

Detección Automática en Señales de Radar Basada en Conocimiento

Por

Santos Martín López Estrada MC., INAOE

Tesis sometida como requisito parcial para obtener el grado de

DOCTOR EN CIENCIAS EN LA ESPECIALIDAD DE CIENCIAS COMPUTACIONALES

en el

Instituto Nacional de Astrofísica, Óptica y Electrónica

Octubre 2009 Tonantzintla, Puebla

Supervisada por:

Dr. René Armando Cumplido Parra Investigador Asociado C

© INAOE 2009 Derechos reservados El autor otorga al INAOE el permiso de reproducir y distribuir copias de esta tesis en su totalidad o en partes.



Resumen

Este trabajo presenta un método alternativo a los reportados previamente en la literatura sobre procesamiento de señales de radar para realizar la detección de blancos inmersos en ruido marítimo, provocado por el reflejo de la onda electromagnética sobre la superficie de las olas. El método presentado es un esquema basado en conocimiento, el cual trabaja en paralelo con la cadena tradicional de procesamiento, con el objetivo de incrementar la probabilidad de detección de blancos pequeños inmersos en ruido no homogéneo.

El esquema basado en conocimiento realiza el reconocimiento del entorno marítimo para seleccionar y configurar el algoritmo de detección adecuado a las condiciones del entorno existentes. Este reconocimiento consiste en una clasificación de textura de los ecos de radar para determinar el estado del mar presente de entre 12 niveles, de acuerdo a la escala de Beaufort/Douglas. La clasificación de textura se realiza mediante descriptores de textura de la matriz de co-ocurrencias de niveles de gris y descriptores de energía, con los se construyen modelos basados en reglas y árboles de decisión para realizar la clasificación de los nuevos datos. La selección y configuración del algoritmo de detección permitirá incrementar su desempeño.

Este trabajo presenta también la implementación en hardware del esquema basado en conocimiento, utilizando una implementación de tipo *pipeline* para asegurar que el esquema completo se ejecute en tiempo real, se destaca la implementación de los módulos de clasificación, la obtención de los descriptores de textura y la implementación de la variante llamada Energy-CFAR paramétrico, la cual surgió como resultado del análisis realizado con los descriptores de textura de energía y que constituye una aportación adicional al método descrito.

Abstract

In this work a cutting edge technology in radar signal processing is presented. A new knowledge-based data processing approach is proposed with the objective of improving the performance of the target detection algorithm by adapting the processing algorithms according to changing environment conditions. This involves combining traditional signal processing with real-time exploitation of a priori knowledge of the environment. Since this mix of knowledge-based and traditional signal processing algorithms requires performing a large number of operations to operate in real-time, custom architectures implemented on reconfigurable hardware have been explored as an alternative processing platform. On this context, this thesis proposes an efficient FPGA-based reconfigurable hardware to implement knowledge-based signal processing for a target detection module in radar signal processing. The discussion is centered in the Knowledgebased signal processing design and the hardware implementation of an architecture for environment knowledge acquisition, knowledge base storage, and inference engine that allow selecting the best algorithm and corresponding parameters according to current environment conditions. The selected target detection algorithms are five variants of the CFAR (Constant False Alarm Rate) algorithm. The environment knowledge acquisition is carried out by texture and energy analysis for clutter recognition. The inference engine is implemented as decision tree and production rules to select the best CFAR algorithm and its parameters according to current environment conditions. Experimental results with simulated and real data are presented and discussed.

Dedicatorias

A mi esposa María Isabel Por su amor y comprensión.

A mis hijos Diego y Jessica Por ser el motor que me impulsa.

A mis padres y hermanos Por su apoyo brindado

Agradecimientos

A la Secretaria de Marina y al Instituto de Investigación y Desarrollo Tecnológico de la Armada de México por las facilidades otorgadas para la realización del presente trabajo.

Al Dr. René Armando Cumplido Parra, por la asesoría y apoyo brindados durante el desarrollo de la investigación.

Al jurado dictaminador por sus valiosos comentarios que enriquecieron el trabajo.

Dra. Claudia Feregrino Uribe Dr. Gustavo Rodríguez Gómez Dr. Carlos Alberto Reyes García Dr. Miguel Octavio Arias Estrada Dr. Esaú Vicente Vivas

A la planta Docente del departamento de Ciencias Computacionales, por sus valiosos conocimientos transmitidos.

Al equipo del proyecto Radar por su valiosa colaboración

Al CONACYT por el apoyo otorgado durante el programa doctoral.

Contenido

Capítulo 1 Introducción	1
1.1 El problema de la detección de blancos	2
1.2 Motivación	3
1.3 Objetivos y alcances	4
1.4 Organización del documento	6
Capítulo 2 _Detección de blancos y su dominio	9
2.1 El problema de la detección de blancos	10
2.2 Ruido marítimo	16
2.2.1 Ruido con distribución Weibull	20
2.2.2 Ruido con distribución K	23
2.3 Algoritmo CFAR	28
2.4 El conocimiento del experto	36
2.5 Cómputo reconfigurable	37
2.6 Resumen	39
Capítulo 3 _Esquema KBSP para detección de blancos	.41
3.1 Introducción a los esquemas KBSP	42
3.2 Adquisición del conocimiento	46
3.2.1 Modelo basado en la matriz de concurrencia de niveles de gris para	
clasificación de textura de estados del mar	48
3.2.2 Modelo basado en descriptores de energía para clasificación de textura de	
estados del mar	53
3.3 Representación del conocimiento	58
3.3.1Descriptores de textura y conocimiento del experto como base de hechos	58
3.3.2 Árboles de decisión y reglas de producción como Base de conocimiento	59
3.4Máquina de inferencias	62
3.5 Filtro de energía	64
3.6 Resumen	67
Capítulo 4 _Resultados experimentales del esquema KBSP	. 69
4.1 Conjunto de datos experimentales	69
4.2 Esquema basado en conocimiento	72
4.2.1 Descriptores GLCM y energía	73
4.2.2 Modelos de clasificación	79
4.3 CFAR parametrito	87
4.4. Filtro de energía	97
4.5 Resumen	104
Capítulo 5 Arquitectura para procesamiento de señales basado en	
conocimiento	107

5.1 Introducción	108
5.2 Hardware para análisis de textura	110
5.3 Clasificador de estados del mar y selección de parámetros CFAR	115
5.4 Procesador CFAR	117
5.5 Filtro de energía	119
5.6 Discusión de resultados	121
5.7 Resumen	123
Capítulo 6 Conclusiones, aportaciones y trabajo futuro	. 125
6.1 Conclusiones y aportaciones del esquema basado en conocimiento	125
6.2 Conclusiones y aportaciones de la arquitectura para procesar señales de radar	
basado en conocimiento	127
6.3 Conclusiones y aportaciones generales	128
6.4 Trabajo futuro	129
6.5 Artículos publicados	127
Glosario de términos	128
Referencias	130

Índice de figuras y tablas

Capítulo 1

Figura 1.	1	Cadena de procesamiento tradicional en señales de radar	3
Figura 1.	2	. Bloque de detección basado en KBSP	5

Capítulo 2

Figura 2. 1. Principio de funcionamiento del radar de pulsos	10
Figura 2. 2. Detección de blancos y determinación de su ubicación.	11
Figura 2. 3. Perfil de rangos típico.	12
Figura 2. 4. Enmascaramiento de blancos por ruido marítimo	13
Figura 2. 5. a) Umbral fijo, el nivel del umbral de comparación representa un	
compromiso entre el número de falsas alarmas y el número de detecciones	
exitosas. b) Umbral variable, si el umbral se adapta a las condiciones de	
amplitud del perfil de rangos se pueden incrementar las detecciones exitosas y	
disminuir las falsas alarmas.	15
Figura 2. 6. Escenario global en el problema de la detección automática.	16
Figura 2. 7. Energía de las olas para distintos estados del mar	17
Figura 2. 8. Energía de las olas para distintos estados del mar	18
Figura 2. 9. Distribución Weibull con tres parámetros de simetría.	23
Figura 2. 10. Distribución K con tres parámetros de simetría.	26
Figura 2. 11 distribución Weibull y K con tres parámetros de simetría.	27
Tabla 2.1. Estado del mar y modelo de ruido asociado.	27
Figura 2. 13. Celdas de guarda en el algoritmo CFAR.	31
Figura 2. 14. Variantes de CFAR y umbrales calculados.	34
Tabla 2.2. Parámetros de configuración de variantes de CFAR.	36
Figura 2. 15. Espectro de cómputo	38

Capítulo 3

Figura 3. 1. Estructura KBSP para procesamiento de señales de radar.	45
Figura 3. 2. Datos reales de radar y acercamiento a su textura	47
Figura 3. 3. Datos sintéticos y acercamiento a su textura.	48
Figura 3. 4. Vecinos del píxel de referencia y direcciones de desplazamiento	49
Figura 3. 5. Imagen de radar generada con datos sintéticos	51
Tabla 3.1 Descriptores de textura para la imagen de la figura 3.5	51
Figura 3. 6. Árboles de decisión generados con weka para datos sintéticos	52
Figura 3. 7. Árboles de decisión generados con weka para datos reales.	53
Figura 3. 8. Modelo para clasificación de estados del mar basado en reglas	53
Figura 3. 9. Mascaras de convolución de laws	55

Figura 3. 10. Árbol de decisión basado en descriptores de energía	55
Figura 3. 11. Árbol de decisiones obtenido con filtros 1D.	56
Figura 3. 12. Niveles de energía en perfiles con blancos	57
Figura 3. 13. Mapa de memoria de la base de hechos.	59
Figura 3. 14. Reglas y árbol de decisión para clasificar estados del mar con datos	
sintéticos y GLCM.	60
Figura 3. 15. Reglas y árbol de decisión para clasificar estados del mar con datos	
reales y descriptores de energía	60
Figura 3. 16. Fusión de variantes de CFAR	63
Tabla 3.2 Comportamiento de las reglas de fusión	64
Figura 3. 17. Estructura del filtro FIR de energía	65
Figura 3. 18. Esquema KBSP mas filtro de energía	65
Figura 3. 19. Filtrado adaptativo de energía y CFAR	67

Capítulo 4

Figura 4. 1. Datos sintéticos, blancos y ruido marítimo.	72
Figura 4. 2. Descriptores de textura con GLCM para datos sintéticos.	73
Figura 4. 3. Descriptores de textura con GLCM para datos reales.	74
Figura 4. 4. Descriptores de textura con Energía para datos sintéticos	76
Figura 4. 5. Acercamiento al descriptor R5L5 para datos sintéticos.	76
Figura 4. 6. Acercamiento al descriptor S5S5 para datos sintéticos	77
Figura 4. 7. Descriptores de textura con Energía para datos reales	78
Figura 4. 8. Acercamiento al descriptor R5L5 para datos reales.	78
Figura 4. 9. Acercamiento al descriptor S5L5 para datos reales.	79
Figura 4. 10. Árbol de decisión para clasificación de estados del mar de datos	
sintéticos basado en la GLCM.	80
Figura 4. 11. Árbol de decisión para clasificación de estados del mar de datos reales	
basado en la GLCM.	81
Figura 4. 12. Matriz de confusión para clasificación de estados del mar de datos	
sintéticos basado en la GLCM.	82
Figura 4. 13. Matriz de confusión para clasificación de estados del mar de datos	
reales basado en la GLCM.	82
Figura 4. 14. Matriz de confusión para el conjunto de prueba datos sintéticos	83
Figura 4. 15. Matriz de confusión para el conjunto de prueba datos reales	84
Figura 4. 16. Modelo de Reglas generadas con datos sintéticos	85
Figura 4. 17. Modelo de Reglas generadas con datos reales	85
Figura 4. 18. Esquema KBSP para procesamiento de señales de radar	86
Figura 4. 19. Centro de fusión de salidas de CFAR	87
Figura 4. 20. Detección con el centro de fusión	89
Figura 4. 21. Métrica de desempeño del centro de fusión	90
Figura 4. 22. Salidas de las variantes CFAR sin configurar.	92

Figura 4. 23. Desempeño de variantes CFAR con Pfa de 10 ⁻³ , M=10, GC=2, Ruido	
Gaussiano.	. 93
Figura 4. 24. Desempeño de variantes CFAR con Pfa de 10 ⁻³ , M=20, GC=2, Ruido	
Gaussiano.	. 94
Figura 4. 25. Desempeño de variantes CFAR con Pfa de 10 ⁻⁴ , M=10, GC=2, Ruido	
Gaussiano.	. 95
Figura 4. 26. Desempeño de variantes CFAR con Pfa de 10 ⁻⁵ , M=10, GC=2, Ruido	
Gaussiano.	. 96
Figura 4. 27. Desempeño de variantes CFAR con Pfa de 10 ⁻³ . Versiones no	
configuradas utilizan M=10, GC=4	. 97
Figura 4. 28. Respuesta en amplitud de las mascaras de Laws	. 98
Figura 4. 29. Respuesta en amplitud y fase de un filtro FIR de orden 5 paso bajas	. 99
Figura 4. 30. Respuesta al impulso de un filtro FIR de orden 5 paso bajas	100
Figura 4. 31. Coeficientes normalizados del filtro FIR de orden 5 paso bajas	100
Figura 4. 32.Respuesta en amplitud y fase del filtro L5.	101
Figura 4. 33 Respuesta la impulso del filtro L5.	101
Figura 4. 34. Salida de la nueva variante Energy-CFAR comparada con TM-CFAR	103
Figura 4. 35 Comparativa de eficiencia entre variantes de CFAR	104

Capítulo 5

110
112
114
116
118
119
120
121
123

Capítulo 1.

Introducción.

La percepción o visión por computadora tiene por objetivo diseñar, realizar y evaluar algoritmos y métodos para generar sistemas que permitan simular las capacidades visuales en los seres vivos, recuperando información relevante de una escena y trasladándola a un plano bidimensional conocido como imagen. La recuperación de la información se realiza mediante el análisis de la salida del sensor utilizado, la información que entrega el sensor pueden ser intensidades de gris o color de píxel, en el caso de una cámara, o bien, la amplitud, frecuencia y fase de una señal, en el caso de un sistema de radio (sonar o radar).

El radar de navegación en particular, proporciona ayuda visual sobre los blancos u objetivos (buques, boyas, tierra) presentes en un escenario de navegación, muestra información de posición y distancia o rango de cada uno de ellos. El desempeño de la ayuda visual se ve afectado por las condiciones climatológicas existentes (estas condiciones provocan variación en la altura de las olas del mar, a lo que llamaremos estados del mar). Es esencial contar con los medios adecuados para discriminar el ruido provocado por las condiciones atmosféricas que permitan una detección adecuada para realizar posteriormente el seguimiento de los blancos.

La discusión se inicia con la descripción del problema de la detección de blancos.

1.1 El problema de la detección de blancos.

La cadena tradicional de procesamiento de señales de radar se muestra en la figura 1.1. En cada uno de los bloques se diseñan e implementan algoritmos de procesamiento para condiciones determinadas del entorno. En particular en el módulo de detección se lleva a cabo la separación de blancos y ruido provocado por las condiciones atmosféricas. Esta separación se lleva a cabo mediante el algoritmo CFAR (Constant False Alarm Rate) [1], el cual es diseñado para ciertas condiciones del entorno o estados del mar. En este diseño el usuario tiene una participación activa al modificar algunos parámetros del algoritmo para incrementar el rendimiento en la detección cuando cambian las condiciones del entorno. En este sentido, en la literatura sobre procesamiento de señales de radar se reportan trabajos que contribuye a mejorar el rendimiento de un sistema, empleando técnicas basadas en conocimiento KBSP por sus siglas en inglés (Knowledge Based Signal Processing), combinadas con el procesamiento tradicional. El KBSP tiene como principio fundamental mejorar el desempeño de un sistema, cambiando los algoritmos de procesamiento a medida que cambian las condiciones del entorno. Como menciona F. Gini en [2] "La clave de KBSP es explotar en tiempo real el conocimiento a priori del entorno, para ello se requiere de procesamiento inteligente en toda la cadena de procesamiento de señales", lo cual representa un amplio y nuevo campo de investigación en el procesamiento de señales. En el caso

particular de señales de radar se requiere en las etapas de filtrado, integración, detección y seguimiento e identificación.

Del párrafo anterior se puede notar que al desarrollar un método para implementar cualquiera de los bloques de procesamiento de señales de radar empleando técnicas KBSP se requiere incluir técnicas de cómputo reconfigurable [3], para realizar el cambio de los algoritmos de procesamiento con el cambio del entorno.



Figura 1. 1 Cadena de procesamiento tradicional en señales de radar.

1.2 Motivación.

El procesamiento de señales de radar requiere de sistemas capaces de mantener su eficiencia ante cambios constantes del entorno, es decir robustos en cuanto al procesamiento. Los sistemas empleados hasta el momento son eficientes sólo en condiciones favorables del entorno [4], en este sentido el empleo de una variante del algoritmo CFAR con sus parámetros adecuados para cada estado del mar proporciona la robustez necesaria para incrementar la detección de blancos.

El empleo de técnicas KBSP para procesar señales de radar constituye la base para las nuevas generaciones de dispositivos de radar, sin embargo, se requiere de métodos y plataformas de procesamiento eficientes para operar en tiempo real. La obtención de métodos de procesamiento que emplean técnicas KBSP constituye un aporte para mejorar el procesamiento tradicional de señales de radar. Estos métodos son aplicados en otros dominios como imágenes de ultrasonido, sistemas de audio y comunicaciones, donde el ruido del entorno es una fuente de interferencia dominante y no puede ser eliminado por los métodos tradicionales de procesamiento de señales.

Por otro lado, en el área de procesamiento de señales de radar constantemente se realiza investigación sobre plataformas que permitan evaluar de manera eficiente y oportuna esquemas de detección, por lo que una plataforma reconfigurable, permitirá realizar el prototipado de esquemas de detección en señales de radar empleando técnicas KBSP.

La reconfiguración cobra importancia cuando las aplicaciones requieren tener diferentes diseños de algoritmos o circuitos digitales en la misma área de un dispositivo, o bien cuando la aplicación debe tener la flexibilidad para cambiar parte de un diseño sin apagar el sistema o reconfigurarlo totalmente. Con esta capacidad es posible realizar actualizaciones de hardware y reconfiguración en tiempo de ejecución [5].

La tecnología basada en FPGA ofrece la flexibilidad para experimentar con distintas arquitecturas y reconfigurar la unidad FPGA tantas veces como sea necesario hasta obtener el modelo que satisfaga las necesidades del problema.

1.3 Objetivos y alcances.

General:

Desarrollar un método de procesamiento de señales de radar basado en conocimiento para mejorar el desempeño en el proceso de detección automática, respecto a los reportados en la literatura y tendiendo al detector ideal, empleando información del entorno proporcionada por un solo sensor.

Particulares:

- Seleccionar y evaluar la información del entorno en escenarios de navegación y su representación para ser usada con las técnicas basadas en conocimiento.
- Desarrollar, evaluar e implementar una arquitectura hardware parametrizable que ejecute el método propuesto en tiempo real.

En este trabajo se presenta un método de procesamiento de señales de radar basado en conocimiento, que permite desarrollar un módulo de detección automática basado en conocimiento y procesamiento tradicional, como se muestra en la figura 1.2. Así mismo, se proporciona el diseño de una arquitectura para realizar el procesamiento del módulo de detección. La arquitectura diseñada se basa en un esquema que implementa el algoritmo de detección CFAR parametrizado, para ser configurado automáticamente de acuerdo a las condiciones del entorno existentes. Se implementa en hardware el módulo KBSP el cual consiste en un sistema experto que analiza las señales de radar para determinar las condiciones del escenario (estados del mar).

El método de procesamiento de señales y la arquitectura diseñada son para radares de superficie monoestáticos (transmisión y recepción por la misma antena) pulsados considerando sólo la componente de amplitud de la señal de radar [1], los blancos a detectar son embarcaciones de diferente sección cruzada de radar (área que presenta el blanco a la radiación electromagnética emitida por el radar) [6]. El escenario o entorno de detección estará formado por múltiples blancos y el ruido generado por el reflejo electromagnético provocado por las olas del mar (ruido marítimo) y la tierra (ruido terrestre) en diferentes condiciones climatológicas (estados del mar).



Figura 1. 2. Bloque de detección basado en KBSP.

1.4 Organización del documento.

En el capítulo dos se expone en detalle el problema de la detección de blancos inmersos en ruido marítimo y terrestre, así como el dominio del problema, es decir, la información del entorno de navegación que se utilizó para generar la base de conocimiento y la base de datos de hechos empleada en el método desarrollado. Se presenta el análisis de trabajos previos en ruido marítimo, variante del algoritmo de detección y el conocimiento del experto que apoyó en este trabajo.

El capítulo tres expone el diseño del método KBSP para detección de blancos, destacando principalmente la adquisición y representación del conocimiento y la máquina de inferencias diseñada. Se expone, también, el diseño de un filtro digital que complementa el procesamiento tradicional de señales y que constituye una aportación adicional al trabajo de investigación.

En el capítulo cuatro se realiza la discusión de los resultados de los experimentos planteados en el capítulo tres, se muestran resultados obtenidos con datos reales de radar y datos simulados.

El capítulo cinco describe el diseño de la arquitectura que procesa el algoritmo KBSP diseñado para evaluar las condiciones del entorno, se detalla la implementación en hardware de los módulos KBSP, el algoritmo de detección y una variante nueva de CFAR propuesta en este trabajo.

Finalmente, el capítulo seis presenta las conclusiones obtenidas con el trabajo desarrollado y se presentan algunas direcciones en las que se puede extender el trabajo para integrar procesamiento KBSP en otros módulos de la cadena de procesamiento de radar.

Capítulo 2.

Detección de blancos y su dominio.

En este capítulo se discutirá el problema de la detección de blancos y su dominio, es decir, los principales elementos que intervienen en el proceso de detección de blancos en un escenario de navegación, conteniendo costa, mar abierto, ninguno o múltiples blancos y distintas condiciones atmosféricas. En las secciones siguientes se abordará la problemática de la detección; una descripción del ruido marítimo desde el punto de vista electromagnético y los modelos empleados para describir su comportamiento; los algoritmos empleados para realizar la detección, la cual consiste en separar el ruido marítimo de los blancos; las acciones que realiza el experto para obtener una buena detección y finalmente las opciones existentes para llevar a cabo una implementación en hardware de un esquema de detección basado en conocimiento. En cada una de las secciones se realizará una revisión de los trabajos reportados en la literatura, principalmente aquellos enfocados al procesamiento basado en conocimiento.

2.1 El problema de la detección de blancos.

El principio de funcionamiento de un radar de pulsos se muestra en la figura 2.1. Un transmisor genera una señal electromagnética con un pulso corto de una señal senoidal, esta señal es radiada al espacio por la antena. Una parte de la energía transmitida es reflejada por los blancos en distintas direcciones, esta energía es conocida como sección cruzada de radar. La energía en dirección al radar es captada por la antena y enviada al receptor, en donde es procesada para detectar la presencia de blancos y determinar su localización, esta información es presentada al usuario como una imagen, la cual se presenta en un display llamado indicador plano de posición PPI por sus siglas en inglés (Plan Position Indicator) como se muestra en la figura 2.2. La localización del blanco se obtiene midiendo la distancia y el ángulo a partir de la emisión de la onda electromagnética, el rango o distancia se obtiene midiendo el tiempo que tarda en viajar la señal hasta el blanco y regresar, el ángulo o azimut se obtiene enviando un pulso en cada posición angular de la antena, esta posición depende del ancho del haz y de la velocidad de giro de la antena. Al conjunto de pulsos emitidos en una vuelta se le llamará tren de pulsos y a la frecuencia de transmisión de ellos se le llama frecuencia de repetición del pulso o PRF por sus siglas en inglés (Pulse Repetition Frequency).



Figura 2. 1. Principio de funcionamiento del radar de pulsos.



Figura 2. 2. Detección de blancos y determinación de su ubicación.

La amplitud de los ecos reflejados tanto por los buques como por los picos de las olas, la tierra y otros elementos atmosféricos es obtenida como una función del tiempo, que es proporcional al rango o distancia de los blancos, esta función del tiempo es conocida como la envolvente de los ecos de radar [7]. La señal o envolvente de los ecos es obtenida en el receptor de manera analógica, la cual es digitalizada. Cada muestra (x(i)), corresponde a un rango o alcance discreto. A esta muestra se le llama celda de rango y al conjunto de celdas de rango se le llama perfil de rango (x(t)). El perfil de rango queda definido como se indica a continuación:

$$x(t) = [x(1), x(2), \dots, x(n)]$$
(2.1)

El número de celdas en el perfil depende del alcance del radar y de la resolución del mismo, la cual a su vez depende del ancho del pulso emitido. Por ejemplo un radar con un ancho de pulso de 0.07µs, tiene una resolución de 10.5 metros, esto es, la mínima distancia para distinguir dos blancos cercanos, por lo que para un alcance de 1.5 millas náuticas el perfil de rangos debe tener al menos 260 muestras. El perfil de rango mostrado en la figura 2.3 es procesado digitalmente para obtener los blancos presentes en la señal.



Figura 2. 3. Perfil de rangos típico.

El entendimiento de las señales de radar inmersas en ruido y los requerimientos de procesamiento son fundamentales para el desarrollo de las nuevas generaciones de sistemas de radar [8]. La combinación adecuada de algoritmos y plataformas de procesamiento de señales, permiten desarrollar sistemas con diversidad de procesamiento adaptativo en espacio, tiempo y frecuencia.

La detección de buques y embarcaciones menores por señales de radar en entornos marítimos sufre serias limitaciones en su desempeño, provocadas por el eco reflejado por los picos de las olas del mar (ruido marítimo) principalmente cuando la amplitud del ruido es mayor que la del blanco, como se observa en la figura 2.4 [9]. Se conoce una clasificación de 12 estados del mar de acuerdo a la altura de las olas y velocidad del viento, conocida como escala de Beaufort/Douglas) [10]. El ruido marítimo en la mayoría de los radares ha sido considerado como una perturbación con distribución Gaussiana, sin embargo, se ha mostrado que en condiciones adversas puede ser representado con distribuciones Rayleigh, Weibull y K [4], [7]. Estos modelos permiten discriminar de una mejor manera el ruido marítimo, solo requieren de seleccionar adecuadamente sus parámetros de acuerdo a las condiciones del entorno existentes.



Figura 2. 4. Enmascaramiento de blancos por ruido marítimo.

En el proceso de detección generalmente se utilizan algoritmos adaptativos como CFAR. Este algoritmo mantiene constantes y en un bajo nivel las falsas detecciones, siempre y cuando se elijan adecuadamente sus parámetros de configuración de acuerdo a las condiciones del entorno (ruido marítimo). La entrada para CFAR es el perfil de rangos proveniente del receptor del radar. El algoritmo utiliza un umbral para discriminar los buques del ruido marítimo. Si el umbral se mantiene fijo en un nivel bajo se incrementan las falsas detecciones. Por otro lado, si el umbral se mantiene en un nivel alto, es posible que el eco de algunos blancos pequeños no alcancen el nivel de comparación y no sean detectados, incrementando con ello los blancos perdidos, como se muestra en la figura 2.5a.

Para realizar la discriminación de blancos y ruido marítimo en condiciones cambiantes del entorno, es necesario utilizar distintos niveles de umbral o bien, tener un umbral que se adapte a la amplitud del perfil de rangos, lo cual implica que se debe conocer la forma en que se comportan las ondas electromagnéticas al ser reflejadas por las olas del mar. Un umbral que se adapte a las condiciones del perfil de rangos (de acuerdo a las condiciones meteorológicas) permite obtener mejores resultados en la detección de blancos y disminuir la probabilidad de detección de ecos falsos, como se observa en la figura 2.5b.

El problema de la detección ha sido atacado por múltiples grupos de investigación y de diferentes formas. Las propuestas de solución se han encaminado a utilizar algún modelo de ruido marítimo (Rayleigh, Weibull, K) que describa las condiciones del mar en toda la escala de Beaufort/Douglas. Otras propuestas se enfocan a mejorar el rendimiento del algoritmo de detección CFAR, realizando modificaciones al algoritmo base como Order Statistics CFAR o Trimmed Mean CFAR. El enfoque de fusión también ha sido abordado para dar solución al problema de detección en entornos con ruido no homogéneo [11]. Recientemente se han incorporado propuestas que emplean técnicas de procesamiento de señales basado en conocimiento (KBSP), en donde el conocimiento a priori del comportamiento del ruido o del entorno puede ser útil para eliminarlo adecuadamente [2], [3], [5].

De acuerdo a las condiciones planteadas en la problemática de la detección, se cuenta con múltiples variantes del algoritmo adaptativo CFAR, donde cada una de estas variantes debe ser utilizada en las condiciones del entorno para las cuales fueron diseñados, por lo que contar con un módulo de detección en donde el algoritmo CFAR sea reconfigurado de acuerdo a las condiciones presentes del entorno, permitirá mantener constantes y en un bajo nivel las falsas detecciones, incrementando la probabilidad de detección. El método de procesamiento que emplee el módulo de detección debe contar con una plataforma de procesamiento eficiente que realice la tarea en tiempo real. La figura 2.6 muestra un escenario global en donde el punto central es un módulo de detección basado en conocimiento mediante una plataforma de procesamiento que permita reconfigurar los parámetros del algoritmo CFAR a medida que cambien las condiciones del entorno. Considerando que una de las tareas principales en el procesamiento de señales de radar es la detección y posteriormente el seguimiento de blancos inmersos en entornos que cambian dinámicamente con las condiciones meteorológicas. Al usar técnicas KBSP se puede tener ejecutando el algoritmo CFAR con sus parámetros adecuados para las condiciones del escenario presentes. A este proceso de selección

de algoritmo y sus parámetros sin la intervención del usuario se le llamará detección automática [6][12].



Figura 2. 5. a) Umbral fijo, el nivel del umbral de comparación representa un compromiso entre el número de falsas alarmas y el número de detecciones exitosas.b) Umbral variable, si el umbral se adapta a las condiciones de amplitud del perfil de rangos se pueden incrementar las detecciones exitosas y disminuir las falsas alarmas.

De acuerdo al escenario planteado en la figura 2.6 debemos analizar la información del entorno, en este caso el ruido marítimo, las variantes del algoritmo de detección empleados y el conocimiento que aplica el experto en diferentes situaciones del entorno, así mismo se deben analizar las alternativas existentes en cuanto a plataformas para la implementación de un esquema KBSP. En las siguientes secciones se describe el análisis realizado en cada una de los componentes

del dominio, finalizando con un resumen de la información útil del entorno para la implementación del esquema KBSP.



Figura 2. 6. Escenario global en el problema de la detección automática.

2.2 Ruido marítimo.

En radares de superficie o radares de navegación se considera ruido marítimo a los ecos provocados por los picos de las olas cuando es emitida una onda electromagnética. Los ecos producidos por el ruido marítimo tienen distintas amplitudes, que pueden ser presentadas en la pantalla del radar como blancos falsos si se trata de olas amplias y altas o bien como ruido de fondo si son olas cortas y pequeñas. Un tipo de olas que se presentan en situaciones críticas son aquellas que presentan gran altura pero de corta duración [13]. En la figura 2.7 se observan las curvas de distribución de probabilidad de la energía de las olas del primer tipo, es decir olas altas amplias y cortas pequeñas, mientras que en la figura 2.8 se muestra la distribución de probabilidades para olas altas y de corta duración. En estas figuras se puede notar que las curvas tienden a ser simétricas y de poca energía para estados del mar bajo, mientras que se hacen más espigadas en estados del mar altos de acuerdo a la escala de Beaufort/Douglas.



Figura 2. 7. Energía de las olas para distintos estados del mar.

El ruido marítimo cambia constantemente en el tiempo de acuerdo a las condiciones meteorológicas existentes, principalmente debido a la capilaridad de las olas y la velocidad del viento, variando los niveles de energía de la ola o estado del mar, en alguno de los 12 niveles de la escala de Beaufort/Douglas como se observa en las figuras 2.7 y 2.8.

Las perturbaciones en el perfil de rangos, provocadas por las olas, se comportan como un proceso dinámico no lineal, como el presente en los procesos de modulación múltiple. En los últimos años se han estudiado tales procesos, enfocándose principalmente en su naturaleza estadística, considerándolo como un proceso estocástico, aunque en los trabajos de Haykin y Baker [14] emplean la teoría del caos para modelar su comportamiento. El ruido marítimo debe ser caracterizado en amplitud, fase y polarización, aunque en un radar pulsado como el empleado en el

presente trabajo sólo se puede caracterizar la amplitud de los ecos. Esta característica, restringe el procesamiento tradicional de señales, ya que el análisis de Fourier se basa en la información de fase y polarización. En este trabajo, las técnicas empleadas procesan la energía de la señal presente en la amplitud de la misma.



Figura 2.8. Energía de las olas para distintos estados del mar.

Continuando con el enfoque estadístico para el análisis de ruido, Cowper y Mulgrew en [15] realizan un estudio de las principales técnicas de modelado de ruido marítimo y concluyen que el empleo de técnicas lineales no genera resultados adecuados debido a la naturaleza no Gaussiana del ruido. Los autores muestran que se tiene una considerable disminución de falsas detecciones al utilizar modelos estocásticos de alta resolución como la distribución K o Weibull.

Davidson y Ouchi en [16] realizan un análisis numérico sobre los métodos de calculo de los parámetros de la distribución K para modelar ruido marítimo. Logran acelerar el tiempo de procesamiento en la estimación de tales parámetros, comparado con los métodos de estimación Monte Carlo. Con el esquema propuesto pueden calcular umbrales de detección con niveles extremos para la probabilidad de falsa alarma como 10⁻¹⁰, lo cual, en el caso de radares de superficie es un valor inalcanzable por el momento, ya que los modelos del ruido no describen exactamente el comportamiento del ruido.

Chong y Zhu en [17] proponen la utilización del modelo Gaussiano, para pequeñas ventanas de procesamiento, utilizan 10-24 celdas del perfil de rangos, logrando buenos resultados pero sólo con blancos inmersos en ruido con distribución Gaussiana presente en los estados del mar 0 y 1 de acuerdo a la escala de Beaufort/Douglas. Los autores proponen la utilización de un modelo K de distribución para el ruido marítimo, con lo que obtienen buenos resultados solo en mares complejos (estado del mar 5 o superiores).

Watts en [18], realiza un estudio sobre el algoritmo CFAR, muestra su confiabilidad al utilizar modelos de ruido con distribución K para estimar los umbrales de detección, utiliza el análisis estadístico del ruido marítimo con una distribución normalizada. Por otro lado, Levanon y Shor en [19], muestran el comportamiento del algoritmo CFAR cuando se utiliza en ruido con distribución Rayleigh y Weibull. El inconveniente es que el parámetro de simetría de las distribuciones debe ser conocido.

López y Cumplido en [20] proponen tres procesadores CFAR corriendo en paralelo. Cada uno de estos algoritmos es combinado con algún modelo de ruido (Weibull, Rayleigh, K). Cada procesador es analizado por separado, obteniendo mayor probabilidad de detección respecto al algoritmo base en condiciones de ruido no homogéneo.

J. Leszek [21] en su tesis doctoral realiza una revisión de los modelos de ruido existentes, concluyendo que se puede emplear una distribución Rayleigh para estados del mar con poco movimiento y olas con periodos largos, mientras que una distribución K es adecuada para mares agresivos en donde se tienen olas con periodos cortos y picos altos.

Por lo anteriormente expuesto se tiene que ninguno de los modelos empleados puede ser usado para los 12 estados del mar, sin embargo, los modelos Weibull, Rayleigh y K cuentan con el parámetro de forma, el cual puede ser ajustado para ser usado en un grupo de estados del mar. En los siguientes apartados se muestran las características de las distribuciones Weibull y K, ya que la distribución Rayleigh es un caso particular de la distribución Weibull.

2.2.1 Ruido con distribución Weibull.

Como se ha mencionado anteriormente, el modelo Weibull puede ser empleado para modelar los ecos provocados por las olas en algunos estados del mar, principalmente aquellos con poco movimiento y olas con periodo largo. El modelo Weibull es ampliamente usado en predicción de tiempos de falla en equipos o en materiales. En el caso de señales de radar puede ser empleado para modelar picos de olas en algunos estados del mar, prediciendo tiempos de presencia de falsas alarmas. Esta distribución es un modelo adaptativo, ya que se tienen coeficientes (coeficientes de Fisher) o parámetros variables: de simetría o forma y escala o apuntamiento (momentos de tercer y cuarto orden respectivamente). En la figura 2.9a se muestran las curvas de funciones de probabilidad simétrica, asimétrica izquierda y asimétrica derecha, mientras que en la figura 2.9b se muestran las curvas de funciones de probabilidad con mayor o menor apuntamiento respecto a la normal. En el caso del modelo de Weibull aplicado a ruido marítimo se busca que las colas de la distribución tengan la menor probabilidad, es decir distribuciones con mayor apuntamiento y asimetría izquierda, para facilitar la separación del ruido marítimo de los blancos.

La distribución Weibull tiene algunos casos especiales [22], [23]: 1.- La distribución es simétrica con un parámetro cercano a 3.6 (distribución

normal), siendo asimétrica izquierda o asimétrica derecha con valores menores o mayores a 3.6 respectivamente.

2.- La distribución es exponencial con un parámetro de simetría de 1.

3.-La distribución es de tipo Rayleigh con un parámetro de simetría de 2.



Figura 2. 9. Funciones de distribución, a) simetría o forma, b) apuntamiento o escala.

Con parámetros de simetría comprendidos entre 0.5 y 2.0 se ha utilizado para describir el ruido marítimo asociado a los estados del mar 2, 3 y 4 [19]. La función de distribución Weibull se muestra en la ecuación 2.2, las ecuaciones 2.3 y 2.4 muestran la media y la desviación estándar y las ecuaciones 2.5 y 2.6, obtenidas con el método de máxima verosimilitud [24], muestran los parámetros de forma y escala.

$$f(x) = \frac{a}{b} \left(\frac{x}{b}\right)^{a-1} e^{-\left(\frac{x}{b}\right)^{a}},$$
 (2.2)

$$\mu = b\Gamma\left(1 + \frac{1}{a}\right),\tag{2.3}$$

$$\sigma = b \sqrt{\Gamma\left(1 + \frac{2}{a}\right) - \Gamma^2\left(1 + \frac{1}{a}\right)},$$
(2.4)

$$\frac{1}{a} = \frac{\sum_{i=1}^{m} x_i^a \ln x_i}{\sum_{i=1}^{m} x_i^a} - \frac{1}{m} \sum_{i=1}^{m} \ln x_i \quad , \qquad (2.5)$$

$$b^{a} = \frac{1}{m} \sum_{i=1}^{m} x_{i}^{a} \quad , \tag{2.6}$$

donde *a* es el parámetro de forma o simetría, *b* es el parámetro de escala o apuntamiento, Γ es la función Gamma, *x* representa el conjunto de celdas del perfil de rango, μ es la media y σ es la desviación estándar. En la figura 2.10 se muestran tres curvas de la distribución Weibull con distintos parámetros de simetría.



Figura 2. 10. Distribución Weibull con tres parámetros de simetría.

2.2.2 Ruido con distribución K.

Uno de los modelos más recientes para analizar las fluctuaciones en la amplitud del ruido marítimo es la distribución K, la cual es una distribución compuesta, formada por la distribución Rayleigh modulada en amplitud por la distribución Gamma [17], [25]. Esta distribución proporciona buenos resultados en la representación estadística para estados del mar con olas altas y espigadas como las presentes en los estados del mar superiores al 5 [18], [25], [26]. Algunos casos especiales de la distribución K se presentan al variar su parámetro de simetría, teniendo los siguientes:

1.- Con la simetría mayor o igual a 4 se convierte en Rayleigh.

2.-Con 0.3<simetría<4 la distribución K modela ruido marítimo con blancos de forma simple.

3.-Con parámetro de simetría pequeño se producen colas largas en la distribución

La función de distribución K se muestra en la ecuación 2.7, mientras que la ecuación 2.8 muestra los momentos de orden r. En las ecuaciones 2.9 y 2.10 se muestran los momentos de primer y segundo orden, es decir la media μ y la desviación estándar σ .

$$f(x) = \frac{2}{b\Gamma(a+1)} \left(\frac{x}{2b}\right)^{a+1} K\left(\frac{x}{b}\right), \qquad (2.7)$$

$$\mathbf{E}(x^{r}) = (2b)^{r} \frac{\Gamma\left(1 + \frac{r}{2}\right)\Gamma\left(a + 1 + \frac{r}{2}\right)}{\Gamma(a + 1)},$$
(2.8)

$$\mu = (2b) \frac{\Gamma\left(1 + \frac{1}{2}\right)\Gamma\left(a + 1 + \frac{1}{2}\right)}{\Gamma(a+1)}, \qquad (2.9)$$

$$\sigma = (2b)^2 \frac{\Gamma\left(1 + \frac{2}{2}\right)\Gamma\left(a + 1 + \frac{2}{2}\right)}{\Gamma(a+1)},$$
(2.10)

Para obtener los parámetros de simetría (*a*) y escala (*b*) se utilizó el método propuesto por Raghavan en [27], el cual estima los parámetros por máxima verosimilitud. El método de Raghavan emplea la media geométrica y la media aritmética de una distribución Gamma (I) para aproximar la distribución K, ya que esta última por ser una distribución compuesta, requiere de una solución numérica de ecuaciones complejas. Al aplicar máxima verosimilitud a la distribución Gamma se obtienen las expresiones mostradas en las ecuaciones 2.11 y 2.12 para los parámetros de simetría y apuntamiento, mientras que las ecuaciones 2.13 y 2.14 muestran la forma de obtener la media aritmética μ_{ar} y la media geométrica μ_{ge} respectivamente.
$$\frac{\mu_{ar}}{\mu_{ge}} = ae^{-\psi(a)} = \prod_{k=0}^{m} \left(1 + \frac{1}{a+k}\right)^{-1} e^{\frac{1}{x+k}} , \qquad (2.11)$$

$$b = \frac{\mu_{ar}}{a} \tag{2.12}$$

$$\mu_{ar} = \frac{1}{n} \sum_{i=1}^{n} X_i \quad , \tag{2.13}$$

$$\mu_{ge} = \left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{n}}$$
(2.14)

donde *x* representa las muestras del perfil de rangos y *n* el número de muestras.

En trabajos desarrollados recientemente se ha propuesto un nuevo método para calcular los coeficientes con mayor rapidez y precisión, este es el propuesto por Radoi [28], el cual, calcula los parámetros en función de los momentos de primer y segundo orden, como se muestra en las ecuaciones 2.15, 2.16, 2.17 y 2.18, se disminuye con ello la complejidad computacional para calcular los parámetros de la distribución. En la figura 2.11 se muestra la distribución K con tres parámetros de simetría, asociados a 3 estados del mar.

$$\alpha = \frac{m_1}{2\Gamma\left(\frac{3}{2}\right)} \quad , \tag{2.15}$$

$$\beta = \frac{m_2}{4\Gamma(2)} \quad , \tag{2.16}$$

$$b = \frac{\Gamma\left(a + \frac{1}{2}\right)}{\alpha\Gamma(a)}, \qquad (2.17)$$

$$b\Gamma(a)\Gamma(a+1) = \left(\frac{\beta}{\alpha^2}\right)\Gamma^2\left(a+\frac{1}{2}\right)$$
(2.18)

donde m_1 , m_2 son los momentos de orden 1 y 2 respectivamente, a y b son el parámetro de simetría y escala respectivamente, α y β son parámetros auxiliares.



Figura 2. 11. Distribución K con tres parámetros de simetría.

Finalmente en la figura 2.12 se muestran juntas las funciones de distribución Weibull y K con sus respetivos parámetros de simetría, donde se observa que se pueden hacer corresponder un modelo y su parámetro de simetría a un estado del mar. La tabla 2.1 muestra un resumen de los modelos de distribución y sus parámetros de forma asociados a cada estado del mar que es parte de la información útil que lleva a desarrollar el sistema basado en conocimiento para suprimir este tipo de ruido.



Figura 2. 12 Distribución Weibull y K con tres parámetros de simetría.

Estado de mar	Altura olas (m)	Descripción	Distribución	Parámetro forma
0	0	Calma	Gaussiana	Simétrico
1	0-0.1	Calma rizado	Gaussiana	Simétrico
2	0.1 – 0.5	Suave con olas	Weibull	2.0
3	0.6 - 1.2	Ligero	Weibull	1.4
4	1.2 – 2.4	Moderado	Weibull	0.8
5	2.4 - 4.0	Rugoso	K	1.9
6	4.0-6.0	Muy rugoso	К	1.7
7	6.0 - 9.0	Alto	K	1.5
8	9.0 - 14	Muy alto	K	1.3
9	Mas de 14	Fenomenal	K	1.1

Tabla 2.1. Estado del mar y modelo de ruido asociado.

En la tabla se muestra que sólo se tienen 10 estados del mar, los que corresponden a la escala Douglas [1] (en la práctica los estados 10 y 11 se presentan en condiciones muy extremas). Los estados del mar 10 y 11 corresponden sólo a la escala Beaufort [1]. En este trabajo se utiliza la escala Beaufor/Douglas, lo que significa que tenemos estados del mar del 0 al 9.

2.3 Algoritmo CFAR.

Como ya se ha mencionado, el proceso de detección de blancos se realiza mediante el algoritmo CFAR, el cual tiene múltiples variantes, diseñadas para distintas condiciones del entorno. El algoritmo CFAR calcula un umbral de comparación (T), basado en la energía del ruido marítimo presente en un grupo de celdas de rango (M) llamadas celdas de referencia (RC) alrededor de una celda de prueba (CUT) como se observa en la figura 2.13. El factor de escalamiento (SF) es obtenido con la probabilidad de falsa alarma (pfa) deseada y el parámetro de simetría o forma (SP) del modelo de ruido utilizado [18]. Múltiples variantes de este algoritmo han sido desarrolladas para mantener en un bajo nivel las falsas alarmas y una alta probabilidad de detección.

El algoritmo base de CFAR es conocido como CA-CFAR (Cell Averaging – CFAR) el cual calcula el umbral de comparación estimando el nivel medio de energía en dos ventanas de celdas de referencia (leading and lagging), como se muestra en la ecuación 2.19.

$$g(x) = mean(f_{lag} + f_{lead}), \qquad (2.19)$$



Figura 2. 13. Variantes de CFAR.

donde f_{lag} es la media de las celdas de rango de la ventana inferior, se calcula con la ecuación 2.20, mientras que f_{lead} es la media de las celdas de rango de la ventana superior, se calcula con la ecuación 2.21.

$$f_{lag}(x_i) = \frac{1}{M/2} \sum_{i=1}^{M/2} x_i , \qquad (2.20)$$

$$f_{lead}(x_i) = \frac{1}{M/2} \sum_{i=M/2}^{M} x_i$$
(2.21)

En ambas ecuaciones M representa el número de celdas empleadas en la ventana de procesamiento.

El factor de escalamiento para esta variante es calculado asumiendo ruido con distribución Gaussiana y se muestra en la ecuación 2.22. CA-CFAR fue diseñado para condiciones de ruido homogéneo (distribución Gaussiana) como el existente en los estados del mar 0 y 1, sin embargo sufre serias limitaciones en presencia de múltiples blancos y condiciones adversas del entorno, con este algoritmo se presentan problemas como enmascaramiento mutuo cuando hay varios blancos cercanos, auto enmascaramiento cuando un blanco ocupa múltiples celdas en la ventana de procesamiento y el problema de enmascaramiento por ruido marítimo debido a transiciones de ruido en el entorno.

$$SF = \sqrt{\ln\left(\frac{1}{p_{fa}}\right)} \tag{2.22}$$

En la gráfica 2.14 se muestra el resultado de un experimento realizado con el algoritmo CFAR básico para resaltar la problemática descrita anteriormente y la solución del enmascaramiento mutuo. En la sección a), se tienen cinco blancos sintéticos con distintas secciones cruzadas de radar (nivel de energía radiado por el blanco en dirección a la antena). En la sección b) se tiene un perfil de rangos de ruido marítimo mezclado con los blancos. En la misma sección se han graficado los umbrales calculados por el algoritmo CFAR. En el primer caso, calculado sin celdas de guarda, lo que ocasiona el problema de auto enmascaramiento y enmascaramiento mutuo de blancos. En el segundo caso se calcula el umbral utilizando 4 celdas de guarda y se hace un acercamiento en las zonas críticas, que es donde ocurre la pérdida de blancos. En el acercamiento se ve el efecto de incluir celdas de guarda en el algoritmo, cuando hay varios blancos cercanos o el blanco ocupa múltiples celdas, al menos una celda logrará pasar el umbral. En la sección c) se muestra la salida de CFAR sin celdas de guarda y se resalta el lugar donde se perdieron los blancos, mientras que en la sección d) se muestra la salida de CFAR con celdas de guarda, se destaca la recuperación de los blancos perdidos en el inciso c). El número de celdas de guarda es uno de los parámetros a configurar en el algoritmo [29].



Figura 2. 14. Celdas de guarda en el algoritmo CFAR.

Múltiples variantes han sido desarrolladas para dar solución a estos problemas, tal es el caso de GO-CFAR (Greatest Of –CFAR) diseñado para resolver el problema del enmascaramiento por transiciones de ruido con ventanas de procesamiento de más de 24 celdas. Esta variante sufre de enmascaramiento mutuo, usa el máximo de las medias de energía de las ventanas de CFAR, calculadas como muestra la ecuación 2.20, 2.21 y 2.23 para calcular el umbral de comparación [29].

$$g(x) = Max(f_{lag}, f_{lead})$$
(2.23)

SO-CFAR (Smallest Of-CFAR) utiliza el mínimo de las medias de energía de las ventanas de CFAR, calculadas como muestra la ecuación 2.20, 221 y 2.24 para calcular el umbral de comparación. Esta variante fue diseñada para resolver el

problema de enmascaramiento mutuo con ventanas de más de 32 celdas, sin embargo, sufre de enmascaramiento por transiciones de ruido en el entorno [29].

$$g(x) = Min(f_{lag}, f_{lead})$$
(2.24)

Algunas variantes utilizan algoritmos de ordenamiento de los datos para estimar el umbral como TM-CFAR (Trimmed mean-CFAR), en esta variante se ordena el valor de las celdas de menor a mayor y se elimina la celda de menor y de mayor valor (x_{min} , x_{max}), utilizando la celda en la posición de la mediana junto con el factor de escala para estimar el umbral de comparación. La función mostrada en 2.25 señala este proceso [30], *sort* representa el ordenamiento de los datos y *mean* la media de esos datos.

$$g(x_i) = mean(sort(f_{lagg}, f_{lead}))$$
(2.25)

OS-CFAR (Order Statistic-CFAR), ordena el valor de las celdas de menor a mayor y seleccionan la k-ésima muestra junto con el factor de escala para calcular el umbral de comparación, [31]. La función mostrada en 2.26 indica que se ordenan los valores de las ventanas de referencia con la función *sort* y de ellas se selecciona con la función *select* la celda en la posición k-ésima. En la ecuación 2.27 se calcula la k-ésima celda empleando el criterio de frecuencia de corte de un filtro, el cual considera que esta se encuentra en un 75% del valor de la señal, en este caso el 75% del número de celdas ordenadas.

$$g(x_i) = select(K^{th}, sort(f_{lag}f_{lead})), \qquad (2.26)$$

$$k^{th} = 0.75M \tag{2.27}$$

En la figura 2.15 se muestra un resumen de los efectos de cada una de las variantes del algoritmo descritas en los párrafos anteriores. Nuevamente en la sección a) se muestran los blancos generados. En la sección b) se tiene el ruido marítimo mezclado con los blancos y los umbrales calculados con cada una de las variantes de CFAR. En las secciones c), d), e), f) y g) se muestra la salida de las variantes CA-CFAR, GO-CFAR, SO-CFAR, TM-CFAR, OS-CFAR respectivamente, en ellas se puede notar la pérdida de blancos e incremento de falsas

detecciones en la variante de CA-CFAR, algunas pérdidas se pueden corregir con la variante GO-CFAR, mientras que las falsas detecciones se disminuyen con la variante SO-CFAR. Las variantes que mejor responden son TM-CFAR y OS-CFAR, debido a que el ruido generado es el equivalente a un mar del 3, por lo que las dos últimas variantes responden de mejor manera, es decir, detectan los blancos y disminuyen las falsas detecciones.

Un aspecto a resaltar es que ninguna de las variantes suprime adecuadamente el ruido marítimo, teniendo excesivas falsas alarmas, ocasionando que la imagen del radar se vea con ruido y dificulte la visualización de los blancos. Para que el algoritmo suprima adecuadamente el ruido se debe elegir un número de celdas de guarda adecuado, el tamaño de la ventana de procesamiento conveniente y calcular el umbral con el parámetro de simetría de la distribución del ruido presente en el entorno. La selección de estos parámetros es parte de la información útil para el esquema KBSP.

En el algoritmo de procesamiento CFAR se han desarrollado trabajos que emplean modelos de ruido con distribución Weibull y K y alguna variante del algoritmo CFAR. En el trabajo de DeumHeller y Lew [32], realizan un procesador de falsas alarmas con poca sensibilidad a los parámetros de la distribución. Emplean una distribución Gamma, realizan una modificación a CFAR para obtener el parámetro de simetría de la distribución y calcular el valor adecuado de umbral. Reportan que sólo es válido para una distribución Gamma con un intervalo en su parámetro de simetría, sugieren el empleo de modelos Gaussianos y modelos con distribución K cuando el parámetro está fuera del intervalo.



Figura 2. 15. Variantes de CFAR y umbrales calculados.

En trabajos recientes se han realizado diseños de variantes de CFAR con enfoques de inteligencia artificial, principalmente filtrado adaptativo y redes neuronales. En este sentido L. Zhao y W. Liu [33] proponen dos variantes de CFAR. AND-CFAR y OR-CFAR, en las cuales realiza la fusión mediante redes neuronales de las variantes CA-CFAR y OS-CFAR, mostrando las mejoras obtenidas en detección, las cuales son superiores a los modelos tradicionales únicamente en el caso de AND-CFAR. Mario y Farina [34] realizan la implementación de un procesador CFAR basado en conocimiento, en el diseño explotan el conocimiento a priori proveniente de un sistema de información geográfica para suprimir posibles áreas de ruido. Algunos trabajos en esta área se encuentran patentados como es el caso del Expert system CFAR [35], el cual utiliza técnicas de inteligencia artificial y fusión de datos para procesar los ecos del entorno y las acciones del usuario experto para seleccionar de un banco de variantes de CFAR y mejorar el desempeño de la detección. Sin embargo, el sistema experto fue desarrollado en software para una estación de trabajo, usando software comercial para manejar la base de conocimiento y la máquina de inferencias, por lo que el sistema no puede ser usado para procesar datos al vuelo, es decir en tiempo real.

Los tres últimos trabajos citados están encaminados a desarrollar procesamiento basado en conocimiento, utilizan en mayor o menor escala el conocimiento a priori del entorno. En el caso de Zhao y Liu utilizan un esquema de fusión basado en red neuronal. En el caso de Mario y Farina el procesador CFAR basado en conocimiento, explota el conocimiento a priori de las zonas de ruido conocidas, obtenidas de un sistema de información geográfica. Finalmente el esquema del CFAR con sistema experto es el más completo, sin embargo, como ya se mencionó, implementa un banco de M procesadores CFAR y un sistema experto que fusiona los resultados del procesamiento, pero la limitación más importante es el hecho de no trabajar en tiempo real por estar implantado en software comercial para una estación de trabajo.

En el trabajo descrito en este documento uno de los experimentos realizados aborda un esquema de fusión de variantes CFAR mediante red neuronal. Se realiza procesamiento basado en conocimiento, obteniendo información del entorno de navegación y el conocimiento del experto para seleccionar la variante adecuada de CFAR. El trabajo expuesto en estas líneas es un esquema basado en conocimiento que obtiene la información del entorno con el mismo sensor, incluyendo el conocimiento del experto y los datos obtenido de la misma cadena de procesamiento para seleccionar la variante adecuada del algoritmo CFAR, todo implantado en una arquitectura reconfigurable que ejecuta el procesamiento al vuelo, es decir en tiempo real. Resumiendo el estudio de las variantes de CFAR para el desarrollo del esquema KBSP, la información útil sobre los parámetros de configuración para usar las variantes de CFAR en distintas condiciones del entorno se muestran en la tabla 2.2. Con esta tabla y la información mostrada en la tabla 1 nos encaminamos a completar la información útil buscada para diseñar el sistema KBSP.

Parámetros	CA	GO	SO	ТМ	OS
М	10-24	10-24	24-32	24-32	24-32
GC	1-4	1-4	1-4	-	-
SP	Ver tabla 2.1				
Pfa	$10^{-3} - 10^{-6}$				
K lag	-	-	-	0.75M	0.75M
K lead	-	-	-	0.75M	0.75M

Tabla 2.2. Parámetros de configuración de variantes de CFAR.

2.4 El conocimiento del experto.

El experto en la operación del radar realiza una inspección visual de las condiciones del entorno y estima de manera aproximada la altura de las olas, con lo cual decide el estado del mar (SS) presente en el entorno. Adicionalmente, realiza una estimación del número de blancos presentes (TN) y la mínima distancia entre ellos para garantizar la detección (MD), estos dos últimos parámetros también pueden ser obtenidos de la cadena tradicional de procesamiento. Con esta información el operador selecciona una variante de CFAR, con la cual realizará la detección hasta que sea configurada otra variante del algoritmo. En los sistemas tradicionales se utiliza una o más variantes del algoritmo CFAR con la mayoría de sus parámetros fijos y sólo algunos pueden ser modificados por el usuario mediante controles de ganancia. Este conocimiento heurístico se incorpora al sistema en forma de reglas para clasificar el estado del mar y seleccionar la variante de CFAR adecuada a las condiciones del entorno, así como seleccionar sus parámetros adecuados. Los principales trabajos que incorporan conocimiento del experto se

discutieron en la sección anterior [12], [34], [35]. En el trabajo de Jylha y Kerminen [36], incorporan el conocimiento sobre la posición de islas y zonas conocidas de ruido, las cuales son omitidas en el procesamiento de CFAR. Este procesamiento lo realizan para un radar aéreo. En este trabajo se utiliza el concepto llamado *clutter maps* que son coordenadas conocida de zonas de ruido para omitirlas del procesamiento, esta información es proporcionada por el experto.

2.5 Cómputo reconfigurable.

Como se mencionó anteriormente, las plataformas de procesamiento de señales son un factor importante en el desarrollo de nuevas generaciones de radar. El cómputo reconfigurable proporciona las técnicas adecuadas para el diseño de tales plataformas. A continuación se describen algunas de sus características principales.

En el espectro de cómputo mostrado en la figura 2.15, se tienen dos métodos convencionales para llevar a cabo el proceso de computación [37]. Por un lado, se tiene procesamiento hardware utilizando circuitos cableados ya sea en ASIC's (Application Specific Integrated Circuit) o bien en una placa de circuito impreso, este método se caracteriza por su rapidez y eficiencia para una aplicación concreta para la cual fue diseñado, el ASIC no puede ser alterado después de la fabricación, restándole flexibilidad. En el otro extremo se tiene procesamiento software ejecutándose sobre microprocesadores con un conjunto de instrucciones. El método incrementa la flexibilidad al poder cambiar la funcionalidad de un sistema pero se reduce la eficiencia debido a la lectura, decodificación y ejecución de la instrucción. Los dispositivos reconfigurables disponen de la eficiencia de procesamiento hardware al explotar el paralelismo en las aplicaciones y un alto grado de flexibilidad al tener arquitecturas adaptables al algoritmo a procesar.



Figura 2. 16. Espectro de cómputo.

La investigación en esta área se ha incrementado notablemente en los últimos años con la introducción del FPGA (Field Programable Gate Array), el nivel de integración de los dispositivos reconfigurables actuales y la posibilidad de modificar su configuración, han permitido diseñar sistemas reconfigurables eficientes para múltiples aplicaciones que utilizan algoritmos adaptativos [38], [39]. En cuanto a implementaciones en FPGA de algoritmos CFAR se tienen los trabajos de Abdullah y Saleh [40], en el cual implementan en un FPGA la variante de TM-CFAR para ser ejecutado en tiempo real, logrando velocidades de 100MHz. Así mismo Magaz y Bencheikh [41] realizan la implementación en FPGA de la variante OS-CFAR. La implementación eficiente de las variables de CFAR en estos trabajos no emplea técnicas basadas en conocimiento, son diseñadas para un modelo de ruido determinado. El aspecto importante en estas implementaciones es la velocidad de procesamiento, lo que permite añadirle bloques adicionales hasta alcanzar un esquema KBSP. En el trabajo desarrollado se llevan a cabo implementaciones eficientes en hardware reconfigurable de variantes de CFAR y módulos de procesamiento y decisión basados en conocimiento para completar un sistema de procesamiento de señales de radar que se ejecute en tiempo real.

2.6 Resumen.

En este capítulo se ha presentado una descripción general del conocimiento del dominio (entorno de navegación) que es de utilidad para realizar el proceso de detección de blancos inmersos en ruido marítimo, por un lado los modelos de ruido y su parámetro de forma asociados a cada estado del mar, se resumen en la tabla 2.1. Las variantes del algoritmo CFAR y sus parámetros que deben ser aplicadas de acuerdo a las condiciones del entorno, resumido en la tabla 2.2. Se presentó el conocimiento del experto que nos ayuda a configurar el algoritmo a emplear. Finalmente se destacó la importancia del hardware disponible para realizar el proceso de computación en tiempo real y sobre todo reconfigurar el algoritmo a medida que cambian las condiciones del entorno de acuerdo a lo planteado en el procesamiento KBSP. Conociendo la información útil para realizar el procesamiento KBSP, en el siguiente capítulo se muestra el esquema diseñado para evaluar un método para detección de blancos basado en conocimiento.

Capítulo 3.

Esquema KBSP para detección de blancos.

En este capítulo se describe el diseño del esquema KBSP para detección de blancos en presencia de ruido marítimo. Como se mencionó en el capítulo 2, el conocimiento del dominio está formado por el ruido marítimo, caracterizado mediante funciones de distribución de probabilidad, asociadas a cada estado del mar mediante su parámetro de simetría. El algoritmo CFAR para supresión de ruido y sus parámetros de configuración, como son el número de celdas de guarda y el tamaño de la ventana de procesamiento, resumidos en la tabla 2.2. El conocimiento del experto sobre el estado del mar, el tipo y cantidad de blancos en el escenario, que permite configurar a CFAR de acuerdo a las condiciones del entorno y una arquitectura que soporte este procesamiento en tiempo real.

Para el diseño KBSP se realiza una revisión del bibliográfica en este tipo de procesamiento de señales. Con esta información se plantea el esquema KBSP que permita alcanzar los objetivos planteados en el capítulo 1. El diseño está enfocado a

proporcionar una solución alternativa al problema de la detección de blancos inmersos en ruido marítimo, para ello se describen en las siguientes secciones las técnicas empleadas para obtener la información necesaria a partir de los ecos de radar.

3.1 Introducción a los esquemas KBSP.

Para mejorar las siguientes generaciones de radar que procesan señales complejas (fase y cuadratura), en tiempo real y en entornos de ruido, DARPA (Defence Advanced Research Projects Agency), recientemente se ha iniciado el proyecto KASSPER (Knowledge Aided Sensor Signal Processing and Expert Reasoning). El objetivo de KASSPER es la utilización de los siguientes conceptos en el procesamiento de las señales de radar: conocimiento asistido, procesamiento adaptativo, tiempo real, arquitecturas de cómputo embebidas de alta eficiencia, entornos multifuente y bases de datos estadísticas [42]. Por otro lado, Simon Haykin en [43] destaca el concepto de radar adaptativo, en donde el objetivo es convertir un radar en un sistema remoto adaptativo, el cual sea capaz de reconocer el entorno y adaptarse a variaciones estadísticas del mismo, planteando el concepto de detección de blancos como un problema de clasificación adaptativa de patrones. Haykin concluye que para tener un radar adaptativo o cognitivo, se requiere del uso de inteligencia artificial, retroalimentación de datos para facilitar el procesamiento inteligente y la preservación de la información proveniente de los ecos de radar.

El esquema diseñado y descrito en el presente documento retoma este concepto para aplicarlo a uno de los bloques de detección, el procesador CFAR, siendo el principal medio de información los ecos de radar con los blancos y el entorno el mar, en sus diferentes estados de acuerdo a la escala de Beaufort/Douglas, por lo que el enfoque de clasificación de patrones se aplica para clasificar el patrón de textura electromagnética reflejado por la superficie marítima. Guerci y Baranoski en [44], proporcionan un resumen de los trabajos que se han realizado en el programa KASSPER, destacando principalmente algoritmos de procesamiento adaptativo tiempo-espacio para radares aéreos. El procesamiento tiempo-espacio también es explorado por Bergin y Teixeira en [45]. Ellos incorporan fuentes de conocimiento tiempo-espacio directamente en la formación del haz para radares aéreos. Básicamente, los autores obtienen un modelo al que le añaden restricciones basadas en modelos de ruido generados con conocimiento a priori del entorno de operación. El objetivo es mejorar el desempeño de los filtros adaptativos para supresión de ruido. Los resultados reportados al aplicar un filtro de este tipo en el modelo KASSPER muestran que sólo se ha tenido éxito con algunos datos experimentales en las bandas de UHF y L.

Capraro y Farina en [46], resaltan los logros alcanzados en el módulo MTI (Moving Target Indicator) utilizado en radares aéreos, incorporando técnicas basadas en conocimiento. Wicks y Rangoswamy proporcionan una revisión general del procesamiento adaptativo tiempo espacio (STAP) por sus siglas en inglés [47], destacando que la clave para este tipo de procesamiento es conocer las características espectrales de la interferencia en un escenario de interés. Para suprimir la interferencia realizan la clasificación del tipo de interferencia presente y para incrementar el conjunto de entrenamiento proponen la utilización de un grupo de antenas que proporcionen información sobre el tipo de interferencia en el entorno.

En el trabajo de Miranda y Baker [48] se discute la utilización de cómputo suave para seleccionar los parámetros de configuración de un radar multifunción con el fin de mejorar su desempeño. Los autores utilizan un conjunto de reglas para imitar el proceso de toma de decisiones humanas en la configuración del radar.

Melvin y Showman [49] proponen una arquitectura para la detección de blancos móviles, en la cual la primera etapa es un filtro de predicción/estimación que incorpora conocimiento a priori de las características del ruido. El objetivo del filtro de predicción/estimación (Knowledge Aided Filter) es remover los componentes predecibles del ruido ayudado por una base de datos de su comportamiento. Los resultados que reportan sugieren un incremento en la probabilidad de detección sobre esquemas tradicionales.

Los trabajos desarrollados en procesamiento de señales basados en conocimiento, reportan la necesidad de un alto poder de cómputo y memoria de almacenamiento. En este sentido French y Suh en [5] evalúan el desempeño de dos arquitecturas de cómputo embebido en donde realizan operaciones típicas de procesamiento. Las dos arquitecturas son Imagine Architecture desarrollada por la Universidad de Stanford, que procesa a 300 MHz y es capaz de realizar 14 GOPS en punto flotante. La segunda es Raw Architecture, desarrollada por el MIT que trabaja a 400MHz y es capaz de realizar 6.4 GOPS en punto flotante. En los resultados reportan que ambas arquitecturas presentaron mejor desempeño que una Power PC G4 la cual es generalmente usada para aplicaciones de procesamiento digital de señales, la cual trabaja a 500MHz y es capaz de realizar 2 GOPS en punto flotante.

Guerci y Baranoski en [44] muestran la primera arquitectura para un radar adaptativo basado en conocimiento operando en tiempo real. Esta arquitectura está basada en múltiples procesadores corriendo en paralelo, en los cuales se ejecutan algoritmos típicos de procesamiento de señales de radar. Es una plataforma de propósito general para evaluar los esquemas propuestos en KASSPER.

De los trabajos analizados anteriormente, podemos notar que el factor común es el empleo de técnicas de inteligencia artificial para mejorar el procesamiento, así mismo se requieren de arquitecturas robustas para ejecutar los esquemas KBSP. En el trabajo que se desarrolla en estas líneas, plantea la estructura y características del sistema basado en conocimiento adecuado para procesar señales de radar como un sistema experto embebido en hardware como el mostrado en la figura 3.1. En la figura se distinguen cuatro bloques principales. *Adquisición del conocimiento*, el cual procesa los ecos o datos crudos del radar, los datos calculados en la cadena de procesamiento tradicional a manera de retroalimentación o proporcionados por el experto como numero de blancos (TN) y distancia entre blancos(MD) y el conocimiento del experto representado por los mapas de ruido que son zonas conocidas con ruido terrestre. *Representación del conocimiento*, en donde se tiene una base de datos de hechos y una base de conocimiento, la cual utiliza principalmente árboles de decisión y estructuras basadas en reglas del tipo IF <condición> THEN <conclusión>. *Maquina de inferencias*, la cual evalúa las reglas almacenadas en la base de conocimiento de acuerdo a los hechos calculados y almacenados en la base de datos de hechos. *CFAR paramétrico* el cual es configurado de acuerdo a la evaluación de las reglas de conocimiento. En la sección 2.3 se discutió que el trabajo desarrollado se enfoca en el procesamiento KBSP, procesando principalmente señales de radar, a diferencia de otros esquemas similares, se emplean distintos modelos de ruido marítimo con diferentes parámetros, asociados a cada estado del mar. En esta sección se detalla el esquema de un sistema experto en donde cada bloque plantado tiene diferencias con los sistemas analizados, tales como los modelos de clasificación de estados del mar y el procesamiento de CFAR basado en energía. En las siguientes secciones se detalla el diseño de cada una de las etapas de la estructura KBSP planteada.



Figura 3. 1. Estructura KBSP para procesamiento de señales de radar.

3.2 Adquisición del conocimiento.

De acuerdo al esquema planteado en la sección anterior, la información que se tiene del entorno es la proporcionada por los ecos de radar, con ella se debe estimar el estado del mar presente para configurar el algoritmo CFAR con los parámetros adecuados de acuerdo con el conocimiento que se tiene del dominio, el cual se resume en las tablas 2.1 y 2.2.

El conocimiento que se tiene sobre los estados del mar es la escala de Beaufort/Douglas, la cual los clasifica en función de la altura de las olas y velocidad del viento. Al digitalizar los ecos de radar se obtiene la distribución espacial de las amplitudes de las olas; el patrón obtenido por el reflejo de las ondas electromagnéticas sobre la superficie del mar puede ser obtenido mediante un análisis de textura. En este sentido, Gangeskar [50] reporta la efectividad de la estadística de segundo orden para aplicaciones donde la distribución espacial de niveles de gris o niveles de energía es importante, tal como en el caso de las señales de radar.

En el área de procesamiento de señales de radar se buscan constantemente métodos de análisis tiempo/escala, tiempo/frecuencia, para mejorar el proceso de detección, reconocimiento y clasificación de blancos. El análisis tiempo/escala, está asociado a la potencia de los ecos recibidos, es decir, la amplitud de la sección cruzada de radar de los blancos o de los picos de las olas. Los perfiles de rango recibidos se agrupan en una matriz para ser convertidos posteriormente a una imagen, por lo que se puede asociar la amplitud de la sección cruzada de radar de los blancos o del gris en la imagen. En el caso de análisis tiempo/frecuencia se puede obtener la energía de los ecos concentrada en el tiempo en el perfil de rangos. En ambos casos existen métodos para realizar el reconocimiento de texturas, en este caso reconocer y clasificar el comportamiento textural desde el punto de vista electromagnético de la superficie marina en diferentes condiciones del entorno. La figura 3.2 muestra los datos de radar reales,

así como un acercamiento a la textura, mientras que la figura 3.3 muestra datos sintéticos. Para el caso del análisis tiempo/escala se usa la matriz de co-ocurrencias de niveles de gris, GLCM por sus siglas en inglés (Grey Level Co-ocurrence Matix) propuesta por Haralik [51] y para el caso del análisis tiempo/frecuencia se usan los filtros de energía propuestos por Laws [52].

El módulo de adquisición del conocimiento tiene dos funciones, la primera es la generación de un modelo para clasificar los estados del mar, el cual se desarrolla como un sistema del tipo *off-line* [53], este modelo forma parte de la base de conocimiento y es discutido más adelante. La segunda función del módulo es la obtención en línea o al vuelo de los hechos del entorno, es decir, los nuevos datos de radar que serán clasificados en alguna categoría de estados del mar. En los siguientes párrafos se describen los métodos para estos procesos.



Figura 3. 2. Datos reales de radar y acercamiento a su textura.



Figura 3. 3. Datos sintéticos y acercamiento a su textura.

3.2.1 Modelo basado en la matriz de concurrencia de niveles de gris para clasificación de textura de estados del mar.

Para clasificar el estado del mar, se calculan siete descriptores estadísticos de textura, derivados de la GLCM, la cual describe la distribución espacial de los datos. La GLCM estima las propiedades relativas a estadísticas de segundo orden de la imagen de radar, considerando la relación espacial entre un píxel de referencia y sus píxeles vecinos. En este caso, el píxel corresponde a una celda del perfil de rangos, esta relación espacial puede ser representada por un vector desplazamiento d = (dx, dy), donde dx, dy representan las coordenadas de la imagen.

La GLCM puede representar desplazamientos en cuatro direcciones llamadas 0° , 45° , 90° , 135° , como se muestra en la figura 3.4. Para este trabajo sólo se

consideran los desplazamientos 0° y 90°, ya que es la dirección en que emite y gira el radar respectivamente, así mismo el vector distancia empleado es d=1, ya que una celda de rango representa la máxima resolución del radar, es decir, la separación mínima entre blancos para detectarlos.



Figura 3. 4. Vecinos del píxel de referencia y direcciones de desplazamiento.

Definimos una imagen de ecos de radar como I(x,y) con $0 \le x \le N-1$ y $0 \le y \le N-1$ con G=256 niveles de gris (datos de amplitud de 8 bits) y N = 512, 128 y 64 pixeles. Los elementos de la GLCM (P_d) se definen como la probabilidad asociada de que 2 pixeles del mismo valor se encuentren separados una distancia den la imagen, esta probabilidad se puede encontrar contando el número de ocurrencias del valor del pixel de referencia con sus vecinos separados una distancia d. La GLCM tiene dimensiones de GxG donde G denota el número de niveles de gris de la imagen y está definida como:

$$P_d(i,j) = \left\{ \left((r,s), (t,v) \right) : I(r,s) = i, I((t,v) = j \right\},$$
(3.1)

donde *d* representa el desplazamiento utilizado, *i,j* son los índices de la GLCM y representan el número de ocurrencias de los niveles de gris de la imagen, *r,s* y *t,v* son índices para direccionar los pixeles de la imagen, con (t,v) = (r+d, s+d).

Los descriptores de textura calculados para una imagen I son los siguientes:

• Homogeneidad.- Es una medida de similitud entre pixeles de toda la GLCM, se obtiene mediante la siguiente ecuación.

$$Homo = \sum_{i,j=0}^{n-1} \frac{P_{i,j}}{1 + |i-j|}$$
(3.2)

 Contraste (Cont).- También llamado suma de varianzas cuadradas, es una medida que representa el nivel de variación entre píxeles de la textura, cuanto mayor es el contraste en la imagen, mayor es el valor de Cont. Se obtiene con la siguiente fórmula.

$$Cont = \sum_{i,j=0}^{n-1} P_{i,j} (i-j)^2$$
(3.4)

• Disimilaridad.- Es una medida similar al contraste, tiene valores altos cuando la región tiene un contraste alto. Se calcula como se indica a continuación.

$$Dis = \sum_{i,j=0}^{n-1} P_{i,j} | i - j |$$
(3.5)

 Entropía (Ent).- Es una medida de la aleatoriedad de la textura, cuanto más suave es esta aleatoriedad, la entropía toma valores bajos. Se obtiene por medio de la siguiente ecuación.

$$Ent = \sum_{i,j=0}^{n-1} -P_{i,j} \ln(P_{i,j}) \to 0 * \ln(0) = 0$$
(3.6)

 Correlación (Cor).- Es una medida de la similitud entre pixeles vecinos dentro de la GLCM, toma valores iguales a 1 cuando los pixeles son iguales, se obtiene mediante la fórmula 3.7.

$$Cor = \sum_{i,j=0}^{n-1} P_{i,j}(\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}})$$
(3.7)

donde $P_{i,j}$ son los componentes de la GLCM, *i*,*j* son los índices de la misma matriz, μ_i es la media y σ_i es la desviación estándar.

 Segundo momento angular (ASM).- Es una medida de la suavidad de la textura, cuanto más suave es la textura mayor valor toma ASM, se obtiene mediante la siguiente ecuación

$$ASM = \sum_{i,j=0}^{n-1} P_{i,j}^{2}$$
(3.8)

 Momento de diferencia inverso (MDI).- Es una medida de contraste en el sentido inverso, es decir cuanto menor es el contraste, mas alto es el valor de MDI, está definido por la ecuación 3.9.

$$MDI = \sum_{i,j=0}^{n-1} \frac{P_{i,j}}{1 + (i-j)^2}$$
(3.9)

En la figura 3.5 se muestra una imagen de datos sintéticos, mientras que en la tabla 3.1 se muestran sus descriptores de textura.



Figura 3. 5. Imagen de radar generada con datos sintéticos.

	Cor	MDI	ASM	DIS	Cont	Ent	Homo
SS5	0.1293	0.6473	0.3534	0.2393	0.2270	0.2509	0.6862

Tabla 3.1 Descriptores de textura para la imagen de la figura 3.5.

Con los descriptores obtenidos de la GLCM se experimentó con distintos modelos para clasificar la textura de los ecos recibidos, probando algoritmos como ID3, C4.5, Naive Bayes y KNN. Esta fase del experimento se realizó utilizando el software WEKA (Waikato Environment for Knowledge Analysis) [54]. Los árboles de decisión mostrados en las figuras 3.6, 3.7 y 3.8 muestran los modelos obtenidos para clasificar la textura de los ecos utilizando datos sintéticos y datos reales respectivamente, los descriptores usados en estos modelos son disimilaridad, segundo momento angular y momento de diferencia inverso, para datos sintéticos; momento de diferencia inverso y correlación para datos reales.



Figura 3. 6. Árboles de decisión generados con WEKA para datos sintéticos.



Figura 3. 7. Árboles de decisión generados con WEKA para datos reales.

Kules generated :	
class ST2 IF : -0.0026747<=corr<=0.0038574 ^ 0.13213<=ndi<=0.13362 ^ 0.0021355<=asm<=0.0021454 ^ 7.0087<=disim<=7.0349 ^ 7	8.475<=cont<=79.074 ^ 6.4012<=ent<=6.408 (7)
class ST3 IF : -0.0038759<=corr<=0.0036278 ^ 0.11164<=ndi<=0.11256 ^ 0.0015312<=asn<=0.0015426 ^ 9.3377<=disin<=9.402 ^ 14	9.8<=cont<=152.09 ^ 6.8042<=ent<=6.8144 (7)
class ST4 IF : -0.0033079<=corr<=0.0010701 ^ 0.12522<=Hdi<=0.12599 ^ 0.0027919<=asH<=0.0028406 ^ 13.852<=disiH<=13.903 ^ 45	53.03<=cont<=462.26 ^ 6.8713<=ent<=6.8791 (7)
class ST5 IF : -0.0026275<=corr<=0.0018231 ^ 0.13331<=ndi<=0.13448 ^ 0.0031338<=asm<=0.0031872 ^ 14.415<=disim<=14.5 ^ 579	.55<=cont<=590.85 ^ 6.8134<=ent<=6.822 (7)
class ST6 IF : -0.0016989<=corr<=0.0034553 ^ 0.13611<=ndi<=0.13784 ^ 0.0034014<=asm<=0.0034564 ^ 14.618<=disin<=14.749 ^ 6	13.14<=cont<=629.72 ^ 6.7878<=ent<=6.7961 (7)
class ST7 IF : -0.0036075<=corr<=0.003826 ^ 0.13963<=ndi<=0.14116 ^ 0.0037527<=asn<=0.0038277 ^ 14.865<=disin<=14.984 ^ 653	2.12<=cont<=663.9 ^ 6.7532<=ent<=6.7657 (7)
class ST8 IF : -6.063E-4<=corr<=0.0037436 ^ 0.14487<=ndi<=0.14664 ^ 0.0043092<=asn<=0.0043962 ^ 15.201<=disin<=15.326 ^ 70'	7.51<=cont<=727.53 ^ 6.702<=ent<=6.7139 (7)
class ST9 IF : -0.0024388<=corr<=0.0031231 ^ 0.15264<=ndi<=0.15449 ^ 0.0051548<=asm<=0.0053334 ^ 15.523<=disim<=15.794 ^ 7/	68.66<=cont<=804.45 ^ 6.632<=ent<=6.6563 (7)
class ST10 IF : -0.0037932<=corr<=0.0021598 ^ 0.16442<=ndi<=0.16601 ^ 0.0068534<=asn<=0.0070268 ^ 15.97<=disin<=16.297 ^ 8	57.52<=cont<=894.93 ^ 6.517<=ent<=6.5425 (7)
class ST11 IF : -0.0020593<=corr<=0.0026172 ^ 0.18337<=ndi<=0.18512 ^ 0.010122<=asn<=0.010389 ^ 16.718<=disin<=16.86 ^ 102'	2.6<=cont<=1039.6 ^ 6.3542<=ent<=6.3692 (7)

Figura 3. 8. Modelo para clasificación de estados del mar basado en reglas.

3.2.2 Modelo basado en descriptores de energía para clasificación de textura de estados del mar.

Otra información que proporcionan los ecos de radar es la energía de los blancos o de las olas del mar, por lo que se realiza también el análisis de textura basado en los descriptores de energía de Laws [52]. El autor desarrolló un conjunto de filtros de energía para obtener un mapa de concentración de energía en una señal o en una imagen. Estos vectores son: media ponderada (level), gradiente (Edge), forma (spots) y rugosidad (ripple).

L3 = (1, 2, 1) Level

E3 = (-1, 0, 1) Edge

S3 = (-1, 2, -1) Spot

Convolucionando cada vector consigo mismo como se muestra en la ecuación 3.10, se pueden obtener vectores de 5 o más coeficientes:

$$L3*L3 = L5 (3.10)$$

donde * denota la operación de convolución

L5 = (1, 4, 6, 4, 1) Level E5 = (-1, -2, 0, 2, 1) Edge S5 = (-1, 0, 2, 0, -1) Spot R5 = (1, -4, 6, -4, 1) Ripple

Este conjunto de vectores es utilizado para encontrar la textura de una señal basado en descriptores de energía. Si se convoluciona cada vector con otro vector transpuesto, se obtiene un conjunto de máscaras de dos dimensiones para encontrar la textura en imágenes. Este proceso se calcula como se indica en la ecuación 3.11, mientras que en la figura 3.9 se muestran algunas de las máscaras de energía utilizadas.

$$L5 * E5' = L5E5 \tag{3.11}$$

Un total de 25 máscaras de convolución son obtenidas durante este proceso, cada una de las máscaras es convolucionada con cada una de las imágenes del radar obtenidas para cada estado del mar. Posteriormente se realiza un proceso de obtención de un mapa de energía en una vecindad de 15x15 celdas de rango, utilizando un proceso de enventanado, como se muestra la ecuación 3.12. Posteriormente se normalizan y se combinan características similares como se muestra en la ecuación 3.13, obteniendo un total de 14 descriptores de energía.

$$New(x, y) = \sum_{i=-71}^{7} \sum_{j=-71}^{7} |Old(x+1, y+j)|$$
(3.12)

$$Descriptor E5L5 = E5L5 + L5E5 \tag{3.13}$$

donde *New(x,y)* es un punto de la imagen al cual se le obtendrá su mapa de energía, *x,y* son las coordenadas de la imagen, *i,j* son los índices de la ventana para la obtención del mapa de energía.

En la figura 3.10 se muestra un árbol de decisiones obtenido con WEKA.



Figura 3. 9. Máscaras de convolución de Laws.



Figura 3. 10. Árbol de decisión basado en descriptores de energía.

El proceso de clasificación mediante descriptores de energía utilizando kernels 2D, utiliza demasiadas operaciones y requiere de una gran cantidad de memoria de almacenamiento. De una imagen se generan 25 más, por lo que para reducir el tiempo de procesamiento y el tamaño de almacenamiento se realizaron experimentos con los filtros en 1D aplicados sobre el perfil de rangos, la distribución de energía se obtiene con la ecuación 3.14.

$$New(x) = \sum_{i=-7}^{7} Old(x+i)$$
 (3.14)

donde New(x) es el nuevo valor del perfil de rangos, x es el índice del perfil, i es el índice de la ventana del mapa de energía, *Old(x)* es el antiguo valor del perfil sin mapa de energía. La figura 3.11 muestra el árbol de decisiones obtenido.



Figura 3. 11. Árbol de decisiones obtenido con filtros 1D.

Utilizando el análisis de energía sobre un perfil de rangos se puede obtener información sobre la presencia o no de blancos en ese perfil, la figura 3.12 muestra los diferentes niveles de energía en distintos perfiles con y sin blancos.



Figura 3. 12. Niveles de energía en perfiles a) sin blancos, b), c) con blancos.

De la figura 3.12 se puede notar que a medida que se incrementa el número de blancos en el perfil de rangos, los niveles de energía se incrementan, esto es debido a que los blancos conservan más energía a lo largo del tiempo, mientras que el ruido marítimo puede tener mayor amplitud pero de poca duración.

Los descriptores de textura obtenidos tanto en la GLCM como de energía, permiten obtener el modelo de clasificación basado en árboles de decisión, así como un conjunto de datos de entrenamiento para nuevos escenarios

3.3 Representación del conocimiento.

Como ya se ha mencionado, el conocimiento del dominio se almacena en forma de árboles de decisión y reglas del tipo IF-THEN. En este sentido el módulo de representación del conocimiento de la figura 3.1 se divide en dos bloques, uno para almacenar el conocimiento obtenido para la clasificación de los estados del mar y el segundo para almacenar los descriptores de textura que servirán como base de hechos para clasificar los nuevos datos provenientes del radar. Ambos bloques se describen a continuación.

3.3.1 Descriptores de textura y conocimiento del experto como base de hechos.

En el esquema planteado se utiliza una base de datos de hechos para ser comparados con el conocimiento almacenado y de esta manera poder clasificar nuevos conjuntos de datos, los cuales cambian con las condiciones del entorno. La base de hechos está formada por los descriptores de textura obtenidos mediante la GLCM y los filtros de energía. Forman parte también de la base de hechos el número de blancos en el escenario y la mínima distancia entre blancos, estos hechos pueden ser proporcionados por el experto o bien pueden ser alimentados como realimentación del mismo sistema, tal como se indica en el diagrama de la figura 1.2. Hechos adicionales pueden ser proporcionados por el escenario que no deben ser procesadas, ya que el ruido generado en ella es de otro tipo, como en el caso de la costa. En la figura 3.13 se muestra el bloque de almacenamiento de hechos, en él se muestran sólo los descriptores que mejor clasificaron la textura, ésto se explica a detalle en el capítulo 4.

ASM
MDI
DISS
R5L5
L5R5
R5R5
TD .
TN
CM
GLCM Rules
ENERGY Rules
HEURISTIC Rules
GLCM Rules ENERGY Rules HEURISTIC Rules

Figura 3. 13. Mapa de memoria de la base de hechos.

3.3.2 Árboles de decisión y reglas de producción como base de conocimiento.

El segundo bloque de la representación del conocimiento es la base de conocimientos, almacenado como un conjunto de reglas y árboles de decisión. El conocimiento generado es dividido en dos grupos de reglas, el primero de ellos es para clasificar el ruido marítimo y conocer el estado del mar actual, estas reglas son extraídas de los árboles de decisión. La figura 3.14 muestra un árbol de decisión obtenido con WEKA para clasificar los estados del mar usando datos sintéticos, mientras que la figura 3.15 muestra el árbol de decisión para clasificar estados del mar usando datos reales, en las mismas figuras se muestra el conjunto de reglas generadas a partir del árbol de decisión, en donde C representa la condición a evaluar.



Figura 3. 14. Reglas y árbol de decisión para clasificar estados del mar con datos



Figura 3. 15. Reglas y árbol de decisión para clasificar estados del mar con datos

reales y descriptores de energía.
El segundo grupo de reglas concentra el conocimiento del dominio resumido en las tablas 2.1 y 2.2, así como el conocimiento del experto, estas reglas son encadenadas a las reglas para clasificación de estados del mar. Las condiciones de estas reglas son: el estado del mar, el número de blancos en el entorno y la distancia entre blancos, la conclusión es la variante de CFAR a utilizar y los parámetros para configurarlo, principalmente el número de celdas de guarda, el número de celdas de referencia y el parámetro de forma. A continuación se muestran algunas de estas reglas:

Conjunto de reglas 1

IF C1 and C2 and C3 THEN CA-CFAR, GC=2, RC=16, SP= Gaussiano IF C1 and C2 and C4 THEN CA-CFAR, GC=2, RC=32, SP= Gaussiano IF C1 and C5 and C3 THEN CA-CFAR, GC=4, RC=16, SP= Gaussiano IF C1 and C5 and C4 THEN CA-CFAR, GC=4, RC=32, SP= Gaussiano Conjunto de reglas 2 IF C6 and C2 and C3 THEN GO-CFAR, GC=2, RC=24, SP=0.9 Weibull

IF C6 and C5 and C4 THEN SO-CFAR, GC=4, RC=32, SP=0.9 Weibull Conjunto de reglas 3

IF C7 and C2 and C8 THEN TM-CFAR, GC=2, RC=32, SP=2.0, W, K1=22

IF C9 and C2 and C8 THEN OS-CFAR, GC=2, RC=32, SP=0.7 K, K1=22

donde

Condiciones: C1=SS1, C2=distancia>MD, C3= TN<2, C4= TN>2, C5= distancia<MD, C6=SS3, C7=SS4, C8= TN<5, C9=SS5.

El conjunto de reglas 1 corresponde a un estado del mar clasificado como 1 (C1), y con blancos presentes en el entorno en distintas condiciones, por lo que la variante a seleccionar es CA-CFAR, sin embargo, la configuración de sus parámetros es diferente debido a los blancos presentes en el entorno. En el segundo conjunto de reglas, el antecedente contiene el estado del mar clasificado como 3

(C6) y algunos blancos en el entorno, el consecuente es la selección de GO-CFAR, con el modelo de ruido Weibull y los parámetros de configuración de acuerdo a las condiciones del antecedente. Finalmente, en el tercer grupo de reglas se muestran clasificaciones de estados del mar 4 y 5, con su consecuente selección de variantes de CFAR y sus parámetros de configuración.

3.4 Máquina de inferencias.

De acuerdo al diagrama de la figura 3.1 el motor de inferencias se encuentra dividido en dos secciones encadenadas, la primera de ellas evalúa el conjunto de reglas que clasifican los estados del mar, las cuales se muestran en las figuras 3.14 y 3.15. El resultado de esta primera clasificación, activa las reglas adecuadas en la segunda sección para la selección de parámetros de configuración del algoritmo CFAR. El motor de inferencias, siempre opera bajo la estrategia de encadenamiento hacia adelante, es decir, el antecedente genera su respectivo consecuente y este a su vez activa la siguiente regla encadenada.

3.5 CFAR paramétrico.

Para completar el esquema planteado en la figura 3.1 se llevaron acabo distintas implementaciones del algoritmo CFAR, una de ellas es la implementación de un esquema de fusión y el otro la implementación de un CFAR paramétrico. A continuación se describen ambas implementaciones.

El esquema de fusión consiste en una fusión de salidas de variantes de CFAR. La figura 3.16 muestra el esquema utilizado. La implementación del centro

de fusión se realizó con una red neuronal del tipo perceptrón de una sola capa, las reglas de fusión usadas son las mostradas en las ecuaciones 3.15-3.17.



Figura 3. 16. Fusión de variantes de CFAR.

Cuando la salida de CA-CFAR es uno puede ser un blanco con falsa alarma, entonces se utilizan las salidas de GO-CFAR y SO-CFAR para decidir si es un blanco o una falsa alarma, como se muestra en la ecuación 3.15.

$$ANDrule1 = CA-CFAR (GO-CFAR+SO-CFAR)$$
(3.15)

Cuando la salida de CA-CFAR es 0 es un blanco perdido, entonces se recupera con las salidas de GO-CFAR y SO-CFAR, como se muestra en la ecuación 3.16.

$$ANDrule2=GO-CFAR(SO-CFAR)$$
(3.16)

La regla OR, mostrada en la ecuación 3.17 recobra todos los blancos perdidos pero incrementa las falsas alarmas.

$$ORrule=GO-CFAR+SO-CFAR$$
(3.17)

La tabla 3.2 resume el comportamiento de las reglas de fusión.

CA	GO	SO	AND-rule	OR-rule	
0	0	0	0	0	
0	0	1	0	1	
0	1	0	0	1	
0	1	1	1	1	
1	0	0	0	0	
1	0	1	1	1	
1	1	0	1	1	
1	1	1	1	1	

Tabla 3.2 Comportamiento de las reglas de fusión.

El esquema de fusión de datos con distintos algoritmos CFAR requiere de un alto costo computacional, por lo que llevarlo a una arquitectura representará un alto consumo de área, debido a que los parámetros del algoritmo dependen de las condiciones del entorno, las cuales cambian constantemente. Otra opción es un algoritmo CFAR parametrizable, en donde sus parámetros de entrada son el tipo de CFAR, el número de celdas de referencia, el número de celdas de guarda, el parámetro de forma y el k-ésimo valor de la celda. Un algoritmo parametrizable permite implementarse en una arquitectura reconfigurable con poca área utilizada. La implementación del bloque CFAR paramétrico y los bloques descritos en las secciones anteriores se discuten en el capítulo cinco.

3.5 Filtro de energía.

Los filtros de energía propuestos por Laws y discutidos anteriormente pueden ser aplicados a señales de radar como el perfil de rangos, y como se vió anteriormente el nivel de energía aumenta con la presencia de blancos, por lo que se puede incluir un bloque de filtrado previo al procesador CFAR como parte de la cadena tradicional de procesamiento. En particular el filtro de gradiente enfatiza la presencia de los blancos, ya que es un filtro de orden 5, cuyos coeficientes se encuentran confinados en una ventana de Hamming concentrada en el lóbulo principal de la señal de radar [55]. El filtro es diseñado como un filtro FIR seguido de un generador del mapa de energía, con lo cual se enfatizan los blancos presentes, permitiendo que el algoritmo CFAR sea más eficiente. En la figura 3.17 se muestra la estructura seleccionada para el filtro, mientras que en la figura 3.18 se muestra el filtro integrado al esquema.



Figura 3. 17. Estructura del filtro FIR de energía.



Figura 3. 18. Esquema KBSP más el filtro de energía.

La primera parte del filtro consiste en obtener la convolución Y(n) de la máscara con la señal de radar X(n), como se muestra en la ecuación 3.18. Posteriormente se calcula el mapa de energía $Y_{new}(n)$ con la fórmula mostrada en la ecuación 3.19, que es la ecuación propuesta por Laws para procesamiento de una dimensión. Los coeficientes del filtro se deben calcular adaptativamente de acuerdo a las condiciones del entorno, ésto se realiza mediante la ecuación 3.20, que es el algoritmo de raíz cuadrada media para minimizar el error del sistema de realimentación planteado y mostrado en la figura 3.19. La variable η es un parámetro de ajuste que depende del nivel de potencia de los ecos de radar y es usado para ajustar por primera vez los parámetros. El error e(n) se calcula mediante la fórmula mostrada en 3.21, donde T(n) es la salida del filtro de energía con el procesamiento CFAR. SC(n) representa el estado del mar presente.

$$Y[n] = E[n] \otimes X[n], \qquad (3.18)$$

$$Y_{new}[n] = \sum_{i=-7}^{i=7} |Y_{old}[n+i]|, \qquad (3.19)$$

$$C_{i}[n+1] = C_{i}[n] + \eta e[n] X[n], \qquad (3.20)$$

$$e[n] = T[n] - SC[n]$$
(3.21)

donde \otimes representa la convolución, *n* es el tamaño del perfil de rangos e *i* es el índice de los coeficientes.



Figura 3. 19. Filtrado adaptativo de energía y CFAR.

3.6 Resumen.

En el presente capítulo se describió cada uno de los módulos que conforman el esquema utilizado para procesar la señal de radar mediante técnicas basadas en conocimiento. Se puede resumir el esquema en dos secciones principales un sistema del tipo *off line* que permite construir un modelo de clasificación de estados del mar basado en descriptores de textura proporcionados por la GLCM y energía. Este clasificador forma parte de la base de conocimientos del sistema experto, al mismo tiempo nos permite obtener los nuevos descriptores de la señal de radar (módulo de adquisición del conocimiento) para realizar nuevas clasificaciones. En la segunda sección se analizaron los módulos de representación del conocimiento en donde se mostró parte de las reglas utilizados tanto para la clasificación de estados del mar como para la selección de los parámetros de CFAR, reglas utilizadas por la máquina de inferencias. Finalmente se mostró el esquema de CFAR paramétrico empleado y el filtro de energía añadido a la cadena tradicional de procesamiento. En el capítulo siguiente se detallan los experimentos realizados en cada uno de los módulos y se obtiene el esquema global que será llevado a la implementación en hardware.

Capítulo 4.

Resultados experimentales del esquema KBSP.

En el presente capítulo se discuten los resultados obtenidos en los experimentos realizados para la obtención del esquema KBSP planteado en el capítulo anterior. Se analizan los resultados obtenidos en los descriptores de textura, así como los modelos obtenidos en WEKA para seleccionar aquellos que permitan clasificar la textura del mar con el mayor porcentaje. Se discuten los resultados obtenidos en la implementación del sistema experto implementado en software, así como el análisis de resultados del algoritmo CFAR parametrizable y el filtro de energía.

4.1 Conjunto de datos experimentales.

En los capítulos anteriores se ha discutido sobre las soluciones propuestas en la literatura para el problema de la detección de blancos inmersos en ruido. Sin embargo, muchas de estas soluciones utilizan un modelo de ruido para todos los estados del mar. Otros presentan resultados obtenidos únicamente con datos simulados en las que se omiten detalles del comportamiento real de los radares. En este sentido, en el presente trabajo se atacan tanto el aspecto de simulación, como el de datos reales, la generación y obtención de estos datos se describen en los siguientes párrafos.

Como ya se mencionó, los datos utilizados para los experimentos realizados en este trabajo son de dos tipos: datos sintéticos y datos reales. Los datos reales fueron adquiridos por un radar comercial que presenta las características mostradas en la tabla 4.1. El perfil de rangos fue digitalizado a una frecuencia de muestreo de 100 MHz, con lo cual se pueden obtener varias muestras para la longitud de pulso emitido, el convertidor analógico digital usado tiene una resolución de 8 bits. Una imagen de radar está formada por una matriz de 4096 perfiles de rango con 4096 celdas en cada perfil para su máximo alcance. Las imágenes utilizadas para este trabajo se extrajeron de una imagen de radar y se utilizaron imágenes de tamaño 512x 512, 256x 256 y 128x128, representativas de ruido marítimo, correspondiente a estados del mar del 1 al 4.

Característica	Valor
Marca KelvinHughes	Modelo 5000A
Banda	Х
Frecuencia	9 GHz
Polarización	Horizontal
Ganancia de la antena	30dB
Velocidad de rotación de la antena	24 rpm
Figura de ruido	3 dB
Longitud del pulso	0.07µseg
Frecuencia de Repetición del Pulso	3 KHz
Alcance	96mn

Tabla 4.1 Características de radar comercial.

Los datos simulados son generados con las funciones de distribución discutidas en el capítulo 2, para lo cual se desarrollaron un conjunto de herramientas en C y Matlab [56]. Los datos fueron generados para tener las mismas características que los datos reales, esto es 8 bits de resolución o 256 niveles de gris y media equivalente a ruido térmico de 316mV. Se generaron perfiles de rango con número de celdas equivalentes a los alcances de un radar comercial, es decir, una celda por cada pulso emitido, equivale a una resolución de 10 metros. Para la determinación de la amplitud se utilizó la fórmula de ganancia de la antena, mostrada en la ecuación 4.1 [57], mientras que para la sección cruzada de radar se utilizó la máxima resolución del radar. En la figura 4.1a se muestran 6 blancos con amplitudes de 30, 25, 20, 15, 10 y 5 dB con 60, 50, 40, 30, 20 y 10 metros de sección cruzada respectivamente, mientras que en la figura 4.1 b se muestra un perfil de rangos de mar del 4 con una media igual al ruido térmico de 316 mV, en la figura 4.1c se muestran los blancos y el ruido mezclados.

La fórmula de ganancia de la antena es:

$$G = 20\log\frac{V_{out}}{V_{in}},\tag{4.1}$$

donde G es la ganancia, *Vout* y *Vin* son los voltajes de salida y entrada respectivamente, por lo que el voltaje de salida buscado para la amplitud de los blancos generados se calcula como se muestra en la ecuación 4.2.

$$V_{out} = 10^{\frac{dB}{20}} * V_{in}, \qquad (4.2)$$

donde *dB* es la amplitud en decibles deseada para el blanco y *Vin* es el voltaje de entrada, correspondiente al ruido térmico del radar.



Figura 4.1. Datos sintéticos, a) blancos, b) ruido marítimo y c) blancos+ ruido.

Con los perfiles de rango se generaron imágenes de 512 x 512, 256x256 y 128x 128. Para los experimentos se cuenta con un total de 100 imágenes sintéticas y 40 imágenes reales.

4.2 Esquema basado en conocimiento.

En esta sección se detallan los experimentos realizados para seleccionar los descriptores de textura que mejor clasifican los estados del mar, mismos que servirán como nuevos hechos que formarán la base de hechos, así como el conjunto

de reglas que serán aplicadas en el proceso de clasificación y selección de parámetros mediante la máquina de inferencias.

4.2.1 Descriptores GLCM y energía.

En los experimentos realizados para obtener los descriptores de textura basados en la GLCM se desarrollaron un conjunto de herramientas en Matlab y C. Con ellas se calculó la GLCM para cada una de las imágenes de radar, tanto sintéticas como reales. En el caso de las imágenes sintéticas se tienen 10 imágenes para cada estado del mar del 2 al 11, los estados del mar cero y uno no se simularon, ya que el modelo Gaussiano funciona adecuadamente en estas condiciones. El siguiente paso en el experimento es obtener los descriptores tal como se describió en el capítulo 3. Los descriptores para datos sintéticos se muestran en la figura 4.2.



Figura 4. 2. Descriptores de textura con GLCM para datos sintéticos.

En la figura 4.2 se muestran los descriptores de textura para un conjunto de estados del mar. En esta imagen se muestra la separabilidad de los valores dinámicos de los descriptores. En el caso de la correlación se observa que no puede ser utilizada para la clasificación, ya que no se puede separar esta característica en distintos estados del mar, debido a que la correlación sólo muestra la relación entre píxeles adyacentes. Otros descriptores que pueden ser utilizados pero que pudieran causar errores en la clasificación son entropía y homogeneidad, ya que muestran valores cruzados en algunos estados del mar. Los descriptores más adecuados para la clasificación son Momento de diferencia inverso (MDI), Disimilaridad (Dis) y Segundo momento angular (ASM). Los descriptores para datos reales se muestran en la figura 4.3.



Figura 4. 3. Descriptores de textura con GLCM para datos reales.

Con los datos reales se puede corroborar que los descriptores adecuados para los datos sintéticos, también lo son para datos reales y aquellos como la correlación y la entropía dificultarían la clasificación ya que sus valores se entrecruzan en distintos estados del mar. Los descriptores de textura se compararon para todo el conjunto de imágenes de radar disponible, es decir 100 imágenes sintéticas y 40 imágenes reales.

En el caso de los descriptores de energía, se convolucionaron cada una de las 14 máscaras con las imágenes de radar. En el siguiente paso, para cada imagen se calculó el mapa de energía de acuerdo a lo descrito en el capítulo 3, obteniendo finalmente el promedio de energía de la imagen. En la figura 4.4 se muestran los descriptores de energía obtenidos para datos sintéticos, donde se puede observar que la energía es la más adecuada para separar los estados del mar. Sólo algunos descriptores presentan una ventana muy estrecha para realizar la separación, tal es el caso de la máscara S5S5, ya que esta máscara fue creada para resaltar puntos de concentración de energía (spot), lo cual es escaso en un entorno marítimo. En cambio, la máscara R5L5 (roughtness, level) es la más apropiada para clasificar la textura del mar, ya que esta máscara se encarga de filtrar la rugosidad de la imagen, combinada con los bordes o media ponderada, lo cual es la principal característica de la textura marítima y de los blancos existentes en el entorno. Las figuras 4.5 y 4.6 muestran un acercamiento a los descriptores calculados con R5L5 y S5S5 respectivamente. En la figura 4.6 se observa que la mayoría de los estados del mar arrojan valores entre 0.05 y 0.06, lo cual es un rango estrecho, sin embargo, se puede realizar la clasificación utilizando este descriptor, como se verá más adelante.



Figura 4. 4. Descriptores de textura con Energía para datos sintéticos.



Figura 4.5. Acercamiento al descriptor R5L5 para datos sintéticos.



Figura 4. 6. Acercamiento al descriptor S5S5 para datos sintéticos.

Los descriptores para datos reales, estados del mar 1 al 4, se muestran en la figura 4.7. Podemos notar en las gráficas que los descriptores de energía son los más adecuados para la clasificación, en este caso el descriptor R5L5 sigue siendo adecuado para la clasificación, sin embargo, el peor descriptor se encuentra en S5L5. Nuevamente los descriptores menos adecuados son aquellos que convolucionan la imagen con alguna máscara que tenga el descriptor para resaltar puntos (spot), lo cual no forma parte de la textura del ruido marítimo. En las figuras 4.8 y 4.9 se realiza un acercamiento al mejor y peor descriptor, los cuales son R5L5 y S5L5 respectivamente.



Figura 4. 7. Descriptores de textura con energía para datos reales.



Figura 4. 8. Acercamiento al descriptor R5L5 para datos reales.



Figura 4.9. Acercamiento al descriptor S5L5 para datos reales.

4.2.2 Modelos de clasificación

En esta parte del experimento se utilizaron los descriptores calculados anteriormente tanto de GLCM como de energía en un conjunto de 70 imágenes sintéticas y 21 reales, con lo cual se generó un conjunto de datos de entrenamiento para extraer un modelo de clasificación basado en árboles de decisión y reglas. Utilizando los clasificadores proporcionados por WEKA, principalmente modelos basado en árboles de decisión como el j48 o C 4.5 y uno basado en reglas como el K-NN [54]. Para la construcción del modelo se utilizó la estrategia de ten –fold cross validation. En la figura 4.10 se muestra un árbol de decisión obtenido para datos sintéticos y GLCM. Se puede observar que la poda del árbol quedó con el segundo momento angular (ASM) y el momento de diferencia inverso (MDI) para realizar la

clasificación. En la figura 4.11 se muestra el árbol obtenido para datos reales y GLCM. Se puede notar que en este caso sólo se utilizó el momento de diferencia inverso (MDI) y la disimilaridad (Dis) para realizar la clasificación. En las figuras 4.12 y 4.13 se muestra la matriz de confusión obtenida para los datos de entrenamiento del modelo, en donde podemos notar que la clasificación de datos sintéticos utilizando ASM y MDI alcanza un 87 %; incluyendo descriptores como la disimilaridad se pueden alcanzar porcentajes del 90%. Mientras que para datos reales se alcanza sólo un 80%, de igual manera combinado ASM se puede mejorar el porcentaje de clasificación.

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2 Relation: training Instances: 70 Attributes: 7 corr mdi asm disim cont ent class Test mode: 10-fold cross-validation === Classifier model (full training set) === J48 pruned tree asm <= 0.003456 | asm <= 0.002145 | asm <= 0.001543: ST3 (7.0) asm > 0.001543: ST2 (7.0) 1 1 asm > 0.002145 1 | asm <= 0.002841: ST4 (7.0) 1 asm > 0.002841 | | asm <= 0.003187: ST5 (7.0) 1 1 Ι 1 asm > 0.003187: ST6 (7.0) asm > 0.003456 asm <= 0.004396 | asm <= 0.003828: ST7 (7.0) 1 asm > 0.003828: ST8 (7.0) asm > 0.004396 | mdi <= 0.15449: ST9 (7.0) Т 1 mdi > 0.15449| mdi <= 0.16601: ST10 (7.0) 1 mdi > 0.16601: ST11 (7.0) Number of Leaves : 10

Figura 4. 10. Árbol de decisión para clasificación de estados del mar de datos sintéticos

basado en la GLCM.

```
=== Run information ===
             weka.classifiers.trees.J48 -C 0.25 -M 2
Scheme:
Relation:
             training
Instances:
             21
             8
Attributes:
             corr
             mdi
             asm
             disim
             cont
             ent
             homo
             class
Test mode:
             10-fold cross-validation
=== Classifier model (full training set) ===
J48 pruned tree
_____
mdi <= 0.19744: ST1 (10.0)
mdi > 0.19744
| disim<= 0.043778: ST2 (5.0)
| disim> 0.043778
  | disim <= 0.37005: ST3 (3.0)
1
| | disim> 0.37005: ST4 (3.0)
Number of Leaves :
                      4
Size of the tree : 7
```

Figura 4. 11. Árbol de decisión para clasificación de estados del mar de datos reales

```
basado en la GLCM.
```

```
=== Stratified cross-validation ===
=== Summary ===
                                        61
Incorrectly Classified Instances
Kappa statistic
Correctly Classified Instances
                                                          87.1429 %
                                         9
                                                           12.8571 %
                                          0.8571
Kappa statistic
Mean absolute error
                                          0.0257
Root mean squared error
Relative absolute error
                                          0.1604
                                        14.2208 %
                                   53.1983 %
-
Root relative squared error
Total Number of Instances
Total Number of Instances
                                         70
```

=== Confusion Matrix ===

a	b	С	d	е	f	g	h	i	j		<-		classified	as
6	0	1	0	0	0	0	0	0	0	Ι	а	=	ST2	
1	6	0	0	0	0	0	0	0	0	Ι	b	=	ST3	
0	0	6	1	0	0	0	0	0	0	Τ	С	=	ST4	
0	0	0	6	1	0	0	0	0	0	Ι	d	=	ST5	
0	0	0	0	6	1	0	0	0	0	Ι	е	=	ST6	
0	0	0	0	0	6	1	0	0	0	Ι	f	=	ST7	
0	0	0	0	0	0	6	1	0	0	Ι	g	=	ST8	
0	0	0	0	0	0	0	6	1	0	Ι	h	=	ST9	
0	0	0	0	0	0	0	0	6	1	Ι	i	=	ST10	
0	0	0	0	0	0	0	0	0	7	Ι	j	=	ST11	

Figura 4. 12. Matriz de confusión para clasificación de estados del mar de datos

sintéticos basado en la GLCM.

=== Stratified cross-validation ===									
=== Summary ===									
Commentary Changeleters Transmost	17	00 0504 0							
Correctly classified instances	17	80.9524 %							
Incorrectly Classified Instances	4	19.0476 %							
Kappa statistic	0.7228								
Mean absolute error	0.0952								
Root mean squared error	0.3086								
Relative absolute error	27.1241 %								
Root relative squared error	73.4184 %								
Total Number of Instances	21								
=== Confusion Matrix ===									
a b c d < classified as									
9100 a = ST1									
0 4 1 0 b = ST2									
0 1 1 1 c = ST3									
0 0 0 3 d = ST4									
,									

Figura 4. 13. Matriz de confusión para clasificación de estados del mar de datos reales

basado en la GLCM.

Para probar los modelos se utilizaron un conjunto de datos de prueba, consistiendo en 30 imágenes con datos sintéticos y 10 con datos reales. Las matrices de confusión obtenidas en esta fase del experimento se muestran en las figuras 4.14 y 4.15. En las figuras se puede observar que los porcentajes de clasificación se mantuvieron en 87% para datos sintéticos y de 90% para datos reales

=== Re-evaluation on test set === User supplied test set Relation: test 30 Instances: Attributes: 7 === Summary === 26 Correctly Classified Instances 86.6667 % Incorrectly Classified Instances 4 13.3333 % Kappa statistic 0.8519 Mean absolute error 0.0267 0.1633 Root mean squared error Relative absolute error 14.8148 % Root relative squared error 54.4331 % Total Number of Instances 30 === Confusion Matrix === abcdefghij <-- classified as 102000000|a = ST20 3 0 0 0 0 0 0 0 0 | b = ST30 0 1 2 0 0 0 0 0 0 | c = ST40 0 0 3 0 0 0 0 0 0 0 | d = ST50 0 0 0 3 0 0 0 0 0 | e = ST60000030000|f=ST7 0 0 0 0 0 0 3 0 0 0 | q = ST80 0 0 0 0 0 0 3 0 0 | h = ST90 0 0 0 0 0 0 0 3 0 | i = ST10 0000000003|j=ST11

Figura 4. 14. Matriz de confusión para el conjunto de prueba datos sintéticos.

```
=== Re-evaluation on test set ===
User supplied test set
Relation:
             test
              10
Instances:
Attributes:
              8
=== Summary ===
Correctly Classified Instances
                                         9
                                                          90
                                                                  ÷
Incorrectly Classified Instances
                                          1
                                                          10
                                                                  ÷
Kappa statistic
                                         0.8611
Mean absolute error
                                         0.05
Root mean squared error
                                         0.2236
                                        13.7255 %
Relative absolute error
                                        52.6152 %
Root relative squared error
Total Number of Instances
                                         10
=== Confusion Matrix ===
 abcd
           <-- classified as
 4 0 0 0 | a = ST1
 0 1 1 0 | b = ST2
 0 0 2 0 | c = ST3
 0 0 0 2 | d = ST4
```

Figura 4. 15. Matriz de confusión para el conjunto de prueba datos reales.

Aquellas instancias mal clasificadas fueron ubicadas en la posición contigua, es decir, la diferencia entre la posición real de la instancia y la asignada es de un estado, por lo que el error que se presente al final del ciclo de detección será de un estado del mar.

Procediendo de manera similar, se generaron modelos basados en reglas para realizar la clasificación, en esta fase del experimento se probó principalmente el algoritmo K-NN proporcionado por WEKA. La figura 4.16 muestra el conjunto de reglas obtenidas para datos sintéticos. Se puede notar que se utilizan todos los descriptores para realizar la clasificación, logrando con ello clasificaciones del 100% tanto en el conjunto de entrenamiento como en el conjunto de pruebas. La figura 4.17 muestra las reglas obtenidas para datos reales.

```
=== Run information ===
Scheme:
              weka.classifiers.rules.NNge -G 5 -I 5
Relation.
              training
Instances:
              70
Attributes:
             7
              corr
             mdi
              asm
              disim
              cont
              ent
              class
Test mode:
              10-fold cross-validation
=== Classifier model (full training set) ===
NNGE classifier
Rules generated :
     class ST2 IF : -0.0026747<=corr<=0.0038574 ^ 0.13213<=mdi<=0.13362 ^ 0.0021355<=asm<=0.0021454 ^
                     7.0087<=disim<=7.0349 ^ 78.475<=cont<=79.074 ^ 6.4012<=ent<=6.408 (7)
     class ST3 IF : -0.0038759<=corr<=0.0036278 ^ 0.11164<=mdi<=0.11256 ^ 0.0015312<=asm<=0.0015426 ^
                     9.3377<=disim<=9.402 ^ 149.8<=cont<=152.09 ^ 6.8042<=ent<=6.8144 (7)
     class ST4 IF : -0.0033079<=corr<=0.0010701 ^ 0.12522<=mdi<=0.12599 ^ 0.0027919<=asm<=0.0028406 ^
                     13.852<=disim<=13.903 ^ 453.03<=cont<=462.26 ^ 6.8713<=ent<=6.8791 (7)
     class ST5 IF : -0.0026275<=corr<=0.0018231 ^ 0.13331<=mdi<=0.13448 ^ 0.0031338<=asm<=0.0031872 ^
                     14.415<=disim<=14.5 ^ 579.55<=cont<=590.85 ^ 6.8134<=ent<=6.822 (7)
     class ST6 IF : -0.0016989<=corr<=0.0034553 ^ 0.13611<=mdi<=0.13784 ^ 0.0034014<=asm<=0.0034564 ^
                     14.618<=disim<=14.749 ^ 613.14<=cont<=629.72 ^ 6.7878<=ent<=6.7961 (7)
     class ST7 IF : -0.0036075<=corr<=0.003826 ^ 0.13963<=mdi<=0.14116 ^ 0.0037527<=asm<=0.0038277 ^
                     14.865<=disim<=14.984 ^ 652.12<=cont<=663.9 ^ 6.7532<=ent<=6.7657 (7)
     class ST8 IF : -6.063E-4<=corr<=0.0037436 ^ 0.14487<=mdi<=0.14664 ^ 0.0043092<=asm<=0.0043962 ^
                     15.201<=disim<=15.326 ^ 707.51<=cont<=727.53 ^ 6.702<=ent<=6.7139 (7)
     class ST9 IF : -0.0024388<=corr<=0.0031231 ^ 0.15264<=mdi<=0.15449 ^ 0.0051548<=asm<=0.0053334 ^
                     15.523<=disim<=15.794 ^ 768.66<=cont<=804.45 ^ 6.632<=ent<=6.6563 (7)
     class ST10 IF : -0.0037932<=corr<=0.0021598 ^ 0.16442<=mdi<=0.16601 ^ 0.0068534<=asm<=0.0070268 ^
                    15.97<=disim<=16.297 * 857.52<=cont<=894.93 * 6.517<=ent<=6.5425 (7)
     class ST11 IF : -0.0020593<=corr<=0.0026172 ^ 0.18337<=mdi<=0.18512 ^ 0.010122<=asm<=0.010389 ^
                     16.718<=disim<=16.86 ^ 1022.6<=cont<=1039.6 ^ 6.3542<=ent<=6.3692 (7)
```

Figura 4. 16. Modelo de Reglas generadas con datos sintéticos.

```
=== Run information ===
Scheme:
               weka.classifiers.rules.NNge -G 5 -I 5
Relation:
               training
Instances:
               21
Attributes:
               8
               corr
               mdi
               asm
               disim
               cont
               ent
               homo
               class
               10-fold cross-validation
Test mode:
=== Classifier model (full training set) ===
NNGE classifier
Rules generated :
      class ST2 IF : -0.16348<=corr<=0.043778 ^ 0.20221<=mdi<=0.20907 ^ 0.036001<=asm<=0.038332 '
      11.599<=disim<=11.81 ^ 223.52<=cont<=240.08 ^ 4.6709<=ent<=4.734 ^ 0.26039<=homo<=0.26667 (5)
class ST1 IF : -0.20926<=corr<=0.18517 ^ 0.16844<=mdi<=0.19744 ^ 0.027224<=asm<=0.036155 ^
13.108<=disim<=14.242 ^ 290.26<=cont<=333.64 ^ 4.6245<=ent<=5.0813 ^ 0.22359<=homo<=0.25257 (10)
```

Figura 4. 17. Modelo de Reglas generadas con datos reales.

Los resultados muestran que el empleo de reglas y árboles de decisión con descriptores de textura nos permiten clasificar los estados del mar con altos niveles de probabilidad 87 y 90%. Con estos resultados se puede concluir que la información de textura de la GLCM y filtros de energía es adecuada para describir y clasificar el ruido marítimo. Los principales descriptores utilizados son MDI, Dis y ASM en el caso de descriptores de la GLCM y R5L5, R5R5 y E5L5 en el caso de descriptores de energía. Desde el punto de vista de la arquitectura es más conveniente calcular sólo estos descriptores que calcular todos para evaluar el modelo basado en reglas. Con esta información se puede completar el diseño del esquema KBSP. Los descriptores arriba mencionados formarán la base de datos de hechos, mientras que los modelos probados formaran la base de conocimiento. Los modelos serán evaluados por la máquina de inferencias, utilizando la estrategia de encadenamiento hacia delante, activando las reglas adecuadas en función del estado del mar clasificado. El diagrama de la figura 4.18 muestra el esquema completo para ser llevado a la arquitectura. La heurística a utilizar es la que se discutió en el capítulo 3.



Figura 4. 18. Esquema KBSP para procesamiento de señales de radar.

4.3 CFAR paramétrico.

Como se mencionó en el capítulo tres, uno de los primeros experimentos consistió en implementar un centro de fusión con distintas variantes de CFAR, este experimento se realizó implementando el centro de fusión en Matlab, utilizando un esquema basado en perceptrón de una capa con factor de desvío y hardlims como función de transferencia, todo ello siguiendo el esquema mostrado en la figura 4.19.



Figura 4. 19. Centro de fusión de salidas de CFAR.

La salida de la red neuronal se calculó como se indica en la ecuación 4.3.

$$Deteccion = Hard \lim s \left(\sum W_i P_i + b \right)$$
(4.3)

donde W_i es el vector de pesos asociados a la entrada, P_i es el vector de entradas al centro de fusión (salida de CA, SO y GO-CFAR), las cuales fueron discutidas en el capítulo 3. El entrenamiento del centro de fusión se llevó a cabo de la siguiente manera:

• Se dieron valores iniciales aleatorios a los pesos y al factor de desvío.

- Se le presentó el primer patrón, que son las salidas de CA, SO y GO-CFAR, mostradas en la tabla 3.1. En este punto se presentó también la salida esperada, la cual corresponde a las reglas AND y OR.
- Se calculó la salida de acuerdo con la ecuación 4.3.
- Si la salida es incorrecta los pesos y el desvío son recalculados con las ecuaciones 4.4 y 4.5 respectivamente.

$$W_{next} = W_{previo} + (DeteccionEsperada - Deteccion) * P$$
(4.4)

$$b_{next} = b_{previo} + (DeteccionEsperada - Deteccion)$$
(4.5)

donde *DetecciónEsperada* son los valores mostrados en la tabla 3.1 para las reglas AND y OR. *Detección* es el valor calculado por la red neuronal.

El resultado de aplicar el centro de fusión a varios perfiles de rango se muestra en la figura 4.20. En la sección a) se muestra un perfil de rangos con blancos sintéticos, y ruido marítimo, así mismo se muestra la salida de los tres algoritmos de CFAR empleados y la salida de la red neuronal. Encerrados en un rectángulo se muestran las diferencias en cada uno de ellos, en algunos casos un algoritmo puede detectar un blanco, mientras que otro lo pierde, con las reglas de fusión se puede recuperar el blanco perdido. El comportamiento del centro de fusión es similar si se utilizan datos reales, como los mostrados en la figura 4.20b. Finalmente en la figura 4.21 se muestra la métrica del desempeño del centro de fusión comparado con un detector ideal y con las variantes de CFAR. El detector ideal calcula la probabilidad de detección asumiendo condiciones ideales sin ruido marítimo y sin pérdidas en la transmisión [1]. El esquema usado en este experimento calcula la probabilidad de detección considerando la fusión de modelo de ruido maritimo y variantes de CFAR. Estas probabilidades son comparadas con las del detector ideal, cuyos resultados se grafican en la figura 4.21. Los resultados de esta sección fueron reportados en [1] de la sección "artículos generados".



Figura 4. 20. Detección con el centro de fusión.



Figura 4. 21. Métrica de desempeño del centro de fusión.

Como se mencionó en el capítulo tres, la opción de fusión de salidas CFAR requiere de utilizar múltiples variantes corriendo en paralelo, lo cual implica áreas muy amplias para ser llevado al hardware, aunado al hecho de que éste método no contempla la selección de los parámetros de CFAR de acuerdo al ruido del entorno. En los siguientes párrafos se muestran los resultados de experimentos realizados con variantes de CFAR, para corroborar el buen desempeño al ser configurados con sus parámetros correctos de acuerdo al estado del mar.

Durante la búsqueda del esquema CFAR adecuado para el sistema KBSP estudiado, se realizaron experimentos en Matlab para determinar el desempeño de cada una de las variantes de CFAR. Cada variante se configuró con los parámetros adecuados para el entorno en el cual deben operar, las variantes programadas son CA-CFAR, GO-CFAR, SO-CFAR, OS-CFAR y TM-CFAR, discutidas en el capítulo dos.

Los experimentos consistieron en probar cada una de las variantes de CFAR con distintos parámetros de celdas de guarda (GC), tamaño de la ventana de procesamiento (M), diferente probabilidad de falsa alarma (Pfa) y parámetro de forma (SP) acorde al tipo de ruido existente. En primera instancia se muestra en la figura 4.22 el resultado de los primeros experimentos. Se muestra la salida de las variantes de CFAR considerando un modelo de ruido Gaussiano, como en la mayoría de los trabajos reportados en la revisión bibliográfica. En la sección *a* de la figura se muestran seis blancos de amplitud variable, de 5 a 30dB, este último por considerar que el radar en un momento determinado puede adquirir un blanco cercano de esa magnitud. En la sección b, se muestra un perfil de rangos de ruido generado para un mar del 5. En la sección c se muestra el perfil de rangos del ruido con los blancos integrados, mientras que en la sección d se muestran los umbrales calculados por cada una de las variantes de CFAR. Programadas todas ellas con el mismo valor de CG, M, Pfa y considerando ruido Gaussiano. En las secciones e, f, g, *h* se muestran las salidas de cada una de las variantes. Se puede notar principalmente el exceso de falsas alarmas sobre todo en donde se encuentran los blancos de menor tamaño. El desempeño de estas variantes se midió modificando sus parámetros y calculando la probabilidad de detección para cada tipo de variante CFAR. En las figuras 4.23 a 4.26 se muestra el rendimiento comparados con el detector ideal.

En la figura 4.23 se muestra el desempeño de cada variante utilizando una Pfa de 10⁻³, una ventana de CFAR de 10 celdas y 2 celdas de guarda para cada una de las dos ventanas. Cada variante se comparó con el detector ideal, el cual se muestra en primera instancia. La variante CA-CFAR es la que tiene el peor desempeño, ya que es la versión inicial del algoritmo, esta versión requiere de blancos muy grandes del orden de 25dB para tener una probabilidad de detección del 60%. Las siguientes versiones corrigen algunos de los problemas de CA-CFAR, por lo que su rendimiento es más cercano al detector ideal como se observa en la gráfica 4.23.



Figura 4. 22. Salidas de las variantes CFAR sin configurar.



Figura 4. 23. Desempeño de variantes CFAR con Pfa de 10⁻³, M=10, GC=2, Ruido Gaussiano.

En la figura 4.24 se muestra el rendimiento de las mismas variantes de la figura anterior, sólo se varió el tamaño de la ventana de procesamiento a 20 celdas. El rendimiento se incremento en todas las variantes excepto CA-CFAR. Los resultados muestran que con una ventana de datos de 20 celdas se tiene un desempeño muy cerca del detector ideal. Sin embargo, este comportamiento es para situaciones de mar en calma, con modelos de ruido Gaussiano.

La figura 4.25 muestra el resultado de evaluar una disminución en la probabilidad de falsa alarma, es decir, disminuir las falsas detecciones. Se utiliza una ventana de procesamiento de 10 celdas y modelos de ruido Gaussiano. El resultado

arroja que el desempeño de todas las variantes disminuye y se alejan del comportamiento del detector ideal.



Figura 4. 24. Desempeño CFAR con Pfa de 10⁻³, M=20, GC=2, Ruido Gaussiano.



Figura 4. 25. Desempeño CFAR con Pfa de 10⁻⁴, M=10, GC=2, Ruido Gaussiano.

En la figura 4.26 se muestra el desempeño de las variantes de CFAR variando la probabilidad de falsa alarma a un valor muy bajo de 10^{-5} , podemos notar que el degradamiento en el desempeño de los algoritmos es excesivo por lo que podemos concluir que existe un compromiso entre probabilidad de detección y probabilidad de falsa alarma, siendo un valor adecuado de 10^{-3} .

Finalmente en la figura 4.27 se muestra una comparación de las variantes de CFAR configuradas para entornos Gaussianos y las variantes de CFAR configuradas para un entorno con mar del 5, ambas variantes con pfa de 10⁻³. El resultado es un desempeño mayor en las variantes que fueron configuradas para el tipo de ruido existente, mientras que las variantes que no fueron configuradas su desempeño se

aleja del detector ideal. El resumen de estos experimentos fue expuesto en el capitulo dos en la tabla 2.1 y 2.2.



Figura 4. 26. Desempeño de variantes CFAR con Pfa de 10⁻⁵, M=10, GC=2, Ruido Gaussiano.


Metrica de desempeño mar 5, Pfa = 0.001, Versiones no conf.Ventana CFAR = 20 GC=4

Figura 4. 27. Desempeño de variantes CFAR con Pfa de 10⁻³. Versiones no configuradas utilizan M=10, GC=4.

4.4. Filtro de energía.

En el capítulo tres se discutió sobre la naturaleza del ruido marítimo, se destacaron principalmente los análisis tiempo-espacio y tiempo-frecuencia, en éste último se realizó el análisis de textura mediante descriptores de energía. Experimentos adicionales se realizaron con los filtros de una dimensión propuestos por Laws para aplicarlos sobre el perfil de rangos, los cuales se describen a continuación.

Primero se realizó el análisis de los filtros de orden 5 propuestos por Laws para la descripción de textura. Este análisis consistió en obtener la respuesta en magnitud y fase de los filtros para identificar sus características, ya que Laws sólo propone las máscaras para determinar ciertas características de textura como rugosidad, gradiente, ondulaciones etc. En la figura 4.28 se grafica la respuesta en amplitud de las 5 máscaras propuestas por Laws E5, R5, S5, L5, W5. La gráfica arroja que las máscaras son un conjunto de filtros pasa bajas (L5), pasa altas (R5) y pasa banda (E5, S5, W5).



Figura 4. 28. Respuesta en amplitud de las mascaras de Laws.

Del análisis de respuesta en amplitud y frecuencia de un filtro digital FIR (Finite Impulse Response) pasa bajas como el mostrado en la figura 4.29, 4.30 y 4.31, muestran que la concentración de energía se encuentra en el lóbulo principal

(figura 4.30), así como la sincronía existente en magnitud y fase, marcadas en color azul y verde respectivamente (figura 4.29). Los coeficientes normalizados del filtro de orden 5 se muestran en la figura 4.31. Para este trabajo sólo se cuenta con la información en amplitud que proporcionan los ecos de radar, así que el análisis se basa únicamente en la respuesta en amplitud. El filtro L5 es un filtro pasa bajas cuya respuesta en amplitud y fase se muestran en la figura 4.32 y la respuesta al impulso en la figura 4.33. Del análisis de estas figuras se puede concluir que la característica principal del filtro es la concentración de energía en el lóbulo principal, mostrado en la figura 4.33, no siendo así el análisis en fase, ya que ésta se encuentra fuera de sincronía, lo cual deja un sector por analizar para un trabajo futuro. Si el objetivo del filtro es enfatizar la energía entonces al aplicarlo sobre un perfil de rangos deberá incrementar los niveles de energía en los lugares donde tenga mayor amplitud y duración la señal, como es el caso de los blancos.



Figura 4. 29. Respuesta en amplitud y fase de un filtro FIR de orden 5 pasa bajas.



Figura 4. 30. Respuesta al impulso de un filtro FIR de orden 5 pasa bajas.

Numerator:
-0.067531838889517
0.284207441492324
0.444228821705622
0.444228821705622
0.284207441492324
-0.067531838889517

Figura 4. 31. Coeficientes normalizados del filtro FIR de orden 5 pasa bajas.



Figura 4. 32. Respuesta en amplitud y fase del filtro L5.



Figura 4. 33 Respuesta la impulso del filtro L5.

Para aplicar el filtro de energía L5 al perfil de rangos, se realiza la convolución del filtro y el perfil, posteriormente se aplica el mapa de energía según lo discutido en el capítulo 3). El perfil de rangos está formado por blancos de diferente sección cruzada de radar y de diferente amplitud mezclado con ruido correspondiente a mar del 5, como se muestra en la figura 4.34 b. El resultado es el mostrado en la figura 4.34d, el cual muestra la salida del conjunto al que hemos llamado L5-CFAR o Energy-CFAR. La gráfica muestra el realce que toman los blancos del perfil, así como la eliminación de falsas detecciones, este resultado es comparado con la salida del algoritmo TM-CFAR configurado para suprimir ruido de mar del 5. La salida se muestra en la figura4.33c, se tienen excesivas falsas detecciones, aunado a la pérdida de blancos pequeños. Finalmente en la figura 4.35 se muestra un comparativo del rendimiento de la nueva variante obtenida en este trabajo con las variantes de CFAR configuradas para suprimir ruido marítimo equivalente a mar del 5. La gráfica muestra que el uso de Energy CFAR configurado para las condiciones de ruido existentes, presenta el mejor desempeño con respecto al detector ideal.



Figura 4. 34. Salida de la nueva variante Energy-CFAR comparada con TM-CFAR.



Figura 4. 35 Comparativa de eficiencia entre variantes de CFAR.

4.5 Resumen.

En este capítulo se han mostrado los resultados obtenidos durante los experimentos realizados para determinar la información útil que lleve a mejorar la detección de blancos en señales de radar inmersos en ruido marítimo. Se pudo comprobar que el análisis de textura proporciona información útil para realizar la clasificación de estados del mar, así como la realimentación del sistema y el conocimiento heurístico del operador de radar permite configurar el algoritmo CFAR con sus parámetros adecuados de acuerdo a las condiciones existentes. De esta manera, se satisfacen parte de los objetivos planteados en el capítulo uno, teniendo un esquema KBSP completo como el mostrado en la figura 4.17 para procesar señales de radar junto con la cadena tradicional de procesamiento.

Adicionalmente en este capítulo se mostraron los resultados obtenidos de una variante aportada en este trabajo para la cadena tradicional de procesamiento, siendo ésta el Energy-CFAR. Para completar con los objetivos planteados se debe realizar todo el procesamiento en tiempo real, para lo cual se debe diseñar una arquitectura que responda en el tiempo adecuado, lo cual será tratado en el siguiente capítulo.

Capítulo 5.

Arquitectura para procesamiento de señales basado en conocimiento.

Con los resultados obtenidos en los experimentos realizados en el capítulo 4 se tiene una parte de los objetivos alcanzados. La parte restante se refiere al procesamiento del esquema KBSP en una arquitectura hardware que permita realizar las tareas en tiempo real. En este capítulo se describe el diseño elaborado para evaluar el esquema final mostrado en la figura 4.17 del capítulo anterior. Se describe el diseño e implementación en hardware de cada uno de los módulos que forman el esquema KBSP, se discuten los resultados obtenidos en la implementación en hardware y su desempeño comparado con una implementación en software.

5.1 Introducción.

En la literatura se han reportado trabajos que utilizan la propiedad de reconfiguración de los FPGA para generar arquitecturas de propósito general con procesadores reconfigurables como Splash2 [58] y Piperench [59], las cuales forman matrices de FPGA interconectados mediante un bus para formar computadoras de alto rendimiento y mapear a los FPGA los distintos algoritmos. En los últimos años los fabricantes de dispositivos reconfigurables han incluido sistemas de acceso a los puertos de configuración desde el interior de los propios dispositivos [60]. Para poder hacer uso de estas posibilidades en los nuevos diseños se requieren metodologías, entornos y herramientas que lo faciliten. Los trabajos de Blodget [61] sobre el desarrollo de la plataforma de auto-reconfiguración Self-Reconfiguring Platform (SRP), proponen una plataforma que permite controlar de forma dinámica la reconfiguración del FPGA bajo el control de un microprocesador embebido. Encontramos también en la literatura múltiples trabajos de implementación en hardware de versiones del algoritmo CFAR, tal es el caso de Alsuwailen y Alshebeili [62], en donde implementan la versión de TM-CFAR en una arquitectura que lo ejecuta en tiempo real, utilizan una frecuencia de muestreo de 100MHz pero sólo una ventana de 16 celdas de referencia y 2 de guarda. Los resultados mostrados indican que es una arquitectura que realiza el ordenamiento de los datos a buena velocidad, sin embargo, no presentan el rendimiento del procesador CFAR en el aspecto de la detección.

Magaz y Bencheik[57] proponen la implementación eficiente de la variante OS-CFAR, de forma directa, es decir, no utilizan modelo de ruido para suprimir las falsas detecciones, lo cual si bien es una opción para los estados del mar 0 y 1, esta versión no tendría resultados satisfactorios en condiciones adversas. En [63], [64], [65], [66] Cumplido, Torres y López muestran algunos de los primeros trabajos realizados para esta investigación. Estos trabajos consistieron en la implementación de algunas variantes del algoritmo CFAR, lo cuál arrojó buenos resultados tanto en la velocidad de procesamiento como la latencia generada en esas implementaciones, estos trabajos sirvieron de base para el diseño del CFAR paramétrico que se explicará más adelante.

La implementación del esquema KBSP en hardware se lleva a cabo de acuerdo al diagrama de la figura 4.17. Una versión desde el punto de vista del hardware de ese diagrama se muestra en la figura 5.1, en la cual se destacan principalmente los módulos a implementarse: adquisición del conocimiento, representación del conocimiento, máquina de inferencias y el módulo CFAR parametrizable. En el primer módulo se obtienen los descriptores de textura derivados de la GLCM y filtros de energía para clasificar los estados del mar. Para facilitar la implementación, ésta se realizó utilizando aritmética de punto fijo, utilizando datos de 16 bits. Los datos de los ecos de radar como se mencionó anteriormente son a 8 bits, por lo que para realizar las operaciones a cada dato se le aplica un corrimiento de 8 bits y se realiza operaciones a 16 bits, al final el resultado global se le realiza nuevamente un corrimiento para tener un resultado a 8 bits, con lo cual se pueden manejar errores de redondeo hasta de milésimas. En el segundo bloque se lleva a cabo la clasificación de los estados del mar con el conocimiento almacenado en forma de reglas, así mismo activa las reglas para seleccionar la variante de CFAR y sus parámetros de acuerdo a las condiciones existentes del entorno. El tercer bloque se encarga de implementar el algoritmo CFAR paramétrico y finalmente el bloque que implementa el filtro de energía. En los siguientes párrafos se describe detalladamente cada uno de estos bloques.



Figura 5. 1. Arquitectura KBSP para reconocimiento de textura y selección del algoritmo CFAR.

5.2 Hardware para análisis de textura.

Como se ha mencionado en capítulos anteriores, las primeras actividades del método KBSP consisten en clasificar los estados del mar, mediante la GLCM y descriptores de energía. Para el caso de la GLCM. Tahir y Roula [67] proponen un co-procesador de GLCM basado en FPGA para medir las características de textura, siendo ésta la parte que consume mayores recursos, ya que la latencia depende del tamaño de la imagen a analizar. Se extrae una imagen representativa de la imagen de radar para acelerar el proceso, debido a que el ruido marítimo es homogéneo a lo largo de toda la imagen. Moroulis y Lakovidis en [68] presenta una implementación de un analizador de textura basado en la GLCM para video en tiempo real, esto es posible si se adquiere video de baja resolución, por lo que el esquema propuesto,

proporciona buenos resultados en la clasificación de textura. Los resultados de estos artículos corroboran el resultado de los experimentos realizados para clasificar textura electromagnética de la superficie del mar.

Para la implementación de las arquitecturas diseñadas por bloque se utilizó un FPGA de Xilinx Virtex2 XC2V1000, usando el lenguaje VHDL y Xilinx ISE 8.2 como herramienta de diseño. Para el análisis de textura se implementó un elemento procesador (EP) para generar la GLCM llamado PEGLCM, como muestra la figura 5.1, cuya entrada es la matriz de perfiles de rango Data (i, j). En esta parte del experimento se probó con imágenes de diferentes tamaños, 16x16, 32x 32, 64x64, 128x128 y 256x256 cada una con una resolución de 8 bits, es decir 256 niveles de gris. El PEGLCM calcula sus elementos de la siguiente manera: se define un conjunto de acumuladores que llevan la cuenta de la ocurrencia de los niveles de gris, la figura 5.2 muestra un diagrama de bloques de este elemento. La entrada es la celda de rango de la imagen de radar Data (i, j), el valor del píxel en esa posición se usa como índice de renglones, mientras que el píxel en la posición (i ,j+d) se usa como índice de columnas, recordemos que el valor del desplazamiento para el radar empleado es d=1. El proceso se realiza para todas las posiciones i, j de la imagen de radar, al final del proceso los acumuladores contienen los valores de la GLCM.



Figura 5. 2. Elemento procesador para la GLCM y sus descriptores.

Una vez obtenida la GLCM se calculan de manera paralela los descriptores de textura que utiliza el árbol de decisión generado. Los descriptores de textura se implementaron como EP de acuerdo a las ecuaciones descritas en el capítulo 3, los cuales son PEASM, PEIDM, PEDIS. En la figura 5.2 se observan los EP para calcular los descriptores de textura. Para facilitar los cálculos, se lleva a cabo una normalización de los datos de la GLCM, el valor de normalización se llama Vn, y a los elementos de la GLCM cuya designación era P_{ij} , ahora se llamarán V_{ij} . Los diagramas de bloque de los EP para calcular los descriptores de la GLCM, se muestran en la figura 5.2. Como se puede observar, los EP están compuestos básicamente de acumuladores para implementar las sumatorias y multiplicadores, adicionalmente se tienen almacenados en LUT (Look Up Table) aquellos valores que no cambian durante los cálculos, facilitando de esta manera la implementación y el tiempo de cálculo.

Las tablas 5.1 y 5.2 muestran los detalles de la síntesis de estos EP, cabe destacar que las operaciones de multiplicación se realizan utilizando los

multiplicadores ya diseñados para el FPGA, lo que se traduce como un ahorro sustancial en el área ocupada para estas operaciones.

Número de arreglos	52 de 5120	1%
Número de Flip Flops	96 de 10240	1%
Número de LUT LUTs	88 de 10240	0%
Número de IOBs	96 de 324	29%
Número de MULT18X18s	6 de 40	15%
Número de GCLKs	1 de 16	6%

Tabla 5.1 Estadísticas de síntesis de la GLCM y descriptores.

Mínimo periodo: 5.695ns (Máxima Frecuencia: 175.562MHz)
Mínimo tiempo de arribo clock: 6.771ns
Máximo tiempo de salida despued de clock: 18.173ns
Máximo retardo combinacional: 20.564ns

Tabla 5.2 Resumen de tiempos de los EP GLCM y descriptores.

Los EP de los descriptores de energía se implementan en dos partes, la primera es la convolución del filtro con los datos del perfil de rango, la segunda es el cálculo del mapa de energía de acuerdo a lo discutido en el capítulo 3, el diagrama de bloque de la implementación se observa en la figura 5.3. En la primera parte de la figura se aprecia la forma de implementar la convolución del filtro con la señal de radar. Se puede observar que el esquema planteado realiza la convolución en línea, es decir, se va ejecutando a medida que llegan los ecos de radar. Las entradas para este módulo son por una parte cada una de las celdas de referencia del perfil de rangos X_i y los coeficientes del filtro C_i , los cuales son leídos de memoria para que puedan actualizarse en cualquier momento (concepto adaptativo). En la segunda parte del EP se tiene el mapa de energía que es la sumatoria de un grupo de celdas

vecinas, esto se lleva a cabo mediante un registro de corrimiento y un sumador, de la misma manera que el componente anterior se realiza el procesamiento al vuelo. Las estadísticas de síntesis del EP se muestran en las tablas 5.3 y 5.4.



Figura 5. 3. Elemento procesador para el Filtro de energía.

Número de arreglos	63 de 5120	1%
Número de Flip Flops	60 de 10240	0%
Número de LUT LUTs	55 de 10240	0%
Número de IOBs	55 de 324	16%
Número de MULT18X18s	5 de 40	12%
Número de GCLKs	1 de 16	6%

Tabla 5.3 Estadísticas de síntesis para descriptores de energía.

Mínimo periodo: 1.778ns (Máxima Frecuencia: 562.588MHz)
Mínimo tiempo de arrivo clock: 2.225ns
Máximo tiempo de salida clock: 20.667ns
Máximo retardo combinacional: 21.062ns

Tabla 5.4 Resumen de tiempos para descriptores de energía.

Los descriptores calculados tanto de energía como de la GLCM, las reglas de clasificación de estados del mar discutidas en el capítulo 3, la selección de algoritmos y su configuración se almacenan en memoria, para ser suministrados al siguiente bloque, de acuerdo a la figura 5.1.

5.3 Clasificador de estados del mar y selección de parámetros CFAR.

En este punto se han reportado algunos trabajos que realizan clasificación utilizando árboles de decisión, tal es el caso de Naraganan y Honbo [69], quienes presentan una arquitectura en FPGA para realizar clasificación basada en árboles de decisión. En la referencia [2] de la sección de trabajos derivados de la investigación realizamos la publicación de un árbol de decisión en hardware como clasificador de estados del mar. Por otro lado Xiao Gong [70] propone la segmentación de textura usando estimaciones iterativas de la energía de la imagen.

Para la clasificación de estados del mar, se implementaron los modelos de clasificación basados en árboles de decisión. la clasificación utiliza un conjunto de comparadores, con ello se realiza la clasificación del estado del mar presente, con esta información se podrá seleccionar el mejor algoritmo de detección asociado con su modelo de ruido. La figura 5.4 muestra el diagrama esquemático del EP, en él se pueden distinguir dos partes, la primera evalúa el árbol de decisión para la clasificación de estados del mar, el resultado de esta clasificación, activa las reglas

del siguiente bloque, las cuales determinan la variante de CFAR a utilizar y su parámetros de configuración. La salida de este bloque es el conjunto de parámetros que utilizará la variante de CFAR seleccionada de acuerdo a las condiciones del mar existentes. En las tablas 5.5 y 5.6 se muestran las estadísticas de la síntesis de este módulo.



Figura 5. 4. Máquina de inferencia para clasificación de estados del mar y configuración de CFAR.

Número de arreglos	50 de 5120	1%
Número de Flip Flops	20 de 10240	0%
Número de LUT LUTs	87 de 10240	0%
Número de IOBs	102 de 324	31%
Número de MULT18X18s	1 de 16	6%
Número de GCLKs		

Tabla 5.5 Estadísticas de síntesis para el módulo de clasificación y selección.

Mínimo periodo: 3.625ns (Máxima Frecuencia: 275.862MHz)
Máximo tiempo de llegada clock: 4.813ns
Máximo tiempo de salida after clock: 5.880ns
Máximo retardo combinacional: 7.025ns

Tabla 5.6 Resumen de tiempos para el módulo de clasificación y selección.

5.4 Procesador CFAR.

Como se mencionó en el capítulo 3, el esquema basado en fusión requiere implementar un número elevado de procesadores CFAR, adicionalmente requiere la implementación de la red neuronal, lo que lleva a la problemática del área requerida para implementar toda la arquitectura, por ello se eligió implementar una versión parametrizable del algoritmo CFAR, la cual se configura con los parámetros seleccionados en el módulo de configuración de CFAR. Behar en [71] propone una arquitectura sistólica paralela para implementar un procesador CFAR para radares aéreos inmersos en ruido electrónico (jamming), proponen la descomposición de las tareas en pequeños elementos procesadores actuando en paralelo para logra el tiempo real.

El EP para este módulo se muestra en la figura 5.5. Al igual que los módulos anteriores realiza el procesamiento de los datos de radar al vuelo, los cuales entran a un registro de corrimiento de tamaño variable según el número de celdas de referencia M seleccionado y el tamaño de celdas de guarda GC a emplear. Se tiene un módulo acumulador-ordenador, en donde se realiza el promedio de las ventanas izquierda y derecha o el ordenamiento de los datos de mayor a menor de acuerdo al tipo de CFAR a implementar. Un módulo que implementa la lógica de cálculo según la variante del algoritmo seleccionado, el cual utiliza los modelos de ruido marítimo para el cálculo del umbral. Un módulo comparador para tomar la decisión de declarar un blanco o no declararlo. Finalmente, se tiene un módulo de control el cual

configura el tamaño del registro de corrimiento, selecciona el tipo de operaciones a realizar en el resto de los módulos y en general suministra los parámetros a cada módulo, con ello tenemos un EP parametrizable que se configura de manera diferente de acuerdo a las condiciones del entorno existentes. En las tablas 5.7 y 5.8 se muestran las estadísticas de la síntesis y el resumen de tiempos de la implementación de este módulo en hardware.



Figura 5. 5 Módulo parametrizable de CFAR.

Número de arreglos	400 de 5120	7%
Número de Flip Flops	596 de 10240	5%
Número de LUTs	229 de 10240	2%
Número de IOBs	75 de 324	23%
Número de GCLKs	1 de 16	6%

Tabla 5.7 Estadísticas de la síntesis de la arquitectura.

Mínimo periodo: 7.028ns (Máxima Frecuencia: 142.288MHz)
Mínimo tiempo de llegada before clock: 4.828ns
Máximo tiempo de salida clock: 11.106ns
Máximo retardo combinacional: 11.226ns

Tabla 5.8 Resumen de tiempos para el EP CFAR.

5.5 Filtro de energía.

De los resultados obtenidos en el capítulo 3 encontramos que el filtro en una dimensión se comporta como un detector de envolvente. En [72] T. Aach demuestra el comportamiento de un filtro de textura para transformarlo a un filtro de cuadratura, esto es, enfatiza la componente de amplitud de los blancos.

La implementación del filtro de energía, se lleva a cabo en dos etapas, primero se realiza la convolución del filtro con el vector de datos de radar X[i], lo cual se lleva a cabo por medio del diagrama mostrado en la figura 5.6, en donde los datos pasan a un registro de corrimiento de tamaño igual al orden del filtro N. Los datos son multiplicados por los coeficientes del filtro almacenados en memoria C[i] y que son actualizados constantemente de acuerdo a los cambios del entorno. Finalmente se realiza la suma de un conjunto de valores para obtener el valor convolucionado Y[n].



Figura 5. 6. EP para realizar la convolución de datos con filtro.

La segunda parte es el mapa de energía, la cual es una implementación de la ecuación 3.14. La figura 5.7 muestra el diagrama de bloques de la implementación. La salida del dato convolucionado Y[n] del bloque anterior se toma como entrada para este módulo, el cual tiene un registro de corrimiento para un mapa de energía de hasta 15 elementos. El módulo realiza la suma del valor absoluto de los elementos del registro de corrimiento. Al igual que los módulos anteriores, procesa los datos al vuelo, al momento que entra un dato se obtiene una salida, sólo requiere de un tiempo de latencia igual al tamaño del registro de corrimiento. La salida Y_{new} [n] es la entrada para el siguiente módulo que es el algoritmo CFAR. Las tablas 5.9 y 5.10 muestran las estadísticas de la síntesis de este EP.



Figura 5. 7. Calculo del mapa de energía.

Número de arreglos	63 de 5120	1%
Número de Flip Flops	60 de 10240	0%
Número de LUT LUTs	55 de 10240	0%
Número de IOBs	55 de 324	16%
Número de MULT18X18s	5 de 40	12%
Número de GCLKs	1 de 16	6%

Tabla 5.9 Estadísticas de síntesis para el EP descriptores de energía.

Mínimo periodo: 1.778ns (Máxima Frecuencia: 562.588MHz)
Mínimo tiempo de llegada clock: 2.225ns
Máximo tiempo de salida after clock: 20.667ns
Máximo retardo combinacional: 21.062ns

Tabla 5.10 Resumen de tiempos para el EP descriptores de energía.

5.6 Discusión de resultados.

El radar utilizado en los experimentos de este trabajo fue descrito en el capítulo_3. Este radar adquiere una imagen cada 2.4 segundos, que es el tiempo máximo para realizar el proceso de clasificación de textura. El tiempo crítico en la arquitectura se encuentra en la selección de los parámetros de CFAR y su ejecución, ya que ésta se debe realizar de acuerdo a la frecuencia de repetición del pulso (PRF) del radar. La figura 5.8 muestra un diagrama de tiempos en el que se indica el tiempo máximo para capturar y procesar los datos de radar antes de que emita un nuevo pulso.



Figura 5.8. Tiempo de procesamiento crítico.

Para probar el rendimiento de la arquitectura en hardware se hizo una comparación de ella misma pero implementada en software. La tabla 5.11 muestra estos resultados. Esta parte de las pruebas se desarrollaron con un perfil de rangos procesado por la variante Energy-CFAR implementada en Matlab, ANSI C y hardware.

Se probaron principalmente los elementos procesadores GLCM, Descriptores, Energy, y CFAR. Para todos los casos se tomó una fracción de la imagen de radar de 256X256 pixeles, el tiempo que tardaron en ejecutarse se muestra en la tabla 5.11, tanto la implementación en Matlab como ANSI C rebasan en mucho el tiempo crítico de procesamiento, sin embargo el tiempo en el que debe trabajar la GLCM y obtener los descriptores es de 2.4 segundos, ya que las condiciones del mar no cambian en cada vuelta del radar. Por lo tanto el tiempo crítico es el de CFAR.

Se puede notar que las implementaciones en software rebasan el tiempo crítico, sin embargo, en hardware el tiempo de procesamiento se reduce, ya que se aprovecha al máximo las capacidades del FPGA, reduciendo la latencia entre datos. Este proceso se ilustra en la figura 5.9.

EP	Matlab	ANSIC	Hardware	% ahorro
GLCM	78ms	8.845ms	0.469ms	95%
Descriptores	187ms	15.642ms	1.347ms	92%
Energy	11ms	0.451ms	86.25µs	80%
CFAR	12-16ms	0.683-0.939ms	45.88-56.79µs	95%

Tabla 5.11 Resultados comparativos de tiempos de ejecución de los EP en softwarey hardware.



Figura 5.9. Latencia del procesador CFAR y EP de energía.

La latencia mas larga depende del tamaño de la ventana utilizada, pero una vez que los datos se han propagado, la latencia se diminuye al mínimo, que es la espera de un nuevo dato, como se muestra en la figura 5.9.

Los resultados de la síntesis mostrados anteriormente nos permiten determinar las velocidades de operación para cada uno de los EP del sistema, lo que nos permite determinar los puntos críticos en los cuales se debe poner atención, sin embargo como podemos notar el problema del procesamiento CFAR es critico en el sistema, sin embargo la experiencia adquirida en implementaciones en hardware del algoritmo CFAR, permitieron implementar el esquema de procesamiento al vuelo basado en un arreglo de datos almacenados en FF (Flip-Flop) tipo latch, lo que nos permite implementar el esquema mostrado en la figura 5.9, en donde la latencia se puede reducir hasta el punto en que se tiene procesamiento al vuelo o en tiempo real.

5.7 Resumen.

En este capítulo se mostró la implementación en hardware del esquema KBSP, en donde podemos notar que la característica de todos los EP es la de actuar de forma

independiente pero interactuando entre si. Otro aspecto importante de la implementación es la propiedad que tiene de adaptarse a las condiciones existentes, es decir cada bloque puede ajustarse a los parámetros que le sean suministrados. Los resultados de la síntesis de los diseños muestran que los EP responden en el tiempo de la aplicación, es decir se obtiene una respuesta del sistema antes de que se presente un cambio significativo en las condiciones del entorno.

Capítulo 6.

Conclusiones, aportaciones y trabajo futuro.

En este capítulo se emiten las conclusiones del presente trabajo, para su explicación las dividiremos en dos secciones: las derivadas de los experimentos realizados con el esquema basado en conocimiento y aquellas surgidas de la implementación del esquema KBSP en hardware, para finalizar con algunas conclusiones generales y los logros alcanzados con el trabajo descrito en este documento.

6.1 Conclusiones y aportaciones del esquema basado en conocimiento.

Para alcanzar el objetivo principal, que es encontrar un método basado en conocimiento para procesar señales de radar y realizar la detección de blancos en presencia de ruido marítimo, se realizaron los experimentos con la única fuente de

información conocida, que es la señal o ecos de radar, a la cual se le realizó el análisis tiempo-escala, tiempo-frecuencia. El primero de ellos mediante la matriz de co-ocurrencias de niveles de gris (GLCM) y sus descriptores de textura, el segundo mediante los descriptores de textura basados en la energía de los ecos.

En este trabajo, uno de los objetivos planteados, es la determinación del tipo de información que puede ser útil para realizar procesamiento de los ecos de radar . El estado del mar es parte de la información útil del entorno que se requiere conocer para realizar detección óptima. Para determinar el estado del mar a partir de la señal de eco recibida por un radar monoestático, se puede concluir que el análisis de textura realizado sobre los ecos recibidos, empleando la GLCM y sus descriptores, permite obtener de manera automática esta información para ser utilizada en un esquema KBSP. En este punto cabe destacar que una de las aportaciones realizadas fue la creación de una arquitectura configurable sobre FPGA para clasificar estados del mar basado en árboles de decisión. Ver referencia [2] de la lista de publicaciones derivadas del presente trabajo.

De los experimentos realizados con los descriptores de energía se concluye que al usar una máscara de 2 dimensiones se realizan demasiadas operaciones para una imagen de radar, sin embargo, la información que proporcionan es de mayor utilidad que la de la GLCM, ya que al aplicar los filtros en una dimensión sobre un perfil de rangos, se comportan como detector de envolvente de los blancos, por lo que enfatizan su presencia. La información de energía extraída de los ecos de radar también forma parte de la información útil del entorno para el esquema KBSP. De los resultados obtenidos con estos experimentos, se desarrollo de una variante de CFAR a la cual se llamó Energy-CFAR, ya que combinando el filtro de energía con el procesador CFAR. Esta variante mejora el proceso de detección y constituye otra de las aportaciones del presente trabajo. Con la información obtenida con los descriptores de textura y la información de energía se puede calcular información adicional como es la cantidad de blancos presentes en el entorno, aunado a la información que pueda aportar el experto operador de radar, como son zonas conocidas de ruido o el numero de blancos y distancias entre ellos, se formó una base de datos de hechos presentes en el entorno de navegación. Con los resultados obtenidos al procesar las señales de radar con los modelos de clasificación de estados del mar, la base de datos de hechos y el conjunto de reglas utilizado para seleccionar y configurar el algoritmo CFAR, se concluye que el esquema basado en conocimiento, planteado como un sistema experto es adecuado para mejorar el desempeño en el proceso de detección. Esta mejora significa que la probabilidad de detección de blancos inmersos en ruido marítimo es mayor, comparada con los esquemas de procesamiento tradicional alta aun con blancos de poca amplitud. Esta conclusión se desprende del análisis realizado en el capitulo 3. Hasta este punto se tiene cubierto la mitad del objetivo principal que es el esquema de procesamiento de señales basado en conocimiento, la otra parte es que este esquema pueda ejecutarse en tiempo real para el tipo de radar utilizado.

6.2 Conclusiones y aportaciones de la arquitectura para procesar señales de radar basado en conocimiento.

Los experimentos realizados con el algoritmo CFAR permitieron conocer los parámetros adecuados para cada una de las condiciones del entorno existentes, completando con ello la información requerida para cumplir con los objetivos planteados. Esta información permitió también plantear una arquitectura basada en un algoritmo parametrizable que nos permita reconfigurarlo a medida que cambia el entorno. La implementación de CFAR parametrizable en hardware, permite disminuir la cantidad de datos a ser transferida durante la reconfiguración lo que hace eficiente éste proceso.

La implementación en hardware de cada uno de los componentes del esquema basado en conocimiento y el resultado del análisis realizado en el capítulo 5 nos lleva a concluir que es posible realizar la implementación en hardware de técnicas de procesamiento tradicionales y basadas en conocimiento en tiempo real para un radar monoestático pulsado. La aportación de está arquitectura está reportada en la referencia 3 de los artículos aceptados, derivados del presente trabajo.

La implementación en hardware del CFAR paramétrico y el filtro de energía constituyen otra aportación, ya que es una alternativa para procesar los perfiles de rango en la cadena tradicional en tiempo real, comparado con otras implementaciones KBSP que no se ejecutan en tiempo real.

6.3 Conclusiones y aportaciones generales.

Los esquemas basados en conocimiento mejoran el rendimiento en el proceso de detección de blancos inmersos en ruido marítimo. Incrementan la probabilidad de detección acercándose a las características del detector ideal. Es posible llevar sistemas basados en conocimiento a sistemas embebidos trabajando en tiempo real, tradicionalmente se desarrollaban para equipos de computo con un conjunto de instrucciones y con ayuda de software especial.

En el presente trabajo se obtuvo el método adecuado para procesar las señales de radar, el cual se describe a continuación:

1.- En un proceso *off line* se obtienen los modelos de clasificación de estados del mar, basados en reglas y árboles de decisión, estos modelos forman parte de la base de conocimiento.

2.- Se obtienen los descriptores de textura de los ecos de radar que describen el entorno de navegación. Estos descriptores formaran la base de datos de hechos y se utilizan para clasificar los estados del mar.

3.- Con la información del entorno y la información heurística del experto se forma el conjunto de reglas que tomaran la decisión de seleccionar el algoritmo y sus parámetros. 4.- Se clasifican los nuevos datos y se obtiene le estado del mar.

5.- Conociendo el estado del mar la máquina de inferencias activará la regla o reglas adecuadas para seleccionar el algoritmo y sus parámetros de configuración.

6.- Se procesan los nuevos datos con el algoritmo seleccionado

6.4 Trabajo futuro.

El procesamiento tiempo-frecuencia esta basado en el filtro de energía, el cual fue diseñado para contemplar la componente de amplitud de la señal de radar, sin embargo se debe considerar que la componente de fase es un trabajo adicional que se debe desarrollar, para tener completo un esquema de procesamiento de señales de radar de cualquier tipo.

Como ya se menciono el radar monoestático solo transmite y recibe en una sola polarización, horizontal para este trabajo, por lo que la información que se obtiene es limitada a esta polarización. Se debe explorar la posibilidad de incluir polarimetría en el análisis de la textura, para ampliar el conjunto de datos disponibles para la clasificación e incluso para hacer reconocimiento.

Se debe extender el procesamiento KBSP a otros módulos del radar para llegar a tener el radar cognitivo planteado por Haykin [43]

Como se ha destacado en la implementación del sistema KBSP en hardware, se optó por un esquema parametrizable, el cual se ajusta a los parámetros suministrados, sin embargo se puede explotar la característica de reconfiguración dinámica de los FPGA para implementar una arquitectura de este tipo totalmente reconfigurable, lo que permitirá extender los resultados obtenidos a otros dominios.

Los descriptores de textura permiten conocer más información del entorno marítimo, por lo que es conveniente continuar con los experimentos en esta área para encontrar información como altura de las olas, dirección del viento y la corriente y utilizar esta información como realimentación que pudiera generar nuevo conocimiento, o bien desarrollar otro tipo de aplicaciones.

Se debe de extender el concepto analizado en este trabajo para llevarlo a otro tipo de ruido, como la lluvia o la niebla, condiciones presentes también en un entorno de navegación.

6.5 Artículos publicados.

- [1].S. Lopez, R. Cumplido. "Fusion Center With Neural Network for Target Detection in Background Clutter". Sixth Mexican International Conference on Computer Science. Pp 189-197. IEEE Computer Society Press, Sept. 2005.
- [2].S. López, R. Cumplido." Decision Tree Based FPGA-Architecture for Texture Sea State Classification". 3rd Intenational Conference on Reconfigurable Computing and FPGA's. IEEE Computer Society Press, Sept. 2006.
- [3].S. Lopez, R Cumplido "FPGA-architecture for Knowledge-Based Target Detection in Radar Signal Processing". Engineering of Reconfigurable Systems and Architectures, ERSA Conf.erence. Julio 2009.
- [4].S. Lopez, R. Cumplido. "A Knowledge-Based System for Sea State Recognition and Target Detection Using a Single Marine Radar Sensor" Computational Intelligence. IASTED. agosto 2009

6.6 Artículo en preparación.

S. Lopez, R. Cumplido "Reconfigurable FPGA-Based Architecture of Adaptive Filtering based on Energy-CFAR processor for Radar Target Detection"

Glosario de términos

CFAR	Constant False Alarm Rate
	Razon constante de falsas alarmas
KBSP	Knowledge Based Signal Processing
	Procesamiento de señales basado en conocimiento
FPGA	Field Program Gate Array
	Arreglo de compuertas programadas en campo
PPI	Plan Position Indicator
	Indicador plano de posición
PRF	Pulse Repetition Frequency
	Frecuencia de repetición de pulso
SS	Sea State
	Estado del mar
Т	Threshold
	Umbral
М	Numero de celdas de rango
RC.	Reference cell
	Celdas de referencia
CUT	Cell under test
	Celda bajo prueba
SF	Scale factor
	Factor de escala
Pfa.	Probabilidad de falsa alarma
SP	Shape parameter
	Parametro de forma
Х	Perfil de rangos
CA-CFAR	Cell Averaging CFAR
	CFAR promediador
GO-CFAR	Greatest Off CFAR
	CFAR mayor
SO-CFAR	Smallest Off CFAR
	CFAR menor
TM-CFAR	Trimmed mean CFAR
	CFAR ponderado
OS-CFAR	Order Statistic CFAR
	CFAR ordenado
TN	Target Numeber
	Número de blancos
MD	Minimum Distance
	Minima distancia
ASIC	Application Specific Integrates Circuit

	Circuito integrado de aplicación especifica
DARPA	Defence Avanced Research Projects Agency
KASSPER	Knowledge Aided Sensor Signal Processing and Expert Reasoning
MTI.	Moving Target Indicator
	Indicador de blancos moviles
STAP	Space Time Adaptive Processing
	Procesamiento adaptativo tiempo espacio
GOPS	Giga Operations Per Second
	Giga operaciones por segundo
GLCM	Grey Level Co-Ocurrence Matrix
	Matriz de co-ocurrencia de niveles de gris
MDI	Moment Diference Inverse
	Momento de diferencia inverso
ASM	Angular Second Moment
	Segundo momento angular
WEKA.	Waikato Environment for Knowledge Analysis
L	Level
	Nivel
E	Edge
	Borde
S	Spot
	Punto
R	Riple
	Rizado
W	Wablet
	Onda
GHz	Giga Hertz
dB	decibel
V	Volts
mV	milivolts
Rpm	Revoluciones por minuto
KHz	Kilo Hertz
Mn.	Millas náuticas
FIR	Finite impulse response
	Respuesta finita al impulso
EP	Elemento procesador
VHDL	Very High Definition Lenguage
	Lenguaje de definición de alto nivel
LUT	Look Up Table
	Tabla de actualización
FF	Flip Flop
	Balculador
Referencias

- [1]. Skolnik M. Introduction to radar system. Mc Graw Hill. New York. 2001.
- [2].Fulvio Gini. "Knowledge Based Systems for Adaptive Radar" IEEE Signal Processing Magazine. January 2006. 14-17.
- [3].M. Wicks. "Incorporating Knowledge Base Techniques in Radar Signal Processing, past, present and future". KBTech. Radar Signal Proc. 2005.
- [4].M. Cowper, B. Mulgrew "Non Linear Processing of High Resolution Radar Sea Clutter". Proceedings at International Joint Conference on Neural Networks. 1999. 2633-2638.
- [5].C. French, J. Suh. "Novel Signal Processing Architecture for Knowledge-Based STAP Algorithms". IEEE. 2004.
- [6].G. Richard. "Electronic Intelligence: The analysis of radar signals". Willey. 1982.
- [7].B. R. Mahafza. Radar Systems Analysis and Design using Matlab. Ed. Chapman & Hall/CRC. Alabama. 2000.
- [8].F. Posner. "Spiky Sea Clutter High Range Resolution and Very Low Grazing Angles". IEEE Transactions on Aerospace and Electronics Systems. Vol. 38 No 1. 2002.
- [9].G. Richard. Radar System Performance Modeling. Artech House. Sec. Ed. 2005.

- [10]. Farina, Gini, Greco. "High Resolution Sea Clutter Data: A Statistical Analysis of Recorded Live Data" IEE Proceedings Radar, Sonar & Navigation. Vol. 144 No. 3. 1997. 121-130.
- S. Leung, W. Minett. "CFAR Data Fusion Using Fuzzy Integration".
 Proceedings of the Fifth IEEE International Conference on Fuzzy Systems. 1996. vol. 2, pp 1291-1295
- [12]. A. D. Mario, A. Farina,G. Foglia. "Knowledge-based recursive least Squire techniques for heterogeneous clutter suppression". Radar Sonar and Navigation. 2007. pp 106-115
- [13]. Fred E. Nathanson. Radar Design Principles Signal Processing and the Envoronment. Scitech Publishing 1999.
- [14]. S. Haykin, R. Bakker. "Uncovering Nonlinear Dynamics: The Case of Study of Sea Clutter". Proc. IEEE. May 2002. vol. 90, pp 860-881
- [15]. Cowper M, Mulgrew B. "Non Linear Processing of High Resolution Radar Sea Clutter". Proceedings at International Joint Conference on Neural Networks. 1999. Vol. 4, pp 2633-2638
- [16]. G. Davidson, K. Ouchi. "Numerical Laplace Analysis of K-Distributed Clutter in Noise". Radar toolbox. www.radarworks.com.
- [17]. J. Chong, M. Zhu "Ship Detection Algorithm of SAR Imagery Based on Local Window K-Distribution". Symposium. Advances in Microwave Remote Sensing Applications 1999.
- [18]. S. Watts. "Cell-averaging CFAR gain in spatial correlated Kdistributed sea clutter". IEE Proceedings Radar, Sonar & Navigation. Vol. 143 No. 5. 1996. 321-327.

- [19]. N. Levanon, M. Shor. "Order Statistics CFAR for Weibull Background". IEE Proceedings, vol. 137 No. 3. 1990, 157-162.
- [20]. S. Lopez, R. Cumplido "A Hybrid Approach for Target Detection Using CFAR Algorithm and Image Processing" Fifth Mexican International Conference on Computer Science. 2004. 108-115.
- [21]. J. Leszek. "Bayesian State-Space Modeling of Spatio Temporal Non Gaussian Radar Return". PhD Teses. Cambridge University. 1998.
- [22]. P.Z.Pebels."Radar Principles". Jhon Wiley & Sons, Inc. New York 1998.
- [23]. NIST/SEMATECH. "e-Handbook of Statisticla Methods". http://www.itl.nist.gov/div898/handbook/index.htm.
- [24]. R. Ravid. N. Levanon. "Maximum Likelihood CFAR for Weibull Background". IEEE Proceedings. Vol. 139. No.3. June 1992.
- [25]. Keith D., Simon Watts Ward. "Scattering, the K Distribution and Radar Performance". IET 2006.
- [26]. A. Abdi. M. Kaveh. "K distribution: an appropriate substitute for Rayleigh-lognormal distribution in fading-shadowing wireless chanels". Elect. Lett. Vol. 34, pp 851-852, 1998.
- [27]. R.S. Raghavan. "A Method for Estimating parameters of K-Distributed Clutter". IEEE Transactions on Aerospace and Electronics Systems. Vol. 27 No. 2. March 1991
- [28]. F. totir, E. Radoi. "Advanced Sea Clutter Models and their Usefulness for Target Detection". HAL-00349339, V 1, 2008.

- [29]. T. Zhang "ACFAR processor for the detection of unknown random signals in nonstationary correlated noise". Signal Processing 48. 1996.
- [30]. J. Minett. W. Leung. "Shadow-Feature-Enhanced ML-CFAR Detection for a Swerling 1 Target in Weibull Clutter". IEEE Radar Conference 2001. pp. 444-447.
- [31]. D. Kevin "OS Characterization for local CFAR Detection". IEEE Transactions on Systems Man and Cybernetics. 1991, pp 1212-1216.
- [32]. D.Drumheller, H.Lew. "A Parameter- Intensive False Alarm Rate Detection Processor". Technical report. Defence Science & Technology Organization. Electronics and Surveillance Research Laboratory. 2001.
- [33]. L. Zhaoe, W. Liu. "A Novel Approach for CFAR Processor Design". IEEE Radar Conference. 2001, pp 284-288.
- [34]. D. Maio, A. Farina. "Design and experimental validation of knowledge-based CFAR detectors". IEEE Radar conference 2006.
- [35]. Wicks M, Baldigo J, Brown R. "Design and experimental validation of knowledge-based CFAR detectors". United States Patent 5499030.
- [36]. Jylha J, Kerminen R. "New Aspects to Knowledge.aided clutter analysis". IEEE radar conferences. 2006.
- [37]. K. Compton, S. Hauck. "Reconfigurable Computing: A Survey of Systems and Software". ACM Computing Surveys, vol. 34 No. 2, Junio 2002, pp 171-210.
- [38]. R. Sidhu, A. Mei, V.K. Prassana. "Genetic Programming using Self-Reconfigurable FPGAs". Lecture Notes in Computer Science, 1999, pp 1673:301-312.

- [39]. K.H. Leung, K.W. Wong, P.H.W. Leong. "FPGA Implementation of a Microcoded Elliptic Curve Cryptographic Processor". Proceedings IEEE Symposium on Field-Programmable Custom Computing Machines. 2000, pp 68-76.
- [40]. Abdulha, M, Saleh A." Design and Implementation of a Configurable Real-Time FPGA-Based TM-CFAR Processor for Radar Target Detection" J. of avtive and passive Electronics Devices Vol. 3 pp 241-256.
- [41]. Magaz B. Bencheikh L. "An Efficient FPGA Implementation of The OS-CFAR Processor". International Radar Symposium. 2008, pp 1-4.
- [42]. "KASSPER: Knowledge-Aided Sensor Signal Processing and Expert Reasoning", KASSPER Workshops, Conf. Proc. Data. Air Force Research Laboratory, DARPA 2004. <u>http://www.darpa.mil/spo/programs/kassper.htm</u>
- [43]. S. Haykin "Cognitive Radar". IEEE Signal Processing Magazine 2006. pp. 30-40
- [44]. J. Guerci, L. Baranoski. "Knowledge-Aided Adaptive Radar at DARPA". IEEE Signal Processing Magazine. January 2006, pp 41-50.
- [45]. J. Bergin, M. Teixeira. "STAP with Knowledge Aided Data pre Whitening" Proc. IEEE Radar Conference Philadelphia 2004, pp 289-294.
- [46]. G. Capraro, A. Farina "Knowledge-Based Radar Signal and Data Processing". IEEE Signal Processing Magazine. January 2006, pp 18-29.
- [47]. C. Wicks, M. Rangaswamy "Space-Time Adaptive Processing". IEEE Signal Processing Magazine. January 2006, pp 51-65.

- [48]. S. Miranda, C. Baker "Knowledge-Based Resource Management for Multifunction Radar". IEEE Signal Processing Magazine. January 2006, pp 66-76.
- [49]. L. Melvin, A. Showman "A Knowledge Aided GMTI Detection Architecture" Proc. IEEE Radar Conference Philadelphia 2004, pp 301-306.
- [50]. Gangeskar R, "Ocean Current Estimated from X-band Radar Sea Surface Images", IEEE Transactions and Remote Sensing, vol. 40, no. 4. 2002, pp 783-792.
- [51]. Haralick R. M., Shammugam K. "Texture Features for image classification". IEEE transaction on systems, Man and Cybernetics. 1973, pp 610-622.
- [52]. K. Laws. Textured Image Segmentation, Ph.D. Dissertation, University of Southern California, January 1980.
- [53]. J. Melendez "Integration of Knowledge-Based, Qualitative, and Numerical Tools for Real Time Dynamics Systems Supervision". PhD. Tesis University of Girona. 1997.
- [54]. Witten I. H., Eibe F.: "Data Mining, Practical Machine Learning Tools and Techniques". 2nd edn. Morgan Kaufman Publishers. 2005.
- [55]. Sanjit Mitra. "Digital Signal Processing" Third Edition Mc Graw-Hill. 2006.
- [56]. S. Lopez, R. Cumplido "Detecta: una Herramienta Software Para Evaluación esquemas de Detección de Blancos en Presencia de Ruido Marítimo". XIII Congreso Internacional de Computación. 2004, 12p

- [57]. B. Magaz, L. Bencheikh. "An Efficient FPGA Implementation of the OS-CFAR Processor" Radar Symposium, 2008 International. 21-23 May 2008, pp 1-4
- [58]. D. A. Buel, J.M. Arnold, W.J. Kleinfelder "Splash 2: FPGAs in a Custom Computing Machine". IEEE Computer Society Press, 1996.
- [59]. S. C. Goldstein, H. Schmit, M. Budiu, "PipeRench: A Reconfigurable Architecture and Compiler", IEEE Computer, Vol. 33, No. 4, 2000, pp 70– 77.
- [60]. Xilinx Corp.: ISE 6.1 Xilinx Libraries Guide. http://support.xilinx.com, 2003.
- [61]. B. Blodget, P. James-Roxby, E. Keller, S. McMillan, P. Sundararajan."A Self-reconfiguring Platform". Lecture Notes in Computer Science, 2003, pp 565-574.
- [62]. M. Alsuwailem, A. Alshebeili. "Design and Implementation of a Configurable Real-Time FPGA-Based TM-CFAR Processor for Radar Target Detection". J. of Active Pasive electronic devices, Vol3. 2008 pp241-256.
- [63]. R. Cumplido, C. Torres, Santos Lopez "A Configurable FPGA-based Hardware Architecture for Adaptive Processing of Noisy Signals for Target Detection Based on Constant False Alarm Rate (CFAR) Algorithms", Global signal Processing Conference, GSPx 2004. SantaClara CA. Septiembre 2004.
- [64]. S. López-Estrada, R. Cumplido-Parra, C.Torres-Huitzil "A Hybrid Approach for Target Detection Using CFAR Algorithm and Image Processing", Fifth Mexican International Conference in Computer Science

(ENC'04), Septiembre 2004, IEEE Computer Society Press, ISBN: 0-7695-2160-6/04. pp. 108-115.

- [65]. Rene Cumplido, César Torres, Santos López. "On the Implementation of an efficient FPGA-based CFAR Processor forTarget Detection", 1st International Conference on electrical and Electronics Engineering, Acapulco, México. Septiembre 2004. IEEE Catalog number: 04EX865C, ISBN: 0-7803-8532-2.
- [66]. César Torres-Huitzil, Rene Cumplido-Parra, Santos López-Estrada, "Design and Implementation of a CFAR Processor for Target Detection", FPL04. Amberes, Bélgica, Agosto 2004. Lectures Notes on CumputerScience Vol. 3203, pp. 943-947. ISBN 3540229892.
- [67]. M. Tahir, M. Roula, A. Bouridane, F. Kurugollu. "An FPGA based co-processor for GLCM texture feature measurement". IEEE Int. Conf. Electronics, Circuits and Systems. 2003, vol3 pp 1006-1009 ISBN: 0-7803-8163-7.
- [68]. D. Moroulis, D. Lakovidis. "FPGA-based System for Real-Time Video Texture Analysis". J. of signal Processing Systems. Vol 53 2008 pp 419-433.
- [69]. R. Nayaranan, D. Honbo "An FPGA Implementation of Decision tree Classification". Design, Automation & Test in Europe Conference. 2007 pp 16-20.
- [70]. X. Gong, NK. Huang. "Texture Segmentation Using Iterative Estimate of Energy States". International Conference on Pattern Recognition. 1998. vol 14 pp51-55.

- [71]. P. Behar, C.kabakchiev, L. Doukovska "Adaptive CFAR PI Processor for Radar Target Detection in Pulse Jamming". J of VLSI Signal Processing 2000. vol. 26 pp383-396.
- [72]. T. Aach, A. Kaup, R. Mester"On Texture Analysis: Local Energy Transforms versus Quadrature Filters". Signal Processing. 1995 vol. 45 pp 173-181.
- [73]. H. Nomiyama, T. Hirayama. "Evaluation of Marine Radar as an Ocean-Wavewe-Field Detector Through Full Numerical Simulation". J. Marine Science & Technology. 2003.
- [74]. D. Doong, L. Wu. "Spatial Wave Fields extracted from marine radar images by Wavelet transform". Third Chinese-German Joint Symposium on Coastal and Ocean Engineering. 2006.
- [75]. J. Nieto-Borge, P. Jarabo-Amores. "Estimation of Ocean Wave Heights from Temporal Sequences of X-Band Marine Radar Images". European Signal Processing Conference. 2006.