

Diseño de un Robot Móvil para Docencia en la Materia de Robótica

Matías García Rivera, Ángel Delgado Rodríguez, Joaquín López Fernández, Miguel Díaz-Cacho Medina
Departamento de Ingeniería de Sistemas y Automática, Universidad de Vigo, España
Email: {mgrivera, joaquin, mcacho}@uvigo.es, andelrod@yahoo.es

Resumen—En este artículo se presenta una plataforma que está siendo utilizada en la enseñanza de la robótica móvil. La plataforma consiste en la base móvil de un juguete, que ha sido modificada añadiendo sensores y actuadores fáciles de encontrar comercialmente, hasta crear un robot móvil totalmente funcional. De esta forma, los alumnos de la materia Robótica pueden construir este robot y proponer rediseños para modificar o añadir funcionalidades. Hasta la fecha se han utilizado robots sencillos para las prácticas de esta materia, con los que realizar ejercicios muy básicos de seguimiento de líneas o salir de un laberinto. Este nuevo robot es lo suficientemente complejo para ejecutar herramientas de alto nivel, con las cuales poder realizar mapas o planificar trayectorias.

I. INTRODUCCIÓN

La materia Robótica es común tanto en las hoy todavía existentes ingenierías informáticas, como en los ya presentes grados en informática. En esta materia integra diversos conocimientos adquiridos por los alumnos en asignaturas precedentes: física, matemáticas, electrónica e informática. El alumno trabajará con sensores y actuadores, controladores de motores, buses de comunicaciones, y programará diversos algoritmos en algún lenguaje de programación.

En las prácticas de Robótica suelen usarse robots como los MARK III [1] y PicBot [2], los dos robots de la parte superior de la figura 1. Estos robots se caracterizan por su sencillez, tienen un conjunto reducido de sensores (infrarrojos, de luz visible, de sonido, bumpers), los actuadores se reducen a los motores y en su microcontrolador se pueden ejecutar sencillos programas en C o ensamblador. Con estos robots se suelen hacer prácticas de rastreo de un camino trazado en el suelo, salir de un laberinto siguiendo sus paredes, o persecución de una fuente de luz o sonido.

Los Robot MARK III y PicBot son productos comerciales, que bien ya han sido diseñados por otros o bien pueden comprarse montados o no. Existen también la posibilidad de construcción de un sencillo robot uno mismo [3] [4].

Otros robots más completos como el PIONEER [5] o el RATO [6], los dos robots de la parte inferior de la figura 1, suelen dedicarse únicamente a tareas de investigación. Estos robots portan un PC o un portátil que permite ejecutar programas más complejos, y disponen de sofisticados sensores que aportan mucha más información sobre el entorno (un conjunto de sonars o un LASER). Este tipo de robot permite realizar mapas de recintos, planificar trayectorias y evitación de obstáculos.

Hasta ahora en las prácticas de la materia Robótica de la titulación de Ingeniero Informático en la Escuela Superior de Ingeniería Informática de Ourense, se han utilizado los sencillos robots MARK III. Pero nuestra idea siempre fue trabajar con un robot más complejo, pero robots como el PIONEER son comprados ya acabados y tienen un coste muy alto. Para ello, se ha diseñado un robot a partir de una plataforma móvil, añadiendo sensores y etapas de potencia comerciales para relativamente económicos. Esto permitirá a los alumnos tanto participar en la construcción del robot, como posteriormente ejecutar programas de alto nivel sobre él.

Los objetivos que se persiguen son tanto a nivel hardware como software:

- Empleo de componentes electrónicos tales como: relés, LEDs, transistores, resistencias, diodos, optoacopladores, sensores, actuadores, microcontroladores, etc.
- Ampliar el conocimiento del alumnado en la aplicación de equipos informáticos en el control de sistemas electrónicos.
- Posibilidad de reutilizar componentes de equipos informáticos, pequeños electrodomésticos y juguetes, viejos o en desuso.
- Aplicación de distintos lenguajes de programación, de bajo nivel como el ensamblador y de alto nivel como C o Java, para resolver problemas de robótica.

En resumen, crear un espacio donde los alumnos diseñen, construyan y programen robots en un entorno basado en la creatividad y la experimentación.



Figura 1. Robots MARK III, PicBot, PIONEER y RATO.

II. DESCRIPCIÓN DEL ROBOT

Los criterios de selección de los componentes tanto hardware como software del robot han sido su bajo coste, facilidad para encontrarse en el mercado (incluso componentes similares de otros fabricantes) y acceso mediante buses de amplio uso.

II-A. Componentes

Los componentes de este robot incluyen la base del robot, el sistema motriz y el sistema sensorial básico. Estos son:

- Plataforma, ruedas, reductoras, motores y batería del Dareway.
- 2 encoders procedentes de los motores EMG30.
- Placa de control de motores MD22.
- Placa de control de encoders MD23.
- 2 circuitos interfaz USB-I2C.
- Láser Hokuyo URG-04LX.
- 8 sensores s3nar SRF08.
- 6 bumpers.
- Chip GPIO14 para control de bumpers.
- Batería de 12 voltios, convertidor de tensi3n a 5 voltios y filtro de corriente continua.
- Ordenador port3til.
- C3mara web.

El Dareway es un veh3culo con bater3a de 12 voltios, que permite a un ni3o montado en 3l avanzar y girar [7]. Perteneciente a la empresa Famosa, fue dise3ado para su uso como juguete probablemente tomando como referencia el Segway [8], pero para evitar la complejidad de mantener el equilibrio con dos ruedas que el Segway soluciona con el uso de un giroscopio, el Dareway incorpora una tercera rueda de apoyo. El veh3culo tiene unas dimensiones de 620 mm de ancho, por 600 mm de largo y una altura de 870 mm.

El Dareway servir3 de base para el robot, ya que de 3l utilizaremos la plataforma, las ruedas, las reductoras, los motores y la bater3a. Eliminaremos la estructura para la sujeci3n del pasajero y ser3 sustituida por una plataforma horizontal donde posteriormente se colocarn3 los sensores sonar y laser, adem3s del un PC portatil. Los motores se pueden accionar individualmente teniendo una configuraci3n diferencial. No obstante no disponen de encoder para realizar la odometr3a o medici3n del desplazamiento, con lo cual deber3 ser a3adidos posteriormente. La figura 2 muestra el Dareway tal como puede ser adquirido, y las modificaciones que se han realizado sobre 3l.

El resto de los componentes que acompa3an al Dareway y sus interconexiones pueden verse en la figura 3. Los encoders del motor EMG30, las placas de control de motores MD22 y MD23, los circuitos interfaz USB-I2C, los sensores s3nar SRF08 y chip GPIO14 para el control bumpers, fueron escogidos por buena disponibilidad y su bajo coste [9].

El EMG30 es un motor de corriente continua de 12 voltios que incluye un encoder o codificador de cuadrante, el cual manda un tren de impulsos cuando gira el eje del motor permitiendo que un circuito externo pueda calcular la velocidad real a la que est3 girando el eje y cuantas vueltas da. El



Figura 2. Dareway: la base del robot

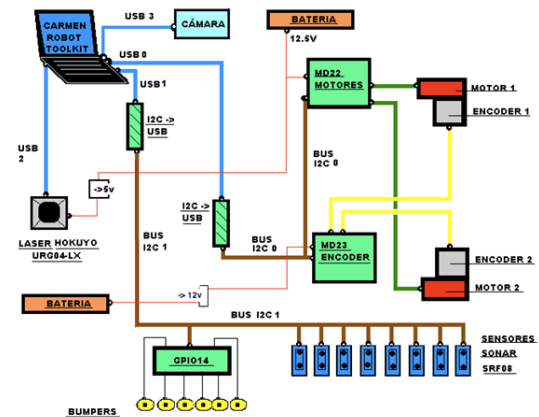


Figura 3. Arquitectura del robot

encoder est3 formado por dos sensores de efecto hall, que proporcionan un total de 360 pulsos por cada vuelta del motor. El motor tiene una potencia nominal de 4,22 vatios y ofrece una fuerza de 1,5 kg/cm, insuficiente para desplazar todo el peso del Dareway m3s un ordenador port3til incorporado. Por lo tanto, los motores usados son los propios del Dareway, pero modificados para incorporarles los encoders del EMG30. Para ello es necesario desmontar los encoder del EMG30 y montarlos ahora en los motores del Dareway tal como muestra la figura 4.a.

El MD22 de la figura 4.b es un controlador para 2 motores de corriente continua de mediana potencia. El controlador usa dos alimentaciones, una a 5 voltios para la l3gica, y otra alimentaci3n para el motor que est3 comprendida entre los 5 y los 24 V. Para cada uno de los motores el controlador es capaz de entregar hasta 5A de intensidad de corriente. Este controlador alimentar3 los motores de la base del Dareway y se controla mediante un bus I2C.

El MD23 en la figura 4.c es capaz de controlar dos motores de corriente continua de hasta 3 A y 12 V. Este circuito cuenta con entradas para leer la se3al de los encoders de los motores. Se accede externamente mediante un bus I2C, y cuenta con numerosos registros que controlan la aceleraci3n, la velocidad, la corriente de cada motor y los contadores de los encoders de cada motor. El circuito act3a a modo de regulador, aplicando

mayor o menor corriente a los motores para conseguir que alcance la velocidad real deseada. De esta forma se consigue que la velocidad sea siempre la misma independientemente de la carga o la resistencia que encuentre el motor. Los impulsos también pueden leerse externamente para determinar la distancia real recorrida o los giros realizados. Esta placa se usará sólo para la lectura de los encoders, ya que los 3 A de corriente que ofrece para cada motor son insuficientes para mover los motores del Dareway.

El GPIO14 de la figura 4.d es un chip para entradas y salidas digitales de propósito general accedido por bus I2C. También cuenta con la posibilidad de emplear 5 líneas como puertos de entrada analógicos con conversión de 10 bits y una salida PWM controlada por un convertidor digital a analógico de 8 bits. Este chip da acceso a la lectura de los sensores de contacto o bumpers de los cuales dispone el robot para detectar choques con obstáculos. Cuando todos los demás sensores han fallado y se produce un impacto, el robot debe detenerse y modificar su trayectoria. El funcionamiento de estos sensores es muy simple, se trata de interruptores que se conectan a un circuito eléctrico y que por lo tanto lo abren o lo cierran.

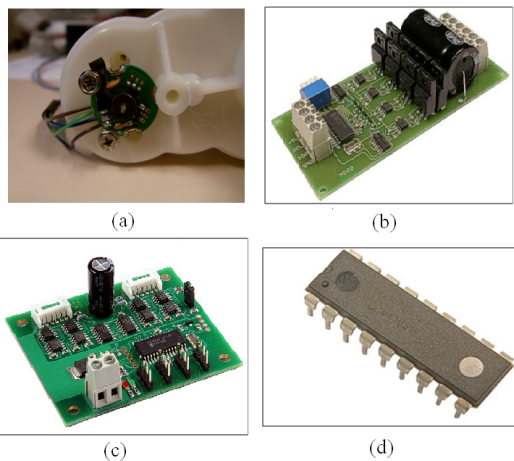


Figura 4. Varios componentes del robot:

El SRF08 de la figura 5.a es un medidor ultrasónico de distancias para robots que representa la última generación en sistemas de medidas de distancias por s3nar, consiguiendo niveles de precisi3n y alcance 3nicos e impensables hasta ahora con esta tecnolog3a. El sensor es capaz de detectar objetos a una distancia de 6 metros con facilidad adem3s de conectarse al micro-controlador mediante un bus I2C, por lo que se pueden conectar cuantos sensores sean necesarios en el mismo bus. Con una alimentaci3n 3nica de 5 voltios, s3lo requiere 15 mA para funcionar y 3mA mientras est3 en reposo, lo que representa una gran ventaja para robots alimentados por pilas. El sensor SRF08 incluye adem3s un sensor de luz que permite conocer el nivel de luminosidad usando igualmente el bus I2C y sin necesidad de recursos adicionales.

EL Hokuyo URG-04LX de la figura 5.b es un l3aser destinado a su uso en rob3tica, con una relaci3n aceptable calidad-precio, que lo hace asequible para desarrollos no industriales.

Ofrece tanto interfaz serie (RS-232) como USB, para ofrecer esc3neres l3aser extremadamente aproximados. El campo de visi3n es de 240 grados, y la resoluci3n angular es de 0.36 grados, con una tasa de actualizaci3n de hasta 10 MHz. Es capaz de detectar objetos situados en un plano horizontal desde 20 mm hasta 4 m. Se alimenta con una tensi3n de 5 voltios y 500 mS.

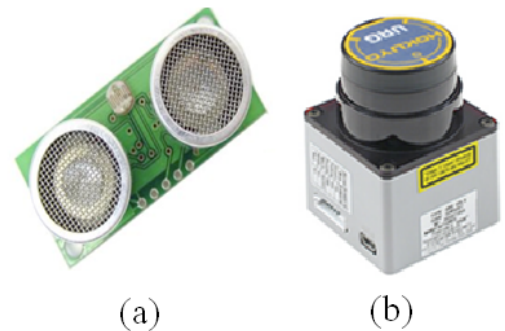


Figura 5. Sonar y laser empleados.

II-B. Interconexiones

Los componentes anteriormente descritos se conectan entre s3 y al ordenador mediante buses I2C y USB, adem3s de alimentarse con tensi3n aquellos que lo necesitan. La estructuraci3n del hardware se observa en la siguiente figura. Las conexiones se realizan de la siguiente forma:

II-B1. Bus USB 0 (puerto serie: /dev/ttyUSB0): Se corresponde con el sistema motriz y sensorial interno. Este primer bus USB se conecta a un interfaz USBI2C, mediante el cual se accede al primer bus I2C. A este primer bus se conectan en serie las placas MD22 y MD23. La placa MD22, de control de motores usa dos alimentaciones, una para la l3gica, a 5 voltios, que obtiene del propio bus I2C, y otra a 12.5 voltios para los motores, que obtiene de la bater3a. La placa tiene dos entradas que la conectan directamente con cada uno de los motores para enviar las ordenes de movimiento. La placa MD23, de control de encoders se alimenta de la misma manera que la placa MD22, una alimentaci3n a 5 voltios generada en la propia placa para la l3gica, y otra a 12.5 voltios para los encoders, que se toma de la bater3a adicional. Antes de la conexi3n con la placa, se conecta el filtro de corriente continua, para mantener una corriente continua constante en la placa, evitando intrusiones de se3ales variables que se pueden producir por ruido en los motores, y que pueden variar la tensi3n de entrada y bloquear la placa. La placa se conecta con cada uno de los encoders para recibir los pulsos enviados y registrarlos. Los motores y encoders usan la tensi3n de 12.5 voltios, que obtienen a partir de la bater3a. Los motores se conectan directamente a ella para alimentarse.

II-B2. Bus USB 1 (puerto serie: /dev/ttyUSB1): Se corresponde con parte del sistema sensorial externo, la parte de los sensores s3nar y los sensores de contacto. El bus USB se conecta a otro interfaz USBI2C, a trav3s del cual accede al

segundo bus I2C. A este bus se conectan en serie cada uno de los sensores s3nar y el chip GPIO14 de control de los bumpers. A parte de la conexi3n al bus I2C, el chip GPIO14 dispone de otros pines de entradas digitales. Se usar3n seis de ellas para conectar a cada uno de los seis bumpers usados. Los bumpers funcionan como interruptores conectados en paralelo en un circuito el3ctrico, por lo tanto al ser activados cerrar3n el circuito mandando una tensi3n a la entrada correspondiente del chip GPIO14.

II-B3. Bus USB 2 (puerto: /dev/ttyACM0): Se corresponde al l3ser, parte del sistema sensorial externo. La conexi3n con el sensor l3ser es directa por bus USB, y este necesita alimentarse externamente con 5 voltios de manera uniforme, por lo que no es v3lida la alimentaci3n del bus. Se usa el convertidor de tensi3n a 5 voltios, conectado a la bater3a para alimentar el sensor.

II-B4. Bus USB 3 (puerto serie: /dev/video0): Es el puerto correspondiente a la conexi3n usada por la c3mara web, utilizada como parte del sistema sensorial externo. La c3mara se conecta 3nicamente con el bus USB, ya que no utiliza una alimentaci3n extra.

II-C. Software

Carmen (Carnegie Mellon Robot Navigation Toolkit) [10], es un conjunto de m3dulos software de c3digo abierto para control de robots m3viles. Es un software modular creado para ofrecer soporte para primitivas como: control de la base y sensorial, registro de datos, evitaci3n de obst3culos, localizaci3n, planificaci3n de rutas y creaci3n de mapas. Las comunicaciones entre los distintos m3dulos de Carmen son manejadas usando un paquete adicional llamado IPC, creado en 1991 por Christopher Fedor y Reid Simmons. IPC se distribuye junto con Carmen, pero es un desarrollo independiente. Este paquete se basa en el paso de mensajes, que constituyen la informaci3n intercambiada. Los m3dulos publican mensajes con informaci3n, y otros m3dulos que necesiten esa informaci3n se suscriben a esos mensajes. Carmen tiene licencia GNU Public License (GPL), y ha sido desarrollada en la universidad de la cual hereda el nombre, Carnegie Mellon University. A continuaci3n se resumen sus funcionalidades principales:

- Es un software modular para control de robots m3viles.
- Usa la plataforma IPC para la comunicaci3n entre procesos.
- Monitorizaci3n de procesos.
- Soporte para el hardware de distintas plataformas rob3ticas.
- Soporte hardware para l3ser SICK o Hokuyo, y recibidores de GPS.
- Simulador en 2 dimensiones del robot y sensores.
- M3dulo de planificaci3n de caminos.
- M3dulo de localizaci3n.
- M3dulo de an3lisis sensorial y creaci3n de mapas a partir de la informaci3n de los sensores.
- Registro de mensajes, y funcionalidad de repetic3n de acciones.
- Servidor de par3metros centralizado.

- Distintas funcionalidades muy 3tiles para trabajo y programaci3n de robots.
- M3dulos desarrollados en lenguaje C, pero con soporte para lenguaje JAVA.
- Ejecuci3n bajo sistema operativo Linux, y disponible bajo lice

El nivel alto del sistema se ocupa de la recopilaci3n de informaci3n recibida y su procesamiento, de la fusi3n sensorial, de la localizaci3n, la planificaci3n de trayectorias, y la supervisi3n, detecci3n y correcci3n de errores. Se corresponde con la herramienta Carmen, de control de robots m3viles aut3nomos, que se ejecuta bajo sistema operativo Linux Ubuntu. La herramienta se compone de diversos m3dulos, cada uno con su funci3n que se comunican mediante el paso de mensajes con la tecnolog3a IPC. Existe un m3dulo central que controla el intercambio de mensajes entre los dem3s. El resto de m3dulos tienen fines distintos:

- Servir par3metros generales del robot.
- Modificaci3n de par3metros generales.
- Obtener informaci3n de la base.
- Obtener informaci3n de los sensores.
- Control del movimiento del robot.
- Localizaci3n.
- Navegaci3n.
- Creaci3n de mapas a partir de archivos de log.
- Interfaz gr3fico usando informaci3n sensorial, odometr3a y mapas.
- Registrar informaci3n en un archivo de log.

Dentro de la herramienta Carmen, se adapta el m3dulo robot, que interact3a directamente con la base del robot, para el funcionamiento deseado en este robot.

III. APLICACIONES

Con el desarrollo de Internet las aplicaciones de teleoperaci3n en los robots m3viles est3n m3s extendidas [11]. La transmisi3n de datos de un punto a otro del planeta en tiempos de milisegundos, hace posible que un usuario pueda dar ordenes de movimiento al robot, mediante el teclado u otro dispositivo, y ver por donde se est3 moviendo este robot. En nuestro caso utilizando la red inal3mbrica de la Escuela de inform3tica hemos podido hacer la teleoperaci3n del robot m3vil tal como se puede apreciar en la figura 6.

Uno de los mayores problemas en la navegaci3n de robots m3viles es su localizaci3n con un alto grado de precisi3n dentro de su entorno [12]. Los problemas en la localizaci3n y construcci3n de mapas son debidos a la existencia de ruido en las medidas de los sensores y su limitaciones de alcance. Carmen puede crear un mapa a partir de las medidas obtenidas teleoperando al robot. Dispone adem3s de herramientas para corregir los errores en las medidas. La figura 7 muestra el mapa obtenido de la tercera planta del edificio de la Escuela de Inform3tica.

Finalmente el robot puede ser utilizado por los alumnos para desplazarse por la Escuela. Se puede indicar en el mapa de la planta, las posiciones inicial y final, y empezar la

navegación autónoma del robot para que, a través de cálculo de trayectorias, evitación de obstáculos y localización, llegue a su destino. La figura 8 muestra la navegación por una zona de la Escuela.

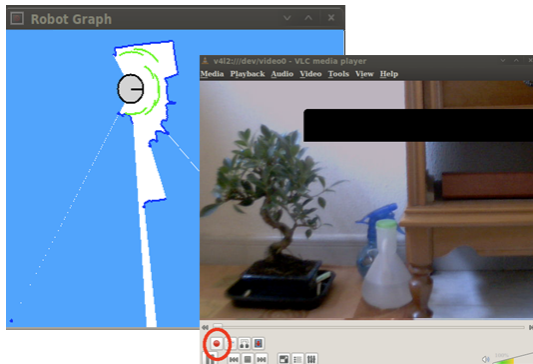


Figura 6. Teleoperación: imágenes de la cámara y de los sensores.

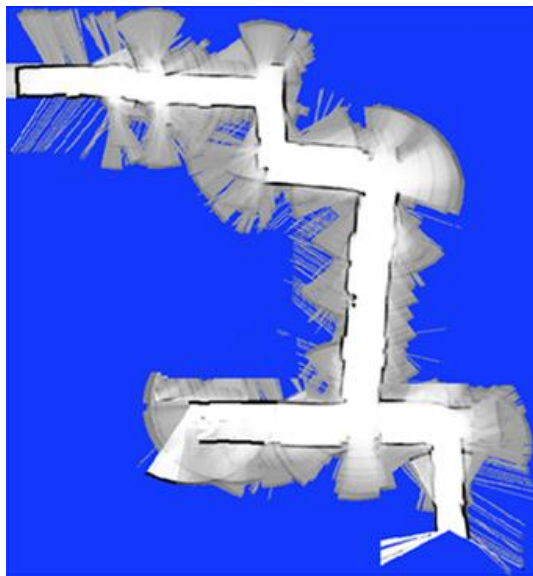


Figura 7. Creación de mapas.

IV. CONCLUSIONES

El robot diseñado puede ser construido con componentes económicos y fáciles de encontrar en el mercado. Deja la abierta la posibilidad de añadir nuevos componentes o ser sustituidos por otros. El alumno tendrá la posibilidad de modificar el diseño del robot y añadir nuevos componentes software a la herramienta Carmen para añadir nuevas funcionalidades.

REFERENCIAS

- [1] Sitio web del robot MARKIII. [Online]. Available: <http://www.junun.org/MarkIII/>
- [2] Sitio web del robot PICBOT. [Online]. Available: <http://www.msebilbao.com/tienda/index.php?cPath=53>
- [3] J. M. Angulo, *Introducción A La Robótica*. Paraninfo, 2005.
- [4] J. W. M. James M. Conrad, *Stiqito: Advanced experiments with a simple and inexpensive robot*. IEEE Computer Society, cop., 1998.

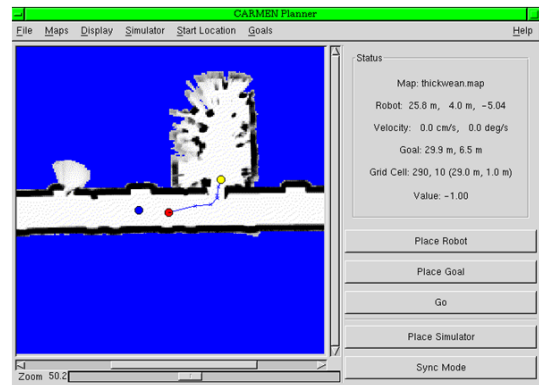


Figura 8. Navegación.

- [5] Sitio web del robot PIONNER. [Online]. Available: <http://www.mobilerobots.com/researchrobots/PioneerP3DX.aspx>
- [6] Sitio web del robot RATO. [Online]. Available: <http://webs.uvigo.es/vigobot/ride.php?content=robocan>
- [7] Sitio web del fabricante del juguete Dareway. [Online]. Available: <http://www.famosa.es>
- [8] Sitio web del distribuidor en España del Segway. [Online]. Available: <http://www.segway.es>
- [9] Distribuidor de componentes electrónicos. [Online]. Available: <http://www.superrobotica.com>
- [10] Carmen: Robot navigation tool.
- [11] V. F. J. Benali A., Wasiak, "Remote robot teleoperation via internet. a first approach," *Proceedings of the 10th IEEE International Workshop on Robot and Human Interactive Communication*, 2001.
- [12] A. J. R. G. Diego Rodríguez-Losada, Fernando Matía and G. Lacey, "Implementing map based navigation in guido, the robotic smartwalker," *In IEEE International Conference on Robotics and Automation*, 2005.