國　立　交　通　大　學

機械工程學系

碩士論文

利用自動聚焦重建微波影像

Microwave Images Reconstruction using Autofocus Algorithm

研 究 生： 吳偉誠

指導教授： 成維華　教授

中華民國九十四年六月

利用自動聚焦重建微波影像

Microwave Images Reconstruction using Autofocus Algorithm

研 究 生：吳偉誠　　Student：Wei-Chang Wu

指導教授：成維華　　Advisor：Dr. Wei-Hua Chieng

國 立 交 通 大 學

機 械 工 程 學 系

碩 士 論 文

A Thesis

Submitted to Institute of Mechanical Engineering

College of Engineering

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Mechanical Engineering

June 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年六月

利用自動聚焦重建微波影像

研究生：吳偉誠　　指導教授:成維華 教授

國立交通大學機械工程學系

摘　　要

微波影像在遙測地表、獲得地面上土壤、水資源等，甚至在軍事方面都是相當重要的影像資訊。由於微波影像的同調性質，需要相當精確之相位計算，稍許的相角誤差會造成影像模糊、解析度降低，而使其價值降低。這裡所提及的相位誤差皆指方位角方向的誤差，並且是因為飛機路徑偏移所造成的。自動聚焦演算法是一種改善微波影像品質的演算法，可以估測訊號的相位誤差，以達到消除相角誤差，改善影像之解析度。

自動聚焦技術主要包含兩個部分：誤差估測和誤差補償。傳統由感測器執行之運動補償功能只能測量到相位的一次項係數，如果加上自動聚焦法，則可以將二階以上的誤差項消除。本篇論文主要是探討在不同的模式下，自動聚焦演算法的成效。利用自動聚焦演算法來補助系統的運動補償功能或成為微波影像系統之一部分架構，以期改善微波影像之品質。

Microwave Images Reconstruction Using Autofocus Algorithm

Student：Wei-Cheng Wu　　Advisor：Dr. Wei-Hua Chieng

Institute of Mechanical Engineering
National Chiao Tung University

Abstract

Microwave images are important for acquiring the surface data of the earth such as the soil of the ground, water resource, and the applications in military. Because of the coherent characteristics of microwave, the phase must be calculated accurately. A little phase error will cause image degrading and useless.

Phase errors mentioned here mean errors in azimuth direction. Autofocus technique is an algorithm to improve microwave images quality. Autofocus technique can estimate phase errors and remove phase errors to improve the images resolution. Autofocus technique generally encompasses two steps：phase estimation and phase compensation. First order coefficient of the phase error function can be estimated by sensor-based motion compensation. Second or higher order term of phase error function could be estimated and cancelled by autofocus technique.

In this paper, the effect of the autofocus algorithm in different modes is presented. The autofocus algorithm is used to assist the capability of motion compensation of the system. And the autofocus algorithm is expected to improve the quality of microwave images.

Acknowledgement

First, I would like to appreciate my advisor Prof. Wei-Hua Chieng. He gives me many assists whether in research or in life. Next, I want to thank for my senior, Chia-Feng Yang. He teaches me lots of simulation and coding skills. Let me smoothly solve problems.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 History

Many algorithms were introduced to implement the autofocus technique. Among them, five practical autofocus technique are discussed in the following paragraphs. The first four techniques are mapdrift (MD), multiple aperture mapdrift (MAM), phase difference (PD), and phase gradient autofocus (PGA). We describe these techniques for only one-dimensional phase error because the azimuth phase error is of the main concern in SAR applications. The fifth technique is prominent point processing (PPP). In more precise applications, PPP provides a full motion compensation capability that includes the ability to determine the scene rotation rate for SAR and ISAR applications.

First, mapdrift is developed in the early to mid-1970s that was a significant event in the evolution of fine-resolution imaging radar[1]. MD directly estimates the quadratic coefficient of the phase error and then, based on this estimate, builds and applies a one-dimensional correction vector to the signal history. Second, the extension of MD, MAM, known as multiple aperture mapdrift autofocus, appeared in 1981 in the classified literature[1]. Third, the development of PD began in the late 1980s[2]. The PD algorithm achieves results comparable to MD with fewer computations. This algorithm is able to generate an accurate quadratic phase error measurement without first canceling the major portion of the error being measure. Fourth, the PGA algorithm for estimating higher order phase errors first appeared in the SAR literature in 1981[3-6]. The algorithm is unique in that it is not model-based；implementation does not require explicit

selection of a maximum order for the phase error being estimated. PGA algorithm has a well-deserved reputation as a superior algorithm for higher order autofocus. Finally, PPP algorithm operates much like an autofocus technique and is considered a refocus technique, although it is not technically automatic. The PPP technique is an interactive procedure making use of the radar signals returned from the target to obtain azimuth coherence for the SAR signal history.

1.2 Motive and Object

Forming a microwave image is different from forming an optical image because a microwave image does not directly reflect the signal intensity of an object such that a photo reveals its intensity in space coordinate. Microwave image must be processed by a series of transform, and reveal its signal intensity in Range or Doppler coordinate. An optical image has fine resolution, and it is easy to read. The source of light is not coherent that won't cause the speckle phenomenon. Microwave image has many advantages such as detecting objects day or night, detecting objects in every kind of weather and observing the object on ground or water.

In microwave imagery, turbulence in the unknown path movements corrupt the phase of the echo signals leading to image degraded. Phase-distortions represent a major obstacle preventing the widespread use of microwave imaging. The removal of blur-causing phase-distortions often requires data-driven phase-retrieval or autofocus techniques. These techniques are widely used in narrow-beam, narrow-bandwidth, spotlight mode Synthetic Aperture Radar (SAR) to improve the system performance. Autofocus algorithm attempt to estimation unknown motion-errors and provide the accuracy unable to be

obtained using an INS. Accurate motion estimation is important because unknown motion error cause image degraded.

What we want to do is using autofocus technique to reconstruct the microwave image and to improve image resolution. We apply Prominent Point Process (PPP) algorithm, Phase Curvature Autofocus (PCA) algorithm, and Strip-map Phase Gradient Autofocus (SPGA) algorithm on images. All of them are not as mature as PGA hence we want to research them.

1.3 Research Orientation

We have already known how PGA works, but it does not match our applications because of the different modes. Therefore, other algorithms like PPP and PCA are our goal.

We will implement our algorithms on PC-based platform. In software aspect, Matlab is used to implement algorithms.

First, we study theorems of both algorithms and derive relative formulas. Next, Matlab is used to simulate algorithms, and we modify algorithms to match our applications at the same time. Finally, algorithms are implemented with real stripmap data. We observe and compare the results.

Chapter 2 Autofocus Fundamental

Autofocus techniques improve image focus by removing a large part of any phase errors. The phase error is presented after conventional motion compensation and data formatting procedures. Autofocus with respect to microwave imagery refers to the computer-automated estimation and subsequent removal of these residual phase errors. Autofocus generally encompasses two distinct steps: error estimation and error compensation. Error estimation operates on the SAR data and estimates the coefficients of an expansion that models the existing phase error. Elementary autofocus methods may determine only the quadratic coefficient while more elaborate methods estimate higher order coefficients as well. Error compensation or cancellation uses the coefficient values that the first step provides. These steps are described in detail in this chapter. And a few different methods have been proposed for the phase estimation step, with different criteria for optimality. Some autofocus implementations require explicit knowledge of the set of error coefficients to insert the proper quantities into the correction procedure. In other implementations, the coefficients are implicit in some composite compensation phase function. For instance, measurements on a high-intensity point scatterer in the image may provide this phase function.

2.1 Motivation for autofocus

For Synthetic aperture systems, the coherence is very important. Synthetic aperture requires much information about the returned signal to get well-known locations. Uncertain trajectories that deviate over fraction of a wavelength (λ/16)

cause the synthetic aperture imagery blurring. SAR imagery suffers from this blurring, although the problem is less significant with the high pulse-repetition-frequency and short integration times of SAR system[7]. Moreover, the accuracy of a typical GPS-locked inertial navigation system (INS) in sufficient to allow close to diffraction-limited SAR imagery[8].

Autofocus algorithms must be used to estimate the phase-distortions caused by path errors. Autofocus algorithms estimate the platform trajectory and remove residual image blurring using the collected data. Another name for this type of algorithm is micronnavigation.

2.2 The autofocus problem

Synthetic aperture autofocus was originally developed in the SAR system. This kind of algorithm is to estimate the phase error and center-frequency of the returned signal. The phase error is the error that estimates the SAR platform velocity. The imagery from a system suffering a phase error has a quadratic defocus[9]. Center-frequency errors cause along-track blurring effects similar to those caused by low-frequency sway errors in stripmap system.

The main purpose for the autofocus algorithm is to estimate the platform's path-deviation at each pulse. Autofocus is often achieved in an iterative framework, using the recorded returned signal. The related problem of autofocus is usually aided by an on-board INS and is not generally iterative.

The estimated phase errors must be less than $\lambda/8$ over the aperture[10]. The motion constraints are derived using the two-way propagation path-echoes still sum coherently if the position errors are less than $\lambda/16$. When unknown motion effects cause phase error greater than $\lambda/8$, the returned signal do not sum

coherently and SAR image suffers degradation. The constraints on high-frequency errors are even limited and are less than λ/60 over the aperture[9][11]. Low-frequency phase errors across the aperture cause main-lobe broadening that cause the image degraded but high-frequency phase errors result in raising the side-lobe[11] that result in a loss of contrast in the image.

2.3 Spotlight vs. stripmap

Imaging in stripmap mode is different form imaging in spotlight mode hence their processes of motion compensation are much different. It is more difficult for stripmap than spotlight systems to focus the obtained the imagery. This is because blurring of stripmap images being space-variant[12], but spotlight images have space-invariant blurring. Space-variant blurring[7][13] means that standard, well researched, spotlight autofocus methods such as mapdrift[14], and PGA[15-17] are unable to be applied to stripmap data without modification. In the case of PGA, an extension to the stripmap case does not exist[13] and is referred to as phase curvature autofocus (PCA) [18][19]. The assumption of space invariance in the spotlight algorithms often makes the extension to stripmap systems challenging. The reason why space-invariant algorithms perform poorly in space-variant problems is straightforward. A space-invariant autofocus algorithm ensemble averages over all scatterers to estimate the path. However, in a space-invariant problem, all scatters have the same blurring. In a space-variant problem, each scatterer has a different blurring. Averaging many different path estimates results in a poor overall path estimate.

Data from stripmap systems is often autofocused by segmenting the image into smaller along-track sections[19-21] in order to overcome the problem caused

by space-variant blurring. Usually a preprocessing step is required[14][20][22] to massage the data into a form that SAR autofocusing algorithms can use. Each section of the data is autofocused independently and the individual estimates combined to remove the distortion form the entire image[21] [23]. Approaches that account for the spatial variance and retain information (such as stripmap phase gradient autofocus (SPGA)) have better performance. To summarize, spotlight autofocus is a special case of stripmap autofocus where the space-variance of the problem is low. Spotlight algorithms often ignore the effect of space variant blurring.

2.4 Autofocus technique

Phase errors are presented in a SAR signal history at the time of the final Fourier transform operation. The most serious degradation is often image defocus caused by the presence of uncorrected quadratic and higher order phase errors. Uncompensated motion between the SAR antenna phase center (APC) and the scene being imaged is a primary source of phase errors. Other sources include algorithm approximations, hardware limitations, and propagation effects.

Autofocus algorithms often need to estimate the phase error across the aperture $\phi(k)$ -- effectively determining the time shift in the image. Much different phase estimation has been used for this purpose. Some of them more widely used are outlined in this section.

The performance of autofocus algorithms relies on the phase estimation technique. The PGA algorithm has been proven to be a robust technique that can provide excellent results over a wide variety of both scene content and structure

of the degrading phase-error function. It is unique in that it is not model-based. The PGA algorithm has a well-deserved reputation as a superior algorithm for higher order autofocus.

2.5 Phase Gradient Autofocus (PGA)

Fig 2.1 shows the actual operation steps in simulation of the PGA algorithm[24]. The process begins with the range compressed image and selects a subset of range bins containing the greatest energy. It performs an azimuth Fourier transform to compress each selected rang bin, locates the peak amplitude within each azimuth-compressed range bin, and selects a window of complex pixels from each range bin centered about the azimuth location of the peak amplitude sample. The procedure continues through a number of steps using these groups of pixels from each selected range bin to compute the first derivative of the signal history associated with the windowed image pixels. These last steps must use of the derivative property of the Fourier transform[25].

The final step of the PGA algorithm integrates the derivative of the azimuth phase error to calculate the phase error itself as a function of azimuth sample number over the aperture. It is a simple task to convert this estimate of the higher order phase error into an error correction vector. A typical procedure applies the correction vector to the signal history and iterates the algorithm several times on the corrected data in order to reduce residual error.

2.5.1 Phase Estimation

We consider a single range bin containing one point scatterer. The time-domain signal from the scatterer is

$$g(t) = a\exp\{j[w_0 t + \phi_0 + \phi_e(t)]\} \qquad -\frac{T}{2} \le t \le \frac{T}{2} \tag{2-1}$$

Here, a_t, w_o and ϕ_0 represent the magnitude, frequency, and phase of the signal-history, respectively；$\phi_e(t)$ represent the phase error of the signal-history. The PGA algorithm first takes a Fourier transform of the signal in azimuth. We can obtain

$$G(w) = aT\sin c[T(w - w_0)]e^{j\phi_0} \otimes \Phi_e(w) \tag{2-2}$$

where $\Phi_e(w)$ is the Fourier transform of the phase error term $\exp[j\phi_e(t)]$；the symbol "$\otimes$" denotes convolution. Next, selecting and symmetrically windowing the peak amplitude point within this range bin and using the weighting function W(w) to produce

$$G_w(w) = \mathrm{W}(w - w_0)aT\sin c[T(w - w_0)]e^{j\phi_0} \otimes \Phi_e(w) \tag{2-3}$$

The window can only include the highest frequency phase error that the PGA can estimate and exclude the interference of other points. Taking the inverse Fourier transform of the windowed signal with respect w-w_0, then

$$g_w(t) = ae^{j\phi_0}e^{j\phi_e(t)} \otimes \mathrm{w}(t) \tag{2-4}$$

where w(t) is the inverse Fourier transform of W(w). The windowed signal function in time-domain, $g_w(t)$ represents the signal history associated with peak point response in the corresponding image domain. If the window length is long enough to include all important frequency component of $\Phi_e(w)$, then there is a little effect on $g_w(t)$ and we can ignore the effect of w(t) in (2-4).

The PGA algorithm utilize the inverse Fourier transform of $jwG_w(w)$ to obtain the first derivation of $g_w(t)$. For a single target, this derivative $\dot{g}_w(t)$ is equal to

$$\dot{g}_w(t) = j\dot{\phi}_e(t)g_w(t) \tag{2-5}$$

where $\dot{\phi}_e(t)$ is the derivative of the phase error. This relationship provides a means to estimate $\dot{\phi}_e(t)$ using the computed values of $\dot{g}_w(t)$.

Assuming the effect of w(t) can be neglected, the estimation of the derivative of the phase error becomes

$$\hat{\dot{\phi}}_e(t) = \frac{\mathrm{Im}[\dot{g}_w(t)g_w^*(t)]}{|g_w(t)|^2} \tag{2-6}$$

Integration of (2-6) provides the phase error estimate to within a constant. For a scene containing many rang bins, the PGA averages both the numerator and the denominator of (2-6) over many range bins to estimate the gradient. Using N range bins, the estimation becomes

$$\hat{\dot{\phi}}_e(t) = \frac{\sum_N \mathrm{Im}[\dot{g}_w(n,t)g_w^*(n,t)]}{\sum_N |g_w(n,t)|^2} \tag{2-7}$$

where $g_w(n,t)$ is the signal associated with the windowed target from the nth range bin and $\dot{g}_w(n,t)$ is its derivative. The expression in (2-7) represents weighted least-squares estimate of the phase error derivative[5].

2.5.2 Implementation

There are four crucial steps in PGA algorithm referred to as circular shifting, windowing, phase gradient estimation, and iterative correction. It appears that the circular shifting operation not only attempts to align strong scatterers, subsequently improving the signal-to-noise ratio for phase estimation, but it also aligns regions undergoing subtle contrast changes. Windowing has the effect of preserving the width of the dominant blur for each range bin while discarding

data that cannot contribute to the phase-error estimation. Phase estimation is to correct the phase error. Last step, the iteration is to estimate and to correct the phase repeatedly.

It is important to select a window in order to offer significant data in the PGA algorithm. The first step is to select a window length that is much larger than the anticipated phase errors require and reduces the window length on subsequent iterations. The initial window length can include the entire azimuth extent of the image. Typically, each PGA iteration reduces the previous window length between 20 and 50 percent to minimum value. The minimum window value dictates the maximum order phase error that the procedure can estimate. With this strategy, the PGA algorithm can have excellent performance in estimating phase errors of arbitrarily high order on a wide variety of scenes[6].

Figure 2.2 show the algorithmic steps and corresponding images in PGA algorithm[26]. The following simulation is the result of the graduate thesis which written by the senior in my lab. I just extract some results of his thesis.

Following the steps of Figure 2.2, we input an image with phase errors shown in first step. Second, we circular shift the input image. Next, the shifted image is windowed with 20dB window. The neighbor image of the windowed images is the accumulation of image intensity in dB. The corrected image after PGA algorithm obviously many phase errors are removed. And we iterate the loop again to see how effect it will be. Figure 2.3 shows the iteration result. The window width is much smaller. That means the information is concentrated, and a lot of phase errors are removed. After iterations, the image does not improve as the first time. We can compare these images in figure 2.4, and only little difference is between the first corrected image and the iterated image. Figure 2.5 show the estimated phase error. The blue solid curve is the error function that we

apply to the original image. The error function is to simulate the path-deviation, the platform vibration, and some low-frequency phase errors. The red dotted curve is the first estimated phase error that is close to the simulation error function. The green dot curve is during two iterations and it almost the same as the red one.

The above result is with the high-SNR image hence the phase error is easier to find out. Iteration times can be less in the high-SNR image. If the image is low-SNR, we must iterate more times to get higher resolution image.

Chapter 3 Prominent Point Processing Algorithm

Formation of fine resolution, well-focused SAR images require accurate knowledge of and compensation for relative motion between APC and the target. Especially in spotlight mode there exists rotational motion, images will be degraded greatly when the rotational motion is not uniform. Traditional autofocus methods such as MAM, PD, and PGA only can compensate space-invariant phase error but can not remove space-variant phase error. Space-invariant phase error is common in spotlight SAR.

The PPP technique is an interactive procedure making use of the radar signals returned from the target to obtain azimuth coherence for the SAR signal history. The procedure isolates the signals backscattered from a number of prominent points in the target field. The signal from the first prominent point provides a reference signal to compensate for translational motion effects. Signals from two additional prominent points provide an estimate of rotational motion to improve image quality by aiding data formatting.

It is natural to develop and use the prominent approach in the framework of the polar format algorithm because the prominent point technique to estimate the image azimuth scale involves the data formatting process. Single and multiple prominent point technique are also applicable to spotlight processing with other formation algorithms.

Single prominent point refocus moving targets that are smeared in conventionally processed SAR imagery because of unknown target translations[27]. More broadly, in an airborne SAR data collection, PPP can be useful to augment or replace instrumentation of the SAR antenna to measure its

motion. A multiple prominent point implementation is especially valuable as a means to accomplish both translational and rotational motion compensation.

3.1 Single Prominent Point Processing Algorithm

The single PPP algorithm identifies a single dominant point scatterer on the target and measures its pulse-to-pulse range position and phase variation. Using these measurements, the algorithm achieves scene center stabilization (one step in motion compensation) by removing the undesired phase effects of translational motion from the collected signal history. In fact, the algorithm estimates and compensates simultaneously for low- and high-frequency azimuth phase errors from all sources, including undesirable relative motion. There is no need for supplemental autofocus in azimuth unless it is necessary t o compensate for smaller, space-variant effects. The steps of this refocus algorithm are[24]:

1. Identify and isolate a prominent point:
2. Compress(in range) the signal history containing the target
3. Track the identified prominent point over the aperture time to measure the phase and the range location of its peak response
4. Use the measurements of the peak location to adjust the frequency of each pulse in the original signal history
5. Use the measurements of the peak phase to adjust the phase of each pulse in the original signal history
6. Process the modified signal history with a two-dimensional FFT.

Selection of the prominent point is an interactive process requiring user inputs.

In the two-dimensional image domain, a poorly focused target which the user of the algorithm must be isolate. In the range-compressed signal domain, a

scatterer appears as a line meandering across the signal history in the azimuth direction. The Figure 3.1(a) shows the two prominent point tracks. The algorithm estimates the phase and frequency of the selected scatterer by measuring the phase and the location of its peak response in each range-compressed pulse. It uses these measurements to adjust the starting phase and the frequency of the SAR signal returned from each transmitted pulse in the original signal history. After these adjustments, the return from the selected prominent point (now the scene center) has zero frequency and constant phase over the entire two-dimensional processing aperture. Range compression of this compensated signal history now generates the signal track of this prominent point scatterer at a constant range shown in Figure 3.1(b). The track of a second target point still meanders somewhat in range during the azimuth aperture interval. An azimuth FFT of this range-compressed signal history yields a focused image of the prominent point. The second scatterer still is not well-focused. The Figure 3.1(c) shows the result with some exaggeration.

3.1.1 Analysis

The radar transmits pulses using a LFM chirp signal with chirp rate γ, period T, and center frequency f_c. The transmission pulse signal

$$s_t(t) = A(t)\cos[2\pi f_c t + \pi\gamma(t - nT)^2] \qquad (3\text{-}1)$$

n is pulse number, and t is fast time

It is convenient to select a rectangular pulse shape of amplitude A_0 for A(t) and to use the complex form to represent this real signal. Defining $\hat{t} = t - nT$, the transmitted signal ：

$$s_x(t) = A_0 rect(\frac{\hat{t}}{T_p}) \exp\{j[2\pi f_c t + \pi\gamma \hat{t}^2]\} \tag{3-2}$$

T_p is the transmitted pulse length. The rect() function ：

$$rect(u) = \begin{cases} 1 & |u| \le \frac{1}{2} \\ 0 & |u| > \frac{1}{2} \end{cases} \tag{3-3}$$

The expression for transmitted signal in (3-2) does not incorporate the effects of transmits amplifier and waveguide on the signal. It does not include the time delay for the signal to reach the antenna. These effects affect the understanding and image formation techniques.

Assuming an ideal scatterer of radar at (X_t, Y_t, Z_t), its amplitude and phase characteristics do not change with frequency and aspect angle. This assumption is realistic for normal situation. The received signal from a single point at (X_t, Y_t, Z_t) at pulse n is

$$s_r(t) = \mathrm{a} \bullet rect(\frac{\hat{t} - t_d}{T_p}) \exp\{j[2\pi f_c (t - t_d) + \pi\gamma (\hat{t} - t_d)^2]\} \tag{3-4}$$

where “a” is proportional to A_0 and t_d is the total time delay from the antenna to the target and back to the antenna.

The derivation of (3-4) ignores any attenuation due to the two-way propagation of transmitted energy to the scene and back to the radar. In fact, the amplitude of the received signal will be a function of many factors including the amplitude of the transmitted signal, distance to the target, and gain of the antenna, in addition to the range cross-section of the target.

It is common practice to make the simplifying assumption that the sensor is stationary during pulse transmission and reception while moving in discrete

increments between pulses. In this situation, an adequate approximation for t_d is

$$t_d \approx \frac{2R_t}{c} \tag{3-5}$$

R_t is the distance from the Antenna Phase Center (APC) to the target. Using the approximation, (3-4) becomes

$$s_r(t) = \mathrm{a} \bullet rect(\frac{\hat{t} - 2R_t / c}{T_p}) \exp\{j[2\pi f_c(t - \frac{2R_t}{c}) + \pi\gamma(\hat{t} - \frac{2R_t}{c})^2]\} \tag{3-6}$$

The distance $R_t(n)$ express

$$R_t = \sqrt{(X_a + X_e - X_t)^2 + (Y_a + Y_e - Y_t)^2 + (Z_a + Z_e - Z_t)^2} \tag{3-7}$$

The slant range vector $\mathbf{R_a}$ = (X_a, Y_a, Z_a) represent the measured distant. The error vector $\mathbf{R_e}$ = (X_e, Y_e, Z_e) represents any error in motion measurement. Figure 3.2 illustrates the geometry.

The last analog signal conditioning step before sampling the received signal is demodulation. A spotlight mode system, occasionally, and stripmap or scan mode systems, often, will not perform this dechirp. We choose to include the dechirp operation in the model development here. The radar demodulates the received signal by mixing it with a replica of the transmitted signal delay by $2R_0/c$.

Stripmap and spotlight modes differ in selection of a motion compensation signal. Figure 3.3 shows the difference. Spotlight compensates each pulse exactly for the scatterer at scene center to convert this scatterer's signal to a constant phase. In Figure 3.3, the spotlight mode compensates for the distance AC and treats the data as if it came from B in subsequent processing.

The radar incorporates a real-time motion compensation operation by mixing the received signal $s_r(n,t)$ of (3-6) with a reference function $s_{ref}(n,t)$ to produce the intermediate frequency(IF) signal $s_{if}(n,t)$. The reference signal is

$$s_{ref}(t) = \exp\{j[2\pi f_c(t - \frac{2R_0}{c}) + \pi\gamma(\hat{t} - \frac{2R_0}{c})^2]\} \quad (3\text{-}8)$$

$R_0(n)$ is reference range from the planned APC location to the scene center. The model does not include the effects of any receiver waveguide dispersion on the signal nor any receiver filter and waveguide time delay. The IF signal that results from mixing (3-6) with (3-8) is

$$s_{if}(t) = a \bullet rect(\frac{\hat{t} - 2R_t / c}{T_p}) \exp[-j\frac{4\pi(f_c + \gamma\hat{t})}{c}(R_t - R_0)] \exp[j\frac{4\pi\gamma}{c^2}(R_t^2 - R_0^2)] \quad (3\text{-}9)$$

It is convenient to write this equation in the form

$$s_{if}(t) = a \bullet rect(\frac{\hat{t} - 2R_1 / c}{T_p}) e^{j\Phi(n,\hat{t})} \quad (3\text{-}10)$$

and

$$\Phi(n,\hat{t}) = -\frac{4\pi\gamma}{c}(\frac{f_c}{\gamma} + \hat{t} - \frac{2R_0}{c})(R_1 - R_0) + \frac{4\pi\gamma}{c^2}(R_1 - R_0)^2 \quad (3\text{-}11)$$

where $R_1(n)$ replace $R_t(n)$.

The next step in the algorithm is to Fourier transform $s_{if}(t)$ for each pulse in fast time over the parameter $\hat{t} - 2R_0 / c$. The range-compressed signal is

$$S_{if}(n, f_{\hat{t}}) = aT_p \sin c[T_p(f_{\hat{t}} + \frac{2\gamma}{c}R_{10})] \exp[-j(\frac{4\pi f_c}{c}R_{10} + \frac{4\pi\gamma}{c^2}R_{10}^2 + \frac{4\pi f_{\hat{t}}}{c}R_{10})] \quad (3\text{-}12)$$

using $R_{10} = R_1 - R_0$ for convenience. The peak intensity of the signal $S_{if}(n, f_{\hat{t}})$ occurs at $-2\gamma R_{10} / c$. A cross-correlation of the range-compressed returns from successive pulses determines the pulse-to-pulse changes in peak location. At this point, it is acceptable to either subtract off a range frequency term in the original

signal history or move the compressed pulses in range using an interpolation routine to align successive pulses. The necessary frequency adjustment to (3-10) is to multiply by a complex exponential having a phase Φ_{adj} given by

$$\Phi_{adj} = \frac{4\pi\gamma}{c}(R_1 - R_0)(\hat{t} - \frac{2R_0}{c}) \tag{3-13}$$

The next step is to measure the phase of the peak response. The phase is

$$\Phi_{pk}(n) = -\frac{4\pi f_c}{c}(R_1 - R_0) + \frac{4\pi\gamma}{c^2}(R_1 - R_0)^2 \tag{3-14}$$

The result of multiplying the frequency adjustment described Φ_{adj} and a phase that is the negative of Φ_{pk} is

$$s_{ifm}(n,\hat{t}) = arect(\frac{\hat{t} - 2R_t / c}{T_p})\exp[j\frac{4\pi\gamma}{c}(\frac{f_c}{\gamma} + \hat{t} - \frac{2R_1}{c})(R_t - R_1)]\exp[j\frac{4\pi\gamma}{c^2}(R_t - R_1)^2] \tag{3-15}$$

for the general scatterer at location (X_t, Y_t, Z_t) and range R_t. The process stabilizes a scene center scatter (for which $R_t = R_1$) gives any signal from this scatterer a constant phase and zero frequency.

The final step is to compress this signal in azimuth and in range dimension via the two-dimensional Fourier transform. The result image is in Figure 3.1(c).

If the target's relative rotation rate is essentially constant over the coherent data processing interval, this single point compensation alone provides an approximately focused image, especially when the target is small enough that polar formatting is not required. However, when the incremental change in rotation angle is not constant from pulse to pulse, a position-dependent blurring or defocusing of scatterers generally occurs.

The azimuth scale, relating image and target distances in the azimuth direction, depends on this rotation rate； therefore, the azimuth scale accuracy depends on auxiliary data accuracy. However, the range scale, relating image and target distances in the range direction, is independent of the relative motion. Thus, the range scale accuracy in PPP is equivalent to that in a motion measurement implementation

3.2 Implementation

Figure 3.4 is the flowchart of PPP algorithm. Unlike PGA, PPP algorithm does not need iteration step. Figure 3.5 is a spotlight image with the random phase error that simulates aircraft sway. Next, Figure 3.6 is the corrected image removed the linear phase. It is obviously that the quality of Figure 3.6 is much better than the Figure3.5. Because of PPP algorithm can't remove linear phase term, the linear phase removal method use the regressive method[26]. Figure 3.7 is the image without adding random phase error. We can compare Figure 3.6 with Figure 3.7. The brightness of corrected image is smaller than the original. We see that the point spread in range is more convergent, in additional the most important is the spread in azimuth is also converge. In Figure 3.8, the red dotted line is the random phase error we added, and the blue solid line is the estimated phase. We see the two lines are close, and their trends are similar. From the two viewpoints of the corrected image and the estimated phase, PPP algorithm works well in spotlight image.

Next, we discuss that the larger random phase error influences PPP phase estimation accuracy or not. There, we consider that the added phase error is 10 times larger than above. Figure 3.9 is the information with the larger error. First,

we see the top right image; it is similar with the original image, the top left one. Following, the bottom block show the added random phase error and the estimated phase where we also see the trends are similar. Although the curve is not so match, the final destination --- image correction is good. The linear phase estimated by regressive method seems not good. But we could say it works successfully with larger phase error.

PPP can also apply to stripmap data. The following implementations are all for stripmap data. First, we see Figure 3.10 that is real image with random error, and the image is distortion. Figure 3.11 is corrected image via PPP algorithm. Obviously, it looks much better than Figure 3.10. Comparing it with the original image without added error, Figure 3.12, the corrected image can be restored as similar as original one. Figure 3.13 is the plot of the random phase error and the estimated phase. The simulation result is same as spotlight.

Next, we implement the algorithm with the actual data without given phase error. The implementation is what we really care about. Figure 3.14 is the implementation result. We observe that the corrected image is not better than original image. Many reasons may cause the result. We think the most probable reason is that the motion compensation is so good that there is not much aircraft sway information for PPP to correct. And some little error may be caused by the hardware like electric circuits, mechanism vibration, thermal noise, and so on.

Because of the failure of above implementation, we decide to use image without conventional motion compensation (mocomp) to understand what reasons cause the failure of PPP with stripmap data. Figure 3.15 is a no mocomp image, and it is obviously worse than the image using mocomp, Figure 3.7. Here, we see how important mocomp is. Figure 3.16 is the no mocomp image corrected via PPP. We carefully observe the two images, and we don’t get a

much better or worse image. In some regions of the image, the scatters are much convergent, but in other places are not. We can't judge it is better or worse. And it also has the similar situation with other stripmap images.

There are many simulations, and most of them show satisfying result except the last two simulation. PPP does not design for stripmap SAR but for ISAR. That's why PPP doesn't have such good result for stripmap images.

Chapter 4 Stripmap Autofocus Algorithm

Stripmap autofocus is different from spotlight autofocus because of synthesizing image also in different way. Our data is stripmap mode hence we have to research it. Most stripmap autofocus algorithms derive from PGA --- spotlight autofocus algorithm. The following two kinds of stripmap autofocus algorithms are presented : PCA and SPGA.

4.1 Phase Curvature Algorithm

Phase curvature autofocus (PCA) [28] is an extension of phase gradient autofocus (PGA) [29] for stripmap geometries. The idea was proposed by Wahl to extend PGA to narrow-band stripmap SAR systems. PCA as originally published is suitable for narrow-band systems with no range curvature.

In narrow-band systems that the sway error can be treated as a phase only function so PCA is prepared to use this kind of system. This assumes that the image blurring is contained in 1-D. PCA uses phase curvature range redundancy of the corrupted, azimuth-chirped image to estimate sway. It averages random phase curvature components form the scene to get the common phase curvature error caused by sway. The common phase curvature error is the double integrated to estimate the sway.

4.1.1 Problems and solution for stripmap mode

A conventional development of stripmap SAR views each point target in the image as having arisen from arisen from a certain linear FM chirp response in the range-compression step in forming an image is accomplished by convolution

with a corresponding chirp sequence. This suggests that in order to obtain aperture domain data suitable for a phase estimation procedure similar to PGA, each point target response in the image domain should be isolated, windowed, and convolved with the proper FM chirp. This effectively reconstructs the original range-compressed data from which the target image was produced (along with noise arising from the clutter included in the window). Multiplication by the known conjugate chirp values at each position then yields an aperture sequence consisting of a target-dependent complex constant, multiplied by the complex exponential associated with the phase error, spanning the particular set of pulses involved in imaging that target. A collection of selected targets produces a set of displaced apertures in the range compressed domain, each of which reveals a particular segment of the phase error function, as depicted in the Figure 4.1(a)、(b) for spotlight and stripmap mode each, and dotted line means the linear phase.

In practice, the aperture domain data used for phase error estimation can be computed much more efficiently than by actually performing the chirp convolution procedure described above. This is a consequence of a theorem stating that the chirp deconvolution involved in stripmap processing is equivalent to a conjugate chirp multiplication followed by FT centered on the target. (A sequel to this paper will detail this development.) as a result, the stripmap autofocus procedure may be implemented with steps very similar to the spotlight mode PGA algorithm. The main difference is that compressed domain corresponding to the locations of the targets in the image domain.

4.1.2 Analysis

There is a difference in the way in which the phase data in the range compressed domain must be processed in the stripmap algorithm, as opposed to the spotlight case. The reason is that the target aperture signals formed as described above fail to reconstruct any linear component of phase error that may be present in the corresponding segment of SAR data. This is of no consequence in the spotlight mode case where the individual target apertures coincide, but it creates a problem in the stripmap case in which the apertures are offset from each other. The result is that partially overlapping aperture that span different segments of the phase error function, with potentially different local linear components, will not necessarily exhibit the same phase gradients in the region of overlap. The curvature of the phase function will, however, coincide for all target apertures that overlap a certain point in the collection and maybe used as the basis for coherent averaging. Thus the stripmap autofocus algorithm computes second differences in phase, averages across the appropriate target apertures, and performs a double integration to produce an estimate of the phase error function. Iteration is again used to improve the estimate.

The phase estimation is similar with PGA that mentions above in 2.5.1. There, we derive the mathematical model from (2-5), and all assumption is the same as PGA in 2.5.1. The derivative of (2-5) is

$$\ddot{g}_w(t) = j\ddot{\phi}_e(t)g_w(t) - \dot{\phi}_e^2(t)g_w(t) \tag{4-1}$$

Also, assuming the effect of w(t) can be neglected, the estimation of the second derivative of the phase error becomes

$$\hat{\ddot{\phi}}_e(t) = \frac{\mathrm{Im}[\ddot{g}_w(t)g_w^*(t)]}{|g_w(t)|^2} \tag{4-2}$$

Double integration of (4-2) provides the phase error estimate to within a constant.

For a scene containing many rang bins, the PCA averages both the numerator and the denominator of (4-2) over many range bins to estimate the gradient. Using N range bins, the estimation becomes

$$\hat{\ddot{\phi}}_e(t) = \frac{\sum_N \mathrm{Im}[\ddot{g}_w(n,t) g_w^*(n,t)]}{\sum_N |\, g_w(n,t)\,|^2} \tag{4-3}$$

where $g_w(n,t)$ is the signal associated with the windowed target from the nth range bin and $\ddot{g}_w(n,t)$ is its second derivative. The expression in (4-3) represents weighted least-squares estimate of the phase error derivative[5]. And the derivation above is in continuous time domain, it is not suitable for actually implement. The following discrete time domain is derived for coding. g(m) is used to substitute g(t), and m denotes discrete index. For simplifying the derivation, signal phase term only retain phase error function $\phi_e(m)$.

$$g(m) = a\exp[j\phi_e(m)] \tag{4-4}$$

The first derivative of phase error function in differential function $\Delta\phi_e$ is defined :

$$\Delta\phi_e = \phi_e(m+1) - \phi_e(m) \tag{4-5}$$

Therefore, the second derivative of phase error function is

$$\begin{aligned} \Delta\phi_e(m+1) - \Delta\phi_e(m) &= \phi_e(m+2) - \phi_e(m+1) - \phi_e(m+1) + \phi_e(m) \\ &= \phi_e(m+2) - 2\phi_e(m+1) + \phi_e(m) \end{aligned} \tag{4-6}$$

Finally, we can get the estimation of second derivative of the phase error function, $\widehat{\Delta^2\phi_e(m)}$,

$$\widehat{\Delta^2\phi_e}(m) = phase\{g(m+2)[g^*(m+1)]^2 g(m)\} \tag{4-7}$$

4.1.3 Implementation

The process of PCA is almost the same as PGA. We see Figure 2.2, and only step 5 is different. In PGA, phase gradient is used to estimate the phase error in step 5, but the phase curvature is used in PCA.

The first simulation we use the same image as PPP simulation, Figure 3.7. PCA has been compared with PGA because their theory is similar. Figure 4.2 is the total information about the simulation for iteration 0. We see that PCA converges faster than PGA. The estimated phase is close to the added phase error. Figure 4.3 is a bigger image that can observe more details and the image is via two iterations of PCA. Figure 4.4 is via two iterations of PGA. Both results are similar. Note that PCA can't remove the linear phase and we use regressive method to cancel the linear phase, too. Next, we see Figure 4.5, the estimated phase. The blue thin line is the estimated phase of PCA, and the green thick one is the estimated phase of PGA. We can't judge which one is better, but PCA is convergent faster than PGA in many simulations.

Figure 4.6 is a part of the real stripmap image, and we add the random phase error to see the error could be estimated or not. Figure 4.6 is the total information of iteration 0 of PCA, and there are the original image, the degraded image, the corrected image (iteration 0) and the estimated phase. We find that the performance is not so good at first correction. The figure 4.7 is the more iteration times image. The image is obviously convergent at iteration 2 and not divergent in following iterations. The figure 4.8 is the estimated phase per iteration. The dotted red curve is added random phase error, and the star blue curve is the estimated phase via five times iteration. It is found that the estimated phase via two iterations is similar with iteration 3, 4, 5 and the phase

is different from iteration 0 and 1. It is worth noting that the final image is as good as the original one.

Following, we discuss the stripmap image without artificial blur we added will be improved or not. Figure 4.9 shows the result. The left one is a part of the original image, and the right is via two iterations of PCA. There is no any difference between each other and no divergence after many iterations. In other word, the quality of image won't become worse. The reason is the same as the PPP simulation in the preceding chapter.

Finally, we also take the map without sensor-based mocomp to implement. Figure 4.10 is the image without sensor-based mocomp. Figure 4.11 is the image via iteration 0 ~ 5. There is obviously much improvement after iteration 2. It is more convenient to compare the difference. Figure 4.12 is the complete size image. It is impossible for us to compensate the no mocomp image as good as image with navigation mocomp because there is too much unknown aircraft dynamics information. PCA successfully makes scatters more convergent that proves the algorithm works!

Besides above implementations, there we discuss the effect of the window size selection. We see Figure 4.13, and there are three kinds of window size. The top one is general size, and the size is most used. The middle one is larger than the first, and we mainly compare these two kinds of window size. In our experience, the second window performs better than the first, but the convergent rate is more slowly. In Figure 4.14, the left one is the result of general size window, and the middle one is corresponding to larger size window. We observe these two in detail, and the middle image is better than the left. The scatters are more convergent as iteration times increase. The iteration time we mark is the times to reach a stable situation. The bottom one of Figure 4.13 is the biggest

window that almost encompasses the entire image. We find that the result, the right one of Figure 4.14, is not good, and the whole image is degraded. Different window sizes correspond to the cases of different scenario. It is by rule of thumb that the first window never encompasses the whole image in actual data.

From this discussion, window selection is always an important issue for PGA based algorithms, and the selection will determine the implementation result good or not.

4.2 Stripmap Phase Gradient Algorithm

The disadvantage of PCA is it does not estimate local linear sways. Thus if there was not at least a point target per synthetic aperture length, i.e., aperture coverage is insufficient, then the reconstructed image can be disjoint. SPGA overcomes this problem by estimating the local linear sway. Therefore we introduce SPGA[30] as new improved algorithm here. The algorithm is suitable for SAS system not for SAR system. We want that someone will take this work and modify the algorithm to be suitable for our application.

The algorithm described here unifies the discussion of stripmap autofocus. Instead of deriving an algorithm based on narrow-band/narrow-beam approximations a model with as few approximations as possible has been used. Traditional autofocus algorithms may be derived from this algorithm by applying appropriate approximations. Thus, all of the algorithms previously discussed in the chapter can be described in the framework of the algorithm presented here.

The stripmap phase gradient autofocus (SPGA) algorithm consists of a number of subproblems that described in separate sections later in this chapter.

The SPGA algorithm operates as follows:

1. Starting with the blurred image f(x,y), a number of prominent targets in the image are selected and the coordinates (x_m,y_m) recorded. Typically, a fixed number of bright scatterers are selected.
2. Applying 2-D window function as following :

$$w_m(x,y) = rect(\frac{x-x_m}{W_x})rect(\frac{y-y_m}{W_y}) \tag{4-8}$$

where W_x and W_y are the range and azimuth widths respectively. These widths are chosen to be as small as possible, but the window must encompass image blurring information.

3. Masking the blurred image with the window

$$f_m(x,y) = f(x,y)w_m(x,y) \tag{4-9}$$

4. Applying 2-D Fourier transform to windowed image

$$F_m(f_x,f_y) = F_{x,y}\{f_m(x,y)\} \tag{4-10}$$

5. Finding the true target position, $(\hat{x}_m,\hat{y}_m)$ of the individual targets that estimate by using the techniques. Using centroiding to estimate the spatial Doppler shift :

$$\Delta f_y = \frac{\iint f_y \left|F_m(f_x,f_y)\right|^2 df_y df_x}{\iint \left|F_m(f_x,f_y)\right|^2 df_y df_x} \tag{4-11}$$

The spatial Doppler shift estimate, $\widehat{\Delta f_y}$, is mapped into an estimate of $(\hat{y}_m - y_m)$ using the wavenumber transform

6. The region of interest images are phase modulated to correct for the shift in target position

$$F_m^{'}(f_x, f_y) = F_m(f_x, f_y)\exp(-jf_x(\widehat{x_m} - x_m) - jfy(\widehat{y_m} - y_m)) \tag{4-12}$$

7. Applying the wavenumber transform

$$X_m(f_x, u) = SC\{F_m^{'}(f_x, f_y)\} \tag{4-13}$$

where the modified wavenumber transform coordinate mapping SC{} is given by

$$\begin{aligned} f_x &= f_x \\ f_y &= f_x(\frac{\hat{y}_m - u}{\hat{x}_m}) \end{aligned} \tag{4-14}$$

8. Phase gradients of $X_m(f_x, u)$ in the along-track direction are calculated and averaged for all m. The ML phase gradient estimator is employed

$$\widehat{\Delta\phi}(u) = Arg\{\int_{f_x} \widehat{\Delta\eta}(f_x, u)df_x\} \tag{4-15}$$

where

$$\widehat{\Delta\eta}(f_x, u) = \sum_m X_m(f_x, u)X_m^*(f_x, u + \Delta u) \tag{4-16}$$

is used to calculate the individual phase gradients.

9. A final phase estimate, $\hat{\phi}(u)$, is generated from the phase gradients via cumulative summation or various other methods depending on the phase estimation kernel selected. When the ML phase gradient estimator is used, this is summarized by

$$\hat{\phi}[u] = \sum_{q=1}^{u-1} \widehat{\Delta\phi}[q] \qquad \hat{\phi}[0] \equiv 0 \tag{4-17}$$

10. The final phase-estimate $\hat{\phi}(u)$ is converted into a sway estimate via

$$\widehat{X}(u) = \frac{\hat{\phi}(u)}{2k_0} \tag{4-18}$$

11. The new sway estimate is used to motion compensate the image and the

algorithm iterates from step 1 until the sway estimate is less than some threshold.

The SPGA algorithmic steps are in Figure 4.15.

4.2.1 Target region selection

Target region selection is important for SPGA algorithm. If poor targets are selected their influence is reduced by the better phase-estimation kernels. However, enough points are required to achieve the desired autofocus accuracy.

SPGA currently selects a predetermined number of target regions in order of descending energy. Regions are selected from any range bin and only need be separated by current resolution at a particular iteration. Full aperture coverage is essential, so enough points must be selected to ensure coverage of each section of the aperture. Typically 100-1000points would be selected for a 50m by 50m SAS image this number should provide aperture coverage and allow accurate autofocus performance. Note that the selection of alias lobes from undersampled imagery should be avoided to prevent autofocus bias.

Autofocus accuracy improves as more points are selected, thus having the aperture covered with multiple targets is desirable. However, a trade-off is not as straightforward as it appears. QPGA for example, selects 4-8 times more targets than traditional PGA but is accurate enough to autofocus without iteration. Thus selecting more targets can reduce computational burden – particularly with the large iteration cost involved in stripmap reconstruction. This observation is often overlooked – autofocus accuracy improves even when using pints with low signal-to-clutter ratios.

Another strategy for improving accuracy is to select widely separated points.

This is performed to ensure that the pints selected are independent i.e., are not different parts of the same object. Phase-estimation improves since the averaging of random target phase is not biased by a few strong-scattering extended targets. To some extent, the 2-D windowing used in SPGA alleviates this problem by ensuring sufficient along-tack and across-track separation.

4.2.2 Window width selection

SPGA windows individual targets to ensure that only one scatterer at one time is used. If the window encompasses a number of targets, the phase estimation has poor results. As the algorithm iterates, the along-track window size decreased, reducing the likelihood of multiple targets within a single window. Selecting many targets at different locations improves sway estimation.

Window width selection is an important task for any PGA-based algorithm. The window acts to improve the signal-to-clutter ratio of the phase estimation and gas a large impact on algorithm performance. In PGA, the window width requires careful selection to avoid discarding useful autofocus information. This is apparent if the window does not fully encompass the target blurring; the algorithm is unable to estimate high order aperture phase components leading to residual blurring. SPGA, like PGA, starts with a wide window that decreases over time. This window must encompass all blurring, allowing estimation of errors contained in both low-order and high-order blurring components. Usually, the initial window width is selected to exceed the expected blurring to avoid data loss.

SPGA has slightly different windowing requirements to traditional PGA/PCA style algorithms due to the along-track position estimation. SPGA

requires that the window width is selected to allow accurate Doppler centroid estimation. This means that a small number of along-track side-lobes of the target peak must be encompassed. Windows are typically 2-3 times bigger for SPGA than for PGA/PCA with equivalent blurring. Curvature-based phase estimation removes the requirement for using larger along-track window width than PGA/PCA. Interestingly, SPGA's linear phase estimation and correction might not be required in the final iterations so the requirement for larger window widths at convergence is not certain. Further investigation is necessary. Another slight difference between SPGA windowing and PGA windowing is that with SPGA the window width should be range-variant since a fixed sway causes increasing blurring extent with range.

Traditional PGA window width selection techniques are not suitable for SAS autofocus. These typically measure the -10dB point of the centre shifted image. These techniques work well when a number of prominent scatterers occur in the image but fail in scenes consisting mostly of clutter. With clutter images the window width is decreased by a fixed fraction each iteration. This is needed since blurring only causes contrast loss in clutter scenes – no peak blurring is evident. SAS images have few prominent scatterers and need to use successive window width reduction.

4.2.3 Azimuth position estimation

The biggest difference between SPGA and other PGA/PCA based algorithms is the estimation of the along-track position of the targets before phase-estimation. This allows phase gradient averaging instead of phase curvature averaging and improves phase-estimation. Target position estimation

is necessary because phase gradient estimation is unable to estimate the linear phase trend caused by target shifting. Unknown linear phase trends prevent averaging of phase gradients and must be removed. In contrast, spotlight imagery has an unknown linear phase trend common to all targets so phase gradient averaging is possible.

The along-track target position is estimated using prior knowledge of the beam-pattern and spatial-frequency coverage of stripmap systems. Linear phase shift cause frequency shifting. Thus the linear phase trend across the target may be estimated by measuring the shifting of the target's along-track spatial Doppler spectrum.

Three methods[30] for determining the Doppler spectrum shift have been employed with SPGA to date. There are:

Doppler centroiding – Doppler centroiding operated by estimating the centroid of the power spectrum averaged over range. This may be described by

$$\Delta f_y = \frac{\iint f_y \left| F_m(f_x, f_y) \right|^2 df_y df_x}{\iint \left| F_m(f_x, f_y) \right|^2 df_y df_x} \tag{4-19}$$

where $F_m(f_x, f_y)$ is the wavenumber-domain data of the target patch. The centroid of the power spectrum is preferred to the centroid of the amplitude spectrum since linear trends cause energy-shifting and the amplitude can be adversely affected by higher order sway. Note that the centroid estimate also causes an underestimation of the shift due to the circular repetition of the Doppler spectrum caused by Doppler aliasing. This aliasing can be resolved using a circular centroid estimator of the form

$$\Delta f_y = \frac{f_{y\max}}{\pi} A\mathrm{rg}\{\iint |F_m(f_x, f_y)|^2 \, df_x \exp(j\frac{2\pi f_y}{f_{y\max}} - f_{y\min})df_y\} \tag{4-20}$$

Modified Doppler centroiding – using the observation that the centre of the target scene is not necessarily the centre of energy and that Doppler centroid estimation aims to estimate the linear trend via energy-shifting, a modified centroid estimation can also be used. This uses the centroid of the image scene and correcting the normal Doppler centroid result. Centroiding of the target scene improves the estimation when multiple targets are selected in a region of interest – the limiting condition on autofocus performance. Similar observations might also aid the correlation estimator discussed next

Doppler correlation – Doppler correlation is another method of estimating the Doppler shift and does not suffer underestimation due to Doppler aliasing. Doppler correlation operates by correlating the spectrum of the target patch with the expected spectrum. This operation is described by

$$\Delta f_y = \max_{f_y}\{\int F_m(f_x, f_y)df_{x*}A(f_y)\} \tag{4-21}$$

where

$$A(f_y) = (D_t \operatorname{sinc}(\frac{f_y D_t}{2\pi})D_r \operatorname{sinc}(\frac{f_y D_r}{2\pi})) \tag{4-22}$$

is the expected amplitude spectrum.

Of the methods, modified Doppler centroid estimation is preferred because of its improved accuracy and reduced computational requirements.

It is worth noting that the noise performance of position estimation is different to that of the phase estimation kernels described earlier. Clutter suffers the same linear shift as the selected target so has the same spatial Doppler spectrum. The result is that the performance of the estimator is determined by the signal-to-noise ratio not the signal-to-clutter ratio. The difference can be as

much as20-40dB

Estimating the true position of the targets allows the use of phase gradient estimation. This is possible when platform yaw is negligible or accurately measured. When unknown yaw is present, workarounds allow the use of phase gradient estimation. In addition, phase curvature based estimation can be used with a decrease in performance. Finally, phase gradient based estimation is sensitive to the accuracy of the along-track target position estimation, and any technique for improving those estimations will benefit the SPGA algorithm.

4.2.4 Wavenumber transform

The modified wavenumber transform coordinate transform SC{} acts to transform the phase error estimates from the individual targets into a space-invariant domain prior to averaging. This is a typical step in space-variant image processing. The success of the wavenumber transform in accomplishing this results from its similarity to the reconstruction process. Both PGA and PCA use approximations of the wavenumber transform tailored to the environment they are designed for : PGA, a Fourier transform; and PCA an along-track chirp spreading.

The wavenumber transform maps image blurring to the phase error in the pulse compressed data that caused the blurring. Once in the pulse compressed data domain, the phase errors have redundancy in along-track position of the target adversely affects the average. However, the difference between the image target position and the estimated trued target position is small for later iterations and has little effect on the average. The use of the wavenumber transform allows the phase error information from multiple targets to be combined in sensible

fashion.

4.2.4.1 Wavenumber geometry based derivation

The wavenumber transform is the basis of the stripmap blurring model used in this thesis and as such is also the basis of the SPGA autofocus method.

The wavenumber transform can also derived using straightforward geometrical arguments.

Starting with the angle to target θ

$$\theta = \tan^{-1}\left(\frac{u - y_n}{x_n}\right)$$

and noting that in the far-field of the target patch the spatial frequency coverage is described by the same angle

$$\theta = \tan^{-1}\left(\frac{f_y}{f_x}\right)$$

The wavenumber transform relation may now be derived as

$$u = y_n - \frac{f_y x_n}{f_x}$$

4.2.5 Phase Estimation

Phase estimation is the heart of the SPGA method. Like PGA, small improvements to the phase estimation kernels can yield large improvements in the end result. The phase estimation kernel used depends on the accuracy of the along-track position estimation. When the along-track position estimates are accurate, a phase gradient method should be used. However, when the along-track position of the targets is unknown or inaccurate, a phase-curvature

kernel must be used. Even with unknown along-track positions, a phase gradient kernel can be used once an estimate of the phase error is obtained using phase curvatures.

The kernel currently used of SPGA is the ML phase gradient estimator. WPGA provides a better phase estimator than the ML estimator suggested. However, the weightings used in WPGA's derivation are calculated based on spotlight SPGA assumptions and the iterative scheme employed cannot currently be used in a stripmap framework.

4.2.6 SPGA vs. traditional algorithm

SPGA provides a framework which can describe a number of traditional autofocus algorithms.

PGA

SPAG implements PGA if narrow-band and narrow-beam approximations are made and the along-track position of the targets is assumed to be zero in the phase compensation and interpolation. Under these approximations, the wavenumber transform becomes the along-track Fourier transform that PGA uses. The centre shifting step of PGA is implicitly implemented by SPGA's centroiding the Fourier transform around the target region of interest. Further work is required on the phase estimation technique of the more sophisticated PGA.

PCA

PCA is implemented by using narrow-band, narrow-beam approximations and using a phase curvature based phase estimation.

PPP

PPP is implemented by allowing SPGA only a single point and using a phase-averaging estimation kernel.

Chapter 5 Conclusion

Many autofocus algorithms are developed for SAR, especially for spotlight mode. We choose these two algorithms to research that because these two algorithms are not so mature and the relative research is not so much. Another reason is that our SAR data is stripmap mode, and stripmap data is also suitable for the two algorithms.

We make a comparison table to understand the effects of these two algorithms. Table 5.1 compares three algorithms PGA, PPP, and PCA. PGA is used to compare with PPP; PGA and PPP algorithm develop for spotlight mode originally.

The meaning of the notation in Table 5.1 : “ O ” expresses “ better ” which means the image can be restored or scatter can be more convergent; “ Δ ” expresses “ similar ” which means some scatters may be convergent but others not; “ X ” expresses “ worse ” that means the image can’t be restored or even worse.

In table 5.1, it is obvious that PPP algorithm is not as good as PCA algorithm for stripmap data. It does not surprise us since PPP algorithm is developed for ISAR system, and there are also some constraints in applying stripmap data. We only implement single point PPP so it may be not enough information for stripmap images. It is successful for PCA algorithm whether in spotlight or stripmap images. From above implementations, we discover stripmap autofocus algorithms also work with spotlight images.

Although PCA algorithm compensates the stripmap image successfully, there are also some disadvantages. The most one is linear phase term removal.

PCA can’t remove linear phase term in its algorithm but SPGA can. SPGA is derived from PCA, and the linear phase term removal is included in SPGA algorithm. Therefore, we think that SPGA performance will be better than PCA and hope the following junior in lab will complete the algorithm.

Reference

[1] Mancill, C.E., and J.M. Swiger, "A Map Drift Autofocus Technique for Correcting Higher Order SAR Phase Errors (U)," 27th Annual Tri-Service Radar Symposium Record, Monterey, CA, June 23-25, 1981, pp.391-400.

[2] United States Patent No. 4,999,635, G.N. Yoji, "Phase Difference Auto Focusing for Synthetic Aperture Radar Imaging," March 12, 1991.

[3] Eichel, P.H., D.C. Ghiglia, and C.V. Jakowatz, Jr., "Speckle Processing Method for Synthetic Aperture Radar Phase Correction," Optics Letters, Vol. 14, No. 1, Jan. 1, 1989

[4] United States Patent No. 4,924,229 P.H. Eichel, D.C. Ghiglia, and C.V. Jakowatz, Jr., "Phase Correction System for Automatic Focusing of Synthetic Aperture Radar," May 8, 1990.

[5] Eichel, P.H., D.C. Ghiglia, and C.V. Jakowatz, Jr., "Phase-Gradient Algorithm as an Optimal Estimator of the Phase Derivative," Optics Letters, Vol. 14, No. 28, Oct. 15, 1989, pp.1101-1109

[6] Wahl, D.E., P.H. Eichel, D.C. Ghiglia, and C.V. Jakowatz, Jr., "Phase Gradient Autofocus — A Robust Tool for High Resolution SAR Phase Correction," IEEE Transactions on Aerospace and Electronic Systems, Vol. 30, No.3, July 1994, pp. 827-834.

[7] D.W. Hawkins. "Synthetic Aperture Imaging Algorithms：with application to wide bandwidth sonar," PhD thesis, Department of Electrical and Electronic Engineering, University of Canterbury, Oct. 1996.

[8] A. Potsis, A. Reigber, J. Mittermayer, A. Moreira, and N. Uzunoglou. "Sub-aperture algorithm for motion compensation improvement in

wide-beam SAR data processing," Electronics Letters, 37(23)：1405-1407, Nov. 2001.

[9] W.G. Carrera, R. S. Goodman, and R. M. Majewski. "Spotlight synthetic aperture radar：signal processing algorithms," Artech House, 1995.

[10] M.P. Hayes and P.T. "Gough. Broad-band synthetic aperture sonar," IEEE Journal of Oceanic Engineering, 17(1)：80-94, Jan. 1992.

[11] S.Wang and X. Huang. "Autofocus techniques for reducing phase errors in UWB-SAR," Aerospace and Electronics Conference, 1997. NAECON 1997, volume 2, pages 14-18. IEEE, July 1997.

[12] P.T. Gough, M.P. Hayes, and D.R. Wilkinson. "An efficient image reconstruction algorithm for a multiple hydrophone array synthetic aperture sonar," In Proceedings of the fifth European Conference on Underwater Acoustics (ECUA) 2000, pages 395-400, July 2000b.

[13] D.E. Wahl, C.V. Jakowatz, Jnr., and P.A. Thompson. "New approach to strip-map SAR autofocus," In Sixth IEEE Digital Signal Processing Workshop, pages 53-56. IEEE, Oct. 1994b.

[14] J.C. Curlander and R.N. Mcdonough. "Sythetic Aperture Radar：Systems and signal processing," John Wiley & Sons, Inc., 605 Third Avenue, New York , NY, 1996.

[15] P.H. Eichel, and C.V. Jakowatz, Jnr., "Phase-Gradient Algorithm as an Optimal Estimator of the Phase Derivative," Optics Letters, Vol. 14, No. 20, Oct. 1989, pp.1101-1103

[16] C.V. Jakowatz, Jnr., D.E. Wahl, P.H. Eichel, D.C. Ghiglia, and P.A. "Thompson Spotlight-Mode Synthetic Aperture Radar：A Signal Processing

Approach," Kluwer Academic Publishers, Boston, 1996.

[17] D.E. Wahl, P.H. Eichel, D.C. Ghiglia, and C.V. Jakowatz, Jnr. "Phase Gradient Autofocus – a robust tool for high resolution SAR phase correction," IEEE Transactions on Aerospace and Electronic Systems, 30(3)：827-835, July 1994a.

[18] J.B. Pat. "Synthetic aperture sonar image reconstruction using a multiple-receiver tow-fish", Master's thesis, Department of Electrical and Electronic Engineering, University of Canterbury, Mar. 2000.

[19] T.J. Sutton, S.A. Chapman, and H.D. Griffiths. "Robustness and effectiveness of autofocus algorithms applied to diverse seabed environments", In Proceedings of the fifth European Conference on Underwater Acoustics (ECUA) 2000, volume 1, pages 407-412, July 2000.

[20] D.G. Thompson, J.S. Bates, D.V. Arnold, and D.G. Long. "Extending the phase gradient autofocus algorithm for low-altitude stripmap mode SAR" In Proceeding of the International Geoscience and Remote Sensing Symposium 1999, volume 1, pages 564-566, 1999.

[21] W.W. Bonifant, Jnr., M.A. Richards, and J.H. McClellan. "interferometric height estimation of the seafloor via synthetic aperture sonar in the presence of motion errors," IEE Proceedings on Radar, Sonar, and Navigation, 147(6)：322-330, Dec. 2000.

[22] J. Pihl, P. Ulriksen, O. Kroling, B. Lovgren, and G. Shippey. "MLO classification using an ROV-mounted wideband synthetic-aperture sonar." In Proceedings of the fifth European Conference on Underwater Acoustics (ECUA) 2000, volume 1, pages 433-438, July 2000.

[23] W.W. Bonifant, Jnr. "Interferometric synthetic aperture sonar processing"

Master's thesis, Georgia Institute of Technology, July 1999.

[24] Walter G. Carrara, ron S. Goodman, and Ronald M Majewski, Spotlight Synthetic Aperture Radar Signal Processing Algorithms, Artech House Boston London.

[25] Oppenheim, A.V., and R.E. Schafer, Digital signal Processing, Englewood Cliffs, NJ, prentice-Hall, Inc., 1975

[26] 楊嘉豐，「機載合成孔徑雷達訊號之運動補償」，國立交通大學，碩士論文，民國 93 年

[27] Werness, S.A., W.G. Carrara, L.S. Joyce, and D.B. Franczak, "Moving Target Imaging Algorithm for SAR Data," IEEE Transactions on Aerospace and Electronic Systems, Vol. 26, No.1, January 1990, pp.57-67.

[28] D. E. Wahl, C. V. Jakowatz, Jnr., and P. A. Thompson. "New approach to stripmap SAR autofocus." In Sixth IEEE Digital Signal Processing Workshop, pages 53-56. IEEE, Oct. 1994.

[29] D. E. Wahl, P.H. Eichel, D.C. Ghiglia, and C.V. Jakowatz, Jnr. "Phase gradient autofocus – a robust tool for high resolution SAR correction." IEEE Transactions on Aerospace and Electronic Systems, 30(3):827-835, July 1994.

[30] Hayden J. Callow B.E. "Signal Processing for Synthetic Aperture Sonar Image Enhancement", University of Canterbury, the thesis of Doctor of Philosophy in Electrical and Electronic Engineering, April 2003.

Figures

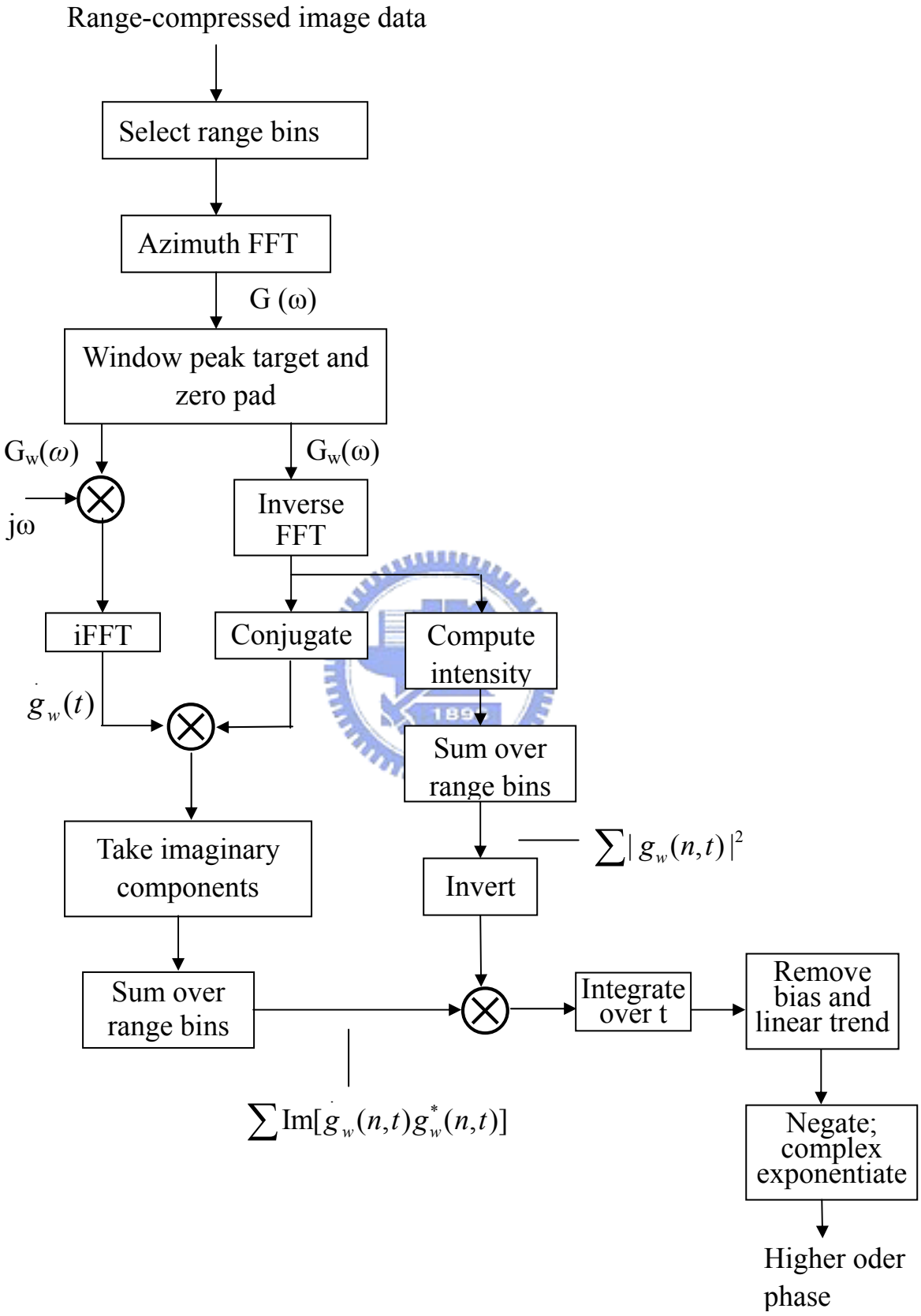


Figure 2.1 Block diagram of the phase gradient autofocus procedure

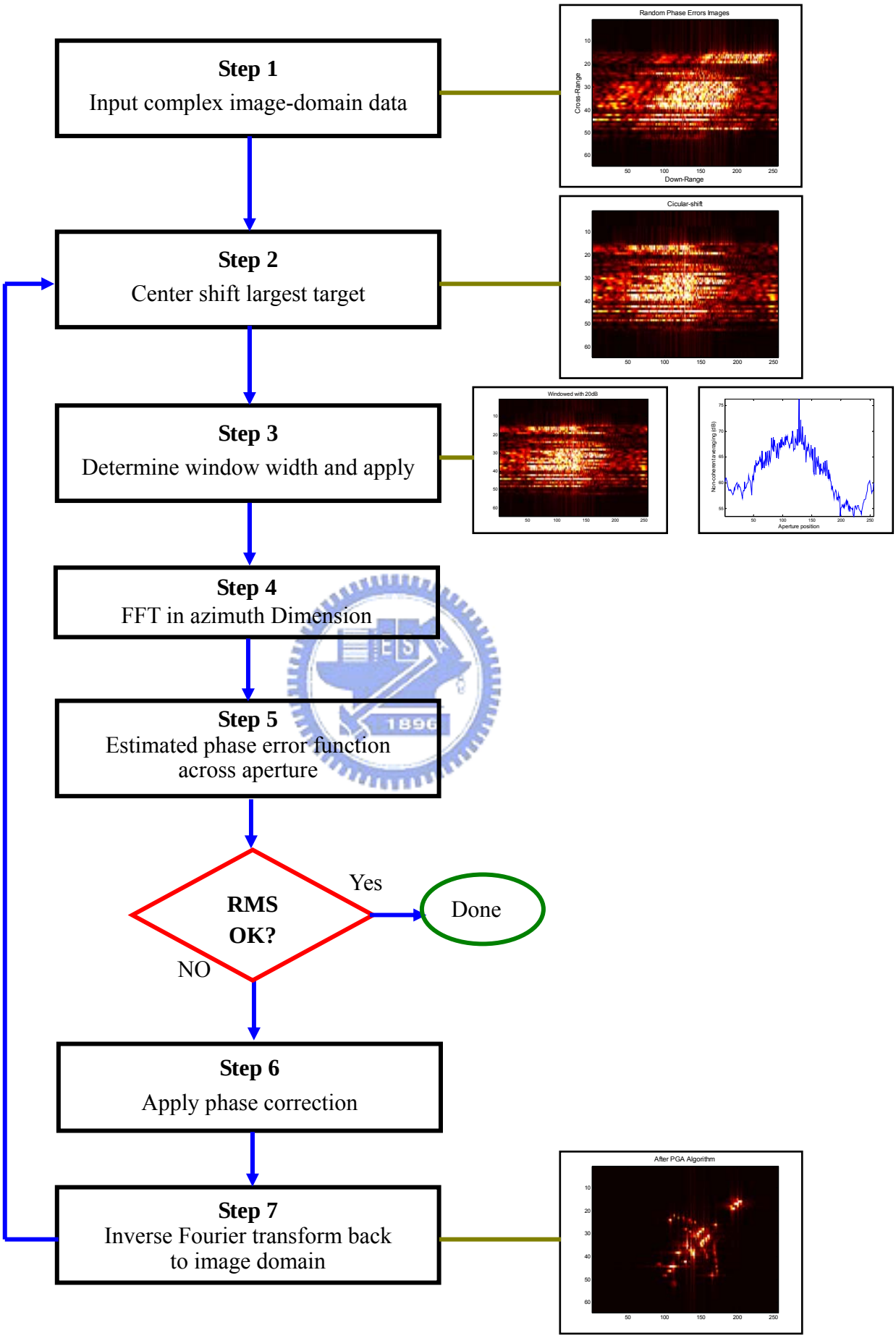


Figure 2.2 PGA algorithmic steps and corresponding images

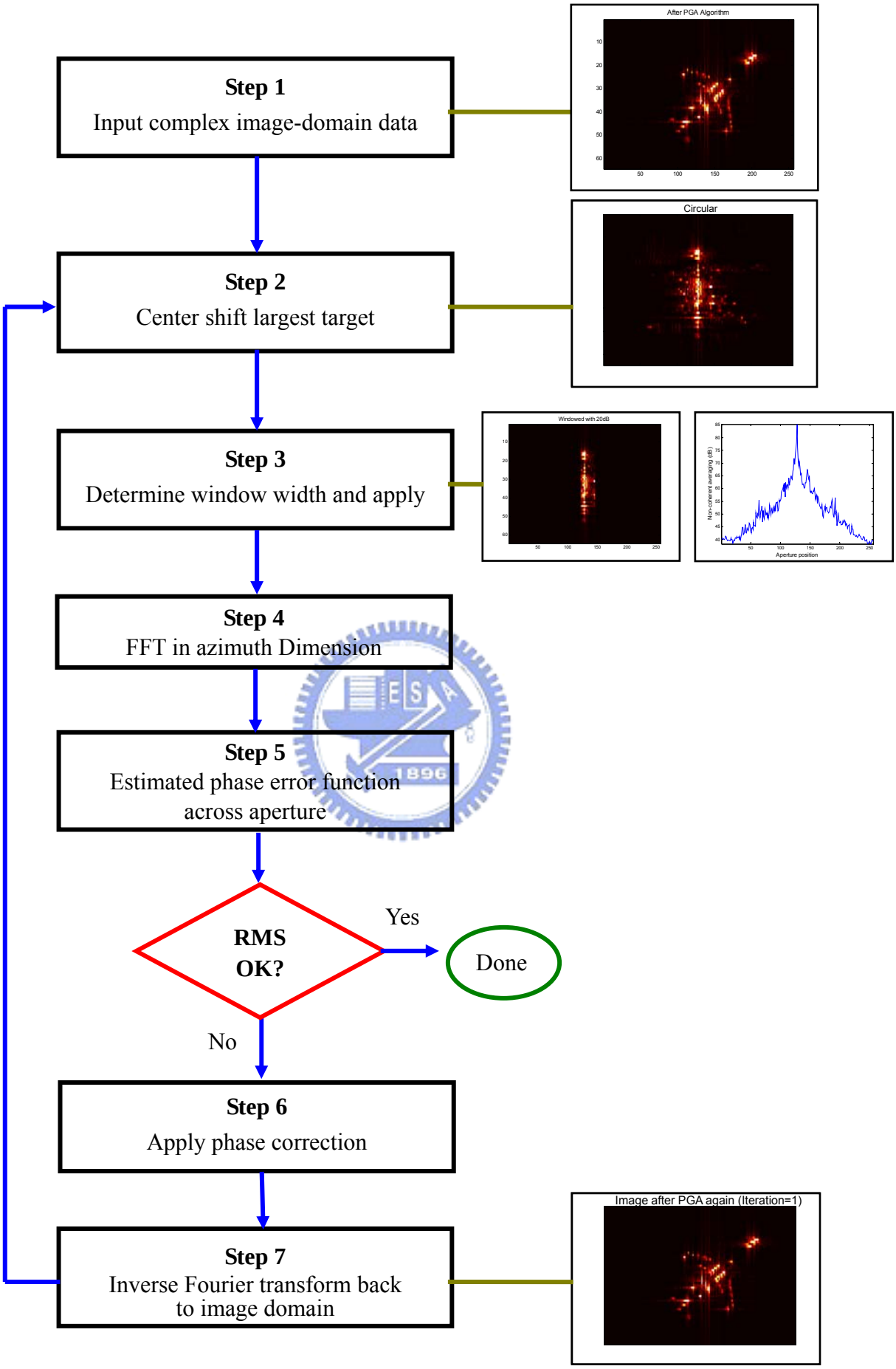
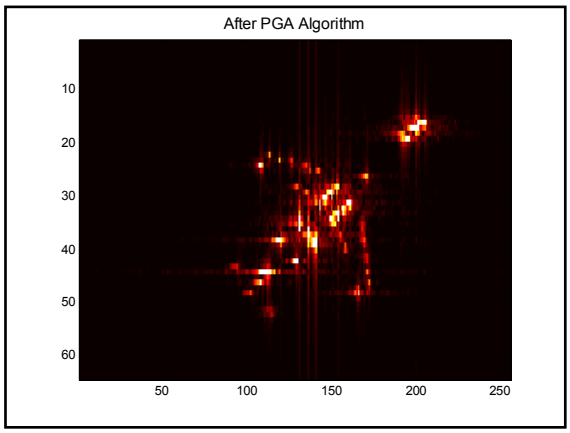
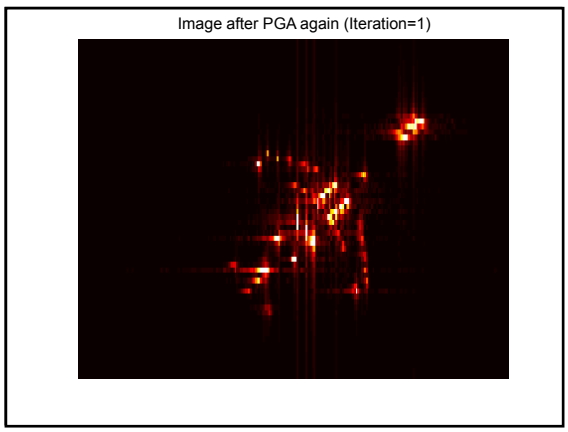


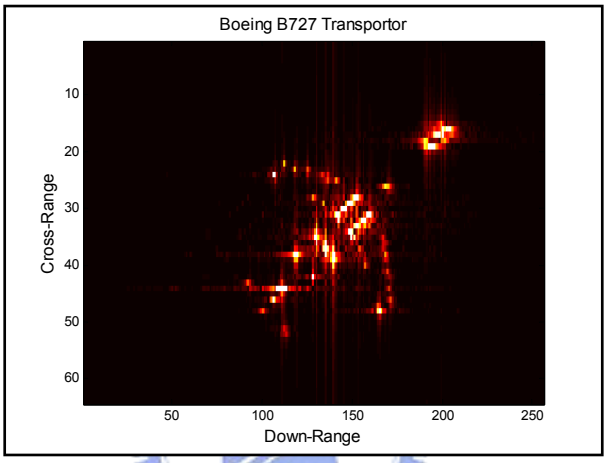
Figure 2.3 PGA algorithmic steps and corresponding images (iteration=1)



(a)



(b)



(c)

Figure 2.4 Compare processed images with original image
(a) processed image (iteration=0) (b) processed image (iteration=1)
(c) original image without phase errors

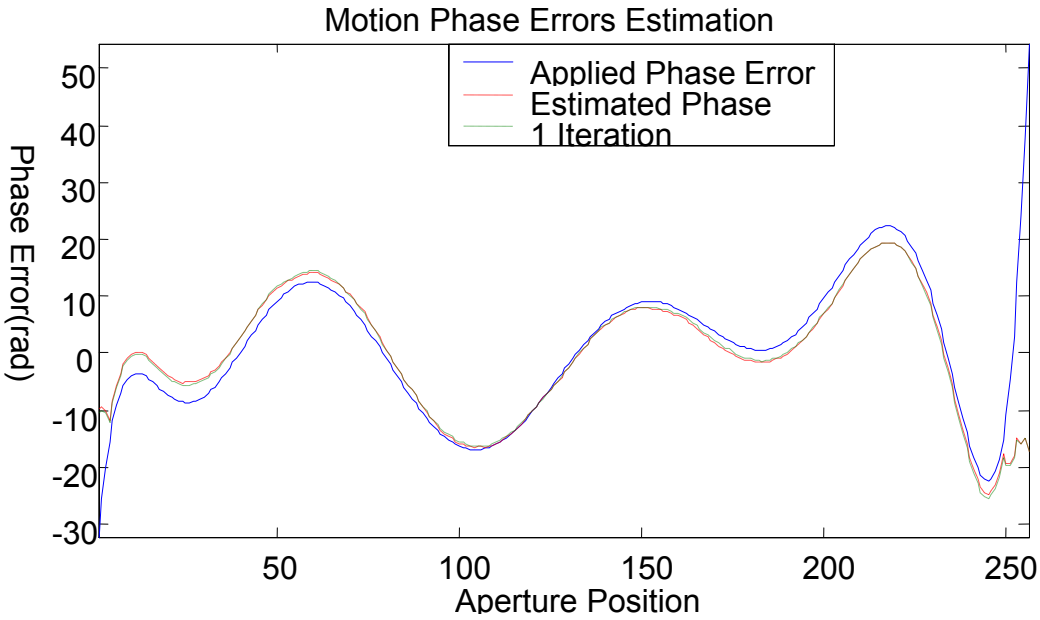


Figure 2.5 Estimated phase error function

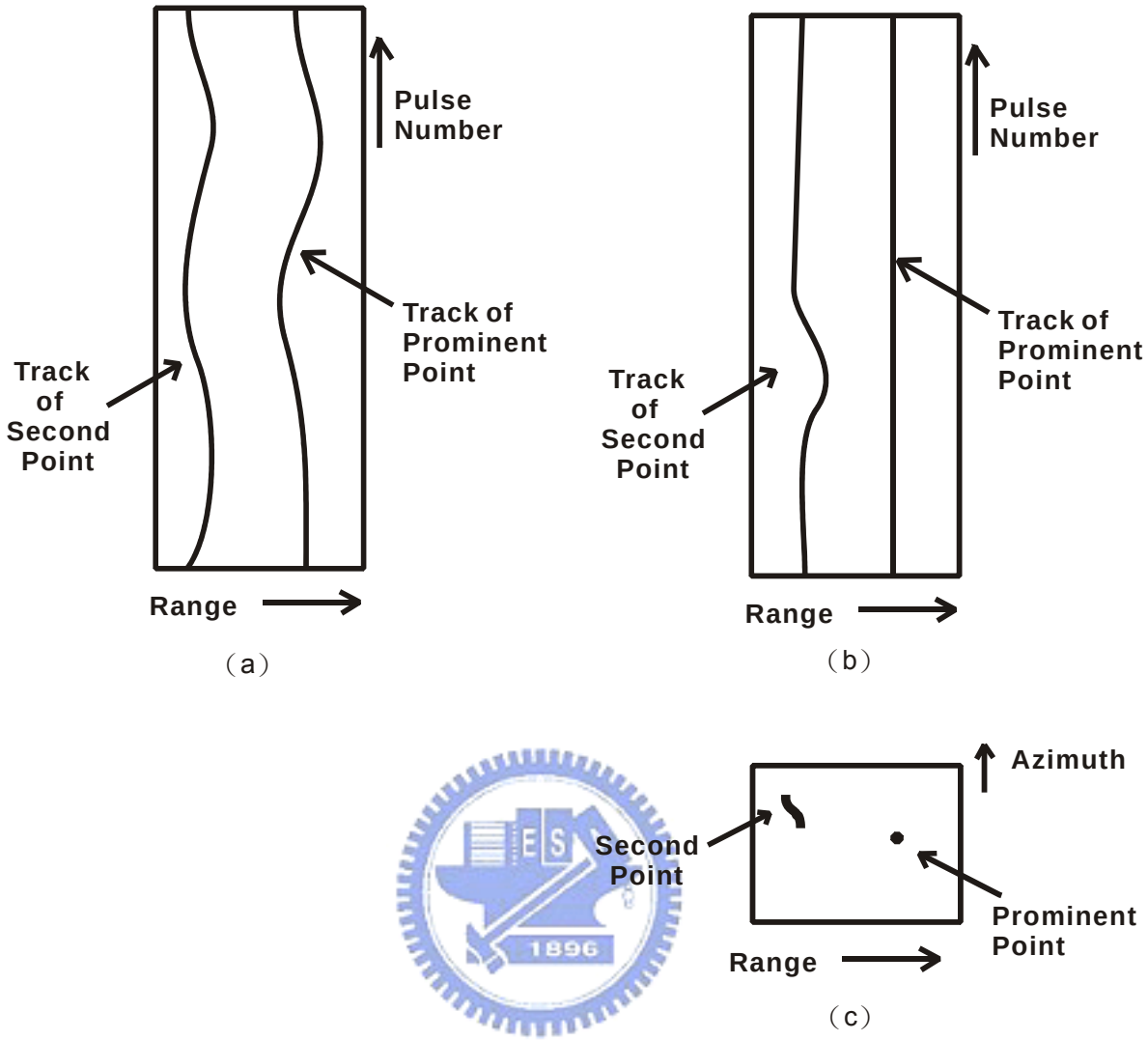


Figure 3.1 View of prominent point data at various stages in algorithm : (a) Track before compensation; (b)Tracks after compensation; (c)focused prominent point image

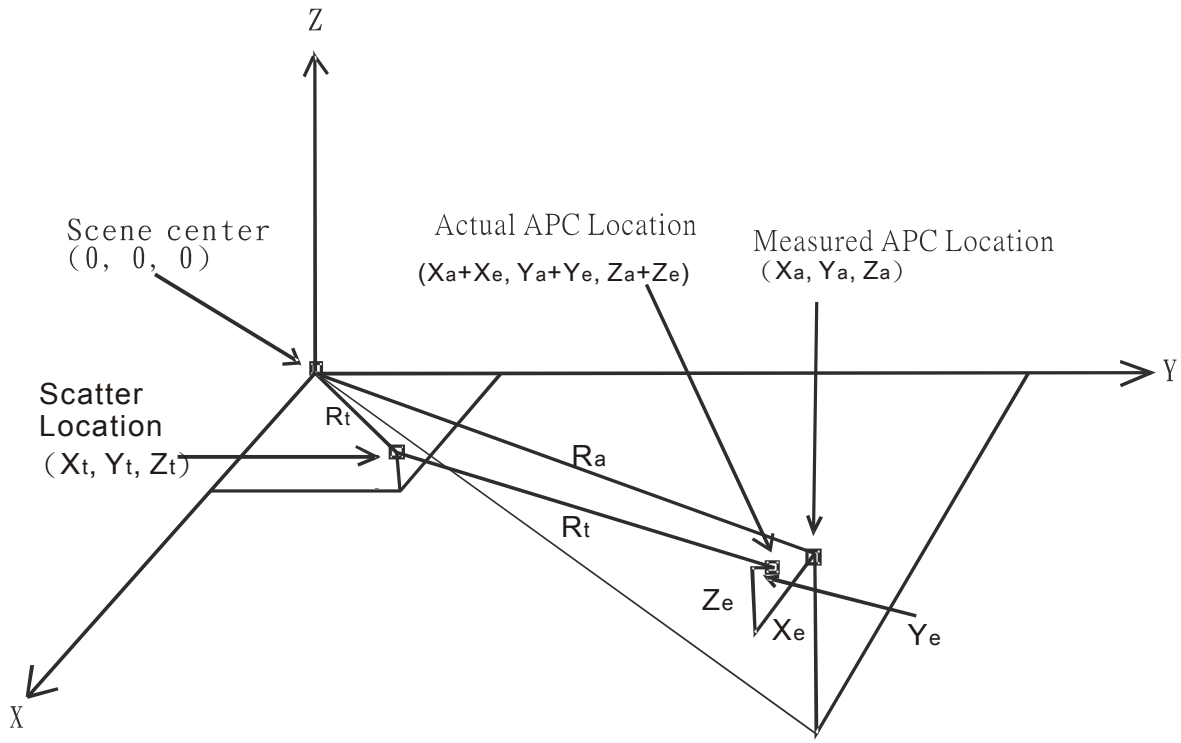


Figure 3.2 Measured and actual locations distances in the data collection geometry

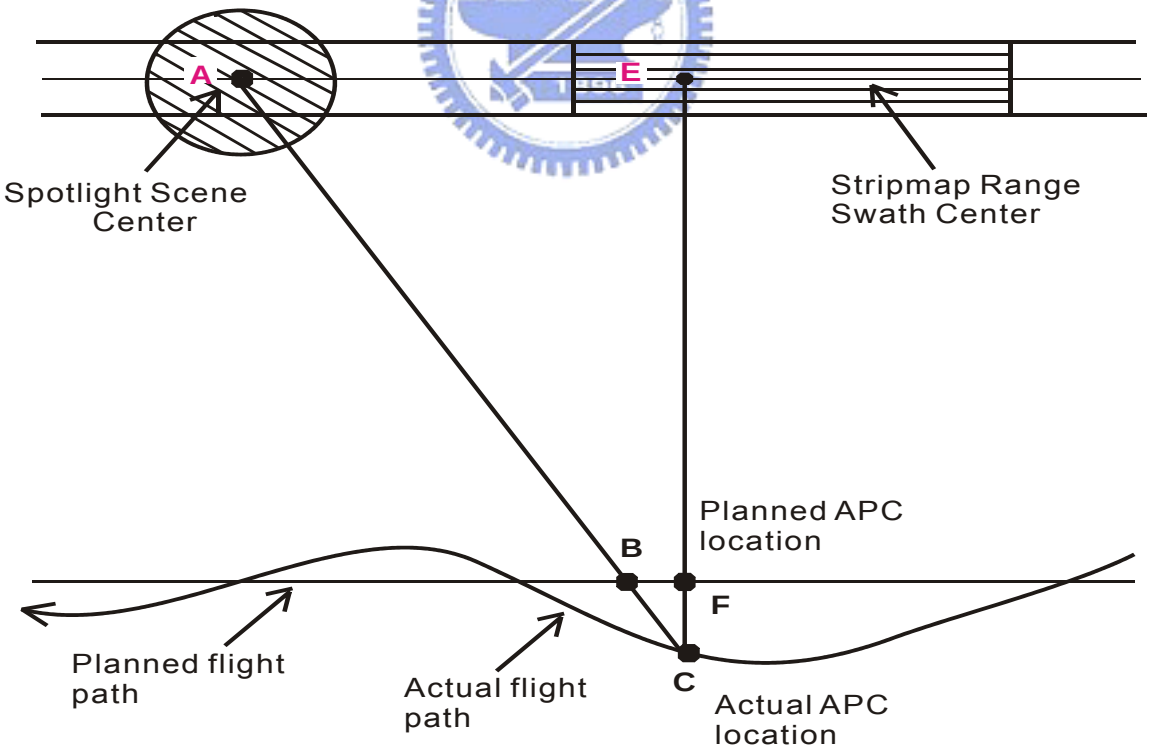


Figure 3.3 Motion compensation comparison

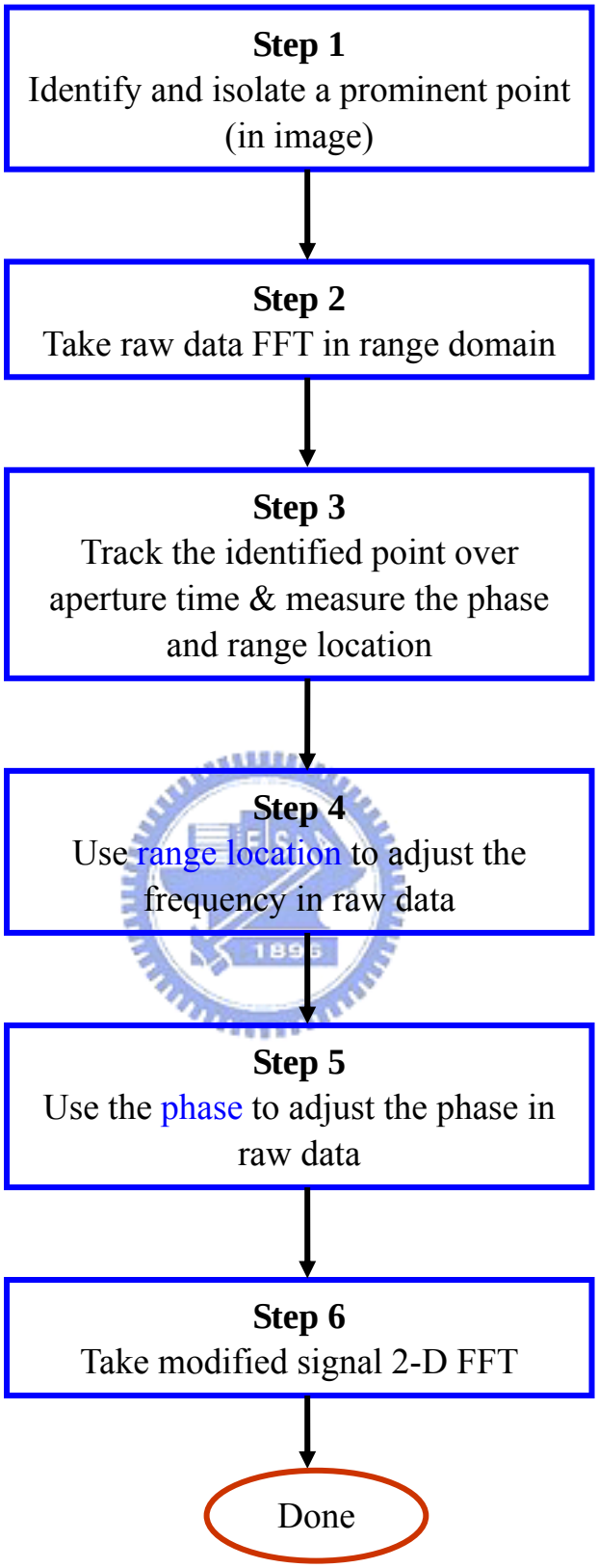


Figure 3.4 PPP algorithmic steps

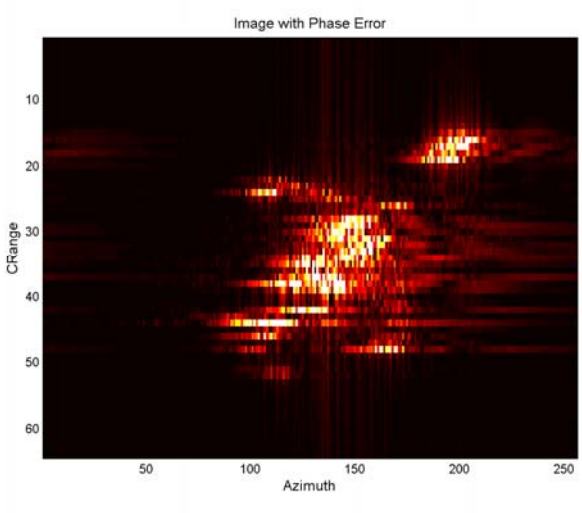


Figure 3.5 Input spotlight image for PPP

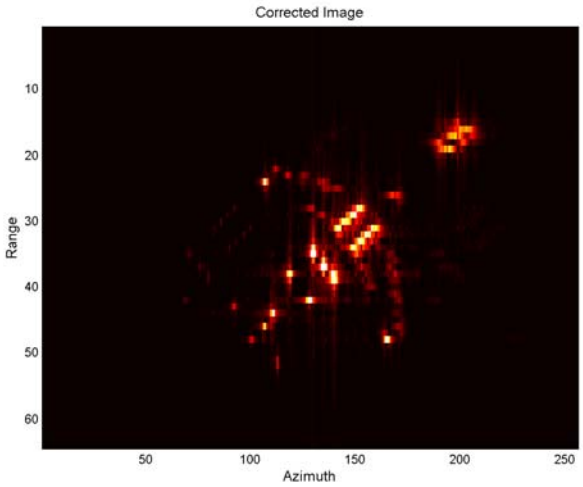


Figure 3.6 Corrected spotlight image by PPP

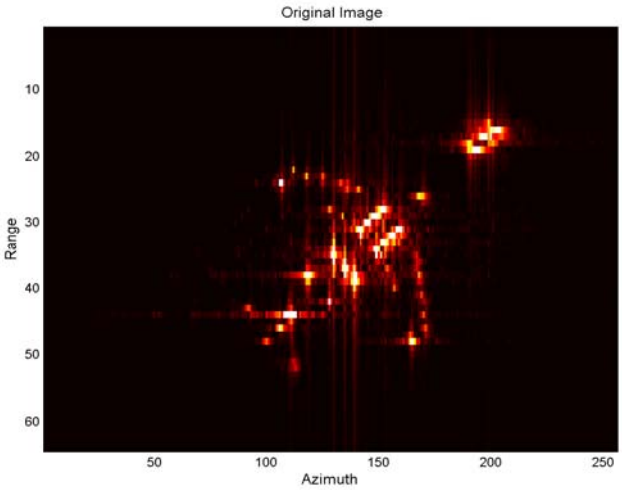


Figure 3.7 Original spotlight image

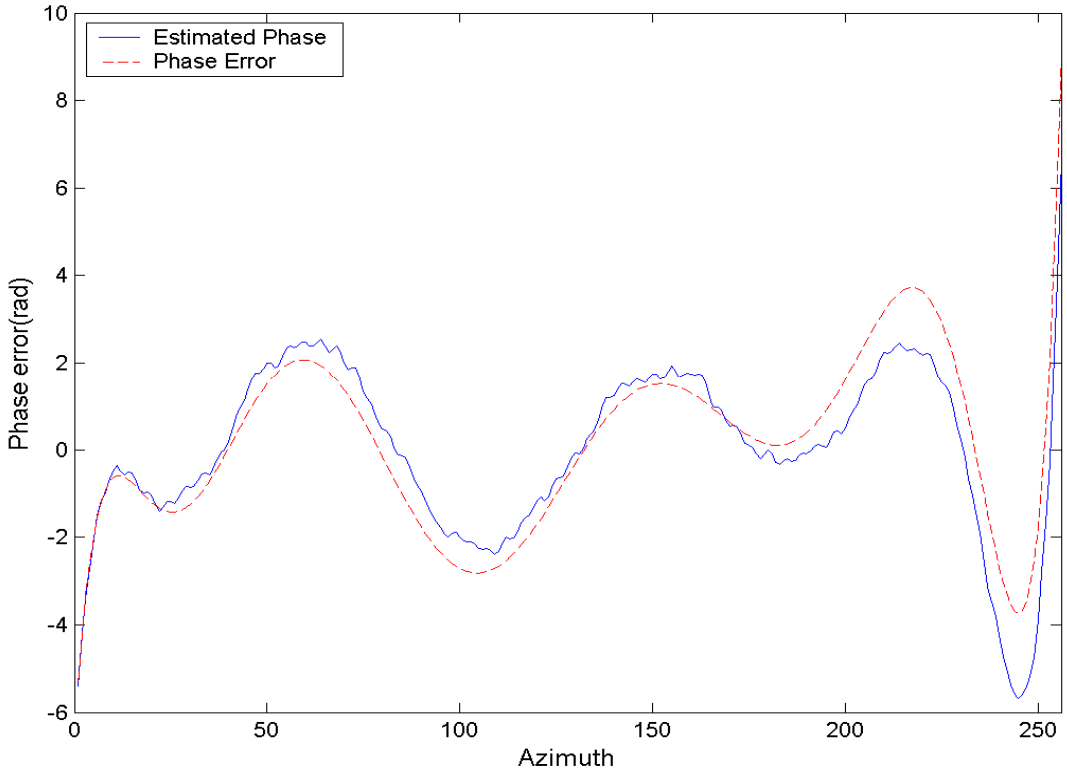


Figure 3.8 Estimated phase and random error

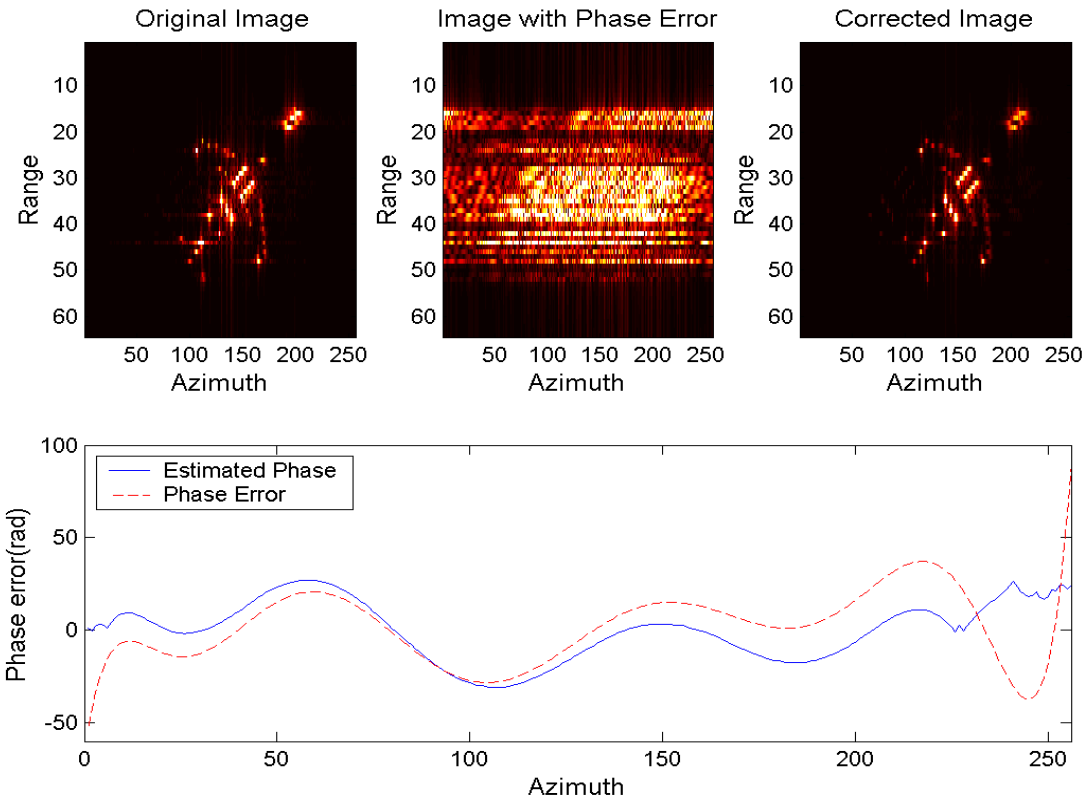


Figure 3.9 The total result with 10 times bigger random phase error

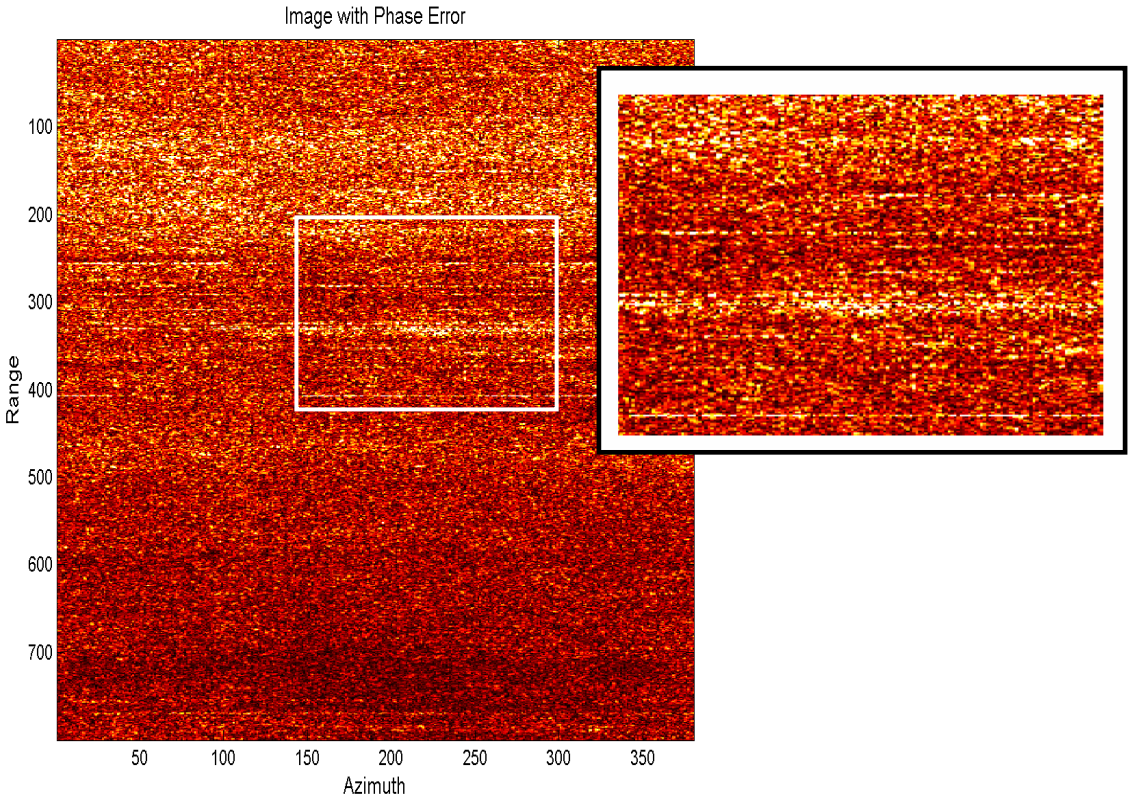


Figure 3.10 Stripmap image with random phase error

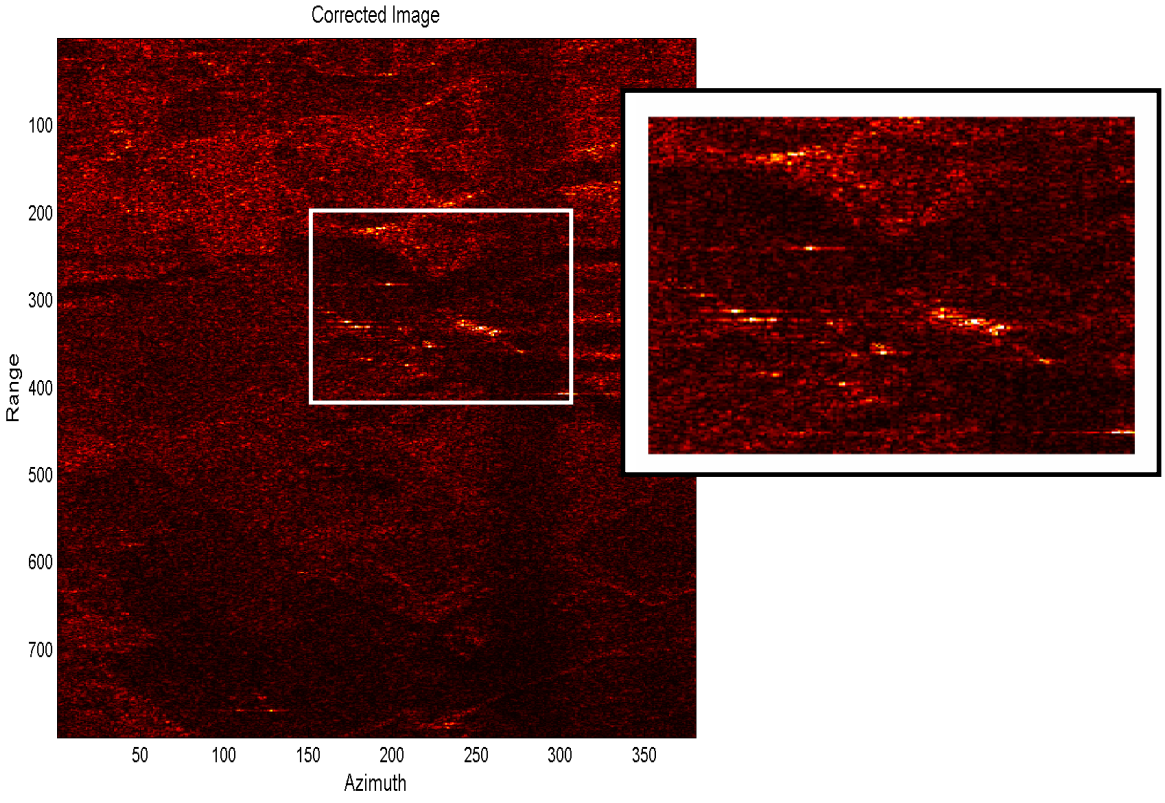


Figure 3.11 Corrected stripmap image

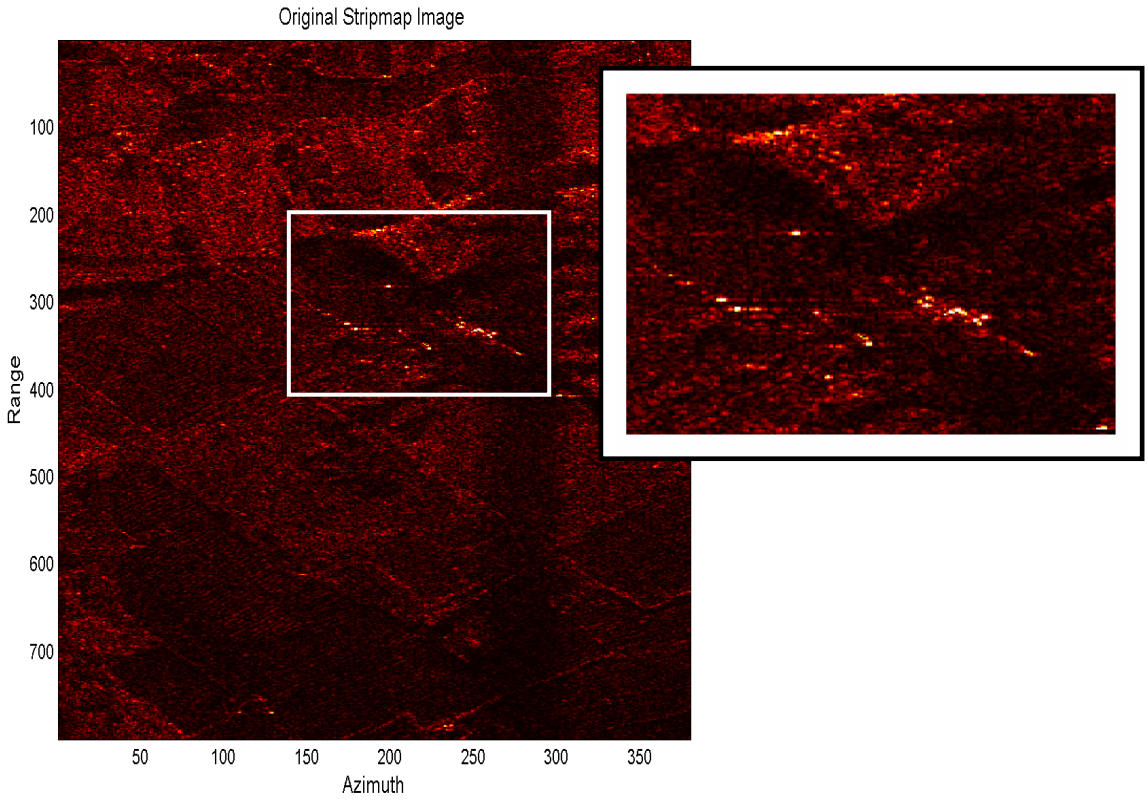


Figure 3.12 Original stripmap image

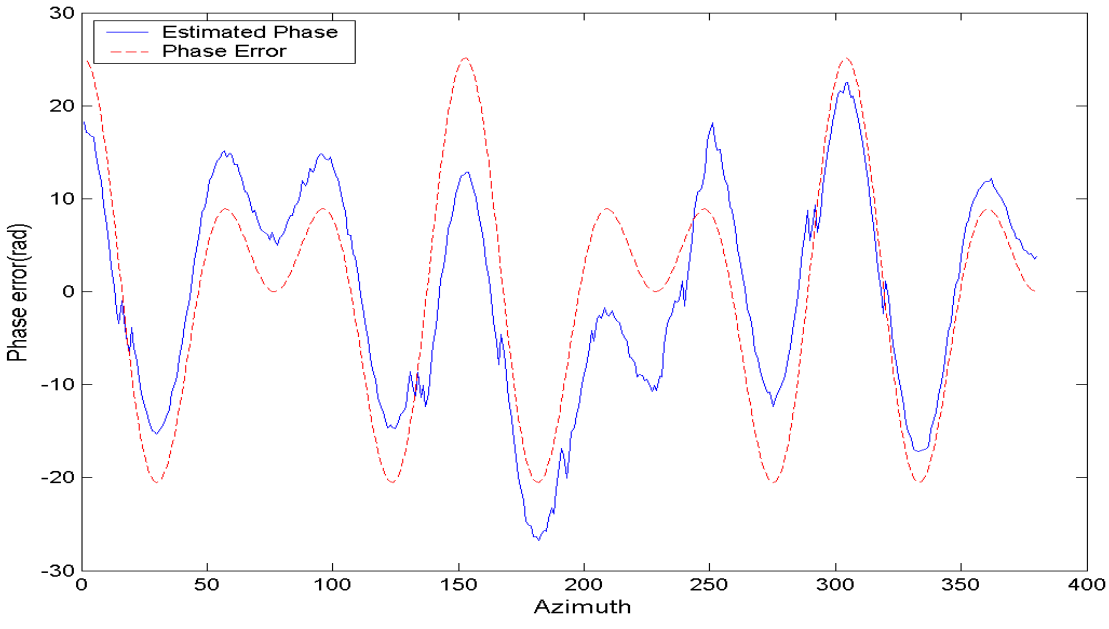


Figure 3.13 Estimated phase and random error

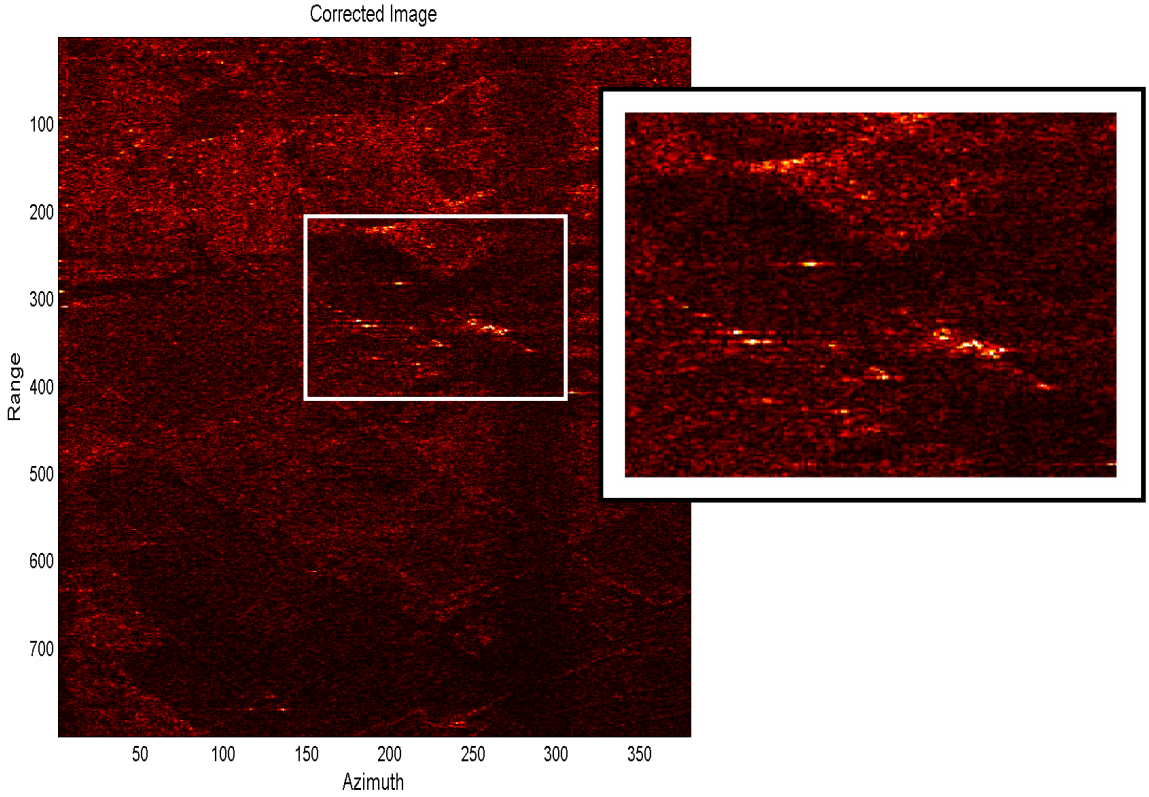


Figure 3.14 Corrected stripmap image without error

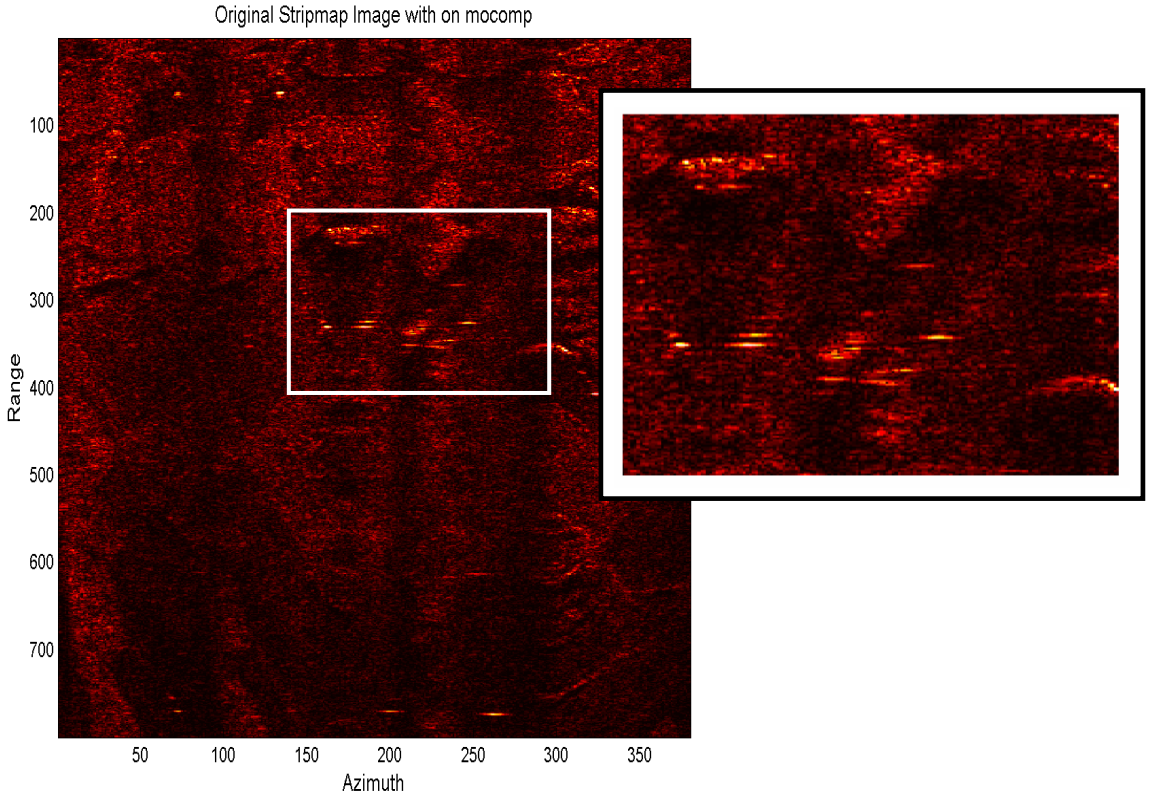


Figure 3.15 Stripmap image without mocomp

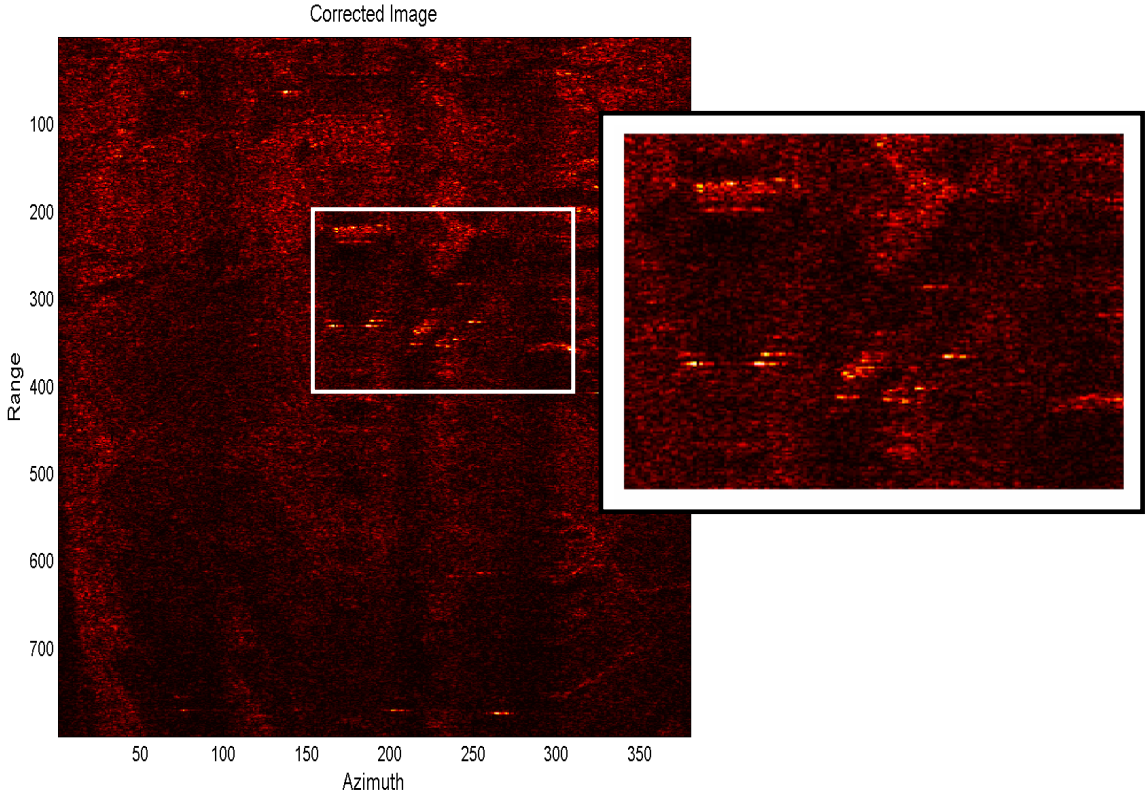


Figure 3.16 Corrected stripmap image without mocomp

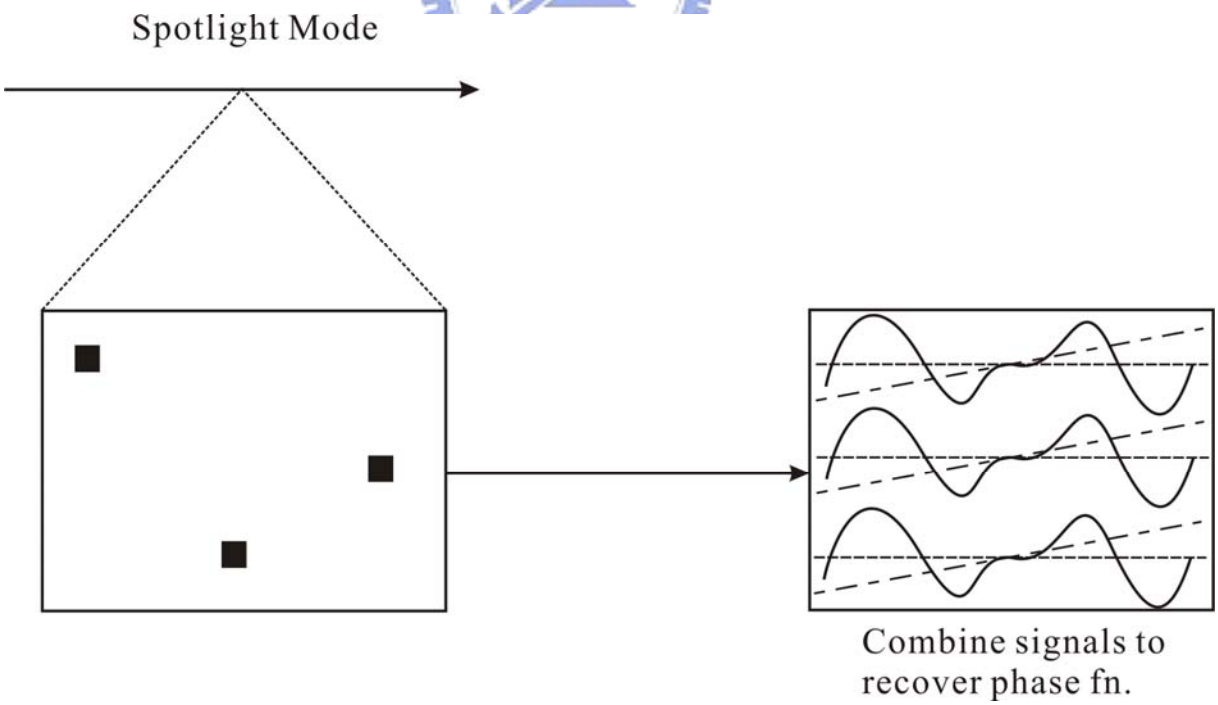


Figure 4.1(a) Phase error and linear phase for spotlight mode

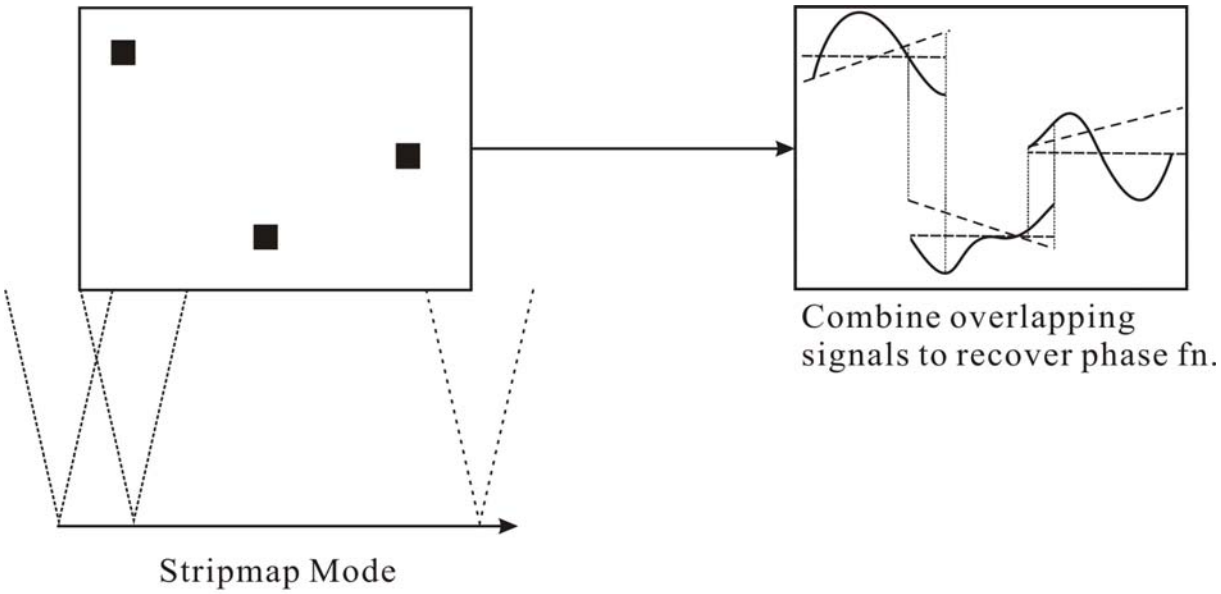


Figure 4.1(b) Phase error and linear phase for stripmap mode

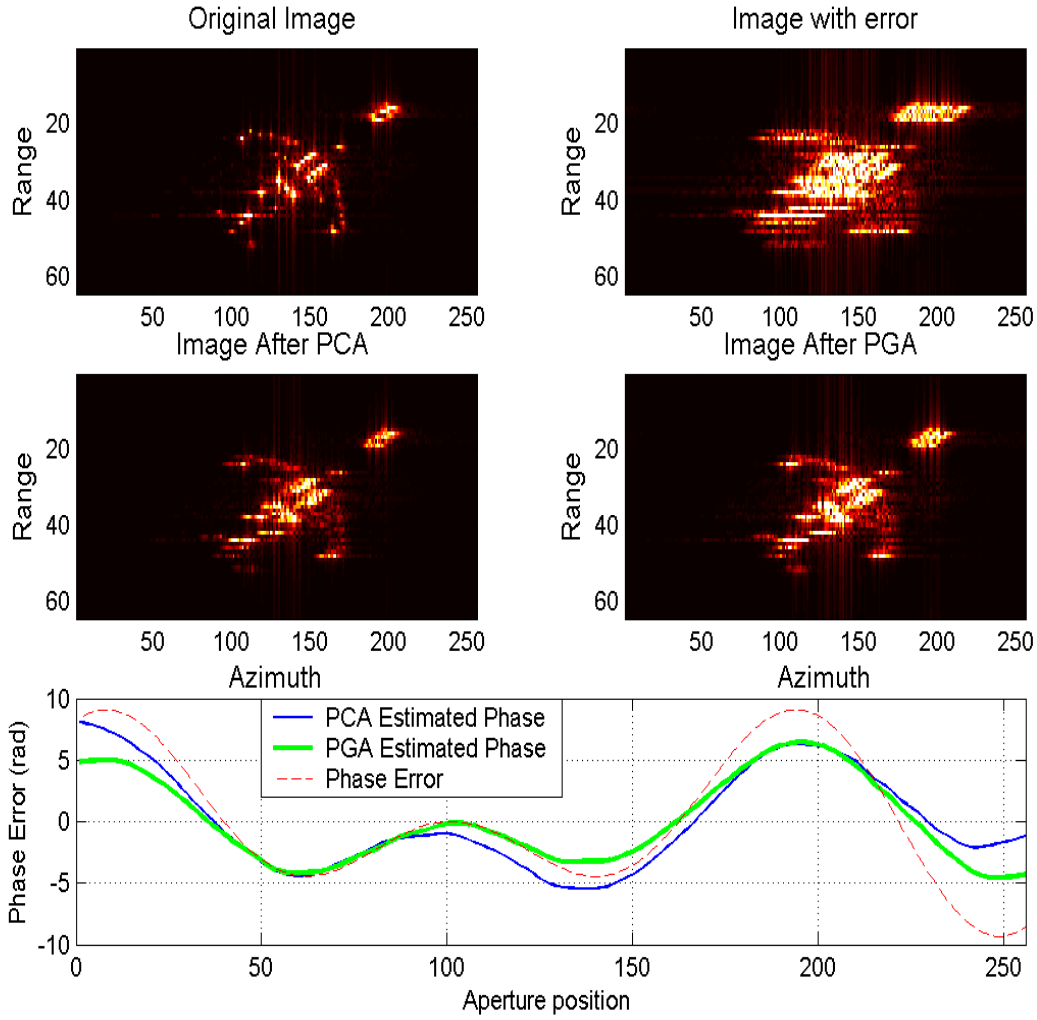


Figure 4.2 Corrected image after PCA and PGA (iteration = 0)

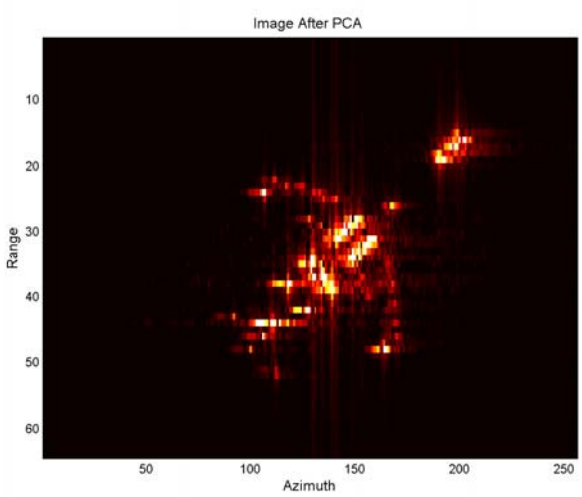


Figure 4.3 Corrected image after PCA (iteration = 2)

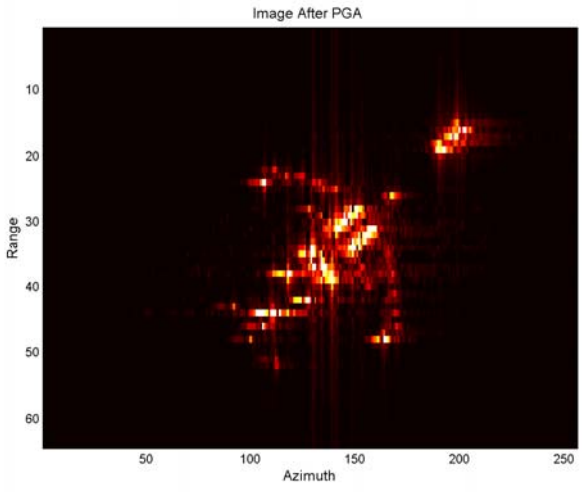


Figure 4.4 Corrected image after PGA (iteration = 2)

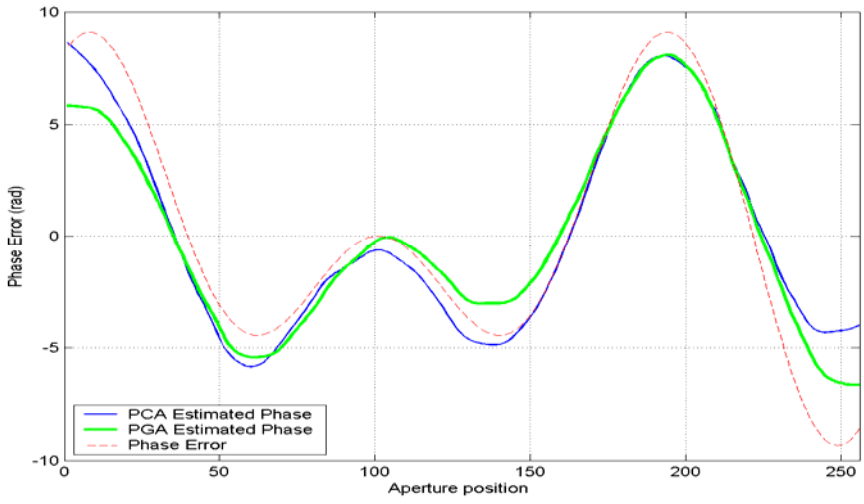


Figure 4.5 Estimated phase by PCA and PGA (iteration = 2)

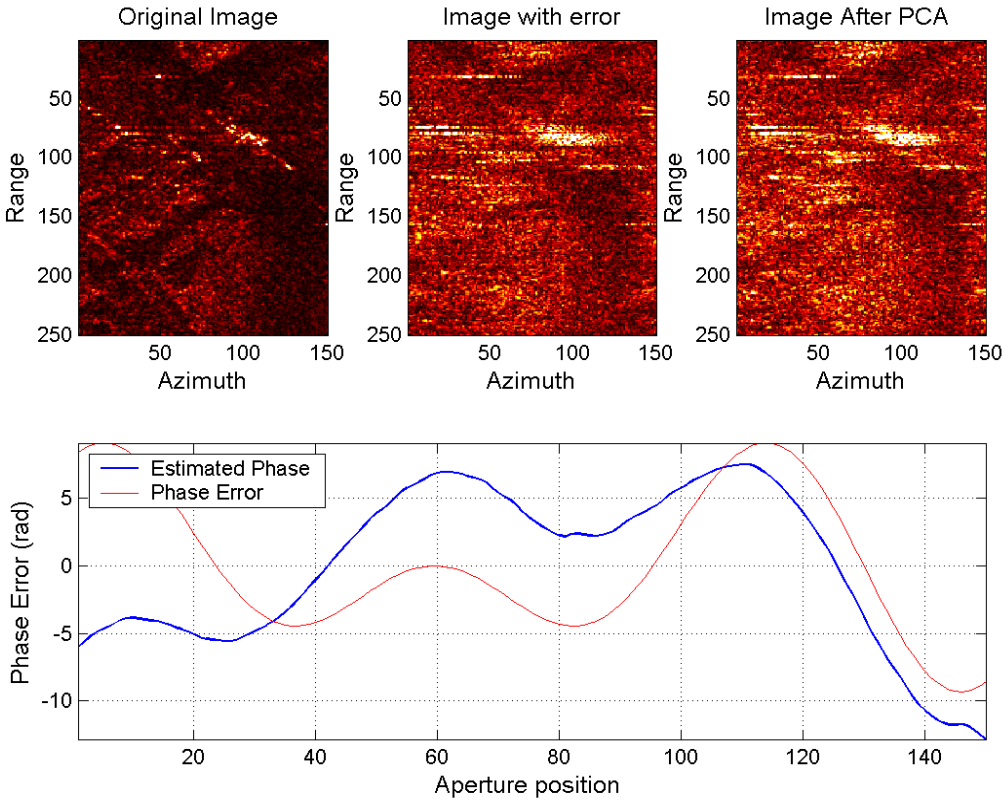


Figure 4.6 Stripmap image with random error by PCA (iteration = 0)

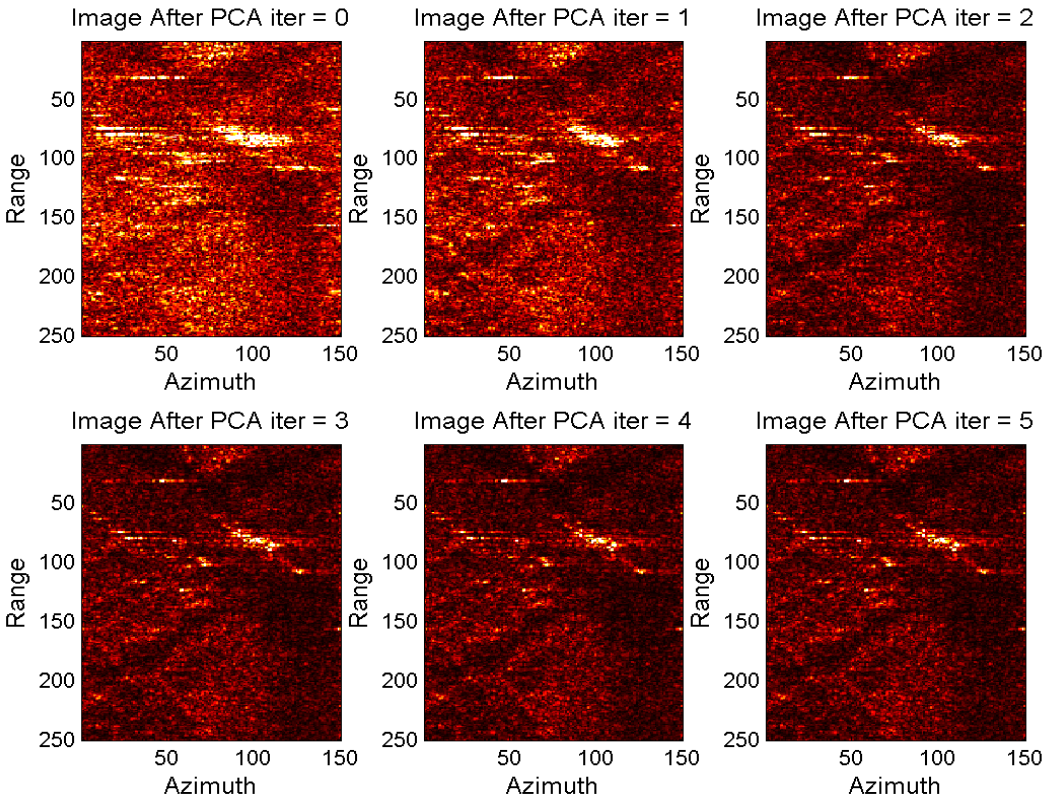


Figure 4.7 Corrected image (iteration = 0 ~ 5)

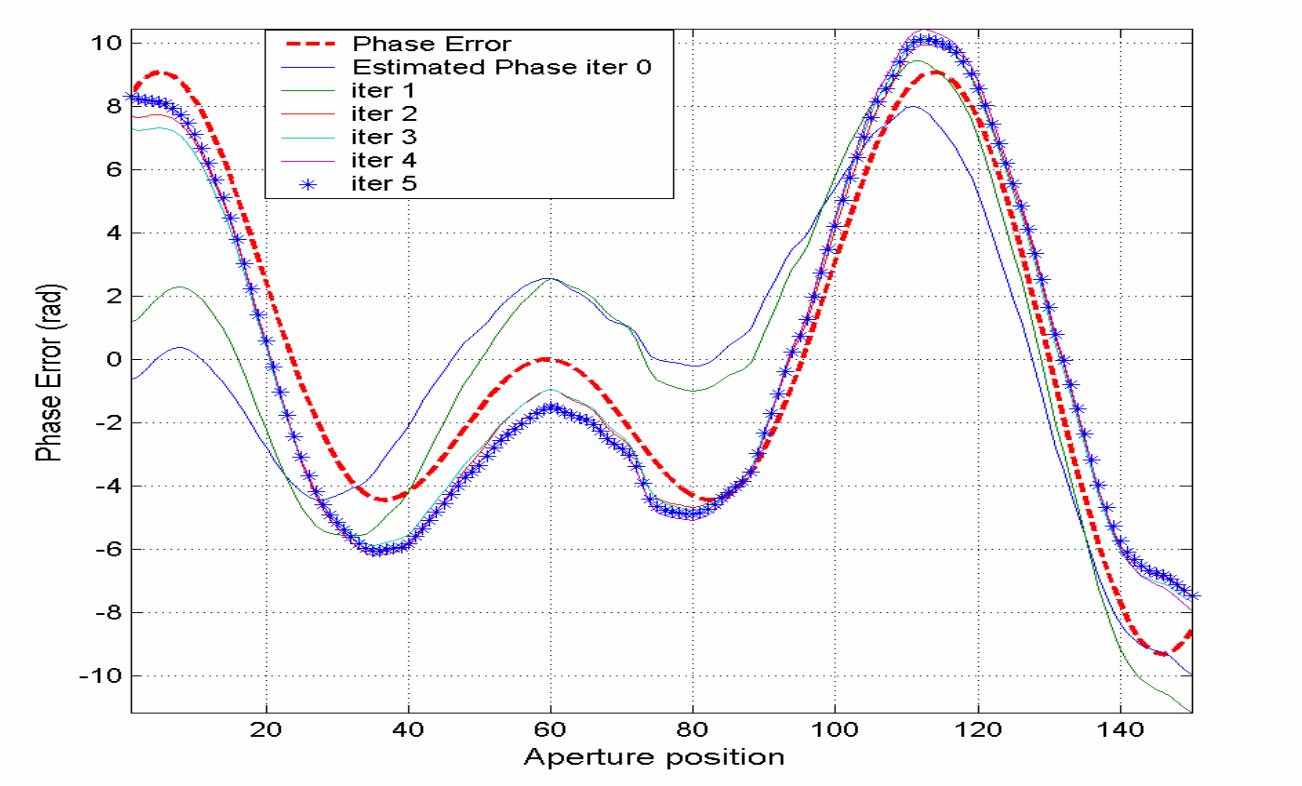


Figure 4.8 Estimated phase (iteration = 0 ~ 5)

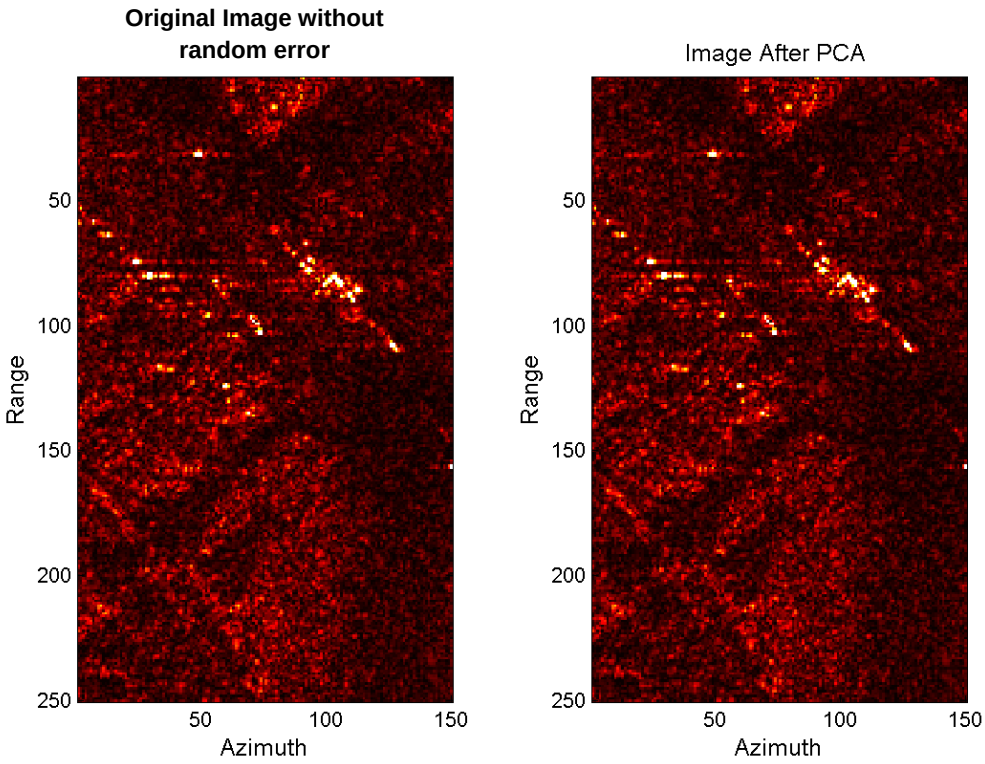


Figure 4.9 Original image without random error via PCA (iteration = 2)

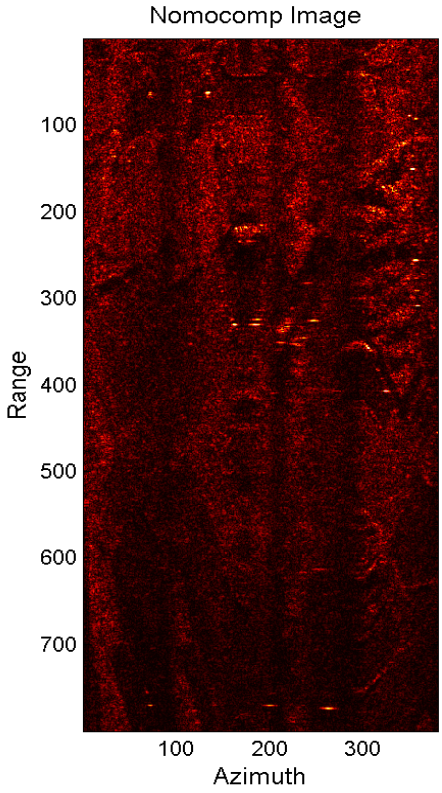


Figure 4.10 No mocomp image

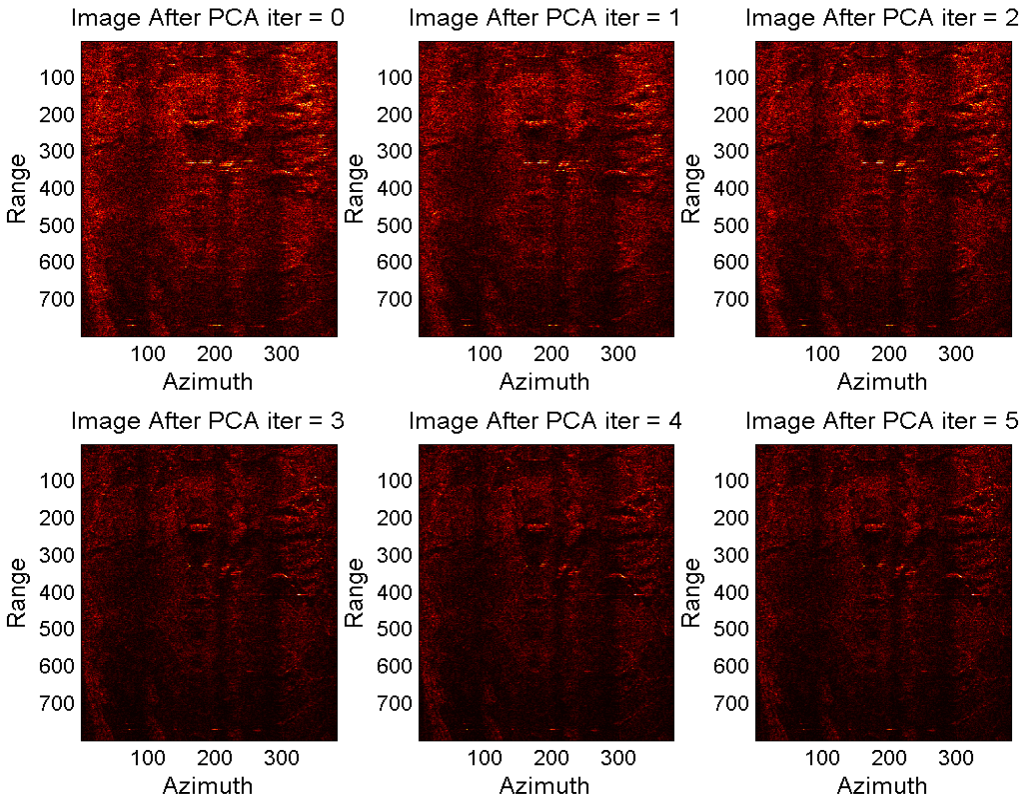


Figure 4.11 Image of the process of iteration 0 ~ 5

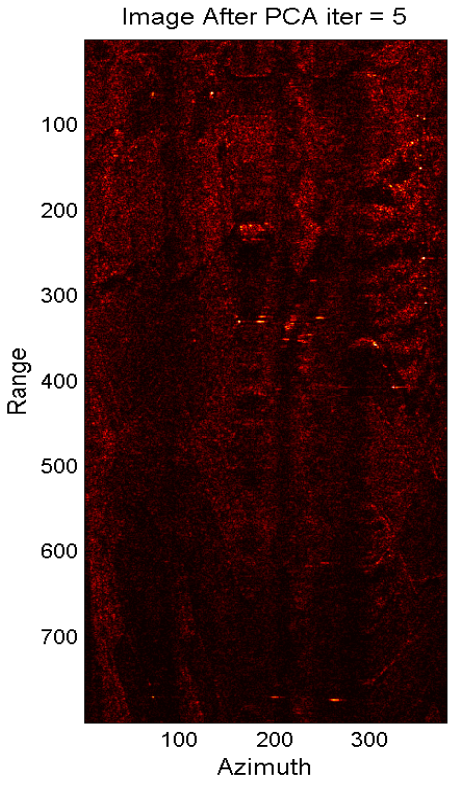


Figure 4.12 No mocomp image after five iterations of PCA

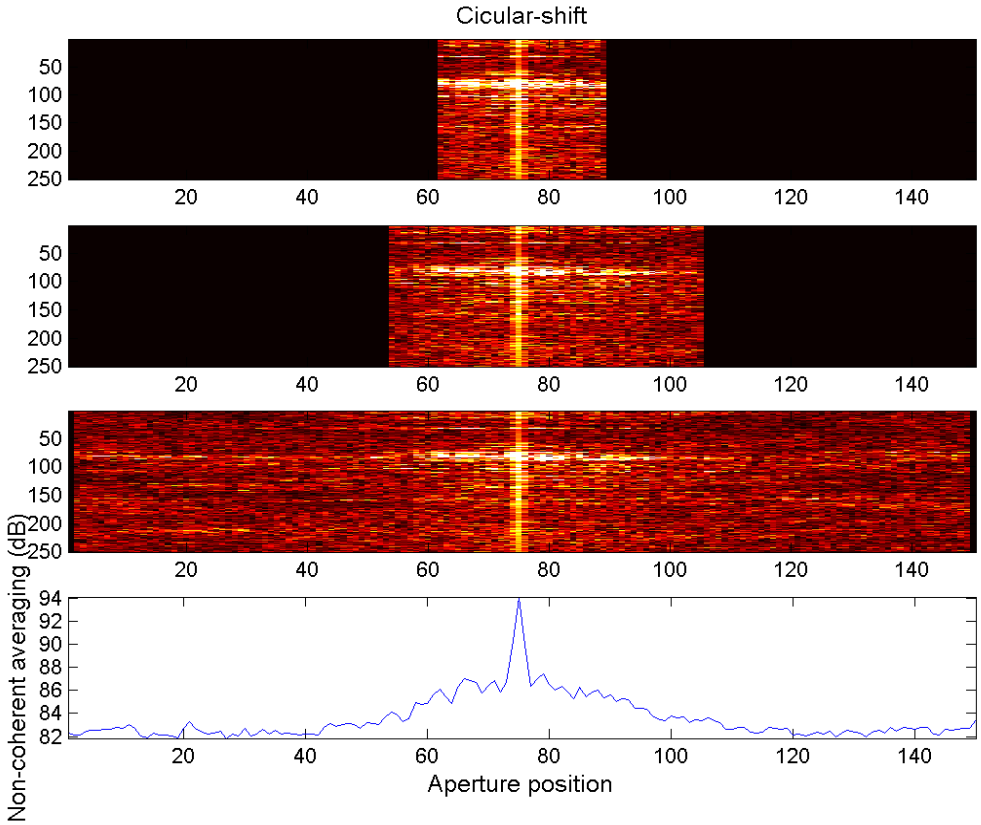
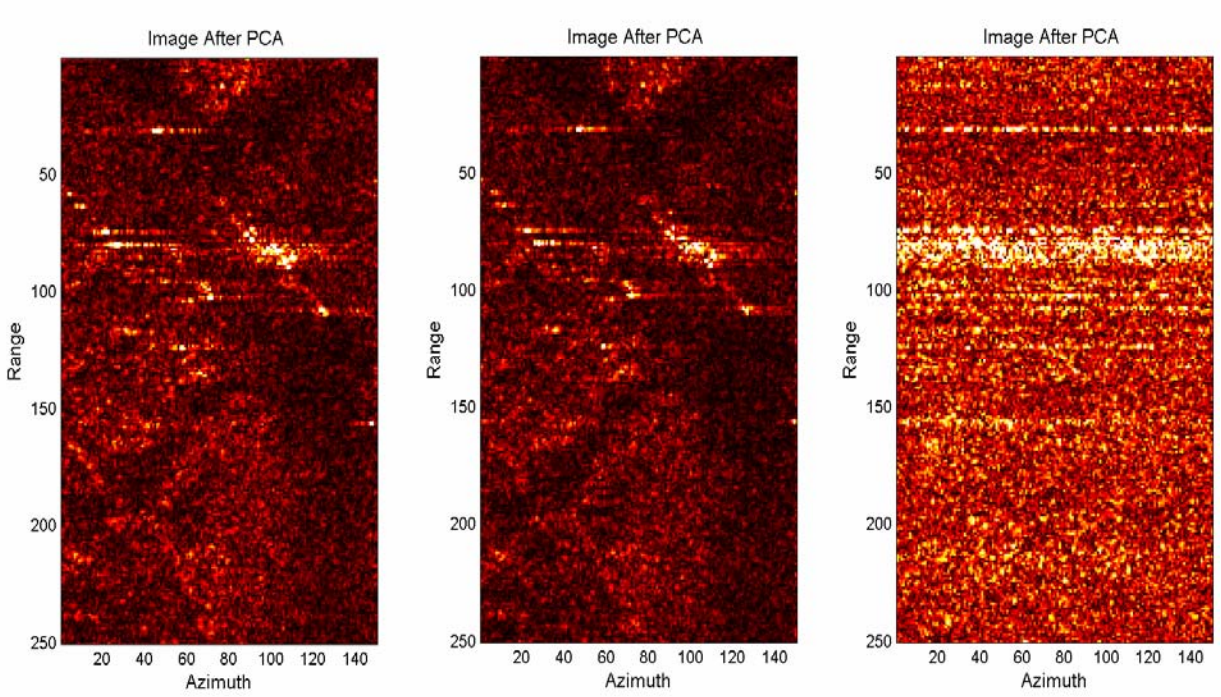


Figure 4.13 Three kinds of window size



(a) small window & iteration = 5

(b) medium window & iteration = 6

(c) large window & iteration = 5

Figure 4.14 Corrected image with three kinds of window size

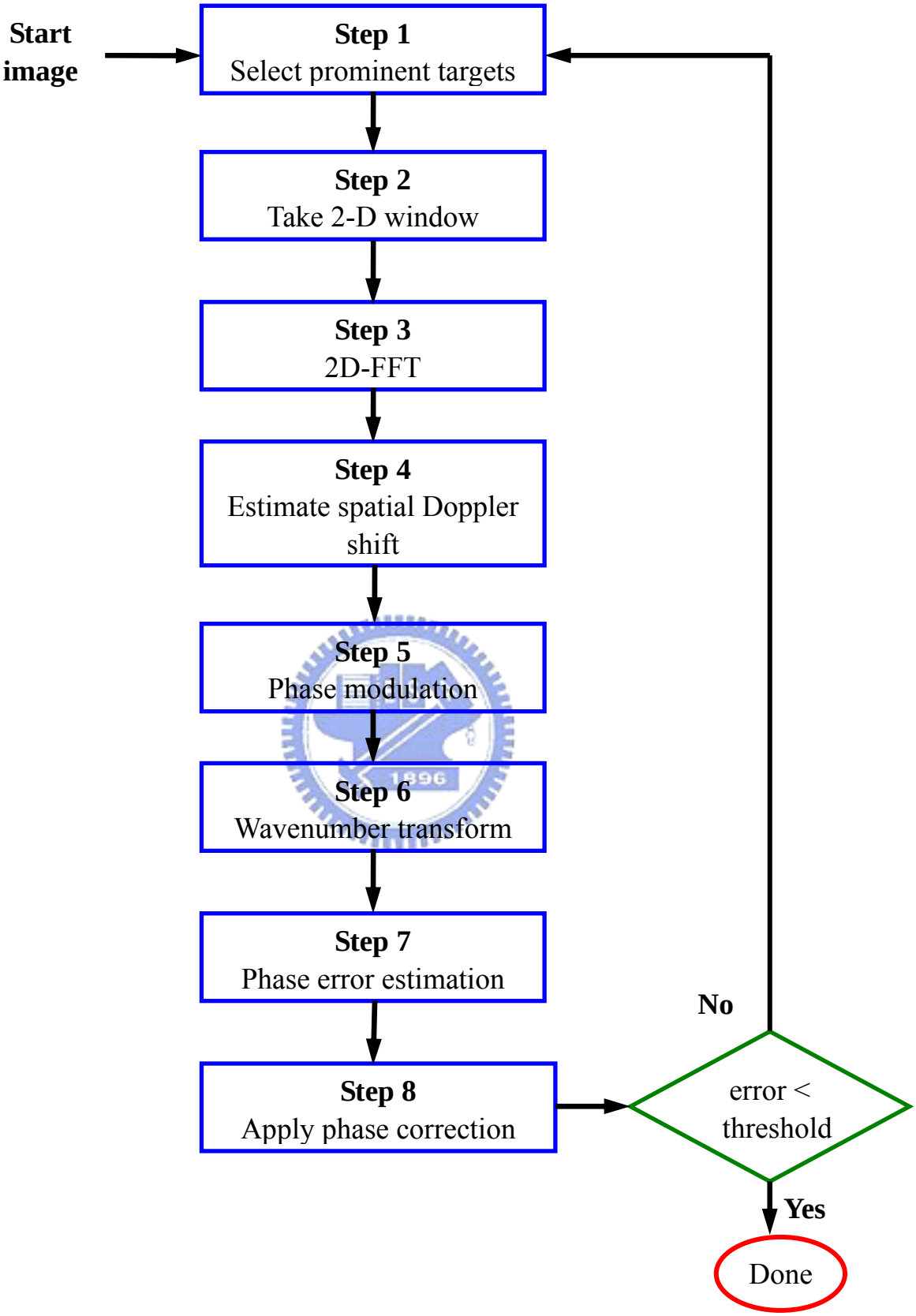


Figure 4.15 SPGA algorithmic steps

Tables

	Spotlight data	Stripmap data with error	Stripmap data without error	Stripmap data with no mocomp
PGA	O	X	X	X
PPP	O	O	X	Δ
PCA	O	O	Δ	O

O : better

Δ : similar

X : worse

Table 5.1 Algorithm implementation comparison