

Diseño e implementación de un extractor de video-radar y seguimiento

Nélida Gálvez, Agustín Campo Kihn, Christian Galasso,
Diego Martínez, Ricardo Cayssials y Juan Cousseau

SIAG, Armada Argentina

Universidad Nacional del Sur
www.uns.edu.ar

UTN Bahía Blanca
www.frbb.utn.edu.ar

Contactos

Nélida Gálvez: ngalvez@uns.edu.ar

Agustín Campo Kihn: agustin.campo.kihn@gmail.com

Christian Galasso: christian_galasso81@yahoo.com.ar

Diego Martínez: dmmartinez7@gmail.com

Ricardo Cayssials: rcayssials@frbb.utn.edu.ar

Juan Cousseau: jcousseau@uns.edu.ar

Nota del editor. El artículo aquí reproducido fue presentado originalmente como trabajo de investigación en el Congreso Argentino de Control Automático, en el marco de la Semana de Control Automático, llevada a cabo por AADECA en 2020.

En este trabajo se presenta el desarrollo de un extractor de video-radar y tracking (EVRT), que consiste en un sistema integrado de software y hardware, cuyo principio de funcionamiento se basa en la detección y posterior seguimiento en tiempo real de varios objetos de interés (embarcaciones, aeronaves, etc.) en forma automática y simultánea, dentro de un área, para brindar al usuario toda la información posible acerca de los objetos detectados (ubicación en rango y acimut, velocidad, aceleración, rumbo, etc.).

El principal objetivo de este proyecto es el de dotar a las unidades de la Armada Argentina de un sistema de alta complejidad con desarrollo propio adaptado a los requerimientos, versátil dentro de ciertos márgenes, con bajo costo de inversión y capacidad de producir tantos equipos como sea necesario. Brindará, además, la posibilidad de interactuar con otros proyectos ya desarrollados y en proceso de desarrollo, lo que permitirá disponer de un sistema completo de simulación, de prueba, visualización de señales radar y de extracción de la información contextual en la vigilancia marítima.



Figura 1. EVRT en funcionamiento

La señal proveniente de los objetos detectados se reproduce en la consola táctica, por lo que antiguamente, la capacidad de detección y seguimiento de los objetos estaba supeditada a la pericia de los operadores que observaban una imagen [15]. Los destellos provenientes de los objetos, también llamados “ecos”, normalmente llegan inmersos en ruido y entorno (se refiere de este modo a los ecos no deseados recibidos por el radar y que son causados por el entorno marino) [2].

La propuesta se implementa sobre un sistema de software y hardware con tecnologías modernas tales como módulos con FPGA, con capacidades de procesamiento y manejo de datos muy superiores a las del sistema anterior.

En la actualidad, los ecos radar se detectan en forma automática, con diferentes metodologías, mediante las cuales se captura la señal y se procesa con diversos algoritmos. El objetivo es lograr una indicación precisa de la ubicación, y así luego poder hacer un seguimiento en forma automática dentro de la consola táctica.

El desarrollo que se describe en el presente trabajo es la actualización de un sistema de detección y seguimiento de señales radar, en el que se utilizó como base para el procesamiento un clúster de procesadores digitales de señales (DSP) [15]. La propuesta se implementa sobre un sistema de software y hardware con tecnologías modernas tales como módulos con FPGA, con capacidades de procesamiento y manejo de datos muy superiores a la del sistema anterior, lo que trajo como consecuencia la necesidad de rediseñar una arquitectura general acorde al hardware

propuesto, como así también el estudio, investigación e implementación de los procesos en este nuevo sistema. Con el presente desarrollo se logró mejorar varios aspectos del diseño como, por ejemplo, inconvenientes en la captura y almacenamiento de los datos, antes realizada por sectores, o cuestiones asociadas a las velocidades de procesamiento. En la figura 1 se muestra el sistema en funcionamiento. En el caso de la detección se trabajó en el criterio de Neyman Pearson [13], mediante el modelado de la señal radar utilizando diferentes funciones de densidad de probabilidad (PDF, por sus siglas en inglés) [18], [14]. Se estudiaron varias técnicas para la estimación de parámetros [13], como así también los procesos CFAR [8] y ventana deslizante (MW, por sus siglas en inglés) [5]. Con respecto al proceso de seguimiento, se trabajó en la resolución de trayectorias en base a los conceptos de “correlación”, “asociación” y “predicción”, con la implementación del filtro Kalman, que aunque tiene más de cincuenta años de antigüedad se sigue utilizando [12], [19], [6].

Si bien en el mercado se presentan productos que brindan funciones similares, por ejemplo el SPx Tracking Server de Cambridge Pixel o la placa Cougar 2 Radar de Curtiss-Wright, que efectúa captura de datos, conversión de video y seguimiento de objetos de interés; tener un desarrollo propio permitirá disponer de un sistema adaptable a las necesidades de la Armada Argentina, con la posibilidad de instalarse en varias unidades de mar, de bajo costo de fabricación, instalación y mantenimiento.

El proceso de investigación para el desarrollo del EVRT lleva varios años de trabajo conjunto entre la Armada Argentina (ARA-SIAG), la Universidad Nacional del Sur (UNS-DIEC) y la Universidad Tecnológica Nacional (UTN-FRBB), en el marco de convenios de cooperación entre las instituciones mencionadas [15], [10], con el financiamiento del Ministerio de Defensa de la República Argentina.

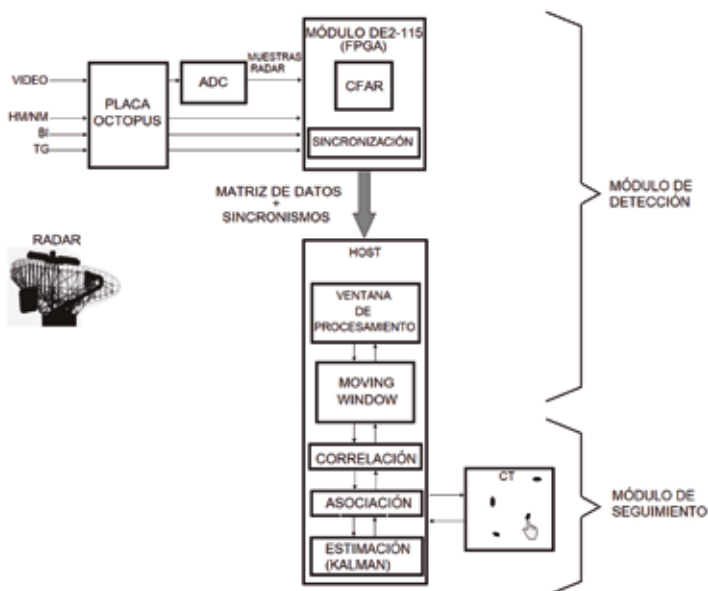


Figura 2. Estructura general del EVRT

Cabe destacar que esta sigue siendo un área de investigación de actualidad [4], y todavía quedan aspectos pendientes del procesamiento de la señal radar que se detallan en este trabajo para futuras investigaciones.

Estructura general del EVRT

Para visualizar los ecos dentro de la consola táctica, se utiliza un conjunto de señales de sincronismo que permiten representar el contenido de la señal de video-radar en forma polar. La arquitectura del EVRT, que se representa en la figura 2, consiste en una estructura sincronizada por medio de estas señales que permiten obtener las referencias necesarias para los procesos del sistema.

Su principio de funcionamiento está basado en la digitalización de la señal de retorno radar y el procesamiento en tiempo real. Los bloques dis-

tivos del sistema son los módulos de detección y seguimiento. En el módulo de detección se realiza la captura de la señal de video-radar por medio de un conversor analógico digital, luego, se procesan las muestras en rango y acimut para distinguir los ecos del ruido y entorno de mar. Estos ecos detectados pasan al módulo de seguimiento, donde mediante algoritmos de correlación, asociación y predicción se determina su trayectoria para, posteriormente, brindar resultados en la consola táctica.

Este sistema es altamente modular, se logró trabajar sobre cada bloque por separado, con entradas/salidas como parámetros, que funcionan en forma independiente, lo que facilitó realizar cada etapa por separado, permitiendo así dar continuidad al desarrollo por diferentes grupos de trabajo.

Con este método, se creó un espacio flexible, que permitió desarrollar y probar distintas alternativas de implementación por separado, con eficacia y eficiencia.

Tener un desarrollo propio permitirá disponer de un sistema adaptable a las necesidades de la Armada Argentina, con la posibilidad de instalarse en varias unidades de mar, de bajo costo de fabricación, instalación y mantenimiento.

Placa Octopus

Se diseñó un circuito de adaptación de niveles de las señales que provienen del radar o de un simu-

lador de señales radar (SIMURAD). El hardware desarrollado tiene el objetivo de adaptar y aislar las señales, con el fin de proteger los equipos involucrados. La placa se denominó "Octopus" ('pulpo' en inglés), por poseer cuatro señales de entrada y cuatro señales de salida: *trigger* (TG), *bearing increment* (BI), *heading mark* o *north mark* (HM/NM) y video-radar.

La Octopus tiene una etapa de adaptación, un buffer de distribución de señales y una etapa de amplificación, con dos circuitos amplificadores de video, uno en modo diferencial y otro en modo común, para adaptar los niveles de la señal de video a las especificaciones de los módulos de las etapas siguientes: el módulo conversor analógico digital y un conversor de barrido radar, que permite visualizar la señal en forma polar en la consola táctica.

A continuación se presenta una descripción de estas señales provenientes del radar:

- » HM/NM. Se debe al movimiento de rotación de la antena y se define como un nivel lógico. De acuerdo a su configuración, puede indicar que la antena alcanza su posición norte (NM), o en el caso de un radar marítimo, indica la proa de la embarcación (HM). Según el tipo de radar, la velocidad de rotación de la antena varía de acuerdo a su aplicación y configuración. Por ejemplo, algunos radares marinos civiles tienen velocidades de rotación entre 22 y 33 rpm, y en algunas embarcaciones de alta velocidad se utilizan velocidades de hasta 45 rpm [2].
- » BI. Es un tren de pulsos producidos por el movimiento angular de la antena. Originalmente proviene de un conjunto de dispositivos mecánicos y electromecánicos que permiten medir el ángulo entre el HM/NM y el objeto de interés [2]. Este tren de pulsos puede tener una resolución de 2.048, 4.096 u 8.192 pulsos por revolución de antena. En la figura 3 se muestra un ejemplo del modo en que se relacionan las señales de HM/NM y BI,

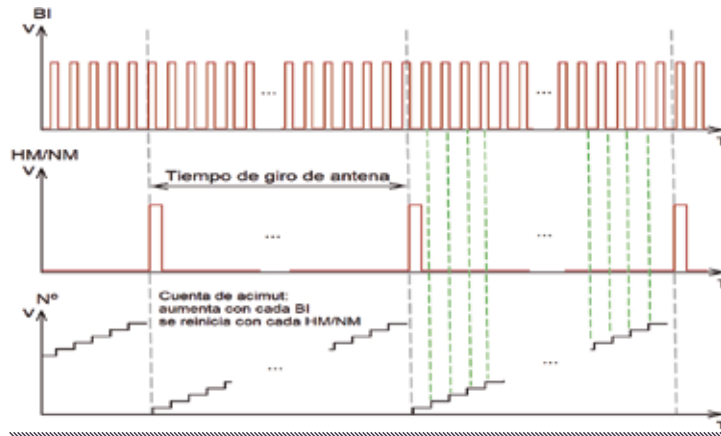


Figura 3. Relación entre las señales de bearing increment y heading mark/north mark

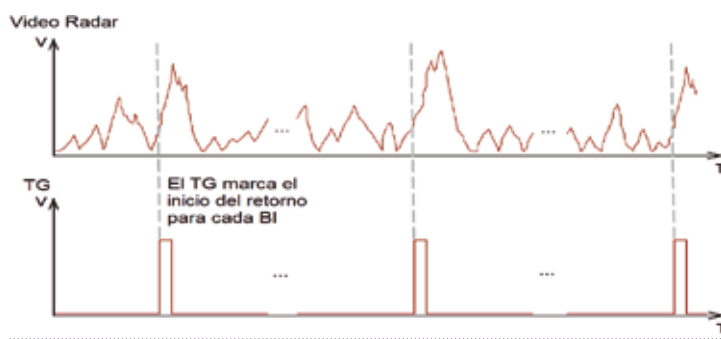


Figura 4. Señales de video y *trigger* del radar

para obtener la cantidad de BI que ocurren durante el tiempo que el radar da un giro de antena completo.

- » TG. El disparo radar consiste en un pulso rectangular muy delgado que indica el instante en el que el radar transmite el pulso de radiofrecuencia. La diferencia de tiempo entre el instante en que se produce cada pulso de *trigger* y el instante en que aparece cada eco, permite al receptor determinar la distancia a la que se encuentran los objetos que lo generan, esta distancia se denomina "rango". La frecuencia de esta señal, también llamada "frecuencia de repetición de pulsos" (PRF,

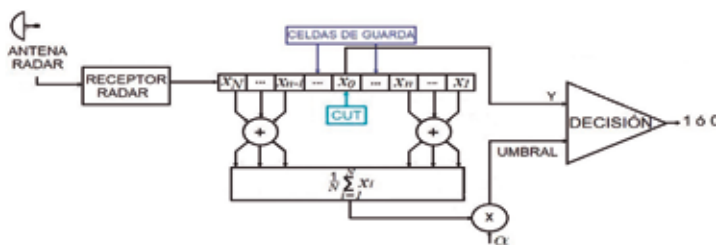


Figura 5. Esquema de celda promedio CFAR

por sus siglas en inglés), es una característica de cada sistema radar. Generalmente, se mide en pulsos por segundo o hertzios. Algunos valores típicos van desde los 850 hasta los 4.000 Hz [2]. El periodo de esta señal define el rango: a mayor periodo, mayor rango de detección. El ancho de este pulso, además, determina la discriminación en rango, o sea la capacidad de distinguir entre dos objetos cercanos.

- » Video-radar. Generada a partir de los ecos electromagnéticos recibidos, es una señal analógica de tensión en función del tiempo, cuya amplitud en un instante de tiempo está estrechamente relacionada con la intensidad en que se recibe el eco. Puede estar contaminada por el entorno marino. Se pueden obtener una o varias señales de video por unidad de ángulo. Esta señal se muestra por medio del conversor analógico digital para su procesamiento digital. La mínima velocidad de muestreo que se puede utilizar dependerá del ancho del pulso de *trigger*. Por ejemplo, si este pulso tiene un ancho de 60 ns, entonces, la mínima velocidad de muestreo deberá ser de 16,67 Msps, para conseguir por lo menos una muestra por pulso. En la figura 4, se puede apreciar la relación entre el *trigger* y la señal de video, donde el *trigger* marca el inicio del retorno para cada *bearing increment*.

Módulo de detección

La función del módulo de detección es detectar la presencia de objetos de interés en la señal de video muestreada. Consta de un proceso de detección en rango y otro en acimut. El proceso de detección en rango se denomina “tasa constante de falsas alarmas” (en adelante, ‘CFAR’, por sus siglas en inglés) y la detección en acimut se lleva a cabo con el proceso de ventana deslizante. La implementación de este módulo consta de los siguientes procesos: CFAR, transmisión de datos hacia el *host* y ventana deslizante.

Respecto de CFAR, el principal desafío que se presenta cuando se diseña un detector radar es distinguir los objetos de interés del entorno en que estos se encuentran inmersos. Mucho se ha estudiado este tema, la función del detector es la de tomar una decisión: determinar si en la celda de rango bajo análisis (CUT, por sus siglas en inglés), hay un objeto de interés, o solamente entorno. El método clásico consiste en establecer un umbral adaptativo que varíe de acuerdo a la energía del entorno contenido en la señal radar. Un algoritmo que se utiliza frecuentemente para realizar esta operación es el esquema CFAR, que estima la energía de entorno en la celda de análisis y ajusta un umbral de detección de acuerdo a la variación en las celdas adyacentes a la celda de análisis. El umbral de detección se calcula de modo tal que el detector mantiene una probabilidad de falsa alarma constante [11].

Un esquema de detección ampliamente estudiado en la literatura radar es el promedio de celdas (CA, por sus siglas en inglés) CFAR. Este procesador establece un umbral de detección por medio de la estimación del nivel medio en una ventana de detección de cierta cantidad de celdas de rango, como puede apreciarse en la figura 5. Este detector es el óptimo (maximiza la probabilidad de detección) en un entorno homogéneo cuando las celdas adyacentes a la celda de análisis contienen muestras de entorno independientes e idénticamente distribuidas caracterizado por

las funciones de densidad de probabilidad exponenciales [11].

Su principio de funcionamiento está basado en la digitalización de la señal de retorno radar y el procesamiento en tiempo real. Los bloques distintivos del sistema son los módulos de detección y seguimiento.

Teniendo en cuenta que, debido a las características del entorno marino y con el advenimiento de los radares de alta resolución, no siempre es posible modelar el entorno mediante las funciones de densidad de probabilidad exponenciales [14]. Por este motivo, se encuentran en la literatura múltiples esquemas de detección propuestos, entre los que se destacan el *greatest off* (GO), *smallest off* (SO), *ordered statistic* (OS), entre otros [16], [18], [4].

En el presente trabajo, se implementó el promedio de celdas CFAR de acuerdo al esquema que se presenta en la figura 5. Se utilizaron dos módulos de desarrollo de la firma Terasic, un conversor analógico digital THDB ADA conectado a un DE2-115, que cuenta con un dispositivo FPGA Cyclone IV de Intel, dentro del cual se implementó la arquitectura del CFAR.

El promedio de celda CFAR se describió mediante registros de desplazamiento de ancho y profundidad parametrizables, por lo que se puede adaptar fácilmente a diferentes velocidades de muestreo, ancho de conversión de la señal de video y profundidad de la ventana de ponderación.

Este esquema de detección, que solamente requiere el cálculo de un promedio, opera bajo el principio de ventana deslizante, según el cual se desplaza el mecanismo detector por toda la zona

de cobertura. Así, se evalúa la existencia de un objeto de interés mediante la comparación del nivel encontrado en la celda de análisis con el umbral de detección, que se calcula como el promedio hallado en las celdas vecinas o de referencia, multiplicado por un factor de escala (α). Las celdas de guarda (las vecinas más cercanas) se remueven del cálculo del promedio para evitar que la detección se desparrame en su vecindad. Si el valor de la celda de análisis (Y) es mayor que el valor de umbral, entonces a la salida se obtiene un '1', lo cual indica que se detectó un objeto de interés. Caso contrario, un '0' indica ausencia de objeto de interés.

Teniendo en cuenta que cada muestra proveniente del conversor analógico digital contiene 14 bits de video digital, a la salida del CFAR, se obtiene un resultado de un solo bit por muestra, que indica la detección o no en la celda de análisis. Este proceso, realizado en el dispositivo FPGA, reduce considerablemente la cantidad de bits a transmitir y procesar, logrando de este modo mejorar notablemente la velocidad de procesamiento del sistema en relación al desarrollo anterior [15].

Respecto de la transmisión de los datos hacia el *host*, para transmitir la salida del CFAR y las señales de sincronismo por Ethernet y, por consiguiente, bajar el ancho de banda requerido para su transmisión, se agrupó la información en palabras de 32 bits que se envían a un *host* por Ethernet, donde continuará el procesamiento utilizando lenguaje Python. Cada palabra debe contener los '1' y '0' de la salida del CFAR y las indicaciones de las señales de sincronismo. En los primeros 19 bits se almacena la salida del CFAR y se lleva una cuenta de la cantidad que ingresa. Las indicaciones de *trigger* y *bearing increment* se codifican, cada una de ellas, en 5 bits de acuerdo a la posición de cuenta, correspondientes a los 19 bits del CFAR. De esta manera, se preserva la relación sincrónica del muestreo de la señal de video y la ocurrencia de las señales de *trigger* y *bearing increment*. Los dos últimos bits se utilizan para

secuenciar las palabras de 32 bits transmitidas con el objetivo de verificar su correcta recepción. Posteriormente, se envía por Ethernet esta palabra, se reinicia la cuenta, se vacía el registro, y se repite el procedimiento. Los datos son recibidos por el *host* y se ordenan en una matriz, denominada "M", donde las filas representan los *triggers* y las columnas, las muestras correspondientes a cada *trigger*.

Este sistema es altamente modular, se logró trabajar sobre cada bloque por separado, con entradas/salidas como parámetros, que funcionan en forma independiente, lo que facilitó realizar cada etapa por separado.

Por su parte, el módulo de ventana deslizante, también denominado "MW", en dos dimensiones, tiene el objetivo de realizar un análisis en acimut, evaluando la cantidad de detecciones que arrojó el CFAR para cada retorno, y filtrando falsas detecciones. Para ello, registra los últimos tres conjuntos de muestras en rango, correspon-

diente a los últimos tres *triggers*. A medida que las muestras del último *trigger* van llegando, analiza la cantidad de '1' provenientes del CFAR, dentro de una ventana de 3x3 (tres muestras en rango por tres muestras en acimut). Este algoritmo utiliza un criterio de tipo m/n, para decidir si se detectó un objeto de interés [5], o sea, si de 'm' muestras, 'n' son '1', se determina la presencia de un *plot*. El conjunto de *plots* detectados se transfiere al módulo de seguimiento.

Módulo de seguimiento

El desarrollo del módulo de seguimiento se realiza en lenguaje Python/NumPy en el *host*. El proceso se inicia desde la consola táctica, por un operador, quien selecciona un eco con el *mouse* o desde una pantalla táctil. Esta selección da inicio al proceso de seguimiento automático, que se realiza del siguiente modo: una vez que el operador envía su solicitud, comienza un proceso iterativo, donde se localiza el objeto, se calcula su velocidad, aceleración, se estima su trayectoria, etc. Este proceso abre una ventana de datos centrada en el punto de inicio del seguimiento o en la última posición del eco. Posteriormente, se predice su próxima ubicación, para desplazar la ventana a esta nueva posición, donde se analizan

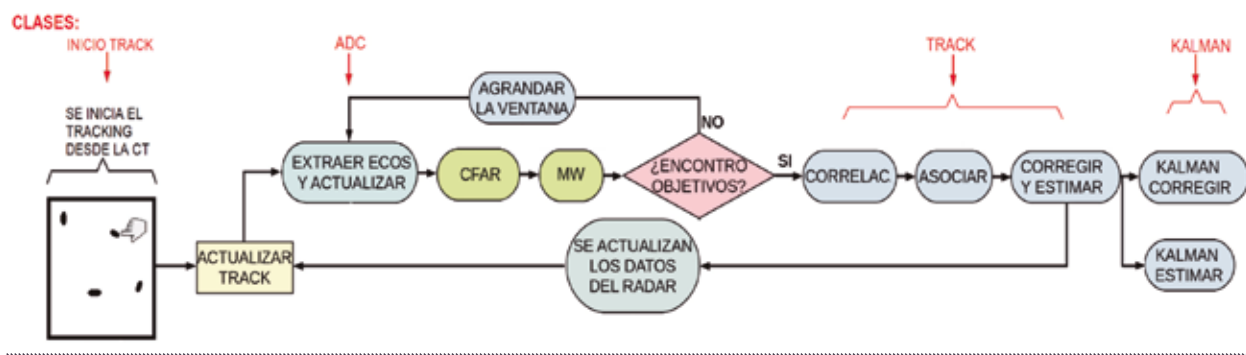


Figura 6. Estructura del software

los datos provenientes del detector en el siguiente giro del radar. Si no se encuentra la detección del eco dentro de esta ventana, se agranda y se vuelven a solicitar los datos al proceso de detección. La ventana se agranda en cada giro del radar hasta un nivel máximo, cuando se consigue este máximo sin detectar un objeto, el objeto se da por perdido, pero en el caso en que la detección sea positiva, se incorporan los ecos a la matriz de correlación y se asocian entre sí, luego, se predice su nueva ubicación.

Dentro de este módulo, se pueden distinguir los bloques, definidos como "clases". El modelado en clases de todo el sistema permite una plataforma de desarrollo y verificación de conceptos Agile [1]. La utilización de Python y cocotb [3] ha posibilitado técnicas de simulación de *hardware-in-the-loop* flexibles para la evaluación y análisis de diferentes métodos de procesamiento en hardware [17]. En la figura 6 se puede apreciar un diagrama que muestra en forma general la estructura del software.

A continuación se describen las clases que componen este módulo:

- » Clase InicioTrack: inicializa el seguimiento y lo actualiza utilizando los datos provenientes de la matriz 'M'. Decide la apertura y tamaño de la ventana, interactúa con la clase Track.

maño de la ventana, interactúa con la clase Track.

- » Clase ADC: se encarga de recibir los nuevos datos CFAR provenientes del conversor analógico digital, también, actualiza y contiene la matriz 'M'.
- » Clase Track: almacena los datos de posición, velocidad y aceleración del eco detectado y al cual se está haciendo el seguimiento. Contiene una instancia de la clase Kalman, que se encarga de hacer las predicciones de las próximas ventanas de búsqueda.
- » Clase Kalman: implementa el filtro homónimo, un algoritmo que permite encontrar los estados de un sistema (posición, velocidad y aceleración), en base a mediciones y estados anteriores. También se encarga de predecir el estado de dichas variables [19].

La separación del proyecto en componentes y una política de documentación permitió sumar alumnos en el marco de sus prácticas profesionales supervisadas.

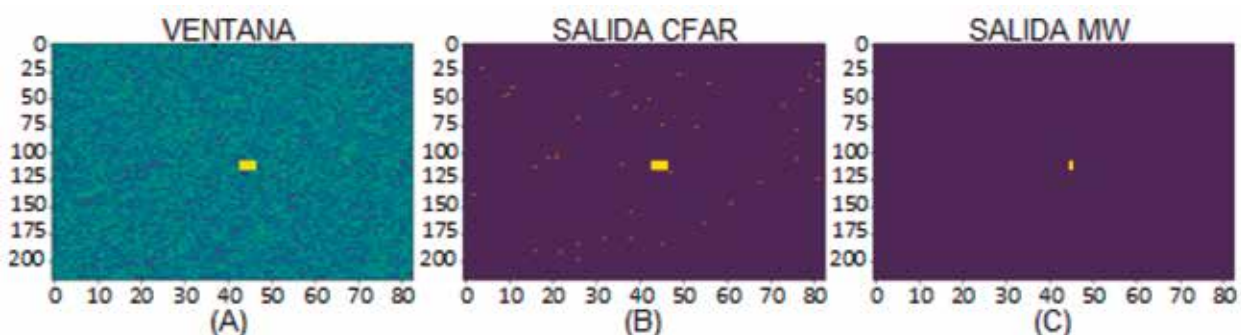


Figura 7. (A) Señal de radar muestreada; (B) salida CFAR; (C) Salida de ventana deslizante

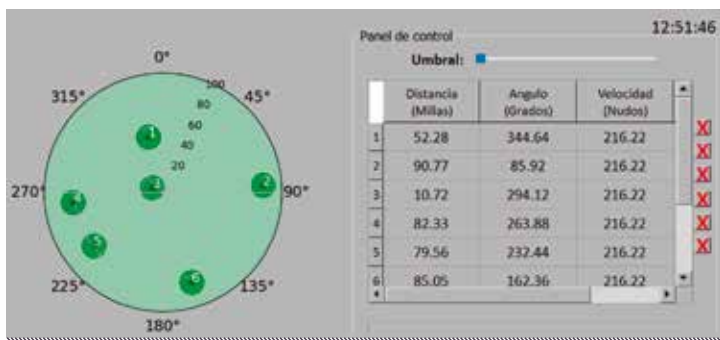


Figura 8. Interfaz gráfica

Interfaz gráfica

Se realizó una interfaz gráfica con el fin de implementar una representación intuitiva en tiempo real de la información mediante un conjunto de herramientas de soporte desplegadas que permiten al operador el fácil entendimiento y respuesta ante eventos que surjan.

Debido a que el rendimiento es un factor importante y altamente requerido en este tipo de sistemas, se realizó un riguroso estudio y análisis de las cantidades de operaciones necesarias entre cada barrido del radar con el fin de brindar toda la información dentro del tiempo disponible. Para ello se optó por una arquitectura multi-hilo que realiza todas las operaciones posibles en paralelo y en forma coordinada.

Esta interfaz se realizó en lenguaje Python, con el objetivo de interactuar con el sistema de una manera amigable. Esta permite que desde la consola táctica, el operador dé inicio al proceso de seguimiento mediante la selección de los ecos.

Algunas de las funciones de esta interfaz son las siguientes:

- » desplegar un gráfico polar que representa la pantalla de barrido radar en la consola táctica, donde se pueden apreciar el desplazamiento de los móviles detectados;

- » marcar sobre la consola táctica el seguimiento de los móviles mediante una identificación y un círculo de color verde que se desplaza sobre cada *plot* seleccionado;
- » Mostrar un conjunto de resultados provenientes de los módulos de detección y seguimiento en una ventana al costado de la pantalla, indicando los datos calculados del *plot* que se detectan: distancia, acimut, velocidad, etc.;
- » incorporar algunos parámetros de configuración del sistema (inicialización, tipos de CFAR, constante multiplicadora de umbral de detección, etc.);
- » facilitar la interacción con otros sistemas, como por ejemplo el conversor de barrido radar.

En el futuro, se espera realizar más ensayos con señales simuladas y reales, como grabaciones de video crudo radar y pruebas a bordo.

Resultados

Para verificar el correcto funcionamiento de los módulos de detección y seguimiento, se simuló una señal radar mediante un conjunto de muestras, modeladas utilizando la PDF Weibull, con un objeto inmerso que responde al modelo de eco Swerling I.

El resultado de este proceso se muestra en la figura 7. Cada figura corresponde a una ventana de búsqueda en rango y acimut de 225 por 85 muestras. En (A) se representa el conjunto de datos simulados; en (B), se observa el resultado del proceso de detección en rango (CFAR), y finalmente en (C) se puede apreciar el resultado de la

detección en acimut luego de aplicar el proceso de ventana deslizante. Este proceso da como resultado el *plot* (indicación del lugar exacto donde se insertó el eco), sobre el que se realizará el seguimiento.

El módulo de seguimiento arroja como resultado la ubicación donde se detecta cada objeto de interés, dando la indicación de su posición en relación al radar, la velocidad, aceleración y trayectoria. Se pueden apreciar algunos de estos resultados en la interfaz gráfica de la figura 8. En esta interfaz se resaltan sobre un gráfico polar los *plots*, especificados con un círculo verde, con un número (ID) asociado. Cada ID corresponde a un seguimiento actual, en tiempo real de acuerdo al movimiento del móvil. El ID del objeto se corresponde con el de la tabla de la derecha, que contiene información relacionada al seguimiento activo, en tiempo real. Allí se vuelvan los resultados del proceso, indicando la posición, la velocidad y la hora en que se produjo la detección del objeto. El sistema también almacena en un archivo del estilo cuaderno de bitácora, toda la información referente a los ecos detectados.

Se encuentran en estudio y en proceso de implementación otros algoritmos, que bajo determinadas condiciones del estado de mar, podrían mejorar los resultados de las etapas de detección, por ejemplo otras estructuras de CFAR, tales como el *OS*, *censored* o *trimmed* CFAR [18], [4]. También se realizaron estudios con técnicas que utilizan redes neuronales [9]. Se investigarán, además, algunos algoritmos que podrían mejorar el seguimiento como por ejemplo, seguimiento de múltiples objetivos o Kalman extendido y *unscented*, filtro de partículas, etc. [7]. Algunas de las limitaciones que podrían presentar estos métodos es la dificultad para desarrollarlos sobre la FPGA en el caso de los procesos de detección, como así también las velocidades de procesamiento de los diferentes algoritmos de seguimiento considerando las limitaciones que estos procesos de tiempo real presentan.

Conclusiones

Este proyecto, se encuentra en una etapa avanzada de su desarrollo. Hasta la fecha, se completó la implementación de cada uno de sus componentes, como así también la integración correspondiente, llegando a realizarse pruebas de desempeño para la detección y seguimiento de ecos fijos y móviles con retornos de prueba. Estos retornos de prueba fueron modelados de acuerdo a la distribución Weibull, para representar el entorno marino con ecos de tipo Swerling I inmersos en algunos de los retornos. Se realizaron, además, pruebas utilizando las señales provenientes de un SIMURAD, diseñado por algunos de los integrantes de este proyecto.

La separación del proyecto en componentes y una política de documentación permitió sumar alumnos en el marco de sus prácticas profesionales supervisadas de carreras tales como Ingeniería Electrónica y en Computación.

Como resultado de este trabajo en equipo se pudo comprobar que esta metodología es muy eficiente, permitió desarrollar simultáneamente diferentes bloques del proyecto, integrando desarrollos basados en temas específicos tales como lenguajes de alto y bajo nivel, como Python, C/C++, lenguajes científicos, sistemas embebidos y diseño de hardware digital.

En el futuro, se espera realizar más ensayos con señales simuladas y reales, como grabaciones de video crudo radar y pruebas a bordo, que permitan cuantificar resultados en un escenario real de funcionamiento. Se aguarda, además, poder utilizar el saber práctico adquirido en nuevos proyectos relacionados con el procesamiento de la señal radar como, por ejemplo, la implementación de un conversor de barrido radar y una nueva versión mejorada de un SIMURAD. ■

Referencias

- [1] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. Martin, S. Mellor, K. Schwaber, J. Sutherland y D. Thomas. "Agile Manifesto," <http://agilemanifesto.org/>, 2001
- [2] A. Bole, B. Dineley and A. Wall. "Radar and Target Tracking for Professional Mariners, Yachtsmen and Users of Marine Radar," Second Edition 2005.
- [3] "Introduction - cocotb 1.0 documentation". [Online]. Available: <https://docs.cocotb.org/en/stable/>
- [4] A. De Maio and A. Aubry, "Radar Detection, Performance Analysis, and CFAR Techniques," in *IEEE Radar Conference (RadarConf)*, Boston, MA, USA, 2019, pp. 1-120, doi: 10.1109/RADAR.2019.8835617
- [5] G. Dillard, "A moving-window detector for binary integration," in *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 2-6, January 1967, doi: 10.1109/TIT.1967.1053967.
- [6] R. Faragher, "Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation [Lecture Notes]," in *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 128-132, Sept. 2012, doi: 10.1109/MSP.2012.2203621.
- [7] Ge, B.; Zhang, H.; Jiang, L.; Li, Z.; Butt, M.M. "Adaptive Unscented Kalman Filter for Target Tracking with Unknown Time-Varying Noise Covariance," in *Sensors* 2019, 19, 1371.
- [8] N. B. Gálvez, J.L. Pasciaroni, and J. E. Cousseau. "CFAR Thresholding Weibull Background", in *Asociación Argentina de Control Automático (AADECA)*, Vol. 1, pp. 1-6, 2006.
- [9] N. B. Galvez, J. E. Cousseau, J. L. Pasciaroni and O. Agamennoni, "Improved Neural Network Based Cfar Detection For Non Homogeneous Background And Multiple Target Situations", in *Latin American Applied Research Buenos Aires: PLAPIQUI (UNS-CONICET)*. 2012 vol. 42 n° 4. issn 0327-0793. pp. 343-350.
- [10] N. B. Gálvez. "Técnicas de Detección Radar en Contextos Marítimos Heterogéneos." Tesis Doctoral, Universidad Nacional del Sur, Bahía Blanca, Buenos Aires, Argentina, 2018.
- [11] P. P. Gandhi and S. A. Kassam, "Analysis of CFAR processors in nonhomogeneous background," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 4, pp. 427-445, July 1988, doi: 10.1109/7.7185
- [12] R.E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," in *Trans. ASME, J. Basic Engineering*, vol 82, , March 1960.
- [13] S. M. Kay. "Fundamental of Statistical Signal Processing, Estimation Theory and Detection Theory". *Prentice Hall Signal Processing Series*, 1993.
- [14] J. R. Machado Fernandez, J. C. and Bacallao Vidal, (2017) "Statistical Distributions for Sea Clutter Modeling: A Survey". *RIELAC*, Vol. XXX-VIII, pp.12-35, ISSN: 1815-5928
- [15] J. Pasciaroni, N. Galvez, J. Cousseau, D. Martínez, C. Geisbuhler and S. Burgos. "Extractor de video-radar Radar (EVR)", in *X Reunion de Trabajo en Procesamiento de la Información y Control (RPIC)*, pp. 1-6, Universidad Nacional de Río Negro, San Carlos de Bariloche, Río Negro, Argentina, 2013.
- [16] H. Rohling, "Radar CFAR Thresholding in Clutter and Multiple Target Situations," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 4, pp. 608-621, July 1983, doi: 10.1109/TAES.1983.309350.
- [17] M. Trapaglia, R. Cayssials, L. D. Pasquale, and E. Ferro, "Flexible Software to Hardware Migration Methodology for FPGA Design and Verification," in *X Southern Conference on Programmable Logic (SPL)*, Apr. 2019, pp. 39-44, doi: 10.1109/SPL.2019.8714377.
- [18] K. D. Ward, R. J. A. Tough and S. Watts "Sea Clutter: Scattering, the K-Distribution and Radar Performance," 2nd ed. London, U.K.: *Inst. Eng. Technol.*, 2013.
- [19] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," TR 95- 041 in *Department of Computer Science, University of North Carolina at Chapel Hill*, 2002