



Disponible en [www.sciencedirect.com](http://www.sciencedirect.com)



Revista Iberoamericana de Automática e Informática industrial 9 (2012) 69–80



[www.elsevier.es/RIAI](http://www.elsevier.es/RIAI)

## Sistema multi-robot para localización e identificación de vehículos

C. Sagues<sup>a,\*</sup>, A.R. Mosteo<sup>a,b</sup>, D. Tardioli<sup>a,b</sup>, A.C. Murillo<sup>a</sup>, J.L. Villaruel<sup>a</sup>, L. Montano<sup>a</sup>

<sup>a</sup>Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, C/Mariano Esquillor s/n, 50018 Zaragoza, España

<sup>b</sup>Centro Universitario de la Defensa, Academia General Militar, Ctra. de Huesca s/n, 50090 Zaragoza, España

### Resumen

En este trabajo se presenta un sistema multi-robot para localización e identificación de vehículos que están estacionados en un recinto abierto o cerrado. El sistema realiza una planificación a priori y una asignación de tareas a los miembros del equipo optimizando el tiempo de la misión. El equipo de robots está dotado de sensores de visión que permiten la localización de los vehículos y la identificación de su matrícula. El controlador de movimiento de cada robot utiliza un sensor láser para el posicionamiento frente al vehículo a identificar y un sistema de control basado en visión realiza el posicionamiento preciso para la adquisición de la imagen de la matrícula que permite su posterior identificación. El sistema multi-robot dispone de capacidad de comunicaciones entre ellos y con una estación central de mando, con la que se intercambian comandos e incidencias y eventualmente datos, con restricciones de tiempo real. Los sensores utilizados están comercialmente disponibles y los algoritmos han sido desarrollados por el grupo Robótica, Percepción y Tiempo Real de la Universidad de Zaragoza en el marco de proyectos financiados por el Ministerio de Ciencia e Innovación. Integra diversas tecnologías de planificación, navegación, percepción y comunicaciones, adaptadas en el proyecto a la aplicación concreta. Copyright © 2012 CEA. Publicado por Elsevier España, S.L. Todos los derechos reservados.

### Palabras Clave:

Sistemas multi-robot, Robótica móvil, Navegación, Planificación y asignación de tareas, Protocolos de comunicación, Visión por computador, Tiempo real.

### 1. Introducción

La utilización de equipos de robots que cooperan para la realización de tareas ha adquirido un importante auge en la investigación y se ha incrementado el abanico de posibles aplicaciones. Cuando se trata de trabajar en un espacio amplio para realizar una misión, la utilización de múltiples robots coordinados permite aumentar la eficacia y eficiencia de la operación. Una posible aplicación, que se presenta en este artículo, es la localización e identificación de vehículos en un aparcamiento, interior o exterior, en actividades de vigilancia, seguridad o exploración. El hecho de que sea interior o exterior modifica algunos aspectos, fundamentalmente los relativos a la percepción y a las comunicaciones.

La aplicación presentada aquí se enmarca en proyectos desarrollados por el Grupo de Robótica, Percepción y Tiempo Real

del Instituto de Investigación en Ingeniería de Aragón de la Universidad de Zaragoza. Estos proyectos han sido financiados por el Plan Nacional de Investigación del Ministerio de Ciencia e Innovación del Gobierno de España, durante el periodo 2004 a 2009: (1) Técnicas de exploración automatizada en aplicaciones de rescate, EXPRES (Montano (2003)); (2) Robots móviles en red para tareas de servicio y de intervención, NERO (Sagues (2006)).

La principal finalidad de estos proyectos ha sido la investigación en estrategias de exploración para diversas aplicaciones de robótica de intervención y de técnicas multi-robot incluyendo las interacciones entre robots, robots con humanos y con el entorno. Se plantearon como objetivos concretos de investigación: (1) Técnicas de cooperación multi-robot con asignación de tareas basadas en métodos de optimización y heurísticas, (2) Percepción basada en escáner láser y en control visual mediante el emparejamiento de características invariantes, incorporando restricciones geométricas y utilizando información visual del propio robot o de otros agentes perceptuales del sistema. Se han utilizado cámaras panorámicas omnidireccionales para reconocimiento y localización a gran distancia y cámaras convencionales para reconocimiento a corta distancia, y (3) Comuni-

\* Autor en correspondencia.

Correos electrónicos: [csagues@unizar.es](mailto:csagues@unizar.es) (C. Sagues), [amosteo@unizar.es](mailto:amosteo@unizar.es) (A.R. Mosteo), [dantard@unizar.es](mailto:dantard@unizar.es) (D. Tardioli), [acm@unizar.es](mailto:acm@unizar.es) (A.C. Murillo), [jlvillaruel@unizar.es](mailto:jlvillaruel@unizar.es) (J.L. Villaruel), [montano@unizar.es](mailto:montano@unizar.es) (L. Montano)

cación de los robots entre sí y de éstos con un centro de mando, basada en redes ad-hoc móviles que soporten tráfico de tiempo real y con calidad de servicio. En este tipo de sistemas y tareas es fundamental mantener las comunicaciones, de tal manera que, independientemente de la distancia y sin necesidad de utilizar una infraestructura con múltiples puntos de acceso, los robots del equipo sean capaces de enviar su propia información local para la coordinación y la toma de decisiones.

Todas estas técnicas se han integrado en la aplicación de localización e identificación de vehículos que se presenta en este artículo. Los principales resultados y su utilización en la aplicación se presentan en las siguientes secciones. En todos los aspectos abordados ha habido contribuciones científicas, reflejadas en las publicaciones referenciadas, y han sido evaluadas experimentalmente en diversos escenarios y situaciones.

Existen empresas que suministran robots de distintas características para tareas de vigilancia y patrullaje (p.e. Roboserv (2011)). Sus prestaciones mecánicas y eléctricas están comprobadas, de forma que pueden realizar estas tareas en condiciones climatológicas muy distintas. Son capaces de detectar intrusos y condiciones anómalas respecto de las establecidas como normales. Sin embargo todavía no incorporan prestaciones avanzadas para reconocer vehículos y sus matrículas. Algunas compañías suministran robots capaces de realizar tareas de vigilancia en aparcamientos (p.e. MoviRobotics (2010)). En este caso la vigilancia se realiza por rutas preasignadas y un guarda puede supervisar la tarea. En nuestro caso el proceso de asignación se suele realizar a priori, y posteriormente se ejecuta. En caso de que se produzca algún bloqueo durante la ejecución, el sistema puede realizar una nueva asignación. Adicionalmente nuestra propuesta es más ambiciosa desde el punto de vista científico y práctico, ya que se plantean esquemas conceptuales en los que varios robots pueden colaborar en la realización de la tarea.

El presente artículo se estructura de la siguiente forma: En la sección 2 se presenta la aplicación y las características de los robots utilizados en su desarrollo. La siguientes secciones se destinan a describir las técnicas utilizadas en la aplicación: planificación y navegación (Sección 3), localización e identificación de vehículos (Sección 4), detección de matrículas y control de posicionamiento (Sección 5) y comunicaciones en tiempo real (Sección 6). La sección 7 se dedica a describir los experimentos y detalles de implementación. Finalmente en la sección 8 se presentan las conclusiones de este trabajo.

## 2. Descripción de la aplicación

La localización de vehículos en aparcamientos plantea diversos problemas a resolver. Si se trata de áreas extensas, la utilización de varios robots que se coordinen para la misión permite reducir el tiempo de búsqueda, cuestión importante en vigilancia y seguridad. La utilización de robots reduce la necesidad de ubicar gran cantidad de sensores estáticos en todo el entorno, haciendo además más flexible y adaptable la ejecución de la misión. El sistema asegura la exploración completa y continua de todo el escenario no dependiente de la intervención humana durante la actuación. Además es importante mantener la



Figura 1: Vista parcial del aparcamiento en el que se han desarrollado los experimentos de la aplicación.

comunicación entre todos los elementos del sistema cumpliendo requisitos temporales para coordinación, control distribuido y supervisión humana. Así, en todo momento la información captada puede ser recibida con calidad y a tiempo por parte del personal a cargo de la vigilancia. Los aspectos de captura y procesamiento robusto de la información son fundamentales, tanto para la propia navegación autónoma de los robots como para el reconocimiento e identificación de los vehículos buscados.

Se presenta una aplicación que combina todos estos elementos y que, hasta donde nosotros conocemos, no se ha desarrollado comercialmente. El escenario es un aparcamiento exterior con varias calles en el que se desea realizar dos tipos de tareas:

- Explorar todo el aparcamiento, registrando periódicamente las matrículas y, si es posible, otra información adicional de los vehículos (marca, color).
- Localizar uno o varios vehículos específicos, reconocibles por color y matrícula.

Los problemas concretos a resolver son:

- Despliegue óptimo de los robots, incluyendo la obtención de la ubicación, los puntos de acceso iniciales más adecuados y las calles que cada robot ha de recorrer para minimizar el tiempo de la misión.
- Navegación autónoma de los robots en cada calle, asegurando la no colisión con otros vehículos o personas. Se trata de un entorno no estructurado en el que los vehículos no están aparcados perfectamente (más adelante, más atrás, no centrados, etc), hay plazas sin ocupar, las calles pueden no ser uniformes, etc. Esto exige una capacidad de adaptación a estas circunstancias que necesita utilizar sensores embarcados.
- Ubicación frente al vehículo a reconocer de manera que pueda ser identificado de forma robusta.
- Identificación del vehículo y su matrícula en condiciones de iluminación variables (mañana, tarde, sombras, reflejos).
- Mantenimiento permanente de las comunicaciones entre robots y con la estación base, para evitar interrupciones en la información recibida y desconexión o pérdida de la



Figura 2: Equipo de robots con los que se han llevado a cabo las pruebas de campo.

supervisión por parte del responsable de la tarea. Ello lleva consigo la necesidad de realizar una navegación coordinada entre los robots.

El experimento presentado en la sección 7 integra parte de estas técnicas. Se ha realizado en un aparcamiento del Campus Río Ebro de la Universidad de Zaragoza. La figura 1 es una vista aérea parcial del mismo. La figura 2 muestra los actores utilizados en la aplicación: Varios robots móviles y una consola o estación base, comunicados entre sí inalámbricamente mediante una red móvil ad-hoc.

### 3. Planificación y Navegación

#### 3.1. Planificación y asignación de tareas

El despliegue autónomo de un equipo multi-robot para la localización de vehículos en un aparcamiento requiere resolver diferentes problemas de planificación y asignación de tareas de manera que se realice la misión en el menor tiempo posible. Dicho de otra manera, se desea minimizar el tiempo de ejecución de la tarea del robot que más tarde en ejecutarla. A su vez esto implica elegir en primer lugar las posiciones y distribución inicial de despliegue, si es que hay más de una posible, y en segundo lugar cómo distribuir las tareas, en este caso los recorridos y calles, que debe explorar cada robot.

En esta aplicación se ha particularizado una técnica general de asignación de tareas propuesta por Mosteo y Montano (2006). Un grupo de técnicas muy extendido en la planificación y asignación de tareas es la basada en *subastas*. Estas técnicas son eficientes para ser implementadas en tiempo real y obtener soluciones rápidamente, pero no dan en general soluciones óptimas. Por ello se han desarrollado técnicas estocásticas para la planificación y asignación de tareas, que tienen el potencial teórico (si bien con tiempo infinito) de hallar mejores soluciones. Están basadas en un método de optimización heurística, *Simulated Annealing* (SA), (Kirkpatrick et al. (1983)), que ha sido adaptado y aplicado a este problema de planificación y asignación de tareas. Las principales ventajas se pueden resumir en que:

- Obtiene soluciones potencialmente óptimas, que mejoran las que pueden proporcionar las técnicas de *subastas*.
- Permite resolver simultáneamente el problema de la localización inicial óptima de los recursos y de la asignación de tareas para su ejecución.
- Puede ejecutarse en tiempo real.

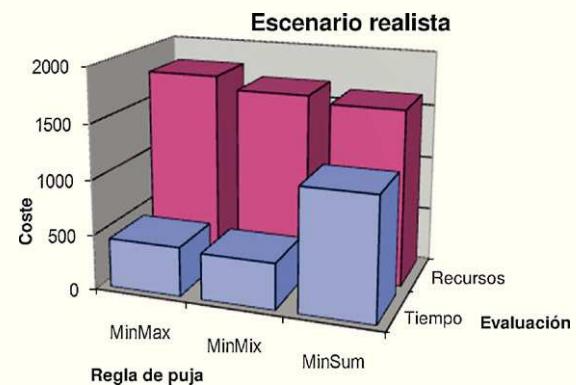


Figura 3: Comparación de costes para los tres criterios.

- Es un algoritmo *anytime*, esto es, que se ejecuta de forma continua y que, considerando el estado actual de la misión, puede encontrar mejoras en todo momento.

Se pueden utilizar diversos tipos de coste a minimizar, dependiendo de la aplicación. Los costes más utilizados son el coste individual máximo ( $C_M$ ) y la suma de los costes individuales ( $C_S$ ):

$$C_M = \max_i C(r_i; a_i),$$

$$C_S = \sum_i C(r_i; a_i),$$

donde  $C(r_i; a_i)$  representa el coste del robot  $r_i$  para ejecutar su asignación  $a_i$ . Los criterios a minimizar son:

$$\text{MINMAX} : A_{MM}^O = \operatorname{argmin}_A C_M, \quad (1)$$

que representa la asignación que minimiza el coste máximo, y

$$\text{MINSUM} : A_{MS}^O = \operatorname{argmin}_A C_S, \quad (2)$$

que representa la asignación que minimiza el coste total  $C_S$ . Habitualmente el primero se asocia a la minimización del tiempo máximo de ejecución de la misión y el segundo al coste total de la misión referido a la suma de desplazamientos realizados por todos los robots. En el caso de la aplicación concreta presentada aquí, el primero es el que tiene mayor sentido, ya que se trata de minimizar el tiempo máximo de la misión.

Se ha definido un coste mixto, que minimiza una combinación lineal de ambos:

$$C_{Mix} = \omega_M C_M + \omega_S C_S,$$

$$\text{MINMIX} : A_{Mix}^O = \operatorname{argmin}_A C_{Mix}, \quad (3)$$

que proporciona soluciones más equilibradas entre los dos costes (Mosteo y Montano (2007a)).

El algoritmo opera sobre un modelo de costes como el del algoritmo del viajante: el coste de cada tarea (calle a explorar) comprende el tiempo desde una posición anterior dada hasta

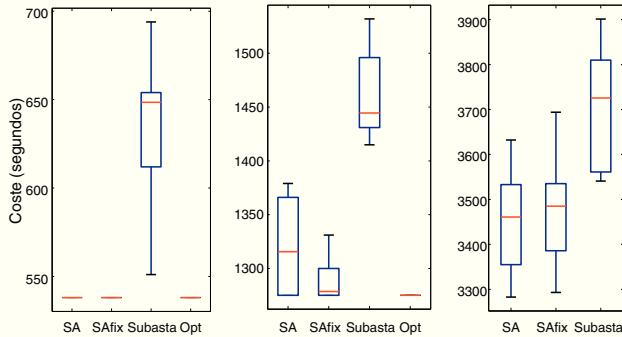


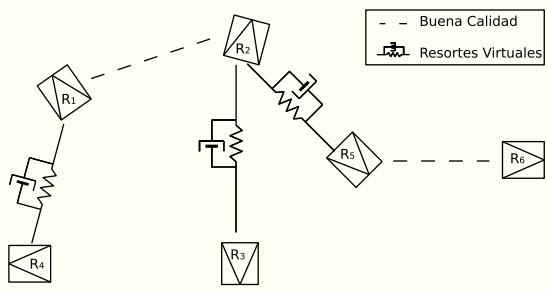
Figura 4: Costes de las soluciones en el problema del aparcamiento para 12, 32 y 43 calles.

el principio de la calle más el tiempo para recorrerla. Para comenzar se construye rápidamente una solución válida aunque probablemente subóptima aplicando inserción voraz. Esta solución sirve de partida para el método estocástico, que funciona aplicando modificaciones a la solución, como por ejemplo intercambiar el orden de dos tareas, trasladar una tarea de un robot a otro, invertir el sentido de una calle, o cambiar el punto de entrada de un robot. La nueva solución se acepta con una probabilidad que depende tanto de su bondad respecto de la anterior como de la temperatura simulada. Las temperaturas elevadas permiten el escape de óptimos locales, mientras que las bajas fuerzan la convergencia a una solución óptima (si bien posiblemente local). El espectro completo de modificaciones, que opera de forma generalizada sobre una descripción jerárquica de tareas, se detalla en Mosteo y Montano (2006). Dicha red jerárquica comprende toda la información necesaria para modelar el problema (tareas, número de robots y función de coste para cada robot), por lo que el algoritmo explora en todo momento el espacio de soluciones del problema.

Se han realizado simulaciones con 100 ejecuciones en diferentes escenarios utilizando los tres criterios y evaluando el coste máximo asociable a tiempo máximo (*Tiempo*) y el coste total (recorrido total, *Recursos*). En este caso, para el término individual  $C(r_i; a_i)$  tiempo y recorrido se consideran proporcionales y por tanto tiene indistintamente ambos significados. La figura 3 representa el resultado de la comparación. La utilización del criterio *MinMix* propuesto establece un buen compromiso frente a los otros criterios, cuando se evalúan el tiempo y el coste total.

La técnica *Simulated Annealing* propuesta se ha comparado con las técnicas clásicas basadas en subastas, en aparcamientos de distinto tamaño. Se han obtenido resultados de simulación para aparcamientos de 12, 32 y 43 calles. Las tareas a asignar a los robots son las diferentes calles. El número de posibles planes es  $2^{12}$ ,  $2^{32}$  y  $2^{43}$  respectivamente, el último considerado como intratable para encontrar la solución óptima.

La figura 4 representa el resultado de la comparación entre técnicas de asignación. En ella se muestran los costes mínimo, máximo, medio, y cuartil sobre 10 ejecuciones de cada algoritmo con 4 robots a los que poder asignar las tareas. Se ha comparado el método *Simulated Annealing* propuesto, en el que se calcula también la distribución inicial de partida de los robots,



con un método de subasta. Dado que este último no puede calcular la distribución de partida (los robots necesitan conocer su ubicación para pujar), se ha prefijado en su caso la mejor calculada por *SA*. Por esta razón se incluye también la evaluación de *SA* utilizando dicha distribución (*SAfix* en la figura 4), por razón de equidad entre los algoritmos, y porque sería el algoritmo a emplear en caso de tener una configuración de partida dada manualmente. Finalmente, se incluye la solución óptima en la comparación.

Se puede comprobar que el método *SA* propuesto, ya sea con configuración inicial prefijada o no, permite obtener mejores prestaciones que el método de subastas en todos los escenarios, y que la solución obtenida se approxima a la óptima en los dos primeros casos. En el escenario más complejo no se ha obtenido la solución óptima debido al coste computacional de su cálculo.

Esta técnica forma parte de un paquete de software de dominio público basado en componentes software que implementa una variedad de técnicas de planificación y asignación de tareas para equipos multi-robot, (Mosteo y Montano (2007b)).

### 3.2. Navegación cooperativa

Una vez realizada la asignación óptima de tareas, cada robot la ejecuta navegando de forma autónoma hacia cada uno de los subobjetivos marcados en la planificación, es decir recorriendo las diferentes calles asignadas. Para realizar una navegación autónoma, los robots utilizan una técnica híbrida que combina planificación de caminos y navegación reactiva. En ella se representan y procesan las partes dinámicas y estáticas del entorno (Montesano et al. (2008)). Ello permite ejecutar la tarea asignada y reaccionar ante posibles vehículos no correctamente aparcados o incluso personas que crucen las calles del aparcamiento, asegurando su integridad. El sensor utilizado, en este caso, es un sensor de rango láser embarcado en los robots.

La navegación autónoma se realiza manteniendo la comunicación entre los robots y con la estación central (Tardioli et al. (2010)). Para ello se ha desarrollado un sistema de comunicaciones multi-robot, explicado en la sección 6, que permite transmitir la información entre robots para la navegación coordinada y, al mismo tiempo, transmitir la información necesaria desde cada robot a la estación central. Para evitar la pérdida de comunicación se ha desarrollado una técnica de navegación cooperativa, que combina la asignación de tareas con una estrategia de control del movimiento basada en una estructura de

resortes virtuales que impiden el avance de los robots en el momento en que la calidad de la señal desciende por debajo de un umbral. La figura 5 muestra la estructura general en la que los resortes-amortiguadores representan que la conexión entre los correspondientes robots ha bajado del umbral de calidad, e impide que los robots se separen más entre sí. En la figura, los resortes virtuales R1-R4, R2-R3 y R2-R5 fuerzan a que las distancias relativas entre ellos se mantengan para evitar una pérdida de comunicación entre ellos y con la base. En este caso el planificador de tareas puede hacer una reasignación para evitar la situación de bloqueo (Mosteo et al. (2008)).

### 3.3. Planificación y ejecución de la navegación

Como se ha presentado en la Sección 2, se contemplan en la aplicación dos posibles misiones: registrar todas las matrículas y localizar un determinado vehículo.

En el primer caso la planificación de la navegación de cada robot se basa en la asignación que el planificador de tareas hace a cada uno de ellos, es decir las calles a recorrer. Un planificador híbrido, que replanifica la trayectoria y ejecuta una navegación reactiva utilizando la técnica Nearness Diagram (*ND*) en cada ciclo de control, a partir del modelo del entorno obtenido por el sensor de rango láser embarcado (Montesano et al. (2008)), realiza la navegación autónoma en las calles asignadas. Mediante la técnica de navegación utilizada el robot es capaz de actuar en un entorno cambiante, dinámico, y por tanto adaptarse a él para evitar obstáculos. Es el encargado de posicionar el robot a una determinada distancia frente a cada vehículo detectado para adquirir la imagen y reconocer la matrícula, independientemente de la ubicación del vehículo en su plaza. Se realiza un procesamiento de la información obtenida a partir de la segmentación de la información de rango para calcular la localización relativa adecuada para el posterior procesamiento de la imagen.

En el segundo caso, el vehículo debe ser localizado en primer lugar. Despues de la asignación de tareas a los robots, cada uno de ellos utiliza la técnica basada en un sistema de visión descrita en la sección 4. Una vez localizado, el planificador de trayectorias híbrido ejecuta un movimiento directo hacia el objetivo, deteniéndose frente al vehículo objetivo utilizando la misma técnica explicada en el caso anterior.

## 4. Identificación basada en visión

Una vez que el sistema multi-robot ha llevado a cabo la fase de planificación de movimientos y la asignación de tareas a cada uno de los robots, para conseguir una óptima inspección del recinto, el equipo comienza la localización de los vehículos en base a su color y la posterior identificación de matrículas, a la vez que realiza la navegación comentada anteriormente.

El proceso seguido por el algoritmo diseñado para la detección de vehículos es el siguiente:

- Procesar la imagen con el algoritmo *Multi-scale retinex with color restoration (MSRCR)* para mejorar la detección de color en pasos posteriores (Sección 4.1).

- Binarizar la imagen usando parámetros de color variables según la saturación del color buscado (Sección 4.2).
- Segmentar esta imagen binaria para reconocer regiones conexas que pueden corresponder al coche que se está buscando (Sección 4.3).

### 4.1. Reconocimiento de vehículos por color

Para la localización del vehículo se usan imágenes omnidireccionales sobre las que se realiza un procesamiento en color. Se pueden plantear distintos métodos de análisis de color tales como los que segmentan en regiones o los que se basan en restricciones de color en imágenes filtradas. Las ideas originales del filtro de procesamiento de imágenes *retinex* fueron exploradas como un modelo de percepción de la luz y el color (Land (1986)). Este filtro se usa para mejorar el reconocimiento del color en situaciones de luminosidad muy variable (Rahman et al. (2006)).

Las imágenes captadas por cada cámara omnidireccional están sujetas a varias circunstancias que hacen complicado el reconocimiento de los colores reales que la componen. Por una parte la propia cámara puede no obtener imágenes de la calidad esperada por factores ópticos (apertura o enfoque, entre otros). Por otra, los escenarios al aire libre presentan diferentes condiciones de iluminación debido a elementos meteorológicos (nubes / sol) o de posicionamiento y luminosidad del sol (mañana / tarde).

Como el objetivo es determinar la presencia de un vehículo del color buscado a la izquierda o derecha del robot, el algoritmo propuesto segmenta en la imagen regiones del color deseado que cumplen ciertas restricciones: de tamaño, posición y situación respecto al robot. Éstas son examinadas y usadas en la siguiente fase de reconocimiento de matrículas.

Se pueden plantear múltiples métodos de análisis de segmentos de color en una imagen, en este caso se consideran dos:

- El primer método consiste en realizar una segmentación de la imagen en regiones con color y textura similar. Se basa en uno de los algoritmos de segmentación de imágenes más utilizados debido a su buena eficiencia (Felzenszwalb y Huttenlocher (2004)).
- El segundo método obtiene una imagen binaria, separando los píxeles que pertenecen al rango de color buscado de los que no, a partir de la imagen original procesada con el filtro *retinex* (Land (1986)).

La segmentación en regiones del primer método puede producir resultados no deseados y poco robustos, dividiendo en varias regiones un mismo objeto de interés o uniendo regiones que corresponden a objetos distintos. Además, pese a ser uno de los algoritmos de segmentación de imagen más eficiente, el tiempo de ejecución comparado con el segundo método es demasiado alto. Por lo tanto, el segundo método resulta más adecuado, por eficiencia y eficacia en el resultado.

La percepción humana de los colores cumple la denominada *constancia de color*, que consiste en la percepción constante de un color de manera independiente a la iluminación del mismo.



Figura 6: Diferencia entre una imagen original (Izda.) y la misma imagen tratada con el filtro *retinex* (Dcha.). Los colores ganan en pureza y la iluminación se homogeneiza.

El primer paso de este proceso intenta mejorar el reconocimiento del color de nuestro algoritmo en situaciones de luminosidad muy variable, mediante el mencionado filtro retinex:

$$R_i(x, y) = \log I_i(x, y) - \log [F(x, y) * I_i(x, y)], \quad (4)$$

con  $R_i(x, y)$  la salida del filtro retinex,  $I_i(x, y)$  el valor de cada canal,  $*$  el operador de convolución y  $F(x, y)$  la función Gaussiana:

$$F(x, y) = K e^{\frac{-(x^2+y^2)}{c^2}}, \quad (5)$$

siendo  $c$  la escala de la Gaussiana.

Dependiendo de la escala se pueden combinar convoluciones Gaussianas de distinta escala para mejorar el resultado multiescala  $M_{r_i}$ :

$$M_{r_i}(x, y) = \sum_{i=1}^N w_n R_{n_i}(x, y), \quad (6)$$

con  $N$  el número de escalas,  $R_{n_i}(x, y)$  el  $i$ -ésimo componente de la escala  $n$  y  $w_n$  el peso asociado a la escala  $n$ . En la figura 6 aparece un ejemplo de aplicación de este filtro.

La salida del algoritmo no es el valor final utilizado, dado que en una segunda fase se debe normalizar la salida al rango de valores de cada canal. Este ajuste se realiza con el valor medio de la salida de *retinex*  $\mu$  y su varianza  $\sigma^2$ , normalizando los valores para estar en el intervalo  $[\mu - \sigma, \mu + \sigma]$ .

#### 4.2. Filtrado de color

A partir de la imagen mejorada se procede a reconocer el color buscado. El análisis del color se realiza en la mayoría de los casos en el espacio de color HSV. El espacio HSV contiene un canal (*Hue*) que corresponde con la idea de color del espectro visible y los otros dos canales (*Saturation* y *Value*) reflejan la pureza y luminosidad del mismo y produce mejores resultados que la comparación con los valores RGB. Sin embargo, el modelo HSV no sirve para comparar colores sin saturación (blancos, negros y grises), dado que no corresponden a un color puro del espectro visible. Por eso, cuando se busca un color de baja saturación, la comparación se realiza en el espacio RGB.

Este filtrado consiste simplemente en asignar uno a los píxeles con un color dentro del intervalo del color buscado, mientras que se asigna valor nulo al resto. Así, al final de este proceso se obtiene una imagen binaria con los píxeles del color deseado en blanco y el resto en negro.

#### 4.3. Reconocimiento de regiones

El último paso del algoritmo consiste en reconocer las regiones conexas en la imagen binaria obtenida tras el filtrado de color, y obtener parámetros de interés tales como su posición o tamaño en la imagen. La búsqueda de regiones conexas (*blobs*) es un problema muy común en visión por computador. Para resolver este paso se utiliza un algoritmo estándar, que consiste en dos pasos. En primer lugar se codifican las líneas horizontales de la imagen con el formato *Run Length Encoding* (RLE). En un segundo paso, se va comprobando la imagen por filas buscando segmentos conexos.

La codificación de datos RLE es un método de compresión en el que secuencias de datos del mismo valor aparecen codificadas junto con el número de repeticiones. Sin embargo, en este caso, el uso de RLE no es para comprimir secuencias de datos, sino para hacer más sencillo el procesado de la conectividad, al reducir la conectividad de todos los píxeles a conectividad entre segmentos. Este paso, por tanto, consiste en el análisis de cada línea horizontal de la imagen para extraer los segmentos conexos de la misma, los cuales se almacenan en una tabla. En el segundo paso del algoritmo se buscan regiones conexas definidas por los segmentos extraídos en el paso anterior. Para ello, se recorre la tabla de segmentos ordenados según la fila a la que pertenecen, creando nuevas regiones, añadiendo segmentos a una región o fusionando regiones según sencillas normas de procesamiento. La figura 7 muestra algunos ejemplos del procesamiento de los segmentos. El procesamiento se hace comparando filas horizontales de dos en dos cada vez. El resultado es una lista de regiones conexas (*blobs*) encontrados en la imagen.

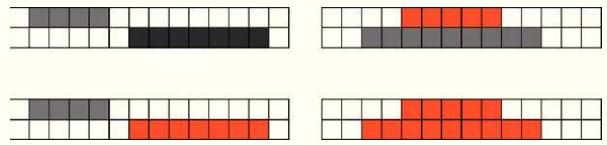


Figura 7: Análisis de segmentos conexos: (Izda.) Se crea una nueva región al encontrar un segmento que no toca con el superior. (Dcha.) Se añade el segmento a la región a la que pertenece el segmento superior con el que toca.

Mientras se hace el procesamiento de regiones conexas, se calcula también el centroide y la distancia al centro de proyección de la cámara. Las regiones con suficiente tamaño, y que cumplen ciertas restricciones de posición se consideran como coches (sabiendo que los robots circulan por el centro de las calles del aparcamiento, se puede establecer una zona en la imagen en la que es factible o no que aparezca un coche).

#### 5. Localización, detección de la placa y control

La aplicación desarrollada en este trabajo se ha centrado en un aparcamiento exterior. La localización continua global de cada robot se realiza utilizando la información GPS y la odometría. El posicionamiento frente al vehículo cuya matrícula se quiere reconocer se hace inicialmente con información del sensor láser y después se aplica una técnica de control visual basada en homografías que permite obtener la imagen siempre de un tamaño similar y en una posición similar. En esta sección se

describen las técnicas para el posicionamiento frente al vehículo y la de reconocimiento de los caracteres de la matrícula.

### 5.1. Localización con escáner láser

Se tiene en cuenta que los vehículos pueden no estar colocados de forma homogénea en sus lugares de aparcamiento y no todos los lugares están ocupados. Para la ubicación precisa del robot frente al vehículo se realiza un procesamiento de la información obtenida a partir de la segmentación de los datos de rango. Se realiza un reconocimiento de patrones de segmentos correspondientes a un vehículo (véase la figura 19) y se posiciona el robot frente a la parte central del segmento frontal, a una determinada distancia que facilita los procesos posteriores.

### 5.2. Reconocimiento de la placa

El objetivo en el último paso para la detección del vehículo es la localización de la placa y la lectura de la matrícula. Esto se hace con una cámara convencional situada en el lateral del robot. Al igual que para las imágenes omnidireccionales, con condiciones de iluminación variables, puede resultar difícil obtener una imagen que permita su perfecta identificación. En sistemas convencionales de detección de matrículas éstas se detectan prácticamente siempre en la misma posición. En nuestro caso, se ha usado adicionalmente un algoritmo de control que permite posicionar al robot en la posición óptima para la lectura de la matrícula. El control preciso del movimiento del robot está basado en homografías (Hartley y Zisserman (2004)).

El proceso de reconocimiento de la matrícula busca en primer lugar todos los cuadriláteros de la imagen que cumplan ciertas restricciones geométricas, y realiza un filtrado de los resultados para buscar el más semejante. Para el procesado de la imagen se hace uso de librerías que permiten llevar a cabo gran parte de tareas de manera sencilla (OpenCV (2010)). Los pasos detallados en la detección de la placa de la matrícula son los siguientes:

1. Se procesa la imagen de manera independiente con filtro Canny (1986) para detectar contornos. Así se obtienen una serie de imágenes sobre las que se van a buscar contornos, tanto interiores como exteriores (Figura 8).

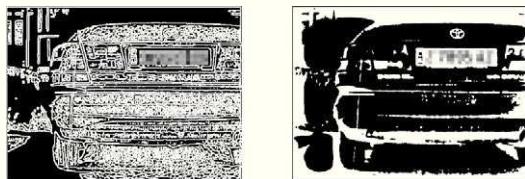


Figura 8: (Izda.) Resultado del filtro Canny. (Dcha.) Umbralización de la imagen para valores de 162 (Entre 0 y 255).

2. Se ajusta un polígono a cada contorno (Douglas y Peucker (1973)). Este polígono se analiza, y solo se tiene en cuenta si cumple cuatro restricciones: (i) Tiene 4 vértices; (ii) El área es mayor que una determinada; (iii) Es convexo; (iv) Los ángulos que forman sus lados son aproximadamente 90°.
3. Se obtiene una lista de rectángulos a procesar (Figura 9).



Figura 9: (Izda.) Rectángulos obtenidos. (Dcha.) Rectángulo identificado como placa de matrícula.

4. Se elige el rectángulo que puede corresponder a la matrícula. Para esto, se eliminan primero todos los cuadrados cuya elongación sea menor de un umbral,

$$\text{Elongación} = \frac{\text{Perímetro}^2}{\text{Área}}. \quad (7)$$

Como una matrícula es un rectángulo de dimensión definida y elongación más o menos constante, valores muy pequeños indican un rectángulo que no corresponde a una matrícula. Tras haber rechazado estos rectángulos, se busca el que más veces ha sido encontrado en las distintas imágenes procesadas (Figura 9).

### 5.3. Control basado en homografías

El objetivo de este control basado en visión es conseguir un posicionamiento preciso frente a la placa de forma que de la imagen obtenida se puedan identificar los caracteres. Se supone que el proceso de planificación ha dejado el robot en una localización adecuada, aunque no lo suficientemente precisa para poder identificar la matrícula. Para la correcta identificación de la matrícula es deseable que esté posicionada en el centro de la imagen, paralela al plano de la cámara y ocupando un área suficiente. Para ello se realiza un movimiento desde una posición en la que se adquiere una imagen hasta una posición en la que se adquiere la imagen objetivo con las características deseadas. Al no disponer de una imagen objetivo, se calcula la disposición deseada de manera aproximada usando las esquinas de la matrícula que permiten crear la imagen objetivo, *virtual*.

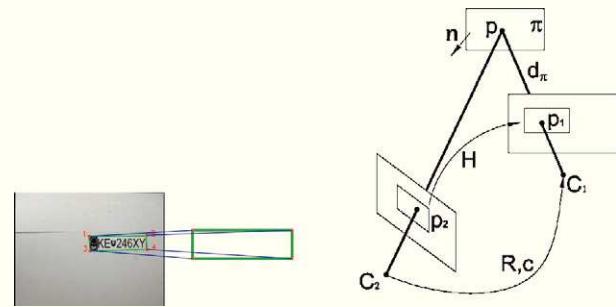


Figura 10: (Izda.) Imagen de la matrícula desde la posición actual del robot y ubicación de la matrícula en la imagen virtual objetivo. (Dcha.) Homografía de un plano entre dos vistas.

Una aproximación clásica para este control es el uso de la geometría epipolar (Lopez-Nicolas et al. (2008)), pero en escenas planas no funciona correctamente, por lo que este control usa el modelo de la homografía. Una homografía es una

transformación que hace corresponder puntos pertenecientes a un plano en el espacio 3D entre dos planos proyectivos (Figura 10). En este caso interesa el plano en el que está la matrícula.

La homografía se puede relacionar con el movimiento de la cámara (Lopez-Nicolas et al. (2010)). El robot puede controlarse modificando su velocidad de rotación ( $w$ ) y translación ( $v$ ). Dado que, como se ha comentado antes, estos valores están relacionados con los valores de la homografía y con la trayectoria deseada, se define una salida de velocidad dependiente de los valores deseados de la homografía ( $H^d$ ) y de los valores actuales de la misma ( $H$ ). Adicionalmente se normaliza con el elemento  $h_{13}$  que es, en este caso, siempre no nulo. En función de las derivadas de dos elementos de la homografía ( $h_{13}$  y  $h_{33}$ ) se tiene,

$$\begin{aligned} \dot{h}_{13} &= \alpha_x h_{33} \omega, \\ \dot{h}_{33} &= \frac{n_z}{d} v - \frac{h_{13}}{\alpha_x} \omega, \end{aligned} \quad (8)$$

siendo  $\alpha_x$  la distancia focal en la dirección  $x$  expresada en píxeles,  $n_z$  la componente  $z$  de la normal al plano de la escena que genera la homografía y  $d$  la distancia al plano desde la referencia global.

Después de obtenida la derivada, se tiene una relación lineal en la entrada/salida del sistema y las ecuaciones necesarias a partir de los elementos de la homografía,  $h_{13}$  y  $h_{33}$ . Se define el sistema de error de seguimiento usando el cambio de variable  $e_{13} = h_{13} - h_{13}^d$  y  $e_{33} = h_{33} - h_{33}^d$ , teniendo en cuenta los valores definidos deseados ( $d$ ), de forma que

$$\begin{pmatrix} \dot{e}_{13} \\ \dot{e}_{33} \end{pmatrix} = \mathbf{L} \begin{pmatrix} v \\ \omega \end{pmatrix} - \begin{pmatrix} \dot{h}_{13}^d \\ \dot{h}_{33}^d \end{pmatrix}, \text{ siendo } \mathbf{L} = \begin{bmatrix} 0 & \alpha_x h_{33} \\ \frac{n_z}{d} & -\frac{h_{13}}{\alpha_x} \end{bmatrix}.$$

Las entradas de control  $(v, w)^T$  que estabilizan el sistema de error de seguimiento son,

$$\begin{pmatrix} v \\ \omega \end{pmatrix} = \mathbf{L}^{-1} \begin{pmatrix} \dot{h}_{13}^d - k_{13}(h_{13} - h_{13}^d) \\ \dot{h}_{33}^d - k_{33}(h_{33} - h_{33}^d) \end{pmatrix}, \quad (9)$$

que se obtienen a partir de los valores de elementos de la homografía antes comentados y los valores deseados  $(h_{13}^d, h_{33}^d)^T$  que representan las trayectorias a seguir. Ganancias  $k_{13}$  y  $k_{33}$  positivas hacen al sistema estable y su valor concreto se elige para obtener un comportamiento adecuado en la respuesta. La matriz de desacoplamiento es:

$$\mathbf{L}^{-1} = \begin{bmatrix} \frac{h_{13}}{\alpha_x^2 h_{33}} \frac{d}{n_z} & \frac{d}{n_z} \\ \frac{1}{\alpha_x h_{33}} & 0 \end{bmatrix}.$$

El control implementado se ve realimentado por nuevas imágenes y nuevas homografías a lo largo del movimiento del robot, hasta que los valores de la homografía indican que se ha logrado el objetivo de posicionar la matrícula en la región deseada de la imagen.

Para comprobar el correcto funcionamiento de los algoritmos se realizaron varias pruebas de control hasta la posición de destino y reconocimiento de la matrícula. El algoritmo de detección de la placa de matrícula funciona correctamente en todo momento aportando los datos necesarios al algoritmo de

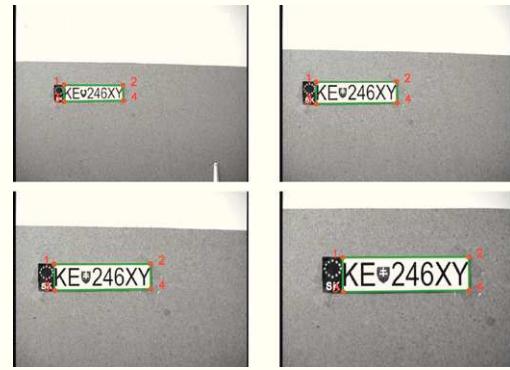


Figura 11: Varias imágenes tomadas durante pruebas de control visual para posicionar la matrícula colocada en un panel. En la imagen inferior izquierda aparece la matrícula en la posición que ha quedado tras ejecutar el algoritmo de control.

control. A su vez, el algoritmo de control permite el posicionamiento correcto del robot frente a la matrícula (Figura 11).

#### 5.4. Reconocimiento de caracteres de las matrículas

Tras el reposicionamiento del robot, el proceso de reconocimiento de matrículas se divide en los siguientes pasos. En primer lugar se delimita la ubicación exacta de la matrícula dentro de la fotografía. Para ello se realiza una detección de regiones mediante métodos gráficos consistentes en localizar el fondo de la matrícula filtrando las zonas blancas y extraer los caracteres mediante un filtrado de las regiones negras resultantes de la umbralización. Una vez delimitada la región, ésta se subdivide progresivamente hasta obtener los caracteres internos mediante la extracción de los blobs correspondientes. Finalmente, mediante la utilización de una red neuronal, se realiza la identificación de los caracteres extraídos en el paso anterior.

Se han utilizado y comparado tres tipos de redes neuronales: *Backpropagation Feedforward Networks* (RPNet), *Radial Basis Function Networks* (RBFNet) y *Learning Vector Quantization Networks* (LVQNet). Los mejores resultados se obtuvieron con redes LVQNet, en torno al 95-96 % de aciertos en buenas condiciones de iluminación. La red tiene 35 neuronas y utiliza la distancia euclídea para llevar a cabo el reconocimiento. Tiene una primera capa de tipo competitivo, que compara los vectores de entrada con las matrices de pesos logradas en el entrenamiento, realizando la clasificación de dichas entradas en los diferentes tipos de patrones a reconocer. La segunda capa está formada por neuronas de activación lineal, cuya función es agrupar los diferentes tipos hallados en el formato de salida deseado.

## 6. Comunicaciones en tiempo real

Una aplicación multi-robot con una central de mando, como la presentada en este trabajo, tiene una serie de requisitos de comunicaciones que se han establecido en:

1. Red conexa en todo momento. La central de mando debe tener información en tiempo real de todos los robots y

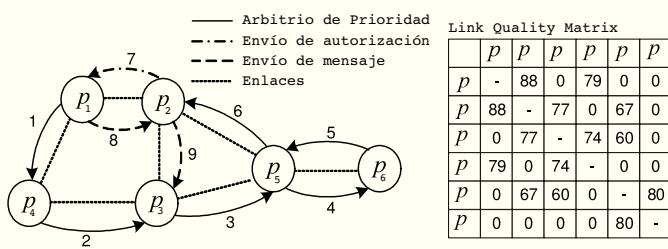


Figura 12: Una situación hipotética descrita por el grafo de la red y la matriz LQM. También se indica la secuencia de saltos del protocolo.

poder emitir órdenes. Asimismo, la coordinación entre los robots necesita de una permanente comunicación.

2. Red ad-hoc multi-salto. Se trata de una aplicación en la que posiblemente no se disponga de una infraestructura de comunicaciones y por lo tanto los diferentes actores (robots y central de mando) se deben auto-organizar en una red ad-hoc. El hecho de que sea multi-salto supone aumentar el rango de actuación, que es imprescindible en grandes aparcamientos.
3. Tráfico de tiempo real. Cualquier control distribuido necesita de comunicaciones en tiempo real. Este es el caso también de equipos de robots.
4. Conocimiento global de la información cinemática y la calidad de los enlaces entre los robots. Son informaciones imprescindibles para el control de movimientos y planificación de la misión de forma que se garantice la conectividad.

Para esta aplicación se ha utilizado el protocolo de comunicaciones RT-WMP (Real Time Wireless Multihop Protocol) que responde a todos los requisitos de comunicaciones comentados (Tardioli y Villarroel (2007)). Por otra parte, el control de la conectividad de la red de comunicaciones se ha realizado mediante una técnica que restringe los movimientos de los robots con ligaduras virtuales basados en la calidad de los enlaces de comunicación (Tardioli et al. (2010)).

El RT-WMP es un protocolo de comunicación para MANETs (Mobile Adhoc NETwork) multi-salto que soporta tráfico de tiempo real. Así, en RT-WMP, el retardo de entrega de un mensaje tiene una duración acotada y conocida. Además el protocolo gestiona prioridades estáticas globales de los mensajes. Con estas dos características las aplicaciones son planificables y analizables, pudiéndose garantizar el cumplimiento de los requisitos temporales.

El RT-WMP permite la comunicación multi-salto para aumentar la cobertura de la red. El protocolo ha sido diseñado para comunicar un grupo relativamente pequeño de nodos móviles (no más de 32). Está basado en un esquema de paso de testigo que permite acceder al medio de forma controlada, característica básica para garantizar tiempos acotados en la entrega de mensajes. El protocolo, que es completamente descentralizado, maneja los cambios de topología a través del intercambio de una matriz que contiene la calidad del enlace entre los nodos.

RT-WMP consiste en tres fases (Figura 12): fase de *arbitraje de prioridad*, en la que los nodos llegan a un consenso acerca

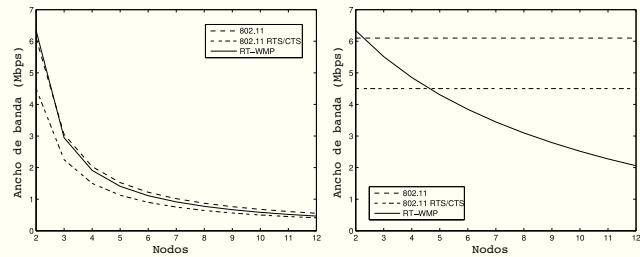


Figura 13: Rendimiento en Mbps de RT-WMP comparado con 802.11. La primera gráfica considera que el nodo emisor y el de destino están a  $n - 1$  saltos (siendo  $n$  el número de nodos). En la segunda gráfica, emisor y destino están a 1 solo salto.

de cuál tiene el mensaje de mayor prioridad, fase de *autorización*, en la que se envía una autorización de transmisión al nodo que posee el mensaje de más prioridad y fase de *transmisión*, en la que se envía el mensaje al nodo destino. Las dos primeras fases se realizan mediante el paso de un testigo o mensaje de control, mientras que la última es el envío efectivo del mensaje con información. El encaminamiento de los mensajes se basa en una matriz LQM (Link Quality Matrix), que mantiene la estimación de la calidad de los enlaces entre los nodos de la red. Esta matriz es compartida entre los nodos de la red. Cada uno de sus elementos es calculado utilizando una media móvil del valor instantáneo de Received Signal Strength Indicator (RSSI) que es una medida de la intensidad de la señal recibida. Los nodos utilizan esta información para buscar el camino más fiable que permita enviar los mensajes de control o de la aplicación. La información contenida en la LQM se comparte entre todos los nodos, ya que viaja con los mensajes de control y llega a todos los miembros de la red en cada fase de arbitraje de prioridad. Los nodos son responsables de actualizar su columna correspondiente de la LQM para informar a los demás nodos de los cambios locales de topología.

RT-WMP está actualmente implementado sobre 802.11 utilizando hardware comercial de bajo coste en los sistemas operativos Linux y MaRTE-OS. Así, RT-WMP permite la utilización de 802.11 en aplicaciones de control distribuido garantizando el tiempo de entrega de los mensajes (tiempo acotado más planificación por prioridades) y además introduce el multi-salto. En la figura 13 puede observarse la sobrecarga que el protocolo introduce sobre 802.11. A pesar de que en el caso de una red hipotéticamente completamente conexa (Figura 13-Dcha) se observa una disminución del rendimiento, éste es en todo caso comparable y asumible teniendo en cuenta los beneficios comentados. En redes de tipo cadena, sin embargo, su comportamiento es muy parecido al protocolo 802.11 (Figura 13-Izda).

El RT-WMP tiene una extensión multicast que permite compartir información entre todos los nodos, a menor coste que con mensajes punto a punto. El protocolo permite el transporte de pequeñas cantidades de datos en la cola de los mensajes de control. En otras palabras, al final de cada mensaje de control se ha añadido un espacio que los nodos pueden utilizar para publicar información que todos los demás nodos pueden leer (Tardioli et al. (2010)). En la presente aplicación, cada nodo publica sus variables cinemáticas. Con esta información y la contenida en

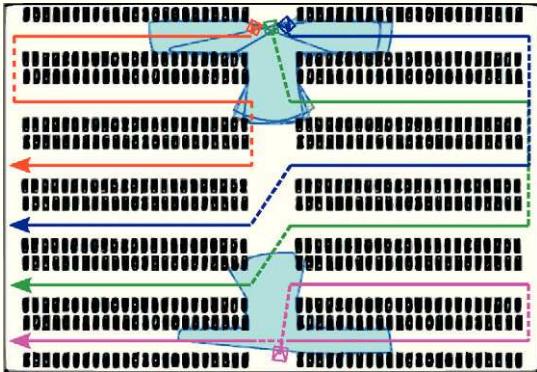


Figura 14: Planificación y asignación de tareas al equipo multi-robot en el aparcamiento. Las calles son las tareas a distribuir entre los robots.

la LQM, los algoritmos de navegación cooperativa explicados actúan para mantener la conectividad de la red de comunicaciones y la asignación de tareas.

## 7. Experimentos para la aplicación

La implementación e integración de las técnicas presentadas se han evaluado en uno de los aparcamientos de la Universidad de Zaragoza en su campus Río Ebro. Se ha hecho público un vídeo e imágenes del experimento (RoPeRT (2009)).

Para la experimentación se utilizaron los PCs empotrados en los robots: VersaLogic VSBC-8 equipados con procesador Pentium III a 800 MHz. Estos tienen instaladas tarjetas WI-FI Cisco 350 Series IEEE 802.11b. Como sistema operativo se utilizó la distribución Debian del sistema operativo Linux con la versión del kernel 2.6.8.

En la figura 14 se representa un ejemplo de ejecución en simulación de la técnica de planificación y navegación en el aparcamiento. En este caso las localizaciones iniciales a resolver son las posibles entradas y las tareas son las calles del aparcamiento, que deben ser recorridas por 4 robots. Esta simulación ha sido realizada en el entorno Player/Stage, y permite obtener a priori una solución para el despliegue de robots en el entorno concreto en el que se desarrolla el experimento. La figura 15 muestra un despliegue inicial de los robots en el aparcamiento. En Mosteo y Montano (2006) se puede encontrar un estudio comparativo cuando se utilizan 2, 4 y 8 robots.



Figura 15: Despliegue inicial coordinado de tres robots en el aparcamiento.

En el experimento se han realizado las dos misiones mencionadas: exploración de todo el aparcamiento registrando las matrículas, localización de un coche objetivo con la cámara omnidireccional y navegación hasta él. En ambos casos se utiliza la técnica híbrida de planificación de trayectoria y navegación reactiva (Sección 3.3) para la ejecución del movimiento en cada robot.

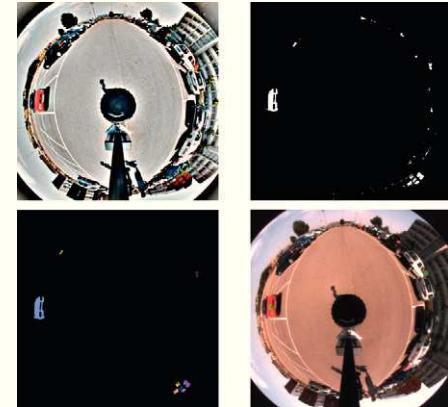


Figura 16: Proceso de reconocimiento de regiones. Se buscan coches de color rojo ( $R = 255, G = 0, B = 0$ ). a) Imagen de entrada procesada con el filtro retinex. b) Filtrado por color. c) Resultado del reconocimiento de regiones, con cada región de un color y las regiones pequeñas eliminadas. d) Imagen original las regiones encontradas marcadas con un círculo verde.

En relación al reconocimiento y localización de vehículos con la cámara omnidireccional, la figura 16 muestra el proceso completo de reconocimiento de vehículos basado en color. En los experimentos reales se ha evaluado la efectividad del algoritmo (teniendo en cuenta aciertos, falsos positivos y falsos negativos). El proceso desarrollado busca un coche a izquierda o derecha con un margen de 80° a los lados de la imagen y da como resultado una imagen en la que se visualizan las regiones reconocidas. A partir de esta salida, se evalúa si la región encontrada corresponde a un coche del color buscado (acertado), a algo que no es un coche del color buscado (falso positivo), y por último se contabilizan los coches del color buscado no encontrados (falsos negativos).

Para el análisis de estos resultados se han usado la *sensibilidad*, que se define como la proporción entre el número de aciertos y el total de coches del color buscado que había en la imagen:

$$Sensibilidad = \frac{Aciertos}{Aciertos + Falsos negativos}, \quad (10)$$

y el *valor predictivo positivo* (*v.p.p.*), que es la proporción de aciertos frente al total de positivos:

$$v.p.p. = \frac{Aciertos}{Aciertos + Falsos positivos}. \quad (11)$$

En la figura 17 aparecen las medidas de sensibilidad y *v.p.p.* para las distintas pruebas. Se pueden observar las mejoras obtenidas al utilizar el filtro retinex, sobre todo para los colores

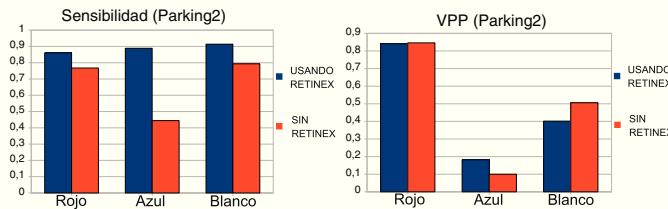


Figura 17: Sensibilidad y v.p.p. de los experimentos de detección de vehículos por color.

Tabla 1: Ancho de banda (Mbps) del protocolo RT-WMP para distinto tamaño (en Bytes) de mensaje y número de robots (Nodos) en la misión.

Tamaño (Bytes)	2 Nodos		3 Nodos		4 Nodos	
	Caso Peor	Caso Mejor	Caso Peor	Caso Mejor	Caso Peor	Caso Mejor
100	0.258	0.284	0.177	0.262	0.110	0.247
256	0.830	0.893	0.387	0.573	0.249	0.478
512	1.47	1.62	0.658	1.05	0.422	0.875
1024	2.06	2.19	0.975	1.54	0.645	1.46
1500	2.28	2.33	1.26	2.27	0.813	1.79

más problemáticos, los azules, ya que son tonos más oscuros que tienden a confundirse más con las sombras o el asfalto.

En la figura 18 se muestra una imagen del robot navegando por una calle mientras toma imágenes de los vehículos. En la figura 19 se representa un esquema de la navegación y detección de los vehículos y una segmentación de la información láser realizada para el posicionamiento del robot frente al vehículo. En la figura 20 se muestran imágenes tomadas desde la posición correcta pero a distintas horas del día, con iluminación diferente. Ello pone de manifiesto algunos de los problemas para el reconocimiento que pueden ocurrir cuando la iluminación no está controlada. Dada la variabilidad de la iluminación a lo largo del día, la tasa de reconocimiento baja desde el 96 % de éxito, en el caso de buena iluminación, al 80 % en casos de una iluminación peor (al atardecer, por ejemplo). Evidentemente hay casos extremos (oscuridad o reflejo del sol) en los que es imposible reconocer la matrícula. El tiempo medio de reconocimiento de la matrícula en una imagen es de 1.5 seg.

La red de comunicaciones con el protocolo desarrollado permite la coordinación de los robots y la asignación de tareas. También permite la comunicación de los robots con la consola de mando y el seguimiento de la misión. En ella se visualiza



Figura 18: Imagen del robot navegando en el aparcamiento.

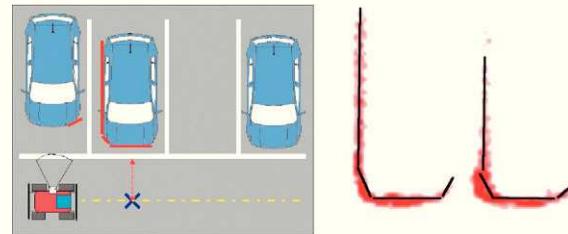


Figura 19: Segmentación de la información del sensor de rango láser para calcular el posicionamiento correcto del robot.



Figura 20: Imágenes tomadas con distinta iluminación una vez posicionado el robot mediante la navegación autónoma.

permanentemente el estado de cada uno de los robots, el avance de la misión, la información percibida por los sensores, y el envío de órdenes, todo ello en tiempo real y cumpliendo requisitos temporales estrictos. Una imagen de la consola desarrollada para la aplicación presentada puede verse en la figura 21.

Con respecto a la implementación del protocolo RT-WMP utilizada, ésta se basa en las capas IP y UDP de 802.11. Sin embargo las cabeceras IP y UDP añaden hasta 28 bytes de sobrecarga que son innecesarios para el funcionamiento del protocolo RT-WMP. Por esta razón, en el futuro, las capas IP y UDP serán eliminadas para mejorar el rendimiento. La tabla 1 muestra las prestaciones de la implementación propuesta en los diversos experimentos llevados a cabo para poner a punto la aplicación.

## 8. Conclusión

El trabajo desarrollado en este proyecto ha integrado diferentes tecnologías orientadas a la aplicación de reconocimiento



Figura 21: Consola de la aplicación en funcionamiento.

y localización de vehículos en aparcamientos, mediante la utilización de equipos de robots. Se han realizado contribuciones en cada una de las tecnologías utilizadas, así como en su integración y adaptación.

Se han realizado experimentos en simulación y en entornos reales, así como un experimento completo de la aplicación en un aparcamiento. En los diferentes experimentos se han evaluado las diferentes técnicas integradas y se han puesto de manifiesto algunas limitaciones existentes en este tipo de aplicación, debidas principalmente a las condiciones de iluminación. Se dispone en este momento de software y hardware para la transferencia tecnológica. Hasta donde conocemos, no existe en el mercado un producto que cubra las posibilidades que ofrecen las tecnologías desarrolladas y presentadas en este trabajo, ya que habitualmente se trabaja con sensores fijos en la infraestructura y en espacios controlados y de extensión limitada.

## English Summary

### Multi-robot system for vehicle localization and identification.

## Abstract

In this work we present a multi-robot system for the localization and identification of vehicles parked indoors or outdoors. In the initial stage, a priori planning is achieved and tasks are assigned to optimize the time of the mission. The robot team has different onboard sensors to achieve the localization and identification of a vehicle as well as its plate. A motion controller uses the laser sensor in each robot for an initial and precise localization in front of the car plate and the vision sensors are used to find the car and recognize the plate. The group of robots is endowed with communications to be used with each other and with a central station. Commands, alarms and data can be shared in real time maintaining the quality of service. The sensors used are commercially available and the algorithms have been designed and implemented by the Robotics, Perception and Real Time group of the University of Zaragoza with financial support from the Spanish Ministry of Innovation and Research. We integrate different technologies related to planning, navigation, perception and communications in the project to carry out the aimed application.

## Keywords:

Multi-robot systems, Mobile robots, Navigation, Task planning and allocation, Communication protocols, Robot vision, Real time.

## Agradecimientos

Los autores agradecen la colaboración a los compañeros del grupo que han colaborado en los experimentos, en especial a Luis Riazuelo.

Este trabajo ha sido financiado por Ministerio de Ciencia e Innovación a través de los proyectos DPI2003-07986, DPI2006-07928 y DPI2009-08126.

## Referencias

- Canny, J., 1986. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (6), 679–698.
- Douglas, D. H., Peucker, T. K., 1973. Algorithms for the reduction of the number of points required to represent a line or its caricature. *The Canadian Cartographer* 10 (2), 112–122.
- Felzenszwalb, P. F., Huttenlocher, D. P., Sep. 2004. Efficient graph-based image segmentation. *International Journal of Computer Vision* 59 (2).
- Hartley, R. I., Zisserman, A., 2004. *Multiple View Geometry in Computer Vision*. Cambridge University Press.
- Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., 1983. Optimization by simulated annealing. *Science* 252 (5020), 671–680.
- Land, E., 1986. Recent advances in retinex theory. *Vision Research* 26 (1), 7–21.
- Lopez-Nicolas, G., Sagües, C., Guerrero, J., Krasic, D., Jensfelt, P., 2008. Switching visual control based on epipoles for mobile robots. *Robotics and Autonomous Systems* 52 (7), 592–603.
- Lopez-Nicolas, G., Gans, N., Bhattacharya, S., Sagües, C., Guerrero, J. J., Hutchinson, S., 2010. Homography-based control scheme for mobile robots with nonholonomic and field-of-view constraints. *IEEE Trans. on Systems, Man and Cybernetics. Part B Cybernetics* 40 (4), 1115–1127.
- Montano, L., 2003. Técnicas de exploración automatizada en aplicaciones de rescate, EXPRES. In: DPI2003-07986.
- Montesano, L., Minguez, J., Montano, L., 2008. Modeling dynamic scenarios for local sensor-based motion planning. *Autonomous Robots* 25 (3), 231–251.
- Mosteo, A. R., Montano, L., 2006. Simulated annealing for multi-robot hierarchical task allocation with flexible constraints and objective functions. In: *Workshop on Network Robot Systems: Toward intelligent robotic systems integrated with environments. Int. Conf. on Intelligent Robots and Systems*.
- Mosteo, A. R., Montano, L., 2007a. Comparative experiments on optimization criteria and algorithms for auction based multi-robot task allocation. *IEEE Int. Conf. on Robotics and Automation (ICRA-07)*, 3345–3350.
- Mosteo, A. R., Montano, L., 2007b. SANCTA: An ada 2005 general-purpose architecture for mobile robotics research. *12th Int. Conf. on Reliable Software Technologies (Ada-Europe)*, Lecture Notes in Computer Science. Springer Verlag (220), 221–234.
- Mosteo, A. R., Montano, L., Lagoudakis, M. G., 2008. Multi-robot routing under the limited communication range. In: *IEEE Int. Conf. on Robotics and Automation*, pp. 1531–1536.
- MoviRobotics, 2010. Página web de empresa suministradora de servicios robóticos.  
URL: <http://www.movirobotics.com/ENvigilancia.php>
- OpenCV, 2010. Open computer vision library.  
Http://playerstage.sourceforge.net/.  
URL: <http://sourceforge.net/projects/opencvlibrary/>
- Rahman, Z., Woodell, G. A., Jobson, D. J., Hines, G. D., 2006. Advanced image processing of aerial imagery. In: *Visual Information Processing XV*, Proc. SPIE 6246.
- Roboserv, 2011. Página web de empresa suministradora.  
URL: <http://www.roboserv.net/robots-profesionales>
- RoPERT, 2009. Página web del proyecto.  
URL: [http://robots.unizar.es/html/ficha\\_resultado.php?id=73&idmember=](http://robots.unizar.es/html/ficha_resultado.php?id=73&idmember=)
- Sagües, C., 2006. Robots móviles en red para tareas de servicio y de intervención, NERO. In: DPI2006-07928, 2006–2009.
- Tardioli, D., Villaruel, J. L., 2007. Real-time communications over 802.11: RT-WMP. In: *The Fourth IEEE Int. Conference on Mobile Ad-hoc and Sensor Systems*, pp. 1–11.
- Tardioli, D., Mosteo, A., Riazuelo, L., Villaruel, J., Montano, L., 2010. Enforcing network connectivity in robot team missions. *The International Journal of Robotics Research* 29 (4), 460–480.