collected within a CPI are depicted in the k-space in figure 2.7 (b). Generally, these returns create the fan-shaped manifold as depicted. A group of neighboring radar returns are fan shaped but fairly rectangular which is important in Section 4.5 as well.

2.6 The Bistatic Receive Signal Model using Stretch Processing

This derivation of the bistatic signal model will begin with the signal model derived in Equation (2.26) and reproduced in Equation (2.44).

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t}-t_d}{T_p})e^{j(\Delta\phi - 2\pi(f_c+\gamma(\hat{t}-t_o))(t_d-t_o)+\pi\gamma(t_d-to)^2)} \tag{2.44}$$

The $\Delta\phi$ term in this equation is the difference between the initial phases of the transmitter and the receiver's LO, such that $\Delta\phi = \phi_t - \phi_r$. The t and r subscripts represent the transmitter and receiver. The term $\hat{t}$ is the time referenced to the start of the Pulse Repetition Interval (PRI), T, such that $\hat{t} = t - nT$. At $t = 0$, the leading edge of the first transmit pulse of the pulse train leaves the transmitter's antenna when $n = 0$, the first PRI. The term n is the discrete index for each PRI. There are N_p PRIs in the Coherent Pulse Interval (CPI) that forms the synthetic aperture. The $rect()$ function is defined by Equation (2.23).

Within the signal model in Equation (2.44), there is a term that can be omitted called the Residual Video Phase (RVP) containing the $(t_d - t_o)^2$ term. This term was discussed briefly in Section 2.4 and can be omitted due to it negligible effects or eventual compensation within an image formation algorithm. This term will be discussed in more detail in Section 2.8.1. With this term removed, the Bistatic Receive Signal model becomes

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t}-t_d}{T_p})e^{j(\Delta\phi - 2\pi(f_c+\gamma(\hat{t}-t_o))(t_d-t_o))} \quad . \tag{2.45}$$

For a bistatic radar system, the term t_d is defined by $t_d(n) = \frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c}$ where $R_{t_{TX}}(n)$ and $R_{t_{RX}}(n)$ are the ranges from the transmitter and receiver to the point target. The term to is defined by $to(n) = \frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}$ where $R_{o_{TX}}(n)$ and $R_{o_{RX}}(n)$ are the ranges from the transmitter and receiver to the SRP, respectively. Figure 2.3 depicts the relationships between bistatic radar systems, a point target, and the SRP. Substituting these ranges for the delays, the bistatic receive signal model becomes

$$s_{B_{IF}}(n,t) = a_t rect(\frac{\hat{t} - \frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c}}{T_p}) \cdot ...$$
$$e^{j(\Delta\phi - 2\pi(f_c+\gamma(\hat{t}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}))(\frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}))} \quad . \qquad (2.46)$$

The terms $R_{o_{TX}}(n)$, $R_{o_{RX}}(n)$, $R_{t_{TX}}(n)$, and $R_{t_{RX}}(n)$ are redefined into differential range terms, $\Delta R_{TX} = R_{t_{TX}}(n) - R_{o_{TX}}(n)$ and $\Delta R_{RX} = R_{t_{RX}}(n) - R_{o_{RX}}(n)$ creating

$$s_{B_{IF}}(n,t) = a_t rect(\frac{\hat{t} - \frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c}}{T_p}) e^{j\Delta\phi} \cdot ...$$
$$e^{-j(\frac{2\pi}{c}(f_c+\gamma(\hat{t}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}))(\Delta R_{TX}(n)+\Delta R_{RX}(n))} \quad . \qquad (2.47)$$

Equation (2.47) is the Bistatic Receive Signal model whose form is similar to Equation (2.29) for a monstatic radar system. This model can also be broken into two separate equations: the Bistatic IF signal and it's phase, Equations (2.48) and (2.49), respectively.

$$s_{B_{IF}}(n,t) = a_t rect(\frac{\hat{t} - \frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c}}{T_p}) e^{j\Delta\phi} e^{-j\Phi(n,\hat{t})} \qquad (2.48)$$

$$\Phi(n,\hat{t}) = \frac{2\pi}{c}(f_c + \gamma(\hat{t} - \frac{R_{o_{TX}}(n) + R_{o_{RX}}(n)}{c}))(\Delta R_{TX}(n) + \Delta R_{RX}(n)) \qquad (2.49)$$

2.7 K-space Defined within the Bistatic Receive Signal Model

Within Equation (2.49) the wave number can be substituted to give Equation (2.51).

$$\Phi(n,\hat{t}) = \frac{4\pi}{\lambda_c}\Big(1+\frac{\gamma}{f_c}\Big(\hat{t}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}\Big)\Big)\frac{\Delta R_{TX}(n)+\Delta R_{RX}(n)}{2} \tag{2.50}$$

$$\Phi(n,\hat{t}) = k_c\Big(1+\frac{\gamma}{f_c}\Big(\hat{t}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}\Big)\Big)\frac{\Delta R_{TX}(n)+\Delta R_{RX}(n)}{2} \tag{2.51}$$

The terms k_c and λ_c represent the center wavenumber and center wavelength of the transmitted waveform. The other terms to examine are $\Delta R_{TX}(n)$ and $\Delta R_{RX}(n)$. These terms can be approximated using a first order Taylor series expansion given the target's distance from the SRP is significantly smaller than the range of the radar system to the SRP [13]. This approximation will be examined in further detail in Section 2.8.2.

The ΔR terms are approximated using Equation (2.52). The X subscript is used as a place holder for the reference to the transmitter, TX, or the receiver, RX.

$$\Delta R(n) \approx x\cos(\theta_X)\cos(\phi_X)+y\sin(\theta_X)\cos(\phi_X)+z\sin(\phi_X) \tag{2.52}$$

The angles within Equation (2.52) have their dependence on n suppressed. Each angle varies as the pulse number, n, changes. Figure 2.4 illustrates a transmitter and a receiver and their relationship to these angles.

Substituting the approximation in Equation (2.52) gives equation (2.53) for the bistatic phase.

$$\begin{aligned}\Phi(n,\hat{t}) = k_c\Big(1+\frac{\gamma}{f_c}\Big(\hat{t}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}\Big)\Big)\cdot ...\\ \Big(\frac{x(\cos(\theta_{TX})\cos(\phi_{TX})+cos(\theta_{RX})\cos(\phi_{RX}))}{2}+...\\ \frac{y(\sin(\theta_{TX})\cos(\phi_{TX})+\sin(\theta_{RX})\cos(\phi_{RX}))}{2}+...\\ \frac{z(\sin(\phi_{TX})+\sin(\phi_{RX}))}{2}\Big)\end{aligned} \tag{2.53}$$

The k-space in a bistatic range return is given in Equation (2.55), where $\boldsymbol{r} = [x,\ y,\ z]$.

$$\Phi(n,\hat{t}) = \boldsymbol{k}(n,\hat{t}) \bullet \boldsymbol{r}(n) \tag{2.54}$$

$$k_r(n,\hat{t}) = k_c\Big(1+\frac{\gamma}{f_c}\Big(\hat{t}-\frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}\Big)\Big) \tag{2.55}$$

Then the inner product of the $\boldsymbol{r}(n)$ vector with the $\boldsymbol{k}(n,\hat{t})$ vector creates the phase of the signal model.

Upon inspection, the k-space of Equation (2.53) and (2.55) are the bistatic k-space Equations (2.57) to (2.60).

$$\boldsymbol{k}(n,\hat{t}) = k_r(n,\hat{t})[\cos(\theta(n))\sin(\phi(n)),\ \sin(\theta(n))\sin(\phi(n)),\ \cos(\phi(n))] \tag{2.56}$$

$$\boldsymbol{k}(n,\hat{t}) = [k_x(n,\hat{t}),\ k_y(n,\hat{t}),\ k_z(n,\hat{t})] \tag{2.57}$$

$$k_x(n,\hat{t}) = k_r(n,\hat{t})\frac{(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))}{2} \tag{2.58}$$

$$k_y(n,\hat{t}) = k_r(n,\hat{t})\frac{(\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))}{2} \tag{2.59}$$

$$k_z(n,\hat{t}) = k_r(n,\hat{t})\frac{(\sin(\phi_{TX}) + \sin(\phi_{RX}))}{2} \tag{2.60}$$

These equations define the three dimensional k-space of the bistatic receive signal illustrated in Figure 2.8. The term $\hat{t}$ ranges between $[\frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c} - \frac{T_p}{2}, \frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c} + \frac{T_p}{2}]$ due to the $rect()$ function in Equation (2.48) for the signal model.

From this derivation, an interesting observation is made: The angular positions of each k-space radial are an average between the angular contributions from the transmitter and receiver to form the bistatic k-space manifold illustrated in Figure 2.8.

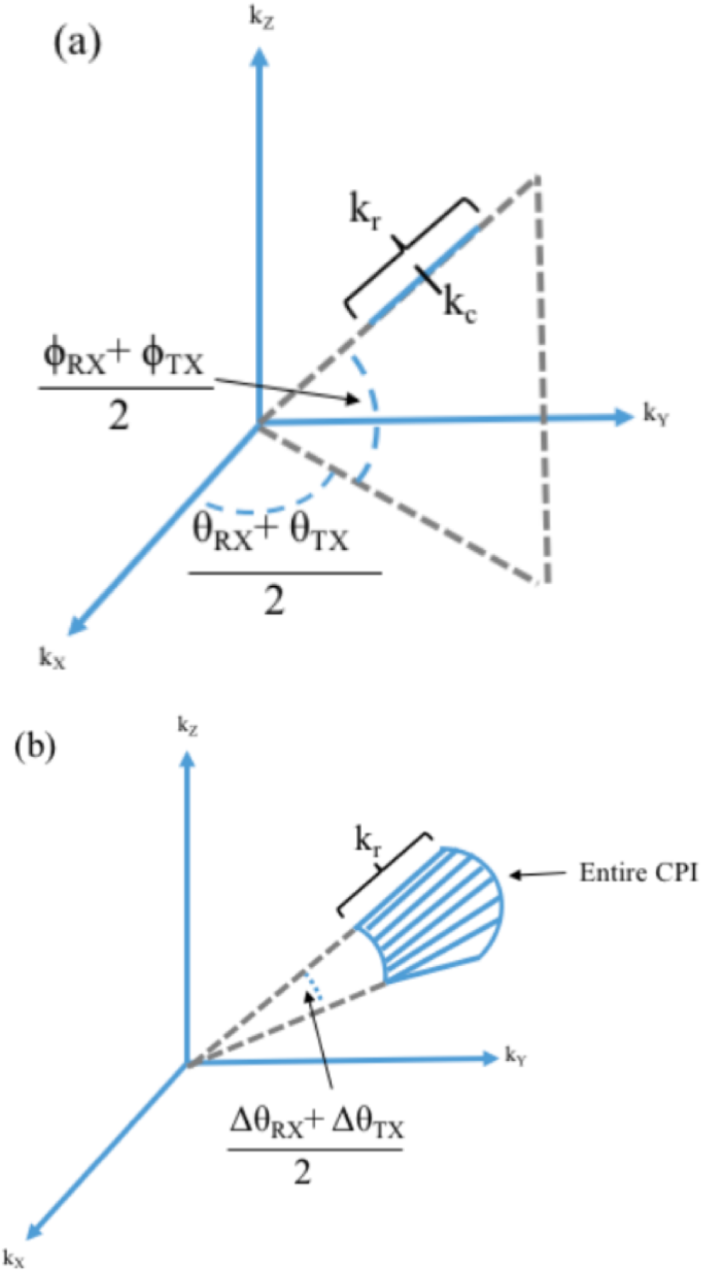

Figure 2.8: *Bistatic k-space representations of the signal. a) a single Bistatic PRI within the k-space b) All the PRIs over a Bistatic CPI transformed into a k-space.*

Figure 2.8 (a) depicts the single bistatic return from a bistatic system illustrated in Figure 2.4 in the k-space domain. This return has the same angular locations in k-space that it did in the SRP referenced Cartesian space where it was collected. The angular equivalence between the spatial and k-space domains is due to the Fourier slice theorem which will be discussed in detail in Section 4.3. All the returns that were collected within a bistatic CPI are depicted in k-space in Figure 2.8 (b). These returns could create the fan-shaped manifold as depicted or a more distorted 3-D space depending on the flight paths of the transmitter and receiver.

2.8 Bistatic's K-space Model Approximations and Limitations

The derivation of the bistatic k-space made four approximations. First, it neglected the Residual Video Phase (RVP) of the deramped signal model. Second, a first order Taylor series expansion to approximate the ΔR terms. Third, a stop and hop approximation was used for the position of the transmitter and receivers. Finally, the model assumed perfect measurement of the transmitter and receiver's position. This section will discuss these approximations and their limitations.

2.8.1 Residual Video Phase approximation

The RVP term in Equation (2.44) is defined as Equation (2.61). This term is an artifact of dechirp on receive processing and can either be neglected for small scene sizes or compensated to mitigate its effects from the image. In [9], this term can be removed without a high computational burden. Also, this term does not appear when matched filtering a receive signal in the range domain because the receive signal is not deramped on receive but coherently mixed with a tone.

$$s_{RVP_{IF}}(n,t) = a_t rect(\frac{\hat{t}-t_d}{T_p})e^{j2\pi\gamma(t_d-to)^2} \tag{2.61}$$

Section 3.10 of reference [9] presents analysis for the monostatic RVP's effects. For the analysis of the receive signal model, the RVP term is removed assuming it is either negligible or a correction will be applied in the pre-processing steps of the image formation algorithms. In [20], there is a description of an algorithm to remove the RVP term from a bistatic signal as well. Table 3.12 of [9] provides limits on the scene size radius versus the resolution of the SAR system at a certain wavelength. For example, a SAR system operating at X-Band with 0.3 meter resolution should limit the scene radius to 1.053 km to avoid the effect of RVP, but the same SAR system creating an image at 1 meter resolution will not see any RVP distortion effects for a 1.2 km scene radius. In [21], the authors cite methods in [9] and develop a method to decrease the effects of RVP using a frequency scaling operation in the appendix of [21].

In [22], the author derives the scene size limits for high chirp rate SAR systems. In addition, the authors allows a maximum amount of quadratic phase error to be $\substack{+\\-}\frac{\pi}{2}$. From this derivation the maximum scene size radius without compensation for monostatic SAR is

$$r_{max} \leq 4\rho_a\sqrt{\frac{R_o}{\lambda_c}} \quad . \tag{2.62}$$

Section 10.3.1 of [13] derives a bistatic pre-processing method to mitigate the effects of RVP. When a SAR system uses stretch processing or dechirp on receive, there are well-known methods to mitigate RVP's effects while still benefiting from the decreased IF bandwidth that stretch processing provides [13].

2.8.2 The ΔR Taylor series expansion approximation

The ΔR Taylor series expansion approximation is a far-field approximation which means R_o, the distance from the radar system to the SRP, must be much larger than the distance from the point scatterer to the SRP. This approximation introduces an effect in high resolution SAR imagery called range curvature. In Section 6.2, the mitigation of these effects were discussed for the images in Section 6.3. Reference [13] in section 10.4.3 discusses in detail the effects of the first order Taylor series approximation for ΔR for bistatic radars. There is a closed form inequality that bounds the scene size radius using a second order Taylor series approximation for ΔR. This bound was developed for monostatic radar [9]. Equation (2.63), which limits the phase error from the Taylor series approximation to $\substack{+\\-}\frac{\pi}{2}$ as

$$r_{M_{max}} \leq 2\rho_a\sqrt{\frac{R_o}{\lambda_c}} \quad . \tag{2.63}$$

In [13], the scene size radius limit for bistatic SAR systems is also developed using a second order Taylor series approximation in Equation (2.64).

$$r_{B_{max}} \leq \sqrt{2}\lambda\Big(\frac{L_{TX}^2}{R_{o_{TX}}^3} + \frac{L_{RX}^2}{R_{o_{RX}}^3}\Big)^{-\frac{1}{2}} \tag{2.64}$$

The terms L_{TX} and L_{RX} are the lengths of the synthetic apertures created by their flight paths. The azimuth or cross-range resolution for a SAR system is dependent on the flight path lengths. Equation (2.65) is the resolution for a bistatic SAR system based on the aperture length.

$$\rho_{a_{BI}} = \frac{\lambda}{\cos(\beta/2)}\Big[\frac{L_{TX}}{R_{o_{TX}}}\sin(\psi_{TX}) + \frac{L_{RX}}{R_{o_{RX}}}\sin(\psi_{RX})\Big] \tag{2.65}$$

The terms ψ_{TX} and ψ_{RX} are the squint angles of the transmit and receive synthetic apertures, respectively. In [13], the relationship between (2.63) and (2.64) is demonstrated when Equation (2.64) is set to broadside monostatic parameters, $\beta = 0$, $\psi_{TX} = \psi_{RX} = \frac{\pi}{2}$, $L_{TX} = L_{RX}$, and $R_{o_T X} = R_{o_R X}$. Then, the monostatic scene size limit in Equation (2.63) is found.

Equations (2.63) and (2.64) are important because they place a bound on the scene size for image formation algorithms like the polar format algorithm and the back projection algorithm (BPA) when stretch processed signals are used. When a target is imaged at the scene size limit for a bistatic SAR image, refenece [6] finds a 10% azimuth impulse response broadening in simulation.

In [23], the authors discuss methods to mitigate the second order effects from the Taylor series expansion in the image. In doing so, a larger scene radius can be imaged without distortion. Another effect from these corrections is that the pixels are placed in the correct locations. With the application of these corrections, the authors of [23] discussed the scene size radius is now limited to Equation (2.66) for monstatic systems. [23]

$$r_{M_{max}} \leq R_o \sqrt[3]{\frac{4\rho_a^2}{\lambda_c R_{gr_o}}} \tag{2.66}$$

The term R_{gr_o} is range to the target in the ground plane, known as the ground range to the target. The authors in [23] gave instances where this increased the scene size radius by a factor of 2 to 4 times the size without the corrections.

2.8.3 Stop and hop approximation

This signal model makes the assumption that at the time the waveform is transmitted and received, the radar system and a scatterer that it interacts with will not move [24]. Most SAR systems are in motion making this assumption unrealistic but in spite of this fact it is useful. The aircraft and the target may have motion. In a monostatic scenario when looking at the scene from a direction perpendicular to the flight path, there is little to no radial velocity so there is not a significant Doppler shift of the target's radar return. Once the radar images a scene at a squinted line of sight to the SRP there is a radial velocity from the aircraft and the signal is Doppler shifted as shown in Figure 2.9.

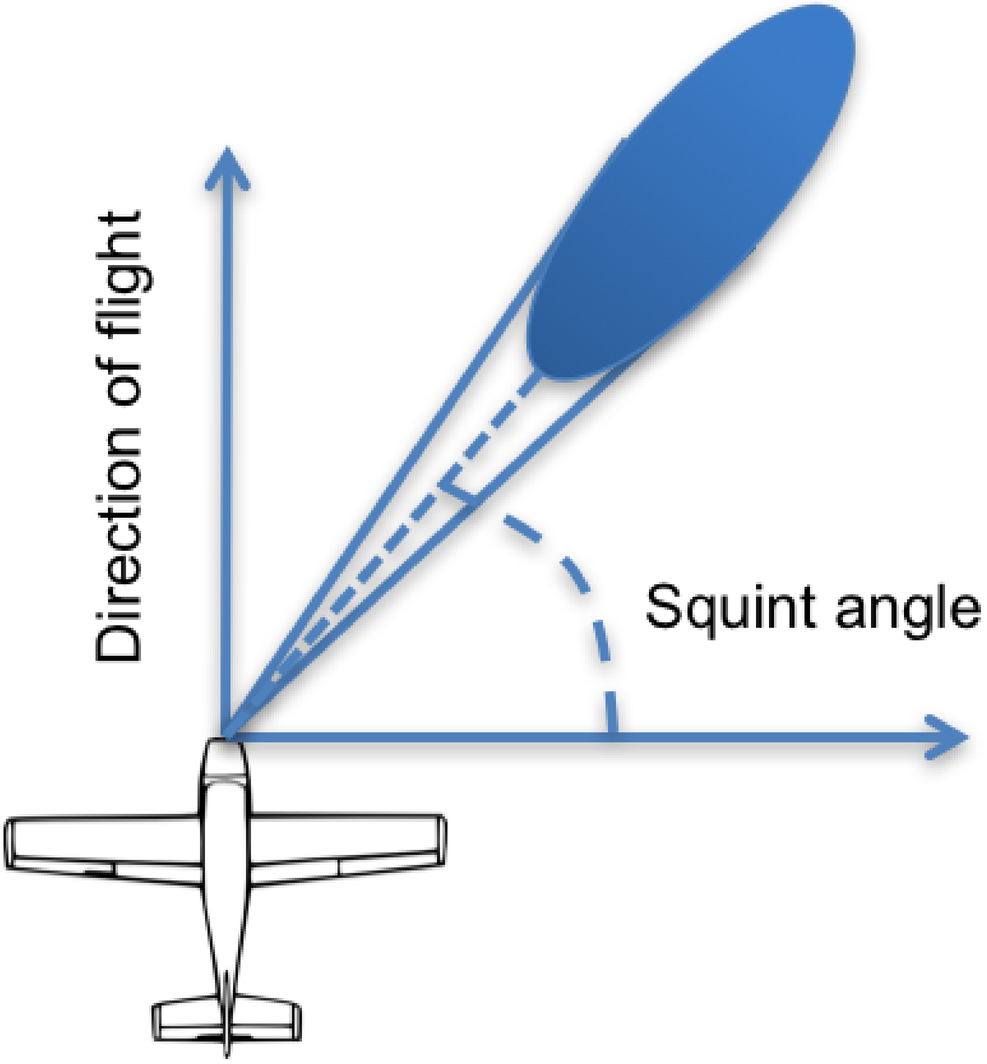

Figure 2.9: *Squint angle defined by an aircraft's flight path.*

The radar system's motion is measured by the SAR system and can be compensated for a more complex model signal if neccessary. In [24], the author goes into detail about the stop and hop assumption and is a detailed reference to understand this assumption. In [9], the author provides a more in depth analysis of the signal model and corrections to mitigate the affects of the platform's motion. In bistatic scenarios, this assumption should be reevaluated. A bistatic configuration, where one platform flies directly at the scene, imparts a significant Doppler shift while not generating cross-range resolution. An example of this scenario is a passive missile flying towards a target as one bistatic system, and another part of the bistatic system carrying an active transmitter with a perpendicular flight path to the missile's flight path.

The target's motion is another aspect of this model that should be considered. Examples of this motion are vehicles, humans, animals in motion, or trees and grass blowing in the wind. This motion is difficult to compensate due to its unknown orientation with respect to the transmitter and receiver.

2.8.4 Perfect Position Measurement

This model also assumed perfect knowledge of the aircraft's position. When SAR systems are implemented, the motion measurement subsystem cannot measure all of the aircraft's motion. Mounting an IMU near the transmitting and receiving antennas is a good method to capture most of the antetenna's motion but still some residual error remains. In [9], there is detailed analysis of these motion measurement errors. Autofocus algorithms such as PGA are intended to remove the these motion measurement errors [25, 26].

Chapter 3

SAR Resolution

3.1 Contrast

The idea of using a radar system mounted on an aircraft to create a synthetic aperture to form an image was conceived by Carl Wiley in 1951 as shown in Figure 3.1 [27]. During this time period, modern digital computing devices did not exist. The technologies required for digital computing, like the integrated circuit or even the transistor were not invented at this time. While at Goodyear Aerospace in Goodyear, Arizona, Wiley developed a way to process the data using optical computing methods by transforming the data that is received through a radar system, known as radar returns, into an image of the ground. Originally Wiley called this image formation method Doppler Beam Sharpening (DPS) but further development by many researchers led to the development of what it is now known as Synthetic Aperture Radar (SAR) [2].

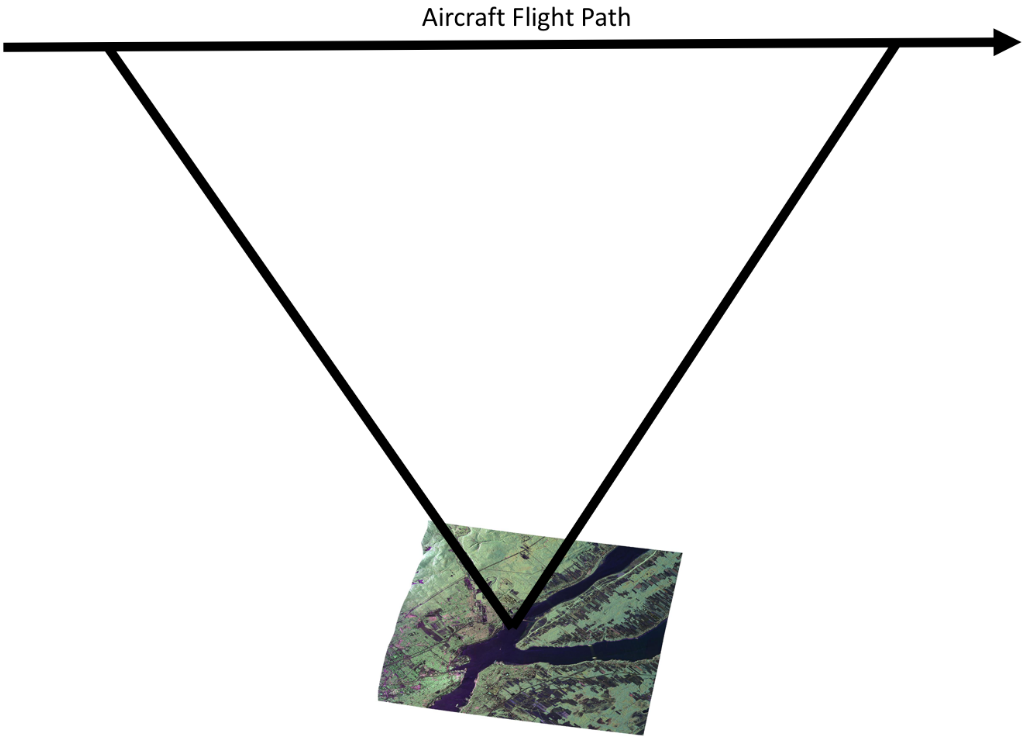

Figure 3.1: *Aircraft flying a synthetic aperture to image the terrain. (SAR Image Courtesy of NASA/JPL)*

All images, including SAR images, require contrast to be of use to humans or computers. Low image contrast makes objects of interest or targets, indiscernible within any image. The definition of contrast, C, for a SAR image is [9],

$$C = \frac{\frac{\sigma_t \cos(\psi_{grazing})}{\Delta cr \Delta r} + \sigma_N}{\sigma_0 + \sigma_N} \quad .$$

Where σ_t is the radar cross section of the target, $\psi_{grazing}$ is the grazing angle to the target, σ_0 is the terrain's backscattering coefficient and σ_N is the image noise which includes the multiplicative noise from the radar system. The terms Δcr and Δr are the cross-range and range resolution, respectively. According to [9], a contrast ratio of 15 dB is a requirement for any SAR system. This means that other ratios, like the radar system signal-to-noise ratio (SNR), will need to be taken into account as well.

The equation for contrast also demonstrates the need for low system noise and low terrain backscattering because both noise contributors are included within σ_N. Changing different system parameters within the radar receiver will control the total noise, σ_N . The total noise, σ_N, is added to the numerator and denominator of the contrast ratio and if the image noise is extremely large, a contrast ratio of 0 dB can be achieved and the final image would be indiscernible. A real world example of this effect is white noise on a television screen, which has contrast but also a completely unrecognizable image on the screen. In most cases, the scene surrounding the target is not easily modifiable, but there are radar system parameters which can control the intensity of the terrain backscatter, σ_0. These system parameters are polarization, frequency, and bistatic angle for bistatic systems and the first two parameters are applicable to monostatic SAR.

Another significant aspect of the contrast ratio for a SAR image is it's dependence on the image resolution. The terms Δr and Δcr are the range and azimuth resolutions respectively. Initial analysis of the equation for contrast shows an inverse relationship between the resolution terms mentioned above and the image contrast discussed as well. As spatial resolution terms decrease and becomes finer the image contrast increases and improves image quality.

3.2 Range Resolution

Range resolution, Δr of a radar system is determined by the radar's transmitted pulse. Depending on the plane where the image is processed, Δr can either represent the resolution in the slant, bistatic, or ground plane or another defined plane with which the image is projected. The fundamental definition of resolution in the slant plane is the minimum range between two targets needed to resolve or determine that each target is present within the pulse compressed radar data [28]. This derivation of range resolution in the radar's slant plane, is

$$\Delta r_{slant} = \frac{c\tau}{2} \quad . \tag{3.1}$$

The equation that defines the resolution is determined by the duration of the pulse called the pulse width, τ, and the speed of the electromagnetic wave traveling trough the medium, c. In most radar applications, the medium is air and the speed of electromagnetic wave is very close to the speed of light in a vacuum.

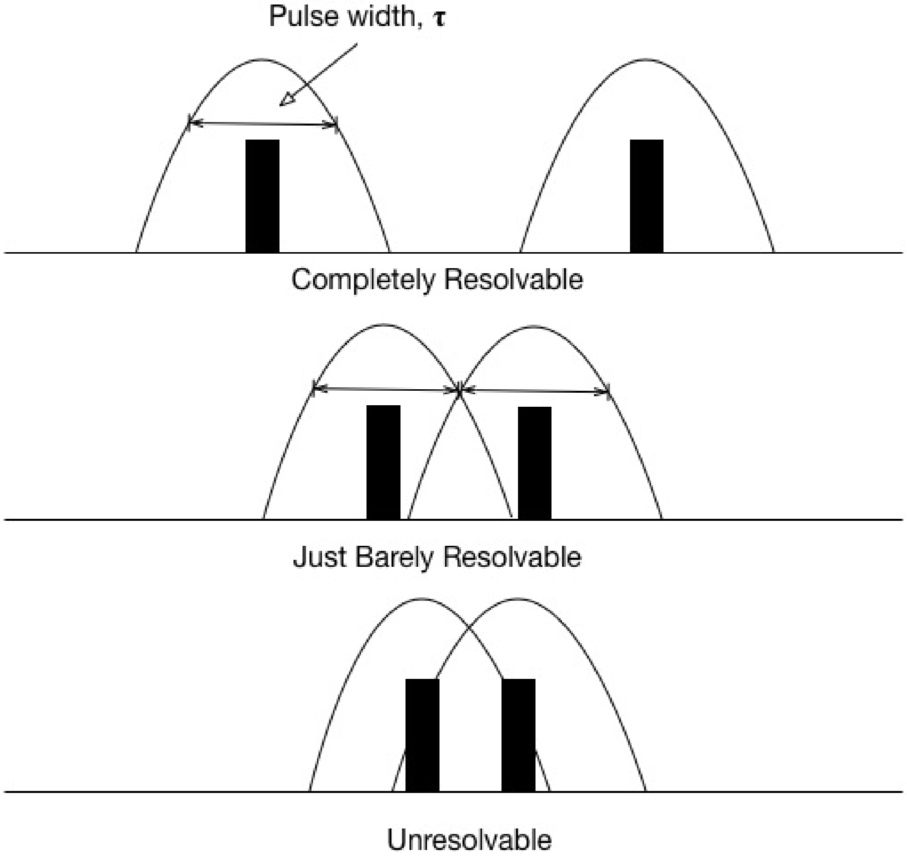

Figure 3.2: *Range resolution determined by pulse width.*

If two targets are unresolvable they appear as a single target to the radar as shown in Figure 3.2. This relationship between resolution and pulse width indicates that the shorter the pulse, the smaller and finer the range resolution of the radar system. Generally, the power transmitted by the radar system is fixed, decreasing the pulse width of the transmitted waveform will also decrease the transmitted signal energy incident on the target. The decreased transmitted signal energy decreases the SNR as well.

It is uncommon and unrealistic to overcome this relationship by increasing the transmitted power. Most radar systems overcome this issue by transmitting longer phase or frequency coded waveforms. The most common coded waveform employed is a linear frequency modulated (LFM) pulse, especially in SAR systems.

An LFM waveform chirps over a frequency span, the difference between the higher and lower frequencies of this frequency span is called the waveform bandwidth, BW. The signal used to generate the LFM waveform is

$$m(t) = a(t)exp(j(\omega_0 t + \pi\gamma t^2) \quad .$$

The function $m(t)$ is the transmit LFM waveform, where $a(t)$, determines the amplitude of the chirp. The function $a(t)$ acts like an on/off switch and would equal 1 while $t < \tau$ and zero otherwise. Different windows may be included within the $a(t)$ function to reduce the ringing from rise and fall of the pulse as well. The angular center frequency of the pulse is the w_0 term and γ is the chirp rate of the LFM waveform. The waveform bandwidth, in Hertz, can be determined by [10]

$$BW = \frac{\gamma\tau}{\pi} \quad [Hz] \quad .$$

From this bandwidth term, the monostatic slant range resolution of the LFM waveform can be determined, as

$$\Delta r_{slant} = \frac{c}{2BW} \quad .$$

This equation for range resolution can be generalized for other waveforms. For the slant range resolution in equation (3.1), by using a modulated pulse with a pulse width τ, the bandwidth is determined through the inverse relationship between time and frequency resulting $BW = \frac{1}{\tau}$ [28]. Substituting the bandwidth, $\frac{1}{\tau}$ leads to our initial equation for range resolution (3.1).

Many radar imaging applications project the range dimension to the ground plane, making the ground range resolution coarse compared to slant range resolution in the slant plane. Figure 3.3 illustrates this projection.

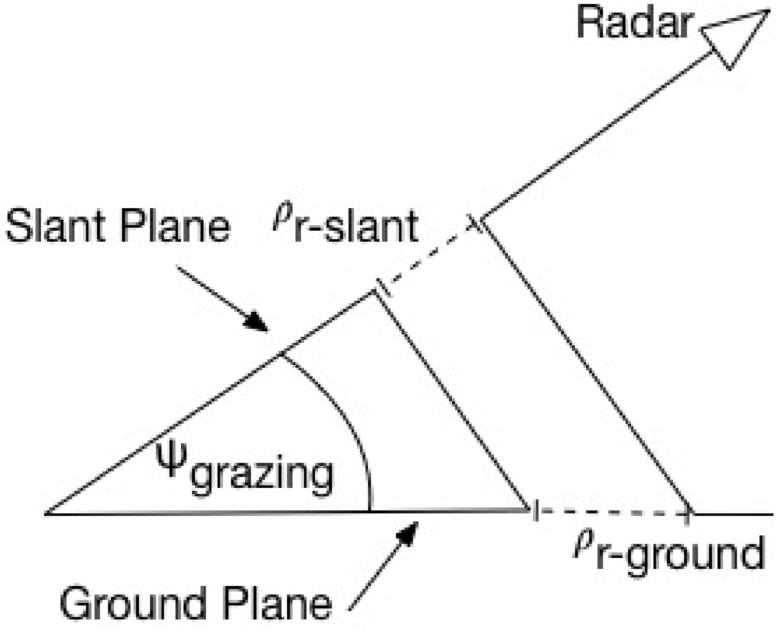

Figure 3.3: *Range Resolution in the ground and slant planes.*

The following equation determines the monostatic range resolution in the ground plane as [29]

$$\Delta r_{ground} = \frac{\Delta r_{slant}}{\sin(\psi_{grazing})} = \frac{c}{2BW\sin(\psi_{grazing})} \quad .$$

3.3 Bistatic Range Resolution

For bistatic ground range resolution the $x - y$ plane will be aligned witha ground or terrian plane in a Cartesian space. In conventional Earth imaging SAR systems, this plane would be the surface of the Earth. From the signal model, the spectral information is projected into the ground plane using the k-space and the $k_x - k_y$ plane. Then the spectral information goes through an inverse Fourier transformed creating a two-dimensional image using Fourier based imaging algorithms like the polar format algorithm [6, 13]. Conversely, individual bistatic radar returns can be pulse compressed and backprojected using a bistatic backprojection algorithms [30]. In the ground plane, it is important to understand the effects of the transmitter's and receiver's position in k-space which will ultimately affect the range resolution in the ground plane. One method to find the resolution is to determine the wavenumber's magnitude in terms of each radar system's azimuth angles, (θ_{TX} and θ_{RX}) and elevation angles (ϕ_{TX} and ϕ_{RX}) to the scene reference point (SRP), illustrating their impact.

The key to determining the ground plane range resolution is an understanding of the k-space within the $k_x - k_y$ plane. Understanding the minimum and maximum values of $|k_{x-y}|$ leads to the definition of range resolution in the ground plane. Hence, the magnitude of an instantaneous sample of a single radar return in k-space is,

$$|k_{x-y}| = \mathbf{k} \cdot \mathbf{u} = \sqrt{k_x^2 + k_y^2} \quad , \tag{3.2}$$

where the vector $\mathbf{u}$ is a unit vector that bisects the angle β. The inner product of $\mathbf{k}$ and $\mathbf{u}$ is the projection of the k-space into the ground plane. Due to the placement of the Cartesian coordinate system, the $k_x - k_y$ plane is aligned with the ground plane. Hence, a projection of the k-space into the ground plane is found by analyzing the k-space within the $k_x - k_y$ plane and it is an approximation of the signal model. Then, substituting Equations (2.19) and (2.20) in Equation (3.2), the magnitude becomes,

$$|k_{x-y}| = \frac{k_i}{2}\sqrt{\Big(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX})\Big)^2 ... \\ +\Big(\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX})\Big)^2} \quad . \tag{3.3}$$

The term k_i represents an arbitrary point in the k-space of a transmitted waveform. Later on in this section, k_i will be replaced with k_{max} and k_{min}, which are proportional to the start and stop frequencies of the transmitted waveform. Next, we expand the squared terms in (3.3) to arrive at

$$|k_{x-y}| = \frac{k_i}{2}\sqrt{\cos^2(\theta_{TX})\cos^2(\phi_{TX}) + \cos^2(\theta_{RX})\cos^2(\phi_{RX}) ... \\ +\sin^2(\theta_{TX})\cos^2(\phi_{TX}) + \sin^2(\theta_{RX})\cos^2(\phi_{RX}) ... \\ +2\cos(\theta_{TX})\cos(\theta_{RX})\cos(\phi_{TX})\cos(\phi_{RX}) ... \\ +2\sin(\theta_{TX})\sin(\theta_{RX})\cos(\phi_{TX})\cos(\phi_{RX})} \quad . \tag{3.4}$$

Using the trigonometric identities $\sin^2(A) + \cos^2(A) = 1$ and $\cos(A - B) = \cos(A)\cos(B) + \sin(A)\sin(B)$, the magnitude of the k-space becomes

$$|k_{x-y}| = \frac{k_i}{2}\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\theta_{TX} - \theta_{RX})cos(\phi_{TX})cos(\phi_{RX})} \quad . \tag{3.5}$$

The bistatic angle between the transmitter and receiver is defined as $\beta = \theta_{TX} - \theta_{RX}$. The cosine function is an even function that results in the same value if $\beta = \theta_{RX} - \theta_{TX}$. Therefore, (3.5) becomes

$$|k_{x-y}| = \frac{k_i}{2}\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\beta)cos(\phi_{TX})cos(\phi_{RX})} \quad . \tag{3.6}$$

Next, the relationship between $|k_{x-y}|_{max}$ and $|k_{x-y}|_{min}$ and its importance in determining ground plane range resolution is demonstrated. The two-way propagation is included in the wavenumber definition in Equation (2.16), so that a factor of two is included in the numerator of the definition of range resolution in

$$\Delta r_g = \frac{2\pi}{|k_{x-y}|_{max} - |k_{x-y}|_{min}} \quad . \tag{3.7}$$

Equation (3.7) is an approximation of the -3dB width of the point spread function in a Rayleigh sense [15, 16]. The term k_{max} occurs when the largest frequency of the waveform is transmitted while k_{min} occurs when the smallest frequency is transmitted. Substitution of $|k_{x-y}|_{max}$ and $|k_{x-y}|_{min}$ leads to

$$\Delta r_g = \frac{4\pi}{(k_{max} - k_{min})\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\beta)cos(\phi_{TX})cos(\phi_{RX})}} \quad . \tag{3.8}$$

Finally, we find that Equation (3.8) allows us to simplify the range resolution within the ground plane. Where $(k_{max} - k_{min}) = \frac{4\pi(f_{max}-f_{min})}{c} = \frac{4\pi BW}{c}$ and the term BW is the transmitted waveform's bandwidth. This simplifies the range resolution in the ground plane to Equation (3.9).

$$\Delta r_g = \frac{c}{BW\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\beta)cos(\phi_{TX})cos(\phi_{RX})}} \tag{3.9}$$

Forward-scatter bistatic SAR range resolution is predicted in Equation (3.9).

Equation (3.9), bistatic range resolution in the ground plane, first appeared in [19], but this equation can be derived from equations (14) and (15) within [4], as well.

When the bistatic angle, $\beta = 180°$ and the elevation angles, $\phi_{TX} = \phi_{RX}$, the square root term in the denominator becomes zero, resulting in the range resolution to be infinite. The larger the difference between ϕ_{TX} and ϕ_{RX}, the finer the range resolution becomes as is shown in Figure 3.4. Recently, the authors of [4] came to the same conclusions about the effects of the elevation angle and bistatic angle on bistatic range resolution. The magnitude of the bistatic k-space in the x-y plane, in Equation (3.6), can be transformed by $\phi = \phi_{TX} = \phi_{RX}$ and then using the trigonometric identity $\cos^2 \frac{A}{2} = \frac{1}{2}(1 + \cos(A))$ to create

$$|k_{x-y}| = k_i \cos(\phi) \cos(\beta/2) \quad . \tag{3.10}$$

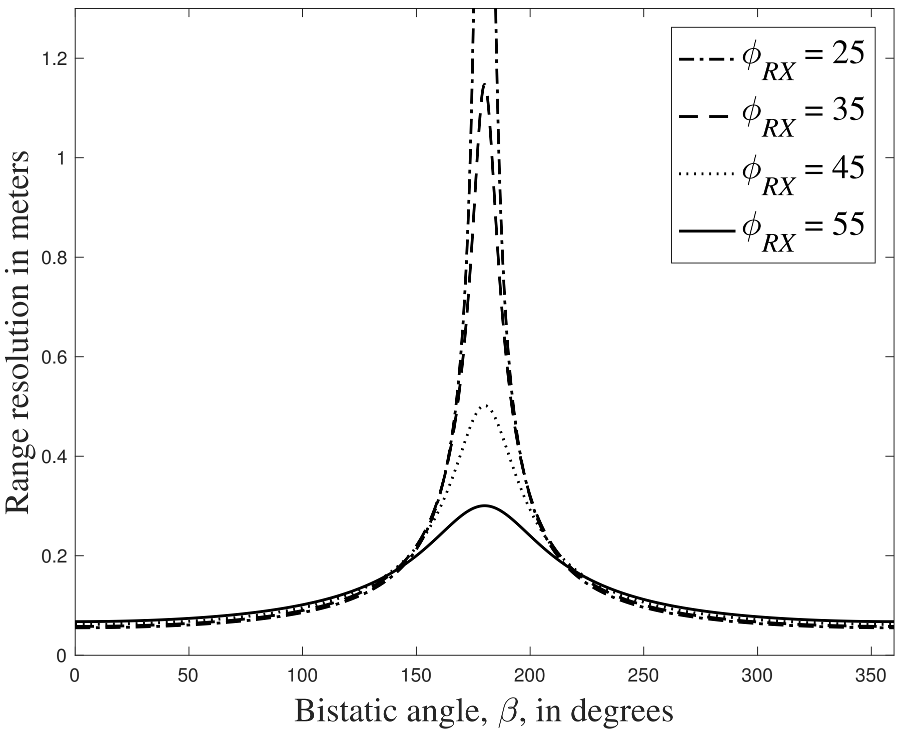

Figure 3.4: *Bistatic range resolutions with fixed $\phi_{TX} = 25^\circ$ and indicated ϕ_{RX} values. The transmit bandwidth is 3 GHz.*

Then (3.7) can be transformed into the equation for bistatic range resolution found in [6, 13], and [17] whereas ϕ is described as the the elevation angle of the bistatic k-space,

$$\Delta r_g = \frac{c}{2BW\cos(\phi)\cos(\beta/2)} \quad . \tag{3.11}$$

Equation (3.9) describes the resolution in small bistatic and unequal elevation angles as well as the resolution in the forward-scatter region based on the arbitrary positions of the transmitter and receiver. The ground plane range resolution equations found in Chapter 10 of [13] and [17] rely on the

k-space manifold in Equations (2.19) to (2.21) in order to find the elevation angle of the k-space manifold. Equation (3.9) is a single relationship based on the individual contributions of each bistatic system revealing how to generate resolution in challenging bistatic configurations like cases when $\beta = 180°$.

The angle β is the major difference between Equation (3.11) and ground plane range resolution equations found in Chapter 10 of [13], [15] and [17]. As discussed, in Section 2.1, β is defined in the ground plane as it was originally in, [31] but in the range resolution equation found in [17] and [32] it is defined in the bistatic plane. Equation (3.9) relies solely on the system parameters of the bistatic system: the transmitted waveform's bandwidth and bistatic systems' positions. Other forms rely on computing the elevation angle of the k-space to project the resolution into the ground plane. In Chapter one of [33], a bistatic range resolution equation is developed for a bistatic system above a flat and curved Earth, but the transmitter and receiver must have equal or coupled heights. Computing resolution based only on uncoupled system parameters aids in understanding when mission planning.

The bistatic range resolution equation was orignially developed in [31] and cited and republished in [7]. In [31], range resolution was defined on a 2D plane and this plane was North referenced. In [7] it is stated that the range resolution equation is an approximation for bistatic systems with low grazing angles. Also within [7] the idea of the bistatic plane was introduced in [34] where the bistatic plane includes the transmitter, receiver, and the target. For flat Earth models, the elevation angles (ϕ_{TX} and ϕ_{RX}) are the grazing angles. If the targets have a grazing angle that is not zero, it means that the bistatic angle defined in [31] is no longer within the bistatic plane. This is the motivation for defining the bistatic angle within the ground plane. The author of this document decided to suspend the idea of the bistatic plane but continue to use the symbol β. The term β can be easily replaced with $\beta = \phi_{TX} - \phi_{RX}$ or $\beta = \phi_{RX} - \phi_{TX}$ for the range and cross-range resolution equations due to the even property of the cosine function. The intention for leaving the bistatic angle within the ground plane, was to connect the range resolution definitions in [7] and [31] with monostatic cross-range resolution equations in [9] and [10], while using the bistatic SAR work developed within [6, 13] and [17].

3.4 Alternative Derivation of Bistatic Range Resloution

In Sections 3.3 and 3.6, range and cross-range resolution in the ground plane are determined by finding the minimum and maximum values of the k-space in the $x - y$ plane due to the placement of the Cartesian coordinate system. The intention is to provide a derivation that allows the readers to understand the genesis of resolution. Finding the minimum and maximum points of the k-space in the ground plane aids in giving meaning to this spectral SAR domain. The inverse relationship between k-space and SAR imagery is challenging for some engineers therefore, adding another layer of complexity. To avoid this complexity the vector projection was mentioned but not employed in the previous section. Range resolution in the ground plane can be derived from a fundamental definition of resolution, using the work found in Section 3.3 and a vector projection. To begin this derivation, a basic definition of range resolution that is stated as

$$\Delta r = \frac{\Delta \tau c}{2} \quad . \tag{3.12}$$

Where Δr is the resolution, c is the speed of light, and $\Delta \tau$ is the pulse width of the transmitted pulse. Then using, $\Delta\tau = \frac{0.866}{\Delta f}$, described in [32], in Equation (16), where Δf is the bandwidth of the signal which generates the resolution, Equation (3.12) becomes

$$\Delta r = \frac{0.866c}{2\Delta f} \quad . \tag{3.13}$$

Then the ground range resolution is defined as,

$$\Delta r_g = \frac{0.866c}{2\Delta f_g} \quad , \tag{3.14}$$

where the Δf_g term is the bandwidth in the ground plane.

A fundamental aspect of any range resolution derivation is how the bandwidth term, BW, is generated. The k-space Equations (2.19), (2.20) and (2.21) within Chapter 2 are neccessary and placed in the vector $\mathbf{k} = [k_x, k_y, k_z]^T$. Then using the relationship $k = \frac{2w}{c} = \frac{4\pi f}{c} = \frac{4\pi}{\lambda}$ [9] leads to

$$\Delta f_g = \frac{\Delta k_g c}{4\pi} \quad , \tag{3.15}$$

where Δk_g is the magnitude of the k-space projected into the ground plane. This may be determined two different ways. The method outlined in Section 3.3 or by projecting the k-space with a unit vector to arrive at a similar range resolution equation.

Next, a unit vector is needed within the ground plane, so that the z component is set to zero.

$$\mathbf{u} = \frac{[\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}), \sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}), 0]}{\sqrt{(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))^2 + (\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))^2}} \tag{3.16}$$

Then as shown in the signal model of [19] the k-space of a single radar return is located at the average positions of the transmitter's and receiver's positions in Equations (2.19)-(2.21) which leads to

$$\mathbf{k_{min}} = \begin{bmatrix} k_{min}\frac{\cos(\theta_{TX})\cos(\phi_{TX})+cos(\theta_{RX})\cos(\phi_{RX})}{2} \\ k_{min}\frac{\sin(\theta_{TX})\cos(\phi_{TX})+\sin(\theta_{RX})\cos(\phi_{RX})}{2} \\ k_{min}\frac{\sin(\phi_{TX})+\sin(\phi_{RX})}{2} \end{bmatrix} \tag{3.17}$$

and

$$\mathbf{k_{max}} = \begin{bmatrix} k_{max}\frac{\cos(\theta_{TX})\cos(\phi_{TX})+cos(\theta_{RX})\cos(\phi_{RX})}{2} \\ k_{max}\frac{\sin(\theta_{TX})\cos(\phi_{TX})+\sin(\theta_{RX})\cos(\phi_{RX})}{2} \\ k_{max}\frac{\sin(\phi_{TX})+\sin(\phi_{RX})}{2} \end{bmatrix} . \tag{3.18}$$

Then subtracting $\mathbf{k_{min}}$ from $\mathbf{k_{max}}$ leads to

$$\mathbf{\Delta k} = \mathbf{k_{max}} - \mathbf{k_{min}} = \begin{bmatrix} (k_{max}-k_{min})\frac{\cos(\theta_{TX})\cos(\phi_{TX})+cos(\theta_{RX})\cos(\phi_{RX})}{2} \\ (k_{max}-k_{min})\frac{\sin(\theta_{TX})\cos(\phi_{TX})+\sin(\theta_{RX})\cos(\phi_{RX})}{2} \\ (k_{max}-k_{min})\frac{\sin(\phi_{TX})+\sin(\phi_{RX})}{2} \end{bmatrix} ; \tag{3.19}$$

where $\mathbf{\Delta k}$ is a vector pointing in the bisector of the bistatic angle within the bistatic plane and has a magnitude equal to $k_{max} - k_{min}$. To determine the bandwidth, BW, this relationship, $k = \frac{2w}{c} = \frac{4\pi f}{c} = \frac{4\pi}{\lambda}$, is used to find

$$(k_{max} - k_{min}) = \frac{4\pi(f_{max} - f_{min})}{c} = \frac{4\pi BW}{c} \quad , \tag{3.20}$$

leading to

$$\mathbf{\Delta k} = (\mathbf{T_x}+\mathbf{R_x})\frac{4\pi BW}{c} = \frac{2\pi BW}{c}\begin{bmatrix}\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}) \\ \sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}) \\ \sin(\phi_{TX}) + \sin(\phi_{RX})\end{bmatrix} \quad . \tag{3.21}$$

The term, $(\mathbf{T_x}+\mathbf{R_x})$ is the sum of the angular contributions of the transmitter and receiver. These angular contributions can be found in the k-space equations, Equations (2.19) - (2.20) by removing the k term. To determine Δk_g, we multiply the unit vector (u) and Δk results in:

$$\Delta k_g = (u) \cdot \Delta k \quad . \tag{3.22}$$

Then, substituting Equation (3.21) into (3.22), the k-space in the ground plane is found as

$$\begin{aligned}\Delta k_g &= (u) \cdot (\mathbf{T_x} + \mathbf{R_x})\frac{4\pi BW}{c} \\ &= \frac{2\pi BW}{c}(u) \cdot \begin{bmatrix}\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}) \\ \sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}) \\ \sin(\phi_{TX}) + \sin(\phi_{RX})\end{bmatrix} \quad .\end{aligned} \tag{3.23}$$

Then, substitution of **u** in Equation (3.16) and including the inner product, leads to,

$$\Delta k_g = \frac{2\pi BW}{c}\frac{(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))^2 + (\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))^2}{\sqrt{(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))^2 + (\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))^2}} \quad . \tag{3.24}$$

Equation (3.24) can be further simplified to

$$\Delta k_g = \frac{2\pi BW}{c}\sqrt{(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))^2 + (\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))^2} \quad . \tag{3.25}$$

Next, using Equations (3.3) to (3.6) from Section 3.3, this equation can be further simplified to

$$\Delta k_g = \frac{2\pi BW}{c}\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\beta)cos(\phi_{TX})cos(\phi_{RX})} \quad . \tag{3.26}$$

Finally, substituting Equations(3.26) into Equations (3.14) and (3.15) it is found that,

$$\Delta r_g = \frac{0.866c}{BW\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\beta)cos(\phi_{TX})cos(\phi_{RX})}} \quad . \tag{3.27}$$

Equation (3.27) has very similar form for ground range resolution as Equation (3.9) with the exception of the 0.866 term within the numerator. This value can be set to unity or replaced with another windowing term. It's useful to understand the ground resolutions in terms of vector projections, but the reader can easily get lost within the math. This was the intention behind creating this separate section. A similar method can be used to find cross-range resolution in the ground plane as well.

3.5 Impacts of a Bistatic Radar Systems' Elevation Angles on K-space

The inverse relationship between bistatic range resolution and k-space is described in Equation (3.7), which shows that the larger the difference between the minimum and maximum wavenumbers, the finer the range resolution becomes. Equation (3.11) is the conventionally used equation for bistatic range resolution. Equation (3.11) indicates that there is extremely coarse resolution in the forward scatter bistatic region even when the elevation angles are unequal, which is opposite of the previously developed Equation (3.9) and the results in Chapters 5 and 6. Understanding the bistatic k-space will help to illustrate how this forward scatter resolution is created.

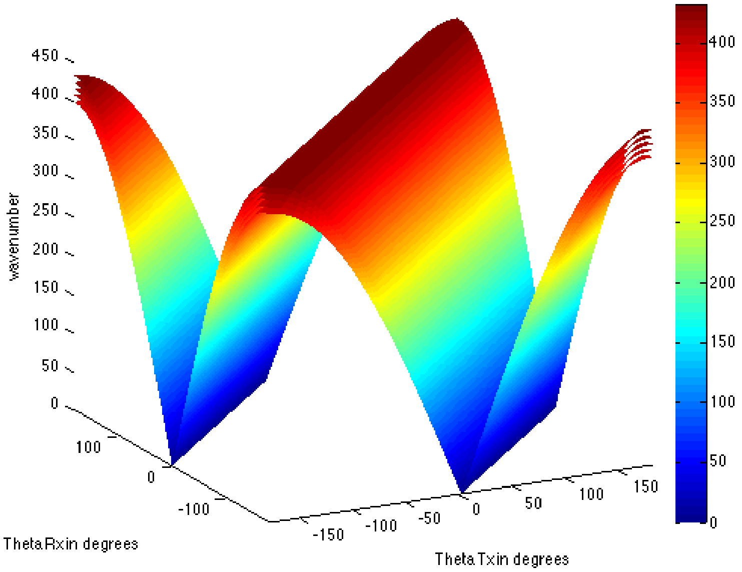

Figure 3.5: *Three-Dimensional View of the k-space while the transmitter and receiver azimuth angles vary. The transmitter's and the receiver's elevation angle is* $0\,°$.

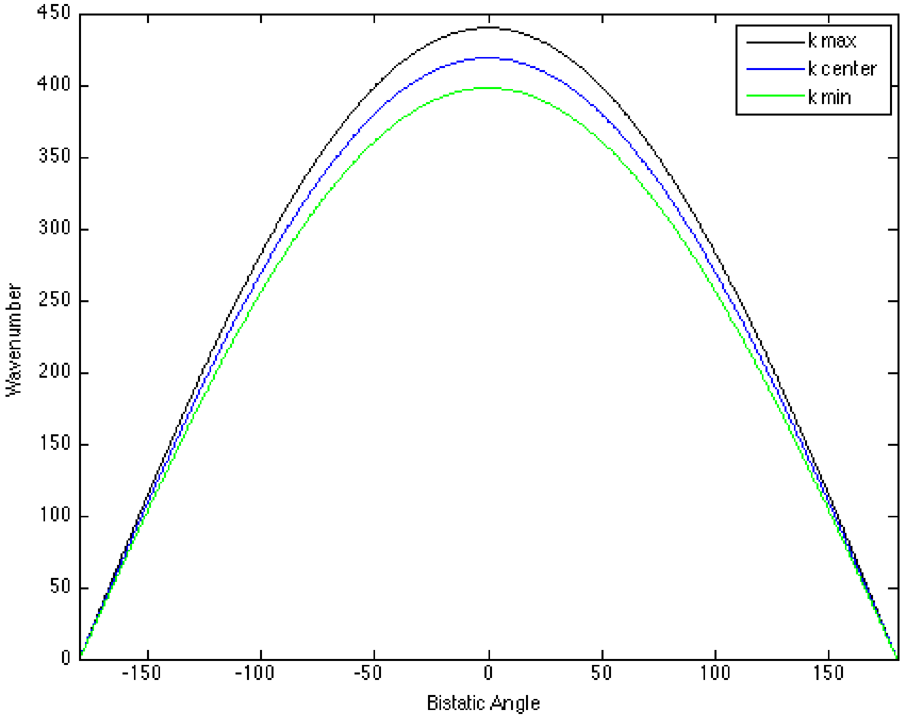

Figure 3.6: *Two-Dimensional View of the k-space based on the bistatic azimuth angle between the receiver and transmitter. The transmitter's and the receiver's elevation angle is* 0°.

Figure 3.5 is a three-dimensional view of the k-space based on the transmitter's and receiver's azimuth angle to the target, which is mathematically represented by Equation (3.5). This same k-space can be viewed in terms of bistatic angle as shown in Figure 3.6 and mathematically represented by Equation (3.6). In Figures 3.5 and 3.6, the center frequency of the transmitted waveform is $10GHz$ and the transmitted bandwidth is $1GHz$. When the difference between θ_{TX} and θ_{RX} is less than 20°, the k-space is in the forward scatter region. This forward scatter region exists in the bluest parts of the plot of Figure 3.5 and also when the absolute value of the bisatic angle, β, is between 160° and 180° as shown in Figure 3.6. In Figures 3.5 and 3.6, both the transmitter's and receiver's elevation angles are equal and set to 0°. From inspection of both Figures 3.5 and 3.6, in the forward scatter region, the difference between the minimum and maximum wavenumbers is small or zero creating extremely coarse range resolution. Although these elevation angles are not realistic, they assist in a better understanding of Equation (3.10) through inspection. They show that in the case where the bistatic radar system's elevation angles are equal, the range resolution is extremely coarse.

It is also important to understand the impact of unequal elevation angles to a bistatic k-space with the same waveform parameters of a $10GHz$ center frequency and bandwidth of $1GHz$. In the next example, the transmitter's elevation angle, ϕ_{TX}, is 45° degrees and the receiver's elevation angle, ϕ_{RX}, is 10°. Using Equations (3.5) and (3.6) this example is plotted in Figures 3.7, 3.8 and 3.9.

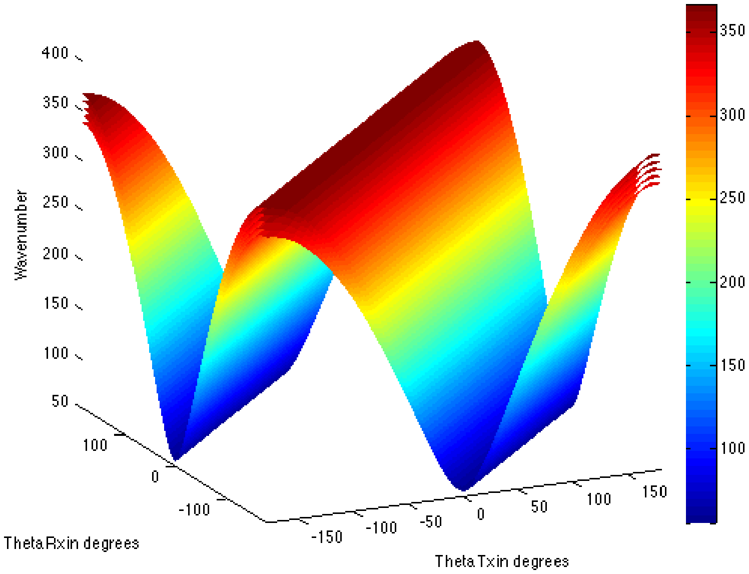

Figure 3.7: *Three-Dimensional View of the k-space while the transmitter and receiver azimuth angles vary.The transmitter's elevation angle is 45° and the receiver's elevation angle is 10°.*

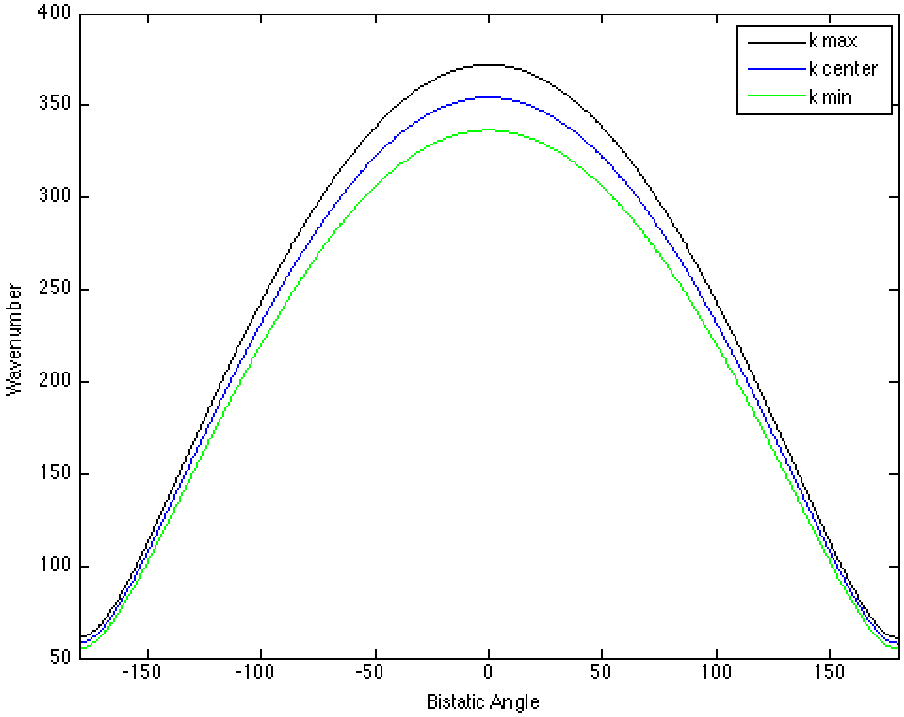

Figure 3.8: *Two-Dimensional View of the k-space based on the bistatic azimuth angle between the receiver and transmitter. The transmitter's elevation angle is 45° and the receiver's elevation angle is 10°.*

The contribution of unequal elevation angles is that a significant bandwidth of the signal exists in the ground plane at large bistatic angles. With unequal elevation angles and a large bistatic angle, the wavenumber does not tend completely towards zero. The latter means that there is some range resolution in the ground plane for forward scatter scenarios when the elevation angles are not the same.

Figure 3.9 shows a more detailed view of this forward-scatter resolution in the k-space domain. Although the distance is smaller than it would be in the backscatter scenarion, $\beta = 0^\circ$, there is a change in wavenumbers from the maximum to the minimum wavenumber generating resolution to form an image.

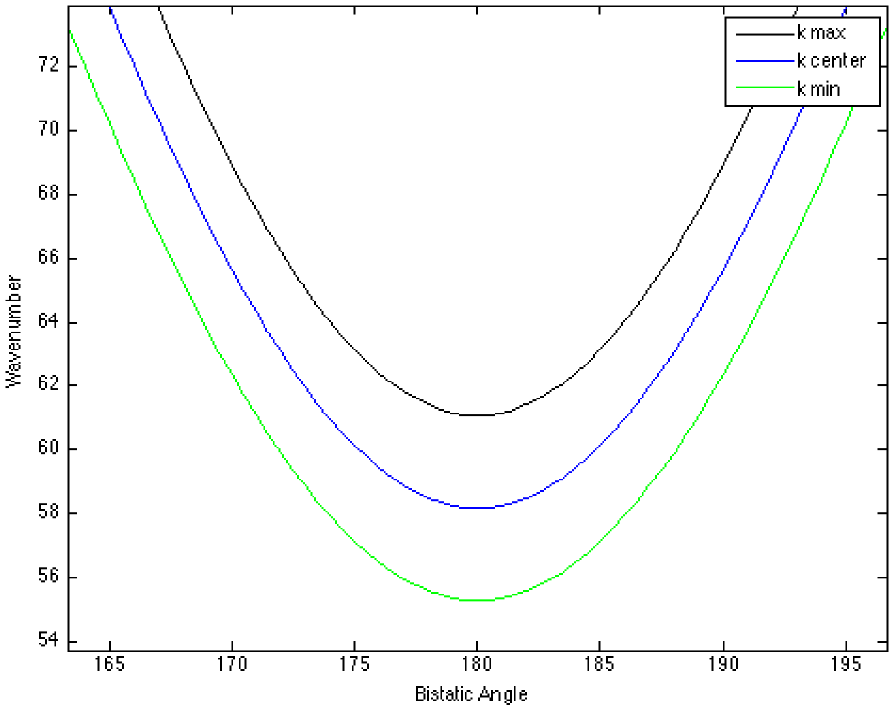

Figure 3.9: *Zoomed in view of the k-space for the forward scatter region. The transmitter's elevation angle is* 45° *degrees and the receiver's elevation angle is* 10°*.*

The distances found between the minimum and maximum wavenumbers led to the development of the bistatic range resolution equation found in the previous Section 3.3 and Equation (3.9). This range resolution equation describes the range resolution found in the backscatter, bistatic, and forward scatter regions.

3.6 Derivation of Bistatic Cross-Range Resolution

The bistatic cross-range resolution is found using a similar methodology to derive the bistatic range resolution found in Section 3.3. Considering the wavenumber manifold defined by Equations (2.19) to (2.21), the cross-range resolution would be defined by the distance in wavenumbers between the edges of the manifold that begins when the data collection starts and ends when the collection stops at the lowest instantaneous wavenumber or frequency, $k_{low} = \frac{4\pi f_{low}}{c}$. Using the lowest wavenumber or frequency allows for an accurate estimate of the cross-range resolution because in many Fourier-domain SAR algorithms, the higher wavenumbers at the edges are unused after an interpolation step of the SAR algorithm [6] as depicted in horizontal dimension of Figure 5.22. Hence

$$\Delta cr_g = \frac{2\pi}{d(k_{low_{stop}}, k_{low_{start}})} \quad , \tag{3.28}$$

where the function $d(k_{low_{stop}}, k_{low_{start}})$ finds the Cartesian distance between these two point in the k_{x-y} plane defined in Equation (3.29). Equation (3.28) is an approximation of the -3dB width of the point spread function in a Rayleigh sense [15, 16]. The Cartesian distance, $d(k_{low_{stop}}, k_{low_{start}})$, is depicted in horizontal dimension of Figure 5.22 and defined as

$$d(k_{low_{stop}}, k_{low_{start}}) = \sqrt{(k_{X_{stop}} - k_{X_{start}})^2 + (k_{Y_{stop}} - k_{Y_{start}})^2} \quad . \tag{3.29}$$

This derivation will begin with expanding each term within (3.29) by substituting their values from Equations (2.19) and (2.20), where the cross-range distance in the k-space is found. The derivation begins by determining

$$
\begin{aligned}
k_{X_{stop}} - k_{X_{start}} = \\
&\frac{k_{low}}{2}\Big(\Big(\cos(\theta_{TX_{stop}}) - \cos(\theta_{TX_{start}})\Big)\cos(\phi_{TX})... \\
&+ \Big(\cos(\theta_{RX_{stop}}) - \cos(\theta_{RX_{start}})\Big)\cos(\phi_{RX})\Big) \quad . \qquad (3.30)
\end{aligned}
$$

Where it is assumed that ϕ_{TX} and ϕ_{RX} are constant during the collection, hence their average values are adopted in Equation (3.30). Then using the trigonometric identity $\cos(X) - \cos(Y) = -2\sin(\frac{X+Y}{2})\sin(\frac{X-Y}{2})$, Equation (3.30) becomes Equation (3.31) where $\Delta\theta_X = \theta_{X_{stop}} - \theta_{X_{start}}$. The subscript X is a place holder for the transmitter or receiver's subscript, TX and RX, respectively.

$$
\begin{aligned}
k_{X_{stop}} - k_{X_{start}} = \\
&- k_{low}\Big(\sin(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2})\sin(\frac{\Delta\theta_{TX}}{2})\cos(\phi_{TX})... \\
&+ \sin(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2})\sin(\frac{\Delta\theta_{RX}}{2})\cos(\phi_{RX})\Big) \qquad (3.31)
\end{aligned}
$$

Then squaring $k_{X_{stop}} - k_{X_{start}}$ leads to

$$
\begin{aligned}
(k_{X_{stop}} - k_{X_{start}})^2 = \\
&k_{low}^2\Big(\sin^2(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2})\sin^2(\frac{\Delta\theta_{TX}}{2})\cos^2(\phi_{TX})... \\
&+ \sin^2(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2})\sin^2(\frac{\Delta\theta_{RX}}{2})\cos^2(\phi_{RX})... \\
&+ 2\sin(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2})\sin(\frac{\Delta\theta_{TX}}{2})\cos(\phi_{TX})... \\
&\cdot \sin(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2})\sin(\frac{\Delta\theta_{RX}}{2})\cos(\phi_{RX})\Big) \quad . \qquad (3.32)
\end{aligned}
$$

Then $k_{Y_{stop}} - k_{Y_{start}}$ is found in a similar manner resulting in

$$\begin{aligned}
&(k_{Y_{stop}} - k_{Y_{start}})^2 = \\
&\qquad k_{low}^2 \Big(\cos^2(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2}) \sin^2(\frac{\Delta\theta_{TX}}{2}) \cos^2(\phi_{TX})... \\
&\qquad + \cos^2(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2}) \sin^2(\frac{\Delta\theta_{RX}}{2}) \cos^2(\phi_{RX})... \\
&\qquad + 2\cos(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2}) \sin(\frac{\Delta\theta_{TX}}{2}) \cos(\phi_{TX})... \\
&\qquad \cdot \cos(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2}) \sin(\frac{\Delta\theta_{RX}}{2}) \cos(\phi_{RX}) \Big) \quad . \qquad (3.33)
\end{aligned}$$

Adding Equations (3.32) and (3.33) and using the identity $\sin^2(A)+\cos^2(A) = 1$ result in

$$\begin{aligned}
&(k_{X_{stop}} - k_{X_{start}})^2 + (k_{Y_{stop}} - k_{Y_{start}})^2 = \\
&\qquad k_{low}^2 \Bigg(\sin^2(\frac{\Delta\theta_{TX}}{2}) \cos^2(\phi_{TX}) + \sin^2(\frac{\Delta\theta_{RX}}{2}) \cos^2(\phi_{RX})... \\
&\qquad + 2\sin(\frac{\Delta\theta_{TX}}{2}) \sin(\frac{\Delta\theta_{RX}}{2}) \cos(\phi_{TX}) \cos(\phi_{RX}) \cdot ... \\
&\qquad \Big(\sin(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2}) \sin(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2}) + ... \\
&\qquad \cos(\frac{\theta_{TX_{stop}} + \theta_{TX_{start}}}{2}) \cos(\frac{\theta_{RX_{stop}} + \theta_{RX_{start}}}{2}) \Big) \Bigg) \quad . \qquad (3.34)
\end{aligned}$$

Then, after using the trigonometric identity, $\sin(A)\sin(B) + \cos(A)\cos(B) =$

$\cos(A-B)$, this equation becomes

$$\begin{aligned}&(k_{X_{stop}}-k_{X_{start}})^2+(k_{Y_{stop}}-k_{Y_{start}})^2 = \\ &\qquad k_{low}^2\Big(\sin^2(\frac{\Delta\theta_{TX}}{2})\cos^2(\phi_{TX})+\sin^2(\frac{\Delta\theta_{RX}}{2})\cos^2(\phi_{RX})... \\ &\qquad +2\sin(\frac{\Delta\theta_{TX}}{2})\sin(\frac{\Delta\theta_{RX}}{2})\cos(\phi_{TX})\cos(\phi_{RX})\cdot ... \\ &\qquad \cos(\frac{\theta_{TX_{stop}}+\theta_{TX_{start}}-\theta_{RX_{stop}}+\theta_{RX_{start}}}{2})\Big) \quad . \end{aligned} \tag{3.35}$$

Then the bistatic angle between two radars is defined as $\beta=\theta_{TX}-\theta_{RX}$, which means $\beta_{stop}-\beta_{start}=\Big(\theta_{TX_{stop}}-\theta_{RX_{stop}}\Big)+\Big(\theta_{TX_{start}}-\theta_{RX_{start}}\Big)$. The bistatic angle for the collection would equal $\beta=\frac{\beta_{stop}-\beta start}{2}$, creating the final form of this equation as Equation (3.36):

$$\begin{aligned}&(k_{X_{stop}}-k_{X_{start}})^2+(k_{Y_{stop}}-k_{Y_{start}})^2 = \\ &\qquad k_{low}^2\Big(\sin^2(\frac{\Delta\theta_{TX}}{2})\cos^2(\phi_{TX})+\sin^2(\frac{\Delta\theta_{RX}}{2})\cos^2(\phi_{RX})... \\ &\qquad +2\sin(\frac{\Delta\theta_{TX}}{2})\sin(\frac{\Delta\theta_{RX}}{2})\cos(\phi_{TX})\cos(\phi_{RX})\cos(\beta)\Big) \end{aligned} \tag{3.36}$$

By combining Equations (3.28), (3.29) and (3.36), the equation for bistatic cross-range resolution is formulated as

$$\begin{aligned}\Delta cr_g = &\frac{2\pi}{k_{low}}\cdot ... \\ &\left(\frac{\sqrt{\sin^2(\frac{\Delta\theta_{TX}}{2})\cos^2(\phi_{TX})+\sin^2(\frac{\Delta\theta_{RX}}{2})\cos^2(\phi_{RX})...}}{+2\sin(\frac{\Delta\theta_{TX}}{2})\sin(\frac{\Delta\theta_{RX}}{2})\cos(\phi_{TX})\cos(\phi_{RX})\cos(\beta)}\right)^{-1} \quad , \end{aligned} \tag{3.37}$$

where $\frac{2\pi}{k_{low}}=\frac{2c}{fc-\frac{BW}{2}}=\frac{\lambda_{max}}{2}$ and f_c is the center frequency; BW is the bandwidth; and λ_{max} is the largest wavelength of the transmitted waveform. Finally, applying the definition $k = \frac{4\pi}{\lambda}$, Equation (3.37) for cross-range resolution becomes

$$\Delta cr_g = \frac{\lambda_{max}}{2} \cdot ... \\ \left(\frac{\sqrt{\sin^2(\frac{\Delta\theta_{TX}}{2})\cos^2(\phi_{TX}) + \sin^2(\frac{\Delta\theta_{RX}}{2})\cos^2(\phi_{RX})...}}{+2\sin(\frac{\Delta\theta_{TX}}{2})\sin(\frac{\Delta\theta_{RX}}{2})\cos(\phi_{TX})\cos(\phi_{RX})\cos(\beta)} \right)^{-1} . \tag{3.38}$$

3.7 Monostatic and Bistatic Cross-Range Resolution

Equation (3.38) shares a similar form with Equation (3.9), the derived bistatic SAR range resolution equation. The bistatic ground-plane cross-range resolution equation. Similar to Equation (3.9), this ground plane cross-range resolution equation, Equation (3.38), relies on uncoupled system parameters to compute ground plane range resolution, the longest transmitted wavelength, λ_{max} and the platform's positions. Cross-range resolution equations found in [6, 13, 17] need the angular extent of the k-space to determine resolution meaning the k-space equations are needed to find the cross-range resolution in the ground plane. When mission planning, a better understanding of spatial resolution is achieved when the resolution is based on uncoupled platform positions and the transmitted waveform. The sign of small angles approximation also found in the monostatic formulation in [10], is used to reduce the equation to

$$\Delta cr_g = \frac{\lambda_{max}}{\sqrt{\Delta\theta_{TX}^2 \cos^2(\phi_{TX}) + \Delta\theta_{RX}^2 \cos^2(\phi_{RX}) + 2\Delta\theta_{TX}\Delta\theta_{RX} \cos(\phi_{TX}) \cos(\phi_{RX}) \cos(\beta)}} \quad , \tag{3.39}$$

where the $\Delta\theta_X$ terms are in units of radians. This form of the equation is analogous to the forms of monostatic SAR cross-range resolution found in [9, 10] and can be solved for the monostatic equation.

First by setting $\Delta\theta_{RX} = \Delta\theta_{TX} = \Delta\theta$, this describes a collection where both the transmitter and receiver have covered the same angular extent throughout the aperture, which is always the case for monostatic SAR. Therefore, we find

$$\Delta cr_g = \frac{\lambda_{max}}{\Delta\theta} \frac{1}{\sqrt{\cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\phi_{TX})\cos(\phi_{RX})\cos(\beta)}} \quad . \tag{3.40}$$

Then in the case where $\phi_{TX} = \phi_{RX} = \phi$, both the transmitter and receiver have the same elevation angle, which is also always the case for monostatic SAR. The cross range resolution becomes Equation (3.41).

$$\Delta cr_g = \frac{\lambda_{max}}{\Delta\theta \cos(\phi)\sqrt{2(1 + \cos(\beta))}} \tag{3.41}$$

Then using the trigonometric identity $2\cos^2 \frac{A}{2} = 1 + \cos(A)$ Equation (3.41) becomes

$$\Delta cr_g = \frac{\lambda_{max}}{2\Delta\theta \cos(\phi) \cos(\frac{\beta}{2})} \quad . \tag{3.42}$$

When $\beta = 0$ this formulation becomes the monostatic formulation for monostatic cross-range resolution found in [9] and [10].

$$\Delta cr_g = \frac{\lambda_{max}}{2\Delta\theta\cos(\phi)} \tag{3.43}$$

Additionally a useful rule of thumb is developed for monostatic and bistatic SAR when the angular extent of the collection is equal for transmit and receive, i.e. $\Delta\theta_{RX} = \Delta\theta_{TX} = \Delta\theta$. In most imaging scenarios it is desired to have equal resolution in both range and cross-range. Equating Equation (3.40) to Equation (3.9) leaves

$$f_{min} = \frac{c}{\lambda_{max}} = \frac{BW}{\Delta\theta} \quad , \tag{3.44}$$

where f_{min} is the lowest instantaneous frequency of the transmitted waveform and generates a useful equation. Equation (3.44) can help easily determine one of the unknown parameters and windowing terms can be factored in if needed.

Chapter 4

SAR Signal Processing

Once the signal is received and motion compensated to a Scene Reference Point (SRP), the signal can be processed using one of the many SAR image formation algorithms. SAR algorithms can be classified into either time or frequency domain algorithms each having different advantages and disadvantages. Time domain image formation techniques are computationally intensive but do not suffer from the k-space approximations, meaning that they are an exact reconstruction of the radar image. Whereas frequency or Fourier based algorithms take advantage of the computational efficiency of the Fast Fourier Transform (FFT). Spatial frequency conventions like k-space approximate the removal of the target's phase due to the aircraft's position but these approximations leave unaccounted higher order phase terms that can distort or defocus the image if left uncorrected. The Fourier slice theorem is a useful theorem when relating these two types of image formation techniques.

4.1 The Polar Format Algorithm a Frequency Domain Approach

The k-spaces developed and discussed in chapters 2 and 3 are the basis for the Polar Format Algorithm (PFA). K-space, also referred to as wavenumber, is spatial Fourier or frequency domain representation of the signal. The polar format algorithm creates an image by transforming the radar data within the k-space domain. The polar format algorithm follows a sequence of steps that are shown in a block diagram in Figure 4.1.

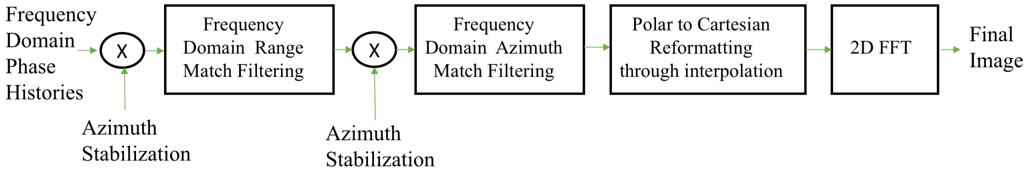
Frequency Domain Phase Histories
X
Azimuth Stabilization
Frequency Domain Range Match Filtering
X
Azimuth Stabilization
Frequency Domain Azimuth Match Filtering
Polar to Cartesian Reformatting through interpolation
2D FFT
Final Image

Figure 4.1: *Polar Formatting Algorithm block diagram.*

K-space domain matched filtering entails multiplying the Fourier transform of the transmitted waveform with the received data in the range dimension. If the receive signal is digitally deramped or deramped in the analog part of the receiver chain, as described in Chapter 2, then the data is in the frequency domain. This deramped receive signal is stretch processed by taking the FFT of the deramped return. The position of the transmitter and receiver, is used to create a matched filter in the cross range dimension and applied to the data as well. The data for the entire Coherent Processing Interval (CPI) moves from the polar to Cartesian reformat through an interpolation step.

The PFA reformats the polar frequency-domain radar returns by interpolating these returns into a Cartesian format. This is where PFA derives it's name. The polar to Cartesian formatting is illustrated in Figure 4.2 [9]. The polar k-space returns within the blue fan shaped envelope are interpolated to the Cartesian k-space in the red grid. This interpolation step enables the final image formation step with a 2D FFT.

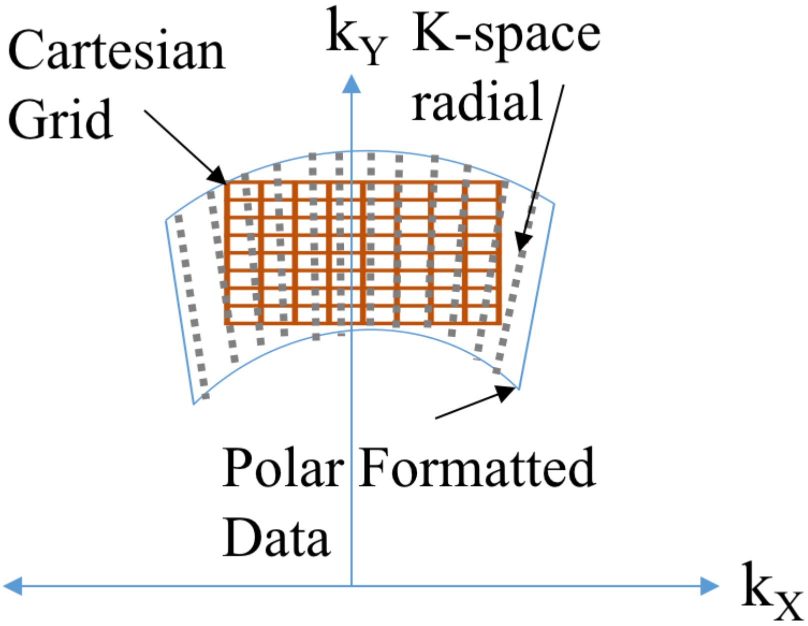

Figure 4.2: *Polar Formatting phase of the Polar Format algorithm.*

4.2 Time Domain Backprojection Algorithm

The Time Domain BackProjection Algorithm (BPA) is a tomographic method found in [3, 9, 35]. This method operates in the time domain by taking the pulse compressed radar returns and interpolating the data. The interpolated data is then summed into a 2D Cartesian imaging grid.

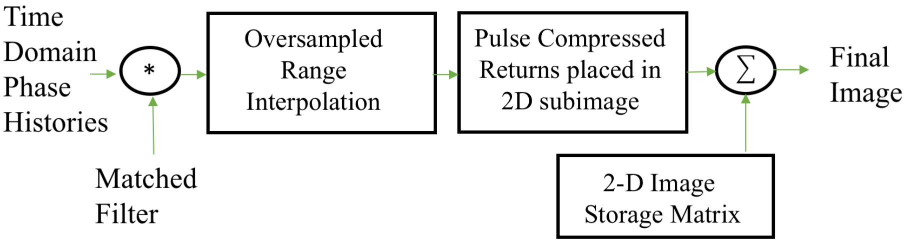
Time Domain Phase Histories
*
Matched Filter
Oversampled Range Interpolation
Pulse Compressed Returns placed in 2D subimage
∑
Final Image
2-D Image Storage Matrix

Figure 4.3: *Time Domain Backprojection block diagram.*

The block diagram in Figure 4.3 shows each of the computationally intensive steps taken to back project the image. In the initial step, the radar returns are matched filtered with the transmit waveform. Then, this pulse compressed data is interpolated in the range dimension. This interpolation is intended to highly oversample the pulse compressed radar return in the range dimension. A sub-image grid is formed with the same dimensions as the final image. Then the range to each pixel within the sub-image is calculated. The appropriate interpolated range bin from the pulse compressed radar return is placed into the sub-image. A range bin is a single element of a range compressed radar return stored in a vector. A range bin will corespond to a specific range value from the radar's Aperture Center Point (ACP).

Curved arcs will form in the sub-image due to multiple pixels in the image having the same or similar ranges to the monostatic or bistatic radar systems' ACPs. These curved arcs are called iso-range contours described in [7, 13, 28, 36]. Figure 4.4 illustrates two sub-images of a scene containing a single point scatterers summed together. Constructive and destructive interference causes the energy of the curved arcs to dissipate and 2D impulses are formed in the location of the scatterers' position in the image plane.

4.3 The Fourier Slice Theorem

Frequency domain analysis of a radar return at the initial stage of Figure 4.3 shows equivalence between time domain algorithms and frequency or k-space domain algorithms using the Fourier Slice Theorem. The Fourier Slice Theorm relates these two types despite their different domains [10, 37]. Both types of algorithms achieve the same goal: SAR image formation.

A radar return is pulse compressed in the time domain using stretch pro-

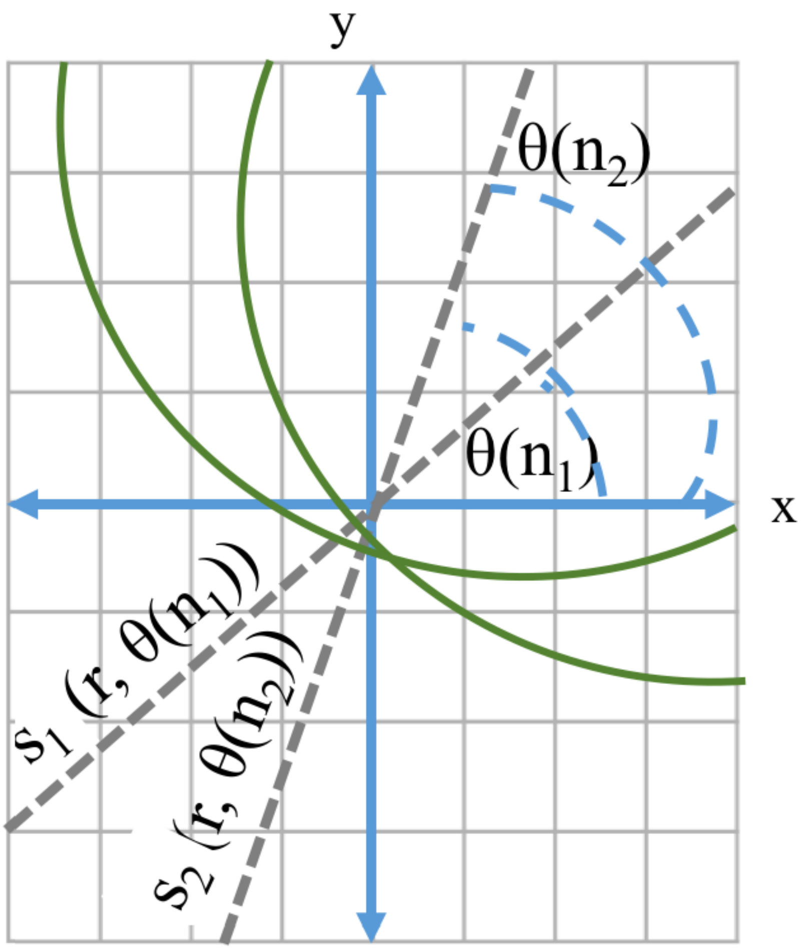

Figure 4.4: *Two radar returns summed into a ground referenced image plane.*

cessing or matched filtering. The BPA algorithm projects the radar return in to the ground plane using it's elevation angle. This will place the radar return in a two-dimensional plane on the ground.

The radar return signal $s(r, \theta)$ where r is the range from the origin of the Cartesian imaging grid and θ is the angle referenced from the x-axis as shown in Figure 4.5. The relationship between r and θ and the Cartesian coordinates are $x = r\cos(\theta)$ and $y = r\sin(\theta)$, where x and y are the Cartesian positions of the radar return signal $s(r, \theta)$. The two previously mentioned polar to

Cartesian conversion are used to transform $s(r, \theta)$ to $s(x, y)$.

In the frequency domain, the terms u and v are the Cartesian frequency locations and analogous to x and y in the time domain. The u and v dimensions are proportional to the k_x and k_y domains in Chapter 2. The two domains are related by this linear relationship $k = \frac{4\pi f}{c}$ from Equation (2.32) where $k_x = \frac{4\pi u}{c}$ and $k_y = \frac{4\pi v}{c}$. The $F\{\bullet\}$ function is the Fourier transform which relates the time domain of the signal to the spatial frequency domain of the signal in

$$S(u, v) = F\{s(x, y)\} = \int\int s(x, y)e^{-j2\pi(xu+yv)}dxdy \quad . \tag{4.1}$$

Similarly, the Cartesian frequency terms u and v are related to the polar frequency terms w and ψ by these two relationships $u = w\cos(\psi)$ and $v = w\sin(\psi)$. Also in the frequency domain, the terms w and ψ are the polar frequency locations and analogous to polar coordinates r and θ in the time domain. These two relationships transform the exponent and differential terms with

$$xu + yv = r\cos(\theta - \psi) \quad , \tag{4.2}$$

and

$$dxdy = rdwd\psi \quad . \tag{4.3}$$

Equation 4.1 now becomes

$$S(w, \psi) = F\{s(r, \theta)\} = \int_0^{2\pi}\int_0^{\infty} s(r, \theta)e^{-j2\pi(r\cos(\theta-\psi))}rdwd\psi \quad . \tag{4.4}$$

The representation of a single radar return is now in the form of $s(r, \theta +$

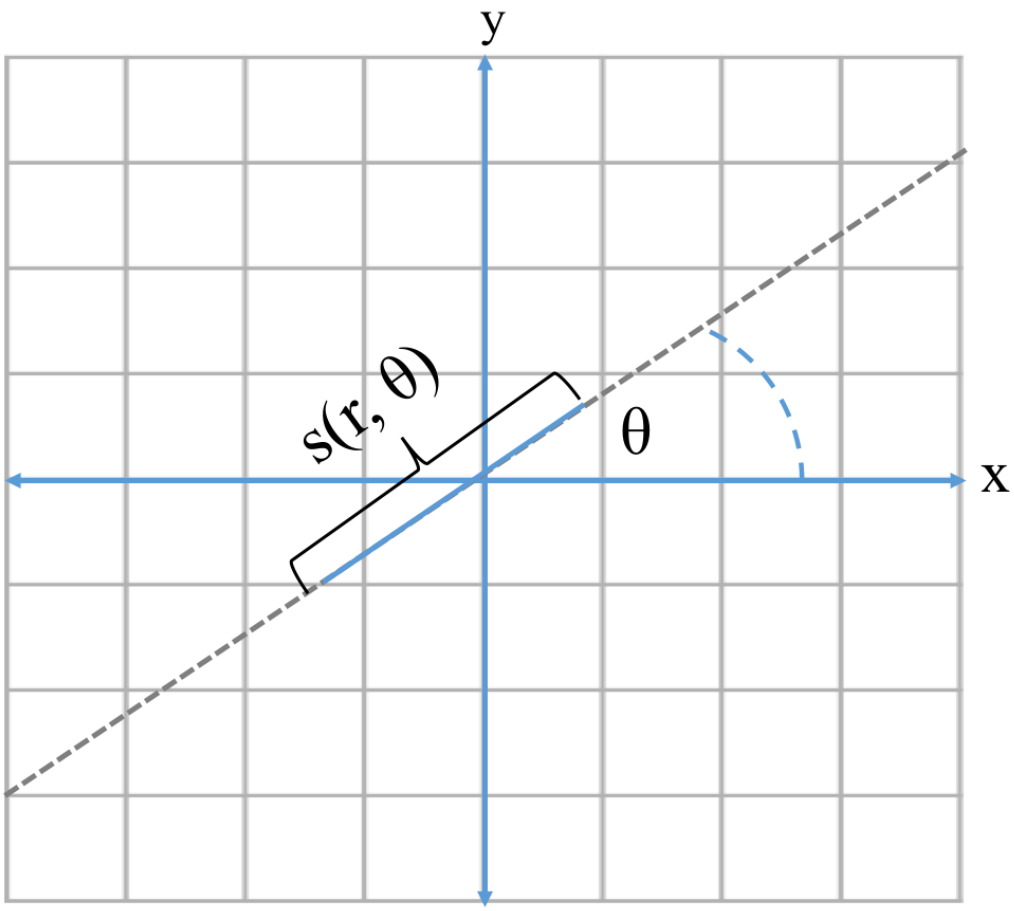

Figure 4.5: *A single radar returns projected into a ground referenced image plane.*

$\theta(n))$, where the θ term is an angular constant. The term $\theta(n)$ is the angular change for each PRI within the CPI. The term n represents the discrete indexing for each PRI within the CPI that contains N_p total PRIs.

Using the polar Fourier transform found in Equation (4.4), a single radar return is transformed into the frequency domain in

$$F\{s(r,\theta+\theta(n))\} = \int_0^{2\pi}\int_0^{\infty} s(r,\theta+\theta(n))e^{-j2\pi(r\cos(\theta-\psi))}rdwd\psi \quad . \tag{4.5}$$

Since the integral is from 0 to 2π the $\theta(n)$ term is transferred to the exponential term as an offset in

$$F\{s(r, \theta + \theta(n))\} = \int_0^{2\pi} \int_0^{\infty} s(r, \theta) e^{-j2\pi(r\cos(\theta - \theta(n) - \psi))} r dw d\phi \quad . \tag{4.6}$$

The even property of the cosine function, $cos(\theta) = cos(-\theta)$, will transform the cosine term in the exponent of Equation 4.6. Then a substitution of variables where, $\phi = \psi + \theta(n)$ and $d\phi = d\psi$, the integral becomes Equation 4.7.

$$F\{s(r, \theta + \theta(n))\} = \int_0^{2\pi} \int_0^{\infty} s(r, \phi) e^{-j2\pi(r\cos(\phi - \theta))} r dw d\psi \tag{4.7}$$

Then using the relationship defined in Equation (4.4) the Fourier transform of the single return is

$$F\{s(r, \theta + \theta(n))\} = \int_0^{2\pi} \int_0^{\infty} s(r, \phi) e^{-j2\pi(r\cos(\phi - \theta))} r dw d\psi = S(w, \phi) \quad . \tag{4.8}$$

Then substituting for ϕ using this relationship $\phi = \psi + \theta(n)$, the Fourier Slice Theorem is defined as

$$F\{s(r, \theta + \theta(n))\} = S(w, \psi + \theta(n)) \quad . \tag{4.9}$$

This relationship defined in Equation 4.9 implies that the same angular in the time domain also occurs in the frequency domain. Applying this derived relationship to the definition of k-space, $k = \frac{4\pi f}{c}$ from Equation (2.32), transforms the Fourier transform of a single radar return into the polar k-space as

$$k = \frac{4\pi}{c} F\{s(r, \theta + \theta(n))\} = \frac{4\pi}{c} S(w, \psi + \theta(n)) \quad . \tag{4.10}$$

Equation (4.10) relates a single radar return that is summed into an imaging plane for BPA into k-space used by the PFA. Figure 4.6 illustrates this relationship in a two-dimensional plane.

Both algorithm types, time and frequency domain, will use interpolation to reformat the polar formatted data into a Cartesian space. The Fourier slice theorem is illustrated in Equation 4.10 shows the relationship of a single radar return and it's angular position within the k-space or frequency and the time domain are the same. This relationship will help to illustrate the relationship between the two image processing paradigms, time-domain and frequency domain.

4.4 Fast Back Projection Algorithms: Prior Art

Fast backprojection algorithms have been proposed in [**?**, 38–42]. One of the earliest papers on Fast Back-Projection (FBP) algorithm is [38] published in 1981. This paper develops FBP algorithms for computed tomography (CT) scans used in medical imaging. This paper describes an FBP algorithm where the one-dimensional CT signal that is back projected is oversampled and nearest neighbor interpolation to sum the value into the image. The algorithm is written in machine code and presented at the end of the algorithm. This implementation of a FBP algorithm is very similar to what is called just called the back projection algorithm in this document.

Fast Factorized BPA algorithms have been proposed by [5,39,40] where the

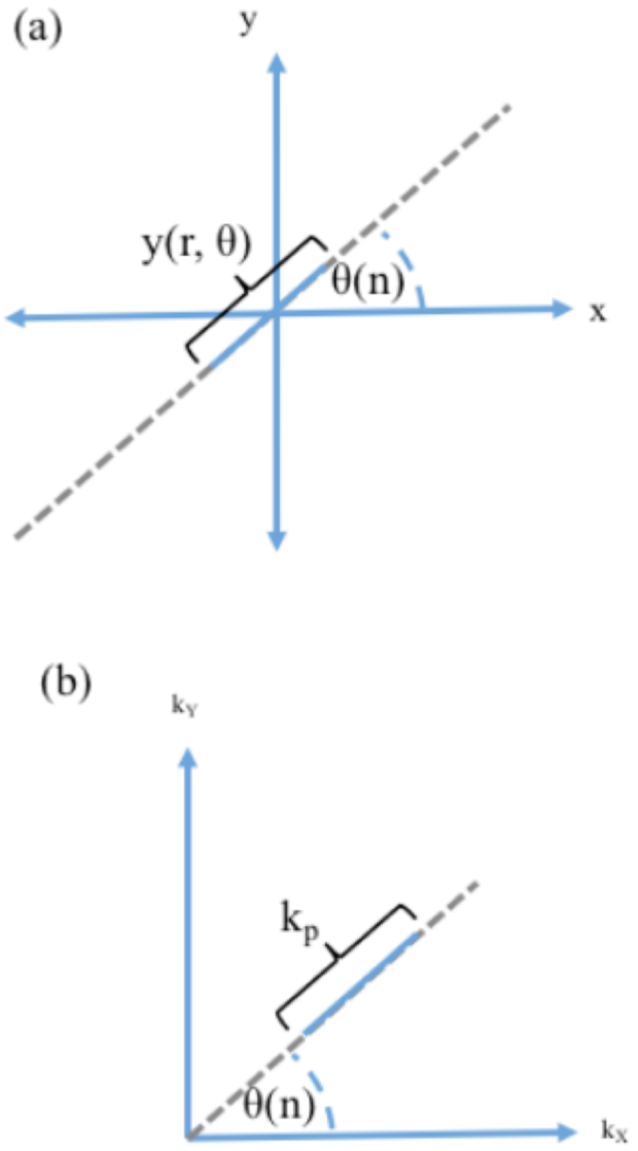

Figure 4.6: *a) a single range return located within an imaging grid b) the same return represented in k-space*

radar returns are broken into sub apertures and can be processed in parallel and then combined to form the desired image. In [41], a decimation in image approach is proposed where the phase histories are split into different sections and undersampled images are formed. This algorithm is the most similar to the fast backprojection algorithm described in this dissertation. In both [39, 42], each propose a back projection method where the radar returns are placed in a polar grid and then back projected.

4.5 Application of the Fourier Slice Theorem

The the Fast Factorized Back Projection Algorithm (FFBPA) described in [5] is a novel adaptation to the back projection algorithm which speeds up the processing using a time-domain interpolation and combination of neighboring radar returns. The trade-off to this approach is understood from the frequency domain. The combination of neighboring radar returns preserves the SNR but decimates the number of samples in the cross-range dimension of the signal space. Due to the cross-range decimation, the number of neighboring radar returns summed together increases and the image folds over in the cross-range due to aliasing. This is true for the FFBPA in [5] and the Fast Decimated Wavenumber Back Projection Algorithm (FDWBPA) within this document. When stretch processing, after analog deramping, the native format of stretch processed SAR data from the analog-to-digital (A/D) converter is polar formatted and within the wavenumber domain. Equations (2.30) and (2.37) can be combined to form an IF signal received in this domain as

$$s_{IF}(n,t) = a_t rect(\frac{\hat{t} - \frac{2R_t(n)}{c}}{T_p})e^{-j\boldsymbol{k}(n,\hat{t})\bullet\boldsymbol{r_t}(n)} \quad . \tag{4.11}$$

Now, a single radar return signal is represented by $s_{IF}(n,t)$. It is important to realize that t affects two aspects, the rectangular window and the wavenumber or k-space the signal occupies in the range dimension. This ultimately depends on the transmit bandwidth and center frequency. The term n varies the positional aspects of the signal, $R_o(n)$, $\theta(n)$, and $\phi(n)$. It's assumed that $R_o(n)$ does not change too significantly for neighboring radar returns, which means $R_o(n)$ is $n \pm m$ where $m = 0, 1, 2$ or 3, for this algorithm, but could be larger or smaller depending on the data collection conditions.

A good rule of thumb for acceptable range error of a SAR signal, is an eighth of a wavelength [9, pp. 320]. For the data set used in this paper, the maximum difference between $R_o(n)$ and $R_o(n+5)$ was .088 wavelengths. This error should be accounted for in a phase error budget as well [9, pp. 220-243]. This error could also be mitigated using digital or analog techniques, but none of the techniques are discussed or implemented in this document.

Using the previously stated assumption, we are now left with a spherically formatted signal model $s_{IF}(k_r, \theta, \phi)$. This signal is recorded as a time domain signal but these samples exist in the k-space domain. This is an important aspect to this algorithm because modern radar systems' A/D converts sample at linear rates, but it's important to note that other SAR radar systems A/D converts have sampled at non-linear rates to motion compensate prior to processing reducing a SAR algorithm's computational complexity [43]. By combining these linearly sampled radar returns in the time-sampled k-space domain, the FDWBPA similarly reduces the computational complexity by removing the interpolation step in the Fourier transformed time domain of the FFBPA but more significantly, it reduces the number of BPA iterations.

The Fourier Slice Theorem (FST) [44, 45] is

$$R_{\theta,\phi}\{f(x,y,z)\} \underset{\rightarrow}{F} R_{\theta,\phi}\{F(u,v,w)\} \quad , \tag{4.12}$$

where $R_{\theta,\phi}\{\cdot\}$ is the rotational operator [44, 45] and $\underrightarrow{F}$ is the Fourier transform. Equation (2.38) is a one-dimensional signal when $R_o(n)$ is assumed to not change and Equation (2.38) can be mapped to $f(x,0,0)$ while the angular components, θ and ϕ, are mapped to the rotational operator. The range domain would be mapped to $F(u,0,0)$ using the relation of the Fourier transform of a stretch processed return [18]. This application of the of the FST generates the relationship

$$s_{IF}(k_r,\theta,\phi)\underrightarrow{F}S_{IF}(r,\theta,\phi) \quad . \tag{4.13}$$

Applying this relationship to neighboring radar returns, these radar returns in the k-space domain have the same 3 dimensional relationship that they do in the range domain with respect to their azimuth and elevation angles. Neighboring radar returns are sampled in the same relative k_r wavenumbers while θ and ϕ change slightly or not at all. Generally speaking, these radar returns are efficiently sampled in the k_r dimension at the rate needed to capture the received signal. This means that neighboring samples are efficiently sampled and aligned. Figure 4.7 depicts a group of neighboring radar returns and their alignment. Many BPAs use nearest neighbor interpolation which relies on over sampling the signal using an FFT for pulse compression. Fewer samples exist prior to the FFT in the wavenumber domain compared to combining radar returns in an over sampled range domain.

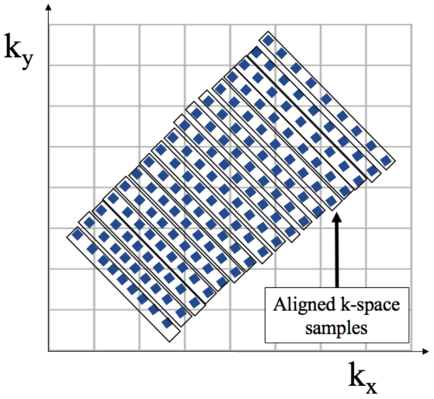

Figure 4.7: *K-space representations of the signal illustrating how neighboring k-space samples are in alignment with each other.*

4.6 Fast Decimated Wavenumber Back Projection Algorithm

Using the properties of the Fourier Slice theorem, a FBP algorithm is developed in the spatial-frequency domain. This preprocessing method occurs prior to the range FFT, which is used to pulse compress and possibly over sample the data, prior to the back-projection algorithm as well. The block diagram in Figure 4.8 shows where the decimated wavenumber processing exists in the signal processing chain of the back-projection algorithm.

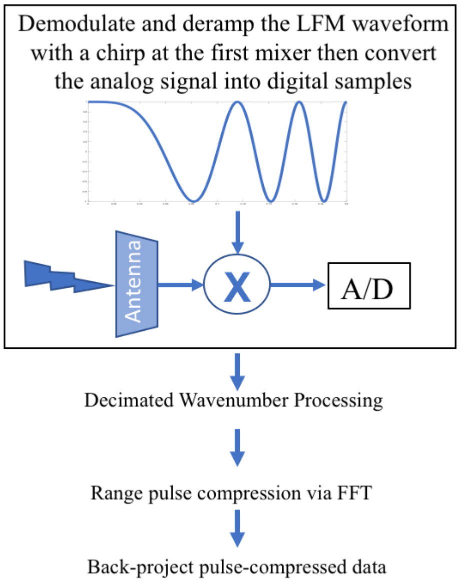
Demodulate and deramp the LFM waveform with a chirp at the first mixer then convert the analog signal into digital samples
Antenna
X
A/D
Decimated Wavenumber Processing
Range pulse compression via FFT
Back-project pulse-compressed data

Figure 4.8: *Flow diagram showing where the decimated wavenumber processing takes place.*

The FBP algorithm described in [5] interpolates pulse compressed range returns to a central point in the range domain, and may be implemented in the back projection stage of this algorithm. In the Fast Decimated Wavenumber BPA (FDWBPA) algorithm, in the decimated wavenumber processing step within Figure 4.8, adjacent wavenumber samples, meaning adjacent deramped phase histories, are averaged together along with the positions of the platform for each of those wavenumbers as shown in Figure 4.9. Within Figure 4.9 the term Decimation Factor, DF, is introduced. Simply stated, DF is the number of radar returns and their positions which are averaged together within this algorithm. These averages will then form a single radar return called a subaperture that is then pulse compressed with an FFT and then back projected to a 2D Cartesian image. After the samples have been combined into a subaperture, the number of FFTs needed to pulse compress the data are reduced by DF, the Decimation Factor as well.

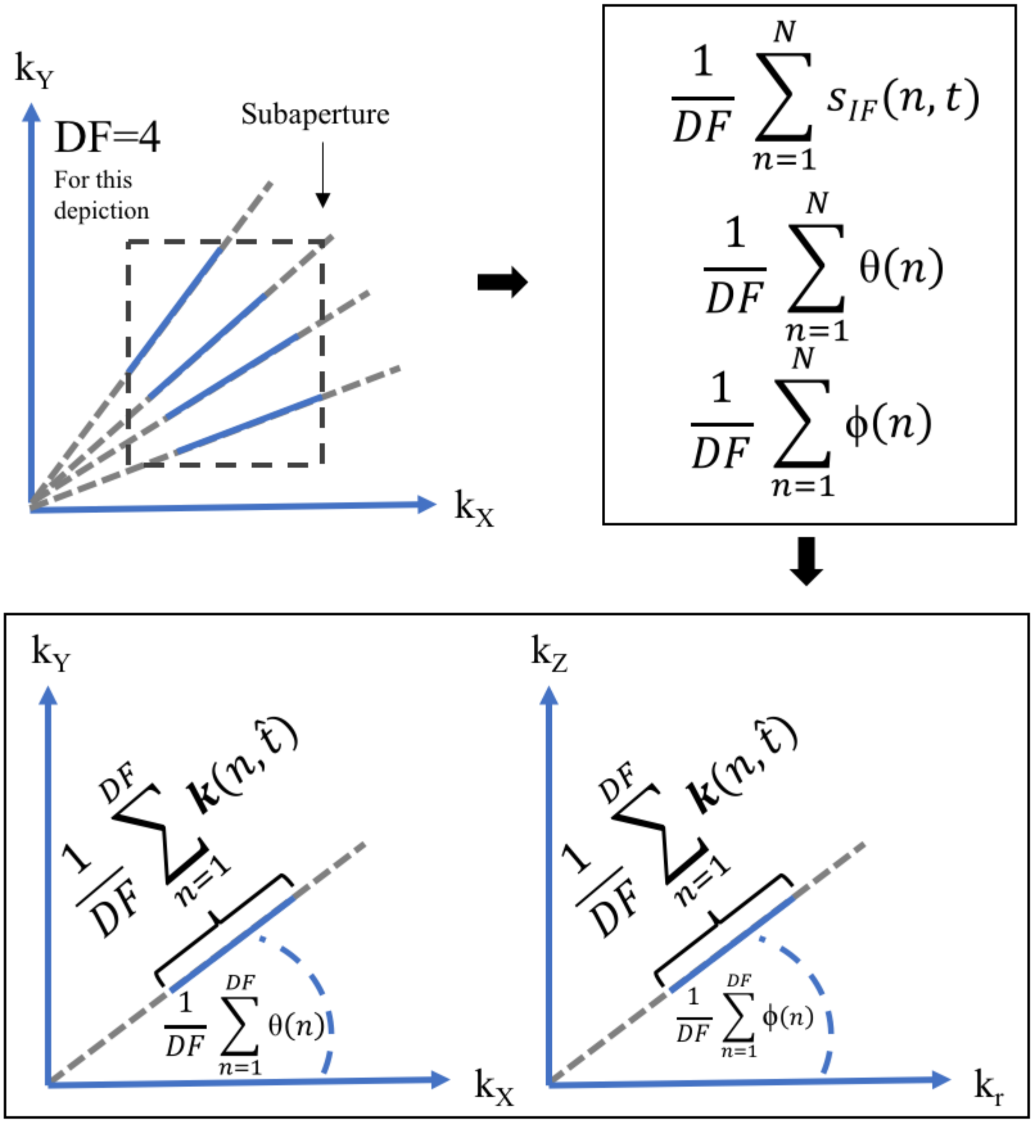
k_Y
DF=4
For this depiction
Subaperture
k_X
$\frac{1}{DF}\sum_{n=1}^{N} s_{IF}(n,t)$
$\frac{1}{DF}\sum_{n=1}^{N} \theta(n)$
$\frac{1}{DF}\sum_{n=1}^{N} \phi(n)$
k_Y
$\frac{1}{DF}\sum_{n=1}^{DF} \boldsymbol{k}(n,\hat{t})$
$\frac{1}{DF}\sum_{n=1}^{DF} \theta(n)$
k_X
k_Z
$\frac{1}{DF}\sum_{n=1}^{DF} \boldsymbol{k}(n,\hat{t})$
$\frac{1}{DF}\sum_{n=1}^{DF} \phi(n)$
k_r

Figure 4.9: *Depiction of the algorithm in k-space.*

The advantage to this approach is that only one FFT is needed to pulse compress each subaperture. The pulse compressed subaperture is converted from its polar format to a Cartesian format within the back-projection algorithm. This approach operates on the data in its native format if the receive signal deramped for strectch processing. If the receive signals were matched filtered, then each PRI would need to be transformed via an IFFT (Inverse Fast Fourier Transform), and then averaged together. In addition, the average would be transformed back to the range domain using an FFT. There could be instances for a specific radar configuration when matched filtered data was transformed using an IFFT for other reasons; and in this case, this approach may be useful.

Chapter 5

SAR Simulation

This chapter discusses the signal model used to create the radar returns from multiple point targets using the signal model developed in Chapter 2. The Matlab code used to generate this model is found in Appendix A and specific Matlab files for the model are referenced throughout this chapter in this form: `filename.m`.

5.1 The Bistatic SAR Signal Model

The bistatic SAR signal model can be found between lines 15 and 37 of `create_bistatic_Phase_History.m`. Initially a for loop loops through all point targets within the scene and sums a radar returned signal from each point target into a single received signal,

$$s_{IF}(n,t) = N(0,\sqrt{P_N}) + jN(0,\sqrt{P_N}) + \sum_{i=1}^{M} s_{IF}(n,t,m) \quad . \tag{5.1}$$

The term M is the integer number of point targets. The term $N(\mu,\sigma)$ is Normal or Gaussian noise with mean μ and standard deviation σ. The term P_N is the noise power of the system [24, 46] where,

$$P_N = kT_0 f_s F_N G_s \quad . \tag{5.2}$$

The system noise power was calculated using Equation (5.2). The terms k, T_0, F_N and G_s are Boltzmann's constant, the system temperature in Kelvins, the system noise figure and the system gain, respectively. The term, f_s, was used instead of the IF bandwidth to account for oversampling and cases where stretch processing was used. The noise power defined in Equation (5.2) is the standard deviation used in the Gaussian noise which was added to both the in-phase and quadrature channels in (5.3). The receiver's system noise was modeled as zero-mean Guassian noise on the I and Q channels in Equations (5.3).

Using the bistatic signal model developed in Section 2, the received SAR signal model was defined by Equation (2.26) as

$$s_{IF}(n,t,m) = a_t(m) rect(\frac{\hat{t}-t_d}{T_p}) e^{j(\Delta\phi - 2\pi(f_c+\gamma(\hat{t}-t_o))(t_d-t_o)+\pi\gamma(t_d-to)^2)} \quad . \tag{5.3}$$

In Equation (5.3), the IF receive signal model, f_c, γ, T_p represents the center frequency, chirp rate and pulse width of the transmit waveform respectively. The term a_t is the square root of the bistatic Radar Cross Section, RCS, of the point target which is set to one for a simple point target model and modeled in the function `targetRCS.m`.

The $\Delta\phi$ term in this equation is the difference between the initial phases of the transmitter and the receiver's LO, such that $\Delta\phi = \phi_t - \phi_r$ and $\Delta\phi$ is set to zero for this model. The t and r subscripts represent the transmitter and receiver, respectively.

The term $\hat{t}$ is the time referenced to the start of the Pulse Repetition Interval (PRI), T, such that $\hat{t} = t - nT$. At $t = 0$, the leading edge of the first transmit pulse of the pulse train leaves the antenna. The term n is the discrete index for each pulse, where $n = 0$ is the first PRI. There are N_p PRIs in the Coherent Pulse Interval (CPI) that forms the synthetic aperture. The $rect()$ function is defined by Equation (2.23). Within the signal model in Equation (5.3), the Residual Video Phase (RVP) is modeled. The RVP was discussed in depth in Section 2.8.1.

5.2 Receiver and Transmitter geometry

The receive signal model was developed using the bistatic time delays found in Section 2 [9, 13]. For a bistatic radar system, the term t_d is defined by $t_d(n) = \frac{R_{t_{TX}}(n)+R_{t_{RX}}(n)}{c}$ where $R_{t_{TX}}(n)$ and $R_{t_{RX}}(n)$ are the ranges from the transmitter and receiver, respectively, to the the point target. The term t_o is defined by $t_o(n) = \frac{R_{o_{TX}}(n)+R_{o_{RX}}(n)}{c}$ where $R_{o_{TX}}(n)$ and $R_{o_{RX}}(n)$ are the ranges from the transmitter and receiver, respectively, to the Scene Reference Point (SRP). Figure 5.1 depicts the ranges previously described between an arbitrary point target and the SRP and the two bistatic systems.

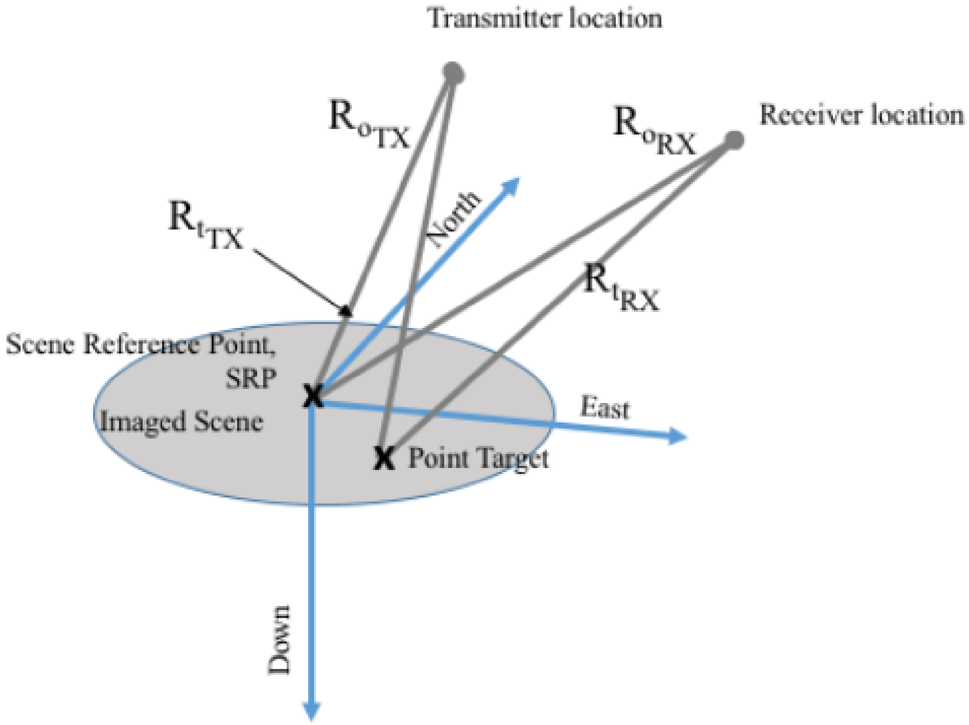

Figure 5.1: *Bistatic Imaging Geometry of the radar, point target and scene in NED.*

The positions of the point targets, transmitter and receiver positions were determined within the following functions, `Plat_Pos_L_Head_Center.m`, `Plat_Pos_Circles.m`, `Plat_Pos_Forward_Scatter.m`, `Plat_Pos_Forward_Scatter_exact.m`, `Plat_Pos_Forward_Scatter_changing_Angs.m`, `Plat_Pos_L_Head_Center.m`, `OU_target.m`, `three_targets.m`, `two_targets.m`, and `one_target.m`. These positions are referenced to a North East Down (NED) frame illustrated in Figure 5.1. In this simulation, each platform traversed a straight and level flight path parallel to the North-East ground plane shown in Figure 5.2. The transmitter traversed at an altitude of 5 km i.e. -5 km down and the receiver had an altitude of 10 km i.e. -10 km down. A monostatic receive signal was easily created by setting the positions of the receiver equal to the transmitter's location in the function `Make_mono.m`.

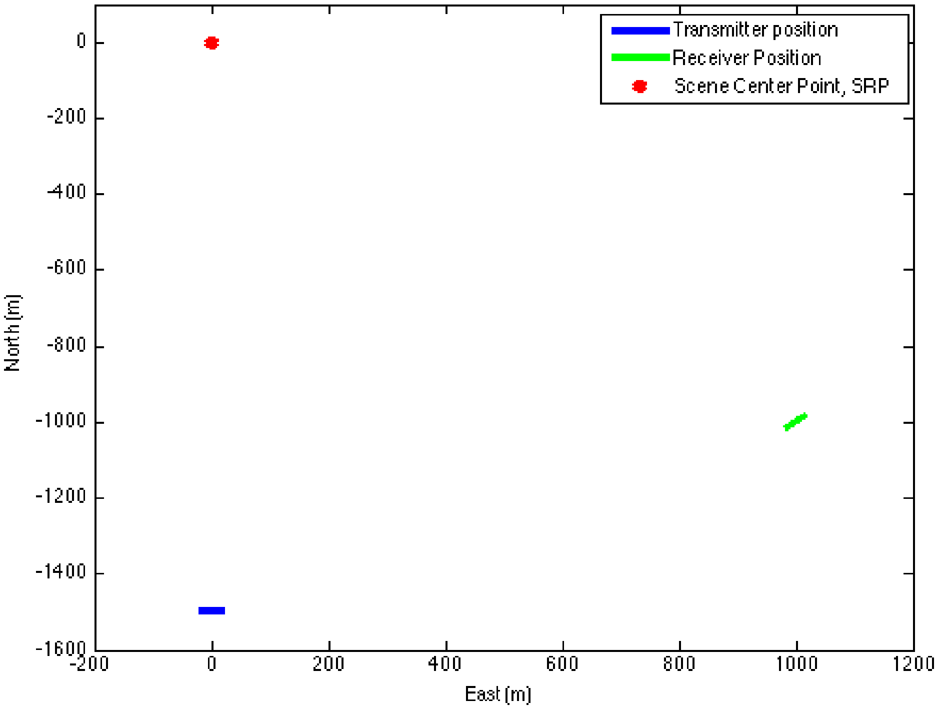

Figure 5.2: *Flight Path Trajectories of the Transmitter and Receiver.*

5.3 Targets

The function `three_targets.m` generates a target model with three point targets each in the North-East plane in the configuration shown in Figure 5.3. Once the simulation was run, the phase histories for each target were created and pulse compressed in the range domain using an Inverse Fast Fourier

Transform (IFFT) as discussed in Section 2.4.

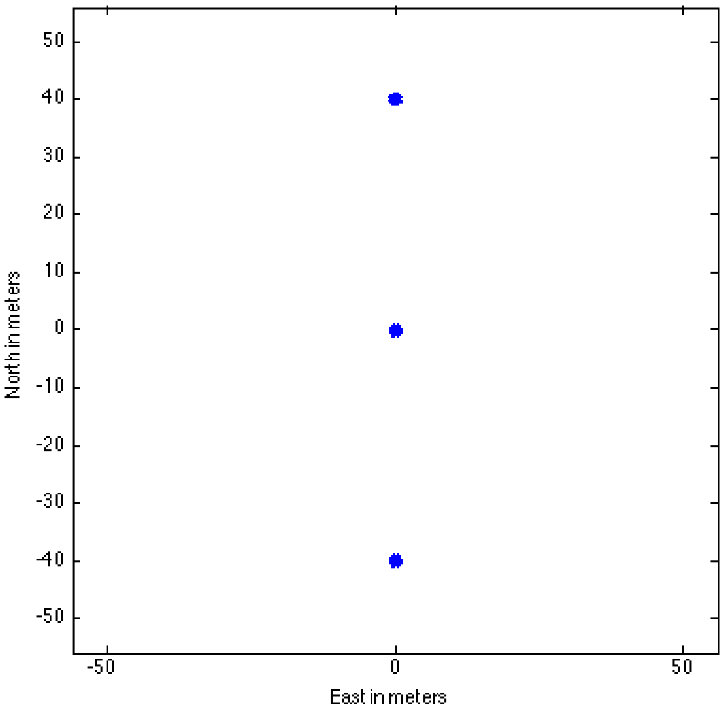

Figure 5.3: *Three point targets in the North-East plane depicted with a blue 'x'.*

The magnitude in dB of range compressed signal models for monostatic and bistatic simulations are shown in Figures 5.7 and 5.8.

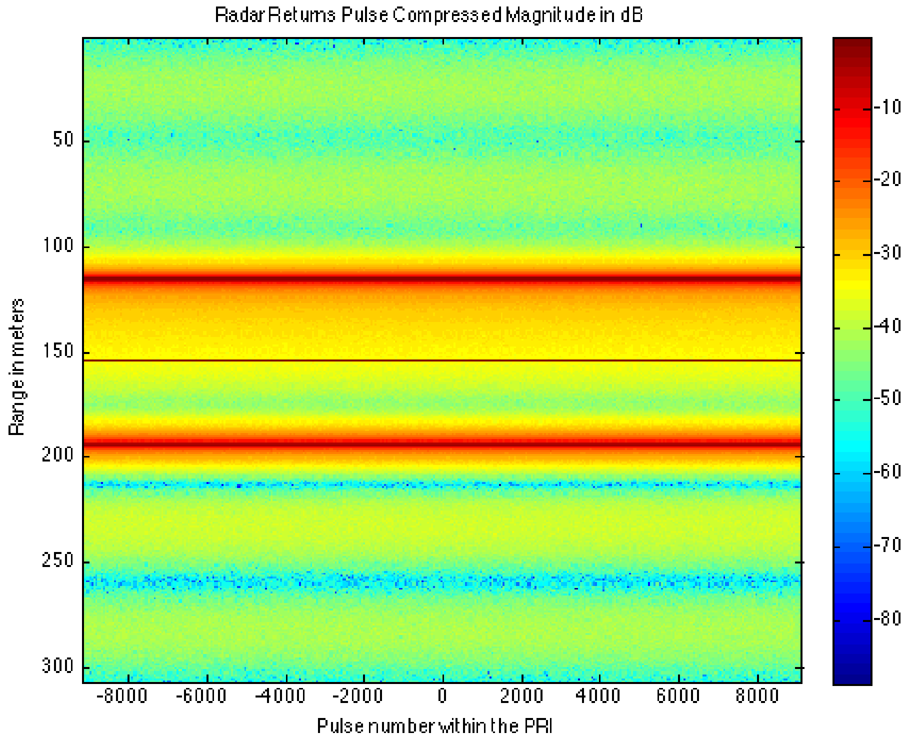

Figure 5.4: *The simulated monostatic signal model containing three point targets which are pulse compressed in range.*

The bistatic angle between the SRP, transmitter and receiver was $\beta = 22°$.

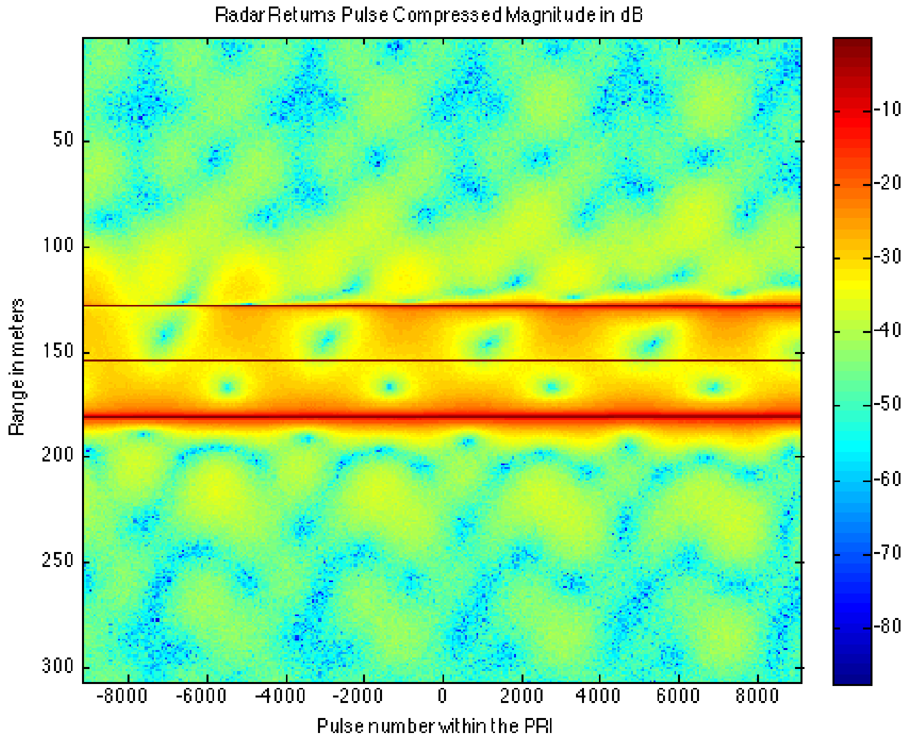

Figure 5.5: *A simulated bistatic receive signal pulse compressed in range consisting of three point targets with a bistatic angle, $\beta = 22°$.*

The OU target, generated in `OU_target.m`, contains point targets configured in the shape of the OU logo. All of the point targets are in the North-East plane which means their Down height is set to zero. Figure 5.6 shows the position of each point scatterer depicted with a blue 'x'.

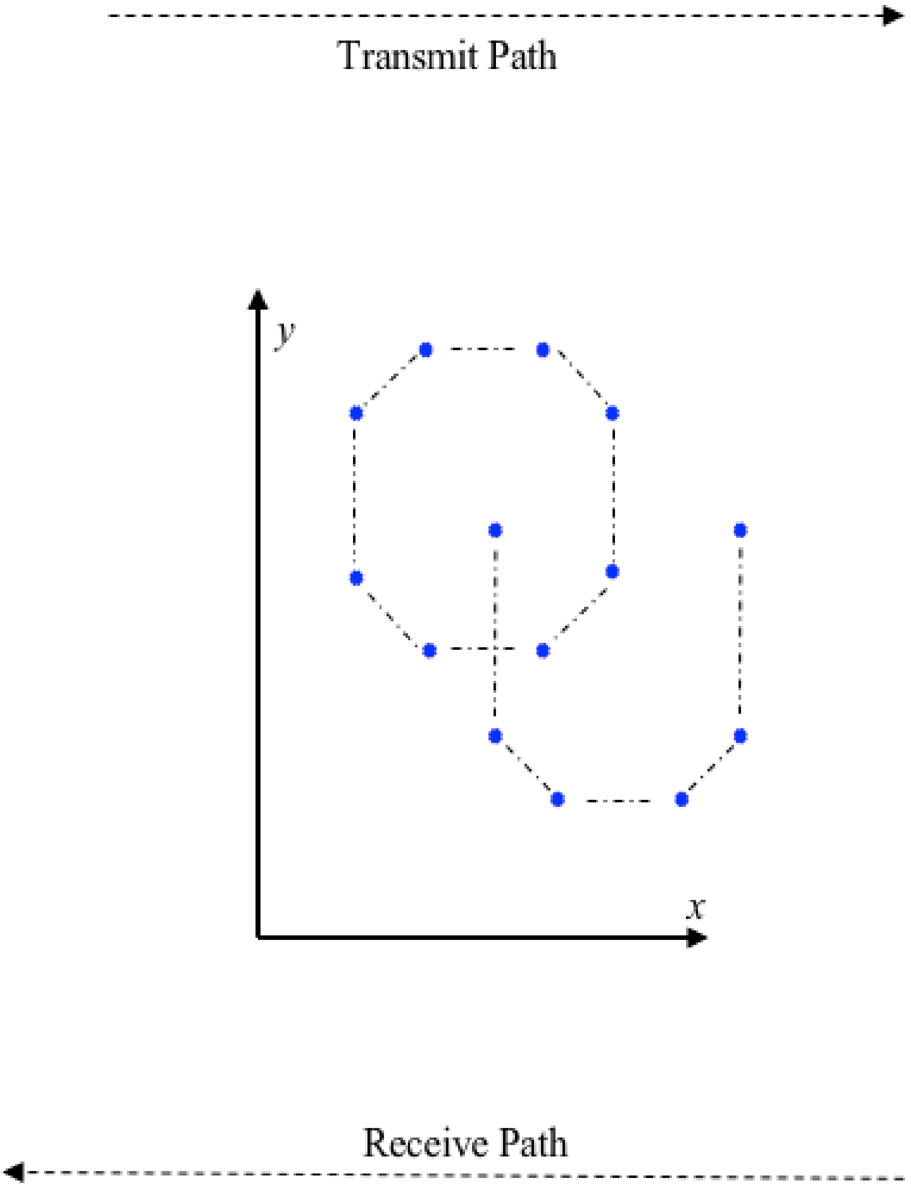

Figure 5.6: *Simulated Flight Paths of the transmitter and receiver. The dashed lines connect the dots to illustrate our university logo, OU.*

Again, the magnitude in dB of range compressed signal models for monostatic and bistatic simulations are shown in Figures 5.4 and 5.5.

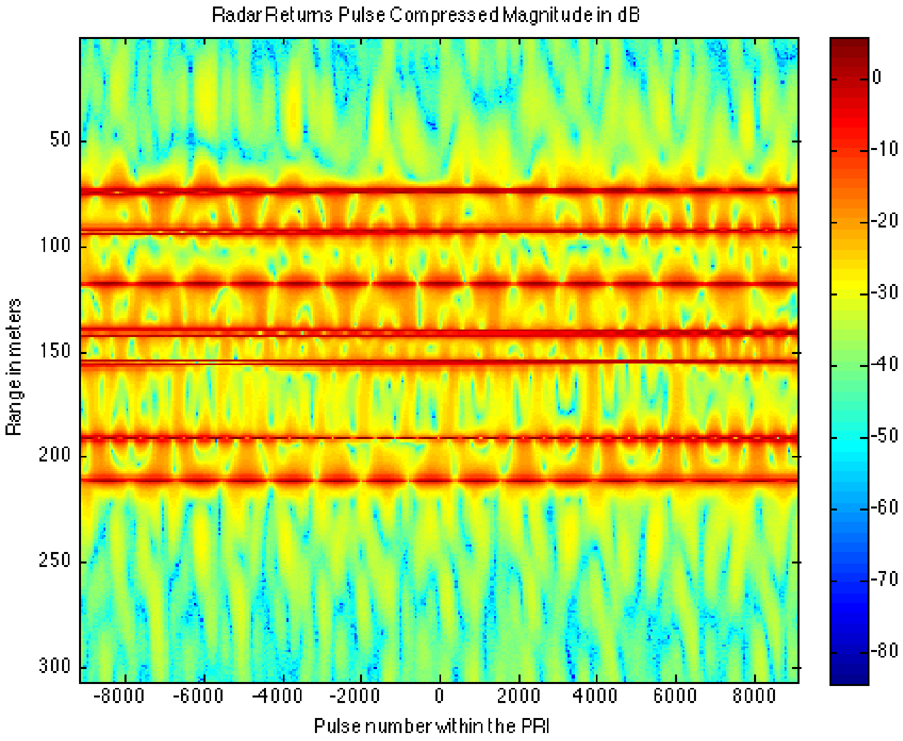

Figure 5.7: *The simulated monostatic signal model using the OU point target and pulse compressed in range.*

Again, the bistatic angle between the SRP, transmitter and receiver was $\beta = 22°$ for the bistatic OU target models range compressed phased histories as well.

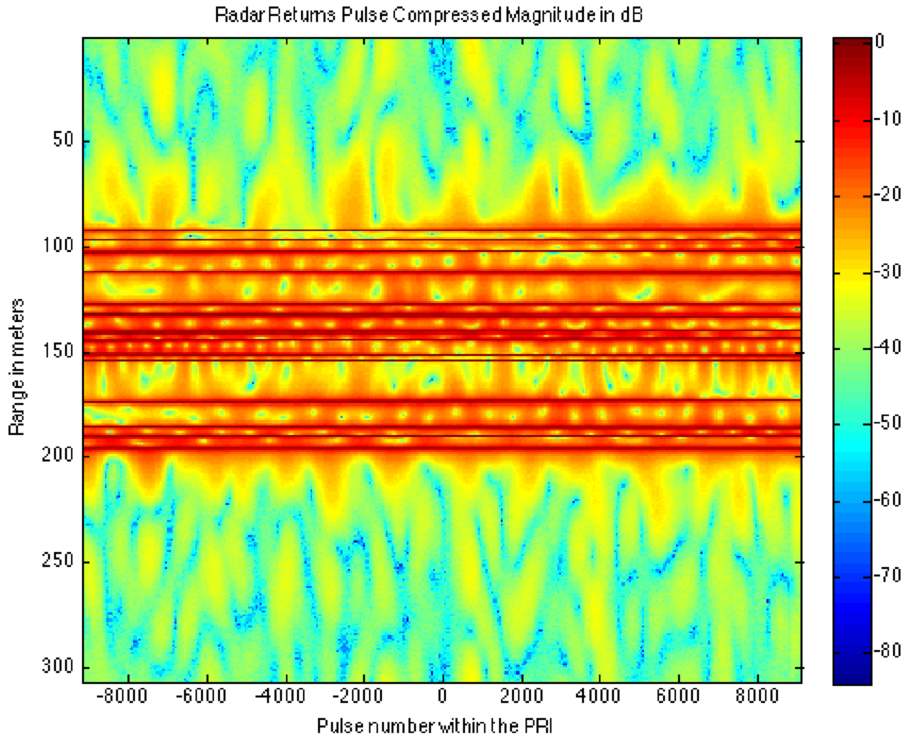

Figure 5.8: *The simulated bistatic signal model using the OU point target and pulse compressed in range with a bistatic angle, $\beta = 22°$.*

5.4 Bistatic k-space

The Polar Format Algorithm (PFA) uses a k-space representation of the signal to reformat the polar radar returns into a Cartesian frequency space for processing SAR imagery using IFFTs. The PFA algorithm has been developed for Bistatic SAR in these references as well [6, 13]. The bistatic k-space developed in Equations (2.57) to (2.60) is shown below in it's Cartesian vector representation in

$$\boldsymbol{k}(n,\hat{t}) = [k_x(n,\hat{t}),\ k_y(n,\hat{t}),\ k_z(n,\hat{t})] \quad , \tag{5.4}$$

$$k_x(n,\hat{t}) = k_r(n,\hat{t})\frac{(\cos(\theta_{TX})\cos(\phi_{TX}) + cos(\theta_{RX})\cos(\phi_{RX}))}{2} \quad , \tag{5.5}$$

$$k_y(n,\hat{t}) = k_r(n,\hat{t})\frac{(\sin(\theta_{TX})\cos(\phi_{TX}) + \sin(\theta_{RX})\cos(\phi_{RX}))}{2} \quad , \tag{5.6}$$

and

$$k_z(n,\hat{t}) = k_r(n,\hat{t})\frac{(\sin(\phi_{TX}) + \sin(\phi_{RX}))}{2} \quad . \tag{5.7}$$

In Section 4, using the Fourier slice theorem, a single radar return was placed within the k-space using its frequency extent and angular position. In Equation 5.4, the angular representation is shown in each of the Cartesian terms multiplied by the $k_r(n,\hat{t})$ function. The $k_r(n,\hat{t})$ function is shown below in Equation (5.8) which was developed in Equation (2.55), as

$$k_r(n,\hat{t}) = k_c\Big(1 + \frac{\gamma}{f_c}\Big(\hat{t} - \frac{R_{o_{TX}}(n) + R_{o_{RX}}(n)}{c}\Big)\Big) \quad . \tag{5.8}$$

The terms k_c and λ_c represent the wavenumber and center wavelength of the transmitted waveform. Where k is defined as

$$k = \frac{4\pi f}{c} = \frac{4\pi}{\lambda} \quad . \tag{5.9}$$

From these bistatic k-space equations, the PFA was developed [6, 20].

5.5 Generating the Bistatic K-space

The simulation calculates the angles θ_{TX}, ϕ_{TX}, θ_{RX}, and ϕ_{RX}, of the bistatic platforms. Each of these angles are dependent on the PRI index n that varies over the CPI. These values can change as the platform moves. These platform angles are illustrated in Figures 2.1 and 2.2. Function `create_platform_angles.m` generates these angles for each of the platform positions.

Once the platform angles are generated, the k-space was created according to Equations (5.4) to (5.7). This takes place on lines 3 through 9 of `gen_Kspace.m`. Since the bistatic k-space's angular location is determined by the average of the transmitter's and receiver's angular location, the bistatic angle, β can be determined by the center of the k-space manifold.

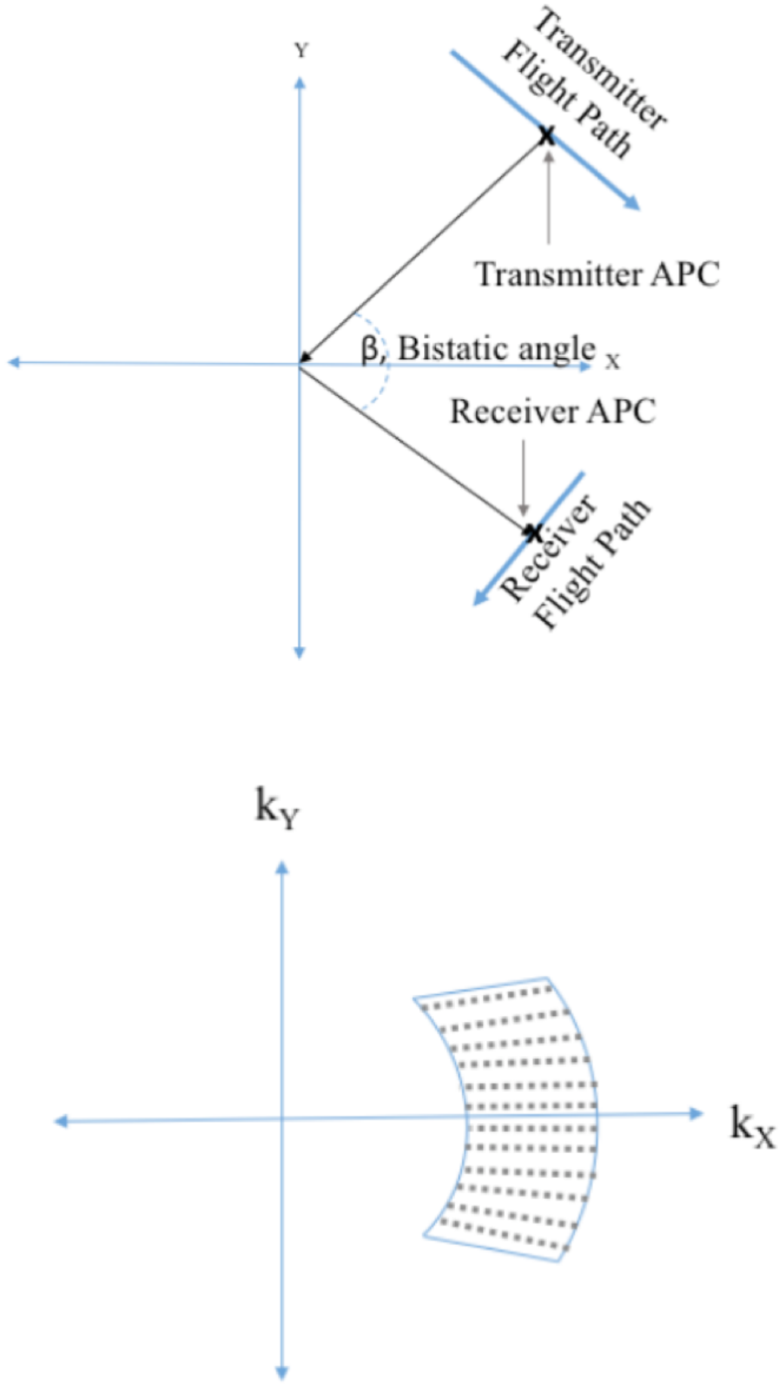

Figure 5.9: *Bistatic angle determined by the transmitter's and receivers ACP*

Once this bistatic angle, β, is determined, the k-space is rotated such that the the bistatic k-space is aligned with the k_x and k_y coordinates. This is done on lines 18 and 19 of `gen_Kspace.m`. This aligns the k-space so that the maximum amount of bandwidth in the range and cross-range is available to process the image. The rotation is preformed using

$$k'_x = k_x \cos(\beta) + k_y \sin(\beta) \quad , \tag{5.10}$$

and

$$k'_y = -k_x \sin(\beta) + k_y \cos(\beta) \quad , \tag{5.11}$$

found in [6] and [13] and depicted in Figure 5.10.

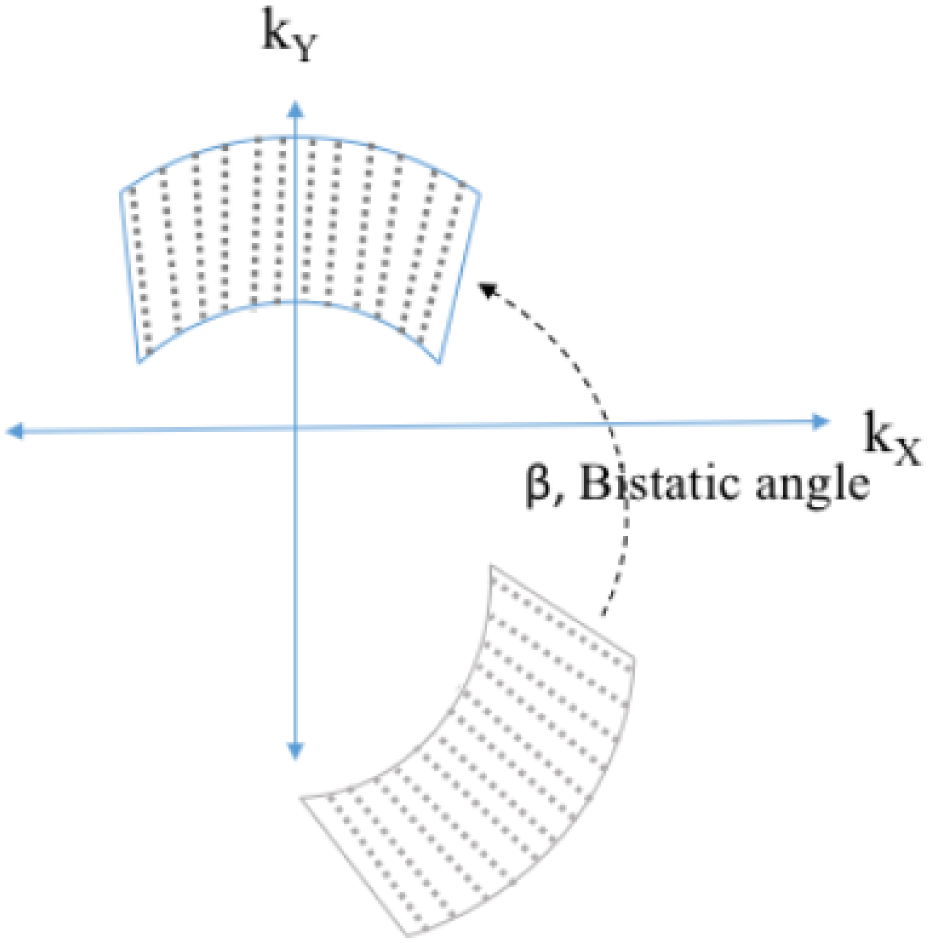

Figure 5.10: *Bistatic angle rotating the k-space into a suitable Cartesian frame.*

Once the k-space is rotated the slope of the k-space's edge is found along with the ground plane bandwidths, B_x and B_y, in the k_x and k_y dimensions. This is shown using the simulated k-space in Figure 5.11.

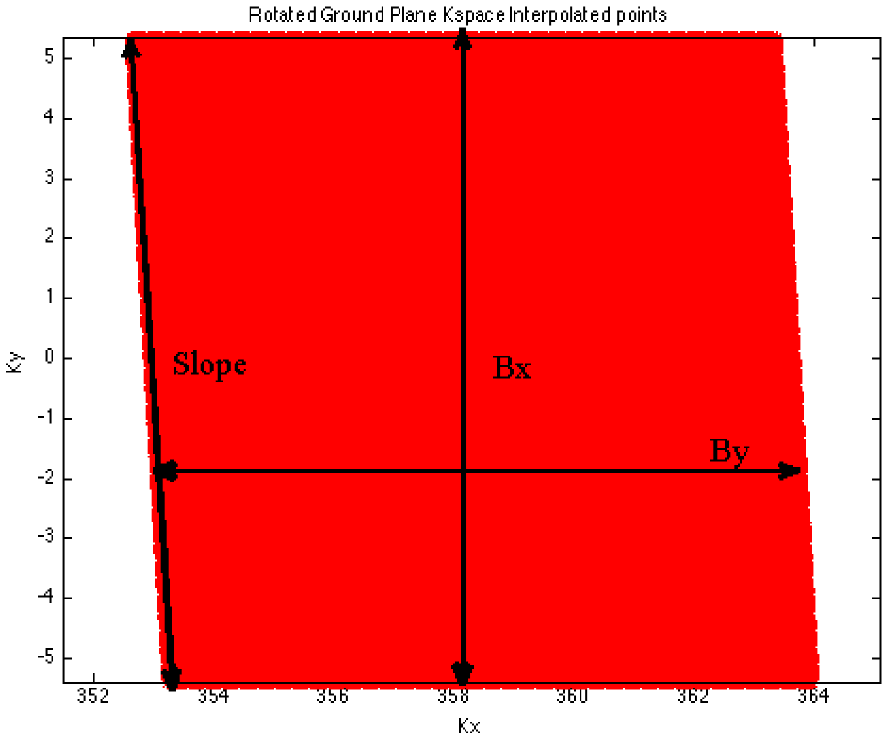

Figure 5.11: *Rotated bistatic k-space illustrating slope and bandwidths in the x and y dimensions where the red area represents the interpolated k-space.*

These parameters were determined in lines 28 to 42 of `gen_Kspace.m`. Once these parameters were found, they were used to calculate the interpolation grid in the rotated space. The interpolated k-space is given by these equations found in [13], where N_y and N_x are the number of pixels in the final image.

$$k'_{y_{intp}} = \frac{4\pi}{c}\Big(\frac{-B_y}{2} + B_y\frac{n_y}{Ny}\Big) \qquad n_y = 0, ..., N_y - 1 \tag{5.12}$$

$$k'_{x_{intp}} = \frac{4\pi}{\lambda_c} + k'_{y_{intp}}M + B_x\frac{n_x}{Nx} \qquad n_x = 0, ..., N_x - 1 \tag{5.13}$$

Comparisons of the interpolated k-space and collected k-spaces, shown in Figure 5.12. The red and blue areas are the interpolated and collected k-spaces, respectively in Figure 5.12. The collection bandwidth in the k_y dimension is larger than the interpolated bandwidth.

Within the algorithm of lines 31 to 34 of `gen_Kspace.m`, the bandwidths are set equal to each other using the smallest bandwidth. This is visually pleasing to a human viewing an image because most humans are accustomed to viewing photographs with equal resolution in each spatial dimension and use this experience as a reference for viewing SAR images. Generally speaking, in a photography the photo receptors are spaced equally in the vertical and horizontal dimensions giving equal bandwidth and resolution in the horizontal and vertical dimensions of tan image. If this image was solely interpreted by a machine, like a classification algorithm, then the bandwidths may not need to be set equal. The classification algorithm may have better performance with a finer resolution in one dimension.

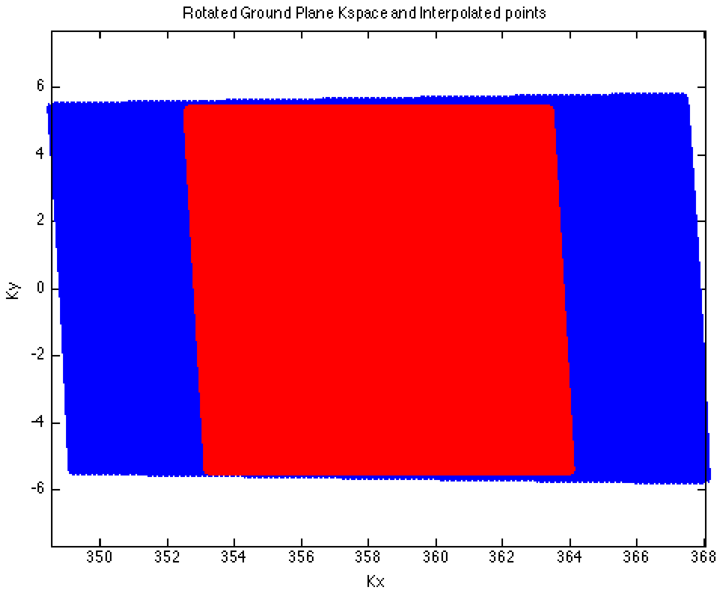

Figure 5.12: *Interpolated k-space (red) superimposed on the collected k-space (blue).*

After the interpolation k-space grid is formed, the data is interpolated using the `griddata()` function in `Bistatic_SAR_Sim.m`. At this point, the image is ready to be formed by taking a two-dimesional FFT in line 98 of `Bistatic_SAR_Sim.m`.

5.6 Image Formation Results

Figures 5.13, 5.14, 5.15, and 5.16 are simulated SAR images using the simulator described in this chapter. In these figures red indicates a high value in the image and correlates with the location on the point targets while blue indicates a low intensity value and most of the time represents simulated radar system noise. The square root of the magnitude of the images was taken to improve the dynamic range of the images, separating the targets from the noise. The system noise, range and cross-range sidelobes of the targets are present in Figures 5.13, 5.14, 5.15, and 5.16. Figure 5.13 contains the monostatic polar formatted image of the three targets in Figure 5.3.

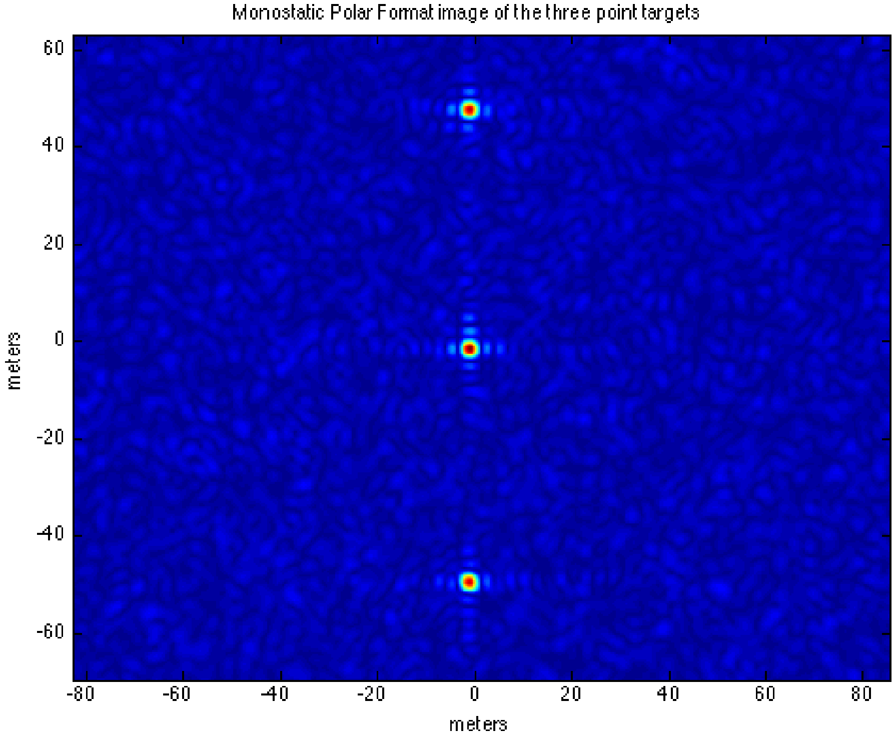

Figure 5.13: *A Simulated Monostatic Polar Formatted Image with three point targets. Red indicates an high value in the image which is the target's location.*

Figure 5.14 contains the bistatic polar formatted image of the three targets in Figure 5.3. In this image, the targets are rotated clockwise 22 ° due to the bistatic angle. This rotation occurred in the k-space processing, and it is represented in Equations (5.12) and (5.13).

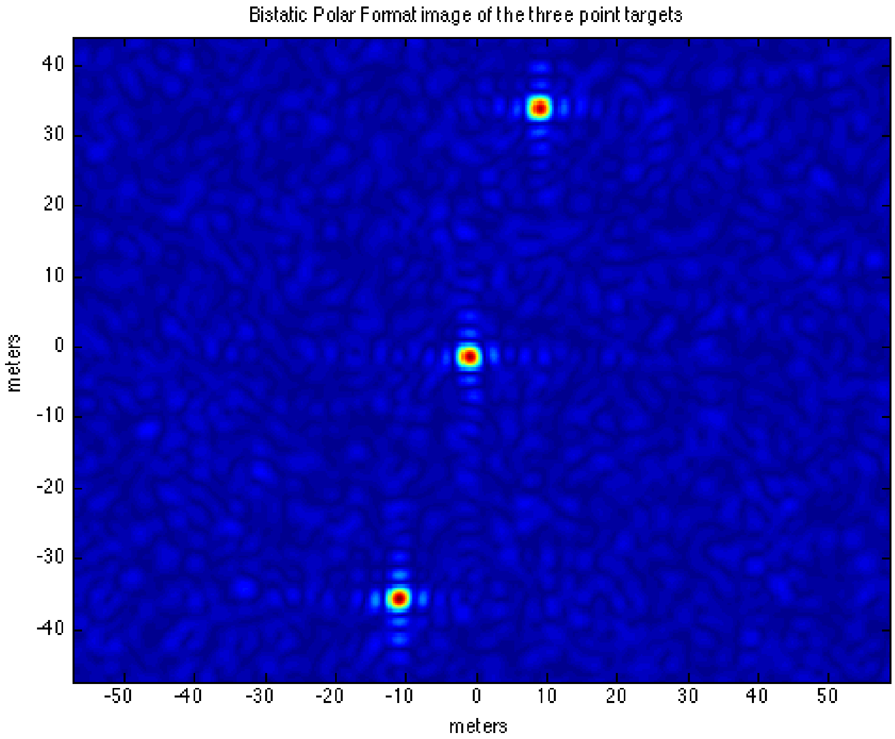

Figure 5.14: *A Simulated Bistatic Polar Formatted Image with three point targets. Red indicates an high value in the image which is the target's location.*

Figure 5.15 displays the monostatic polar formatted image of the OU target in Figure 5.6.

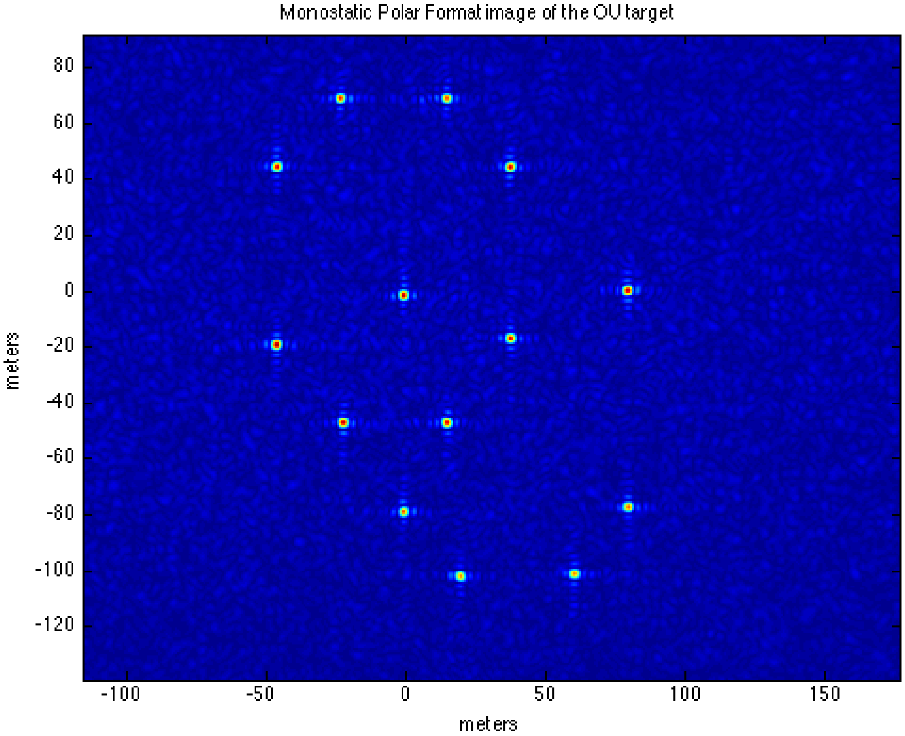

Figure 5.15: *The Monostatic OU point target Polar Formatted Image.*

Figure 5.16 contains the processed bistatic polar formatted image of the OU target in Figure 5.6. Again, in Figure 5.16, the targets are rotated clockwise 22 ° due to the bistatic angle like the bistatic polar formatted image in Figure 5.14.

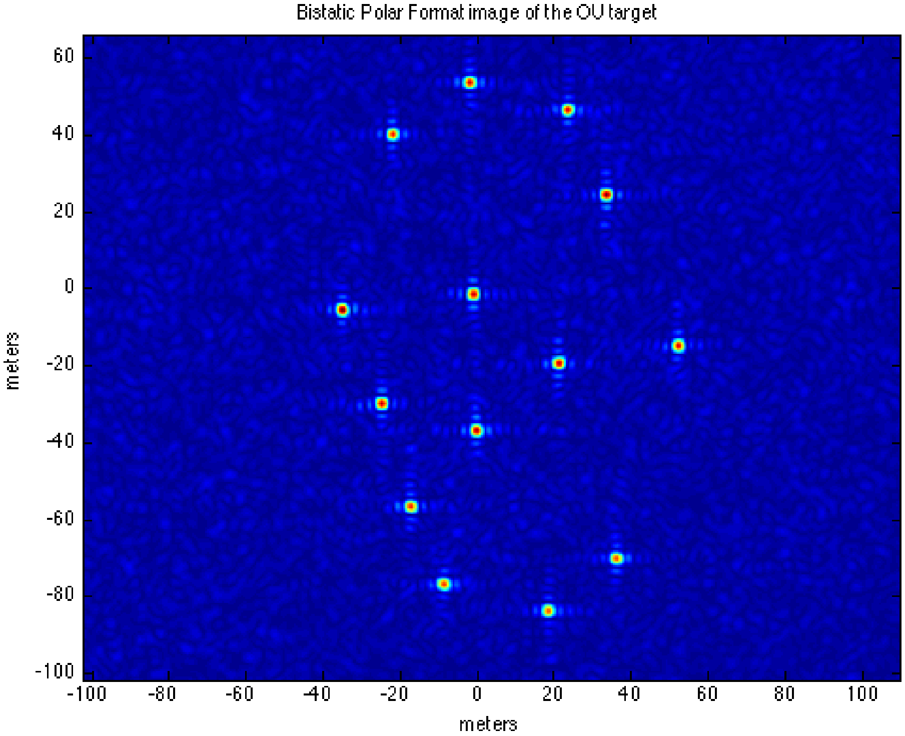

Figure 5.16: *The Bistatic OU point target Polar Formatted Image.*

5.7 Forward-Scatter Bistatic SAR Simulation

The bistatic SAR simulation was developed using a stretch processing signal model developed in [13], and [6], and the signal was processed using the bistatic polar format algorithm described in these citations as well. In each of the simulated images in this section, the transmitter and receiver had a 180° bistatic angle at each PRI and each system had a fixed elevation angle with respect to the SRP. Each platform flew a straight flight path depicted in a bird's eye view of the scene in Figure 5.6.

The receiver's height above the flat Earth was quadratic so that the bistatic and elevation angles remained constant. This slight bend in the receiver's height over the flight paths are so that the angular exactness is maintained. Figures 5.17 and 5.18 show a simulated case where the transmitter and receiver had elevation angles of 10° and 45°, respectively.

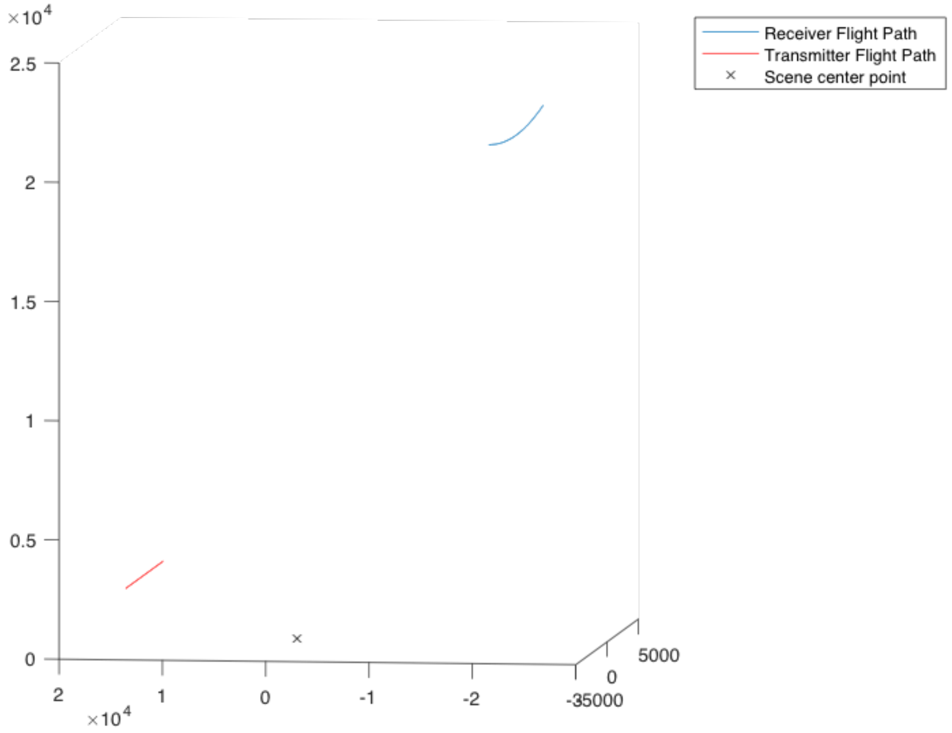

Figure 5.17: *Three-dimensional view of the simulated transmitter and receiver flight paths when their elevation angles were* 10° *and* 45°*, respectively.*

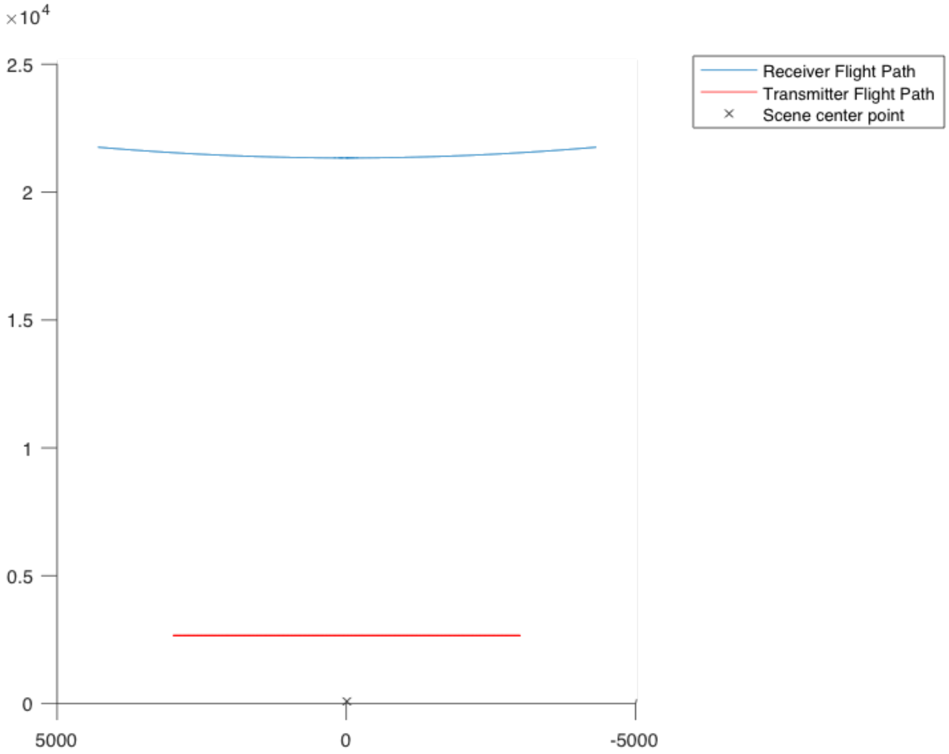

Figure 5.18: *Profile view of the simulated transmitter and receiver flight paths when their elevation angles were* 10° *and* 45°*, respectively.*

Figure 5.19 shows a birds eye view of their flight paths which is very similar to the flight path depictions in Figure 5.6.

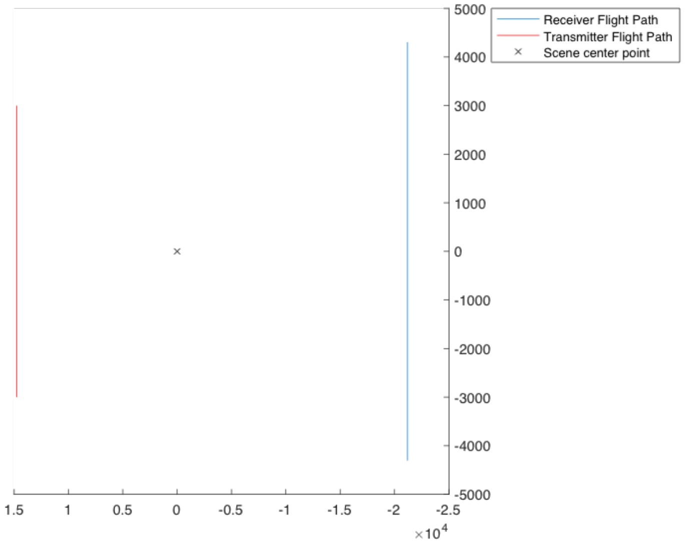

Figure 5.19: *Bird's eye view of the simulated transmitter and receiver flight paths when their elevation angles were* 10° *and* 45°*, respectively.*

Each point target within Figure 5.6 resides within the $x-y$ plane of Figure 2.1 (a) and (b). The transmitter and receiver had a slant range of 15 and 30 km to the scene reference point (SRP), respectively. The elevation angle for the transmitter was fixed at $\phi_{TX} = 10°$ for each simulated image while the receiver's elevation angle was constant during each simulation as well, but varied for each simulated bistatic SAR image as indicated in the resulting images in Figure 5.23. The fixed elevation angles and bistatic angle were used to illustrate the geometric effects on the range resolutions and cross-range resolutions described in Equations (3.9) and (3.39) respectively. The transmit waveform parameters were a 10 GHz center frequency, f_c, and 3 GHz bandwidth, BW.

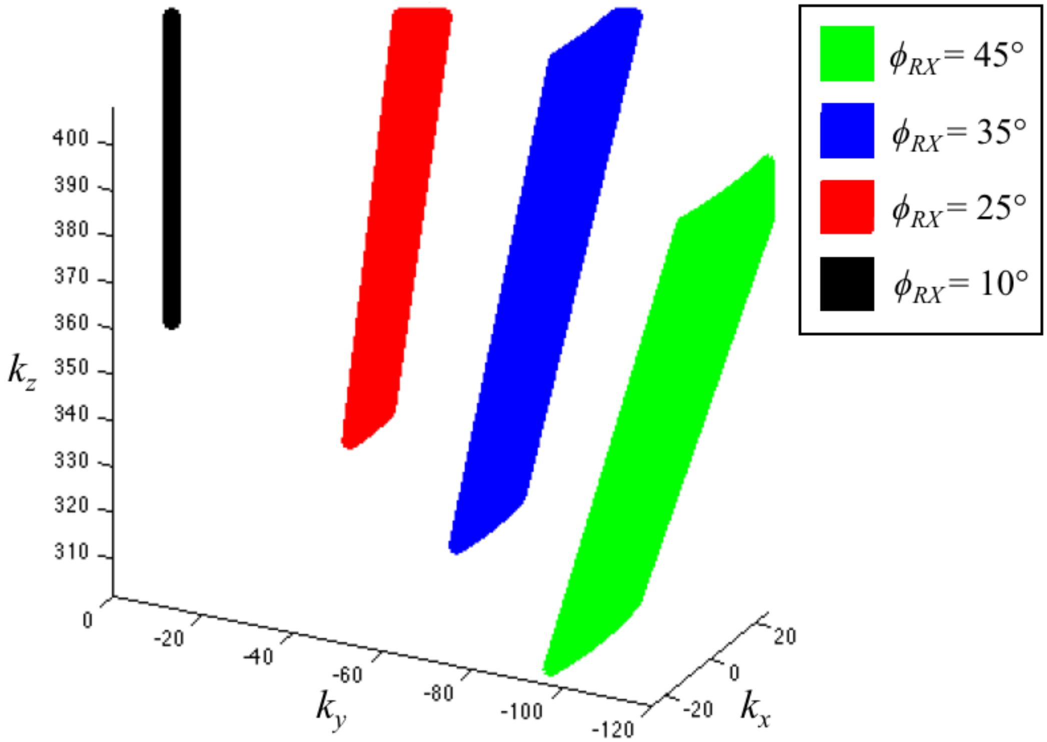

Figure 5.20: *Three-dimensional view of each of the k-spaces simulated. The elevation angle for the transmitter is fixed at $\phi_{TX} = 10°$ for each simulated image.*

Viewing the k-space created by each of the collections during the simulations gives a better understanding of how the resolution in each of the images was created. In Figure 5.20, each k-space is plotted from the simulation in a three-dimensional perspective. When elevation angles are equal, as in the case when $\phi_{RX} = 10°$, the k-space projects to a point in the ground plane as shown in Figure 5.21 with the black dot centered the point $(0, 0)$. No range or cross-range resolution was achieved in this case, which is apparent in Figure 5.21, when each of these k-spaces are projected into the ground plane. The k-space for $\phi_{RX} = 35°$ overlaps the k-space for $\phi_{RX} = 45°$ in Figure 5.21. The

k-space for $\phi_{RX} = 45°$ has been reproduced in Figure 5.22. The width of the ground plane projected k-space in the y-dimension of Figures 5.21 and 5.22 is inversely proportional to the range resolution as described in Equation (3.7). The cross-range resolution in the ground plane projected k-space within the x-dimension in these plots, is also inversely related as described in Equation (3.28). Figure 5.22 has both of these k-space distances noted within the figure.

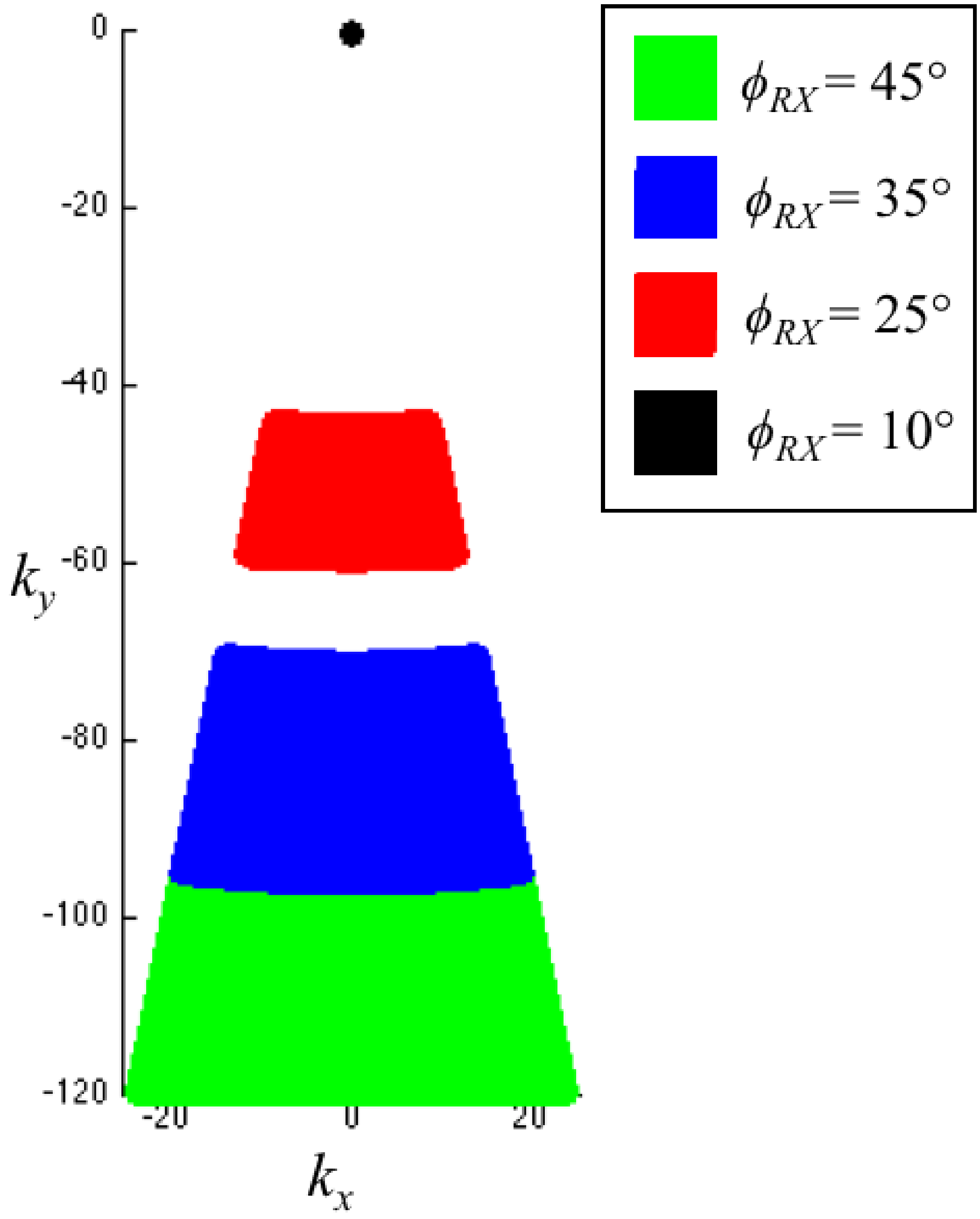

Figure 5.21: *Three-dimensional k-spaces projected into the ground plane. The elevation angle for the transmitter is fixed at $\phi_{TX} = 10°$ for each simulated k-space.*

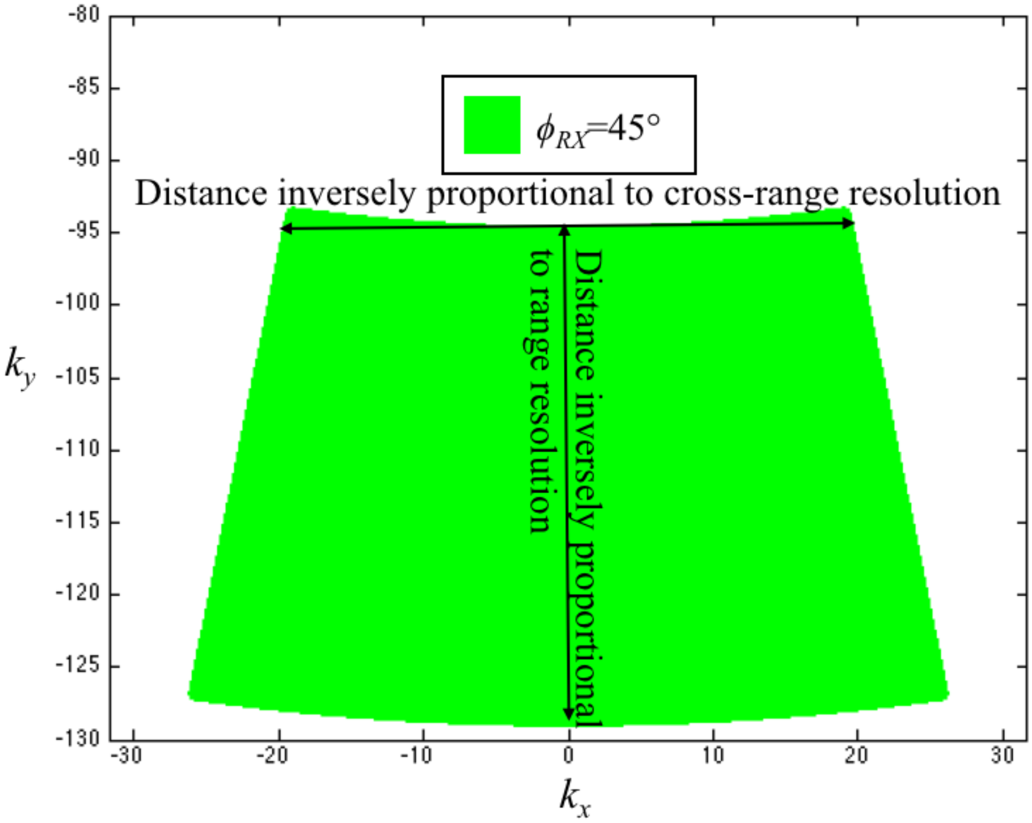

Figure 5.22: *The* $\phi_{RX} = 45°$ *k-space is into the ground plane. The elevation angle for the transmitter is* $\phi_{TX} = 10°$*.*

The imaging results from the simulation are found in Figure 5.23. As expected, from Equations (3.9) and (3.39), no image appears in the case where the elevation angles are equal because the resolution tends to infinity. This implies that a target cannot be resolved in range and cross-range. As the difference between the elevation angles increases, the point targets in the shape of the University of Oklahoma logo, OU, as depicted in Figure 5.6, appears and the resolution becomes finer.

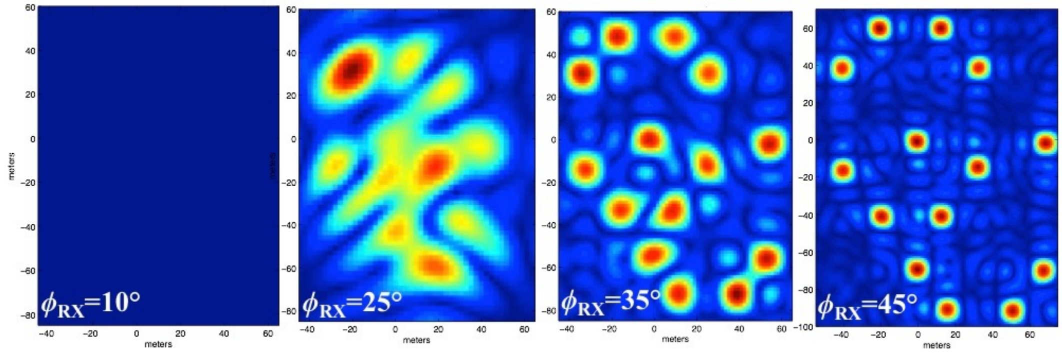

Figure 5.23: *Bistatic forward-scatter SAR simulation results with fixed* $\phi_{TX} = 10°$ *and* ϕ_{RX} *values as indicated in each sub-image. These are the images of simulated point scatters in the shape of the University of Oklahoma's logo, OU.*

Chapter 6

SAR Imagery Results

6.1 Forward-Scatter Bistatic SAR Test Results

To validate the new approach for the range and cross-range resolution calculations in Sections 3.3 and 3.7 respectively, anechoic chamber measurements were prepared. Tests in an anechoic chamber were also performed to test the ability to resolve targets in a forward-scatter scenario. Within this chamber, there were two masts that held the antennas. The transmitter's mast was maintained at a constant elevation angle of 25° while the receiver's mast changed elevation angles. A turn table was placed directly in the center of the two masts and rotated two corner reflectors 3.3 meters apart. Figure 6.1 is a bird's eye view of the setup. The placement of the turn table and two masts created a bistatic angle of $\beta = 180°$.

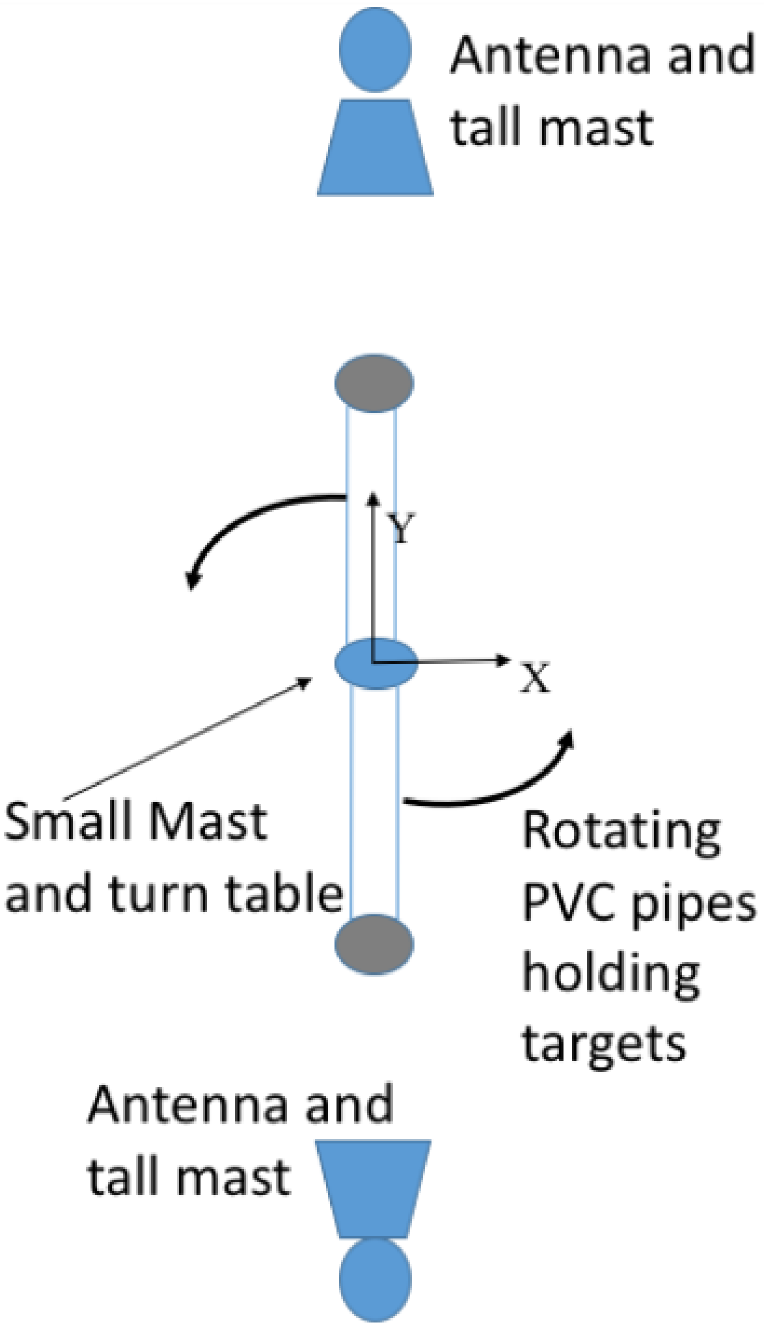

Figure 6.1: *Bird's eye view of the test setup in the anechoic chamber.*

A network analyzer was connected to a computer which was controlled by LabView®. Mr. Simon Duthoit generated the Labview®functions which controlled the turntable and the network analyzer. Without his help this testing would have been much more difficult for the author to pursue. The author is grateful for Mr. Duthoit's contribution. The computer, in Figure 6.4, controlled the turn table within the center of the chamber. The turn table moved 0.0125° for each collection of frequency responses. The network analyzer stepped 8001 frequency points from 8.5 GHz to 11.5 GHz centered at 10 GHz. The range resolutions for the four experiments are found in Figure 3.4 when $\beta = 180°$ using Equation (3.9), but due to the configuration in the chamber, the bistatic angle is approximately 180° for the targets. The center of the turn table is the only location in the chamber where the bistatic angle is always 180° for this configuration. Using Equation (3.44), the angular extent needed was determined. As discussed prior, this equation was developed by equating (3.9) and (3.40) since the angular extent for this collection was approximately equal for transmit and receive, i.e. $\Delta\theta_{RX} \approx \Delta\theta_{TX} \approx \Delta\theta$. Once again, the center of the turn table is the only location where they are truly equal for this configuration.

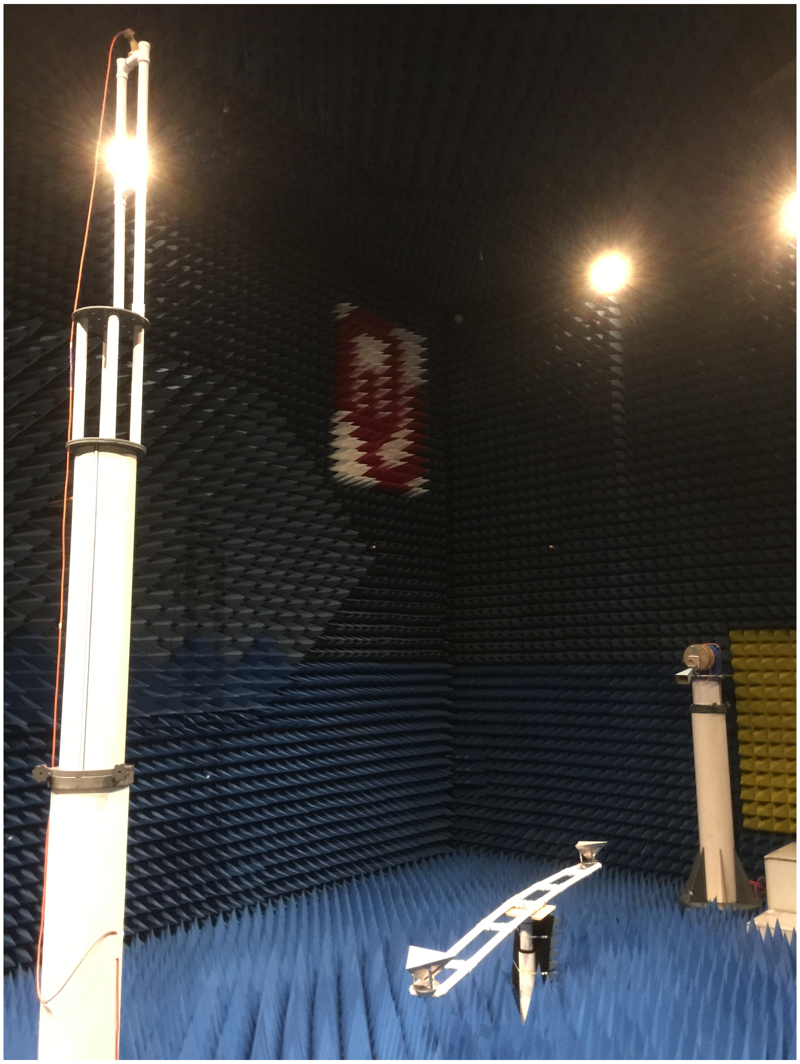

Figure 6.2: *Test setup within the anechoic chamber with* $\phi_{RX} = 55^\circ$ *and* $\phi_{TX} = 25^\circ$.

Once all the data was collected for a specific receiver position, it was then processed in a series of three steps. First, the data was shifted to the center of the chamber where the SRP was located at the top and center of the turn table. Second, the data was CLEAN processed which will be discussed in more detail in the next paragraph. Then, the data was processed with the bistatic polar format algorithm described in [13] and [6]. Figure 6.2 is a picture of the setup within the chamber with the two corner reflectors.

The first set of data was collected in the chamber with the corner reflectors and the PVC pipe removed. A corner reflector was placed on the top of the turn table and rotated to find the center point within the chamber. This data was used to determine the scene center for the following two data sets. A second data set was collected where the PVC pipe was placed on the turn table without the reflectors. A third data set was collected with the reflectors attached to the PVC pipe. Both the second data set and the third data set were centered to the center of the turn table using the results from the first data set. Then the data set without the reflectors was subtracted from the data set with the reflectors to remove the reflections from the turn table and the PVC pipe in similar manner to the CLEAN algorithm [47–49]. Once the data set was preprocessed using the previously described manner, the data was ready to be processed using the bistatic polar format algorithm in [6, 13], and also used in the simulation in Section 5.7.

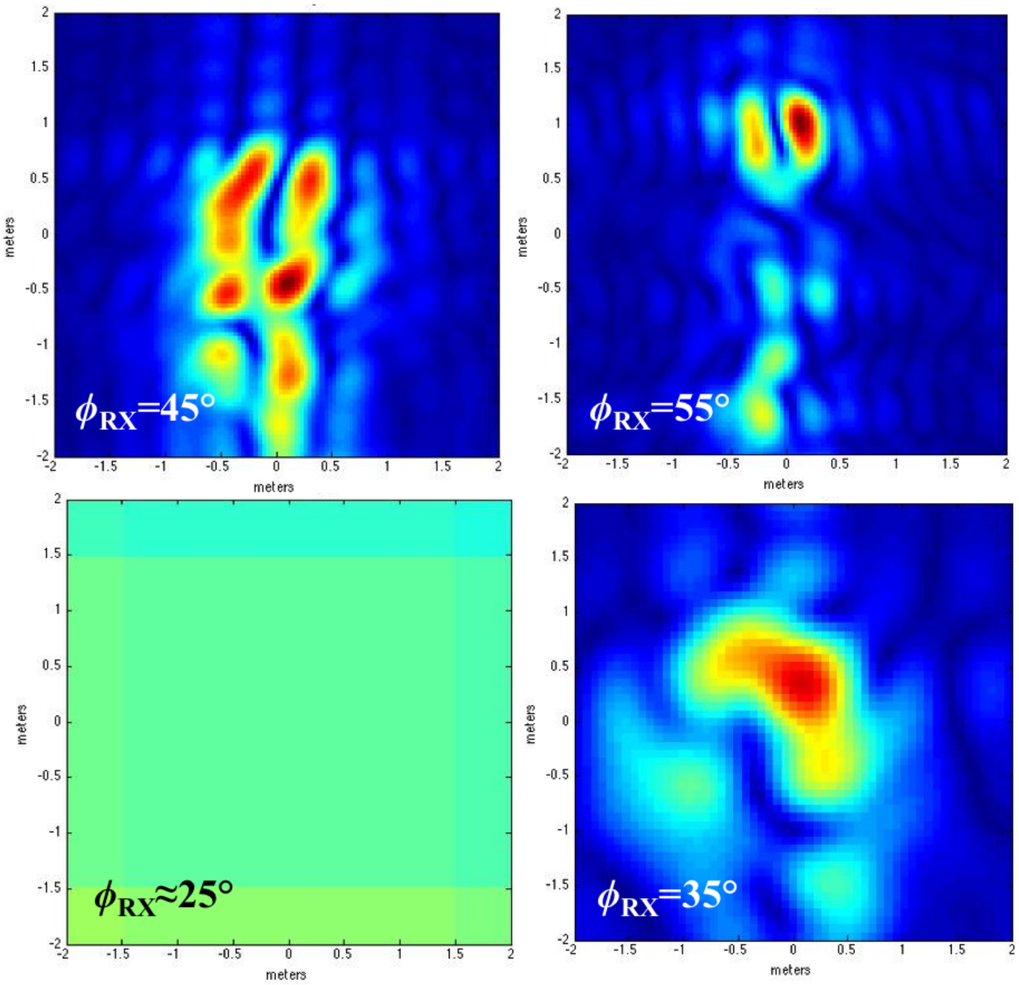

Figure 6.3: *Bistatic forward-scatter SAR test results with fixed $\phi_{TX} = 25°$ and various ϕ_{RX} values as indicated in each sub-image.*

Within each of the resulting images found in Figure 6.3, the two reflectors became more resolvable as the difference between the elevation angles increases. Other reflections occurred during these tests and could not be removed with the CLEAN processing. The reflectors were also secured by a metal pedestal which may have created a multi-path bounce near the reflector, and the CLEAN method would not remove multi-path bounces between the reflectors and other scatterers in the chamber like the PVC pipe or the turn table.

6.2 FDWBPA Processing

This algorithm was implemented using the Gotcha Large Scene Data Example, which is provided by the Air Force Research Laboratory (AFRL), Sensor Data Management System (SDMS) [50]. This data set is open to the public and has appeared in many publications. To form the SAR images in this document, a circular flight path fit, which was determined in [51, 52] is used as the actual flight path. This fit is a radius of 10.4994 km, an azimuth extent ($\Delta\theta$) of 3.322° (meaning the number of degrees the aircraft traveled) and an elevation angle (ϕ) of 44.341°.

The processed SAR images takes into account a curved Earth and range curvature of the received signal. The Earth's curvature is approximated as a sphere since the location of the data collection was unknown. The average of the Earth's radius is determined to be 6,371,008.8 m. Then, range curvature corrections were applied to each pixel of the image by subtracting the corrections from the range. The first and second order range curvature corrections used are found in [23] while second order corrections for the z direction of the image due to the curved Earth approximation are found in [12]. If a flat Earth were assumed, then there would be no need for z direction corrections.

The other aspect of this algorithm is processing time. To reduce the amount of processing time the BPA takes, an NVIDA®Tesla K40c GPU formed the images along with Matlab's®GPU toolbox to implement the algorithm. The Matlab®functions used to develop and implement this algorithm are found in Appendix B. An image of the GPU is in Figure 6.4. At first, writing the backprojection algorithm in CUDA code was pursued but the FFT functions were not as flexible as Matlab's®GPU toolbox. These FFT's were needed for stretch processing the return. Matlab's®Built in GPU toolbox provided the needed flexibilty for development [53].

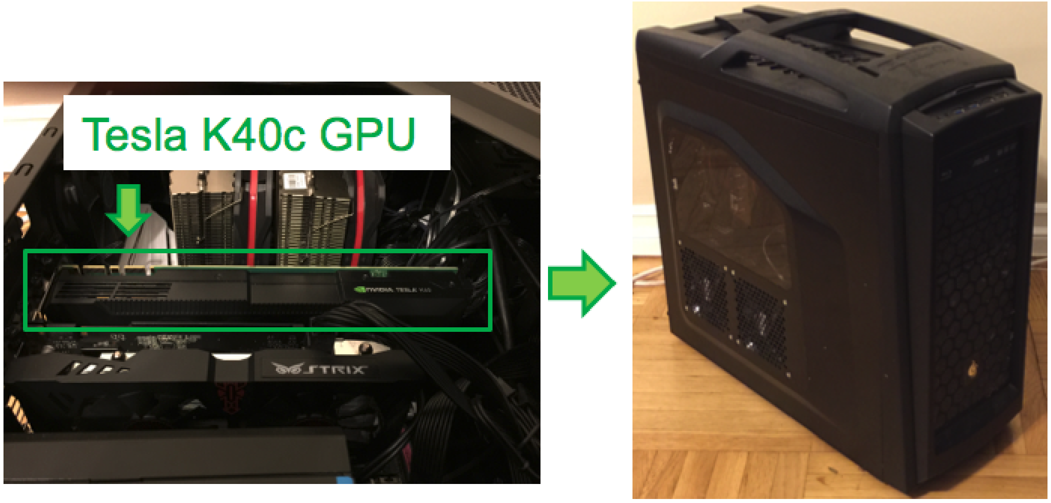

Figure 6.4: *Picture of the NVIDA Tesla K40c GPU and computer running the GPU.*

The individual processing time of each image was mainly due to the processing configuration, i.e. the NVIDA®Tesla K40c GPU and Matlab's®Built in GPU toolbox. The computer's configuration included a solid state drive and large amount of RAM which played a minor role increasing the processing time as well. In the future, this algorithm could be sped up through a CUDA implementation. The processing time in Figure 6.5 corresponds with the resolutions and Decimation Factors in Section 6.3.

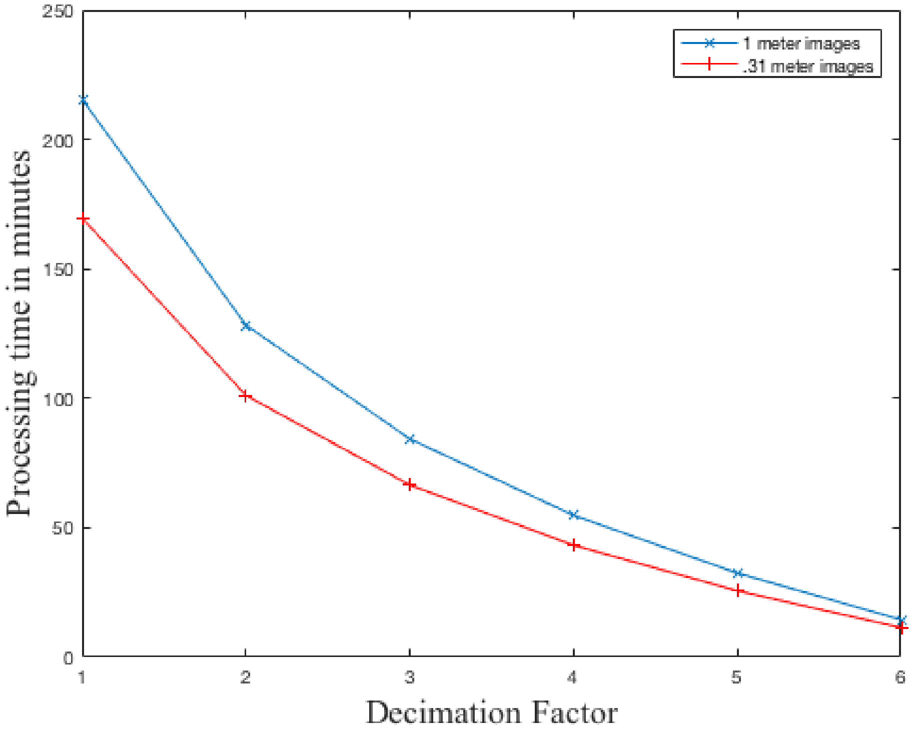

Figure 6.5: *Processing time of each Back Projected SAR image .*

From the processing times in Figure 6.5, it is apparent that as DF increases the processing time will decrease compared to the BPA. A BPA image is the case when $DF = 1$. Comparing the $DF = 1$ case or BPA image to the other cases demonstrates the algoritm's processing time improvements. In Figure 6.6, the ratio of the BPA image is compared to the FDWBPA image which quantifies this improvement.

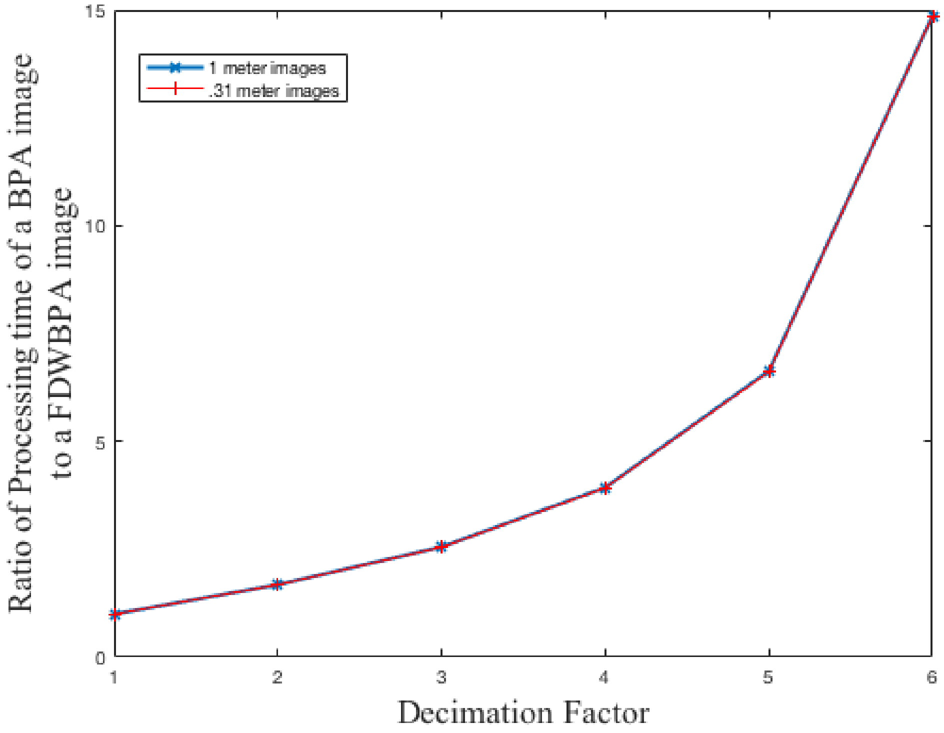

Figure 6.6: *Ratio of a BPA image to a FDWBPA image.*

From Figure 6.6, a FDWBPA image with $DF = 6$ is processed over 14 times faster than the BPA image.

6.3 FDWBPA Results

This section contains two sets of SAR images. Each image in this section was created by a single dataset, the Gotcha Large Scene Data collection [50]. A variety of other researchers have found this dataset to be useful for the evaluation of innovative SAR algorithm development [54–58]. The authors of [59] use this data set to test a recurise back-projection algorithm. In [60], the authors use this data set to present a new radar processing architecture for simultaneous radar modes. Table 14-2 in [61] provides a variety of details about the GOTCHA radar as well.

The first set of images illustrates how the algorithm works and when it is appropriate to use this algorithm. This algorithm is intended for high resolution SAR imagery with a reasonable but limited scene size in the cross-range dimension. For many SAR applications, when cross-range resolution becomes finer the scene size tends to decrease unless large scene size and high resolution SAR imagery is intended which is why choosing the correct Decimation Factor, DF, is important. The second set shows the overall results of this algorithm for a high resolution SAR images with a limited scene size.

This algorithm decimates the signal by summing neighboring radar returns and their positions together causing the frequency support of the sampled signal to decrease and aliasing to occur in the cross-range dimension. Figures 6.7 to 6.12 demonstrate this aliasing while increasing DF of the image for a large scene SAR image. These figures are provided to illustrate what is being traded while using this algorithm and why in certain cases like large scene SAR images this algorithm could have less of an impact.

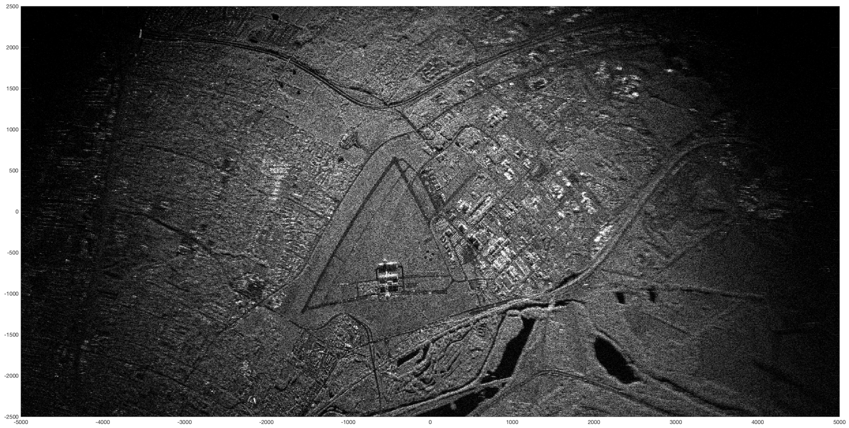

Figure 6.7: *Back Projected Image without Wavenumber Decimation, i.e. Decimation Factor of 1, 1 meter resolution. This image took 3.59 hours to process.*

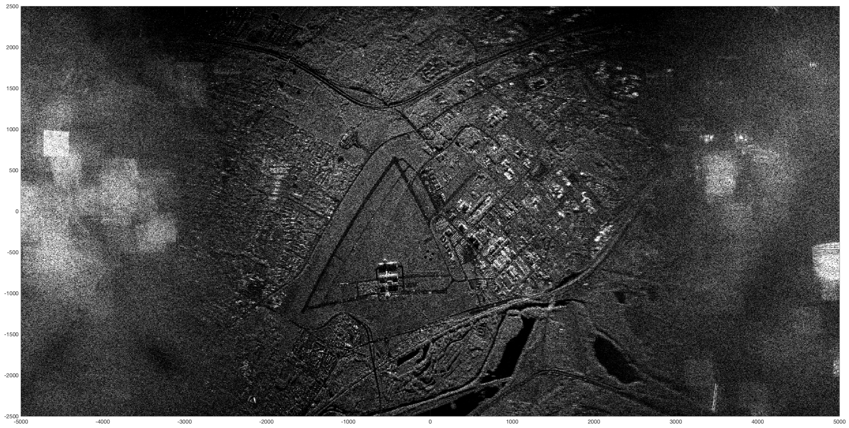

Figure 6.8: *Back Projected Image with Wavenumber Decimation Factor of 2, 1 meter resolution. This image took 2.14 hours to process.*

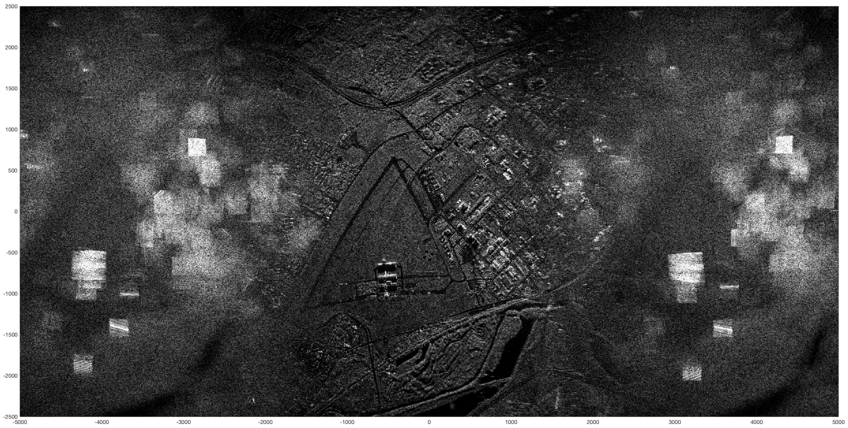

Figure 6.9: *Back Projected Image with Wavenumber Decimation Factor of 3, 1 meter resolution. This image took 1.41 hours to process.*

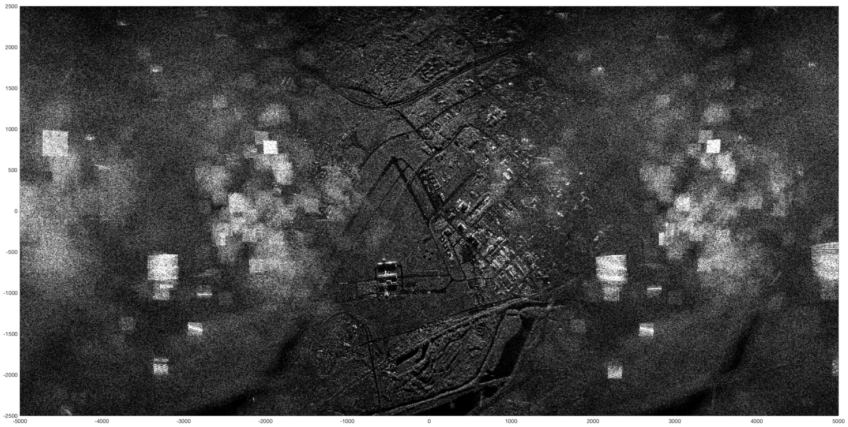

Figure 6.10: *Back Projected Image with Wavenumber Decimation Factor of 4, 1 meter resolution. This image took 54.8 minutes to process.*

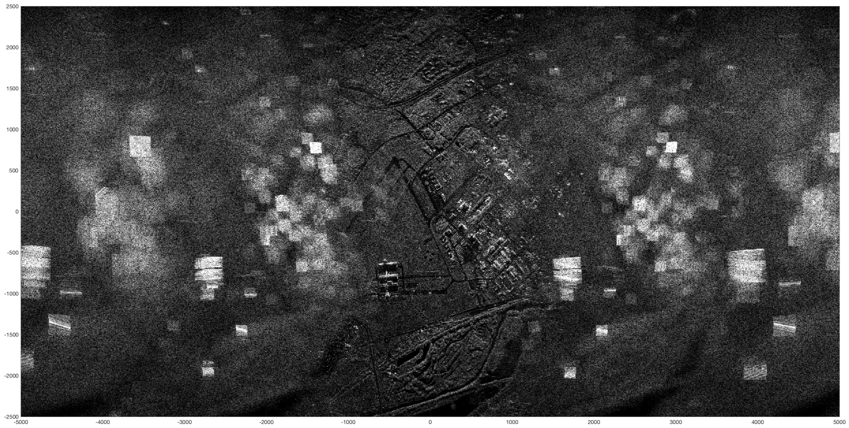

Figure 6.11: *Back Projected Image with Wavenumber Decimation Factor of 5, 1 meter resolution. This image took 32.5 minutes to process.*

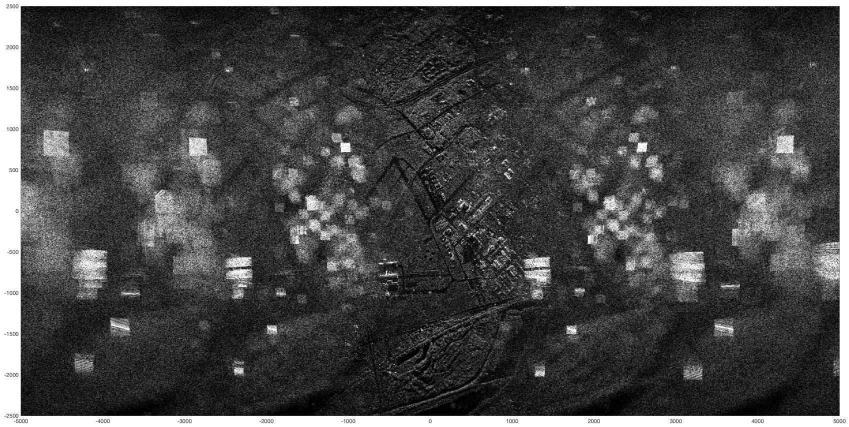

Figure 6.12: *Back Projected Image with Wavenumber Decimation Factor of 6, 1 meter resolution. This image took 14.5 minutes to process.*

As DF increases, the aliasing becomes more severe and the alias artifacts encroach on the center region of the image. It's important to remember that the SRP exists within the center of the image and the radar system intends to illuminate this area since it pointed the antenna in this location when spotlighting the system. This demonstrates how Decimated Wavenumber aliasing works in favor of this algorithm. Figure 6.13 has a DF of 6 and indicates the usable image area within this image.

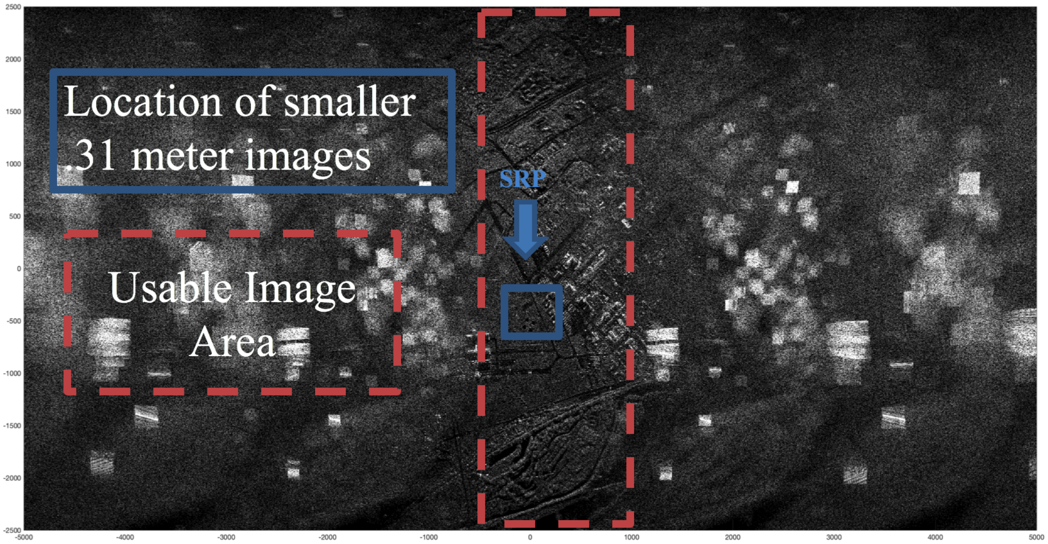

Figure 6.13: *Back Projected Image with Wavenumber Decimation Factor of 6, 1 meter resolution. Figures 6.14 to 6.19 are located within the blue box. The usable image area is within the red dashes box.*

Within Figure 6.13, there is a smaller blue box that indicates the imaging area of the next set of .31m resolution images in Figures 6.14 to 6.19. Each of these images are intended to illustrate how reducing the Decimation Factor, DF, did not significantly reduce the image quality for this specific image location until a DF of 6 is applied.

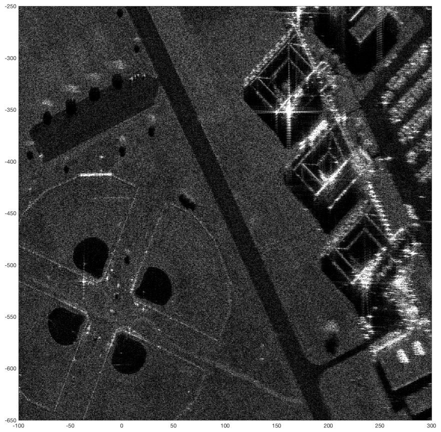

Figure 6.14: *Back Projected Image without Wavenumber Decimation Preprocessing, i.e. Decimation Factor of 1, 0.31 meter resolution near SRP. This image took 2.81 hours to process.*

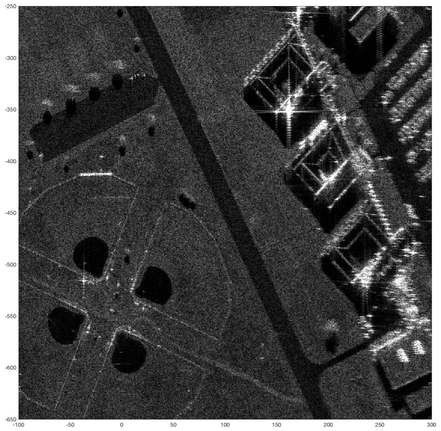

Figure 6.15: *Back Projected Image with Wavenumber Decimation Factor of 2, 0.31 meter resolution near SRP. This image took 1.68 hours to process.*

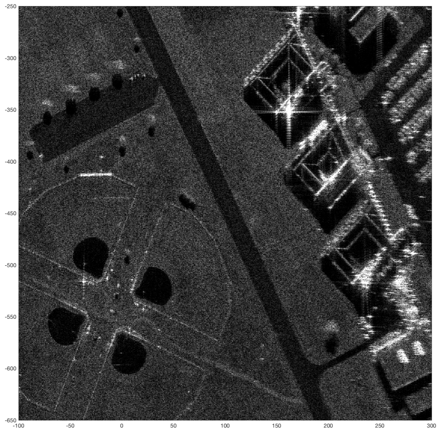

Figure 6.16: *Back Projected Image with Wavenumber Decimation Factor of 3, 0.31 meter resolution near SRP. This image took 1.11 hours to process.*

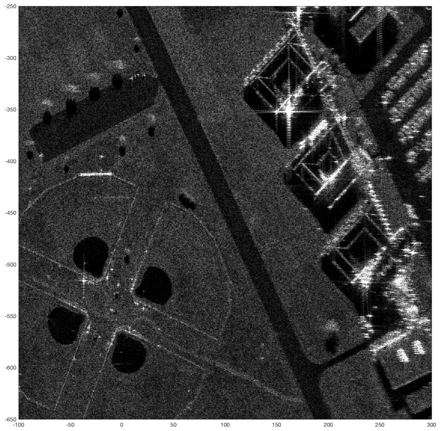

Figure 6.17: *Back Projected Image with Wavenumber Decimation Factor of 4, 0.31 meter resolution near SRP. This image took 43.2 minutes to process.*

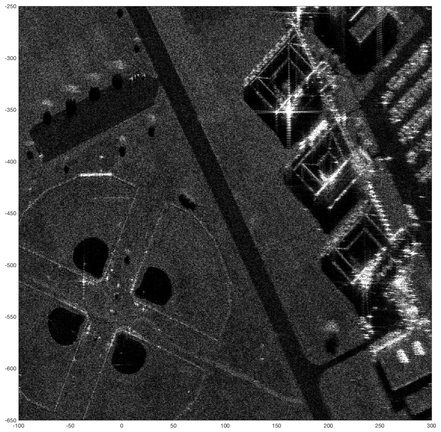

Figure 6.18: *Back Projected Image with Wavenumber Decimation Factor of 5, 0.31 meter resolution near SRP. This image took 26.2 minutes to process.*

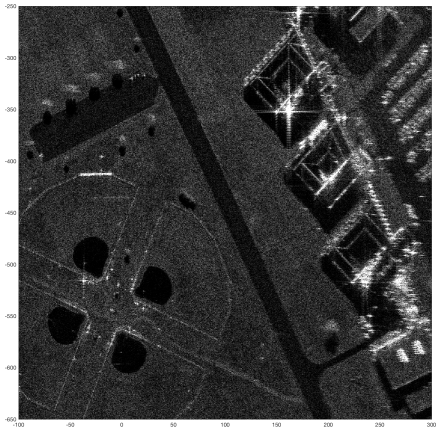

Figure 6.19: *Back Projected Image with Wavenumber Decimation Factor of 6, 0.31 meter resolution near SRP. This image took 11.4 minutes to process.*

Figures 6.14 to 6.19 demonstrate how the image quality of this specific image is preserved as DF increases. Comparing Figure 6.14 to Figure 6.19, the images have very little difference. The only difference that can be seen is in the upper right corner of the image where some of the aliasing has crept in. If this is undesirable, a DF of 5 can be used, which does not appear to have this artifact as shown in Figure 6.18. The coherent summation process within the algorithm is intended to preserve the image SNR compared to just decimating the number of radar returns by a factor of DF. Figure 6.20 illustrates why the summation of the radar returns is necessary. Figure 6.20 b) has very poor image quality in comparision to Figure 6.20 a), which used the FDWBPA with a DF of 6.

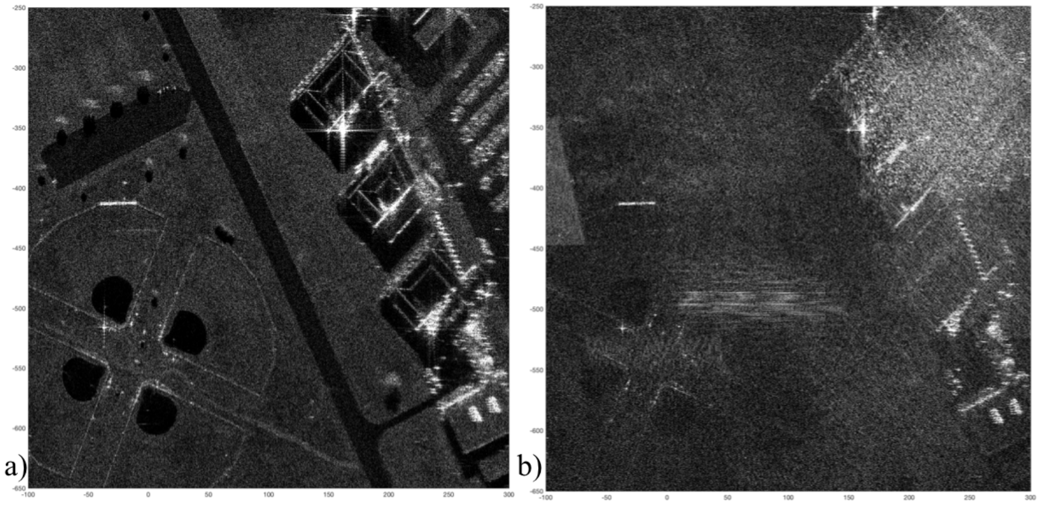

Figure 6.20: *Comparison of Back Projecting every 6th pulse compared to FDWBPA with a DF of 6. a) FDWBPA image with a DF of 6 and b) every 6th radar return back projected*

Figure 6.20 illustrates why averaging the radar returns and their positions is an important step within this algorithm. Without this step, the SNR of the image would be degraded.

Chapter 7

Future Considerations For SAR systems

One of the most important aspects for any imaging radar is how this system creates cross-range resolution. Most SAR systems use an aircraft's motion to create cross-range resolution while ISAR systems use the motion of the target. Developing a test plan for a bistatic radar system will start with the resolutions equations in Chapter 3. Simulation results from Chapter 5 will also demonstrate some of these key aspects.

7.1 Test Plan for an Experimental Bistatic Radar

A way to test a bistatic radar system is to place a target, like a car or truck, on a rotating platform and image it with a transmitter and receiver in different locations as shown in Figure 7.1. The angular rotation of the platform will give the cross-range resolution needed to image the target. The rotation of the platform will need to be measured or determined by some means. This method will create the angular change to create a monostatic and bistatic ISAR image.

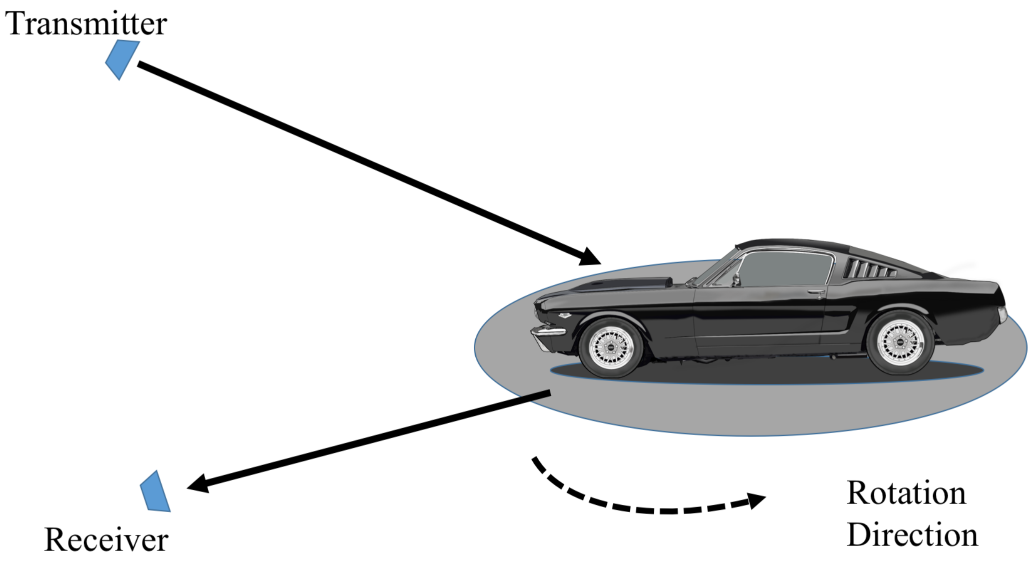

Figure 7.1: *Illustration of a car imaged on a rotating platform. (Ford Mustang image used under CC0 license, url: https://pixabay.com/p-146580)*

Because $\theta_{TX}(n)$ and $\theta_{RX}(n)$ will be changing at the same rate, this will create a k-space similar to a monostatic k-space. A k-space, where the two platforms are separated by a 90 ° bistatic angle, β, and the target is rotating at the same angular rate, creates a k-space similar to one generated with a monostatic system. A k-space like this has been simulated and then plotted in Figure 7.2. This bistatic k-space is similar to a monostatic SAR k-space allowing monostatic SAR processing techniques to generate an image. Other complex shaped bistatic k-spaces can be processed with monostatic algorithms like the backprojection algorithm or the polar format algorithm modified for bistatic systems [6, 30].

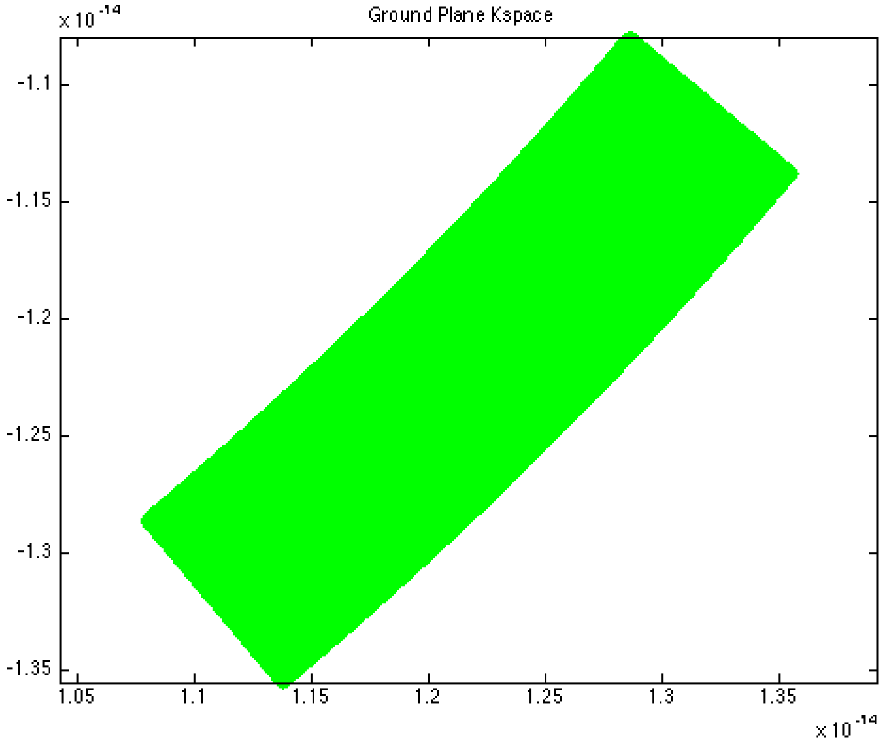

Figure 7.2: *Bistatic k-space from platforms with a 90 degree bistatic angle and same angular rotation.*

This method of testing will produce results that will be essential in collecting other sets of data that will lead to more exotic shaped k-space collections. To create more of these stressing k-space collections illustrated in Figure 7.3 (b), the transmitter and receiver will need to move independently.

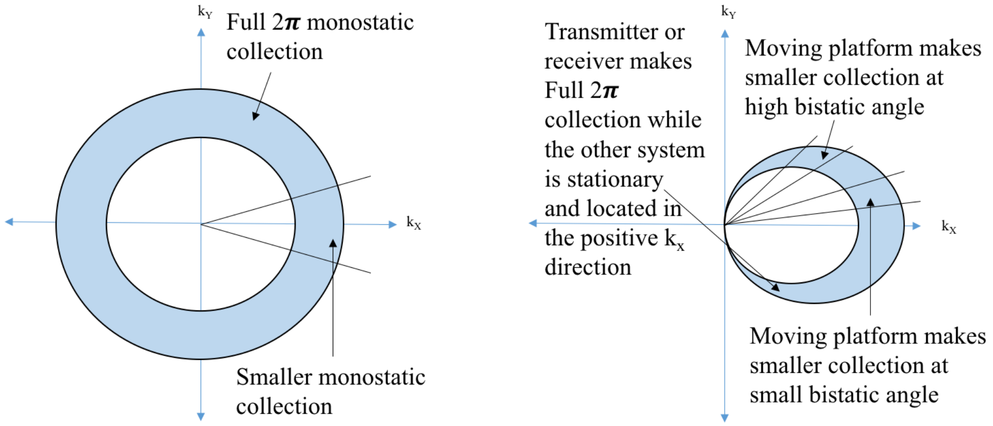
k_Y
Full 2π monostatic collection
k_X
Smaller monostatic collection
k_Y
Transmitter or receiver makes Full 2π collection while the other system is stationary and located in the positive k_x direction
Moving platform makes smaller collection at high bistatic angle
k_X
Moving platform makes smaller collection at small bistatic angle

Figure 7.3: *a) Monostatic k-space b) bistatic k-space where one of the radar systems is stationary and lies on the positive k_x axis*

7.2 Test Plan to Collect Bistatic SAR Data from Independent Moving Radar Systems for a Bistatic Radar system

Most SAR systems are tested by flying the system on an aircraft to create the synthetic aperture. This method is usually very expensive because it requires owning an airborne platform and it is also the conventinal method for creating the aperture. Another method proposed is mounting the radar to a ground vehicle and driving it across a bridge or a cliff and to image a scene below. SAR systems on a truck have been described in these two patents [62, 63] and other technical papers [64].

A bistatic radar system can consist of an active monostatic system and a passive bistatic radar system. To remove the complexity and cost in testing the system on a truck or a plane, one system can remain stationary while moving the other radar system. Either the monostatic or the passive bistatic radar system can have the motion measurement system attached to it and moved on a vehicle to generate a bistatic SAR image.

Ideally, the monostatic system would be the mobile system, so that a second monostatic image could be generated to create an image. Figure 7.3 depicts both bistatic and monostatic k-spaces in this scenario. The authors of [65] have also illustrated a k-space for a bistatic system where one system is stationary. This k-space is based on the bistatic angle between the stationary system and the system which rotates 2π radians around a target.

The finest range or cross-range resolution that any SAR system can create around a point target is $\Delta cr_{min} = \frac{\lambda_c}{4}$ [10], given there are no physical obstructions to the target. At X-band, 10 GHz, the minimum range resolution, Δr_{min}, is 0.75 cm. For a bistatic SAR system, where one system is stationary, the finest achievable cross-range resolution is doubled to $\Delta cr_{min} = \frac{\lambda_c}{2}$ and at X-band, 10 Ghz, this is 1.5 cm. Practically speaking, most SAR systems will only need to traverse a small arc to achieve the desired range resolution. If the minimum achievable cross-range resolution was increased by a factor of 100, a bistatic SAR system with a stationary passive receiver will only need to traverse a $3.6\,^{\circ}$ arc with no bistatic angle between the transmitter's and receiver's APC to achieve 1.5 meter resolution.

It is important to determine the length of a SAR aperture at a specific range for planning purposes. In the case of an aircraft flying, a pilot will need this distance to determine what airspace is necessary to fly the aperture. In the case of a truck mounted SAR system this will determine the length of a roadway the truck will need to image the scene. Equation (3.39) has been transformed in terms of aperture length of a single platform by setting θ_{RX}

equal to zero and assuming the platform's trajectory is not squinted. Figure 2.9 shows a squinted trajectory which means that this equation assumes that the angle θ in the figure is equal to zero.

7.3 Motion Measurement Requirements

To move one of the platforms independently will require a motion measurement system attached to the radar. The intention of this system is to measure the motion of the location where the electromagnetic wave originates, which is the phase center of the antenna. This requires a motion measurement system that will sample the position at a certain rate and with a specific amount of accuracy. The two-way monostatic Line Of Sight (LOS) phase error is given by Equation (7.1) [9].

$$\Phi_{MO_e} = \frac{4\pi r_e}{\lambda_c} \tag{7.1}$$

The term r_e is the slow time varying range error in the measurement. The one-way phase error for bistatic systems is given as Equation (7.2). A two-way monostatic phase error assumption will tighten any requirements compared to the one-way bistatic error due to the extra factor of 2 from the range error. Also, the moving system could be a monostatic radar system of a bistatic radar. This system would be used to create a second monostatic image. This means that requirements for the motion measurement system for a bistatic radar could be derived with a monostatic assumption leading to

$$\Phi_{BI_e} = \frac{2\pi r_e}{\lambda_c} \quad . \tag{7.2}$$

Assuming that the nonlinear slowly varying motion should be less than λ_c can limit phase errors to $\frac{\pi}{4}$ or 45 ° [9]. In [9], equations to specify requirements for a monostatic motion measurement system were developed. Tables 7.3 and Equation 7.2 calculate the requirements for a motion measurement system for a bistatic radar if it was mounted to a truck or an airborne platform. Comparing a land based vehicle to an airborne platform, the acceptable velocity errors are found to be an order of magnitude in $\frac{m}{s}$ and two orders of magnitude for the acceleration errors in $\frac{mm^2}{s}$. The position errors remained the samein the comparison.

Table 7.1: SAR system parameters effecting Motion Measurement Errors.

Term	Term Name	System Value
f_c	center frequency	10 GHz
λ_c	center wavelength	3 cm
V_{xa}	velocity	$15\frac{m}{s}$, Truck; $100\frac{m}{s}$, Aircraft
ρ_a	azimuth resolution	.75 m
PSR	Peak to Sidelobe Ratio	-30 dB
$ISLR$	Integrated Sidelobe Ratio	-21 dB
L_{sc}	Scene location accuracy	50 m
θ_{squint}	Squint angle	90 deg
R	Range	$32km \approx 20miles$

Motion Measurement Error	Image Effect	Allowable Motion	Aircraft Aperture	Truck Aperture
Position	Scene Center Offset	(position error) $P_e \approx L_{sc}$	50 m	50 m
LOS Velocity	Scene Center Offset	$v_{LOS_e} \approx \frac{V_{xa}\sin(\theta_{squint})}{R} L_{sc}$	$0.47\frac{mm}{s}$	$3.2\frac{mm}{s}$
Along-track velocity	Defocus	$v_{xe} \approx \frac{\rho_a^2 V_{xa}\sin^2(\theta_{squint})}{\lambda_c R}$	$8.8\frac{mm}{s}$	$58.5\frac{mm}{s}$
LOS acceleration	Defocus	$a_{LOS_e} \approx \frac{2\rho_a^2 V_{xa}^2\sin^2(\theta_{squint})}{\lambda_c R^2}$	$8\frac{mm^2}{s}$	$366\frac{mm^2}{s}$
Residual Low-frequency LOS position	Defocus	$R_e \approx \frac{\lambda_c}{16}$	$1.875mm$	$1.875mm$
Sinusoidal LOS position (range vibration magnitude)	Peak sidelobes	$a_s \approx \left(\frac{\lambda_c}{2\pi}\right)\sqrt{PSR}$	$.151mm$	$.151mm$
RMS wideband LOS position (range measurement noise power)	Loss of contrast	$\sigma_r \approx \left(\frac{\lambda_c}{4\pi}\right)\sqrt{ISLR}$	$.213mm$	$.213mm$

Table 7.2: Motion Measurement Error Requirements for the Mobile Monostatic System

7.4 Inverse Vehicle SAR

A vehicle traveling down a road travels within a lane generally. By surveying the road a priori, an estimate of the position of a car can be made using radar data. The positions of each of the bistatic radar systems will also be known. Using a range-Doppler map, the position and velocity can be estimated to determine the location of the vehicle as shown in Figure 7.4. These position estimates are used to motion compensate the vehicles radar returns for the monostatic and bistatic system. Residual position errors will be removed by an auto-focus algorithm. The final image would be an ISAR image of the vehicle.

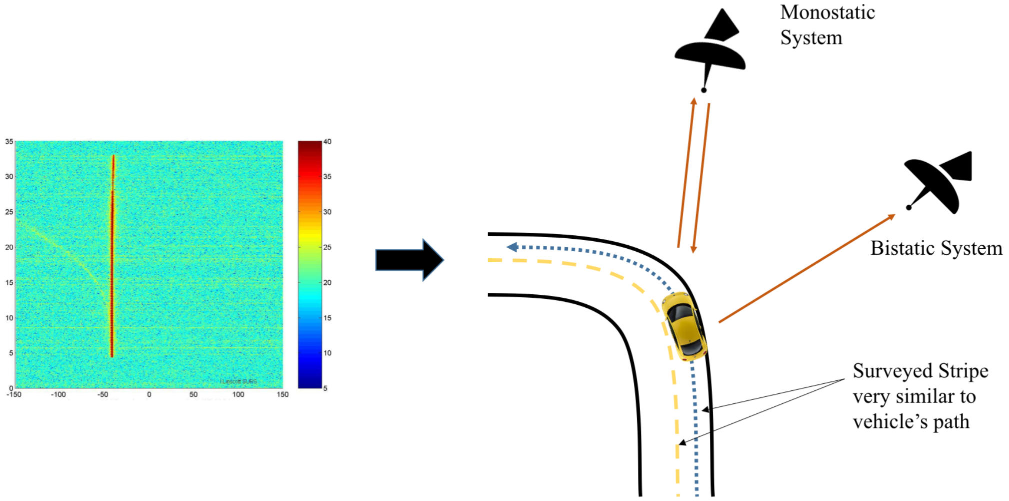

Figure 7.4: *Concept for Inverse Vehicle SAR. (Range Doppler Image Courtesy of NASA/JPL)*

7.5 Waveform Resource Management

One of the inefficiencies of SAR systems relying on a single center frequency is the polar to Cartesian conversion of the k-space. Since the k-space is captured in a polar domain and then converted to a Cartesian domain for imaging, there are unused areas of the k-space. Some radar systems can transmit LFM waveforms at various center frequencies. Some radar systems do this to increase the range resolution of the system.

Varying the center frequency of the pulse and using a priori knowledge of the imaging area allow the radar system to reduce the number of PRIs in a CPI. Waveform resource management also allows transmit at center frequencies that can provide a larger bandwidth across the k-space. When the radar is making a bistatic collection, the k-space becomes more distorted and irregular. Figure 7.5 illustrates how being adaptively varying the center frequency can cover square Cartesian k-spaces. Methods like these can help to overcome such irregularities. Most real-time SAR systems are able to predict ahead of their next position using a Kalman filter and a motion measurement system. Algorithms developed for real time optimization of the k-space in the range domain will be extremely useful.

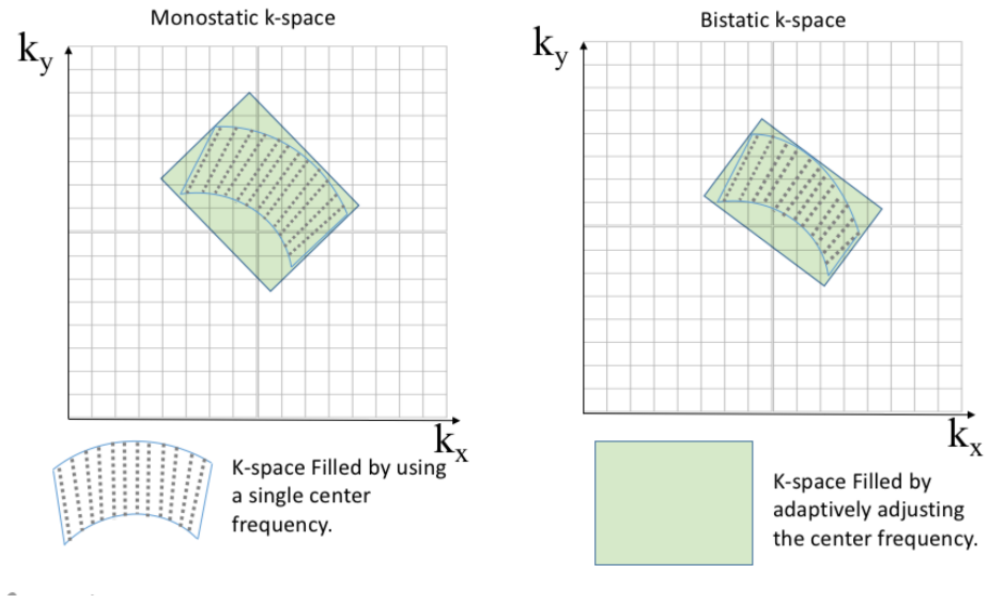
Monostatic k-space
k_y
k_x
K-space Filled by using a single center frequency.
Bistatic k-space
k_y
k_x
K-space Filled by adaptively adjusting the center frequency.

Figure 7.5: *Examples of Optimizing the transmit center frequency for a monostatic and bistatic collection.*

7.6 Introfermetic SAR (IFSAR) via a bistatic system

Both the monostatic and bistatic radar systems on a bistatic radar can be mounted to an aircraft in an introfermetic configuration. The antennas for each system will be mounted to the wings of the aircraft to create a horizontal baseline for the IFSAR systems [9]. Two motion measurement systems will need to measure the motion of each wing since each wing will move independently of the other.

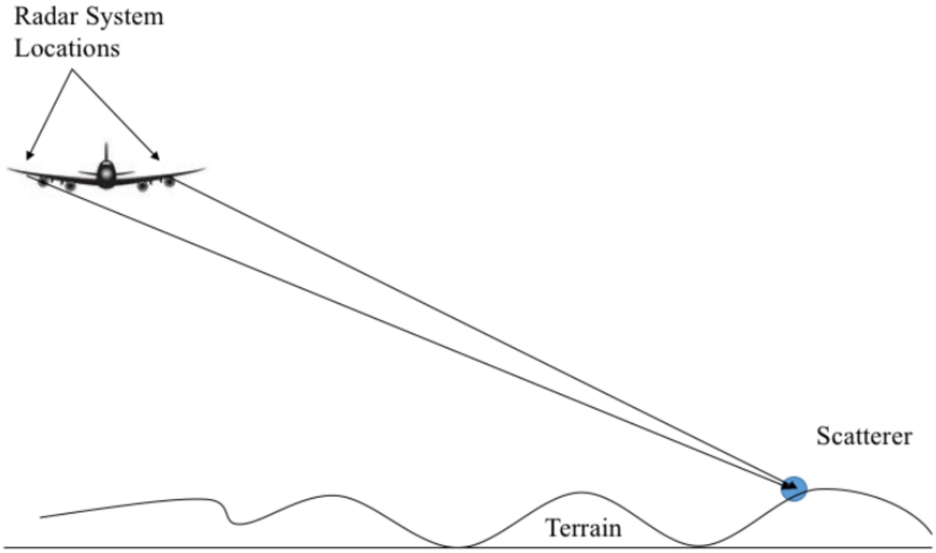

Figure 7.6: *Aircraft IFSAR diagram*

Once the data is collected, the phase changes due to the depression angle are used to determine the elevation of a specific pixel. This information is used to create a 3-D map of the terrain. This IFSAR method can also be employed on a land based platform like a truck discussed in Sections 7.2 and 7.3.

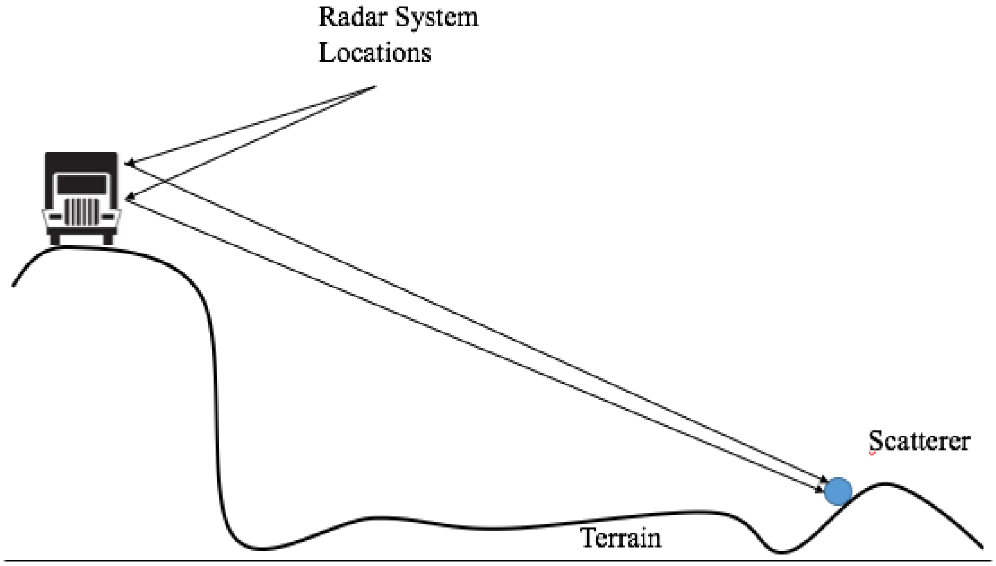

Figure 7.7: *Truck IFSAR diagram*

7.7 Adaptive Cross-Range Pulse Compression

The Adaptive Pulse Compression (APC) algorithm was designed to adaptively improve the impluse response of a matched filter. This algorithm walks through each range bin of a radar return and iteratively finds the best impulse response for the range bin [66]. Using the formulation found in [13] for the matched filtered response of a single pixel in a SAR, this algorithm can be modified to adaptively find the impulse response in the cross-range dimension. This can be done by replacing the matched filter in range with the motion compesation of a specific pixel in cross-range multiplied by the elements of the Discrete Fourier Transform vector used to determine the cell. This method can represent a new auto-focus method for SAR images that will adaptively focus each individual pixel compared to developing a compensation vector in cross-range like the Phase Gradient Autofocus (PGA) algorithm. [25]

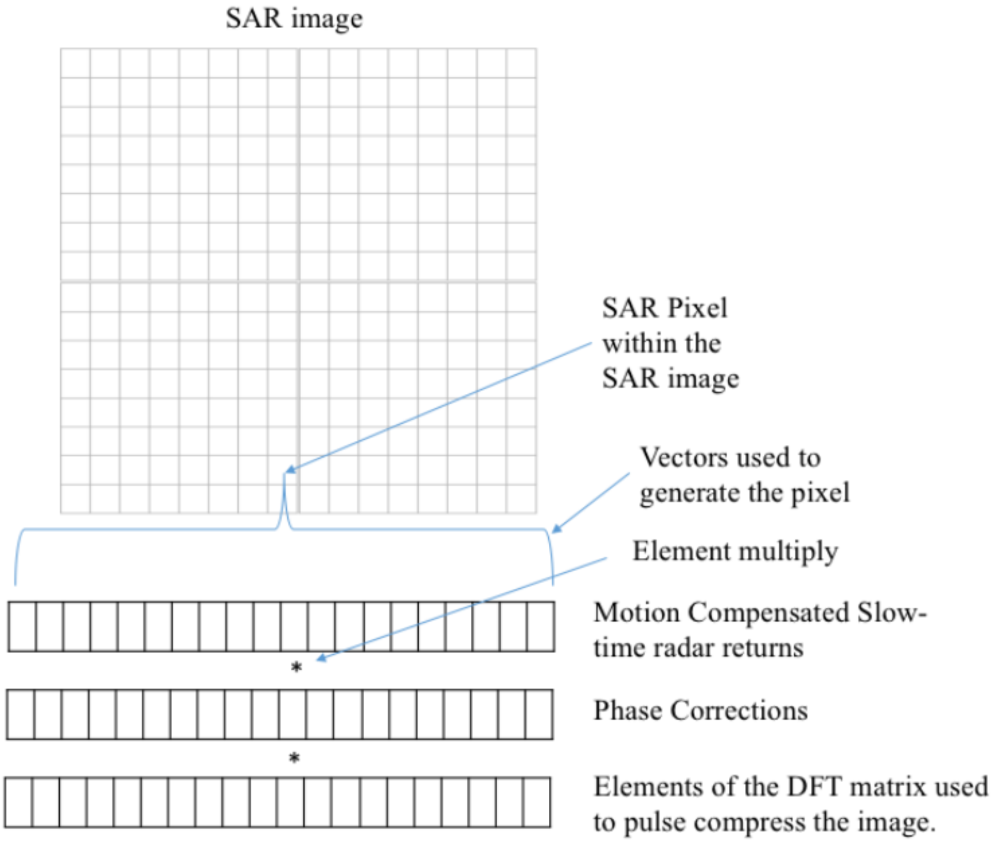
SAR image
SAR Pixel within the SAR image
Vectors used to generate the pixel
Element multiply
Motion Compensated Slow-time radar returns
*
Phase Corrections
*
Elements of the DFT matrix used to pulse compress the image.

Figure 7.8: *Illustration of Adaptive Pulse Compression in the cross-range dimension*

Chapter 8

Conclusion

This dissertation documents two significant advances in SAR from a wavenumber or k-space perspective. The author's intention was to gain a deeper understanding of this domain used in SAR imagery while advancing SAR image processing and systems. The author's initial experience working with radar imaging systems in industry, was this domain being perceived as black box that was important to understand, however, this took significant time and research in doing so. After learning some of these concepts and ideas they now seem easy and simple to understand but looking back to when the author first started working on SAR problems it seemed very difficult to know where to begin. When issues arise with production radar systems, these issues need to be solved quickly which leaves little time to dig into the deeper aspects of the radar system. The time the author spent at the University of Oklahoma researching SAR radars has given the author the ability to apply what was learned in industry and gain a deeper understanding of this topic while developing new advances in Synthetic Aperture Radar.

The two advances in SAR are discussed in detail within this dissertation. One is the generalized closed form bistatic ground range and cross-range resolution equations. These equations have a simple relationship between bandwidth and azimuth extent when the transmitter and receiver's azimuth extent is equal and equal resolution is desired in range and cross-range dimensions. These formulations for bistatic range and cross-range resolution, given in Equations (3.9), as

$$\Delta r_g = \frac{c}{BW\sqrt{cos^2(\phi_{TX}) + \cos^2(\phi_{RX}) + 2\cos(\beta) cos(\phi_{TX}) cos(\phi_{RX})}} \quad ,$$

and (3.39), as

$$\Delta cr_g = \frac{\lambda_{max}}{\sqrt{\Delta\theta_{TX}^2 \cos^2(\phi_{TX}) + \Delta\theta_{RX}^2 \cos^2(\phi_{RX}) + 2\Delta\theta_{TX}\Delta\theta_{RX} \cos(\phi_{TX}) \cos(\phi_{RX}) \cos(\beta)}} \quad ,$$

respectively, and their demonstration in simulation and laboratory testing are the major contributions of this dissertation. Separating the angular contributions of the elevation angles allows for a new forward-scatter SAR paradigm and could open the door to other forward-scatter modalities if the range resolution exists in a plane which has unequal elevation angles between the transmitter and receiver.

Both equations show a similar dependence on what contributes to finer resolution in the forward-scatter region, the elevation angles. The transmit bandwidth and the angular extent that the transmit and receive platforms travel will contribute to the magnitude of the resolution. The bistatic angle in the ground plane plays a significant role in the resolution in the ground plane as well.

The relationship that relates bandwidth to azimuth extent is Equation (3.44),

$$f_{min} = \frac{c}{\lambda_{max}} = \frac{BW}{\Delta\theta} \quad .$$

This equation is only valid in cases when equal resolution is desired in both range and cross-range which is common in most SAR applications and when the transmitter and receiver have the equal azimuth extent as well i.e. $\Delta\theta_{TX} = \Delta\theta_{RX}$, which is always the case with monostatic SAR. This relationship can be easily memorized and used quickly when a radar system engineer is mission planning a SAR collection.

The second advance in SAR is a fast BPA that exploits a stretch processed k-space. This algorithm exploits the properties of the Fourier Slice Theorem by combining neighboring pulse repetition intervals in the k-space domain. There are two significant reasons for preprocessing the radar returns this way. First it uses the native format of many high resolution SAR systems when they stretch process the receive pulse. The signal is preprocessed after a deramp step but before match filtering. This method also removes a computationally intensive interpolation step in a previously developed time-domain fast BPA found in [5] by operating in the k-space domain. Once combined, fewer radar returns are back projected speeding up the BPA. This algorithm fits in the backprojection processing chain and can be combined with other fast processing techniques like the one found in [5].

As discussed in the Introduction, this dissertation used a traditional SAR paradigm, k-space, and discovered new ways to understand SAR systems. New relationships between bistatic and monostatic SAR systems and the image resolution were discovered, and a faster method to produce radar imagery, was developed using the a newly formed in-depth understanding of the k-space domain. Ultimately, the author gained a deeper understanding of SAR systems while advancing our understanding of Synthetic Aperture Radar. The authors hopes and intentions are that this dissertation will do the same for it's readers as well.

Appendix A : Bistatic SAR Simulation

Appendix A contains the Matlab code used in the simulations of the point target model Chapter 5.

The file `Bistatic_SAR_Sim.m` is the main script that runs the simulation.

```
%Bistatic_SAR_Sim.m
%Developed by Bill Dower 2016
clear all;
clc;
% close all;

% Intialization Paramerters

%Simulation parameters
createMonostatic=0;
digitalDeramp   =0;
setRngCxRngEqual=0;

%Physical Parameters
c = 2.99792458e8; %m/s speed of light
kB= 1.38064852e-23; %J/K boltsmann's constant

% System Parameters
noNoise=1;
```

```
Fn=5; %db System Noise figure
Gs=30; %db System Gain
To=290; %k system noise temp
PRF=800; %hz
fc=10e9; %Hz
BWtx=600e6; %Hz
Tp=30e-6; %s Pulsewidth
gamma=BWtx/Tp; %Chirp Rate frequency/second
ΔR=90; %m rangeSwath

if ¬digitalDeramp
    BW=(2∗gamma∗ΔR)/c;
else
    BW=BWtx;
end

Ks=4;%unitless Oversampling;
fs=BW∗Ks; %Hz Sampling rate including over sampling
Ts=(2∗ΔR)/c; %s swath delay
Ki=8;%unitless Image Oversampling;

%Debug flags
debug.posPlots=1;
debug.targPlots=1;
debug.phaseHistPlots=0;
debug.FspacePlots=0;
debug.FspaceInterpPlots=0;
debug.KspacePlots=1;
debug.KspaceInterpPlots=1;
debug.interpolation=1;

% Target Locations
% Positon Generation scripts
targLocFnc=4;

switch targLocFnc
    case 1,
        OU_target;
    case 2,
        three_targets;
    case 3,
        two_targets;
    case 4,
        one_target;
    case 5,
        plane_Raider_Tracer;
end

%% Radar Positions
%Positon Generation scripts
posGenFnc=5;

switch posGenFnc
    case 1,
        Plat_Pos_L_Head_Center
    case 2,
        Plat_Pos_Circles
    case 3,
        Plat_Pos_Forward_Scatter
    case 4,
        Plat_Pos_Forward_Scatter_exact
    case 5,
        Plat_Pos_Forward_Scatter_changing_Angs
    case 6,
        Chamber_Forward_Scatter
end


%Position angles

if createMonostatic
    %Set all the receiver position variables from the ...
       previous function to
    %the transmitters

    Make_mono
end
%Begin Bistatic Polar Format Algorithm
create_platform_angles;

create_bistatic_Phase_History;

gen_Kspace;

%Interpolate data
yIFIP=griddata(kxP,kyP,yIF,kxIP,kyIP);
if debug.interpolation
    figure, subplot(2,2,1), imagesc(abs(yIFIP)), title('Abs ...
       interpolated'), colorbar, subplot(2,2,2), ...
       imagesc(abs(yIF)), title('Abs data'), colorbar,
    subplot(2,2,3), imagesc(angle(yIFIP)), title('Angle ...
       interpolated'), colorbar, subplot(2,2,4), ...
       imagesc(angle(yIF)), title('Angle data'), colorbar,
end

NfftY=2^nextpow2(Ki*Ny);
```

```
NfftX=2^nextpow2(Ki*Nx);

%Image Formation
SARimage=fliplr(fftshift(fft2(yIFIP, NfftY, NfftX)));
% x=(c/(Ki*Bx))*([0:Ki*Nx-1]-(Ki*Nx/2));
% y=(c/(Ki*By))*([0:Ki*Ny-1]-(Ki*Ny/2));


xs=inv((max(max(kxIP))-min(min(kxIP))))*4*pi*Nx/NfftX;
ys=inv(max(kyIP(floor(Ny/2),:))-min(kyIP(floor(Ny/2),:)))...
    *4*pi*Ny/NfftY;
y=([0:NfftY-1]-NfftY/2)*ys;
x=([0:NfftX-1]-NfftX/2)*xs;

figure, imagesc(y,x,abs(SARimage)), title('Polar Format ...
    image'), axis xy equal, xlabel('meters'), ylabel('meters')
```

The following three files, `OU_target.m`, `three_targets.m`, and `one_target.m` create the point target models used in the simulation.

```
%OU_target.m

%Template
% targNED()=[0,0,0]; targType()=1;

scaleFactor=10;

targNED(1,:)=[5,-1.1,0];    targType(1)=1;
targNED(2,:)=[5,2.1,0];     targType(2,1)=1;
targNED(3,:)=[3,-3,0];      targType(3)=1;
targNED(4,:)=[3,4,0];       targType(4)=1;
targNED(5,:)=[-2.2,-3,0];   targType(5)=1;
targNED(6,:)=[-2,4,0];      targType(6)=1;
targNED(7,:)=[-4.5,-1,0];   targType(7)=1;
targNED(8,:)=[-4.5,2.1,0];  targType(8)=1;
targNED(9,:)=[.5,-.2,0];    targType(9)=1;
targNED(10,:)=[.5,6.5,0];   targType(10)=1;
targNED(11,:)=[-6,-.2,0];   targType(11)=1;
targNED(12,:)=[-6,6.5,0];   targType(12)=1;
targNED(13,:)=[-8,1.5,0];   targType(13)=1;
targNED(14,:)=[-8,4.9,0];   targType(14)=1;


targNED(9:14,2)=targNED(9:14,2)+1;
targNED(9:14,1)=targNED(9:14,1)-1.2;

targNED=targNED-ones(size(targNED,1),1)*[-.7, .8, 0];  ...
    %centers the left top part of the U at 0,0

targNED(:,1:2) =scaleFactor*targNED(:,1:2);

sceneCenter= [0,0,0];

if exist('debug') && debug.targPlots
    figure, plot(targNED(:,2),targNED(:,1), 'b+', ...
        'LineWidth', 10), axis([-10*scaleFactor ...
        10*scaleFactor -15*scaleFactor 10*scaleFactor]), ...
        axis('square')
    ylabel('North in meters'), xlabel('East in meters')
    if scaleFactor==10
         axis([-46.4912   79.2398 -97.0029   66.7398])
    end
end
```

```
%three_targets.m

%Template
% targNED()=[0,0,0]; targType()=1;

scaleFactor=8;

targNED(1,:)=[ 1.5, 0, 0];       targType(1)=1;
%targNED(2,:)=[ 0, 0,0];       targType(2,1)=1;
targNED(2,:)=[-1.5, 0, 0];      targType(2)=1;


targNED(:,1:2) =scaleFactor*targNED(:,1:2);

sceneCenter= [0,0,0];

if exist('debug') && debug.targPlots
    figure, plot(targNED(:,2),targNED(:,1), 'bx', ...
        'LineWidth', 4), axis([-7*scaleFactor 7*scaleFactor ...
        -7*scaleFactor 7*scaleFactor]), axis('square')
    ylabel('North in meters'), xlabel('East in meters')
end
```

```
%one_target.m

%Template
% targNED()=[0,0,0]; targType()=1;

```

```
scaleFactor=8;

targNED(1,:)=[ 0, 0, 0];       targType(1)=1;



targNED(:,1:2) =scaleFactor*targNED(:,1:2);

sceneCenter= [0,0,0];

if exist('debug') && debug.targPlots
    figure, plot(targNED(:,2),targNED(:,1), 'bx'), ...
        axis([-7*scaleFactor 7*scaleFactor -7*scaleFactor ...
        7*scaleFactor]), axis('square')
end
```

The transmitter and receiver locations are generated in the function, `Plat_Pos_L_Head_Center.m`.

```
%Plat_Pos_L_Head_Center.m

%Platform Positions based given the follwoing inputs:
%Aperture Lengths, heading and aperture center point


headTx= 90; %heading in degrees
headRx= 45; %heading in degrees

lTx=2e2;  %transmit aperture length m
lRx=2e2;  %receive  aperture length m

```

```
txCentNED = [-1.5e3; 0; -5e3];       %Transmit aperture ...
    center position NED
rxCentNED = [-1e3;   1e3; -10e3];    %Receive  aperture ...
    center position NED

velTx = 100; %m/s
velRx = 125; %m/s

Np = max(floor(PRF*lTx/velTx), floor(PRF*lRx/velRx));% ...
    number of Pulses

if mod(Np,2)
    t=[-Np/2:1:Np/2]/PRF;
else
    t=[-(Np-1)/2:1:(Np-1)/2]/PRF;
end

txPosNed=[velTx*cosd(headTx); velTx*sind(headTx); ...
    0]*t+txCentNED*ones(size(t));
rxPosNed=[velRx*cosd(headRx); velRx*sind(headRx); ...
    0]*t+rxCentNED*ones(size(t));

rTx=sqrt(sum((txPosNed-sceneCenter.'*ones(1,Np)).^2));
rRx=sqrt(sum((rxPosNed-sceneCenter.'*ones(1,Np)).^2));


if debug.posPlots
    figure, plot(txPosNed(2,:), txPosNed(1,:), 'b', ...
        'LineWidth', 4), hold on
    plot(rxPosNed(2,:), rxPosNed(1,:), 'g', 'LineWidth', 4)
    plot(sceneCenter(2), sceneCenter(1), 'rx', 'LineWidth', ...
        4), hold off
    ylabel('North (m)')
    xlabel('East (m)')
    ylim([-1600 100])
    legend('Transmitter position', 'Receiver Position', ...
        'Scene Center Point, SRP')

    figure, plot(t, rTx, t, rRx)
    ylabel('Range (m)')
    xlabel('time (sec)')

    figure, plot(t, rTx-mean(rTx), t, rRx-mean(rRx))
    ylabel('Range minus mean (m)')
    xlabel('time (sec)')

    figure, plot(t(1:end-1), diff(rTx), t(1:end-1), diff(rRx))
    ylabel('diff of Range (m)')
    xlabel('time (sec)')
end
```

The function Make_mono.m sets the position of the receiver equal to the transmitter's position to create a monostatic flight path if this function is called in the Bistatic_SAR_Sim.m script.

```
%Make_mono.m

%Sets receiver's position equal to the transmitters




%headRx= headTx; %heading in degrees
%lRx=lTx;  %receive  aperture length m
```

```
rxCentNED = txCentNED;    %Receive  aperture center ...
   position NED
%velRx = velTx; %m/s
rxPosNed=txPosNed;
rRx=rTx;


if debug.posPlots
    figure, plot(txPosNed(2,:), txPosNed(1,:), ...
        rxPosNed(2,:), rxPosNed(1,:), sceneCenter(2), ...
        sceneCenter(1), 'rx')
    ylabel('North (m)')
    xlabel('East (m)')
    title('Make mono')

    if exist('t')
    figure, plot(t, rTx, t, rRx)
    ylabel('Range (m)')
    xlabel('time (sec)')
    title('Make mono')

    figure, plot(t, rTx-mean(rTx), t, rRx-mean(rRx))
    ylabel('Range minus mean (m)')
    xlabel('time (sec)')
    title('Make mono')

    figure, plot(t(1:end-1), diff(rTx), t(1:end-1), diff(rRx))
    ylabel('diff of Range (m)')
    xlabel('time (sec)')
    title('Make mono')
    end
end
```

The function `create_bistatic_Phase_History.m` creates the phase histories for the SAR signal.

```
%create_bistatic_Phase_History.m
%This function creates the phase histories of the bistatic ...
    signal

Mt=size(targNED,1);
K=floor((Tp+Ts)*fs);
K=K+mod(K,2); %make K an even number

yIF=zeros(K,Np);

to=ones(K,1)*(rRx+rTx)/c;

k=[1:K].';
n=1:Np;

for iM=1:Mt
    rTxd=sqrt(sum((txPosNed-targNED(iM,:).'*ones(1,Np)).^2));
    rRxd=sqrt(sum((rxPosNed-targNED(iM,:).'*ones(1,Np)).^2));
    td=ones(K,1)*(rRxd+rTxd)/c;
    tHat=((k/fs)-((Ts+Tp)/2))*ones(1,Np)+to;

    posTx=(txPosNed-targNED(iM,:).'*ones(1,Np));
    posRx=(rxPosNed-targNED(iM,:).'*ones(1,Np));

    thetaRxTgt=atan2(posRx(1,:), posRx(2,:));
    thetaTxTgt=atan2(posTx(1,:), posTx(2,:));

    phiRxTgt=acos(-1*posRx(3,:)./sqrt(sum(posRx.^2,1)));
    phiTxTgt=acos(-1*posTx(3,:)./sqrt(sum(posTx.^2,1)));

    if digitalDeramp & targLocFnc>3
        disp('This part may not be right')
        fo = fc*(td-to); fo=fo(:,1); %Need to keep this ...
           hack in mind clean this up
        at=sqrt(targetRCS(targType(iM), fo, thetaTxTgt, ...
           phiTxTgt, thetaRxTgt, phiRxTgt));
    elseif length(targType)==1 & targLocFnc>3
        fo = fc+gamma*tHat-gamma*to; fo=fo(:,1);  %Need to ...
           keep this hack in mind clean this up
        at=sqrt(targetRCS(targType(iM), fo, thetaTxTgt, ...
           phiTxTgt, thetaRxTgt, phiRxTgt));
    else
        at=sqrt(targetRCS(targType(iM)));
    end


    if digitalDeramp
        yIF=yIF+at.*rect(tHat,td,Tp).*exp(-j*2*pi*fc*(td-to))...
            .*exp(j*pi*gamma*(tHat-td).^2);
    else
        yIF=yIF+at.*rect(tHat,td,Tp)...
            .*exp(-j*2*pi*(fc+gamma*tHat-gamma*to)...
            .*(td-to)).*exp(j*pi*gamma*(td-to).^2);
    end
end



%Zero-mean Guassian Noise from reciever.
```

```
Pn=kB*To*fs*10^((Fn*Gs)/10);
ni = normrnd(0, sqrt(Pn), [K, Np]);
nq = normrnd(0, sqrt(Pn), [K, Np]);

if ¬noNoise
    yIF= yIF + ni + j*nq;
end

clear ni nq

if debug.phaseHistPlots
    r=(2*c/(BW))*([0:K-1]-(K/2));
    figure, imagesc(r, [], 20*log10(abs(fftshift(ifft(yIF, ...
        [],1),1)))),
    ylabel('Range in meters'), xlabel('Pulse number within ...
        the PRI'), title('Radar Returns Pulse Compressed ...
        Magnitude in dB')

    figure, imagesc(r, [], angle(fftshift(ifft(yIF, ...
        [],1),1))), title('Radar Returns complex phase')
    ylabel('Range in meters'), xlabel('Pulse number within ...
        the PRI')
end

if digitalDeramp
    yIF=yIF.*exp(-j*pi*gamma*(tHat-to).^2);

    if debug.phaseHistPlots
        figure, imagesc(abs(yIF)), title('Uncompenstated ...
            Phase Histories')
        figure, imagesc(abs(fftshift(ifft(yIF,[],1),1))), ...
            title('Frequency Domain Phase Histories')
    end
end
```

The function `create_platform_angles.m` finds the angles from the SRP to the platforms used by the k-space generation function.

```
%create_platform_angles.m

thetaRx=atan2(rxPosNed(1,:), rxPosNed(2,:));
thetaTx=atan2(txPosNed(1,:), txPosNed(2,:));

phiRx=acos(-1*rxPosNed(3,:)./sqrt(sum(rxPosNed.^2,1)));
phiTx=acos(-1*txPosNed(3,:)./sqrt(sum(txPosNed.^2,1)));
```

The function `gen_Kspace.m` creates the k-spaces and frequency spaces needed to Cartesian format the polar formatted data.

```
%gen_Kspace.m

fi=[-BWtx/2:(BWtx/(K-1)):BWtx/2];
k=(4*pi*(fc+fi.')/c);
f=fc+fi.';

kx= (k/2)*(cos(thetaTx).*cos(phiTx)+cos(thetaRx).*cos(phiRx));
ky= (k/2)*(sin(thetaTx).*cos(phiTx)+sin(thetaRx).*cos(phiRx));
kz= (k/2)*(sin(phiTx)+sin(phiRx));

thetaB=atan2(ky, kx);

ACPthetaB=mean(thetaB(:, floor(end/2)));
```

```
ACPPhiB  =mean(phiRx+phiTx)/2;

%disp(['The Bistatic angle for the aperture center is ', ...
    num2str((ACPthetaB)*180/pi), ' degrees.'])

kxP= kx*cos(ACPthetaB)+ky*sin(ACPthetaB);
kyP=-kx*sin(ACPthetaB)+ky*cos(ACPthetaB);

if debug.KspacePlots
figure, plot(kx, ky, 'g*'), title('Ground Plane Kspace'), ...
    axis equal
figure, plot3(kx, ky, kz, 'g*'), title('Kspace'), axis equal
figure, plot(kxP, kyP, 'g*'), title('Rotated Ground Plane ...
    Kspace'), axis equal
end

By = (c/(4*pi))*2*min(abs([kyP(end,1), kyP(end,end), ...
    kyP(1,1), kyP(1,end)]));
% Bx = (c/(4*pi))*min(abs(kxP(1,end)-kxP(end,end)),...
% abs(kxP(1,1)-kxP(end,1)));
Bx = (c/(4*pi))*(min(kxP(end,:))-max(kxP(1,:)));

if setRngCxRngEqual
    Bx = min(By, Bx);
    By = Bx;
end

%Determine kspace slope
slope=[(kxP(end,end)-kxP(end,1))/(kyP(end,end)-kyP(end,1)),...
     (kxP(1,end)-kxP(1,1))/(kyP(1,end)-kyP(1,1))];

[¬,idxSlope]=min(abs(slope));
```

```

slope = slope(idxSlope); clear idxSlope;

Ny=min(size(yIF));
Nx=Ny;

nx=[0:Nx-1].';
ny=[0:Ny-1];

% kcIP=mean((kxP(1,:)+kxP(end,:))/2);
kcIP=mean((min(kxP(end,:))+max(kxP(1,:)))/2);

kyIP = (4*pi)*repmat(-By/2 + By*ny/Ny, [Nx,1])/c;
kxIP = kcIP+ (4*pi)*repmat(Bx*nx/Nx -Bx/2, [1,Ny])/c + ...
    slope*kyIP;

if debug.KspaceInterpPlots
figure, plot(kxIP, kyIP, 'r*'), title('Rotated Ground Plane ...
    Kspace Interpolated points'), axis equal
figure, plot(kxP, kyP, 'b*'), hold on, plot(kxIP, kyIP, ...
    'r*'), hold off, title('Rotated Ground Plane Kspace and ...
    Interpolated points'), axis equal

figure, subplot(2,2,1), imagesc(kxP), colorbar, ...
    title('Kx'), subplot(2,2,2), imagesc(kxIP), colorbar, ...
    title('Kx Interpolated')
        subplot(2,2,3), imagesc(kyP), colorbar, ...
            title('Ky'), subplot(2,2,4), imagesc(kyIP), ...
            colorbar, title('Ky Interpolated')
end
```

The function `targetRCS.m` is a simple point target model that can be expanded for future target modeling.

```
%targetRCS.m
function [ RCS] = targetRCS( targetType, fo, thetaT, phiT, ...
    thetaR, phiR)
%This function creates the RCS of the target

addpath /home/user/matlab /home/user/matlab/test
if targetType==1
    RCS=1;
end
if targetType==2
    %Path to the p files
    addpath('/Users/bill/Documents/Research/matlab/Raider ...
        Tracer/From Rigling/');
    theta0 = 0*(pi/180); %Target azimuth rotation
    phi0 = 0*(pi/180);   %Target elevation rotation
    fname = 'dihedral.facet';
    %fname = 'from_web_simple_airplane.facet';
    maxBounce = 5;
    nLevels = 3;

    [rayList] = RaiderTracerIncident...
        (fname,maxBounce,nLevels,thetaT+theta0,phiT+phi0);
    [RCS]     = RaiderTracerReturned...
        (fname,nLevels,fo,rayList,thetaR+theta0,phiR+phi0,1);
end
end
```

Appendix B : Decimated Wavenumber Processing Scripts

Appendix B contains the Matlab code used for testing and processing the Decimated Wavenumber fast BPA.

The file `processDataBP.m` is the main script that runs the image processing algorithms.

```
%Physical Parameters
kB= 1.38064852e-23; %J/K boltsmann's constant
c = 2.99792458e8; %m/s speed of light

prompt = {'Enter figure title:','Enter figure filename:'};
dlg_title = 'Figure names (don''t change for generic ...
    responses)';
num_lines = [1,70; 1,70];
defaultans = {'','BPimage'};
figureData = inputdlg(prompt,dlg_title,num_lines,defaultans);
if size(figureData,1)==0
    return;
    figureData= {'';'BPimage'};
end

if ¬exist('phdata') | size(phdata,2)≠3e4
    loadAllData
end
```

```


tic;
if 0
    mocompData;
else
    AntXsmooth = AntX;
    AntYsmooth = AntY;
    AntZsmooth = AntZ;
end

%Rotates the image so that the fight path is parallel to ...
    the x axis of the
%image
dShift=-90-atan2d(AntYsmooth(end/2), AntXsmooth(end/2));
AntXhold=AntXsmooth;
AntXsmooth = AntXsmooth*cosd(dShift)-AntYsmooth*sind(dShift);
AntYsmooth = AntXhold  *sind(dShift)+AntYsmooth*cosd(dShift);
clear AntXhold

ps=3;%[meters] pixel spacing

%Image sizes
ib=[-2.5e3, 2.5e3, -2.5e3, 2.5e3];%[meters] (xmin, xmax, ...
    ymin, ymax) image boundaries

Nx=floor((ib(2)-ib(1))/ps); %number of image pixels in the ...
    X direction
Ny=floor((ib(4)-ib(3))/ps); %number of image pixels in the ...
    y direction


```

```
if realData
    fc=minF+∆F*(nSamp/2); %Further verify if the -.5 is ...
        needed this says its not (2*minF+∆F*nSamp)/2
else
    nSamp=K;
    ∆F=BWtx*Ks/(nSamp-1);
end

%Remove Residual Video Phase
if 0
    Ns=size(phdata,1);
    nsdiv2=ceil(Ns/2);
    fTHat=[-nsdiv2+mod(Ns,2):-1, 0:nsdiv2-1].'*∆F;
    BW=∆F*Ns;
    Tp=.1e-6;
    RVPcomp=single(exp(-1i*pi*fTHat.^2*Tp/BW))...
        *ones(1,size(phdata,2),'single');
else
    RVPcomp=ones(size(phdata),'single');
end


BPver=12;

if BPver==1
    IM=        backProjectData(ps, ib,  AntXsmooth, ...
        AntYsmooth, AntZsmooth, phdata.*RVPcomp, fc, ∆F);
elseif BPver==2
    error('This function was not completed')
    IM=        backProjectCuda(ps, ib, AntXsmooth, ...
        AntYsmooth, AntZsmooth, phdata.*RVPcomp, fc, ∆F);
elseif BPver==3
```

```
    IM=backProjectDataNoParFor(ps, ib,  AntXsmooth, ...
       AntYsmooth, AntZsmooth, phdata.*RVPcomp, fc, ∆F);
elseif BPver==4
    IM=     backProjectDataGPU(ps, ib,  AntXsmooth, ...
       AntYsmooth, AntZsmooth, phdata.*RVPcomp, fc, ∆F);
elseif BPver==5
    IM= fastBackProjectDataGPU(ps, ib,  AntXsmooth, ...
       AntYsmooth, AntZsmooth, phdata.*RVPcomp, fc, ∆F);
elseif BPver==6
    IM= preprojectBackProjectDataGPU2(ps, ib,  AntXsmooth, ...
       AntYsmooth, AntZsmooth, phdata.*RVPcomp, fc, ∆F);
elseif BPver==7
%The following BPver elseifs are decimated wavenumber ...
   processing methods
    IM=     backProjectDataGPU(ps, ib,  ...
       (AntXsmooth(1:2:end)+AntXsmooth(2:2:end))/2, ...
            (AntYsmooth(1:2:end)+AntYsmooth(2:2:end))/2, ...
            (AntZsmooth(1:2:end)+AntZsmooth(2:2:end))/2, ...
            (phdata(:,1:2:end)+phdata(:,2:2:end))...
            .*RVPcomp(:,1:2:end)/2, fc, ∆F);
elseif BPver==8

    IM=     backProjectDataGPU(ps, ib, ...
            (AntXsmooth(1:3:end)+AntXsmooth(2:3:end)...
            +AntXsmooth(3:3:end))/3, ...
            (AntYsmooth(1:3:end)+AntYsmooth(2:3:end)...
            +AntYsmooth(3:3:end))/3, ...
            (AntZsmooth(1:3:end)+AntZsmooth(2:3:end)...
            +AntZsmooth(3:3:end))/3, ...
            (phdata(:,1:3:end)+phdata(:,2:3:end)...
            +phdata(:,3:3:end)).*RVPcomp(:,1:3:end)/3, ...
            fc, ∆F);
```

```
elseif BPver==9

    IM=     backProjectDataGPU(ps, ib, ...
            (AntXsmooth(1:4:end)+AntXsmooth(2:4:end)...
            +AntXsmooth(3:4:end)+AntXsmooth(4:4:end))/4, ...
            (AntYsmooth(1:4:end)+AntYsmooth(2:4:end)...
            +AntYsmooth(3:4:end)+AntYsmooth(4:4:end))/4, ...
            (AntZsmooth(1:4:end)+AntZsmooth(2:4:end)...
            +AntZsmooth(3:4:end)+AntZsmooth(4:4:end))/4, ...
            (phdata(:,1:4:end)+phdata(:,2:4:end)...
            +phdata(:,3:4:end)...
            +phdata(:,4:4:end)).*RVPcomp(:,1:4:end)/4, ...
            fc, ∆F);
elseif BPver==10

    IM=     backProjectDataGPU(ps, ib, ...
            (AntXsmooth(1:5:end)+AntXsmooth(2:5:end)...
            +AntXsmooth(3:5:end)...
            +AntXsmooth(4:5:end)+AntXsmooth(5:5:end))/5, ...
            (AntYsmooth(1:5:end)+AntYsmooth(2:5:end)...
            +AntYsmooth(3:5:end)...
            +AntYsmooth(4:5:end)+AntYsmooth(5:5:end))/5, ...
            (AntZsmooth(1:5:end)+AntZsmooth(2:5:end)...
            +AntZsmooth(3:5:end)...
            +AntZsmooth(4:5:end)+AntZsmooth(5:5:end))/5, ...
            (phdata(:,1:5:end)+phdata(:,2:5:end)...
            +phdata(:,3:5:end)...
            +phdata(:,4:5:end)+phdata(:,5:5:end))...
            .*RVPcomp(:,1:5:end)/5,fc, ∆F);

elseif BPver==11

```

```
    IM=     backProjectDataGPU(ps, ib, ...
            (AntXsmooth(1:6:end)+AntXsmooth(2:6:end)...
            +AntXsmooth(3:6:end)+AntXsmooth(4:6:end)...
            +AntXsmooth(5:6:end)+AntXsmooth(6:6:end))/6, ...
            (AntYsmooth(1:6:end)+AntYsmooth(2:6:end)...
            +AntYsmooth(3:6:end)+AntYsmooth(4:6:end)...
            +AntYsmooth(5:6:end)+AntYsmooth(6:6:end))/6, ...
            (AntZsmooth(1:6:end)+AntZsmooth(2:6:end)...
            +AntZsmooth(3:6:end)+AntZsmooth(4:6:end)...
            +AntZsmooth(5:6:end)+AntZsmooth(6:6:end))/6, ...
            (phdata(:,1:6:end)+phdata(:,2:6:end)...
            +phdata(:,3:6:end)+phdata(:,4:6:end)...
            +phdata(:,5:6:end)+phdata(:,6:6:end))...
            .*RVPcomp(:,1:6:end)/6,fc, ∆F);
elseif BPver==12
    IM=     backProjectDataDebug(ps, ib,  AntXsmooth,...
        AntYsmooth, AntZsmooth, phdata.*RVPcomp,...
        fc, ∆F);
end

toc;

figure, imagesc([ib(1)+ps/2:ps:ib(2)-ps/2], ...
    [ib(3)+ps/2:ps:ib(4)-ps/2], sqrt(abs(fliplr(IM)))),
title(figureData{1},'Color','white');
set(gcf,'color','black');
set(gca,'Xcolor','white');
set(gca,'Ycolor','white');
colormap('gray'), axis xy;% equal;
xlim([ib(1), ib(2)]); ylim([ib(3), ib(4)]);
saveas(gcf, [figureData{2}, '.fig']);
```

The file `backProjectDataGPU.m` backprojects the data into an image using GPUs.

```matlab
function [IM]=backProjectDataGPU(ps, ib, AntX, AntY, AntZ, ...
    phdata, fc, ΔF, Rcorrect)

if ib(1)>ib(2) || ib(3)>ib(4)
    error('Image boundry min is greater than image boundry ...
        max.')
end

c =299792458; %[m/s] speed of light;
lambda=c/fc;
k = 100; %oversample factor
st = 1/ΔF;% swath time
rngSwath = st*c/2;
Re=6371008.8; %Average Radius of the Earth in meters

Ns = size(phdata, 1)*k;
Np = size(phdata, 2);

Nx=floor((ib(2)-ib(1))/ps); %number of image pixels in the ...
    X direction
Ny=floor((ib(4)-ib(3))/ps); %number of image pixels in the ...
    y direction

IM=single(zeros(Nx,Ny));
subIM=IM;

pixRng = zeros(Nx,Ny);
pixNumX = repmat(1:Nx, Ny, 1);
```

```
pixNumY = repmat((1:Ny).', 1, Nx);

pixPosX = (pixNumX-(Nx+1)/2)*ps + ib(1) + (ib(2)-ib(1))/2;
pixPosY = (pixNumY-(Ny+1)/2)*ps + ib(3) + (ib(4)-ib(3))/2;

if 0
    dSRP    =  sqrt (pixPosX.^2+pixPosY.^2);
    pixPosZ =  -Re+sqrt(Re^2-dSRP.^2);
else
    pixPosZ = zeros(size(pixPosX));
end

R0 = sqrt(AntX.^2 + AntY.^2 + AntZ.^2);

rngSwath = st*c/2;
Ns = size(phdata, 1)*k;
rngVal =[-rngSwath/2:rngSwath/(Ns-1):rngSwath/2];


Ns = size(phdata, 1)*k;

%Variables to transfer to the Parallel Processors
%delete(gcp('nocreate'))

pPX     = gpuArray(pixPosX);
pPY     = gpuArray(pixPosY);
pPZ     = gpuArray(pixPosZ);

%%%%%%!!!!!! You changed this !!!!!!!!!!!!!!!
AX      = gpuArray(AntX);
AY      = gpuArray(AntY);
AZ      = gpuArray(AntZ);
```

```

pd       = gpuArray(phdata);
rV       = gpuArray(rngVal);
rS       = gpuArray(rngSwath);
Ro       = gpuArray(R0);
ns       = gpuArray(Ns);
nsp1     = ns+1;
nsdiv2   = gpuArray(ceil(Ns/2));
IMgpu    = gpuArray(zeros(Nx,Ny,'single'));
ks       = gpuArray(-1i*4*pi/lambda);
rngSp    = rS/(ns-1); %Range Pixel Spacing
% pPX = pixPosX;
% pPY = pixPosY;
%
% AX = AntX;
% AY = AntY;
% AZ = AntZ;
%
% pd = phdata;
% rV = rngVal;
% rS = rngSwath;
% Ro = R0;
% ns = Ns;
%R0 = sqrt(AntX.^2 + AntY.^2 + AntZ.^2);
for ind=1:Np
    %pixRng = round((sqrt((pixPosX-AntX(ind)).^2 + ...
        (pixPosY-AntY(ind)).^2 + ...
        (AntZ(ind))^2)-(R0(ind)-rngSwath/2))/(rngSwath/Ns));
    rangeProfile=fftshift(fft(pd(:,ind), ns));
    pixRng    = sqrt((pPX-AX(ind)).^2 + (pPY-AY(ind)).^2 + ...
        (pPZ-AZ(ind)).^2);
%     pixRngNum = int32(round((sqrt((pPX-2*AX(ind)).^2 + ...
    (pPY-2*AY(ind)).^2 + (2*AZ(ind))^2)-rS/2)/rngSp));

    pixRngNum = int32(round((pixRng-(Ro(ind)-rS/2))/rngSp));
%    pixRngNum = int32(round((sqrt((pPX-AX(ind)).^2 + ...
    (pPY-AY(ind)).^2)+rS/2)/rngSp));

%     pixRngOut=gather(pixRng);
%     pixRngNumOut=gather(pixRngNum);
%     keyboard;
%     pixRngNum(pixRngNum > ns || pixRngNum <  1)=ns+1;

%     indexZero=[find(pixRngNum > ns); find(pixRngNum <  1)];

    indexZero=[(pixRngNum > ns) | (pixRngNum <  1)];

    rangeProfile(nsp1)=0;
    pixRngNum(indexZero)=nsp1;

%     subIM = ...
    rangeProfile(pixRngNum).*exp(ks*(pixRng-Ro(ind)));
    subIM = rangeProfile(pixRngNum).*exp(ks*pixRng);
    IMgpu = IMgpu + subIM;
end
%delete(gcp('nocreate'))

IM = gather(IMgpu);

end
```

The file `backProjectData.m` backprojects the data into an image using the parallel processing toolbox.

```
function [IM]=backProjectData(ps, ib, AntX, AntY, AntZ, ...
    phdata, fc, ∆F, Rcorrect)

if ib(1)>ib(2) || ib(3)>ib(4)
     error('Image boundry min is greater than image boundry ...
         max.')
end

c =299792458; %[m/s] speed of light;
lambda=c/fc;
k = 100; %oversample factor
st = 1/∆F;% swath time
rngSwath = st*c/2;
Ns = size(phdata, 1)*k;
Np = size(phdata, 2);

Nx=floor((ib(2)-ib(1))/ps); %number of image pixels in the ...
    X direction
Ny=floor((ib(4)-ib(3))/ps); %number of image pixels in the ...
    y direction

IM=single(zeros(Nx,Ny));
subIM=IM;

pixRng = zeros(Nx,Ny);
pixNumX = repmat(1:Nx,    Ny, 1);
pixNumY = repmat([1:Ny].', 1, Nx);

pixPosX = (pixNumX-(Nx+1)/2)*ps + ib(1) + (ib(2)-ib(1))/2;
pixPosY = (pixNumY-(Ny+1)/2)*ps + ib(3) + (ib(4)-ib(3))/2;

```

```
R0 = sqrt(AntX.^2 + AntY.^2 + AntZ.^2);

rngSwath = st*c/2;
Ns = size(phdata, 1)*k;
rngVal =[-rngSwath/2:rngSwath/(Ns-1):rngSwath/2];


Ns = size(phdata, 1)*k;

%Variables to transfer to the Parallel Processors
%delete(gcp('nocreate'))

pPX = parallel.pool.Constant(pixPosX);
pPY = parallel.pool.Constant(pixPosY);

AX = parallel.pool.Constant(AntX);
AY = parallel.pool.Constant(AntY);
AZ = parallel.pool.Constant(AntZ);

pd = parallel.pool.Constant(phdata);
rV = parallel.pool.Constant(rngVal);
rS = parallel.pool.Constant(rngSwath);
Ro = parallel.pool.Constant(R0);
ns = parallel.pool.Constant(Ns);


% pPX = pixPosX;
% pPY = pixPosY;
%
% AX = AntX;
% AY = AntY;
% AZ = AntZ;
```

```
%
% pd = phdata;
% rV = rngVal;
% rS = rngSwath;
% Ro = R0;
% ns = Ns;

parfor ind=1:Np
    %pixRng = round((sqrt((pixPosX-AntX(ind)).^2 + ...
        (pixPosY-AntY(ind)).^2 + ...
        (AntZ(ind))^2)-(R0(ind)-rngSwath/2))/(rngSwath/Ns));
    rangeProfile=[single(fftshift(fft(pd.Value(:,ind), ...
        ns.Value))); 0];
    pixRngNum = round((sqrt((pPX.Value-AX.Value(ind)).^2 + ...
        (pPY.Value-AY.Value(ind)).^2 + ...
        (-AZ.Value(ind))^2)-(Ro.Value(ind)-rS.Value/2))...
        /(rS.Value/(ns.Value-1)));
    pixRng    =        sqrt((pPX.Value-AX.Value(ind)).^2 + ...
        (pPY.Value-AY.Value(ind)).^2 + (-AZ.Value(ind))^2);

    pixRngNum(pixRngNum > ns.Value | pixRngNum <  ...
        1)=ns.Value+1;

    %subIM = rangeProfile(pixRng);

    subIM = ...
        rangeProfile(pixRngNum).*exp((-j*4*pi/lambda)*pixRng);
    IM = IM + subIM;
end
delete(gcp('nocreate'))

end
```

The file `loadAllData.m` loads all of the radar data.

```
path=uigetdir([cd, filesep, '..'], 'Select Data Directory');
tic
% if exist('subDataAll.mat','file')
%     disp('loading subDataAll.mat.....')
%     load('subDataAll.mat');
% else
D=dir('subData*.mat');

for indx=1:length(D)
    load(D(indx).name);

    F = fieldnames(subData);
    for ind=1:length(F),
        if indx==1
            eval([F{ind}, '=subData.', F{ind}, ';']);
        else
            eval([F{ind}, '=[', F{ind}, ',subData.', ...
                F{ind}, '];']);
        end
    end,
    clear subData ans;
end
clear D F ind indx

nSamp=size(phdata,1);

AntX=reshape(AntX, [size(AntX,1)*size(AntX,2), 1]);
AntY=reshape(AntY, [size(AntY,1)*size(AntY,2), 1]);
```

```
AntZ=reshape(AntZ, [size(AntZ,1)*size(AntZ,2), 1]);
R0=reshape(R0, [size(R0,1)*size(R0,2), 1]);


if length(unique(∆F))==1
    ∆F=∆F(1);
end

if length(unique(minF))==1
    minF=minF(1);
end

phdata = single(phdata);

%     save('subDataAll.mat', '-v7.3')
%     disp('Saved subDataAll.mat')
% end
toc
```

The file `mocompData.m` motion compensates the data to a circular flight path. This is important.

```

rA=10.4994e3; %meters
thetaA=44.341; %degrees
phiT=3.322; %degrees

Na=length(AntX);
phiTvec=[-(Na)/2:Na/(Na-1):(Na)/2].'*(phiT/Na);

AntXsmooth=rA*cosd(thetaA)*cosd(phiTvec);
```

```
AntYsmooth=rA*cosd(thetaA)*sind(phiTvec);
AntZsmooth=rA*sind(thetaA)*ones(size(phiTvec));
```

References

[1] J. L. Walker, *Range-Doppler Imaging of Rotating Objects*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-16, no. 1, pp. 23–52, Jan 1980.

[2] A. Love, *In memory of Carl A. Wiley*, *Antennas and Propagation Society Newsletter, IEEE*, vol. 27, no. 3, pp. 17–18, June 1985.

[3] M. Soumekh, *Synthetic aperture radar signal processing.* New York: Wiley, 1999, vol. 7.

[4] A. Moccia and A. Renga, *Spatial Resolution of Bistatic Synthetic Aperture Radar: Impact of Acquisition Geometry on Imaging Performance*, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 10, pp. 3487–3503, Oct 2011.

[5] L. Ulander, H. Hellsten, and G. Stenstrom, *Synthetic-aperture radar processing using fast factorized back-projection*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 3, pp. 760–776, July 2003.

[6] B. Rigling and R. Moses, *Polar format algorithm for bistatic SAR*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 4, pp. 1147–1159, Oct 2004.

[7] N. J. Willis, *Bistatic Radar.* SciTech Publishing, 1995.

[8] G. P. Cardillo, *On the use of the gradient to determine bistatic SAR resolution*, in *Antennas and Propagation Society International Symposium, 1990. AP-S Merging Technologies for the 90's Digest*, May 1990, pp. 1032–1035 vol.2.

[9] W. G. Carrara, R. S. Goodman, and R. M. Majewski, *Spotlight Synthetic Aperture Radar- Signal Processing Algorithms.* Norwood, MA: Artech House, 1995.

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

[11] R. E. Kell, *On the derivation of bistatic RCS from monostatic measurements*, *Proceedings of the IEEE*, vol. 53, no. 8, pp. 983–988, Aug 1965.

[12] B. D. Rigling and R. L. Moses, *Taylor expansion of the differential range for monostatic SAR*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 60–64, Jan 2005.

[13] N. Willis and H. Griffiths, *Advances in Bistatic Radar*, ser. Electromagnetics and Radar. SciTech Publishing, 2007.

[14] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts and Techniques.* Simon & Schuster, 1992.

[15] I. Walterscheid, J. Klare, A. R. Brenner, J. H. Ender, and O. Loffeld, *Challenges of a bistatic spaceborne/airborne SAR experiment*, *EUSAR 2006*, 2006.

[16] J. Ender, *The meaning of k-space for classical and advanced SAR-techniques*, *Proc. PSIP*, vol. 2001, 2001.

[17] B. D. Rigling and R. L. Moses, *Flight path strategies for 3-D scene reconstruction from bistatic SAR*, *IEE Proceedings - Radar, Sonar and Navigation*, vol. 151, no. 3, pp. 149–157, June 2004.

[18] M. Richards, *Fundamentals of Radar Signal Processing*, ser. Professional Engineering. McGraw-Hill Education, 2005.

[19] W. Dower and M. Yeary, *Bistatic SAR: Forecasting Spatial Resolution*, *IEEE Transactions on Aerospace and Electronic Systems*, Under Review 2017.

[20] B. D. Rigling, *Advances in Bistatic Radar.* SciTech Publishing, 2007, ch. 10, Fundamentals of Bistatic Synthetic Aperture Radar, pp. 320–431.

[21] J. Mittermayer, A. Moreira, and O. Loffeld, *Spotlight SAR data processing using the frequency scaling algorithm*, *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 37, no. 5, pp. 2198–2214, Sep 1999.

[22] A. Doerry, *Patch diameter limitation due to high chirp rates in focused SAR images*, *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 30, no. 4, pp. 1125–1129, Oct 1994.

[23] M. Horvath, L. Gorham, and B. Rigling, *Scene Size Bounds for PFA Imaging with Postfiltering*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 1402–1406, APRIL 2013.

[24] M. Richards, *Fundamentals of Radar Signal Processing, Second Edition.* McGraw-Hill Education, 2014.

[25] P. Eichel, D. Ghiglia, J. Jakowatz, C.V., and D. Wahl, *Phase Gradient Autofocus for SAR Phase Correction: Explanation and Demonstration of Algorithmic Steps*, pp. 6.6.1–6.6.2, Sep 1992.

[26] M. Preiss, D. Gray, and N. Stacy, *The effect of polar format resampling on uncompensated motion phase errors and the phase gradient autofocus algorithm*, vol. 3, pp. 1442–1444 vol.3, 2001.

[27] C. A. Wiley, *Pulsed doppler radar methods and apparatus*, Patent, July 20, 1965, US Patent 3,196,436.

[28] S. Kingsley and S. Quegan, *Understanding Radar Systems.* SciTech Publishing, 1999.

[29] B.-C. Wang, *Digital signal processing techniques and applications in radar image processing.* John Wiley & Sons, 2008, vol. 91.

[30] V. T. Vu, T. K. Sjogren, and M. I. Pettersson, *Fast Time-Domain Algorithms for UWB Bistatic SAR Processing*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 3, pp. 1982–1994, July 2013.

[31] M. C. Jackson, *The geometry of bistatic radar systems*, *IEE Proceedings F, Communications, Radar and Signal Processing*, vol. 133, no. 7, pp. 604–612, December 1986.

[32] T. Zeng, M. Cherniakov, and T. Long, *Generalized approach to resolution analysis in BSAR*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 2, pp. 461–474, April 2005.

[33] M. Cherniakov, *Bistatic radars: Emerging technology.* John Wiley & Sons, 2008.

[34] D. E. N. Davies, *Use of bistatic radar techniques to improve resolution in the vertical plane*, *Electronics Letters*, vol. 4, no. 9, pp. 170–171, May 1968.

[35] J. Munson, D.C., J. O'Brien, and W. Jenkins, *A tomographic formulation of spotlight-mode synthetic aperture radar*, *Proceedings of the IEEE*, vol. 71, no. 8, pp. 917–925, Aug 1983.

[36] M. I. Skolnik, *Introduction to radar systems*, 2001.

[37] J. T. Bushberg and J. M. Boone, *The essential physics of medical imaging.* Lippincott Williams & Wilkins, 2011.

[38] T. Peters, *Algorithms for fast back-and re-projection in computed tomography*, *IEEE transactions on nuclear science*, vol. 28, no. 4, pp. 3641–3647, 1981.

[39] H. Xie, S. Shi, D. An, G. Wang, G. Wang, H. Xiao, X. Huang, Z. Zhou, C. Xie, F. Wang et al., *Fast Factorized Backprojection Algorithm for One-Stationary Bistatic Spotlight Circular SAR Image Formation*, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 4, pp. 1494–1510, 2017.

[40] D. E. Wahl, D. A. Yocky, and C. V. Jakowatz Jr, *An implementation of a fast backprojection image formation algorithm for spotlight-mode SAR*, *Algorithms for Synthetic Aperture Radar Imagery XV*, vol. 6970, p. 69700H, 2008.

[41] S. I. Kelly and M. E. Davies, *A fast decimation-in-image back-projection algorithm for SAR*, in *2014 IEEE Radar Conference*, May 2014, pp. 1046–1051.

[42] A. F. Yegulalp, *Fast backprojection algorithm for synthetic aperture radar*, in *Proceedings of the 1999 IEEE Radar Conference. Radar into the Next Millennium*, 1999, pp. 60–65.

[43] A. W. Doerry, *Synthetic aperture radar processing with polar formatted subapertures*, in *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, vol. 2, Oct 1994, pp. 1210–1215 vol.2.

[44] D. H. Garces, W. T. Rhodes, and N. M. Peña, *Projection-slice theorem: a compact notation*, *Journal of the Optical Society of America*, vol. 28, no. 5, pp. 766–769, 2011.

[45] N. Baddour, *Operational and convolution properties of three-dimensional Fourier transforms in spherical polar coordinates*, *Journal of the Optical Society of America*, vol. 27, no. 10, pp. 2144–2155, 2010.

[46] D. M. Pozar, *Microwave and RF design of wireless systems.* John Wiley & Sons, Inc., 2000.

[47] S. D. Blunt, W. Dower, and K. Gerlach, *Hybrid Adaptive Receive Processing for Multistatic Radar*, in *The 2nd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, 2007. CAMPSAP 2007.*, Dec 2007, pp. 5–8.

[48] S. D. Blunt, W. Dower, and K. Gerlach, *Hybrid interference suppression for multistatic radar*, *IET Radar, Sonar Navigation*, vol. 2, no. 5, pp. 232–333, October 2008.

[49] J. Tsao and B. D. Steinberg, *Reduction of sidelobe and speckle artifacts in microwave imaging: the CLEAN technique*, *IEEE Transactions on Antennas and Propagation*, vol. 36, no. 4, pp. 543–556, Apr 1988.

[50] *Large Scene Gotcha Data Example, Air Force Research Laboratory (AFRL), Sensor Data Management System (SDMS)*. [Online]. Available: https://www.sdms.afrl.af.mil

[51] L. A. Gorham, *Large Scene SAR Image Foramtion*, PhD dissertation, Wright State University, 2015.

[52] L. Gorham and B. Rigling, *Fast corrections for polar format algorithm with a curved flight path*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 6, pp. 2815–2824, December 2016.

[53] *Matlab's Built-In Functions on a GPU*. [Online]. Available: https://www.mathworks.com/help/distcomp/run-built-in-functions-on-a-gpu.html

[54] U. Majumder, M. Soumekh, M. Minardi, S. Scarborough, L. Gorham, C. Casteel, M. Judge, and J. Kirk, *Synthetic Aperture Radar moving target indication processing of along-track monopulse nonlinear gotcha data*, in *2009 IEEE Radar Conference*, May 2009, pp. 1–6.

[55] V. Murthy, F. Uysal, and S. Scarborough, *Target geolocation in Gotcha data using cross-channel interferometry*, in *2014 IEEE Radar Conference*, May 2014, pp. 0017–0021.

[56] S. I. Kelly, G. Rilling, M. Davies, and B. Mulgrew, *Iterative image formation using fast (Re/Back)-projection for spotlight-mode SAR*, in *2011 IEEE RadarCon (RADAR)*, May 2011, pp. 835–840.

[57] G. O. Glentis, K. Zhao, A. Jakobsson, and J. Li, *Non-Parametric High-Resolution SAR Imaging*, *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1614–1624, April 2013.

[58] D. Pritsker, *Efficient Global Back-Projection on an FPGA*, in *2015 IEEE Radar Conference (RadarCon)*, May 2015, pp. 0204–0209.

[59] R. L. Moses and J. N. Ash, *An Autoregressive Formulation for SAR Backprojection Imaging*, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2860–2873, OCTOBER 2011.

[60] R. K. Hersey and E. Culpepper, *Radar processing architecture for simultaneous SAR, GMTI, ATR, and tracking*, in *2016 IEEE Radar Conference (RadarConf)*, May 2016, pp. 1–5.

[61] W. Melvin and J. Scheer, *Principles of Modern Radar Vol. II: Advanced Techniques. Edison*, 2013.

[62] R. Samaniego, *Synthetic aperture radar incorporating height filtering for use with land-based vehicles*, July 27 2010, US Patent 7,764,220.

[63] J. A. Pruett, T. E. Adams, C. T. Moshenrose, and J. M. Grimm, *Vehicular Surveillance System Using a Synthetic Aperture Radar*, Oct. 11 2011, US Patent 8,035,545.

[64] H. Lee, J.-H. Lee, K.-E. Kim, N.-H. Sung, and S.-J. Cho, *Development of a Truck-Mounted Arc-Scanning Synthetic Aperture Radar*, *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 52, no. 5, pp. 2773–2779, May 2014.

[65] A. K. Mishra and B. Mulgrew, *Generation of sar image for real-life objects using general purpose em simulators*, *IETE Technical Review*, vol. 26, no. 1, pp. 18–27, 2009.

[66] S. Blunt and K. Gerlach, *Adaptive pulse compression via MMSE estimation*, *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 42, no. 2, pp. 572–584, April 2006.