



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

**RECONOCIMIENTO Y SEGUIMIENTO DE PERSONAS CON
ROBOTS MÓVILES EN AMBIENTES DINÁMICOS EMPLEANDO
TÉCNICAS DE VISIÓN COMPUTACIONAL**

TESIS

QUE PARA OPTAR POR EL GRADO DE:

MAESTRO EN INGENIERÍA (COMPUTACIÓN)

PRESENTA:

JUAN CARLOS ÁLVAREZ REBOLLAR

TUTOR:

**DR. JESÚS SAVAGE CARMONA
FACULTAD DE INGENIERÍA**

MÉXICO, D.F. FEBRERO 2015



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

Quiero agradecer en primer lugar a mis sinodales los Doctores, Nidiyare Hevia, Boris Escalante, Edgar Garduño y al Maestro Abel Pacheco por sus tiempos y aportaciones para la mejora de este trabajo.

Al doctor Jesús Savage Carmona que me acepto como tesista y como parte de su equipo de trabajo, lo que me ayudó en gran medida en mi desarrollo profesional y académico, también quiero agradecer a mis compañeros del Laboratorio de Biorobótica, principalmente a Jesús, Nonato, Marco y Cecilia por su apoyo.

Así mismo, al Posgrado en Ciencias e Ingeniería de la Computación por brindarme la oportunidad de estudiar en su programa, a todos los académicos que durante estos 2 años estuvieron brindándome su conocimiento, y también al CONACYT por cumplir adecuadamente con lo convenido durante todo el periodo.

Agradezco a la DGAPA-UNAM por el apoyo proporcionado para la realización de esta tesis a través del proyecto PAPIIT IN117612, “Robot de Servicio para Asistencia a Adultos Mayores y en Sistemas Hospitalarios”.

Obviamente también a Justina y Judy por su increíble disponibilidad de tiempo en la realización de las pruebas.

Y por último quiero agradecer a mi familia (a mis padres, a mi hermano y a mi amada esposa Diana) por apoyarme en este proyecto, también a mis amigos de la maestría Zamira, Ivan, Israel y Oscar por su apoyo de los diferentes proyectos en las materias cursadas.

Resumen

En el presente trabajo se presentan los métodos que se desarrollaron para conseguir que una robot móvil pueda seguir a un humano en ambientes dinámicos. La acción de seguir conlleva al movimiento en un espacio limitado, complejo y desconocido para la robot. Gracias a las cámaras RGB-D (una de éstas el *KinectTM*) y a sensores láser es posible obtener información espacial de la escena, tales como posiciones y distancias que a su vez se transforman en espacio navegable, obstáculos y localización del objetivo; con la ayuda de las máquinas de estados se puede iniciar una serie de movimientos mecatrónicos, procesos de visión y de desplazamiento de la robot por la ruta correcta de acuerdo al entorno.

La robot deberá visualizar el o los colores característicos de la prenda que le cubra el torso del humano a seguir (comunmente es la camisa) para realizar el reconocimiento de éste, ya que a partir de éste se determinará su posición y se determinará la dirección y velocidad de la robot.

Finalmente los algoritmos desarrollados en esta tesis para seguir a humanos son utilizados por las robots Justina y Judy, en las tareas que ellas ejecutan para ayudar a adultos mayores, la atención hospitalaría y para la humanidad en general.

Índice general

Agradecimientos	I
Resumen	III
Índice de figuras	VII
Índice de tablas	XIII
1. Introducción	1
1.1. Presentación	1
1.2. Objetivos	3
1.3. Motivación del trabajo de tesis	3
1.4. Estructura del documento	5
2. Fundamentos y antecedentes	7
2.1. Antecedentes	7
2.2. Herramientas de <i>software</i>	14
2.2.1. Sistemas operativos	14
2.2.2. Bibliotecas	15
2.3. Marco Teórico	16
2.3.1. Espacios de Color RGB y HSV	16
2.3.2. Cuantización vectorial	18
2.3.3. Láser	19
2.3.4. Cámaras RGB-D	20
2.3.5. Campos potenciales	21
2.3.6. Filtro de Kalman	24
3. Desarrollo e implementación	25
3.1. Sensado de personas con láser	25
3.2. Reconocedor de piernas basado en pendientes	26
3.3. Detección de humanos	29

3.4.	Entrenamiento y reconocimiento de camisa	35
3.4.1.	Almacenamiento del histograma	39
3.4.2.	Centroides de colores	39
3.4.3.	Reconocimiento de la camisa	40
3.5.	Rastreo con filtro de Kalman	42
4.	Aplicación en robots móviles	43
4.1.	Robots Justina y Judy	43
4.2.	Control de la Robot	48
4.3.	Sensado con Láser	48
4.4.	Sensado con Visión	51
4.5.	Rastreando al operador	54
5.	Pruebas y análisis de resultados	59
5.1.	Procedimiento	59
5.2.	Resultados	62
5.2.1.	Experimento 1 - Cuarto vacío	62
5.2.2.	Experimento 2 - Pasillo largo con humanos	67
5.2.3.	Experimento 3 - Habitación con muebles y humanos	72
5.2.4.	Competencias	76
5.3.	Análisis	78
6.	Conclusiones y trabajo a futuro	79
6.1.	Conclusiones	79
6.2.	Trabajo a futuro	80
6.2.1.	Detección de piernas y personas	80
6.2.2.	Reconocimiento de operador	80
6.2.3.	Planeación de movimientos	81
A.	Prueba <i>Follow Me</i>	83
A.1.	Prueba <i>Follow Me</i>	83
A.2.	Objetivo	83
A.3.	Configuración	83
A.4.	Tarea	84
A.5.	Reglas y observaciones adicionales	85
A.6.	Instrucciones del Comité de Organización y Árbitros	86
A.7.	Hoja de Puntuación	87
	Bibliografía	87

Índice de figuras

2.1. <i>Electric dog</i> fue el primer robot móvil creado con la capacidad de seguimiento [CT].	8
2.2. La tortuga de Walter tenía la capacidad de interacción social, poseían sensores de luz, proximidad y comportamientos reactivos simples de forma analógica [FND03].	8
2.3. Robot <i>Beast</i> tenía la capacidad de navegar por pasillos, corredores, buscar contactos de electricidad y auto conectarse [WS05].	9
2.4. Robot <i>Shakey</i> fue el primer robot móvil con comportamiento no reactivo, tenía un sistema de planeación de acciones y un sistema de visión artificial [Cen84].	9
2.5. Robot <i>Hilare</i> tenía una cámara monocromática y un escáner láser, de tal manera que en conjunto generaban una cámara con noción de profundidad [GCV90].	10
2.6. Robots <i>Rhino</i> y <i>Minerva</i> fueron construidos para la asistencia e interacción con las personas en lugares públicos, tenían cámara estéreo, sonares, táctiles e infrarrojos [BCF+99,BCF+98].	10
2.7. Robot <i>Roomba</i> [®] puede aspirar la casa, pero lo hace sin supervisión humana dado que es autónomo, pequeño y de forma circular [FD06].	11
2.8. Robot <i>iBotGuard</i> constituido por dos partes, un robot móvil y un servidor, el sistema en el servidor podía reconocer personas y seguirlas mientras estuvieran frente a él [LWF05].	11
2.9. Robot <i>Nao</i> mide solo 58 cm de altura y tiene forma humanoide, posee habilidades de reconocimiento, habla y escucha [NAO06].	12
2.10. Robot <i>Security Warrior</i> contaba con dispositivos de sensores incluida una cámara de colores, láser, ultrasónicos y <i>wireless</i> [LCLT07].	13
2.11. Evolución del robot <i>Asimo</i> que posee un sistema capaz de manipular su mapa de navegación, manipulación de objetos, creación de mejores rutas, tareas de habla, de acción, de procesamiento de imágenes, uso de comandos, monitoreo de su estado de comportamiento, entre otros [Wor13,SWA+02].	13

2.12.	Diagrama del <i>BlackBoard</i> y su comunicación con los demás módulos. . .	15
2.13.	Calibración y seguimiento del esqueleto utilizando OpenNI TM , en (a) la posición en la que el humano debe de estar para iniciar el seguimiento, y en (b) una pantalla de OpenNI TM realizando una detección y seguimiento de un esqueleto.	16
2.14.	Representación gráfica de los modelos de espacio de color: (a) modelo RGB y (b) modelo HSV.	17
2.15.	Pantalla de la clasificación de 10 cúmulos de puntos en 8 partes utilizando dos algoritmos: (a) <i>K-means</i> y (b) cuantización vectorial.	19
2.16.	Graficación de una lectura del láser Hokuyo obtenida cuando la robot Justina está en un pequeño cuarto de 1.2×1.2 metros aproximadamente y enfrente se encuentra una persona.	20
2.17.	Pantallas de la graficación de la información obtenida por el <i>Kinect</i> TM , en (a) se muestra la información de profundidad respecto al <i>Kinect</i> TM , mientras más cercas más amarillos y mientras más lejos más negros son los píxeles, y en (b) la imagen RGB.	21
2.18.	Muestra los efectos de los campos potenciales de atracción y repulsión, identificados con el color verde y rojo respectivamente, de color azul el robot y con una línea azul la reacción probable del robot ante esos campos.	22
3.1.	Patrones de piernas PS (piernas separadas o aisladas), PD (piernas desfasadas o caminando) y PJ (pierna juntas) cuando el láser está en forma horizontal enfrente de las piernas.	26
3.2.	Geometria de los puntos principales para la identificación de piernas: los puntos p_{ini} y p_{fin} son los puntos de inicio y final del cúmulo de puntos a evaluar, respectivamente, p_m el punto medio del segmento $\overline{p_{ini}p_{fin}}$ y p_o el origen (posición del láser).	27
3.3.	Detección en patrones de piernas: (a) es una sola pierna, (b) son dos piernas juntas, (c) dos piernas, una adelante de la otra, y (d) un cúmulo que no es pierna.	28
3.4.	Muestra un punto $K_P[i][j]$ y su referencia con respecto a \mathbb{S}_R (sistema coordinado del robot) y \mathbb{S}_K (sistema coordinado del <i>Kinect</i> TM), también se muestran las distancias t_x, t_y y t_z , y los ángulos de giro α y ϕ para realizar la transformación de \mathbb{S}_K a \mathbb{S}_R	30
3.5.	Muestra la localización de los puntos vecinos $K_P[i-2][j-2]$, $K_P[i+2][j+2]$, $K_P[i-2][j+2]$, $K_P[i+2][j-2]$, y la $K_{\bar{n}}[i][j]$ con respecto a $K_P[i][j]$	31

3.6.	Graficación de la información de una escena obtenida con el <i>KinectTM</i> donde (a) es la imagen RGB, (b) son las normales $K_{\vec{n}}$, (c) es el componente x de las normales $K_{\vec{n}}$, (d) es el componente y de las normales $K_{\vec{n}}$, (e) es el componente z de las normales $K_{\vec{n}}$ y $K_{\vec{n}}$ es la matriz de normales de la escena.	32
3.7.	Muestra 4 pantallas de la salidas de la discriminación del no torso, los incisos (a) y (b) son dos imágenes en las cuales se muestran los píxeles que se encuentran a la altura de la camisa del humano, los incisos (c) y (d) son imágenes en las cuales unicamente se muestran los píxeles que pueden ser humanos.	34
3.8.	Patrones iniciales del entrenamiento, el inciso (a) es la parte de enfrente del operador (pecho y estómago) y el inciso (b) es la espalda.	35
3.9.	Distorsiones de rotación sobre un patrón original, en el inciso (a) la camisa sin el brazo izquierdo, en (b) la camisa sin ambos brazos y en (c) la camisa sin el brazo derecho.	36
3.10.	Distorsiones de escalamiento sobre un patrón de rotación, en el inciso (a) la camisa sin brazos con resolución <i>default</i> , en (b) la camisa sin brazos con $\frac{3}{4}$ de la resolución <i>default</i> y en (c) la camisa sin brazos con $\frac{1}{2}$ de la resolución <i>default</i>	37
3.11.	Distorsiones de iluminación sobre un patrón original, en el inciso (a) es el patrón original obscurecido, en (b) es el patrón original y en (c) es el patrón original aclarado.	38
3.12.	Ecuaciones del filtro de Kalman.	42
4.1.	Diagrama VirBot presenta la composición del sistema VirBot y las conexiones de sus subsistemas, este sistema está dividido en 4 areas: (i) entradas, (ii) planificación, (iii) gestión del conocimiento y (iv) ejecución, para más detalles se puede ver [SCH98].	44
4.2.	Imágenes de (a) Justina y (b) Judy.	47
4.3.	Diagrama de verificación de piernas.	49
4.4.	Muestra una serie de imágenes tomadas con la robot Justina, visualizando la información capturada por las lecturas del láser mientras una persona camina de derecha a izquierda y de regreso, existiendo algunos objetos a su alrededor pero sin generar oclusión. Sobre las imágenes están marcados con un rectángulo rojo los cúmulos de puntos de las piernas detectadas.	50
4.5.	Diagrama de entrenamiento del operador, describe paso a paso el método general utilizado para entrenar al operador.	51
4.6.	Diagrama de comparación de camisas, donde se describe paso a paso el método general utilizado.	52

4.7.	Muestra una serie de imágenes tomadas con la robot Justina, estas imágenes tienen marcados los humanos detectados con rectángulos, si el rectángulo es de color verde significa que el humano es reconocido como el operador, de lo contrario, el rectángulo estará de color rojo, las imágenes corresponden a una prueba de seguimiento a través de un pasillo con varias personas alrededor, y algunas haciendo oclusiones.	53
4.8.	Diagrama interno de la metodología propuesta para el seguimiento de personas.	54
4.9.	Muestra una serie de imágenes de rutas hechas por la robot Judy y el operador, en experimentos realizados y documentados en el Capítulo 5, en (a) es el experimento 1, en (b) es el experimento 3, y en (c) es el experimento 2. Los recorridos constan de 2 líneas: la línea verde es la ruta del operador y la línea azul es la ruta seguida por la robot	55
4.10.	Diagrama de la máquina de estados, muestra el diagrama de la máquina de estados que se utiliza para el seguimiento del operador, donde cada cuadro representa un estado que la robot debe de ejecutar.	56
5.1.	Imagen que muestra la distancia y ángulo instantaneo entre la robot y el operador.	62
5.2.	Esquema de la distribución territorial del experimento 1, las áreas de color rosa representan obstáculos u objetos del ambiente, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.	63
5.3.	Imágenes del lugar donde se probó el experimento 1 donde (a) es la vista desde el pasillo y (b) es la vista desde una esquina del cuarto.	63
5.4.	Gráfica de distancias obtenidas con láser en el experimento 1.	65
5.5.	Gráfica de distancias obtenidas con <i>KinectTM</i> en el experimento 1.	65
5.6.	Gráfica de distancia obtenidas con láser+ <i>KinectTM</i> en el experimento 1.	65
5.7.	Gráfica de orientaciones obtenidas con láser en el experimento 1.	66
5.8.	Gráfica de orientaciones obtenidas con <i>KinectTM</i> en el experimento 1.	66
5.9.	Gráfica de orientaciones obtenidas con láser+ <i>KinectTM</i> en el experimento 1.	66
5.10.	Esquema de la distribución territorial del experimento 2, las áreas de color rosa representan obstáculos u objetos del ambiente, los círculos amarillos representan personas, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.	67
5.11.	Imágenes del lugar donde se probó el experimento 2 donde (a) es la vista desde donde se inicia el recorrido y (b) es la vista desde donde se termina el recorrido.	68

5.12. Gráfica de distancias obtenidas con láser en el experimento 2.	69
5.13. Gráfica de distancias obtenida con <i>KinectTM</i> en el experimento 2. . . .	70
5.14. Gráfica de distancia obtenidas con láser+ <i>KinectTM</i> en el experimento 2.	70
5.15. Gráfica de orientaciones obtenidas con láser en el experimento 2. . . .	70
5.16. Gráfica de orientaciones obtenidas con <i>KinectTM</i> en el experimento 2. .	71
5.17. Gráfica de orientaciones obtenidas con láser+ <i>KinectTM</i> en el experi- mento 2.	71
5.18. Esquema de la distribución territorial del experimento 3, las áreas de color rosa representan obstáculos u objetos del ambiente, los círculos amarillos representan personas, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.	72
5.19. Imágenes del lugar donde se probó el experimento 3 donde (a) es la vista desde donde se inicia el experimento, (b) es la vista desde el pasillo al interior de la habitación y (c) es la vista del pasillo.	73
5.20. Gráfica de distancias obtenidas con láser en el experimento 3.	74
5.21. Gráfica de distancias obtenidas con láser+ <i>KinectTM</i> en el experimento 3.	75
5.22. Gráfica de orientaciones obtenidas con láser en el experimento 3. . . .	75
5.23. Gráfica de orientaciones obtenidas con láser+ <i>KinectTM</i> en el experi- mento 3.	75
5.24. Judy ejecutando <i>follow me</i> en <i>Robocup</i> 2014 donde (a) es Judy antes de la oclución, (b) es Judy en el elevador y (c) es Judy terminando la segunda fase.	77

Índice de tablas

5.1. Valores promedio para determinar el desempeño de la robot en los ex- perimentos.	61
5.2. Resultados obtenidos con láser en el experimento 1.	64
5.3. Resultados obtenidos con <i>KinectTM</i> en el experimento 1.	64
5.4. Resultados obtenidos con láser+ <i>KinectTM</i> en el experimento 1.	64
5.5. Resultados obtenidos con láser en el experimento 2.	68
5.6. Resultados obtenidos con <i>KinectTM</i> en el experimento 2.	69
5.7. Resultados obtenidos con láser+ <i>KinectTM</i> en el experimento 2.	69
5.8. Resultados obtenidos con láser en el experimento 3.	74
5.9. Resultados obtenidos con láser+ <i>KinectTM</i> en el experimento 3.	74
5.10. Puntuación en prueba <i>Follow me Robocup@home</i> 2014.	76
5.11. Promedios de los experimentos.	78

Capítulo 1

Introducción

1.1. Presentación

El ser humano es un ser único cuyo funcionamiento todavía no podemos igualar artificialmente e incluso, en algunos casos, ni siquiera entender. Sin embargo, se han estado haciendo grandes esfuerzos para que ésto cambie y se han conseguido muchos avances en varias disciplinas como en las matemáticas [Váz04], física y biología [EF89], que nos han ayudado para formar las bases de otras como en la computación y la robótica, es aquí donde el hombre se ha propuesto uno de sus mayores retos, crear una máquina a su imagen y semejanza.

Por ejemplo, se han logrado avances como en el juego del ajedrez en el cual una computadora ha logrado vencer al campeón mundial (a base de fuerza bruta al realizar muchos cálculos) [IBM13]; en el cómputo matemático al poder realizar muchas operaciones en poco tiempo [Coe04] y fuerza sobrehumana al cargar o mover miles de kilogramos [IH05], entre otros. Sin embargo, falta algo muy importante que para el ser humano aparentemente es muy sencillo: el sentido común. Una de las cosas que podemos hacer gracias a él, es el de seguir una persona, que es el tema principal de esta tesis.

La acción de seguir, ya sea a una persona o a un objeto, es muy común en nuestros días, ésto lo podemos ver desde nuestra infancia al seguir a nuestros padres o abuelos, cuando nos muestran algún lugar, para jugar, o incluso al cazar [AM01].

Todo lo anterior ha sido llevado a la robótica y más específico a los robots de servicio doméstico [Rob13]. La acción de seguir en estos robots es de gran utilidad para las labores que pudieran llegar a desempeñar, algunos ejemplos pueden ser el mostrarle la casa o para mostrarle algún lugar fuera como la tiendita de la esquina o simplemente un sitio de interés como un parque o la escuela de los niños. Además, puede consolidarse como un importante vínculo para la interacción humano-robot [MV12], al poder compartir de esta manera más espacios entre sí, sin mencionar que el hogar es uno de los lugares más íntimos que pueda tener la familia, por lo que no querríamos un robot de sirviente que rompa o tire nuestras cosas, simplemente porque no ve bien o su navegación es deficiente.

Dado que vivimos en un mundo físico donde nuestra principal herramienta es la vista [CO02], es importante trabajar arduamente en la visión computacional de los robots y tener importantes avances dada la importancia de los colores para los humanos y la relación humano-robot que se puede tener en el futuro. Cuando seguimos a una persona nos fijamos en pequeños detalles como es la vestimenta, el físico y hasta el modo de caminar; pero ésto no acaba ahí, también tenemos que calcular la distancia y dirección de la persona para generar una ruta válida y óptima; para llegar a él sin descartar la velocidad, ya que, si vamos lento lo perderemos y si vamos rápido podemos chocar, causando accidentes.

Así es que tenemos mucho trabajo que hacer, por un lado debemos elaborar algoritmos de visión computacional para ver a las personas a partir de sus características especiales, y por el otro generar un mecanismo capaz de generar toda la cadena de movimientos necesarios para conseguir el objetivo de la forma más natural posible sin importar que sea un espacio sin obstáculos, una casa-habitación u oficina.

1.2. Objetivos

Esta investigación se enfocó en el desarrollo de un sistema seguidor de personas, por lo cual el objetivo general es:

- Detectar, reconocer y seguir a una persona en movimiento con un robot móvil, a través de escenarios complejos y desconocidos para el robot usando una ruta no pre-definida.

Y a partir de éste podemos desglosarlo en los siguientes objetivos particulares:

- Detectar la presencia de seres humanos dentro de la escena.
- Reconocer a la persona a seguir con base en un conjunto de patrones característicos invariantes a cambios de perspectiva, iluminación, rotación y escala.
- Diseñar el planeador de acciones y movimientos necesario para seguir a dicha persona, robusto a oclusiones parciales o totales.

1.3. Motivación del trabajo de tesis

La robótica es una ciencia muy importante en la actualidad porque contribuye al desarrollo tecnológico y económico de los países [GTR07]. Se han creado una gran variedad de robots para cubrir las tareas más relevantes de nuestras vidas, para ello la *Robocup* ha creado diferentes áreas que sirven como motivación en el desarrollo de éstas, tales como áreas de rescate, deportivas (especialmente juego de fútbol), para prácticas de estudiantes, de servicio doméstico, entre otros [dav13].

Las guerras son hechos fuertemente criticados por la mayoría, sin embargo, no olvidemos que nos ha ayudado intensamente al desarrollo tecnológico, a tal grado que si no hubiese sido por ellas probablemente la robótica no tendría el alcance que tiene hoy, pero éste tiene un gran costo en sangre, sufrimiento y dolor al mundo en general, así es que la humanidad tuvo que buscar otras maneras más civilizadas, que por una parte, se desarrolle tecnología a una velocidad creciente, y al mismo tiempo, tener la dicha de poder derrotar al prójimo sin pagar la factura mencionada anteriormente [Zie].

Por ello se crearon muchas competencias, de las cuales aquí mencionaremos únicamente dos como ejemplo por tener una fuerte presencia en el país, la primera

es el Torneo Mexicano de Robótica que es un concurso a nivel nacional cuyo objetivo es “Impulsar el desarrollo de robots autónomos capaces de resolver problemas de interés para la sociedad”¹, y el segundo es la *Robocup* que es un evento muy importante a nivel internacional cuyo objetivo es “Conseguir para el año 2050 un equipo de robots capaz de vencer al ganador de la copa mundial de la FIFA (*Fédération Internationale de Football Association*) siguiendo sus propias reglas”².

En ambas competencias existen varias categorías, *Robocup@home* es una de ellas, la cual busca mejorar las habilidades de los robots de servicio [ifr13]. Al participar en *Robocup@home* los robots realizan una serie de prácticas que intentan semejar situaciones cotidianas de nuestras vidas, entre ellas está la prueba de *Follow me* la cual se centra en el seguimiento y reconocimiento de una persona desconocida, la interacción básica y capacidades de señalización, y la seguridad de la navegación en entornos desconocidos y estrechos con otras personas caminando o bloqueando el camino (en el Apéndice A se encuentra descrita de forma más detallada dicha prueba) [Rul14], lo que puede servir para asistencia a adultos mayores y en sistemas hospitalarios.

Para que los robots puedan ejecutar satisfactoriamente las tareas es necesario un conjunto de elementos entre *software* y *hardware* (entre ellos el sistema visión, planeación de movimientos, sistema de voz, motores, ruedas, pinzas, micrófonos, etc.) y obviamente un buen equipo de trabajo, todos comunicándose de manera eficaz y efectiva. Estas competencias abren muchas oportunidades para estudiantes de diferentes categorías, desde hace 17 años en el Laboratorio de Biorrobótica de la Facultad de Ingeniería de la UNAM alumnos desde licenciatura hasta de doctorado, de diferentes especialidades como son: electrónica, computación, mecánica, artes visuales, entre otras, han construido robots de servicios, mejorado continuamente en las diferentes disciplinas, entre ellas la visión computacional donde esta tesis está enfocada, y dirigida por del Dr. Jesús Savage Carmona fundador de este Laboratorio.

¹<http://ccc.inaoep.mx/tmr2013/>

²<http://www.robocup.org/about-robocup/objective/>

1.4. Estructura del documento

Este documento se encuentra dividido de la siguiente manera: en el Capítulo 2 describe los conceptos básicos a utilizar para los algoritmos que darán solución a la problemática, así como un poco de historia, en el Capítulo 3 se muestra como se llevó a cabo cada uno de los algoritmos desarrollados, este Capítulo se separó en tres partes: *(i)* detección y reconocimiento: que se encarga de encontrar al individuo, *(ii)* filtro de Kalman para determinar un camino para llegar a la persona, y *(iii)* planeador de movimientos que realizará el control de las acciones y movimientos de la robot a la hora de su ejecución, el Capítulo 4 presenta el procedimiento para el seguimiento de personas, el Capítulo 5 presenta los resultados obtenidos de los experimentos realizados siguiendo la metodología sugerida por esta tesis y por último, al final del documento se presentan las conclusiones a las que se llegaron al terminar el proyecto, así como el trabajo a futuro que restaría por hacer, la descripción de la prueba *Follow me* de *Robocup* y las referencias usadas.

Capítulo 2

Fundamentos y antecedentes

La robótica es una ciencia relativamente joven en comparación a otras como las matemáticas o la física, sin embargo, cada vez es más común oír, hablar y ver robots, están tomando una fuerte presencia en nuestras vidas, por ejemplo, participando en la medicina o fábricas.

En este Capítulo se verán 3 cosas referentes a este trabajo, primero un poco de historia sobre seguimiento en robots móviles y algunos robots transcendentales obtenidos en su mayoría de [Bec12], después se verán algunas herramientas utilizadas por los robots hoy en día, y al final se verán algunos conceptos teóricos que se utilizaron.

2.1. Antecedentes

Desde la creación de los robots móviles hace un poco más de un siglo se han creado una gran variedad de modelos de los cuales aquí se mencionarán algunos:

1912 *Electric dog* fue el primer robot móvil creado con la capacidad de seguimiento; fue creado con fines militares por John Hammonf Jr. y Benjamin Miessner en los Estados Unidos, su mecanismo consistía en tener dos celdas fotovoltaicas que controlaban su dirección al seguir un rayo de luz, mientras más luz recibía más avanzaba (ver Figura 2.1) [CT].

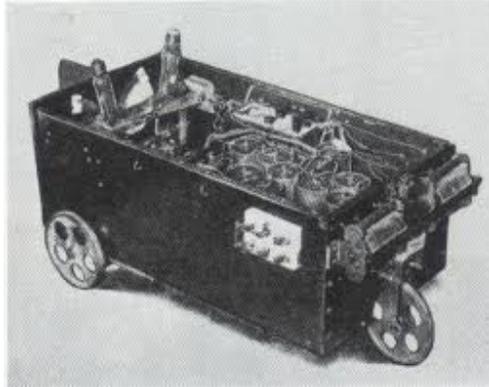


Figura 2.1: *Electric dog* fue el primer robot móvil creado con la capacidad de seguimiento [CT].

1940 *Elmer* y *Elsie* también conocidas como las tortugas de Walter en honor a su creador Gray Walter; los robots tenían la capacidad de interacción social, poseían sensores de luz, proximidad y comportamientos reactivos simples de forma analógica, pero algo destacado era la realización de tareas o actividades entre ellos (ver Figura 2.2) [FND03].

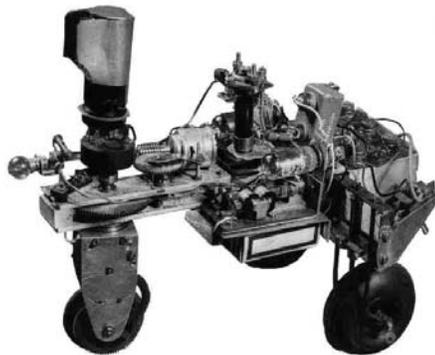


Figura 2.2: La tortuga de Walter tenía la capacidad de interacción social, poseían sensores de luz, proximidad y comportamientos reactivos simples de forma analógica [FND03].

1960 *Beast* tenía la capacidad de navegar por pasillos y corredores sin chocar utilizando un comportamiento reactivo, también podía buscar un contacto de electricidad y auto conectarse cuando lo necesitase (ver Figura 2.3) [WS05].

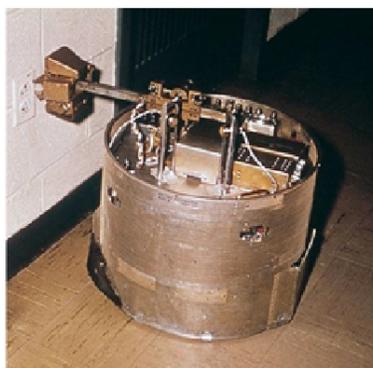


Figura 2.3: Robot *Beast* tenía la capacidad de navegar por pasillos, corredores, buscar contactos de electricidad y auto conectarse [WS05].

1966 *Shakey* fue el primer robot móvil con comportamiento no reactivo, tenía un sistema de planeación de acciones y un sistema de visión artificial con el que lograba el reconocimiento de objetos (ver Figura 2.4) [Cen84].

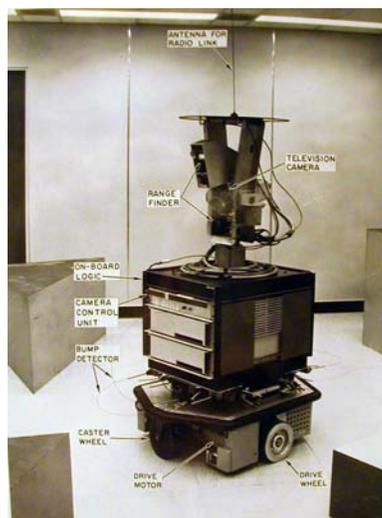


Figura 2.4: Robot *Shakey* fue el primer robot móvil con comportamiento no reactivo, tenía un sistema de planeación de acciones y un sistema de visión artificial [Cen84].

1968 *Hilare* llevaba una cámara monocromática y un escáner láser, de tal manera que en conjunto generaban una cámara con noción de profundidad, cargaba 14 sensores, la idea era obtener una navegación con planeación de movimientos utilizando un anillo de sensores (ver Figura 2.5) [GCV90].

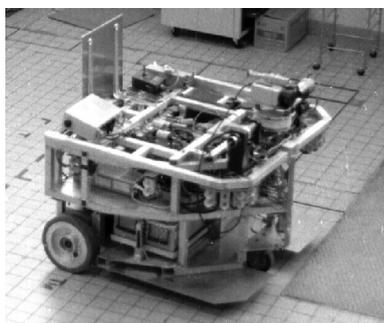


Figura 2.5: Robot *Hilare* tenía una cámara monocromática y un escáner láser, de tal manera que en conjunto generaban una cámara con noción de profundidad [GCV90].

1997 *Rhino* y *Minerva* fueron construidos para la asistencia e interacción con las personas en lugares públicos como el *Deutsches Museum Bonn* donde fueron desarrollados, tenían cámara estéreo, sensores sonares, táctiles e infrarrojos; *Minerva* era la versión mejorada de *Rhino* (ver Figura 2.6) [BCF+99,BCF+98].



(a) Robot Rhino.



(b) Robot Minerva.

Figura 2.6: Robots *Rhino* y *Minerva* fueron construidos para la asistencia e interacción con las personas en lugares públicos, tenían cámara estéreo, sonares, táctiles e infrarrojos [BCF+99,BCF+98].

2002 *Roomba*[®] se puede adquirir en el mercado, su única tarea es aspirar la casa, pero lo hace sin supervisión humana dado que es autónomo, pequeño y de forma circular (ver Figura 2.7) [FD06].



Figura 2.7: Robot *Roomba*[®] puede aspirar la casa, pero lo hace sin supervisión humana dado que es autónomo, pequeño y de forma circular [FD06].

2003 *iBotGuard* un sistema de guardia de seguridad creado por Liu en Taiwán, se constituía por dos partes, una era un robot móvil y la otra era un servidor que procesaba las tareas, el sistema podía reconocer personas y seguirlas mientras estuvieran frente a él (ver Figura 2.8) [LWF05].



Figura 2.8: Robot *iBotGuard* constituido por dos partes, un robot móvil y un servidor, el sistema en el servidor podía reconocer personas y seguirlas mientras estuvieran frente a él [LWF05].

2005 se realizó el sistema de visión artificial de Zadjel, podía detectar y seguir varias personas con el movimiento y color a través de algoritmos de redes bayesianas, operado desde la plataforma móvil de un robot y con una cámara [ZK05].

2006 *Nao* el robot de la compañía de *Aldebaran*, con tan solo 58 cm de altura y de forma humanoide, fue construido para que se convirtiera en un compañero o amigo, dado a que posee habilidades de reconocimiento, habla y escucha, además de que es usado en la competencia de *Robocup* para jugar fútbol creándose una liga donde solo pueden participar robots *Nao* (ver Figura 2.9) [NAO06].



Figura 2.9: Robot *Nao* mide solo 58 cm de altura y tiene forma humanoide, posee habilidades de reconocimiento, habla y escucha [NAO06].

2006 la visión panorámica de Chakravarty y Jarvis utilizaba una cámara panorámica estacionaria con la que el robot se podía dirigir, junto con un láser con el que obtenía información espacial y seguía a múltiples personas simultáneamente [CJ06].

2007 *Security Warrior* creado por Luo, diseñado para el servicio domiciliario contaba con dispositivos de sensores incluida una cámara de colores, láser, ultrasónicos y *wireless* (ver Figura 2.10) [LCLT07].



Figura 2.10: Robot *Security Warrior* contaba con dispositivos de sensores incluida una cámara de colores, láser, ultrasónicos y *wireless* [LCLT07].

2013 *Asimo* un robot construido por la compañía japonesa Honda, desde sus inicios ha tenido varias evoluciones; posee un sistema capaz de manipular su mapa de navegación (conjunto de información que el robot guarda internamente sobre el área donde desempeñara sus actividades), manipulación de objetos, creación de mejores rutas, tareas de habla, de acción, de procesamiento de imágenes, uso de comandos, monitoreo de su estado de comportamiento, entre otros (ver Figura 2.11) [Wor13,SWA⁺02].

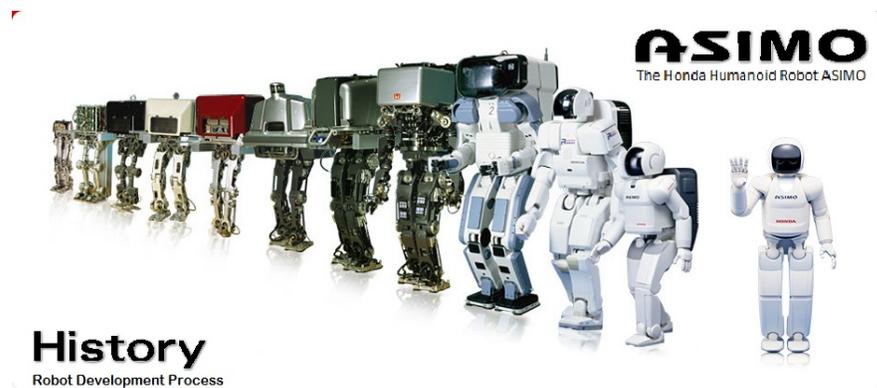


Figura 2.11: Evolución del robot *Asimo* que posee un sistema capaz de manipular su mapa de navegación, manipulación de objetos, creación de mejores rutas, tareas de habla, de acción, de procesamiento de imágenes, uso de comandos, monitoreo de su estado de comportamiento, entre otros [Wor13,SWA⁺02].

2.2. Herramientas de *software*

En la actualidad, el mundo de la robótica ha evolucionado notablemente, ahora es relativamente más fácil tener un robot que de inmediato hable, tome objetos y reconozca personas. Un robot como una computadora se compone principalmente por dos cosas: *hardware* y *software*. Si la institución cuenta con varios miles de euros puede comprar un robot nuevo, lo que resuelve la mitad del problema, la otra mitad ya está resuelta, gracias a comunidades internacionales de *software* ya es posible descargar código de módulos reemplazables y adaptarlos al robot para que éste pueda reaccionar por sí solo.

Para realizar este trabajo se obtuvo la oportunidad de trabajar con dos robots llamadas Justina y Judy, en la sección 4.1 se menciona de manera más detallada las características de los robots. A continuación se mencionan algunas herramientas de *software* consideradas de importancia para la implementación del trabajo en los robots.

2.2.1. Sistemas operativos

ROS es un sistema operativo para robots (llamado así por sus siglas en inglés, *Robot Operating System*), es un conjunto de bibliotecas de *software* y herramientas que ayudan a crear aplicaciones robóticas. Desde los *drivers* (*software* controlador de *hardware*) hasta los algoritmos del estado del arte, con potentes herramientas de desarrollo, y todo es de código abierto [ros14]. Pero en este proyecto no se utilizará este sistema dado a que los robots a utilizar cuentan con su propia metodología llamada *Blackboard* el cual se verá en el siguiente párrafo y cuyas formas de trabajar son similares.

El *Blackboard* es un sistema operativo centralizado que modera el tráfico de información entre diferentes módulos, los mensajes en formato TCP (*Transmission Control Protocol*) viajan de un módulo a otro a través del *Blackboard* dependiendo de cada una de las necesidades, comunicándose entre sí. La principal desventaja es que si falla el *Blackboard* falla todo el sistema, pero las ventajas son que si se muere algún otro módulo la afectación podría no ser fatal, y el agregar o eliminar módulos es relativamente fácil, en la Figura 2.12 se muestra la comunicación entre los módulos y el *Blackboard*.

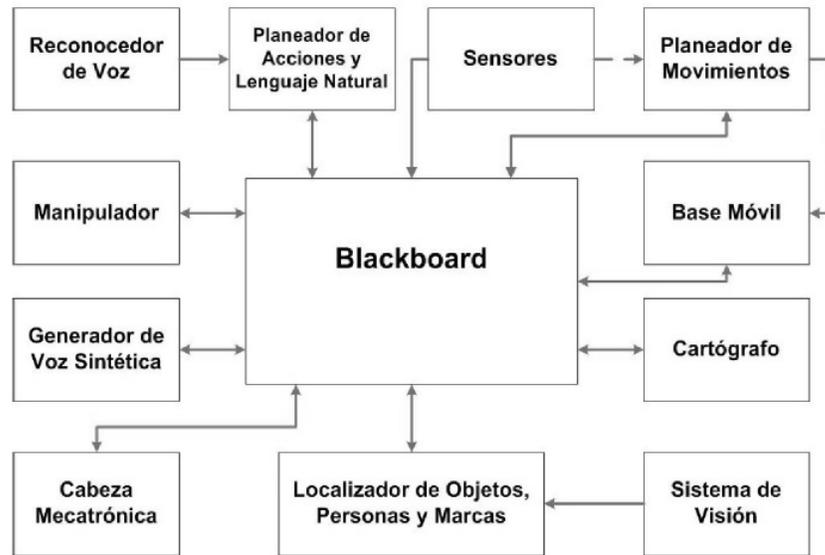


Figura 2.12: Diagrama del *BlackBoard* y su comunicación con los demás módulos.

2.2.2. Bibliotecas

OpenNITM (*Open Natural Interaction*) es un *software* multi-lenguaje y multi-plataforma que define las API (*Application Programming Interface*) para escribir aplicaciones que utilicen interacción natural o IN, la cual es la forma en que los humanos se relacionan con su exterior. Las API de OpenNITM se componen de un conjunto de interfaces para escribir aplicaciones de IN. “La API estándar de OpenNITM permite a los desarrolladores de aplicaciones desarrollar algoritmos usando escenas de 3D, utilizando los tipos de datos que se calculan a partir de la entrada de un sensor (por ejemplo, la representación de un cuerpo completo, la representación de un lugar de la mano, una matriz de los píxeles en un mapa de profundidad y así sucesivamente)”¹. Las aplicaciones pueden ser escritas independientemente de los proveedores de sensores o *middleware*.

OpenNITM cuenta con un algoritmo de seguimiento de humanos (ver Figura 2.13b), pero no será utilizado porque para conseguir el seguimiento se necesita de una calibración del humano para dar inicio, ésto significa que el humano debe de adoptar una posición específica mostrada en la Figura 2.13a para iniciar el seguimiento, y aunque es muy bueno no es lo que necesitamos en este proyecto, ya que una vez perdido al humano se necesita de volver a calibrar lo cual no es bien visto ni es permitido en las reglas de *Robocup*.

¹https://github.com/OpenNI/OpenNI/blob/master/Documentation/OpenNI_UserGuide.pdf

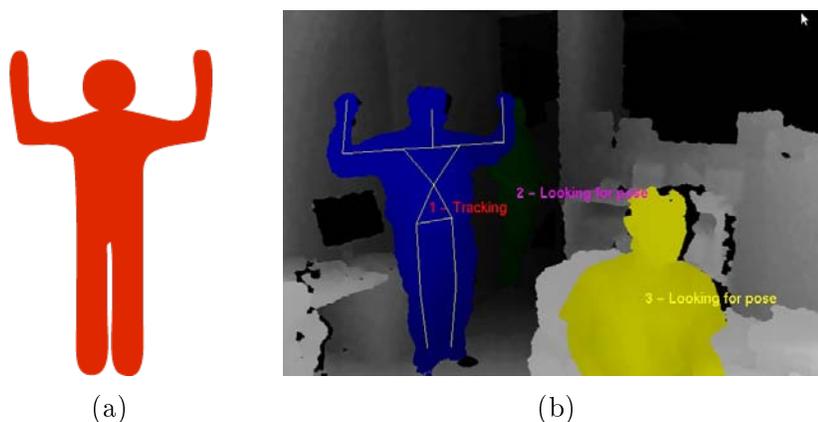


Figura 2.13: Calibración y seguimiento del esqueleto utilizando OpenNITM, en (a) la posición en la que el humano debe de estar para iniciar el seguimiento, y en (b) una pantalla de OpenNITM realizando una detección y seguimiento de un esqueleto.

Por otro lado, la biblioteca de OpenCV (*Open Source Computer Vision Library*) es la que será utilizada para la generación de los algoritmos de visión por sus múltiples ventajas como son: multilinguajes (C++, C, Python y Java), y multiplataformas (Windows, Linux, Mac OS, iOS y Android), posee más de 2,500 algoritmos optimizados, y además es compatible con CUDATM *Software Development Kit* de la compañía NVIDIA[®] y OpenCLTM (*Open Computing Language*), programas creados para agilizar el manejo de nubes de puntos. Entre los algoritmos de OpenCV existen algunos para la detección y reconocimiento de rostros, identificación de objetos, extracción de nube de puntos, así como, funciones básicas de manejo de colores, escalamiento y bordes, programadas de forma nativa en C++ [ope14].

2.3. Marco Teórico

2.3.1. Espacios de Color RGB y HSV

El estándar de las cámaras fotográficas digitales es entregar una matriz con los códigos de colores en formato RGB (Rojo (R), Verde (G) y Azul (B), por sus siglas en inglés *red*, *green* y *blue*) organizados de forma similar a la que tenía la escena al momento de tomar la fotografía, lo que facilita su procesamiento.

Existen otros modelos como el HSV (Matiz (H), Saturación (S) y Valor (V), por sus siglas en inglés *Hue*, *Saturation* y *Value*) que posee una mejor distribución de los colores cuando se desea reorganizarlos de acuerdo a su proximidad a los

colores base (pensemos que un color cualquiera pudiera ser un metal y es arrojado a un campo de fuerzas magnéticas influenciado por 3 imanes: rojo, verde y azul ubicados en posiciones estratégicas, el metal, o sea, el color será más atraído al iman cuyo color sea más parecido al suyo), esta distribución se concentra sobre un canal similar al de la visión humana [Smi78, CCC09], en la Figura 2.14 se presentan ambos modelos en su representación espacial, en el inciso (a) el modelo RGB y en el inciso (b) el modelo HSV.

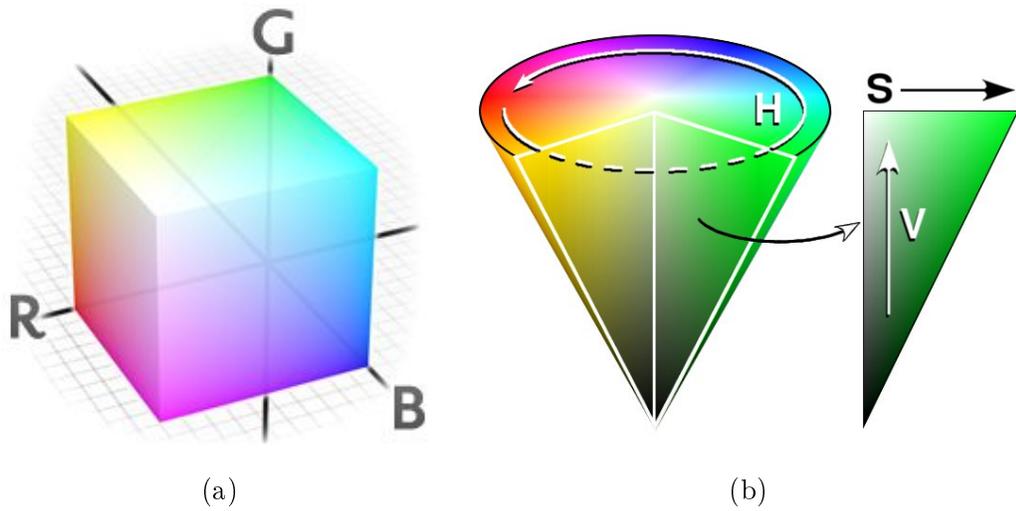


Figura 2.14: Representación gráfica de los modelos de espacio de color: (a) modelo RGB y (b) modelo HSV.

La transformación del espacio RGB (representado por las variables c_r , c_g y c_b referentes a los valores rojo, verde y azul, respectivamente), al espacio HSV (representado por las variables c_h , c_s y c_v referentes a los valores de la matiz, saturación y valor, respectivamente) es como sigue:

Primero se encuentra el canal de valor utilizando

$$c_v = \text{máx}(c_r, c_g, c_b), \quad (2.1)$$

donde la función $\text{máx}()$ obtiene el valor máximo de los valores ingresados y la función $\text{mín}()$ obtiene el valor mínimo de los valores ingresados, luego se continua con el canal de saturación usando

$$c_s = \begin{cases} \frac{c_v - \text{mín}(c_r, c_g, c_b)}{c_v}, & \text{si } c_v \neq 0, \\ 0, & \text{en caso contrario,} \end{cases} \quad (2.2)$$

y por último el canal de matiz con

$$c_h = \begin{cases} \frac{60(c_g - c_b)}{c_v - \min(c_r, c_g, c_b)}, & \text{si } c_v = c_r, \\ \frac{120 + 60(c_b - c_r)}{c_v - \min(c_r, c_g, c_b)}, & \text{si } c_v = c_g, \\ \frac{240 + 60(c_r - c_g)}{c_v - \min(c_r, c_g, c_b)}, & \text{si } c_v = c_b, \end{cases} \quad (2.3)$$

2.3.2. Cuantización vectorial

La cuantización vectorial es un método utilizado para la segmentación de puntos a base de distancias, como por ejemplo la distancia Euclidiana obtenida con $dist()$, es muy parecido al método de clasificación de *K-means*.

Para describir como funciona la cuantización vectorial supongamos que tenemos N_p cantidad de puntos $p_i \rightarrow \mathbb{R}^2$ a segmentar, y creamos un punto centroide inicial c_1 en el centro de los puntos encontrado a través de

$$c_1 = \frac{1}{N_p} \sum_{i=1}^{N_p} p_i, \quad (2.4)$$

después se realiza una iteración duplicando la cantidad de centroides creando a c_2 , teniendo a c_1 y c_2 con valores parecidos al antiguo c_1 diferenciándose únicamente por un pequeño umbral τ_{cv} . Ahora se generan dos cúmulos de puntos, uno para cada punto c_j usando

$$p_i \in P_j, \text{ si } dist(c_j, p_i) \text{ es mínima } \forall i \text{ y } \forall j, \quad (2.5)$$

donde P_j son conjuntos de puntos p_i que ubican a los centroides c_j . Ahora utilizando los puntos de P_j reubicamos a los c_j con

$$c_j = \frac{1}{N_{p_j}} \sum_{i=1}^{N_{p_j}} p_i \mid p_i \in P_j, \quad (2.6)$$

donde N_{p_j} es el número total de puntos pertenecientes al conjunto P_j . Luego solo se repite de nuevo la iteración empezando por la duplicación del número total de centroides.

El proceso continua hasta tener la cantidad de centroides que el usuario necesite, la principal desventaja es el procesamiento que ésto requiere ya que hay que recorrer todos los puntos y calcular su distancia a cada centroide, cada que se realice alguna división o reubicación, sin embargo, las ventajas son que la segmentación queda mejor definida como se puede apreciar en la Figura 2.15 donde

los puntos están mejor segmentados por la cuantización vectorial (b), que por el *K-means* (a), y que no requiere de información a priori como en el caso de *K-means*.

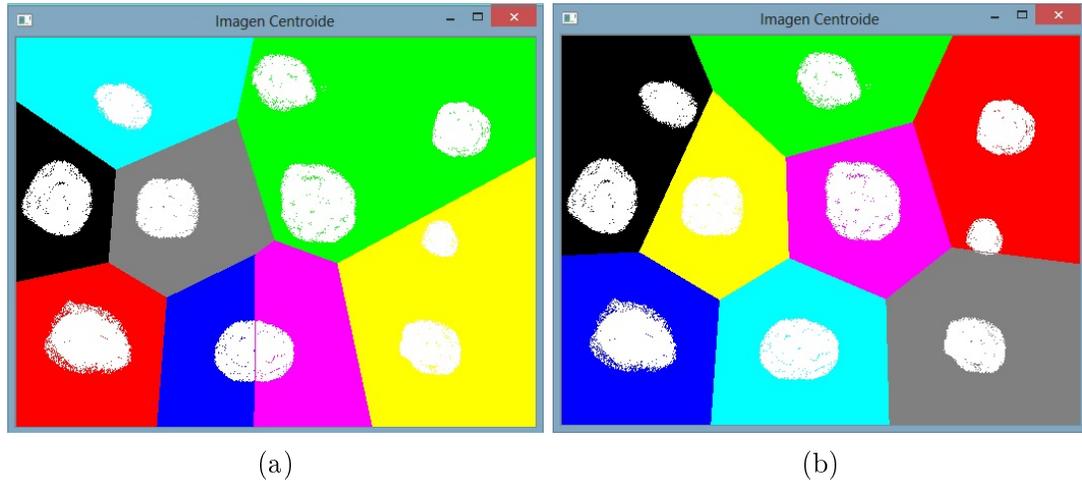


Figura 2.15: Pantalla de la clasificación de 10 cúmulos de puntos en 8 partes utilizando dos algoritmos: (a) *K-means* y (b) cuantización vectorial.

2.3.3. Láser

El láser de la marca Hokuyo utilizado en este proyecto es un sensor infrarrojo activo de rango que mide distancias en forma radial, es muy preciso con errores de apenas milímetros. Las características técnicas de este equipo son descritas en la sección 4.1.

El láser es capaz de detectar obstáculos enfrente de él, reportandolos con un formato de vector de valores (distancias); en la Figura 2.16 se muestra graficado este vector de información obtenido por el láser Hokuyo cuando la robot Justina está en un pequeño cuarto de 1.2×1.2 metros aproximadamente y enfrente se encuentra una persona.

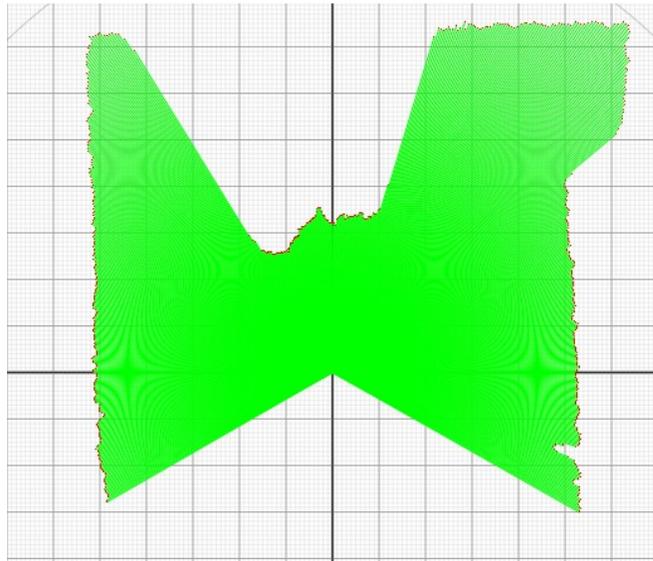


Figura 2.16: Graficación de una lectura del láser Hokuyo obtenida cuando la robot Justina está en un pequeño cuarto de 1.2×1.2 metros aproximadamente y enfrente se encuentra una persona.

2.3.4. Cámaras RGB-D

Las cámaras de profundidad son un *hardware* que permite obtener además de una imagen a color, la información de profundidad de cada píxel visualizado, a ésto se le llama proyección de puntos 3D discretos o comúnmente llamado nube de puntos.

El *Microsoft KinectTM* es una cámara que ofrece esta tecnología, aunque sus orígenes fueron los videojuegos, ahora en el área científica, es usado para la navegación de robots, ya que se puede modelar mapas, detectar obstáculos, objetos y en este caso personas con él [NWM13].

En la Figura 2.17a se muestra la graficación de la información de profundidad de la escena obtenida por el *KinectTM*, mientras más cercas más amarillo y mientras más lejos más negros son los píxeles, y en la Figura 2.17b la imagen RGB de la escena.



Figura 2.17: Pantallas de la graficación de la información obtenida por el *KinectTM*, en (a) se muestra la información de profundidad respecto al *KinectTM*, mientras más cercas más amarillos y mientras más lejos más negros son los píxeles, y en (b) la imagen RGB.

2.3.5. Campos potenciales

Los campos potenciales artificiales son una forma simple de navegación en robótica, el objetivo es guiar al robot con movimientos similares a los movimientos de una partícula con campos de atracción y repulsión artificiales, que para este caso en particular, el campo de atracción (fuerza de atracción) es el humano a seguir, los campos de repulsión (fuerza de repulsión) lo conforman el resto de las personas, objetos u obstáculos que se encuentren en el ambiente y al conjunto de ambas fuerzas le llamaremos simplemente las fuerzas.

Una forma metafórica de verlo es, por ejemplo, pensemos cuando tenemos una canica y la soltamos en el suelo, y si éste no es parejo pareciera que la canica obtuviera todas las pendientes de su alrededor y se decidiera andar por la más prolongada, y que ésto lo hace en cada momento, hasta terminar en una pendiente igual a cero o mínimo local. Ésto es semejante a lo que tiene que hacer el robot, navegar por las fuerzas sin caer en mínimos locales, lo cual es un gran problema sin conocimiento a priori, y éste no siempre se puede tener gracias a los entornos dinámicos.

En la Figura 2.18 se muestran los efectos de los campos potenciales de atracción y repulsión, identificados con el color verde y rojo respectivamente, de color azul el robot y con una línea azul la reacción probable del robot ante esos campos.

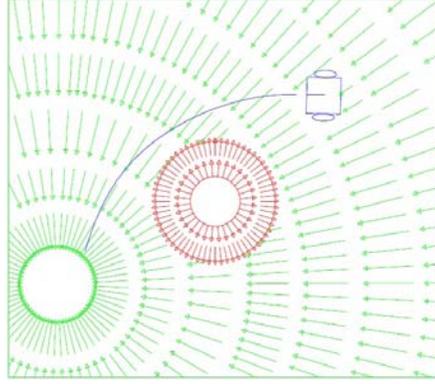


Figura 2.18: Muestra los efectos de los campos potenciales de atracción y repulsión, identificados con el color verde y rojo respectivamente, de color azul el robot y con una línea azul la reacción probable del robot ante esos campos.

Las fórmulas para el cálculo de las diferentes fuerzas fueron obtenidas de [Cru].

La fuerza de atracción f_{atr} es considerada como un vector en \mathbb{R}^2 y \mathbb{R} es el conjunto de números reales, f_{atr} es obtenido a partir de una constante de atracción k_{atr} con valor 5.8988, un vector unitario que va desde la posición del robot p_{rob} hasta la posición objetivo p_{obj} , como sigue:

$$f_{atr} = k_{atr} \left(\frac{p_{obj} - p_{rob}}{|p_{obj} - p_{rob}|} \right). \quad (2.7)$$

Para la fuerza de repulsión se utilizan:

$$f_{rep_{nx}} = \sqrt{\frac{1}{d_n} - \frac{1}{d_{inf}}} \left(\frac{1}{d_n^3} \right) (-d_n \cos(\alpha_n + \theta_R)), \quad (2.8)$$

$$f_{rep_{ny}} = \sqrt{\frac{1}{d_n} - \frac{1}{d_{inf}}} \left(\frac{1}{d_n^3} \right) (-d_n \sin(\alpha_n + \theta_R)) \quad (2.9)$$

y

$$f_{rep} = \frac{k_{rep}}{N_{obs}} \sum_{n=0}^{N_{obs}} f_{rep_n} \quad (2.10)$$

donde k_{rep} es una constante de repulsión con valor 5.9769, f_{rep} es la fuerza de repulsión total, $f_{rep_{nx}}$ y $f_{rep_{ny}}$ son las descomposiciones de f_{rep_n} en los ejes x y y respectivamente, d_n y α_n son la distancia y el ángulo de un obstáculo n repulsivo, N_{obs} es la cantidad de obstáculos repulsivos que se encontraron a través de d_{inf} , d_{inf} es la distancia de influencia de los obstáculos al robot valiendo 0.8634 metros, por lo tanto se debe de cumplir que $d_n < d_{inf}$ para que el obstáculo pueda ser tomado en cuenta, y θ_R es la orientación del robot.

La fuerza resultante f_{res} se obtiene de $f_{res} = f_{atr} + f_{rep}$ y para obtener la posición deseada p_{des} a la cual se debe de enviar el robot ocupamos $p_{des} = p_{rob} + f_{res}$.

A pesar de ser un método seguro para la evasión de colisiones y elegante por su sencillez matemática, los campos potenciales tienen las siguientes desventajas [CWZ07]:

Mínimo local que se referencia a la fuerza de atracción y la de repulsión cuando $-f_{atr} = f_{rep}$. La fuerza resultante es cero y el robot no alcanzará la meta, éste es el principal problema de los campos potenciales.

Meta inalcanzable que es observada si existe un obstáculo cerca del objetivo final, la fuerza repulsiva aumentará a medida que el robot se acerque a ésta. Si $f_{atr} < f_{rep}$, el robot no llegará al objetivo final, sino que se detendrá en un punto cercano donde $f_{res} = 0$.

Oscilaciones para determinadas configuraciones de obstáculos, por ejemplo en pasillos angostos, f_{rep} alejará al robot de un obstáculo; mientras el robot se aleja esta fuerza disminuye y si existe un obstáculo cercano que empuje de nuevo al robot hacia el primer obstáculo, el robot se alejará y acercará de éstos continuamente por lo que el robot se desplazará como una onda oscilatoria en el plano, lo que hacen que los campos potenciales se usen en combinación con otros métodos para la planeación de trayectorias.

2.3.6. Filtro de Kalman

El filtro Kalman es un método de estimación inventado por Rudolf E. Kalman en 1960 para dar seguimiento o predicción (según se use) a una partícula, funciona con un conjunto de ecuaciones matemáticas que dan una solución óptima recursiva al problema del filtrado lineal de datos discretos. A partir de una serie de observaciones, puede estimar un resultado en el tiempo t con las $t - 1$ observaciones previas; consiste en 2 pasos: predicción y corrección [MJ12].

En la predicción se calcula a priori con:

$$X'_t = AX_{t-1} \quad (2.11)$$

y

$$P'_t = AP_{t-1}A^T + Q, \quad (2.12)$$

donde A es la matriz de transición de estados, X_t es el estimado a priori del vector de estados, P_t es la covarianza del error asociada a la estimación a priori y Q es la matriz de covarianza del proceso (también se puede ver como una variable para regular a P_t).

En la corrección se realiza primero una actualización del resultado obtenido en la predicción y después un cálculo a posteriori, por cada ciclo se genera un nuevo estado que sería la nueva estimación usando [Ram03]:

$$K_t = \frac{P'_t H^T}{HP'_t H^T + R}, \quad (2.13)$$

$$X_t = X'_t + K_t (Z_t - HX'_t) \quad (2.14)$$

y

$$P_t = (I - K_t H) P'_t, \quad (2.15)$$

donde K es una matriz auxiliar, Z_t es el vector de mediciones (valores recibidos) al momento t , H_t es la matriz que indica la relación $\frac{Z_t}{X_t}$ en una suposición ideal de que no hubiera ruido en las mediciones y R es la matriz de covarianza del ruido de las mediciones.

Ahora conocemos algunas de las herramientas más utilizadas en los robots de hoy, éstas les ayudan a realizar sus tareas de la mejor forma posible haciendo uso de la tecnología disponible, así también los conocimientos vistos hasta ahora comprenden los fundamentos básicos para el desarrollo de los algoritmos a utilizar. Cada sección es una columna teórica que soporta la arquitectura sugerida para el seguimiento de personas.

Capítulo 3

Desarrollo e implementación

En este Capítulo se presentan los algoritmos desarrollados para conseguir el objetivo del proyecto, para ello se utiliza un sensor láser Hokuyo y un sensor *KinectTM*, equipo *hardware* que estructura la arquitectura de Justina y Judy descrita detalladamente en la sección 4.1. Se desarrollaron 2 métodos de seguimiento adaptados a los métodos ya implementados en las robots.

3.1. Sensado de personas con láser

Un barrido completo del láser acumula una serie de puntos que proveen en cada uno de ellos la distancia del objeto más cercano al láser, dado que el escaneo se hace sobre un plano únicamente, se generan varias problemáticas al momento de querer sensar personas, en la implementación se utilizó el láser que se encuentra de manera semifija en el dorso de la robot, a una altura de entre 30 a 40 cm sobre el nivel del suelo, ésto es una ventaja porque se pueden utilizar las piernas para sensar a las personas y es útil ya que no importa mucho la estatura o edad de la persona mientras tenga un par de piernas, pero también es importante su limitación si la persona en la parte de abajo posee otra cosa que no sean un par de piernas (como por ejemplo un bastón, muletas, sillas de ruedas o andadera, entre otros) la detección será más difícil o imposible con el algoritmo propuesto.

Otro factor importante es la posición de la persona, si se encuentra caminando o parada, con las piernas juntas o separadas, ya sea de frente o perfil, cada una de estas posiciones será una manera diferente de ver a las personas y tendremos que identificarlas. Acotando el caso podemos llegar a tres simples formas que son las siguientes: (i) PS (piernas separadas o aisladas) cuando las piernas poseen una cierta separación entre ellas, (ii) PD (piernas desfasadas o caminando) cuando la persona esté caminando, por lo general tendrá una pierna adelante de la otra; y

(iii) PJ (piernas juntas) cuando las piernas están juntas, sin embargo, si éste es el caso únicamente se podrá detectar una sola pierna.

En la Figura 3.1 se muestra los 3 patrones de piernas mencionados anteriormente cuando el láser está de forma horizontal enfrente de las piernas.

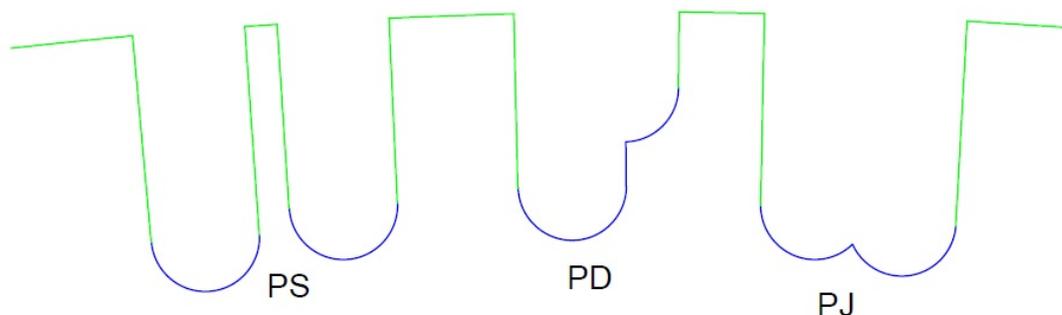


Figura 3.1: Patrones de piernas PS (piernas separadas o aisladas), PD (piernas desfasadas o caminando) y PJ (pierna juntas) cuando el láser está en forma horizontal enfrente de las piernas.

3.2. Reconocedor de piernas basado en pendientes

Una vez obtenido el mapa de distancias con el láser, se procede a identificar las piernas, y lo podemos hacer por medio del ángulo entre el cúmulo de puntos y la posición física del láser. El láser entrega un vector de puntos p_{dis} ordenado de acuerdo al ángulo en que se encuentran los puntos con respecto al láser, la separación de este vector en otros vectores o cúmulos de puntos es a partir de un umbral τ_{prof} , tal que si $|p_{dis_i} - p_{dis_{i+1}}| > \tau_{prof}$ se cierra un cúmulo en i y se abre uno nuevo en $i + 1$, donde p_{dis_i} es la distancia del punto i al láser e i es la posición del punto en el vector p_{dis} .

Para el reconocimiento de piernas analizaremos cada cúmulo por separado, primero podemos calcular el ancho del cúmulo utilizando $dist(p_{ini}, p_{fin})$ donde p_{ini} y p_{fin} son los puntos de inicio y fin del cúmulo y que nos ayudarán de forma biométrica a la identificación: (i) si es una sola pierna, si mide de 10 a 25 cm, (ii) si son dos piernas juntas, si mide de 20 a 50 cm o (iii) si es otra cosa, si es menor a 10 cm o mayor a 50 cm.

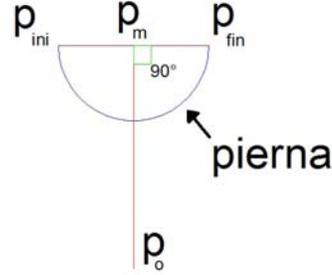


Figura 3.2: Geometría de los puntos principales para la identificación de piernas: los puntos p_{ini} y p_{fin} son los puntos de inicio y final del cúmulo de puntos a evaluar, respectivamente, p_m el punto medio del segmento $\overline{p_{ini}p_{fin}}$ y p_o el origen (posición del láser).

Una vez que el ancho coincida con las medidas biométricas se revisarán las pendientes. Sean los puntos p_{ini} y p_{fin} los puntos a evaluar, p_m el punto medio del segmento $\overline{p_{ini}p_{fin}}$ y p_o el origen (posición del láser), en la Figura 3.2 se muestra la distribución de estos puntos.

Ahora p_m es obtenido con

$$p_m = \frac{p_{ini} + p_{fin}}{2} \quad (3.1)$$

y dado que las piernas tienen una forma semicircular (en el mejor de los casos), el ángulo θ entre los segmentos $\overline{p_{ini}p_{fin}}$ y $\overline{p_m p_o}$ debe de ser casi recto (90°) obteniéndose con

$$m_{p_{fin}p_{ini}} = \frac{p_{fin_y} - p_{ini_y}}{p_{fin_x} - p_{ini_x}}, \quad m_{p_m p_o} = \frac{p_{o_y} - p_{m_y}}{p_{o_x} - p_{m_x}} \quad (3.2)$$

y

$$\tan \theta = \frac{m_{p_{ini}p_{fin}} - m_{p_m p_o}}{1 + m_{p_{ini}p_{fin}} m_{p_m p_o}}, \quad (3.3)$$

así que se puede manejar un umbral τ_θ para el ángulo θ , y poder tomar la decisión de si es pierna o no.

En la Figura 3.3 se muestran los posibles resultados al utilizar la detección geométrica con cada patrón descrito anteriormente, donde la opción a) es una sola pierna, la opción b) son dos piernas juntas, la opción c) dos piernas, una adelante de la otra, y la opción d) un cúmulo que no es pierna el cual debería ser rechazado.

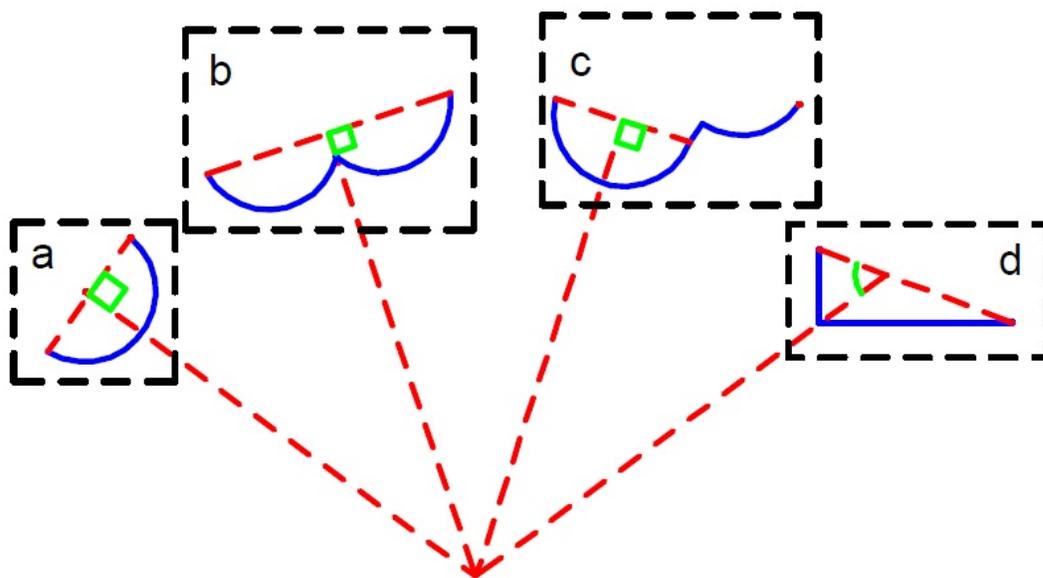


Figura 3.3: Detección en patrones de piernas: (a) es una sola pierna, (b) son dos piernas juntas, (c) dos piernas, una adelante de la otra, y (d) un cúmulo que no es pierna.

Para lograr estos resultados se realizaron una serie de experimentos, éstos consistieron en tomar varias muestras de diferentes piernas orientadas en diferentes direcciones, concluyendo con el método propuesto y sugiriendo el valor de τ_θ . Este método no es perfecto dado arroja también resultados falsos en calidad de positivos pero con la implementación de un buen método de rastreo, es posible no confundirse durante la realización de la tarea principal.

En el Algoritmo 1 primero se obtiene la pendiente $m_{p_{ini}p_{fin}}$ correspondiente a los puntos de la pierna, donde MAX es un valor muy grande, después se obtiene el punto medio p_m y la pendiente $m_{p_m p_o}$, luego obtenemos el ángulo θ para compararlo con el umbral τ_θ y entonces regresar un valor booleano como respuesta, realizando esto para cada cúmulo de puntos.

Algorithm 1: Verificación de piernas**Entrada:** Los valores p_{ini_x} , p_{ini_y} , p_{fin_x} y p_{fin_y} **Salida:** **cierto** si p_{ini_x} , p_{ini_y} , p_{fin_x} y p_{fin_y} corresponden a una pierna y **falso** en caso contrario

```

1  $res \leftarrow \text{falso}$ 
2 si  $p_{ini_x} \neq p_{fin_x}$  entonces
3    $m_{p_{ini}p_{fin}} \leftarrow \frac{p_{fin_y} - p_{ini_y}}{p_{fin_x} - p_{ini_x}}$ 
4 si no
5    $m_{p_{ini}p_{fin}} \leftarrow MAX$ 
6  $p_{m_x} \leftarrow \frac{p_{ini_x} + p_{fin_x}}{2}$ 
7  $p_{m_y} \leftarrow \frac{p_{ini_y} + p_{fin_y}}{2}$ 
8 si  $p_{m_x} \neq 0$  entonces
9    $m_{p_{m}p_o} \leftarrow \frac{p_{m_y}}{p_{m_x}}$ 
10 si no
11    $m_{p_{m}p_o} \leftarrow MAX$ 
12  $\theta \leftarrow \left| \frac{m_{p_{m}p_o} - m_{p_{ini}p_{fin}}}{1 + (m_{p_{m}p_o} m_{p_{ini}p_{fin}})} \right|$ 
13 si  $\theta > \tau_\theta$  entonces
14    $res \leftarrow \text{cierto}$ 
15 retorna  $res$ 

```

3.3. Detección de humanos

Utilizando la información 3D (también conocida como nube de puntos) del *KinectTM*, es posible detectar humanos al computar sus normales y procesarlas de acuerdo a las características que la robot tenga del humano como su altura.

Esta nube de puntos posee su propio sistema coordenado \mathbb{S}_K , y los valores de sus puntos se encuentran representados por una matriz $K_P[N_{K_i}][N_{K_j}]$ de vectores de profundidad y una matriz $K_{RGB}[N_{K_i}][N_{K_j}]$ de los colores RGB, ambas de tamaño $N_{K_i} \times N_{K_j}$. Ahora, por la estructura física de la robot (que se verá más adelante en la sección 4.1) el *KinectTM* utilizado se encuentra en la parte superior de su cabeza, lo que genera que la robot maneje otro sistema coordenado \mathbb{S}_R y dado a que el método desarrollado utiliza la discriminación de información desde \mathbb{S}_R (llamada discriminación del no torso), necesitamos realizar una transformación geométrica de \mathbb{S}_K a \mathbb{S}_R , tal que $\mathbb{S}_K, \mathbb{S}_R \rightarrow \mathbb{R}^3$.

Dado que la cabeza solo posee la capacidad de hacer cabeceo y paneo (movimientos hacia ambos lados, para arriba y para abajo), para realizar la transformación necesitamos usar la matriz de transformación

$$M_{R_{xy}T}(\alpha, \phi, t_x, t_y, t_z) = \begin{bmatrix} \cos \phi & 0 & -\sin \phi & t_x \\ (\sin \alpha)(\sin \phi) & \cos \alpha & -(\sin \alpha)(\cos \phi) & t_y \\ -(\cos \alpha)(\sin \phi) & \sin \alpha & (\cos \alpha)(\cos \phi) & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

donde α y ϕ son los ángulos de inclinación de \mathbb{S}_K a \mathbb{S}_R sobre los ejes x y y , respectivamente, t_x, t_y y t_z son los desplazamientos desde el origen de \mathbb{S}_K al origen de \mathbb{S}_R y $M_{R_{xy}T}$ es el resultado de

$$M_{R_{xy}T} = M_{R_x} M_{R_y} M_T, \quad (3.5)$$

donde M_{R_x} y M_{R_y} son las matrices homogéneas de rotación x y y , respectivamente, y M_T es la matriz homogénea de traslación.

En la Figura 3.4 se muestra un punto $K_P[i][j]$ y su referencia con respecto a \mathbb{S}_R y \mathbb{S}_K , también se muestran las distancias t_x, t_y y t_z , y los ángulos de giro α y ϕ .

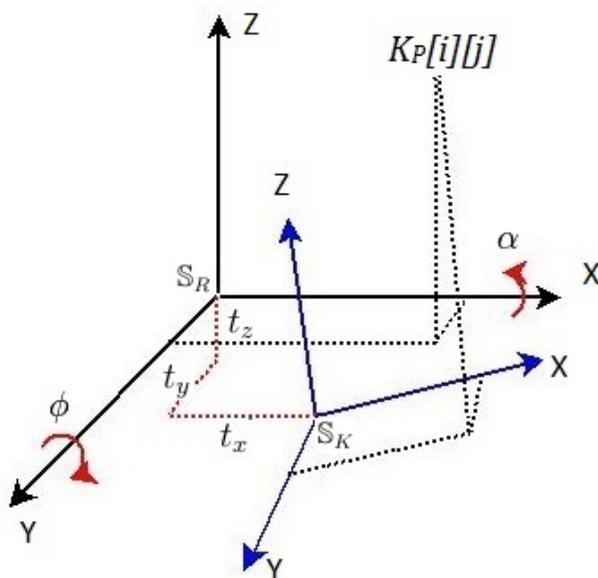


Figura 3.4: Muestra un punto $K_P[i][j]$ y su referencia con respecto a \mathbb{S}_R (sistema coordinado del robot) y \mathbb{S}_K (sistema coordinado del *Kinect*TM), también se muestran las distancias t_x, t_y y t_z , y los ángulos de giro α y ϕ para realizar la transformación de \mathbb{S}_K a \mathbb{S}_R .

Para obtener la normal $K_{\vec{n}}[i][j]$ de $K_P[i][j]$ donde $i \in N_{K_i}$ y $j \in N_{K_j}$ es necesario realizar

$$\vec{v} = \text{normalizar}(v) = \frac{v}{|v|}, \quad (3.6)$$

$$\vec{u} = \text{normalizar}(K_P[i-2][j-2] - K_P[i+2][j+2]), \quad (3.7)$$

$$\vec{v} = \text{normalizar}(K_P[i-2][j+2] - K_P[i+2][j-2]) \quad (3.8)$$

y

$$K_{\vec{n}}[i][j] = \vec{u} \times \vec{v} \quad (3.9)$$

donde los puntos $K_P[i-2][j-2]$, $K_P[i+2][j+2]$, $K_P[i-2][j+2]$ y $K_P[i+2][j-2]$ son 4 vecinos de $K_P[i][j]$, \vec{u} y \vec{v} son vectores auxiliares normalizados para encontrar a $K_{\vec{n}}[i][j]$, en la Figura 3.5 se visualizan éstos datos.

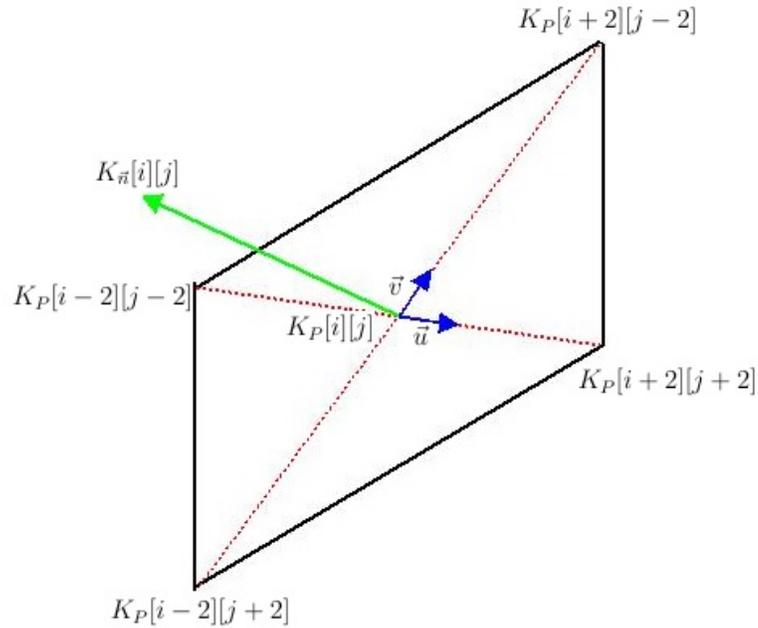


Figura 3.5: Muestra la localización de los puntos vecinos $K_P[i-2][j-2]$, $K_P[i+2][j+2]$, $K_P[i-2][j+2]$, $K_P[i+2][j-2]$, y la $K_{\vec{n}}[i][j]$ con respecto a $K_P[i][j]$.

En la Figura 3.6 se muestra la información de una escena obtenida con el *Kinect*TM, y en cada inciso se graficó lo siguiente: en (a) la imagen RGB (K_{RGB}) de la escena, en (b) las normales $K_{\vec{n}}$ de la escena, en (c) el componente x de las normales $K_{\vec{n}}$ de la escena, en (d) el componente y de las normales $K_{\vec{n}}$ de la escena y en (e) el componente z de las normales $K_{\vec{n}}$ de la escena.

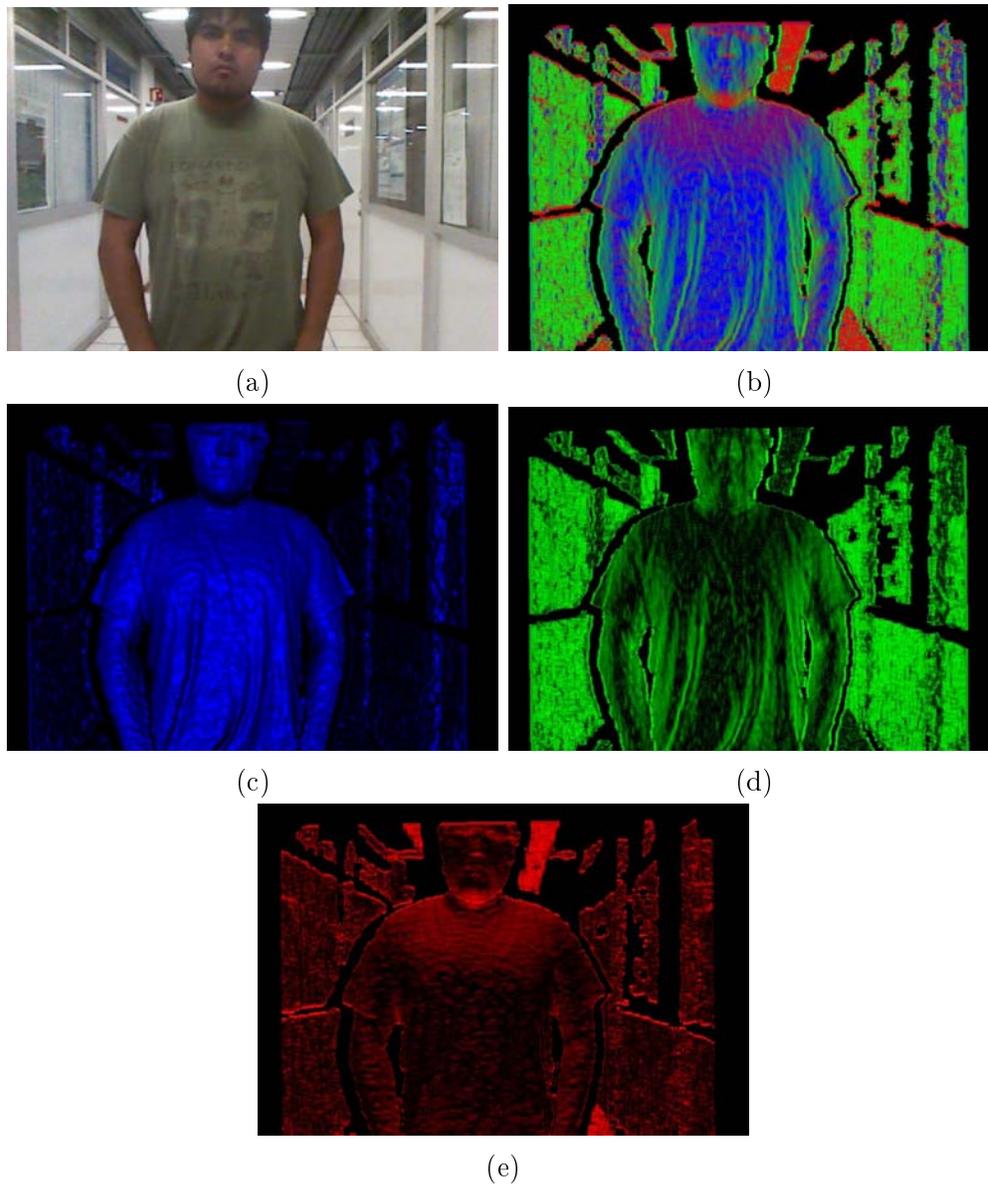


Figura 3.6: Graficación de la información de una escena obtenida con el *Kinect*TM donde (a) es la imagen RGB, (b) son las normales $K_{\vec{n}}$, (c) es el componente x de las normales $K_{\vec{n}}$, (d) es el componente y de las normales $K_{\vec{n}}$, (e) es el componente z de las normales $K_{\vec{n}}$ y $K_{\vec{n}}$ es la matriz de normales de la escena.

Para la detección de humanos se realizaron una serie de pruebas que consistieron en tomar varias muestras de humanos enfrente de un *Kinect*TM en diferentes posiciones. Se analizaron los diferentes patrones visualizados en estas pruebas dando como resultado que solo es necesario procesar el componente x de las normales $K_{\bar{n}}$, debido a que éste fue, él que mejor resultado arrojó con el menor costo de procesamiento durante el desarrollo de las pruebas, determinando un umbral $\tau_{\bar{n}} = 0.3$ a dicho componente, por lo cual se establece que

$$K_{\tau_{\bar{n}}}[i][j] = \begin{cases} 1, & \text{si } K_{\bar{n}_x}[i][j] > \tau_{\bar{n}}, \\ 0, & \text{en caso contrario,} \end{cases} \quad (3.10)$$

donde $K_{\tau_{\bar{n}}}[i][j]$ es el valor asignado de acuerdo a $K_{\bar{n}_x}[i][j]$ que es el valor del componente x de la normal $K_{\bar{n}}[i][j]$. Ahora para agilizar el procesamiento, a partir de $K_{\tau_{\bar{n}}}$ se creó una malla cuadrículada (K_M), donde cada cuadro es inicializado en 0 y hace referencia a un conjunto de 8×8 celdas de $K_{\tau_{\bar{n}}}$. Para procesar la presencia de un posible torso se realizará un ventaneo de 15×7 sobre K_M , si

$$K_M[i_M][j_M] = \begin{cases} 1, & \text{si } \sum_{i_2=0}^{55} \sum_{j_2=0}^{119} K_{\bar{n}_x}[i+i_2][j+j_2] > \tau_T, \\ K_M[i_M][j_M], & \text{en caso contrario,} \end{cases} \quad (3.11)$$

donde τ_T es un umbral para establecer si la sumatoria obtenida de la ventana es un torso, $i = 0, \dots, N_{K_i} - 55$, $j = 0, \dots, N_{K_j} - 119$, $i_M = \left\lfloor \frac{i}{8} \right\rfloor, \left\lfloor \frac{i+8}{8} \right\rfloor, \dots, \left\lfloor \frac{i+55}{8} \right\rfloor$ y $j_M = \left\lfloor \frac{j}{8} \right\rfloor, \left\lfloor \frac{j+8}{8} \right\rfloor, \dots, \left\lfloor \frac{j+119}{8} \right\rfloor$, los valores 119 y 55 corresponden al tamaño de la ventana en la matriz $K_{\bar{n}}$, es decir, $15(8) \times 7(8) = 120 \times 56$.

Cuando termina el ventaneo, los puntos $K_M[i_M][j_M] = 1$ serán referenciados a $K_{\tau_{\bar{n}}}$ y serán agrupados por vecindad (si son vecinos son un grupo en caso contrario) para pasar al siguiente paso.

A continuación en la Figura 3.7 se muestran 4 pantallas de las salidas de la discriminación del no torso, los incisos (a) y (b) son dos imágenes en las cuales se muestran los píxeles que se encuentran a la altura de la camisa del humano a seguir, los incisos (c) y (d) son imágenes en las cuales únicamente se muestran los píxeles que pueden ser humanos.

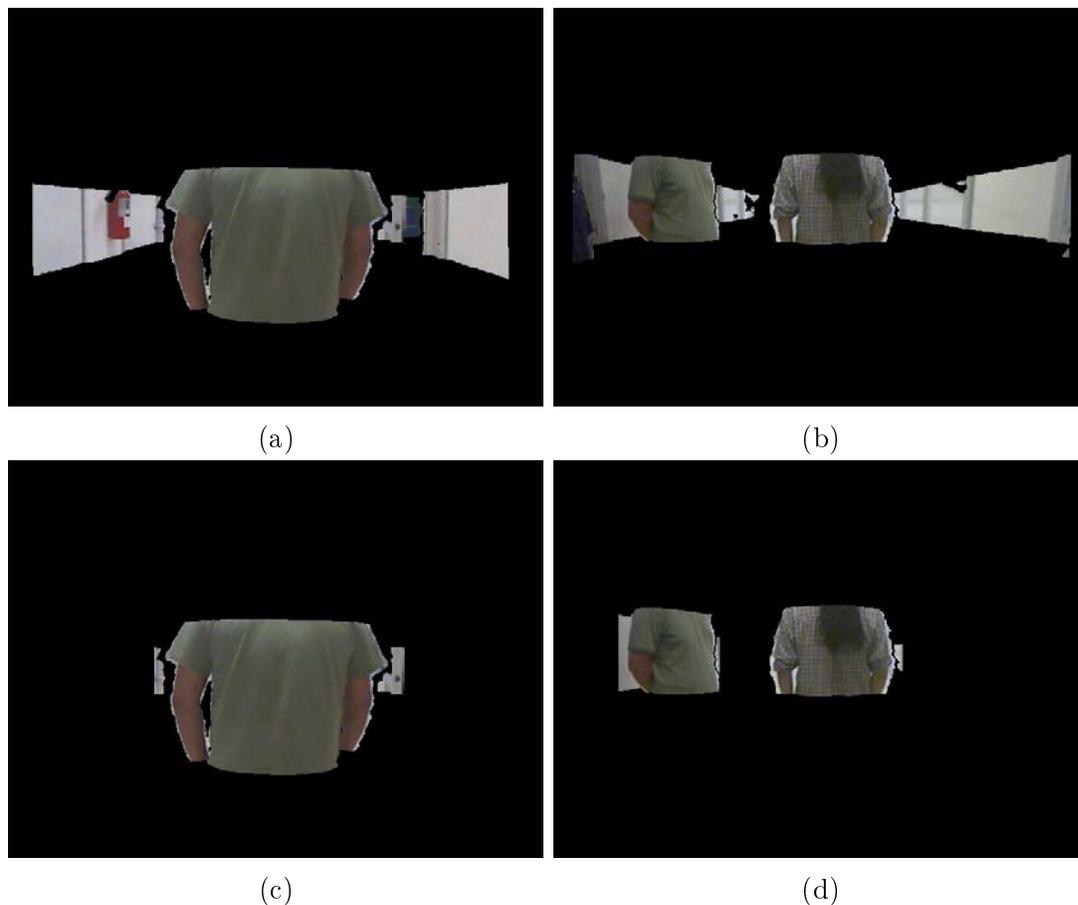


Figura 3.7: Muestra 4 pantallas de las salidas de la discriminación del no torso, los incisos (a) y (b) son dos imágenes en las cuales se muestran los píxeles que se encuentran a la altura de la camisa del humano, los incisos (c) y (d) son imágenes en las cuales únicamente se muestran los píxeles que pueden ser humanos.

3.4. Entrenamiento y reconocimiento de camisa

Para el reconocimiento del humano (operador) es necesario primero, al momento de iniciar la prueba, que el operador se presente enfrente de la robot para que la robot lo observe y se grabe las características del operador (su altura y los colores de la prenda que lleva sobre su torso tanto de enfrente como de espalda) para realizar su reconocimiento (a este proceso le llamaremos entrenamiento del operador), lo que implica realizar una discriminación por altura, una discriminación del no torso y suponiendo que en el momento de estar entrenando al operador, el operador es la persona más cercana a la robot, se tomará el torso más cercano a la robot de los torsos detectados, ahora se hará referencia al grupo de píxeles del torso utilizando la conexión de K_P a K_{RGB} y se depositaran en una imagen para escalarla a un tamaño *default* de 120×160 píxeles (medidas promedio extraídas durante el desarrollo) para considerarlas como los patrones iniciales del entrenamiento.

En la Figura 3.8 se muestran los dos patrones originales del entrenamiento, el inciso (a) es la parte de enfrente del operador (pecho y estómago) y el inciso (b) es la espalda.

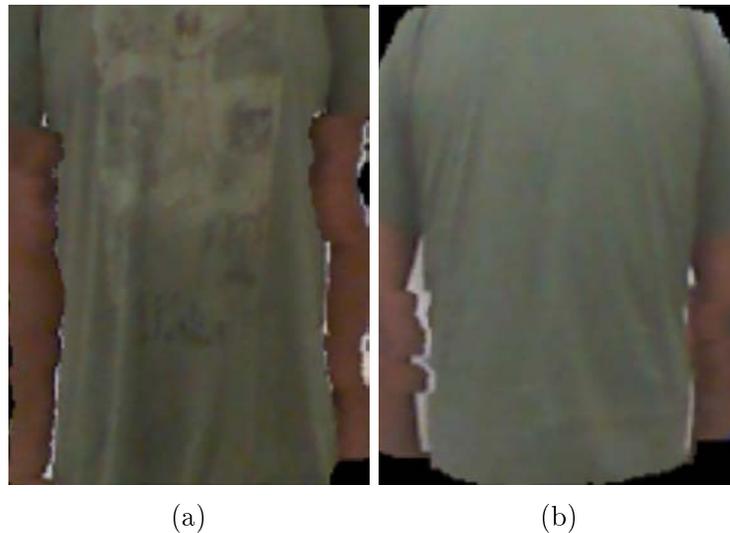


Figura 3.8: Patrones iniciales del entrenamiento, el inciso (a) es la parte de enfrente del operador (pecho y estómago) y el inciso (b) es la espalda.

Por cada patrón se generan varios patrones más con diferentes distorsiones o cambios, con la idea de emular las situaciones más frecuentes, en las que se enlistan rotación, escalamiento e iluminación.

Para el caso de las rotaciones los patrones creados consisten en quitar los píxeles del brazo izquierdo, los del brazo derecho o quitar los de ambos brazos más cierto porcentaje del torso y luego escalarlo al tamaño *default* con una función de interpolación cúbica. En la Figura 3.9 se presentan estos cambios, en el inciso (a) la camisa sin el brazo izquierdo, en (b) la camisa sin ambos brazos y en (c) la camisa sin el brazo derecho.

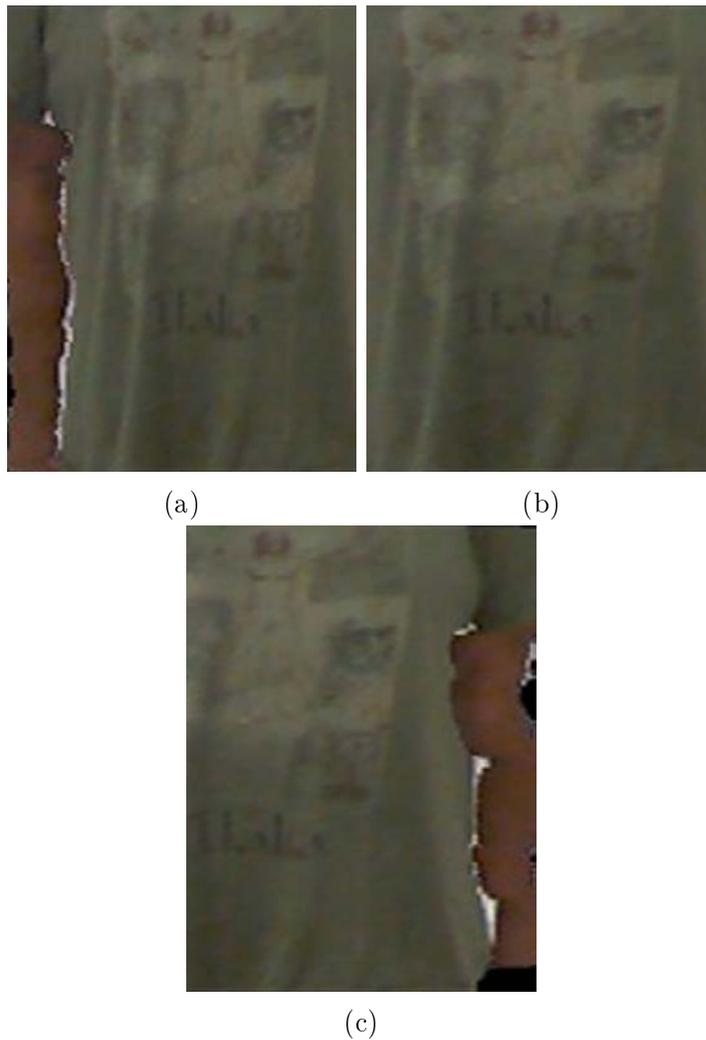


Figura 3.9: Distorsiones de rotación sobre un patrón original, en el inciso (a) la camisa sin el brazo izquierdo, en (b) la camisa sin ambos brazos y en (c) la camisa sin el brazo derecho.

Para las distorsiones en las escalas se crearon nuevos patrones a partir de los patrones anteriores, el proceso consiste en reducir la imagen hasta cierto punto con una función de interpolación cúbica, aplicar un filtro de suavizado gaussiano y por último regresar la imagen resultante a la escala *default* con una función de interpolación cúbica. En la Figura 3.10 se presentan estos cambios, en el inciso (a) la camisa sin brazos con resolución *default*, en (b) la camisa sin brazos con $\frac{3}{4}$ de la resolución *default* y en (c) la camisa sin brazos con $\frac{1}{2}$ de la resolución *default*.

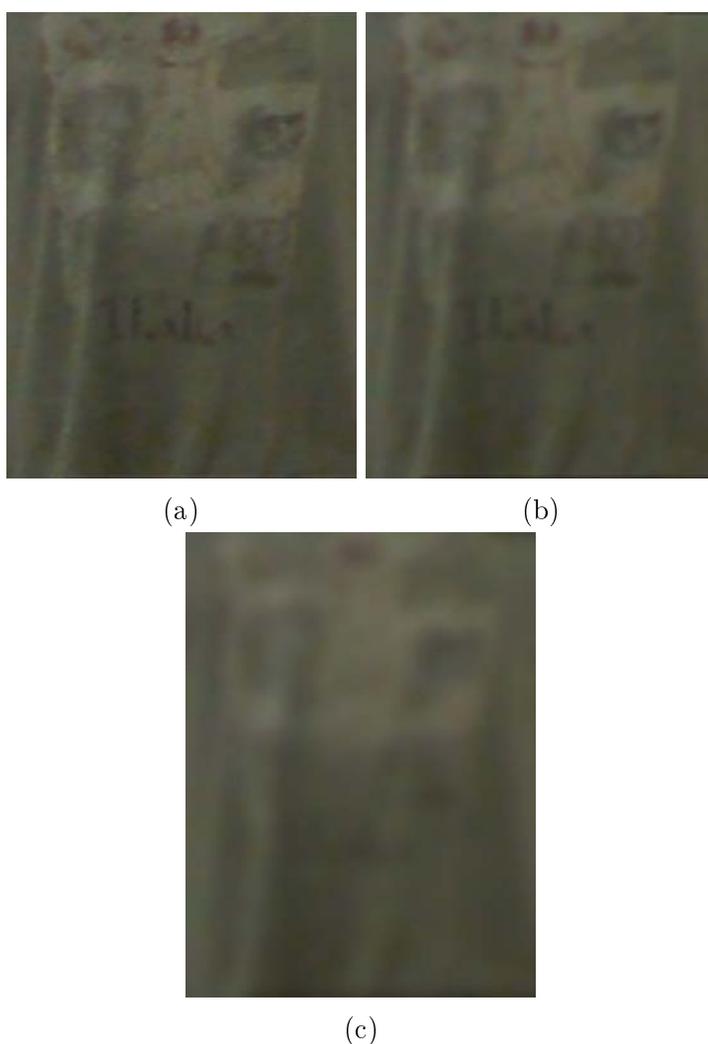


Figura 3.10: Distorsiones de escalamiento sobre un patrón de rotación, en el inciso (a) la camisa sin brazos con resolución *default*, en (b) la camisa sin brazos con $\frac{3}{4}$ de la resolución *default* y en (c) la camisa sin brazos con $\frac{1}{2}$ de la resolución *default*.

Por último, para crear las distorsiones de iluminación se tomaron todos los patrones anteriormente descritos y se modificaron sus valores originales, aumentando y disminuyendo el valor de cada canal, para modificar los niveles de gris como hiperintensos haciendo

$$v_{nuevo} = ((1 - v_{actual}) p_{hip}) + v_{actual} \quad (3.12)$$

o hipointensos haciendo

$$v_{nuevo} = v_{actual} - (v_{actual} p_{hip}), \quad (3.13)$$

donde v_{nuevo} es el resultado de la operación, v_{actual} es el valor actual del pixel en una escala de $[0, 1]$ y p_{hip} es el porcentaje de hiperintensos o hipointensos del pixel.

Esta distorsión debe de afectar a las demás distorsiones, ya que la iluminación siempre está presente. En la Figura 3.11 se presentan estos cambios, en el inciso (a) es el patrón original obscurecido, en (b) es el patrón original y en (c) es el patrón original aclarado.

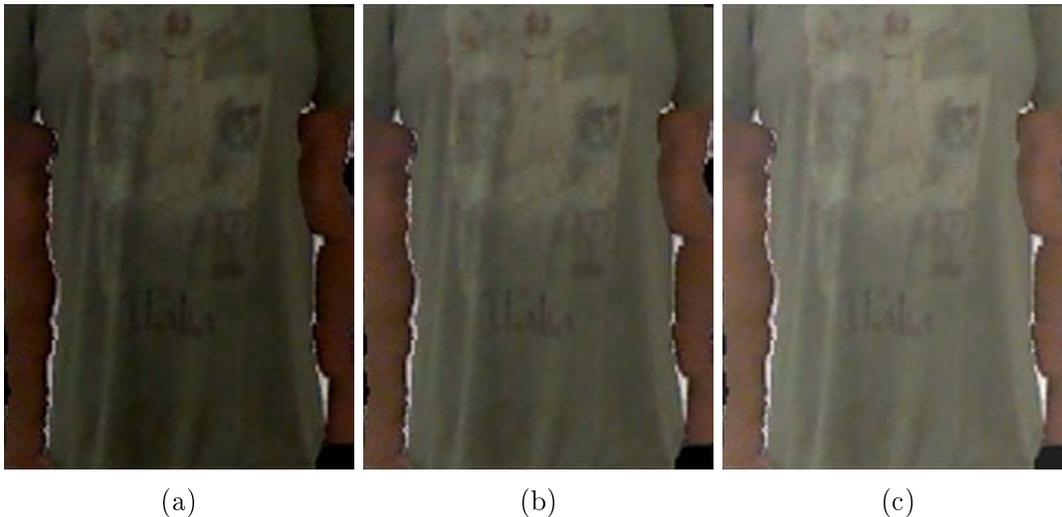


Figura 3.11: Distorsiones de iluminación sobre un patrón original, en el inciso (a) es el patrón original obscurecido, en (b) es el patrón original y en (c) es el patrón original aclarado.

Por cada patrón se genera un descriptor el cual se construye en 2 partes: su histograma y los centroides de colores, cuyos valores se almacenan en una base de datos.

3.4.1. Almacenamiento del histograma

La configuración almacenada se obtiene primero transformando la imagen patrón del modelo RGB al HSV utilizando (2.3), (2.2) y (2.1) en general. Para ésto se utilizará una función ya programada de *OpenCV*, con la cual se construye el histograma con la siguiente estructura: dado que los valores obtenidos en el canal H están en el rango $[0, 179]$ (debido a que la función de *OpenCV* utiliza un entero de 8 bits), el histograma se divide en 6 bloques, cada bloque tiene un tamaño de 30 unidades, el primer bloque tiene como rango $[15, 44]$, el segundo $[45, 74]$ y así sucesivamente hasta terminar con el sexto con un rango de $[165, 179]$ y $[0, 14]$, y cada pixel estará relacionado con un bloque si el valor de su matiz cae dentro del rango del bloque.

Cuando todos los píxeles están relacionados se realiza la contabilidad de cada bloque, cada bloque debe de tener 3 datos que son: (i) el conteo de cuantos píxeles están relacionados con el bloque, (ii) el promedio de los valores del canal S de los píxeles que están relacionados con el bloque, y (iii) el promedio de los valores del canal V de los píxeles que están relacionados con el bloque, en total son 18 valores por patrón que se guardan en la base de datos de la robot.

3.4.2. Centroides de colores

Los centroides de colores consisten en generar 8 centroides a partir de los valores de los píxeles en RGB, utilizando la cuantización vectorial (visto en la sección 2.3.2). Sin embargo se realizó una pequeña modificación al algoritmo de cuantización vectorial; al momento de realizar la división en lugar de separarlos por un umbral τ_{cv} , la separación se realiza con el promedio de los puntos que estén debajo y sobre el centroide original, para un punto y otro, para ello se debió realizar la clasificación cuando se selecciona el centroide más cercano.

La información que se almacena en la base de datos son los 3 valores de los 8 colores encontrados.

3.4.3. Reconocimiento de la camisa

Al final del entrenamiento se obtienen dos descriptores diferentes, cuando se obtiene la camisa tentativa se genera su histograma y se compara con los almacenados. En el Algoritmo 2 se visualiza el código que se siguió para la comparación de los histogramas donde $P_h[n_{img}][6][3]$ es una matriz que guarda la base de datos de los histogramas, n_{img} es el número del patrón a comparar, $T_h[6][3]$ es una matriz con la información de la camisa tentativa y *respuesta* es la variable que devuelve el resultado de la verificación.

Algorithm 2: Verificación del histograma

Entrada: Histograma del patrón $P_h[n_{img}][6][3]$ y de evaluación $T_h[6][3]$.

Salida: **cierto** si $T_h[6][3]$ es la camisa y **falso** en caso contrario.

```

1 congruencia ← 0
2 respuesta ← falso
3 para  $i \leftarrow 0$  hasta 5 hacer
4   si  $|T_h[i][0] - P_h[n_{img}][i][0]| < \tau_{h1}$  entonces
5     si  $|T_h[i][1] - P_h[n_{img}][i][1]| < \tau_{h2}$  entonces
6       si  $|T_h[i][2] - P_h[n_{img}][i][2]| < \tau_{h3}$  entonces
7          $congruencia \leftarrow congruencia + 1$ 
8 si  $congruencia = 6$  entonces
9    $respuesta \leftarrow$  cierto
10 retorna respuesta

```

Si el patrón a evaluar coincide bajo ciertos umbrales τ_{h1} , τ_{h2} y τ_{h3} (determinados durante el desarrollo de las pruebas), la camisa tentativa pasa a la verificación de los centroides de colores. Esta verificación se realiza en dos partes utilizando un centroe del patrón tentativo y un centroe del patrón guardado en la base de datos: (i) se coteja la relación de los valores entre los tres canales sea la misma en ambos patrones, es decir, se toma la diferencia entre el canal rojo y el azul, del canal azul y el verde, y del canal verde y el rojo, para ambos centroides y se compara con el umbral τ_{c2} , y (ii) se analiza la diferencia de cada canal a través de un umbral τ_{c1} . Si las coincidencias supera un número dado se acepta que es la camisa, si no se llega al límite se procede con el siguiente patrón, y así sucesivamente hasta terminar.

En el Algoritmo 3 se muestra el código de verificación de centroides donde $P_c[n_{img}][8][3]$ es una matriz que posee los valores de los centroides en la base de datos, n_{img} es el número del patrón a comparar, $T_c[6][3]$ es la información de los centroides de la camisa tentativa y *respuesta* es la variable que devuelve el resultado de la verificación.

Algorithm 3: Verificación de los centroides

Entrada: Grupo de centroides del patrón $P_c[n_{img}][8][3]$ y de evaluación $T_c[8][3]$.

Salida: **cierto** si $T_c[6][3]$ es la camisa y **falso** en caso contrario.

```

1 spun ← 0
2 ipun ← -1
3 respuesta ← falso
4 obt ← falso
5 rpun ← 1
6 para i ← 0 hasta 7 hacer
7   casilla[i] ← falso
8 para j ← 0 hasta 7 hacer
9   obt ← falso
10  para k ← 0 hasta 7 y obt = falso hacer
11    si casilla[k] = falso entonces
12      si  $|(P_c[n_{img}][j][0] - P_c[n_{img}][j][1]) - (T_c[k][0] - T_c[k][1])| < \tau_{c2}$  y
13         $|(P_c[n_{img}][j][0] - P_c[n_{img}][j][2]) - (T_c[k][0] - T_c[k][2])| < \tau_{c2}$  y
14           $|(P_c[n_{img}][j][2] - P_c[n_{img}][j][1]) - (T_c[k][2] - T_c[k][1])| < \tau_{c2}$ 
15            entonces
16              si  $|P_c[n_{img}][j][0] - T_c[k][0]| < \tau_{c1}$  y
17                 $|P_c[n_{img}][j][1] - T_c[k][1]| < \tau_{c1}$  y
18                   $|P_c[n_{img}][j][2] - T_c[k][2]| < \tau_{c1}$  entonces
19                    spun ← spun + 1
20                    casilla[k] ← cierto
21                    obt ← cierto
22      entonces
23        si spun > 5 entonces
24          respuesta ← cierto
25  retorna respuesta

```

3.5. Rastreo con filtro de Kalman

Parte fundamental en este proyecto es el rastreo del operador, para lo cual se utilizó el filtro de Kalman para estimar la posición en la que el operador pueda estar. Una ventaja que nos permite el filtro de Kalman es estimar una posición después de una oclusión utilizando las ecuaciones ilustradas en la Figura 3.12 como se puede ver en [WKT06].

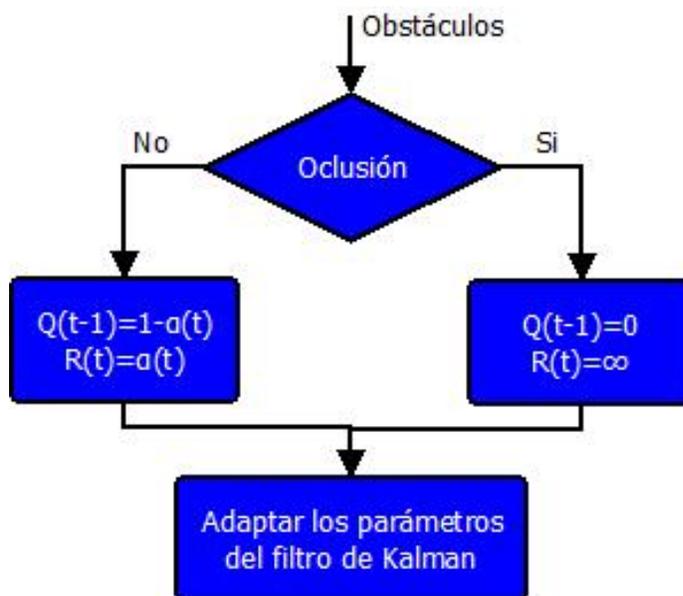


Figura 3.12: Ecuaciones del filtro de Kalman.

La función $a(t)$ es una variante que modifica las variables de Kalman y que depende del porcentaje de oclusión, sin embargo, se decidió no usarla y sustituirla por un 0 debido a que en ocasiones solo conocemos si esta ocluido o no, como cuando usamos el láser.

Con la implementación del diagrama de la Figura 3.12, el seguimiento se ve levemente afectado cuando hay oclusiones, siempre y cuando la dirección y la velocidad del operador sean casi constantes desde poco antes de la oclusión y durante ésta.

En este Capítulo se desarrollaron e implementaron los algoritmos para la detección, reconocimiento y seguimiento de un humano con base en sus características espaciales y visuales, ahora es necesario aplicarlo para ver alguna utilidad hacia la humanidad.

Capítulo 4

Aplicación en robots móviles

La aplicación de este trabajo en una robot móvil, consiste en el desplazamiento de ésta a lo largo del ambiente siguiendo al operador, dicho desplazamiento se puede ejecutar de dos formas: *(i)* la física o real que consta de una robot física con equipo *hardware* tales como láser, cámaras, motores, etcétera, y *(ii)* la virtual que consta de un simulador que tenga la capacidad de emular todo el equipo físico antes mencionado y también pueda agregar elementos como personas y lugares como si fuera un ambiente real, a demás de poder interactuar con los módulos de la robot real, para que estos mismos puedan controlar a ambos robots.

4.1. Robots Justina y Judy

Para el caso de este trabajo se contó con dos robots físicas llamadas Justina y Judy que pertenecen al Laboratorio de Biorrobótica, este Laboratorio cuenta con todos los instrumentos necesarios para el objetivo, y cada robot cuenta con un modelo por bloques interno que le permite interactuar con nuestro mundo y sus personas, este modelo también llamado VirBot está compuesto por varios subsistemas que interactúan entre si intercambiando información para lograr los objetivos impuestos a las robots. En el diagrama de la Figura 4.1 se presenta la composición del VirBot y las conexiones de sus subsistemas, este sistema está dividido en 4 áreas: *(i)* entradas, *(ii)* planificación, *(iii)* gestión del conocimiento y *(iv)* ejecución; y para más detalles se puede ver [SCH98].

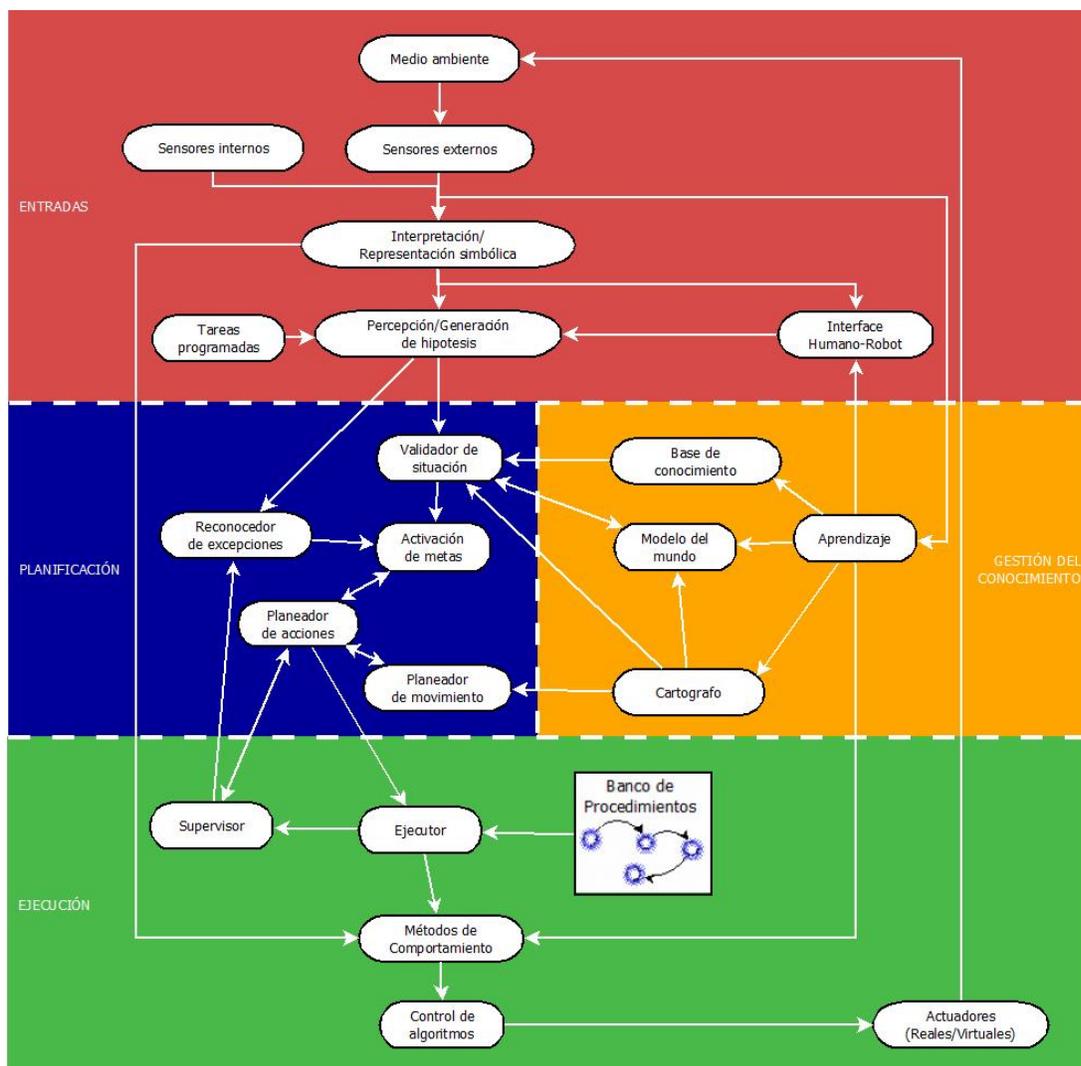


Figura 4.1: Diagrama VirBot presenta la composición del sistema VirBot y las conexiones de sus subsistemas, este sistema está dividido en 4 áreas: (i) entradas, (ii) planificación, (iii) gestión del conocimiento y (iv) ejecución, para más detalles se puede ver [SCH98].

A continuación se describe como están formadas las robots Justina y Judy. En cuestión de *Hardware* ambas poseen:

- Cabeza mecatrónica.
 - *KinectTM XBOX 360* con las siguientes características (ver [pla15] para más información):
 - ◇ Sensores:
 - Lentes de color y sensación de profundidad.
 - Microfono multi-arreglo.
 - Ajuste de sensor con su motor de inclinación.
 - ◇ Campo de visión:
 - Campo de visión horizontal: 0.99 rad.
 - Campo de visión vertical: 0.75 rad.
 - Rango de inclinación física: ± 0.47 rad.
 - Rango de profundidad del sensor: 1.2 – 3.5 metros.
 - Data Streams* (Flujo de datos).
 - 320 × 240 a 16 bits de profundidad a 30fps.
 - 640 × 480 32-bit de color a 30fps.
 - Audio de 16-bit a 16 kHz.
 - Cámara web de 1280 × 1024 píxeles a 30fps.
 - Micrófono de 16-bit a 16 kHz.
 - Cabeceo rango $\frac{\pi}{2}$ rad.
 - Paneo rango π rad.
- Torso:
 - *KinectTM XBOX 360* ya especificado (únicamente Justina).
 - Láser Hokuyo con las siguientes características (ver [CO.14] para más información):
 - ◇ Distancia de lectura es de 20 mm a 5,600 mm en 240°.
 - ◇ Tiempo de lectura 100 ms.
 - ◇ Resolución angular 0.36°.
 - ◇ Ruido de 25dB.
- Brazos con 7 grados de libertad.
- Base móvil de par diferencial con 2 baterías de litio de 6 y 12 V.

- Computadora Lenovo Y510P con:
 - Procesador Intel® Core™ i7-3630QM (2.40 GHz, 4MB L3, 1,333 MHz FSB).
 - Sistema operativo Windows 7 Professional 64 bits.
 - Pantalla de 15.6" HD+ (1600 × 900) (220 nits).
 - Soporte para gráficos duales: NVIDIA® Quadro® K2000M 2GB VRAM.
 - Memoria de 8 GB DDR3L – 1,333 Mhz.
 - Unidad de disco duro de hasta 1 TB.
 - Sonido con Dolby® Home Theater® v4.
 - Comunicaciones integradas:
 - ◊ ThinkPad b/g/n Wireless.
 - ◊ WWAN inalámbrica Qualcomm® Gobi™ 3K 14.4Mbps/HSPA + /EVDO HSPA + 21Mbps.
 - Conectores:
 - ◊ 1 Mini DisplayPort con audio.
 - ◊ 1 VGA.
 - ◊ 1 Combo de auriculares y micrófono de 3.5 mm.
 - ◊ 2 USB 3.0.
 - ◊ 1 USB 2.0.
 - ◊ 1 USB 2.0 con conexión eléctrica.
 - ◊ 1 Lector de tarjetas SD 4 en 1 (Ranura para SD/SDHC/SDXC/MMC).
 - ◊ 1 Express Card 34 mm.
 - ◊ 1 Lector Smart Card.
 - Cámara HD de 720p para ambientes con poca luz.
 - Batería de hasta 5 horas.
 - Peso de 2.7 kg (5.95 lb).
 - Dimensiones 245.1mm × 372.8mm × 31.8mm (9.65" × 14.68" × 1.25").

y en *Software* los módulos a utilizar son:

- Sistema de visión.
- Sistema de cabeza.
- Planeador de movimientos.

- Planeador de acciones.
- Sistema de láser.
- *Blackboard*.
- Sistema de voz.
- Sistema de brazos.

En la Figura 4.2 se muestran estos componentes y su ubicación sobre las robots, el inciso (a) es Justina y el inciso (b) es Judy. La robot Justina posee un *Kinect*TM más y tiene unos 40 cm más de altura que Judy.

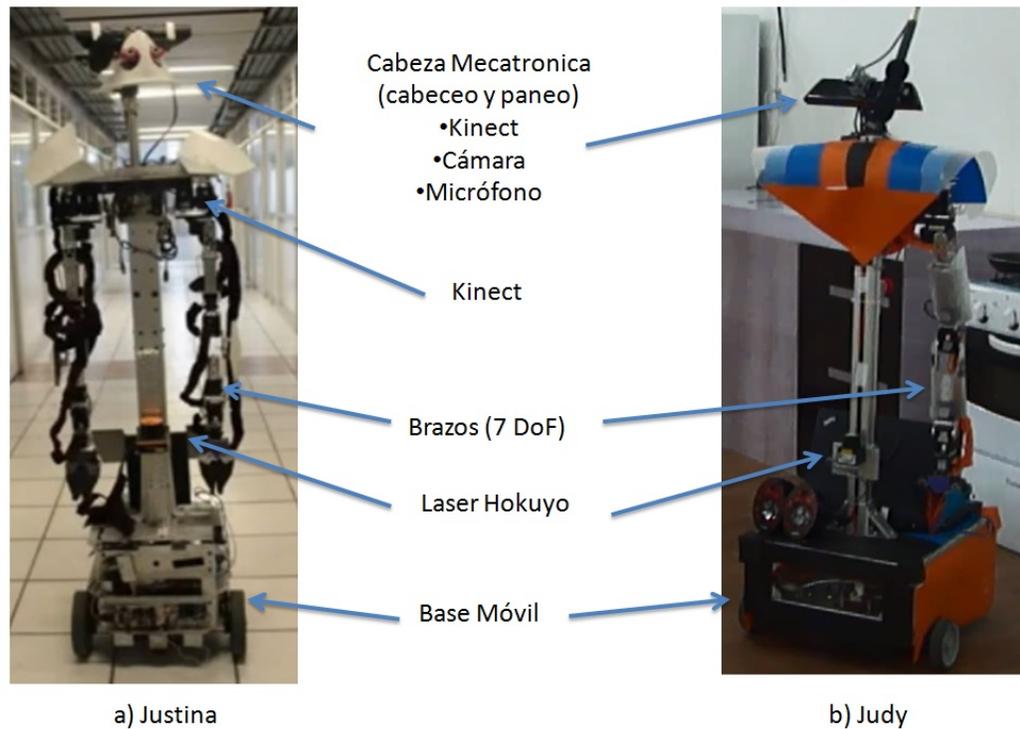


Figura 4.2: Imágenes de (a) Justina y (b) Judy.

4.2. Control de la Robot

Utilizando las ecuaciones de los campos potenciales para determinar la distancia y dirección que la robot debe de avanzar (visto en la sección 2.3.5), ahora debemos encontrar las ecuaciones que darán la velocidad adecuada al par de ruedas diferenciales que la robot posee, entonces debemos generar una fórmula para cada rueda, sin embargo, el control de la base ya se había realizado y se encuentra documentado en [Cru], por lo cual aquí solo se pondrán algunas fórmulas consideradas de importancia:

La fórmula de velocidad para la rueda derecha es

$$v_d(e_\theta) = \frac{v_{max}}{e^{\frac{e_\theta^2}{2\alpha^2}}} + \frac{l}{2}w_{max} \left(\frac{2}{1 + \frac{1}{e^{\frac{e_\theta}{\beta}}}} - 1 \right) \quad (4.1)$$

y la fórmula de velocidad para la rueda izquierda es

$$v_i(e_\theta) = \frac{v_{max}}{e^{\frac{e_\theta^2}{2\alpha^2}}} - \frac{l}{2}w_{max} \left(\frac{2}{1 + \frac{1}{e^{\frac{e_\theta}{\beta}}}} - 1 \right), \quad (4.2)$$

donde e_θ es el ángulo entre la orientación de la robot y el vector de posición de la posición deseada, v_{max} es la velocidad de traslación máxima, α es el parámetro de la velocidad de traslación, w_{max} es el ángulo de la velocidad angular, β es el ángulo de cambio de la velocidad máxima y mínima.

4.3. Sensado con Láser

El sensado con el láser se debe hacer lo más rápido posible ya que la única información con la que se cuenta son las coordenadas del lugar donde se encontraba el operador en la anterior lectura (posición anterior), sin embargo, esto no es una gran problemática dado que el procesamiento de la información del láser es veloz. Utilizando la ultima posición registrada del operador y el Algoritmo 1 (visto en la sección 3.2) es posible determinar quién es el operador en la nueva lectura hecha por el láser, calculando la distancia Euclidiana de la última posición y de la posible nueva posición, si dicha distancia es menor a 0.25 metros (valor obtenido de las diferentes pruebas durante el desarrollo) se asume que es la persona, si existen más de dos personas que puedan ser el operador se opta por la que esté más cercana a la posición anterior, dado que el sensado del entorno es constante en periodos cortos de tiempo y que la velocidad promedio de las

personas al caminar es de 1.4 m/s no causa mayores inconvenientes.

A continuación en la Figura 4.3 se muestra el diagrama de verificación de piernas donde se ilustra de forma técnica los pasos a seguir en la verificación de piernas.



Figura 4.3: Diagrama de verificación de piernas.

En la Figura 4.4 se muestra una serie de imágenes tomadas con la robot Justina, estas imágenes visualizan la información capturada por las lecturas del láser mientras una persona camina de derecha a izquierda y de regreso, existiendo algunos objetos a su alrededor pero sin generar oclusión. Sobre las imágenes están marcados con un rectángulo rojo los cúmulos de puntos de las piernas detectadas.

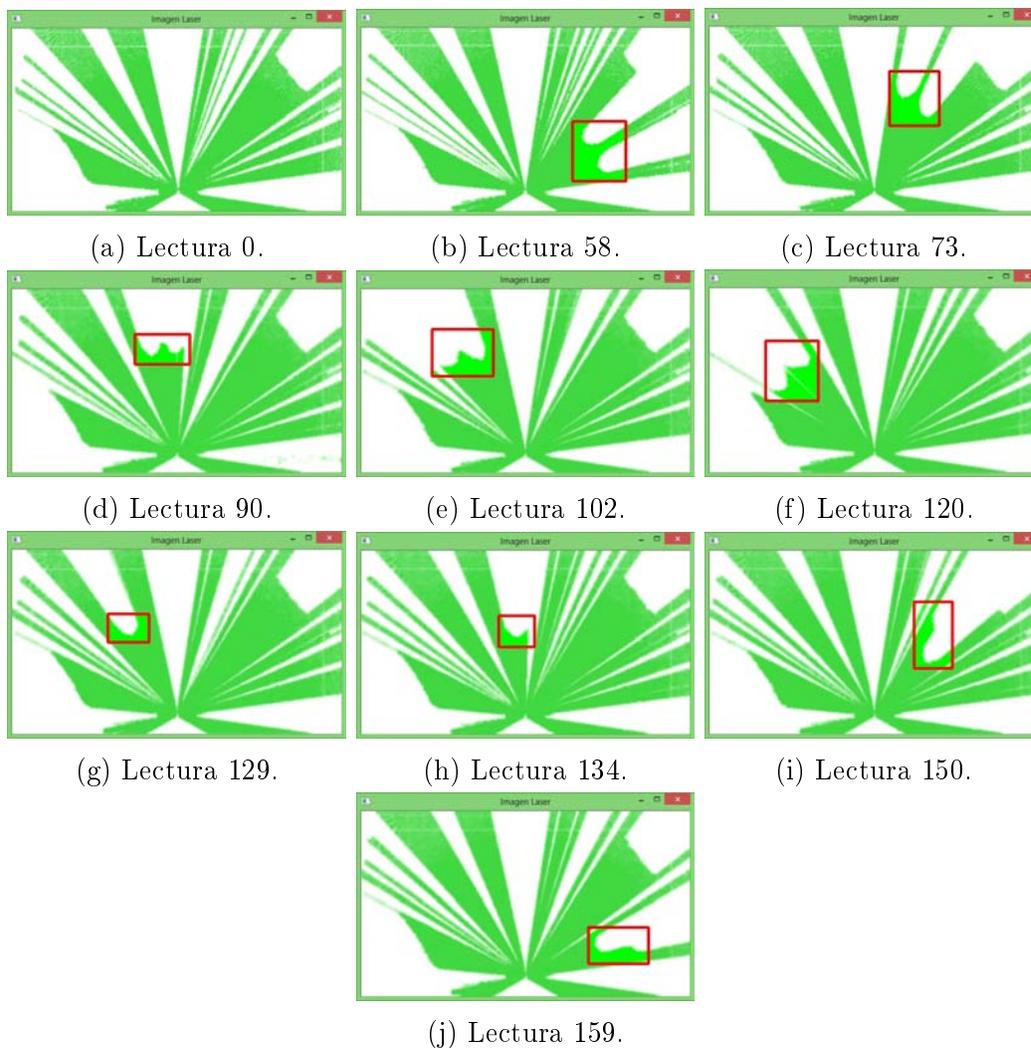


Figura 4.4: Muestra una serie de imágenes tomadas con la robot Justina, visualizando la información capturada por las lecturas del láser mientras una persona camina de derecha a izquierda y de regreso, existiendo algunos objetos a su alrededor pero sin generar oclusión. Sobre las imágenes están marcados con un rectángulo rojo los cúmulos de puntos de las piernas detectadas.

4.4. Sensado con Visión

En el sensado con visión, la carga de procesamiento se vuelve mayor, sin embargo, aquí existe un factor que nos permite confirmar la identidad del operador, hablamos de la camisa. Al utilizar la camisa es posible distinguir al operador de los otros humanos para extraer su coordenada y poder seguirlo.

Sin embargo, primero necesitamos entrenar la camisa del operador, para ello es necesario esperarlo si es que no se encuentra enfrente de la robot. En la Figura 4.5 se muestra un diagrama que describe paso a paso el método general utilizado para entrenar al operador.

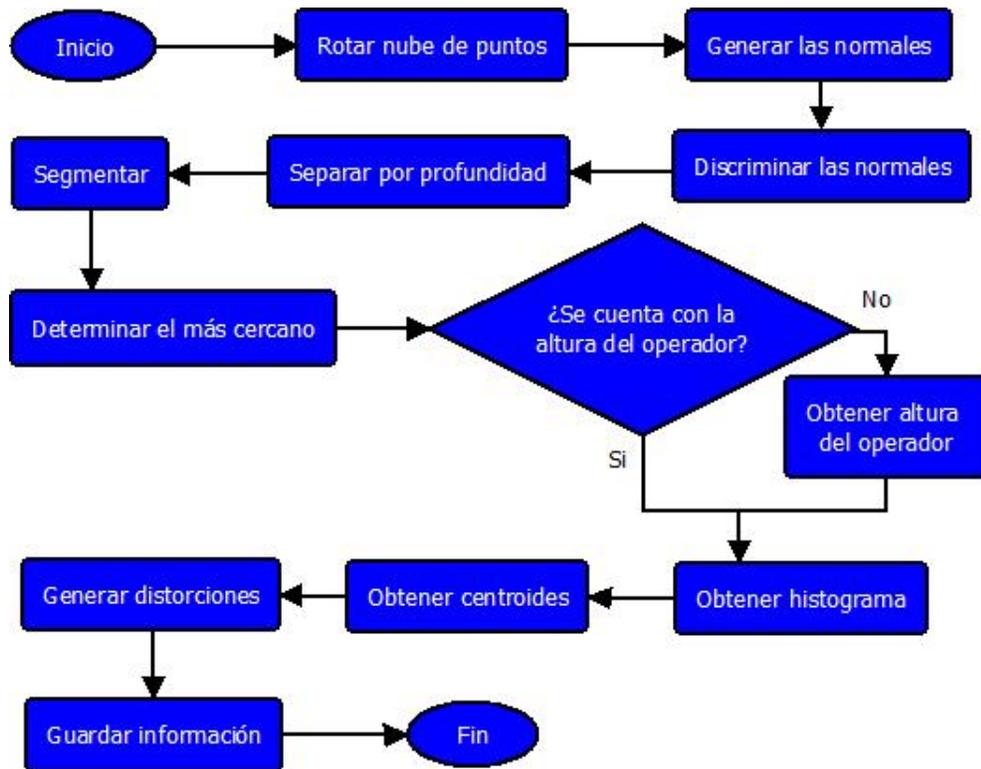


Figura 4.5: Diagrama de entrenamiento del operador, describe paso a paso el método general utilizado para entrenar al operador.

Una vez terminado el entrenamiento se comienza el seguimiento utilizando los Algoritmos 2 y 3 (vistos en la sección 3.4.3); la visión, al igual que el láser, es utilizada frecuentemente por la robot y aunque ambos métodos se deben mezclar al mismo tiempo para obtener un mejor resultado (ver la sección 5.3), cada método puede trabajar de forma separada. En la Figura 4.6 se visualiza el diagrama de comparación de camisas donde se describe paso a paso el método general utilizado.

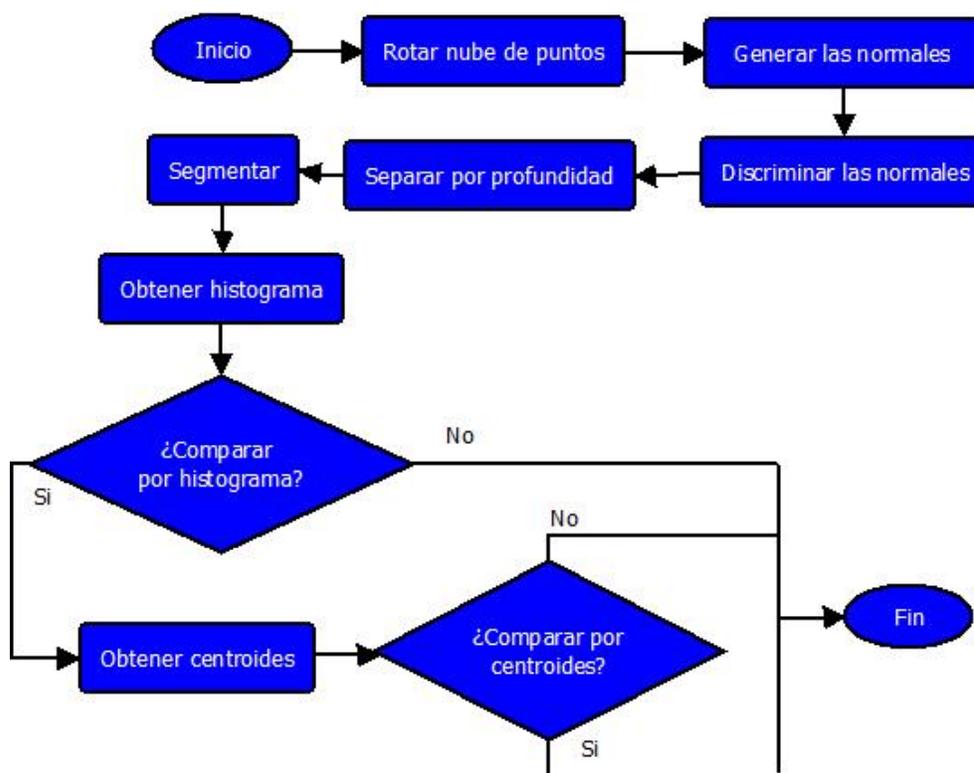


Figura 4.6: Diagrama de comparación de camisas, donde se describe paso a paso el método general utilizado.

En la Figura 4.7 se muestran una serie de imágenes tomadas con la robot Justina, estas imágenes tienen marcados los humanos detectados con rectángulos, si el rectángulo es de color verde significa que el humano es reconocido como el operador, de lo contrario, el rectángulo estará de color rojo, las imágenes corresponden a una prueba de seguimiento a través de un pasillo con varias personas alrededor, y algunas haciendo oclusiones.

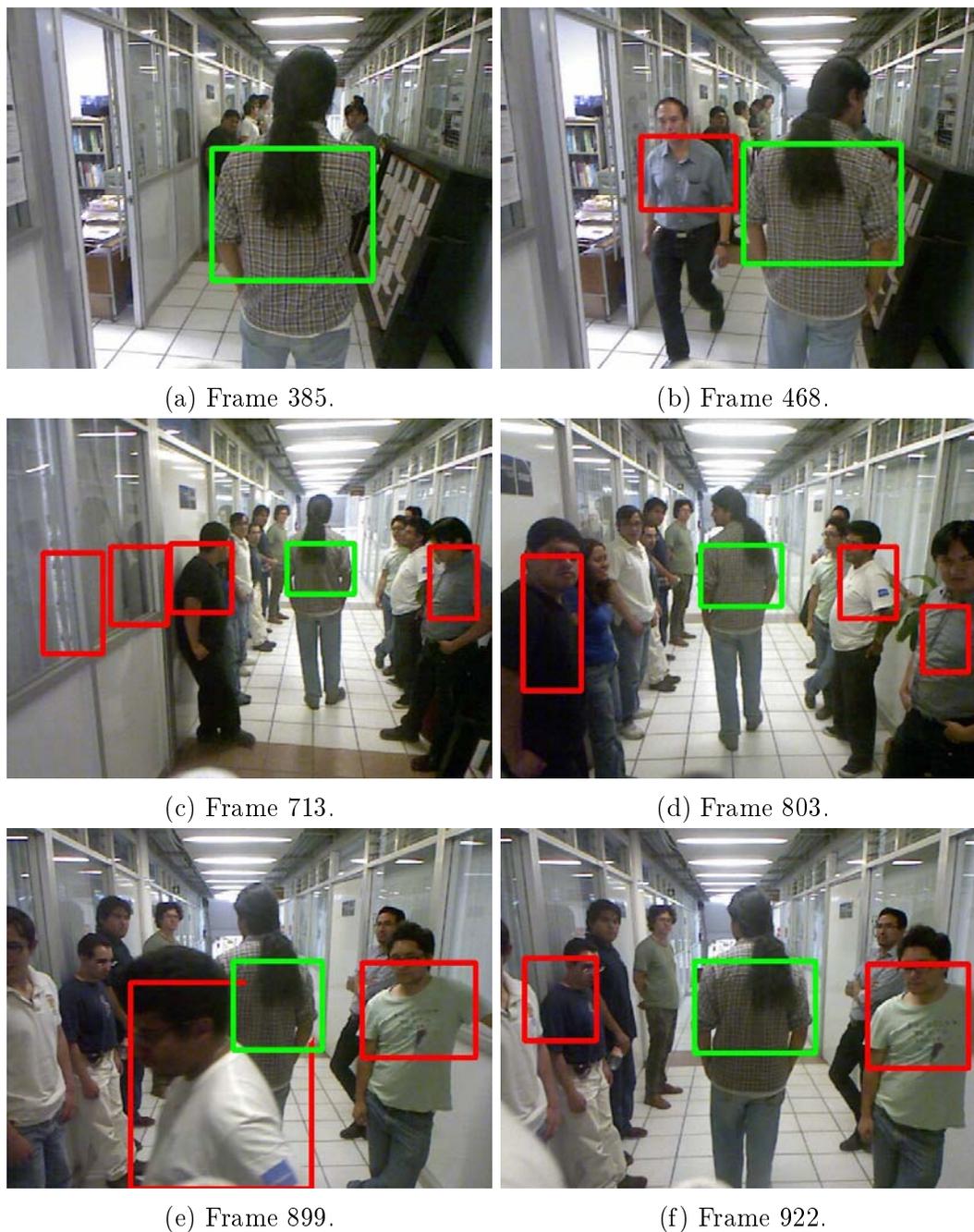


Figura 4.7: Muestra una serie de imágenes tomadas con la robot Justina, estas imágenes tienen marcados los humanos detectados con rectángulos, si el rectángulo es de color verde significa que el humano es reconocido como el operador, de lo contrario, el rectángulo estará de color rojo, las imágenes corresponden a una prueba de seguimiento a través de un pasillo con varias personas alrededor, y algunas haciendo oclusiones.

4.5. Rastreando al operador

El escenario general de operación consiste en lo siguiente: (i) un operador (humano a seguir), (ii) humanos extras bloqueando parcial o totalmente el camino entre el operador y la robot, y (iii) objetos materiales. Así que la robot empieza en una locación y el operador enfrente de ella, luego de que la robot realiza el procedimiento de entrenamiento al operador (ver Figura 4.5), el operador avanza y la robot lo debe de seguir, mientras ésto sucede habrá personas que interferirán entre ellos.

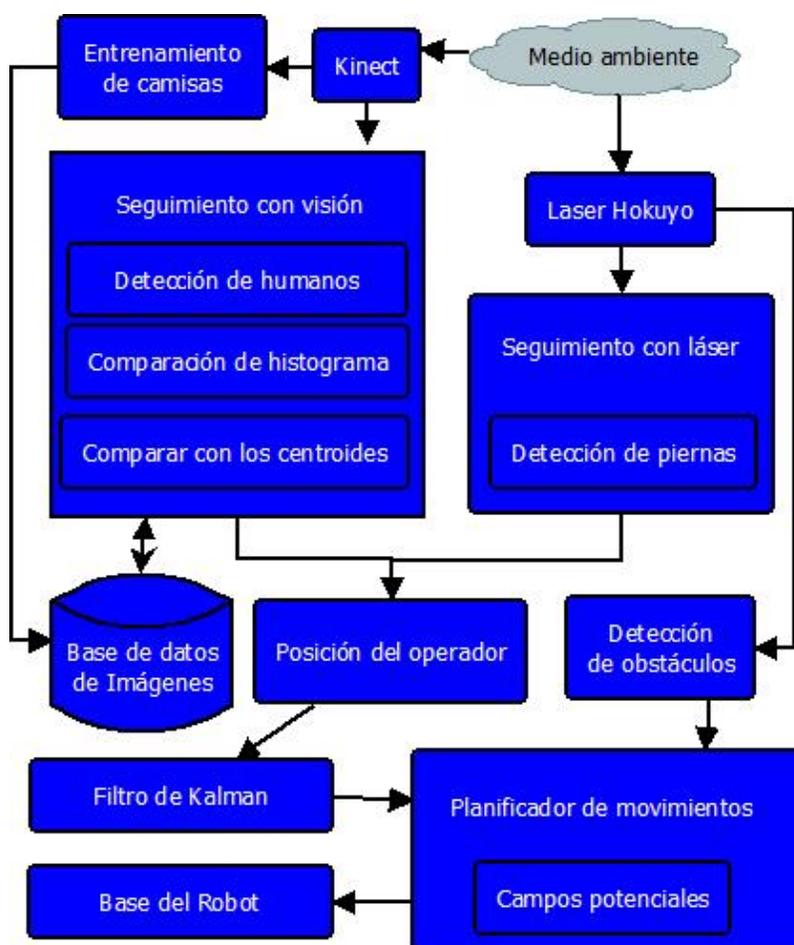


Figura 4.8: Diagrama interno de la metodología propuesta para el seguimiento de personas.

En Figura 4.8 se muestra el diagrama interno de la metodología propuesta para el seguimiento de personas, esta metodología es el proceso del flujo de co-

municación de las secciones vistas en el Capítulo 3, desde que la información del medio ambiente es recolectada por los sensores, hasta el movimiento de la base móvil para encaminar al robot; entre ellas destaca la participación de los Algoritmos 2 y 3 del seguimiento con visión, el Algoritmo 1 del seguimiento con el láser, el filtro de Kalman y los campos potenciales artificiales para lograr el objetivo.

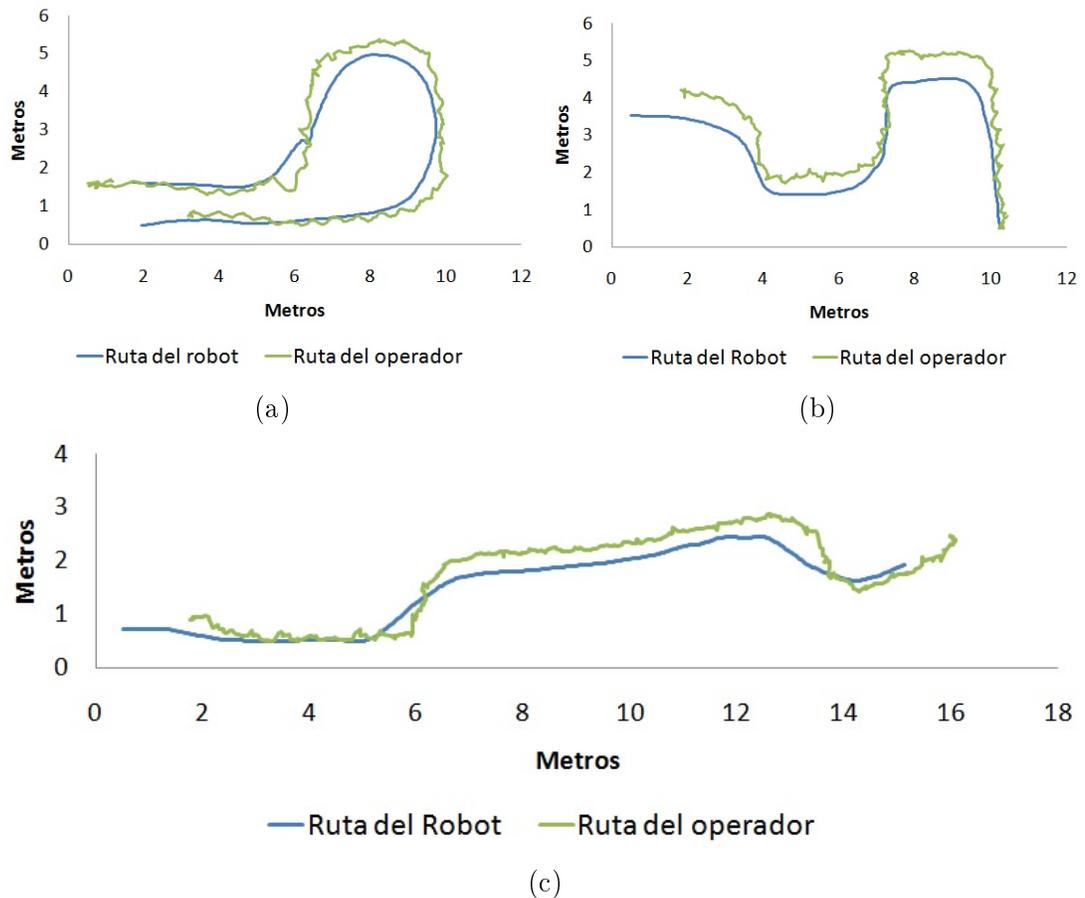


Figura 4.9: Muestra una serie de imágenes de rutas hechas por la robot Judy y el operador, en experimentos realizados y documentados en el Capítulo 5, en (a) es el experimento 1, en (b) es el experimento 3, y en (c) es el experimento 2. Los recorridos constan de 2 líneas: la línea verde es la ruta del operador y la línea azul es la ruta seguida por la robot

En las Figuras 4.9a, 4.9b y 4.9c se muestran tres diferentes rutas hechas por el operador y la robot, estos recorridos son con base a los mejores resultados de los experimentos 1, 3 y 2 respectivamente, realizados y documentados en el Capítulo 5, los recorridos constan de 2 líneas: la línea verde es la ruta del operador

y la línea azul es la ruta seguida por la robot, aunque la línea verde es variante respecto a su trayectoria, el operador en ningún momento se movió en zigzag pero la línea quedó así debido a los errores de los motores y sensores de la robot, pero con ayuda del filtro de Kalman la robot puede continuar con la tarea. La información de las líneas fue obtenida con los sensores de la robot debido a que no se tuvieron los recursos para capturar las coordenadas reales de los experimentos.

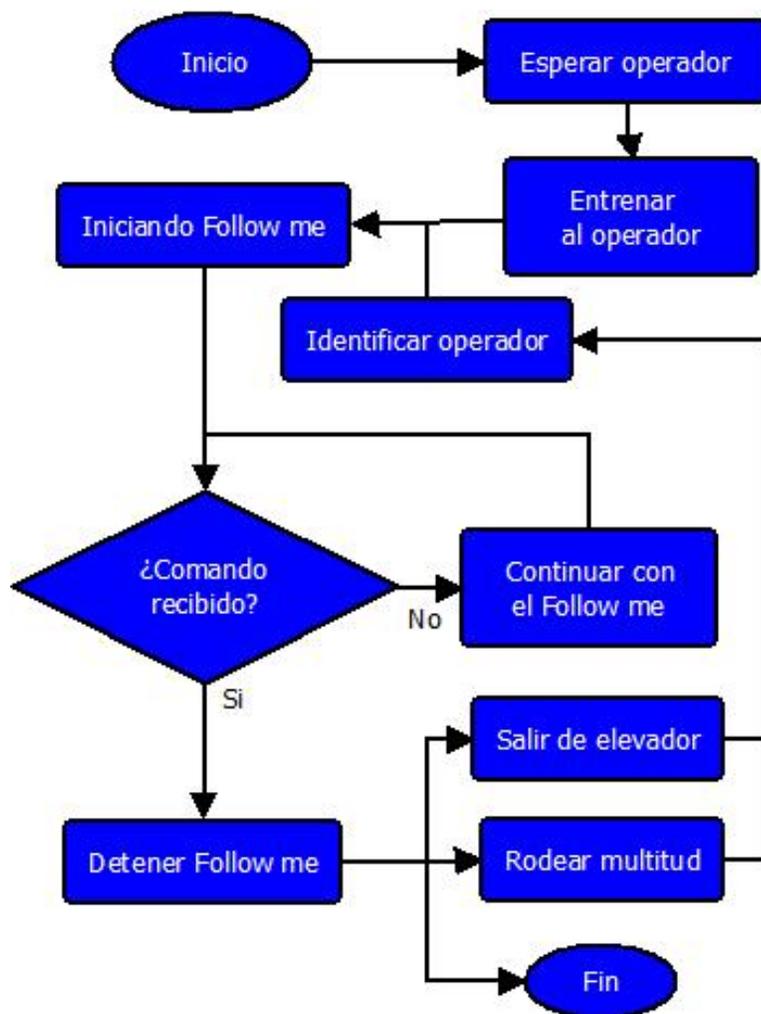


Figura 4.10: Diagrama de la máquina de estados, muestra el diagrama de la máquina de estados que se utiliza para el seguimiento del operador, donde cada cuadro representa un estado que la robot debe de ejecutar.

En la Figura 4.10 se muestra el diagrama de la máquina de estados que se utiliza para el seguimiento del operador, donde cada cuadro representa un estado que la robot debe de ejecutar, y es implementado dentro del planeador de acciones para su ejecución. Los bloques Iniciando *Follow me*, Continuar con el *Follow me* e Identificar operador son la representación del diagrama de la Figura 4.8 y el bloque de Entrenar al operador hace referencia al diagrama de la Figura 4.5.

Este proyecto puede tener aplicaciones en el ambito de asistencia a adultos mayores al acompañarlos a surtir su despensa en el supermercado por ejemplo, al seguirlos mientras ellos indican al robot que debe de tomar de los estantes para meterlo en el carrito, así también en sistemas hospitalarios ayudando a transportar a los pacientes de una habitación a otra mientras sigue al doctor/enfermera.

Capítulo 5

Pruebas y análisis de resultados

5.1. Procedimiento

Con apoyo de un grupo de voluntarios del Laboratorio de Biorobótica y del público en general, utilizando la información de los Capítulos 2, 3 y 4 anteriormente descritos como marco teórico, una robot llamada Justina y otra llamada Judy pertenecientes al Laboratorio, se realizaron varios experimentos físicamente ya que no se pudo utilizar un simulador por la estructura del proyecto.

Debido a la complejidad del trámite que se necesita hacer para sacar a las robots del edificio donde se encuentran y a la difícil traslación de éstas, se decidió realizar los experimentos utilizando 3 escenarios que se encuentran ubicados dentro del edificio donde están las robots, uno para cada experimento. Cada experimento se repitió 10 veces utilizando las dos formas diferentes de sensado y su combinación, haciendo un total de 90 ejecuciones. Como no se tuvieron los medios para obtener las posiciones y orientaciones reales de la robot y del operador, se recolectó la información obtenida por los sensores de la robot durante cada ejecución y fue dividida en 2 partes: (i) la información sobre la distancia (d_{ins}) que existe entre el operador y la robot medida en metros, y (ii) la información sobre el ángulo (o_{ins}) que existe entre el operador y la orientación de la robot medida en radianes.

Para poder calificar los desempeños de la robot, fue necesario crear un nuevo grupo de información a partir de los datos obtenidos por los sensores de las robots utilizando

$$d_{pro_s} = \frac{1}{21} \sum_{i=s-10}^{s+10} \left(\frac{1}{21} \sum_{i=s-10}^{s+10} d_{ins_i} \right) \quad (5.1)$$

y

$$o_{pro_s} = \frac{1}{21} \sum_{i=s-10}^{s+10} \left(\frac{1}{21} \sum_{i=s-10}^{s+10} o_{ins_i} \right), \quad (5.2)$$

donde el subíndice s es la posición del valor a procesar, d_{pro} es el valor de los promedios de las distancias y o_{pro} es el valor de los promedios de los ángulos.

La información sobre las N_{exp} posiciones de la robot y el operador recolectadas por los sensores, fue organizada en una tabla con la siguiente estructura: $(i) \#$ es el número de ejecución del experimento, $(ii) d_{dp}$ es el promedio de d_{ins} expresada con

$$d_{dp} = \frac{1}{N_{exp}} \sum_{i=1}^{N_{exp}} d_{ins_i}, \quad (5.3)$$

$(iii) d_{dm}$ es la diferencia máxima entre d_{pro} y d_{ins} expresada con

$$d_{dm_i} = \max(|d_{pro_i} - d_{ins_i}|) | i = 1, \dots, N_{exp}, \quad (5.4)$$

$(iv) d_{dd}$ es el promedio la diferencia entre d_{pro} y d_{ins} expresada con

$$d_{dd} = \frac{1}{N_{exp}} \sum_{i=1}^{N_{exp}} |d_{pro_i} - d_{ins_i}|, \quad (5.5)$$

$(v) d_{dl}$ es la cantidad de veces que la $d_{ins} \geq 2$ (que podría ser negativo para el experimento), si $d_{ins} < 2$, entonces no se contabiliza, $(vi) o_{op}$ es el promedio de o_{ins} expresado con

$$o_{op} = \frac{1}{N_{exp}} \sum_{i=1}^{N_{exp}} |o_{ins_i}|, \quad (5.6)$$

$(vii) o_{om}$ es la diferencia máxima entre o_{pro} y o_{ins} expresado con

$$o_{om} = \max(|o_{pro_i} - o_{ins_i}|) | i = 1, \dots, N_{exp}, \quad (5.7)$$

$(viii) o_{od}$ es el promedio de la diferencia entre o_{pro} y o_{ins} expresado con

$$o_{od} = \frac{1}{N_{exp}} \sum_{i=1}^{N_{exp}} |o_{pro_i} - o_{ins_i}|, \quad (5.8)$$

(ix) o_{ol} es la cantidad de veces que $|o_{ins}| > 0.3491$ (que podría ser negativo para el experimento), si $|o_{ins}| < 0.3491$, entonces no se contabiliza, (x) $Perd$ indica si la robot se perdió durante la ejecución del experimento, (xi) $Conf$ indica si la robot se confundió durante la ejecución del experimento y (xii) $Calif$ indica una calificación que se le da a la ejecución del experimento a partir de la información antes mencionada si la robot se perdió o se confundió la calificación que recibe es de 5, de lo contrario se utiliza

$$Calif = 10 - \left(f_2(d_{dm}) + f_3(d_{dd}) + f_1(d_{dl}) + f_2(o_{op}) + f_2(o_{om}) + f_3(o_{od}) + f_1(o_{ol}) \right), \quad (5.9)$$

donde

$$f_1(v_e) = \frac{1}{1 + \exp(-0.2(v_e - \bar{v}_e))} - 0.2, \quad (5.10)$$

$$f_2(v_e) = \frac{1}{1 + \exp(-200(v_e - \bar{v}_e))} - 0.2, \quad (5.11)$$

$$f_3(v_e) = \frac{1}{1 + \exp(-2000(v_e - \bar{v}_e))} - 0.2, \quad (5.12)$$

y \bar{v}_e es el valor promedio de v_e , los valores promedio se muestran en la Tabla 5.1.

	\bar{d}_{dm}	\bar{d}_{dd}	\bar{d}_{dl}	\bar{o}_{op}	\bar{o}_{om}	\bar{o}_{od}	\bar{o}_{ol}
Promedio	0.2613	0.034	108	-0.0419	0.2741	0.0496	28.3904

Tabla 5.1: Valores promedio para determinar el desempeño de la robot en los experimentos.

Casi todos los promedios fueron obtenidos tomando en cuenta los resultados de todas las ejecuciones, excepto los siguientes valores: (i) \bar{d}_{dl} es el promedio de los valores en el campo d_{dl} del segundo experimento solo usando el *KinectTM*, (ii) \bar{o}_{ol} es el promedio de los valores del campo o_{ol} en el tercer experimento usando el láser y el láser+*KinectTM* (debido a las grandes diferencias).

En (5.10), (5.11) y (5.12) se sugirió una función sigmoide dado que ésta puede castigar o premiar significativamente a $Calif$, dependiendo del valor que se le pase.

También se graficó la información generada en la mejor prueba (de acuerdo a las calificaciones obtenidas), por cada prueba se visualizan 2 gráficas, la primera es el comportamiento de d_{ins} y d_{pro} , y la segunda es el comportamiento de o_{ins} y o_{pro} , en el diagrama de la Figura 5.1 se muestra su distribución.

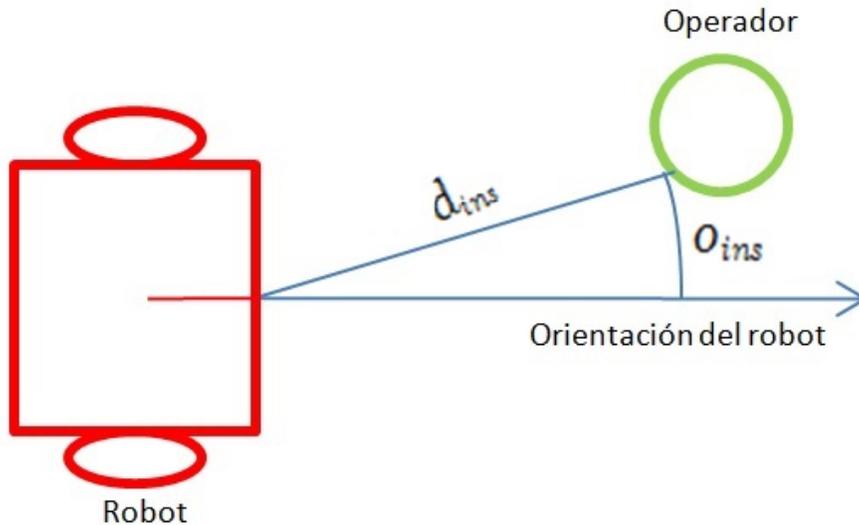


Figura 5.1: Imágen que muestra la distancia y ángulo instantáneo entre la robot y el operador.

5.2. Resultados

5.2.1. Experimento 1 - Cuarto vacío

Se realizó el seguimiento del operador con la robot en un cuarto de 4×4 metros y un pedazo de pasillo de 2×2 metros completamente vacíos. En la Figura 5.2 se muestra el recorrido del circuito que consiste de salir del lado derecho de la robot, rodear el cuarto, y llegar por la parte izquierda donde las áreas de color rosa representan obstáculos u objetos del ambiente, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.

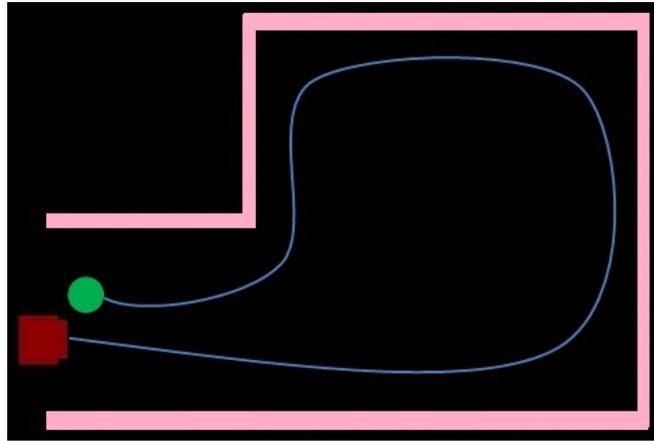


Figura 5.2: Esquema de la distribución territorial del experimento 1, las áreas de color rosa representan obstáculos u objetos del ambiente, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.

En la Figura 5.3 se muestran fotografías del lugar donde fue realizado el experimento donde (a) es la vista desde el pasillo y (b) es la vista desde una esquina del cuarto.



(a)

(b)

Figura 5.3: Imágenes del lugar donde se probó el experimento 1 donde (a) es la vista desde el pasillo y (b) es la vista desde una esquina del cuarto.

En la Tabla 5.2, 5.3 y 5.4 se tabulan los resultados obtenidos con la robot Judy utilizando los 3 métodos de seguimiento.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	<i>Perd</i>	<i>Conf</i>	<i>Calif</i>
1	1.3544	0.2416	0.046	20	-0.034	0.1672	0.0375	0	No	No	8.5722
2	1.3451	0.2827	0.0398	16	-0.0368	0.2106	0.0432	2	No	No	7.8799
3	1.1457	0.219	0.0439	0	-0.0715	0.2313	0.0425	4	No	No	9.2
4	1.3696	0.2388	0.0403	0	-0.0596	0.1869	0.0528	0	No	No	8.4015
5	1.3561	0.2308	0.0438	0	-0.0748	0.243	0.055	1	No	No	8.4
6	1.25	0.2531	0.0285	0	-0.0546	0.2753	0.0396	13	No	No	9.6425
7	1.2947	0.1954	0.0244	0	-0.0584	0.1687	0.056	1	No	No	9.2
8	1.2739	0.2566	0.0321	0	-0.0514	0.235	0.0391	1	No	No	9.9204
9	1.1087	0.2043	0.0376	0	-0.0654	0.2046	0.0562	10	No	No	8.4008
10	1.2545	0.1469	0.03	0	-0.056	0.2223	0.0408	4	No	No	10

Tabla 5.2: Resultados obtenidos con láser en el experimento 1.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	<i>Perd</i>	<i>Conf</i>	<i>Calif</i>
1	1.7184	0.6001	0.0439	81	-0.0450	0.1946	0.044	0	No	No	8.2515
2	1.7293	0.4499	0.0374	32	-0.0631	0.315	0.0373	0	No	No	7.6013
3	1.8072	0.5548	0.0479	98	-0.048	0.1892	0.0416	0	No	No	8.3707
4	1.9519	0.4979	0.0487	230	-0.0538	0.249	0.0424	1	No	No	7.6
5	1.9746	0.1447	0.0345	228	-0.0527	0.1888	0.0263	0	No	No	8.6678
6	2.0704	0.2717	0.0356	311	-0.0485	0.1933	0.0298	0	No	No	7.7377
7	2.0896	0.1423	0.0357	317	-0.0568	0.1664	0.0298	0	No	No	8.4326
8	1.9307	0.1415	0.0372	222	-0.0486	0.1678	0.034	0	No	No	8.3949
9	1.7506	0.1531	0.0315	125	-0.046	0.1468	0.0337	0	No	No	9.1279
10	0	0	0	0	0	0	0	0	Si	No	5

Tabla 5.3: Resultados obtenidos con *KinectTM* en el experimento 1.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	<i>Perd</i>	<i>Conf</i>	<i>Calif</i>
1	1.3529	0.2062	0.0343	0	-0.0613	0.2306	0.05	0	No	No	9.0837
2	1.1	0.121	0.0229	0	-0.07	0.2297	0.0585	14	No	No	9.2
3	1.2323	0.1173	0.0261	0	-0.0733	0.2003	0.0565	6	No	No	9.2
4	1.2111	0.1755	0.0252	0	-0.0756	0.2763	0.0512	12	No	No	8.8244
5	1.2545	0.16	0.0244	0	-0.068	0.1776	0.0362	0	No	No	10
6	1.1931	0.1732	0.0309	0	-0.0406	0.2244	0.0451	7	No	No	9.6354
7	1.1852	0.1232	0.0384	0	-0.0652	0.1801	0.0456	2	No	No	9.2001
8	1.1497	0.261	0.0258	0	-0.041	0.2675	0.0551	8	No	No	8.5627
9	1.1281	0.2139	0.0396	0	-0.0485	0.177	0.0492	3	No	No	9.0733
10	1.179	0.1339	0.0263	0	-0.0678	0.2247	0.0546	1	No	No	9.2

Tabla 5.4: Resultados obtenidos con láser+*KinectTM* en el experimento 1.

En las Figuras 5.4, 5.5 y 5.6 se graficó la información de distancia obtenidas con la robot Judy utilizando los 3 métodos de seguimiento.

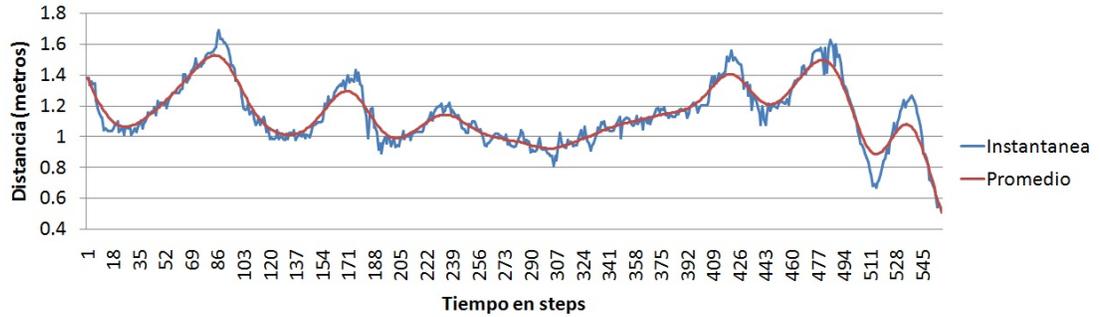


Figura 5.4: Gráfica de distancias obtenidas con láser en el experimento 1.



Figura 5.5: Gráfica de distancias obtenidas con *Kinect™* en el experimento 1.

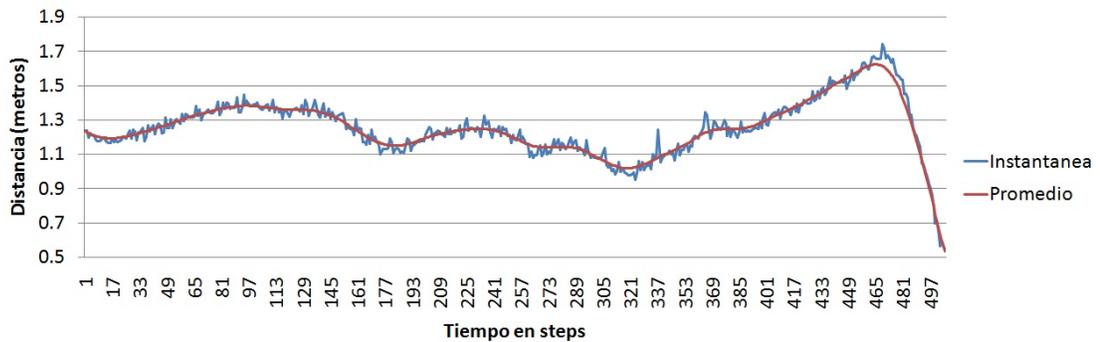


Figura 5.6: Gráfica de distancia obtenidas con láser+ *Kinect™* en el experimento 1.

En las Figuras 5.7, 5.8 y 5.9 se graficó la información de orientación obtenidas con la robot Judy utilizando los 3 métodos de seguimiento.

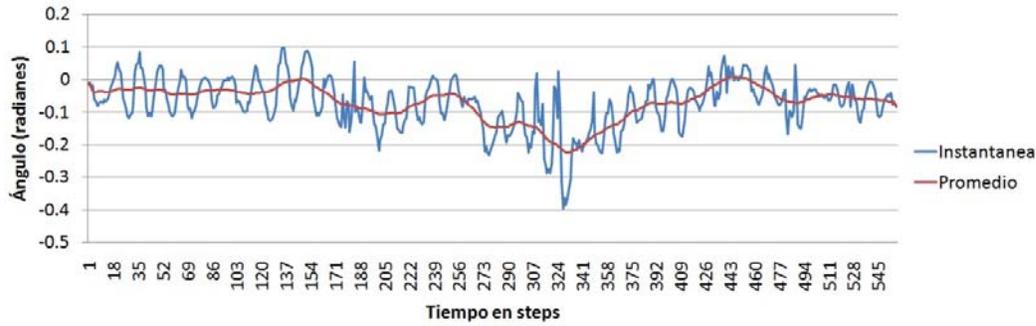


Figura 5.7: Gráfica de orientaciones obtenidas con láser en el experimento 1.

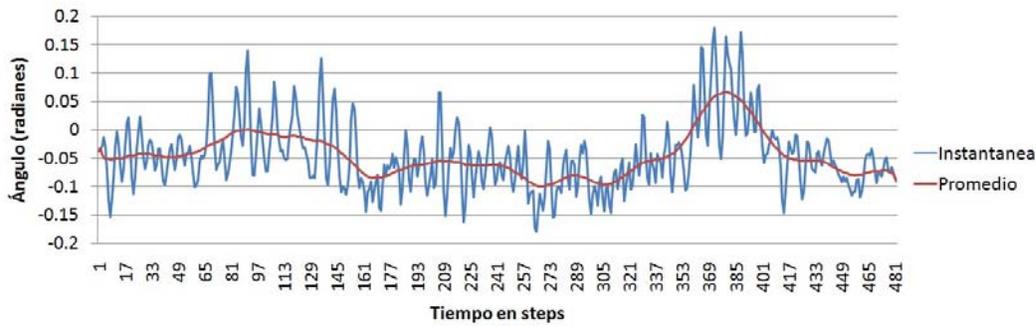


Figura 5.8: Gráfica de orientaciones obtenidas con *Kinect™* en el experimento 1.

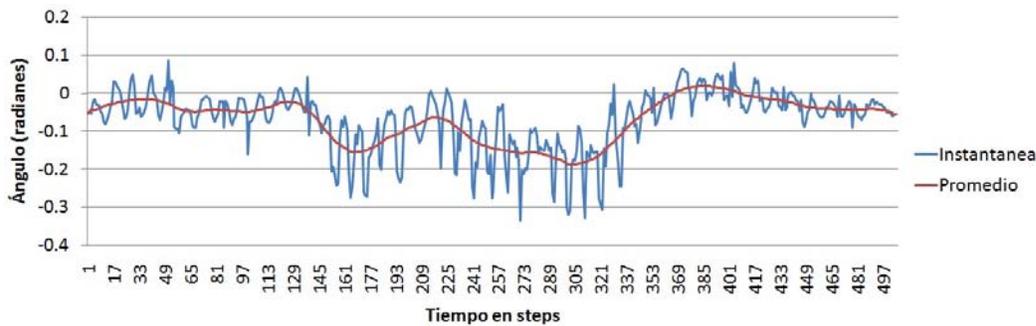


Figura 5.9: Gráfica de orientaciones obtenidas con láser+*Kinect™* en el experimento 1.

5.2.2. Experimento 2 - Pasillo largo con humanos

En este experimento se incrementó el grado de dificultad tanto de distancia como en personas involucradas, fue realizado en un corredor de 2.3×18 metros. En la Figura 5.10 se muestra el recorrido del circuito que consiste de avanzar por todo el largo pasillo empezando del lado izquierdo y terminando de lado izquierdo de la Figura, evitando a 7 personas distribuidas a todo lo largo; donde una persona realiza una oclusión completa, y la robot tiene que pasar de forma cercana a las personas y hacer giros un poco bruscos. Las áreas de color rosa representan obstáculos u objetos del ambiente, los círculos amarillos representan personas, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.



Figura 5.10: Esquema de la distribución territorial del experimento 2, las áreas de color rosa representan obstáculos u objetos del ambiente, los círculos amarillos representan personas, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.

En la Figura 5.11 se muestran unas fotografías del lugar donde fue realizado el experimento donde (a) es la vista desde donde se inicia el recorrido y (b) es la vista desde donde se termina el recorrido.

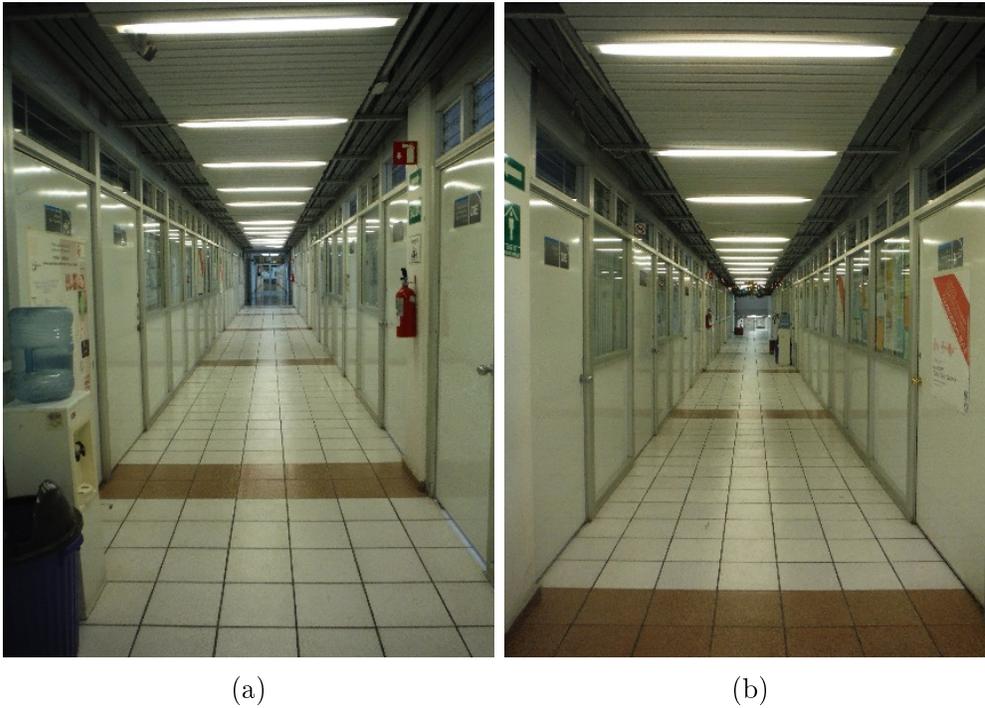


Figura 5.11: Imágenes del lugar donde se probó el experimento 2 donde (a) es la vista desde donde se inicia el recorrido y (b) es la vista desde donde se termina el recorrido.

En la Tabla 5.5, 5.6 y 5.7 se tabulan los resultados obtenidos con la robot Judy utilizando los 3 métodos de seguimiento.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	<i>Perd</i>	<i>Conf</i>	<i>Calif</i>
1	1.2349	0.2712	0.0245	0	-0.0697	0.2538	0.0483	8	No	No	9.3216
2	1.2889	0.1421	0.0323	0	-0.0646	0.2519	0.0521	8	No	No	9.2065
3	1.3976	0.1899	0.0348	0	-0.0816	0.2242	0.0474	4	No	No	9.3765
4	0	0	0	0	0	0	0	0	No	Si	5
5	1.3256	0.1887	0.0246	0	-0.0725	0.2195	0.0506	0	No	No	9.3119
6	1.242	0.1346	0.0352	0	-0.075	0.252	0.0438	4	No	No	9.2896
7	0	0	0	0	0	0	0	0	No	Si	5
8	1.2777	0.1255	0.034	0	-0.054	0.212	0.0425	5	No	No	9.7053
9	0	0	0	0	0	0	0	0	No	Si	5
10	0	0	0	0	0	0	0	0	No	Si	5

Tabla 5.5: Resultados obtenidos con láser en el experimento 2.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	Perd	Conf	Calif
1	1.8307	0.268	0.0395	47	-0.0457	0.3311	0.0517	2	No	No	6.9051
2	1.7917	0.2989	0.0374	4	-0.047	0.4151	0.0569	8	No	No	6.7351
3	1.7901	0.4286	0.0434	50	-0.0374	0.3981	0.0533	8	No	No	6.2884
4	1.8337	0.6661	0.0685	86	-0.0186	0.4068	0.0701	15	No	No	6.0094
5	0	0	0	0	0	0	0	0	Si	No	5
6	1.9635	0.5139	0.0538	202	-0.0352	0.5112	0.0552	9	No	No	5.4064
7	1.9144	0.529	0.053	132	-0.0384	0.2948	0.0594	2	No	No	5.555
8	0	0	0	0	0	0	0	0	Si	No	5
9	1.9107	0.4981	0.0519	166	-0.0061	0.5672	0.0718	21	No	No	5.2008
10	1.8848	0.3876	0.0425	104	-0.0147	0.5974	0.0671	19	No	No	5.8943

Tabla 5.6: Resultados obtenidos con *KinectTM* en el experimento 2.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	Perd	Conf	Calif
1	1.2288	0.258	0.0372	0	-0.0465	0.258	0.0242	1	No	No	8.9779
2	1.2143	0.1229	0.0244	0	-0.0631	0.2453	0.0526	4	No	No	9.2022
3	0	0	0	0	0	0	0	0	Si	No	5
4	1.2355	0.1725	0.025	0	-0.0574	0.2138	0.0504	0	No	No	9.3669
5	1.1287	0.1556	0.023	0	-0.0571	0.2118	0.0489	0	No	No	10
6	1.2644	0.157	0.028	0	-0.0614	0.2468	0.0423	1	No	No	10
7	1.2634	0.1236	0.0241	0	-0.0756	0.247	0.0276	6	No	No	10
8	1.3496	0.2424	0.0246	0	-0.068	0.2541	0.0387	1	No	No	10
9	1.3741	0.2485	0.0263	0	-0.0493	0.2544	0.0351	6	No	No	10
10	1.2622	0.209	0.0276	0	-0.0605	0.2285	0.0292	2	No	No	10

Tabla 5.7: Resultados obtenidos con láser+*KinectTM* en el experimento 2.

En las Figuras 5.12, 5.13 y 5.14 se graficó la información de distancia obtenidas con la robot Judy utilizando los 3 métodos de seguimiento.



Figura 5.12: Gráfica de distancias obtenidas con láser en el experimento 2.

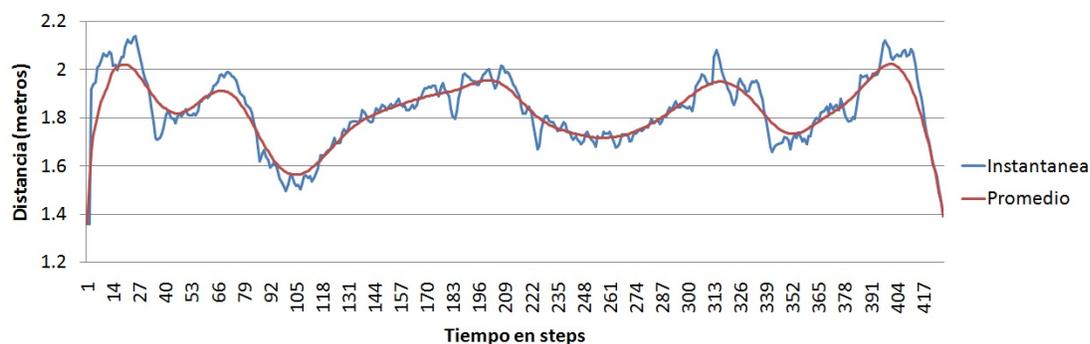


Figura 5.13: Gráfica de distancias obtenida con *KinectTM* en el experimento 2.

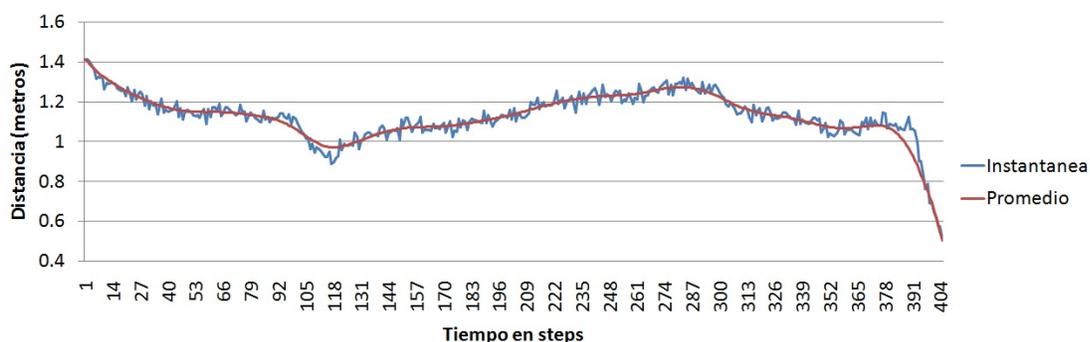


Figura 5.14: Gráfica de distancia obtenidas con láser+*KinectTM* en el experimento 2.

En las Figuras 5.15, 5.16 y 5.17 se graficó la información de orientación obtenidas con la robot Judy utilizando los 3 métodos de seguimiento.

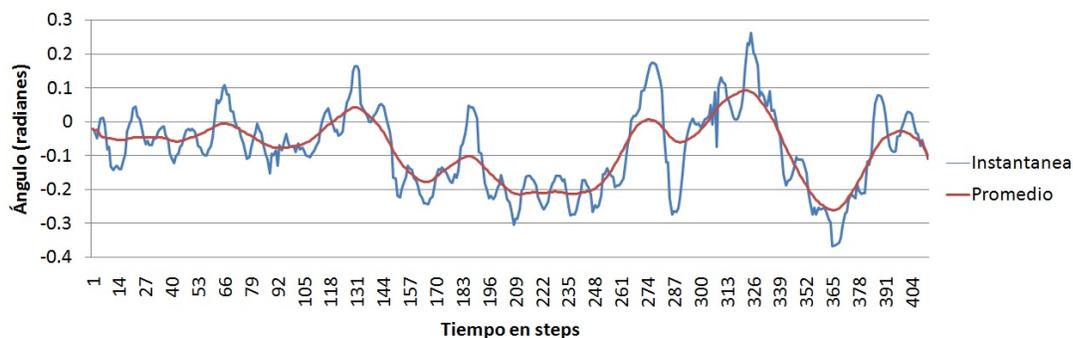


Figura 5.15: Gráfica de orientaciones obtenidas con láser en el experimento 2.

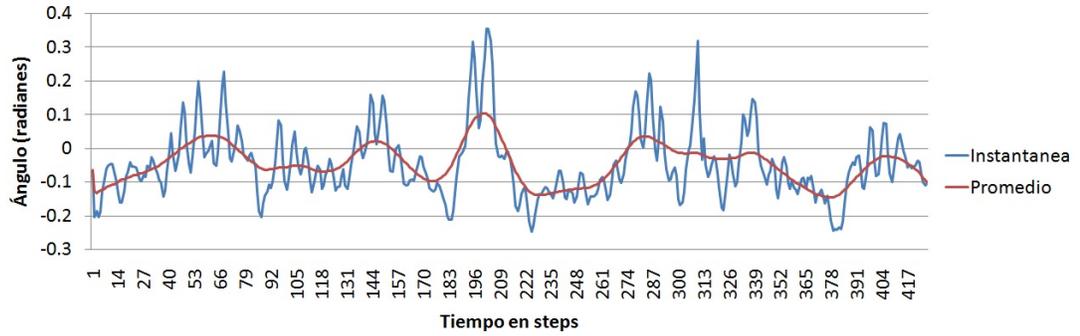


Figura 5.16: Gráfica de orientaciones obtenidas con *Kinect™* en el experimento 2.

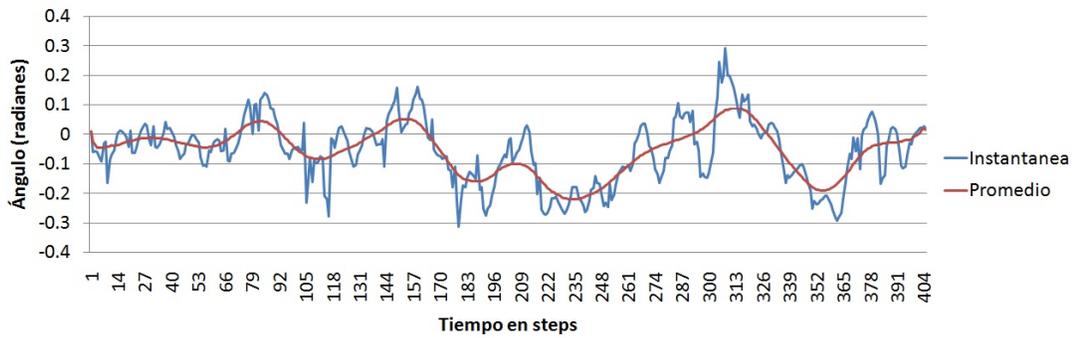


Figura 5.17: Gráfica de orientaciones obtenidas con láser+ *Kinect™* en el experimento 2.

5.2.3. Experimento 3 - Habitación con muebles y humanos

Este último experimento fue realizado en una habitación (Laboratorio de Bio-robótica) de 5.2×10.3 metros y un pasillo junto a la habitación de 2.3×5.3 metros. En la Figura 5.18 se muestra el recorrido del circuito que consiste de salir desde atrás de la habitación ubicada en el lado izquierdo de la imagen, salir de la habitación y avanzar por el pasillo hasta terminar del lado derecho inferior de la imagen, evitando los muebles y el humano existentes. Los muebles y el humano generan espacio reducido forzando a la robot a realizar giros bruscos generando una nueva dificultad, donde las áreas de color rosa representan obstáculos u objetos del ambiente, los círculos amarillos representan personas, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.

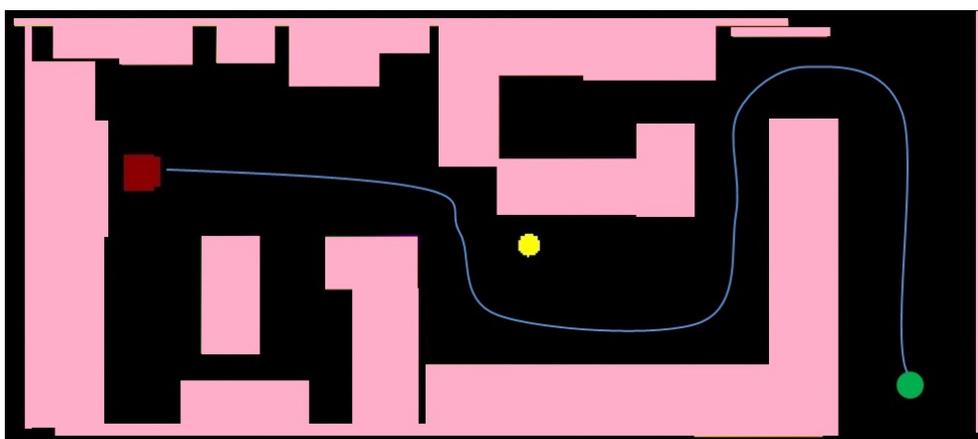
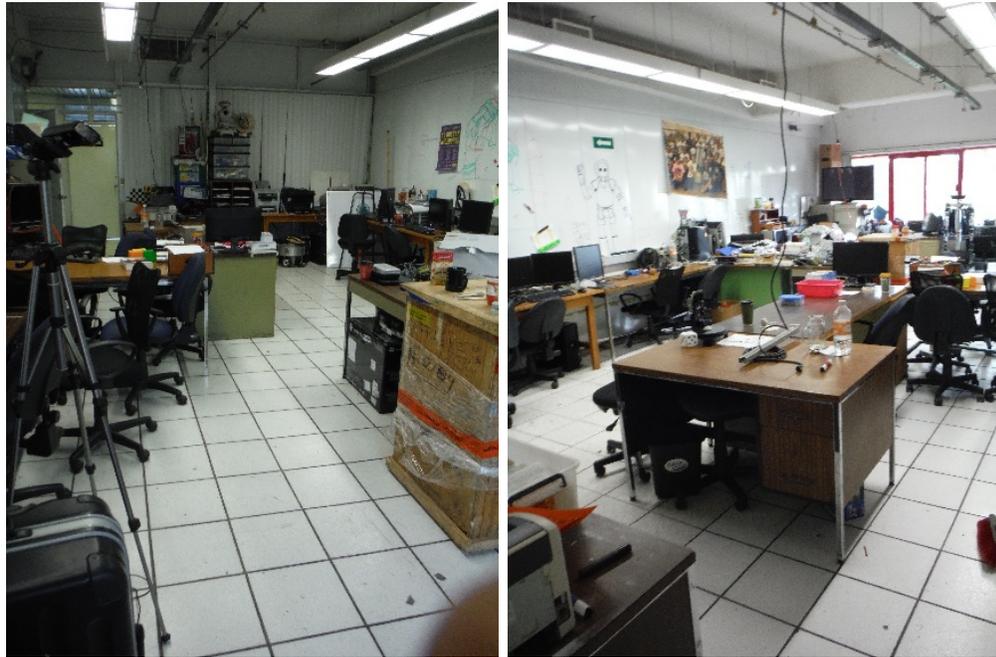


Figura 5.18: Esquema de la distribución territorial del experimento 3, las áreas de color rosa representan obstáculos u objetos del ambiente, los círculos amarillos representan personas, la línea azul indica el trayecto a seguir, el círculo verde representa el final del circuito y el rectángulo un poco deforme de color rojo la posición inicial de la robot.

En la Figura 5.19 se muestran fotografías del lugar donde fue realizado el experimento donde (a) es la vista desde donde se inicia el experimento, (b) es la vista desde el pasillo al interior de la habitación y (c) es la vista del pasillo.



(a)

(b)



(c)

Figura 5.19: Imágenes del lugar donde se probó el experimento 3 donde (a) es la vista desde donde se inicia el experimento, (b) es la vista desde el pasillo al interior de la habitación y (c) es la vista del pasillo.

En la Tabla 5.8 y 5.9 se tabulan los resultados obtenidos con la robot Judy utilizando 2 métodos de seguimiento.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	Perd	Conf	Calif
1	0.8891	0.1216	0.022	0	-0.0437	0.35	0.0584	27	No	No	7.9575
2	0.9577	0.144	0.0266	0	0.006	0.3718	0.0666	36	No	No	6.9792
3	0.9463	0.1636	0.0263	0	0.0194	0.3309	0.0639	44	No	No	6.8422
4	0.9727	0.1656	0.0265	0	-0.0395	0.3301	0.0614	16	No	No	7.9849
5	0.8991	0.1839	0.0257	0	0.0036	0.2951	0.0666	22	No	No	7.5884
6	0.9281	0.1631	0.0225	0	0.0142	0.3645	0.0509	28	No	No	7.3925
7	0	0	0	0	0	0	0	0	Si	No	5
8	0.9380	0.2007	0.0228	0	0.0285	0.3635	0.0561	27	No	No	7.3776
9	1.033	0.1708	0.024	0	-0.0255	0.3659	0.051	43	No	No	6.9499
10	0	0	0	0	0	0	0	0	Si	No	5

Tabla 5.8: Resultados obtenidos con láser en el experimento 3.

#	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	Perd	Conf	Calif
1	0.9719	0.2048	0.027	0	-0.0063	0.2893	0.0716	32	No	No	7.1737
2	0	0	0	0	0	0	0	0	Si	No	5
3	1.0264	0.1345	0.0233	0	-0.0317	0.2985	0.0484	11	No	No	8.5221
4	1.0889	0.1665	0.0234	0	-0.0226	0.2685	0.051	25	No	No	8.29
5	0.9661	0.1341	0.0239	0	0.0345	0.3283	0.059	32	No	No	7.127
6	0	0	0	0	0	0	0	0	Si	No	5
7	0.9893	0.1694	0.0264	0	0.0152	0.3242	0.0563	23	No	No	7.563
8	1.0867	0.1574	0.0241	0	0.0329	0.3152	0.0516	34	No	No	7.0743
9	0.9978	0.1395	0.0249	0	-0.0081	0.327	0.0628	29	No	No	7.2648
10	1.043	0.1893	0.0243	0	-0.0156	0.3683	0.0667	31	No	No	7.1752

Tabla 5.9: Resultados obtenidos con láser+*Kinect*TM en el experimento 3.

En las Figuras 5.20 y 5.21 se graficó la información de distancia obtenidas con la robot Judy utilizando los 2 métodos de seguimiento.



Figura 5.20: Gráfica de distancias obtenidas con láser en el experimento 3.

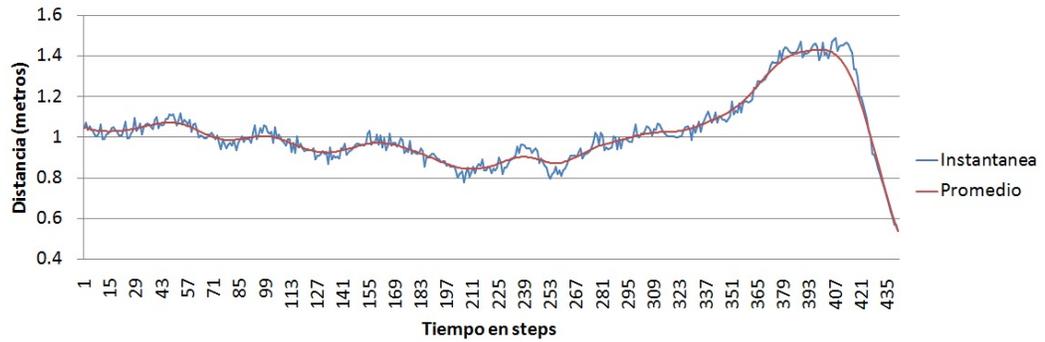


Figura 5.21: Gráfica de distancias obtenidas con láser+*Kinect*TM en el experimento 3.

En las Figuras 5.22 y 5.23 se graficó la información de orientación obtenidas con la robot Judy utilizando los 2 métodos de seguimiento.



Figura 5.22: Gráfica de orientaciones obtenidas con láser en el experimento 3.

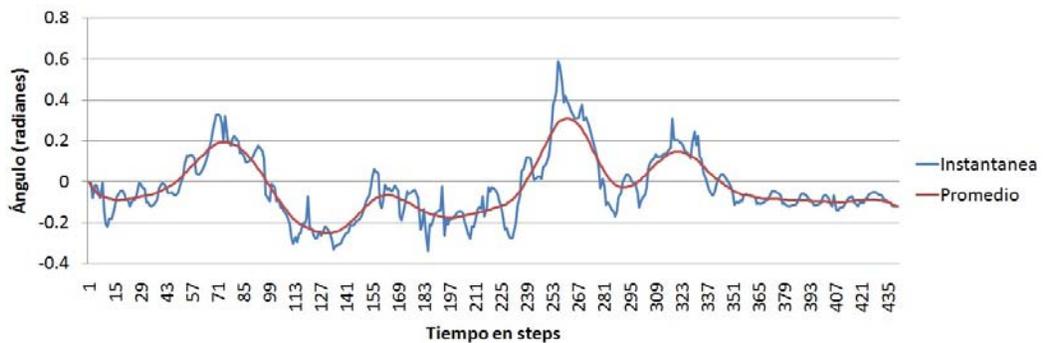


Figura 5.23: Gráfica de orientaciones obtenidas con láser+*Kinect*TM en el experimento 3.

La información del tercer experimento usando solo el *KinectTM* no pudo ser recolectada, dado a que nunca se pudo terminar satisfactoriamente las pruebas debido a las vueltas cerradas con las que contaba la prueba.

5.2.4. Competencias

Un poco antes de la terminación del trabajo se nos dió la oportunidad de ponerlo en marcha en una serie de competencias a nivel nacional e internacional. Se compitió con el equipo *pumas@home* del Laboratorio de Biorobótica de la Universidad Nacional Autónoma de México.

Las competencias son las siguientes:

- Torneo Mexicano de Robótica (TMR2014) celebrado en Ciudad del Carmen, Campeche, en el cual se ganó el 1er lugar, y el equipo logró el mayor puntaje de todos los equipos, en esta ocasión se operó con la robot Justina.
- *Robocup* 2014 celebrado en João Pessoa, Brasil, en el cual se ganó el 5to lugar, y el equipo logró el 4to mejor puntaje de todos los equipos en la prueba de *follow me*, así como terminar la primera fase con el mejor record y la segunda con el segundo mejor record de todos; en esta ocasión se operó con la robot Judy. En la Tabla 5.10 se muestran los puntajes recibidos de cada equipo en dicha prueba.

#	1	2	3	4	5
Equipo	Wright Eagle	NimbRo	ToBi	Pumas	TU/e
Puntuación	811	631	600	575	484
#	6	7	8	9	10
Equipo	MRL	AUT	UChile	Reem	Tinker
Puntuación	287	100	100	100	0

Tabla 5.10: Puntuación en prueba *Follow me Robocup@home* 2014.

Cabe señalar que estos resultados fueron logrados únicamente con la utilización del láser. En la Figura 5.24 se muestran algunas fotografías de Judy mientras realizaba la prueba *Follow me* en la competencia de *Robocup*.



(a)

(b)



(c)

Figura 5.24: Judy ejecutando *follow me* en *Robocup* 2014 donde (a) es Judy antes de la oclución, (b) es Judy en el elevador y (c) es Judy terminando la segunda fase.

5.3. Análisis

En la Tabla 5.11 se muestran los promedios de los experimentos donde se pueden visualizar las diferencias entre ellos.

Exp	d_{dp}	d_{dm}	d_{dd}	d_{dl}	o_{op}	o_{om}	o_{od}	o_{ol}	Calif	Fin
Exp1l	1.2753	0.2269	0.0366	3.6	-0.0563	0.2145	0.0463	3.5424	8.9617	1
Exp1k	1.8914	0.3284	0.0392	182.6667	-0.0514	0.2012	0.0354	0.1111	7.9184	0.9
Exp1l+k	1.1986	0.1685	0.0294	0	-0.0611	0.2188	0.0502	5.2009	9.198	1
Exp2l	1.2944	0.1753	0.0309	0	-0.0696	0.2356	0.0475	4.8368	7.6211	0.6
Exp2k	1.8649	0.4488	0.0487	98.875	-0.0304	0.4402	0.0607	10.5	5.7994	0.8
Exp2l+k	1.2579	0.1877	0.0267	0	-0.0599	0.24	0.0388	2.3271	9.2547	0.9
Exp3l	0.9455	0.1642	0.0245	0	-0.0046	0.3465	0.0594	30.2837	6.9072	0.8
Exp3l+k	1.0213	0.1619	0.0247	0	-0.0002	0.3149	0.0584	27.0569	7.019	0.8

Tabla 5.11: Promedios de los experimentos.

El solo uso del láser obtuvo un buen resultado, debido a que el láser posee un amplio ángulo de visión de hasta 240° y que su procesamiento es ligero, sin embargo, cuando existe oclusión puede fallar. El usar solo el *KinectTM* no fue tan bueno a consecuencia quizás, de su reducido ángulo de visión y la baja velocidad al procesar, lo que lo vuelve más sensible a errores, tal es el hecho que el tercer experimento no lo pudo concluir ni una sola vez y hasta el primer experimento fallo una vez, el *KinectTM* tuvo menores calificaciones pero cuando se trato de reconocimiento acertó en más ocasiones. Al combinar lo mejor de ambas técnicas se obtienen buenos resultados, ésto se puede ver en el experimento número 2 donde el promedio en las calificaciones es mayor por casi 1.5 puntos.

En las gráficas de distancia se puede observar que las distancias son muy estables, en cambio las gráficas de orientación son un poco menos estables presentando más alteraciones con respecto a la línea promedio. Los números en los campos de d_{dd} y o_{dd} también arrojan esta característica. Por otro lado, en la competencia de *Robocup* se obtuvo una buena puntuación, la cual está a $\frac{3}{4}$ de la puntuación máxima, solo 50 puntos por debajo de la segunda mejor calificación y casi 100 puntos por arriba de la calificación más alta por debajo de la nuestra.

Capítulo 6

Conclusiones y trabajo a futuro

6.1. Conclusiones

En el presente trabajo se desarrolló un sistema de seguimiento de personas con una robot de servicio, durante su realización se crearon una serie de experimentos y se desarrolló *software* especial para dicha tarea, donde se aprendió que la problemática del seguimiento puede pasar de ser un proceso casi trivial a ser un caso de lo más complicando, y su nivel de dificultad varía dependiendo del medio ambiente en el que se desarrolle esta actividad.

El seguimiento por piernas resulto ser muy rápido y por ende robusto, sin embargo, cuando sucede una oclusión puede ser el fin del seguimiento, por el contrario, el seguimiento por visión no es muy bueno, dado el poco ángulo de visión del *KinectTM* y el constante tráfico generado por los módulos involucrados, aunque el reconocimiento es mucho mejor al contemplar la vestimenta del operador. Con la mezcla de ambas técnicas se obtiene un resultado sobresaliente, al combinar lo mejor de cada una de ellas se logra un seguimiento firme y utilizando el reconocimiento con visión la probabilidad de que se confunda es menor.

Los números que arrojaron las pruebas comprueban lo anteriormente dicho, las pruebas con solo el láser siempre presentaron mejores resultados que las pruebas que utilizaban solo *KinectTM*, salvo el segundo experimento donde el láser se confundía y el *KinectTM* no.

El modelo HSV es mejor que el modelo RGB para clasificar los colores a través de un canal, pero su debilidad está en los tonos grises, por el contrario, el modelo RGB en un solo canal no determina el color, pero por su separación de los canales se puede distinguir los tonos grises de manera más eficaz, así que al combinarlos se mejora el reconocimiento de las prendas estableciendo umbrales para la identificación de los colores.

El utilizar el filtro de Kalman nos ayuda en el hecho de tener una sugerencia de la posición del operador haciéndolo más robusto a las oclusiones, y usando los campos repulsivos la evasión de obstáculos se realiza con mayor naturalidad.

6.2. Trabajo a futuro

Ahora se muestra algunas propuestas consideradas como continuación de este trabajo:

6.2.1. Detección de piernas y personas

El método geométrico que se utilizó resulta muy eficiente, sin embargo, se cree que se puede obtener mejores resultados utilizando diferentes técnicas como el empleo de redes neuronales para capturar mejor el patrón tanto de las piernas como de los cuerpos, dado que es algo constante no habría necesidad de reentrenamientos simplemente se carga la red y se pone a funcionar, esto con la idea de reducir los falsos positivos que se obtienen.

6.2.2. Reconocimiento de operador

El algoritmo utilizado por medio de la información de color resulta bueno pero tiene sus desventajas al estar atado únicamente a rotaciones horizontales, para este proyecto funciona, sin embargo, existen más variedad de casos que en ciertas situaciones se pudieran presentar y el algoritmo debería estar preparado. De igual manera se podría ocupar algunos métodos más heurísticos para robustecer el reconocimiento y utilizar programación paralela para procesar la información.

6.2.3. Planeación de movimientos

La planeación de movimientos a corta distancia es muy buena pero no perfecta, dado que se tienen muchos problemas en vueltas muy cerradas, así como en espacios estrechos de menos de 80cm de ancho, sería bueno tener un segundo panorama con un radio de vista mucho mayor, al mismo tiempo que nos permita anticiparnos a este tipo de situaciones y resolverlas con mayor elegancia, el conocer el ambiente serviría pero no siempre se puede tener esta información.

Otro factor importante es el tipo de seguimiento, en este caso es posicionado en la parte de atrás, sin embargo, eso en nuestro mundo no es muy natural, generalmente debería de ser a lado, para lo cual se requiere otro tipo de movimiento diferente al estipulado aquí.

Apéndice A

Reglamento de la prueba de seguimiento de personas del *Robocup@Home*.

A.1. Prueba *Follow Me*

El robot debe de seguir satisfactoriamente a una persona desconocida a través de un espacio público.

A.2. Objetivo

Esta prueba se centra en el seguimiento y reconocimiento de una persona previamente desconocida, con iteración básica y capacidades de señalización, y navegación segura en ambientes desconocidos y espacios estrechos con otras personas caminando o bloqueando el camino.

A.3. Configuración

1. Lugar: La prueba se realiza en un lugar fuera de la arena en un espacio público.
2. Operador: Un operador “profesional” es seleccionado por el CT(Comité Técnico) para probar el robot.
3. Otras personas: No hay restricciones sobre otras personas caminando o que estén de pie alrededor mientras se completa la tarea.

4. Ruta: Una ruta es preparada (pero no anunciada) de antemano. La ruta completa es dividida en tres secciones por dos puntos de tiempos intermedios (ITPs).

A.4. Tarea

1. Inicio: El robot inicia en una designada posición inicial, y espera por el operador “profesional”. Cuando el árbitro comienza el tiempo, le es permitido al equipo (por poco tiempo) instruir al operador. Después de las instrucciones, el operador se coloca delante del robot y le dice que lo siga (sin señal de arranque).
2. Memorizando al operador: El robot debe memorizar al robot, durante esta fase el robot puede dar instrucciones al operador a seguir ciertos procedimientos de movimientos.
3. Siguiendo al operador: Cuando el robot recibe la señal de que está realmente listo para iniciar, el operador camina - de forma natural - en la dirección designada. El robot necesita seguir al operador. El robot se ocupa de diferentes obstáculos (personas individuales, habitaciones de ascensores estrechos y pequeñas multitudes) en diferentes secciones. Cada sección tiene una medición de tiempo por separado, pero la tarea completa debe realizarse dentro del tiempo de tarea general.
 - a) 1ra sección (desde el inicio hasta el primer ITP): dos personas bloquean directamente el paso sobre el camino del primer ITP (en diferentes posiciones). El operador guía al robot alrededor de las personas. Una de las personas comienza a caminar cuando se acerca el robot, y recorre el camino entre el robot y el guía. Justo en medio, la persona deja de caminar durante 3 segundos antes de alejarse.
 - b) 2da sección (entre el primer y segundo ITP): El operador guía al robot dentro de un pequeño cuarto estrecho (ej. un elevador). No hay espacio suficiente para operar libremente en este cuarto (ej. 1m x 2m). Otras personas podrían estar ahí causando que el operador y el robot tengan que salir en orden inverso (de conforme entraron). El equipo puede elegir de acuerdo a las siguientes opciones y instruir al operador en consecuencia cuando inicie la prueba.
 - 1) El operador entra primero, y ordena al robot cuando salir. El robot se espera a salir del cuarto, (La puerta debería estar abierta), espera por el operador, y automáticamente sigue al guía de nuevo.

- 2) El operador ordena al robot que entre en primer lugar, y sale primero al salir. El robot se espera entrar al cuarto (la puerta debe estar abierta), espera al operador, y de forma automática seguir cuando el operador abandona la habitación de nuevo.

Ambos comandos podrían ser naturalmente hablados tales como “sal del elevador”. Controlar remotamente al robot con formas de posiciones específicas, orientaciones o movimientos no es permitido. Entre entrar y salir del cuarto, la puerta estará cerrada por al menos 5 segundos. En caso de que el cuarto no esté directamente accesible, el tiempo se detiene mientras se espera.

- c) 3ra sección (entre la segunda ITP y la línea final): Un pequeño grupo de personas (4-5) podrían estar esperando afuera del cuarto, bloqueando el camino entre el segundo ITP y la línea final. El operador se cuela a través de la multitud y espera al robot en el otro lado. El robot no puede pasar a través de las personas y debe navegar de forma autónoma alrededor del grupo. Mientras espera, el operador tiene permitido hacer señales al robot desde donde se encuentra parado (ej. agitando o haciendo un sonido), pero no tiene permitido moverse hacia atrás, Después de que el robot está siguiendo al operador de nuevo, el operador se dirige a la línea final.

A.5. Reglas y observaciones adicionales

1. Preparación: El robot tiene que esperar por lo menos durante 1 minuto antes de que aparezca el operador en frente del robot. Durante este tiempo de espera, el equipo no se le permite tocar el robot.
2. Caminar Natural: El operador tiene que caminar “natural”, es decir, avanzar hacia adelante. Si no se menciona lo contrario, el operador no se le permite caminar hacia atrás, detenerse, señalar el robot o seguir algún procedimiento de recalibración.
3. Pidiendo el paso: El robot se le permite (suavemente) pedir a personas individuales hacerse a un lado, pero no se le permite gritar a ciegas en grupos de personas.
4. Las perturbaciones del exterior: Si una persona de la audiencia (severamente) interfiere con el robot de una manera que hace que sea imposible resolver la tarea, el equipo puede repetir la prueba inmediatamente.

5. Instrucción: El robot interactúa con el operador, no el equipo. Es decir, al equipo sólo se les permite (¡muy!) brevemente instruir al operador.
 - Cómo el decirle al robot seguir,
 - Cómo visualmente / acústicamente señalar que (por ejemplo, agitando o llamando el nombre del robot), y
 - Cómo ordenar al robot dentro o fuera del cuarto estrecho (ITP 2); se permite que el operador indique tanto la entrada y la salida del ascensor sólo por un único comando.
6. Llamar a la operadora de vuelta: Cuando el robot ha perdido el operador, puede llamar a la operadora de vuelta una vez por sección (se perderán los puntos de esa sección).
7. Medida del tiempo: El tiempo de la primera sección (desde el principio hasta el primer ITP) se inicia cuando el operador pasa en frente del robot.

A.6. Instrucciones del Comité de Organización y Árbitros

En cualquier momento antes de la prueba:

- Definir la ruta y los ITPs donde se mide el tiempo.
- Seleccionar al operador(es) “profesional”.

Durante la prueba:

- Mostrar el camino y los ITPs a los operadores.
- Tomar el tiempo (acumulado) en cada ITP y en la línea de meta.
- Comprobar que la puerta del ascensor se cierra por el tiempo especificado.
- Comprobar buen funcionamiento del robot; el robot debe ser detenido inmediatamente si una persona va a ser tocada por el robot.

A.7. Hoja de Puntuación

El tiempo máximo de la prueba es de 8 minutos.

Acción	Puntaje
Puntos para las tres secciones	
para el mejor tiempo Δt_1 archivado en la sección 1, menos puntos son dados para el tiempo más largo (ver la Normalización)	300
para el mejor tiempo Δt_2 archivado en la sección 2, menos puntos son dados para el tiempo más largo (ver la Normalización)	300
para el mejor tiempo Δt_3 archivado en la sección 3, menos puntos son dados para el tiempo más largo (ver la Normalización)	300
Puntos adicionales	
completar la tarea completa	100
Sancciones especiales y bonificaciones	
No atender (ver sec. 3.7.1 de [?])	-500
Excelente rendimiento	100
Puntuación total (excluyendo multas y bonificaciones)	1000

Puntuación de observaciones: La puntuación está basada en el tiempo (acumulado) tomado en cada ITP (punto en el tiempo intermedio) y en el punto de meta final (ej. $t_1 = 125\text{seg}$; $t_2 = 305\text{seg}$; $t_3 = 425\text{seg}$). A continuación, los intervalos de tiempo necesarios para cada sección se calculan (ej. $\Delta t_1 = 125\text{seg}$; $\Delta t_2 = 180\text{seg}$; $\Delta t_3 = 120\text{seg}$). Las puntuaciones son normalizadas a continuación, en relación con el mejor intervalo de tiempo logrado para esta sección (ej. $\Delta t_1^{\min} = 100\text{seg}$; $\Delta t_2^{\min} = 90\text{seg}$; $\Delta t_3^{\min} = 120\text{seg}$). En ejemplo una puntuación completa de $240 + 150 + 300 = 790$ puntos es obtenida:

Normalización: Si un equipo necesita más tiempo en una sección que el mejor equipo, la puntuación de esa sección i se calcula por:

$$\text{normalizedscoreforsection } i = 300 * (\Delta t_i^{\min} = \Delta t_i) \quad (\text{A.1})$$

donde $\Delta t_1 = t_1 - 0$; $\Delta t_2 = t_2 - t_1$; $\Delta t_3 = t_3 - t_2$, and Δt_1^{\min} , Δt_2^{\min} , Δt_3^{\min} son el mínimo tiempo obtenido por el equipo.

Bibliografía

- [AM01] Carlos Aschero y Jorge Guillermo Martínez, “Técnicas de caza en antofagasta de la sierra, puna meridional argentina”, *Relaciones de la Sociedad Argentina de Antropología*, vol. 26, 2001.
- [BCF⁺98] Wolfram Burgard, Armin B Cremers, Dieter Fox, Dirk Hähnel, Gerhard Lakemeyer, Dirk Schulz, Walter Steiner, y Sebastian Thrun, “The interactive museum tour-guide robot”, *AAAI/IAAI*, 1998, pp. 11–18.
- [BCF⁺99] Wolfram Burgard, Armin B Cremers, Dieter Fox, Dirk Hähnel, Gerhard Lakemeyer, Dirk Schulz, Walter Steiner, y Sebastian Thrun, “Experiences with an interactive museum tour-guide robot”, *Artificial intelligence*, vol. 114, Elsevier, 1999, pp. 3–55.
- [Bec12] Marco Becerra, “Sistema de seguimiento de personas para un robot móvil de servicio”, Tesis de Maestría, 2012, *Universidad Nacional Autónoma de México*.
- [CCC09] Tse-Wei Chen, Yi-Ling Chen, y Shao-Yi Chien, “Fast image segmentation and texture feature extraction for image retrieval”, *IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)*, IEEE, 2009, pp. 854–861.
- [Cen84] Artificial Intelligence Center, “Shakey the robot”, Reporte técnico, SRI International, 1984.
- [CJ06] Punarjay Chakravarty y Ray Jarvis, “Panoramic vision and laser range finder fusion for multiple person tracking”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2006, pp. 2949–2954.
- [CO02] Elisa Colombo y Beatriz O’Donell, “Manual de iluminación eficiente: Capítulo 2 luz, color y visión”, *Universidad Tecnológica Nacional y Eli Argentina*, 2002.

- [CO.14] Hokuyo Automatic CO., *Electronic references*, https://www.hokuyo-aut.jp/02sensor/07scanner/urg_04lx_ug01.html, 2014.
- [Coe04] Carlos A Coello, “Breve historia de la computación y sus pioneros”, *Fondo de Cultura Económica*, 2004.
- [Cru] Jesús Cruz, “Optimización de campos potenciales para navegación mediante algoritmos genéticos”, Tesis de Licenciatura, 2013, *Universidad Nacional Autónoma de México*.
- [CT] Roberto Cordeschi y Guglielmo Tamburrini, “Intelligent machines and warfare”.
- [CWZ07] Yuwan Cen, Lihua Wang, y Handong Zhang, “Real-time obstacle avoidance strategy for mobile robot based on improved coordinating potential field with genetic algorithm”, *IEEE International Conference on Control Applications*, IEEE, 2007, pp. 415–419.
- [dav13] davidbuckley.net, *Electronic references*, <http://www.davidbuckley.net/DB/HistoryMakers.htm>, 2013.
- [EF89] Luis Estrada y Jorge Flores, “Perspectivas en la biología y en la física”, *UNAM, Centro Universitario de Comunicación de la Ciencia*, 1989.
- [FD06] Jodi Forlizzi y Carl DiSalvo, “Service robots in the domestic environment: a study of the roomba vacuum in the home”, *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, ACM, 2006, pp. 258–265.
- [FND03] Terrence Fong, Illah Nourbakhsh, y Kerstin Dautenhahn, “A survey of socially interactive robots”, *Robotics and autonomous systems*, vol. 42, Elsevier, 2003, pp. 143–166.
- [GCV90] Georges Giralt, Raja Chatila, y Marc Vaisset, “An integrated navigation and motion control system for autonomous multisensory mobile robots”, *Autonomous robot vehicles*, Springer, 1990, pp. 420–443.
- [GTR07] GTRob, “Libro blanco de la robótica: de la investigación al desarrollo tecnológico y aplicaciones futuras”, *Comité Español de Automática*, 2007.
- [IBM13] IBM, *Electronic references*, <http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/deepblue/>, 2013.

- [ifr13] ifr.org, *Electronic references*, <http://www.ifr.org/service-robots/>, 2013.
- [IH05] Keijiro Ishii, Mineo Yamamoto y Kazuhito Hyodo, “Stand-alone wearable power assist suit-development and availability”, *Journal of robotics and mechatronics*, vol. 17, FUJI TECHNOLOGY PRESS LTD., 2005, p. 575.
- [LCLT07] RC Luo, YJ Chen, CT Liao, y AC Tsai, “Mobile robot based human detection and tracking using range and intensity data fusion”, *IEEE Workshop on Advanced Robotics and Its Social Impacts*, IEEE, 2007, pp. 1–6.
- [LWF05] James NK Liu, Meng Wang, y Bo Feng, “iBotGuard: an internet-based intelligent robot security system using invariant face recognition against intruder”, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 35, IEEE, 2005, pp. 97–105.
- [MJ12] Mahmoud Mirabi y Shahram Javadi, “People tracking in outdoor environment using Kalman filter”, *Third International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, IEEE, 2012, pp. 303–307.
- [MV12] Francesc Mestres y Josep Vives, “Los cyborgs y los robots: evolución humana y aumentación”, *Ludus Vitalis*, vol. 20, 2012, pp. 225–252.
- [NAO06] Aldebaran SoftBank Group NAO, *Electronic references*, <http://www.aldebaran.com/en>, 2006.
- [NWM13] Bingbing Ni, Gang Wang, y Pierre Moulin, “RGBD-hudaact: a color-depth video database for human daily activity recognition”, *Consumer Depth Cameras for Computer Vision*, Springer, 2013, pp. 193–208.
- [ope14] opencv.org, *Electronic references*, <http://opencv.org/>, 2014.
- [pla15] play.com, *Electronic references*, <http://www.play.com/stores/CHIPSWORLD/listing/686819009>, 2015.
- [Ram03] Álvaro Solera Ramírez, “El filtro de Kalman”, *Documento de trabajo del Banco Central de Costa Rica, elaborado en la División Económica, Departamento de Investigaciones Económicas* (2003).

- [Rob13] Robocup, *Electronic references*, <http://www.robocup.org/>, 2013.
- [ros14] ros.org, *Electronic references*, <http://www.ros.org/about-ros/>, 2014.
- [Rul14] RoboCup@Home Rulebook, “RoboCup@Home Rulebook”, *RoboCup@Home Rulebook / Draft version for RoboCup*, 2014.
- [SCH98] Billingham Mark Savage-Carmona, Jesus y Alistair Holden, “The Vir-Bot: a virtual reality robot driven with multimodal commands”, *Expert Systems with Applications*, vol. 15, Elsevier, 1998, pp. 413–419.
- [Smi78] Alvy Ray Smith, “Color gamut transform pairs”, *ACM Siggraph Computer Graphics*, vol. 12, ACM, 1978, pp. 12–19.
- [SWA⁺02] Yoshiaki Sakagami, Ryujin Watanabe, Chiaki Aoyama, Shinichi Matsunaga, Nobuo Higaki, y Kikuo Fujimura, “The intelligent asimo: System overview and integration”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, IEEE, 2002, pp. 2478–2483.
- [Váz04] Juan Luis Vázquez, “Matemáticas, ciencia y tecnología: una relación profunda y duradera”, *De la aritmética al análisis: historia y desarrollos recientes en matemáticas*, Ministerio de Educación, 2004, p. 183.
- [WKT06] Shih-Ku Weng, Chung-Ming Kuo, y Shu-Kang Tu, “Video object tracking using adaptive Kalman filter”, *Journal of Visual Communication and Image Representation*, vol. 17, Elsevier, 2006, pp. 1190–1208.
- [Wor13] Honda Worldwide, *Electronic references*, <http://world.honda.com/ASIMO/history/>, 2013.
- [WS05] David P Watson y David H Scheidt, “Autonomous systems”, *Johns Hopkins APL technical digest*, vol. 26, 2005, pp. 368–376.
- [Zie] Herbert Ziegler, “El legado de la II guerra mundial”, *Centro Virtual Estudios Judaicos*.
- [ZK05] Zivkovic Zoran Zajdel, Wojciech y BJA Krose, “Keeping track of humans: have I seen this person before?”, *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, IEEE, 2005, pp. 2081–2086.