BYU MICRO-SAR: A VERY SMALL, LOW-POWER LFM-CW SYNTHETIC APERTURE RADAR

by

Michael I. Duersch

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Electrical and Computer Engineering

Brigham Young University

December 2004

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a thesis submitted by

Michael I. Duersch

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date	David G. Long, Chair
Date	Karl F. Warnick
Date	Dah Jye Lee

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the thesis of Michael I. Duersch in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

David G. Long
Chair, Graduate Committee

Accepted for the Department

Michael A. Jensen
Graduate Coordinator

Accepted for the College

Douglas M. Chabries
Dean, Ira A. Fulton College of
Engineering and Technology

ABSTRACT

BYU MICRO-SAR: A VERY SMALL, LOW-POWER LFM-CW SYNTHETIC APERTURE RADAR

Michael I. Duersch

Department of Electrical and Computer Engineering

Master of Science

Brigham Young University has developed a low-cost, light-weight, and low power consumption SAR for flight on a small unmanned aerial vehicle (UAV) at low altitudes. This micro-SAR, or μSAR, consumes only 18 watts of power, ideal for application on a small UAV. To meet these constraints, a linear frequency modulation-continuous wave (LFM-CW) transmit signal is utilized.

Use of an LFM-CW signal introduces some differences from the typical strip map SAR processing model that must be addressed in signal processing algorithms. This thesis presents a derivation of the LFM-CW signal model and the associated image processing algorithms used for the μSAR developed at BYU.

A data simulator for the BYU LFM-CW SAR is detailed and results are provided for the case when the simulated data are processed using the μSAR algorithms. Data processing schemes are discussed, including compression, receive signal phase detection, interference filtering and auto-focusing. Finally, data collected from the instrument itself are processed and presented.

ACKNOWLEDGMENTS

I want to express appreciation to everyone who has assisted me in my education and research.

Most of my thanks "beLong" to my mentor and adviser, Dr. David Long. He has spent many hours instructing me about radar, as well as life in general. Even sometimes on late notice, he has spent much time reviewing and editing this thesis.

Thanks to my colleagues, David Duncan and Derek Hudson, who have provided guidance and insight into SAR and have helped me in my research.

Finally, I would also like to thank my family for their love and support. Especially, I want to thank my Michelle for her many hours of proofreading, patience and kindness.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Purpose

Synthetic aperture radar is a valuable technique in remote sensing with applications ranging from environmental to military. There is a wide variety of types of synthetic aperture radar (SAR). However, SAR has been traditionally quite expensive and only affordable on a governmental basis.

For over a decade, Brigham Young University (BYU) has been developing low-cost, compact SAR instruments. Because these instruments are relatively inexpensive by comparison to previous sensors, they allow the application of SAR in areas that would otherwise be impossible due to the prohibitive cost. BYU has innovated on typical SAR design by combining standard PC hardware with original RF subsystems that dramatically reduce the cost of a functional SAR system.

The latest innovation in SAR technology at BYU is the smallest, lightest, and lowest power-consuming design yet created. This new system is termed μSAR (pronounced *micro*-SAR). The goal of the μSAR project is to design a synthetic aperture radar small enough and with low enough power requirements to operate on a small unmanned aerial vehicle (UAV).

The μSAR design includes a number of fundamental aspects, including a linear frequency modulation continuous wave (LFM-CW) transmit signal. As a result, existing SAR processing algorithms must be modeled to process μSAR data.

The purpose of this thesis is to discuss the signal processing algorithms developed for the μSAR. A discussion of this development begins with conventional strip

map SAR techniques and extends to their application for the μSAR. Specifically, signal processing idiosyncrasies pertaining to LFM-CW radar are addressed, as well as data filtering and autofocusing techniques used to improve image quality.

1.2 Contributions

This thesis contributes to the base of synthetic aperture radar knowledge in the case of continuous-wave signal transmission. Previously, most SAR systems have been designed using a pulsed transmit signal. Advantages of LFM-CW SAR for short range applications are given and processing algorithm modifications required to accommodate this type of radar are shown.

Included in this discussion are processing techniques employed to refine the SAR imagery. Though the transmitted signal is continuous, its repetition must be tracked precisely for proper image formation. Due to external interference in the RF spectrum used by the μSAR, a method for data filtering is presented. Finally, inability to accurately measure the velocity of the SAR platform also leads to important imaging issues resolved with auto-focusing.

1.3 Outline

The remainder of this thesis is organized into the following chapters:

- Chapter 2 contains SAR background and a derivation of the signal model used for synthetic aperture radar. It includes differences from the usual SAR model that are specific to the BYU μSAR. Finally, the mathematical procedure for processing the data into images is presented.

- Chapter 3 introduces the simulator created specifically to generate sample μSAR data for testing the image processing algorithms.

- Chapter 4 provides the data processing methods particular to this project. Each of the contributions in the signal processing of μSAR data is discussed as well as results from the μSAR itself.

- Chapter 5 concludes the thesis and offers areas in which there are further research opportunities.
- Appendix A gives the specifications and performance, signal block diagram and pictures of the μSAR.
- Appendix B describes the software operating commands for the μSAR, including both the data collection and image processing programs.

Chapter 2

Background

This chapter describes the essential SAR background supporting the discussion of μSAR in this thesis. Contributions of BYU, fundamental concepts of SAR processing and the geometric model characterizing SAR are introduced. Range and azimuth resolution are described, motivating the use of SAR technology. Next, model adaptations particular to μSAR are discussed. Finally, the SAR processing model is presented, first for the range direction, then for the cross-range or azimuth direction.

2.1 SAR Fundamentals

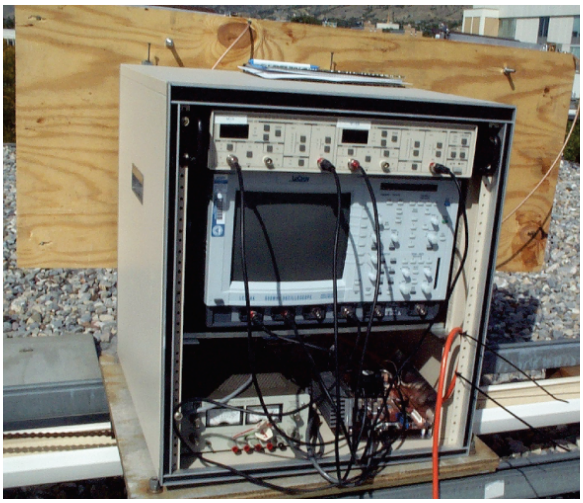
Synthetic aperture radar is an important tool in earth remote sensing. In conventional radar systems, image resolution is dependent on the illumination footprint size on the ground. Much finer resolution is attainable, however, using SAR techniques. These techniques are based on transmission of linear frequency modulation (LFM) signals and consideration of Doppler frequency shifts in received signals.

While SAR techniques lead to dramatic improvements in spatial resolution over conventional, real-aperture radar systems, SAR technology is traditionally very large, extremely expensive and thus unavailable in a wide variety of applications. The goal of the Brigham Young University Microwave Earth Remote Sensing (MERS) Laboratory has been to create smaller, lower cost SAR systems. Pictures of several BYU SAR projects are shown in Fig. 2.1.

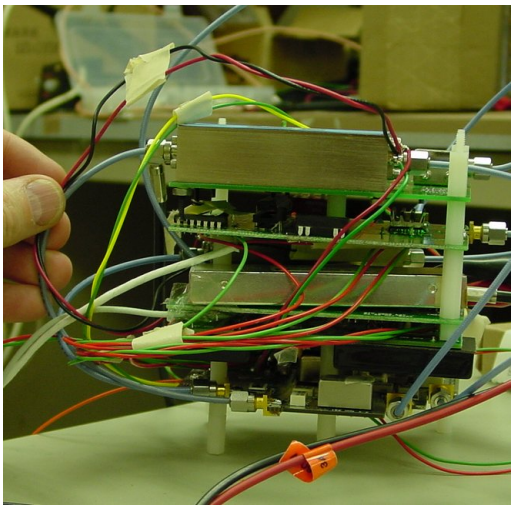
BYU's first SAR was YSAR, which was developed to prove the feasibility of small, low-cost SAR [1]. It flew over Israel to image archaeological sites. YINSAR is the second SAR project created at BYU and was funded by a grant from NASA [1].

(a) YSAR

(b) YINSAR



(c) rail-SAR



(d) μSAR

Figure 2.1: Images of BYU SAR projects.

It expanded on the YSAR design by including two receive antennas for the purpose of interferometry. This allows YINSAR to create images in three dimensions. Additionally, the hardware systems of YINSAR are more sophisticated, allowing better results with a more compact, light-weight design.

The third-generation SAR developed was rail-SAR, a lab experiment to determine the possibility of using low-cost, off-the-shelf technology for SAR along a track on the ground [2]. This SAR is also the first SAR at BYU to use continuous wave (CW) signal transmission as opposed to pulsed transmission. Finally, the fourth and most recent SAR system created at BYU is μSAR.

μSAR is the smallest, most light-weight and low-cost design yet. Because its purpose is to fly in a small UAV, it also improves over previous SAR designs in low power consumption and compact size. It was developed with the specific goal of consuming less than 20 Watts. Additional particulars of μSAR are discussed in Section 2.4

2.2 Geometry

To understand the derivation of the strip map SAR model detailed in following sections, the SAR geometric model is described first. The SAR geometry model is shown in Fig. 2.2. In the figure, $(\hat{x}, \hat{y}, \hat{z})$ refers to the directions range, elevation, and azimuth, respectively. The SAR platform located at coordinates (x,y,z) flies along a straight path in $\hat{z}$ (azimuth) at a constant height, h, and velocity, v. Assuming the platform begins at the origin and moves only in azimuth, the platform's location is simplified to $(0, 0, z(t))$. The range function from the platform to a target located at (x_o, y_o, z_o) is given by

$$R_{slant}(x_o, y_o, z_o, z) = \sqrt{(x_o)^2 + (y_o)^2 + (z_o - z)^2}. \tag{2.1}$$

This distance is known as slant range [3]. Using a fan-beam antenna, the radar transmits signals to the side of the flight path. The return signals are received and then processed.

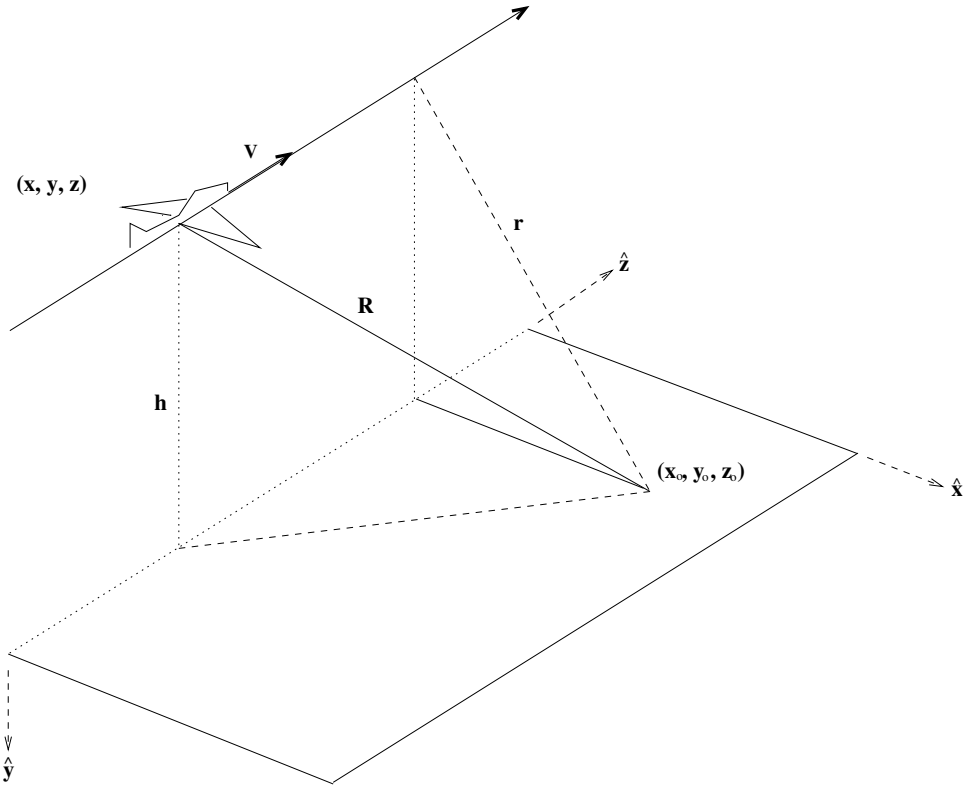


Figure 2.2: SAR imaging geometry.

Many algorithms simplify the slant range equation by combining the range and height terms (x_o and y_o respectively). Thus, substituting

$$r = \sqrt{(x_o)^2 + (y_o)^2}$$

into Eq. (2.1), leads to the result

$$R_{slant}(r, z, z_o) = \sqrt{(r)^2 + (z_o - z)^2}. \tag{2.2}$$

Contributions of each target's signature is present in data for as long as it is illuminated inside the antenna footprint. The data from many segments are combined to create a synthetic aperture much longer than the real aperture of the antenna, as shown in Fig. 2.3. The target enters the antenna footprint at time t_1 and exits it at t_2.

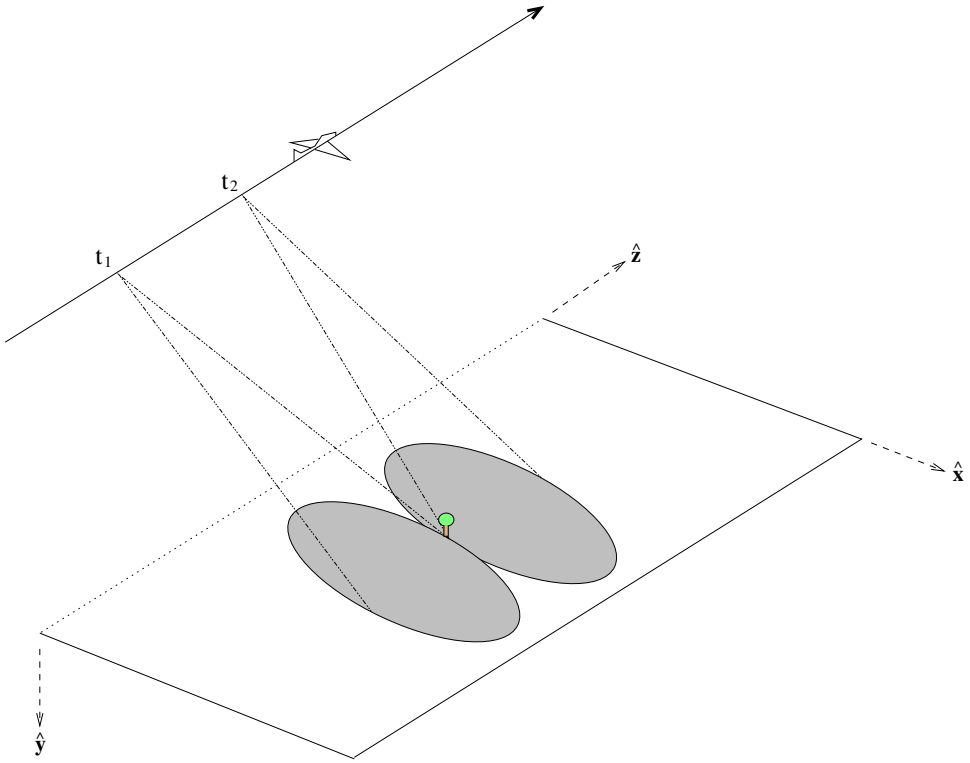


Figure 2.3: Sketch showing a target as it enters and leaves the antenna footprint as the platform flies past.

2.3 Image Resolution

For a given antenna size, resolution is a measure of the resolvability of individual targets by a radar. SAR techniques lead to finer resolution than conventional real-aperture systems, thus making them desirable in radar imaging. The following subsections discuss resolution for real and synthetic aperture radar, in both the range and azimuth directions.

2.3.1 Range Resolution

Range resolution may be obtained using a pulse of very short transmit duration. The resolution resulting from a simple square pulse shape with duration T_p is

$$\Delta r = \frac{c_o T_p}{2} \tag{2.3}$$

in meters per pixel where c_o is the speed of light. Consequently, the shorter the pulse length, the finer the resolution in range. Taking this to the logical limit suggests a delta function as the pulse shape. Unfortunately, this ideal is impossible as the delta function has infinite magnitude. In practice, hardware designed to transmit pulses of very short length is very expensive and such systems suffer from unusably low signal-to-noise ratio (SNR). To alleviate this, it is possible to obtain high resolution using longer pulses that are frequency modulated. Linear frequency modulation (LFM) is by far the most common signal type used in practical SAR applications [4].

An LFM chirp is a sinusoidal signal with linearly varying frequency. A complex LFM chirp can be written as $\exp(j\pi\beta t^2)$ where β is the chirp rate. The time derivative of this phase term gives the frequency $2\pi\beta t$, a linear function in time. Multiplying by a time-shifted copy of its conjugate (matched filtering) results in a signal with linear phase whose frequency is proportional to the time shift. This property is used in range compression, shown in the Section 2.5.

There are several different methods for deriving the range resolution of an LFM chip [5] [6] [7]; however, one simple approach is to recognize that radar bandwidth is approximately equal to $1/T_p$, so Eq. (2.3) can be expressed as

$$\Delta r = \frac{c_o}{2BW}, \tag{2.4}$$

where BW is the bandwidth of the transmitted signal. Increasing the transmit bandwidth thus improves the resolution. This result is consistent with various derivations, and is a common rule-of-thumb in SAR signal processing.

2.3.2 Azimuth Resolution

Azimuth resolution is achieved through the use of a synthetic aperture as opposed to a real aperture. In real aperture radar, the azimuth resolution is proportional to the azimuth beamwidth of the antenna while synthetic aperture resolution is inversely proportional to the azimuth beamwidth. It can be shown [8] that the azimuth

resolution of a real aperture radar is

$$\Delta z = 2r \tan\left(\frac{\theta_a}{2}\right)$$

where r is the near range and θ_a is the azimuth beamwidth in radians. Substituting the approximation $\theta_a \approx \lambda / L_z$ into $\Delta z \approx r\theta_a$, the azimuth resolution is

$$\Delta z \approx \frac{r\lambda}{L_z} \tag{2.5}$$

where λ is the carrier wavelength, and L_z is the antenna length in azimuth [9]. As is apparent, an increase in the length of the antenna yields finer resolutions. In practice, however, the antenna length is limited by the length the platform can support. For practical systems, obtaining very fine azimuth resolution with real aperture radar is impractical due to the large antenna size required.

Synthetic aperture radar is based on the Doppler shift added to the return signal of the target as the platform travels along its path. The shift is positive as the platform approaches each target, is zero at boresight, and goes negative as the platform moves away from each target. It can be shown [10] that the achievable azimuth resolution for SAR is the velocity divided by azimuth bandwidth

$$\Delta z = \frac{v}{BW_z}. \tag{2.6}$$

The following derivation shows how the azimuth bandwidth can be approximated using the phase change of the radar return off each target. A more mathematically rigorous derivation can be found in [11]. The phase change ϕ_D of the received signal is related to the change in range δ_r from the platform to the target

$$\begin{aligned} \phi_D &= \frac{4\pi f \delta_r}{c_o} \\ &= \frac{4\pi\delta_r}{\lambda}. \end{aligned} \tag{2.7}$$

The variable δ_r can be estimated from the slant range R:

$$\begin{aligned} R &= \sqrt{(r)^2 + (vt)^2} \\ &\approx r + \frac{(vt)^2}{2r} \end{aligned} \tag{2.8}$$

$$\delta_r \approx \frac{(vt)^2}{2r}. \tag{2.9}$$

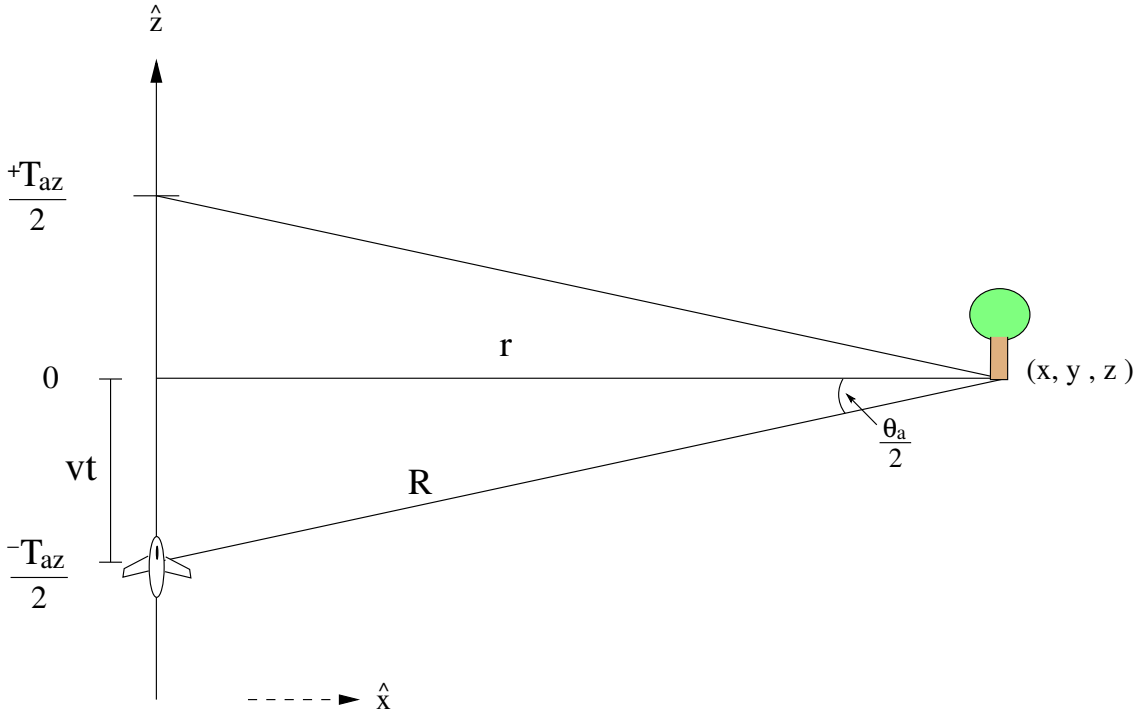


Figure 2.4: Overhead view of time/distance relationship of platform moving past a fixed target.

An example of this is shown in Fig. 2.4. Equation (2.8) is found assuming $vt \ll R$ and with the Taylor expansion approximation

$$\sqrt{1+x} \approx 1 + \frac{x}{2} \qquad \text{where} \qquad x \ll 1,$$

with error $x^2/4$. Substituting δ_r into Eq. (2.7) gives

$$\phi_D(t) = \frac{2\pi v^2 t^2}{\lambda r}. \tag{2.10}$$

Noting that the phase changes quadratically with time, this phase change may be considered an azimuth chirp. The azimuth frequency f_z is found by differentiating the phase to obtain

$$f_z = \frac{2v^2 t}{\lambda r}$$

and the azimuth chirp rate is

$$\dot{f}_z = \frac{2v^2}{\lambda r}.$$

The azimuth bandwidth BW_z is found from the chirp rate $\dot{f} \triangleq BW/T$, where T is the period. Figure 2.4 shows the geometric relation used to find the azimuth period T_z. It can be seen from the figure that

$$\tan\left(\frac{\theta_a}{2}\right) = \frac{vT_z}{2r}.$$

Applying the small angle approximation $\tan(\theta_a/2) \approx \theta_a/2$, the azimuth period can be obtained:

$$T_z = \frac{\theta_a r}{v}.$$

This allows the azimuth bandwidth to be solved for in terms of the approximated azimuth period,

$$\begin{aligned} BW_z &= \dot{f}_z T_z \\ &= \frac{2v^2}{\lambda r}\frac{\theta_a r}{v} \\ &= \frac{2v\theta_a}{\lambda}. \end{aligned} \tag{2.11}$$

Finally, employing the previously used $\theta_a \approx \lambda/L_z$ and substituting this result into Eq. (2.6) yields the SAR azimuth resolution, Δz,

$$\begin{aligned} \Delta z &= \frac{v}{BW_z} \\ &= \frac{\lambda}{2\theta_a} \\ \Delta z &\approx \frac{L_z}{2}. \end{aligned} \tag{2.12}$$

Comparing this result with Eq. (2.5) shows the obvious advantage of synthetic aperture radar over real aperture radar. For the same size antenna, application of SAR processing leads to much finer azimuth resolutions and makes available high resolution radar imaging on aircraft.

This result suggests that an infinitely small antenna provides the best azimuth resolution. However, the pulse repetition frequency (PRF) must meet the minimum Nyquist criterion, which is twice the azimuth bandwidth:

$$PRF_{min} = \frac{4v\theta_a}{\lambda}, \quad \text{or}$$
$$\approx \frac{4v}{L_z}.$$

Thus, PRF_{min} is inversely proportional to the length of the antenna. Conversely, the maximum allowable PRF is limited in practice by hardware restrictions and desirable data bandwidth. Therefore, a trade-off must be made between azimuth resolution and the minimum PRF.

2.4 μSAR Particulars

As discussed, the type signal used in transmission is crucial to obtaining the benefits seen in SAR imaging. As shown in Section 2.3.1, improved resolution in the range direction is achieved through use of an LFM transmit signal. There are two types of LFM transmit signals used in SAR, pulsed and continuous wave. An example of pulsed and continuous-wave radar is shown in Fig. 2.5.

Pulsed signal transmission is employed by traditional SAR systems and is by far the most commonly used. It involves transmitting a quick burst of electromagnetic radiation and recording the received reflections off landscape targets, then waiting a short period of time and repeating the process. The term *pulse repetition interval* (PRI) refers to the elapsed time between pulse transmissions. PRF refers to the number of pulses transmitted per second, and is equivalent to the reciprocal of the PRI.

Linear frequency modulation continuous-wave (LFM-CW) radar, on the other hand, continuously transmits and receives simultaneously. Although the signal is continually being transmitted in LFM-CW radar, it is possible to draw an analogy between pulsed and LFM-CW methods. As seen in Fig. 2.5, the frequency of the transmit signal increases, then decreases, then increases again and so on. The time

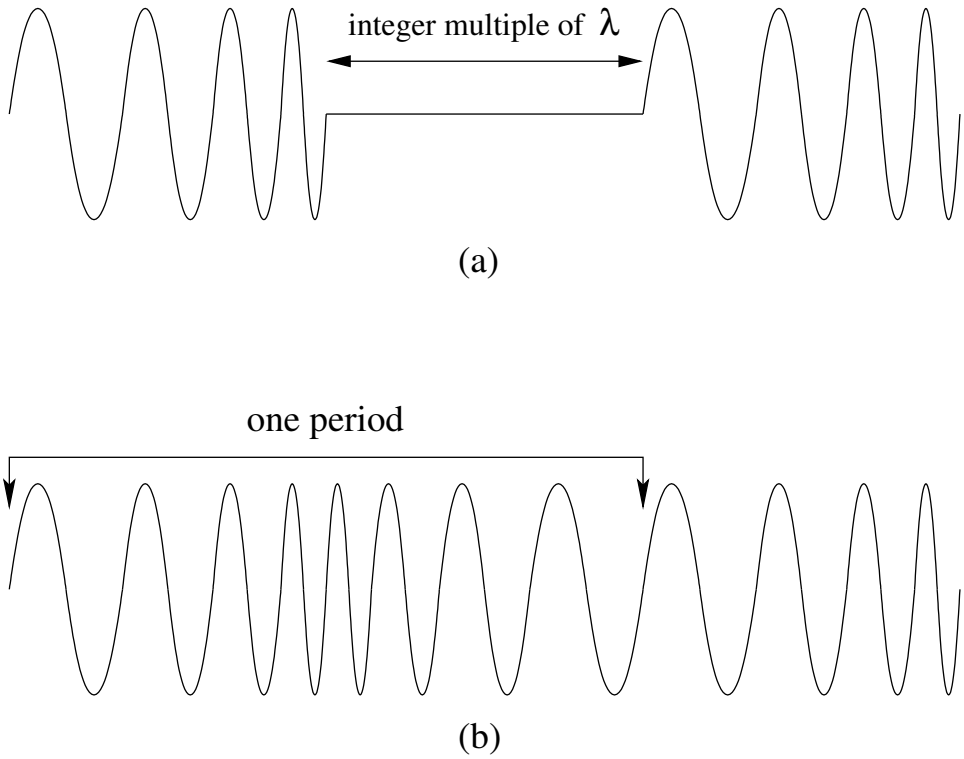


Figure 2.5: (a) Pulsed radar. (b) Continuous-wave radar.

interval it takes for the frequency oscillation to ramp up and back down is considered the PRI, and hence the number of up/down ramps per second is the PRF.

Figure 2.6 shows the LFM-CW transmit/receive model of frequency versus time. Received returns off targets are a time delay, $t_0 ... t_n$, of the transmit signal. Targets farther in range correspond to a greater frequency difference. In the figure, T represents the duration of one up-ramp, or half the PRI.

A key difference between the two types of SAR is that pulsed radar generally mixes the received signal down to some intermediate frequency, then samples and stores the data, while LFM-CW radar mixes the received signal with the transmitting signal (essentially a delayed copy of itself), then samples. This has the advantage of requiring a lower sampling frequency, but the disadvantage of requiring a higher dynamic sampling range as discussed in Section 2.5.

LFM-CW radar also has the advantage of consuming less power than pulsed radar. This is because LFM-CW transmits pulses of much longer duration. Longer

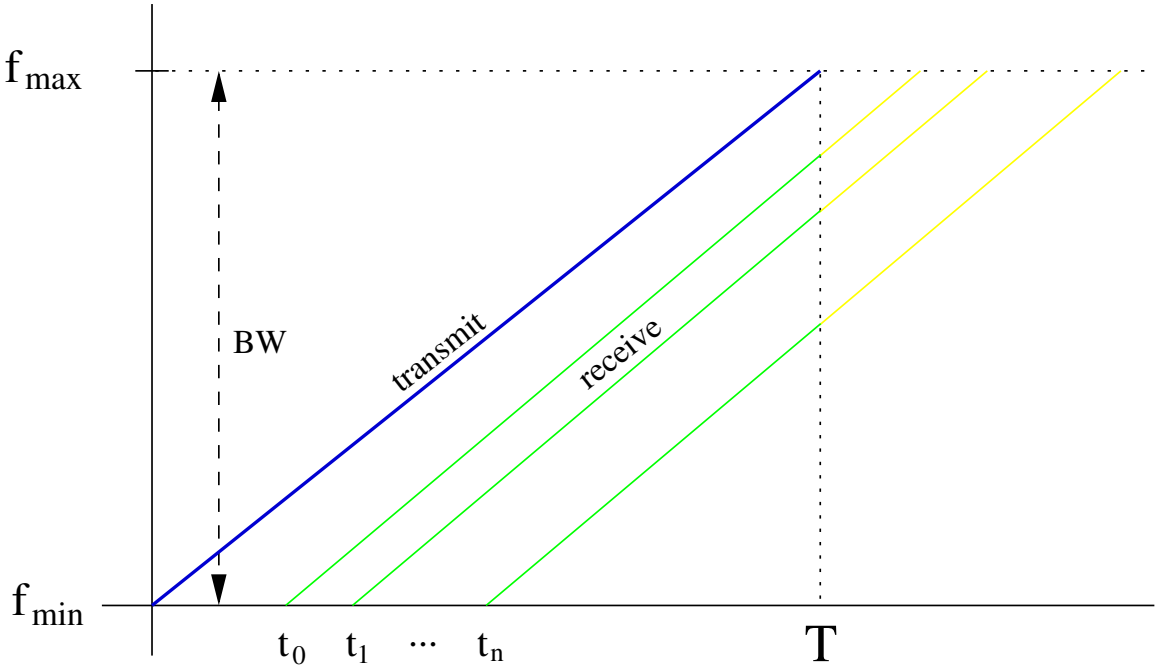


Figure 2.6: Sketch showing LFM-CW signal return off multiple targets.

pulse length yields more energy contained in a single pulse; hence, LFM-CW SAR transmits with less power to maintain the same SNR as conventional, pulsed SAR.

Because μSAR must operate with low power and cost requirements to be feasible on a UAV, an LFM-CW transmit signal was chosen. An LFM-CW also simplifies the sampling hardware and lowers the overall cost and size of the system.

2.5 Range Compression

When data are recorded, each target's two-dimensional signature is spread in both the range and azimuth directions. Figure 2.7 shows an example of two point targets, each at position 0m in azimuth, but at 40m and 90m in range. As seen, without processing of the signals the targets are very difficult to resolve.

The term *compression* involves "squeezing" the signature of each target into one energy bin. This process is preformed first in the range direction, then in the azimuth direction. This section describes the process of range compression, first for pulsed radar, then for LFM-CW radar.

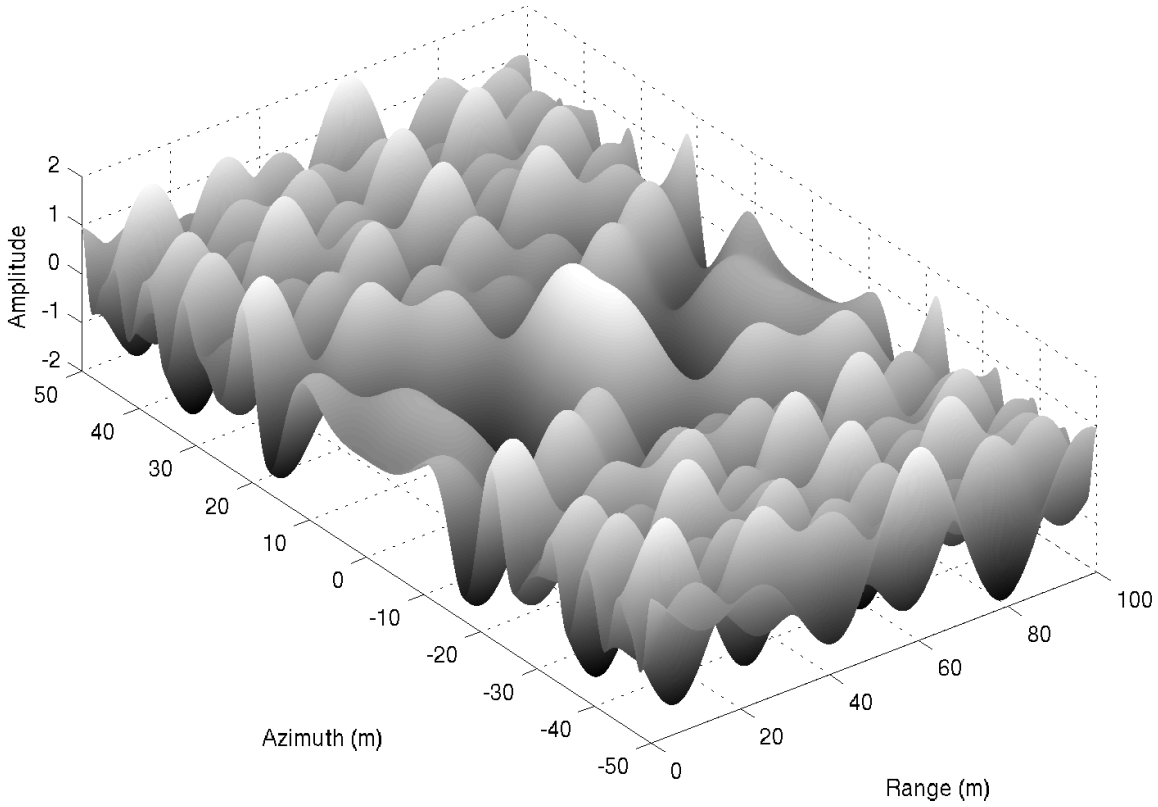


Figure 2.7: Example SAR raw waveform of two point targets.

2.5.1 Pulsed Radar

Received raw data are recorded as one long vector, which is first processed by breaking up the data into segments, each corresponding to one pulse period. This forms a two dimensional image with range on one axis and azimuth on the other, as shown in Fig. 2.7. For LFM pulsed radar, each target's signature is spread in range according to frequency.

After pulse sequences are divided, they are range compressed through the process of matched filtering with the transmitted chirp. Matched filtering is performed in the frequency domain, multiplying the received signal by the conjugate of the transmitted signal and returning to the time domain. This pushes most of the energy of single targets into individual bins in range, as shown in Fig. 2.8. A derivation of this process follows.

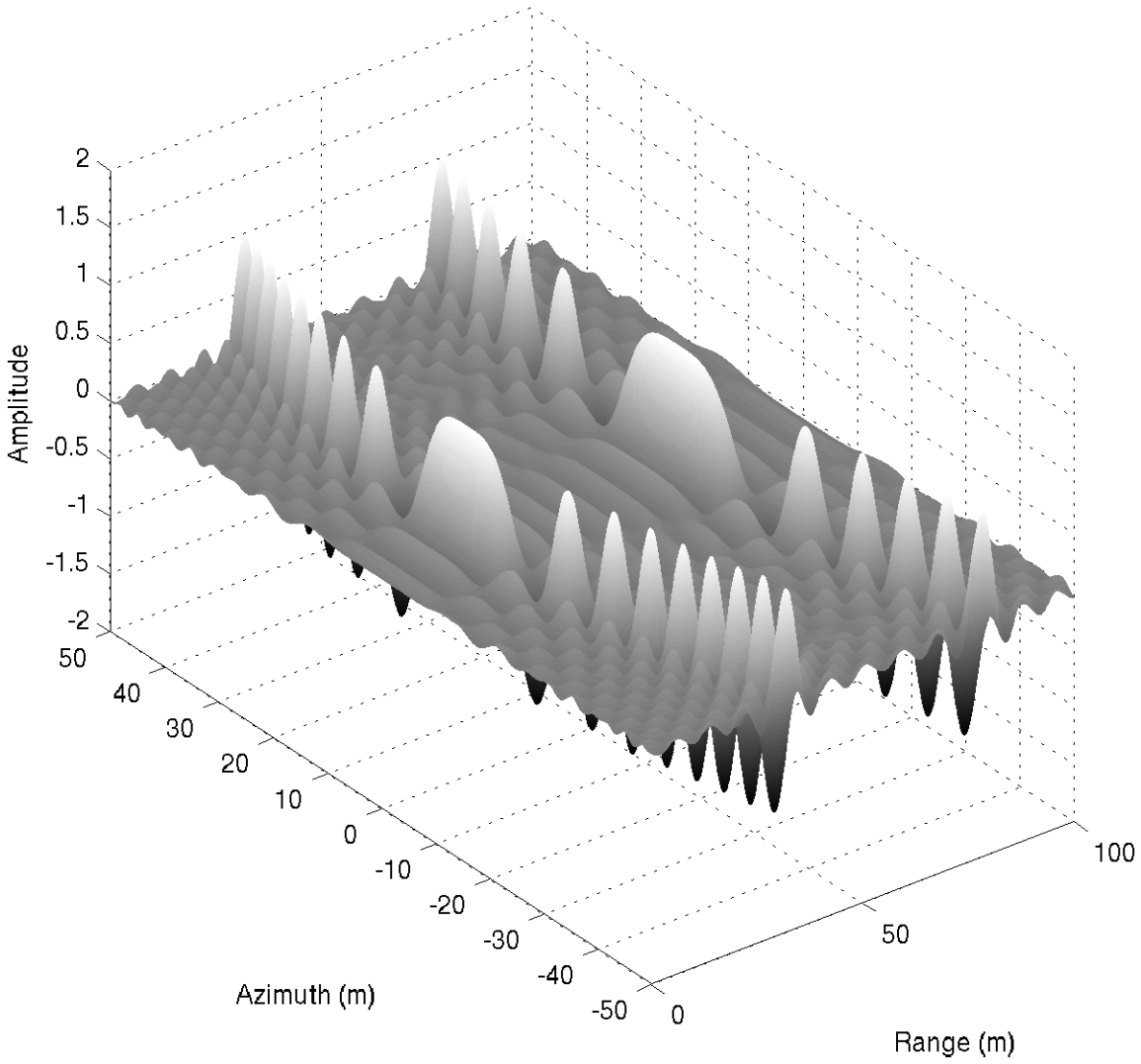


Figure 2.8: Example SAR range compressed waveform.

For simplicity, this derivation ignores azimuth effects and assumes that a reflected target's signal is a time delay, t_o, of the original signal. It is also assumed that the signal is single-side band as opposed to double-side band (a full derivation of double-side band SAR is developed by Robertson [5]). These assumptions simplify the derivation but still provide an accurate model of range compression. This derivation resembles Smith's derivation [2], which is based on Robertson's [5].

The transmit and received signals are modeled by

$$x_t(t) = \frac{1}{2}\cos\left(\omega_c t + \beta t^2\right) \qquad \text{and} \tag{2.13}$$

$$x_r(t, t_o) = \frac{1}{2}\cos\left(\omega_c(t - t_o) + \beta(t - t_o)^2\right), \tag{2.14}$$

where $\beta = \frac{\pi BW}{T}$ represents the slope of the linear frequency vs time. We note here that t is in "fast time." Hardware mixes the signal down to an intermediate frequency, w_d, and then low-pass filters it,

$$\begin{aligned} m_{if}(t) &= x_r(t) \otimes \cos(w_d t), \\ &= \frac{1}{2}\cos\left(w_c(t - t_o) + \beta(t - t_o)^2\right)\cos(w_d t), \\ &= \frac{1}{4}\cos\left(w_c(t - t_o) + \beta(t - t_o)^2 - w_d t\right) + \frac{1}{4}\cos\left(w_c(t - t_o) + \beta(t - t_o)^2 + w_d t\right), \\ &\overset{LPF}{\Longrightarrow} \frac{1}{4}\cos\left(w_c(t - t_o) + \beta(t - t_o)^2 - w_d t\right). \end{aligned} \tag{2.15}$$

This is generally signal that is digitized by the hardware and recorded. As Robertson states, as long as the data are sampled above the Nyquist rate, the rest of derivation may be done in continuous time for simplification [5]. Next, substituting $\omega_o = \omega_c - \omega_d$ and taking the Hilbert Transform yields

$$\begin{aligned} m_{if}(t) &= \frac{1}{4}e^{j\left(\omega_c(t-t_o)+\beta(t-t_o)^2-(\omega_c-\omega_o)t\right)}, \\ &= \frac{1}{4}e^{j\left(\omega_o t+\beta(t-t_o)^2-\omega_c t_o\right)}. \end{aligned} \tag{2.16}$$

At this point, it is useful to calculate the minimum Nyquist sampling rate. By examining the phase, $\phi(t)$, it can be seen that the phase derivative, $\dot{\phi}(t)$ is

$$\begin{aligned} \phi(t) &= \omega_o t + \beta(t - t_o)^2 - \omega_c t_o, \\ \dot{\phi}(t) &= \omega_o + 2\beta(t - t_o). \end{aligned} \tag{2.17}$$

$\dot{\phi}(t)$ represents frequency, whose maximum occurs for every $t - t_o = T$. Thus, based on the Nyquist criterion the minimum sampling rate is

$$S_{rate-min} = 2(f_o + BW). \tag{2.18}$$

The matched filter is given by

$$h(t) = e^{j(\omega_o t + \beta t^2)}. \tag{2.19}$$

Although $h(t)$ is infinite in length, for a finite-length matched filter with large T, this simplifying assumption is valid. The matched filtering is then performed using an autocorrelation function with $m_{if}(t)$ and $h(t)$:

$$\begin{aligned}
m(t) &= m_{if} \otimes h^*(t), \\
&= \int m_{if}(t+\tau) h^*(\tau) d\tau, \\
&= \frac{1}{4} \int e^{j\left(\omega_o(t+\tau)+\beta(t-t_o+\tau)^2-\omega_c t_o\right)} e^{-j(\omega_o\tau+\beta\tau^2)} d\tau, \\
&= \frac{1}{4} e^{j\left(\omega_o t+\beta(^-2t_o t+t_o^2+t^2)-\omega_c t_o\right)} \int e^{j\beta(2t\tau-2t_o\tau)} d\tau.
\end{aligned}$$

(applying limits of integration:)

$$\begin{aligned}
&= \frac{1}{4} e^{j\left(\omega_o t+\beta(t-t_o)^2-\omega_c t_o\right)} \int_{-t+t_o}^{T-t+t_o} e^{j\beta(2t\tau-2t_o\tau)} d\tau, \\
&= \frac{1}{4} \frac{e^{j\left(\omega_o t+\beta(t-t_o)^2-\omega_c t_o\right)}}{2j\beta(t-t_o)} \left(e^{2j\beta(T-t+t_o)(t-t_o)} - e^{2j\beta(-t+t_o)(t-t_o)}\right), \\
&= \frac{1}{4} \frac{e^{j\omega_o t - j\omega_c t_o}}{2j\beta(t-t_o)} \left(e^{2j\beta T(t-t_o)} e^{-j\beta(t-t_o)} - e^{-j\beta(t-t_o)}\right), \\
&= \frac{1}{4} \frac{e^{j\omega_o t - j\omega_c t_o}}{\beta(t-t_o)} e^{j\beta T(t-t_o)} \sin\left(\beta T(t-t_o)\right) e^{-j\beta(t-t_o)}, \\
m(t) &= \frac{T}{4} e^{j\omega_o t - j\omega_c t_o} e^{j\beta(t-t_o)(T-1)} \operatorname{sinc}\left(\beta T(t-t_o)\right).
\end{aligned} \tag{2.20}$$

Examining Eq. (2.20) leads to some important points. The $\frac{T}{4}$ amplitude term shows that the SNR is a function of the chirp period—thus a longer chirp leads to higher SNR. The sinc effect as a result of autocorrelation is also visible. The width of the sinc is inversely proportional to the bandwidth of the chirp. This suggests that the resolution in range is also a function of chirp bandwidth and not period, as shown previously.

2.5.2 Continuous-wave

The process of range compression for LFM-CW SAR is somewhat simpler than pulsed SAR, mostly due to the way data are recorded. In pulsed radar, the signal is recorded just as it was received. In LFM-CW SAR, however, the received signal is first mixed with the transmitting signal. This process is called *dechirping*.

Dechirping allows data to be sampled at a much lower frequency and still meet the minimum Nyquist rate. This is due to the fact that most of the energy of each target's signature has already been compressed. There is a drawback, however. While pulsed SAR must be recorded at a high sampling rate because of its high frequency, it has low dynamic range and can be sampled using only a few bits. Because the signature of an LFM-CW signal has already been compressed, a higher sampling dynamic range is required and quantization noise becomes a factor. In general, wider dynamic sampling at a slower rate is less complicated and simpler A/D hardware can be utilized, resulting in an overall lower costing system. A derivation of range compression for a dechirped, LFM-CW SAR follows.

The derivation for LFM-CW SAR begins as above with models for the transmitted and received signals, except the signals are expressed in exponential form for convenience:

$$x_t(t) = e^{j\left(\omega_o t + \frac{\beta t^2}{2} + \psi\right)} \qquad \text{and} \tag{2.21}$$

$$x_r(t, t_o) = e^{j\left(\omega_o (t - t_o) + \frac{\beta (t - t_o)^2}{2} + \psi\right)}. \tag{2.22}$$

The difference in amplitude between the transmitted signal and received is ignored here, and ϕ represents some arbitrary phase present in the signal. The hardware mixes these two signals together and low-pass filters:

$$\begin{aligned} m(t) &= x_t(t) \otimes x_r^*(t, t_o), \\ &= e^{j\left(\omega_o t + \frac{\beta t^2}{2} + \psi\right)} e^{-j\left(\omega_o (t - t_o) + \frac{\beta (t - t_o)^2}{2} + \psi\right)} \\ &= e^{j(\omega_o t_o + \beta t_o t - \frac{\beta t_o^2}{2})}. \end{aligned} \tag{2.23}$$

This is the signal that is digitized and recorded. As shown above, by examining the phase, $\phi(t)$, it can be seen that

$$\begin{aligned}\phi(t) &= \omega_o t_o + \beta t_o t - \frac{\beta t_o^2}{2}, \\ \dot{\phi}(t) &= \beta t_o. \end{aligned} \tag{2.24}$$

$\dot{\phi}(t)$ represents the frequency difference between the transmitting chirp and received chirp, whose maximum is the frequency difference corresponding to the farthest target in range, represented by time t_n. Thus, the minimum sampling rate is

$$S_{rate-min} = 2\beta t_n. \tag{2.25}$$

Comparing this result with Eq. (2.18), LFM-CW results in a much lower minimum sampling rate requirement than pulsed chirp radar, and thus can be implemented much easier.

Range compression is performed by taking the Fourier transform. The limits of integration are ignored at present but are treated in the next paragraph.

$$\begin{aligned} I(w) &= \int e^{j(\omega_o t_o + \beta t_o t - \frac{\beta t_o^2}{2})} e^{-j\omega t} dt, \\ &= e^{j(\omega_o t_o - \frac{\beta t_o^2}{2})} \int e^{\beta t_o t} e^{-j\omega t} dt \\ &= e^{j(\omega_o t_o - \frac{\beta t_o^2}{2})} 2\pi\delta(\beta t_o - \omega), \end{aligned} \tag{2.26}$$

resulting in the compressed signal $I(\omega)$. It is important to note that $I(\omega)$ is in the spatial domain due to the frequency-domain nature of the dechirped signal.

As shown in Fig. 2.9, the signal for the first target is only present from time t_o to time T. This results in a window, which can be represented in frequency as

$$W_{CW}(\omega) = \begin{cases} (T - t_o) e^{-j\omega \frac{T - t_o}{2}} \operatorname{sinc}\left(\frac{\omega(T - t_o)}{2}\right), & if \;\; t_o < T \\ 0, & if \;\; 0 > t_o \; or \; t_o > T. \end{cases} \tag{2.27}$$

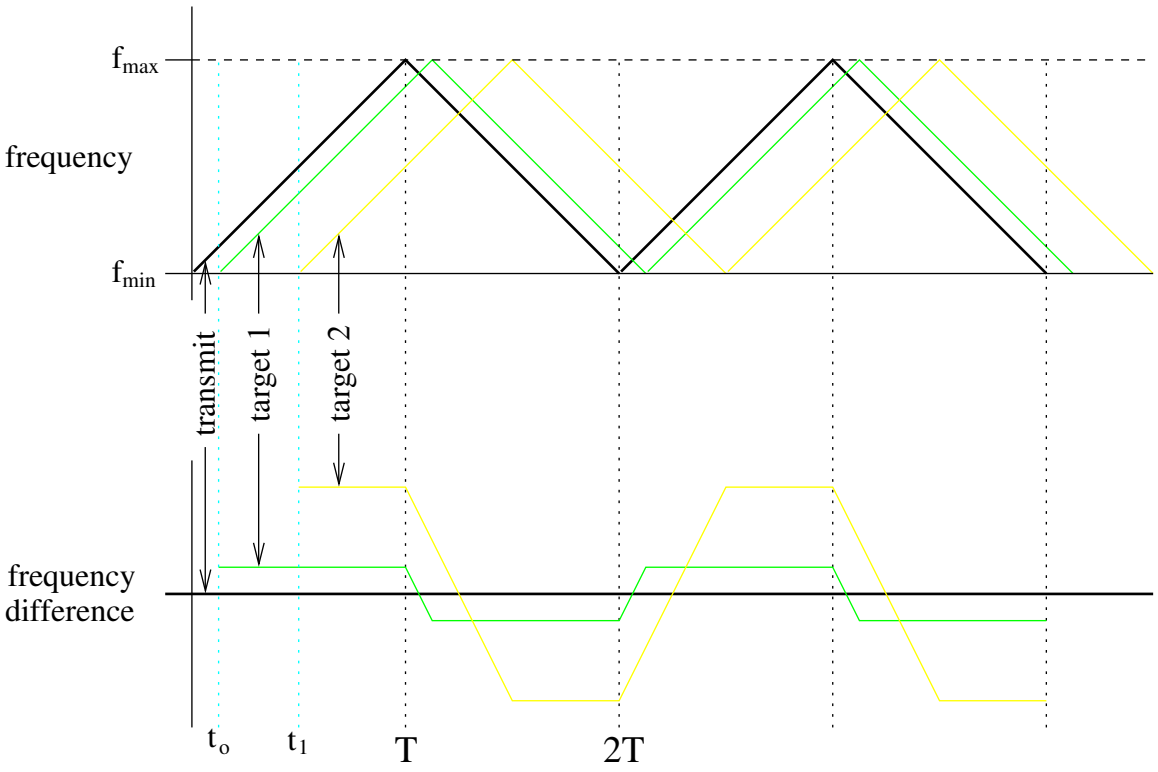


Figure 2.9: Sketch showing dechirping LFM-CW frequency difference of two point targets.

Equation (2.26) is convolved with Eq. (2.27), resulting in the complete range compressed LFM-CW signal

$$\begin{aligned} I_{CW} &= \int_{\infty}^{\infty} I(\omega-\rho)W_{CW}(\rho)d\rho \\ &= \int_{\infty}^{\infty} e^{j(\omega_o t_o - \frac{\beta t_o^2}{2})} 2\pi\delta(\beta t_o - \omega - \rho)(T-t_o)e^{-j\rho\frac{T-t_o}{2}} \operatorname{sinc}\left(\rho\frac{T-t_o}{2}\right) d\rho \\ &= 2\pi(T-t_o)e^{j(\omega_o t_o - \frac{\beta t_o T}{2} + \frac{\omega(T-t_o)}{2})} \operatorname{sinc}\left((\beta t_o-\omega)\frac{T-t_o}{2}\right) d\rho. \end{aligned} \tag{2.28}$$

There are several important points from this result to notice. The first is that the amplitude for each target is proportional to $(T-t_o)$, showing that the SNR improves with increased pulse length. Thus, because LFM-CW pulses are longer in transmit length than conventional SAR pulses, less transmit power is required in LFM-CW to maintain the same SNR. This also suggests, however, that targets farther away in range suffer from reduced power compared to closer targets if compression is simply performed by the Fourier transform.

The argument $(\beta t_o - \omega)\frac{T-t_o}{2}$ of the sinc function is also worthy of notice. Since

$$\beta = \frac{\pi BW}{T}, \tag{2.29}$$

we see that

$$\text{sinc}\left((\beta t_o - \omega)\frac{T-t_o}{2}\right) = \text{sinc}\left((\pi BW - \omega T)\frac{T-t_o}{2T}\right). \tag{2.30}$$

This shows that the width of the sinc function decreases as the bandwidth increases. Hence, resolution increases for increased bandwidth. It is also interesting to note that as t_o increases for targets farther in range, the width of the sinc function increases and the resolution degrades.

2.6 Azimuth Compression

Once the process of range compression has been performed, the last step of SAR processing is to compress the target's signature that is spread in the azimuth direction. This process is known as *azimuth compression*. After range compression for either pulsed or LFM-CW SAR has been performed, azimuth compression is similar and is derived below. Figure 2.10 shows an example azimuth compressed waveform.

Azimuth compression is performed similarly to its counterpart in range for pulsed radar: matched filtering. However, whereas pulsed radar data are matched with the transmitted LFM chirp, in azimuth compression the data are matched with each range line's corresponding Doppler chirp. This chirp is a shift in the phase of each target's signature and is a direct result of the motion of the platform. Based on the assumption of accurately known platform velocity and range distance to target, the Doppler chirp for each range line can be analytically calculated.

Returning to the range compressed form of Eq. (2.20) and setting $t = t_o$ (remembering that now t represents "slow time" and the difference between t and t_o is inconsequential), the result is

$$m(t) = \frac{T}{4}e^{j\omega_o t_o - j\omega_c t_o}. \tag{2.31}$$

Because $\omega_c \gg \omega_o$, the $\omega_o t_o$ term may be assumed to be zero. It is the $\omega_c t_o$ term in Eq. (2.20) and Eq. (2.31) that is responsible for the azimuth chirp. Continuing from

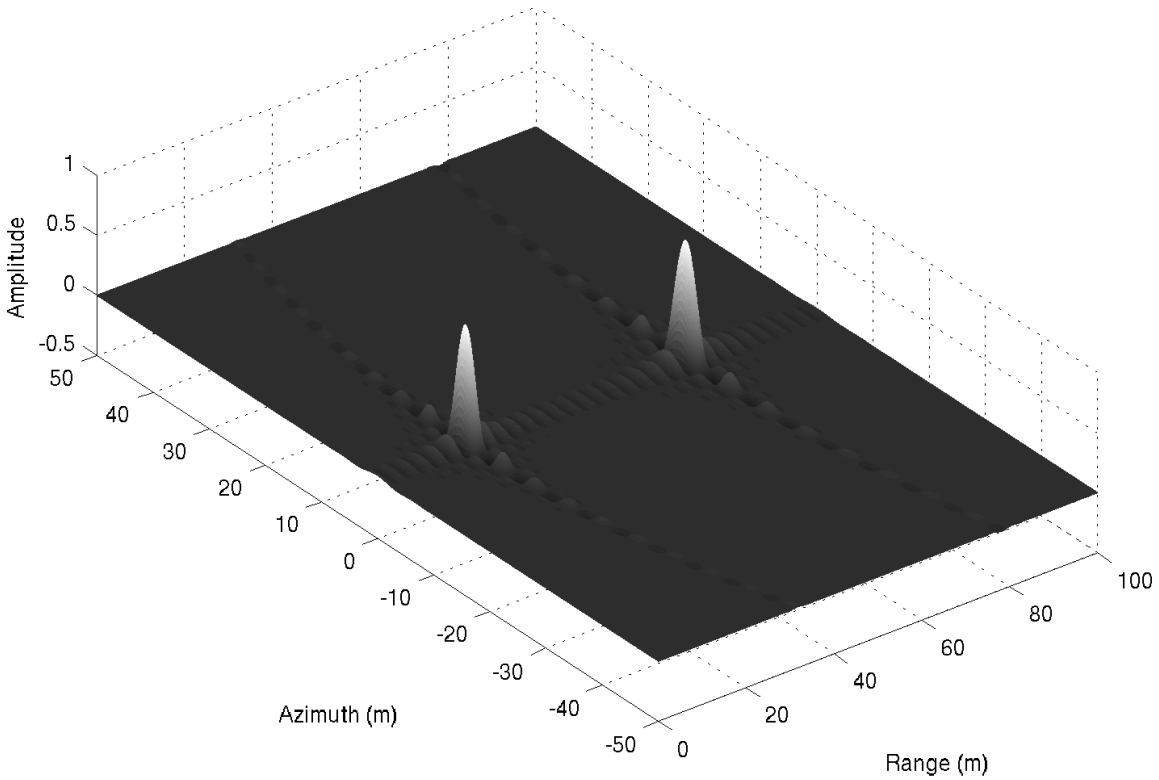


Figure 2.10: Example azimuth compressed waveform from a SAR.

above,

$$s(t) \simeq \frac{T}{4} e^{-j\omega_c t}. \tag{2.32}$$

Because the constant $\frac{T}{4}$ varies depending on the type of SAR model and is unimportant to the derivation, it is replaced with α for convenience. Letting $R(t)$ be the range to target, then the two-way time for the signal to target is given by

$$t = \frac{2R(t)}{c_o}. \tag{2.33}$$

Substituting this into Eq. (2.32) yields

$$\begin{aligned} s(t) &= \alpha e^{-j\omega_c t}, \\ &= \alpha e^{-j\frac{2\omega_c R(t)}{c_o}}, \\ &= \alpha e^{-j\frac{4\pi R(t)}{\lambda}}. \end{aligned} \tag{2.34}$$

$R(t)$ is of interest as it is responsible for the time varying phase shift of the signal. Returning to Eq. (2.10),

$$\phi_D(t) = \frac{2\pi v^2 t^2}{\lambda r},$$

which yields the phase term used for $R(t)$. Combining the previous result with Eq. (2.8) leads to

$$\frac{4\pi R(t)}{\lambda} \simeq \frac{4\pi R_o}{\lambda} + \frac{2\pi v^2 t^2}{\lambda R_o}. \tag{2.35}$$

Substituting this result back into Eq. (2.34), the final form of the signal used in azimuth compression can be written as

$$\begin{aligned} s(t) &= \alpha \exp\left(-j\frac{4\pi}{\lambda}R(t)\right), \\ &= \alpha \exp\left(-j\frac{4\pi R_o}{\lambda}\right) \exp\left(\frac{-j2\pi v^2 (t-t_o)^2}{\lambda R_o}\right). \end{aligned} \tag{2.36}$$

The final term contains the azimuth chirp, and becomes the basis for the matched filter

$$h_{az}(t) \triangleq \exp\left(\frac{-j2\pi v^2 t^2}{\lambda R_o}\right). \tag{2.37}$$

The two signals in Eq. (2.36) and Eq. (2.37) are correlated:

$$\begin{aligned} v(t) &= s(t) \otimes h^*_{az}(t), \\ &= \int s(t+\tau) h^*(\tau) d\tau \qquad \frac{-T_a}{2} < \tau + t - t_o < \frac{T_a}{2}, \\ &= \alpha\ e^{\frac{-j4\pi R_o}{\lambda}} \int e^{\frac{-j2\pi v^2}{\lambda R_o}(t-t_o+\tau)^2} e^{\frac{j2\pi v^2 \tau^2}{\lambda R_o}} d\tau, \\ &= \alpha\ e^{\frac{-j4\pi R_o}{\lambda}} \int e^{\frac{-j2\pi v^2}{\lambda R_o}\left((t-t_o)^2 + 2\tau(t-t_o)\right)} d\tau. \end{aligned}$$

Rearranging and applying the limits of integration,

$$\begin{aligned} v(t) &= \alpha\ e^{\frac{-j4\pi R_o}{\lambda} - \frac{j2\pi v^2 (t-t_o)^2}{\lambda R_o}} \int_{-\frac{T_a}{2}-(t-t_o)}^{\frac{T_a}{2}-(t-t_o)} e^{\frac{-j4\pi v^2 \tau (t-t_o)}{\lambda R_o}} d\tau, \\ &= \frac{j\alpha c_a}{c_b} e^{-jc_b\tau} \Big|_{\tau=-\frac{T_a}{2}-(t-t_o)}^{\tau=\frac{T_a}{2}-(t-t_o)}, \end{aligned} \tag{2.38}$$

where

$$c_a = e^{\frac{-j4\pi R_o}{\lambda} - \frac{j2\pi v^2(t-t_o)^2}{\lambda R_o}}, \tag{2.39}$$

$$c_b = \frac{4\pi v^2 \tau (t-t_o)}{\lambda R_o}. \tag{2.40}$$

Evaluating Eq. (2.38),

$$\begin{aligned} v(t) &= \frac{j\alpha c_a}{c_b}\left(e^{-jc_b\left(\frac{T_a}{2}-(t-t_o)\right)} - e^{-jc_b\left(\frac{-T_a}{2}-(t-t_o)\right)},\right) \\ &= \frac{j\alpha c_a}{c_b} e^{jc_b(t-t_o)}\left(e^{-jc_b\frac{T_a}{2}} - e^{jc_b\frac{T_a}{2}}\right), \\ &= \frac{-j\alpha c_a}{c_b} e^{jc_b(t-t_o)}\ 2j\sin\left(\frac{c_b T_a}{2}\right), \\ &= \alpha T_a c_a e^{jc_b(t-t_o)}\,\mathrm{sinc}\left(\frac{c_b T_a}{2}\right). \end{aligned} \tag{2.41}$$

Substituting c_a and c_b back in,

$$\begin{aligned} v(t) &= \alpha T_a e^{\frac{-j4\pi R_o}{\lambda} - \frac{j2\pi v^2(t-t_o)^2}{\lambda R_o} + \frac{j4\pi v^2(t-t_o)^2}{\lambda R_o}}\,\mathrm{sinc}\left(\frac{2\pi T_a v^2(t-t_o)}{\lambda R_o}\right), \\ &= \alpha T_a e^{\frac{-j4\pi R_o}{\lambda}} e^{\frac{j2\pi v^2(t-t_o)^2}{\lambda R_o}}\,\mathrm{sinc}\left(\frac{2\pi T_a v^2(t-t_o)}{\lambda R_o}\right). \end{aligned} \tag{2.42}$$

At this point, the signal $v(t)$ is given in terms of target dependent variables T_a and R_o. Rearranging the terms, it is possible to obtain a solution where the only target dependent variable is R_o, thus simplifying $v(t)$. If L_z is defined as the distance the radar platform moves while the target is illuminated, and v is the velocity of the platform, then

$$T_a = \frac{L_z}{v}. \tag{2.43}$$

This geometry is shown in Fig. 2.11. Continuing,

$$\begin{aligned} \tan\left(\frac{\theta_a}{2}\right) &= \frac{L/2}{R_o}, \\ &= \frac{T_a v}{2R_o}, \\ T_a &= \frac{2R_o}{v}\tan\left(\frac{\theta_a}{2}\right). \end{aligned} \tag{2.44}$$

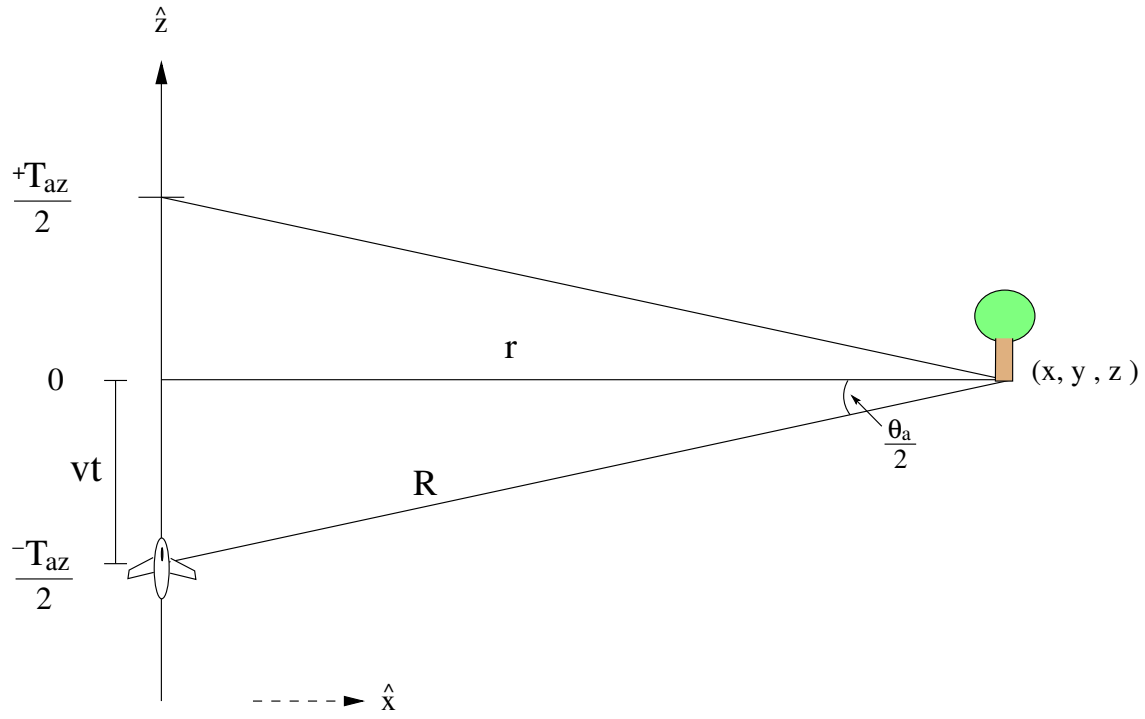


Figure 2.11: Overhead view of time/distance relationship of moving platform.

Substituting back into Eq. (2.42),

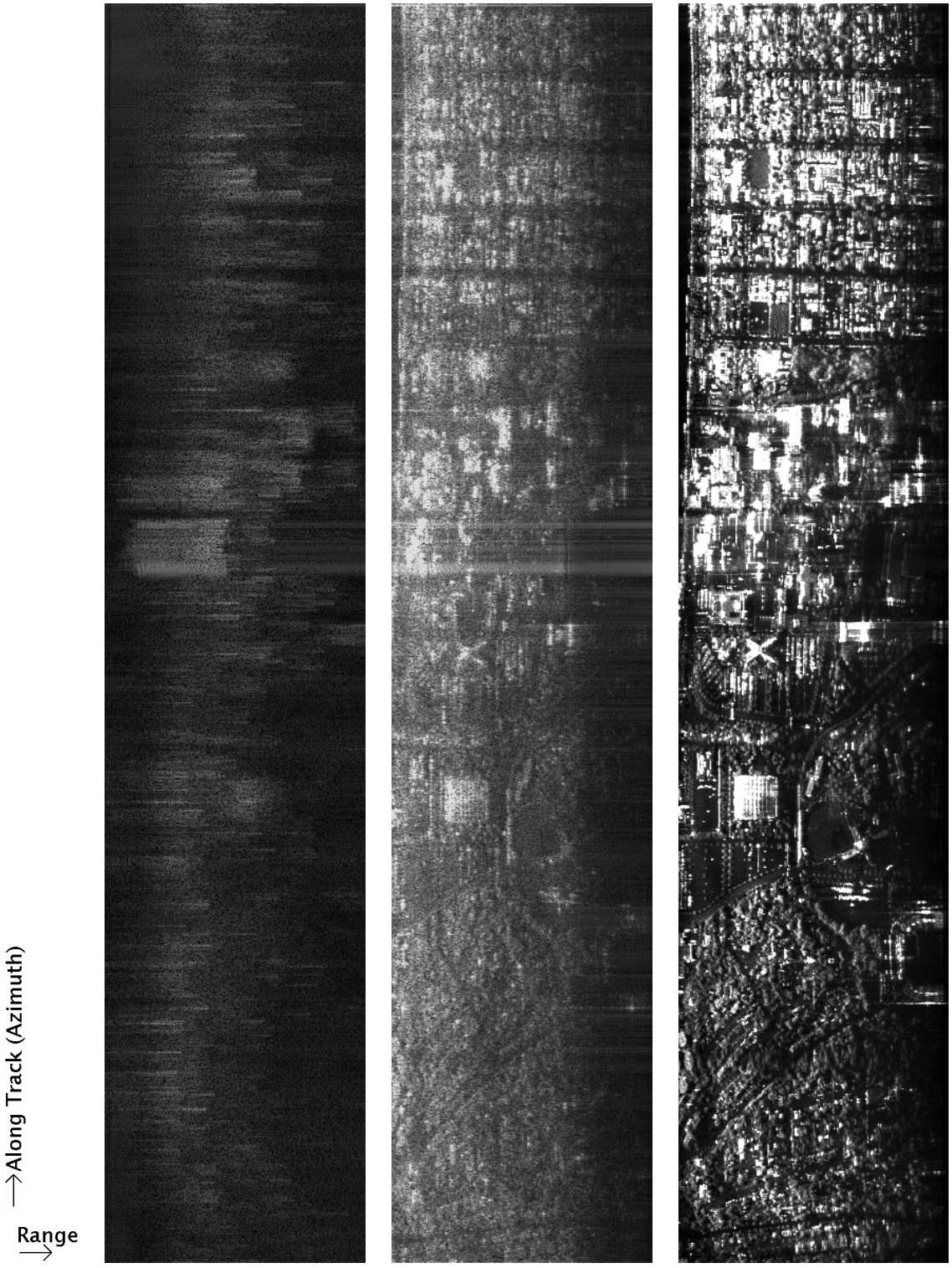
$$v(t) = \alpha T_a e^{\frac{-j4\pi R_o}{\lambda}} e^{\frac{j2\pi v^2 (t-t_o)^2}{\lambda R_o}} \operatorname{sinc}\left(\frac{4\pi v(t-t_o)}{\lambda} \tan\left(\frac{\theta_a}{2}\right)\right). \tag{2.45}$$

The sinc function is now independent of target parameters; however, the two exponential functions are still in terms of target-dependent R_o. The first is simply a constant phase, but the second is quadratic phase in time. Figure 2.12 shows an example of range and azimuth compression from the YINSAR.

A few remarks are necessary. While range compression is performed using a single known matched filter, namely the transmitted signal itself, azimuth compression must be performed on each range line independently. As seen in Eq. (2.37), the matched filter is dependent on both target range, R_o, and platform velocity, v. While the range to target can be calculated from the data given the range resolution, the velocity can not be calculated explicitly. Either the velocity must be recorded

as the platform is moving, or some kind of azimuth auto-focusing algorithm must be employed. Such an algorithm is discussed later.

After azimuth compression, the data are in complex form. Simple images are created by representing each pixel as the magnitude of each complex data point. Phase representations of the data are also possible. However, once the SAR compression schemes are completed and processed images are available, other processing techniques can be performed, such as interferometry [1] [5], as well as motion compensation [9] [12]. Interferometry involves using target phase differences from multiple images to create a topographical height-map of the imaged landscape. Motion compensation involves correcting for motion errors due to deviation from the assumed straight line, constant heading, constant velocity movement of the platform, thus producing a higher quality image.



(b) Raw Data (c) Range Compressed (d) Azimuth Compressed

Figure 2.12: Raw and compressed images of the BYU campus from YINSAR Flight-S, Oct 13.

Chapter 3

SAR Simulator

Sample data are necessary to validate the functionality of signal processing code. While data for this purpose could be collected from the μSAR itself, simulated data simplifies the process of verifying operational code. Using simulated data proves advantageous, as validated processing code also aides in fine-tuning the completed hardware. This chapter discuses the radar background and derivation used to design the simulator, the parameters of the LFM-CW μSAR that enter into the model, implementation of the simulator, and several results.

3.1 Radar Background

The simulator operates by simulating the motion of a platform and the radar above an imaginary scene. Point targets are placed in the scene and the radiation backscatter is calculated for each target as the platform moves over the entire scene. This operation is performed for each target individually, and the results are summed together to produce the complete backscatter waveform for the scene.

The model for the signal received by the platform from a single point target is

$$s_n(t) = P_D^{(n)}(t) \cos\left(2\pi\phi^{(n)}(t)\right), \tag{3.1}$$

where $s_n(t)$ is the received time-varying signal for the nth target, $P_D^{(n)}(t)$ is the power density received from target n, and $\phi^{(n)}(t)$ is a phase modulation term.

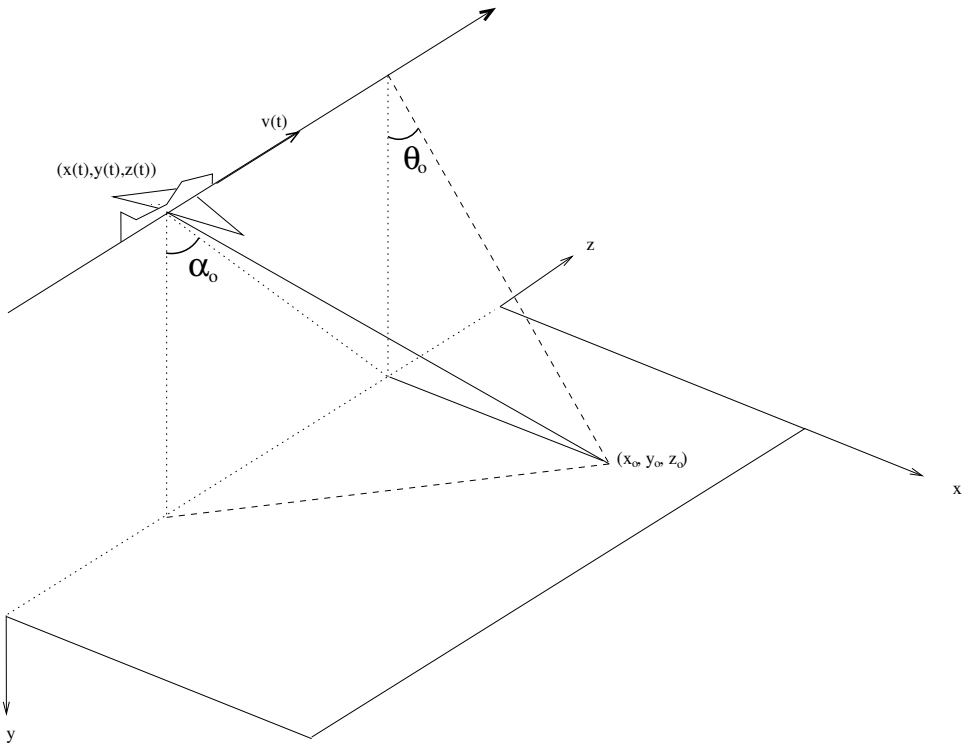


Figure 3.1: Simulator imaging geometry.

To model the backscatter power intensity off each target, the radar equation used is [13]

$$P_D(t) = \frac{P_t G^2(t) \lambda^2 \sigma_o}{(4\pi)^3 R_o^4(t)}, \tag{3.2}$$

where P_t is the transmitted power, $G(t)$ is the platform to target angle dependent antenna gain, λ is radar wavelength, σ_o is the normalized scattering coefficient

$$\sigma_o = \frac{\sigma}{\text{target area}}, \tag{3.3}$$

and $R_o(t)$ is the time-varying one-way distance from the platform to the target. Note that the superscript $^{(n)}$ has been dropped from target-dependent terms for notational convenience. This form is slightly modified for simulator use, by separating the antenna's azimuth gain, $G_{az}\big(\alpha_o(t)\big)$, from range gain, $G_r\big(\theta_o(t)\big)$, so that

$$P_D(t) = \frac{P_t G_{az}^2\big(\alpha_o(t)\big) G_r^2\big(\theta_o(t)\big) \lambda^2 \sigma_o}{(4\pi)^3 R_o^4(t)}, \tag{3.4}$$

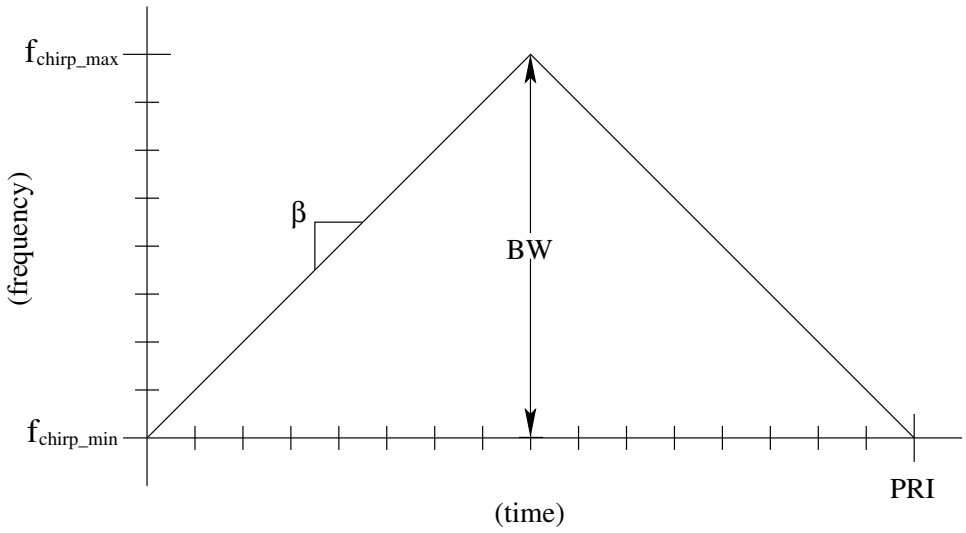


Figure 3.2: LFM-CW diagram of the chirp bandwidth and β versus time.

where

$$\theta_o(t) = \tan^{-1}\left(\frac{x(t) - x_o}{y(t) - y_o}\right) \qquad \text{and} \tag{3.5}$$

$$\alpha_o(t) = \tan^{-1}\left(\frac{z(t) - z_o}{x(t) - x_o}\right). \tag{3.6}$$

θ_o refers to the angle from the platform to the target in range, and α_o is the angle in azimuth. This geometry is shown in Fig. 3.1.

The phase term, $\phi(t)$, from Eq. (3.1) is given by

$$\phi(t) = \int_0^t \beta \frac{2R_o(\tau)}{c_o} + f_d(\tau)\, d\tau, \tag{3.7}$$

where $f_d(t)$ is the Doppler frequency shift, β is the slope of the up/down chirp, and c_o is the speed of light. The inside of the integral is expressed in terms of the frequency received after mixing.

The slope, β, is

$$\beta = \frac{BW}{PRI/2}, \tag{3.8}$$

where BW is the chirp bandwidth and PRI is the pulse repetition interval. A model of this is shown in Fig. 3.2.

The transmitted and received frequencies are

$$f_t(t) = f_o + \beta t, \tag{3.9}$$

$$\begin{aligned} f_r(t) &= f_o + \beta\,(t - t_o) + f_d \\ &= f_o + \beta\left(t - \frac{2R_o}{c_o}\right) + f_d, \end{aligned} \tag{3.10}$$

where t_o is the time delay from signal transmission to reception, f_o represents the carrier frequency of the radar, and R_o and f_d represent the instantaneous range to target and Doppler frequency shift at the moment of reception. Mixing the two signals and low-pass filtering yields

$$\begin{aligned} f(t) &= f_t(t) \otimes f_r(t), \\ &= \pm\beta\frac{2R_o}{c_o} + f_d. \end{aligned} \tag{3.11}$$

Integrating this term over time gives the phase term, $\phi(t)$, in Eq. (3.7). The $\pm$ stems from the fact that the frequency difference between the received and transmitted signals is positive for the up-ramping chirp, and negative for the down-ramping chirp. In practice, as only the magnitude of the frequency is sampled in hardware, the up-ramp and down-ramp components are processed separately.

The Doppler frequency shift $f_d(t)$ is given as [13]

$$f_d(t) = \frac{2v_{rel}(t)}{\lambda}, \tag{3.12}$$

where v_{rel} is the relative velocity of the platform with respect to the target. The relative velocity is calculated as

$$v_{rel}(t) = v(t)\sin(\theta_o)\sin(\alpha_o), \tag{3.13}$$

where θ_o and α_o are given in Eq. (3.5) and Eq. (3.6). Figure 3.3 shows an example of relative velocity. Substituting θ_o and α_o into Eq. (3.13) yields

$$\begin{aligned} v_{rel}(t) &= v(t)\sin\left(\tan^{-1}\left(\frac{x(t)-x_o}{y(t)-y_o}\right)\right)\sin\left(\tan^{-1}\left(\frac{z(t)-z_o}{x(t)-x_o}\right)\right), \\ &= v(t)\frac{x(t)-x_o}{\sqrt{(x(t)-x_o)^2+(y(t)-y_o)^2}}\frac{z(t)-z_o}{\sqrt{(z(t)-z_o)^2+(x(t)-x_o)^2}}. \end{aligned} \tag{3.14}$$

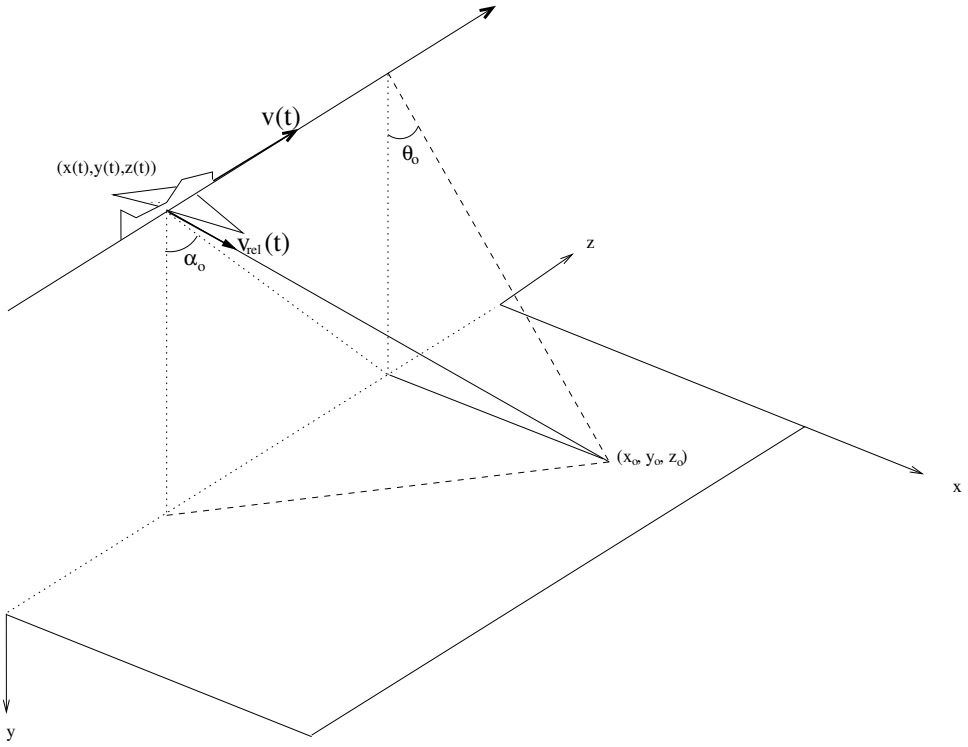


Figure 3.3: 2D antenna geometry used in simulation.

Substituting these into Eq. (3.7) along with Eq. (3.4) into Eq. (3.1) results in the equation used by the simulator:

$$\begin{aligned} s_n(t) &= P_D^{(n)}(t)\cos\left(2\pi\phi^{(n)}(t)\right), \\ &= \frac{P_t G_{az}^2 G_r^2 \lambda^2 \sigma}{(4\pi)^3 R_o^4(t)} \cos\left(2\pi \int_0^t \beta \frac{2R_o(\tau)}{c_o} + \frac{2v(\tau)\sin(\theta)\sin(\alpha)}{\lambda}\, d\tau\right). \end{aligned} \tag{3.15}$$

Finally, the signals of all N targets are summed together to produce the desired simulated data

$$s(t) = \sum_{n=1}^{N} s_n(t). \tag{3.16}$$

3.2 Implementation

As stated previously, the simulator operates by simulating the motion of a platform above given point targets. The platform "flies" above the scene on a given path in discrete increments. At each stop, one pulse repetition interval worth of samples are calculated according to equations (3.15) and (3.16). Although the signal

Table 3.1: Parameters used in simulation. Actual μSAR values are listed above the double line.

Parameter	*Value*
Wavelength (λ)	0.0544m
Chirp bandwidth (BW)	80MHz
Pulse repetition frequency (PRF)	330Hz - 3300Hz
Power (P_t)	1W
Sampling rate (S_{rate})	330kHz
Azimuth gain (G_{az})	12dB
Range gain (G_r)	12dB
Azimuth beamwidth (ϑ_{az})	12°
Range beamwidth (ϑ_{az})	45°
Backscatter coefficient (σ_o)	-15dB
Range to target (R_o)	40m - 1000m
Platform velocity (v)	20m/s - 50m/s

is continuous, the simulator discretely samples according to the PRF, which is chosen to meet all Nyquist requirements. Integration then is simply the cumulative sum of the samples.

To speed processing, signal returns of targets are only calculated if the azimuth angle for a given pulse is less than 40° off broadside. Otherwise, the contributing pulse is assumed to be negligible and is set to zero. A separate function is used to calculate the phase modulation term (the $cos()$ term in Eq. (3.15)), which begins with the value of the ending phase of the previous pulse. The calculations for each contributing target are summed together and the simulated platform moves to the next interval.

This process is continued through the entire path of the platform. Once its destination is reached, the simulated data have been completely generated and are then saved to disk for processing. At this point, noise may also be added to the simulated data to test processing robustness.

Table 3.1 lists the parameters used in the simulation. Note that the true hardware parameters are given above the double line. The backscatter coefficient,

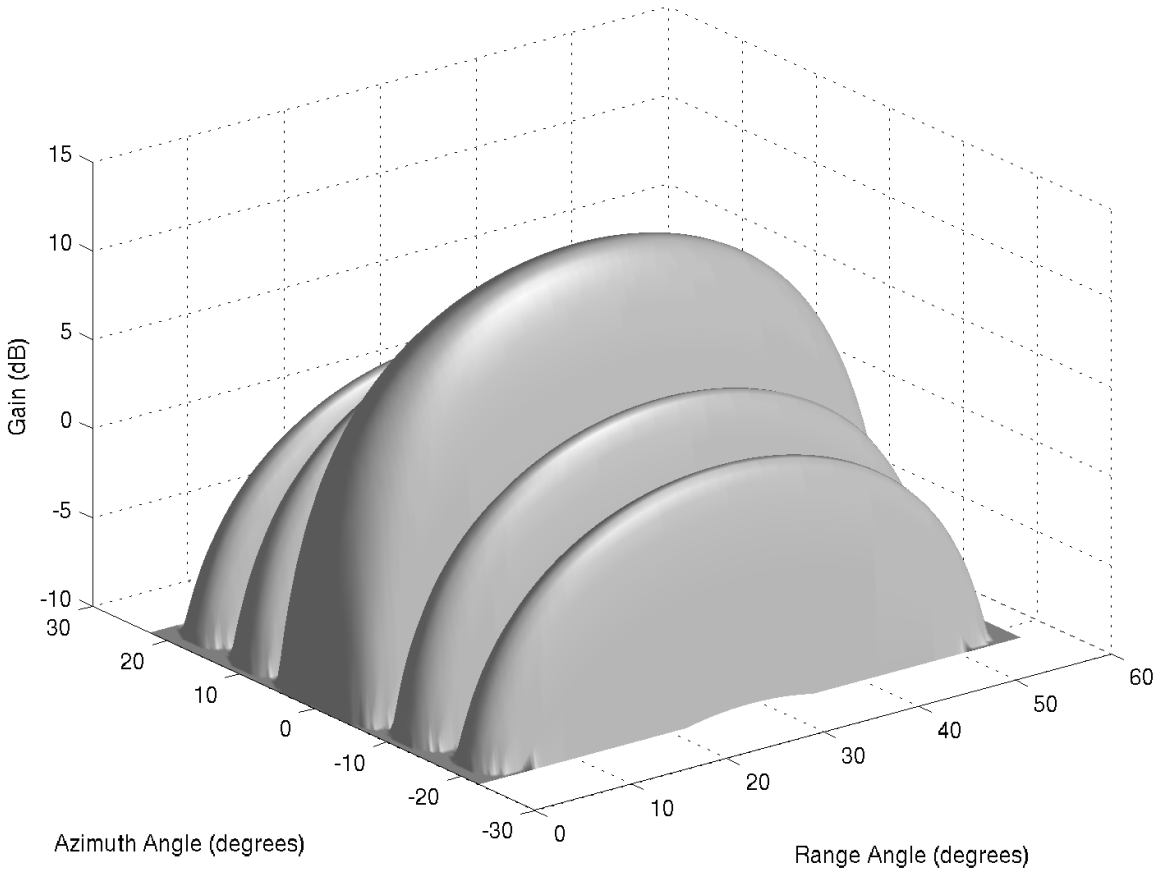


Figure 3.4: 2D antenna gain pattern used in simulation.

σ_o, is chosen somewhat arbitrarily as it adds a scale factor to the final data without actually affecting the simulation results. The antenna gain and beamwidth are chosen to be realistic, although they were not picked with respect to actual design parameters because they were not yet available. Figure 3.4 shows a 2D representation of the antenna gain used in the simulation. The peak value at the center is 12dB.

Simulated targets are placed in various patterns to test the range and azimuth functionality, as well as resolvability of geographically close targets. Figure 3.5 shows the various processing steps of compressing a single point target. The first plot shows the location of the single target. Following it is a plot of the simulated raw data, where the waveform vector is divided into range segments corresponding to samples that lie within one pulse repetition interval. Note that the discontinuity in the center is not due to the target's position, but rather the switch between up

and down ramping chirps. Next is the processed range compressed image, and lastly the azimuth compressed image. Range and azimuth compression are performed as presented in Sections 2.5 and 2.6. The final image shows the target "spread" slightly in azimuth. This imperfect compression is due to the *sinc* style antenna pattern used in simulation.

Figure 3.6 is an example of processing several point targets arranged in various patterns. Superposition of each target's contributing signal is clearly expressed in the raw data plot. The points in these patterns are aligned closely to show resolvability of the targets after compression. As can be seen, even targets closer than half a meter apart in azimuth can be separately distinguished. As expected, targets farther in range are less bright.

The final test of the simulator involves using an actual photograph as the input scene to the simulator. A scene point target is created for each pixel in the photograph, where the σ_o value of each target is a scaled intensity of the corresponding pixel in the photograph of Fig. 3.7(a). The points are then aligned in a grid-like fashion, spaced so the image fills the antenna swath in range and are a meter apart in azimuth. The same simulation is then performed on the entire scene, and compressed using the same techniques described previously.

Figure 3.7(b) shows the image of the processed output. The data have been looked (averaged) in the azimuth direction to maintain the same aspect ratio as the input photograph. Although the mean brightnesses of the two images vary and the simulation output shows SAR processing artifacts, the output is easily distinguishable as originating from the input photograph.

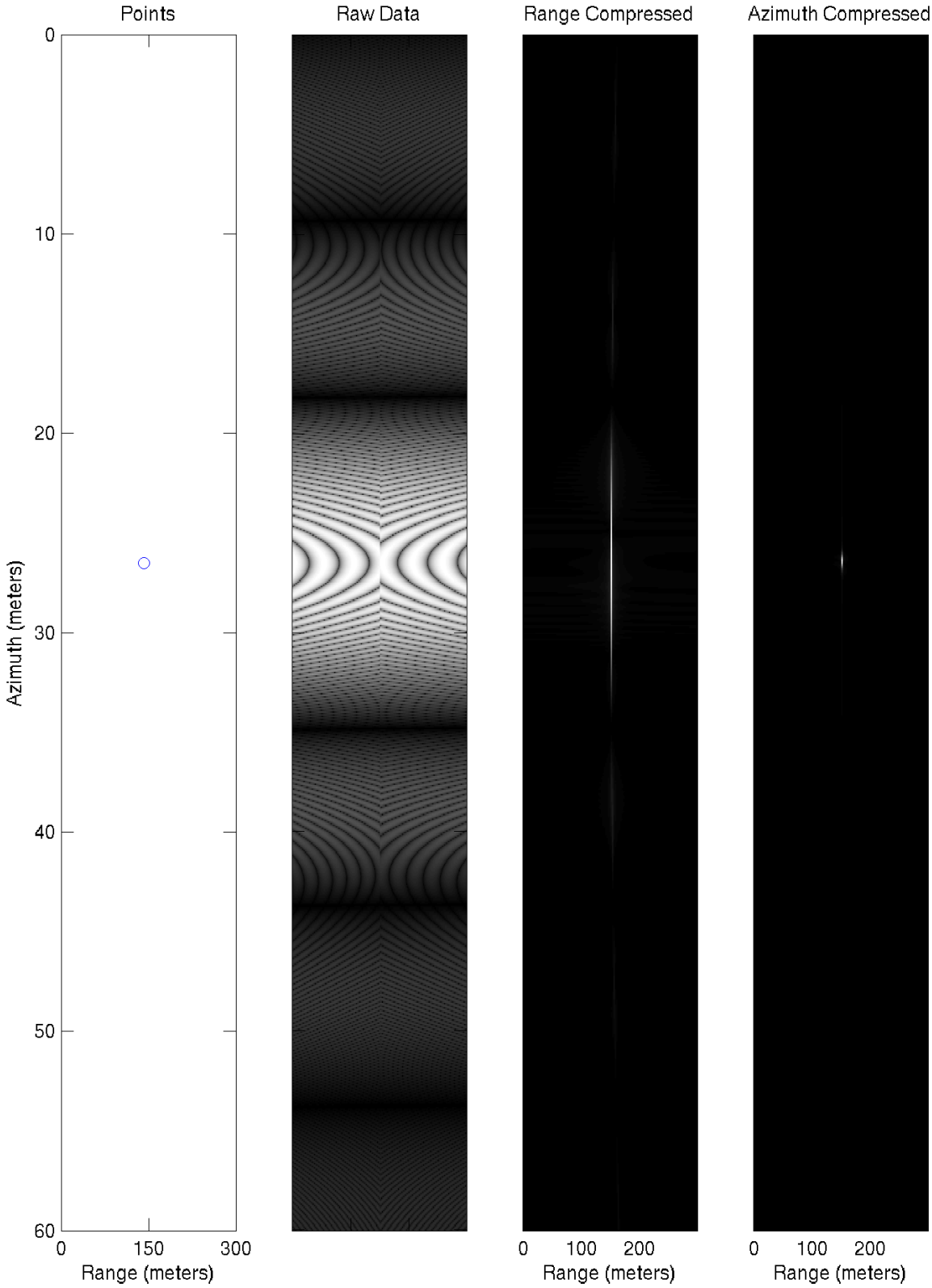


Figure 3.5: Simulation of a single point target through various processing steps.

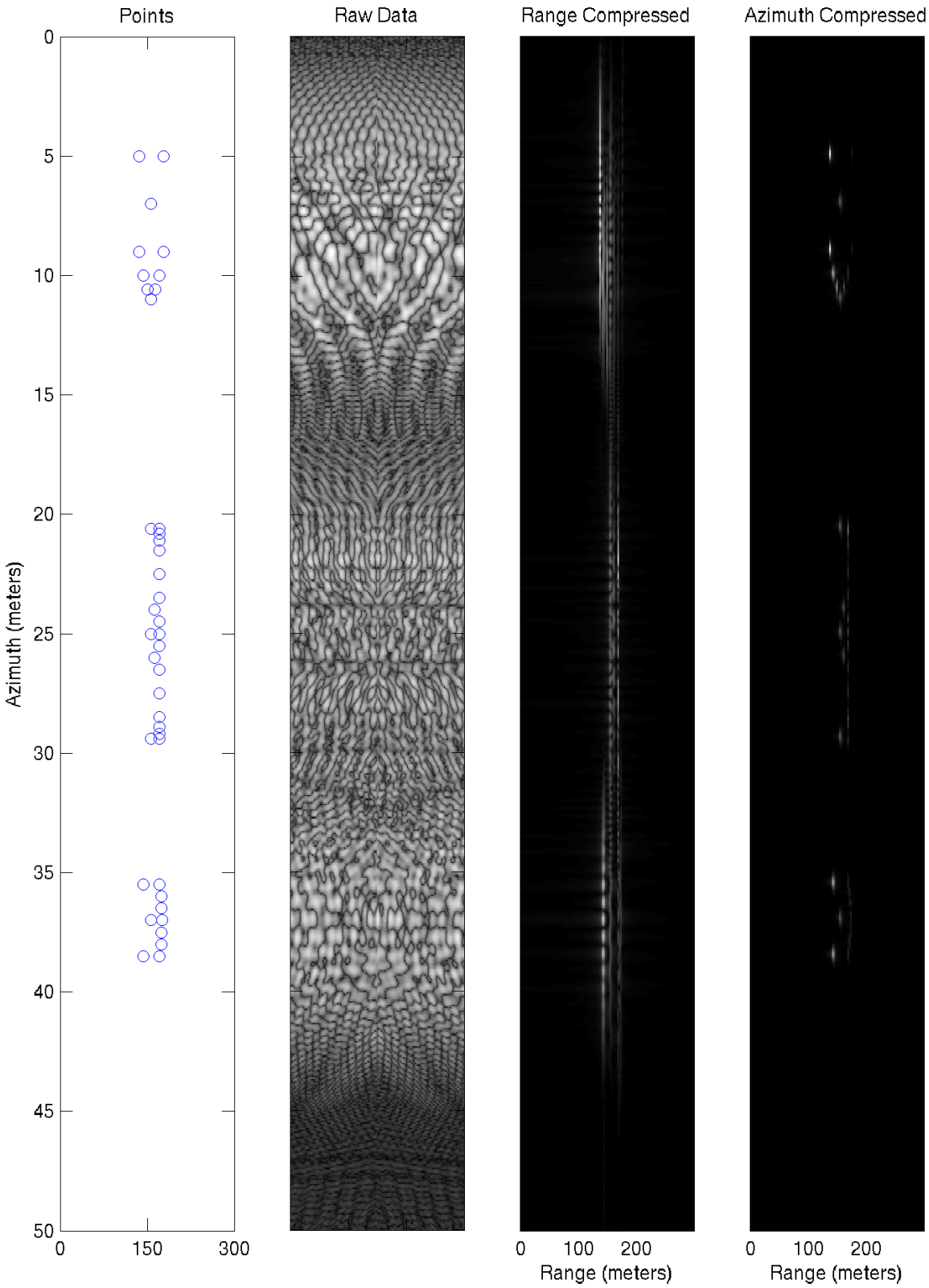
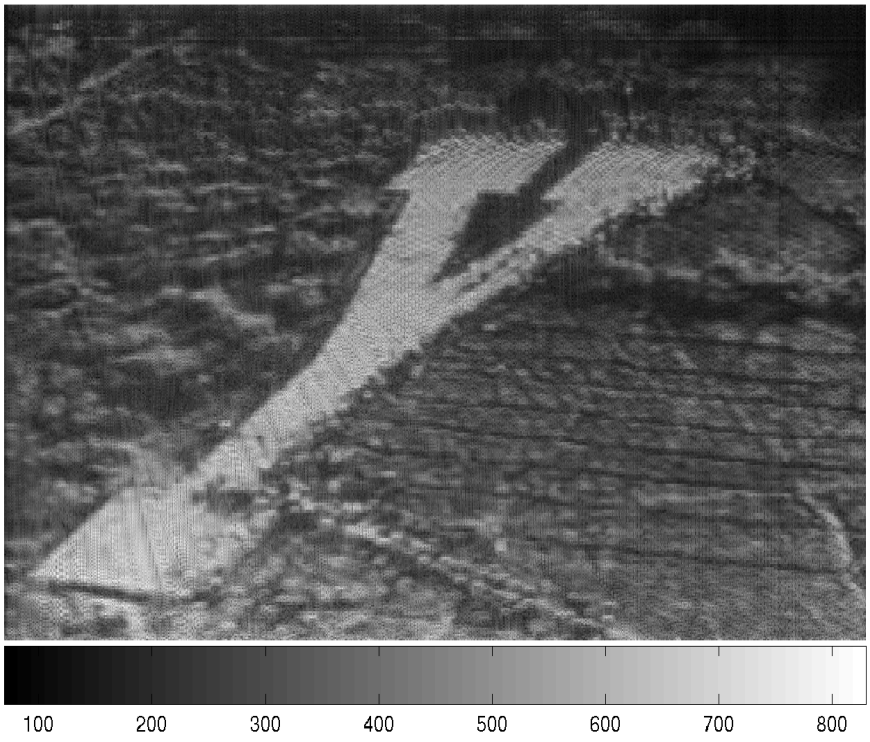


Figure 3.6: Simulation showing multiple point targets. The targets are spaced closely together to exhibit target resolvability.

(a) Photograph used as input to the simulator.



(b) Simulator output of processed photograph.

Figure 3.7: Simulated distributed target scene input and output.

Chapter 4

Data Processing

This chapter discusses the implementation of the μSAR image processing algorithms. The μSAR compression techniques are detailed, followed by the chirp detection, data filtering and auto-focusing algorithms. Finally, results from actually collected data are presented.

4.1 Compression

As discussed in Chapter 2, the received signal is de-chirped to lower the A/D sampling rate requirements. De-chirping, multiplication of the transmitted signal with the received signal and low-pass filtering, results in a frequency which is linearly related to the target range. This signal is stored and subsequently processed.

Because frequency in the sampled signal corresponds to the time interval between signal transmission and reception, the frequency of each target's signature is range dependent. Therefore, the received data can be modeled as existing in the frequency domain for the range direction. The discrete Fourier transform (DFT) separates these signals by their corresponding frequency, thus separating the targets in range. This means the process of range compression for LFM-CW SAR is simply to take the DFT of the raw data in the range direction, essentially transforming it back into the spatial domain.

Azimuth compression is performed exactly as outlined in Chapter 2. Each target's signature is present in azimuth over many range lines, each bin differing only in phase. This phase is equivalent to a frequency modulated chirp that can be analytically calculated based on the relative platform velocity to target. Matched filtering

of this chirp with the range compressed data constitutes the process of azimuth compression. The relative velocity of the platform and target is different at each range line; thus, a new chirp must be calculated for each range. Matched filtering is performed separately on each range line with its associated azimuth Doppler chirp, with the result being the final range compressed image.

4.2 Chirp-start Detection

In pulsed SAR, A/D data are only collected for a small time interval after each pulse is transmitted. Because of the nature of pulsed SAR, each pulse is separately received and recorded. Each individual pulse can then be compressed in range and azimuth. For LFM-CW SAR, however, sampling is continuous. Without information to indicate what the transmit signal is doing, it is not immediately obvious how to distinguish the beginnings of each repeated up/down frequency ramp. (i.e. without triggering, how do we find the start of cycle with only the raw data?)

For the purpose of maintaining phase continuity and predicting the azimuth phase, it is essential that each pulse be separated at the boundaries between transmitted up/down frequency ramps. One possible way to achieve this is to trigger the data collection externally from the transmitter. While this solution creates the desired result, it has the drawback of complicating the design and rendering a collection useless if there are any data sampling or triggering problems. Therefore, it is desirable to have a method for determining the pulse separations given the data itself.

Several techniques attempting to achieve chirp-start separation were tried. These processes include matched filtering using a known up/down ramp, amplitude tracking, and cross-correlation of the perspective up and down ramp separations. The former two methods performed satisfactorily only in cable feed-through tests, thus unable to provide results reliable enough to be used in production. The latter algorithm using cross-correlation between separated up/down ramps is the only method dependable enough to be used in practice.

This method begins by separating the data into suspected up and down chirp portions and range compressing these portions separately. These two range compressed images are formed into vectors. The cross-correlation of the two vectors is then calculated and saved. The process is then repeated for a new predicted chirp-start offset. The prediction with the highest cross-correlation is taken as the correct beginning offset. In practice, the predictions are some subset of all possible starting-sample offsets. The theory is that range compressed images separately created with up and down chirp data are best correlated when the two are divided correctly. The mathematical description of the algorithm follows.

The collected raw data vector, $s[\ ]$, is separated into matrices $\mathbf{r}^{up}$ and $\mathbf{r}^{down}$, corresponding to the up and down chirp segments of the data respectively. This is represented as

$$\mathbf{r}^{up}_{m,n} = s\Big[mN + n + k\Big] \qquad \text{where } 0 \leq n \leq \tfrac{N}{2} \tag{4.1}$$

$$\mathbf{r}^{down}_{m,n} = s\Big[(m + \tfrac{1}{2})N + n + k\Big], \tag{4.2}$$

where N is the number of samples per period, m indicates the m^{th} azimuth line, and k is the chirp-start offset. Taking the Fourier transform,

$$\mathbf{R}_{up} = \mathcal{F}_n\{\mathbf{r^{up}}\} \tag{4.3}$$

$$\mathbf{R}_{down} = \mathcal{F}_n\{\mathbf{r^{down}}\}, \tag{4.4}$$

where $\mathcal{F}_n$ represents the Fourier transform in the n (range) direction. Continuing,

$$\vec{u}_{up} = \text{vec}\{\mathbf{R}_{up}\} \tag{4.5}$$

$$\vec{u}_{down} = \text{vec}\{\mathbf{R}_{down}\} \tag{4.6}$$

$$v[k] = \Big|\vec{u}_{up} \star \vec{u}_{down}\Big|, \tag{4.7}$$

where the $\star$ operator represents cross-correlation and the vec operator creates a column vector from a matrix. Note that the sample data are transformed from a vector into a matrix, then back to a vector again for the purpose of range compression. The index, k, of v with highest magnitude corresponds to the chirp-start offset with the highest correlation.

For each period of samples, there are two offsets that produce near identical peaks. One peak corresponds to the start of the up-ramp samples, and the other to the start of the down-ramp samples. By performing azimuth compression on data processed at both offsets, the up-ramp portion is easily identified since up-ramp azimuth compression on the down-ramp data further blurs the image.

Auto-determining the beginning of the chirp using this algorithm proves to be very effective, even in low SNR situations. Once the chirp starting offset is located, μSAR processing continues as before.

An example result of this algorithm is displayed in Fig. 4.1. The cross-correlation magnitude is displayed for different starting offsets. There are $N = 466$ samples per up/down chirp and the plot displays $1.25N$ offsets. Highest peaks appear at offsets 191 and 424, exactly $N/2$ samples apart. After azimuth compressing one window of data, offset 191 is determined to be the peak corresponding to the beginning of the up-chirp. Similar results are obtained for all data collections to date.

4.3 Interference Filtering

The center frequency for the μSAR is chosen at 5.5GHz, as it is an unlicensed band. However, 5.5GHz falls in the center of the frequency band used by IEEE 802.11a (Wi-Fi) [14]. In urban areas the wide use of Wi-Fi results in interference. This will not be the case for the actual μSAR operation off the coast of Alaska.

The interference due to Wi-Fi causes artifacts in processed images. These artifacts are frequency bands in the raw data which correspond to the frequencies used by various Wi-Fi channels. While these bands change from collection to collection, only portions of the data corresponding to those channels are corrupted. Thus, a method is developed for filtering these bands in order to achieve higher quality compressed images.

The filtering operation begins by detecting portions of the raw data where a significant number of sample points exhibit much higher than average power. Then,

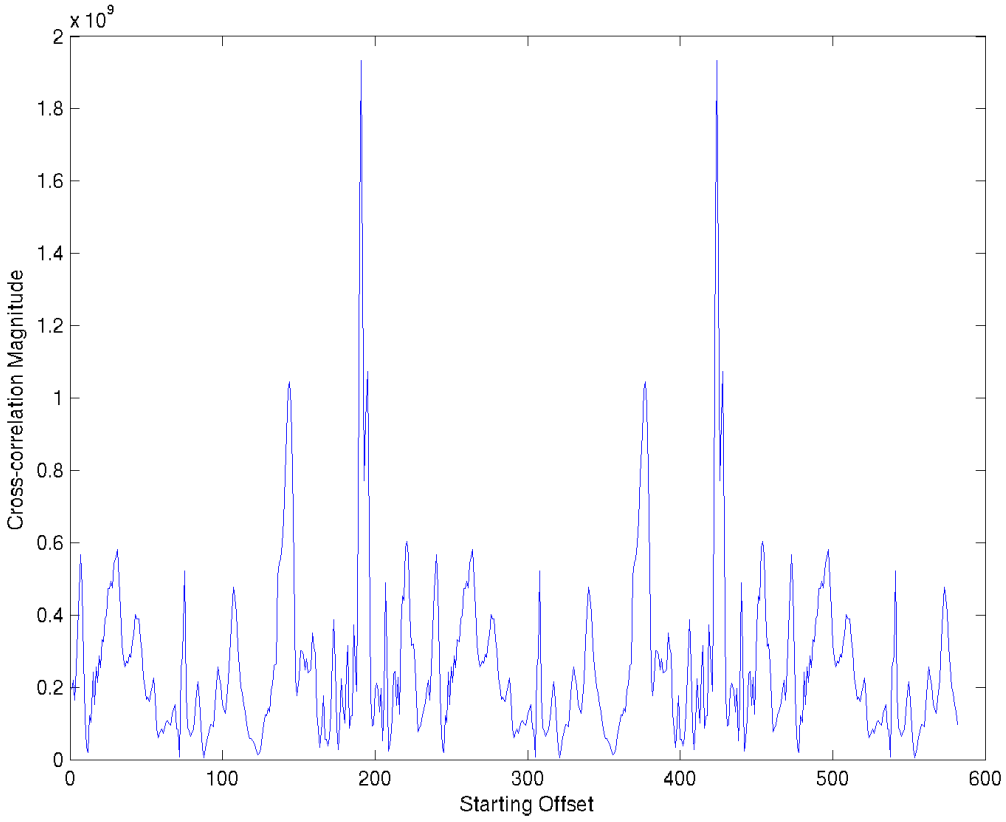


Figure 4.1: Plot showing the results of the chirp-start detection algorithm.

a rectangular window encapsulating those portions of data is created and the samples there are set to the mean value of the data.

Figure 4.2 shows an azimuth compressed image with and without Wi-Fi filtering from a collection made on June 21, 2004 at 1:23pm from a van (as described later). Each of the bright points corresponds to an individual corner reflector or street lamp. As seen in the figure, this processes of filtering, or excision, leads to a clearer image with more of the scene visible. Note that some of the targets appear spread in azimuth, due to acceleration and deceleration of the van, since a constant velocity was assumed for azimuth processing.

4.4 Auto-focus Algorithm

The velocity of a SAR platform is not constant and sometimes imprecisely measured. Small deviances in velocity affect the azimuth matched filter calculated,

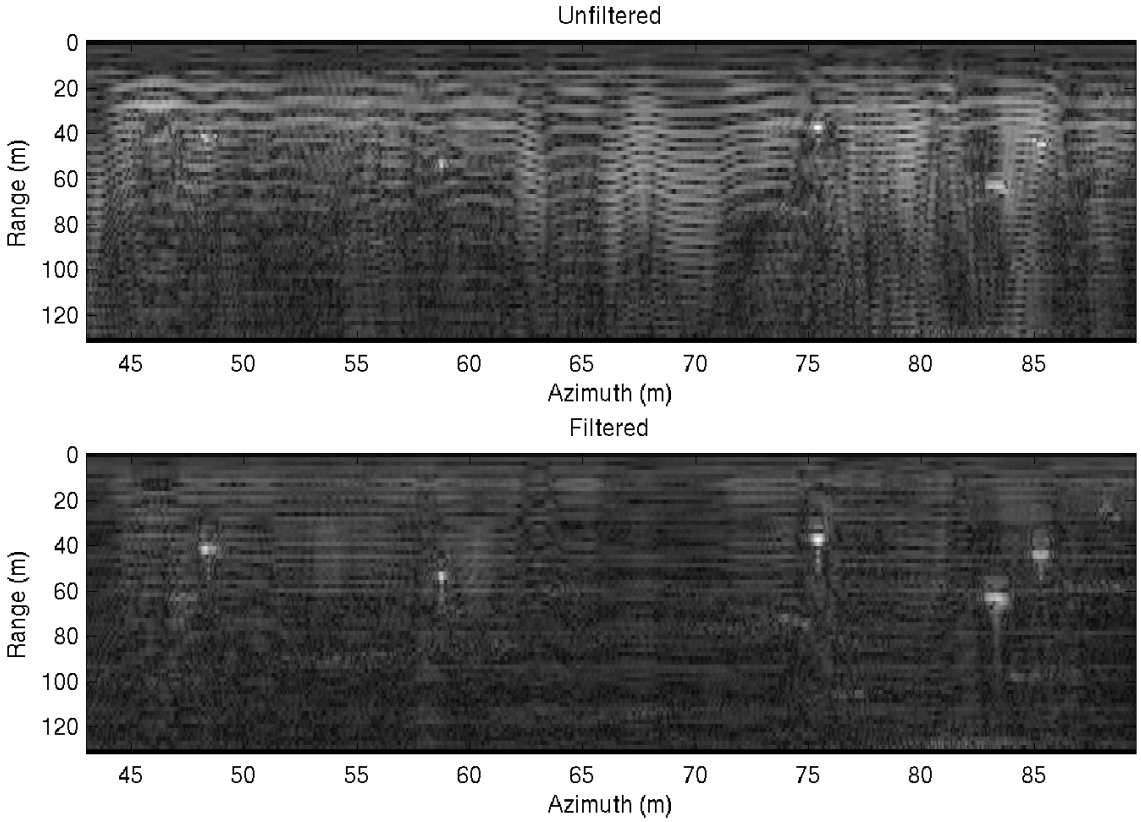


Figure 4.2: The upper image shows an azimuth compressed scene without data filtering, while the lower image shows the same scene with filtering out of interfering Wi-Fi signals. The bright targets correspond to corner reflectors and street lamps.

resulting in target smearing in azimuth. Because azimuth compression is so strongly dependent on precise platform velocity, it is desirable to produce accurate velocity data for a given data collection.

For this reason, an auto-focusing algorithm is employed to create a more focused image through velocity manipulation. The auto-focusing algorithm is based on increasing the contrast in the azimuth compressed image by adjusting the velocity used to calculating the matched filter. The algorithm begins with range compressed data and operates by performing azimuth compression over a range of velocities, centered at the assumed platform velocity.

At each velocity tested, azimuth compression is performed and the first derivative is calculated in the azimuth direction for each range line. The maximum value

for each range line is calculated (each range line's sharpest, brightest target) and of these, the maximum value is chosen. This value corresponds to the sharpest, brightest target in the entire image and is stored, then compared to all others for tested velocities. The maximum of these values represents the velocity with the sharpest, brightest point target and is used to create the azimuth compressed image.

Because velocity is seldom constant over an entire collection, the collection is divided into separate velocity windows. The velocity calculated for each window is then used in processing, with an overlap-and-save technique.

An example of this process is shown in the images of Fig. 4.3. The upper image is the interference-filtered, azimuth compressed scene of Fig. 4.2 with a constant velocity of 12.0 m/s used in compression. The lower image shows the auto-focused version using velocity windows of about 18.7 meters. The dark vertical lines barely visible every 18.7 meters are the edges of the windows and are due to discontinuities in velocities across the window boundaries. The auto-focused velocities represent a difference spanning about 1.1 m/s. While point targets are still clearly distinguishable in the upper image, targets in the auto-focused image are more sharply focused and have higher SNR as a function of time. The velocity used over the entire scene is plotted in Fig. 4.4.

A plot showing the contrast versus chosen velocities is presented in Fig. 4.5. This plot represents the maximum azimuth derivative at each velocity chosen for compression in the first velocity window of the image. The nominal velocity for the collection is 12.0 m/s, so with a contrast peak at -1.5 m/s, the resulting velocity used in azimuth compression for the first window is 10.5 m/s.

4.5 Results

While μSAR is designed for UAV operation, test collections are made with the μSAR antennas mounted to the side of a van. This configuration is shown in the image of Fig. 4.6. The data are collected as the van drives by different scenes, including urban areas and a canyon. Results of two data collections are presented. The first is a corner-reflector test collection made in a parking lot. The second is a

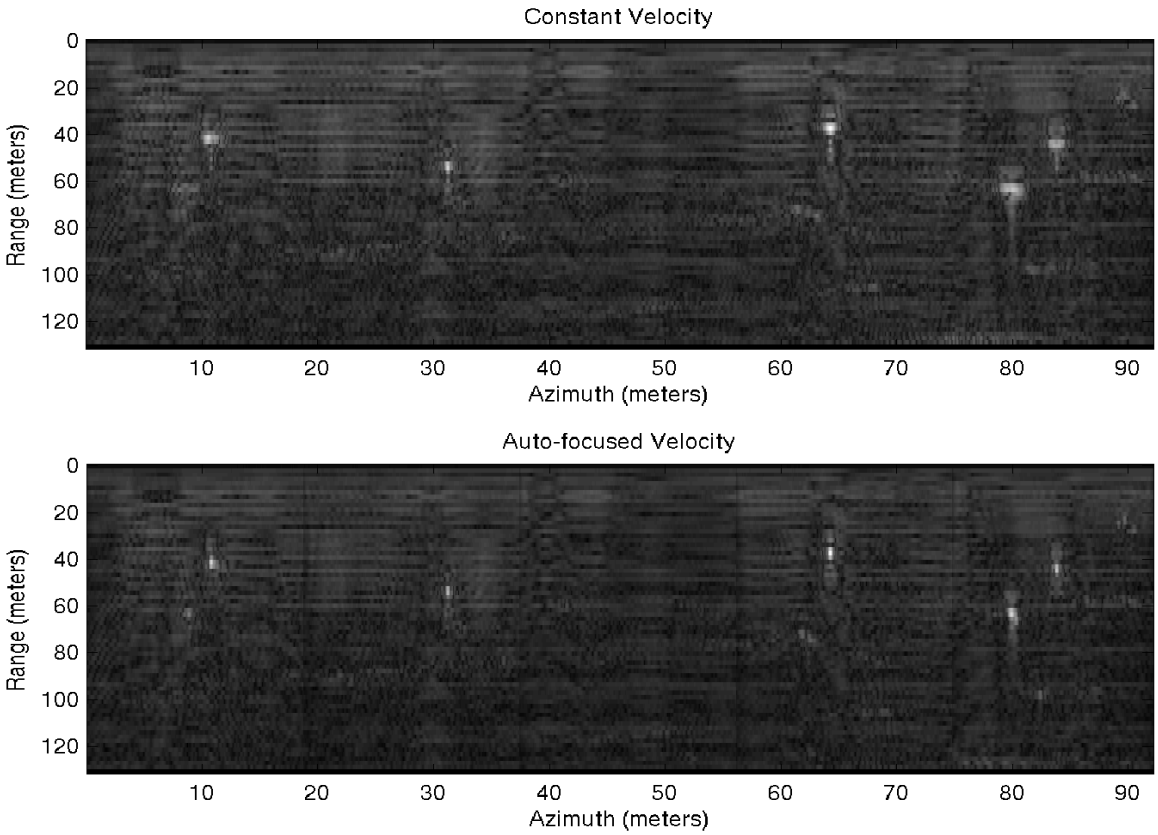


Figure 4.3: The upper image shows a scene using a constant velocity, while the lower image shows the same scene with auto-focusing. The auto-focused image results in better compressed targets.

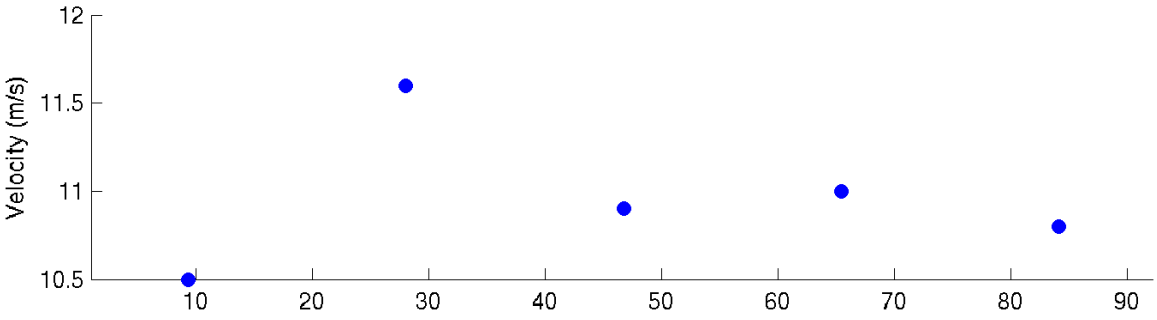


Figure 4.4: Velocities chosen for each of the velocity windows.

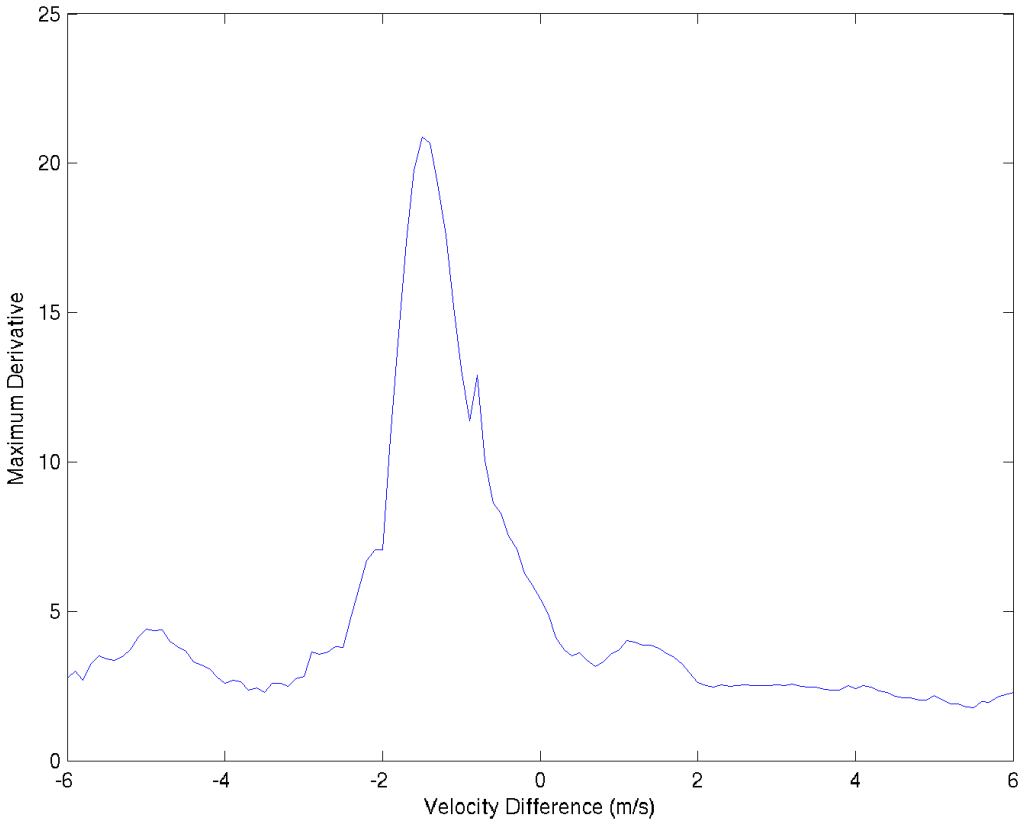


Figure 4.5: Contrast vs. velocity difference for the first velocity window. The peak at -1.5 m/s corresponds to the velocity producing the highest contrast.

collection that more closely resembles an actual UAV platform, made by recording data while driving through a canyon.

4.5.1 Corner-reflector Scene

The corner-reflector scene test collection was made on June 21, 2004 at 1:06pm in the parking lot South-East of LaVell Edwards Stadium in Provo, Utah. This collection represents a test of the μSAR equipment to ensure proper functionality of both the hardware and compression algorithms by imaging point-targets. Three corner reflectors are placed on the ground facing normal to the path of the van. The platform velocity was roughly measured at 30 mph.

Figure 4.7 shows the raw, unfiltered μSAR data. As data are collected, they are stored as a single vector representing the entire stream of digitized samples. The

Figure 4.6: Photograph of the μSAR test setup with antennas mounted to the side of a van.

vector of samples is then transformed into a matrix, with each PRI of samples on a separate row. This forms a two-dimensional image with azimuth on one axis and range on the other.

The up-ramping portion of the data is located between range bins 0 and 232 and the down-ramping between 233 and 466. Heavy interference is seen running in azimuth at the center and both edges of the image. These areas are candidates for filtering. The sinusoidal waves centered near azimuth bins 6000 and 7000 correspond to bright corner reflectors.

The two images in Fig. 4.8 are the range compressed images, separated by up and down chirps and interference filtered. The process of range compression is the same for both up and down chirp images, except that the frequency is reversed for

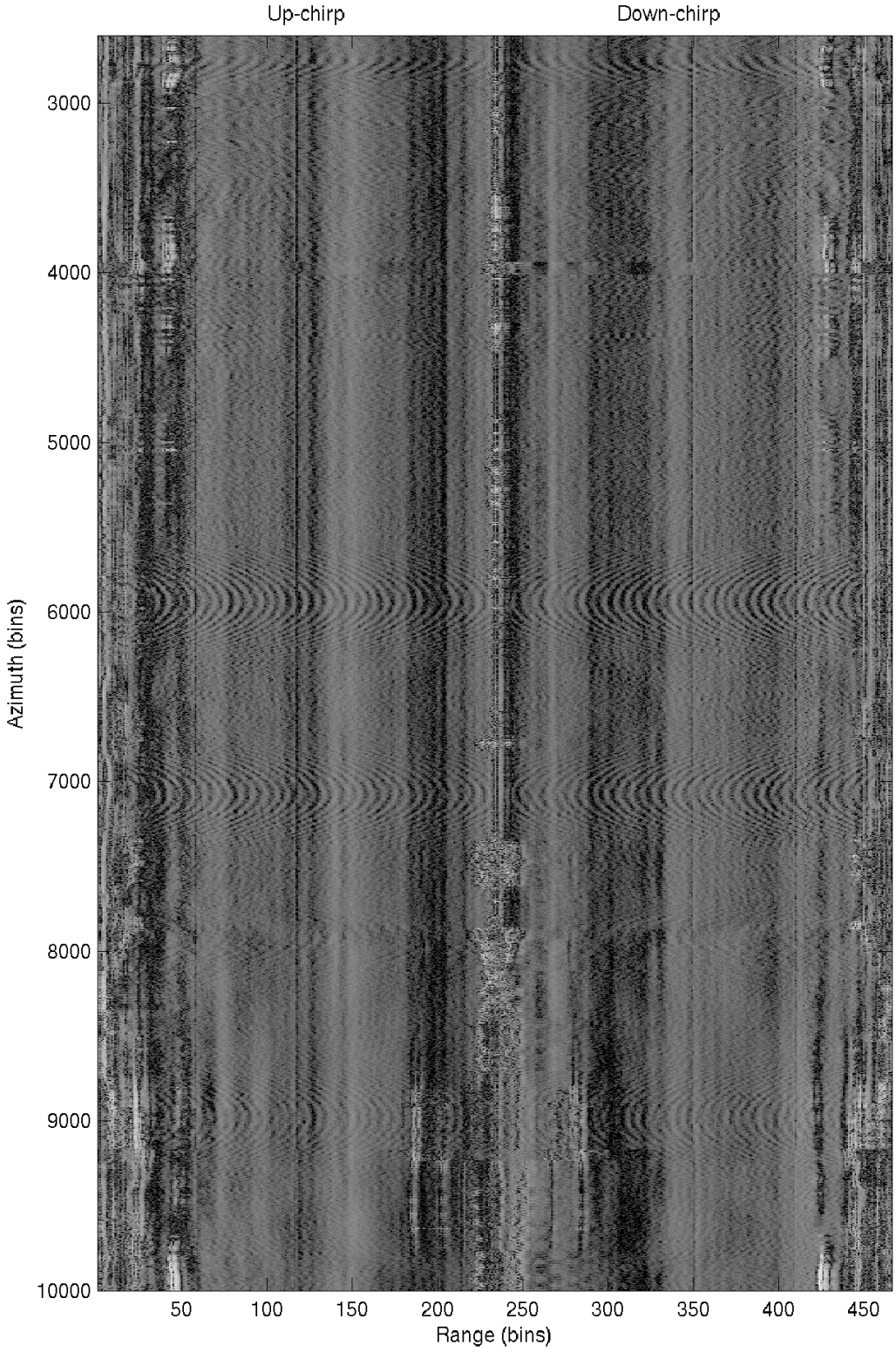


Figure 4.7: Image showing the raw data without filtering. Range bin 233 separates the up-chirp (left side of the image) from the down-chirp (right side of the image).

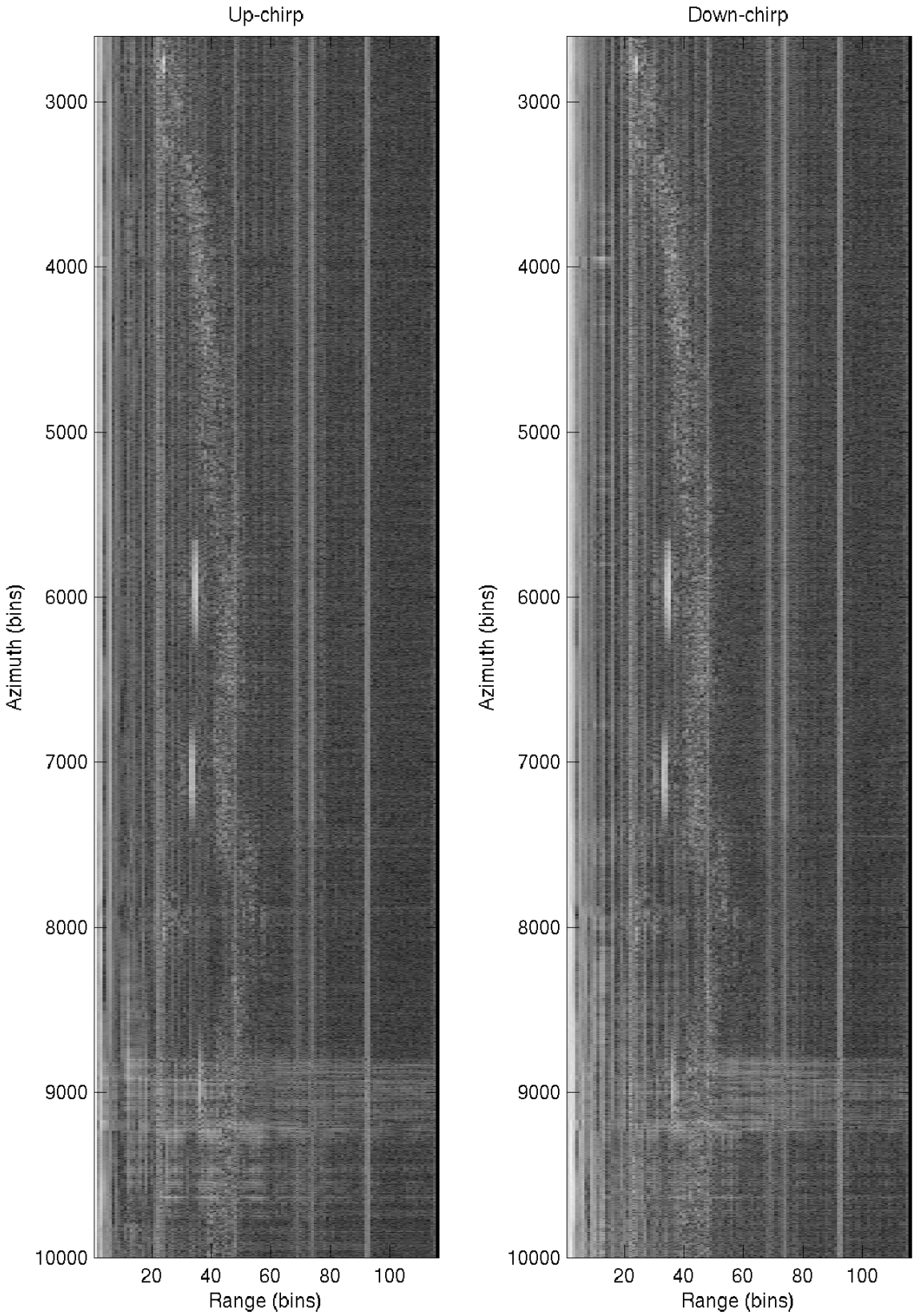


Figure 4.8: Display of the separated, range compressed images.

the down-chirp image. Reversing the range bins of the down-chirp image yields the same spatial reference as the up-chirp image. The two bright targets previously seen centered near 6000 and 7000 in azimuth are now completely compressed in range. The two images appear nearly identical, except for a few areas due mostly to noise.

Figure 4.9 gives the final, azimuth compressed up/down ramp images. Azimuth compression for the two images differs only by the fact that the complex conjugate of the Doppler matched filter must be used in the case of the down-chirp image, again due to frequency reversal. Residual noise artifacts are seen near 80 meters in azimuth.

Figure 4.10 shows the average of the compressed up/down ramp images. Averaging the images results in higher SNR. Targets are now more visible and noise is less apparent. The long line at range 170 meters is an artifact due to a harmonic of the power supply.

The final image shows some corner reflectors placed between 20 and 30 meters away from the track of the van. The rough patterns just farther in range than the corner reflectors are a hill with trees behind the reflectors. There is some interference present at about 80m in azimuth.

4.5.2 Canyon Scene

The canyon scene collection was made on November 23, 2004 at 3:52pm in Provo Canyon, Utah. The van drove East, imaging the walls of the canyon. Because of the steep slope of the mountains, targets are visible in range bins throughout the imaging field of view of the radar. This provides a scene with conditions similar to those on a UAV. The platform velocity was measured at 50 mph.

The results of the imaging process are shown in Fig. 4.11, along with the velocities chosen for each of the azimuth windows. A digital video also recorded the scene while traveling through the canyon. Individual slices of the movie frames are pasted together and shown in Fig. 4.12. The scene is split into three sections, with the approximate corresponding SAR azimuth location marked along the right sides.

Ranges near the van appear dark since the ground slopes down below the road for approximately the first 150 meters in range. As the ground slopes upward, it becomes visible in the image. Trees and their shadows are apparent at these ranges in the image, especially near the beginning and end of the collection. The long lines at 400 meters in range are artifacts of a power supply harmonic.

Natural terracing in the mountains of Bridal Veil Falls are seen it 1600 meters in azimuth. The small dip near 1800 is the location of the falls itself. As the van continues its course in azimuth, the mountain range moves a few hundred meters farther away in range, as seen.

While the ideal SAR track is a straight line, the road winding through the canyon is not. At curves in the road, range migration occurs in the data as targets are shifted in range. This also causes a non-uniform velocity at each range line, affecting the ability of the azimuth matched filter to compress the image. The result is a smearing of the targets in azimuth. These effects due to the curvature of the road are seen near 1300, 2300, and 3100 meters in azimuth.

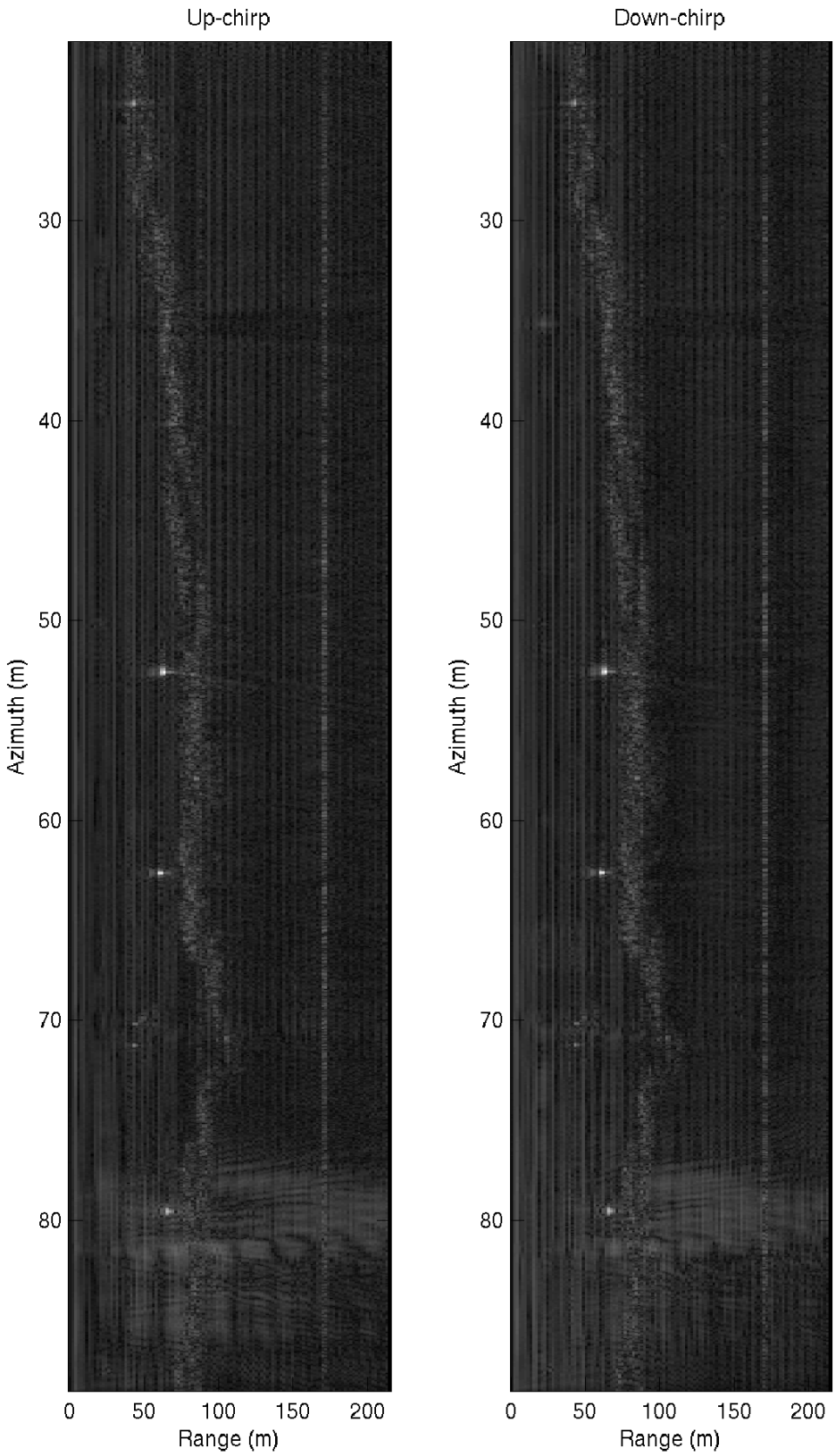


Figure 4.9: Image showing the azimuth compressed data.

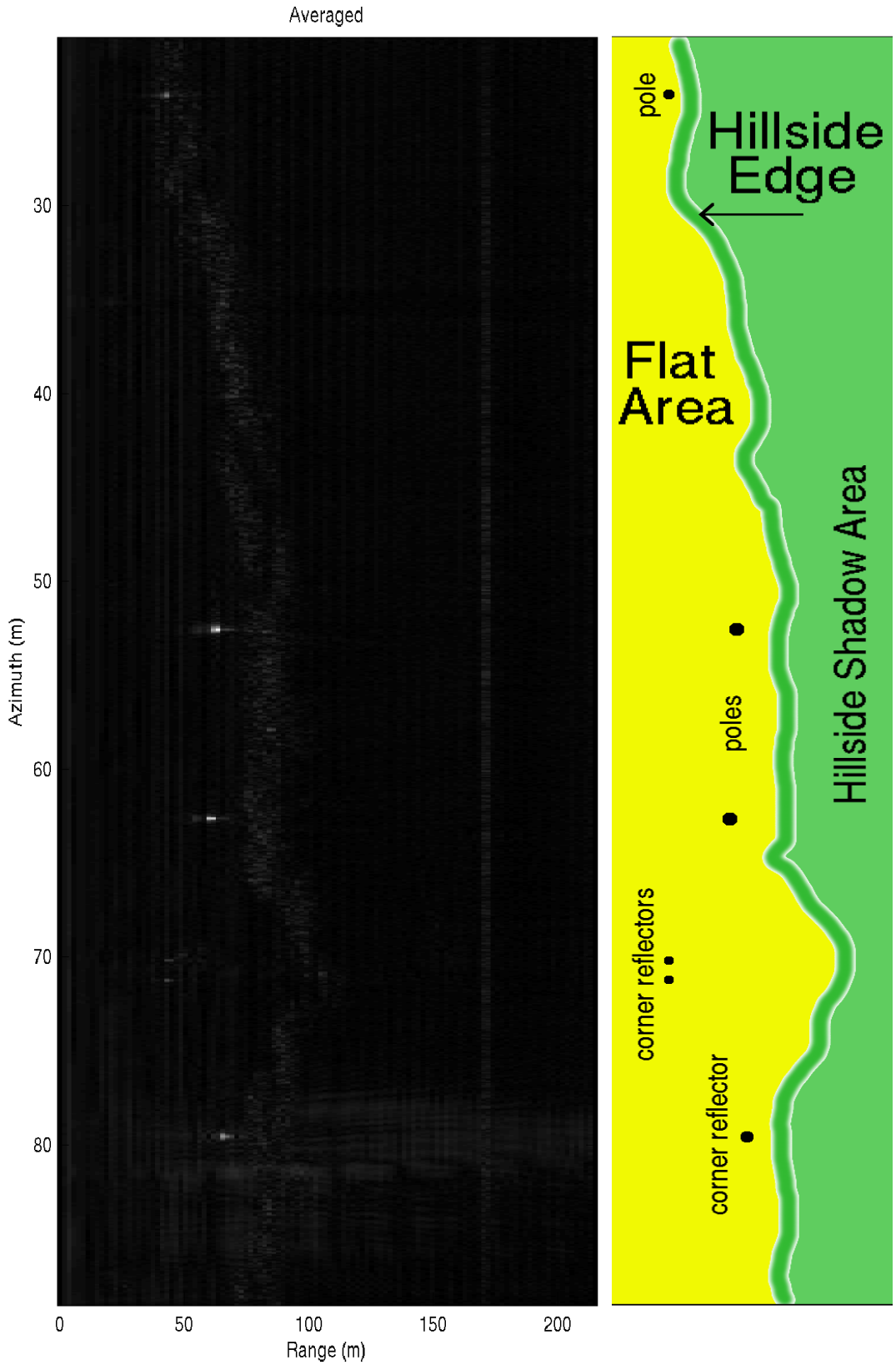


Figure 4.10: Image showing the averaged up/down chirp azimuth compressed data.

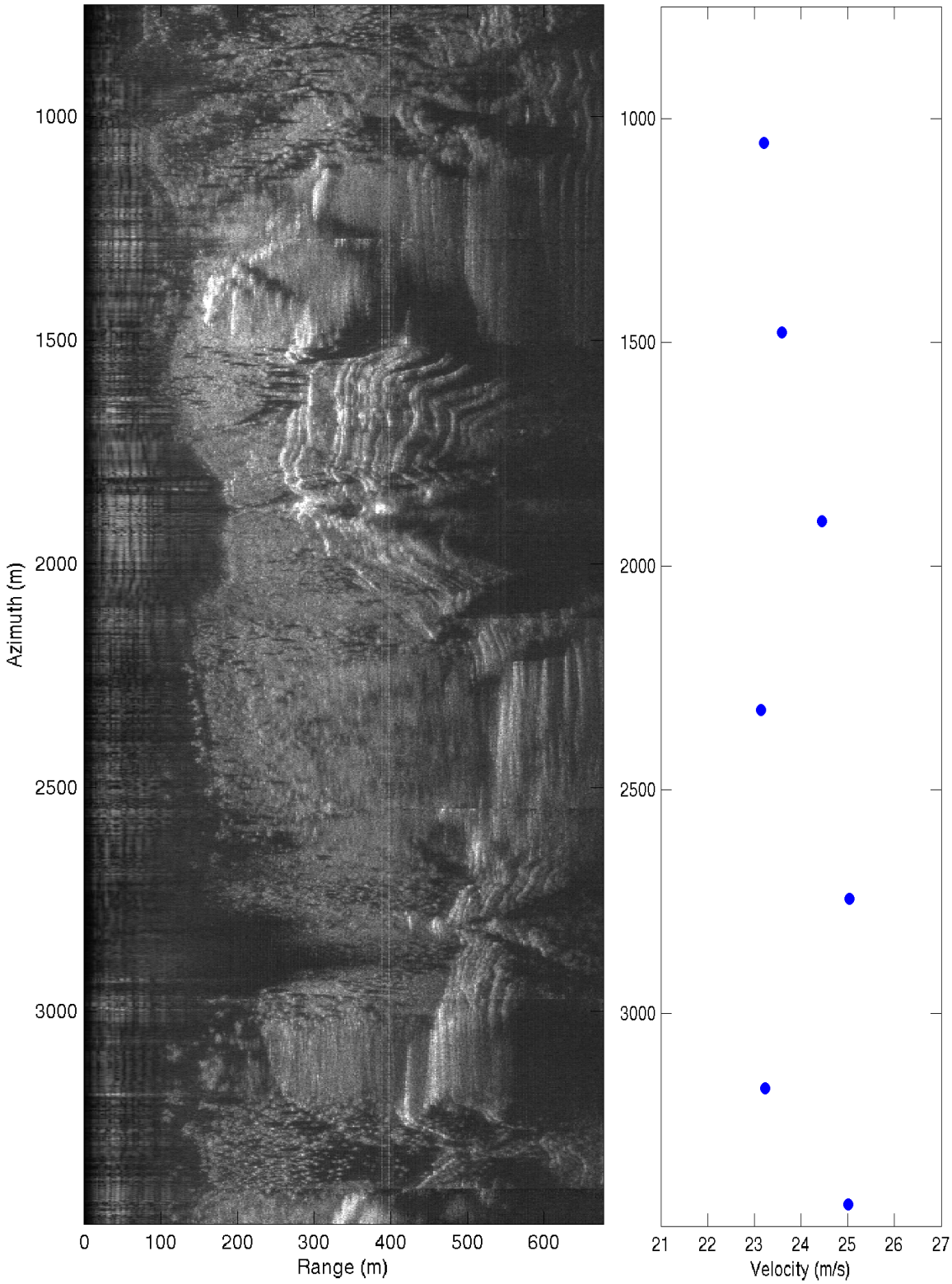


Figure 4.11: μSAR image showing Provo Canyon mountains. The velocity for each of the azimuth windows is shown along the side.

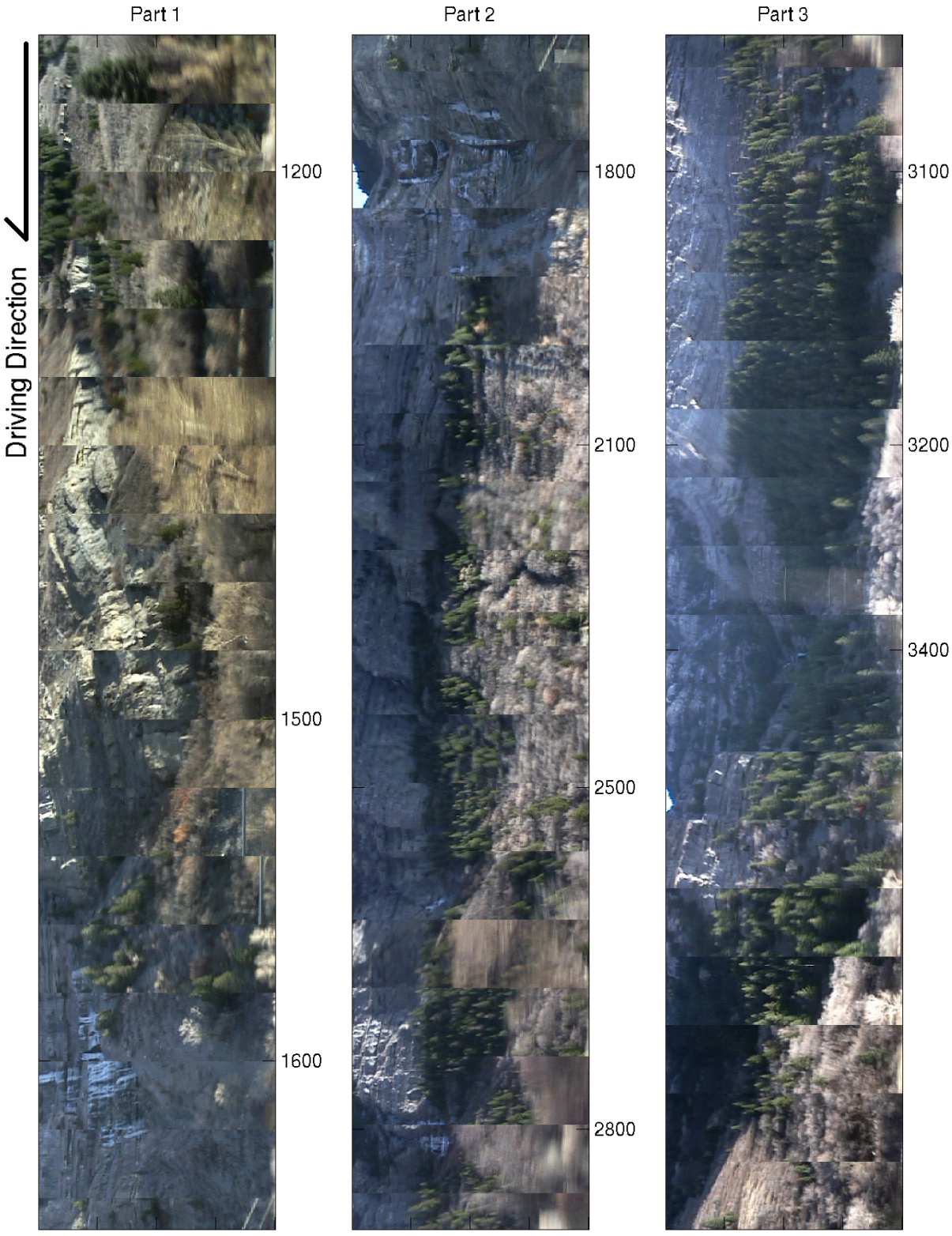


Figure 4.12: Slices of individual movie frames pieced together showing the radar scene imaged. Approximate corresponding azimuth locations are given in meters.

Chapter 5

Conclusion

The research presented in this thesis describes the processing algorithms for the linear frequency modulation-continuous wave μSAR developed at Brigham Young University. This extends the body of knowledge surrounding LFM-CW SAR and its signal processing methods.

Essential background information is presented, providing a foundation for the SAR processing techniques. An LFM-CW data simulator is derived, including results of simulated data on the actual μSAR processing algorithms. Simulated data are important to validate the processing code functionality, as well as to fine-tune the hardware.

Range and azimuth compression are detailed, including both the conventional SAR compression techniques and peculiarities specific to LFM-CW compression. Additionally, algorithms are developed to auto-locate the beginning of each transmit frequency ramp, filter interference and auto-focus the azimuth compressed image for velocity variability.

These techniques describe the fundamental aspects of LFM-CW that vary from the usual pulsed SAR model. LFM-CW is chosen for the transmit signal of μSAR to create a short-range, light-weight, low-power consumption SAR for flight on a small UAV. This research furthers the value of synthetic aperture radar as a tool in earth remote sensing, as well as demonstrates the practicality of utilizing SAR technology in low-cost, low-power applications.

5.1 Contributions

The contributions made by this research include development of:

- An LFM-CW data simulator.
- A chirp-start location algorithm for LFM-CW SAR, which eliminates the need for complicated synchronization software.
- A velocity auto-focusing for LFM-CW SAR.
- An interference excision filter.
- MATLAB and C image compression code for LFM-CW SAR. This includes code for the data simulator (MATLAB only), range compression, azimuth compression, chirp-start detection, an interference filter, and auto-focus algorithm.
- Test and demonstration collections made with μSAR.

5.2 Future Work

Areas in which further research may be performed are

- Design and implementation of motion compensation for μSAR. One main assumption generally made in SAR processing is that the radar platform moves along a straight course with constant velocity. This is not the case, however, and these variances in motion result in degradation of the compressed image. As shown in previous works [12], [9], motion compensation can dramatically improve image quality by counteracting motion effects. Because motion compensation requires motion measurements not available on a UAV, alternative motion estimation methods must be developed.
- Interferometric LFM-CW SAR. Interferometry utilizes two offset collections of the same scene to create a three-dimensional image. This is be performed using either two receive antennas separated by some baseline during collection, or making two separate passes over the same scene. The phase histories of the

data are then co-located and the difference calculated. After some computation, the height of each point in the image is revealed [1], [5].

- Improving the auto-focusing algorithm used in azimuth processing. The current algorithm requires an estimate of the platform velocity as well as bright targets in the image. A more sophisticated auto-focusing algorithm could alleviate this deficiency.

- Implement real-time compression algorithm. With the low sampling rate of received data, it is possible to create a real-time implementation of the processing software, producing real-time images while the platform moves.

Appendix A

Hardware

This appendix gives the specifications, performance, system block diagrams and pictures of μSAR. All of the following information is provided by Dr. David Long and Derek Hudson.

A.1 Specifications and Performance

The following tables list the specification and performance parameters for μSAR. The physical specifications are shown in Table A.1. Table A.2 shows the general parameters of the SAR. The antennae are described in Table A.3. Finally, Table A.4 lists the flying parameters of the platform.

Table A.1: μSAR Physical Specifications.

Parameter	*Value*
Transmit Power	28 dBm
Supply Power	30 - 40 W (18 VDC)
Approximate Mass	2.67 kg
CPU, ADC	0.44 kg
Two antennae	0.7 kg
Cables to antennae	0.69 kg/m
Rest of unit, including cables	0.84 kg
Size (without CPU, ADC)	17.5 x 12.5 x 10 cm (L x W x H)

Table A.2: General Parameters.

Parameter	*Value*
Transmit signal	LFM-CW
Frequency	5520 - 5600 MHz
PRF	138 - 2886 Hz (variable)
A/D sample rate	328.947 kHz
Maximum collection time	28 min per 1 GB

Table A.3: Antennae.

Parameter	*Value*
Configuration	Bi-static
Azimuth 3dB beamwidth	8.8 degrees
Azimuth 10dB beamwidth	15.2 degrees
Azimuth sidelobe level	-18 dB
Elevation 3dB beamwidth	50.0 degrees
Elevation 10dB beamwidth	88 degrees
Elevation sidelobe level	-17 dB
Type	2 X 8 patch array
Weighting	Dolph-Chebyshev
Feed	Coplanar waveguide
Size	34.8 X 12.4 X 0.23 cm

Table A.4: Platform Parameters.

Parameter	*Value*
Flight altitude	16 - 344 m (PRF dependent)
Velocity	18 - 385 m/s (PRF dependent)
Platform	UAV (built by Aerosonde)

A.2 Signal Block Diagram

Figures A.1 and A.2 give the system block diagrams for μSAR.

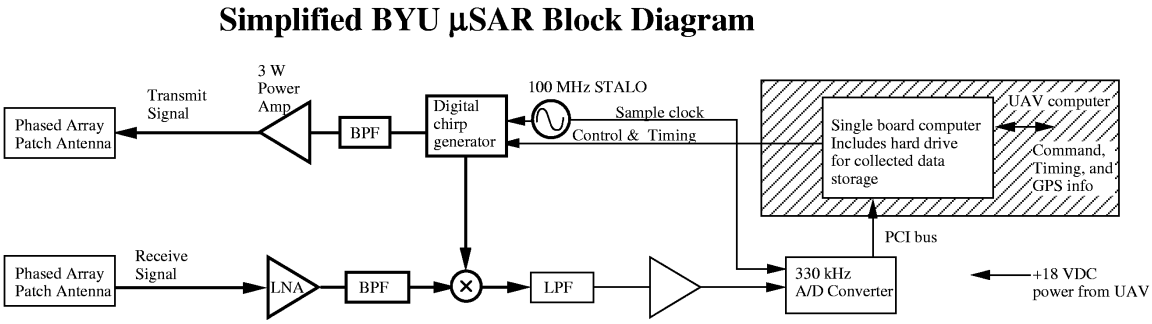


Figure A.1: Simplified signal block diagram.

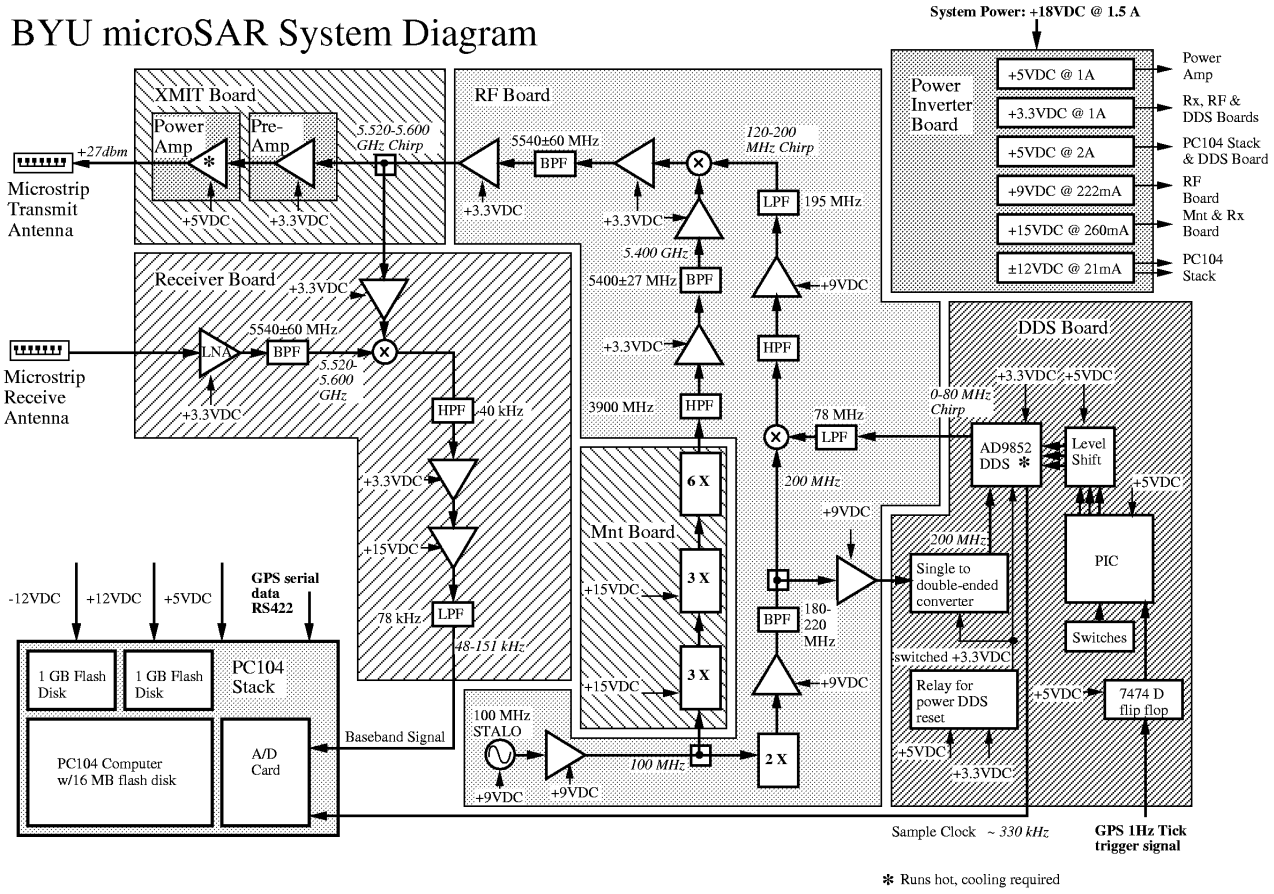


Figure A.2: BYU μSAR signal block diagram.

A.3 Pictures

This section presents images of μSAR. Figure A.3 shows μSAR with all the radar boards and cables connected together. Figure A.4 shows the μSAR computer with the A/D board connected.

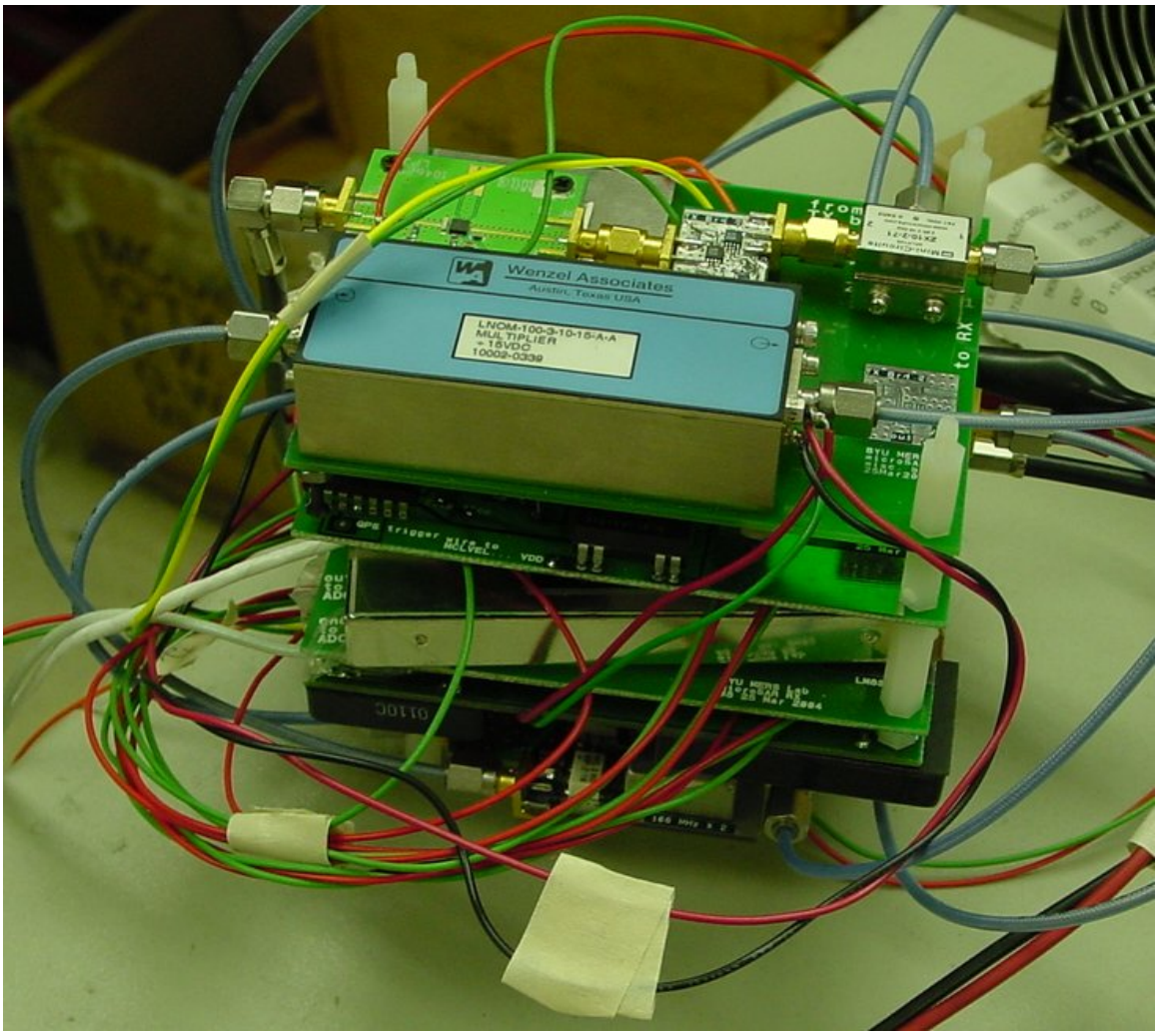


Figure A.3: Image of μSAR radar boards joined together.

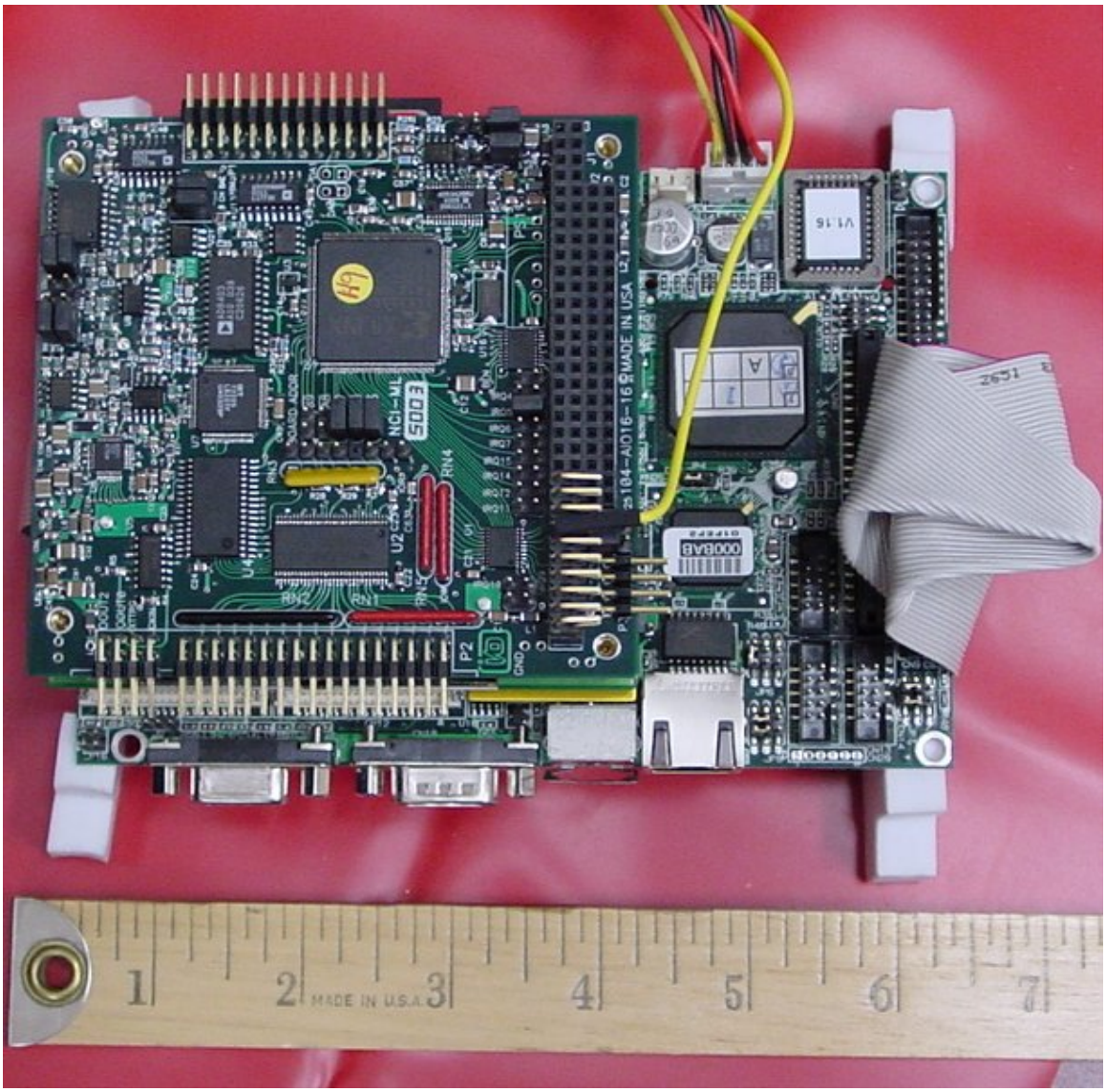


Figure A.4: Image of μSAR computer with A/D board on top.

Appendix B

Operating Software

The purpose of this appendix is to describe the operation of the μSAR processing software. Collection of data on the μSAR instrument is described in the first section, followed by a section on the actual processing of the collected data.

B.1 Data Collection

Data collection is performed on a Crusoe processor running a variant of Linux. A kernel module controlling the A/D board is loaded at boot and the driver is then accessed by reading from the file `/dev/a2d`. The `cat` command is normally sufficient for this.

Data collection begins when unit finishes booting after power-on, however a timer may be set to delay collection. The typical storage available on a 1GB flash disk is just under 30 minutes. Once collected, the data may be retrieved by extracting the flash memory card and reading it on a PC.

B.2 Image Processing

The various command-line parameters and outputs are described in detail to offer a functional understanding of the program.

The μSAR processing program has the following syntax:

```
sargenie [-inputtype] filename [options] [parameters]
```

The *inputtype* parameter is one of three options

```
-raw            Raw floating-point data file (default)
```

```
-rc             Range compressed file
-ac             Azimuth compressed file
```

This allows outside signal processing and modifications to be made in the data at any step, then continue processing from that point forward.

The *options* that may be specified are

```
-v              Verbose/debug output
-noautostart    Turn off auto-start location algorithm
-noautofocus    Turn off auto-focusing
-nobmp          Don't create output bmp of processed image
-nowisdom       Don't use saved FFT wisdom
```

The *-v* option turns on verbose output for debugging purposes. The *-noautostart* and *-noautofocus* turn off these processing algorithms explained in 4. *-nobmp* disables an output bitmap of the scene from being generated. By specifying this option, the final processed data will simply be saved in complex-data form without an actual image being created. To speed Fast Fourier Transform (FFT) algorithms, the FFTW library is used [15]. During processing, a file named `sar.wisdom` is created and updated, speeding transforms. Specifying the *-nowisdom* option disables this feature and dramatically increases processing time.

The μSAR processing *parameters* that may be specified are:

```
-vel #          Set platform velocity (30)
-prf #          Set pulse repetition freq (330)
-srate #        Set sampling rate ( )
-period #       Set number of samples per period ()
-bw #           Set the chirp bandwidth ()
-wlen #         Set az-comp window length (4096)
-offset #       Set start of chirp sample number ()
```

Each of the defaults are specified in parenthesis. The *-vel* parameter is used to set the nominal velocity of the platform and is specified in meters per second. While the

autofocusing algorithm may be performed to estimate the platform velocity, better results are obtainable by providing a velocity parameter. *-prf* sets the pulse repetition frequency, meaning the number of up/down ramps per second. The A2D sampling rate is specified with the *-srate* parameter. The chirp bandwidth in Hz is given with *-bw*. Azimuth compression is performed in chunks to speed processing. The number of range lines per window may be set with *-wlen*. Finally, for proper processing, compression must begin on the sample corresponding to the beginning of the up-chirp. The beginning offset is manually set by the *-offset* parameter.

Bibliography

[1] Douglas G. Thompson, *Innovative Radar Interferometry*, PhD thesis, Brigham Young University, 2001.

[2] Ryan L. Smith, "Micro Synthetic Aperture Radar Using FM/CW Technology", Master's thesis, Brigham Young University, 2002.

[3] Fawwaz T. Ulaby, Richard K. Moore, and Adrian K. Fung, *Microwave Remote Sensing Active and Passive*, vol. 1, Artech House, 1981.

[4] Walter G. Carrara, Ron S. Goodman, and Ron M. Majewski, *Spotlight Synthetic Aperture Radar: Signal Processing Algorithms*, Arthech House, Inc., Norwood, MA, 1995.

[5] Adam E. Robertson, "Multi-baseline Interferometric SAR for Iterative Height Estimation", Master's thesis, Brigham Young University, 1998.

[6] Mehrdad Soumekh, *Synthetic Aperture Radar Signal Processing with MATLAB Algorithms*, John Wiley & Sons, Inc., New York, NY, 1999.

[7] Giorgio Franceschetti and Riccardo Lanari, *Synthetic Aperture RADAR PROCESSING*, CRC Press, Boca Raton, FL, 1999.

[8] Charles Elachi, *Spaceborne Radar Remote Sensing: Applications and Techniques*, IEEE, New York, NY, 1988.

[9] David P. Duncan, "Motion Compensation of Interferometric Synthetic Aperture Radar", Master's thesis, Brigham Young University, 2004.

[10] David V. Arnold, "Vector Quantization of Synthetic Array Radar Data", Master's thesis, Brigham Young University, 1987.

[11] Nadav Levanon, *Radar Principles*, John Wiley & Sons, Inc., New York, NY, 1988.

[12] Richard B. Lundgreen, "Method of Motion Compensation of YINSAR Data", Master's thesis, Brigham Young University, 2001.

[13] Bassem R. Mahafza, *Radar Systems Analysis and Design using MATLAB*, Chapman & Hall/CRC, Boca Raton, FL, 2000.

[14] Steve Kapp, "802.11a. More bandwidth without the wires", *IEEE Internet Computing*, vol. 6, pp. 75–79, July-Aug. 2002.

[15] Matt Frigo and Steven G. Johnson, "FFTW: an adaptive software architecture for the FFT", *International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1381–1384, May 1998.