

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA



TESIS

**IMPLEMENTACIÓN Y COMPARACIÓN DE TRES
ALGORITMOS DE FORMACIÓN DE IMÁGENES SAR
EMPLEANDO UN RADAR DE APERTURA SINTÉTICA DE
ESTACIÓN TERRENA (GB-SAR) PARA UNA APLICACIÓN DE
MONITOREO DE DESLIZAMIENTOS**

**PARA OBTENER EL TÍTULO PROFESIONAL DE:
INGENIERO ELECTRÓNICO**

**ELABORADO POR:
LUIS DE LA CRUZ PAINADO**

**ASESOR:
ING. ANGEL LUIS ALBURQUEQUE GUERRERO**

**LIMA - PERÚ
2021**

**IMPLEMENTACIÓN Y COMPARACIÓN DE TRES
ALGORITMOS DE FORMACIÓN DE IMÁGENES SAR
EMPLEANDO UN RADAR DE APERTURA SINTÉTICA DE
ESTACIÓN TERRENA (GB-SAR) PARA UNA APLICACIÓN DE
MONITOREO DE DESLIZAMIENTOS**

DEDICATORIA

A mis padres Pepe y Carmen, por el cariño inmenso, por estar presentes conmigo en los buenos y difíciles momentos de mi vida, y por apoyarme en el logro de cada uno de mis metas, en especial este gran logro profesional. Les estaré agradecidos toda una vida.

A mis hermanas Helen, Vianney y Noemí, por ser mis guías y ejemplos a seguir, por la motivación y apoyo constante en cada meta trazada.

AGRADECIMIENTOS

Agradezco al Instituto Geofísico del Perú (IGP), sede Radio Observatorio de Jicamarca (ROJ) por todas las facilidades brindadas, al Dr. Marco Milla por el asesoramiento durante el desarrollo de este trabajo, al personal del área CIELO por los consejos y orientaciones, y al Fondo Nacional de Desarrollo Científico y Tecnológico (FONDECYT) por la financiación al proyecto bajo el convenio de adjudicación No. 099-2014-FONDECYT-DE.

RESUMEN

En el presente trabajo se describe la implementación y comparación de tres algoritmos de formación de imágenes para radares de Apertura Sintética de Estación Terrena (GB-SAR), con el objetivo principal de determinar el algoritmo más adecuado para aplicarlo en la etapa inicial de un Sistema de monitoreo de deslizamientos con radares SAR.

Los algoritmos que se implementaron fueron los siguientes: Frequency Domain Back Projection (FDBP), Range Migration Algorithm (RMA) y Discrete Linear Inverse Problem (DLIP). Cada uno de estos algoritmos se probaron con datos simulados, mientras que solo los dos primeros se probaron con datos reales. Las pruebas con datos simulados consistieron en la obtención de las imágenes SAR de blancos teóricos, estos blancos fueron definidos por el usuario y los datos fueron generados por un programa de computador. Por otro lado, las pruebas con datos reales consistieron en la obtención de las imágenes SAR de un blanco real. El blanco real usado fue un cerro propenso a deslizamiento, y los datos fueron adquiridos con el radar GB-SAR 'IGP-ROJ', perteneciente al Instituto Geofísico del Perú (IGP).

En el análisis comparativo se consideraron los siguientes tres parámetros de comparación: calidad de reconstrucción de la imagen, calidad de enfoque y coste computacional. Además se analizó cómo afectan los resultados de cada algoritmo en la etapa final del Sistema de monitoreo de deslizamientos con radares SAR, que es la generación de los mapas y gráficos de deslizamientos del blanco real.

De los resultados de comparación usando tanto datos obtenidos de la simulación como datos reales, se concluyó que el algoritmo FDBP es el más adecuado para una aplicación de monitoreo de deslizamientos.

ABSTRACT

In this work, three imaging algorithms using Ground-Based Synthetic Aperture Radar (GB-SAR) were implemented and then compared, in order to select the most suitable one, and to apply it to the first stage of a SAR Landslide Monitoring System application.

The Imaging Algorithms are: “Frequency Domain Back Projection” (FDBP), “Range Migration Algorithm” (RMA), and “Discrete Linear Inverse Problem” (DLIP). These algorithms were tested with simulated data, but only the former two were tested with real data. The simulated data was obtained by the means of a program that generated a matrix of values, accordingly to a user defined scenario. On the other hand, the real data measurements were carried out using the ‘IGP-ROJ’ GB-SAR radar (developed by Instituto Geofísico del Perú - IGP), which was focused on a part of a mountain, where a landslide has occurred.

These three imaging algorithms were compared in terms of image reconstruction quality, focusing quality and computational burden. Besides, an analysis was made to figure out the effect of the algorithms on the generation of the landslides graphs and maps, which are the results of the last stage of the landslide monitoring system.

Results with both, simulated and real data, showed that the FDBP Algorithm is most appropriate for the mentioned application.

ÍNDICE

PRÓLOGO	1
CAPÍTULO I	
INTRODUCCIÓN	3
1.1 Panorama de los deslizamientos en el Perú	3
1.2 Antecedentes del monitoreo de deslizamientos con radares SAR	4
1.3 Problemática	5
1.4 Estado del arte	6
1.5 Objetivos	7
1.5.1 Objetivo General	7
1.5.2 Objetivos Específicos	7
CAPÍTULO II	
MARCO TEÓRICO	8
2.1 Fundamentos de radares	8
2.1.1 Ecuación de radar	8
2.1.2 Formas de onda de las señales de transmisión y recepción	10
2.2 Fundamentos de radares GB-SAR	11
2.2.1 Geometría	12
2.2.2 Resolución	13
2.2.3 Conceptos básicos de formación de imágenes	15
2.3 Conceptos de PDS usados en formación de imágenes SAR	16
2.3.1 Transformada de Fourier	16
2.3.2 Interpolación	27
2.3.3 Decimación	28
2.3.4 Filtros	30
2.4 Algoritmos de formación de imágenes SAR	32
2.4.1 Revisión del algoritmo FDBP	33
2.4.2 Revisión del algoritmo RMA	35
2.4.3 Revisión del algoritmo DLIP	39
2.4.4 Criterios de comparación de algoritmos	42

2.5	Lenguajes de programación para implementación de algoritmos	46
CAPÍTULO III		
DESCRIPCIÓN DEL SISTEMA DE ADQUISICIÓN DE DATOS		48
3.1	Radar GB-SAR ‘IGP-ROJ’	48
3.1.1	Descripción del hardware y software	49
3.1.2	Funcionamiento	50
3.1.3	Adquisición de los datos	51
3.2	Criterios de selección de parámetros	53
3.3	Geometría	55
3.4	Simulación de datos	56
CAPÍTULO IV		
IMPLEMENTACIÓN DE ALGORITMOS DE FORMACIÓN DE IMÁGENES		58
4.1	Implementación del algoritmo FDBP	58
4.1.1	Pasos	59
4.1.2	Resultados	64
4.1.3	Mejoras a la imagen	66
4.1.4	Resumen de los pasos	68
4.2	Implementación del algoritmo RMA	69
4.2.1	Pasos	70
4.2.2	Resultados	82
4.2.3	Mejoras a la imagen	82
4.2.4	Resumen de los pasos	87
4.3	Implementación del algoritmo DLIP	89
4.3.1	Planteamiento	89
4.3.2	Pasos	91
4.3.3	Resultados	93
4.3.4	Limitaciones de implementación	93
4.3.5	Resumen de los pasos	95
CAPÍTULO V		
ANÁLISIS Y DISCUSIÓN DE RESULTADOS		96
5.1	Análisis de los resultados simulados	96
5.1.1	Algoritmo FDBP	97
5.1.2	Algoritmo RMA	100
5.1.3	Algoritmo DLIP	104

5.2	Comparación de algoritmos	105
5.2.1	Calidad de reconstrucción de la imagen	105
5.2.2	Calidad de enfoque	114
5.2.3	Coste computacional	117
5.3	Análisis de los resultados experimentales	121
5.3.1	Análisis de las imágenes SAR	121
5.3.2	Análisis de los mapas de desplazamientos	123
5.4	Resumen	126
	CONCLUSIONES	128
	RECOMENDACIONES	130
	BIBLIOGRAFÍA	131
	ANEXOS	134
A	Pseudocódigos para la implementación de los algoritmos FDBP, RMA y DLIP .	135
B	Demostración del paso compresión en rango del algoritmo FDBP	141

ÍNDICE DE FIGURAS

Figura 1.1:	Etapas principales de la técnica GB-DInSAR: (a) diagrama de bloques, (b) esquema gráfico. (Fuente propia)	5
Figura 2.1:	Geometría usada para la obtención de la ecuación del radar. (Adaptado de [8])	9
Figura 2.2:	Señal SFCW en el plano Tiempo-Frecuencia. (Adaptado de [8])	10
Figura 2.3:	Modo Stripmap del radar GB-SAR.(Adaptado de [8])	11
Figura 2.4:	Geometría usada para la formación de imágenes SAR. (Fuente propia)	12
Figura 2.5:	Resolución en rango y azimut del radar GB-SAR. (Adaptado de [14])	13
Figura 2.6:	Representación gráfica del perfil de rango y azimut para un blanco puntual. (Fuente propia)	16
Figura 2.7:	Perfil de Rango (abajo), después de una IDFT a una fila del histórico de fase (arriba) presente en el dominio frecuencial. (Fuente propia)	19
Figura 2.8:	Ejemplo de la periodicidad de la DFT tanto en el dominio del espacio (imagen superior) como de la frecuencia (imagen inferior). (Fuente propia)	21
Figura 2.9:	Ejemplo de la propiedad de desplazamiento circular de la DFT. En este caso no se presenta desplazamiento. (a): Espectro, (b) Perfil de Rango en magnitud, (c) Perfil de Rango en fase. (Fuente propia)	23
Figura 2.10:	Ejemplo de la propiedad de desplazamiento circular de la DFT. En este caso sí se presenta desplazamiento con $m = 3$. (a) Espectro, (b) perfil de Rango en magnitud, (c) perfil de Rango en fase. (Fuente propia)	24
Figura 2.11:	Ejemplo de <i>zero padding</i> . (a) Espectro con relleno de ceros, (b) Perfil de rango mejor definido. (Fuente propia)	25
Figura 2.12:	Perfil de rango del espectro con un número de muestras L , (a) sin <i>zero-padding</i> , (b) con <i>zero-padding</i> . Perfil de rango del espectro con un número de datos $L' > L$, (c) sin <i>zero-padding</i> , (d) con <i>zero-padding</i> . (Fuente propia)	26

Figura 2.13: Interpolación constante por partes, aplicado a un perfil de rango, en sus 3 variantes: (a) <i>Previous</i> , (b) <i>Next</i> , (c) <i>Nearest</i> . (Fuente propia)	27
Figura 2.14: (a)Interpolación lineal e (b) interpolación polinomial cuadrática, aplicado a un perfil de rango. (Fuente propia)	29
Figura 2.15: Decimación del perfil de rango. (a) Espectro sin decimar, (b) Perfil de rango sin decimar, (c) Espectro decimado, (d) Perfil de rango decimado. (Fuente propia)	29
Figura 2.16: Lóbulos laterales presentes en un perfil de rango. (Fuente propia)	31
Figura 2.17: Funciones de enventanado junto a sus espectros. (a)Ventana de Hanning y (b)Ventana de Hamming. (Fuente propia)	32
Figura 2.18: Perfil de rango empleando: (a) Ventana Hanning, (b) ventana Hamming. (Fuente propia)	32
Figura 2.19: Esquema gráfico de la etapa de etapa de interpolación del algoritmo FDBP. (Extraído de [19])	35
Figura 2.20: Reconstrucción de un blanco puntual simulado usando el algoritmo RMA. (Extraído de [21])	38
Figura 2.21: Reconstrucción de 3 blancos puntuales simulados usando el algoritmo RMA. (Extraído de [21])	38
Figura 2.22: Reconstrucción de un blanco real usando el algoritmo RMA. Izquierda: blanco, Derecha: imagen SAR reconstruida. (Extraído de [20])	39
Figura 2.23: Geometría asumida para explicar el procedimiento del algoritmo <i>SAR migration</i> . (Extraído de [22])	40
Figura 2.24: Parámetros de un perfil de rango/azimut para el caso de un blanco puntual simulado: resolución Δr , PSLR e ISLR. (Adaptado de [27])	43
Figura 2.25: Procedimiento para obtener las gráficas de ‘Desplazamiento vs Tiempo’, cada gráfica se obtiene respecto a una cota en específico (P1). (Fuente propia)	45
Figura 3.1: Diagrama de flujo del sistema escogido de formación de imágenes con radares SAR. (Fuente propia)	48
Figura 3.2: Elementos mecánicos y electrónicos del radar GB-SAR ‘IGP-ROJ’: 1) antenas de bocina, 2) analizador vectorial de redes (VNA) y 3) sistema de posicionamiento lineal. (Fuente propia)	49
Figura 3.3: Representación gráfica del histórico de fase. (Fuente propia).	52

Figura 3.4:	Contenido de un archivo HDF5 descargado del radar GB-SAR, en el cual se puede observar tanto el histórico de fase como los parámetros del experimento. (Fuente propia)	52
Figura 3.5:	Ángulo de visión máximo de la imagen SAR ' θ_{max} '. La imagen formada en la zona gris estará libre del efecto de solapamiento. (Fuente propia)	54
Figura 3.6:	Geometría detallada usada para la formación de imágenes SAR, mostrando la ubicación de la apertura sintética y la escena de la imagen. (Fuente propia)	55
Figura 3.7:	Histórico de fase simulado en magnitud y fase para el ejemplo de los tres blancos puntuales. (Fuente propia)	57
Figura 4.1:	(a) Blanco usado en las pruebas de los algoritmos de formación de imágenes, (b) histórico de fase simulado respectivo. (Fuente propia)	59
Figura 4.2:	Pasos propuestos del algoritmo FDBP. (Fuente propia)	60
Figura 4.3:	Etapa de compresión en rango del algoritmo FDBP para el caso de los 3 blancos puntuales. Se extrae un perfil de rango de la matriz resultante correspondiente a una posición intermedia de la apertura sintética. (Fuente propia)	61
Figura 4.4:	Representación del perfil de Rango en el plano de la escena de la imagen. (Fuente propia)	61
Figura 4.5:	Proceso de interpolación en el algoritmo FDBP. (Fuente propia)	62
Figura 4.6:	Etapa de interpolación del algoritmo FDBP para el caso de 3 blancos puntuales. Se muestran las imágenes SAR en magnitud, correspondientes a las siguientes posiciones en el riel: (a) inicial, (b) intermedia y (c) final. (Fuente propia)	63
Figura 4.7:	Etapa de suma coherente para el caso de los 3 blancos puntuales. En la parte inferior se muestra la imagen SAR resultante.	64
Figura 4.8:	Imagen SAR del ejemplo de los 3 blancos puntuales, obtenida con el algoritmo FDBP. (Fuente propia)	64
Figura 4.9:	Imagen SAR ampliada en magnitud del ejemplo de los 3 blancos reconstruidos usando el algoritmo FDBP. (Fuente propia)	65
Figura 4.10:	Imagen SAR de un blanco puntual en $(0,5)m$, obtenida empleando el algoritmo FDBP y usando 4 interpoladores: (a) constante, (b) lineal, (c) cuadrático y (d) cúbico. (Fuente propia)	66

Figura 4.11: Reconstrucción de un blanco puntual aplicando el algoritmo FDBP, y usando 2 funciones de enventanado: (a) blanco teórico, (b) sin ventana, (c) con una ventana Hanning, (d) con una ventana Hamming. (Fuente propia)	68
Figura 4.12: Comparación de perfiles de rango empleando las ventanas de Hanning y Hamming, para el caso de un blanco puntual reconstruido con el algoritmo FDBP. (Fuente propia)	69
Figura 4.13: Etapas del algoritmo RMA. (Fuente propia)	70
Figura 4.14: Etapa de compresión en rango del algoritmo RMA, para el caso de 2 grupos de blancos puntuales	71
Figura 4.15: Perfil de rango para 3 blancos puntuales obtenidas en la etapa de compresión en rango del algoritmo RMA.	71
Figura 4.16: Proceso de <i>zero-padding</i> en el paso de 1D-DFT del algoritmo RMA. (Fuente propia)	73
Figura 4.17: Histórico de fase desplazado en $X_c = L_s/2$ hacia la derecha, y el <i>zero-padding</i> respectivo. (Fuente propia)	73
Figura 4.18: Etapa de 1D-DFT (azimut) del algoritmo RMA, aplicado al ejemplo de los 3 blancos puntuales. Se muestran las imágenes resultantes en magnitud y fase. (Fuente propia)	74
Figura 4.19: Distancia del centro de la escena de la imagen hacia el eje de Azimut. (Fuente propia)	75
Figura 4.20: Etapa de filtro adaptado del algoritmo RMA, aplicado al ejemplo de los 3 blancos puntuales. Se muestran las imágenes resultantes en magnitud y fase. (Fuente propia)	76
Figura 4.21: Desplazamiento de la escena de la imagen en el eje del Rango por un valor R_s , y definición de un nuevo plano $x' - y'$. (Basado en [17])	76
Figura 4.22: Efecto de corrección de curvatura realizado por la etapa de filtro adaptado del algoritmo RMA. (Basado en [17])	77
Figura 4.23: Corrección de curvatura para el ejemplo de los 3 blancos puntuales dada por la etapa de filtro adaptado del algoritmo RMA. (a) Antes de aplicar el filtro adaptado, (b) Después de aplicar el filtro adaptado. (Fuente propia)	77

Figura 4.24: Corrección de las curvaturas residuales realizada por la etapa de Interpolación STOLT del algoritmo RMA. (Basado en [17])	78
Figura 4.25: Cambio de coordenadas realizado en la etapa de Interpolación STOLT del algoritmo RMA. (Basado en [17])	78
Figura 4.26: Proceso de interpolación 1D a lo largo del eje k_y realizado en la etapa de Interpolación STOLT del algoritmo RMA. (Basado en [17])	79
Figura 4.27: Etapa de Interpolación STOLT del algoritmo RMA, aplicado al ejemplo de los 3 blancos puntuales. Se muestran las imágenes resultantes en magnitud y fase. (Fuente propia)	80
Figura 4.28: Corrección de la curvatura para el ejemplo de los 3 blancos puntuales después de efectuar la Interpolación STOLT. (Fuente propia)	81
Figura 4.29: Imagen SAR del ejemplo de los 3 blancos puntuales obtenida con el algoritmo RMA. (Fuente propia)	82
Figura 4.30: Imagen SAR ampliada en magnitud del ejemplo de los 3 blancos reconstruidos usando el algoritmo RMA. (Fuente propia)	83
Figura 4.31: Efecto de la función de enventanado en la reconstrucción de un blanco puntual empleando el algoritmo RMA: (a)Blanco reconstruido sin ventana, (b)Blanco reconstruido con una ventana Hamming. (Fuente propia)	83
Figura 4.32: Ejemplo de decimación en el dominio de la frecuencia con $D_x = 4$ y $D_y = 2$. (a) Espectro de la señal original, (b) espectro decimado. (Fuente propia)	85
Figura 4.33: Imagen SAR en magnitud (lado izquierdo) junto a su espectro (lado derecho) de un blanco puntual aplicando el algoritmo RMA, para los siguientes casos: (a,b) sin decimación; (c,d) decimación $(D_x, D_y)=(2,2)$ y (e,f) decimación $(D_x, D_y)=(5,5)$. (Fuente propia)	86
Figura 4.34: Recorte de espectro realizado después de la etapa de 1D-DFT del algoritmo RMA, para el ejemplo de los 3 blancos puntuales. (a) Espectro sin recortar, (b) máscara de recorte de espectro. (Fuente propia)	88
Figura 4.35: Pasos del algoritmo DLIP. (Fuente propia)	91
Figura 4.36: Imagen SAR del ejemplo de los 3 blancos puntuales obtenida empleando el algoritmo DLIP. (Fuente propia)	94
Figura 5.1: Blanco usado para el análisis de los algoritmos de formación de imágenes SAR. (Fuente propia)	96

Figura 5.2:	Imagen SAR de un blanco pseudo-continuo obtenido con el algoritmo FDBP. Se muestra la magnitud y fase de la imagen. (Fuente propia)	98
Figura 5.3:	Blanco ubicado dentro de la zona de interpolación. (Fuente propia)	99
Figura 5.4:	Imagen SAR de un blanco dentro de la zona de interpolación obtenido con el algoritmo FDBP. Se muestra la imagen en magnitud y fase. (Fuente propia)	99
Figura 5.5:	Imagen SAR de un blanco pseudo-continuo obtenido con el algoritmo RMA. Se muestra la magnitud y fase de la imagen. (Fuente propia)	100
Figura 5.6:	Blanco usado para comprobar la presencia de una zona de reconstrucción en el algoritmo RMA. (Fuente propia)	101
Figura 5.7:	Imagen SAR de un blanco pseudo-continuo ubicado en la zona de no reconstrucción, obtenido con el algoritmo RMA. Se muestra la imagen en magnitud y fase. (Fuente propia)	101
Figura 5.8:	Imagen SAR de un blanco pseudo-continuo reconstruido con el algoritmo FDBP, incrementando el valor de L_s . Se muestra el resultado en magnitud y fase. (Fuente propia)	102
Figura 5.9:	Zonas de reconstrucción del algoritmo RMA. Las zonas oscuras de la escena de la imagen indican las zonas de no reconstrucción, mientras que las zonas claras son las zonas de reconstrucción. (Fuente propia)	102
Figura 5.10:	Imagen SAR de un blanco que cumple las recomendaciones del algoritmo RMA. Se muestra la imagen en magnitud y fase. (Fuente propia)	103
Figura 5.11:	Efecto de la apertura sintética L_s en la fase de la imagen SAR obtenida con el algoritmo RMA. Reconstrucción de la imagen con: (a) $L_s = 1.2m$, (b) $L_s = 4m$. (Fuente propia)	104
Figura 5.12:	Imagen SAR obtenida con el algoritmo DLIP en magnitud y fase. (Fuente propia)	105
Figura 5.13:	(a) Imagen teórica en magnitud de un blanco puntual. Imagen SAR en magnitud del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP.	107
Figura 5.14:	(a) Imagen teórica en fase de un blanco puntual. Imagen SAR en fase del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP.	108

Figura 5.15: (a) Imagen teórica en magnitud de un blanco pseudo-continuo en forma de “T”. Imagen SAR en magnitud del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)	110
Figura 5.16: (a) Imagen teórica en fase de un blanco pseudo-continuo en forma de “T”. Imagen SAR en fase del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)	111
Figura 5.17: (a) Imagen teórica en magnitud de un blanco pseudo-continuo de la palabra “UNI”. Imagen SAR en magnitud del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)	112
Figura 5.18: (a) Imagen teórica en fase de un blanco pseudo-continuo de la palabra “UNI”. Imagen SAR en fase del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)	113
Figura 5.19: Perfiles de Rango obtenidos con 3 algoritmos distintos de un blanco puntual ubicado en la posición (<i>Azimut, Rango</i>): (a)(0,5) <i>m</i> , (b)(0,2) <i>m</i> , (c)(0,7) <i>m</i> , (d)(0,9) <i>m</i> . (Fuente propia)	115
Figura 5.20: Perfiles de Azimut obtenidos con 3 algoritmos distintos de un blanco puntual ubicado en la posición (<i>Azimut, Rango</i>): (a)(0,5) <i>m</i> , (b)(0,2) <i>m</i> , (c)(0,7) <i>m</i> , (d)(0,9) <i>m</i> . (Fuente propia)	117
Figura 5.21: Cerro Pucruyacra, vista aérea. Se muestra el blanco y la ubicación aproximada del radar GB-SAR. (Fuente: Google Maps)	122
Figura 5.22: Imágenes SAR en magnitud de un blanco real, obtenido con los algoritmos: (a) FDBP y (b) RMA. (Fuente propia)	123
Figura 5.23: Mapa de desplazamientos (interferograma) del blanco real generados con dos imágenes SAR, a su vez obtenidas con los algoritmos: (a) FDBP y (b) RMA. (Fuente propia)	124
Figura 5.24: Ubicación de los 4 puntos de prueba en el mapa de desplazamientos. (Fuente propia)	125
Figura 5.25: Gráficas de deslizamientos para 4 puntos de prueba: (a) P1, (b) P2, (c) P3 y (d) P4. (Fuente propia)	125

ÍNDICE DE TABLAS

Tabla 2.1:	Características de las funciones de inventanado. [18]	31
Tabla 2.2:	Parámetros usados en las pruebas.	33
Tabla 2.3:	Parámetros que fueron usados en las pruebas con el algoritmo RMA	36
Tabla 3.1:	Principales parámetros eléctricos y mecánicos de un sistema de formación de imágenes con radares SAR.	55
Tabla 3.2:	Parámetros de la escena de la imagen SAR.	56
Tabla 3.3:	Parámetros del sistema usados en la simulación de datos.	56
Tabla 4.1:	Parámetros eléctricos y mecánicos usados en la explicación de los pasos del algoritmo FDBP.	58
Tabla 4.2:	Tiempos de procesamiento del algoritmo FDBP usando 4 interpoladores.	67
Tabla 4.3:	Obtención de la resolución y máximo SSL de los perfiles de rango empleando las ventanas de Hanning y Hamming, para el caso de un blanco puntual reconstruido usando el algoritmo FDBP.	69
Tabla 5.1:	Parámetros geométricos del blanco pseudo-continuo.	97
Tabla 5.2:	Parámetros empleados para la simulación e implementación de los algoritmos de formación de imágenes SAR.	97
Tabla 5.3:	Parámetros de salida de la reconstrucción del blanco pseudo-continuo realizado con el algoritmo FDBP	98
Tabla 5.4:	Parámetros de salida de la reconstrucción del blanco pseudo-continuo realizado con el algoritmo RMA	100
Tabla 5.5:	Parámetros de salida de la reconstrucción del blanco pseudo-continuo realizado con el algoritmo DLIP	104
Tabla 5.6:	Parámetros eléctricos y mecánicos empleados en la comparación de algoritmos de formación de imágenes SAR.	106
Tabla 5.7:	Parámetros geométricos usados para la comparación de algoritmos de formación de imágenes SAR.	106
Tabla 5.8:	Errores de magnitud y fase de las imágenes SAR de un blanco puntual, usando 3 algoritmos: FDBP, RMA, DLIP.	109

Tabla 5.9: Errores de magnitud y fase de las imágenes SAR de un blanco pseudo-continuo en forma de “T”, usando 3 algoritmos: FDBP, RMA, DLIP.	111
Tabla 5.10: Errores de magnitud y fase de las imágenes SAR de un blanco pseudo-continuo de la palabra “UNI”, usando 3 algoritmos: FDBP, RMA, DLIP.	114
Tabla 5.11: Posiciones de los blancos usados en las pruebas de PSLR.	115
Tabla 5.12: Valores de PSLR (en dB) a lo largo del eje de Rango para 4 blancos puntuales empleando los algoritmos FDBP, RMA y DLIP.	116
Tabla 5.13: Valores de PSLR (en dB) a lo largo del eje de Azimut para 4 blancos puntuales empleando los algoritmos FDBP, RMA y DLIP.	117
Tabla 5.14: Complejidad computacional aproximada del algoritmo FDBP	118
Tabla 5.15: Complejidad computacional aproximada del algoritmo RMA	119
Tabla 5.16: Complejidad computacional aproximada del algoritmo DLIP	120
Tabla 5.17: Complejidad computacional de los algoritmos implementados	120
Tabla 5.18: Tiempo de procesamiento para 3 casos de blancos simulados: 1 blanco puntual y 2 blancos pseudo-continuos, calculado para los algoritmos FDBP, RMA y DLIP.	121
Tabla 5.19: Parámetros eléctricos, mecánicos y geométricos usados en las pruebas con datos reales.	122
Tabla 5.20: Tiempo de procesamiento de los algoritmos FDBP y RMA para un blanco real.	123
Tabla 5.21: Desplazamientos promedios de los 4 puntos de prueba en función a los algoritmos FDBP y RMA. Todos los valores se muestran en (mm). . .	126
Tabla 5.22: Cuadro comparativo de los tres algoritmos implementados: FDBP, RMA y DLIP.	127

PRÓLOGO

El propósito del presente trabajo es determinar el algoritmo de formación de imágenes más adecuado para una aplicación de monitoreo de deslizamientos con radares GB-SAR. Para ello en primer lugar se implementan tres algoritmos, los cuales son Frequency Domain Back Projection (FDBP), Range Migration Algorithm (RMA) y Discrete Linear Inverse Problem (DLIP). Como segundo paso, se realiza una comparación entre ellos bajo los siguientes criterios, los cuales son la calidad de reconstrucción de la imagen, la calidad de enfoque y el coste computacional.

Esta tesis espera demostrar que alguno de los algoritmos implementados cumpla los requerimientos para ser usado en el proyecto “Desarrollo de un radar de apertura sintética para monitoreo de deslizamientos”, realizado en el Instituto Geofísico del Perú (IGP).

La validación de los algoritmos implementados se realiza empleando datos obtenidos por simulación, y dos tipos de blancos: puntuales y pseudo-continuos. Estos datos fueron generados mediante un algoritmo implementado con el lenguaje de programación Python 3.

El trabajo se limita solo al desarrollo de la primera etapa de un Sistema de Monitoreo de Deslizamientos con radares GB-SAR, es decir la etapa de formación de imágenes SAR, sin embargo para las pruebas y análisis de los algoritmos con datos reales se usan directamente las ecuaciones de otras técnicas de procesamiento de imágenes SAR, sin presentar muchos detalles de ellas.

Esta tesis se organiza en cinco capítulos y dos anexos.

En el Capítulo I, se describe una breve introducción del uso de los radares GB-SAR para monitorear deslizamientos, luego se presenta la problemática, el estado del arte y finalmente el objetivo general y los objetivos específicos.

En el Capítulo II, se presenta un breve marco teórico en el cual se describen los conceptos más importantes para entender este trabajo, tales como: radares y radares GB-SAR, procesamiento digital de señales usados en formación de imágenes SAR, y una introducción al lenguaje de programación Python, el cual se usó para implementar los algoritmos. También se revisa la literatura relacionada a la implementación de cada uno de los tres algoritmos y a los criterios de comparación de algoritmos.

En el Capítulo III, se presenta la descripción del radar GB-SAR que se usará, la geometría,

y el proceso para obtener los datos simulados.

En el Capítulo IV, se describe el desarrollo de la implementación de los tres algoritmos de formación de imágenes SAR: FDBP, RMA y DLIP.

En el Capítulo V, se presenta el análisis de los resultados que consisten de las pruebas de cada algoritmo con datos simulados y reales, y la comparación entre ellos en términos de calidad de reconstrucción de la imagen, calidad de enfoque y coste computacional.

Luego se presentan las conclusiones, recomendaciones, los anexos y la bibliografía. Respecto al contenido de los Anexos, en el Anexo A se presentan los programas de cada algoritmo implementado, ellos se muestran en forma de pseudocódigo para el mejor entendimiento de los programas, además se incluye un enlace de un repositorio digital de todos los programas realizados. En el Anexo B se presenta la demostración de uno de los pasos del algoritmo FDBP.

Este trabajo forma parte del proyecto “Desarrollo de un radar de apertura sintética para monitoreo de deslizamientos”, realizado en el Instituto Geofísico del Perú (IGP) - Sede Radio Observatorio de Jicamarca (ROJ), el cual involucra la construcción de un radar SAR, el procesamiento de los datos y el análisis de los mismos para estimar mapas de desplazamientos de tierra.

CAPÍTULO I INTRODUCCIÓN

1.1 Panorama de los deslizamientos en el Perú

Los deslizamientos se caracterizan por la ruptura y desplazamiento de pequeñas o grandes masas de suelo, rocas, rellenos artificiales o combinaciones de estas. Debido a la morfología del territorio peruano, un 80% del territorio es altamente susceptible a los diferentes tipos de movimientos en masa, entre ellos los deslizamientos de tierra, esto de acuerdo al mapa de susceptibilidad del Instituto Geológico Minero y Metalúrgico (INGEMMET) [1]. Los lugares más susceptibles a deslizamientos de tierra, de acuerdo a este mapa, se encuentran cerca a la cordillera de los andes, debido a las altas pendientes que se encuentran en estos lugares. En el Perú este fenómeno natural es fuente de varios daños a la población, principalmente en la temporada de lluvias. De acuerdo al Instituto Nacional de Defensa Civil (INDECI), durante el periodo de los años 2003-2012 se registraron a nivel nacional un total de 172 500 personas afectadas solo por este fenómeno natural [2]. Como caso particular se presenta uno de los deslizamiento más recientes y de gran magnitud ocurrido en el distrito de Cuenca, Huancavelica en el año 2014, que afectó a 150 familias y dañó el 50% de la infraestructura física, además de obstruir el cauce del río Mantaro ocasionando inundaciones en su margen derecha, todo esto de acuerdo a un reporte del Instituto Geofísico del Perú (IGP) [3].

Las principales causas de los deslizamientos son las lluvias y los movimientos sísmicos. Las lluvias pueden ser intensas en corto tiempo, o en menor intensidad pero de larga duración, los cuales remojan el suelo y la roca provocando desplazamientos de tierra; estos se presentan principalmente entre los meses de setiembre y mayo en la sierra y selva del territorio Peruano.

La observación, investigación y conocimiento de los deslizamientos es la estrategia base planteada por el INDECI para la prevención y atención de este desastre natural. Es por ello que varias instituciones, como el INGEMMET vienen realizando estudios de deslizamientos en diversas partes del Perú, elaborando posteriormente informes técnicos (mapas de riesgo) los cuales sirven como base para la posterior toma de decisiones. El monitoreo de deslizamientos forma parte de estos estudios para poder analizar la evolución temporal de este fenómeno natural.

1.2 Antecedentes del monitoreo de deslizamientos con radares SAR

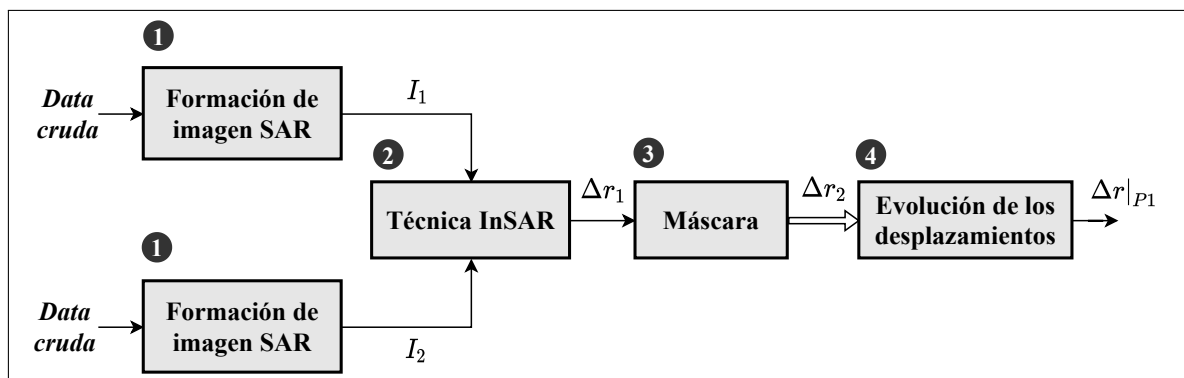
Las técnicas tradicionales para monitorear deslizamientos de superficies de tierra se caracterizan por proveer información precisa de un conjunto de puntos dentro del área de deslizamiento, ya sea usando técnicas de tipo geotécnico como extensómetros, inclinómetros, distanciómetros, etc; o de tipo topográfico como triangulación manual, estación total o mediciones con GPS [4]. Estas técnicas son efectivas midiendo información puntual de una zona de deslizamiento, pero se vuelven imprácticas si se quiere monitorear toda una zona continua, debido a la cantidad de instrumentos que se tendrían que colocar y a la inaccesibilidad de algunos lugares dentro de la zona de estudio para colocar los instrumentos, motivo por el cual ciertas zonas se quedarían sin monitorear.

El monitoreo de deslizamientos usando la técnica de los radares de Apertura Sintética Interferométrica Diferencial de Estación Terrena (GB-DInSAR) [5], solucionan los inconvenientes de las técnicas tradicionales mencionados anteriormente, debido a las ventajas que posee. La primera de ellas, es que es una técnica de sensado remoto, eso implica que el instrumento se puede ubicar en un lugar alejado de la zona de monitoreo, al frente del objetivo por ejemplo, lo que conlleva a una instalación más fácil y segura de los instrumentos. La segunda ventaja es que posee una amplia área de cobertura, de unos cuantos km^2 generalmente, permitiendo obtener mediciones de lugares inaccesibles por las técnicas tradicionales. Una tercera ventaja es la alta precisión de las mediciones de los desplazamientos, en el orden de las unidades de milímetros para el radar usado en este trabajo. La última ventaja es que el radar puede operar en todo tipo de climas, con presencia de lluvias o neblina por ejemplo, y también puede operar las 24 horas del día por periodos largos de tiempo (varios meses) [4].

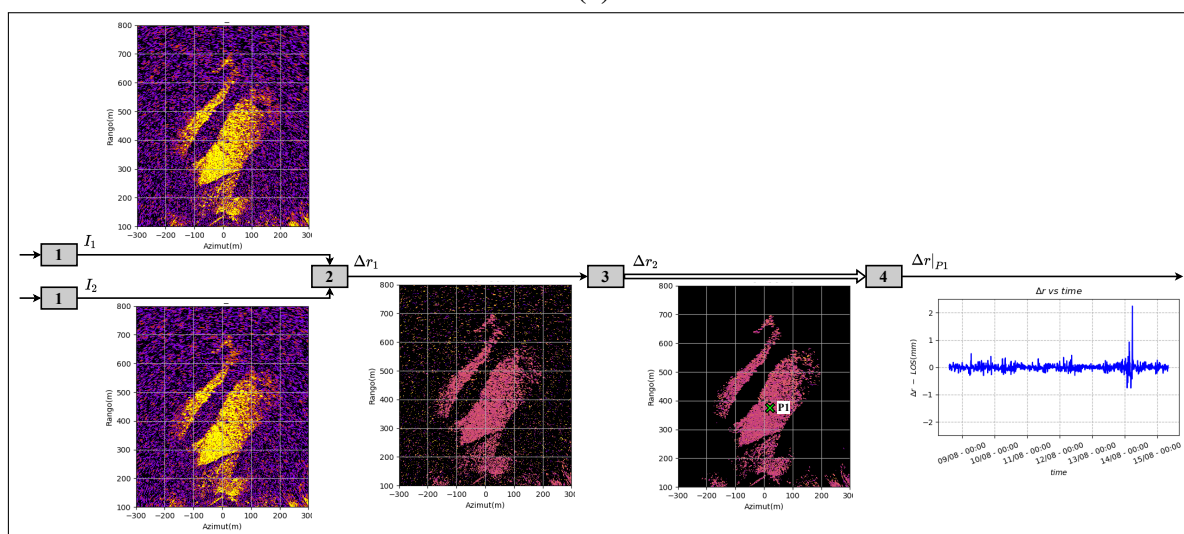
La técnica GB-DInSAR se basa en los conceptos de radares de apertura sintética (SAR) y en técnicas interferométricas (InSAR). Los resultados esperados se representan mediante gráficas de la evolución temporal de los desplazamientos en puntos deseados o en zonas particulares del blanco. Las figuras 1.1(a) y (b) muestran las etapas principales de la técnica GB-DInSAR. En primer lugar, se forman las imágenes SAR complejas I_n a partir de los datos adquiridos con un radar GB-SAR (data cruda). Luego se generan los mapas de desplazamientos (interferogramas) Δr_1 empleando la técnica InSAR, la cual consiste en restar las fases de dos imágenes SAR consecutivas para obtener un interferograma, tal como se muestra en la siguiente ecuación:

$$\Delta r_1 = \Delta \phi \frac{\lambda}{4\pi} \quad (1.1)$$

La tercera etapa de la técnica GB-DInSAR consiste en aplicar una máscara para considerar solo las zonas de alta coherencia, es decir las zonas que tienen una alta relación señal a ruido



(a)



(b)

Figura 1.1: Etapas principales de la técnica GB-DInSAR: (a) diagrama de bloques, (b) esquema gráfico. (Fuente propia)

(SNR), obteniéndose así los interferogramas Δr_2 . Finalmente se grafica y analiza la evolución temporal de los desplazamientos en ciertas zonas o cotas del mapa. Los desplazamientos obtenidos están en el orden de la longitud de onda empleada λ , en específico varían desde $-\lambda/4$ hasta $\lambda/4$, tal como se muestra en la ecuación (1.1). Por ejemplo, si se usa un radar con frecuencia 15 GHz se pueden medir desplazamientos menores a los 5 mm, por ello que se dice que esta técnica tiene una precisión milimétrica. Esto limita las aplicaciones de la técnica, es decir solo se puede aplicar efectivamente a fenómenos cuyos cambios progresivos en el tiempo sean lentos (en el orden de la longitud de onda), como es el caso de los deslizamientos de tierra lentos, para otros casos donde hay cambios bruscos en un corto intervalo de tiempo como los huaycos los resultados no son tan confiables.

1.3 Problemática

En el Instituto Geofísico del Perú (IGP) - sede Radio Observatorio de Jicamarca (ROJ) se viene desarrollando el proyecto “Desarrollo de un radar de apertura sintética para el moni-

toreo de deslizamientos” [6], [7], el cual consiste en realizar el monitoreo de deslizamientos usando la técnica GB-DInSAR. Una de los pasos principales de la técnica es la formación de las imágenes SAR, y su importancia se debe a que los resultados finales, es decir los mapas de desplazamientos, dependen en gran medida con el grado de exactitud con que se obtienen las imágenes SAR, en especial su fase.

Las imágenes SAR se obtienen empleando alguno de los algoritmos de formación de imágenes SAR presentes en la literatura, pero tienen que cumplir algunos requisitos dadas por el tipo de aplicación en específico que se necesita. El primero es que debe de ser un algoritmo capaz de formar una imagen tanto en campo cercano como en campo lejano. El segundo requerimiento, es la no simplificación o aproximación del modelo de señal en el algoritmo, tanto en magnitud como en fase, esto para poder preservar la fase de las imágenes SAR y así tener el menor error posible. El tercer requerimiento es que el algoritmo reconstruya una imagen 2D cuyo tamaño en el eje de azimut (eje del riel) sea mucho mayor al tamaño de la apertura sintética dada por el tamaño del riel, esto se requiere ya que el radar posee una apertura sintética de tamaño 1-2m, mientras que el tamaño de las imágenes formadas son de 1 Km por lado aproximadamente. Una amplia mayoría de algoritmos no cumplen esta condición en especial los que son usados con radares SAR aéreos o satélites en los que su relación de tamaños es contrario, es decir, el tamaño de las aperturas sintéticas son mucho más grandes que el tamaño de las imágenes formadas. Considerando estos requerimientos, surge la necesidad de buscar los algoritmos que cumplan las condiciones dadas y escoger el algoritmo más adecuado para aplicarlo al proyecto mencionado.

1.4 Estado del arte

Los radares de apertura sintética SAR, se desarrollaron como una técnica para formar imágenes de blancos remotos ubicados sobre un terreno. En 1953 se obtuvo la primera imagen SAR de una parte de la localidad de Key West en Florida, EE.UU; para ello se empleó un avión que transportó la instrumentación SAR [8]. En 1978 la NASA puso por primera vez un radar SAR sobre un satélite para usarlo en aplicaciones oceanográficas, luego de ello varios otros países hicieron lo mismo. Las primeras aplicaciones de los radares SAR fueron para vigilancia, tales como detectar aviones o tanques enemigos, luego se expandieron a varias disciplinas de la ciencia, yendo desde la geofísica hasta la arqueología.

Durante la década de los años 1990 a 2000 se empezaron a utilizar los radares GB-SAR para monitorear los desplazamientos diferenciales en estructuras de reservorios de agua [9]. Luego se extendieron a aplicaciones de monitoreo de deslizamientos, glaciares, laderas cubiertas de nieve, minas a tajo abierto, entre otras aplicaciones que requirieron del monitoreo

de deformaciones [10]. Las ventajas que presentan los radares tipo GB-SAR son su alta sensibilidad a las pequeñas deformaciones, su amplia cobertura de algunos Km^2 y la capacidad de formar imágenes continuamente.

Uno de los primeros algoritmos de formación de imágenes para radares GB-SAR que se desarrollaron fue el algoritmo Back Projection [11]. Luego se desarrollaron otros algoritmos más, tales como [12]: Range-Doppler, Range Migration (RMA) y Chirp Scaling. Los parámetros más usuales que se buscaron mejorar al desarrollar un nuevo algoritmo fueron la reducción del tiempo de procesamiento, el menor consumo de recursos computacionales y la facilidad de la implementación.

El algoritmo que más se emplea en aplicaciones de monitoreo de deslizamientos con radares GB-SAR es el algoritmo Back Projection [5], [13]. Las ventajas que presenta son: la reconstrucción de imágenes cuya dimensión en azimut es mucho mayor a la apertura sintética, el bajo consumo de recursos computacionales y la flexibilidad para escoger la ubicación de la escena de la imagen así como el número de puntos, con un mínimo impacto en el tiempo de procesamiento. A la actualidad se sigue empleando este algoritmo en la mayoría de casos. Por otra parte, no se encontró información de estudios comparativos de algoritmos de formación de imágenes para esta aplicación, lo cual permita escoger otros algoritmos o confirmar el uso del algoritmo Back Projection.

1.5 Objetivos

Esta tesis presenta un objetivo general y tres objetivos específicos, los cuales se enuncian a continuación.

1.5.1 Objetivo General

- Implementar y comparar tres algoritmos de formación de imágenes SAR, usando un radar de apertura sintética de estación terrena GB-SAR, para una aplicación de monitoreo de deslizamientos superficiales de tierra.

1.5.2 Objetivos Específicos

- Implementar los algoritmos de formación de imágenes Frequency Domain Back Projection (FDBP), Range Migration Algorithm (RMA) y Discrete Linear Inverse Problem (DLIP).
- Evaluar los tres algoritmos implementados con datos obtenidos por simulación y datos reales.
- Comparar los tres algoritmos implementados en términos de calidad de reconstrucción de la imagen, calidad de enfoque y coste computacional.

CAPÍTULO II MARCO TEÓRICO

2.1 Fundamentos de radares

Los radares de apertura sintética de estación terrena GB-SAR ¹ se basan en conceptos fundamentales de la teoría de radares, por ello se explicarán algunos de estos conceptos que se usarán en este trabajo como son: la ecuación del radar y la forma de onda SFCW.

2.1.1 Ecuación de radar

El radar GB-SAR usado presenta una configuración biestática debido a la presencia de una antena para transmisión y otra para recepción, por lo tanto la ecuación del radar para este caso (hard targets) se muestra a continuación: [8]

$$\frac{P_o}{P_{in}} = \frac{G_1 G_2 \lambda^2 \sigma}{(4\pi)^3 \cdot (R_1 R_2)^2} \quad (2.1)$$

Donde:

- P_{in} es la potencia de transmisión
- P_o es la potencia de recepción
- G_1, G_2 son las ganancias de la antena de transmisión y de recepción respectivamente
- σ es el Radar Cross Section (RCS) del blanco
- λ es el número de onda asociada a la frecuencia usada
- R_1, R_2 es la distancia del blanco hacia el transmisor y receptor respectivamente.

En la ecuación (2.1) no se consideran las pérdidas del sistema como atenuaciones de canal, de la atmósfera, entre otros. En la figura 2.1 se muestra la geometría usada para la terminología.

El término RCS ² o también llamado sección transversal de dispersión, cuyo símbolo es σ , se refiere al área efectiva de un objeto (blanco) cuando es iluminado por una onda plana proveniente de la onda transmitida, y el cual presenta unidades de superficie (m^2). El objeto que presenta un alto valor de RCS es el reflector de esquina ³ (diedro y triedro), motivo por el cual son usados como blancos en varias pruebas de algoritmos de formación de imágenes SAR. Este término es proporcional a la reflectividad, la cual es la relación de energías electromagnéticas (EM) entre la onda reradiada hacia el receptor y la onda que

¹Ground Based - Synthetic Aperture Radar

²Radar Cross Seccion

³También llamado Corner Reflector, es una superficie que consta de 2 o 3 planos perpendiculares entre ellos, y que refleja los rayos incidentes directamente hacia la fuente.

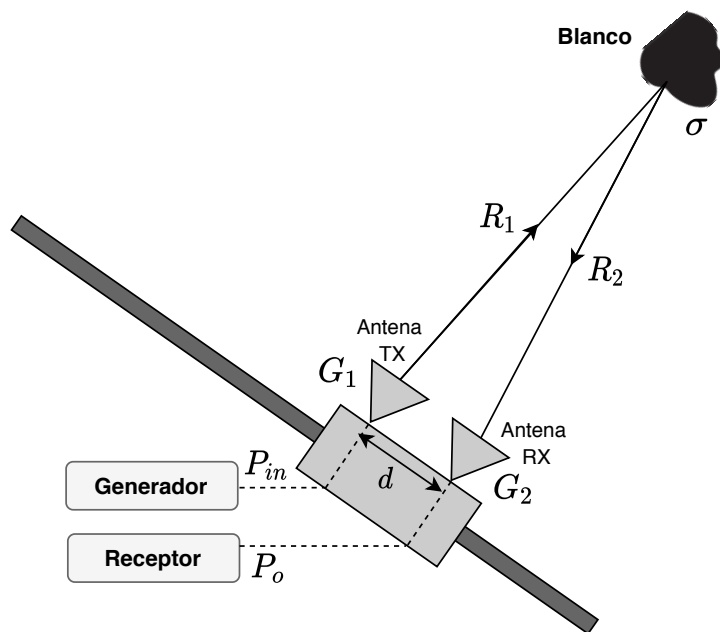


Figura 2.1: Geometría usada para la obtención de la ecuación del radar. (Adaptado de [8])

intercepta al blanco. Por esta razón en varias fuentes [12], [5] se asocia el término σ a la reflectividad del blanco, y en este trabajo también se adoptará lo mismo.

La ecuación (2.1) se puede simplificar con las siguientes consideraciones. Ambas antenas del radar GB-SAR están sujetas a una plataforma móvil y debido a que la distancia de separación entre antenas d es pequeña respecto a la distancia de los blancos ($d \ll R_1$ & $d \ll R_2$), se realiza la siguiente aproximación:

$$R_1 \approx R_2 = R \quad (2.2)$$

La ganancia de las dos antenas son iguales para el caso del radar GB-SAR empleado, por esto se cumple la siguiente relación:

$$G_1 = G_2 = G \quad (2.3)$$

Por lo tanto, la ecuación del radar quedaría del siguiente modo, tomando en cuenta las dos consideraciones anteriores:

$$\frac{P_o}{P_{in}} = \frac{G^2 \lambda^2 \sigma}{(4\pi)^3 R^4} \quad (2.4)$$

La ecuación (2.4) es similar a la ecuación del radar para el caso monoestático [8], por consiguiente el radar GB-SAR usado se considera como un caso monoestático aproximado.

Una consecuencia importante de la ecuación del radar es la expresión del rango de Detección del Radar, el cual es la máxima distancia del blanco (R_{max}) que puede ser detectado

sobre el piso de ruido del radar, y está dado por: [8]

$$R_{max} = \left(\frac{P_{in}}{P_{min}} \frac{A_{eff}^2 \sigma}{4\pi\lambda^2} \right)^{1/4} \quad (2.5)$$

Donde:

- P_{min} es la mínima potencia recibida para que el radar aun detecte señales recibidas.
- A_{eff} es el área de apertura efectiva de la antena.

2.1.2 Formas de onda de las señales de transmisión y recepción

El radar GB-SAR empleado usa una señal de onda continua de frecuencia a pasos SFCW⁴ [8]. Esta señal se forma emitiendo un conjunto de subondas continuas de corta duración y de frecuencia única. Las frecuencias de cada subonda son incrementadas secuencialmente cada paso Δf , tal como se muestra en la figura 2.2. En un solo barrido de señal SFCW, N subondas son generadas, cada una con una frecuencia discreta de $f_n = f_0 + n.\Delta f$ con $n = 0, 1, 2, \dots, (N - 1)$. Cada subonda tiene una duración pequeña τ y es generada cada cierto tiempo T partiendo del inicio. El ancho de banda BW de la señal es:

$$BW = (f_{N-1} - f_0) + \Delta f \quad (2.6)$$

$$BW = N.\Delta f \quad (2.7)$$

La señal SFCW presenta la ventaja de barrer un amplio ancho de banda usando un hardware simple, como un analizador vectorial de redes (VNA) en este caso. Este tipo de señal permite obtener el rango de los blancos empleando el siguiente procedimiento, el cual se usará en la implementación de los algoritmos de formación de imágenes SAR. Suponiendo que el blanco está a un rango R_0 del radar GB-SAR, con un solo barrido de señal SFCW transmitida hacia el blanco, la fase de la onda recibida es proporcional al rango y esta dado por la siguiente

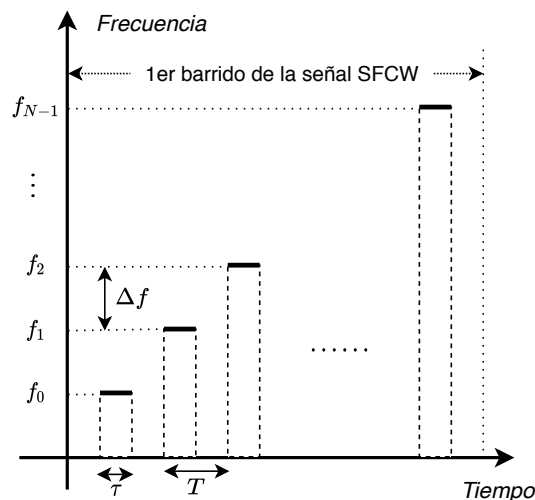


Figura 2.2: Señal SFCW en el plano Tiempo-Frecuencia. (Adaptado de [8])

⁴Stepped Frequency Continuous Wave

ecuación: [8]

$$E^s[\mathbf{f}] = Ae^{-j2\mathbf{k}R_0} \quad (2.8)$$

Donde $E^s[\mathbf{f}]$ es la señal recibida (campo eléctrico), A es la amplitud y \mathbf{k} el número de onda correspondiente al vector de frecuencias \mathbf{f} . El valor 2 presente en la fase se debe al recorrido de ida y vuelta de la onda. Por lo tanto se puede obtener el rango R_0 tomando la transformada inversa de Fourier (IFT) de $E^s[\mathbf{f}]$, a la señal resultante se le suele llamar perfil de rango del blanco ⁵.

2.2 Fundamentos de radares GB-SAR

El radar de apertura sintética SAR ⁶ es una técnica de sensado remoto usado para formar imágenes de blancos sobre un terreno o escena. Se caracteriza por generar una antena larga efectiva (apertura sintética) usando técnicas de procesamiento de señales y antenas físicas pequeñas, con el objetivo de conseguir resoluciones pequeñas en la dimensión de azimut durante el proceso de formación de una imagen SAR. La apertura sintética se forma mediante el movimiento de la plataforma que transporta a la instrumentación del radar SAR. Los medios usuales que se emplean como plataformas del radar son: aviones (*airborne SAR*), satélites (*spaceborne SAR*) y estaciones terrenas (GB-SAR).

Los radares de Apertura Sintética de Estación Terrena (GB-SAR) suelen emplear como plataforma una estructura mecánica móvil montada sobre un riel fijo al suelo, el riel puede ser lineal como el radar GB-SAR que se usa en este trabajo o también curvo. Estos radares son comúnmente usados en aplicaciones de alcance cercano como monitoreo de deslizamientos o de edificios [5], [4]. Posee una instrumentación portable y un tiempo de ejecución relativamente corto (10 m en el caso del radar usado) el cual permite la adquisición de los

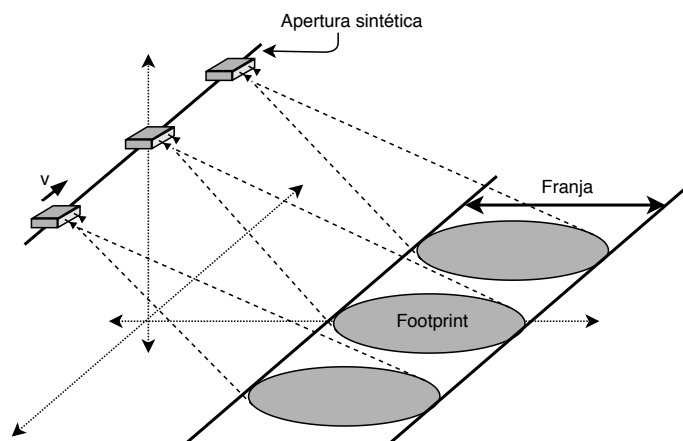


Figura 2.3: Modo Stripmap del radar GB-SAR.(Adaptado de [8])

⁵Range Profile

⁶Synthetic Aperture Radar

datos de manera simple, rápida, y con gran estabilidad.

El modo de operación que emplea el radar GB-SAR es *stripmap SAR*⁷, en este modo se usan antenas de lado, dos en este caso, las cuales permanecen fijas durante todo el periodo de ejecución, con su ancho de haz apuntando siempre en la misma dirección y barriendo una franja del terreno paralelo a la dirección del movimiento, tal como se presenta en la figura 2.3. La huella del haz de la antena sobre el terreno se llama *footprint*. En este modo la región de la imagen debe de permanecer siempre en el ancho de haz de la antena real en cada posición de la apertura sintética.

2.2.1 Geometría

Las imágenes SAR en este trabajo se formarán sobre un plano de ejes coordenados: Azimut y Rango. Estas coordenadas o dimensiones se muestran en la figura 2.4 y se definen del siguiente modo:

- **Azimut:**⁸ Es la primera coordenada de la imagen SAR la cual corresponde a la dirección alineada con el vector velocidad de la plataforma o a la trayectoria de la apertura sintética. En este trabajo, el eje coordenado de la dimensión de azimut coincide con la dirección del riel lineal, tal como se muestra en la figura 2.4.
- **Rango:** Es la distancia entre el radar y el blanco. En este trabajo este término también se referirá a la segunda coordenada de la imagen SAR, la cual corresponde a la dirección perpendicular al eje de azimut. Se presentan 2 tipos:
 - **Rango inclinado:**⁹ Es la distancia de línea de vista (LOS¹⁰) ‘ r ’ entre el radar y el blanco (mostrado en la figura 2.4). También hace referencia al eje coordenado

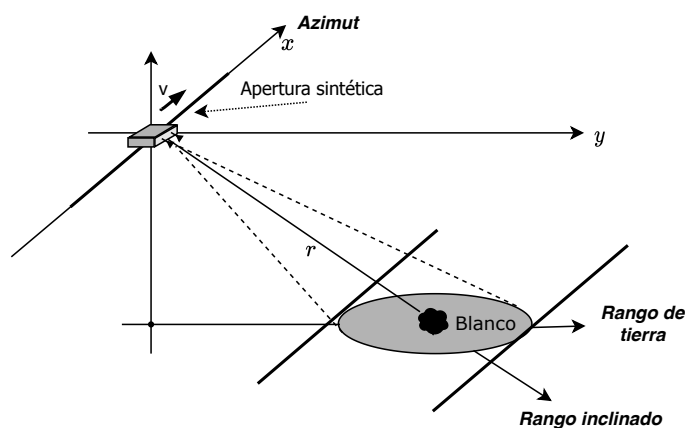


Figura 2.4: Geometría usada para la formación de imágenes SAR. (Fuente propia)

⁷En el modo Stripmap el ancho de haz de las antenas barren una franja de terreno paralelo a la dirección de movimiento.

⁸Cross range

⁹Slant range

¹⁰Line Of Sight

perpendicular al eje de azimut en la dirección LOS radar-blanco. Al mencionar la coordenada de rango a lo largo del desarrollo de este trabajo, se referirá a este tipo de rango.

- **Rango de tierra:**¹¹ Es la distancia proyectada sobre el terreno en la dirección radar-blanco, mostrada en la figura 2.4.

2.2.2 Resolución

La resolución en radares SAR se entiende como la capacidad de distinguir o discriminar entre dos blancos cercanos, es decir, si 2 blancos se separan por una distancia menor a la resolución del sistema, estos no se podrán distinguir separadamente en la imagen final. En radares GB-SAR se encuentran 2 tipos de resoluciones: rango y azimut, uno por cada dimensión de la imagen, los cuales se explicarán a continuación.

Resolución en rango

La resolución en rango está determinado por la habilidad del radar de distinguir 2 blancos puntuales por separado en la dirección del rango inclinado, el cual se muestra en la figura 2.5. Para radares GB-SAR que usan la señal SF-CW, la resolución en rango esta dado por la siguiente expresión: [8]

$$\Delta r = \frac{c}{2BW} \quad (2.9)$$

Como se observa en la ecuación (2.9), la resolución en rango depende exclusivamente del ancho de banda de forma inversamente proporcional, es decir, para obtener valores pequeños de resolución es necesario aumentar el ancho de banda BW de la señal. El radar GB-SAR que se usa suele emplear 100MHz como ancho de banda, para esta frecuencia se obtiene una resolución de 1.5 m, si se aumenta el ancho de banda a 300MHz se logra una resolución menor de 0.5 m.

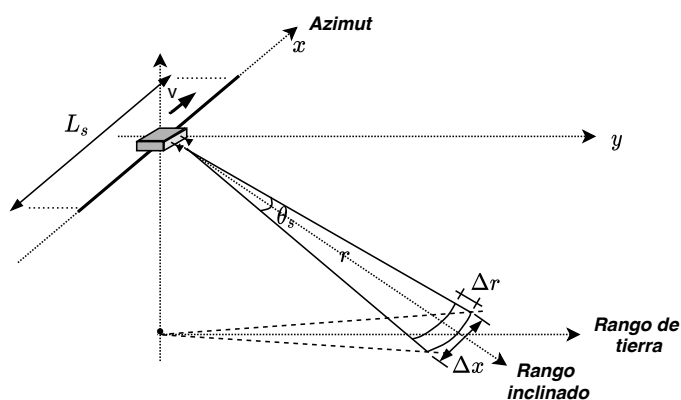


Figura 2.5: Resolución en rango y azimut del radar GB-SAR. (Adaptado de [14])

¹¹Ground range

Resolución en azimut

Es la resolución medida en la dirección de azimut. En radares de apertura real (RAR), esta resolución depende del tamaño de la antena:

$$\theta = \lambda / l \quad (2.10)$$

$$\Delta x = r\theta = r\left(\frac{\lambda}{l}\right) \quad (2.11)$$

Donde “ r ” es el rango del blanco y “ l ” es el tamaño de la antena. Analizando las ecuaciones anteriores se observa que para obtener una buena resolución (Δx pequeño) se necesita una antena de dimensiones grandes. Por ejemplo, para obtener una resolución de 1 m a un rango de 1 Km con una frecuencia de 15 GHz, se necesitaría una antena de 20 m, el cual es un valor alto, y si se quisiera aumentar el tamaño de la imagen, las dimensiones crecerían aún mas, lo cual lo vuelve impráctico en la mayoría de los casos.

Los radares SAR solucionan este inconveniente, permitiendo obtener valores pequeños de resolución usando antenas de dimensiones pequeñas. La expresión de la resolución en azimut para radares GB-SAR es el siguiente, el cual se muestra en la figura 2.5: [8]

$$\Delta x = r \frac{\lambda}{2L_s} \quad (2.12)$$

Donde “ L_s ” es el tamaño de la apertura sintética. Como se puede observar en la ecuación anterior la resolución ya no depende del tamaño de la antena sino de la apertura sintética, e indica que para obtener resoluciones más pequeñas basta con incrementar el tamaño de la apertura sintética. Lo anteriormente dicho conlleva a la idea de poder aumentar la apertura sintética lo más que se pueda para obtener resoluciones cada vez más pequeñas, sin embargo hay un límite en el incremento de la apertura sintética a partir del cual la resolución ya no disminuye más. La razón del límite es el modo de operación *stripmap* empleado, el cual condiciona que la escena de la imagen debe de ser iluminada por el ancho de haz de la antena en todas las posiciones de la apertura sintética. Por este hecho, se presentan 2 expresiones de la resolución en azimut de acuerdo a [14], la clasificación depende de un rango crítico R_c , si el blanco esta a un rango mayor a R_c la resolución se mantiene como el dado en la ecuación (2.12), pero si el rango es menor a R_c la resolución es un valor constante e igual a:

$$\Delta x = l/2 \quad (2.13)$$

Donde l es el tamaño de la antena real. Para calcular el valor de R_c se usa la siguiente expresión:

$$R_c = \frac{L_s l}{\lambda} = \frac{L_s}{\theta} \quad (2.14)$$

Para el radar GB-SAR usado, se calculó el valor de R_c , se consideró el tamaño de la aper-

tura sintética $L_s = 1.4$ m y una antena de bocina de ancho de haz de $\theta = 18^\circ$, reemplazando estos valores se obtiene:

$$R_c = \frac{L_s}{\theta} = \frac{1.4}{18 \times \frac{\pi}{180}}$$

$$R_c = 4.5 \text{ m}$$

El resultado anterior indica que para distancias menores a 4.5 m la resolución en azimut se mantiene constante y es igual a la mitad de la longitud de la antena. Para distancias mayores, la resolución en azimut varía dependiendo al tamaño de la apertura sintética y al rango r .

2.2.3 Conceptos básicos de formación de imágenes

Una imagen SAR posee 2 dimensiones, azimut y rango en este caso, con sus respectivas resoluciones en cada dimensión. Cada celda de resolución contiene información de amplitud y fase. Como ya se mencionó anteriormente, la resolución en rango está dado por la técnica de la forma de onda SFCW, mientras que la resolución en azimut lo determina la técnica SAR. Por lo tanto la imagen SAR es formada por el conjunto de celdas de resolución. Para generar las imágenes SAR se emplean algoritmos de formación de imágenes, los cuales emplean términos básicos como: histórico de fase, resolución en rango, resolución en azimut, y se explicarán a continuación.

Histórico de Fase

El histórico de fase, también llamado data cruda, es el conjunto de datos adquiridos por el radar GB-SAR, y conforman los datos de entrada a los algoritmos de formación de imágenes. Si se usa un radar GB-SAR con forma de onda SF-CW, entonces los datos adquiridos son organizados en una matriz de 2 dimensiones: posiciones del riel y frecuencias. Por este hecho, se menciona que el histórico de fase está en el dominio frecuencia-espacio. El tipo de dato de la matriz es complejo, es decir cada dato posee magnitud y fase.

Perfil de rango

El perfil de rango es la forma de onda recibida del blanco que ha sido previamente iluminado con el radar con suficiente ancho de banda en frecuencia (barrido de frecuencia). Para este caso, en que se transmite una señal SFCW, el perfil de rango se obtiene mediante una transformada inversa de Fourier (IFT) de la señal recibida. El perfil de rango permite obtener el rango del blanco desde la ubicación actual del radar. En la parte superior de la figura 2.6 se muestra un ejemplo del perfil de rango de un blanco puntual, en el cual el pico de la señal corresponde al rango R_o .

Perfil de azimut

El perfil de azimut permite determinar la posición del blanco en azimut, y puede ser obtenido con las señales retornadas del blanco, haciendo un barrido de todas las posiciones del riel para una frecuencia constante. En la parte inferior de la figura 2.6 se muestra un ejemplo del perfil de azimut para un blanco puntual, en el cual el valor máximo en magnitud corresponde a la posición en azimut del blanco.

2.3 Conceptos de PDS usados en formación de imágenes SAR

Los algoritmos de formación de imágenes a implementar están basados en conceptos fundamentales de procesamiento digital de señales (PDS) tales como: transformada discreta de Fourier (DFT y FFT), decimación, interpolación, muestreo de datos y filtros/ventanas. A continuación se explicarán cada uno de estos conceptos con enfoque en la formación de imágenes SAR.

2.3.1 Transformada de Fourier

La transformada de Fourier (FT) juega un papel importante en la formación de imágenes SAR, ya que es una herramienta útil para convertir las señales del dominio del espacio al dominio de la frecuencia, y viceversa. Los conceptos de la FT se utilizan a menudo en el desarrollo de los algoritmos FDBP y RMA. A continuación se explicarán los conceptos más importantes de la teoría de la transformada de Fourier que se usan en aplicaciones de formación de imágenes SAR.

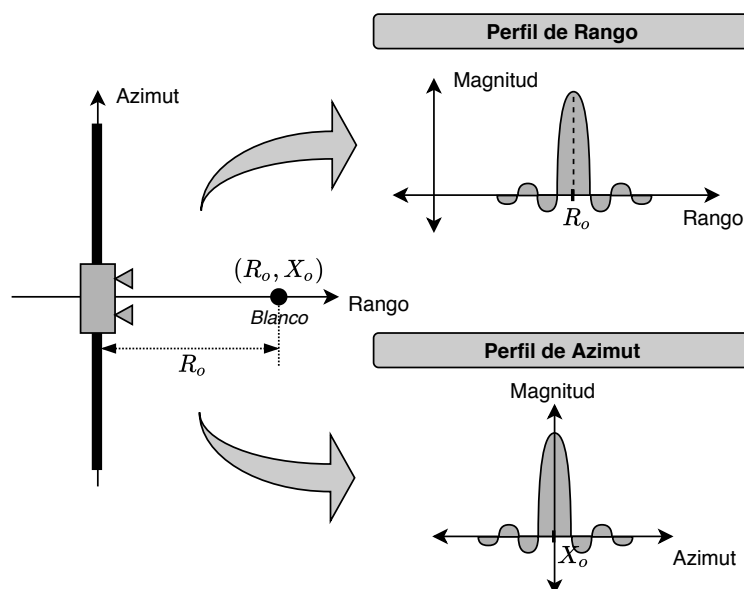


Figura 2.6: Representación gráfica del perfil de rango y azimut para un blanco puntual. (Fuente propia)

Transformada continua de Fourier (CTFT)

La transformada continua de Fourier (CTFT) está definida para señales aperiódicas en tiempo continuo, tal como un pulso en el dominio del tiempo. El par de transformadas de Fourier tiene la siguiente expresión: [15]

$$S(j\omega) = \int_{-\infty}^{+\infty} s(t)e^{-j\omega t} dt \quad (2.15)$$

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S(j\omega)e^{j\omega t} d\omega \quad (2.16)$$

Donde $S(j\omega)$ es el espectro continuo de la señal $s(t)$. La ecuación (2.15) es la definición de la transformada de Fourier, mientras que la ecuación (2.16) es de la transformada inversa de Fourier.

La CTFT principalmente se emplea para realizar un análisis teórico, mas no se puede emplear en una aplicación real de procesamiento de señales, ya que como el procesamiento se realiza en computadoras, una señal continua necesitaría infinitos puntos para poder representarla, y esto implicaría que la computadora tenga una memoria (RAM y disco duro) de tamaño infinito también, lo cual resulta impráctico y hasta imposible en la realidad.

Transformada discreta de Fourier en tiempo continuo (DTFT)

Para poder implementar una transformada de Fourier en la práctica es necesario discretizar la señal continua $s(t)$, la cual está presente, por ejemplo, en el dominio del tiempo. Por lo tanto, el tipo de transformada de Fourier que se aplica a las señales discretas es la transformada discreta de Fourier en tiempo continuo (DTFT). El resultado es la señal $S(e^{j\omega})$ (espectro) presente en el dominio de la frecuencia, la cual es una señal continua: [15]

$$S(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} s[n]e^{-j\omega n} \quad (2.17)$$

$$s[n] = \frac{1}{2\pi} \int_{2\pi} S(e^{j\omega})e^{j\omega n} \quad (2.18)$$

Por lo tanto tampoco se puede usar esta transformada en una aplicación práctica ya que no se podría representar el espectro de una señal, ya que es continua.

Transformada discreta de Fourier (DFT)

Es en este sentido que nace la transformada discreta de Fourier (DFT), el cual está definido para señales discretas en ambos dominios, del tiempo y de la frecuencia.

La FT presenta varios tipos de expresiones dependiendo del tipo de señal usado, motivo por el cual se identificarán en primer lugar la clase de señales que se usan en aplicaciones de formación de imágenes SAR. La señal de entrada a los algoritmos es el histórico de fase, el cual es una señal discreta y finita, con uno de sus dominios en el dominio del espacio y la otra en el dominio de la frecuencia. Por otro lado, la señal de salida es una imagen SAR

la cual esta compuesta por píxeles, y por lo tanto también pertenece a una señal discreta y finita, con ambas dimensiones en el dominio del espacio.

Debido a que ambas señales son discretas y finitas, y además el procesamiento u obtención de ellas se realizarán en un computador, el tipo de FT que se empleará será la transformada discreta de Fourier (DFT) y su complemento la transformada inversa de Fourier discreta (IDFT). Como se usan señales en el dominio del espacio, la definición de la DFT se muestra en este dominio. La DFT transforma una señal discreta y finita del dominio del espacio $s[n]$ al dominio de la frecuencia $S[k]$, por otro lado la IDFT realiza el proceso inverso llevando la señal discreta y finita del dominio de la frecuencia al dominio del espacio, sus expresiones se muestran en las ecuaciones (2.19) y (2.20) respectivamente: [16]

$$S[k] = \sum_{n=0}^{N-1} s[n] e^{-j2\pi \frac{n}{N} k} \quad (2.19)$$

Donde: $k = 0, 1, 2, \dots, (N-1)$

$$s[n] = \frac{1}{N} \sum_{k=0}^{N-1} S[k] e^{j2\pi \frac{k}{N} n} \quad (2.20)$$

Donde: $n = 0, 1, 2, \dots, (N-1)$

Puesto que las señales usadas son imágenes, se debe de calcular la DFT en dos dimensiones (2D-DFT). Sin embargo, debido a la propiedad de separabilidad este cálculo se puede realizar determinando la 1D-DFT en cada dimensión por separado y luego multiplicando ambas expresiones. Una etapa del proceso de generación de imágenes SAR donde se emplea muy a menudo la DFT es en la obtención del perfil de rango, la cual servirá como ejemplo para explicar posteriormente las propiedades y conceptos adicionales de la DFT. El perfil de rango se obtiene mediante una aplicación directa de la IDFT al histórico de fase $S_{k,i}$, en específico se selecciona una fila del histórico de fase correspondiente a una posición del riel, quedando representado por S_k , y se aplica la IDFT obteniendo s_n . En la figura 2.7 se muestra en la parte superior la gráfica del histórico de fase simulado en el dominio de la frecuencia (una fila) de un blanco puntual con rango teórico $R_o = 3$ m, y en la parte inferior de la misma figura se muestra el perfil de rango representado en el dominio del espacio, donde la ubicación de la señal de mayor intensidad indica aproximadamente el rango del blanco, el cual es de $R_o = 2.84$ m en este caso.

Terminología del espectro

Antes de explicar los términos más importantes dentro de la gráfica del espectro de una señal como resultado de su DFT, se determinarán los dominios empleados. Como ya se mencionó anteriormente, la imagen (señal) esta definido en el dominio del espacio y su DFT está representado en el dominio de la frecuencia espacial o del número de onda k . Como una

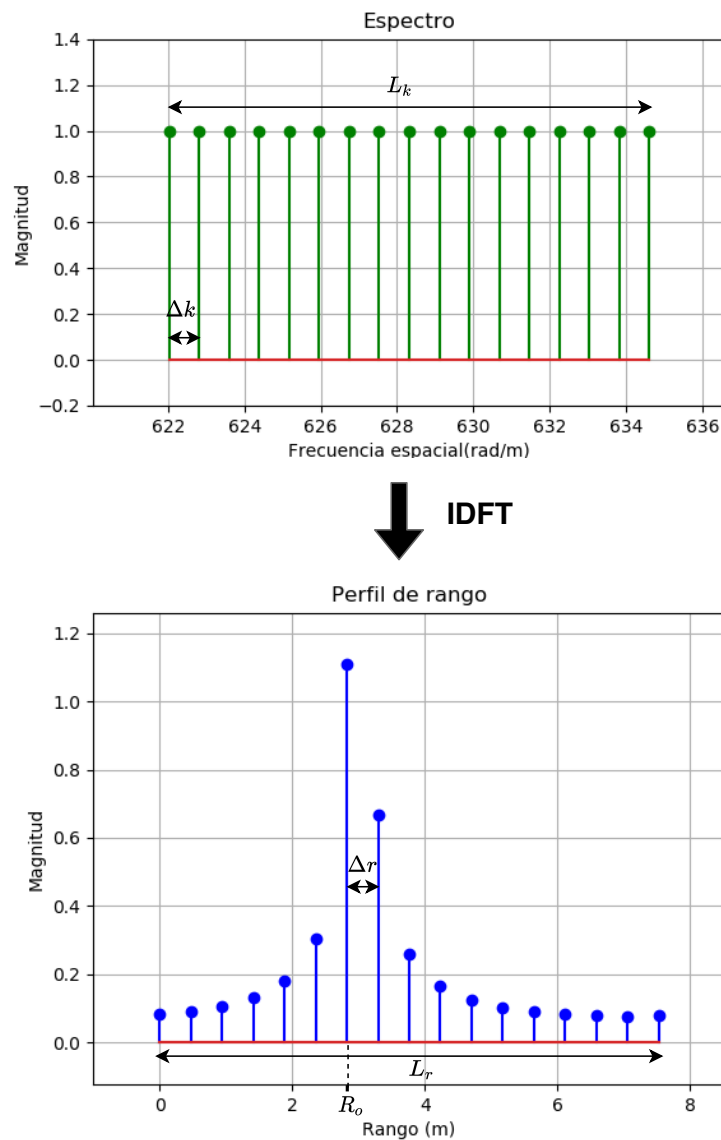


Figura 2.7: Perfil de Rango (abajo), después de una IDFT a una fila del histórico de fase (arriba) presente en el dominio frecuencial. (Fuente propia)

de las dimensiones del histórico de fase esta definido en el dominio de la frecuencia f , se tendrá que convertir al dominio k . La relación típica entre frecuencia f y número de onda k' es el siguiente:

$$k' = \frac{2\pi}{c} f \quad \left(\frac{rad}{m} \right) \quad (2.21)$$

Sin embargo, al usar radares SAR esta expresión cambia a la siguiente: [8], [17]

$$k = 2k' \quad (2.22)$$

$$k = \frac{4\pi}{c} f \quad (2.23)$$

El primer término dentro del espectro es la resolución en frecuencia espacial Δk , el cual está determinado por la longitud del intervalo de observación L_r del dominio del espacio (ver

parte inferior de la figura 2.7), y posee la siguiente expresión:

$$\Delta k = \frac{2\pi}{L_r} \quad (2.24)$$

El segundo término dentro del espectro es el intervalo de observación en frecuencia L_k , el cual está determinado por la resolución espacial Δr , mostrado en la figura 2.7, y posee la siguiente expresión:

$$L_k = \frac{2\pi}{\Delta r} \quad (2.25)$$

Las ecuaciones (2.24) y (2.25) también se usan para calcular Δr , L_r a partir de Δk , L_k cuando se cambia del dominio de la frecuencia espacial hacia dominio del espacio usando la IDFT.

Propiedades de la DFT

En el presente trabajo se usarán 2 propiedades importantes de la DFT: la periodicidad y el desplazamiento circular, explicados a continuación.

La primera propiedad es la periodicidad, la cual señala que el par $s[n]$ y $S[k]$ son señales discretas periódicas de periodo N , osea:

$$s(n + N) = s(n)$$

Con $n = 0, 1, 2, \dots, N - 1$.

$$S(k + N) = S(k)$$

Con $k = 0, 1, 2, \dots, N - 1$

En la figura 2.8 se muestra un ejemplo de esta propiedad, como se puede ver tanto el espectro como el perfil de rango son periódicos. Puede parecer un poco difícil entender esta propiedad ya que cuando se aplica la DFT a una señal, el resultado no parece ser periódico, sin embargo la periodicidad existe, solo que es implícita y el resultado obtenido conforma solo un periodo de la DFT. Por lo tanto antes de aplicar la DFT o IDFT a una señal discreta finita, es conveniente pensar que la señal discreta es periódica y que las muestras de la señal conforman un periodo, así también es conveniente pensar en el resultado de la DFT e IDFT como una señal discreta periódica donde las muestras obtenidas conforman un periodo. La propiedad de periodicidad de la DFT permite entender otras propiedades y conceptos de la DFT como el relleno de ceros o el desplazamiento circular, así mismo permite realizar el cálculo correcto de la DFT. Una segunda propiedad de la DFT es el desplazamiento circular de una secuencia. De acuerdo a [16] un desplazamiento circular de una señal discreta $s[n]$ de N muestras equivale a un desplazamiento lineal de su extensión periódica $\tilde{s}[n]$, en la cual la señal desplazada circularmente se suele representar como el índice de modulo N :

$$s'(n) = s(n - m, \text{modulo } N) \quad (2.26)$$

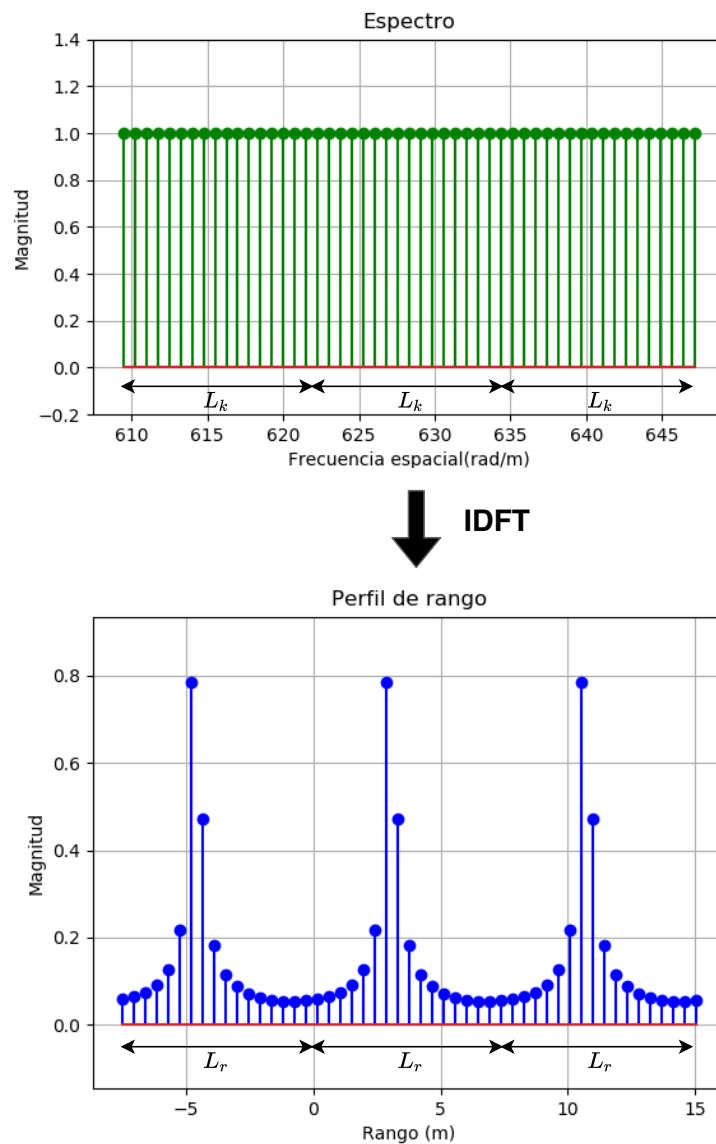


Figura 2.8: Ejemplo de la periodicidad de la DFT tanto en el dominio del espacio (imagen superior) como de la frecuencia (imagen inferior). (Fuente propia)

$$s'(n) = s((n - m))_N \quad (2.27)$$

donde $m \in \mathbb{Z}$.

Para valores positivos de m el desplazamiento se da en sentido antihorario o hacia la derecha en su representación periódica, y para valores negativos en el sentido contrario.

La DFT de la señal desplazada $s'(n)$ es similar a la DFT de la señal original $s(n)$ en magnitud, pero con un cambio en la fase tal como se muestra a continuación: [16]

Si:

$$s(n) \xrightleftharpoons[\text{IDFT}]{\text{DFT}} S(k)$$

Entonces:

$$s((n - m))_N \xrightleftharpoons[\text{IDFT}]{\text{DFT}} S(k)e^{-j(\frac{2\pi k}{N})m} \quad (2.28)$$

Donde $0 \leq n \leq N - 1$. Como se puede ver en la segunda expresión, el cambio de fase es una adición de una fase lineal a la fase del espectro original.

De manera análoga, un desplazamiento circular en el dominio de la frecuencia hace variar la fase de la IDFT, tal como se muestra a continuación: [16]

Si:

$$x(n) \xrightleftharpoons[\text{IDFT}]{\text{DFT}} X(k)$$

Entonces:

$$x(n)e^{j(\frac{2\pi n}{N})m} \xrightleftharpoons[\text{IDFT}]{\text{DFT}} X((k-m))_N \quad (2.29)$$

Donde $0 \leq n \leq N - 1$.

Para comprobar que esta propiedad solo hace variar la fase de la señal mas no su magnitud se usará el ejemplo del perfil de rango. Se desplazará 3 unidades a la derecha ($m = 3$) la señal en el dominio de la frecuencia S_k y se obtendrá su IDFT s_n , luego se graficará la magnitud y fase de la señal sin desplazar, el cual se muestra en la figura 2.9, y desplazada, tal como se muestra en la figura 2.10. En ambas figuras se puede apreciar que la magnitud se mantiene igual mientras que la fase cambia.

Zero padding

La propiedad del relleno con ceros o también llamado *zero-padding* consiste en alargar el periodo de la señal mediante la adición de ceros [16] con el objetivo de sobremuestrear el espectro de la señal para que tenga una representación más precisa. Una de las ventajas que ofrece es que permite usar interpolaciones sencillas, como una interpolación lineal, para obtener una representación más aproximada del espectro de la señal. Teniendo un espectro más preciso de la señal es más fácil identificar características importantes como amplitudes y posiciones de los picos, los cuales en el contexto de formación de imágenes SAR permiten obtener las posiciones más aproximadas de los blancos. La aplicación del *zero-padding* se puede realizar tanto en el dominio del espacio como de la frecuencia. La forma de efectuar el *zero-padding* en este trabajo, y la más común, es agregando ceros al final de la secuencia de datos de la señal, es decir, si se tiene una señal en el dominio del espacio $s(n)$ de longitud L y se quiere realizarle un *zero-padding* agregándole $(N - L)$ ceros, la nueva señal $s_{zp}(n)$ de longitud N tendrá la siguiente expresión:

$$s_{zp}(n) = \underbrace{\{x(0), x(1), \dots, x(L-1)\}}_L \underbrace{\{0, \dots, 0\}}_{(N-L)}$$

En la figura 2.7 se muestra el perfil de rango sin realizar un relleno de ceros, como se puede observar no se identifica bien el rango del blanco, debería ser 3 m pero de la gráfica

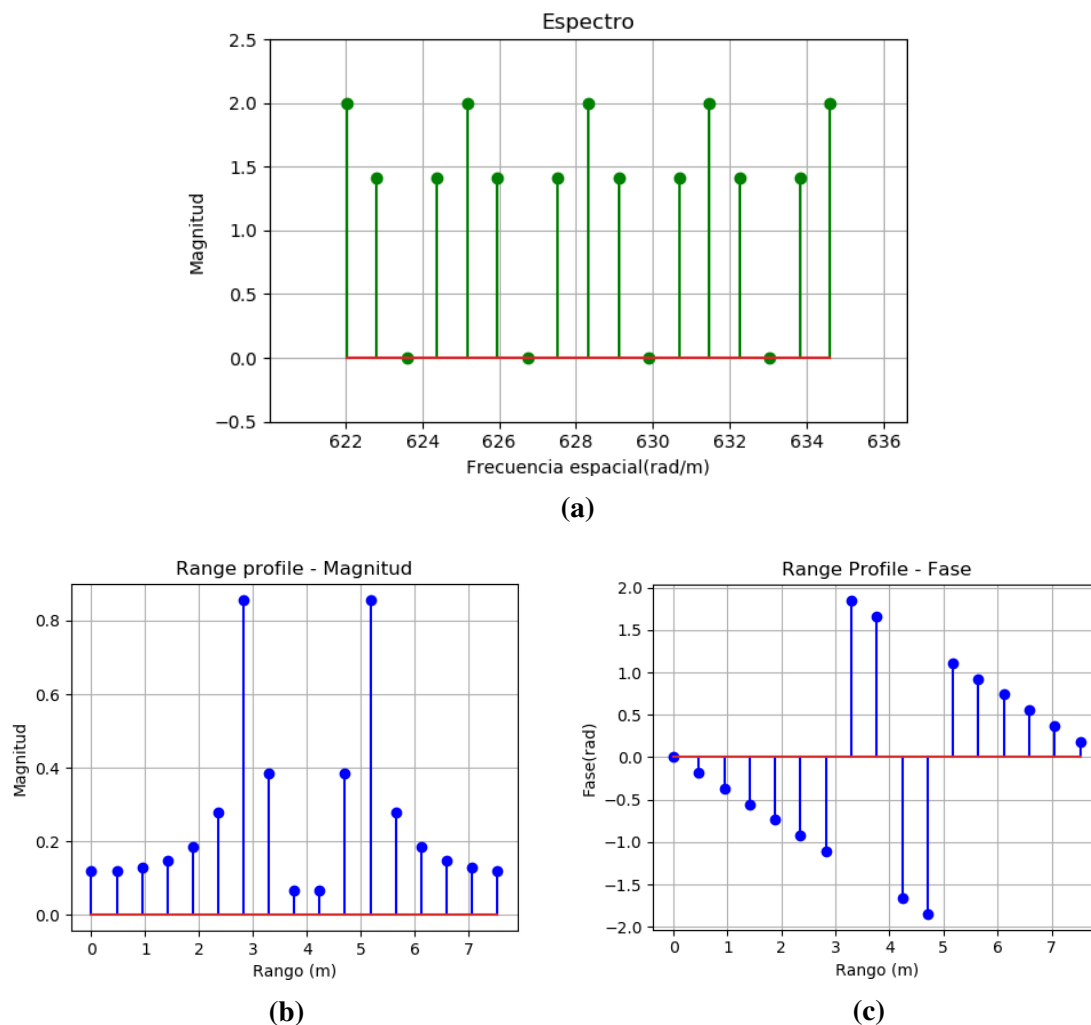


Figura 2.9: Ejemplo de la propiedad de desplazamiento circular de la DFT. En este caso no se presenta desplazamiento. (a): Espectro, (b) Perfil de Rango en magnitud, (c) Perfil de Rango en fase. (Fuente propia)

se obtiene un valor de 2.8m. Para identificar mejor el rango, se realiza el relleno de ceros, al espectro de la señal se le agrega 4 veces el tamaño de la señal con ceros al final de la secuencia, es decir $N = 5L$, obteniéndose un perfil de rango mas preciso y pudiendo identificar un rango del blanco de 3.006m, el cual es un valor más cercano al rango teórico del blanco, estos resultados se muestran en la figura 2.11.

Cuando se le agregan muchos ceros a la señal, el espectro de su DFT se asemeja mucho al espectro de su transformada de Fourier en tiempo discreto (DTFT). La DTFT se define para una señal discreta no periódica en el dominio del tiempo, siendo su espectro una señal continua y periódica. Por lo tanto, se dice que el *zero-padding* tiene el efecto de aproximar la DFT de una señal al de su DTFT, mientras más ceros se agreguen más se cumplirá esta relación.

El *zero-padding* no mejora la resolución frecuencial del sistema, simplemente proporcio-

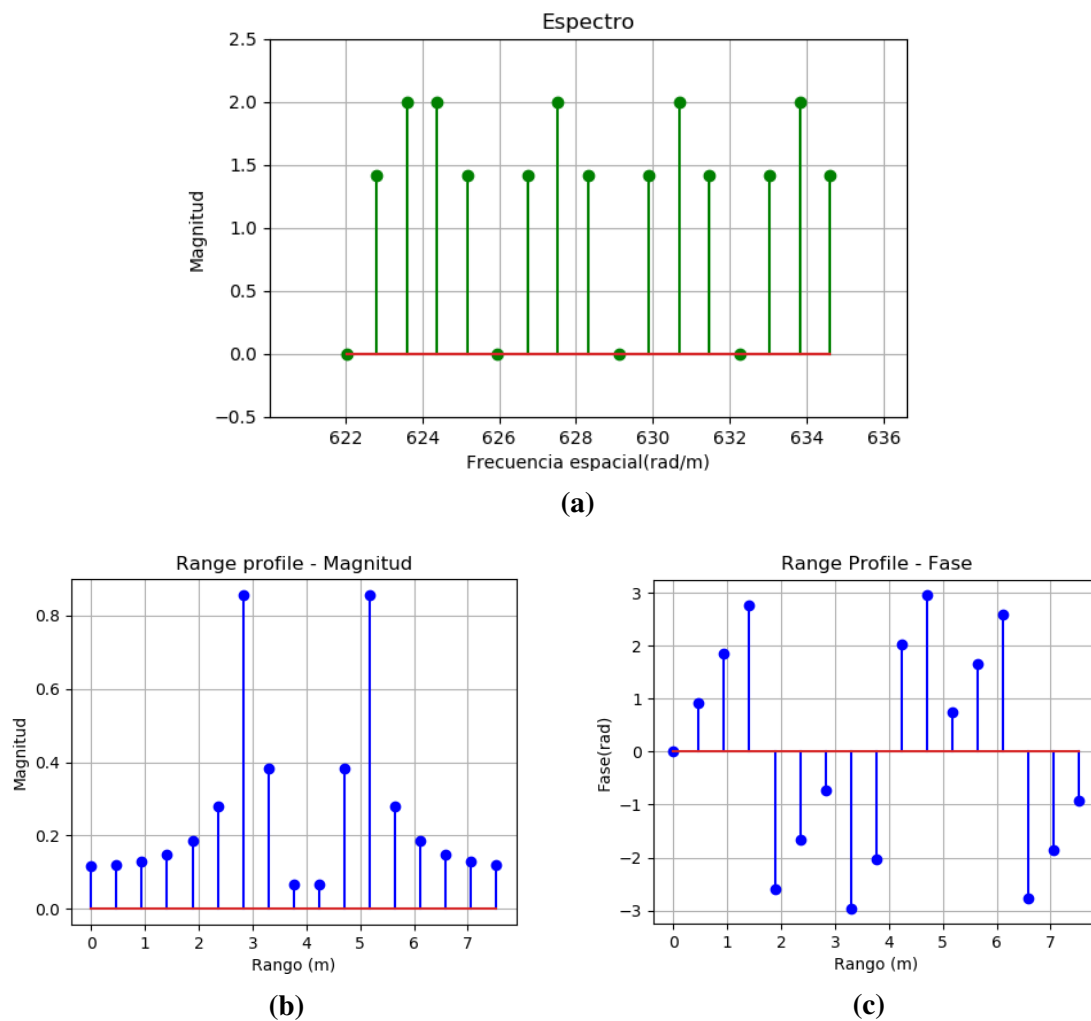


Figura 2.10: Ejemplo de la propiedad de desplazamiento circular de la DFT. En este caso sí se presenta desplazamiento con $m = 3$. (a) Espectro, (b) perfil de Rango en magnitud, (c) perfil de Rango en fase. (Fuente propia)

na puntos de interpolación adicionales los cuales dan la apariencia de una resolución menor en el resultado final. En otras palabras, el proceso de relleno de ceros de una señal es similar al hecho de calcular la DFT a la señal original y luego realizar una interpolación tipo *Sinc* al espectro de la señal del tamaño del *zero-padding*. El término resolución frecuencial del sistema hace referencia a la capacidad de distinguir o discriminar el espectro de 2 señales cuyas frecuencias son cercanas, en el contexto de los radares SAR tiene la misma definición de resolución explicada en la sección 2.2.2, la cual señala que la resolución es la capacidad de distinguir 2 blancos cuando están separados una cierta distancia corta.

La resolución frecuencial del sistema es distinto a la resolución frecuencial Δk , y para mejorarlo se tiene que aumentar más datos a la señal original, mientras que una mejora de Δk se puede obtener con un *zero-padding*. La misma idea se aplica cuando la señal se encuentra en el dominio de la frecuencia, por lo que el relleno con ceros al espectro no

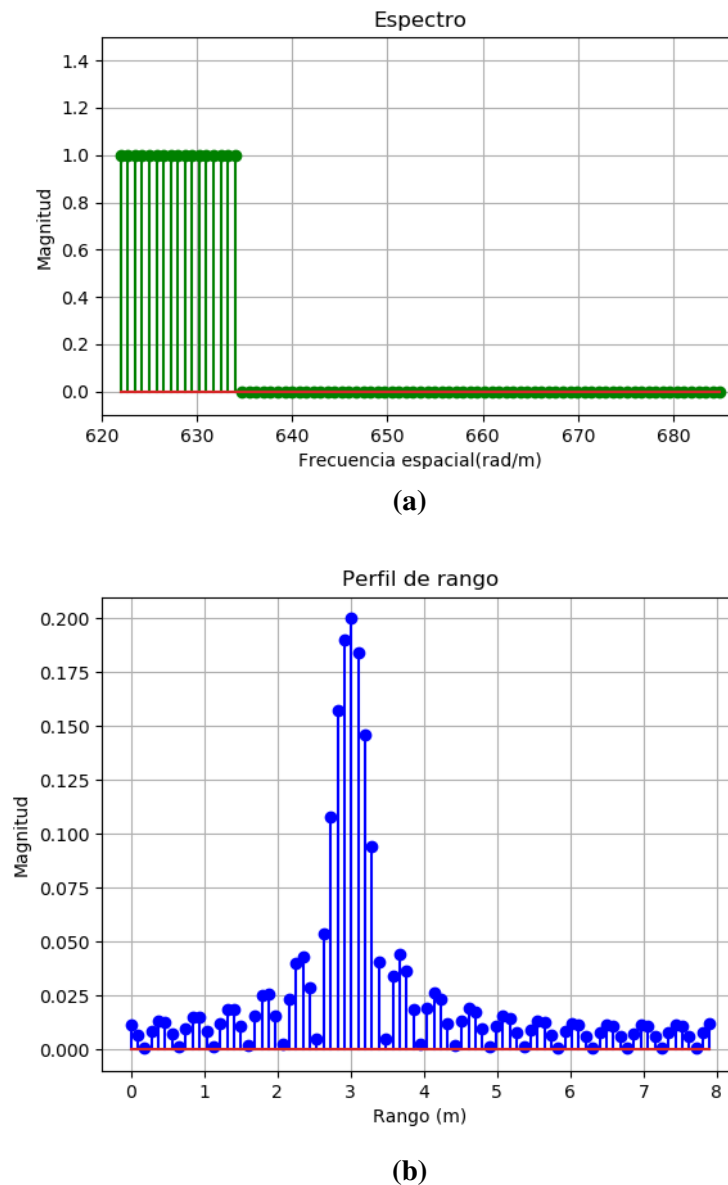


Figura 2.11: Ejemplo de *zero padding*. (a) Espectro con relleno de ceros, (b) Perfil de rango mejor definido. (Fuente propia)

mejora la resolución espacial del sistema.

Como ejemplo se presenta el caso del perfil de rango de 2 blancos puntuales simulados cuyos rangos son 3 m y 3.4 m, el objetivo es poder discriminar la ubicación de sus rangos en el perfil de rango obtenido después de sacar la IDFT al espectro de la señal, en la parte superior de la figura 2.12 se muestra el perfil de rango sin y con relleno de ceros no pudiendo discriminar la ubicación exacta de cada blanco aún cuando se haya realizado el *zero – padding*, para mejorar la resolución y así identificar las posiciones, se incrementa el número de muestras originales L y se obtienen los perfiles de rango respectivos sin y con relleno de ceros (mostrado en la parte inferior de la figura 2.12), como se puede observar ya se pueden discriminar bien las ubicaciones de cada blanco por separado, siendo más fácil la

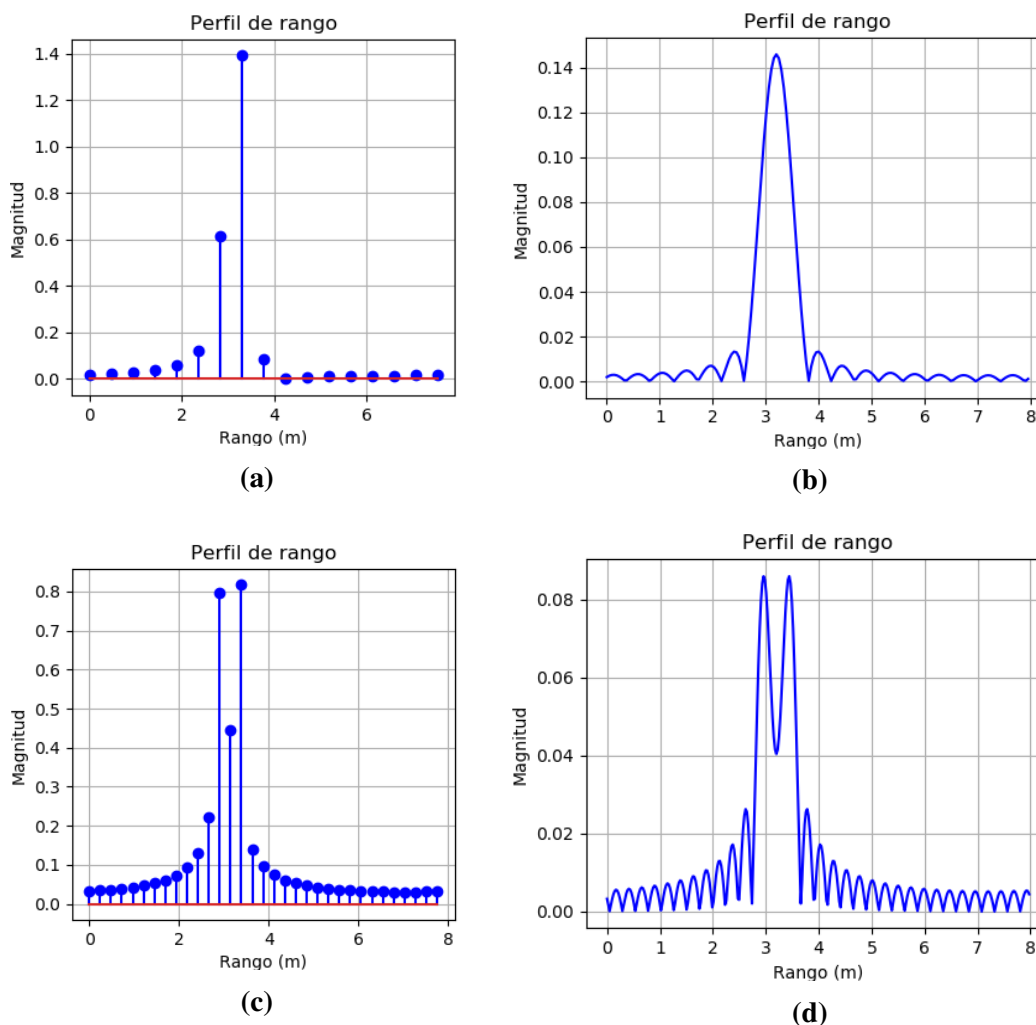


Figura 2.12: Perfil de rango del espectro con un número de muestras L , (a) sin *zero-padding*, (b) con *zero-padding*. Perfil de rango del espectro con un número de datos $L' > L$, (c) sin *zero-padding*, (d) con *zero-padding*. (Fuente propia)

identificación cuando hay relleno de ceros.

Transformada Rápida de Fourier (FFT)

La Transformada Rápida de Fourier (FFT) es una manera eficiente y rápida de evaluar la DFT de una señal. Normalmente para efectuar la DFT se necesitan N^2 operaciones matemáticas, sin embargo al emplear técnicas de FFT como el de Cooley-Tukey, la cantidad de operaciones se pueden reducir al orden de $N \log_2(N)$, cuando N es una potencia de 2, el cual se puede lograr con un *zero-padding* a la secuencia original [16].

Para el cálculo de la FFT se emplea la función *fft* de Python, cuyo resultado se interpreta del siguiente modo: la primera mitad de la secuencia corresponden al lado positivo del espectro, mientras que la otra mitad corresponde al lado negativo, esto se justifica por la propiedad de periodicidad de la DFT; para poder centrar el espectro, es decir, ubicar las frecuencias negativas primero y luego las positivas se puede hacer uso de la función *fftshift*.

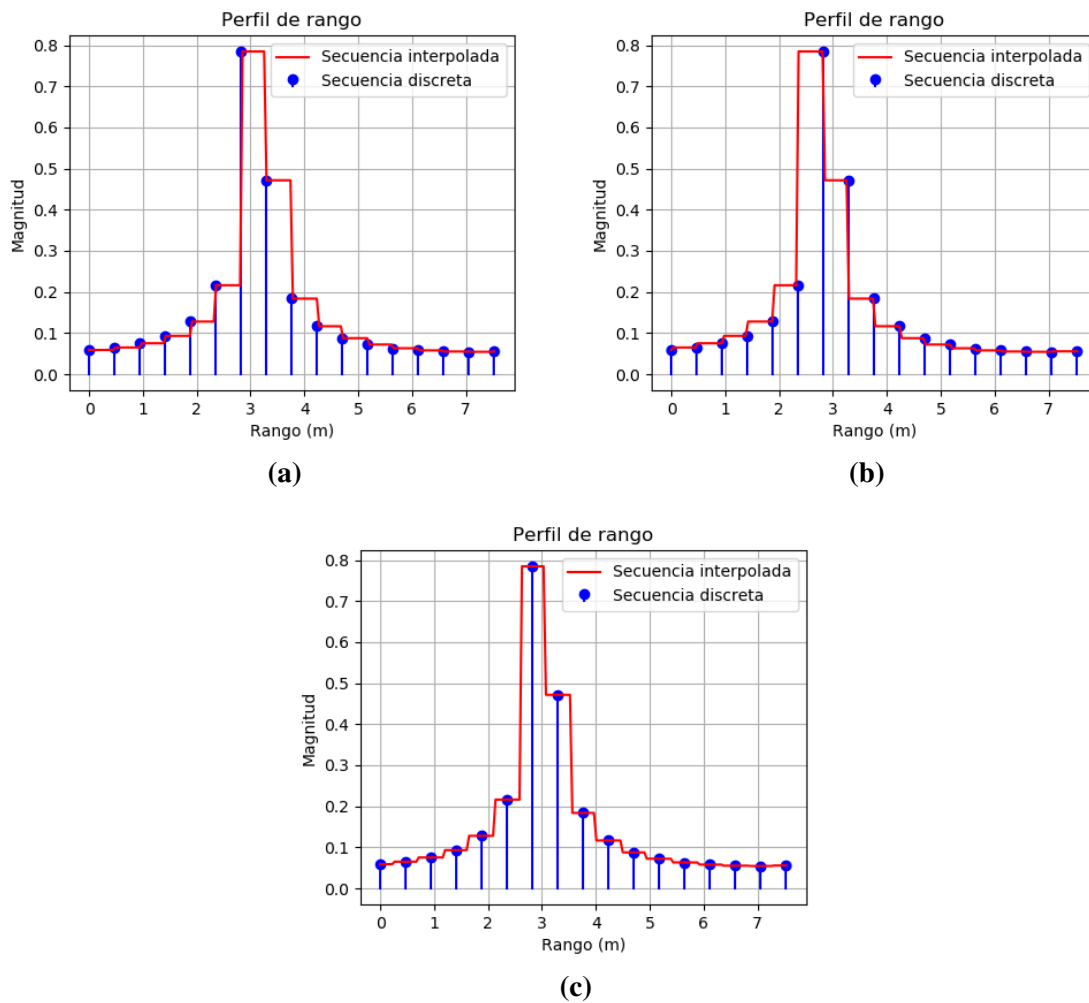


Figura 2.13: Interpolación constante por partes, aplicado a un perfil de rango, en sus 3 variantes: (a) *Previous*, (b) *Next*, (c) *Nearest*. (Fuente propia)

2.3.2 Interpolación

La interpolación es una herramienta matemática que permite calcular los valores desconocidos entre las muestras de una señal discreta. Por ejemplo, si se tiene la señal del perfil de rango $s[n]$ con dominio $r[n]$, y se desea saber el valor de $s[n]$ para un valor de rango entre $r[n_i]$ y $r[n_i + 1]$, la interpolación resuelve este problema. De acuerdo a [18] la interpolación es una operación para convertir una secuencia de datos discretos $s[n]$ en una función continua $s(t)$. Además señala que un importante requerimiento de la función interpolada es que al momento de evaluar en puntos múltiples del periodo de muestreo de la señal T_s , estas tienen que ser iguales a las muestras correspondientes, es decir:

$$s(t)|_{t=n.T_s} = s[n] \quad (2.30)$$

La función continua $s(t)$ permite obtener valores en otros puntos de la señal que no están presentes en $s[n]$. Los valores de t tienen que pertenecer al dominio de la secuencia $r[n]$.

Los tipos de interpoladores definen el tipo de función matemática que tendrá $s(t)$, siendo

uno de ellos los interpoladores locales [18]. Los interpoladores locales, también llamados *spline*, definen una función matemática por partes, es decir, en cada intervalo del dominio $r[n]$, donde cada intervalo comúnmente va de una muestra $r[n_i]$ a la siguiente consecutiva $r[n_i + 1]$. De acuerdo a la función matemática, este interpolador a su vez se divide en las siguientes 3 clases:

1. Interpolador Constante

Es el interpolador más simple, y la función matemática que se usa es una función constante de valor Y_o . La constante Y_o puede tener alguno de los siguientes 3 valores: la muestra inferior $s[n_i]$, la muestra superior $s[n_i + 1]$, o la misma muestra $s[n_i]$ pero con un cambio de intervalo alrededor de ella:

$$s(t) = s[n_i] \quad , \quad (n_i - \frac{1}{2})T_s \leq t < (n_i + \frac{1}{2})T_s \quad (2.31)$$

Dentro de la función de Python *interp1d*, la cual es usada para realizar interpolación de señales, a los 3 tipos mencionados debido al valor de Y_o se les llama interpolador constante tipo *Previous*, *Next* y *Nearest* respectivamente.

En la figura 2.13 se muestra un ejemplo del perfil de rango de la figura 2.7 interpolado con un interpolador constante de los 3 tipos: *Previous*, *Next* y *Nearest*.

2. Interpolador Lineal

Este tipo de interpolador forma la función $s(t)$ simplemente conectando con una línea recta dos muestras consecutivas $s[n_i]$ y $s[n_i + 1]$. En la figura 2.14 (a) se muestra un ejemplo del perfil de rango de la figura 2.7 interpolado linealmente.

3. Interpolador Polinomial

Este tipo de interpolador define un polinomio en cada intervalo. Los polinomios pueden ser de grados distintos en cada intervalo y la respuesta es más suave y sin muchas oscilaciones. Se prefiere hacer esto cuando se tiene una cantidad considerable de puntos de interpolación. Los tipos de interpoladores polinomiales más comunes son la cuadrática y la cúbica. Como ejemplo se muestra en la figura 2.14 el perfil de rango de la figura 2.7 interpolado con un interpolador polinomial por partes cuadrático.

2.3.3 Decimación

También llamado submuestreo, de acuerdo a [18] hacer decimación por M significa crear una secuencia con menor tasa de muestreo, manteniendo una muestra por cada M muestras de la señal original. Los valores de M son enteros positivos mayores a 1. Debido a que esta operación descarta $(M - 1)$ de M muestras, puede causar pérdida o distorsión de la información en la secuencia original, y para entender ello se realiza un análisis en el dominio de la frecuencia.

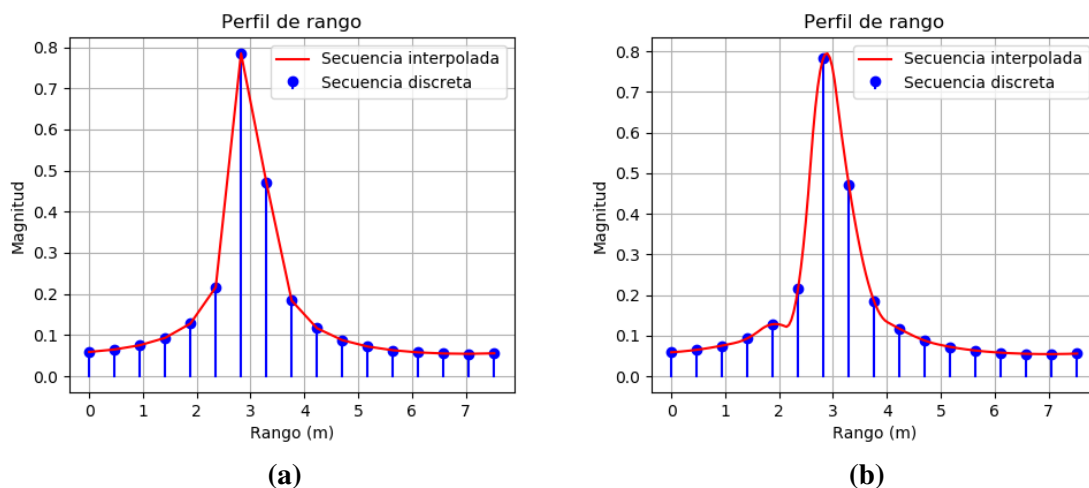


Figura 2.14: (a) Interpolación lineal e (b) interpolación polinomial cuadrática, aplicado a un perfil de rango. (Fuente propia)

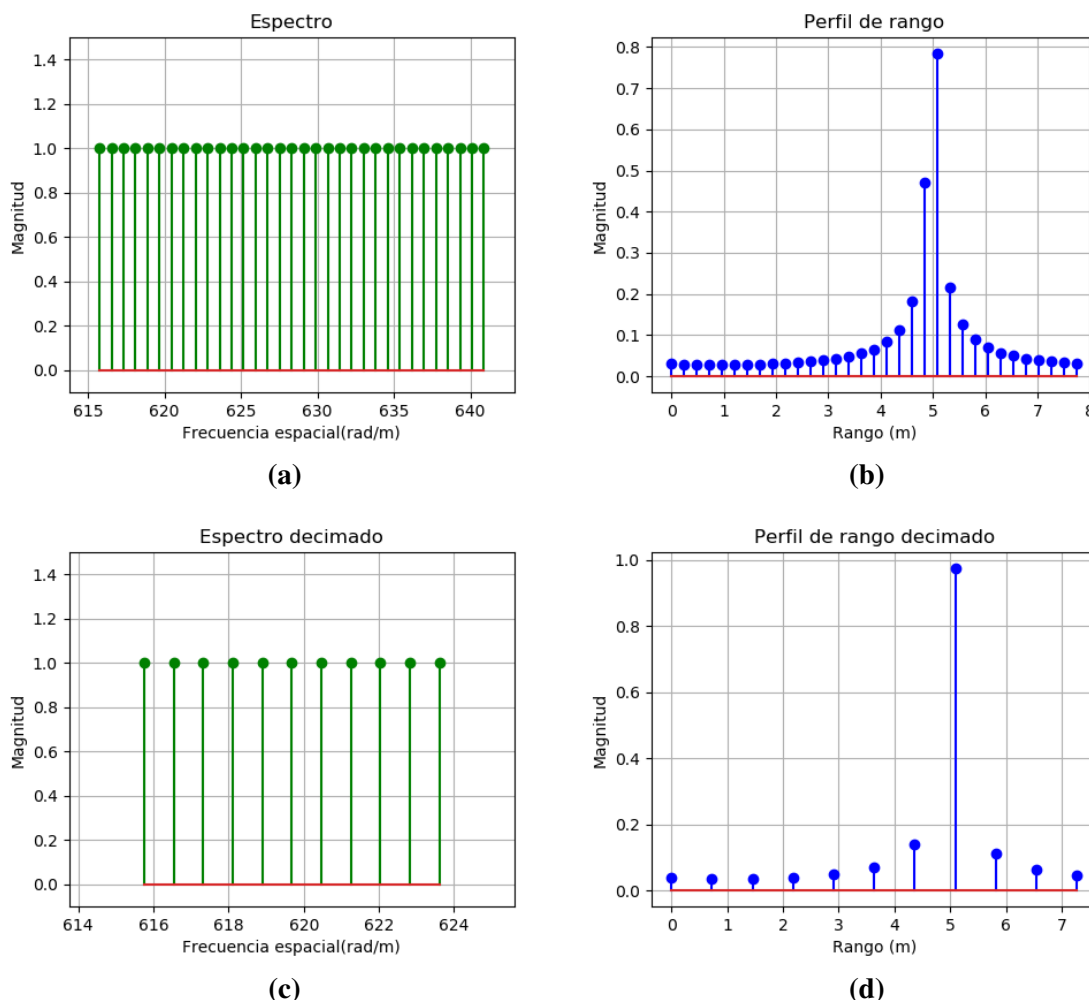


Figura 2.15: Decimación del perfil de rango. (a) Espectro sin decimar, (b) Perfil de rango sin decimar, (c) Espectro decimado, (d) Perfil de rango decimado. (Fuente propia)

Decimación en el dominio de la frecuencia

Una señal decimada en el dominio de la frecuencia tiene la siguiente expresión: [18]

$$S_{MD}(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} S(e^{j(\omega-2\pi k)}) \tag{2.32}$$

La DTFT se usa como referencia, sin embargo, la explicación irá dirigida a la DFT. De la expresión anterior se puede decir que el espectro resultante se compone de la suma de M copias superpuestas del espectro original $S(e^{j\omega})$, donde cada copia está desplazada en frecuencia por un múltiplo entero de $\frac{2\pi}{M}$, es decir, por un múltiplo entero de $\frac{L_k}{M}$ para el caso de la DFT, y el resultado de la suma es extendido en el eje de frecuencia por un factor de M .

Debido a la superposición la señal decimada está propensa a un solapamiento¹², en el cual el espectro de la señal decimada ya no puede recuperar el espectro de la señal original debido a la distorsión que se genera, por lo que habría pérdida de información en el dominio del espacio. Para evitar el solapamiento la señal se puede filtrar antes de efectuar la decimación. El filtrado se realiza con un filtro pasabajo de frecuencia de corte $\omega_c = \pi/M$, para el caso de la DFT la frecuencia de corte sería $\frac{L_k}{2M}$.

En la figura 2.15 se muestra como ejemplo la decimación del perfil de rango por un factor $M = 3$, en la parte superior se observa el perfil de rango y su espectro sin decimar, el número total de muestras es $N_p = 33$; mientras que en la parte inferior se muestra el perfil de rango decimado junto a su espectro, el número total de muestras es de $N'_p = N_p/M = 11$ tal como lo esperado por la decimación. La decimación se obtuvo en el dominio de la frecuencia aplicando previamente un filtro pasabajo al espectro, en forma práctica, se recortó el espectro en 3 partes iguales y se seleccionó solo el primer intervalo para finalmente calcular su IDFT. Un detalle mostrado en la figura es que el periodo de muestreo Δr varía, el cual es el objetivo de la decimación, pero no lo hace el intervalo de observación L_r , el cual se mantiene igual debido a que no hay cambio de Δk , este hecho obedece a las ecuaciones del espectro de la DFT, mostrado en las ecuaciones (2.24) y (2.25).

La decimación se usa durante la implementación del algoritmo RMA para disminuir la cantidad de píxeles de la imagen SAR, así como para reducir el tiempo de procesamiento del algoritmo.

2.3.4 Filtros

Una gráfica de perfil de rango presenta lóbulos laterales, los cuales se hacen más evidentes cuando se le aplica un *zero-padding*, el cual se muestra en la figura 2.16. Del mismo modo que en el perfil de rango, una imagen SAR también posee lóbulos laterales, los cuales pueden distorsionar la imagen final ante la presencia excesiva de estos, por lo que una alternativa para reducirlos es el uso de los filtros.

Los filtros o ventanas reducen la magnitud de los lóbulos laterales indeseados, haciendo que una imagen SAR se vea más suave y mejor definida. El único inconveniente que presenta

¹²Aliasing

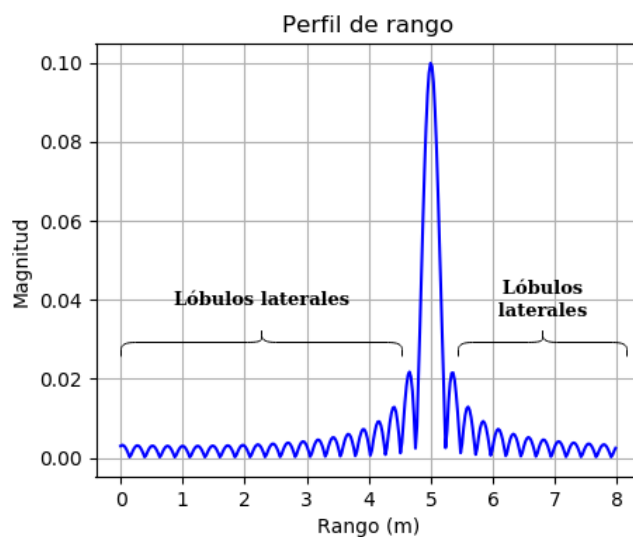


Figura 2.16: Lóbulos laterales presentes en un perfil de rango. (Fuente propia)

es que bajan la resolución de la imagen final. Las funciones descritas anteriormente que se aplican en el dominio de la frecuencia se les llama filtros y cuando se aplica en el dominio del espacio se les llama ventanas, para efectos de explicación se llamarán a ambos términos “ventana”. Los funciones de enventanado más usados para este propósito son:

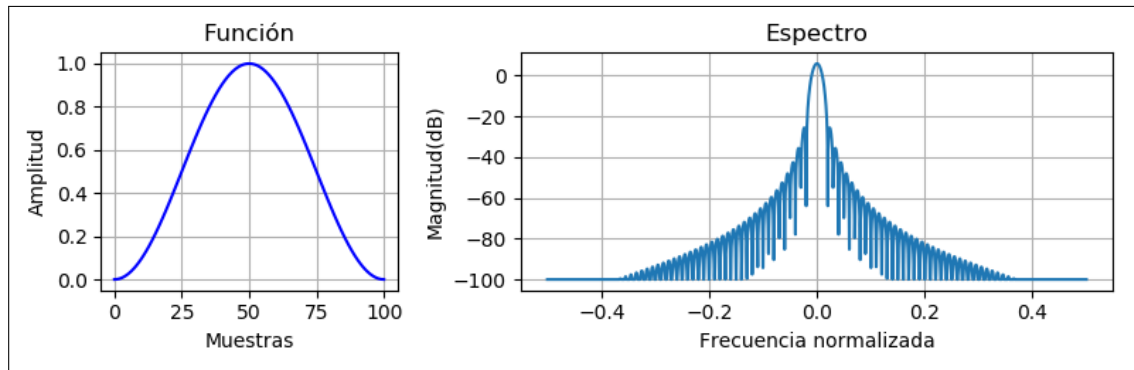
- Ventana de Hanning
- Ventana de Hamming

Tabla 2.1: Características de las funciones de enventanado. [18]

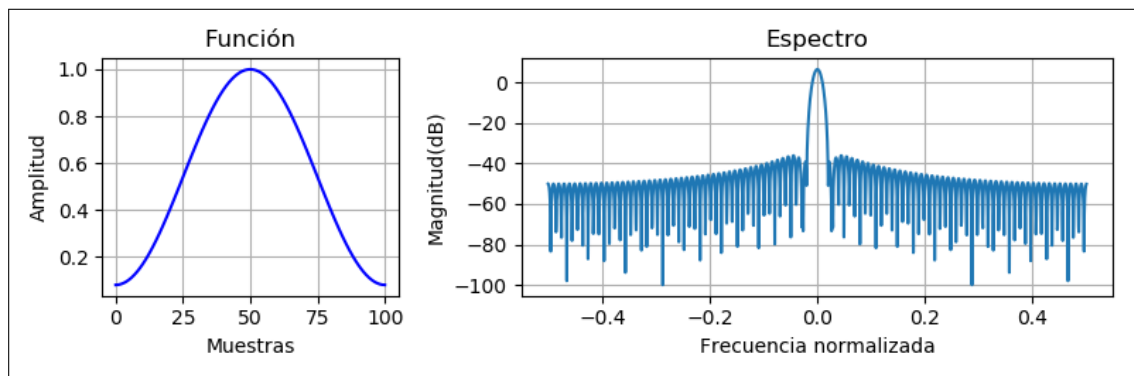
Función de enventanado	Expresión	Ancho del Lóbulo Principal (-3dB)	Máximo SSL(dB)
Hanning	$h[n] = 0.5(1 - \cos \frac{2\pi n}{N-1})$	1.40	-32
Hamming	$h[n] = 0.53836 - 0.46164 \cos \frac{2\pi n}{N-1}$	1.33	-43

En la tabla 2.1 se muestran las características principales de las 2 funciones de enventanado y en la figura 2.17 se muestran sus gráficas respectivas. El término “Máximo SSL(dB)” hace referencia a la diferencia de magnitudes entre el lóbulo principal y el mayor lóbulo lateral expresado en dB. Ambas características de la tabla están referidas al espectro de las ventanas. De acuerdo a la tabla 2.1 la ventana Hamming presenta mejores características, debido a que no baja mucho la resolución (menor ancho de lóbulo principal) y atenúa más los lóbulos laterales (menor valor de SSL).

Como ejemplo, se presenta el caso del perfil de rango de la figura 2.16, a la cual se le aplica las ventanas de Hanning y Hamming. Para ello se multiplica el espectro de la señal con la función de enventanado respectivo. En la figura 2.18 se muestran los resultados, como se puede observar en ella la magnitud de los lóbulos laterales disminuyeron, siendo mayor el efecto en la ventana Hanning, además se observa un aumento del ancho del lóbulo principal,



(a)



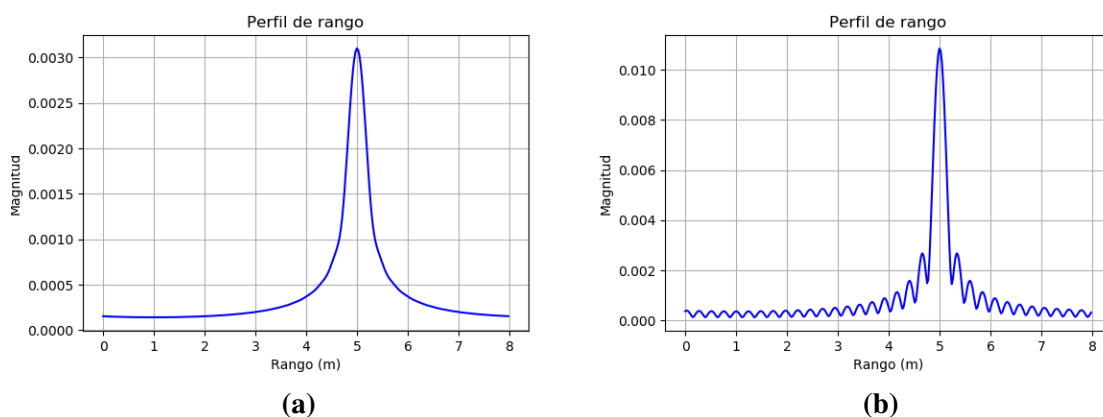
(b)

Figura 2.17: Funciones de enventanado junto a sus espectros. (a) Ventana de Hanning y (b) Ventana de Hamming. (Fuente propia)

en mayor magnitud cuando se usa la ventana de Hanning.

2.4 Algoritmos de formación de imágenes SAR

En esta sección se revisó el estado del arte de los algoritmos a implementar posteriormente, enfocándose principalmente en la 'Implementación del algoritmo'. Los requisitos que deben tener los algoritmos revisados son: aplicables a radares GB-SAR y uso de una forma de onda del tipo SF-CW; esto debido a que ya se tiene la instrumentación con estas características.



(a)

(b)

Figura 2.18: Perfil de rango empleando: (a) Ventana Hanning, (b) ventana Hamming. (Fuente propia)

Tabla 2.2: Parámetros usados en las pruebas.

Frecuencia Central (f_c)	5.7 GHz
Ancho de Banda (B)	600 MHz
Número de frecuencias (N_f)	401
Longitud de escaneo (L)	63 cm
Número de puntos de escaneo (N_p)	22

2.4.1 Revisión del algoritmo FDBP

En la literatura se investigó información acerca del algoritmo FDBP, más que todo relacionado a la implementación y que contenga las condiciones mencionadas anteriormente, encontrándose el artículo [19]. En ella, usan la técnica GB-DInSAR para una aplicación de Terrain Mapping ¹³, y como parte de su procedimiento realizan la formación de imágenes SAR usando el algoritmo FDBP. Respecto al algoritmo, se explican sus pasos, las consideraciones de algunos parámetros de medición, además de mencionar la instrumentación usada.

Instrumentación usada y obtención del histórico de fase

Respecto a la instrumentación empleada, se usa un radar GB-SAR el cual consta de 3 partes principales: una unidad transmisora y receptora, un par de antenas y un riel lineal. La primera parte es la unidad transmisora y receptora de señales del tipo SF-CW, para ello utilizan un analizador de redes (HP 8753D). La segunda parte lo conforman 2 antenas de bocina y la tercera parte es un riel lineal que genera la apertura sintética. El control de la parte mecánica y electrónica es realizada por una computadora.

Los parámetros eléctricos y mecánicos mas importantes que se usan se muestran en la tabla 2.2. Respecto a los valores de los parámetros de la tabla anterior, se puede deducir lo siguiente: la frecuencia central obedece al fenómeno que se quiere estudiar así que la frecuencia de 5.7 GHz posiblemente es usada en aplicaciones de Terrain Mapping, con el ancho de banda dado se puede obtener una resolución en rango de aproximadamente 25 cm, el número de frecuencias repercute en la obtención del Rango no ambiguo (R_U) el cual realizando el cálculo sería de aproximadamente 100m, la apertura sintética determina la resolución en azimut la cual tendría un valor máximo de 4.2m correspondiente al rango máximo teórico R_U , finalmente el número de pasos del riel determina el ángulo de visión máximo de la imagen el cual sería aproximadamente de 27° respecto a una referencia perpendicular el eje del riel.

La obtención del histórico de fase se realiza del siguiente modo. La unidad transmisora y receptora consta principalmente de una sintetizador de microondas y un receptor de micro-

¹³Terrain mapping consiste en realizar mapas topográficos de un terreno dado. (Fuente: <https://www2.gov.bc.ca/>)

ondas para la transmisión y la recepción de la señal respectivamente. El sintetizador genera paso a paso señales continuas a valores discretos de frecuencia barriendo un ancho de banda B . Luego, el sintetizador alimenta la antena transmitiendo la señal y proveyendo de una referencia para la instrumentación receptora, la cual mide las componentes real (I^m) y compleja (Q^m) de la señal recibida. Por otro lado, la apertura sintética es obtenida realizando un barrido a lo largo de un riel horizontal ubicado de tal forma de iluminar el blanco en estudio. Por lo tanto las componentes de la señal recibida, los cuales conforman el histórico de fase, son medidas por cada frecuencia i del ancho de banda B y por cada posición k del riel. El histórico de fase E_{ik}^m presenta la siguiente expresión compleja:

$$E_{ik}^m = I_{ik}^m + jQ_{ik}^m \quad (2.33)$$

Pasos

El procedimiento que se sigue para formar la imagen es el siguiente. El valor en el punto n de la imagen (I_n) es obtenido sumando todas las contribuciones de señal y tomando en cuenta su histórico de fase E_{ik}^m , representadas en las siguientes ecuaciones:

$$I_n = \frac{1}{N_f N_p} \sum_k R_{n,k}^2 F_n^{(k)} \quad (2.34)$$

$$F_n^{(k)} = \sum_i E_{i,k} e^{j \frac{4\pi f_i}{c} (R_{n,k} - R_o)} \quad (2.35)$$

Donde:

- $R_{n,k}$ es la distancia entre el punto n de la imagen y la posición k del riel (u_k).
- R_o es una constante que toma en cuenta el retraso (delay) de los cables.
- c es la velocidad de la luz.

Ambas expresiones se resuelven usando el siguiente procedimiento. La ecuación (2.35) se evalúa como la transformada inversa de Fourier discreta (IDFT) del histórico de fase E_{ik} respecto a la dimensión de la frecuencia i . Lo anterior se realiza por cada posición k del riel. Las muestras de la IDFT son ubicadas en rango equidistantemente cada $\Delta r = c/2B$, así que por cada punto n de la imagen, se selecciona la muestra m de la IDFT, cuya distancia en rango está más cerca a la distancia entre el punto n y la posición k del riel (mostrado en la figura 2.19), lo anterior mencionado se puede entender como una interpolación constante tipo *Nearest*, luego se agrega el valor de esta muestra al valor respecto a la posición anterior del riel $k - 1$, el valor final de I_n se obtiene al iterar todas las posiciones del riel N_p .

En los parámetros de medición se consideran 2 ajustes. La primera es que los rangos de todos los blancos deben ser menores al Rango No Ambiguo (R_U), esto para que la escena de la imagen esté correctamente representada. En la ecuación (2.36) se muestra la expresión de

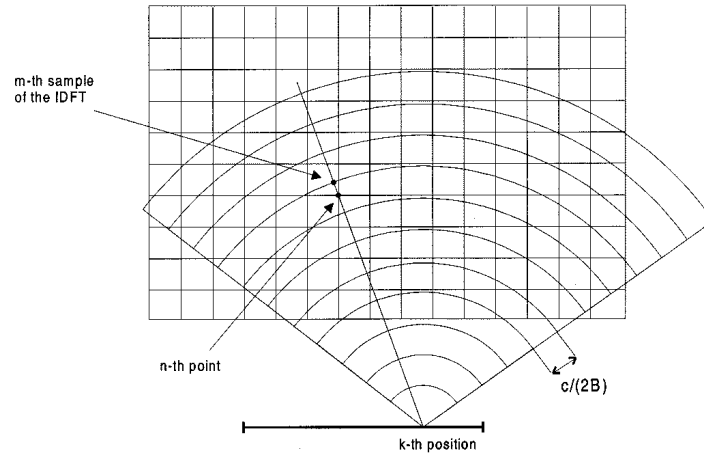


Figura 2.19: Esquema gráfico de la etapa de etapa de interpolación del algoritmo FDBP. (Extraído de [19])

R_U .

$$R_U = \frac{c}{2\Delta f} \quad (2.36)$$

Donde Δf es el paso de frecuencia. Como regla práctica se sugiere que este valor de R_U sea igual a 2 veces la distancia del blanco deseado más distante que el radar pueda detectar.

El segundo ajuste es el muestreo a lo largo del eje del riel el cual debe cumplir las siguientes condiciones:

$$L \geq \frac{c}{2f_c \Delta \theta} \quad (2.37)$$

$$\Delta x \leq \frac{c}{4f_c} \quad (2.38)$$

Apreciaciones

De acuerdo a la revisión hecha a la fuente [19], los aspectos positivos son la explicación del procedimiento usado y de las consideraciones de algunos parámetros de medición, asimismo se detalla la instrumentación usada. Cabe resaltar que la instrumentación usada en esta referencia es similar a la instrumentación que se dispone para la realización de este trabajo: analizador vectorial de redes (VNA) como unidad transmisora y receptora, 2 antenas de bocina y un riel lineal de mayor apertura sintética. Sin embargo, la fuente no lo aplica a un caso de deslizamientos, no hace análisis de las imágenes reconstruidas ni tampoco hace pruebas de la imagen reconstruida con datos simulados.

2.4.2 Revisión del algoritmo RMA

También denominado algoritmo $\omega - k$, este algoritmo se origina de técnicas usadas en el campo del procesamiento de señales sísmicas [12]. Se caracteriza por funcionar en el dominio de la frecuencia y por requerir una interpolación 1D en particular llamada “Interpolación STOLT”.

Se buscó en la literatura información acerca de la implementación del algoritmo RMA y que cumplan las 2 condiciones: aplicable a radares GB-SAR y que emplee una señal del tipo SF-CW. Se encontró una referencia que cumple ambas condiciones en el artículo [20], en el cual se explica la instrumentación usada, los pasos y las pruebas con datos reales. Aparte de ello, en el libro [21] se explican los pasos del algoritmo con un ejemplo sencillo y se realizan pruebas con datos simulados. Una referencia adicional es el libro [17], en el cual se detallan cada paso del algoritmo mostrando sus ecuaciones y explicaciones gráficas.

Instrumentación usada

La instrumentación que se usa en el artículo [20] es un radar GB-SAR el cual está conformado principalmente por un Analizador Vectorial de Redes (VNA) de la marca Rhode & Schwarz ZVB20 para la transmisión y recepción de señales tipo SF-CW, 2 antenas de bocina y de un sistema de posicionamiento lineal para generar la apertura sintética. Una computadora controla la parte eléctrica y mecánica.

Los parámetros eléctricos y mecánicos más importantes que se usan se muestran en la tabla 2.3. Respecto a los valores de los parámetros de esta tabla, se puede resaltar lo siguiente: posee un ancho de banda alto con el cual se obtiene una resolución en rango (Δr) de 3.75 cm el cual es un valor pequeño, con el número de frecuencias dado se puede obtener un valor de $R_U = 19$ m aproximadamente con el cual se puede formar una imagen de rango máximo 19 m y por último presenta una resolución en azimut de 2.34 cm (rango = 100 cm, $f = 10$ GHz). El histórico de fase es obtenido de la misma forma que el caso del algoritmo FDBP ya que usa la misma forma de onda de transmisión: SF-CW, generándose una matriz en 2 dimensiones: frecuencias y posiciones de la apertura sintética.

Pasos

De acuerdo a la referencia [17], el algoritmo RMA consta de 4 pasos:

1. Transformada de Fourier 1D respecto a la dimensión espacial
2. Filtro adaptado ¹⁴
3. Interpolación STOLT
4. Transformada de Fourier inversa 2D

Tabla 2.3: Parámetros que fueron usados en las pruebas con el algoritmo RMA

Frecuencia Central (f_c)	10 GHz
Ancho de Banda (B)	4 GHz
Número de frecuencias (N_f)	512,1024,2048
Longitud de escaneo (L)	128 cm
Número de puntos de escaneo (N_p)	64

¹⁴También llamado *Matched filter* en la literatura.

En las referencias [20], [21], se presenta la implementación práctica de estos 4 pasos, los cuales se pasarán a describir resumidamente.

- **Transformada de Fourier 1D respecto a la dimensión espacial**

Se calcula la 1D-DFT del histórico de fase respecto a la componente espacial, osea en la dirección de azimut. Esto convierte el histórico de fase $s(x_n, \omega)$ en $s(k_x, \omega)$. Adicionalmente, se realiza el siguiente cambio $k_r = \omega/c$, resultando en $s(k_x, k_r)$.

- **Filtro adaptado**

En primer lugar se genera el factor del filtro adaptado :

$$s_{mf}(k_x, k_r) = e^{(jR_s \sqrt{k_r^2 - k_x^2})} \quad (2.39)$$

Donde R_s es la distancia en rango desde la apertura sintética hasta el centro de la escena de la imagen.

Luego, se multiplica el resultado del primer paso con el factor del filtro adaptado :

$$s_{matched}(k_x, k_r) = s(k_x, k_r) \cdot s_{mf}(k_x, k_r) \quad (2.40)$$

- **Interpolación STOLT**

Es el paso crítico dentro de todo el algoritmo. Se transforma la matriz $s_{matched}(k_x, k_r)$ del dominio (k_x, k_r) al dominio (k_x, k_y) , mediante el siguiente cambio de variable:

$$k_y = \sqrt{k_r^2 - k_x^2} \quad (2.41)$$

Con esto, una interpolación 1D es aplicado a lo largo de todos los valores de k_r para mapearlos en k_y resultando en la matriz interpolada Stolt $s_{st}(k_x, k_y)$.

- **Transformada de Fourier inversa 2D**

En este último paso, se aplica una transformada inversa de Fourier en ambas dimensiones (2D-IDFT) a la matriz $s_{st}(k_x, k_y)$ para obtener la matriz $S(X, Y)$ que representa la imagen SAR.

Pruebas con datos simulados

En la siguiente referencia [21], se prueba el algoritmo RMA con datos simulados, es decir datos generados por un programa de computador, en cuya fuente también se indica como se generaron estos datos. Se prueban 2 casos. El primero es de un blanco puntual con coordenadas de azimut $x_{t1} = 0$ ft y de rango $y_{t1} = -10$ ft cuyo resultado se muestra en la figura 2.20, un comentario de este resultado es que se puede apreciar la reconstrucción en el lugar correcto, se pueden notar además la presencia de lóbulos laterales, y el eje del rango tiene valores negativos debido a que se asumió la geometría dada en [17]. El segundo caso es de 3 blancos puntuales con coordenadas:

$$(x_{t1}, y_{t1}) = (3, -10)\text{ft}, \quad (2.42)$$

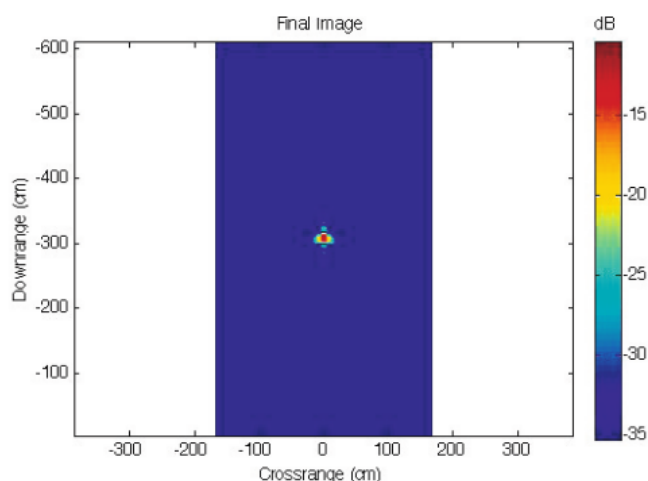


Figura 2.20: Reconstrucción de un blanco puntual simulado usando el algoritmo RMA. (Extraído de [21])

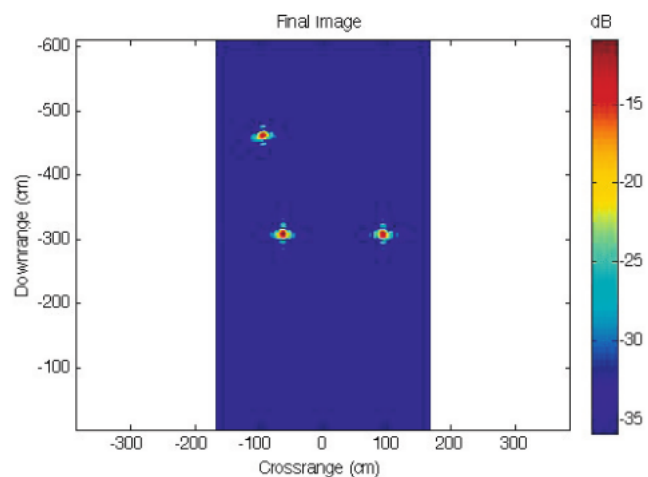


Figura 2.21: Reconstrucción de 3 blancos puntuales simulados usando el algoritmo RMA. (Extraído de [21])

$$(x_{t2}, y_{t2}) = (-3, -15)\text{ft}, \quad (2.43)$$

$$(x_{t3}, y_{t3}) = (-2, -10)\text{ft}, \quad (2.44)$$

El resultado del segundo caso se muestra en la figura 2.21. También se puede observar la correcta reconstrucción. Un detalle a notar después de mostrar ambas pruebas es que el tamaño de la imagen en azimut se limita a la longitud de la apertura sintética, osea a 8 ft (2.4 m). En la fuente [17] también ocurre lo mismo, se realizan pruebas cuyo tamaño de la imagen en azimut es menor o igual a la apertura sintética.

Pruebas con datos reales

En la referencia [20] se realizan pruebas con datos reales adquiridos de diversos blancos como: placa plana, cilindro, corner reflector, un modelo a escala de avión, entre otros. La imagen reconstruida que resalta es del avión, el cual se muestra en la figura 2.22, se le

acompaña una imagen óptica del blanco para comparar e identificar el objeto reconstruido. Sin embargo no se muestran valores de los ejes coordenados para analizar la ubicación del blanco reconstruido.

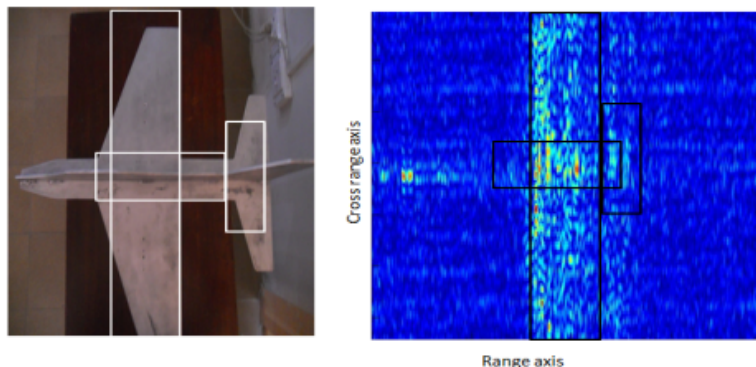


Figura 2.22: Reconstrucción de un blanco real usando el algoritmo RMA. Izquierda: blanco, Derecha: imagen SAR reconstruida. (Extraído de [20])

Apreciaciones

De la revisión del paper, se pudo obtener información para la implementación del algoritmo RMA en 3 fuentes distintas, más que todo sobre la instrumentación usada y los pasos detallados. También se obtuvo información de pruebas con datos simulados y una explicación de la generación de los mismos.

Asimismo, se obtuvo información acerca de la realización de pruebas básicas con datos reales de blancos de formas geométricas básicas, pero sin entrar mucho en el análisis de las mismas. Lo que se pudo notar sin embargo es que las 3 fuentes se limitan a reconstruir imágenes cuyo tamaño de la imagen en azimut es menor o igual al tamaño de la apertura sintética, lo cual es una limitación ya que se busca reconstruir imágenes grandes cuyo tamaño en azimut sea mucho mayor al tamaño de la apertura sintética.

2.4.3 Revisión del algoritmo DLIP

Como se mencionó en la sección introductoria se implementará el algoritmo DLIP, en el cual se resolverá el problema inverso usando alguna técnica de Problemas de Inversión. Para ello se buscó en la bibliografía algún trabajo referencial encontrándose la siguiente fuente [22]. En esta referencia se puede encontrar el planteamiento del problema directo usando un modelo de señal dado y el modelo *Exploiding source*, los cuales servirán para el desarrollo posterior del algoritmo.

Planteamiento del Problema Directo

El problema directo consta de obtener el histórico de fase a partir del conocimiento de la ubicación y la reflectividad de cada uno de los blancos. En la referencia [22], se realiza el planteamiento del problema directo como parte de los pasos del algoritmo *SAR migration*. A

continuación se explicará este punto. En primer lugar, se asume un radar SAR con un patrón de radiación omnidireccional el cual transmite un pulso $p(t)$ desde la ubicación $(0, y, 0)$ hacia el blanco (mostrado en la figura 2.23). El blanco consiste de N blancos puntuales, modelado con el modelo *Exploiding source*¹⁵, cada uno con una posición $(0, y_i, z_i)$ y reflectividad σ_i . A continuación, la señal recibida es modelada del siguiente modo:

$$u(y, t) = \sum_{i=1}^N \sigma_i p\left(t - 2 \frac{\sqrt{(y - y_i)^2 + z_i^2}}{c}\right) \quad (2.45)$$

Donde c es la velocidad de la luz. En este modelo se ignora el término $1/R^2$. A continuación se toma una transformada de Fourier respecto a la variable t , obteniéndose la señal recibida en el dominio de la frecuencia:

$$u(y, \omega) = \tilde{p}(\omega) \sum_{i=1}^N \sigma_i e^{-j2\frac{\omega}{c} \sqrt{(y - y_i)^2 + z_i^2}} \quad (2.46)$$

Los pasos del algoritmo aun continúan pero la última ecuación refleja ya el planteamiento del problema directo para el caso particular de este trabajo de tesis, debido a que el histórico de fase adquirido con los instrumentos está en el dominio (y, ω) . Observando la ecuación (2.46) se puede notar que conociendo las posiciones (y_i, z_i) y las reflectividades (σ_i) de los blancos, se puede obtener el histórico de fase $u(y, \omega)$ simulado. En la fuente [21], utilizan una ecuación similar a la ecuación (2.46) para obtener el histórico de fase simulado, mostrado en la ecuación (2.47). El cambio que hicieron fue la supresión del término $\tilde{p}(\omega)$.

$$s(x_n, \omega) = \sum_{i=1}^N a_{ti} e^{-j2\frac{\omega}{c} \sqrt{(x_n - x_{ti})^2 + y_{ti}^2}} \quad (2.47)$$

Donde a_{ti} es la reflectividad y (x_{ti}, y_{ti}) son las ubicaciones de los blancos puntuales. El radar

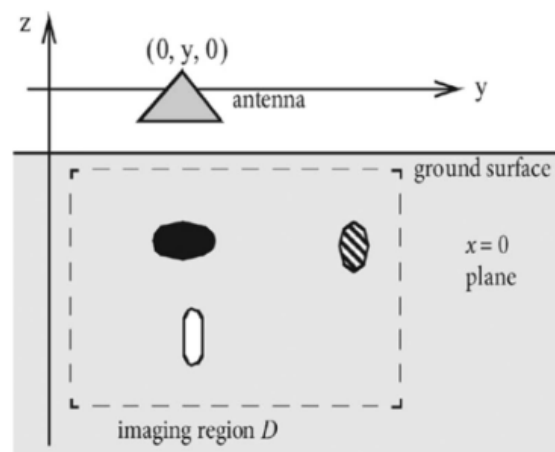


Figura 2.23: Geometría asumida para explicar el procedimiento del algoritmo SAR migration. (Extraído de [22])

¹⁵El modelo *Exploiding source*, presenta las siguientes características principales de acuerdo a [22]: es independiente de la onda incidente, explota y se propaga absolutamente hacia el receptor a lo largo de la dirección de la onda incidente y no hay interacción entre las fuentes.

GB-SAR está ubicado en un plano cartesiano $X - Y$, con la apertura sintética en el eje x . Por lo tanto por temas de simplicidad se usará la ecuación (2.47) como planteamiento del problema directo .

Técnicas para resolver problemas de Inversión

Después de plantear el problema directo como un sistema lineal, se intentará resolver el problema inverso usando los concepto de matriz inversa y pseudoinversa, cuyos principios matemáticos básicos se explican a continuación.

Sea un sistema de ecuaciones lineales de la forma:

$$Ax = b \quad (2.48)$$

Si A es una matriz cuadrada no singular, la solución del sistema se obtiene tomando la inversa de la matriz A : $x = A^{-1}b$. Sin embargo si la matriz A no es una matriz cuadrada, no se podrán obtener soluciones únicas y exactas, solo soluciones aproximadas. Para obtener estas soluciones aproximadas se recurre al concepto de pseudoinversa de la matriz A .

La obtención de la pseudoinversa de A depende del rango de la matriz A y se presentan 2 casos. El primer caso se da cuando la matriz A tiene rango completo ¹⁶. Se dice que la matriz A de tamaño $m \times n$ es de rango completo si su rango es el mínimo valor entre m y n :

$$r = \text{mín}(m, n) \quad (2.49)$$

En este caso, se presentan 2 subcasos, uno para los sistemas subdeterminados y el otro para los sistemas sobredeterminados: [23]

- **Sistemas subdeterminados** ($m < n$)

Los sistemas subdeterminados presentan más incógnitas que ecuaciones, presenta además un rango $r(A) = m$ y generalmente hay infinitas soluciones, por lo que se escoge aquel que tiene la mínima norma euclidiana . Es decir, se minimiza $|x|$ sujeto a la condición $Ax = b$. La solución se obtiene usando el método de los multiplicadores de Lagrange, el cual da por resultado la expresión de la pseudo-inversa para este subcaso:

$$A^\dagger = A^T(AA^T)^{-1} \quad (2.50)$$

- **Sistemas sobredeterminados** ($m > n$)

Los sistemas sobredeterminados presentan más ecuaciones que incógnitas, tienen un rango $r(A) = n$ y generalmente no es posible encontrar soluciones, por lo que se escoge aquella solución que presente el mínimo error. Por lo tanto, la solución es aquella que minimice este error $|b - Ax|$. La solución se obtiene empleando el método de los mínimos cuadrados, el cual da por resultado la expresión de la pseudo-inversa para

¹⁶Full Rank

este subcaso:

$$A^\dagger = (A^T A)^{-1} A^T \quad (2.51)$$

La solución general para ambos casos es:

$$x = A^\dagger \cdot b \quad (2.52)$$

El segundo caso, se da cuando la matriz A tiene rango deficiente ¹⁷, es decir cuando el rango de la matriz A es menor al mínimo valor entre m y n , osea:

$$r(A) < \min(m, n) \quad (2.53)$$

La pseudoinversa para este caso se calcula con la técnica de Descomposición en Valores Singulares (SVD). En primer lugar se calcula la SVD de la matriz A :

$$A_{m \times n} = U_{m \times m} S_{m \times n} V_{n \times n}^T \quad (2.54)$$

Donde S es una matriz diagonal $m \times n$ del mismo tamaño que la matriz A , de la forma:

$$S = \begin{bmatrix} \sigma_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \sigma_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \sigma_p & 0 \end{bmatrix} \quad (2.55)$$

Con $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0$, donde $p = \min\{m, n\}$. Los valores σ_i son los valores singulares de la matriz A .

En segundo lugar, se calcula la pseudoinversa usando la siguiente ecuación: [24]

$$A^\dagger = V S^{-1} U^T \quad (2.56)$$

Donde S^{-1} tiene la siguiente forma para los valores singulares diferentes de cero:

$$S^{-1} = \begin{bmatrix} 1/\sigma_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1/\sigma_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1/\sigma_p & 0 \end{bmatrix} \quad (2.57)$$

Si hay algún valor singular igual a cero, se completa con cero en su lugar correspondiente.

Finalmente, la solución de mínima longitud presenta la siguiente expresión:

$$x = A^\dagger b \quad (2.58)$$

2.4.4 Criterios de comparación de algoritmos

Se buscó en la literatura los criterios para comparar algoritmos de formación de imágenes con radares GB-SAR, encontrándose 2 referencias. La primera referencia [25] emplea 2 primeros criterios de comparación: Calidad de enfoque y ámbito de aplicación, mientras que

¹⁷Rank Deficient

la segunda referencia [26] emplea 3 criterios de comparación: Ámbito de aplicación, coste computacional y precisión en la medición del desplazamiento. A continuación, se explican cada uno de estos criterios de comparación.

Calidad de enfoque

La calidad de enfoque ¹⁸ se obtiene evaluando los perfiles de rango y azimut del blanco reconstruido y calculando sus parámetros principales (mostrado en la figura 2.24): resolución en rango y azimut (medidos a -3 dB del máximo del lóbulo principal), el factor PSLR ¹⁹ y el factor ISLR ²⁰. El parámetro PSLR es la relación entre la amplitud del mayor lóbulo lateral y la amplitud del lóbulo principal, mientras que el parámetro ISLR es la relación entre la energía de los lóbulos laterales (zonas oscuras en la figura 2.24) y la energía del lóbulo principal (zona clara en la figura 2.24). Se realizaron 2 pruebas para evaluar este criterio. La primera prueba consistió en reconstruir un blanco puntual simulado y calcular los parámetros del perfil de rango. Las coordenadas polares del blanco son (r, ϕ) . El perfil de rango se obtiene por cada valor de ϕ y variando los valores de r . A partir de la gráfica del perfil de rango se obtienen tres parámetros: resolución, PSLR e ISLR. Este procedimiento se repite para cada algoritmo. Finalmente se compara numéricamente los parámetros obtenidos con cada algoritmo, tanto para campo cercano como campo lejano.

La segunda prueba se realiza con datos reales, el cual consiste en reconstruir un blanco real y obtener los perfiles de rango y azimut de las zonas de la imagen con mayor reflectividad. Luego se compara gráficamente los perfiles obtenidos con los distintos algoritmos,

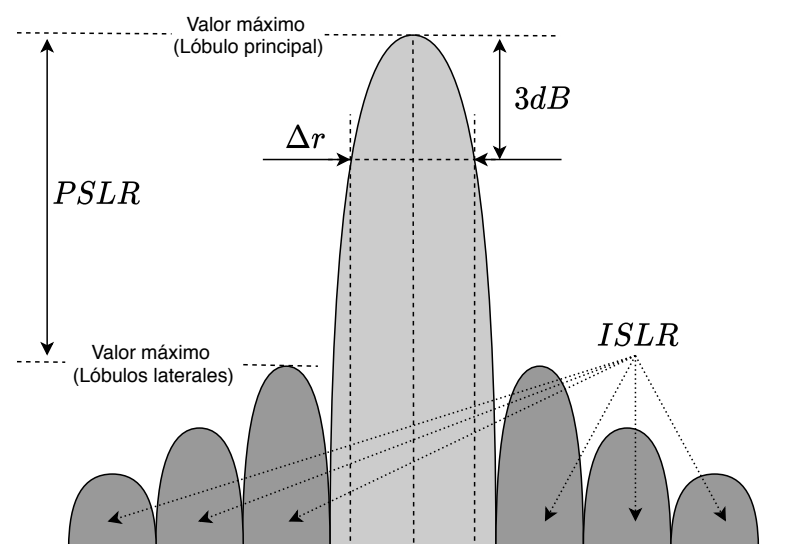


Figura 2.24: Parámetros de un perfil de rango/azimut para el caso de un blanco puntual simulado: resolución Δr , PSLR e ISLR. (Adaptado de [27])

¹⁸Focusing quality

¹⁹Peak to Sidelobe Ratio

²⁰Integrated to Sidelobe Ratio

tomando como base el perfil obtenido con el algoritmo que genera menores aproximaciones en su modelo de señal.

Ámbito de aplicación

El ámbito de aplicación ²¹ se obtiene de las restricciones de las ecuaciones que rigen cada algoritmo, se determina calculando los límites geométricos de la escena de la imagen que puede reconstruir el algoritmo, límites en ambas coordenadas ya sean cartesianas (x, y) o polares (r, ϕ) . Esto permite determinar si el algoritmo puede reconstruir imágenes solo para campo cercano, campo lejano o ambos. Adicionalmente, según [25] se analizan los errores de los algoritmos que poseen aproximaciones en su modelo de señal, por ejemplo errores de posición (range migration) o de fase, para así determinar el ámbito de aplicación en las zonas donde los errores son menores.

Coste computacional

En este criterio se evalúa el factor de coste computacional, el cual es la cantidad de recursos que requiere un algoritmo para ejecutarlo, ya sea tiempo o memoria. El procedimiento que se sigue es la obtención del coste computacional teórico y real, y luego la realización de una comparación numérica.

El coste computacional teórico se obtiene evaluando la complejidad computacional de cada paso del algoritmo y expresándolos en términos de la notación ‘Big-O’²². Por otra parte, el coste computacional real se determina mediante una simulación calculando el tiempo de procesamiento del algoritmo en 2 casos: campo cercano y campo lejano.

De la revisión de la literatura usando el parámetro de coste computacional se observó que el tamaño de las imágenes de prueba no eran iguales para todos los algoritmos, y que se podrían obtener resultados más confiables del coste computacional real si todas las imágenes fueran del mismo tamaño.

Precisión en la medición de los desplazamientos

Bajo el criterio de precisión en la medición de los desplazamientos, se evalúa el factor de precisión en la medición de los desplazamientos, realizándose una comparación numérica y gráfica, tomándose para ello como referencia el factor proveniente de uno de los algoritmos.

El procedimiento para obtener este factor se ilustra en la Figura 2.25 y explica a continuación. En primer lugar se formaron las imágenes SAR usando un algoritmo de formación de imágenes y datos reales adquiridos con el radar GB-SAR, se generaron un total de 60

²¹ Application Scope

²² Big-O es una métrica usado en ciencias de la computación para describir la eficiencia o complejidad de un algoritmo

imágenes para este caso. Luego se obtienen los mapas de desplazamientos procesando las imágenes SAR con la técnica interferométrica PS ²³. Finalmente se escogen unas cotas de prueba del mapa de desplazamientos, seis en este caso, para analizar su evolución temporal, obteniéndose como resultado final gráficas de ‘Desplazamientos vs Tiempo’ por cada cota. Este procedimiento se repite para cada algoritmo en evaluación.

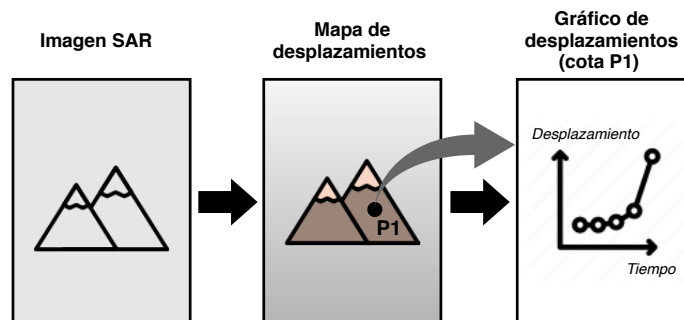


Figura 2.25: Procedimiento para obtener las gráficas de ‘Desplazamiento vs Tiempo’, cada gráfica se obtiene respecto a una cota en específico (P1). (Fuente propia)

La comparación se realiza gráfica y numéricamente. La comparación gráfica involucra evaluar la similitud entre las gráficas de desplazamientos obtenidos con cada algoritmo y una gráfica de referencia obtenido con el algoritmo que no posee simplificaciones en su modelo de señal. La comparación numérica se realiza calculando valores promedios y desviaciones estándar de cada gráfica de desplazamiento.

Como comentario, se podría emplear la técnica GB-InSAR en reemplazo de la técnica PS, la cual se explicó en la sección 1.2, debido a que realiza la misma función de obtener mapas de desplazamientos a partir de imágenes SAR.

Calidad de reconstrucción de la imagen

El criterio de reconstrucción de la imagen está basado en la obtención de los errores en magnitud y fase de las imágenes SAR reconstruidas. Los errores que se usarán son: el error cuadrático medio (MSE), la raíz cuadrada del error cuadrático medio (RMSE), el error absoluto medio (MAE) y el error máximo (ME).

Se usará la siguiente notación. Sea I una imagen SAR obtenida con algún algoritmo de formación de imágenes y sea \hat{I} la imagen original definida teóricamente con la cual se obtiene el histórico de fase simulado. Por otro lado, sea N_x y N_y el número de píxeles de la imagen SAR en los ejes x e y respectivamente. Con esta notación se expresan las fórmulas de los errores:

1. Error cuadrático medio (MSE)

²³Persistent Scatter es una técnica de sensorado remoto que tiene la capacidad de medir y monitorear desplazamientos de superficies de tierra en el tiempo.

El error MSE mide el promedio de los errores cuadráticos individuales. No presenta valores negativos y es cero para una reconstrucción perfecta de la imagen. Una ventaja de este error es que resalta valores inesperados de los píxeles de la imagen. Está definido por la siguiente expresión:

$$MSE = \frac{1}{N_x N_y} \sum_{j=1}^M \sum_{k=1}^N (I_{j,k} - \hat{I}_{j,k})^2 \quad (2.59)$$

2. Raíz cuadrada del error cuadrático medio (RMSE)

El error RMSE es la raíz cuadrada del error MSE. Se crea para hacer que el orden del error esté en el mismo orden que los valores de los píxeles de la imagen. Está definido por:

$$RMSE = \sqrt{\frac{1}{N_x N_y} \sum_{j=1}^M \sum_{k=1}^N (I_{j,k} - \hat{I}_{j,k})^2} \quad (2.60)$$

3. Error absoluto medio (MAE)

El error MAE es el promedio de los valores absolutos de los errores individuales. Tiene un comportamiento lineal ya que todos los errores individuales son pesados igualmente en el promedio. Está dado por:

$$MAE = \frac{1}{N_x N_y} \sum_{j=1}^M \sum_{k=1}^N |I_{j,k} - \hat{I}_{j,k}| \quad (2.61)$$

4. Error máximo (ME)

El error ME indica el máximo error absoluto de toda la imagen, esta dada por la siguiente expresión:

$$ME = \max |I_{j,k} - \hat{I}_{j,k}| \quad (2.62)$$

2.5 Lenguajes de programación para implementación de algoritmos

De acuerdo a la información revisada, los lenguajes de programación más usados para implementar los algoritmos de formación de imágenes SAR son Python y Matlab. Entre estos 2 lenguajes, Python tiene las siguientes ventajas: es de código abierto permitiendo instalarlo de manera gratuita en varios sistemas operativos, tiene una sintaxis sencilla, presenta varios paquetes (librerías) para realizar procesamiento de señales e imágenes y tiene buen soporte por la comunidad de desarrolladores y usuarios. Por estas razones se decidió usar Python en este trabajo.

A continuación se describe este lenguaje y se detallan los paquetes usados. Python es un lenguaje de programación interpretado y multiparadigma, que se caracteriza por poseer una sintaxis sencilla que facilita la programación. Python agrega librerías, a los cuales se les llama paquetes, en la cabecera de los programas para usar varias de sus funciones. Los

paquetes que se usarán en este trabajo son los que están relacionados al procesamiento de señales, álgebra lineal y la realización de gráficas, estas son:

- **SciPy:** Es un paquete usado para computación científica, el cual incluye paquetes de procesamiento de señales e imágenes.
- **Numpy:** Es un paquete usado para realizar operaciones con arreglos y matrices.
- **Matplotlib:** Es un paquete usado para realizar gráficas.

Respecto a la instalación de Python, el programa de instalación se puede descargar de su sitio oficial de manera gratuita <https://www.python.org/downloads/>, hay instaladores para Windows, Mac OS y Linux. Sin embargo, en las últimas versiones de Mac OS y Linux (Ubuntu), Python ya viene instalado por defecto. Hay 2 versiones de Python que se pueden usar: Python 2 y Python 3, sin embargo Python 2 se quedó sin soporte por parte de los desarrolladores desde el año 2020. En este trabajo se usó la versión de Python 3 en un sistema operativo Linux (Ubuntu). El IDE que se usó fue Spyder debido a que está orientado al análisis de datos, su interfaz gráfica de usuario (GUI) es similar a Matlab con ventanas de ayuda, directorio actual, visualizador de variables, consola, entre otros; y ya viene instalado con los paquetes de análisis de datos más importantes como SciPy, Numpy y Matplotlib.

CAPÍTULO III DESCRIPCIÓN DEL SISTEMA DE ADQUISICIÓN DE DATOS

Un sistema de formación de imágenes con radares SAR está conformado por 2 etapas principales: la adquisición de los datos y la implementación del algoritmo, el cual se muestra en la figura 3.1. En el presente capítulo se explica detalladamente la primera etapa, es decir, la “Adquisición de los datos”, y en el capítulo IV se desarrolla la segunda etapa.

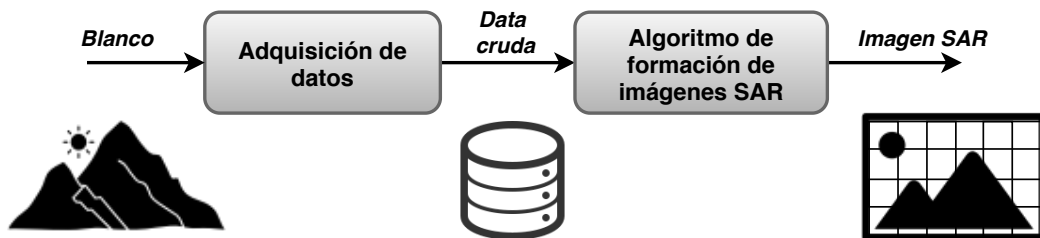


Figura 3.1: Diagrama de flujo del sistema escogido de formación de imágenes con radares SAR. (Fuente propia)

El sistema de adquisición de datos se encarga de obtener los datos crudos, también llamado histórico de fase, de un radar GB-SAR. En este trabajo se emplean los datos crudos del radar GB-SAR ‘IGP-ROJ’, perteneciente al Instituto Geofísico del Perú (IGP) - sede Radio Observatorio de Jicamarca (ROJ), quienes lo diseñaron y construyeron para realizar estudios de deslizamientos de tierra. En las siguientes secciones se describen las partes principales de este radar, su funcionamiento y la forma de adquirir los datos. Asimismo se explican los criterios para seleccionar los parámetros de un experimento, se establece la geometría del sistema y finalmente se explica la forma de generar un histórico de fase con datos simulados.

3.1 Radar GB-SAR ‘IGP-ROJ’

En esta sección se describe tanto el hardware como el software del radar GB-SAR ‘IGP-ROJ’, así como su funcionamiento.

Las características generales de este radar son:

- **Tipo de radar SAR:** GB-SAR, con apertura lineal.
- **Modo de operación:** *Strip-map SAR*.
- **Forma de onda de las señales:** SF-CW.

En la figura 3.2 se muestra una vista del radar ‘IGP-ROJ’, el cual fue instalado en la localidad de Cuenca, Huancavelica para realizar el estudio de deslizamientos en la zona.

3.1.1 Descripción del hardware y software

El radar GB-SAR construido posee una parte física (hardware) conformado por los distintos instrumentos y equipos, y una parte de software conformado por los programas de funcionamiento. La descripción que se realiza a continuación de cada una de las partes está basada en las publicaciones realizadas por el IGP sobre el diseño y construcción del radar SAR, principalmente de la referencia [6].

Los elementos principales del hardware se clasifican en 2 tipos: electrónicos y mecánicos. Cada uno de estos elementos se explican en la siguiente lista:

1. Elementos electrónicos

■ Antenas

Son los elementos encargados de irradiar la señal de radiofrecuencia (RF) hacia el área de interés (blanco). Se usan 2 antenas en total, una de transmisión (TX) y otra de recepción (RX). El tipo de antena que se usa es de bocina ¹, modelo LB-62-15 de la empresa A-Info. Sus especificaciones técnicas más importantes son: banda de operación Ku, ganancia típica de 15 dB y ancho de haz de 30°. Una imagen referencial de esta antena es mostrada en la figura 3.2.

■ Analizador Vectorial de Redes (VNA)

Es la parte fundamental del radar SAR. Se usa un VNA modelo Anritsu VNA Master

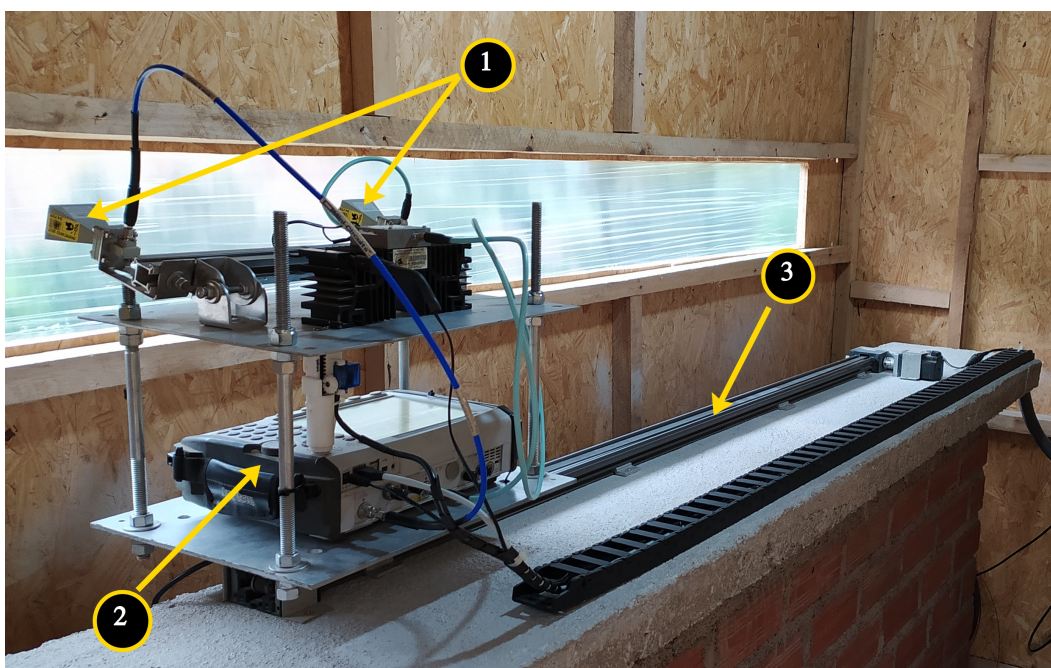


Figura 3.2: Elementos mecánicos y electrónicos del radar GB-SAR 'IGP-ROJ': 1) antenas de bocina, 2) analizador vectorial de redes (VNA) y 3) sistema de posicionamiento lineal. (Fuente propia)

¹Antenas Horn

MS2028C. En transmisión funciona como generador de señales escalonadas del tipo SF-CW. En recepción a través de la señal recibida calcula el parámetro de dispersión S_{21} del sistema de 2 puertos conformado por las 2 antenas y el área de interés. El analizador de redes usado puede generar frecuencias que van desde los 15 KHz hasta los 20 GHz, y hasta 4001 puntos de muestreo en frecuencia por cada medición realizada. En la figura 3.2 se muestra una imagen del VNA usado.

- **Amplificador de potencia (PA)**

Es usado para incrementar la potencia de la señal de transmisión y así llegar a mayores distancias en rango. Se usa un amplificador ZVE-3W-183+ de la empresa Minicircuits, el cual posee una ganancia típica de 35 dB y una potencia máxima de 3 W.

2. Elementos mecánicos

- **Sistema de posicionamiento lineal**

También llamado riel lineal, se compone de una correa dentada de la marca Igus, el cual es un actuador lineal que permite el movimiento horizontal de las antenas del radar que van fijadas a él en una plataforma. Tiene una dimensión total aproximada de 1700 mm. La longitud usada de la correa dentada determina el tamaño de la apertura sintética. En la figura 3.2 se muestra una imagen de la correa dentada usada junto al motor a pasos.

- **Motor a pasos**

Es el elemento que provee el movimiento a la correa dentada. Tiene una resolución de 20000 pasos por revolución y un desplazamiento lineal de 66 mm por revolución. Respecto al torque máximo que puede mover, tiene una capacidad de 20 Kg. Las señales eléctricas para controlarlo son enviadas por el driver SMD7612 de la empresa National Instruments, que a su vez es controlado por un microcontrolador ATmega 328P.

El software del radar GB-SAR lo conforman 2 programas principales desarrollados en el Instituto Geofísico del Perú (IGP): el programa de control del radar y el programa de adquisición de datos. Ambos programas fueron implementados en un servidor web asociados a una base de datos.

3.1.2 Funcionamiento

El procedimiento que sigue el radar GB-SAR para obtener el histórico de fase se explica a continuación. En primer lugar se determina la longitud de la apertura sintética L_s , la cual debe de tener un valor menor o igual a la longitud del riel. Luego se determina el paso del riel d_p y como consecuencia se calcula el número de puntos de escaneo $N_p = \frac{L_s}{d_p} + 1$, si el valor de N_p no es un entero positivo se aproxima el resultado al entero inmediato superior y se determina un nuevo valor para $L_s = d_p(N_p - 1)$. La plataforma que transporta al VNA, las antenas y

al PA se moverá a lo largo del riel partiendo de una posición inicial y deteniéndose sólo en posiciones múltiplos del paso del riel, hasta recorrer toda la longitud L_s , o equivalentemente haberse detenido N_p veces. En cada posición detenida, el VNA genera y transmite una señal tipo SF-CW, compuesta de N_f frecuencias y barriendo un ancho de banda BW , acto seguido el VNA capta la información de los ecos recibidos y los almacena en un disco duro presente en la caja de control del radar. La información captada es el parámetro “ S_{21} ” del sistema de 2 puertos formado por las 2 antenas (TX y RX) y el área de interés (blanco), este parámetro es proporcional al campo eléctrico reflejado por el blanco.

El histórico de fase se obtiene cuando el radar finaliza de recorrer toda la apertura sintética. Se puede representar en una matriz $S_{k,i}$ de 2 dimensiones, donde las filas de $S_{k,i}$ representan el dominio del espacio, y esta formada por todas las posiciones del riel; mientras que las columnas representan el dominio de la frecuencia y esta formada por todas las frecuencias de la señal SF-CW. La matriz tiene la siguiente forma:

$$S_{k,i} = [S] = \begin{bmatrix} S_{(0,0)} & \cdots & S_{(0,i)} & \cdots & S_{(0,N_f-1)} \\ S_{(1,0)} & \cdots & S_{(1,i)} & \cdots & S_{(1,N_f-1)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ S_{(k,0)} & \cdots & S_{(k,i)} & \cdots & S_{(k,N_f-1)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ S_{(N_p-1,0)} & \cdots & S_{(N_p-1,i)} & \cdots & S_{(N_p-1,N_f-1)} \end{bmatrix}_{N_p \times N_f}$$

Donde:

$$S_{k,i} = S(u_k, f_i) = S\left(u_k = \frac{-L_s}{2} + k \cdot d_p, f_i = f_c - \frac{BW}{2} + i \cdot \Delta f\right) \quad (3.1)$$

Con $k = 0, 1, 2, \dots, (N_p - 1)$; $i = 0, 1, 2, \dots, (N_f - 1)$. El término Δf es el paso entre frecuencias de la señal SF-CW. En la figura 3.3 se muestra una representación gráfica del histórico de fase obtenido en el dominio espacio - frecuencia.

3.1.3 Adquisición de los datos

Los datos crudos se almacenan en el mismo radar, y para poder acceder a ellos el usuario se tiene que conectar por red al radar, por ende tanto el radar como el usuario tienen que estar conectados en una misma red local. La descarga de los datos se realiza mediante el protocolo SFTP. El formato de la datos descargados es HDF5, el cual es un formato de archivo estándar que se usa para almacenar datos científicos. Un archivo descargado HDF5 contiene un bloque de histórico de fase, y los parámetros del experimento tales como: la frecuencia inicial y final, el número de frecuencias, la posición de la apertura sintética inicial y final, el número de posiciones de escaneo, el ángulo de visión de la imagen, la potencia de

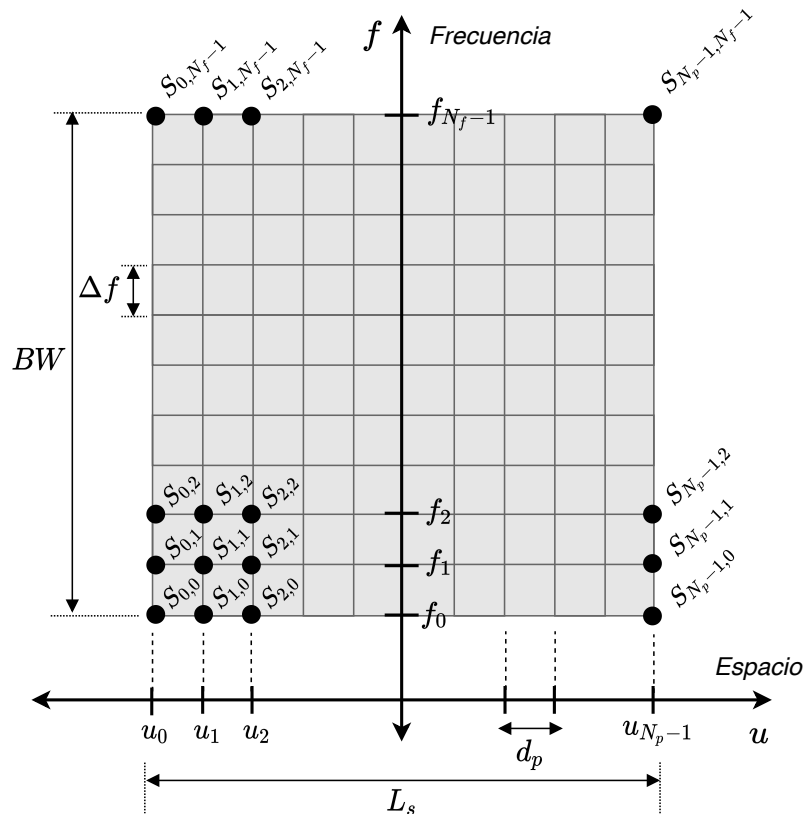


Figura 3.3: Representación gráfica del histórico de fase. (Fuente propia).

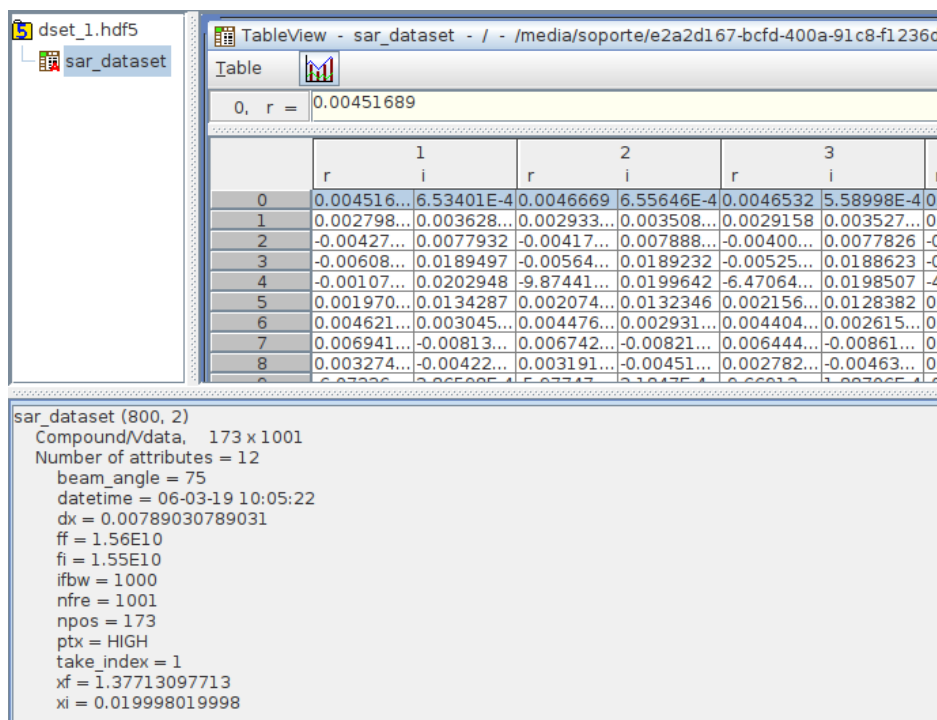


Figura 3.4: Contenido de un archivo HDF5 descargado del radar GB-SAR, en el cual se puede observar tanto el histórico de fase como los parámetros del experimento. (Fuente propia)

transmisión, la fecha de la toma de datos, entre otros.

En la figura 3.4 se muestra el contenido de un archivo HDF5 descargado, abierto con el

programa *hdfview*. En la parte superior derecha de esta figura se observa el histórico de fase como una matriz de posiciones del riel (filas) y frecuencias (columnas), además debido al tipo de dato complejo, cada dato es mostrado en su parte real e imaginaria. Por otro lado, en la parte inferior se muestran los parámetros del experimento en formato de texto, con un total de 12 parámetros. Tanto el histórico de fase como los parámetros conforman los datos de entrada a los algoritmos de formación de imágenes SAR.

3.2 Criterios de selección de parámetros

Antes de realizar un experimento de formación de imágenes SAR, se tienen que seleccionar cuidadosamente los parámetros del experimento, los cuales dependen principalmente de las características de la imagen SAR que se desea reconstruir, tales como: resolución, tamaño de la imagen y tipo de blanco. A continuación se describen los criterios de selección para cada parámetro:

■ Frecuencia central f_c

La elección de la frecuencia central depende del tipo de aplicación. En aplicaciones de monitoreo de deslizamientos usando radares GB-SAR se suelen usar frecuencias en la banda K_u [13], [5]. Una de las razones por las que se usa esta banda, es para obtener una alta resolución en las imágenes SAR, y poder detectar variaciones pequeñas de desplazamientos de tierra. En el caso del radar GB-SAR 'IGP-ROJ', este opera a una frecuencia central de 15.5 GHz, la cual se emplea en las pruebas con datos reales. Por otro lado, sin perder generalidad se usa una frecuencia central de $f_c = 15$ GHz para las pruebas con datos simulados .

■ Ancho de banda BW

El ancho de banda BW está determinado por la resolución en rango Δr que se quiere obtener, mostrado en la ecuación (3.2), ambas variables están relacionados de forma inversamente proporcional, lo cual significa que para resoluciones pequeñas en rango se necesita un ancho de banda amplio.

$$BW = \frac{c}{2\Delta r} \quad (3.2)$$

■ Número de frecuencias N_f

El número de frecuencias N_f influye en la magnitud del Rango No Ambiguo (R_U). El valor de R_U determina el rango máximo que se puede obtener en la imagen final y está dado por la teoría de PDS. Además R_U es igual a la longitud del intervalo de observación del dominio del espacio L_r explicado en la sección 2.3.1. Sin embargo, este término es diferente a R_{max} , el cual es la distancia máxima del blanco que el radar puede detectar y está determinada por la ecuación del radar. Por lo tanto, todos los rangos de los blancos deberían ser menores a R_U para que estén incluidos en la escena de la imagen SAR. Manteniendo constante Δr , la

relación entre N_f y R_U es directamente proporcional:

$$N_f = \frac{R_U}{\Delta r} \quad (3.3)$$

■ **Paso del riel d_p**

El paso del riel ' d_p ' está determinado principalmente por el ángulo de visión máximo ' θ_{max} ' que tendrá la imagen SAR, tal como se muestra la figura 3.5. La relación entre ambas variables es inversamente proporcional tal como se muestra en la ecuación (3.4), por lo tanto para obtener pasos del riel más grandes se tiene que disminuir el ángulo de visión, pero esto conlleva a tener menor área disponible para graficar correctamente la imagen SAR. La ecuación (3.4) previene el efecto de solapamiento al procesar la señal.

$$d_p \leq \frac{\lambda}{4 \sin \theta_{max}} \quad (3.4)$$

■ **Apertura Sintética L_s**

El tamaño de la apertura sintética L_s determina la resolución en azimut Δx de la imagen. Para obtener un valor aproximado de L_s , se puede escoger un valor de resolución en azimut máximo Δx_{max} correspondiente a un rango máximo R_{max} , lo anterior se representa en la siguiente ecuación:

$$L_s = \frac{\lambda}{2\Delta x_{max}} R_{max} \quad (3.5)$$

Los Δx que se escogen deberían ser mayores a $\Delta x_{min} = l/2$, donde l es el tamaño real de la antena, tal como se explicó en la sección 2.2.2.

En la tabla 3.1 se resume los criterios de selección de cada uno de los parámetros.

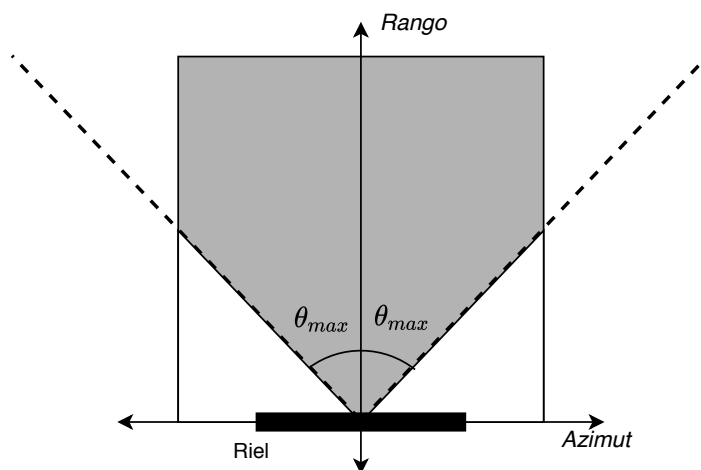


Figura 3.5: Ángulo de visión máximo de la imagen SAR ' θ_{max} '. La imagen formada en la zona gris estará libre del efecto de solapamiento. (Fuente propia)

Tabla 3.1: Principales parámetros eléctricos y mecánicos de un sistema de formación de imágenes con radares SAR.

Parámetro	Símbolo	Dependencia
Frecuencia central	f_c	15 , 15.5 GHz
Ancho de banda	BW	$\propto 1/\Delta r$
Numero de frecuencias	N_f	$\propto R_U$
Paso del riel	d_p	$\propto 1/\theta_{max}$
Apertura Sintética	L_s	$\propto 1/\Delta x$

3.3 Geometría

Se usa un plano coordenado con ejes: Azimut y Rango, para la formación de las imágenes SAR, tal como lo explicado de manera preliminar en la sección 2.2.1. En esta sección se explicarán más detalles de la definición de este plano, tales como la ubicación de la apertura sintética y de la escena de la imagen.

El plano azimut-rango se hace coincidir con un plano cartesiano, donde el eje x coincide con el eje de azimut y el eje y con el eje del rango, tal como se muestra en la figura 3.6. La apertura sintética L_s , se ubica en el origen de coordenadas centrado en Azimut, y la primera posición de la antena empieza en el extremo izquierdo del riel, moviéndose de izquierda a derecha periódicamente. La escena de la imagen SAR presenta forma rectangular (área gris de la figura 3.6) con dimensiones: L_x en el eje de azimut, y L_y en el eje de rango; y una grilla

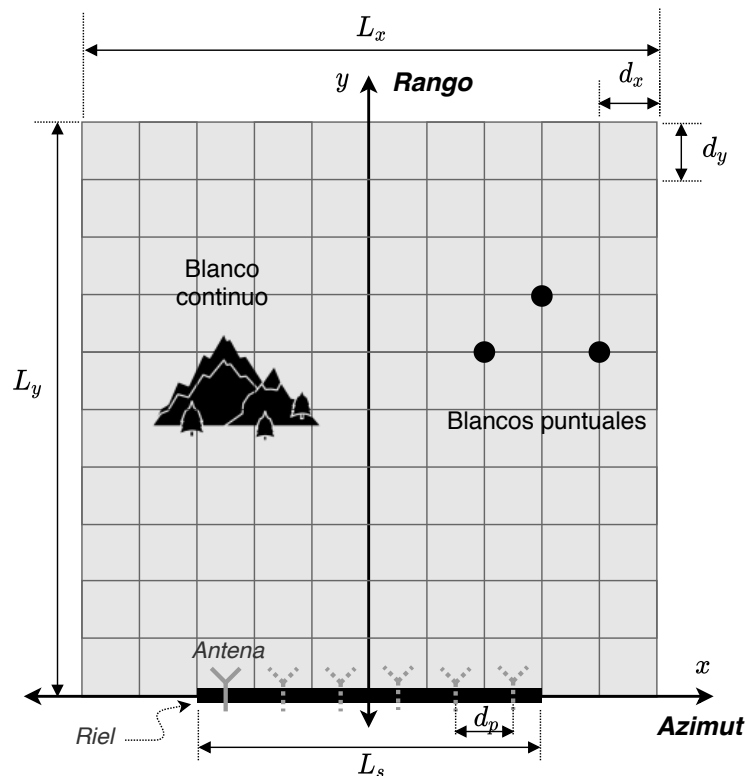


Figura 3.6: Geometría detallada usada para la formación de imágenes SAR, mostrando la ubicación de la apertura sintética y la escena de la imagen. (Fuente propia)

de tamaño: d_x en el eje de azimut y d_y en el eje del rango. La escena de la imagen se ubica en el origen de coordenadas centrado en azimut e iniciando desde el eje positivo en rango.

Finalmente se debe distinguir los siguientes términos para evitar confusiones:

- **rango:** Es la distancia entre el radar y el blanco.
- **Eje de Rango:** Es una coordenada cartesiana, perpendicular al eje de azimut. Coincide con el eje cartesiano y .
- **Eje de Rango (polar):** Es una coordenada polar del rango de los blancos.

Todas las imágenes SAR se forman en un plano cuyo eje vertical es el eje de Rango. Por otro lado, todos los perfiles de rango tienen como eje el eje de Rango (polar).

Un resumen de los parámetros geométricos de la escena de la imagen se muestra en la tabla 3.2.

Tabla 3.2: Parámetros de la escena de la imagen SAR.

Parámetros geométricos	Símbolo
Ancho de la imagen	L_x
Alto de la imagen	L_y
Longitud de la grilla en el eje de Azimut	d_x
Longitud de la grilla en el eje del Rango	d_y

3.4 Simulación de datos

Para realizar las pruebas de los algoritmos y la comparación entre ellos se obtuvo de manera teórica el histórico de fase, a cuyo procedimiento se le llama simulación de datos. La simulación de datos está basado en el planteamiento y solución del problema directo, el cual consta de calcular el histórico de fase a partir del conocimiento de la ubicación y la reflectividad de los blancos, el cual se explicó en la sección 2.4.3. Por lo tanto, primero se define los blancos, tanto sus posiciones como sus reflectividades, si es un blanco continuo se divide en varios blancos puntuales, y luego se aplica la siguiente ecuación: [21]

$$S(u_k, f_i) = \sum_{n=1}^N I_n e^{(-j \frac{4\pi}{c} f_i \sqrt{(u_k - x_n)^2 + y_n^2})} \quad (3.6)$$

Donde (x_n, y_n) representa la posición e I_n la reflectividad de cada blanco puntual. El valor de la reflectividad es compleja y su fase juega un papel importante en el cálculo de los deslizamientos del blanco. Sin perder generalidad y para la simplificación de los cálculos, se

Tabla 3.3: Parámetros del sistema usados en la simulación de datos.

Parámetro	Valor
Frecuencia central (f_c)	15 GHz
Ancho de banda (BW)	600 MHz
Numero de frecuencias (N_f)	41
Longitud del riel (L_s)	1.2m
Numero de posiciones (N_p)	238

asume $I_n = 1$ para todos los casos de obtención de histórico de fase teórico dentro de este trabajo.

Como ejemplo se presenta el caso de un conjunto de 3 blancos puntuales con las siguientes posiciones:

$$(x_1, y_1) = (0, 2)\text{m}, (x_2, y_2) = (-2, 8)\text{m}, (x_3, y_3) = (2, 8)\text{m}$$

Considerando los parámetros del radar SAR mostrados en la tabla 3.3, y reemplazando en la ecuación (3.6), se obtiene su histórico de fase. La matriz del histórico de fase es de tamaño 238×41 y está formado por 9758 datos complejos, en la figura 3.7 se muestra esta matriz graficada tanto en magnitud como en fase. Cada fila del histórico de fase genera un perfil de rango para una posición del riel dado.

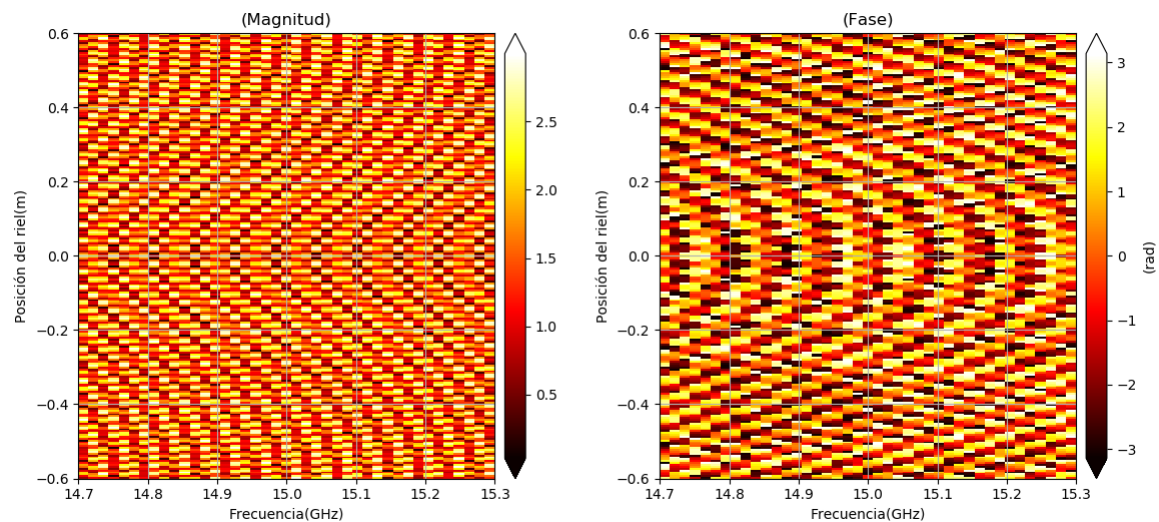


Figura 3.7: Histórico de fase simulado en magnitud y fase para el ejemplo de los tres blancos puntuales. (Fuente propia)

CAPÍTULO IV IMPLEMENTACIÓN DE ALGORITMOS DE FORMACIÓN DE IMÁGENES

En el presente capítulo se implementan tres algoritmos de formación de imágenes para radares GB-SAR. Estos algoritmos son: Frequency Domain Back-Projection (FDBP), Range Migration Algorithm (RMA), y Discrete Linear Inverse Problem (DLIP). La característica que poseen en común es que sus ecuaciones no presentan demasiadas aproximaciones, lo cual permite formar una imagen con el menor error posible.

Para una mejor explicación de cada algoritmo se toma como ejemplo la reconstrucción de 3 blancos puntuales con posiciones:

$$(Azimut, Rango) = [(0, 2); (0, 5); (0, 8)]m$$

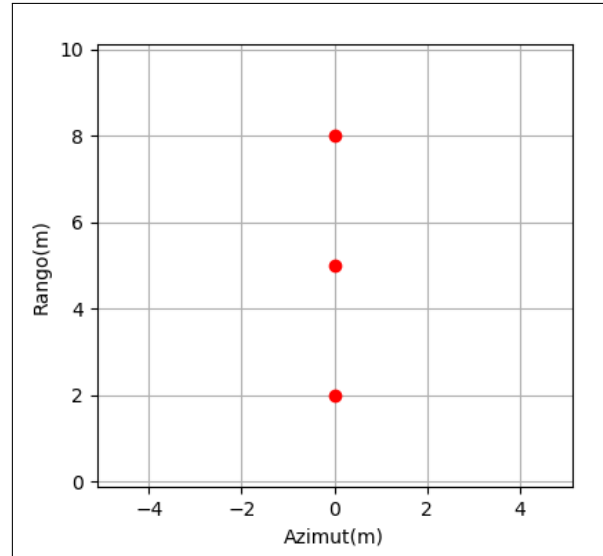
Se consideran los parámetros de la tabla 4.1. Se obtiene su histórico de fase simulado (mostrado en la figura 4.1b) haciendo uso de la ecuación (3.6) y la geometría dada en el capítulo III.

Tabla 4.1: Parámetros eléctricos y mecánicos usados en la explicación de los pasos del algoritmo FDBP.

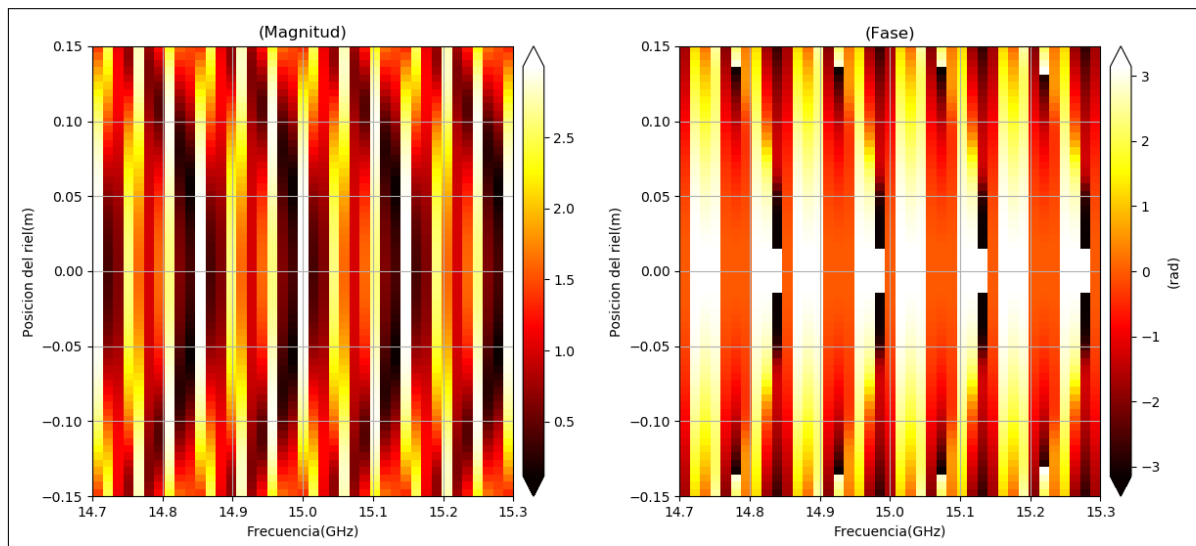
PARÁMETRO	VALOR
Frecuencia central f_c	15 GHz
Ancho de banda BW	600 MHz
Número de frecuencias N_f	41
Apertura sintética L_s	0.3 m
Ángulo de visión θ	90°
Número de posiciones N_p	62
Ancho de la imagen L_x	10 m
Alto de la imagen L_y	10 m
Grilla en azimut d_x	1.25 cm
Grilla en rango d_y	1.25 cm

4.1 Implementación del algoritmo FDBP

La implementación del algoritmo Frequency Domain Back Projection (FDBP) se realizó basándose en las referencias de la sección 2.4.1. Además de haber usado los pasos de la referencia, se realizaron mejoras a la imagen final como el uso de otros tipos de interpoladores y el uso de funciones de enventanado.



(a)



(b)

Figura 4.1: (a) Blanco usado en las pruebas de los algoritmos de formación de imágenes, (b) histórico de fase simulado respectivo. (Fuente propia)

4.1.1 Pasos

Los pasos se derivaron de las ecuaciones (2.34) y (2.35), los cuales se volvieron a reescribir usando los términos establecidos en el capítulo III:

$$I_n = \frac{1}{N_f N_p} \sum_k R_{n,k}^2 F_n^{(k)} \quad (4.1)$$

$$F_n^{(k)} = \sum_i S_{k,i} e^{j \frac{4\pi f_i}{c} (R_{n,k} - R_o)} \quad (4.2)$$

Donde:

- I_n es el valor de la imagen compleja reconstruida en el punto “ n ”
- $R_{n,k}$ es la distancia entre el punto n de la imagen y la posición k del riel (u_k).
- R_o es una constante que toma en cuenta el retraso de los cables.

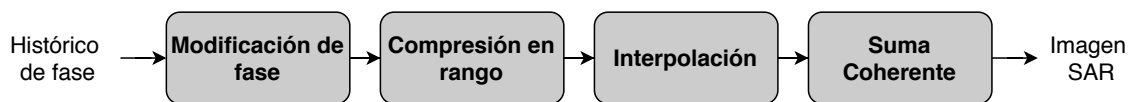


Figura 4.2: Pasos propuestos del algoritmo FDBP. (Fuente propia)

- $S_{k,i}$ es el histórico de fase.
- c es la velocidad de la luz.

La imagen compleja pudo ser obtenida usando directamente las ecuaciones (4.1) y (4.2), pero tuvieron el inconveniente de presentar un tiempo de procesamiento muy alto. Para poder efectuar el algoritmo de una forma más rápida, la ecuación (4.2) se evaluó como la IDFT del histórico de fase respecto a la frecuencia, esto por cada posición del riel. La demostración de lo anteriormente señalado se muestra en el Anexo B. Esta conversión permitió plantear los pasos del algoritmo mostrados en la figura 4.2, los cuales se explican a continuación.

Modificación de Fase

En esta primera etapa se varió la fase de la data cruda debido al término R_o , el cual es expresado por la siguiente ecuación:

$$S'_{k,i} = S_{k,i} e^{-j\left(\frac{4\pi R_o \Delta f_i}{c}\right)} \quad (4.3)$$

En este trabajo se ignoró el término R_o tanto en las pruebas con datos simulados como reales.

Compresión en Rango

Esta etapa consistió en obtener los perfiles de rango por cada posición del riel. Para ello se efectuó una 1D-IDFT a $S'_{k,i}$ a lo largo de la dimensión de la frecuencia:

$$S_2(u_k, r_i) = \mathcal{F}_i^{-1}\{S'_{k,i}\} \quad (4.4)$$

El resultado S_2 es una matriz en el dominio del espacio en ambas dimensiones, cuyas filas constituyen los perfiles de rango, como hay N_p filas entonces existen la misma cantidad de perfiles de rango. Al ejemplo de los 3 blancos puntuales, se le aplicó una compresión en rango, de cuya matriz resultante S_2 se graficó un perfil de rango correspondiente a una fila intermedia, el cual se muestra en la figura 4.3. En esa figura se puede notar que los picos corresponden aproximadamente a las posiciones reales de los blancos, en la generación de esta gráfica se empleó la operación de *zero-padding*.

Después de haber obtenido los perfiles de rango, estos se representaron sobre el plano de la escena de la imagen, la forma de realizarlo fue mediante círculos concéntricos. Un perfil de rango se representó como un conjunto de círculos concéntricos, cuyos centros fueron ubicados en la posición de la apertura sintética respectiva u_k , la separación entre ellos fue la resolución espacial Δr producto de la IDFT, y el valor de cada círculo fueron las muestras

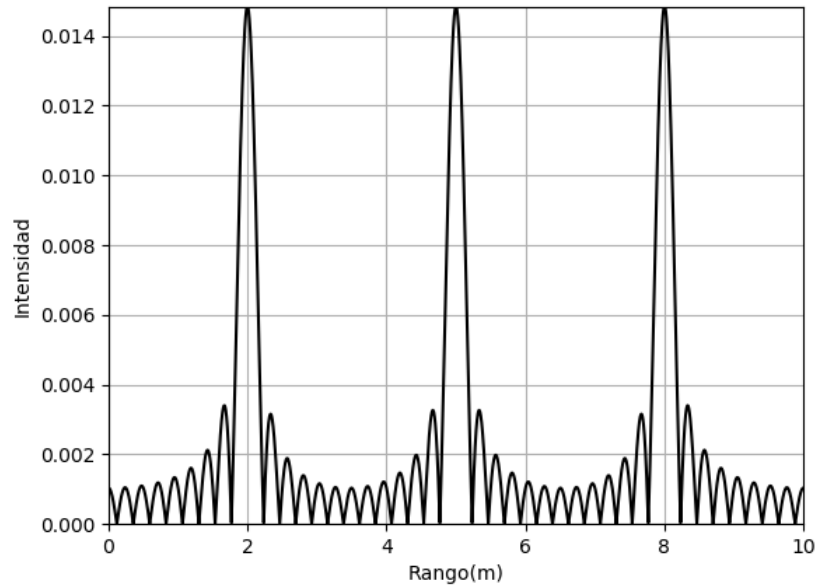


Figura 4.3: Etapa de compresión en rango del algoritmo FDBP para el caso de los 3 blancos puntuales. Se extrae un perfil de rango de la matriz resultante correspondiente a una posición intermedia de la apertura sintética. (Fuente propia)

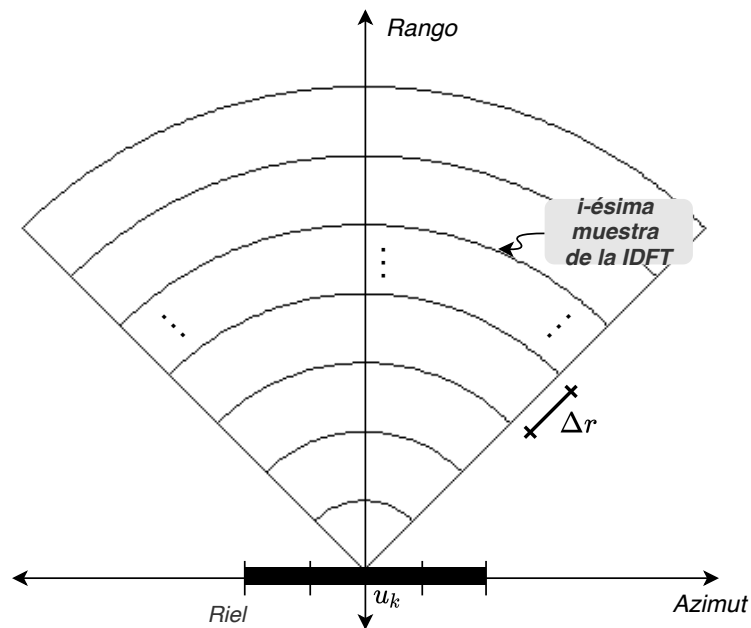


Figura 4.4: Representación del perfil de Rango en el plano de la escena de la imagen. (Fuente propia)

de la IDFT, en la figura 4.4 se muestra esta representación. Todos los perfiles de rango se representaron de forma similar, con la diferencia de que estuvieron desplazados en el eje de azimut por un valor igual al paso del riel d_p .

Interpolación

Este paso tuvo por objetivo calcular los valores en toda la escena de la imagen a partir de los perfiles de rango.

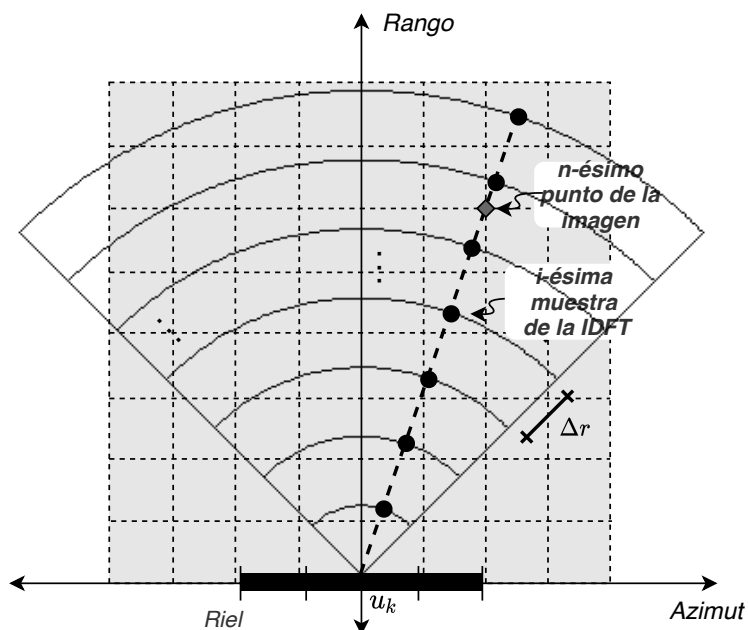


Figura 4.5: Proceso de interpolación en el algoritmo FDBP. (Fuente propia)

Para ello se determinó en primer lugar las dimensiones L_x y L_y de la escena de la imagen, así como el tamaño de la grilla d_x y d_y , conociendo previamente que la escena de la imagen se ubica en el eje positivo en Rango y está centrado en Azimut. Luego se determinaron los valores en todos los puntos o píxeles de la imagen, mediante una interpolación a partir de los perfiles de rango representados sobre el plano de la escena de la imagen. Por cada perfil de rango, correspondiente a una posición del riel u_k , se obtuvo una imagen. Los valores discretos de entrada a la función de interpolación fueron el dominio r_i y los valores del perfil de rango respectivo $S_2(u_k, r_i)$, los puntos a evaluar fueron las distancias de todos los píxeles de la imagen respecto a u_k , en la figura 4.5 se muestra parte de este proceso. Al término de esta etapa de interpolación se obtuvieron N_p imágenes distintas.

En la figura 4.6 se muestra el resultado de la interpolación para el caso de los 3 blancos puntuales, para ello se usó un interpolador lineal. En esta figura se puede observar 3 círculos concéntricos con radios 2, 5 y 8m, los cuales corresponden al valor de los rangos de cada blanco. Estos círculos concéntricos aparecen por el lugar geométrico de la ecuación para un valor de rango constante. También se puede observar que los círculos concéntricos están desplazados ligeramente en el eje de azimut, debido a que están referidos a una posición inicial, media y final de la apertura sintética aproximadamente.

Suma coherente

En este último paso se obtuvo la imagen SAR, para ello se realizó una suma coherente de las N_p imágenes obtenidas en el paso anterior (mostrado en la figura 4.7). La suma coherente consistió en sumar todas las imágenes píxel con píxel tanto en magnitud como en fase. Pre-

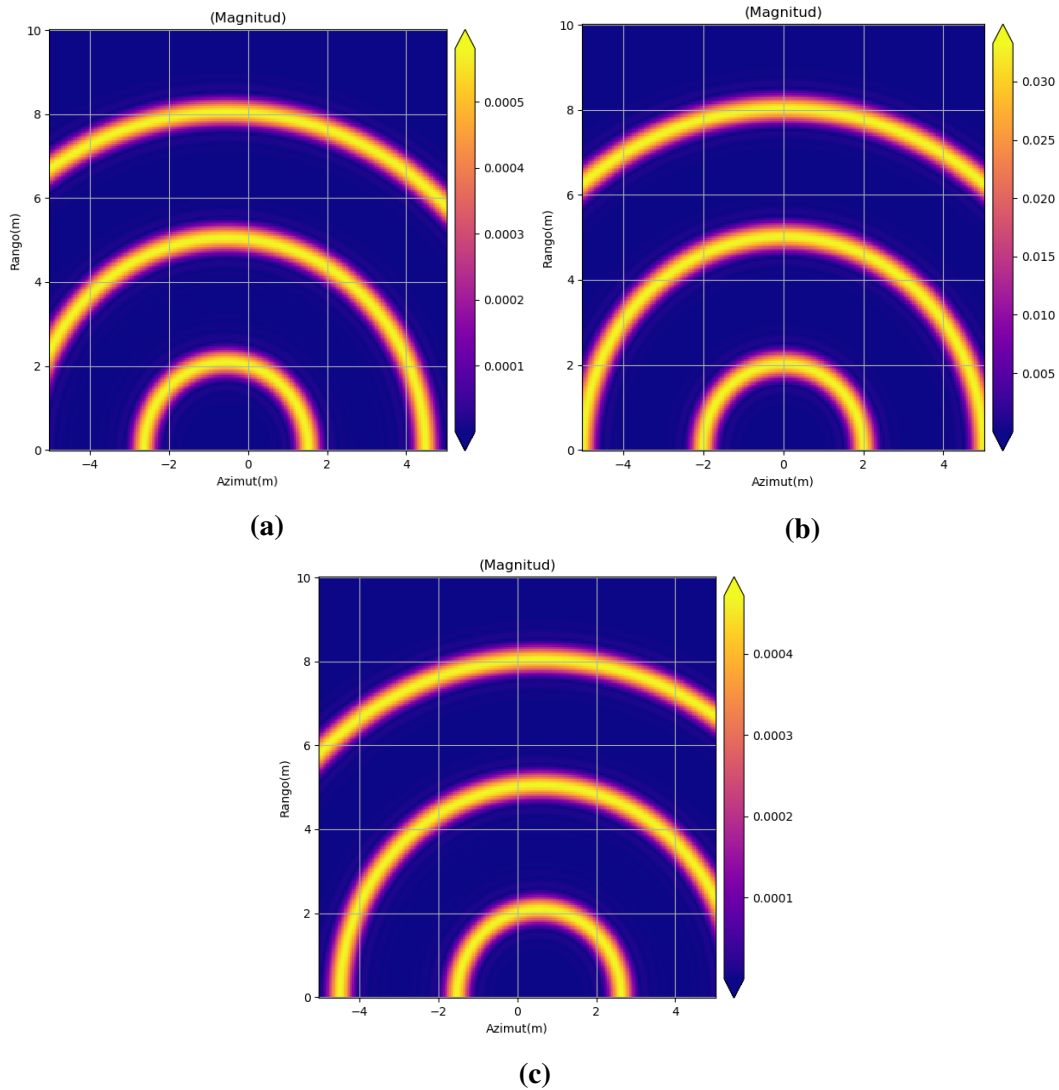


Figura 4.6: Etapa de interpolación del algoritmo FDBP para el caso de 3 blancos puntuales. Se muestran las imágenes SAR en magnitud, correspondientes a las siguientes posiciones en el riel: (a) inicial, (b) intermedia y (c) final. (Fuente propia)

viamente se le tuvo que agregar a cada imagen un factor de fase, opcionalmente se le puede agregar un factor de distancia cuadrático ($R_{n,k}^2$) para hacer notar mejor las zonas más alejadas del blanco en la escena de la imagen. La expresión matemática de esta suma para cada punto de la imagen I_n es la siguiente:

$$I_n = \frac{1}{N_f N_p} \sum_k R_{n,k}^2 F_n'^{(k)} e^{\frac{j4\pi}{c} f_o (R_{nk} - R_o)} \quad (4.5)$$

Donde:

- $e^{\frac{j4\pi}{c} f_o (R_{n,k} - R_o)}$ es el factor de fase
- $F_n'^{(k)}$ es la imagen interpolada
- f_o es la frecuencia inferior

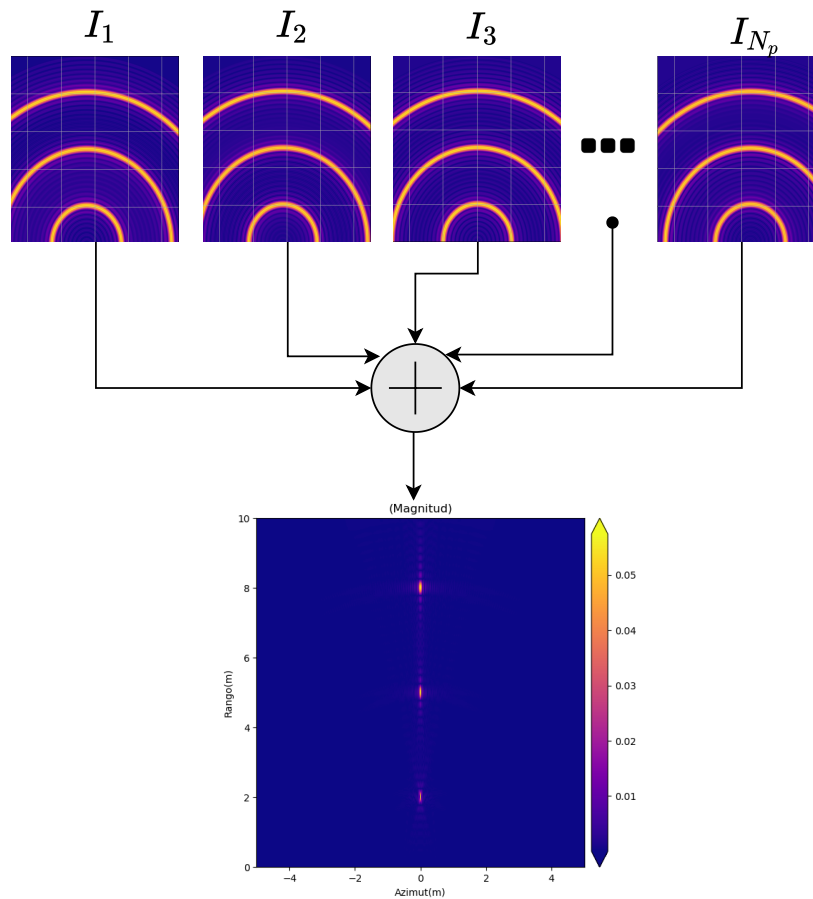


Figura 4.7: Etapa de suma coherente para el caso de los 3 blancos puntuales. En la parte inferior se muestra la imagen SAR resultante.

4.1.2 Resultados

Al finalizar de ejecutar los pasos del algoritmo se obtuvo la imagen SAR, y a continuación se detallarán sus características más importantes, para ello se usará el ejemplo de los 3 blancos puntuales.

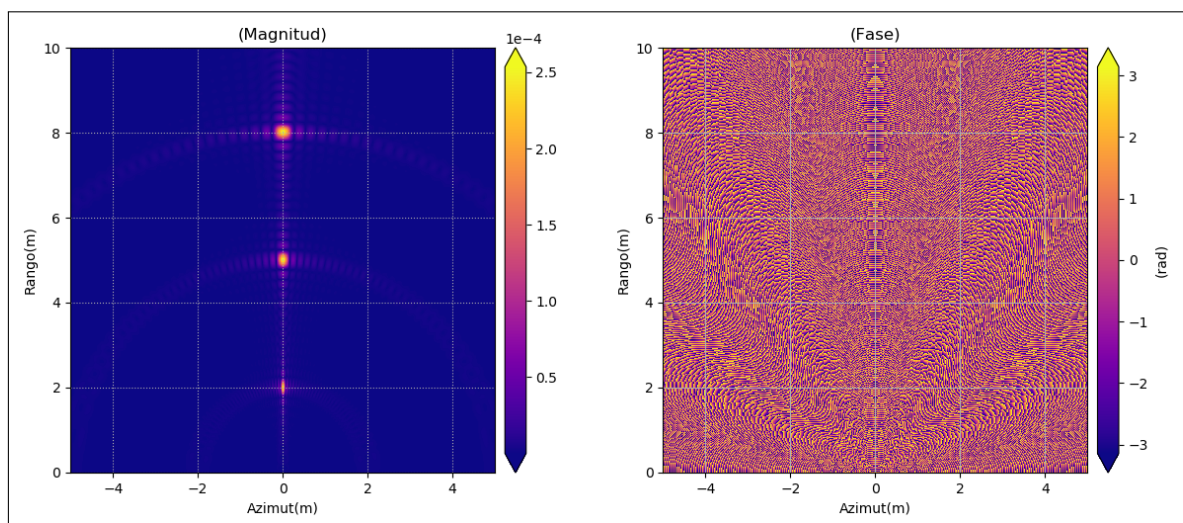


Figura 4.8: Imagen SAR del ejemplo de los 3 blancos puntuales, obtenida con el algoritmo FDBP. (Fuente propia)

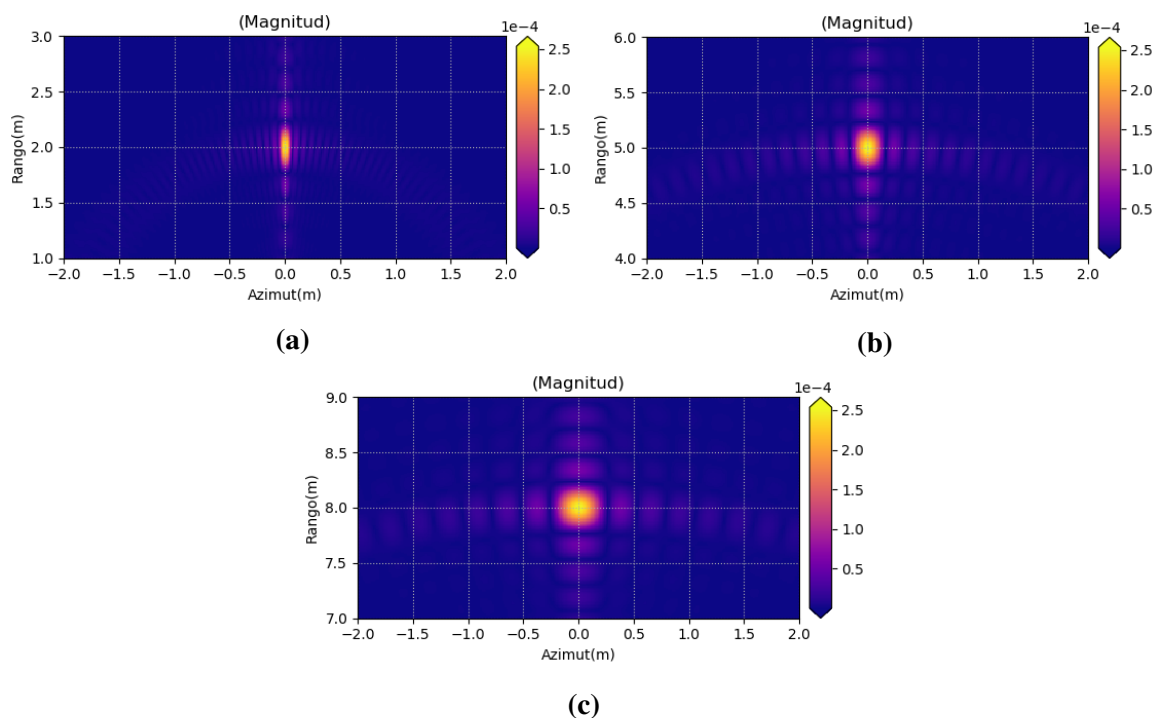


Figura 4.9: Imagen SAR ampliada en magnitud del ejemplo de los 3 blancos reconstruidos usando el algoritmo FDBP. (Fuente propia)

En la figura 4.8 se muestra la imagen SAR del ejemplo mencionado después de haber ejecutado todos los pasos del algoritmo. Como se puede observar en ella, la imagen SAR posee magnitud y fase, las unidades de la magnitud pueden ser unidades absolutas de reflectividad como los mostrados en la figura, o también pueden ser unidades logarítmicas (dB) las cuales tienen la ventaja de permitir observar más detalles de la imagen como los lóbulos laterales en ambas dimensiones. Dentro de la imagen en magnitud se pueden observar los 3 blancos reconstruidos en color amarillo (escala más alta), los cuales se encuentran en sus posiciones correctas. Como se puede observar también en la figura 4.9, cada blanco posee lóbulos laterales en ambas direcciones: rango y azimut, eso es debido al ancho de banda limitado que se usó; por otro lado se nota que la resolución en rango de los 3 blancos es similar, mas no lo es la resolución en azimut, lo cual es un resultado esperado debido a que teóricamente su expresión depende del rango del blanco y como todos los blancos están a un rango distinto consecuentemente tienen una resolución en azimut distinta, siendo mayor para los blancos con mayor rango. La resolución en rango teórico es 0.25 m, mientras que la resolución en azimut para el blanco mas alejado es 0.27 m.

Respecto a la fase, las unidades se expresan en radianes, el valor esperado de cada blanco puntual es 0 rad ya que se le definió teóricamente con fase nula, y de acuerdo a la imagen SAR en fase se observa efectivamente que cada blanco tiene un valor cercano a 0 rad. La fase de las imágenes SAR influye en la obtención de los mapas de desplazamientos, motivo por

el cual tiene importancia en reconstruirlo correctamente.

4.1.3 Mejoras a la imagen

Las mejoras que se le aplicaron a la imagen SAR fueron las siguientes: la elección de un interpolador y la reducción de los lóbulos laterales.

Elección de un interpolador

La elección del interpolador, el cual tiene que ser realizada en la etapa de Interpolación, determina la calidad de la imagen SAR, entendiéndose por calidad como una mejor representación de los blancos reconstruidos, con mayor continuidad y suavidad.

Se probaron 4 tipos de interpoladores: constante, lineal, polinomial cuadrático y polinomial cúbico; cuyos resultados se muestran en la figura 4.10. Como se puede observar en

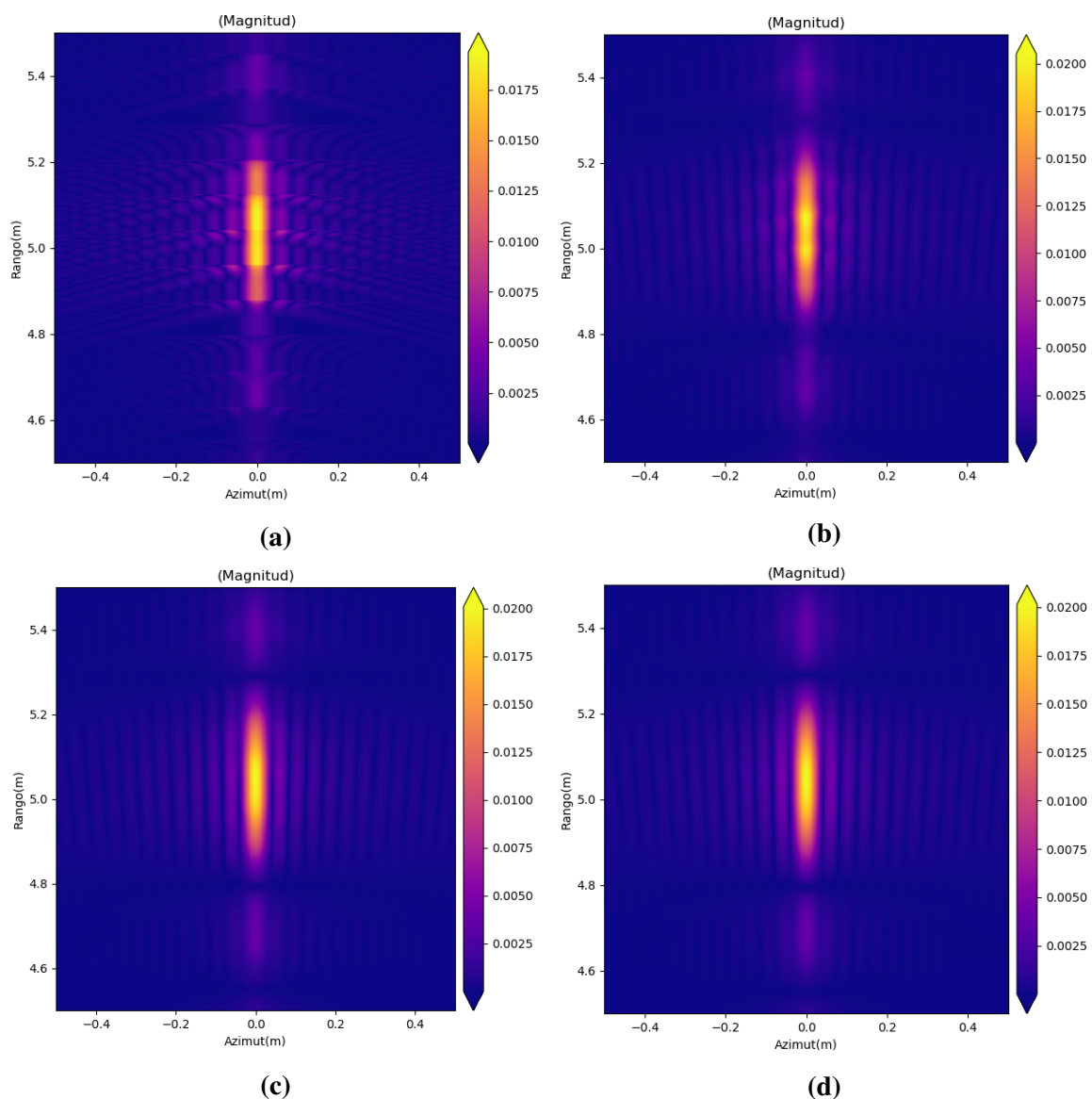


Figura 4.10: Imagen SAR de un blanco puntual en $(0,5)m$, obtenida empleando el algoritmo FDBP y usando 4 interpoladores: (a) constante, (b) lineal, (c) cuadrático y (d) cúbico. (Fuente propia)

ella, el interpolador constante genera la imagen de menor calidad con discontinuidades en varios partes del blanco reconstruido, por otra parte ambos interpoladores polinomiales generan imágenes más continuas, por último el interpolador lineal genera una imagen de calidad intermedia.

El segundo factor que se consideró para elegir el tipo de interpolador fue el tiempo de procesamiento. Por teoría se sabe que los interpoladores polinomiales poseen un alto tiempo de procesamiento, mientras que los interpoladores constantes presentan el menor tiempo de procesamiento, lo cual se validó con pruebas. En la tabla 4.2 se muestra los tiempos de procesamiento del algoritmo al emplear los 4 tipos de interpoladores. En ella se observa que el comportamiento es casi opuesto al factor de calidad, donde el interpolador lineal y constante poseen los menores tiempos, mientras que los interpoladores polinomiales cuadrático y cúbico poseen los mayores tiempos. Habiendo probado los 4 tipos de interpoladores tanto en calidad como en tiempo de procesamiento, se seleccionó la interpolación lineal debido al bajo tiempo de procesamiento y a la calidad media de la imagen, la cual fue aceptable en la mayoría de casos, pudiendo distinguirse bien los blancos y sin presentar discontinuidades.

Tabla 4.2: Tiempos de procesamiento del algoritmo FDBP usando 4 interpoladores.

Interpolador	Tiempo de ejecución
Constante	7.47
Lineal	6.22
Cuadrático	10.78
Cúbico	13.72

Reducción de los lóbulos laterales

En la figura 4.9 se observa que la imagen reconstruida presenta lóbulos laterales en ambas direcciones: rango y azimut, los cuales pueden interferir en la reconstrucción de blancos débiles cercanos. Para reducir estos lóbulos laterales se usaron las funciones de enventanado como Hanning y Hamming.

Se aplicaron las 2 ventanas mencionadas a un blanco puntual de posición $(x,y) = (0,5)m$ para observar el efecto de reducción de los lóbulos laterales, y en la figura 4.11 se muestran los resultados. Se puede notar dos cambios, el primero es la reducción de los lóbulos laterales tal como se esperaba, pero también se nota que aumenta el valor de la resolución. Para cuantificar el valor de cada uno estos cambios usando ambas ventanas, se realizó un corte en $Azimuth = 0m$ a ambas imágenes con filtros y se graficaron, estas gráficas se muestran en la figura 4.12. A continuación, se calculó el ancho de lóbulo principal para obtener la resolución y el máximo SSL para evaluar la atenuación de los lóbulos laterales. Estos resultados se muestran en la tabla 4.3, en ella se observa que las ventanas atenúan los lóbulos laterales en

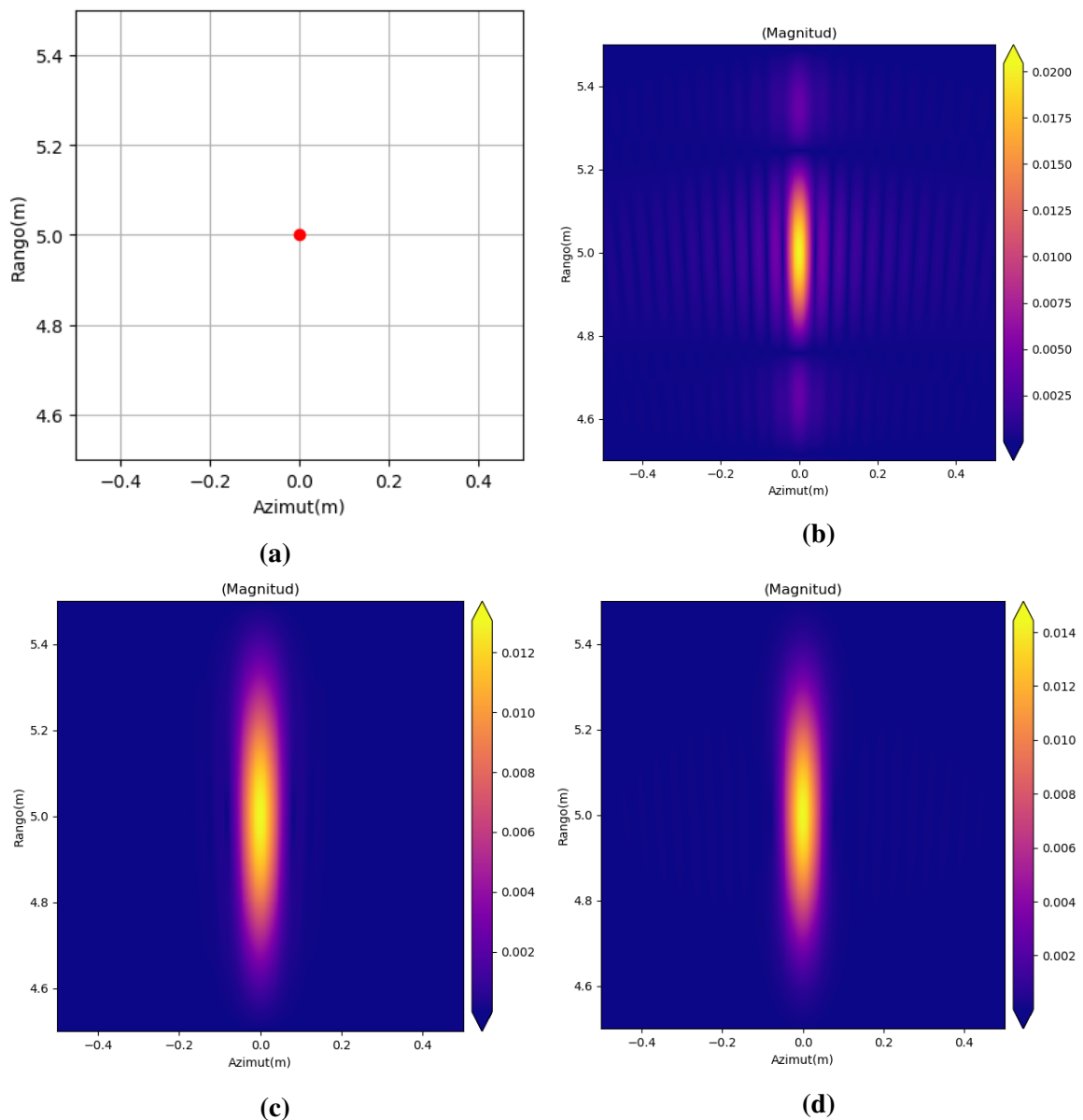


Figura 4.11: Reconstrucción de un blanco puntual aplicando el algoritmo FDBP, y usando 2 funciones de enventanado: (a) blanco teórico, (b) sin ventana, (c) con una ventana Hanning, (d) con una ventana Hamming. (Fuente propia)

más de 3 y 2 veces en unidades de dB , mientras que la resolución aumenta en 28% y 44% para las ventanas de Hamming y Hanning respectivamente. Como se puede observar de los resultados, la ventana Hamming es el que presentó mejores resultados, por lo que se escogió como función de enventanado en este algoritmo.

4.1.4 Resumen de los pasos

A continuación se resumen los pasos del algoritmo FDBP que se usaron, incluyendo las mejoras y la adquisición de datos:

1. Se obtuvieron los datos de entrada como el histórico de fase y los parámetros mecánicos y eléctricos.

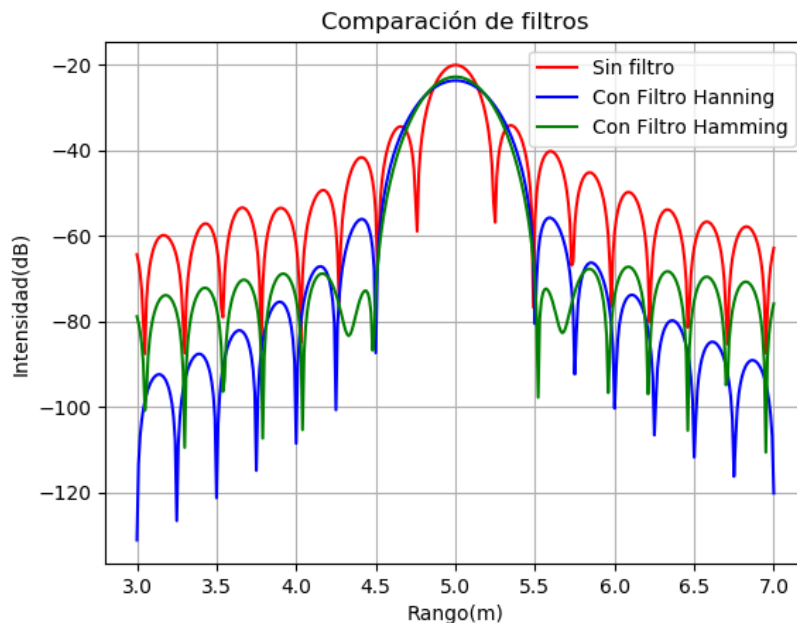


Figura 4.12: Comparación de perfiles de rango empleando las ventanas de Hanning y Hamming, para el caso de un blanco puntual reconstruido con el algoritmo FDBP. (Fuente propia)

Tabla 4.3: Obtención de la resolución y máximo SSL de los perfiles de rango empleando las ventanas de Hanning y Hamming, para el caso de un blanco puntual reconstruido usando el algoritmo FDBP.

Función de enventanado	Ancho del Lóbulo Principal (a -3dB)	Máximo SSL(dB)
Ninguno	0.25m	-14.5
Hanning	0.36m	-36
Hamming	0.32m	-45

2. Se aplicó una ventana Hamming al histórico de fase en ambas dimensiones.
3. Se realizó una compensación de fase empleando la ecuación (4.3).
4. Se efectuó una compresión en rango, obteniéndose N_p perfiles de rango, uno por cada posición del riel.
5. Se definió la escena de la imagen como el tamaño y la grilla, y se computó los valores en cada punto de la imagen mediante una interpolación lineal a partir del perfil de rango. Se obtuvieron N_p imágenes.
6. Se realizó una suma coherente mediante la ecuación (4.5) entre todas las N_p imágenes formadas en el paso anterior para finalmente obtener la imagen SAR, tanto en magnitud como en fase. La fase de la imagen resultante se usa posteriormente en la obtención de los mapas de desplazamientos.

4.2 Implementación del algoritmo RMA

La implementación del algoritmo Range Migration (RMA) se realizó basándose en las referencias de la sección 2.4.2. Además de haber usado los pasos de la referencia se realizaron

mejoras al algoritmo como el uso de funciones de enventanado, decimación y recorte de espectro.

4.2.1 Pasos

Los pasos o etapas principales del algoritmo RMA se muestran en la figura 4.13, los cuales se basaron de las referencias mostradas en la sección 2.4.2. En esta sección se explicarán a detalle cada uno de ellos.

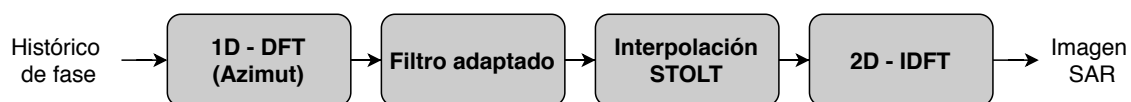


Figura 4.13: Etapas del algoritmo RMA. (Fuente propia)

Para una mejor explicación de los pasos del algoritmo se usa como ejemplo la reconstrucción de los 3 blancos puntuales situados en las siguientes posiciones:

$$(azimut, rango) = [(0, 2), (0, 5), (0, 8)]m$$

Los parámetros se encuentran en la tabla 4.1 y su histórico de fase se muestra en la figura 4.1(b). Se mostrará las señales resultantes después de cada paso del algoritmo.

Pre-procesamiento

Esta es una etapa opcional, el cual consiste en analizar el comportamiento del histórico de fase en el dominio del espacio. Se realiza una compresión en rango y se obtienen los perfiles de rango para conocer los posibles rangos de los blancos. En la figura 4.14 se muestra la compresión en rango para 2 casos de blancos puntuales obtenidos con datos simulados que presentan las siguientes coordenadas:

$$(Rango, Azimut) = [(0, 4), (0, 6), (0, 8)]m$$

$$(Rango, Azimut) = [(0, 4), (2, 6), (-1, 8)]m$$

Donde se usó una apertura sintética de tamaño $L_s = 8m$. El eje vertical de la gráfica de compresión en rango corresponde al eje del Rango polar.

La figura 4.14 presenta los siguientes detalles. En primer lugar hay tantas curvas como número de blancos, en este caso 3, las cuales tienen forma de hipérbola [20]. En segundo lugar, las posiciones de los blancos coinciden aproximadamente con los vértices de las hipérbolas (mostrado en la figura 4.14b). Estas dos características permiten conocer previamente las posibles ubicaciones de los blancos. La longitud en azimut de la matriz comprimida está limitada al tamaño de la apertura sintética L_s , razón por la cual solo se puede aplicar correctamente este método a blancos cuyas posiciones en azimut sean menores a $L_s/2$.

El perfil de rango se obtiene directamente de una fila de la matriz comprimida. En la

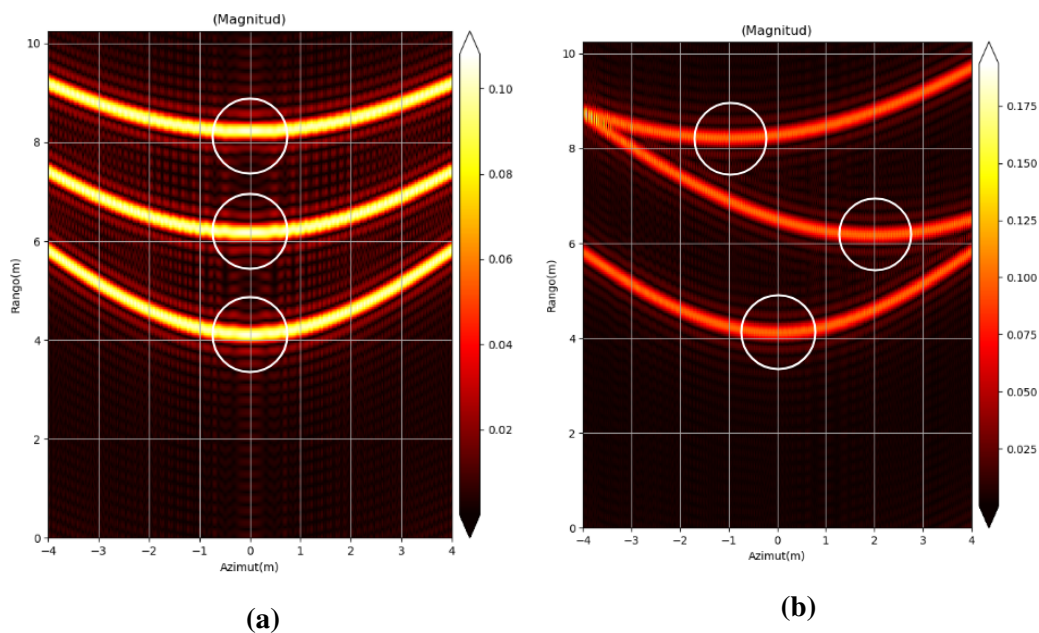


Figura 4.14: Etapa de compresión en rango del algoritmo RMA, para el caso de 2 grupos de blancos puntuales en las posiciones: (a) $(rango, azimuth) = [(0, 4), (0, 6), (0, 8)]$ y (b) $(rango, azimuth) = [(0, 4), (2, 6), (-1, 8)]$. (Fuente propia)

figura 4.15, se muestra un perfil de rango para 3 blancos con coordenadas $(rango, azimuth) = [(0, 4), (0, 6), (0, 8)]m$, para la posición intermedia de la apertura sintética.

1D-DFT (Azimut)

Esta primera etapa tiene como propósito cambiar el dominio del histórico de fase, del dominio espacio - frecuencia al dominio frecuencia - frecuencia. Para ello se efectuó una DFT al

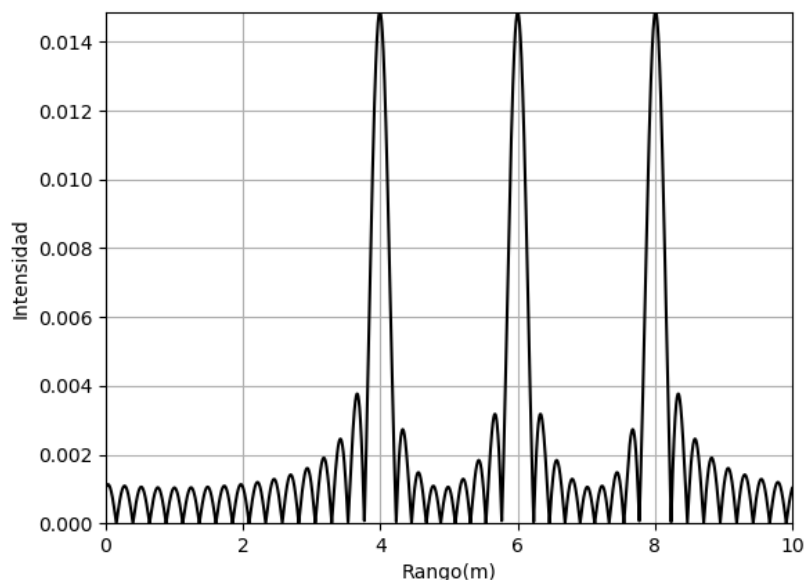


Figura 4.15: Perfil de rango para 3 blancos puntuales con posiciones: $(Rango, Azimut) = [(0, 4), (0, 6), (0, 8)]$, obtenidas de la etapa de compresión en rango del algoritmo RMA. (Fuente propia)

histórico de fase a lo largo del eje de azimut perteneciente al dominio espacial:

$$S(k_x, \omega) = \mathcal{F}_u\{S(u, f)\} \quad (4.6)$$

Donde $\omega = 2\pi f$, y k_x varía linealmente en el intervalo $[-\pi/d_p; \pi/d_p]$. Adicionalmente se realizó el siguiente cambio de variable:

$$k_r = \omega/c \quad (4.7)$$

Con este cambio, la señal $S(k_x, \omega)$ presente ya en el dominio de la frecuencia en ambas dimensiones, se convierte en la siguiente señal:

$$S(k_x, \omega) \rightarrow S(k_x, k_r) \quad (4.8)$$

Un punto importante a considerar en esta etapa inicial es el ancho de la imagen L_x . Aplicando directamente la 1D-DFT al histórico de fase, la máxima distancia L_x que se puede reconstruir es del tamaño de la apertura sintética L_s , en este caso $L_s \approx 1.2$ m mientras que L_x tiene un valor superior a los 100 m. Para poder reconstruir imágenes con $L_x > L_s$, se aplicó un *zero-padding* al histórico de fase en la dimensión del espacio antes de realizar la 1D-DFT. Si la longitud de $S_{k,i}$ en el dominio del espacio es N_p el cual equivale a una longitud en azimut de L_s , entonces la longitud de la señal después del relleno con ceros será N_z el cual equivale aproximadamente a la distancia deseada L_x :

$$N_z \times d_p \approx L_x \quad (4.9)$$

De ahí que la longitud del *zero-padding* sea:

$$z_{pad} = N_z - N_p \quad (4.10)$$

En la figura 4.16 se muestra este proceso del relleno con ceros. Como se puede ver en ella, se tiene que rellenar con ceros a ambos extremos de la señal:

$$S'_{k',i} = \overbrace{\{ \underbrace{0, \dots, 0}_{\frac{(N_z - N_p)}{2}}, \underbrace{S_{0,i}, S_{1,i}, \dots, S_{N_p-1,i}}_{N_p}, \underbrace{0, \dots, 0}_{\frac{(N_z - N_p)}{2}} \}}^{N_z} \quad (4.11)$$

En forma práctica, realizar esta operación de rellenar con ceros a ambos extremos resultó más operativo, razón por la cual se optó por otro método alternativo basado en la propiedad de desplazamiento circular de la DFT. En primer lugar se desplazó $S_{k,i}$ en el dominio del espacio un valor $X_c = L_s/2$ hacia la derecha, de tal modo que la señal se encuentre en el lado positivo del dominio del espacio a partir de cero ($u_0 = 0$ m), en la figura 4.17 se muestra esta idea. Por lo que la señal $S(u, f)$ quedó expresado en términos de la señal desplazada $S_{aux}(u, f)$ del siguiente modo:

$$S(u, f) = S_{aux}(u + X_c, f) \quad (4.12)$$

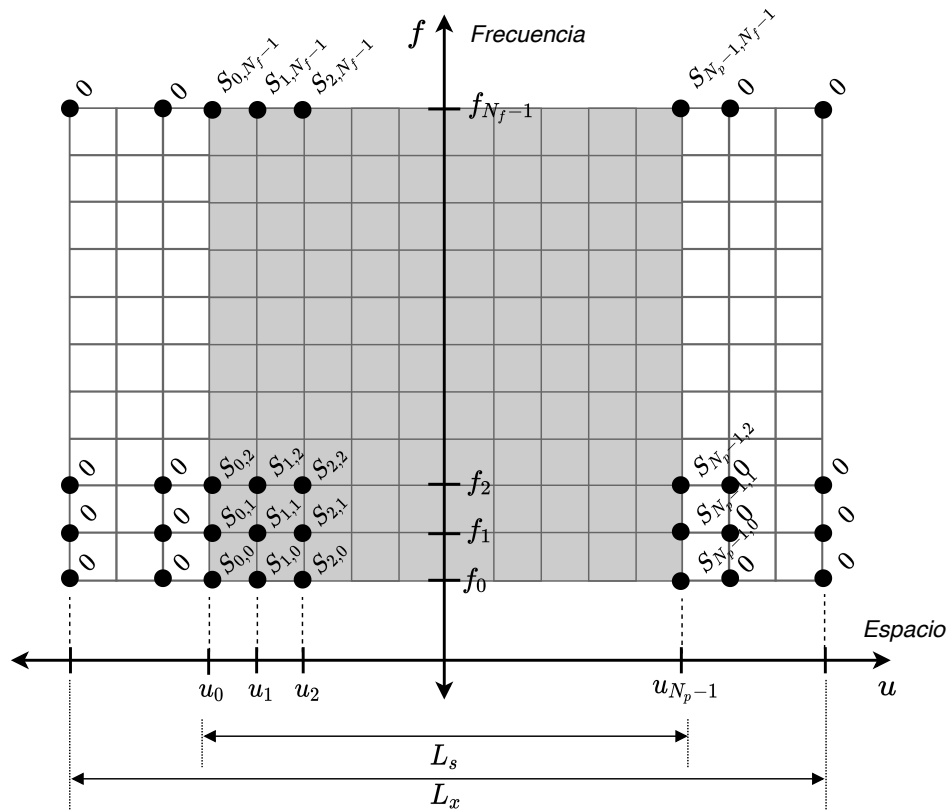


Figura 4.16: Proceso de *zero-padding* en el paso de 1D-DFT del algoritmo RMA. (Fuente propia)

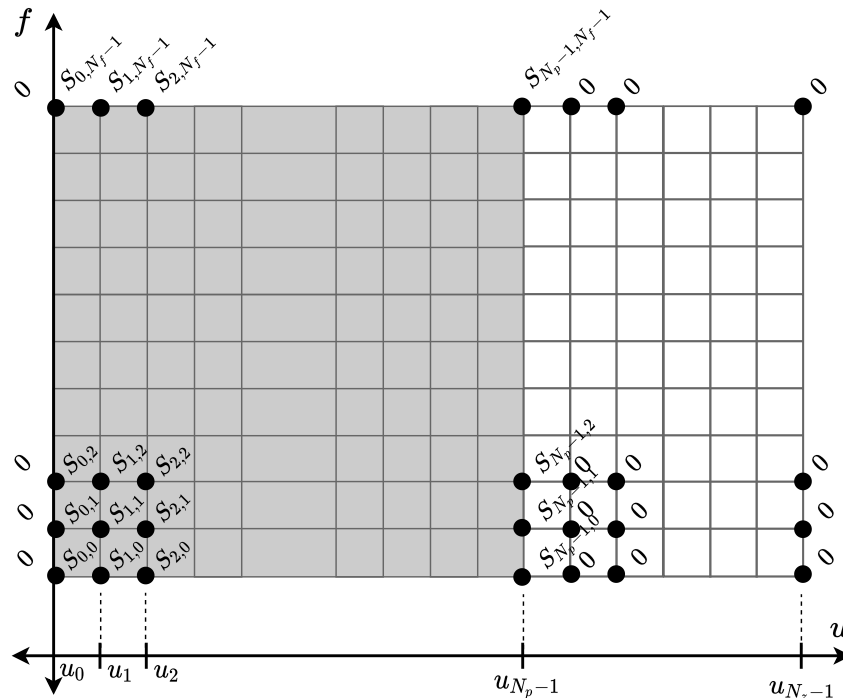


Figura 4.17: Histórico de fase desplazado en $X_c = L_s/2$ hacia la derecha, y el *zero-padding* respectivo. (Fuente propia)

Luego se realizó el relleno con ceros al final de la secuencia, tal como se observa en la

figura 4.17 y en la siguiente expresión:

$$S_{aux}(u_k, f_i) = \underbrace{\{S_{0,i}, S_{1,i}, \dots, S_{N_p-1,i}\}}_{N_p} \underbrace{\{0, \dots, 0\}}_{N_z - N_p} \quad (4.13)$$

Donde $k = 0, 1, \dots, (N_p - 1), N_p, \dots, (N_z - 1)$; con $N_z > N_p$.

A continuación, se calculó la DFT sin *fftshift* de $S_{aux}(u_k, f_i)$:

$$S'(k_x, k_r) = \mathcal{F}_u\{S_{aux}(u, f)\} \quad (4.14)$$

Finalmente se calculó la DFT de $S(u, f)$:

$$\mathcal{F}_u\{S(u, f)\} = \mathcal{F}_u\{S_{aux}(u + X_c, f)\} \quad (4.15)$$

Por la propiedad de desplazamiento circular, se obtuvo el resultado final:

$$\mathcal{F}_u\{S(u, f)\} = \mathcal{F}_u\{S_{aux}(u, f)\} \cdot e^{jk_x \cdot X_c} \quad (4.16)$$

$$S(k_x, k_r) = S'(k_x, k_r) \cdot e^{jk_x \cdot X_c} \quad (4.17)$$

En la figura 4.18 se muestra el resultado de aplicar la 1D-DFT al ejemplo de los 3 blancos puntuales. Se aplicó *zero-padding* debido a que el valor de $L_x = 10\text{m}$ fue más grande que $L_s = 0.3\text{m}$.

Filtro adaptado

Esta etapa consiste en aplicar la siguiente compensación de fase:

$$S_m(k_x, k_r) = S(k_x, k_r) e^{jR_s \sqrt{k_r^2 - k_x^2}} \quad (4.18)$$

Donde R_s es la distancia del centro de la escena de la imagen hacia el eje de Azimut (mostrado en la figura 4.19). Como se definió la escena de la imagen en el eje positivo de

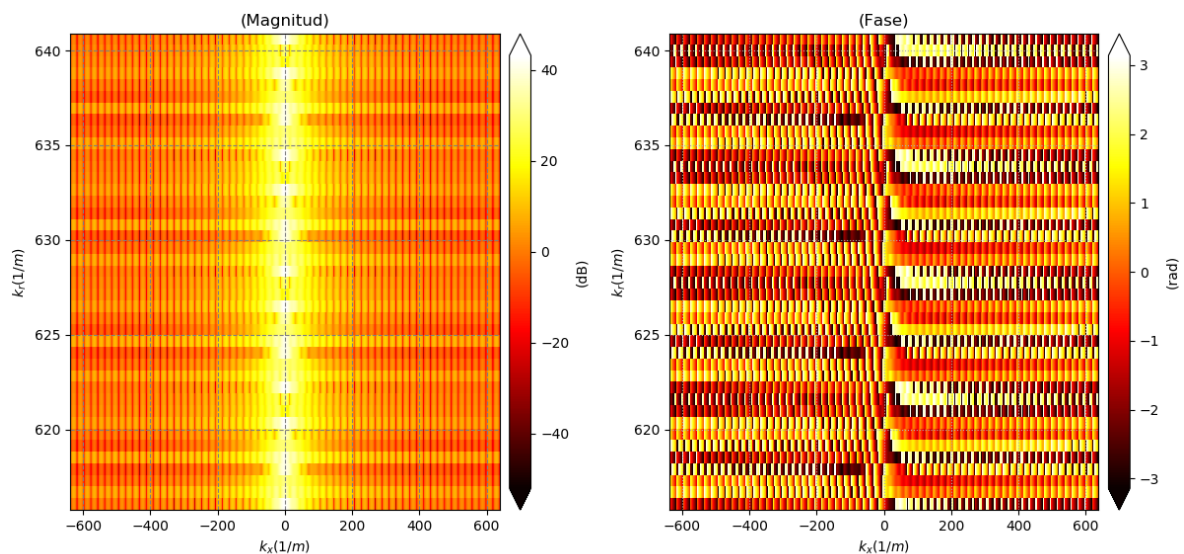


Figura 4.18: Etapa de 1D-DFT (azimut) del algoritmo RMA, aplicado al ejemplo de los 3 blancos puntuales. Se muestran las imágenes resultantes en magnitud y fase. (Fuente propia)

Rango, entonces R_s fue la mitad del alto de la imagen:

$$R_s = \frac{L_y}{2} \quad (4.19)$$

El filtro adaptado se interpretó como una adición de fase debido al desplazamiento de la escena de la imagen en un valor igual a R_s a lo largo del eje del Rango. Este desplazamiento en el dominio del espacio ocasiona la adición de la fase en el dominio de la frecuencia, debido a la propiedad de desplazamiento circular de la DFT. La razón del desplazamiento es poder enfocar el área de interés de reconstrucción de la imagen, el cual está alrededor de R_s .

Si por algún caso se ignora este paso la imagen resultante sale centrada en el origen, el cual tiene la desventaja de reconstruir solo hasta la mitad del rango teórico máximo R_U .

En la figura 4.20 se muestra la aplicación de este paso al ejemplo de los 3 blancos puntuales. Se observa un cambio en la fase de la señal mas no en la magnitud, tal como lo esperado.

Una consecuencia del filtro adaptado es la definición y uso de un nuevo plano cartesiano [17]. El nuevo plano $x' - y'$ esta centrado en la escena de la imagen desplazada, es paralelo al plano Azimut-Rango, y el eje y' esta definido en sentido contrario al eje de Rango, tal como se muestra en la figura 4.21.

Se usa este nuevo plano $x' - y'$, a partir de este paso del Filtro Adaptado. Una vez reconstruida la imagen, se pasará del plano $x' - y'$ al plano Azimut-Rango, mediante las siguientes relaciones:

$$\text{Azimut} = x' \quad (4.20)$$

$$\text{Rango} = R_s - y' \quad (4.21)$$

El filtro adaptado también permitió corregir perfectamente la curvatura en rango de todos los blancos con coordenada $x' = 0$. Para el resto de rangos la compensación fue solo parcial

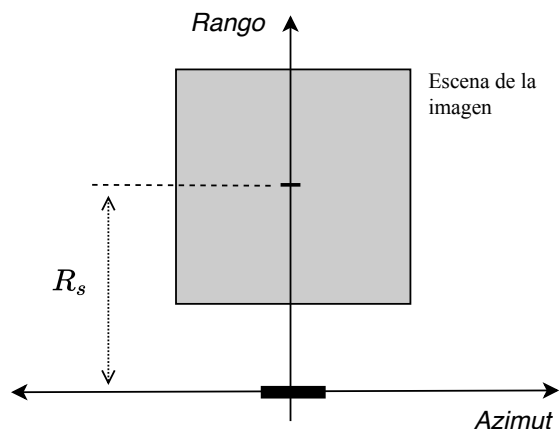


Figura 4.19: Distancia del centro de la escena de la imagen hacia el eje de Azimut. (Fuente propia)

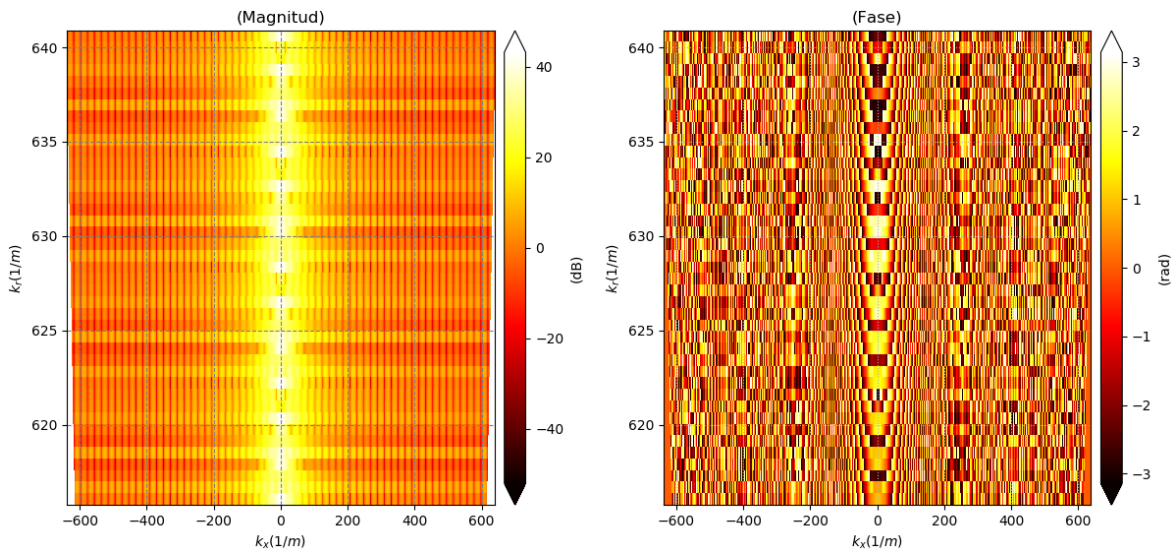


Figura 4.20: Etapa de filtro adaptado del algoritmo RMA, aplicado al ejemplo de los 3 blancos puntuales. Se muestran las imágenes resultantes en magnitud y fase. (Fuente propia)

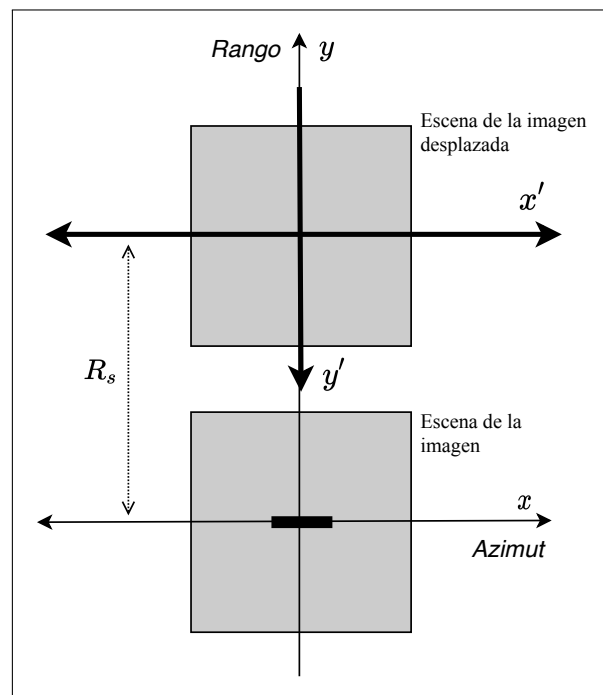


Figura 4.21: Desplazamiento de la escena de la imagen en el eje del Rango por un valor R_s , y definición de un nuevo plano $x' - y'$. (Basado en [17])

y fue tarea del siguiente paso remover las curvaturas residuales (mostrado en la figura 4.22).

En la figura 4.23 se muestra el efecto de la curvatura para el ejemplo de los 3 blancos puntuales, antes y después de realizar el filtro adaptado. Se aumentó los valores de L_s y N_p a modo de ejemplo para apreciar mejor las señales en ambos ejes. También se observa un cambio en la curvatura de los 3 blancos, sin embargo solo el blanco intermedio presenta una corrección perfecta de la curvatura tal como lo esperado ya que coincide con el centro de la escena de la imagen desplazada. Debido a esta etapa, las blancos cercanos al centro de la

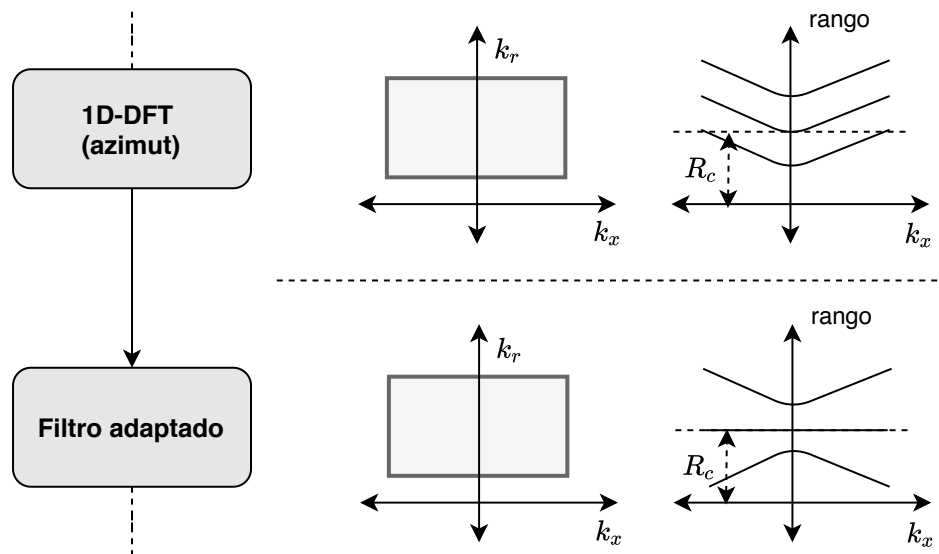


Figura 4.22: Efecto de corrección de curvatura realizado por la etapa de filtro adaptado del algoritmo RMA. (Basado en [17])

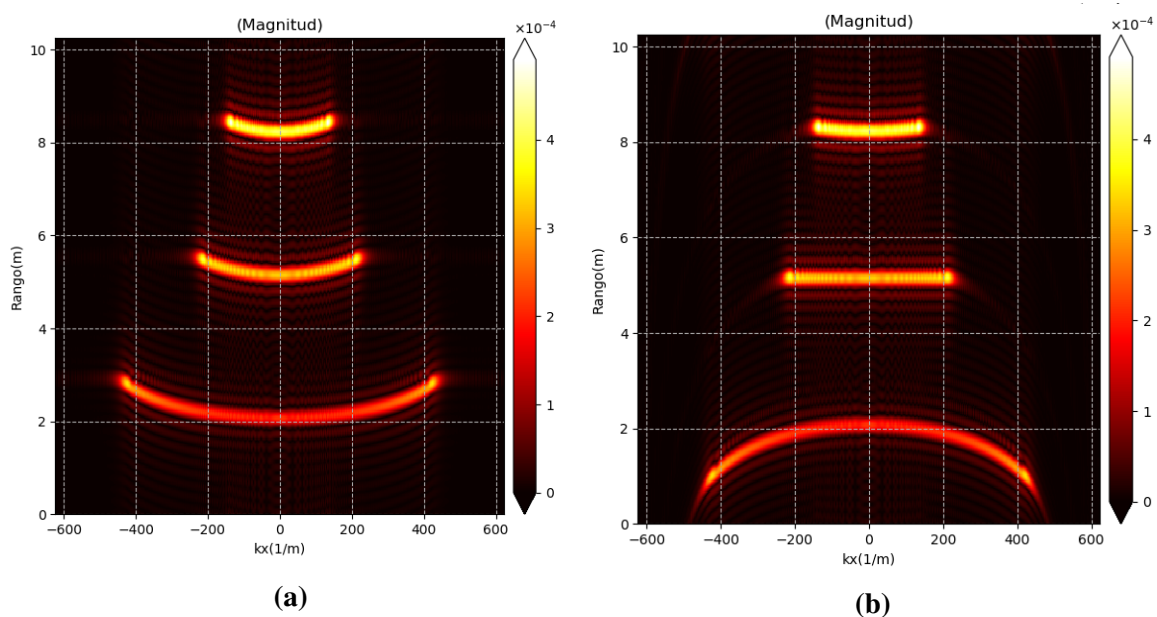


Figura 4.23: Corrección de curvatura para el ejemplo de los 3 blancos puntuales dada por la etapa de filtro adaptado del algoritmo RMA. (a) Antes de aplicar el filtro adaptado, (b) Después de aplicar el filtro adaptado. (Fuente propia)

escena de la imagen se reconstruyeron mejor que los blancos alejados del centro.

Interpolación STOLT

Esta etapa tiene el propósito de corregir las curvaturas residuales dejadas en la etapa del filtro adaptado (mostrado en la figura 4.24), y realizar un cambio de coordenadas. Comprende 2 pasos: cambio de coordenadas y una interpolación 1D; los cuales se explican a continuación.

1. Cambio de coordenadas

En este primer paso se realizó un cambio de coordenadas en el dominio de la frecuencia,

se pasó del plano $k_x - k_r$ al plano cartesiano $k_x - k_y$, para ello se emplearon las siguientes ecuaciones:

$$k_x = k_x \quad (4.22)$$

$$k_y = \sqrt{k_r^2 - k_x^2} \quad (4.23)$$

Al hacer este cambio, el plano del dominio $k_x - k_r$ se curva en el dominio $k_x - k_y$, las líneas de k_r constantes en el dominio $k_x - k_r$ son representadas como circunferencias en el dominio $k_x - k_y$, tal como se muestra en la figura 4.25.

De la ecuación (4.23) se observa que k_y puede tener tanto valores reales como imaginarios, sin embargo para el algoritmo solo se consideraron los valores reales, es decir, solo las combinaciones que cumplan la siguiente relación:

$$|k_r| \geq |k_x| \quad (4.24)$$

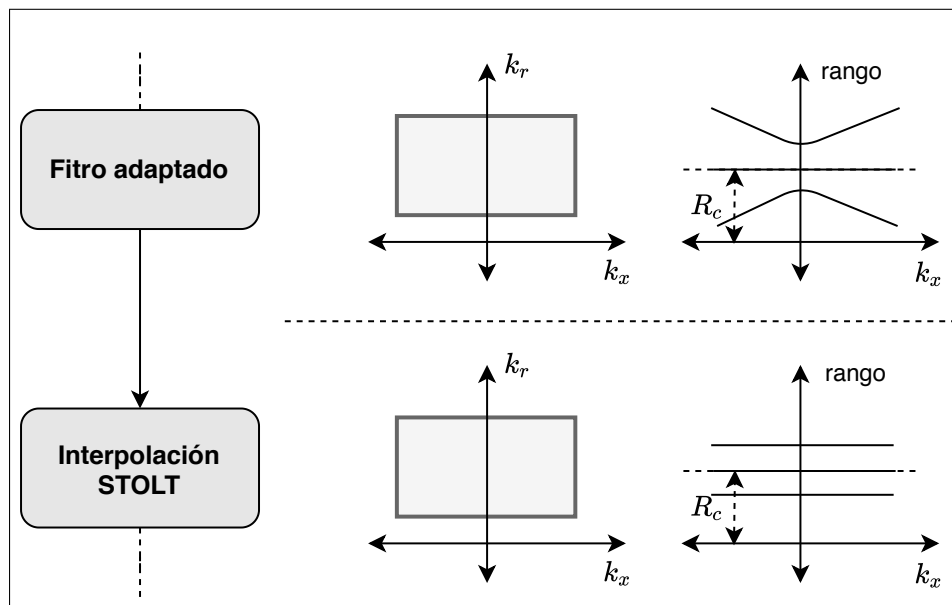


Figura 4.24: Corrección de las curvaturas residuales realizada por la etapa de Interpolación STOLT del algoritmo RMA. (Basado en [17])

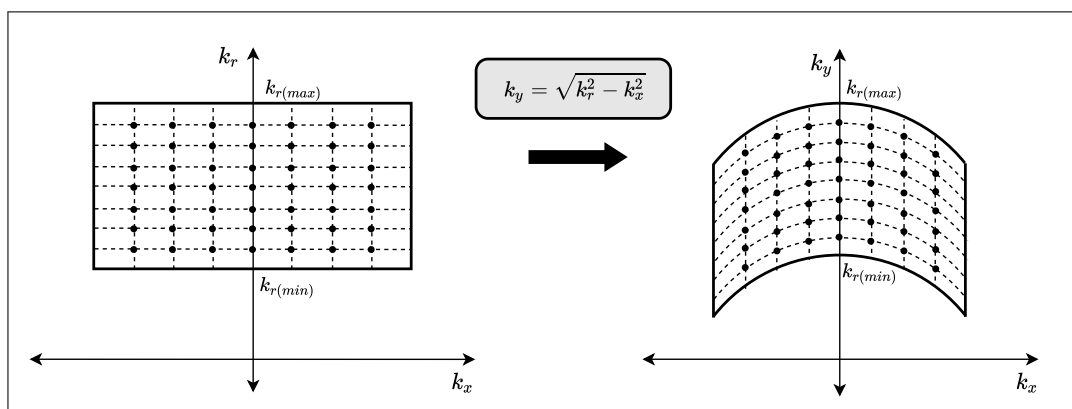


Figura 4.25: Cambio de coordenadas realizado en la etapa de Interpolación STOLT del algoritmo RMA. (Basado en [17])

Los casos que no cumplan esta relación se completaron con un valor infinito positivo.

2. Interpolación 1D

Al realizar el cambio de coordenadas la grilla de $S(k_x, k_y)$ se vuelve no uniforme (mostrado en la figura 4.25), por lo que en este paso se volvió a convertir en una grilla uniforme para poder efectuar la siguiente etapa del algoritmo.

La señal $S(k_x, k_y)$ tiene un grilla no uniforme solo en el eje k_y , por lo que se realiza el cambio solo en este eje, usando interpolación. Por lo tanto se realiza una interpolación 1D a lo largo del eje k_y , para obtener la señal $S_{it}(k_x, k_y)$ con grilla uniforme.

A continuación se explica el proceso de interpolación (mostrado en la figura 4.26). En primer lugar se determinó el dominio de interpolación $k_{y(it)}$, sus límites abarcan desde $\text{mín}(k_y)$ hasta $\text{máx}(k_y)$ los cuales se calculan de la ecuación (4.23), y tiene el mismo paso Δk que el vector k_r , esto para garantizar que la imagen se reconstruya hasta el rango máximo teórico. Su expresión se muestra en la siguiente ecuación:

$$k_{y(it)} = [\text{mín}(k_y), \dots, k_{y(it)}(i), k_{y(it)}(i+1), \dots, \text{máx}(k_y)] \quad (4.25)$$

Donde: $\Delta k = k_{y(it)}(i+1) - k_{y(it)}(i)$.

En segundo lugar se realizó la interpolación a lo largo del eje k_y , esto por cada valor de k_x . Las muestras de entrada, representados con puntos negros en la figura 4.26, lo conforman los valores de $S(k_x, k_y)$ ubicados en la grilla no regular. Las muestras de

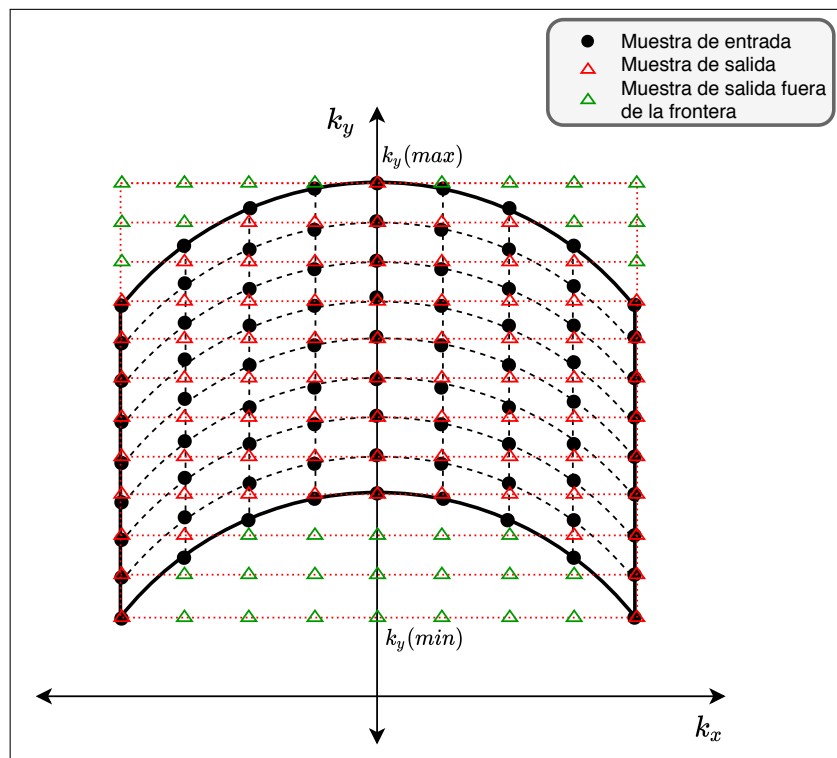


Figura 4.26: Proceso de interpolación 1D a lo largo del eje k_y realizado en la etapa de Interpolación STOLT del algoritmo RMA. (Basado en [17])

salida, representados con triángulos rojos en la figura 4.26, lo conforman los valores de $S_{it}(k_x, k_y)$ ubicados en una grilla regular, los cuales se calcularon de la interpolación. Los valores fuera del dominio de interpolación se completaron con ceros, en la figura 4.26 se muestran con marcadores triangulares verdes. Se escogió una interpolación lineal para las pruebas con datos teóricos y una interpolación polinomial cúbica para las pruebas con datos experimentales. La interpolación se calculó con la función *Interp1D* en Python, la cual ofrece la posibilidad de considerar con valores iguales a 0 a las muestras de salida que estén fuera de la frontera, todo dentro de una misma función.

En la figura 4.27 se muestra la señal interpolada tanto en magnitud como en fase para el ejemplo de los 3 blancos puntuales. En ella se puede observar efectivamente que la señal se curva, presentando una curvatura considerable debido al uso de una frecuencia de alto valor, $f_c = 15$ GHz. Las zonas blancas son los valores extrapolados, los cuales se completaron con ceros. Además se puede observar que el nuevo intervalo de k_y aumentó en 13 veces aproximadamente, y con ello aumentó también el tamaño de la matriz $S_{it}(k_x, k_y)$.

Por otro lado, en la figura 4.28 se observa la corrección de la curvatura para el ejemplo de los 3 blancos puntuales. Debido a la Interpolación STOLT se corrigió las curvas residuales ubicados en los extremos, las cuales fueron dejadas en la etapa del filtro adaptado (mostrado en la figura 4.23). Ahora el lugar geométrico de cada blanco, se convirtió en líneas horizontales.

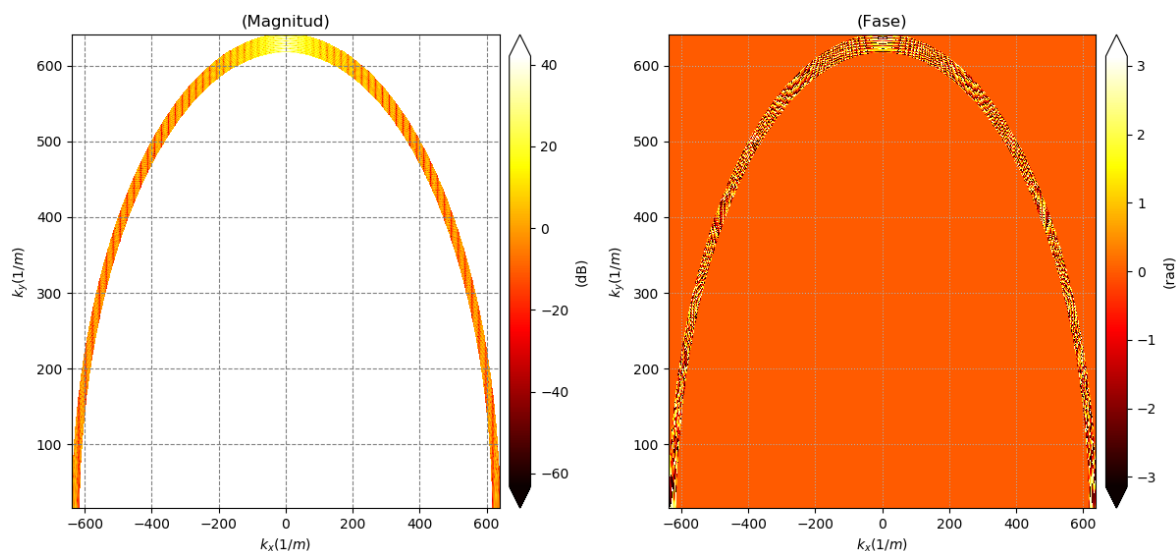


Figura 4.27: Etapa de Interpolación STOLT del algoritmo RMA, aplicado al ejemplo de los 3 blancos puntuales. Se muestran las imágenes resultantes en magnitud y fase. (Fuente propia)

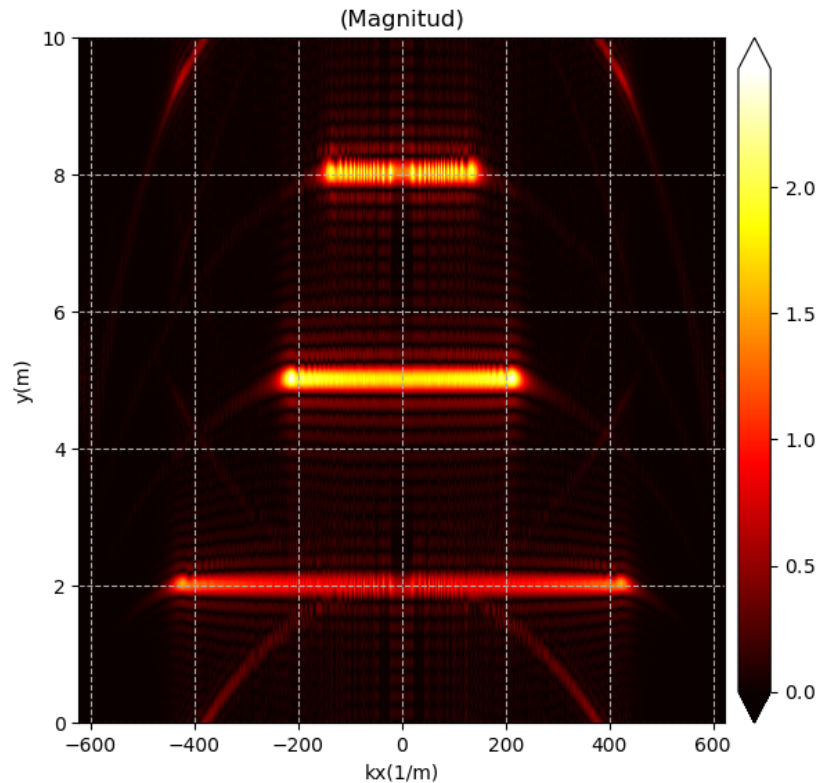


Figura 4.28: Corrección de la curvatura para el ejemplo de los 3 blancos puntuales después de efectuar la Interpolación STOLT. (Fuente propia)

2D-IDFT

En esta última etapa se realizó una Transformada Inversa de Fourier en ambas dimensiones (2D-IDFT) a la señal $S_{it}(k_x, k_y)$, dando como resultado la imagen SAR $s(x, y)$ en el dominio del espacio.

En la práctica esta etapa incluyó los siguientes pasos:

- Se aplicó un *fftshift* a $S_{it}(k_x, k_y)$ respecto a k_x .

$$S_{it}(k_x, k_y) = \text{fftshift}|_{k_x}\{S_{it}(k_x, k_y)\} \quad (4.26)$$

- Se calculó la 2D-IDFT de $S_{it}(k_x, k_y)$

$$s(x', y') = \mathcal{F}_{k_x, k_y}\{S_{it}(k_x, k_y)\} \quad (4.27)$$

- Se aplicó un *fftshift* a $s(x', y')$ en ambos ejes

$$s(x', y') = \text{fftshift}|_{x', y'}\{S_{it}(k_x, k_y)\} \quad (4.28)$$

- Se realizó un cambio de coordenadas, del plano $(x' - y')$ al plano Azimut-Rango, mediante las siguientes ecuaciones:

$$\text{Azimut} = x' \quad (4.29)$$

$$\text{Rango} = R_s - y' \quad (4.30)$$

La imagen SAR resultante al término de esta etapa está en el dominio Azimut-Rango.

4.2.2 Resultados

En la figura 4.29 se muestra la imagen SAR reconstruida para el ejemplo de los 3 blancos puntuales, al terminar de efectuar los pasos del algoritmo RMA. Se muestra la imagen SAR en magnitud y fase en el plano Azimut-Rango, y se observan los blancos ubicados en las posiciones esperadas. En la figura 4.30 se muestra cada blanco reconstruido por separado. Se observa la presencia de los lóbulos laterales en ambas dimensiones. Respecto a la resolución en rango, se observa que son iguales para los 3 blancos, mientras que la resolución en azimut varía con el rango, siendo por ello mayor en el blanco con $Rango = 8\text{ m}$, ambos comportamientos de la resolución son esperados.

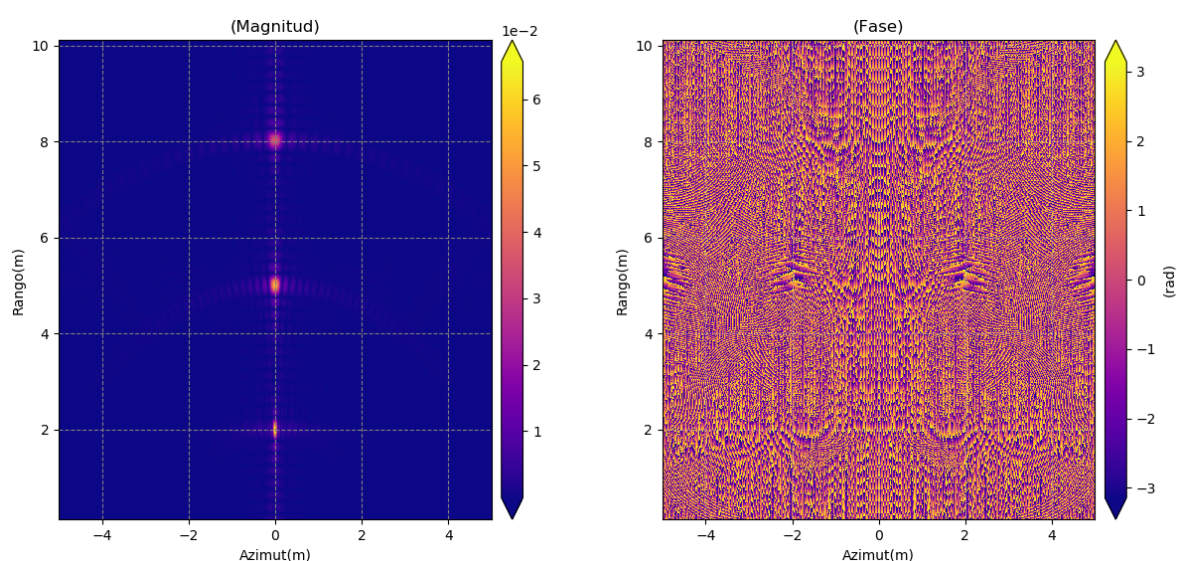


Figura 4.29: Imagen SAR del ejemplo de los 3 blancos puntuales obtenida con el algoritmo RMA. (Fuente propia)

4.2.3 Mejoras a la imagen

Reducción de lóbulos laterales

De igual forma que en el algoritmo FDBP, se usaron funciones de enventanado para atenuar los lóbulos laterales. En este caso se usó la ventana de Hamming debido a los mejores resultados que genera. La función de enventanado se aplicó al histórico de fase en ambas dimensiones antes de iniciar con los pasos del algoritmo.

En la figura 4.31 se muestra la reconstrucción del ejemplo de los blancos puntuales empleando una ventana Hamming. Se observa que los lóbulos laterales en ambos ejes se atenúan y la resolución aumenta de valor en ambos ejes, tal como lo esperado.

Disminución de las muestras

La decimación permite disminuir la tasa de muestreo de una señal discreta, lo que se traduce en una reducción de sus muestras. Una ventaja sería disminuir el tiempo de procesamiento debido a que se usarían señales discretas de menor tamaño. El único inconveniente es que la

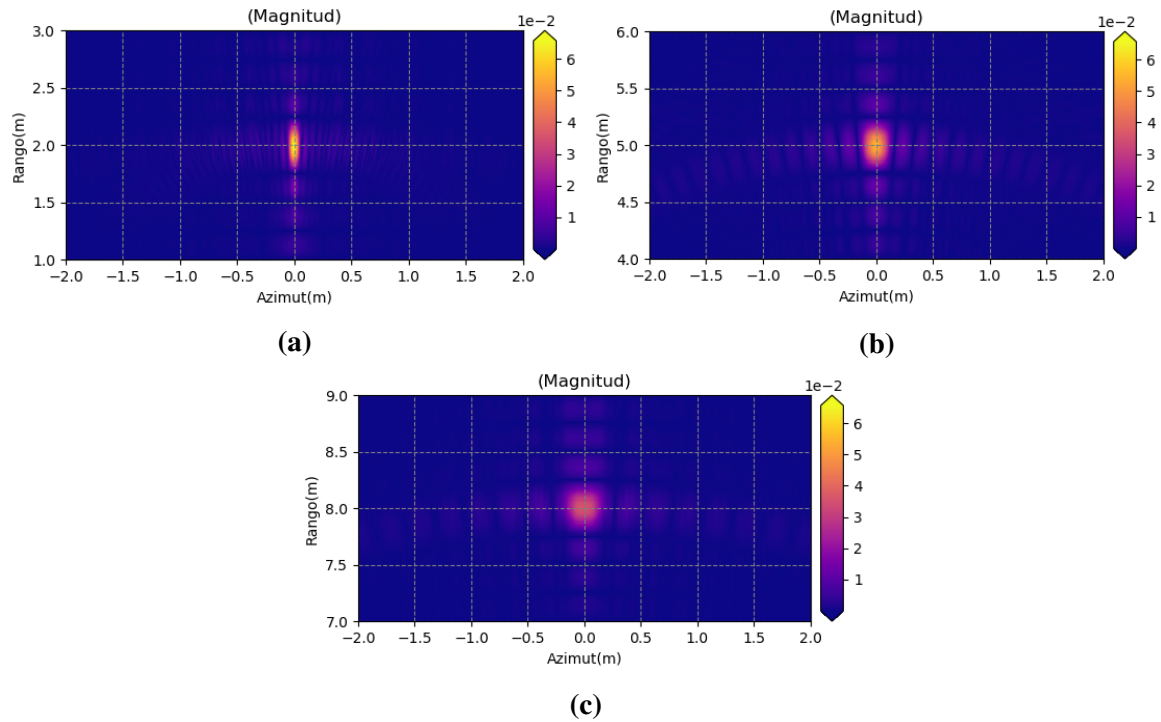


Figura 4.30: Imagen SAR ampliada en magnitud del ejemplo de los 3 blancos reconstruidos usando el algoritmo RMA. (Fuente propia)

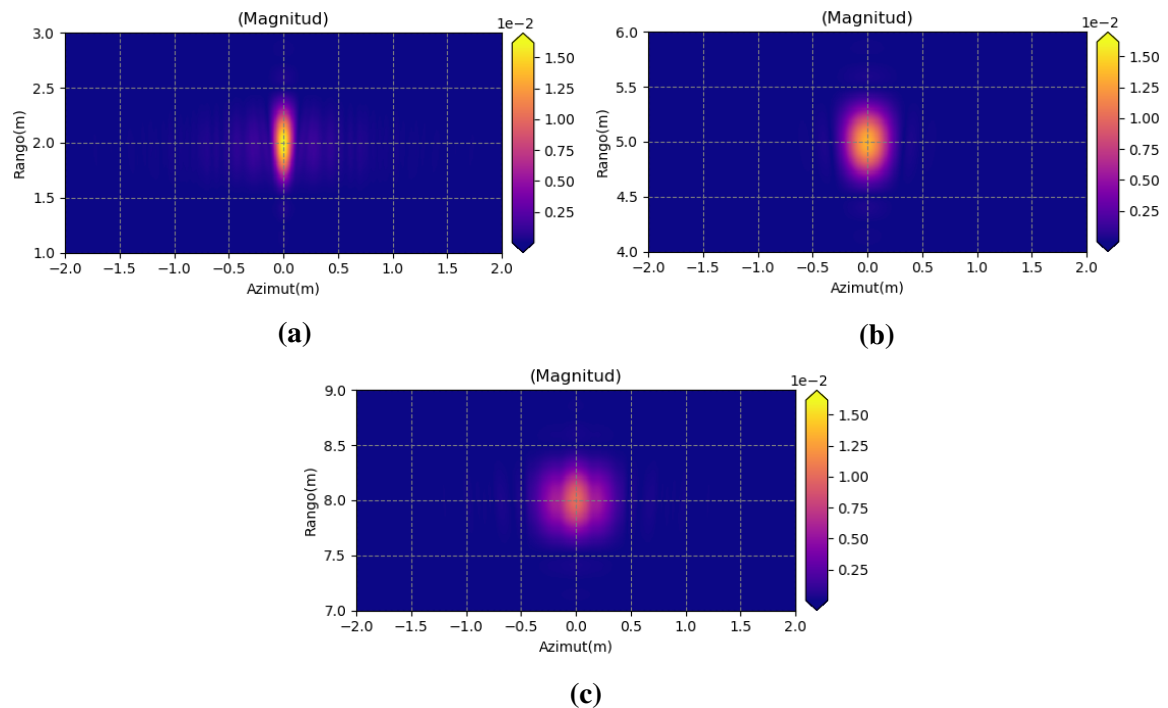


Figura 4.31: Efecto de la función de enventanado en la reconstrucción de un blanco puntual empleando el algoritmo RMA: (a)Blanco reconstruido sin ventana, (b)Blanco reconstruido con una ventana Hamming. (Fuente propia)

señal resultante de la decimación esta propensa al efecto de *aliasing*, generando distorsiones

en la señal ¹. En este caso del algoritmo RMA, se uso la decimación para tener control de la cantidad de píxeles de la imagen SAR (muestras de la señal) y para disminuir la cantidad de muestras de la señal $S_{it}(k_x, k_y)$, generada al término de la etapa de Interpolación STOLT, con el propósito de efectuar después la 2D-IDFT en un menor tiempo y sin problemas de recursos computacionales (memoria RAM).

En la práctica, la decimación se efectuó en el dominio de la frecuencia después de la interpolación STOLT. El primer paso fue determinar los valores de decimación en los ejes x e y a los cuales se les nombró D_x y D_y respectivamente, estos valores se calcularon teniendo en cuenta el número de píxeles deseados de la imagen final, denominados N_x y N_y , y el tamaño de la matriz $S_{it}(k_x, k_y)$ a cuyas dimensiones se les nombró N_{k_x} y N_{k_y} :

$$D_x = \text{int}(\text{ceil}(\frac{N_{k_x}}{N_x})) \quad (4.31)$$

$$D_y = \text{int}(\text{floor}(\frac{N_{k_y}}{N_y})) \quad (4.32)$$

Donde la función $\text{int}(\text{ceil}())$ aproxima al entero inmediato superior, mientras que $\text{int}(\text{floor}())$ aproxima al entero inmediato inferior.

Como segundo paso se dividió el espectro en tantas veces como lo indican los valores de decimación D_x y D_y y en la dirección respectiva, para finalmente sumar todas estas partes entre ellos. La señal resultante de la decimación tiene un tamaño $N_x \times N_y$. El eje k_x disminuye su tamaño a N_x muestras, mientras que el eje k_y disminuye a N_y muestras. Estos pasos se muestran en la figura 4.32, donde se realiza a modo de ejemplo la decimación de una señal, con los valores de decimación $D_x = 4$ y $D_y = 2$.

En la figura 4.33 se muestra la reconstrucción de un blanco, con posición (*azimut, rango*) = (0, 5), empleando el algoritmo RMA con decimación. En ella se muestran 3 casos: la de la parte superior es la imagen sin decimación, la del medio posee decimación $(D_x, D_y) = (2, 2)$ y la inferior $(D_x, D_y) = (5, 5)$. En cada parte de la figura 4.33 se muestra el espectro decimado y su respuesta en el dominio del espacio, se nota como mientras más aumentan los factores de decimación la imagen se hace más borrosa debido a la disminución de píxeles en la imagen, pero a la vez el tiempo de procesamiento disminuye, por ejemplo la señal sin decimar demora 3.53s mientras que la señal con decimación $D_x = D_y = 5$ demora 2.57s. La disminución del tiempo de procesamiento es más notorio cuando se procesan imágenes de mayor cantidad de píxeles.

El procedimiento de decimación anterior se aplicó a las reconstrucciones hechas con datos

¹ A las distorsiones en la señal también se les llama *artifacts* en la literatura. Es cualquier característica extraña que aparece en la imagen, pero que no está presente en la imagen original.

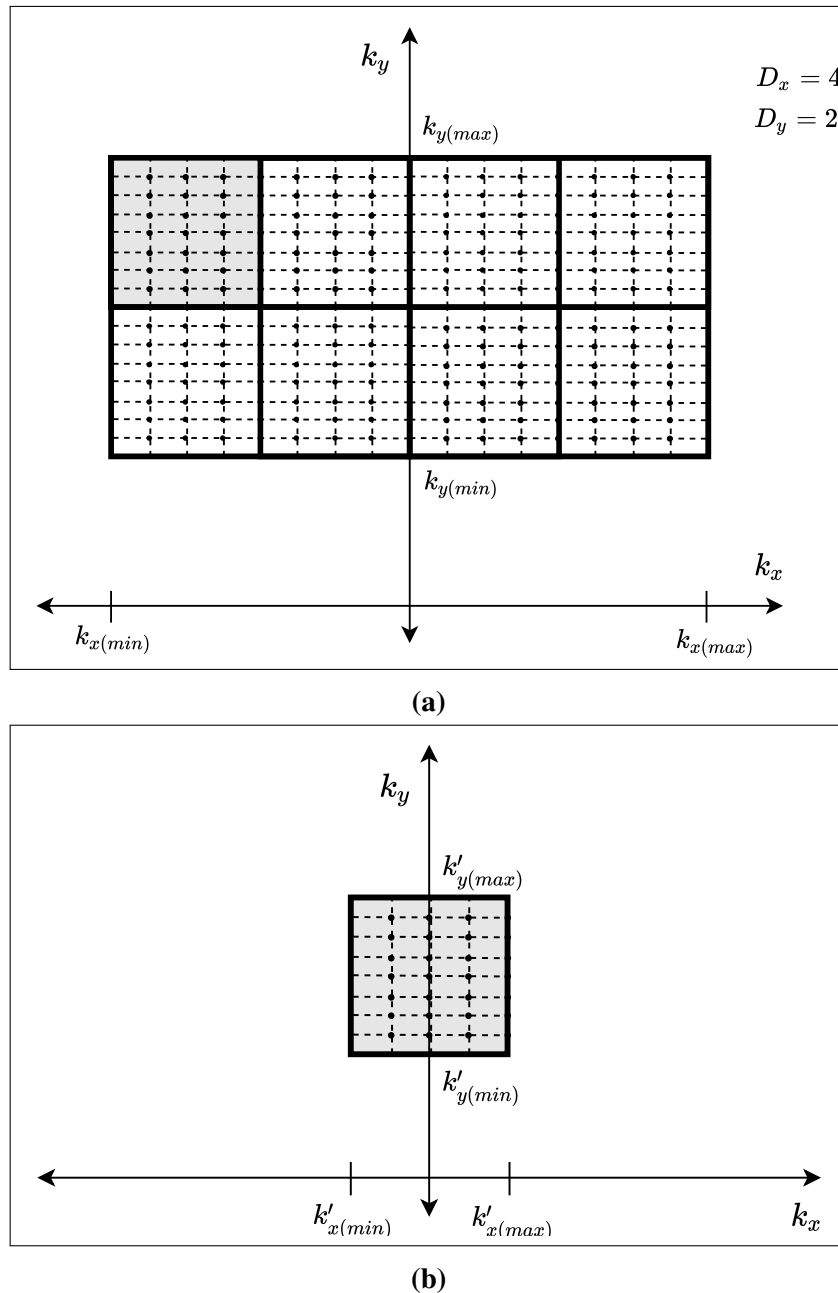


Figura 4.32: Ejemplo de decimación en el dominio de la frecuencia con $D_x = 4$ y $D_y = 2$. (a) Espectro de la señal original, (b) espectro decimado. (Fuente propia)

simulados ya que se tenía control sobre el tamaño de las imágenes, sin embargo el procedimiento cambió al momento de reconstruir con datos experimentales debido a la mayor cantidad de muestras del histórico de fase y las grandes dimensiones de la escena de la imagen. La decimación en este último caso se aplicó junto a la interpolación STOLT. Esto significó decimar inmediatamente después de interpolar una fila, con el propósito de evitar la creación de matrices grandes en la etapa de la Interpolación STOLT. Sin embargo, una desventaja de este método fue el alto tiempo de procesamiento, de 5 minutos aproximadamente para el ejemplo presentado. Por lo tanto, se optimizó el método anterior para bajar el tiempo

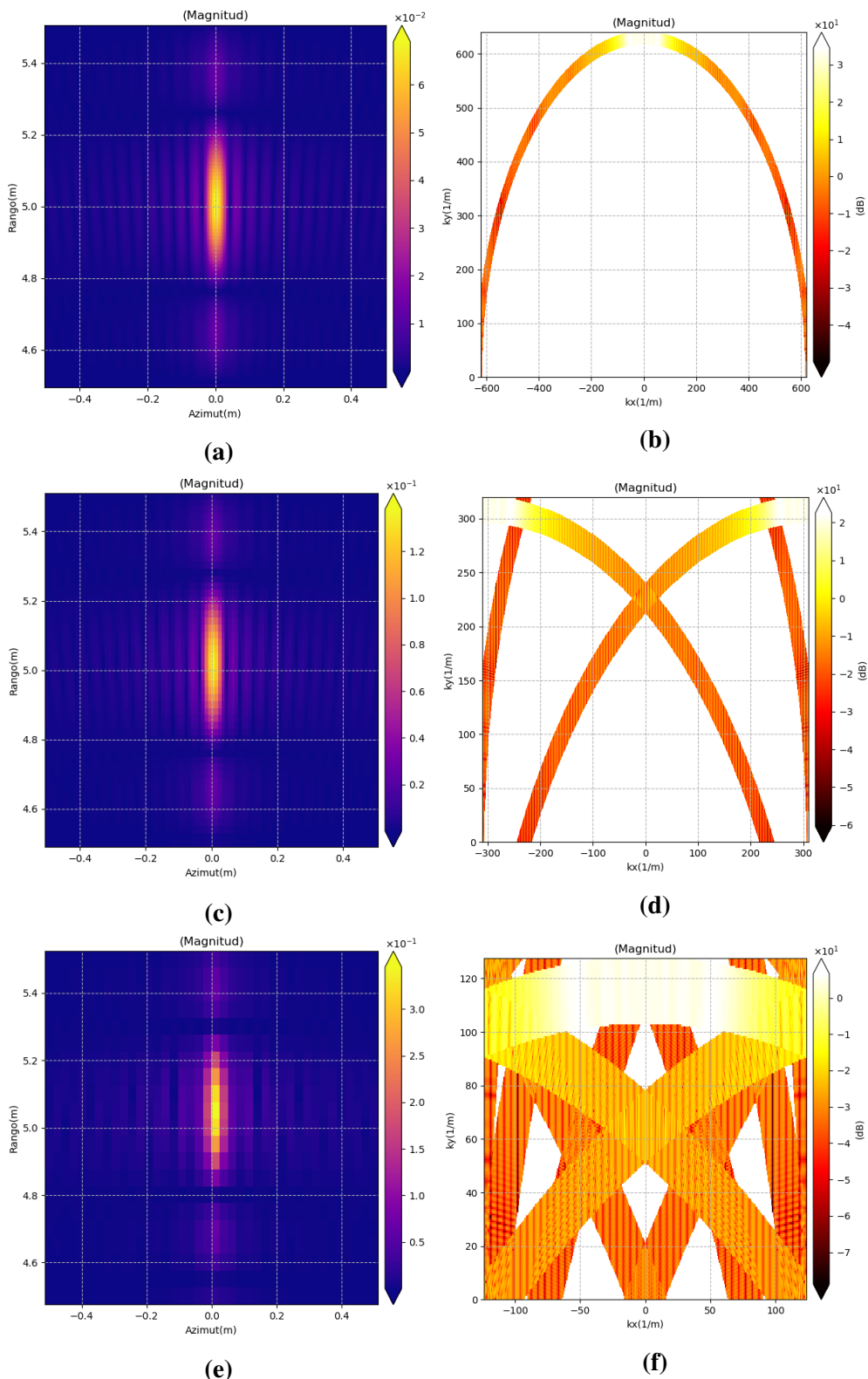


Figura 4.33: Imagen SAR en magnitud (lado izquierdo) junto a su espectro (lado derecho) de un blanco puntual aplicando el algoritmo RMA, para los siguientes casos: (a,b) sin decimación; (c,d) decimación $(D_x, D_y)=(2,2)$ y (e,f) decimación $(D_x, D_y)=(5,5)$. (Fuente propia)

de procesamiento. Se evitó interpolar en las zonas fuera de las fronteras ya que siempre se asignan por convención un mismo valor, 0 en este caso, y además conforman más del 50% de la señal interpolada final, mostrado en las zonas blancas de la figura 4.33. Mediante la creación de zonas de decimación, se determinó las zonas de señal alguna y se ignoró las que no tenían señal. Con este procedimiento se disminuyó el tiempo de procesamiento de 5 minutos sin decimación a 2 minutos con decimación.

Recorte del espectro

Esta mejora está relacionado a disminuir el tiempo de procesamiento del algoritmo recortando parte del espectro que no tiene contribución significativa en el resultado final. El recorte de espectro disminuye el tamaño de las señales, lo cual permite emplear menores recursos computacionales y disminuir el tiempo de procesamiento.

El recorte del espectro se aplico en las etapas iniciales del algoritmo con el propósito de disminuir el tamaño de las señales desde el principio. Específicamente, se le aplico a la señal $S(k_r, k_x)$ después de la etapa de 1D-DFT. El criterio que se usó para el recorte de espectro fue que todos los valores menores a la milésima parte (1/1000) del valor máximo o equivalentemente los que se encuentren a -60dB del máximo de la señal pueden ser ignorados. Para ello se creó una máscara solo con los valores que estuvieran fuera del criterio de exclusión, es decir los que sean mayores a la centésima parte del máximo, y se determinó el nuevo intervalo ya sea de k_x o k_y el cual contenga a todos los valores de la máscara. Este procedimiento se realizó manualmente, analizando para cada caso en particular.

Para el ejemplo de los 3 blancos puntuales, no se pudo realizar el recorte ya que la máscara es casi todo la señal, sin embargo cuando se le aplicó una ventana Hamming la máscara presentaba zonas que recortar los cuales son las zonas blancas mostradas en la figura 4.34. Observando la señal en esta figura se nota que solo se puede hacer recorte en el eje k_x donde el máximo y mínimo valor son aproximadamente +200 y -200 respectivamente (mostradas con líneas punteadas azules en esta figura), osea:

$$k_{x'} \in [-200, 200]$$

Con este recorte, el tamaño de la señal disminuyó en más del 50% en la dirección del eje k_x , disminuyendo también el tamaño de las siguientes señales que se crean a partir de ella.

4.2.4 Resumen de los pasos

A continuación se resume los pasos del algoritmo RMA que se aplicaron, junto con las mejoras, y en el orden respectivo.

1. Se obtuvo el histórico de fase.
2. Se aplicó una ventana Hamming.

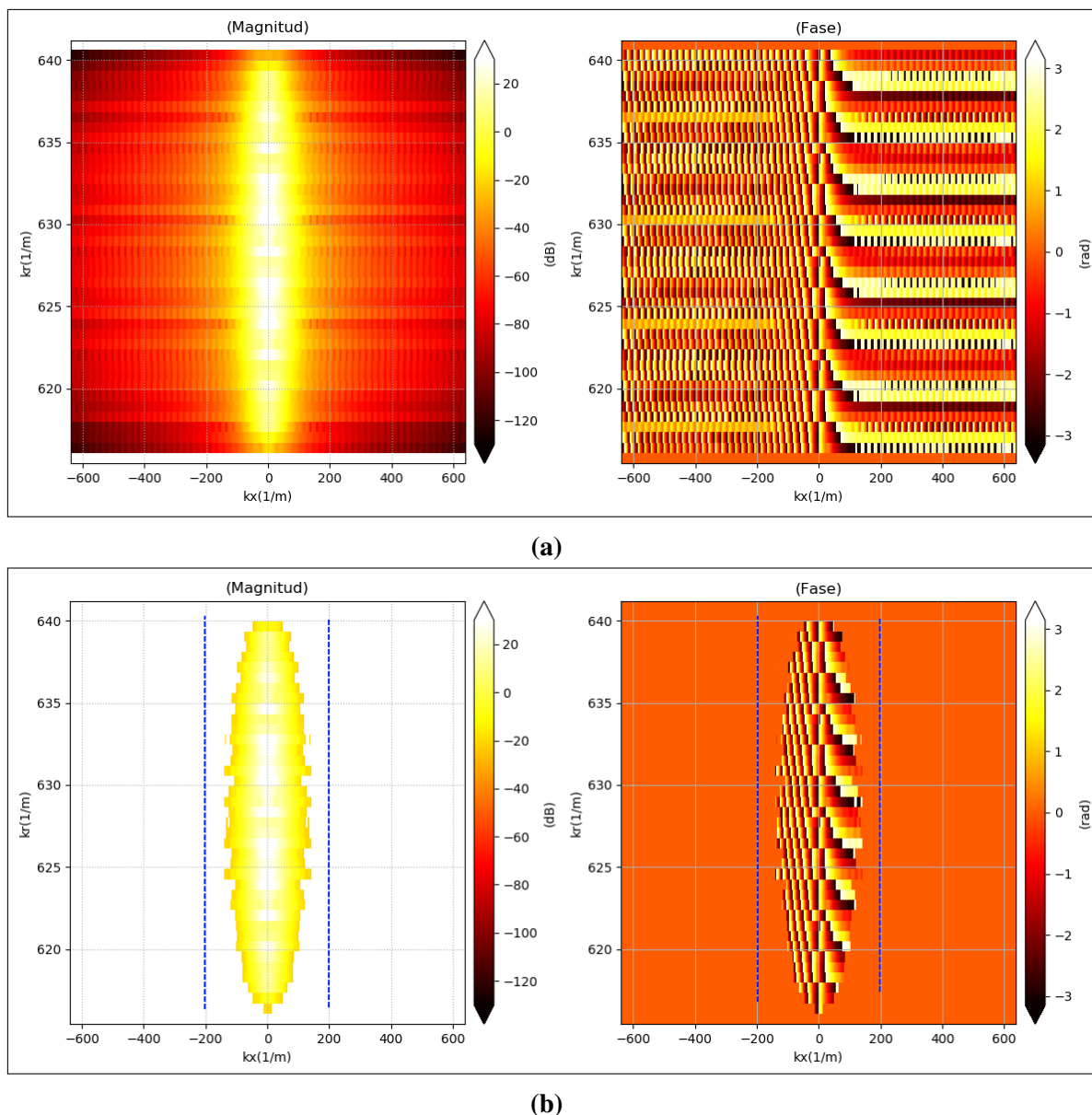


Figura 4.34: Recorte de espectro realizado después de la etapa de 1D-DFT del algoritmo RMA, para el ejemplo de los 3 blancos puntuales. (a) Espectro sin recortar, (b) máscara de recorte de espectro. (Fuente propia)

3. Se calculó una 1D-DFT respecto al eje de azimut, previamente se realizó un *zero – padding*.
4. Opcionalmente se aplicó un recorte de espectro.
5. Se realizó una compensación de fase mediante el filtro adaptado.
6. Se realizó una Interpolación STOLT.
7. Se aplicó la decimación en ambos ejes.
8. Se calculó una 2D-IDFT, obteniendo la imagen SAR.
9. Se cambió el plano de la imagen SAR al de Azimut-Rango.

4.3 Implementación del algoritmo DLIP

El algoritmo Discrete Linear Inverse Problem (DLIP) es un algoritmo de formación de imágenes SAR propuesto basado en la solución del problema inverso. En esta sección se explica el planteamiento, los pasos y las limitaciones del algoritmo. Para una mejor explicación de los pasos del algoritmo, se usa el ejemplo de los 3 blancos puntuales ubicados en:

$$(azimut, rango) = [(0, 2), (0, 5), (0, 8)]m$$

Se usan los parámetros de la tabla 4.1 con un solo cambio en el tamaño de la grilla: $d_x = d_y = 0.25$ m.

4.3.1 Planteamiento

El planteamiento de este algoritmo se basó en las ideas de la sección 2.4.3. Como se menciona en ella, la idea es aproximar el problema directo a un sistema lineal, y luego resolver el problema inverso.

Para la aproximación del problema directo a un sistema lineal se usó la ecuación (2.47), el cual se vuelve a reescribir empleando los términos del capítulo III:

$$S(u_k, f_i) = \sum_{n=1}^N I_n e^{(-j\frac{4\pi}{c} f_i \sqrt{(u_k - x_n)^2 + y_n^2})} \quad (4.33)$$

Donde I_n es la reflectividad y $(azimut, rango) = (x_n, y_n)$ es la posición de cada blanco puntual. La ecuación (4.33) permite dividir un blanco continuo en varios blancos puntuales. Para este caso, se extiende el concepto de I_n y (x_n, y_n) como la reflectividad y posición de cada píxel de la imagen SAR, asumiendo que el blanco se encuentra dentro de la escena de la imagen.

A continuación se expresó la ecuación (4.33) en forma de un sistema lineal de ecuaciones utilizando el formato dado en la referencia [22]. Por lo tanto el sistema lineal del problema directo tiene la siguiente expresión:

$$[S] = [A][I] \quad (4.34)$$

Con:

$$[S] = \begin{bmatrix} S_{(1,1)} \\ \vdots \\ S_{(k,i)} \\ \vdots \\ S_{(N_p, N_f)} \end{bmatrix}_{N_p N_f \times 1} \quad (4.35)$$

$$S_{(k,i)} = S(u_k, f_i)$$

$$[I] = \begin{bmatrix} I_1 \\ \vdots \\ I_n \\ \vdots \\ I_N \end{bmatrix}_{N \times 1} \quad (4.36)$$

$$I_n = I(\bar{r}_n)$$

$$[A] = \begin{bmatrix} \hat{A}_{(1,1),1} & \cdots & \hat{A}_{(1,1),n} & \cdots & \hat{A}_{(1,1),N} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \hat{A}_{(k,i),1} & \cdots & \hat{A}_{(k,i),n} & \cdots & \hat{A}_{(k,i),N} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \hat{A}_{(N_p,N_f),1} & \cdots & \hat{A}_{(N_p,N_f),n} & \cdots & \hat{A}_{(N_p,N_f),N} \end{bmatrix}_{N_p N_f \times N} \quad (4.37)$$

$$\hat{A}_{(k,i),n} = \hat{A}(u_k, f_i, \bar{r}_n)$$

Donde:

- $[S]$ es el histórico de fase mostrado en un vector columna de tamaño $N_p N_f$, en donde N_p es el número de posiciones del riel y N_f es el número de frecuencias usadas.
- $[I]$ es el vector de reflectividades, mostrados en un vector columna de tamaño N , en donde N es el número de píxeles totales de la escena de la imagen.
- $[A]$ es la matriz que identifica al sistema. Tiene dimensión igual a $(N_p N_f \times N)$.

La matriz $[A]$ tiene la siguiente expresión:

$$\hat{A}(u_k, f_i, \bar{r}_n) = e^{[-j \frac{4\pi}{c} f_i \sqrt{x_n^2 + (u_k - y_n)^2}]} \quad (4.38)$$

Donde: $\bar{r}_n = (x_n, y_n)$

Métodos de solución

La ecuación lineal del problema directo es:

$$[S] = [A][I] \quad (4.39)$$

Lo que se busca es calcular el vector $[I]$ teniendo el vector $[S]$, este procedimiento se le conoce como el problema inverso. Para resolver el problema inverso se recurre al concepto de pseudoinversa, ya que la matriz $[A]$ es una matriz rectangular en su expresión más general.

Usando la pseudoinversa, la solución está dada por la siguiente ecuación:

$$[I] = [A]^\dagger [S] \quad (4.40)$$

Donde $[A]^\dagger$ es la pseudoinversa de $[A]$ y presenta una de las siguientes expresiones dependiendo al rango de la matriz $[A]$:

1. Rango completo

Dentro de este caso, se encuentran 3 subcasos.

a) **Sistemas subdeterminados** ($N_p N_f < N$)

$$[A]^\dagger = A^T (AA^T)^{-1} \quad (4.41)$$

Donde AA^T debe de tener inversa.

b) **Sistemas sobredeterminados** ($N_p N_f > N$)

$$[A]^\dagger = (AA^T)^{-1} A^T \quad (4.42)$$

Donde $A^T A$ debe de tener inversa.

c) **Sistemas cuadráticos** ($N_p N_f = N$)

$$[A]^\dagger = A^{-1} \quad (4.43)$$

Donde A debe de tener inversa.

2. Rango deficiente

La solución de mínima longitud es:

$$A^\dagger = VS^{-1}U^T \quad (4.44)$$

Donde U, V, S son las matrices obtenidas de la descomposición en valores singulares (SVD) de la matriz A .

4.3.2 Pasos

Del planteamiento del algoritmo se obtuvieron los pasos, los cuales se muestran en la figura 4.35, y se procederán a explicar a continuación.

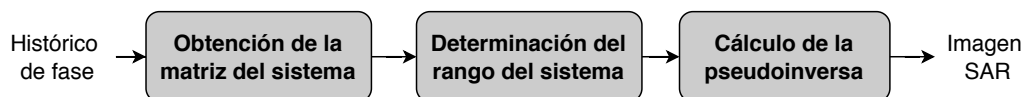


Figura 4.35: Pasos del algoritmo DLIP. (Fuente propia)

Pre-procesamiento

Antes de empezar con los pasos del algoritmo se realizaron dos operaciones. La primera de ellas fue definir la geometría de la imagen, osea, L_x, L_y, d_x, d_y . Con estos valores se definieron el número de píxeles totales N de la escena de la imagen, y por consiguiente se determinó la dimensión del vector $[I]$. La segunda operación fue redimensionar la matriz del histórico de fase, de una matriz rectangular convertirlo en un vector columna siguiendo el orden dado por la ecuación (4.35):

$$[S]_{N_p \times N_f} \rightarrow [S]_{N_p \cdot N_f \times 1}$$

Se determinaron las nuevas dimensiones de los vectores $[I]$ y $[S]$ para el ejemplo de los 3 blancos puntuales. La expresión de la dimensión del vector $[I]$ es:

$$[I] = [I]_{N \times 1}$$

Donde:

$$N = N_x \cdot N_y$$

$$N = \frac{L_x}{d_x} \cdot \frac{L_y}{d_y}$$

Reemplazando términos:

$$L_x = L_y = 10m$$

$$d_x = d_y = 25cm$$

$$\Rightarrow N = \left(\frac{10}{0.25} + 1\right) \cdot \left(\frac{10}{0.25} + 1\right)$$

$$N = 1681$$

Por lo tanto, la dimensión final de $[I]$ es:

$$[I] = [I]_{1681 \times 1}$$

Respecto a la dimensión de $[S]$, este tiene la siguiente expresión:

$$[S] = [S]_{N_p \cdot N_f \times 1}$$

Reemplazando términos:

$$N_p = 62$$

$$N_f = 41$$

Por lo tanto:

$$[S] = [S]_{2542 \times 1}$$

Obtención de la matriz del sistema

Esta etapa consiste en obtener la matriz $[A]$ que caracteriza al sistema. Para obtener los elementos de la matriz dada se empleó la ecuación (4.38) y se ubicaron en la matriz $[A]$ siguiendo el orden especificado por la ecuación (4.37).

Continuando con el ejemplo de los 3 blancos puntuales se obtuvo la matriz $[A]$ y se calcularon sus dimensiones:

$$[A] = [A]_{N_p \cdot N_f \times N}$$

Reemplazando valores:

$$[A] = [A]_{62 \cdot 41 \times 1681}$$

$$[A] = [A]_{2542 \times 1681}$$

La matriz $[A]$ posee más filas que columnas, por lo tanto pertenece a un sistema sobredeterminado.

Determinación del rango del sistema

En esta etapa se obtiene el rango de la matriz $[A]$. Continuando con el ejemplo de los 3 blancos puntuales, se calculó el rango de la matriz $[A]$:

$$\text{ran}(A) = 1681$$

Como el $\text{ran}(A)$ es igual al menor valor de las dimensiones de $[A]_{2542 \times 1681}$, entonces la matriz se clasificó como Rango completo.

Cálculo de la pseudoinversa

En esta última etapa se obtiene la imagen SAR. Para ello en primer lugar se obtiene la pseudoinversa de la matriz $[A]$, empleando una de las expresiones mencionadas en la sección 4.3.1. Luego por medio de la ecuación (4.40) se calcula el vector $[I]$. Finalmente se redimensiona el vector $[I]$ para obtener la imagen SAR:

$$[I]_{N \times 1} \rightarrow [I]_{N_y \times N_x}$$

Continuando con el ejemplo de los 3 blancos puntuales, se calculó en primer lugar la pseudoinversa $[A]^\dagger$. Como la matriz $[A]$ es de Rango completo y además pertenece a un sistema sobredeterminado, se usó la ecuación (4.42) para calcular la pseudoinversa. Una vez obtenido $[A]^\dagger$, esta matriz se multiplicó con el vector $[S]$, y se redimensionó para finalmente obtener la imagen SAR.

4.3.3 Resultados

Después de aplicar todos los pasos anteriores se obtiene la imagen SAR, se usa el ejemplo de los 3 blancos puntuales para explicar sus características más importantes.

En la figura 4.36 se muestra la imagen SAR para el ejemplo, tanto en magnitud como en fase. Se observa en la gráfica de magnitud que los 3 blancos fueron reconstruidos en sus posiciones correctas. Respecto a los lóbulos laterales, estos se encuentran a -280 dB del máximo aproximadamente, el cual es un valor muy atenuado, debido a los valores pequeños de los lóbulos laterales ya no se emplea ninguna función de enventanado. También se observa la aparición de distorsiones de señal en ambos extremos de la parte inferior de la gráfica de magnitud, los cuales poseen un valor de -145 dB del máximo aproximadamente.

4.3.4 Limitaciones de implementación

El algoritmo DLIP presenta 2 limitaciones al momento de implementarlos. La primera está relacionada a la condicional de que ciertas matrices como A , AA^T y $A^T A$ tienen que tener inversas para computar la pseudoinversa y así resolver la ecuación lineal y poder reconstruir la imagen. Esta condicional por lo tanto restringe a que solo algunos casos de la matriz A que cumplan esta condición puedan reconstruir correctamente imágenes usando este algoritmo.

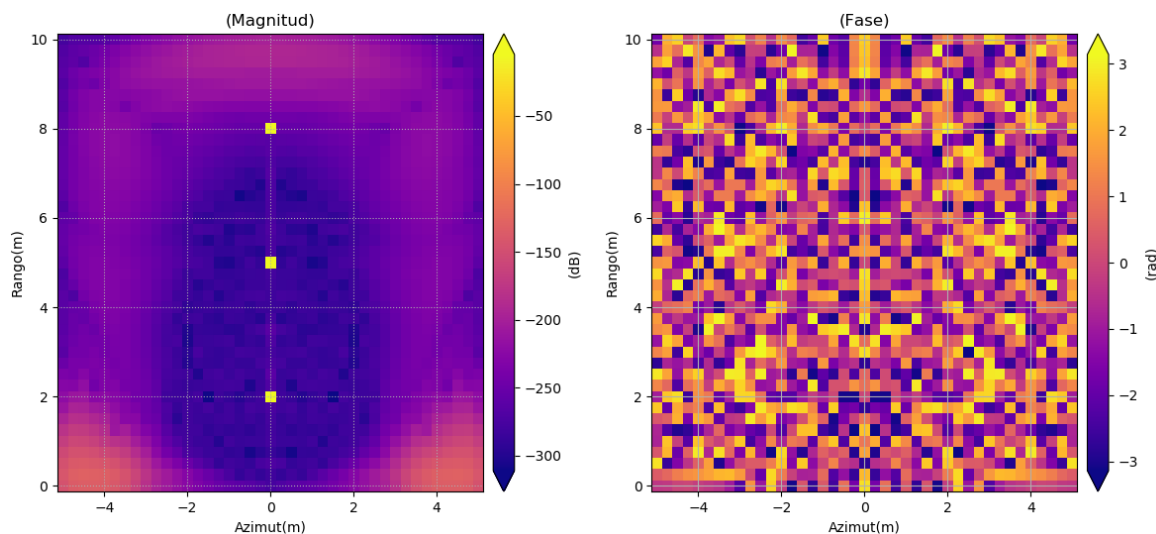


Figura 4.36: Imagen SAR del ejemplo de los 3 blancos puntuales obtenida empleando el algoritmo DLIP. (Fuente propia)

La segunda limitación está relacionada a la cantidad de recursos computacionales que se necesita para usar este algoritmo. Se puede notar que a lo largo de la implementación del algoritmo, la matriz $[A]$ es la variable de mayor tamaño, con dimensiones $(N_p N_f \times N)$, en otras palabras, el tamaño de la matriz es directamente proporcional al producto del número de frecuencias, el número de posiciones y el número de píxeles de la imagen. Para ver el tamaño de $[A]$ cuantitativamente se presentan 2 casos: una imagen pequeña simulada y una imagen real. En el primer caso se usó el ejemplo de los 3 blancos puntuales, el cual tiene los siguientes parámetros: Con los parámetros dados, la matriz $[A]$ tiene las siguientes dimensiones:

L_x, L_y	10m
d_x, d_y	0.25m
N_f	41
N_p	62

$$[A]_{2542 \times 1681}$$

Finalmente para obtener el tamaño de la variable $[A]$ en bytes se necesita saber el tamaño de cada elemento, como los elementos de la matriz son complejos estos tienen un tamaño de 16 bytes en Python, por lo tanto el tamaño de la matriz $[A]$ es:

$$2542 * 1681 * 16 \text{ bytes}$$

$$65.2 \text{ MB}$$

El tamaño de la variable $[A]$ es manejable y por lo tanto puede que este caso no requiera demasiados recursos computacionales.

El segundo caso es de una imagen real, el cual tiene los siguientes parámetros:

L_x	500m
L_y	1000m
d_x, d_y	1m
N_f	1000
N_p	150

Con los parámetros dados, la matriz $[A]$ tiene las siguientes dimensiones:

$$[A]_{150000 \times 500000}$$

Finalmente el tamaño de la variable $[A]$ es:

$$150000 * 500000 * 16 \text{ bytes}$$

$$1118 \text{ GB}$$

En este caso el tamaño de la variable $[A]$ es muy grande, lo cual se hace necesario tener grandes recursos computacionales, principalmente memoria RAM, para definir y procesar este tipo de variables, y por consecuencia para poder usar este algoritmo. La computadora que se usó para el procesamiento solo tenía 8 GB de memoria RAM, el cual permite procesar bien solo imágenes similares al primer caso. Por este motivo este algoritmo se limitó solo a pruebas con imágenes simuladas pequeñas.

4.3.5 Resumen de los pasos

A continuación se resumen los pasos del algoritmo DLIP:

1. Se cambió la dimensión del histórico de fase a un vector columna.
2. Se calculó la matriz $[A]$ que caracteriza al sistema.
3. Se calculó el rango de la matriz $[A]$
4. Se calculó la pseudoinversa de $[A]$
5. Se calculó el vector de reflectividades $[I]$
6. Finalmente, se obtuvo la imagen SAR redimensionando el vector $[I]$.

CAPÍTULO V ANÁLISIS Y DISCUSIÓN DE RESULTADOS

En el presente capítulo se analizan los resultados de la implementación de los 3 algoritmos con datos simulados. Luego se realiza la comparación entre estos algoritmos bajo los siguientes parámetros de comparación: calidad de reconstrucción de la imagen, calidad de enfoque y coste computacional. Por último, se analizan las imágenes SAR obtenidas con datos reales empleando 2 algoritmos, así como su influencia en la obtención de los mapas de desplazamientos.

5.1 Análisis de los resultados simulados

Una vez implementado los 3 algoritmos de formación de imágenes SAR se procedió a realizar pruebas principalmente con datos simulados y analizar los resultados obtenidos, con el objetivo de conocer los alcances y limitaciones de cada algoritmo. Se escogió un blanco que facilite el análisis en toda la escena de la imagen, por ello se decidió usar un blanco pseudo-continuo en forma de cuadrado que abarque casi toda la escena de la imagen. Este blanco estuvo conformado por un conjunto de blancos puntuales separados entre ellos una distancia Δl , tal como se muestra en la figura 5.1. En la tabla 5.1 se muestran los parámetros geométricos de este blanco.

Cada blanco puntual tiene magnitud uno, $I_n = 1$, y fase cero, $\phi_n = 0\text{rad}$. Por otro lado,

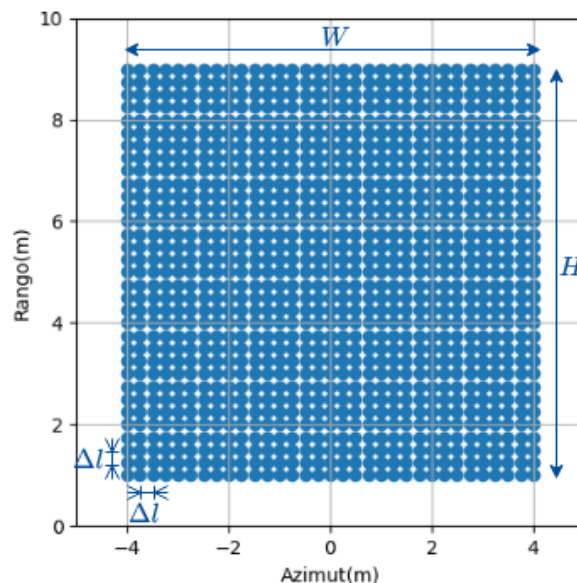


Figura 5.1: Blanco usado para el análisis de los algoritmos de formación de imágenes SAR. (Fuente propia)

Tabla 5.1: Parámetros geométricos del blanco pseudo-continuo.

Parámetro	Valor
Ancho W	9 m
Alto H	9 m
Grilla Δl	0.25 m
Posición inicial	(-4,1) m

para la simulación y la obtención de las imágenes SAR se usaron los parámetros de la tabla 5.2. La mayoría de los parámetros eléctricos y mecánicos se asemejan a los parámetros reales de un experimento de formación de imágenes SAR. Respecto a los parámetros geométricos, se eligieron valores pequeños para realizar pruebas rápidas, y especialmente para probar el algoritmo DLIP el cual presenta problemas de coste computacional cuando las imágenes son muy grandes.

Cada algoritmo usa como base el blanco y los parámetros mencionados, pero puede que se cambien algunos valores para realizar pruebas adicionales.

Tabla 5.2: Parámetros empleados para la simulación e implementación de los algoritmos de formación de imágenes SAR.

PARÁMETRO	VALOR
Frecuencia central f_c	15 GHz
Ancho de banda BW	600 MHz
Número de frecuencias N_f	41
Apertura sintética L_s	1.2 m
Ángulo de visión θ	90°
Número de posiciones N_p	248
Ancho de la imagen L_x	10 m
Alto de la imagen L_y	10 m
Grilla en azimut d_x	6.25 cm
Grilla en rango d_y	6.25 cm

5.1.1 Algoritmo FDBP

Con el blanco de la figura 5.1 y los parámetros de la tabla 5.2 se formó la imagen SAR empleando el algoritmo FDBP. Durante la implementación del algoritmo se usó una Ventana Hamming para atenuar los lóbulos laterales, y un interpolador lineal.

En la figura 5.2 se muestra la imagen SAR reconstruida tanto en magnitud como en fase, y en la tabla 5.3 se muestran los parámetros de salida como Δr , Δx_{max} , R_U , y t_s .

La imagen SAR obtenida en magnitud se analiza a continuación. En primer lugar se observa que el blanco fue reconstruido en su posición correcta, abarcando desde -4 m hasta 4 m en Azimut y de 1 m hasta 9 m en Rango. También se nota la presencia de unos artefactos en las esquinas de la parte superior de la imagen, impidiendo la reconstrucción correcta en esas zonas. La razón de la aparición de los artefactos se debe a que esas zonas pertenecen

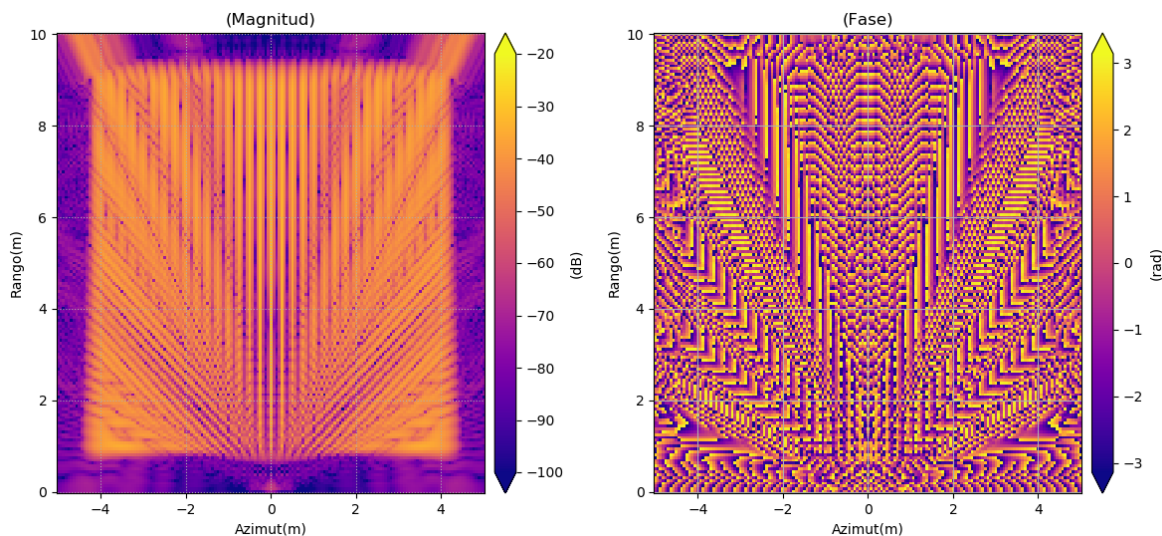


Figura 5.2: Imagen SAR de un blanco pseudo-continuo obtenido con el algoritmo FDBP. Se muestra la magnitud y fase de la imagen. (Fuente propia)

Tabla 5.3: Parámetros de salida de la reconstrucción del blanco pseudo-continuo realizado con el algoritmo FDBP

Parámetro	Valor
Resolución en rango Δr	0.25 m
Resolución en azimuth máxima Δx_{max}	0.075 m
Rango máximo permitido R_U	10.25 m
Tiempo de procesamiento t_s	1.03 s

al intervalo de extrapolación, el cual empieza a partir de la frontera de interpolación. La frontera o límite de interpolación está dada por el valor de R_U , el cual se puede representar gráficamente de forma aproximada como una circunferencia de radio R_U con centro en el origen, las zonas de la imagen menores a R_U pertenecen a la zona de interpolación, mientras que las zonas mayores a R_U pertenecen a la zona de extrapolación. Los blancos presentes en la zona de extrapolación no se reconstruyen bien y se generan artefactos en la imagen SAR. En este caso de la reconstrucción del blanco, el valor de $R_U = 10.25$ m, si se traza una circunferencia de 10.25 m con centro en $(0,0)$, las zonas de las esquinas superiores quedan fuera de la frontera, y es por eso que se generan artefactos. Por lo tanto se recomienda ubicar el blanco dentro de la zona de interpolación:

$$rango_{max} < R_U \quad (5.1)$$

Preferiblemente ubicarlo al centro de la escena de la imagen. Si no es posible cumplir la condición anterior, se debería de incrementar el valor de R_U a un valor que permita cumplir la condición, y eso se consigue disminuyendo el paso en frecuencia Δf de la señal SF-CW.

Siguiendo la sugerencia anterior, se usó otro blanco ubicado al centro de la escena de la imagen, mostrado en la figura 5.3, y se reconstruyó usando los mismos parámetros de la

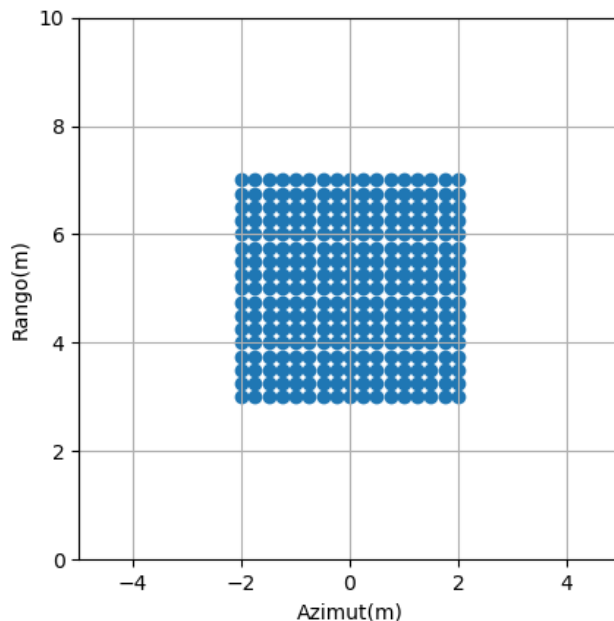


Figura 5.3: Blanco ubicado dentro de la zona de interpolación. (Fuente propia)

tabla 5.2. En la figura 5.4 se muestra la imagen SAR reconstruida en magnitud y fase. Como se puede observar en la gráfica de magnitud de la imagen, se reconstruyó la totalidad del blanco.

Otro detalle a observar en las gráficas de magnitud de las figuras 5.2 y 5.4 es el efecto de la resolución en la reconstrucción del blanco. En el eje de Azimut la resolución máxima es de $\Delta x = 0.075$ m y la separación entre blancos es $\Delta l = 0.25$ m, como Δx es menor a Δl el algoritmo los reconstruye como blancos individuales en la dirección de Azimut, es por ello que se observa los blancos individuales en ese eje. También se puede notar que Δx aumenta de valor a medida que aumenta el valor del Rango. Por otro lado en el eje del Rango sucede

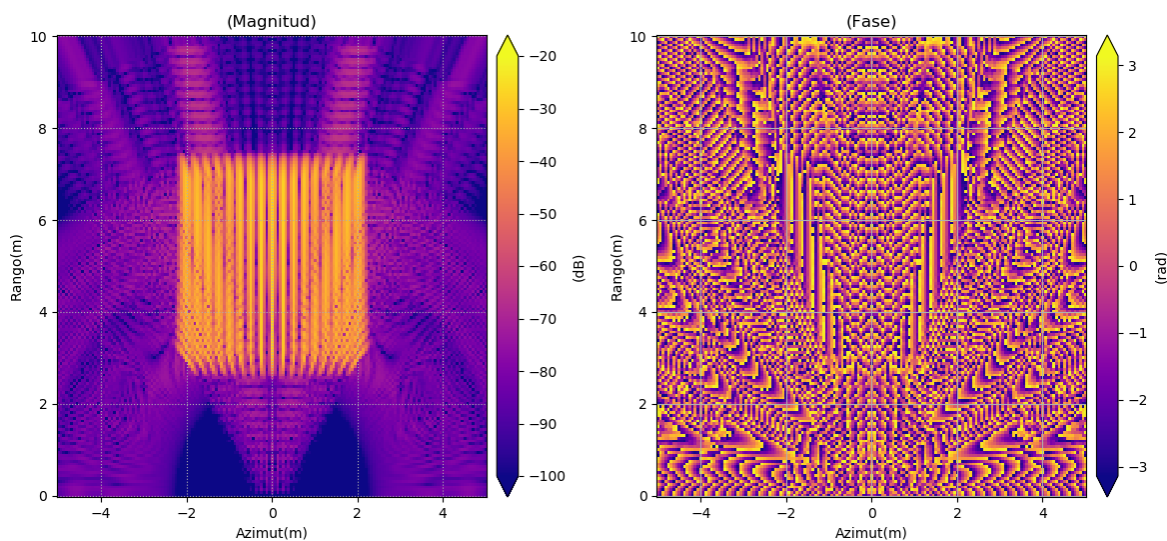


Figura 5.4: Imagen SAR de un blanco dentro de la zona de interpolación obtenido con el algoritmo FDBP. Se muestra la imagen en magnitud y fase. (Fuente propia)

lo contrario, como la resolución en rango Δr es igual a Δl el algoritmo lo reconstruye como un blanco continuo, es por ello que en la imagen se aprecian líneas verticales y diagonales, no pudiendo distinguir los blancos individuales.

5.1.2 Algoritmo RMA

Con el blanco de la figura 5.1 y los parámetros de la tabla 5.2 se reconstruyó el blanco empleando el algoritmo RMA. Se usó una ventana Hamming para reducir los lóbulos laterales, decimación en ambos ejes y recorte de espectro.

En la figura 5.5 se muestra la imagen SAR en magnitud y fase, y en la tabla 5.4 los parámetros de salida. Analizando la imagen SAR obtenida en magnitud, en primer lugar se puede observar que no se reconstruyó bien la parte inferior del blanco, específicamente recién a partir de un ángulo respecto al eje de azimut, mayor a 45° , el blanco se reconstruye en su ubicación correcta.

Se efectuó una segunda prueba para verificar si sigue habiendo estos problemas de reconstrucción y la presencia de este ángulo. Se eligió un blanco teórico ubicado en la parte inferior derecha de la escena de la imagen, justo en la zona de no reconstrucción, en la figura 5.6 se muestra este blanco. La imagen SAR reconstruida se muestra en la figura 5.7 tanto en magnitud como en fase. Como se puede observar en su gráfica de magnitud, la parte infe-

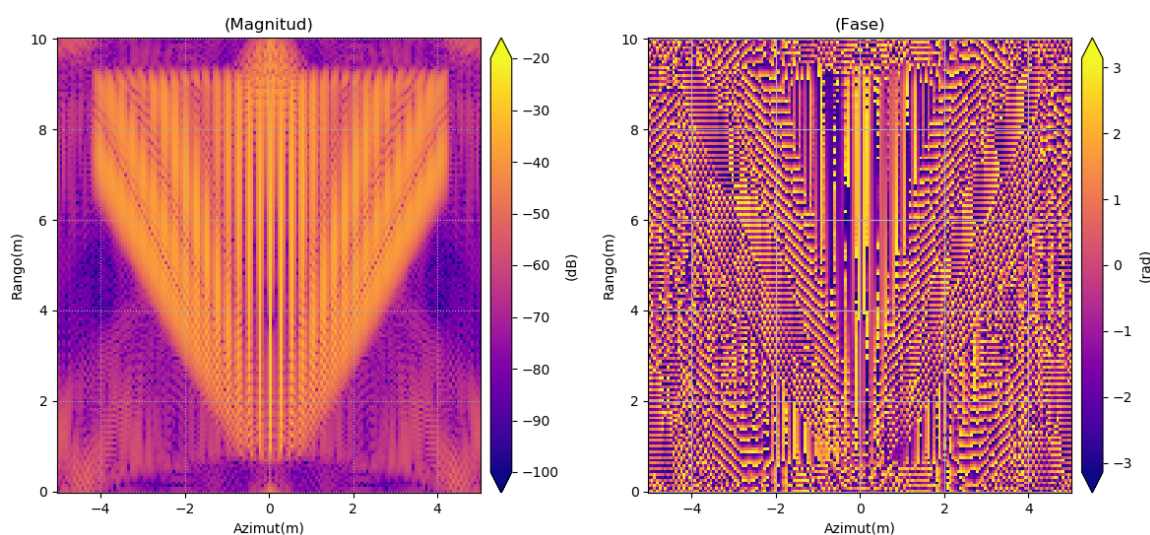


Figura 5.5: Imagen SAR de un blanco pseudo-continuo obtenido con el algoritmo RMA. Se muestra la magnitud y fase de la imagen. (Fuente propia)

Tabla 5.4: Parámetros de salida de la reconstrucción del blanco pseudo-continuo realizado con el algoritmo RMA

Parámetro	Valor
Resolución en rango Δr	0.25 m
Resolución en azimut máxima Δx_{max}	0.075 m
Rango máximo permitido R_U	10.25 m
Tiempo de procesamiento t_s	0.12 s

rior del blanco sigue sin reconstruirse, además sigue presentando un ángulo límite. Con este resultado, se comprobó que el algoritmo RMA no reconstruye blancos menores a un cierto ángulo.

Para comprobar si la no reconstrucción en ciertas áreas de la escena de la imagen está relacionado a algún parámetro propio del sistema, se realizó una tercera prueba. La hipótesis fue que la apertura sintética L_s cambia el ángulo límite y la zona de no reconstrucción. Para la prueba se incrementó el valor de L_s a 4 m, con esto cambió también el valor de N_p , se usó el blanco de la figura 5.1 y el resto de parámetros de la tabla 5.2. En la figura 5.8

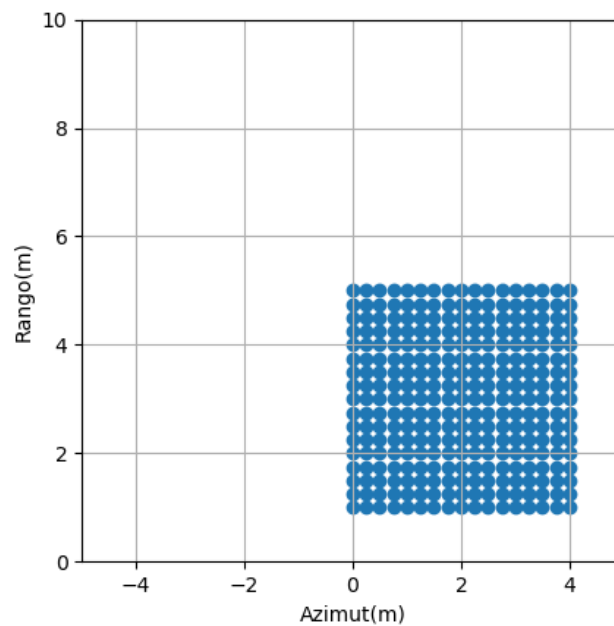


Figura 5.6: Blanco usado para comprobar la presencia de una zona de reconstrucción en el algoritmo RMA. (Fuente propia)

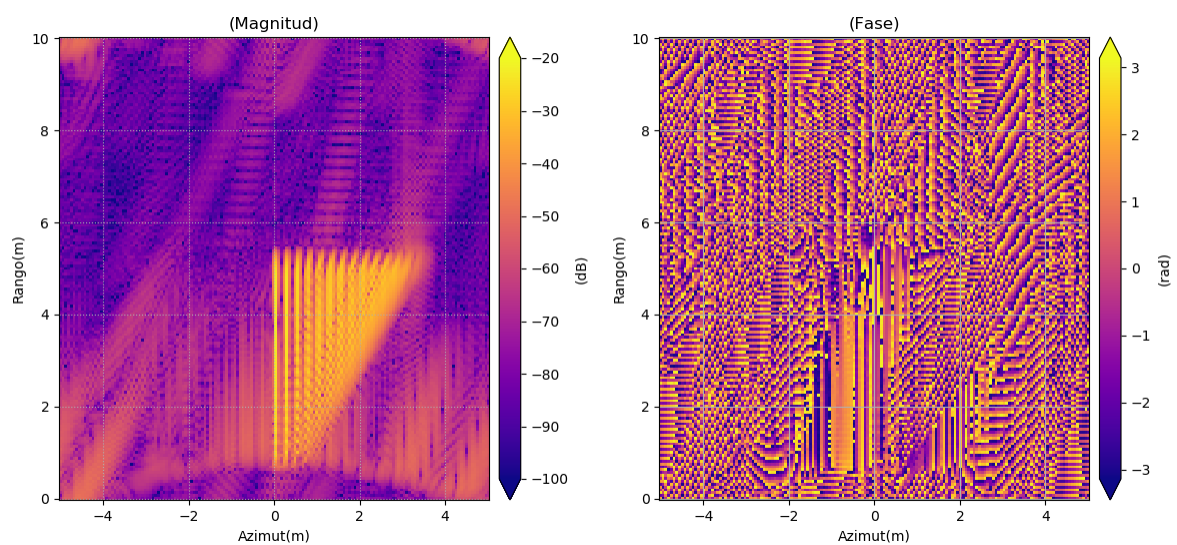


Figura 5.7: Imagen SAR de un blanco pseudo-continuo ubicado en la zona de no reconstrucción, obtenido con el algoritmo RMA. Se muestra la imagen en magnitud y fase. (Fuente propia)

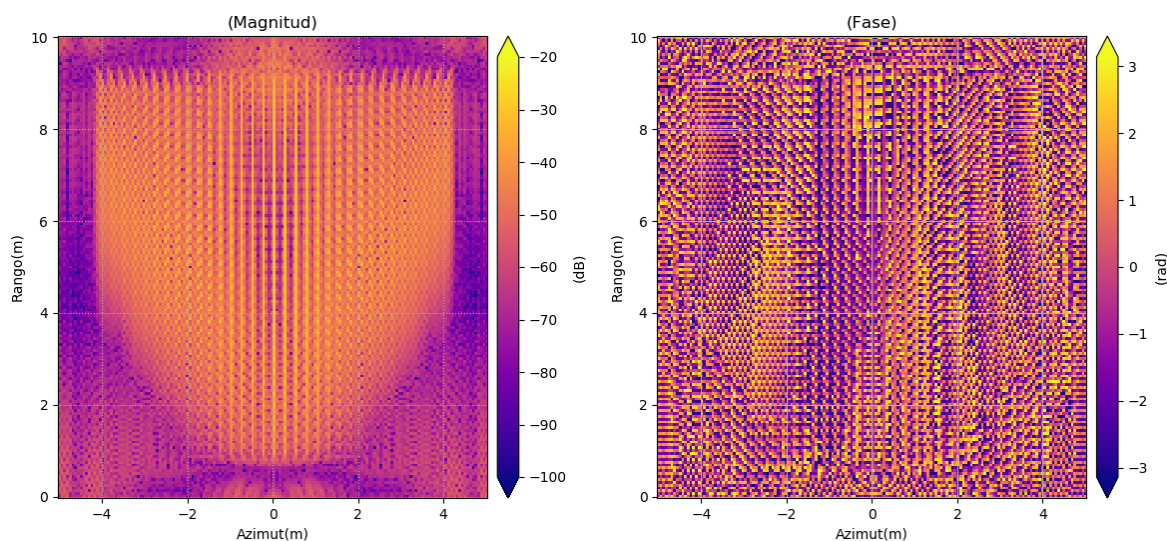


Figura 5.8: Imagen SAR de un blanco pseudo-continuo reconstruido con el algoritmo FDBP, incrementando el valor de L_s . Se muestra el resultado en magnitud y fase. (Fuente propia)

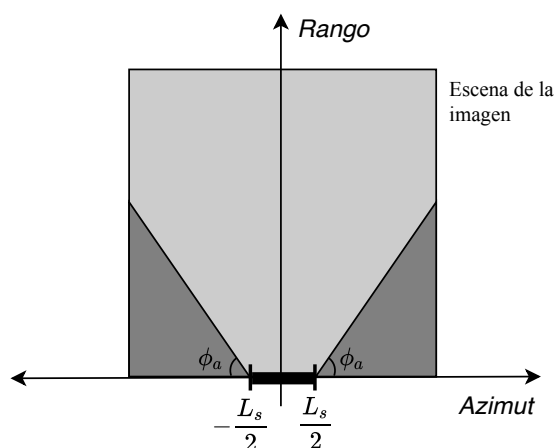


Figura 5.9: Zonas de reconstrucción del algoritmo RMA. Las zonas oscuras de la escena de la imagen indican las zonas de no reconstrucción, mientras que las zonas claras son las zonas de reconstrucción. (Fuente propia)

se muestra la imagen SAR reconstruida en magnitud y fase. Como se puede observar de su gráfica de magnitud, efectivamente hay más zonas del blanco que fueron reconstruidos, además el ángulo disminuyó a 45° aproximadamente. Otro detalle que se observa es que la zona de no reconstrucción empieza en ± 2 m aproximadamente, el cual es la posición que finaliza la apertura sintética, lo mismo se aprecia en la figura 5.5, en el cual el ángulo recién empieza en ± 0.6 m aproximadamente. Por lo tanto, con esta prueba se determina que la longitud de la apertura sintética L_s influye en la creación de un ángulo ϕ_a a partir del cual los blancos recién son reconstruidos correctamente, y que el ángulo inicia a partir de $\pm L_s/2$, tal como se muestra en la figura 5.9, las zonas oscuras en esta figura indican las zonas de no reconstrucción.

Por lo tanto para poder reconstruir un blanco completamente usando el algoritmo RMA se recomienda ubicar el blanco en la zona de reconstrucción, tal como se muestra en las zonas claras de la figura 5.9. Para conocer la zona de reconstrucción, se puede realizar una prueba preliminar de reconstrucción empleando un blanco similar al de la figura 5.1 el cual abarque la parte inferior de la escena de la imagen principalmente, luego se calcula ϕ_a , y con ese valor se determina las zonas correctas de reconstrucción. Otra recomendación complementaria es ubicar el blanco a una distancia menor a R_U .

A modo de ejemplo, se eligió un blanco siguiendo las recomendaciones anteriores. Como se continuó usando una apertura sintética de valor $L_s = 1.2$ m, entonces ella genera una zona de reconstrucción mostrada en la figura 5.5, y por lo tanto se elige un blanco dentro de esta zona, el cual es mostrado en la figura 5.3. Luego se reconstruye la imagen usando el algoritmo RMA, cuyo resultado se muestra en la figura 5.10. Como se puede observar en la imagen SAR en magnitud, el blanco fue reconstruido en su posición correcta de manera total.

La fase de la imagen SAR también está estrechamente relacionada con la apertura sintética. A continuación se realiza una prueba de ello. Se compara las fases de las imágenes SAR reconstruidas con distintas aperturas sintéticas. Previamente se cambian los siguientes parámetros:

$$d_x = d_y = 0.25$$

Se usa el blanco de la figura 5.1 y el resto de parámetros de la tabla 5.2. A continuación se forma la imagen SAR con el algoritmo RMA para 2 casos, en el primero $L_s = 1.2$ m y en el segundo $L_s = 4$ m. En la figura 5.11(a) se muestra la fase de la imagen SAR para $L_s = 1.2$ m

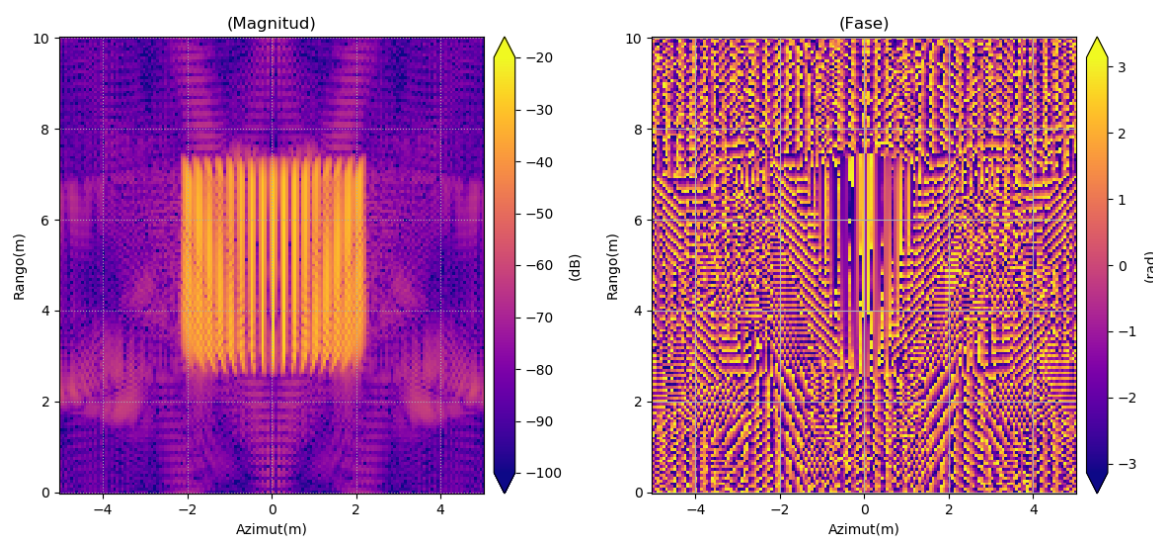


Figura 5.10: Imagen SAR de un blanco que cumple las recomendaciones del algoritmo RMA. Se muestra la imagen en magnitud y fase. (Fuente propia)

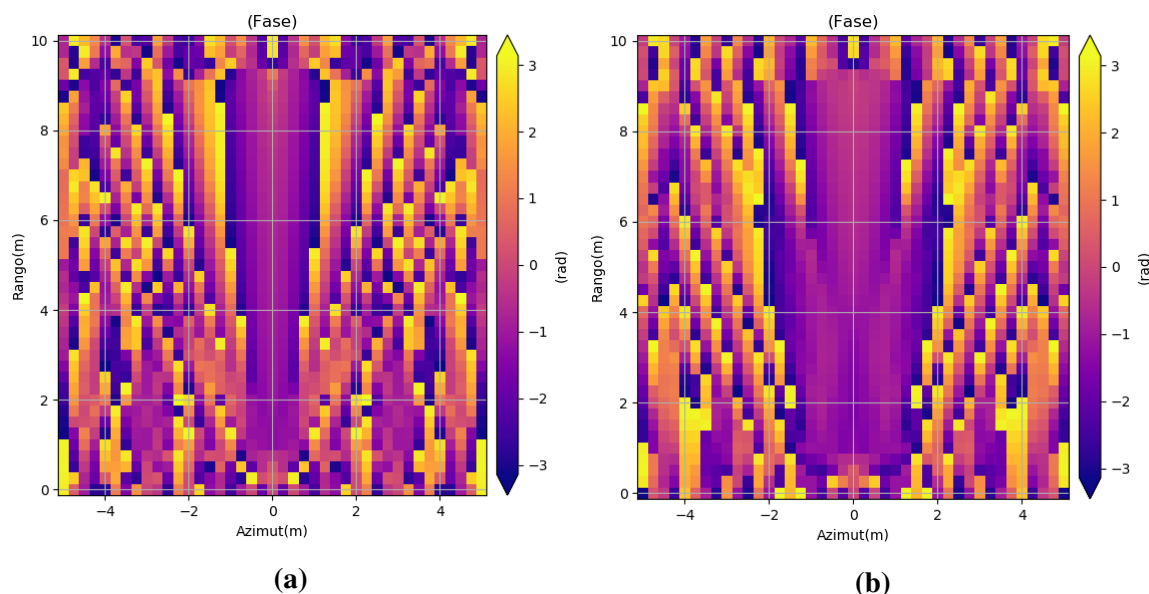


Figura 5.11: Efecto de la apertura sintética L_s en la fase de la imagen SAR obtenida con el algoritmo RMA. Reconstrucción de la imagen con: (a) $L_s = 1.2m$, (b) $L_s = 4m$. (Fuente propia)

Tabla 5.5: Parámetros de salida de la reconstrucción del blanco pseudo-continuo realizado con el algoritmo DLIP

Parámetro	Valor
Resolución en rango Δr	0.25 m
Resolución en azimuth máxima Δx_{max}	0.075 m
Rango máximo permitido R_U	10.25 m
Tiempo de procesamiento t_s	120 s

y en la figura 5.11(b) la fase para $L_s = 4m$.

Como se pueden observar en los resultados de las imágenes SAR en fase, hay más zonas continuas con fases similares para el segundo caso, $L_s = 4m$, es decir, cuando la apertura sintética es mayor. Las zonas de fase continua corresponden aproximadamente a la longitud de L_s . En el primer caso la zona continua es una franja de ancho 1.2 m, mientras que en el segundo caso, el ancho de la franja es mayor, llegando hasta 4 m aproximadamente. Por lo tanto se puede concluir que la apertura sintética influye en la obtención de zonas continuas de fase, mientras más grande es L_s mayor es la zona continua reconstruida.

5.1.3 Algoritmo DLIP

Se usó el blanco de la figura 5.1 y los parámetros de la tabla 5.2 con algunos cambios:

$$d_x = d_y = 0.25 \text{ m}$$

A continuación se reconstruyó el blanco empleando el algoritmo DLIP. En la figura 5.12 se muestra la imagen SAR en magnitud y fase. En la tabla 5.5 se muestran los parámetros de salida. De los resultados se puede observar que el blanco fue reconstruido en las posiciones correctas tanto en magnitud como en fase. La única observación del algoritmo es el alto tiem-

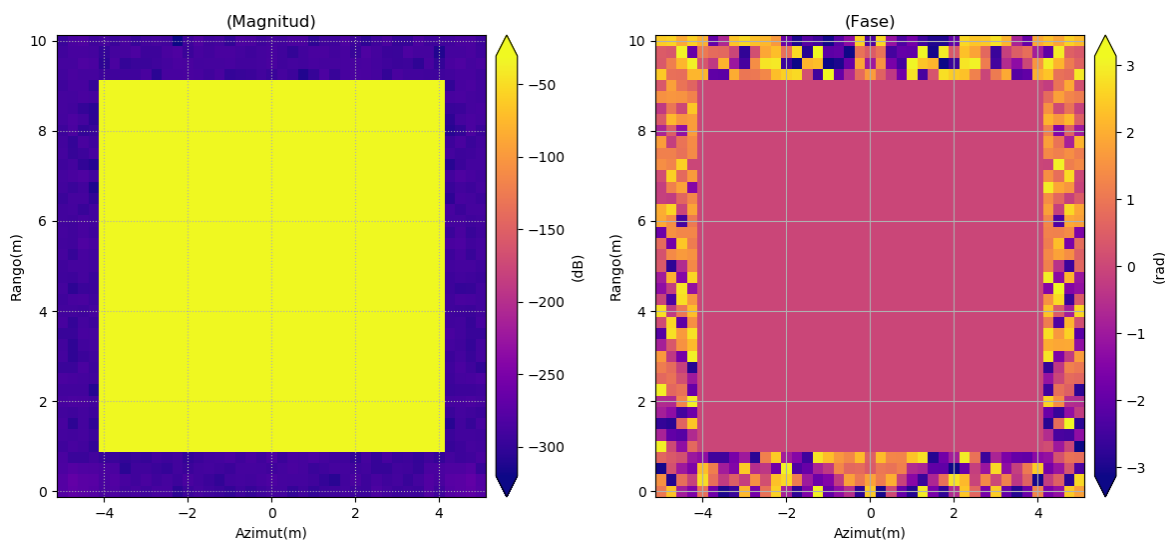


Figura 5.12: Imagen SAR obtenida con el algoritmo DLIP en magnitud y fase. (Fuente propia)

po de procesamiento $t_s = 120$ s. De igual forma que los algoritmos anteriores se recomienda que el blanco esté ubicado a un rango menor a R_U .

Cada caso de formación de imágenes SAR con el algoritmo DLIP es distinto y particular, ya que su solución viene de resolver un Problema de Inversión, el cual presenta varios casos y soluciones aproximadas. Por lo tanto no se pueden generalizar los resultados de unos casos al resto de casos.

5.2 Comparación de algoritmos

En esta sección se describe el análisis comparativo entre los 3 algoritmos. Los criterios que se consideraron son: calidad de reconstrucción de la imagen, calidad de enfoque y coste computacional. Estos criterios estuvieron basadas en la literatura de comparación de algoritmos SAR, explicados en la sección 2.4.4.

5.2.1 Calidad de reconstrucción de la imagen

El parámetro de calidad de reconstrucción de la imagen se refiere a la fidelidad con la que se reconstruye una imagen SAR, y se cuantifica con el error entre la imagen original (definida teóricamente) y la imagen SAR obtenida, tanto en magnitud como en fase. Para este propósito se usaron datos simulados debido a que permiten definir la imagen teórica. Los tipos de errores que se calcularon para el error de magnitud fueron: MSE, RMSE, MAE y ME; mientras que para computar el error de fase se usó el error RMSE. Se usaron los parámetros eléctricos y mecánicos de la tabla 5.6 para la obtención del histórico de fase simulado y la implementación de los algoritmos.

Tabla 5.6: Parámetros eléctricos y mecánicos empleados en la comparación de algoritmos de formación de imágenes SAR.

PARÁMETRO	VALOR
Frecuencia central f_c	15 GHz
Ancho de banda BW	600 MHz
Número de frecuencias N_f	41
Apertura sintética L_s	1.2 m
Ángulo de visión θ	75°
Número de posiciones N_p	238

Homogenización

El primer paso que se realizó fue homogenizar las imágenes. Esto significa usar los mismos parámetros geométricos de la escena de la imagen: tamaño y grilla, en la implementación de los 3 algoritmos. Por esta razón se eligieron parámetros que todos los algoritmos pudieran usar sin problema alguno, como coste computacional, entre otros. Como el algoritmo DLIP es el que presenta más problemas de coste computacional cuando el tamaño o grilla de la imagen es muy grande, la elección de los parámetros quedó determinado por este algoritmo. Los valores elegidos se muestran en la tabla 5.7. Para controlar el número de píxeles de la imagen final, el algoritmo RMA usa decimación, mientras que los otros algoritmos ya lo definen en uno de sus pasos.

Los límites de los valores de reflectividad también se homogenizaron. Se escogieron los siguientes valores:

$$v_{max} = -20 \text{ dB}$$

$$v_{min} = -100 \text{ dB}$$

El último aspecto a homogenizar fue la elección de un blanco adecuado. El blanco tiene que cumplir las recomendaciones dadas en la sección anterior para una correcta reconstrucción.

Las 2 recomendaciones se resumen a continuación:

1. Ubicar el blanco a una distancia menor a R_U .
2. Ubicar el blanco en la zona de reconstrucción, debido a las restricciones del algoritmo RMA.

Tabla 5.7: Parámetros geométricos usados para la comparación de algoritmos de formación de imágenes SAR.

PARÁMETRO	VALOR
Ancho de la imagen L_x	10 m
Alto de la imagen L_y	10 m
Grilla en azimut d_x	0.25 m
Grilla en rango d_y	0.25 m

Obtención de los errores

Las pruebas para el cálculo de errores se realizaron con 3 blancos distintos: 1 blanco puntual y 2 blancos pseudo-continuos. Para cada uno de ellos se empleó el siguiente método:

- Se definió el blanco teórico y la imagen teórica.
- Se obtuvo su imagen SAR empleando los 3 algoritmos de formación de imágenes SAR.
- Se calcularon los errores en magnitud y en fase de las imágenes SAR respecto a la imagen teórica. Los errores en magnitud que se calcularon fueron: MSE, RMSE, ME y MAE. El error en fase que se calculó fue el RMSE. Para computar todos estos errores se usaron las ecuaciones de la sección 2.4.4.

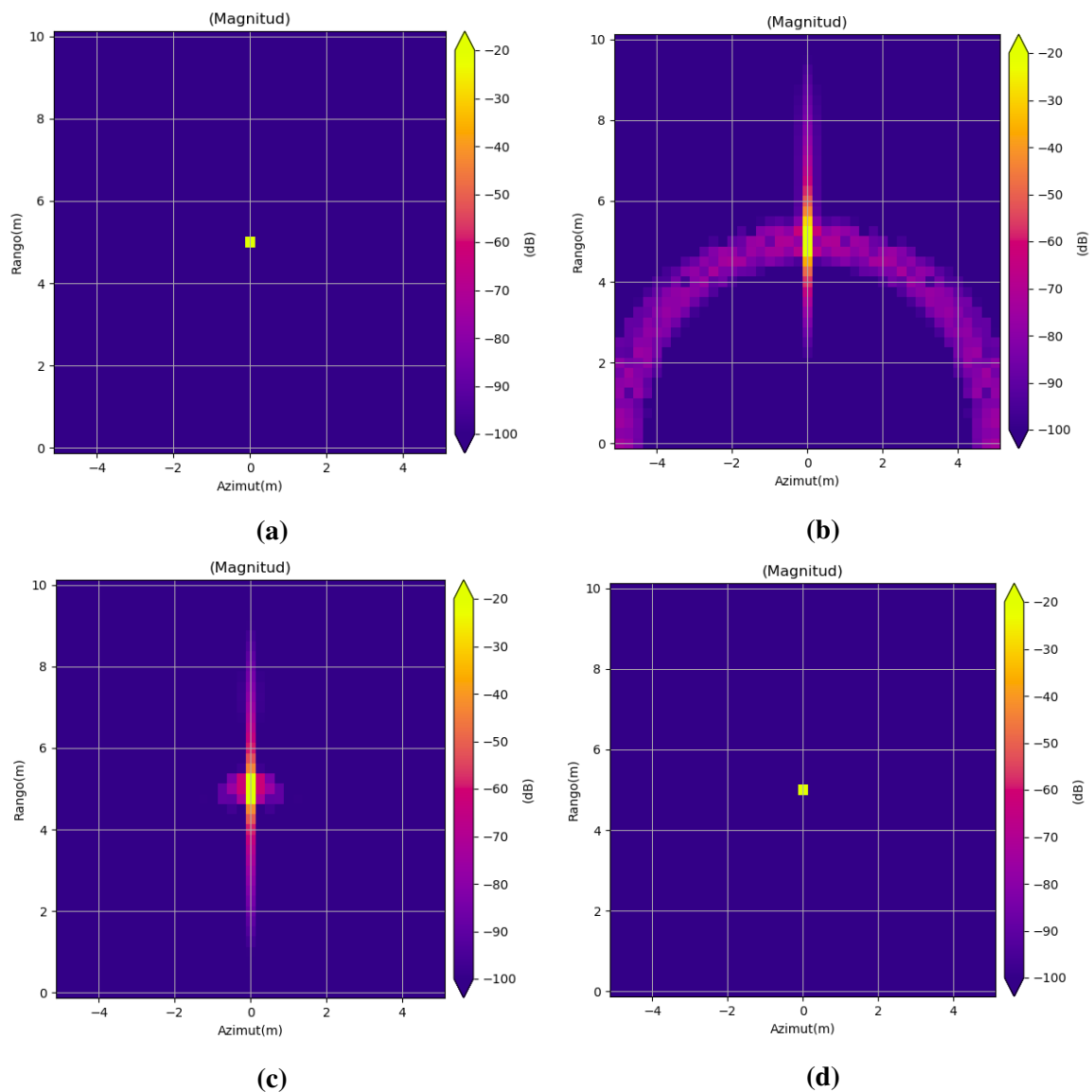


Figura 5.13: (a) Imagen teórica en magnitud de un blanco puntual ubicado en $(Azimut, Rango) = (0, 5)m$. Imagen SAR en magnitud del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)

1. PRIMER CASO: Blanco puntual

El primer caso fue un blanco puntual. Para la definición teórica, se utilizaron los siguientes parámetros:

- Posición: $(Azimut, Rango) = (0, 5)m$.
- Magnitud: $I_0 = 1$
- Fase: $\phi_0 = 0rad$

En posiciones distintas al blanco, la magnitud y fase se definieron valores iguales a cero. En la figura 5.13(a) se muestra la magnitud y en la figura 5.14(a) se muestra la fase de la imagen teórica. Luego se formaron las imágenes SAR con los 3 algoritmos. Los

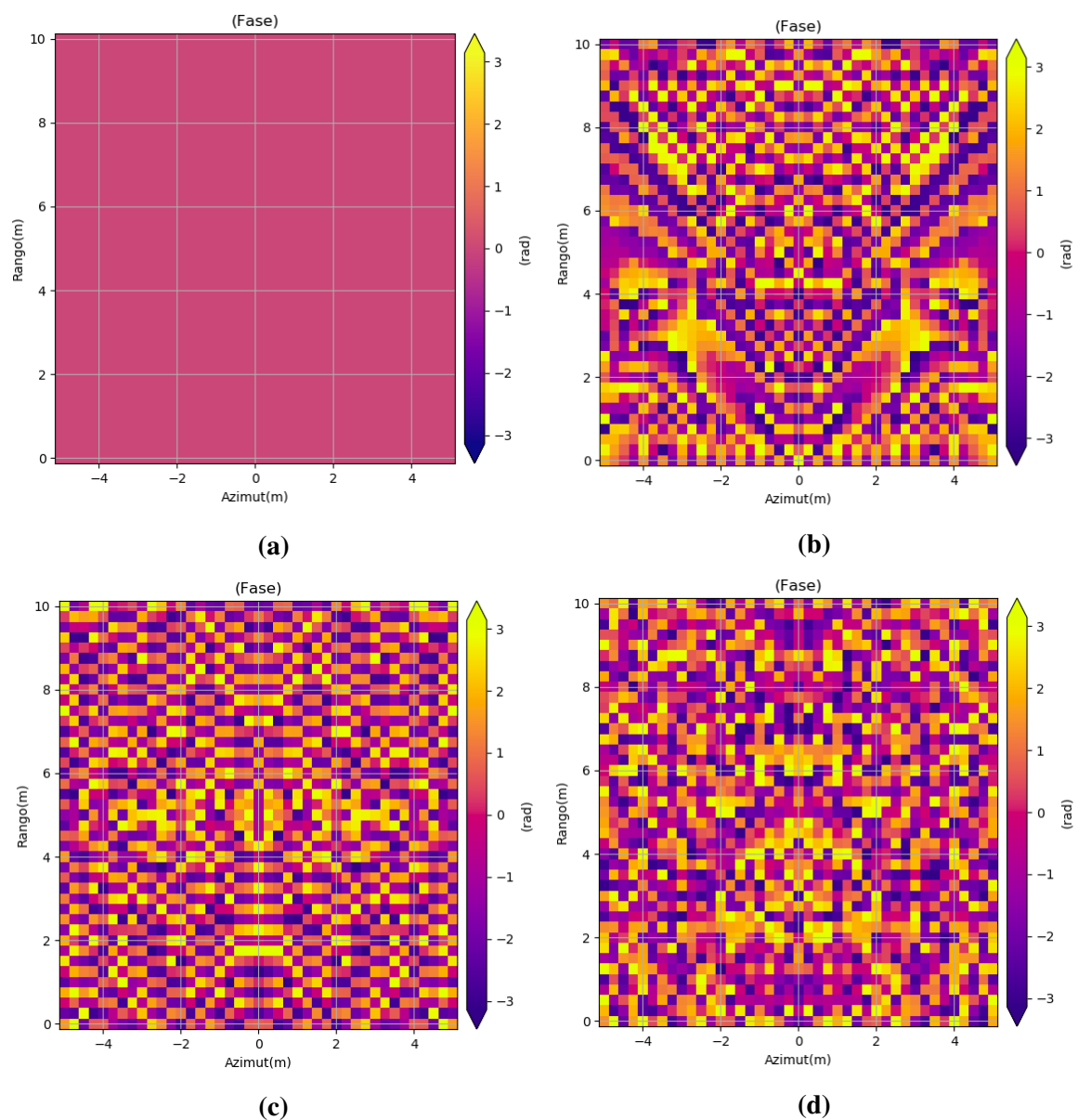


Figura 5.14: (a) Imagen teórica en fase de un blanco puntual ubicado en $(Azimut, Rango) = (0, 5)m$. Imagen SAR en fase del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)

resultados de la imagen SAR reconstruida en magnitud se muestra en las figuras 5.13(b), (c) y (d) respectivamente, mientras que la fase de las imágenes SAR se muestran en las figuras 5.14(b),(c) y (d). A continuación se realiza una comparación cualitativa de los resultados. De las imágenes SAR en magnitud, se puede observar que el blanco está en su posición correcta con los 3 algoritmos, también presenta lóbulos laterales notorios en los algoritmos FDBP y RMA, mientras que los lóbulos laterales en el algoritmo DLIP se encuentran debajo de $v_{min} = -100\text{dB}$ es por ello que no se visualizan en la imagen. Respecto a los resultados de fase, los límites de la fase van desde $[-\pi; \pi]$, y se observa que la fase en la posición del blanco es cercano a 0 rad en los 3 algoritmos. Como paso final se calcularon los errores en magnitud y en fase. Los resultados se muestran en la tabla 5.8. Como se puede observar en ella, el algoritmo DLIP es el que presentó los errores más bajos tanto en magnitud como en fase, seguido del algoritmo RMA, y finalmente por el algoritmo FDBP.

Tabla 5.8: Errores de magnitud y fase de las imágenes SAR de un blanco puntual, usando 3 algoritmos: FDBP, RMA, DLIP.

Error		Algoritmo		
		FDBP	RMA	DLIP
<i>Magnitud</i>	MSE	2.356e-04	2.255e-04	2.55e-28
	RMSE	1.53e-02	1.50e-02	5.93e-15
	ME	5.12e-01	4.23e-01	4.939e-14
	MAE	6.6e-04	6.23e-04	3.196e-15
<i>Fase</i>	RMSE	0.131	0.115	8.834e-16

2. SEGUNDO CASO: Blanco pseudo-continuo en forma de “T”

El segundo caso presentado es de un blanco pseudo-continuo en forma de “T”, formado por un conjunto de blancos puntuales. Para la definición teórica se utilizaron los siguientes parámetros:

- Separación entre blancos puntuales: $\Delta l = 0.25\text{ m}$
- Magnitud de los blancos puntuales: $I_0 = 1$
- Fase de los blancos puntuales: $\phi_0 = 0\text{ rad}$

En posiciones distintas al blanco la magnitud y fase se definieron con valores igual a cero. En la figura 5.15(a) se muestra la magnitud y en la figura 5.16(a) la fase de la imagen teórica.

Como segundo paso se obtuvieron las imágenes SAR empleando los 3 algoritmos de formación de imágenes. En las figuras 5.15(b), (c) y (d) se muestran las imágenes SAR en magnitud usando los algoritmos FDBP, RMA y DLIP respectivamente. En esas figuras se puede observar que el blanco fue reconstruido en la posición correcta, también se observa

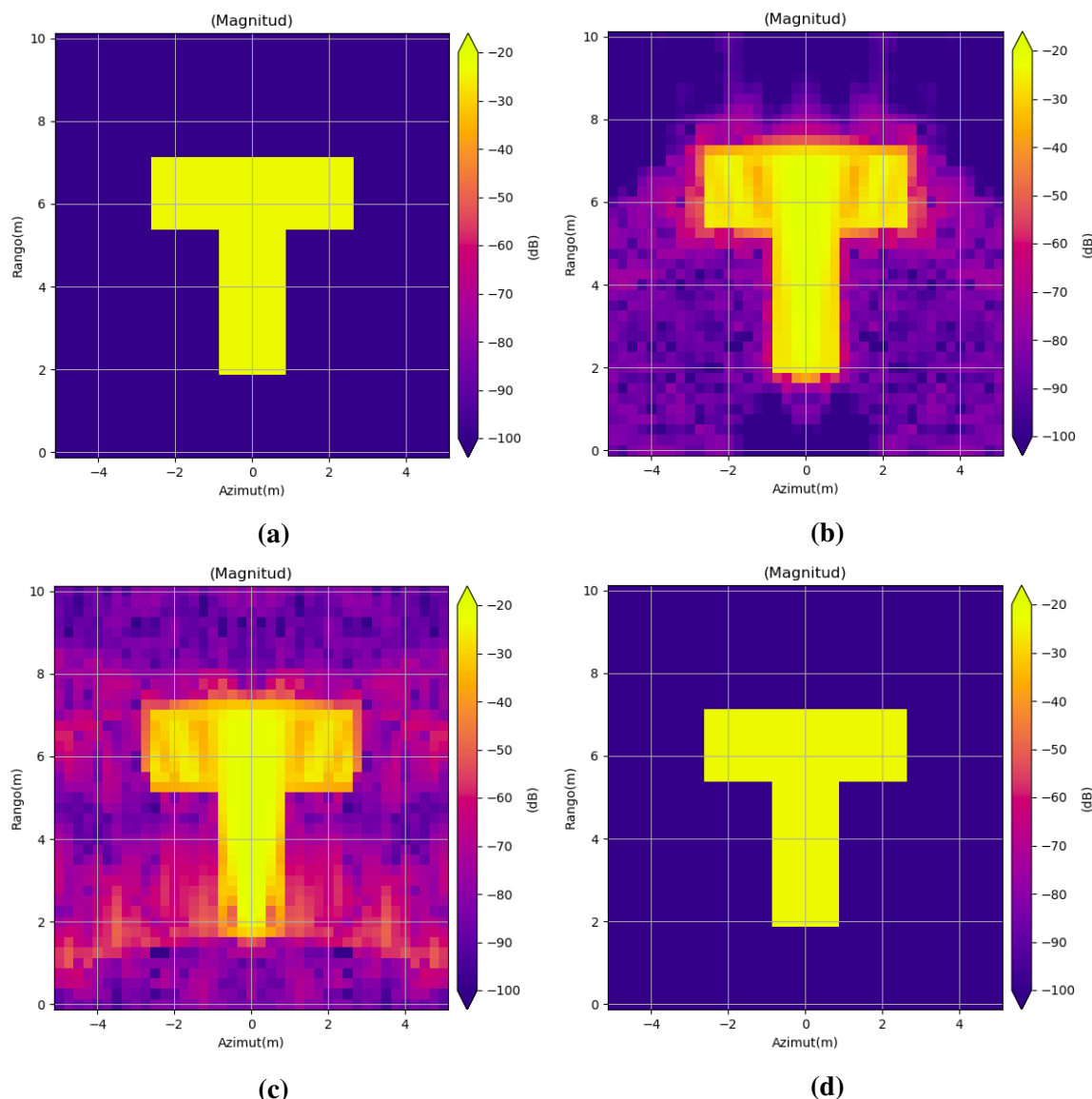


Figura 5.15: (a) Imagen teórica en magnitud de un blanco pseudo-continuo en forma de “T”. Imagen SAR en magnitud del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)

el mismo comportamiento de los lóbulos laterales que el primer caso, siendo notorios para los algoritmos FDBP y RMA, mientras que para el algoritmo DLIP sus lóbulos laterales no están visibles ya que se encuentran debajo de los -100 dB.

Respecto a los resultados de las imágenes en fase, estas se muestran en las figuras 5.16(b), (c) y (d) usando los algoritmos FDBP, RMA y DLIP respectivamente. Se puede notar que los algoritmos FDBP y DLIP tienen fase constantes y cercanas a cero en todo el blanco, mientras que el algoritmo RMA solo tiene fases constantes en partes del blanco, principalmente alrededor de $azimut = 0$ m. Finalmente se calculan los errores tanto en magnitud como en fase de las imágenes SAR respecto a la imagen teórica. En la tabla 5.9 se muestran los resultados de los distintos tipos de errores en magnitud y fase. El

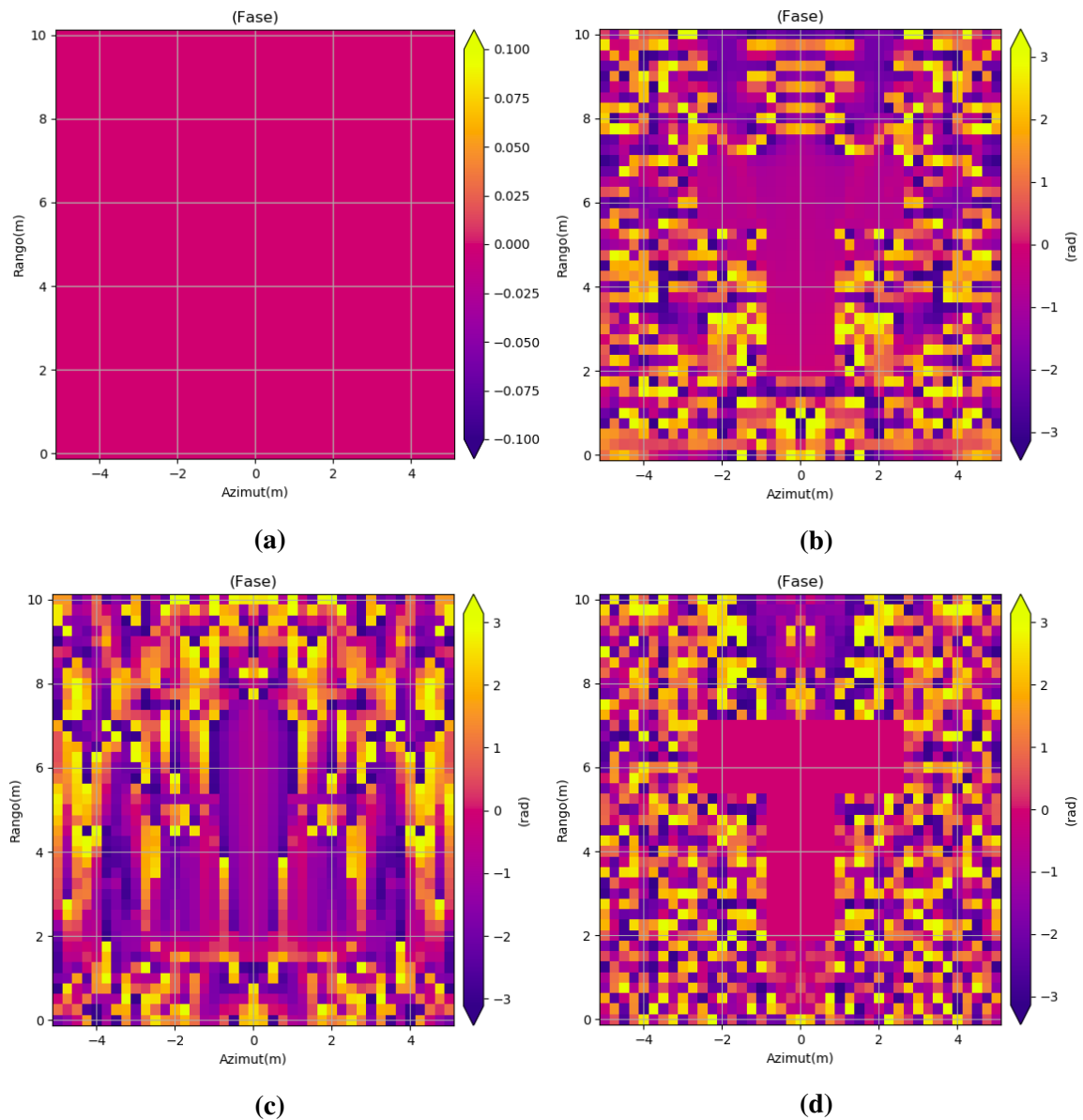


Figura 5.16: (a) Imagen teórica en fase de un blanco pseudo-continuo en forma de “T”. Imagen SAR en fase del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)

Tabla 5.9: Errores de magnitud y fase de las imágenes SAR de un blanco pseudo-continuo en forma de “T”, usando 3 algoritmos: FDBP, RMA, DLIP.

Error		Algoritmo		
		FDBP	RMA	DLIP
<i>Magnitud</i>	MSE	9.26e-05	16.08e-05	2.55e-28
	RMSE	9.62e-03	12.7e-03	1.596e-14
	ME	4.84e-02	6.05e-02	1.553e-13
	MAE	3.5e-03	4.94e-03	5.68e-15
<i>Fase</i>	RMSE	0.757	1.335	2.047e-14

algoritmo DLIP sigue teniendo los valores más bajos en el cálculo de los errores tanto de magnitud como de fase, seguido del algoritmo FDBP y finalmente por el algoritmo RMA.

Respecto al alto valor del error de fase con el algoritmo RMA, error de 1.335 rad, una explicación es que el tamaño de la apertura sintética L_s esta limitando la reconstrucción correcta de la fase a su longitud, tal como se explicó en la sección 5.1.2. Como el valor de $L_s = 1.2\text{ m}$, solo se reconstruye bien la fase en el intervalo $[-0.6; 0.6]\text{ m}$ en Azimut, lo cual coincide con su gráfica de fase.

3. TERCER CASO: Blanco pseudo-continuo en forma de la palabra “UNI”

La tercera prueba consistió de un segundo blanco pseudo-continuo en forma de la palabra “UNI”. Para la definición del blanco teórico se utilizó los mismos parámetros de la segunda prueba. En la figura 5.17(a) se muestra la magnitud y en la figura 5.18(a) la fase de la imagen teórica.

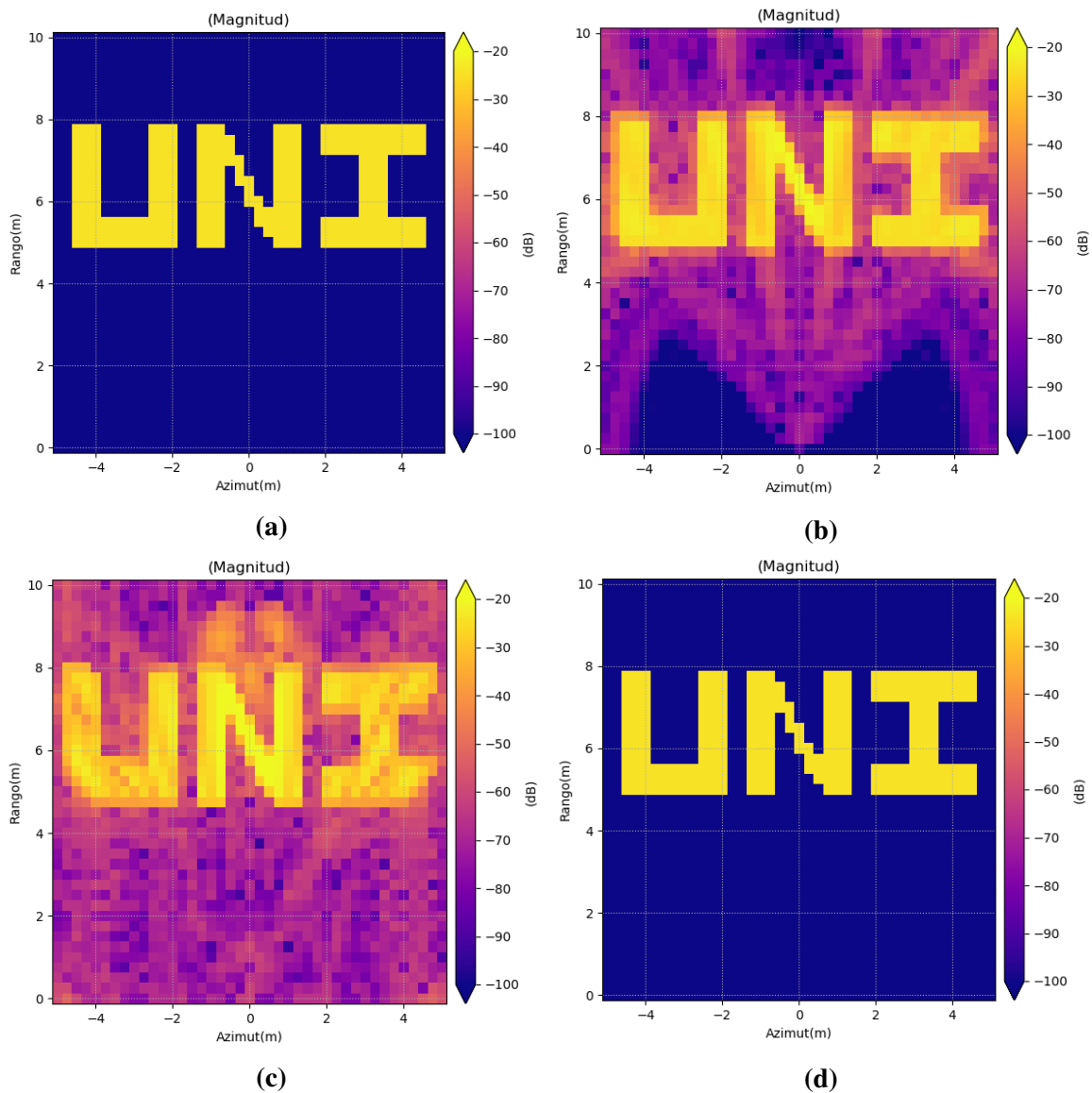


Figura 5.17: (a) Imagen teórica en magnitud de un blanco pseudo-continuo de la palabra “UNI”. Imagen SAR en magnitud del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)

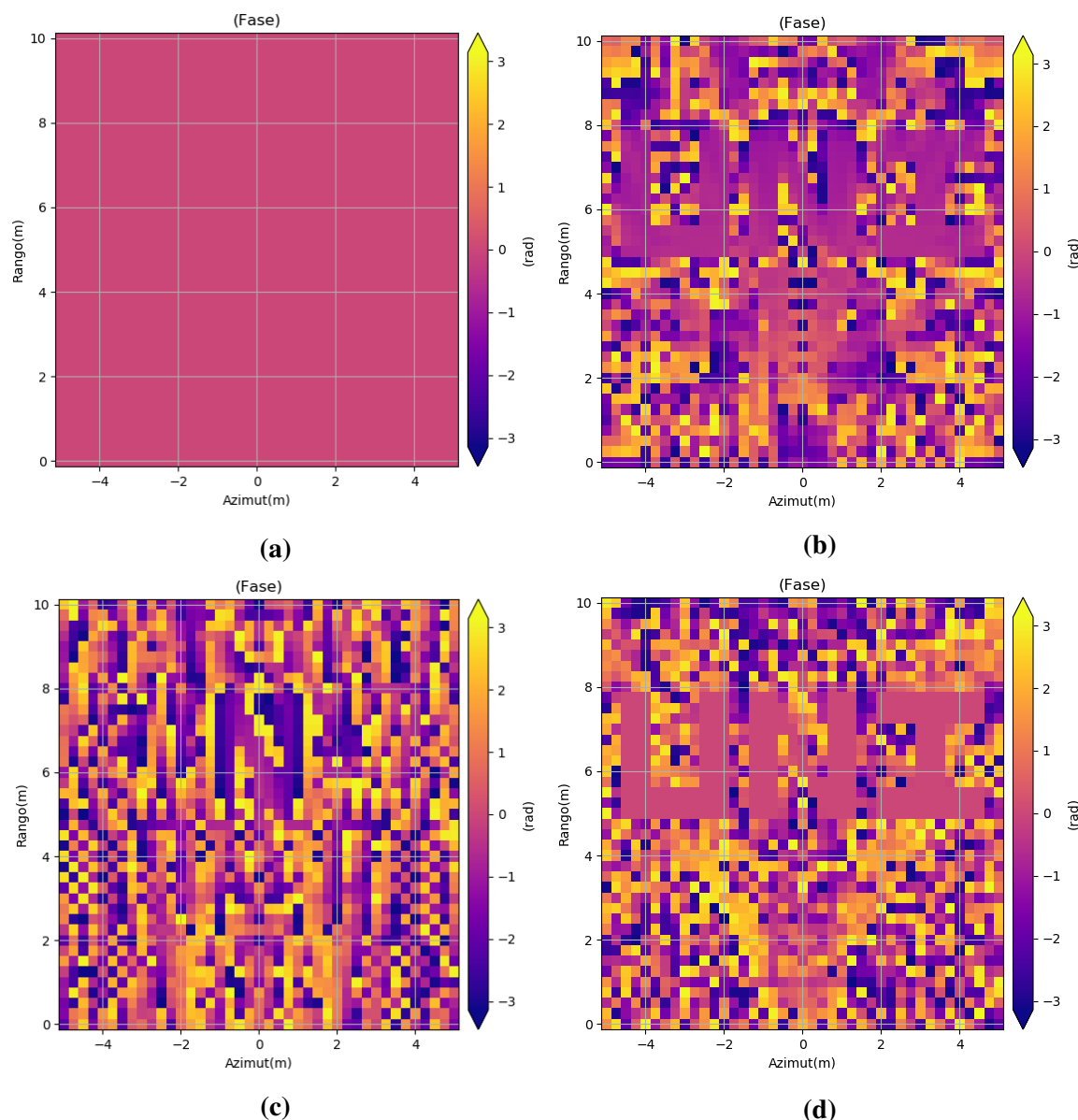


Figura 5.18: (a) Imagen teórica en fase de un blanco pseudo-continuo de la palabra “UNI”. Imagen SAR en fase del blanco, empleando el algoritmo: (b) FDBP, (c) RMA y (d) DLIP. (Fuente propia)

Luego se obtuvieron las imágenes SAR empleando los 3 algoritmos de formación de imágenes. En las figuras 5.17(b), (c) y (d) se muestran las imágenes reconstruidas en magnitud usando los algoritmos FDBP, RMA y DLIP respectivamente. Se puede observar que el blanco fue reconstruido en su posición correcta usando los 3 algoritmos. Además se nota la presencia de lóbulos laterales con los algoritmos FDBP y RMA, en el caso del algoritmo DLIP no se observan lo cual significa que están a menos de -100 dB. Finalmente se observan algunos artefactos en el centro de la parte superior de la imagen SAR con el algoritmo RMA.

Por otro lado, en las figuras 5.18(b), (c) y (d) se muestran las imágenes SAR en fase

usando los algoritmos FDBP, RMA y DLIP respectivamente. Se puede observar que el blanco posee una fase constante con los algoritmos FDBP y DLIP, sin embargo con el algoritmo RMA solo una parte del blanco, la letra ‘N’, posee una fase constante.

Finalmente se calcularon los errores de las imágenes SAR tanto en magnitud como en fase, los cuales se muestran en la tabla 5.10. Respecto a los errores en magnitud, se puede notar que el algoritmo DLIP sigue teniendo los valores más bajos, en el orden de 10^{-14} . Le sigue en orden el algoritmo FDBP, con errores en el orden de 10^{-3} , y finalmente le sigue el algoritmo RMA. Respecto a los errores en fase, el algoritmo DLIP tiene el menor error, en el orden de 10^{-14} , seguido por el algoritmo FDBP, con un valor de 1.27 rad, el cual incrementó de valor respecto al segundo caso, y por último se ubica el algoritmo RMA, con un valor de 1.77 rad, también con un incremento de valor.

Tabla 5.10: Errores de magnitud y fase de las imágenes SAR de un blanco pseudo-continuo de la palabra “UNI”, usando 3 algoritmos: FDBP, RMA, DLIP.

Error		Algoritmo		
		FDBP	RMA	DLIP
<i>Magnitud</i>	MSE	4.75e-05	15.21e-05	8.80e-29
	RMSE	6.89e-03	12.33e-03	9.38e-15
	ME	5.67e-02	9.94e-02	9.77e-14
	MAE	2.61e-03	5.30e-03	5.21e-15
<i>Fase</i>	RMSE	1.27	1.77	2.13e-14

De acuerdo a las 3 pruebas realizadas, el algoritmo DLIP es el que posee los menores errores de reconstrucción de imagen SAR, tanto en magnitud como en fase, con errores máximos en el orden de 10^{-14} para los errores de magnitud y de fase. Le sigue en orden ascendente (mayor error) el algoritmo FDBP y finalmente el algoritmo RMA, ambos algoritmos presentan errores máximos en el orden de 10^{-1} para los errores de magnitud, y de 1.27 rad y 1.77 rad para el error de fase con los algoritmos FDBP y RMA respectivamente.

5.2.2 Calidad de enfoque

Este parámetro evalúa la magnitud de los lóbulos laterales, los cuales repercuten en la reconstrucción de blancos cercanos. Para ello se empleó el factor PSLR, el cual es la relación entre la amplitud del mayor lóbulo lateral y la amplitud del lóbulo principal. Mientras menor sea el valor de PSLR, menor repercusión tendrán los lóbulos laterales de un blanco en la reconstrucción de otro blanco cercano.

El factor PSLR se determinó a partir de los perfiles de rango y azimut. En este caso, el perfil de rango se determinó obteniendo en primer lugar la imagen SAR y luego graficando todos los valores del eje del Rango para un valor de Azimut constante. De manera similar se determinó el perfil de azimut, graficando todos los valores del eje de Azimut para un valor de

Tabla 5.11: Posiciones de los blancos usados en las pruebas de PSLR.

Blanco	Posición: (Azimut, Rango)
B1	(0, 5)m
B2	(0, 2)m
B3	(0, 7)m
B4	(0, 9)m

Rango constante. Teniendo el perfil de rango o azimut con el eje de magnitud en unidades de dB, el PSLR se calculó restando la magnitud del mayor lóbulo secundario con la magnitud del lóbulo principal.

Se realizaron dos pruebas. La primera prueba consistió en calcular el PSLR a lo largo del eje de Rango, y la segunda prueba a lo largo del eje de Azimut. En ambas pruebas se emplearon 4 blancos puntuales por separado cuyas posiciones se muestran en la tabla 5.11.

En la primera prueba se empezó obteniendo las imágenes SAR de los 4 blancos empleando los algoritmos FDBP, RMA y DLIP. Para ello se usaron los parámetros de las Tablas 5.6 y 5.7, además se usó la ventana Hamming en los algoritmos FDBP y RMA. Luego se compu-

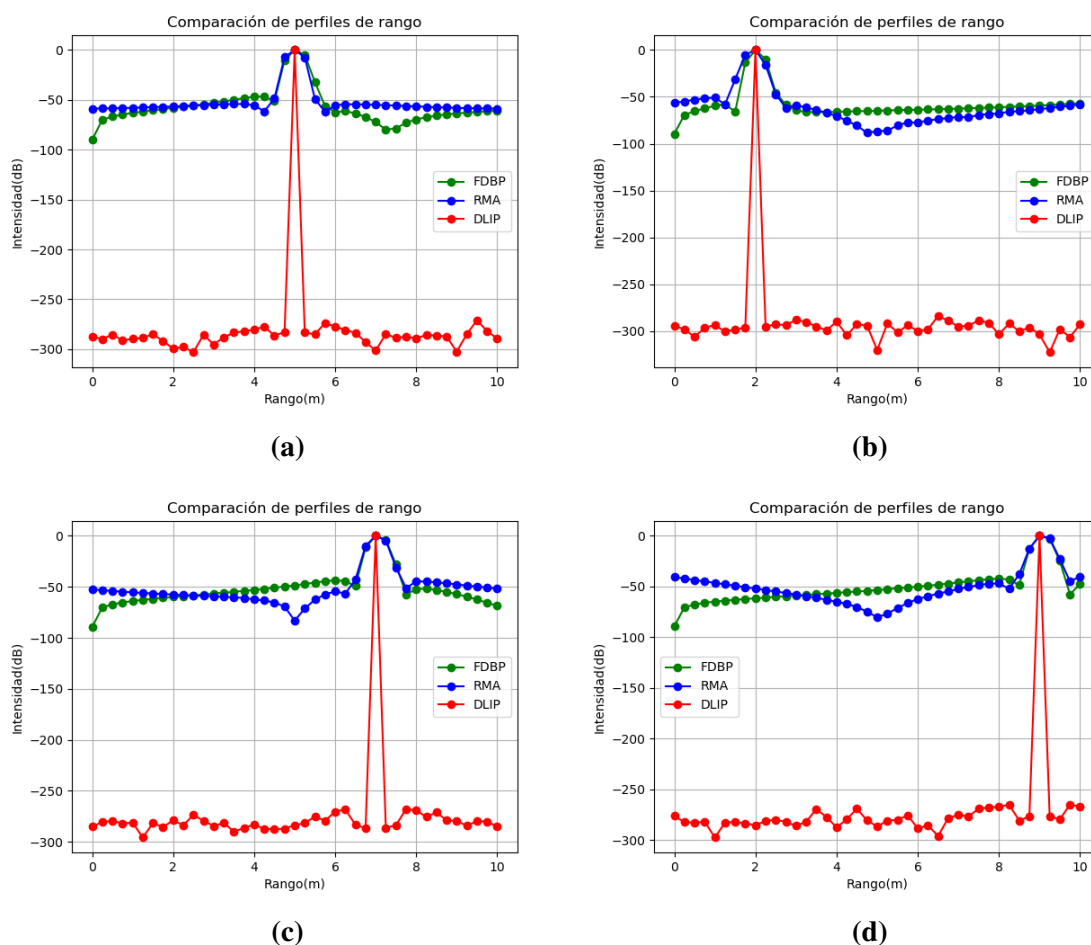


Figura 5.19: Perfiles de Rango obtenidos con 3 algoritmos distintos de un blanco puntual ubicado en la posición (Azimut, Rango): (a)(0, 5)m, (b)(0, 2)m, (c)(0, 7)m, (d)(0, 9)m. (Fuente propia)

taron los perfiles de rango para un valor de $Azimuth = 0m$, mostrándose los resultados en la figura 5.19. En esa figura se puede observar que el algoritmo DLIP es el que presenta los lóbulos laterales más atenuados, cerca a los -300 dB en todos los casos. En el siguiente paso de la primera prueba, se calculó el valor de PSLR para cada caso. Debido a que las imágenes SAR tuvieron pocos píxeles y por consecuencia los perfiles de rango también tuvieron pocas muestras, no se pudo identificar fácilmente el valor máximo del lóbulo principal ni de los lóbulos laterales para calcular el factor PSLR. Por lo tanto para obtener el valor de PSLR, se consideró el valor máximo del lóbulo principal y del mayor lóbulo lateral como el primer y segundo valor máximo de las muestras del perfil de rango respectivamente. Con esta consideración se obtuvieron los resultados de PSLR, los cuales se muestran en la tabla 5.12. De esta tabla se puede notar que el algoritmo DLIP presenta los valores más bajos de PSLR en los 4 casos, con un valor promedio de -272 dB. Le sigue el algoritmo FDBP con un valor promedio de -5.4 dB y finalmente el algoritmo RMA con un valor promedio de -4.8 dB.

Tabla 5.12: Valores de PSLR (en dB) a lo largo del eje de Rango para 4 blancos puntuales empleando los algoritmos FDBP, RMA y DLIP.

Blanco	FDBP	RMA	DLIP
B1	-4.91	-7.03	-270.98
B2	-10.05	-5.13	-283.82
B3	-3.79	-4.62	-267.74
B4	-2.80	-2.36	-265.10

En la segunda prueba se emplea un procedimiento similar. En primer lugar se obtienen las imágenes SAR para los 4 blancos empleando los 3 algoritmos, considerando las ventanas de Hamming para los algoritmos FDBP y RMA. Luego se obtienen los perfiles de azimuth, en las figuras 5.20(a),(b),(c) y (d) se muestran los perfiles en azimuth de los blancos B1, B2, B3 y B4, correspondientes a la posiciones en Rango: 5 m, 2 m, 7 m y 9 m respectivamente. A continuación se genera un cuadro comparativo de PSLR el cual es mostrado en la tabla 5.13. Como se puede observar en ella el comportamiento es similar a la prueba anterior, con el algoritmo DLIP presentando los valores más bajos de PSLR, con un valor promedio de -286 dB, seguido del algoritmo FDBP con un valor promedio de -48.7 dB y finalmente el algoritmo RMA con un valor promedio de -34.7 dB.

Finalmente se resume los resultados señalando que el algoritmo DLIP presentó los valores más bajos de PSLR tanto en la dirección de Rango como de Azimut, seguido del algoritmo FDBP y por último el algoritmo RMA. Estos dos últimos algoritmos tuvieron valores cercanos de PSLR.

El hecho que el algoritmo DLIP tenga los valores más bajos de PSLR significa que la

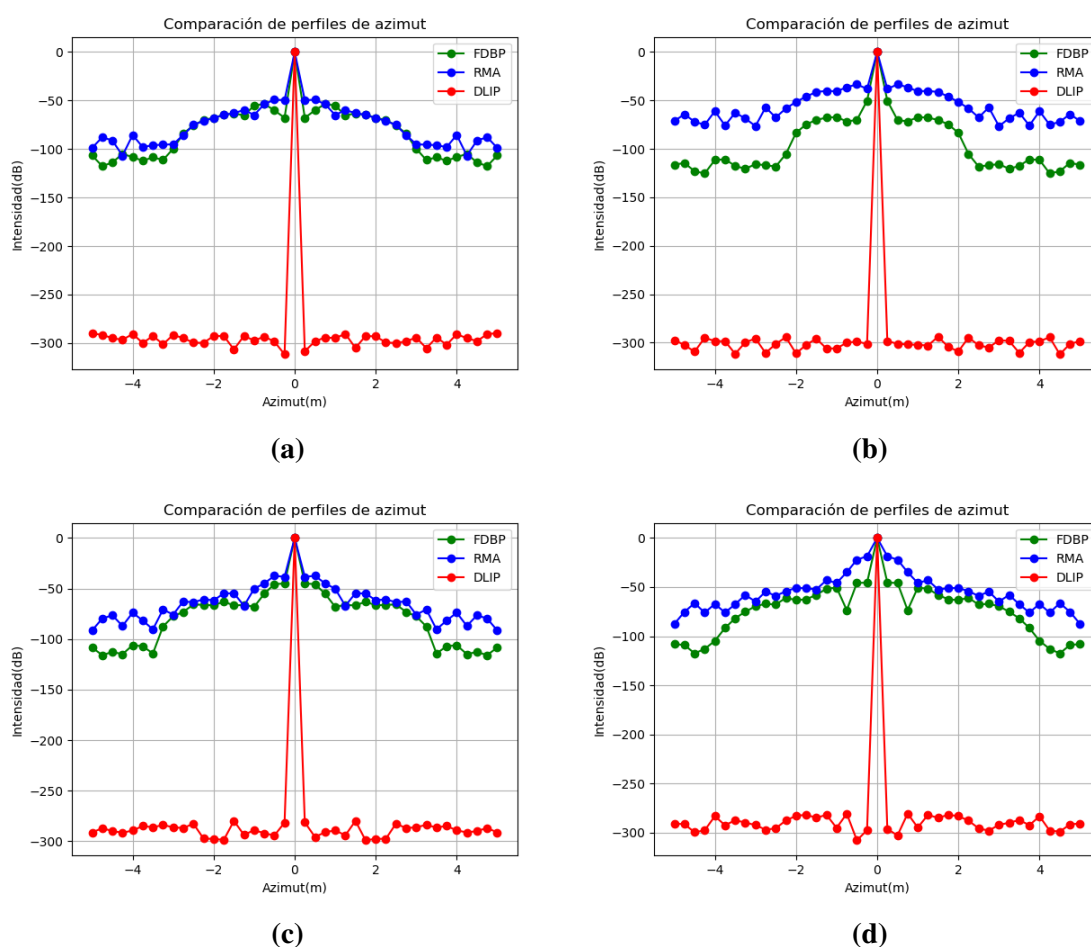


Figura 5.20: Perfiles de Azimut obtenidos con 3 algoritmos distintos de un blanco puntual ubicado en la posición (Azimut, Rango): (a) $(0, 5)m$, (b) $(0, 2)m$, (c) $(0, 7)m$, (d) $(0, 9)m$. (Fuente propia)

Tabla 5.13: Valores de PSLR (en dB) a lo largo del eje de Azimut para 4 blancos puntuales empleando los algoritmos FDBP, RMA y DLIP.

Blanco	FDBP	RMA	DLIP
B1	-53.61	-49.19	-289.66
B2	-50.44	-33.47	-293.94
B3	-45.10	-37.46	-279.87
B4	-45.51	-18.64	-280.69

reconstrucción de un blanco interfiere menos en la reconstrucción de otros blancos cercanos y por lo tanto la imagen SAR resultante podría presentar menores errores de reconstrucción, efectivamente esta hipótesis es cierta ya que los resultados de calidad de reconstrucción de imagen indicaron que el algoritmo DLIP presenta los menores errores, tanto en magnitud como en fase.

5.2.3 Coste computacional

Como se mencionó en la sección 2.4.4, el coste computacional es la cantidad de recursos que requiere un algoritmo para ejecutarlo, ya sea tiempo o memoria. Depende principalmente de

la complejidad de las ecuaciones matemáticas que rigen cada algoritmo.

Para comparar los algoritmos bajo este parámetro, primero se calculó la complejidad computacional de cada algoritmo y luego el tiempo de procesamiento.

Complejidad computacional

La complejidad computacional es el coste computacional teórico. Se expresa en términos de la notación ‘Big-O’. Se obtiene a partir de los pasos principales de los algoritmos. A continuación se muestra el procedimiento de obtención de este parámetro para los algoritmos FDBP, RMA y DLIP.

1. Algoritmo FDBP

A partir de los pasos de este algoritmo se determinó su complejidad computacional aproximada. Se asumió que el histórico de fase tiene N_f frecuencias y N_p posiciones del riel, que el tamaño de la imagen resultante es de $N_x \times N_y$ píxeles, que se usa una interpolación lineal y que no se emplea ninguna función de enventanado. Luego se planteó el costo computacional de cada paso del algoritmo, el cual se muestra en la tabla 5.14.

Tabla 5.14: Complejidad computacional aproximada del algoritmo FDBP

PASOS	SUB PASOS	COSTE
Compresión en rango	-	$N_p N_f \log N_f$
Interpolación y suma coherente	Vector de distancias	$N_p N_x N_y$
	Factor de fase	$N_p N_x N_y$
	Interpolación 1D	$N_f N_p N_x N_y$
	Suma coherente	$N_p N_x N_y$

Sumando los costes computacionales de todos los pasos:

$$= N_p N_f \log N_f + N_p N_x N_y + N_p N_x N_y + N_f N_p N_x N_y + N_p N_x N_y \quad (5.2)$$

$$= N_p N_f \log N_f + 3N_p N_x N_y + N_f N_p N_x N_y \quad (5.3)$$

Asumiendo:

$$N_f = N_p = N_x = N_y = N \quad (5.4)$$

La expresión quedó del siguiente modo:

$$= N^2 \log N + 3N^3 + N^4 \quad (5.5)$$

Finalmente se escogió la mayor expresión matemática para calcular la complejidad:

$$\mathbf{O(N^4)} \quad (5.6)$$

2. Algoritmo RMA

A partir de los pasos de este algoritmo se determinó su complejidad computacional aproximada. Se asumió que el histórico de fase tiene N_f frecuencias y N_p posiciones del riel, que el tamaño de la imagen resultante es de $N_x \times N_y$, que se aplican las decimaciones D_x

y D_y , que se usa una interpolación lineal y que no se usa ninguna función de enventanado. Luego se planteó el costo computacional de cada paso del algoritmo, el cual se muestra en la tabla 5.15.

Tabla 5.15: Complejidad computacional aproximada del algoritmo RMA

PASOS	SUB PASOS	COSTE
1D-DFT (Azimut)	-	$D_x N_x N_f \log D_x N_x$
Matched Filter	-	$D_x N_x N_f$
Interpolación STOLT	Cambio de coordenadas	$D_x N_x N_f$
	Interpolación 1D	$D_x D_y N_x N_y N_f$
	Decimación en x	$D_x D_y N_x N_y$
	Decimación en y	$D_y N_x N_y$
2D-IDFT	-	$N_x N_y \log N_x N_y$

Sumando los costes computacionales de todos los pasos:

$$= D_x N_x N_f \log D_x N_x + D_x N_x N_f + D_x N_x N_f + D_x D_y N_x N_y N_f + D_x D_y N_x N_y + D_y N_x N_y + N_x N_y \log (N_x N_y) \quad (5.7)$$

$$= D_x N_x N_f \log D_x N_x + 2D_x N_x N_f + D_x D_y N_x N_y N_f + D_x D_y N_x N_y + D_y N_x N_y + N_x N_y \log (N_x N_y) \quad (5.8)$$

Asumiendo:

$$N_f = N_x = N_y = N \quad (5.9)$$

$$D_x = D_y = D \quad (5.10)$$

La expresión quedó del siguiente modo:

$$= DN^2 \log(DN) + 3DN^2 + D^2 N^3 + D^2 N^2 + 2N^2 \log(N) + D^2 N^3 \quad (5.11)$$

Finalmente se escogió la mayor expresión matemática para obtener la complejidad:

$$\mathbf{O(D^2 N^3)} \quad (5.12)$$

3. Algoritmo DLIP

A partir de los pasos de este algoritmo se calculó su complejidad computacional aproximada. Se asumió que el histórico de fase tiene N_f frecuencias y N_p posiciones del riel, que el tamaño de la imagen resultante es de $N_x \times N_y$ píxeles y que se considera un caso con rango completo y subdeterminado para calcular el valor de A^\dagger . Luego se planteó el costo computacional de cada paso del algoritmo, el cual se muestra en la tabla 5.16.

Sumando los costes computacionales de todos los pasos:

$$= N_p^2 N_f^2 N_x N_y + N_p^3 N_f^3 + N_x N_y N_p^2 N_f^2 + N_p N_f N_x N_y \quad (5.13)$$

$$= N_p^3 N_f^3 + 2N_x N_y N_p^2 N_f^2 + N_p N_f N_x N_y \quad (5.14)$$

Tabla 5.16: Complejidad computacional aproximada del algoritmo DLIP

PASOS	SUB PASOS	COSTE
Cálculo de A^\dagger	$m_1 = AA^T$	$N_p^2 N_f^2 N_x N_y$
	$m_2 = [m_1]^{-1}$	$N_p^3 N_f^3$
	$A^\dagger = A^T m_2$	$N_x N_y N_p^2 N_f^2$
Cálculo de I	-	$N_p N_f N_x N_y$

Asumiendo:

$$N_f = N_p = N_x = N_y = N \quad (5.15)$$

La expresión quedó del siguiente modo:

$$= N^6 + 2N^6 + N^4 \quad (5.16)$$

Finalmente se escogió la mayor expresión matemática para obtener la complejidad:

$$\mathbf{O}(N^6) \quad (5.17)$$

A modo de resumen, se muestran las complejidades computacionales de los 3 algoritmos en la tabla 5.17.

Tabla 5.17: Complejidad computacional de los algoritmos implementados

	FDBP	RMA	DLIP
Complejidad computacional	$O(N^4)$	$O(D^2 N^3)$	$O(N^6)$

Como se puede observar en ella, el algoritmo DLIP presenta el valor más alto de complejidad computacional, casi N^2 veces más que el algoritmo FDBP. Los otros 2 algoritmos presentan valores menores, el menor de ellos va depender de cada caso en específico, por ejemplo para el siguiente caso.

Sea:

$$N = 10 \quad \wedge \quad D = 10$$

Se calcularon los valores de complejidad para ambos algoritmos obteniéndose:

- **FDBP:** *complejidad* $\propto 10^4$
- **RMA:** *complejidad* $\propto 10^5$

En este caso el algoritmo FDBP presenta menor coste que el algoritmo RMA, pero para un valor menor de la variable D el algoritmo RMA podría ser menor. En la siguiente sección se muestran resultados experimentales para validar estos resultados teóricos.

Tiempo de procesamiento

El tiempo de procesamiento es el coste computacional real y permite validar los resultados de la complejidad computacional. Por lo tanto, se obtuvo este parámetro para 3 casos distintos: un blanco puntual y 2 blancos pseudo-continuos. Estos blancos son los mismos que fueron usados en las pruebas de comparación bajo el parámetro de calidad de reconstrucción de la

Tabla 5.18: Tiempo de procesamiento para 3 casos de blancos simulados: 1 blanco puntual y 2 blancos pseudo-continuos, calculado para los algoritmos FDBP, RMA y DLIP.

BLANCO	FDBP	RMA	DLIP
Blanco puntual	0.3317 s	0.3333 s	102.63 s
Blanco pseudo-continuo en forma de ‘T’	0.106 s	0.149 s	113.36 s
Blanco pseudo-continuo en forma de ‘UNI’	0.138 s	0.166 s	117.79 s

imagen, ver sección 5.2.1.

Se calculó el tiempo que demora en ejecutarse el programa de cada algoritmo en un computador con las siguientes características: 8 GB de memoria RAM y procesador Intel Core i5-4590 @ 3.3 GHz x 4. En la tabla 5.18 se muestran los resultados.

De acuerdo a esta tabla, el algoritmo FDBP es el que presenta el menor tiempo de procesamiento (resaltado en color plomo en la tabla), seguido del algoritmo RMA, y por último el algoritmo DLIP. El tiempo que demora el algoritmo DLIP es aproximadamente 300 veces más que el obtenido con el algoritmo FDBP. Estos resultados están acorde a los resultados de la tabla 5.17.

Por lo tanto de los resultados de complejidad computacional y tiempo de procesamiento se pueden sacar dos conclusiones. La primera de ellas es que el algoritmo DLIP tiene el mayor coste computacional. Debido también a su alto consumo de recursos computacionales (ver sección 4.3.4), lo convierten en un algoritmo impráctico para procesar imágenes reales grandes. La segunda conclusión es que de acuerdo a las pruebas de tiempo de procesamiento realizadas el algoritmo FDBP es el que tiene el menor coste computacional.

5.3 Análisis de los resultados experimentales

En esta sección se muestran los resultados de la implementación de los algoritmos FDBP y RMA con datos reales. No se empleó el algoritmo DLIP para este análisis debido a su alto coste computacional con datos reales. Después de la implementación, se analizaron las imágenes SAR resultantes, comparando los tiempos de procesamiento de cada algoritmo. Finalmente se obtuvieron los mapas de desplazamientos a partir de las imágenes SAR con ambos algoritmos y luego se analizaron estos resultados.

5.3.1 Análisis de las imágenes SAR

En primer lugar se formaron las imágenes SAR empleando 2 algoritmos: FDBP y RMA. El histórico de fase fue obtenido con el radar GB-SAR ‘IGP-ROJ’. El blanco escogido fue el cerro Pucruhacra ubicado en el distrito de San Mateo, provincia de Huarochirí, región Lima. Las razones de su elección fueron los antecedentes de deslizamientos en este cerro y la cercanía de la ubicación del blanco a la ciudad de Lima para la instalación del radar

GB-SAR. En la figura 5.21 se muestra el blanco, el cual está encerrado con un polígono rojo, además se muestra la ubicación aproximada del radar GB-SAR. Además el blanco posee franjas concéntricas, tal como se observa en la figura 5.21, los cuales servirán como guía para identificar el blanco cuando las imágenes SAR hayan sido generadas. Se usaron los parámetros eléctricos, mecánicos y geométricos de la tabla 5.19. Durante la implementación de ambos algoritmos se usó una ventana de Hamming para la reducción de los lóbulos laterales, además en el algoritmo RMA se usó una interpolación polinomial cúbica en la etapa de Interpolación STOLT.

En la figura 5.22 se muestran las imágenes SAR en magnitud obtenidas con los algoritmos FDBP y RMA. Las zonas de alta reflectividad posiblemente corresponden a zonas de escasa vegetación (rocas, suelo árido) mientras que las zonas con baja reflectividad corresponden



Figura 5.21: Cerro Pucruchacra, vista aérea. Se muestra el blanco y la ubicación aproximada del radar GB-SAR. (Fuente: Google Maps)

Tabla 5.19: Parámetros eléctricos, mecánicos y geométricos usados en las pruebas con datos reales.

PARÁMETRO	VALOR
Frecuencia central f_c	15.55 GHz
Ancho de banda BW	100 MHz
Número de frecuencias N_f	1001
Apertura sintética L_s	1.4 m
Ángulo de visión θ	75°
Número de posiciones N_p	178
Ancho de la imagen L_x	400 m
Alto de la imagen L_y	650 m
Posición inicial eje de Rango y_i	300 m
Grilla de la imagen d_x, d_y	1 m

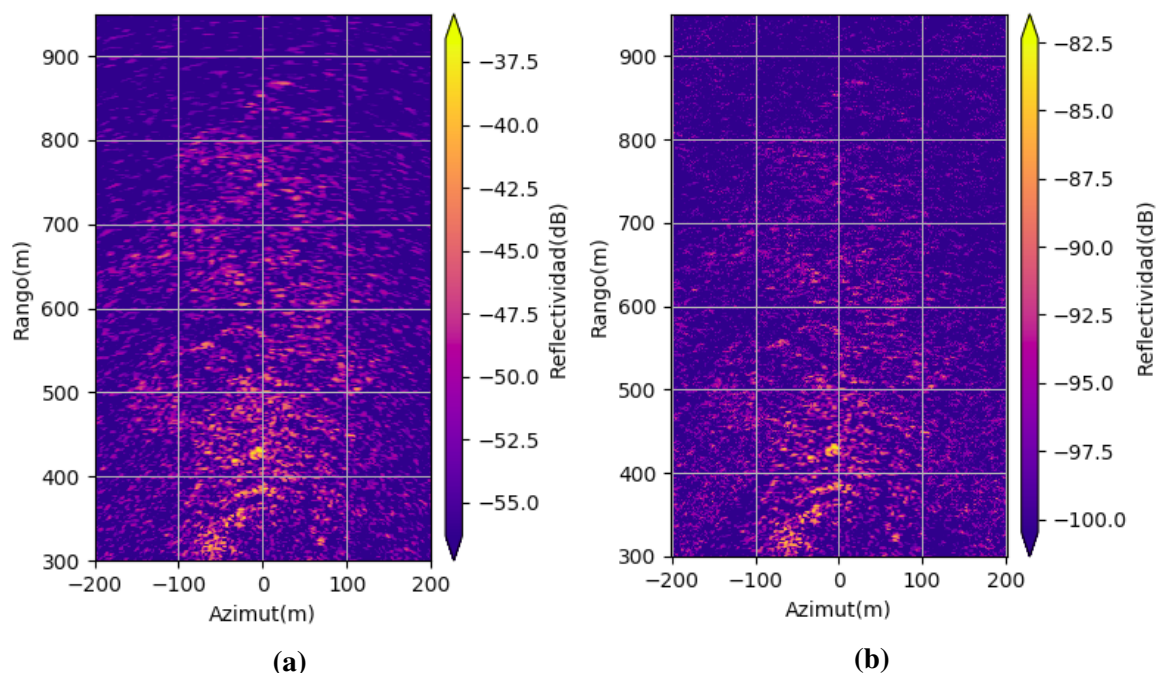


Figura 5.22: Imágenes SAR en magnitud de un blanco real, obtenido con los algoritmos: (a) FDBP y (b) RMA. (Fuente propia)

a zonas con vegetación alguna. Se puede observar en ambas imágenes que las zonas de mayor reflectividad están ubicados entre 300 m y 500 m en Rango, que es donde se ubican las franjas concéntricas del blanco aproximadamente. Observando varias imágenes SAR de este mismo blanco, se notó la presencia de las franjas, entre 300 m y 500 m en Rango, en las transiciones de color de azul a amarillo (anaranjado). Al obtener posteriormente los mapas de desplazamientos se observarán mejor estas franjas.

Respecto al tiempo de procesamiento, sus valores para cada algoritmo se muestran en la tabla 5.20. Como se puede ver en ella, el algoritmo FDBP presentó el menor tiempo, mientras que el algoritmo RMA demoró en ejecutarse unas 18 veces más que el algoritmo FDBP aproximadamente.

Tabla 5.20: Tiempo de procesamiento de los algoritmos FDBP y RMA para un blanco real.

Algoritmo	Tiempo (s)
FDBP	6.02
RMA	112.54

5.3.2 Análisis de los mapas de desplazamientos

En una aplicación de monitoreo de deslizamientos con radares SAR, las imágenes SAR obtenidas son posteriormente procesadas y convertidas en mapas de desplazamientos (deslizamientos). A partir de estos mapas se elaboran gráficas de deslizamientos en función al tiempo, en ciertas zonas o puntos. Estas gráficas junto a los mapas son los resultados finales del sistema de monitoreo. A continuación se evalúa el efecto de las imágenes SAR generadas

con cada algoritmo en la obtención de estos resultados finales.

Los mapas de desplazamientos, también llamados interferogramas, se obtienen usando la técnica GB-DInSAR. Por cada dos imágenes SAR sucesivas se obtiene un interferograma. Se obtuvieron un total de 51 imágenes SAR con cada algoritmo, con un periodo de adquisición de 9 minutos, $T = 9m$. Luego se procesaron las imágenes SAR y se obtuvieron un total de 50 interferogramas. En la figura 5.23 se muestran dos interferogramas obtenidos con las imágenes SAR número 10 y 11, cada interferograma correspondiente a los algoritmos FDBP y RMA. Las unidades de los mapas están dadas en milímetros, presentando un intervalo de $[-4.5; 4.5]$ mm. Las zonas negras no se pueden analizar debido a la falta de datos confiables, y se obtuvieron a partir de un mapa de coherencia con valor límite 0.7. Se puede observar que hay más zonas que se pueden analizar con el algoritmo FDBP que con el algoritmo RMA, por lo que se puede decir que el algoritmo FDBP tiene mayor zona de alta coherencia que el algoritmo RMA. Anteriormente se mencionó que las franjas concéntricas del blanco iban a ser más notorios en los interferogramas, y efectivamente observando cualquiera de las imágenes SAR en la figura 5.23 se pueden observar las franjas entre 300 m y 500 m en Rango.

Luego se obtuvieron los gráficos de deslizamientos, que es la evolución temporal de los desplazamientos en una cierta zona o puntos de prueba. Se escogieron 4 puntos de prueba

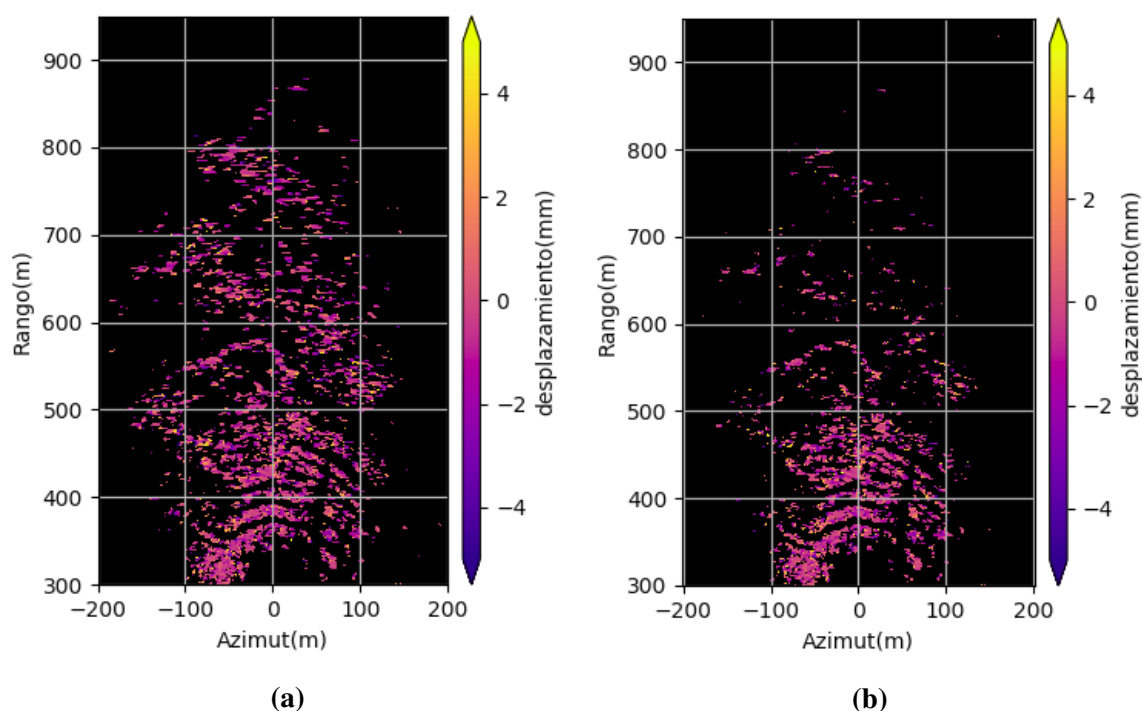


Figura 5.23: Mapa de desplazamientos (interferograma) del blanco real generados con dos imágenes SAR, a su vez obtenidas con los algoritmos: (a) FDBP y (b) RMA. (Fuente propia)

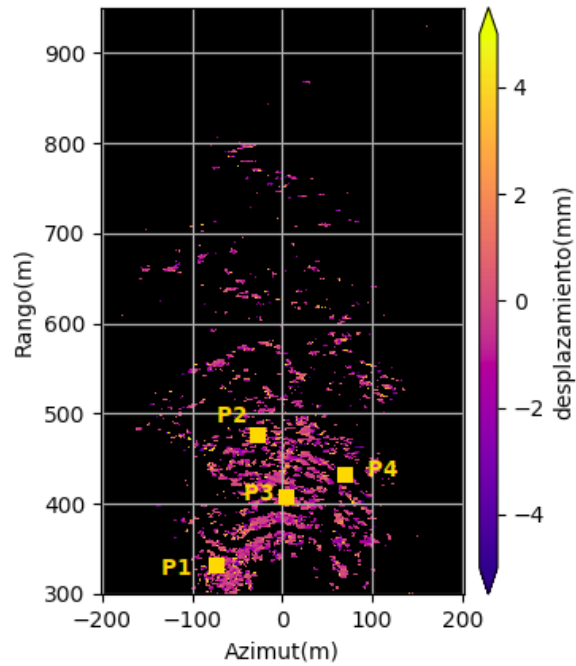


Figura 5.24: Ubicación de los 4 puntos de prueba en el mapa de desplazamientos. (Fuente propia)

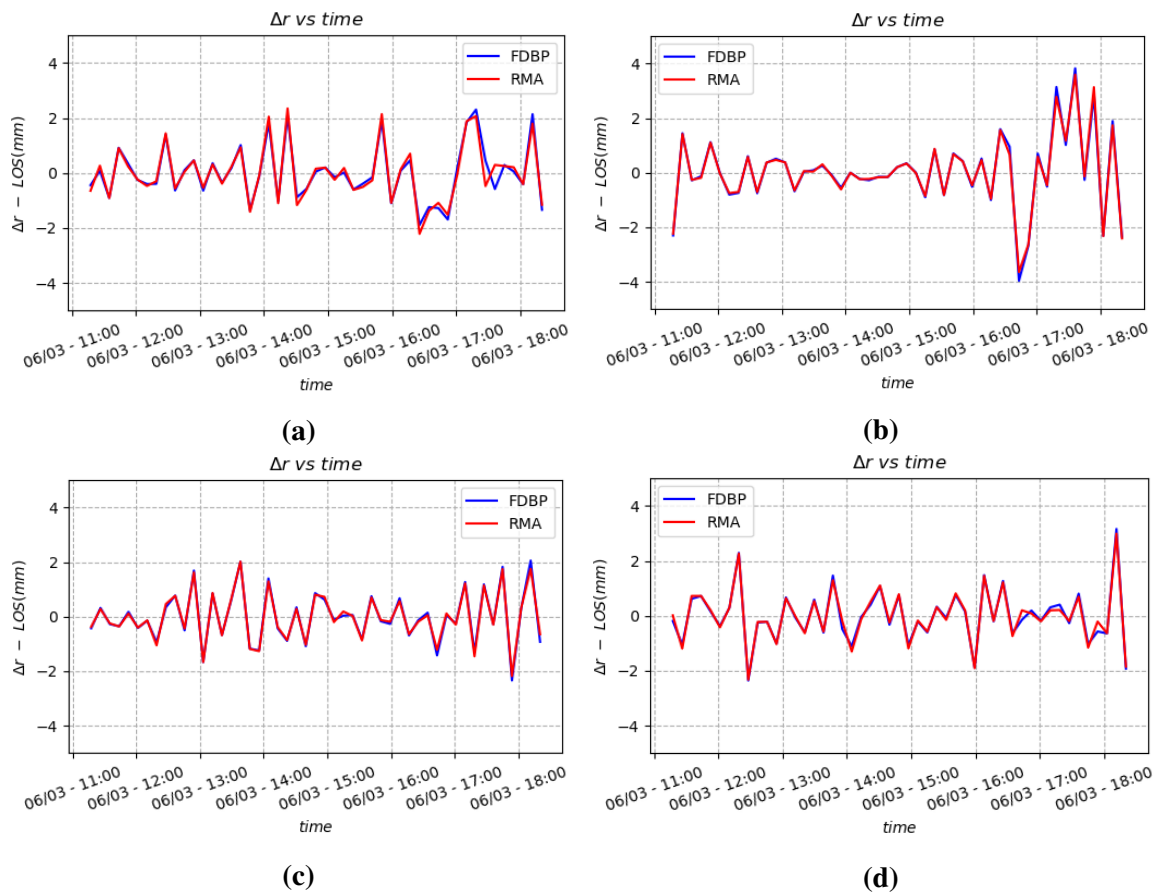


Figura 5.25: Gráficas de deslizamientos para 4 puntos de prueba: (a) P1, (b) P2, (c) P3 y (d) P4. (Fuente propia)

Tabla 5.21: Desplazamientos promedios de los 4 puntos de prueba en función a los algoritmos FDBP y RMA. Todos los valores se muestran en (mm).

Puntos de prueba	FDBP	RMA
P1	0.0009	-0.0017
P2	0.0301	0.0268
P3	-0.0096	-0.0115
P4	-0.0013	-0.00012

los cuales se muestran en la figura 5.24. Por cada punto de prueba se obtuvo su gráfica de deslizamientos. Cada muestra de esta gráfica corresponde al desplazamiento del punto de prueba en el interferograma. Se usaron los 50 interferogramas, los cuales corresponden a un intervalo de tiempo de 7.5 horas aproximadamente. En la figura 5.25 se muestran las 4 gráficas de deslizamientos, uno por cada punto de prueba y dentro de cada gráfica una curva por cada algoritmo.

A continuación se procede al análisis de estas gráficas. En primer lugar se observa un comportamiento similar de las curvas en los 4 puntos de prueba, con ligeras diferencias en algunas zonas. Luego se procedió a calcular los valores promedios de las curvas, mostrándose los resultados en la tabla 5.21. Los valores promedios son similares en ambos casos para todos los puntos de prueba, con una diferencia máxima de 0.0033 mm para el punto P2.

Resumiendo los resultados obtenidos, de las pruebas con datos reales el algoritmo FDBP presentó las siguientes ventajas sobre el algoritmo RMA: menor tiempo de procesamiento y mayor área de alta coherencia en los interferogramas. Respecto a las gráficas de deslizamientos, ambos algoritmos presentaron resultados similares, tanto en valores instantáneos como en valores promedios.

5.4 Resumen

Tres algoritmos de formación de imágenes SAR: FDBP, RMA y DLIP, fueron probados con datos simulados y luego comparados en términos de: calidad de reconstrucción de la imagen, calidad de enfoque y coste computacional. Luego se probaron con datos reales y compararon en términos de: coste computacional, medición de desplazamientos y las zonas que presentan alta coherencia en los interferogramas.

En la tabla 5.22 se muestra un cuadro comparativo resumido, teniendo en cuenta los criterios de comparación mencionados.

Tabla 5.22: Cuadro comparativo de los tres algoritmos implementados: FDBP, RMA y DLIP.

Parámetro	FDBP	RMA	DLIP
Calidad de reconstrucción de la imagen	medio	bajo	alto
Calidad de enfoque	medio	bajo	alto
Coste computacional	bajo	medio	alto
Zonas de alta coherencia en los interferogramas	mayor área	menor área	-
Medición de los desplazamientos	similar	similar	-

CONCLUSIONES

1. Se implementaron tres algoritmos de formación de imágenes SAR, Frequency Domain Back Projection (FDBP), Range Migration Algorithm (RMA) y Discrete Linear Inverse Problem (DLIP), y luego se compararon bajo tres parámetros: calidad de reconstrucción de imagen, calidad de enfoque y coste computacional, para una aplicación de monitoreo de deslizamientos superficiales de tierra.
2. Luego de la comparación de los tres algoritmos de formación de imágenes SAR bajo el parámetro de calidad de reconstrucción de imagen empleando tres blancos diferentes y datos obtenidos por simulación, se concluyó que el algoritmo DLIP es el que presenta los menores errores de reconstrucción de imagen, con un error RMSE promedio en magnitud de $10.42e-15$ y un error RMSE promedio en fase de $1.42e-14$ rad.
3. Después de comparar los tres algoritmos de formación de imágenes SAR bajo el parámetro de calidad de enfoque empleando cuatro blancos puntuales y datos obtenidos por simulación, se concluyó que el algoritmo DLIP es el que presentó los valores más bajos de PSLR, con un valor promedio de -272 dB a lo largo del eje del rango y un valor promedio de -286 dB a lo largo del eje de azimut.
4. Luego de la comparación de los tres algoritmos de formación de imágenes SAR bajo el parámetro de coste computacional empleando tres blancos diferentes y datos obtenidos por simulación y datos reales, se concluyó que el algoritmo FDBP presenta el menor tiempo de procesamiento en cada caso, con un valor promedio de 0.19 s.
5. El algoritmo DLIP presentó los valores más altos de coste computacional y consumo de recursos computacionales, con un tiempo de procesamiento promedio de 111.3 s, motivo por el cual las pruebas con este algoritmo se realizaron solo con datos obtenidos por simulación de tamaño reducido.
6. Las pruebas con datos reales se realizaron empleando solo los algoritmos FDBP y RMA, y producto de la comparación entre ambos se determinó que el algoritmo FDBP presenta una mayor área de alta coherencia en los interferogramas generados, y que ambos algoritmos presentan valores instantáneos y promedios similares en las gráficas de deslizamientos.
7. De todas las pruebas realizadas tanto con datos obtenidos por simulación como con da-

tos reales, se determinó que el algoritmo FDBP es el más apropiado para una aplicación de monitoreo de deslizamientos con radares GB-SAR. Las razones son las siguientes: tiene el coste computacional más bajo lo cual permite obtener rápidamente las imágenes SAR, en los interferogramas genera más zonas de alta coherencia permitiendo analizar los deslizamientos en más regiones del mapa de desplazamientos, presenta valores intermedios en los errores de reconstrucción de la imagen SAR, principalmente en la fase, y finalmente tiene valores intermedios de PSLR. Con esta conclusión se cumple el objetivo general de la tesis.

8. Se desarrolló un método para generar el histórico de fase teóricamente, lo cual permitió obtener los datos simulados para hacer las distintas pruebas.
9. Se implementó exitosamente el algoritmo RMA para el caso en que el tamaño de la imagen SAR en Azimut es mucho mayor a la apertura sintética.
10. En las pruebas del algoritmo RMA con datos obtenidos por simulación se observó que el valor de la apertura sintética L_s influye en la correcta obtención de las imágenes SAR, tanto en fase como en magnitud, por ello se determinó las zonas de la imagen SAR de correcta reconstrucción en función a los valores de L_s .

RECOMENDACIONES

1. Se recomienda usar funciones de enventanado, tales como Hanning o Hamming, durante la implementación de los algoritmos FDBP y RMA, debido a que reducen los lóbulos laterales de la imagen SAR, generando una imagen más notoria, más que todo en magnitud.
2. Debido al alto coste computacional del algoritmo DLIP, además del alto consumo de recursos computacionales como memoria RAM, se recomienda realizar las pruebas solo con datos generados por simulación y con imágenes pequeñas en un computador estándar.
3. Para evitar la presencia de *artifacts* en las imágenes SAR reconstruidas, se recomienda ubicar el blanco al centro de la escena de la imagen y dentro de la zona de correcta reconstrucción en el caso del algoritmo RMA.
4. En las imágenes SAR formadas con datos reales posiblemente no se visualice notoriamente el blanco, mas que todo en magnitud, por lo que se recomienda variar el rango dinámico de la imagen y/o cambiar el tamaño de la escena de la imagen enfocando las zonas donde se ubica el blanco.
5. Se recomienda optimizar el algoritmo DLIP con el objetivo de bajar el tiempo de procesamiento, para luego realizar pruebas con datos reales y posteriormente compararlos con los algoritmos FDBP y RMA.

BIBLIOGRAFÍA

- [1] Instituto Geológico Minero y Metalúrgico(INGEMMET). Mapa de susceptibilidad por movimientos en masa del Perú, 2010. URL <https://sigrid.cenepred.gob.pe/sigridv3/documento/3654>.
- [2] Secretaria de Gestión del Riesgo de Desastres(SGRD) Centro Nacional de Estimación, Prevención, y Reducción del Riesgo de Desastres(CENEPRED) Instituto Nacional de Defensa Civil(INDECI) Sistema Nacional de Gestión del Riesgo de Desastres(SINAGERD), Presidencia del Consejo de Ministros(PCM). Plan nacional de gestión del riesgo de desastres planagerd 2014-2021. Technical report, May 2014.
- [3] Instituto Geofísico del Perú. SCTS estudia deslizamientos que afectan a la población de Cuenca y Huayllapampa, 2016. URL <https://www.igp.gob.pe/version-anterior/scts-estudia-deslizamientos-que-afectan-poblacion-cuenca-huayllapampa>.
- [4] Veronica Tofani, Chiara Del Ventisette, Sandro Moretti, and Nicola Casagli. Integration of remote sensing techniques for intensity zonation within a landslide area: A case study in the northern apennines, italy. *Remote Sensing*, 6(2):907–924, Jan 2014. ISSN 2072-4292. doi: 10.3390/rs6020907.
- [5] D Tarchi, Nicola Casagli, Riccardo Fanti, David D. Leva, Guido Luzi, Alessandro Pasuto, Massimiliano Pieraccini, and Sandro Silvano. Landslide monitoring by using ground-based sar interferometry: An example of application to the tessina landslide in italy. *Engineering Geology*, 68:15–30, Feb 2003. doi: 10.1016/S0013-7952(02)00196-5.
- [6] A. Florentino, S. Charapaqui, C. De La Jara, and M. Milla. Implementation of a ground based synthetic aperture radar (gb-sar) for landslide monitoring: system description and preliminary results. In *2016 IEEE XXIII International Congress on Electronics, Electrical Engineering and Computing (INTERCON)*, pages 1–6, Aug 2016. doi: 10.1109/INTERCON.2016.7815579.

- [7] D. Ortecho, M. Milla, S. Charapaqui, E. Arboleda, A. Florentino, and C. De La Jara. Design and implementation of a mechanical system for a ground based synthetic aperture radar with automatic antenna pointing: Preliminary results. In *2017 Electronic Congress (E-CON UNI)*, pages 1–4, Nov 2017. doi: 10.1109/ECON.2017.8247302.
- [8] Caner Özdemir. *Inverse Synthetic Aperture Radar Imaging with MATLAB Algorithms*. John Wiley & Sons, 2012.
- [9] D. Tarchi, H. Rudolf, G. Luzi, L. Chiarantini, P. Coppo, and A. J. Sieber. Sar interferometry for structural changes detection: a demonstration test on a dam. In *IEEE 1999 International Geoscience and Remote Sensing Symposium. IGARSS'99 (Cat. No.99CH36293)*, volume 3, pages 1522–1524 vol.3, 1999.
- [10] O. Monserrat, M. Crosetto, and G. Luzi. A review of ground-based sar interferometry for deformation measurement. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93:40 – 48, 2014. doi: <https://doi.org/10.1016/j.isprsjprs.2014.04.001>.
- [11] J. Fortuny and A. J. Sieber. Fast algorithm for a near-field synthetic aperture radar processor. *IEEE Transactions on Antennas and Propagation*, 42(10):1458–1460, 1994.
- [12] J. M. Lopez-Sanchez and J. Fortuny-Guasch. 3-d radar imaging using range migration techniques. *IEEE Transactions on Antennas and Propagation*, 48(5):728–737, May 2000. doi: 10.1109/8.855491.
- [13] G. Antonello, N. Casagli, P. Farina, D. Leva, G. Nico, A. J. Sieber, and D. Tarchi. Ground-based sar interferometry for monitoring mass movements. *Landslides*, 1:21–28, March 2004. doi: 10.1007/s10346-003-0009-6.
- [14] Mónica Sensat Nogués. Formación de imágenes radar de apertura sintética (sar) de precisión mediante técnicas de retroproyección. 2015.
- [15] A.V. Oppenheim, A.S. Willsky, and A.H. Nawab. *Señales y sistemas 2 ed.* Prentice Hall, 1998. ISBN 970170116X.
- [16] J.G. Proakis and D.G. Manolakis. *Tratamiento digital de señales*. Pearson Educación, 2007. ISBN 9788483223475.
- [17] R.S. Goodman W.G. Carrara and R.M. Majewski. *Spotlight Synthetic Aperture Radar Signal Processing Algorithms*. Artech House, 1995.

- [18] Martin Vetterli Paolo Prandoni. *Signal Processing for Communications*. EPFL Press, 2008.
- [19] M. Pieraccini, G. Luzi, and C. Atzeni. Terrain mapping by ground-based interferometric radar. *IEEE Transactions on Geoscience and Remote Sensing*, 39(10):2176–2181, Oct 2001. doi: 10.1109/36.957280.
- [20] J. Burki, T. Ali, and S. Arshad. Vector network analyzer (vna) based synthetic aperture radar (sar) imaging. In *INMIC*, pages 207–212, Dec 2013. doi: 10.1109/INMIC.2013.6731351.
- [21] Gregory L. Charvat. *Small and Short-Range Radar Systems*. CRC Press, 2014.
- [22] L. Li, W. Zhang, and F. Li. Derivation and discussion of the sar migration algorithm within inverse scattering problem: Theoretical analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 48(1):415–422, Jan 2010. doi: 10.1109/TGRS.2009.2024690.
- [23] Noble B. *Applied Linear Algebra*. Prentice Hall, 1969.
- [24] Demmel J.W. *Applied Numerical Linear Algebra*. SIAM, 1997.
- [25] P. Guccione, M. Zonno, L. Mascolo, and G. Nico. Focusing algorithms analysis for ground-based sar images. In *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*, pages 3895–3898, July 2013. doi: 10.1109/IGARSS.2013.6723683.
- [26] C. Hu, Q. He, W. Tian, Y. Deng, and X. Dong. A performance analysis of typical imaging algorithms in ground-based synthetic aperture radar. In *2017 XXXIInd General Assembly and Scientific Symposium of the International Union of Radio Science (URSI GASS)*, pages 1–4, Aug 2017. doi: 10.23919/URSIGASS.2017.8105149.
- [27] D. Massonnet and J.C. Souyris. *Imaging with Synthetic Aperture Radar*. CRC Press, 2008.

Anexos

ANEXO A

PSEUDOCÓDIGOS PARA LA IMPLEMENTACIÓN DE LOS ALGORITMOS FDBP, RMA Y DLIP

En esta sección se incluyen los pseudocódigos de los tres algoritmos implementados en este trabajo: FDBP, RMA y DLIP.

Adicionalmente se adjunta en un CD los códigos de programación realizados en Python 3 de los tres algoritmos implementados. También es posible revisar los códigos en el siguiente repositorio digital en GitHub:

https://github.com/LuisDLCP/GBSAR_Imaging_Algorithms

A.1 Generador de datos simulados

Los pasos del pseudoalgoritmo así como la terminología usada se basaron en las ecuaciones de la sección 3.4.

Algorithm 1 Función generadora de datos simulados

Input: $c, f_c, BW, N_f, L_s, N_p, I_n, r_n$

Output: $S_{k,i}[N_p, N_f]$

```
1: function GET-RAWDATA():
2:   for  $k = 0$  to  $(N_p - 1)$  do
3:      $u_k = \frac{-L_s}{2} + kd_p$  ▷ // Lista de posiciones
4:   end for
5:   for  $i = 0$  to  $(N_f - 1)$  do
6:      $f_i = f_c - \frac{BW}{2} + i\Delta f$  ▷ // Lista de frecuencias
7:   end for
8:    $S = \text{zeros}(\text{size} = (N_p, N_f))$ 
9:    $S_{aux} = \text{zeros}(\text{size} = (N_p, N_f))$ 
10:  for  $k = 0$  in  $(\text{size}(u_k) - 1)$  do
11:    for  $i = 0$  in  $(\text{size}(f_i) - 1)$  do
12:      for  $n = 0$  to  $(\text{size}(I_n) - 1)$  do
13:         $S_{aux}[k, i] = I_n[n] \times \exp(-1j \frac{4\pi}{c} f_i[i] \|r_n[n] - u_k[k]\|)$ 
14:         $S = S + S_{aux}$ 
```

```

15:         end for
16:     end for
17: end for
18: return S
19: end function

```

A.2 Algoritmo FDBP

Los pasos del pseudoalgoritmo así como la terminología usada se basaron en las ecuaciones de la sección 4.1.

Algorithm 2 Algoritmo Frequency Domain Back Projection(FDBP)

Input: $S_{k,i}, L_x, d_x, L_y, d_y, L_s, N_p, d_p, c, f_c, BW, N_f, f_i, u_k$

Output: I

```

1: function FDBP-ALGORITHM():
2:      $S_1 = S_{k,i} \times \text{hamming-filter}(axis=both)$ 
3:     for  $i = 0$  to  $(N_f - 1)$  do ▷ // Etapa preliminar: Modificación del histórico de fase
4:          $S_2 = S_1 \times \exp(-j\frac{4\pi}{c}R_0\Delta f \times i)$ 
5:     end for
6:      $S_3 = S_2 \times \text{zero-padding}(axis = columns)$  ▷ // Primera etapa: Compresión en rango
7:      $S_4 = \text{ifft}(S_3, axis = columns)$ 
8:      $\Delta r = c/2BW$  ▷ // Segunda etapa: Interpolación
9:      $r_n = \text{get-range-coordinates}(size_1(S_4), \Delta r)$ 
10:     $x[N_x] = \text{getXcoordinates}(d_x, L_x)$ 
11:     $y[N_y] = \text{getYcoordinates}(d_y, L_y)$ 
12:     $r_{xy}[N_x N_y] = \text{getXYcoordinates}(x, y)$ 
13:    function GET-INTERPOLATION( $p, R_d$ )
14:         $H = \text{interp1D}(r_n, S_4[p])$ 
15:        return  $H(R_d)$ 
16:    end function
17:     $S_5 = \text{zeros}(size = size(r_n))$  ▷ // Tercera etapa: Suma coherente
18:    for  $k = 0$  to  $(N_p - 1)$  do
19:         $R_d = \text{get-distances}(u_k[k], r_{xy})$ 
20:         $F = \text{GET-INTERPOLATION}(k, R_d)$ 
21:         $K_e = \exp(1j\frac{4\pi}{c}f_{low}(R_d - R_0))$ 
22:         $S_5 = S_5 + F \times K_e$ 

```

```

23:   end for
24:    $S_6 = S_5 / (N_f N_p)$ 
25:    $S_7 = \text{reshape}(S_6, \text{size} = (N_y, N_x))$ 
26:    $I = \text{normalize}(S_7)$ 
27:   return  $I$ 
28: end function

```

A.3 Algoritmo RMA

El programa mostrado es válido para las pruebas con datos simulados, para las pruebas con datos reales se hacen los siguientes cambios: en la tercera etapa, interpolación STOLT, se usa una interpolación cúbica; además se emplea un método optimizado para realizar la decimación en ambos ejes.

Los pasos del pseudoalgoritmo así como la terminología usada se basaron en los pasos del algoritmo RMA explicados en la sección 4.2.

Algorithm 3 Algoritmo Range Migration (RMA)

Input: $S_{k,i}, L_x, d_x, L_y, d_y, L_s, N_p, d_p, c, f_c, BW, N_f, f_i, u_k$

Output: I, x, y

```

1: function RMA-ALGORITHM():
2:    $S_1 = S_{k,i} \times \text{hamming-filter}(\text{axis}=\text{both})$ 
3:    $S_2 = S_1 \times \text{zero-padding}(\text{size} = (\text{int}(L_x/d_p) - \text{size}(S_1)))$            ▷ // Primera etapa:
    $1D\text{-FFT}(\text{Azimut})$ 
4:    $S_3 = \text{fft}(S_2, \text{axis}=\text{rows})$ 
5:    $S_3 = \text{fftshift}(S_3, \text{axis}=\text{rows})$ 
6:    $S_3^* = \text{cutSpectrum}(S_3, \text{cond} < -60\text{dB})$                                ▷ // * Opcional
7:    $k_r = \frac{4\pi}{c} \times f_i$                                                    ▷ // Segunda etapa: Matched Filter
8:    $k_x = \frac{2\pi}{d_p} \text{fftShift}(\text{fftFreq}(\text{size}_0(S_3)))$ 
9:    $X_{c1} = -L_y/2$ 
10:   $X_{c2} = -L_s/2$ 
11:   $\text{ffase} = -X_{c2}k_x - X_{c1}\sqrt{k_r^2 - k_x^2}$ 
12:   $S_4 = S_3 \times e^{(1j \times \text{ffase})}$ 
13:   $k_y = \sqrt{k_r^2 - k_x^2}$                                                ▷ // Tercera etapa: Interpolacion STOLT
14:   $k_{y-it} = \text{arange}(\text{min}(k_y), \text{max}(k_y), \Delta k_r)$ 
15:   $S_5 = \text{zeros}(\text{size} = (\text{size}(k_x), \text{size}(k_{y-it})))$ 

```

```

16:   for  $i$  in  $size(k_x)$  do
17:        $G_{it} = \text{interp1D}(k_y[i], S_4[i])$ 
18:        $S_5[i] = G_{it}(k_{y-it})$ 
19:   end for
20:    $F_{kx} = \text{int}(L_x/d_x) + 1$ 
21:    $dec_x = \text{int}(size(k_x)/F_{kx})$ 
22:    $S_6 = \text{zeros}(size = ((size_0(S_5)/dec_x), size_1(S_5)))$ 
23:    $ind_1 = \text{int}(size_0(S_5)/dec_x)$ 
24:   for  $i = 0$  to  $(dec_x - 1)$  do
25:        $S_6 = S_6 + S_5[: ind_1]$ 
26:        $S_5 = S_5[ind_1 :]$ 
27:   end for
28:    $S_6 = S_6/dec_x$ 
29:    $k_{xn} = \text{fftshift}(\text{fftfreq}(ind_1) \times ind_1 \times \Delta k_x)$ 
30:    $F_{ky} = \text{int}(L_y/d_y) + 1$ 
31:    $dec_y = \text{int}(size(k_{y-it})/F_{ky})$ 
32:    $S_{6-aux} = S_6.T$  ▷ //  $T$ : Operador de matriz transpuesta
33:    $S_7 = \text{zeros}(size = ((size_0(S_{6-aux})/dec_y), size_1(S_{6-aux})))$ 
34:    $ind_2 = \text{int}(size_0(S_{6-aux})/dec_y)$ 
35:   for  $i = 0$  to  $(dec_y - 1)$  do
36:        $S_7 = S_7 + S_{6-aux}[: ind_2]$ 
37:        $S_6 = S_{6-aux}[ind_2 :]$ 
38:   end for
39:    $S_7 = (S_7.T)/dec_y$ 
40:    $k_{yn} = \text{arange}(ind_2) \times \Delta k_{y-it} + \min(k_{y-it})$ 
41:    $S_8 = \text{ifftshift}(S_7, axis = rows)$  ▷ // Cuarta etapa: 2D-IFFT
42:    $S_8 = \text{ifft2}(S_8)$ 
43:    $S_8 = \text{fftshift}(S_8, axis = (rows, columns))$ 
44:    $I = \text{normalize}(S_8)$ 
45:    $x = \text{linspace}(\frac{-\pi}{\Delta k_{xn}}, \frac{\pi}{\Delta k_{xn}}, size_0(I))$ 
46:    $y = \text{linspace}(\frac{-\pi}{\Delta k_{yn}}, \frac{\pi}{\Delta k_{yn}}, size_1(I)) - Xc_1$ 
47:   return  $I, x, y$ 
48: end function

```

NOTA: La función **arange**($x_i, x_f, \Delta x$) genera una lista de elementos, con valor inicial x_i , valor final x_f y un paso de Δx . Por otro lado, la función **linspace**(x_i, x_f, N) genera una lista de elementos, con valor inicial x_i , valor final x_f y número de elementos N .

A.4 Algoritmo DLIP

Los pasos del pseudoalgoritmo así como la terminología usada se basaron en los pasos del algoritmo DLIP explicados en la sección 4.3.

Algorithm 4 Algoritmo Discrete Linear Inverse Problem (DLIP)

Input: $S_{k,i}, L_x, d_x, L_y, d_y, L_s, N_p, d_p, c, f_c, BW, N_f, f_i, u_k$

Output: I

```

1: function DLIP-ALGORITHM():
2:    $S_1 = \text{reshape}(S_{k,i}, \text{size} = (N_f N_p, 1))$            ▷ // Primera etapa: Obtención del vector
   columna  $S_1$ 
3:    $x = \text{get-Xcoordinates}(L_x, d_x)$                        ▷ // Segunda etapa: Obtención de la matriz  $A$ 
4:    $y = \text{get-Ycoordinates}(L_y, d_y)$ 
5:    $r_{xy} = \text{get-XYcoordinates}(x, y)$ 
6:    $k_s = \frac{2\pi}{c} f_i$ 
7:   function DISTANCE( $r_n, x_k$ )
8:      $d = \text{sqrt}((r_n[0] - x_k)^2 + r_n[1]^2)$ 
9:     return  $d$ 
10:  end function
11:  for  $n = 0$  to ( $\text{size}(r_{xy}) - 1$ ) do
12:     $m = 0$ 
13:    for  $i = 0$  to ( $N_f - 1$ ) do
14:      for  $k = 0$  to ( $N_p - 1$ ) do
15:         $A[m, n] = \exp(-j2k_s[i] \times \text{DISTANCE}(r_{xy}[n], u_k[k]))$ 
16:         $m = m + 1$ 
17:      end for
18:    end for
19:  end for
20:   $rg = \text{rank}(A)$                                            ▷ // Tercera etapa: Cálculo de la pseudoinversa de  $A$ 
21:   $A_p = \text{pseudoinverse}(A)$ 

```

```
22:    $S_2 = \text{matrixProduct}(A_p, S_1)$  ▷ // Cuarta etapa: Cálculo del vector de reflectividades
23:    $S_3 = \text{reshape}(S_2, \text{size} = (L_y/d_y, L_x, d_x))$ 
24:    $I = \text{normalize}(S_3)$ 
25:   return  $I$ 
26: end function
```

ANEXO B
DEMOSTRACIÓN DEL PASO COMPRESIÓN EN RANGO DEL ALGORITMO
FDBP

Las ecuaciones principales del algoritmo FDBP son:

$$I_n = \frac{1}{N_f N_p} \sum_k R_{n,k}^2 F_n^{(k)} \quad (\text{B.1})$$

$$F_n^{(k)} = \sum_i S_{k,i} e^{j \frac{4\pi f_i}{c} (R_{n,k} - R_o)} \quad (\text{B.2})$$

Donde:

- I_n es el valor de la imagen compleja reconstruida en el punto “ n ”
- $R_{n,k}$ es la distancia entre el punto n de la imagen y la posición k del riel (u_k).
- R_o es una constante que toma en cuenta el retraso(delay) de los cables.
- $S_{k,i}$ es el histórico de fase.
- c es la velocidad de la luz.

En este anexo se demostrará que la expresión $F_n^{(k)}$ se puede obtener calculando una transformada inversa de Fourier (IDFT) al histórico de fase $S_{k,i}$.

Partiendo de la ecuación B.2, se realizarán 2 reemplazos de variables. El primero de ellos es:

$$\Delta r = \frac{c}{2BW} \quad (\text{B.3})$$

Expresando $R_{n,k}$ en función de Δr :

$$R_{n,k} \approx n \cdot \Delta r \quad (\text{B.4})$$

Donde: $n = 0, 1, 2, \dots, (N - 1)$

El segundo reemplazo es:

$$f_i = f_0 + \Delta f \cdot i \quad (\text{B.5})$$

Donde: $i = 0, 1, 2, \dots, (M - 1)$; M es el número de frecuencias y f_0 es la menor frecuencia.

Además:

$$\Delta f \approx \frac{BW}{M} \quad (\text{B.6})$$

Reemplazando las ecuaciones B.4 y B.5 en B.2:

$$F_n^{(k)} \approx \sum_i S_{k,i} e^{1j \frac{4\pi}{c} (f_0 + \Delta f \cdot i) (\Delta r \cdot n - R_o)}$$

$$F_n^{(k)} \approx \sum_i S_{k,i} e^{1j \frac{4\pi}{c} (f_0 \cdot n \cdot \Delta r - f_0 \cdot R_o + i \cdot \Delta f \cdot n \cdot \Delta r - R_o \cdot i \cdot \Delta f)}$$

$$\begin{aligned}
F_n^{(k)} &\approx \sum_i S_{k,i} e^{1j \frac{4\pi}{c} f_0 \cdot n \cdot \Delta r} \cdot e^{-1j \frac{4\pi}{c} f_0 \cdot R_0} \cdot e^{1j \frac{4\pi}{c} i \cdot \Delta f \cdot n \cdot \Delta r} \cdot e^{-1j \frac{4\pi}{c} R_0 \cdot i \cdot \Delta f} \\
F_n^{(k)} &\approx e^{1j \frac{4\pi}{c} f_0 (\Delta r \cdot n - R_0)} \sum_i \{S_{k,i} e^{-1j \frac{4\pi}{c} R_0 \cdot i \cdot \Delta f}\} e^{1j \frac{4\pi}{c} i \cdot \Delta f \cdot n \cdot \Delta r}
\end{aligned} \tag{B.7}$$

A la fase del último término exponencial:

$$1j \cdot \frac{4\pi}{c} \cdot i \cdot \Delta f \cdot n \cdot \Delta r$$

Se le hace un reemplazo de las expresiones de Δf y Δr , dadas por la ecuaciones B.6 y B.3 respectivamente:

$$1j \cdot \frac{4\pi}{c} \cdot i \cdot n \cdot \frac{c}{2BW} \cdot \frac{BW}{M}$$

Simplificando, queda:

$$1j \frac{2\pi}{M} \cdot i \cdot n \tag{B.8}$$

Reemplazando la ecuación B.8 en B.7, y haciendo un cambio de variable:

$$F_n^{(k)} \approx K_e \sum_i \{S_{k,i} e^{-1j \frac{4\pi}{c} R_0 \cdot i \cdot \Delta f}\} e^{1j \cdot \frac{2\pi}{M} \cdot i \cdot n}$$

Donde: $K_e \approx e^{1j \frac{4\pi}{c} f_0 (\Delta r \cdot n - R_0)} = e^{1j \frac{4\pi}{c} f_0 (R_{n,k} - R_0)}$.

Ordenando la expresión matemática y realizando otro cambio de variable:

$$F_n^{(k)} \approx K_e \sum_{i=0}^{M-1} S'_{k,i} e^{1j \cdot 2\pi \cdot \frac{i}{M} \cdot n} \tag{B.9}$$

Donde: $S'_{k,i} = S_{k,i} \cdot e^{-1j \frac{4\pi}{c} R_0 \cdot i \cdot \Delta f}$.

La expresión B.9 tiene la forma de una transformada inversa de Fourier discreta (IDFT), por lo tanto expresándolo en esta forma, quedaría:

$$\begin{aligned}
F_n^{(k)} &\approx K'_e \cdot \mathcal{F}_i^{-1} \{S'_{k,i}\} \\
F_n^{(k)} &\approx K'_e \cdot S'_{k,n}
\end{aligned} \tag{B.10}$$

Donde: $n = 0, 1, 2, \dots, (M-1)$; $K'_e = M \cdot K_e$; y \mathcal{F}_i^{-1} es el operador transformada inversa de Fourier respecto a la variable i .

A continuación se interpreta el resultado de la ecuación B.10. Para una posición fija del riel 'k', la expresión inicial de $F_n^{(k)}$ calcula este valor en cualquier punto n de la Imagen, sin embargo la ecuación B.10 calcula $F_n^{(k)}$ en puntos n , los cuales son valores discretos múltiplos de la resolución en rango Δr . Por lo que para calcular el valor en un punto cualquiera a partir de los valores en puntos discretos, se puede usar una interpolación.

Por lo tanto, con el resultado de la ecuación B.10 quedó demostrado que $F_n^{(k)}$ se puede obtener a partir de una 1D-IDFT respecto a la variable frecuencial al histórico de fase modificado $S'_{k,i}$.