

## Inspección visual subacuática mediante robótica submarina

Marc Carreras, Pere Ridao, Rafael García, David Ribas, Narcís Palomeras

*Instituto de Informática y Aplicaciones, Universidad de Girona, Campus Montilivi, 17071 Girona, Spain.*

### Resumen

Este artículo presenta una aplicación industrial de robótica submarina que consiste en un sistema para realizar inspecciones visuales del fondo subacuático. El sistema consta de un robot submarino para adquirir imágenes visuales a poca distancia (1-2 metros) de la superficie a inspeccionar y un localizador submarino instalado en una embarcación o boyo. Este localizador permite conocer la posición absoluta del robot durante la inspección y se basa en un sistema acústico de tipo USBL (Ultra Short Base Line), una unidad de navegación inercial (INS) y un GPS. Además, el robot tiene su propio sistema de navegación a bordo, basado en EKF, que utiliza un sensor de velocidad basado en efecto Doppler y una INS con un giroscopio de fibra óptica (FOG). La arquitectura de control del robot permite realizar la inspección de forma teleoperada, semi-autónoma o completamente autónoma. El robot puede realizar inspecciones de superficies 2D verticales y horizontales. Una vez adquiridas las imágenes y todos los datos de navegación y percepción, se realiza un proceso fuera de línea de fusión de los datos y procesado de las imágenes que concluye con la generación de un mosaico 2D georeferenciado de la superficie inspeccionada. El artículo detalla las tecnologías desarrolladas y describe una campaña realizada en el embalse de Mequinenza (Aragón) para detectar poblaciones de mejillón cebra. Copyright © 2012 CEA. Publicado por Elsevier España, S.L. Todos los derechos reservados.

**Palabras Clave:** vehículos autónomos, sistemas marinos, control inteligente, navegación de robots, visión de robots

### 1. Introducción

Este artículo presenta una aplicación industrial de robótica submarina que consiste en un sistema para realizar inspecciones visuales del fondo subacuático. El sistema consta de un robot submarino para adquirir imágenes visuales a poca distancia (1-2 metros) de la superficie a inspeccionar y un localizador submarino instalado en una embarcación o boyo. Este localizador permite conocer la posición absoluta del robot durante la inspección y se basa en un sistema localización acústico de tipo USBL (Ultra Short Base Line), una unidad de navegación inercial (INS) y un GPS. Además el robot tiene su propio sistema de navegación a bordo, basado en EKF, que utiliza un sensor de velocidad basado en efecto Doppler y una INS con un giroscopio de fibra óptica (FOG). La arquitectura de control del robot permite realizar la inspección de forma teleoperada, semi-autónoma (con los controles opcionales de profundidad, altura y rumbo) o completamente autónoma. En este último caso, la trayectoria puede definirse en relación a la situación de la superficie a inspeccionar. Para percibir el entorno, el robot dispone de un sensor acústico de rango, un sensor de escaneo de

imágenes acústicas y dos cámaras. El robot puede realizar inspecciones de superficies 2D verticales o horizontales. Una vez adquiridas las imágenes y todos los datos de navegación y percepción, se realiza un proceso offline de fusión de los datos y procesado de las imágenes que concluye con la generación de un mosaico 2D georeferenciado de la superficie inspeccionada. Esta imagen gigante permite observar el fondo marino con alta resolución pudiendo relacionar cada pixel de la imagen con su posición absoluta, y permite también hacer un seguimiento de la inspección comparando la evolución de la superficie inspeccionada en varias observaciones.

Las tecnologías necesarias para llevar a cabo esta aplicación fueron desarrolladas en los proyectos de investigación fundamental del Ministerio de Educación y Ciencia: AIRSUB (DPI 2005-09001) y FOTOGEO (CTM 2004-04205) y sus continuaciones; y la industrialización de la tecnología se está llevando a cabo en el proyecto de la Generalitat de Catalunya "INSPECSUB: Sistema Robótico de Inspección Submarina". Los experimentos llevados a cabo en el proyecto AIRSUB fueron realizados en la pared de una presa de una central hidroeléctrica para inspeccionar el estado del hormigón, (Ridao et al., 2010b,a). Este artículo presenta los resultados obtenidos en otra aplicación de inspección, en este caso la inspección del fondo de un embalse donde habitan poblaciones de mejillón cebra. Esta especie in-

Correos electrónicos: [marc.carreras@udg.edu](mailto:marc.carreras@udg.edu) (Marc Carreras), [pere@eia.udg.edu](mailto:pere@eia.udg.edu) (Pere Ridao), [rafael.garcia@udg.edu](mailto:rafael.garcia@udg.edu) (Rafael García), [dribas@udg.edu](mailto:dribas@udg.edu) (David Ribas), [npalomer@eia.udg.edu](mailto:npalomer@eia.udg.edu) (Narcís Palomeras)

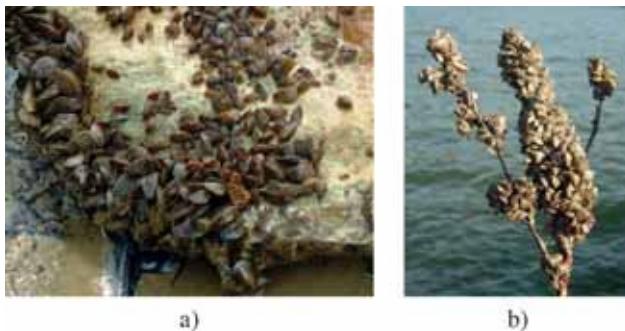


Figura 1: Imágenes de la Confederación Hidrográfica del Ebro en las que se puede apreciar el mejillón cebra proliferando en una roca (a) y en una rama (b).

vasora de aguas interiores provoca graves efectos ecológicos y económicos, motivo por el cual las administraciones y las empresas hidroeléctricas gestoras de las presas están interesadas en hacer inspecciones para valorar la expansión de esta especie (ver Figura 1). El artículo detalla las tecnologías y metodologías utilizadas en una campaña que se realizó en el embalse de Mequinenza (Aragón) con el robot Ictineu (ver Figura 2) fruto de un convenio con una empresa consultora de ecosistemas acuáticos.



Figura 2: Robot Ictineu en el embalse de Mequinenza. Se puede observar la turbidez del agua ya en superficie, así como los brazos que el robot dispone para sujetar los focos que iluminan la escena.

La contribución del artículo recae en los sistemas de navegación y control del vehículo, localización del robot mediante la unidad de superficie y sistema de generación de mosaicos. Aunque existen en la actualidad muchas empresas dedicadas a la fabricación de robots submarinos, ninguna de ellas ofrece la posibilidad de realizar inspecciones visuales de características similares. Más bien proponen la utilización de pequeños robots teleoperados (ROV) que funcionan como cámaras operadas remotamente, poniendo el ojo del inspector en el lugar ocupado tradicionalmente por un buzo profesional. Existe muy poca literatura al respecto y sólo unas pocas empresas/centros de investigación han desarrollado trabajos con un valor añadido en cuanto a la capacidad y calidad de inspección. Entre las más relevantes podemos citar los trabajos desarrollados por el Institut de recherche HydroQuébec (Cote and Lavallee, 1995) en Canadá, que han construido el sistema ROV3. Se trata de un

pequeño ROV posicionado mediante un sistema acústico LBL (Long Base Line). El robot utiliza un sonar multi-beam para evitar colisiones y es capaz de regular la profundidad, altura, orientación y distancia a los objetos. Por supuesto, incorpora cámaras y también un sistema láser para realizar medidas 2D y 3D. El sistema incorpora además un sistema gráfico de realidad virtual que representa, en tiempo real, el movimiento del robot en 3D siendo de gran ayuda en la teleoperación. Por otro lado, la empresa francesa COMEX conjuntamente con la empresa Electricité De France (EDF) llevaron a cabo un proyecto (Poupard et al., 2001) en el cual se utilizó un ROV para la inspección de presas. Para la localización se utilizó un sistema LBL con 5 transpondedores. Para la inspección se disponía de un sistema de captura de vídeo utilizando diversas cámaras así como un sistema de medida 2D basado en la proyección de 2 puntos láser. Finalmente, la empresa británica SonicWorks comercializa un sistema LBL especialmente concebido para la inspección de presas. Un conjunto de 5 transpondedores situados estratégicamente en el embalse, posicionados con GPS y autocalibrados permiten posicionar un ROV con precisión centimétrica. El sistema puede conectarse a un sistema de control que permite gobernar un ROV equipado con cámaras. Aunque el sistema es capaz de realizar un volcado de imágenes georeferenciadas, no es posible encajarlas entre ellas de forma suave y automática.

El artículo se estructura de la siguiente manera, después de la introducción, la sección 2 describe el robot submarino Ictineu que fue utilizado en los experimentos y el sistema de posicionamiento absoluto basado en USBL. La sección 3 detalla la arquitectura de control que controla todos los sistemas del robot (actuadores, sensores, sistemas de percepción, navegación, controladores del vehículo, control basado en comportamientos y controlador de misión). A continuación se detalla el funcionamiento del sistema de navegación en la sección 4, y el sistema de generación de mosaicos visuales en la sección 5. La sección 6 describe los experimentos y resultados obtenidos en la inspección del embalse de Mequinenza. Finalmente, la sección 7 concluye el artículo y apunta las posibilidades de comercialización de esta tecnología que se están contemplando.

## 2. Ictineu AUV

La plataforma robótica utilizada en este proyecto para demostrar las tecnologías desarrolladas fue el robot Ictineu (ver Figura 3.a) junto con una unidad de superficie para localizar el vehículo (ver Figura 3.b).

### 2.1. El robot

El robot Ictineu (Ribas et al., 2007) fue concebido como un vehículo con una estructura abierta, como las comúnmente utilizadas en ROVs comerciales, por su simplicidad, robustez y bajo coste. Estas estructuras son poco eficientes hidrodinámicamente respecto a estructuras cerradas, pero son muy adecuadas para aplicaciones que no requieren movimientos a velocidades altas o durante largas distancias. El chasis del robot está hecho de Delrin, un material plástico similar al nilón que es ligero,

durable y resistente a la corrosión de los líquidos. Otro aspecto del diseño es la concepción modular de los componentes, lo cual simplifica la actualización del vehículo con nuevos equipos y también facilita el acceso a los componentes para su reparación o mantenimiento. Algunos de los módulos (los propulsores y mayoría de sensores) son resistentes al agua y, por lo tanto, se instalan directamente al chasis del robot. Por otro lado, dos cilindros estancos hechos de aluminio alojan los módulos de potencia y procesado mientras que otro más pequeño y de Delrin contiene una unidad giroscópica de fibra óptica.

El módulo de potencia contiene un conjunto de baterías de plomo, el cual suministra 24V 24Ah y puede proporcionar una autonomía de 1 hora. Un convertidor DC-DC proporciona voltaje estabilizado a los diferentes componentes. También, este módulo contiene una placa electrónica para operar los 6 propulsores, los cuales utilizan un bus serie I2C. Esta placa se controla mediante una conexión serie RS-232. Finalmente, un simple circuito con relés conmuta entre la energía de las baterías y la externa proveniente de un cable umbilical, según se desee operar el vehículo de forma autónoma o mediante cable. El módulo de procesado tiene dos PCs, uno para el control y otro para procesado de imágenes visuales y acústicas. Estos PCs se conectan entre ellos y con el exterior (a través del cable umbilical) mediante un switch de 100MBs. El PC de control es un AMD GEODE-300MHz de formato PC104 alimentado por una fuente de alimentación de 50W. El segundo PC es un Via C3 1GHz Pentium de formato mini-ITX.

El movimiento del robot se consigue mediante 6 propulsores SBT150 de la marca Seabotix. Sus reducidas dimensiones y peso, junto con una propulsión de unos 22N hace de ellos una buena opción para vehículos pequeños. En cuanto a su disposición, 4 de ellos están situados en el plano horizontal en una configuración de rombo que permite la propulsión omnidiagonal en el plano horizontal. Así pues, es posible mover simultáneamente el robot en los grados de libertad lineales de avance (surge) y deriva (sway), y angular de guiñada (yaw). Los otros 2 propulsores están situados en el plano vertical y pueden actuar los grados de libertad de arfada (heave) y balance (roll), aunque sólo el primero se controla debido a la estabilidad pasiva del robot en balance (roll) y cabeceo (pitch), ya que el centro de flotabilidad está por encima del centro de gravedad. Esta estabilidad es el resultado de una distribución precisa de los componentes pesados en la parte inferior del chasis combinado con el efecto de la espuma técnica situada en la parte superior, que con sus 10.5 litros de volumen y un peso de 0.6 Kg proporciona una ligera flotabilidad positiva al vehículo. Así pues, el robot Ictineu tiene cuatro grados de libertad.

Una característica interesante de este vehículo es que puede operar como un ROV (vehículo remotamente operado) o como un AUV (vehículo autónomo). Un cable umbilical opcional se puede conectar a los dos cilindros estancos para proporcionar energía y comunicaciones al vehículo. Este modo de operación es muy útil no sólo para operar Ictineu como ROV sino para trabajar como vehículo autónomo en el laboratorio permitiendo el desarrollo de software y su prueba durante largos períodos de tiempo. Una fuente de alimentación externa de 1200VA, conectada a la red eléctrica o a un generador, alimenta al vehículo

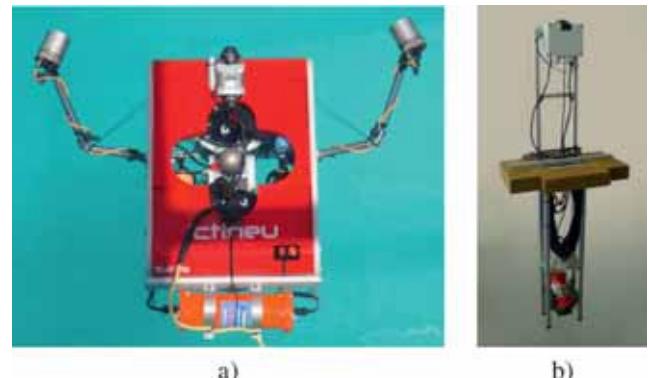


Figura 3: (a) Robot Ictineu con brazos articulados para separar los focos de la cámara. (b) Sistema de posicionamiento absoluto basado en USBL. Dispone también de un DGPS y una unidad de navegación inercial.

con una tensión de 24V. Además, el cable umbilical aloja una conexión estándar Ethernet, permitiendo la transmisión directa de datos desde el exterior al interior y viceversa. Esta conexión de 100Mbps permite operar y supervisar el vehículo, y también transmitir video en tiempo real de las cámaras del vehículo. En modo AUV, el cable umbilical no se utiliza y el vehículo usa las baterías para alimentar todos los sistemas.

El robot Ictineu dispone de un completo conjunto sensorial utilizado para detectar el estado del robot y de su entorno:

**Sonar de imagen** El sonar Tritech Miniking es un pequeño y compacto sonar de imagen de escaneo mecánico (MSIS) diseñado para aplicaciones submarinas como evitación de obstáculos y reconocimiento de objetivos para AUVs y ROVs. Este sonar escanea en un plano 2D rotando un haz en forma de vano en pequeños incrementos de ángulo. El sensor está instalado en la parte frontal y superior del robot para proporcionar una visión clara sin occlusiones del entorno.

**Velocímetro Doppler** El SonTek Argonaut DVL es un sensor especialmente diseñado para aplicaciones ROV/AUV que estima la velocidad del vehículo respecto al fondo y las corrientes del océano utilizando un preciso sistema que se basa en el efecto Doppler. El sensor está equipado con sensores adicionales de: orientación, inclinación, presión y temperatura.

**Unidad inercial con FOG** El sensor Tritech iGC combina 3 giroscopios, 3 acelerómetros y 3 magnetómetros para medir la orientación e inclinación del vehículo. Está diseñado para aplicaciones con AUVs, proporcionando toda la información esencial de movimiento. Este sensor combinado con el Tritech iFG, que es un giroscopio de fibra óptica (FOG), permite tener una estimación del rumbo del robot con muy poca deriva, no afectada por perturbaciones magnéticas. El sensor además de la orientación, estima la velocidad angular y la aceleración lineal, todo ello en 3D.

**Cámaras y focos** El robot está equipado con dos cámaras, modelo Camcolbul2DC, y dos focos halógenos, modelo Aqualux Vario Pro S80. La primera cámara es una cámara en color enfocando la parte frontal del robot para teleoperación. La segunda cámara es de blanco y negro de mucha sensibilidad, y se utiliza para capturar imágenes del fondo marino. Para poder operar en condiciones de poca luz (a mucha profundidad o en aguas muy turbias), el vehículo dispone de 2 focos que se pueden mover y orientar manualmente mediante unos brazos articulados. Para obtener una buena iluminación es necesario separar el ángulo de iluminación de los focos del punto de vista de la cámara.

**Ecosonda** Para evitar colisiones, Ictineu dispone de una ecosonda, modelo Smart sensor de la marca Airmar, que puede instalarse en la dirección frontal o vertical. Este sensor estima acústicamente la distancia entre el robot y un obstáculo frontal o el fondo marino, respectivamente.

**Sensores internos** Finalmente, el robot dispone de sensores de seguridad para medir variables internas como la temperatura, presión y presencia de agua de los cilindros estancos.

**USBL y modem acústico** Desde el exterior, el sistema Linkquest Tracklink 1510BR USBL permite localizar el vehículo de forma absoluta y también mantener una comunicación con el robot mediante un modem acústico. La siguiente sección describe este equipo.

## 2.2. Unidad de superficie

La unidad de superficie está formada por el sensor Linkquest Tracklink 1500 USBL, un DGPS y una unidad de navegación inercial Xsens MTi MRU. El objetivo de este equipo es determinar la posición absoluta del robot en coordenadas de la Tierra. En primer lugar, el sensor USBL realiza una estimación de la posición absoluta del robot respecto al sistema de coordenadas interno de la unidad de superficie USBL. Este equipo utiliza un transceptor en superficie y un transpondedor instalado en el robot. El equipo determina la posición del vehículo calculando el rango y ángulos obtenidos después de transmitir y responder un pulso acústico entre el transceptor y el transpondedor.

Una vez obtenida esta posición relativa entre el robot y la unidad de superficie es necesario calcular la posición respecto a los ejes de la Tierra. Para ello la unidad de navegación inercial se utiliza para determinar la orientación del sistema de coordenadas del USBL, y el DGPS para determinar la posición del origen del sistema. Todos los componentes están fijados en una misma estructura (ver Figura 3.b) la cual, dependiendo del tipo de experimento, puede ser instalada en una embarcación o en una boya de superficie fondeada. La adquisición de los datos se realiza mediante un ordenador externo conectado mediante 3 conexiones serie RS-232. Para integrar la información capturada con los datos del robot, se realiza una sincronización de todos los computadores antes de empezar los experimentos. Es

decir, mediante una conexión de área local, los dos computadores del robot y el computador de la unidad de superficie sincronizan la hora, para después poder agrupar todos los datos capturados en una misma base de tiempo.

Este sistema funciona correctamente siempre que la embarcación o boya no tenga movimientos rotacionales bruscos. Es decir, en situaciones en que las olas sean suaves o nulas. En caso contrario, aunque la unidad INS corrige las inclinaciones, el USBL tomaría una estimación de la posición del robot incorrecta ya que la posición del transceptor habría cambiado desde la transmisión a la recepción. En estas condiciones, los equipos USBL deben instalarse en embarcaciones más estables.

## 3. Arquitectura de control

La arquitectura de control tiene por objetivo realizar la misión que ha sido predefinida por el usuario. La arquitectura es de tipo híbrida (Arkin, 1998), combinando una capa reactiva implementada con comportamientos y una capa de control de misión (ver Figura 4). La capa reactiva, formada por comportamientos, se encuentra englobada en la llamada arquitectura de control software (SCA) que reúne todos los componentes necesarios para realizar una tarea de manera autónoma. La SCA incluye por lo tanto el acceso y control de sensores y actuadores, los sistemas de percepción y navegación, el conjunto de comportamientos activos y su coordinador y, finalmente, un controlador de velocidad que actúa sobre todos los grados de libertad del vehículo. Para realizar un conjunto de ellas, lo que llamamos una misión autónoma, es necesario añadir el Sistema de Control de Misión (MCS). El MCS sigue el estado de la misión mediante la información generada por el sistema de percepción y los comportamientos, y cambia a la siguiente tarea mediante la activación, configuración o desactivación de los comportamientos. Las siguientes secciones detallan los principales componentes de la arquitectura de control

### 3.1. Arquitectura de control software (SCA)

La arquitectura del vehículo tiene que garantizar la funcionalidad del AUV. Desde un punto de vista de implementación, la interfaz de tiempo real POSIX junto con CORBA-RT (Tao, 2003), han sido utilizados para desarrollar la arquitectura como un conjunto de objetos distribuidos con capacidades de tiempo real, garantizando el periodo de ejecución de los procesos como los controladores o la sensorización, aunque en este caso no es muy exigente (máximo 10Hz). Otra parte importante de este sistema es el registro de datos. Un componente de la arquitectura se encarga de registrar mediante archivos los datos de los sensores, los actuadores y de todos los sistemas de percepción y control. Además, todos los computadores de la red se sincronizan previamente, de manera que todos los datos registrados pueden ser relacionados temporalmente. La SCA se divide en tres módulos: módulo de interfaz, módulo de percepción y módulo de control.

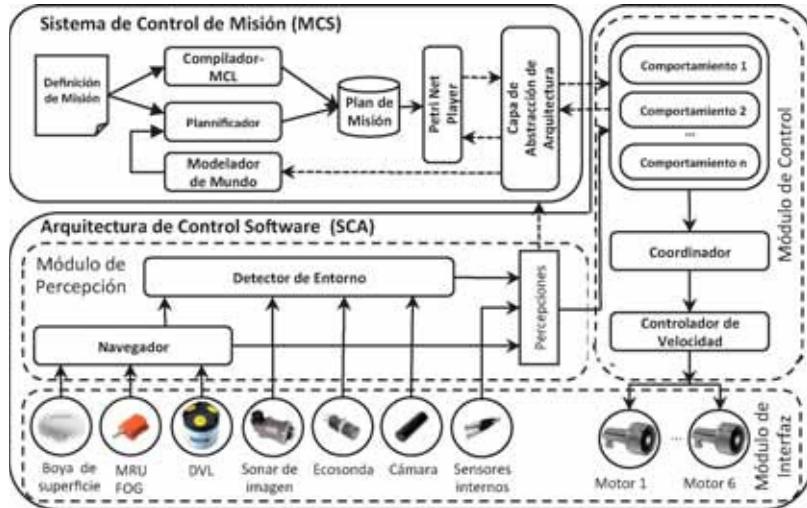


Figura 4: Arquitectura de control del robot Ictineu.

### 3.2. Módulo de interfaz

Este módulo contiene los objetos software que interactúan con el hardware. Los objetos sensores son responsables de leer los datos de los sensores, y los objetos actuadores son responsables de enviar instrucciones a los actuadores. Además, mediante estos objetos es posible configurar todos las variables de estos sistemas y leer la configuración activa. También es posible utilizar una versión virtual de estos objetos para conectar de manera transparente la arquitectura de control con un simulador gráfico a tiempo real, permitiendo simulaciones "hardware in the loop" (Ridao et al., 2004). En este tipo de simulaciones, se ejecutan los programas en los computadores del robot permitiendo encontrar cualquier tipo de problema que pudiera surgir en un experimento real.

### 3.3. Módulo de percepción

Este módulo contiene dos componentes: el navegador y el detector del entorno. El objeto navegador tiene el objetivo de estimar la posición y velocidad del robot (Sección 4) combinando los datos obtenidos por los sensores de navegación. El módulo de control utiliza los datos de la navegación sin necesidad de acceder directamente a los sensores físicos, y manteniendo así la capa de control independiente. En el contexto de este artículo, el detector del entorno tiene por objetivo estimar la altitud del robot y la presencia de objetos en el plano horizontal en el que se encuentra el robot.

### 3.4. Módulo de control

Este módulo actúa como capa reactiva de nuestra arquitectura de control híbrida. Recibe como entrada la información sensorial tratada por el módulo de percepción, manteniendo el módulo independiente de los sensores físicos. Como salida, genera las consignas a seguir por los propulsores, las cuales son mandadas a través del módulo de interfaz. El módulo contiene una librería de comportamientos que pueden ser utilizados para realizar diferentes tareas. Los comportamientos pueden ser habilitados y deshabilitados, y sus parámetros pueden

ser modificados durante la propia ejecución de la misión mediante acciones enviadas por el MCS a través de la Capa de Abstracción de la Arquitectura (AAL). Los comportamientos, a su vez, generaran eventos que indicaran al MCS, por ejemplo, que un objetivo ha sido alcanzado, o que un fallo ha sido detectado en la arquitectura del vehículo. Cada comportamiento activo generará una consigna de movimiento. El coordinador recogerá todas las consignas de los comportamientos activos y generará una consigna coordinada, fruto de las prioridades y niveles de activación de cada comportamiento (Carreras et al., 2007). Es decir, si dos comportamientos (o más) proponen consignas de movimiento incompatibles, el coordinador escogerá una, otra o una mezcla de las dos, según la prioridad de cada comportamiento y su nivel de activación. La consigna de movimiento final será enviada al controlador de velocidad del vehículo, el cual generará las consignas para cada propulsor.

En particular, los siguientes comportamientos fueron desarrollados para realizar trayectorias de inspección del fondo marino:

- **GoTo:** Realiza una trayectoria desde la posición actual a una posición especificada por la posición X, Y y Z, y la orientación o YAW. El comportamiento toma la posición actual del robot del objeto navegador.
- **BottomInspection:** Utilizado para seguir un trayectoria 2D en X y Y de manera omnidiireccional siguiendo un conjunto de puntos destino y manteniendo una orientación predefinida.
- **AchieveAltitudeDist:** Permite al robot mantener una altura predefinida entre el robot y el fondo subacuático.
- **KeepDepth:** Permite al robot mantener una profundidad predefinida.
- **KeepHeading:** Permite al robot mantener una orientación respecto al norte magnético predefinida.

- **Surface:** Realiza un movimiento vertical desde la posición actual hasta la superficie.
- **Teleoperation:** Permite generar consignas de movimiento siguiendo las órdenes de un operador. Sólo puede utilizarse en modo teleoperado (ROV).
- **StartCamera:** Comportamiento utilizado para habilitar o deshabilitar la cámara para tomar imágenes del fondo subacuático.
- **Alarms:** Comportamiento responsable de generar un evento si se detecta una situación anómala en el interior del robot (entrada de agua, temperatura o presión elevada).

### 3.5. Sistema de control de misión (MCS)

El MCS (Palomeras et al., 2009), actúa como una capa de liberativa y de control de ejecución de la arquitectura de control híbrida. En la actualidad no incluye todavía un planificador de la misión. En su lugar, el usuario describe la misión utilizando un lenguaje de alto nivel llamado Lenguaje de Control de Misión (MCL), y entonces la misión se traduce automáticamente a un Sistema de Eventos Discretos (DES) representados por una red de Petri. Esta red define los comportamientos que van a ejecutarse en cada momento dependiendo de los eventos observados.

El MCS ha sido diseñado de manera genérica. Para ello presenta una interfaz que puede ser utilizada en cualquier arquitectura de control de cualquier vehículo, y que está basada en *acciones* y *eventos*. Las acciones son órdenes específicas del MCS a la SCA que indican los comportamientos activos y sus parámetros. Los eventos son hechos detectados por la SCA que se envían al MCS. Entre el MCS y la SCA existe una Capa de Abstracción de la Arquitectura (AAL), que adapta estas *acciones* y *eventos* a cada tipo de arquitectura (ver Figura 4). Esta adaptación consiste básicamente en realizar la interfaz informática entre el MCS y la SCA, y en relacionar las acciones y eventos del MCS con los comportamientos y sistemas de percepción de la SCA. La AAL depende de la SCA específica que se utilice, permitiendo al MCS ser independiente de la arquitectura.

## 4. Módulo de navegación

El módulo de navegación es uno de los elementos clave del sistema. Se utiliza para hacer una estimación de la posición del vehículo durante la ejecución de la misión y también para proporcionar la trayectoria seguida por el robot al sistema de creación de mosaicos visuales. El navegador ejecuta un filtro de Kalman extendido (Kalman, 1960; Thrun et al., 2005), un algoritmo de fusión sensorial estocástico para estimar el estado del vehículo (posición, orientación y velocidad) y su correspondiente incertidumbre, siguiendo un proceso recursivo de dos pasos el cual alterna las predicciones de un modelo con las actualizaciones de las mediciones sensoriales. A continuación, se realizará una descripción detallada de este filtro.

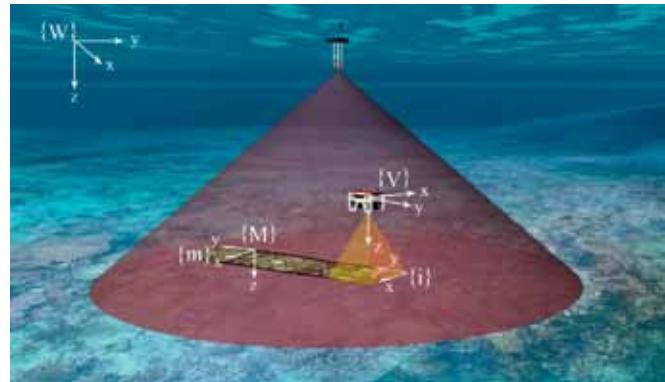


Figura 5: Sistemas de coordenadas involucrados en la inspección y generación del mosaico:  $\{W\}$ , global;  $\{V\}$ , vehículo;  $\{M\}$ , mosaico;  $\{m\}$ , primera imagen (2D);  $\{i\}$ , imagen actual (2D).

### 4.1. Vector de estado

Toda la información estimada por el filtro está contenida en el vector de estado. En nuestra implementación, el vector de estado contiene información acerca de la posición y velocidad del vehículo en el instante  $k$ :

$$\mathbf{x}(k) = [x \ y \ z \ \psi \ u \ v \ w \ r]^T \quad (1)$$

donde, siguiendo la nomenclatura propuesta en (Fossen, 1994), el vector  $[x \ y \ z \ \psi]$  representa la posición y orientación del vehículo en el sistema de coordenadas global  $\{W\}$  (ver Figura 5), mientras  $[u \ v \ w \ r]$  son las velocidades lineales y angulares las cuales están representadas en el sistema de coordenadas del vehículo  $\{V\}$ . Dado que Ictineu es pasivamente estable en *roll* y *pitch*, sus correspondientes ángulos y velocidades angulares no han sido incluidos en el vector de estado.

### 4.2. Inicialización del filtro

Al principio de la misión y antes de empezar a ejecutar el filtro de Kalman, el valor inicial del vector de estado  $\mathbf{x}(0)$  debe ser determinado. Mediante las mediciones de la unidad de superficie, es posible determinar la posición inicial del vehículo con respecto al sistema de coordenadas global  $\{W\}$ . Así pues, la primera medición de la unidad de superficie, y su incertidumbre asociada, será utilizada para definir la posición inicial del vehículo. Dado que  $\{W\}$  está alineado con el norte, la misma estrategia puede ser utilizada para iniciar la orientación, esta vez mediante el compás con FOG del robot. Durante la fase de inicialización, el vehículo se mantiene casi estático. Así pues, las velocidades se inician a cero con un poco de incertidumbre para tener en cuenta posibles perturbaciones. La estimación del vector de estado en el instante 0 es:

$$\hat{\mathbf{x}}(0) = [x_u \ y_u \ z_u \ \psi_c \ 0 \ 0 \ 0 \ 0]^T,$$

$$\mathbf{P}(0) = \begin{bmatrix} \sigma_{Ux}^2 & \sigma_{Uxy} & \sigma_{Uxz} & 0 & 0 & 0 & 0 & 0 \\ \sigma_{Uyx} & \sigma_{Uy}^2 & \sigma_{Uyz} & 0 & 0 & 0 & 0 & 0 \\ \sigma_{Uzx} & \sigma_{Uzy} & \sigma_{Uz}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_c^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_u^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_v^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \sigma_w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma_r^2 \end{bmatrix} \quad (2)$$

donde el subíndice  $U$  indica USBL y  $C$  indica compás. Es importante destacar que la submatriz de covarianza de la posición no es diagonal. Los datos obtenidos de la unidad de superficie son el resultado de combinar la información del GPS, de la INS y de la medición de posición relativa del USBL, lo que significa que la posición global está correlacionada. Estas correlaciones pueden ser determinadas introduciendo la covarianza de cada sensor según las especificaciones del fabricante y realizando las transformaciones necesarias para representar la posición del vehículo en coordenadas globales. Esta transformación se obtiene combinando la lectura del GPS con la medición angular de la INS para obtener la posición y orientación de la unidad de superficie, entonces se introduce la estimación del USBL (en coordenadas esféricas) para obtener la posición del vehículo en coordenadas globales.

#### 4.3. Modelo del sistema

Se utiliza un simple modelo con 4 grados de libertad basado en una cinemática a velocidad constante, sin considerar la aceleración ni las fuerzas generadas por los motores, para predecir como evoluciona el estado desde el instante  $k - 1$  al  $k$ :

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{n}(k-1)),$$

$$\begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \end{bmatrix}_{(k)} = \begin{bmatrix} x + (ut + n_u \frac{t^2}{2})\cos(\psi) - (vt + n_v \frac{t^2}{2})\sin(\psi) \\ y + (ut + n_u \frac{t^2}{2})\sin(\psi) + (vt + n_v \frac{t^2}{2})\cos(\psi) \\ z + wt + n_w \frac{t^2}{2} \\ \psi + rt + n_r \frac{t^2}{2} \\ u + n_u t \\ v + n_v t \\ w + n_w t \\ r + n_r t \end{bmatrix}_{(k-1)} \quad (3)$$

donde  $t$  es el periodo de tiempo y  $\mathbf{n} = [n_u \ n_v \ n_w \ n_r]^T$  es el vector de ruido Gausiano de las aceleraciones, de media igual a cero y valores de covarianza introducidos empíricamente según el funcionamiento observado del modelo de velocidad constante. Para ello se realizan pruebas y se observan las desviaciones de los estados. Estas desviaciones se utilizan en las nuevas ejecuciones del navegador como valores de covarianza. La covarianza del vector  $\mathbf{n}$  se representa con la matriz de ruido del sistema  $\mathbf{Q}$ :

$$E[\mathbf{n}(k)] = \mathbf{0}, \quad E[\mathbf{n}(k)\mathbf{n}(j)^T] = \delta_{kj}\mathbf{Q}(k), \quad (4)$$

$$\mathbf{Q} = \begin{bmatrix} \sigma_{n_v}^2 & 0 & 0 & 0 \\ 0 & \sigma_{n_u}^2 & 0 & 0 \\ 0 & 0 & \sigma_{n_w}^2 & 0 \\ 0 & 0 & 0 & \sigma_{n_r}^2 \end{bmatrix}. \quad (5)$$

El modelo descrito en (3) no es lineal y, por lo tanto, la predicción debe ser realizada con las ecuaciones del filtro extendido de Kalman (EKF) (Thrun et al., 2005).

#### 4.4. Modelo de medición

El vehículo está equipado con un conjunto de sensores que proporcionan observaciones directas de elementos particulares del vector de estado. El modelo lineal general para estas mediciones se escribe de la siguiente forma:

$$\mathbf{z}(k) = \mathbf{Hx}(k|k-1) + \mathbf{m}(k),$$

$$\begin{bmatrix} z_{uD} \\ z_{vD} \\ z_{wD} \\ z_{zP} \\ z_{\psiC} \\ z_{xU} \\ z_{yU} \\ z_{zU} \end{bmatrix}_{(k)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \psi \\ u \\ v \\ w \\ r \end{bmatrix}_{(k)} + \begin{bmatrix} m_{uD} \\ m_{vD} \\ m_{wD} \\ m_{zP} \\ m_{\psiC} \\ m_{xU} \\ m_{yU} \\ m_{zU} \end{bmatrix}_{(k)} \quad (6)$$

donde el subíndice U indica USBL, C compás, D DVL, P sensor de presión,  $\mathbf{z}$  es el vector de medición y  $\mathbf{m}$  representa un vector de ruido blanco Gausiano de media cero que afecta el proceso de observación. La matriz de covarianza del ruido de la medición  $\mathbf{R}$  es:

$$E[\mathbf{m}(k)\mathbf{m}(j)^T] = \delta_{kj}\mathbf{R}(k), \quad (7)$$

$$\mathbf{R}(k) = \begin{bmatrix} \sigma_{Du}^2 & \sigma_{Duv} & \sigma_{Duw} & 0 & 0 & 0 & 0 & 0 \\ \sigma_{Dvu} & \sigma_{Dv}^2 & \sigma_{Dvw} & 0 & 0 & 0 & 0 & 0 \\ \sigma_{Dwu} & \sigma_{Dwv} & \sigma_{Dw}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_p^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_\psi^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{Ux}^2 & \sigma_{Uxy} & \sigma_{Uxz} \\ 0 & 0 & 0 & 0 & 0 & \sigma_{Uyx} & \sigma_{Uy}^2 & \sigma_{Uyz} \\ 0 & 0 & 0 & 0 & 0 & \sigma_{Uzx} & \sigma_{Uzy} & \sigma_{Uz}^2 \end{bmatrix}_{(k)} \quad (8)$$

Los valores de covarianza para la matriz  $\mathbf{R}$  han sido asignados según las especificaciones del fabricante de cada sensor. Es decir, según la precisión que indica el fabricante, se calcula una covarianza en acorde con ella. La forma de la matriz de observación  $\mathbf{H}$  cambia según las mediciones disponibles de los sensores en cada iteración  $k$ .

Cabe mencionar que la sub-matriz de covarianza del DVL no es diagonal. Estas correlaciones existen debido a que las mediciones del DVL no son directamente observables, sino que se calculan proyectando la velocidad del vehículo en los diferentes rayos del sensor. La correlación de las mediciones depende de la geometría de los rayos y, por lo tanto, debe ser determinada en cada caso (Brokloff, 1994). En nuestra implementación particular, estas correlaciones han sido determinadas para la configuración de un DVL de 3 rayos (Ribas et al., 2010). Lo mismo ocurre con la sub-matriz de covarianza del USBL, la cual es completa debido a que la estimación del USBL se representa en coordenadas cartesianas en lugar de coordenadas cilíndricas.

#### 4.5. Suavizado de la trayectoria

Con el fin de generar una trayectoria más precisa y menos ruidosa, el vector de estado puede ser aumentado con un histórico de posiciones (Smith et al., 1990; Leonard and Rikoski, 2001). Ejecutar un filtro de Kalman con estado aumentado permite la propagación de la información sensorial a estados pasados a través de su correlación con la estimación actual. Como resultado de este proceso, sus valores son refinados y se obtiene una estimación de la trayectoria más suave. Para limitar la carga computacional de un vector de estado aumentado, es posible quitar los valores antiguos del estado después de un cierto tiempo. En un estimador como el propuesto aquí, los valores más recientes son los que están más relacionados con el estado actual. Así pues, la información de una medida sensorial se propagará en gran medida a estos valores y, en cambio, casi no tendrá efecto en los valores más antiguos.

La realización de este suavizado no es necesario durante la misión. No obstante, se ha utilizado durante el post procesado de los datos de navegación para generar una mejor estimación de la trayectoria para la construcción de los fotomosaicos.

### 5. Construcción de fotomosaicos del fondo subacuático

Esta sección detalla el proceso necesario para construir los fotomosaicos del fondo subacuático. Esta tarea se realiza una vez finalizado el experimento y para llevarla a cabo son necesarias las imágenes adquiridas junto con la navegación del robot.

#### 5.1. Registro de imágenes consecutivas

Una parte crucial en la creación de mosaicos es el registro de dos o más imágenes de la misma escena tomadas en diferentes momentos y desde diferentes puntos de vista, para alinearlas geométricamente. En este trabajo se ha seguido el método SURF (Bay et al., 2006) para detectar los puntos de correspondencia en todos los pares de imágenes consecutivas, mediante: (1) el uso de un detector Hessiano para identificar los puntos característicos, (2) un descriptor de dichos puntos utilizando la información del gradiente en orientaciones y frecuencias espaciales concretas y (3) un registro basado en la distancia Euclídea entre los descriptores. Por desgracia, debido a la poca variedad de características en las imágenes submarinas, los registros iniciales suelen generar correspondencias incorrectas o valores atípicos. Por lo tanto, se usa un algoritmo de estimación robusta (RANSAC (Fischler and Bolles, 1981)) para rechazar emparejamientos incorrectos. Los emparejamientos restantes se utilizan para calcular la homografía que registra las dos imágenes:

$$\begin{pmatrix} \lambda'x' \\ \lambda'y' \\ \lambda' \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (9)$$

donde los  $h_{ij}$  son los valores de la homografía,  $(x, y)$  es un punto en la primera imagen y  $(x', y')$  es su correspondencia en la segunda imagen.  $\lambda'$  es un factor de escala arbitrario.

La estimación de los valores de la homografía ( $h_{ij}$ ) se obtiene mediante la minimización de una función de coste. Si únicamente tuviéramos 4 correspondencias, podríamos obtener una

solución exacta para  $\mathbf{H}$ . Sin embargo, a pesar de que 4 parejas de puntos son suficientes, en la práctica es aconsejable (debido al ruido inherente) utilizar más puntos. Dado que no siempre existe una homografía que pueda satisfacer  $\{\mathbf{x}'_i = \mathbf{H}\mathbf{x}_i\}$ ,  $i = 1 \dots n$  para todas las correspondencias en el caso de  $n > 4$ , se puede determinar una homografía aproximada solucionando la siguiente ecuación, que tiene la forma  $\mathbf{Ah} = \mathbf{0}$ :

$$\begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1x_1 & -x'_1y_1 - x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1x_1 & -y'_1y_1 - y'_1 \\ & & & & & & \vdots & \\ x_n & y_n & 1 & 0 & 0 & 0 & -x'_nx_n & -x'_ny_n - x'_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -y'_nx_n & -y'_ny_n - y'_n \end{pmatrix} \begin{pmatrix} h_{11} \\ h_{12} \\ \vdots \\ h_{32} \\ h_{33} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} \quad (10)$$

donde  $n$  es el número de correspondencias  $\mathbf{x} \leftrightarrow \mathbf{x}'$ , y  $\mathbf{h} = [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}]^T$  es el vector de incógnitas.

La ecuación (10) tiene más filas que columnas si  $n > 4$ . En ese caso, debemos hallar por mínimos cuadrados la solución  $\mathbf{h}$  que minimiza el vector de residuos  $\|\mathbf{Ah}\|$ . Es importante encontrar una solución diferente de cero, ya que  $\mathbf{h} = \mathbf{0}$  minimizaría de forma trivial  $\|\mathbf{Ah}\|$ . Dicha solución diferente de cero se puede obtener a escala, por lo que es necesario fijar la escala, normalmente imponiendo norma unitaria  $\|\mathbf{h}\| = 1$  o fijando un elemento de  $\mathbf{h}$  (p.ej.,  $h_{33} = 1$ ).

La solución de  $\mathbf{h}$  que minimiza  $\|\mathbf{Ah}\|$  sujeta a  $\|\mathbf{h}\| = 1$  se puede hallar mediante descomposición en valores singulares, siendo dicha solución el vector singular unitario correspondiente al valor singular más pequeño de  $\mathbf{A}$ . En nuestro caso, hemos implementado una solución mediante el análisis de los valores propios. Así, el término de error  $\|\mathbf{Ah}\|$  se puede expresar como

$$\begin{aligned} \|\mathbf{Ah}\| &= (\mathbf{Ah})^2 \\ &= (\mathbf{Ah})^T(\mathbf{Ah}) \\ &= \mathbf{h}^T \mathbf{A}^T \mathbf{Ah} \end{aligned} \quad (11)$$

Derivando la ecuación (11) respecto de  $\mathbf{h}$  e igualándola a cero (para minimizar) nos conduce a la siguiente ecuación:

$$\mathbf{0} = \frac{1}{2}(\mathbf{A}^T \mathbf{A} + (\mathbf{A}^T \mathbf{A})^T)\mathbf{h} \quad (12)$$

Por lo tanto,  $\mathbf{h}$  es igual al vector propio de  $\mathbf{A}^T \mathbf{A}$  que tiene un valor propio más cercano a cero.

#### 5.2. Registro de imágenes no consecutivas

Con el fin de obtener un mosaico coherente y global, el siguiente paso es la detección de imágenes superpuestas no consecutivas. Cuando la trayectoria del vehículo cruza sobre sí misma, se captura la misma área del fondo marino, proporcionando nuevas restricciones para la alineación global del fotomosaico. Los cruces candidatos son identificados utilizando la estimación de movimiento a partir de imágenes consecutivas y, cuando esté disponible, a partir de los datos de navegación del vehículo. Se analizan un conjunto de imágenes cercanas al punto de cruce para identificar pares de imágenes candidatas que luego

se comprobaran. Los pares aceptados, con todos los puntos de correspondencia asociados, son utilizados como entrada para el alineamiento global.

### 5.3. Alineamiento global

Los pequeños errores que se producen durante el registro de la imágenes causan un desalineamiento cuando las imágenes se asignan al sistema de coordenadas global del mosaico  $\{M\}$  (ver Figura 5). Para solventar este problema es necesario realizar un alineamiento global calculado a partir de parámetros de movimiento entre imágenes consecutivas en el tiempo. Primero estimamos la posición y orientación 3D de la cámara a partir del registro de las imágenes consecutivas y componiendo el movimiento asociado. Dado que sólo tenemos una cámara, esta estimación sólo se puede hacer a escala. Mediante un altímetro o mediante la navegación del robot se puede resolver el factor de escala. La distancia de la cámara al robot se utilizará para determinar las dimensiones del plano mosaico, utilizando el modelo geométrico de la cámara. En el trabajo presentado en este artículo, esta distancia se ha extraído de las medidas de la ecosonda. Después, realizamos una optimización que minimiza la función de coste definida como un vector de residuos, tal y como se muestra en la ecuación (13):

$$\text{argmín} \left( \sum_{\substack{1 \leq l \leq m \\ 1 \leq k \leq n}} \left( \|{}^l x_k^l - {}^j H \cdot {}^j x_k^l\|^2 + \|{}^j x_k^l - {}^i H \cdot {}^i x_k^l\|^2 \right) \right) \quad (13)$$

donde  $m$  es el número de correspondencias entre la imagen  $i$  y la imagen  $j$ ,  $n$  es el número de parejas de imágenes que tienen solapamiento y se han registrado correctamente,  $l = \{1..m\}$  y  $k = \{1..n\}$ .  ${}^i x_k^l$  es un punto detectado en la imagen  $i$ , mientras que  ${}^j x_k^l$  es su correspondencia en la imagen  $j$ , expresados en coordenadas homogéneas. Las homografías relativas  ${}^j H$  y  ${}^i H$  entre la imagen  $i$  y la imagen  $j$  se calculan mediante una transformación plana entre cada imagen y el sistema de coordenadas del mosaico (Ferrer et al., 2007).

### 5.4. Detección de lazos e iteración de optimización

Podemos mejorar la alineación del mosaico a través de varias iteraciones de detección de lazo y optimización. Después de cada iteración, la trayectoria resultante optimizada de la cámara se utiliza como punto de partida para una nueva iteración de detección de correspondencias, detección de lazo y optimizado para encontrar nuevas restricciones (es decir, más imágenes superpuestas cerca de los lazos donde la trayectoria del robot cruza una zona ya visitada anteriormente). Las iteraciones se van repitiendo hasta que no se detectan nuevos lazos.

### 5.5. Fundido fotométrico de las imágenes

El mosaico final necesita un procesamiento posterior para compensar las diferencias de exposición entre imágenes y la iluminación no uniforme, especialmente en aguas turbias, como en lagos y presas. De esta forma se evita la visibilidad de las costuras entre imágenes (contorno de unión de dos imágenes en el mosaico). Así, la representación de cada píxel se realiza mediante una función que penaliza la distancia de ese píxel al

centro de la imagen, ya que los píxeles centrales están normalmente mejor expuestos que los píxeles situados en el borde de la imagen. Seguidamente, este mosaico compuesto es transferido al dominio del gradiente, en el que compensamos los gradientes en las zonas correspondientes a las costuras entre imágenes. A continuación se recuperan los valores de luminancia de los campos de gradiente  $G_x$  y  $G_y$ , resolviendo la ecuación de Poisson  $\nabla^2 I = \tilde{G}$  ((Prados, 2007)), donde  $\nabla^2$  es el operador Laplaciano  $\nabla^2 = \frac{\delta^2 I}{\delta x^2} + \frac{\delta^2 I}{\delta y^2}$  y  $\tilde{G}$  es la divergencia del campo de vectores  $G$ , definida como  $\tilde{G} = \frac{\delta G_x}{\delta x} + \frac{\delta G_y}{\delta y}$ . Esta solución nos permite volver al dominio espacial obteniendo un mosaico que combina de forma óptima las contribuciones de las diferentes imágenes, minimizando así el impacto de las costuras entre imágenes. Dado que la resolución del sistema de ecuaciones diferenciales requiere la especificación de condiciones de contorno, se utilizan condiciones de frontera de Neumann, definidas de forma que la derivada en la dirección normal  $\mathbf{n}$  sea cero ( $\nabla I \cdot \mathbf{n} = 0$ ) y resolviendo el sistema mediante un *multigrid Poisson solver* (Kazhdan and Hoppe (2008)).

## 6. Experimentos

Los experimentos presentados en este artículo son el fruto de una colaboración con una empresa consultora que tenía el encargo de la Confederación Hidrográfica del Ebro de estudiar la proliferación del mejillón cebra en el embalse de Mequinenza o mar de Aragón. Este embalse tiene una superficie de 7.540 hectáreas con una anchura media de 600 m y su profundidad llega a superar los 60 m. En los últimos años ha proliferado el mejillón cebra, originario del mar Caspio y el mar Negro, adaptándose con éxito al nuevo hábitat y constituyendo una verdadera plaga. Para realizar dicho estudio la empresa consultora decidió estudiar la posibilidad de caracterizar la poblaciones de mejillón cebra a partir de imágenes adquiridas con un sonar de barrido lateral remolcado desde una embarcación. De esta manera, de manera rápida se podría visualizar la evolución de la población. Para validar esta caracterización les interesaba verificar en ciertos puntos la presencia de esta especie, y es aquí cuando fue necesario el uso de un robot submarino con capacidad de generar fotomosaicos georeferenciados. El robot Ictineu fue empleado en varias inmersiones para realizar estas inspecciones visuales. Toda la campaña, desde las pruebas preliminares a las inmersiones de inspección, se realizó en 4 días.

A diferencia de los experimentos realizados en el proyecto de investigación Airsub de inspección de presas, en el que el robot se lanzaba al agua desde la pared del embalse, en Mequinenza el acceso al lugar de trabajo se hizo mediante una pequeña embarcación de la propia empresa consultora. Una vez cargados todos los equipos en la embarcación, el equipo se dirigió a cada uno de los puntos a inspeccionar guiados por el GPS de la barca y las cartas náuticas.

Durante la inspección, para evitar desplazamientos de la embarcación producidos por el viento, se utilizó un fondeo con varias anclas que los minimizaba. Dado que la unidad de posicionamiento absoluto de superficie iba instalada en la embarcación, era importante evitar estos desplazamientos para no sacar

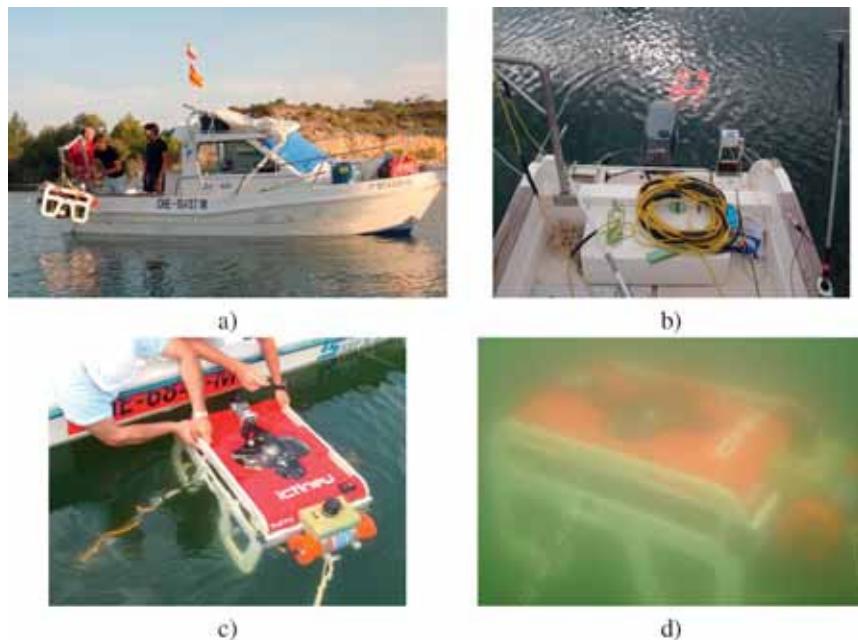


Figura 6: Imágenes de la campaña realizada en el embalse de Mequinenza. a) Embarcación utilizada con el robot Ictineu colgado del pescante. b) Popa de la embarcación con el cable umbilical del robot y la unidad de posicionamiento de superficie a la derecha del motor. c) Robot en el agua durante el proceso de colocar los focos submarinos con sus brazos de sujeción. d) Imagen subacuática en la que se aprecia el robot y la turbidez del agua.

el robot del campo de detección del USBL. Las inspecciones fueron realizadas a un máximo de 30 metros de profundidad e intentando que el punto central de cada mosaico estuviese en la vertical del USBL para trabajar en las mejores condiciones de este equipo. También era importante no tener desplazamientos de la embarcación para no enredar o tirar del cable umbilical con el que trabajaba el robot.

La embarcación disponía de un pequeño pescante con un sistema de poleas que fueron utilizadas para introducir el vehículo al agua e izarlo después (ver Figura 6a). La unidad de superficie, como se ha mencionado, estaba trincada a la embarcación en la aleta de babor (ver Figura 6b, a la derecha del motor). Una vez el robot en el agua, se desplegaban y ajustaban los brazos que sostienen los focos para la iluminación de la escena. Este procedimiento se hacía desde una embarcación auxiliar (ver Figura 6c). Finalmente el robot iniciaba su funcionamiento de inspección. La turbidez del agua (ver Figura 6d) era muy elevada, absorbiendo completamente la luz del Sol ya a los pocos metros. Sólo era posible ver a muy poca distancia del fondo, entre 40cm y 80cm, y siempre con los focos encendidos. Esta turbidez hacía imposible recuperar el robot en caso de fallo (sólo submarinistas profesionales habrían podido recuperarlo) y por lo tanto se optó desde el principio por utilizar el cable umbilical. La utilización del cable también era beneficioso para no depender de la duración de las baterías, para supervisar el funcionamiento y para teleoperar el robot en caso que fuera necesario.

En el interior de la embarcación se instalaron dos ordenadores para operar la unidad de superficie y el robot. Todo el sistema iba alimentado por un generador eléctrico. En las prime-

ras inmersiones se teleoperó el robot hasta llegar al fondo para buscar manualmente poblaciones de mejillón cebra. Ya desde el principio se constató la dificultad del entorno, con un fondo muy escarpado con muchas rocas que provocaban grandes desniveles en relación al tamaño del robot y a la cercana distancia a la que éste debía navegar para visualizar correctamente el fondo. Una vez encontrado el mejillón cebra, se intentaron realizar trayectorias de inspección de manera autónoma con los comportamientos "BottomInspection" y "AchieveAltitudeDist". No obstante, la dificultad del entorno no permitió obtener buenos resultados en este modo. La irregularidad del fondo hacía subir y bajar el robot según la distancia detectada por la echosonda, provocando una secuencia de imágenes que a veces estaban demasiado cercanas y a veces completamente oscuras. Se decidió actuar el robot en modo semi-autónomo, con los comportamientos "KeepDepth" y "KeepHeading" activos, y también el comportamiento "Teleoperation" que tenía prioridad sobre los otros cuando se movía el robot en vertical o se cambiaba la orientación.

Se realizaron inspecciones en 4 zonas distintas en profundidades que iban de 20 a 30 metros. En cada experimento se tomaron los datos de navegación y las imágenes, y se generaron los mosaicos visuales correspondientes. La Figura 7 muestra una parte de una zona inspeccionada en la que se puede apreciar el resultado del mosaico con y sin la fase final de mezclado. La Figura 8 muestra la totalidad de la misma zona, en la que se pueden observar colonias de mejillón cebra. La imagen mide unos 6x3 metros y tiene suficiente resolución para incluso hacer el recuento de los mejillones, o calcular la densidad de la población. Es importante notar el no despreciable relieve de las rocas

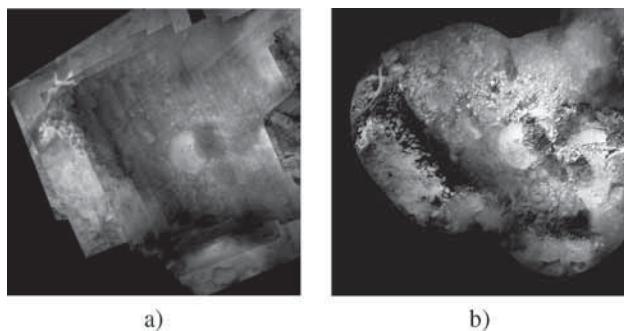


Figura 7: Detalle de mosaico con alineamiento global y optimización. Se puede observar la necesidad de la última fase de mezclado para homogeneizar las imágenes: a) sin mezclado; b) con mezclado.

que dificulta en gran parte el registro de las imágenes. El resultado fue muy satisfactorio teniendo en cuenta la dificultad del entorno. Las tecnologías desarrolladas en este trabajo han permitido acceder a un entorno completamente turbio y peligroso para una persona, generando una imagen de alta resolución que permite ver el fondo como si el agua fuese cristalina.

## 7. Conclusión

Este artículo ha presentado los resultados de dos proyectos de investigación sobre robótica y visión submarina para la realización de inspecciones visuales del fondo subacuático. En particular se ha detallado el vehículo utilizado, su arquitectura de control, el sistema de navegación y el sistema de generación de fotomosaicos. En el apartado de experimentos se ha detallado una campaña realizada en el embalse de Mequinenza para detectar la presencia de poblaciones de mejillón cebra.

En la actualidad esta tecnología está siendo optimizada para simplificar la preparación, ejecución y post-proceso de los datos. Se dispone para ello de un proyecto de valorización de la tecnología, llamado INSPECSUB, financiado por la Generalitat de Catalunya. De esta manera se podrán realizar inspecciones de manera más robusta, en un tiempo menor y con un número de personas menor, haciendo posible su comercialización. En el estado actual, debido a la complejidad del sistema, sólo se contempla la realización de servicios por parte de operarios especializados. Estos operarios son los encargados de: comprobar y preparar los dispositivos del robot, programar la misión, instalar la boya de superficie y el robot, calibrar y ajustar los sistemas in-situ, ejecutar la misión, post-procesar los datos de navegación, ajustar correctamente los algoritmos de generación de mosaicos y ejecutar el sistema de generación de mosaicos. Para una inspección de un día, en la que se pueden cubrir unos 400 metros cuadrados, es necesario un día de preparación previa y otros 3 días de post-procesado, en total una semana. Se está en contacto con empresas e instituciones interesadas en la realización de inspecciones visuales en sitios tan dispares como en presas, pantanos, centrales nucleares, praderas de posidonia, emisarios marinos y yacimientos arqueológicos. En el futuro se

podría también contemplar la posibilidad de integrar esta tecnología en robots submarinos comerciales.

## English Summary

### Underwater Visual Inspection using Robotics

#### Abstract

This article presents an industrial application of underwater robotics for visual inspection of the seabed. The system consists of an underwater robot to acquire visual images at a very short distance (1-2 meters) from the surface to be inspected and an underwater localization system installed in a boat or in a mooring buoy. This system estimates the robot absolute position during the inspection in Earth coordinates using a USBL acoustic locator, an inertial navigation system (INS) and a GPS. Moreover, the robot has its own navigation system onboard, based on EKF, which uses a velocity sensor based on Doppler effect and an INS with a fiber optic gyroscope (FOG). The control architecture of the robot allows the inspection to be in teleoperation, semi-autonomous or autonomous mode. The robot can perform the inspection in 2D vertical or horizontal surfaces. Once the images and all navigation data have been acquired, an offline process is executed for data fusion, and the processing of the images ends up with a 2D georeferenced image mosaic from the inspected area. The paper details the developed technologies and describes a campaign in the Mequinenza reservoir to detect zebra mussel populations.

#### Keywords:

Autonomous vehicles, Marine systems, Intelligent control, Robot navigation, Robot vision.

#### Agradecimientos

Este trabajo ha sido realizado gracias al apoyo del Ministerio de Educación y Ciencia (proyectos DPI2005-09001, CTM2004 04205, DPI2008-06548 y CTM2010-1521), al Centro de Innovación y Desarrollo Empresarial de la Generalitat de Catalunya (Proyecto INSPECSUB) y a la colaboración con la empresa Ecohydros S.L..

#### Referencias

- Arkin, R. C., 1998. Behavior-Based Robotics. The MIT Press, Cambridge, MA, USA.
- Bay, H., Tuytelaars, T., Gool, L. V., may 2006. SURF: Speeded up robust features. In: European Conference on Computer Vision. Graz, Austria, pp. 404–417.
- Brokloff, N. A., September 1994. Matrix algorithm for Doppler sonar navigation. In: Proceedings of the Oceans MTS/IEEE. Vol. 3. Brest, France, pp. 378–383.

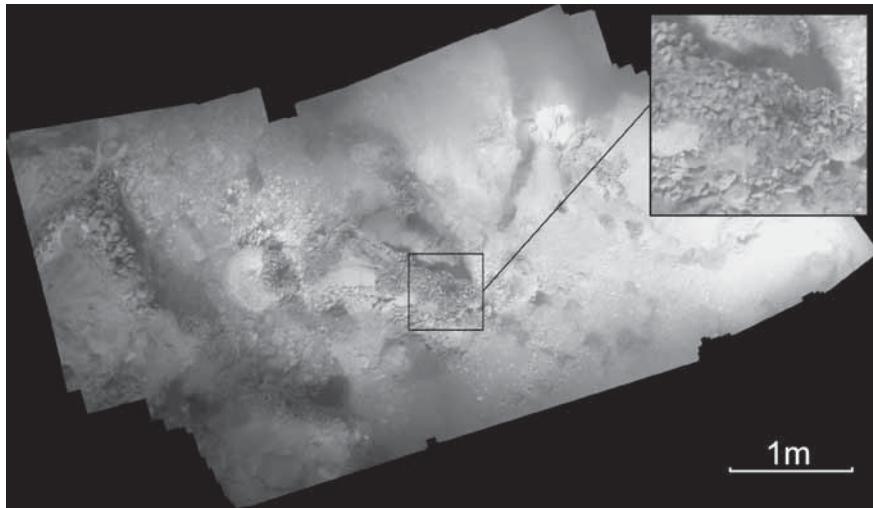


Figura 8: Mosaico generado en una de las inspecciones. Se pueden apreciar las colonias de mejillón cebra y el entorno rocoso del fondo subacuático.

- Carreras, M., Yuh, J., Batlle, J., Ridao, P., 2007. Application of sonql for real-time learning of robot behaviors. *Robotics and Autonomous Systems* 55 (8), 628 – 642.
- Cote, J., Lavallee, J., 1995. Augmented reality graphic interface for upstream dam inspection. In: Proceedings of SPIE Telemanipulator and Telepresence Technologies II. Vol. 2590. Philadelphia, PA, pp. 33–39.
- Ferrer, J., Elibol, A., Delaunoy, O., Gracias, N., García, R., November 2007. Large-area photo-mosaics using global alignment and navigation data. In: Proceedings of the Oceans MTS/IEEE. Vancouver, Canada, pp. 1–9.
- Fischler, M. A., Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* 24 (6), 381–395.
- Fossen, T. I., 1994. *Guidance and Control of Ocean Vehicles*. John Wiley & Sons Ltd.
- Kalman, R. E., March 1960. A new approach to linear filtering and prediction problems. *Transactions of the ASME, Journal of Basic Engineering* 82 (Series D), 35–45.
- Kazhdan, M. M., Hoppe, H., 2008. Streaming multigrid for gradient-domain operations on large images. *ACM Trans. Graph.* 27 (3).
- Leonard, J. J., Rikoski, R. J., 2001. Incorporation of delayed decision making into stochastic mapping. Vol. 271. Springer Verlag, pp. 533–542.
- Palomeras, N., Ridao, P., Carreras, M., Silvestre, C., 2009. Using petri nets to specify and execute missions for autonomous underwater vehicles. In: International Conference on Intelligent Robots and Systems. St. Louis, MO, pp. 4439–4444.
- Poupart, M., Benefice, P., Plutarque, M., November 2001. Subacuatic inspections of EDF (Electricité de France) dams. In: OCEANS, 2000. MTS/IEEE Conference and Exhibition. Vol. 2. pp. 939–942.
- Prados, R., May 2007. Image blending techniques and their application in underwater mosaicing. Master's thesis, University of Girona.
- Ribas, D., Palmer, N., Ridao, P., Carreras, M., Hernández, E., April 2007. Ictineu AUV wins the first SAUC-E competition. In: Proceedings of the IEEE International Conference on Robotics and Automation. Roma, Italy, pp. 151–156.
- Ribas, D., Ridao, P., Neira, J., 2010. Underwater SLAM for Structured Environments Using an Imaging Sonar. No. 65 in Springer Tracts in Advanced Robotics. Springer Verlag.
- Ridao, P., Batlle, E., Ribas, D., Carreras, M., November 9–12 2004. NEPTUNE: A HIL simulator for multiple UUVs. In: Proceedings of the Oceans MTS/IEEE. Vol. 1. Kobe, Japan, pp. 524–531.
- Ridao, P., Carreras, M., García, R., Ribas, D., Batlle, J., May 2010a. Advancing underwater robotics. *International Water Power & Dam Construction* 62 (5), 40–43.
- Ridao, P., Carreras, M., Ribas, D., García, R., November - December 2010b. Visual inspection of hydroelectric dams using an autonomous underwater vehicle. *Journal of Field Robotics* 27 (6), 759–778.
- Smith, R., Self, M., Cheeseman, P., 1990. Estimating uncertain spatial relationships in robotics. In: Autonomous robot vehicles. Springer-Verlag New York, Inc., New York, NY, USA, pp. 167–193.
- Tao, 2003. *TAO Developer's Guide Version 1.3a*. Vol. 2. Object Computing Inc.
- Thrun, S., Burgard, W., Fox, D., 2005. *Probabilistic Robotics*. The MIT Press.