VNIVERSIDAD NACIONAL
AVTNOMA DE
MEXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

IMPLEMENTACIÓN DE LA PRUEBA FOLLOW ME DEL CONCURSO ROBOCUP AT HOME UTILIZANDO MODELOS DE DIÁLOGO Y UNA ARQUITECTURA COGNITIVA

TESIS

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS (COMPUTACIÓN)

PRESENTA:

ING. ARTURO RODRÍGUEZ GARCÍA

DIRECTOR DE TESIS:

DR. LUIS ALBERTO PINEDA CORTÉS

MÉXICO, DF. 2012

Agradecimientos

A todos los que he seguido,

y a todos los que me han seguido…

Mención especial al Consejo Nacional de Ciencia y Tecnología (CONACYT) que apoyó mis estudios de maestría.

Agradezco el apoyo de los proyectos CONACYT 81965 y PAPPIT-UNAM IN-115710 por apoyar con recursos a esta tesis.

Índice

Resumen

En este trabajo de investigación se describe la forma en que se implementó en un robot móvil la habilidad de seguir a una persona, brindándole las habilidades necesarias para poder resolver en forma exitosa la prueba *Follow Me* del concurso *RoboCup@Home*. En esta competencia, el robot no sólo debe desplazarse detrás de una persona, sino que debe interactuar con ella para lograr superar una serie de eventos que retan su capacidad de entender comandos verbales y gestuales, reconocer a la persona correcta en diversas circunstancias y navegar en un ambiente desconocido. El marco de trabajo para la creación de este sistema interactivo multimodal fue el uso de modelos de diálogo y de una arquitectura cognitiva. Se buscaron e integraron las herramientas tecnológicas y algoritmos necesarios para que el robot fuera capaz de resolver la tarea. Además, se construyó un agente de visión dedicado al reconocimiento de personas y gestos. El sistema fue sometido a una serie de evaluaciones, que permitieron detectar fallas y puntos débiles, y que fueron corregidos mediante la incorporación de rutinas de recuperación en los modelos de diálogo.

Capítulo 1

Introducción

Tener un robot para uso personal que ayude en tareas cotidianas como lavar platos, cortar el césped o preparar una taza de café, ha sido uno de los grandes anhelos de la humanidad desde hace muchos años. La literatura, el cine y la televisión se han encargado de alimentar esta idea mostrando un futuro en el que los humanos delegan las tareas rutinarias del hogar o la oficina a los robots.

Aunque faltan todavía muchos años para que los robots sean producidos en serie e invadan los hogares del mundo formando parte de la vida cotidiana de las personas, el diseño y construcción de robots de servicio es un tema de interés para la comunidad científica y para la industria. Uno de los objetivos de la investigación en torno a robots para asistir a los humanos, es lograr que estos puedan ejecutar tareas cada vez más complejas, y además se busca que lo hagan de la manera más parecida a como lo hacen los humanos.

En este trabajo, se implementa en un robot móvil autónomo la habilidad de seguir a una persona. No sólo se busca que el robot camine paso tras paso detrás de un humano, sino que establezca una interacción con él. Este robot seguidor debe tener distintas habilidades, como el poder entender comandos verbales o gestuales del usuario, poder evadir obstáculos del ambiente mientras sigue a la persona, ser capaz de identificar de nuevo a la persona cuando la ha perdido de vista, poder expresar al usuario mensajes para darle instrucciones o pedirle ayuda, etc.

Un robot que es capaz de seguir a una persona, puede tener un gran impacto en diversas aplicaciones destinadas a asistir a los humanos en sus labores cotidianas. Guiar al robot hasta determinado lugar para que allí ejecute determinada tarea, o pedirle que siga a una persona para que la cuide, son ejemplos de aplicaciones que se le podrían dar a esta habilidad del robot.

En este capítulo introductorio se expone en la sección 1.1 la importancia que tiene la habilidad de seguimiento en los robots de servicio. En la sección 1.2 se aborda brevemente la naturaleza de la actividad de seguimiento cuando es realizada por humanos. A continuación, en la sección 1.3 se delimita el alcance de este trabajo de investigación. Enseguida, en la sección 1.4 se describe en forma breve la metodología que será utilizada para el diseño y la implementación del robot seguidor. En la sección 1.5 se hacen explícitos los objetivos de la investigación y se aborda cuál es la contribución de este proyecto. Finalmente, en la sección 1.6 se explica la forma en que están organizados los capítulos de este trabajo.

1.1 Habilidad de seguimiento en robots móviles

Un robot que puede seguir a una persona e interactuar con ella puede ser de gran utilidad en diversas situaciones de la vida real. La habilidad de seguimiento podría ser usada en robots de asistencia para centros comerciales. Imaginemos que un robot tiene la función de resolver dudas a los clientes. El usuario podría necesitar información extra de determinado producto o de un cartel promocional, entonces podría acercarse al robot y decirle "sígueme", para llevarlo hasta el lugar específico donde se encuentra el elemento en cuestión, y una vez en ese sitio decirle "explícame esto", seguido de un gesto como apuntar hacia el producto o cartel. Esta misma funcionalidad serviría para un robot guía de un museo, pudiendo seguir al visitante hasta una determinada zona con el propósito de brindarle una explicación.

Un robot seguidor también podría desempeñar la función de acompañante. Sería práctico tener a un robot que se pudiera llevar al supermercado y pudiera seguir al dueño entre los pasillos, siendo su función cargar en una canastilla los productos que el usuario elige, evitándole así estar cargando la mercancía o arrastrando un carrito de compras. Esto sería de gran ayuda para personas con discapacidad.

Las posibles aplicaciones de la habilidad de seguimiento son muchas, pero es necesario separar cuáles de ellas son factibles en el momento actual y cuáles no. Una de las limitaciones es el factor económico, pues en la actualidad los robots de servicio tienen un costo muy alto. Por esta razón es difícil pensar que hoy en día, un robot de este tipo pueda ser fabricado en masa y utilizado en hogares reales. Sin embargo, existen lugares donde sí hay oportunidad de tener un robot de servicio, por ejemplo, en un museo, en un parque de diversiones o en un centro comercial de lujo, donde el robot por sí mismo podría ser parte del atractivo del lugar y atraer más gente.

Otra de las limitaciones está relacionada a los obstáculos técnicos que existen hoy en día en el desarrollo de tecnologías para robots de servicio, pues la mayoría de ellos funcionan correctamente sólo en ambientes muy específicos. Un robot puede realizar muy bien determinada tarea si está en un lugar donde las condiciones ambientales (como iluminación, sonido ambiental, etc.) están muy controladas, pero al modificar las condiciones de ese escenario o solicitar al robot que haga esa misma tarea en otro lugar, el resultado es por lo general desastroso.

La oportunidad de que los robots de servicio tengan un impacto real en el bienestar de los humanos todavía está a varios años de consumarse, pero esto no debe ser un impedimento para continuar con la investigación en torno a este tipo de aplicaciones, pues es muy probable que en el futuro sea más fácil encontrarlos en distintos lugares de la vida real, como resultado de un mejoramiento en las tecnologías que utilizan y de una disminución en su costo.

1.2 ¿Qué es seguir a una persona?

De acuerdo con la Real Academia Española, seguir puede definirse como "ir detrás de alguien", pero también puede definirse como "ir en compañía de alguien". El robot seguidor que se implementa en este trabajo, más que ir detrás de una persona, la acompañará. Esto quiere decir que se desea que el robot no camine en forma pasiva tras el usuario, sino que establezca una interacción en la cual el robot le informe al humano sobre su estado, lo que está esperando, lo que está entendiendo, lo que hará, etc.

Seguir es una tarea en la que existe al menos un sujeto que asume el rol de guía y al menos un sujeto que asume el rol de seguidor. En lo sucesivo se considerará el caso en que existe sólo un guía y sólo un seguidor.

Cuando dos humanos entablan en forma consciente una actividad de seguimiento se establece una conversación que involucra en forma activa a ambos sujetos. Esto no ocurre cuando uno de los humanos no está consciente de lo que sucede, por ejemplo, si un ladrón sigue a otra persona sin que se dé cuenta, no existe una conversación entre ambos individuos. La conversación que se establece involucra comunicación verbal y no verbal.

La comunicación verbal en la actividad de seguimiento se da a través de mensajes entre las personas que participan. "sígueme", "ven por aquí", "apúrate" o "cuidado con ese escalón" son ejemplos de mensajes que le puede dar un guía a su seguidor. Por otra parte, "voy detrás de ti", "espera un momento, vas muy rápido" o "no te veo, ¿dónde estás?" son mensajes que el seguidor puede darle a su compañero.

También hay comunicación no verbal en la actividad de seguimiento. Aquí podemos mencionar a distintos comportamientos de los interlocutores (que pueden ser conscientes o inconscientes, y que se pueden dar en forma aislada o en forma paralela a la emisión de mensajes verbales):

- Comportamientos kinésicos, que se refieren a la forma en que una persona hace uso del movimiento de su cuerpo para comunicar cosas. Entre ellos se encuentran los gestos (como cuando una persona previene a quien le sigue al detectar un elemento peligroso en el ambiente, señalándolo con el dedo índice), la postura corporal (por ejemplo, cuando una persona es seguida no le da la espalda en forma total al otro, sino que mantiene cierto ángulo que le permita ver todo lo que hay enfrente pero también mantener contacto visual esporádico con su seguidor como una forma de demostrarle que le está prestando atención), la expresión facial (una persona puede expresar cansancio con su rostro como una forma de decirle al otro que no camine tan rápido), el énfasis de un movimiento (por ejemplo, si un guía se da cuenta que hay un borde en el suelo que puede provocar la caída de su seguidor, le dirá "Ten cuidado con este borde", y puede exagerar intencionalmente el movimiento de sus pies al brincar el obstáculo con el objetivo de resaltar la posición del mismo y asegurar que su compañero lo vea), etc.

- Comportamientos proxémicos, que se refieren a la forma en que una persona hace uso de su posición en el espacio con respecto a los demás para comunicar cosas. Por ejemplo, las personas regulan la distancia física con respecto a otras para indicar diferentes cosas, tomando determinada distancia dependiendo de la relación social (desconocidos, amigos, novios, etc.) que exista entre ellos. Una persona que sigue a otra, trata de mantener una distancia adecuada para no invadir el espacio personal de su compañero, al menos que exista una relación sentimental entre ellos, en cuyo caso irán demasiado juntos.
- Comportamientos cronémicos, que se refieren a la forma en que una persona hace uso del tiempo para comunicar cosas. Una persona que es seguida puede caminar muy rápido con el objetivo de indicar que es necesario llegar pronto. La frecuencia con la que un guía voltea a ver a su seguidor puede indicar el grado de interés hacia su compañero.
- Comportamientos hápticos, que se refieren a la forma en que una persona hace uso del contacto táctil con los demás para comunicar cosas. Por ejemplo, cuando alguien toca la espalda de la persona a la que sigue para llamar su atención y provocar que voltee a verlo. Otro ejemplo es cuando un guía tiene suficiente confianza con su seguidor y lo agarra de la mano mientras realizan la actividad de seguimiento.
- Comportamientos oculésicos, que se refieren a la forma en que una persona hace uso del contacto visual con los demás para comunicar cosas. Por ejemplo, si la persona desea darle un mensaje muy importante a quien lo está siguiendo, puede voltear por completo y encararlo para enfatizar la importancia de su mensaje.

Los ejemplos brindados para cada uno de los comportamientos anteriores son sólo uno de entre miles de posibilidades. Esto deja claro que para los humanos, seguir es una actividad mucho más compleja que simplemente estar viendo a la persona e ir dando pasos detrás de ellas. El robot seguidor que se implementa en este trabajo no sólo camina detrás, sino que intenta establecer una conversación con el humano que lo utiliza.

1.3 Delimitación del trabajo

Es claro que resulta muy difícil introducir todos los elementos de comunicación que se dan cuando dos humanos realizan la actividad de seguimiento, pero algunos de ellos se considerarán al momento de diseñar el sistema de seguimiento de personas para un robot móvil que resultará de este proyecto de investigación. Entre mayor diversidad de estos comportamientos se integren en el robot, la actividad de seguimiento será más natural.

En este trabajo el problema de seguir a una persona no se aborda desde la perspectiva de Visión Computacional, en donde el objetivo sería encontrar los mejores algoritmos para detectar, localizar o identificar personas en tiempo real. Comparar cuál es el mejor algoritmo o generar uno nuevo queda fuera del alcance de este proyecto.

De hecho, la actividad de seguimiento es en realidad independiente del sentido que un individuo utiliza para detectar la posición del otro. Sin utilizar el sentido de la vista, una persona puede seguir a otra utilizando el sentido del oído, guiándose con la voz de su compañero.

En este trabajo, el objetivo es implementar la actividad de seguimiento dando énfasis en que seguir es una conversación que involucra conductas comunicativas verbales y no verbales entre el guía y el seguidor.

Aún queda un gran problema por resolver. En la sección anterior se explicó la gran riqueza de conductas comunicativas involucradas cuando un humano sigue a otro. Ya que sería un trabajo muy arduo integrar todas ellas en el sistema, se debe establecer un criterio que permita definir cuáles de estos comportamientos serán integrados y cuáles no, es decir, cuáles de ellos podrá entender el robot cuando un humano los realice, o cuáles debe realizar el robot para comunicarse con el usuario. Para esto se utilizará como referencia a la prueba *Follow Me* del concurso *RoboCup@Home*[1].

RoboCup@Home es una competencia internacional cuyo objetivo es promover el desarrollo de tecnología de robots de servicio y asistencia para aplicaciones domésticas, y que fue creada como resultado del gran interés por parte de la comunidad académica y de la industria en el desarrollo de dichas tecnologías. Una de las pruebas de esta competencia se llama *Follow Me*. En ella, el robot debe seguir a una persona que acaba de conocer a través de una ruta no predefinida, es decir, el usuario puede caminar hacia cualquier parte. Además, en esta prueba, ocurren una serie de eventos que desafían la capacidad del robot para no perder a la persona que sigue. El robot debe ser capaz de:

- Expresar instrucciones y avisos al usuario mediante voz en determinados momentos de la prueba.
- Entender comandos del usuario (que se pueden dar con voz o mediante un gesto con los brazos) como “sígueme” o “detente”.
- Reconocer a la persona correcta aunque haya perdido temporalmente el contacto visual con ella.
- Reconocer a la persona correcta aunque esté muy alejada del robot.
- Reconocer a la persona correcta aunque existan más personas en el lugar.
- Navegar en un ambiente totalmente desconocido sin chocar con los objetos o las personas que se encuentran en él.

La ejecución de esta tarea implica la cooperación interactiva entre el usuario y el robot con el fin de completar la prueba. La descripción de *Follow Me* se utilizará como guía para el diseño del robot seguidor de este trabajo de investigación, el cual deberá ser capaz de resolver todos los retos involucrados.

[1] http://www.robocupathome.org

1.4 Metodología

El robot seguidor que se desea implementar es un sistema interactivo que involucra diversas modalidades de entrada y salida, como lenguaje hablado, visión computacional, acción de los motores, entre otras. Construir una aplicación de este tipo no es una labor sencilla porque representa un gran reto de integración y coordinación de distintas tecnologías encaminadas a resolver problemas específicos (por ejemplo, sintetizadores de voz, reconocedores de voz, algoritmos de reconocimiento de personas, etc.).

El primer paso es buscar una metodología y un marco de trabajo que permita la creación de sistemas interactivos multimodales. Una opción para realizarlo es el uso de modelos de diálogo y de una Arquitectura Cognitiva para Interacción Humano-Computadora Multimodal (Pineda, Meza y Salinas, 2010).

Para entender este marco de trabajo se debe partir del hecho de que cuando dos o más sujetos interactúan entre ellos para resolver una tarea, se está dando una conversación. Allen *et. al.* (2001) definen a un diálogo práctico como un tipo específico de conversación que está enfocado en completar una tarea concreta. También plantean la hipótesis de que "las competencias conversacionales requeridas para diálogos prácticos, si bien son complejas, son significativamente más simples de lograr que las competencias conversacionales humanas en general" (*Ibídem*: 3)[2].

Cuando una persona sigue a otra, y ambas tienen como objetivo completar la tarea (es decir, tanto el guía como el seguidor tienen como meta llegar juntos a un lugar específico) se establece un diálogo práctico entre ellos.

Un diálogo práctico sigue un protocolo construido a través de una secuencia de situaciones conversacionales. Tomando en cuenta el contexto, en cada una de estas situaciones conversacionales hay un conjunto limitado de expectativas de lo que puede pasar, y también un conjunto limitado de acciones a realizar en caso de que pase alguna de ellas (Pineda *et. al.*, 2010).

Por ejemplo, supongamos que en la actividad de seguir, la situación actual consiste en que el guía está caminando y su seguidor va detrás de él. Ambos conocen el contexto en el que se encuentran, es decir, saben lo que están haciendo y el objetivo que pretenden lograr. El seguidor está consciente de que el guía en cualquier momento le puede dar una instrucción relacionada con la tarea, por ejemplo, le podría decir "detente", "apresúrate", etc. Pero el seguidor no espera que el guía le diga "pásame la sal" o "realiza esa multiplicación", las cuales son instrucciones que están totalmente fuera del contexto de la situación planteada. Por lo tanto, en esta situación hay un conjunto limitado de expectativas que tiene el seguidor. Con respecto a las acciones que haría el seguidor, también están enmarcadas dentro del contexto, por ejemplo, si el guía le dice "detente",

[2] Todas las citas textuales presentadas en este trabajo y cuya fuente original estaba en inglés, fueron traducidas al español por Arturo Rodríguez García.

el seguidor dejará de caminar, pero esa misma instrucción en otro contexto podría significar dejar de hablar, dejar de pedalear una bicicleta o dejar de empujar una mesa.

Los modelos de diálogo son una metodología que aprovecha el contexto para especificar de una manera muy sencilla protocolos conversacionales en términos de situaciones, expectativas y acciones, y se abordarán con detalle en el Capítulo 3, en donde también se incluye una explicación acerca de una Arquitectura Cognitiva Orientada a la Interacción que permite la construcción de aplicaciones con interacción humano-computadora y que está centrada en el uso de modelos de diálogo.

Una de las grandes ventajas de este marco de trabajo es el hecho de que el modelo de la estructura de la tarea que se obtiene es una descripción totalmente funcional de la tarea que responde únicamente al *¿qué se debe hacer?*, y no está relacionada con cuestiones algorítmicas o implementacionales, que responden al *¿cómo se debe hacer?*. Esto permite que el diseño del sistema se divida en una serie de etapas, dando inicio con un análisis enfocado exclusivamente a la tarea por resolver, y posteriormente pensar en qué algoritmos y tecnologías se deben incorporar al sistema.

1.5 Objetivos y contribución del trabajo

El objetivo general que guía a este trabajo de investigación es implementar en un robot móvil la habilidad de seguir a una persona utilizando modelos de diálogo y la Arquitectura Cognitiva Orientada a la Interacción, de modo que el robot tenga los elementos necesarios para resolver con éxito la tarea *Follow Me* de la competencia *RoboCup@Home*.

Los objetivos específicos se listan a continuación:

- Resolver la tarea de seguimiento de una persona conceptualizándola como un problema de comunicación entre el robot seguidor y la persona.
- Identificar y analizar el protocolo conversacional que se da entre un robot seguidor y un humano al tratar de llevar a cabo el seguimiento de una persona.
- Diseñar modelos de diálogo para el protocolo conversacional anterior.
- Identificar las habilidades visuales, de lenguaje, motoras, etc., que debe poseer el robot seguidor para poder seguir a una persona.
- Buscar las herramientas tecnológicas y algoritmos necesarios que permitan brindar a un robot las habilidades anteriores.
- Integrar las herramientas tecnológicas y algoritmos anteriores en un sistema de seguimiento de personas sobre un robot móvil.
- Evaluar el funcionamiento del robot seguidor.
- Identificar los problemas más comunes del sistema en funcionamiento y tratar de resolverlos mediante la incorporación de rutinas de recuperación robustas en los modelos de diálogo.

El problema de implementar la habilidad de seguimiento en un robot móvil es un tema que ha sido abordado de muchas maneras, pero en la mayoría de la literatura se trata como si fuera un problema exclusivamente de visión computacional. La aportación general de este trabajo consiste en definir y modelar la actividad de seguimiento como un diálogo práctico entre el robot y el humano utilizando modelos de diálogo y una Arquitectura Cognitiva Orientada a la Interacción.

Las aportaciones específicas de este trabajo son:

- Proponer una taxonomía de robots seguidores basada en la dificultad de la tarea que realizan (Capítulo 2).
- Proponer un agente de visión para reconocimiento de personas y gestos cuyo funcionamiento está basado en casos, los cuales son determinados por las expectativas del sistema en determinada situación (Capítulo 5).
- Proponer formas de evaluar el funcionamiento del robot seguidor (Capítulo 6).
- Proponer el uso del sensor *Kinect* junto con las librerías de *OpenNI* como una opción para el reconocimiento de personas en robots seguidores y caracterizar su funcionamiento (Apéndice B).

1.6 Organización de la tesis

Después de este capítulo en el que se presentó al lector la problemática a resolver en el trabajo, mostrando la importancia de que los robots móviles puedan seguir a las personas y el gran reto que representa brindarles esa habilidad, el presente trabajo de investigación contiene los siguientes capítulos:

- Capítulo 2.- Su objetivo es mostrar los antecedentes, describiendo en forma detallada los aspectos relacionados con robots de servicio y la habilidad de seguimiento, incluyendo una descripción detallada de la tarea *Follow Me*.
- Capítulo 3.- Este capítulo está orientado a presentar el marco de trabajo que será utilizado para diseñar e implementar el sistema, describiendo a detalle qué son los modelos de diálogo y qué es la Arquitectura Cognitiva Orientada a la Interacción.
- Capítulo 4.- Analiza detalladamente la estructura de la tarea que debe resolver el robot seguidor, y se presentan sus correspondientes modelos de diálogo.
- Capítulo 5.- Se muestra la forma en que se integraron todas las habilidades (de visión, de lenguaje, de comportamiento motor, etc.) necesarias para que el robot pudiera resolver la tarea asignada.
- Capítulo 6.- Tiene como objetivo la evaluación del robot seguidor en funcionamiento al ser probado varias veces para identificar los puntos débiles del sistema y en los que es necesario mejorar los algoritmos y herramientas tecnológicas.
- Capítulo 7.- Se presentan las conclusiones y el trabajo futuro.

Capítulo 2

Robots de servicio y la prueba *Follow Me*

Desde sus orígenes, el desarrollo tecnológico ha estado orientado a mejorar la vida de los seres humanos mediante la creación de artefactos que satisfagan sus necesidades. Uno de los resultados de la tecnología es la automatización de procesos, que ha permitido que máquinas y sistemas computarizados sustituyan a los humanos en tareas repetitivas o que implican un esfuerzo físico desgastante. Un ejemplo claro de ello es el sector industrial, donde las máquinas han reemplazado el trabajo de millones de obreros en el mundo.

El interés por hacer llegar la automatización de las tareas hasta los entornos cotidianos de las personas, como el hogar o la oficina, se ha visto reflejado en el desarrollo de dispositivos tecnológicos que simplifican las labores rutinarias en estos espacios. Tareas específicas, como regar el césped o el encendido del calentador de agua, ya son realizadas en forma independiente por las máquinas utilizando los principios del control automático de procesos. Hoy en día, existe el deseo por delegar tareas cotidianas aún más complejas a las máquinas, y esto se ha manifestado en intentos por desarrollar robots que ayuden a los humanos en su vida diaria.

En la sección 2.1 se explica en forma breve el concepto de robot de servicio, así como la gran cantidad de aplicaciones que pueden tener. A continuación, en la sección 2.2 se introduce la descripción de la prueba *Follow Me*, que es la tarea que debe resolver el robot seguidor resultante de este proyecto. En la sección 2.3 se identifican las diversas habilidades que requiere el robot seguidor para poder cumplir con éxito la tarea. Finalmente, en la sección 2.4 se presenta el trabajo previo concerniente a robots seguidores.

2.1 Robots de servicio

El concepto de un hombre artificial fue descrito por primera vez en el folklore judío. Era llamado "El Golem" y hay muchas versiones de esta historia. Una de las más contadas tiene lugar en Praga durante el siglo XVI. La leyenda relata que el jefe rabino Judah Loew construyó un hombre artificial de arcilla con el objetivo de proteger a los judíos que eran perseguidos por las autoridades locales. El Golem adquiría vida cuando alguien escribía en su cuerpo el nombre de Dios o la palabra hebrea *emet*, que significa "verdad" o "realidad". Al igual que los robots modernos, el Golem no podía pensar por si mismo, pero podía llevar a cabo tareas de acuerdo a las instrucciones que le eran dadas (Freedman, 2011).

El término robot fue acuñado por el pintor, escritor y periodista Josef Čapek, para denotar a un trabajador artificial producido bioquímicamente y evocando a la palabra checa *robota*, que significa trabajo forzado o servidumbre (Ambros, 2010). Este término apareció por primera vez escrito en la obra *Rossums's Universal Robots (R.U.R.)* del dramaturgo checo Karel Čapek, hermano de Josef, y estrenada en 1921. Desde entonces, la palabra robot ha formado parte de la literatura global concerniente a máquinas inteligentes y autómatas (Angelo, 2007).

El diccionario de la Real Academia Española (2001), define a un robot como una "máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas sólo a personas".

De acuerdo con la United Nations Economic Commission for Europe y la International Federation of Robotics (2005: 21-30), los robots que existen en la actualidad se pueden agrupar en dos categorías:

a) Robot industrial manipulador.- Definido en la ISO 8373 como "un manipulador programable en tres o más ejes[3], de control automático, reprogramable y multipropósito, que puede ser fijo o móvil, para uso en aplicaciones automáticas industriales". Un ejemplo de este tipo de robots son los brazos mecánicos utilizados en el ensamble de automóviles.
b) Robot de servicio.- Ya que aún no existe una definición aceptada internacionalmente, la UNECE y la IFR han adoptado una definición preliminar. Un robot de servicio es "un robot que opera de forma semi o totalmente autónoma para realizar servicios útiles al bienestar de humanos y equipo, excluyendo operaciones de manufactura". De acuerdo al área de aplicación se clasifican en:
    i. Robots personales/domésticos.- que incluyen robots para tareas domésticas, de entretenimiento, de asistencia para minusválidos, de transporte personal, de seguridad y vigilancia del hogar, entre otros.
    ii. Robots de servicio profesional.- aquellos dedicados a aplicaciones de campo (agricultura, minería, robots espaciales, etc.), limpieza profesional (pisos,

[3] En robótica, un eje es la dirección usada para especificar el movimiento del robot. Puede ser en modo lineal o rotacional.

albercas, ventanas, etc.), inspección, construcción, demolición, ayuda médica, defensa, rescate, seguridad, de asistencia (en restaurantes, supermercados, hoteles, etc.), misiones bajo el agua, entre muchas otras.

En la actualidad, los robots de servicio son un tema de gran interés tanto en la comunidad científica como en la industria privada. No sólo se busca que los robots cumplan con la tarea asignada, sino que tengan un alto grado de interacción con los humanos que le rodean. Es en este contexto donde aparece *RoboCup@Home*, un concurso internacional que pretende impulsar la construcción de robots que puedan asistir a las personas y a quienes se les pueda encargar tareas como preparar el desayuno, barrer la oficina, vigilar la casa, cortar el césped, recibir a un invitado, etc.

RoboCup[4] es un proyecto internacional que busca promover la investigación en Inteligencia Artificial, Robótica y otros campos relacionados. La idea fue introducida en 1993 por Alan Mackworth y en julio de 1997 se celebró la primera competencia en Japón. Originalmente, *RoboCup* escogió el juego de futbol soccer como tema central de investigación, buscando como resultado la generación de innovaciones que pudieran ser aplicadas en la resolución de problemas sociales relevantes o en la industria. El objetivo final del proyecto consiste en poder desarrollar para el año 2050 un grupo de robots humanoides completamente autónomos que puedan ganarle al equipo humano de campeones mundiales en soccer (Xian-yi y De-shen, 2005). En la actualidad, el torneo *RoboCup* se realiza cada año y está dividido en los siguientes cuatro dominios de competencia: *RoboCup Soccer*, *RoboCup Rescue*, *RoboCup@Home* y *RoboCup Junior*.

RoboCup@Home fue introducido en el concurso de 2006 efectuado en Bremen, Alemania. El manual de reglas y regulaciones de *RoboCup@Home* (2011b: 1) señala que el objetivo de la competencia es "desarrollar tecnología de robots de servicio y asistencia con alta relevancia para aplicaciones domésticas futuras", poniendo especial atención en habilidades relacionadas con Interacción y Cooperación Humano-Robot, Navegación y Mapeo en ambientes dinámicos, Visión Computacional y Reconocimiento de Objetos bajo condiciones de luz natural, Manipulación de Objetos, Comportamientos Adaptativos, entre otras.

La competencia *RoboCup@Home* está dividida en tres etapas: Stage 1, Stage 2 y las finales. Cada stage consiste de un conjunto de pruebas en las que se evalúan las habilidades de los robots para desempeñar tareas domésticas. Al final de cada stage, los equipos que hayan acumulado más puntos pasan a la siguiente etapa, en donde tendrán que enfrentar pruebas con un mayor grado de dificultad.

El escenario donde se realizan las pruebas simula un ambiente de la vida cotidiana de los seres humanos, por ejemplo, una casa o un supermercado. La mayoría de las pruebas se llevan a cabo en una arena semejante a un hogar que consta de varios cuartos interconectados y amueblados (sala, cocina, recamara, etc.).

[4] http://www.robocup.org/

2.2 *Follow Me*

Una de las pruebas del Stage 1 de *RoboCup@Home 2011* es *Follow Me*, la cual es descrita en el libro de reglas de la competencia (*RoboCup@Home*, 2011b). En ella, el robot debe seguir a una persona desconocida a través de un ambiente dinámico. La prueba se debe desarrollar en un tiempo máximo de 8 minutos y se realiza en un lugar fuera de la arena de la competencia, por ejemplo, en un supermercado o en un corredor de las instalaciones donde se lleva a cabo el evento. El Comité Técnico de la competencia elige a una persona para probar el funcionamiento del robot, a la cual se denominará operador. Los miembros del equipo tienen 2 minutos antes de la prueba para hablar con el operador y explicarle la forma de dar instrucciones al robot, teniendo la oportunidad de brindarle una nota que contenga información acerca del proceso de calibración y de los comandos hablados o gestuales para interactuar con el robot.

Antes de que empiece la prueba, el operador debe estar alejado por lo menos tres metros del robot. La señal de inicio de la prueba es presionar el botón de inicio del robot. Cuando comienza la prueba, el operador se acerca al robot y le dice que lo siga. El robot tiene que anunciar cuando ha finalizado su calibración y empieza a seguir al operador. Durante la calibración, el robot le puede dar instrucciones al operador (por ejemplo, ponte enfrente de mi, date la vuelta, alza los brazos, dime tu nombre, etc.).

Durante el tiempo restante, el robot debe seguir al operador y superar una serie de acontecimientos antes de llegar a la línea de meta. El operador camina en forma lenta y regular dándole la espalda al robot y esperándolo si va demasiado lento, nunca regresa al menos que vaya a interactuar con el robot. El robot debe mantener una distancia de al menos un metro, excepto si el operador es quien regresa para interactuar. Hay cuatro checkpoints en la prueba, y en cada uno de ellos una acción específica debe ser ejecutada. Si el robot falla en una de ellas, la prueba termina automáticamente. El sistema de puntaje de la prueba se detalla en la Tabla 2.1. Los checkpoints se describen a continuación:

1. Oclusión temporal.- El operador se detiene. Luego una segunda persona pasa caminando lentamente entre el robot y el operador. Después de eso, el operador reanuda la caminata y el robot debe seguirlo.
2. Rastreando desde la distancia.- El operador le dice al robot que se detenga y espere. El robot se detiene en esa posición durante diez segundos, anunciando cuando el periodo de diez segundos da inicio. El operador camina y se aleja tres metros del robot en un modo en el que no haya objetos entre el robot y él. Cuando los diez segundos concluyen, el robot se debe acercar al operador y empezar a seguirlo de nuevo.
3. Reconociendo al usuario.- El operador le dice al robot que se detenga. El operador se mueve a algún lugar en el que el robot no pueda verlo. Después el operador y otra persona caminan hacia el robot. Los humanos se paran a un metro de distancia entre ellos, y cada uno debe estar a metro y medio enfrente del robot. El robot debe reconocer cual de los dos es el operador y seguirlo nuevamente.

4. Línea de meta.- Todas las partes del robot que tocan el suelo deben pasar la línea de meta.

Checkpoint	Nombre	Acción	Puntaje
1	Oclusión temporal	Seguir al operador después de la oclusión	+100
2	Rastreando a lo lejos	Entender el comando del usuario	+50
		Bono por usar un gesto como comando	+100
		Esperar un mínimo de 10 segundos y después buscar y seguir al operador	+100
3	Reconociendo al usuario	Entender el comando del usuario	+50
		Bono por usar un gesto como comando	+100
		Reconocer y seguir al operador después de que regresa	+200
4	Línea de meta	Cruzar la línea de meta	+100
	No tocar	Hacer los cuatro checkpoints sin haber tocado ningún objeto o humano en el escenario	+100
	Bono especial	Rendimiento destacado del robot	+100
	Penalización especial	No atender	-500
PUNTAJE MÁXIMO			+1100

Tabla 2.1 Hoja de puntuación de la prueba *Follow Me*

El bono especial por rendimiento destacado es otorgado por el Comité Técnico de la competencia si considera que el robot hizo algo más de lo que la prueba exigía para acumular puntos y mostró algo innovador. En caso de recibirlo, puede ser de hasta 10% del puntaje obtenido en la prueba.

La penalización por no atender ocurre si el robot no participa en la prueba y no se dio aviso de ello a los jueces al menos con quince minutos de anticipación.

Si se utilizará un gesto como comando, se deben tomar en cuenta las siguientes reglas:

- El equipo tiene derecho a definir el gesto a utilizar.
- El gesto no debe de involucrar más que el movimiento de ambos brazos. Ejemplo de gestos permitidos: gestos de apuntar o lenguaje por señas.
- El equipo tiene la obligación de explicarle al operador el comando gestual antes de empezar la prueba.

2.3 Habilidades que necesita un robot para resolver la tarea *Follow Me*

A partir de la descripción brindada en la sección anterior, es necesario identificar las habilidades que requiere un robot seguidor para poder ejecutar en forma completa la tarea. En esta sección se explicarán en forma general estas habilidades que se deberán integrar y coordinar dentro del sistema.

Reconocimiento automático del habla

Consiste en convertir una señal acústica (emitida por el humano al hablar) en una cadena de palabras (texto). Este es un problema que todavía está muy lejos de ser resuelto, considerando que se pretende hacer en forma automática una transcripción de lo que dice cualquier hablante en cualquier ambiente (Jurafsky y Martin, 2007). Esta habilidad es necesaria en el robot seguidor, ya que el guía le dirá instrucciones en forma hablada.

Síntesis de voz

La síntesis de voz, también conocida como *text-to-speech* o TTS, consiste en generar habla (ondas acústicas) a partir de una entrada de texto (*Ibídem*). El robot seguidor debe poseer esta habilidad para decirle al usuario diferentes cosas, como presentarse, darle instrucciones o indicarle las acciones que va a realizar.

Reconocimiento de personas

El reconocimiento de personas se descompone en tres subproblemas: la detección (¿hay una persona?), la localización (¿dónde está esa persona?) y la identificación (¿quién es esa persona?) (Cielniak y Duckett, 2004).

La detección de personas consiste en determinar si un objeto presente en la escena es o no es una persona. La localización de una persona consiste en determinar la ubicación de esa persona en determinado sistema de referencia. La identificación de una persona consiste en determinar si la persona que se está detectando es la misma que una persona específica que ya se había detectado con anterioridad, es decir, re-conocer a un conocido. Estas tres habilidades son necesarias para que el robot pueda resolver por completo la tarea *Follow Me*.

Tracking de personas

El tracking de una persona consiste en determinar la trayectoria de una persona dentro del ambiente. Hacer el tracking de una persona es muy difícil, incluso en ambientes en los que no haya mucha gente alrededor. Si la persona se sale del campo de visión del sensor y regresa después de cierto tiempo se requiere la habilidad de identificarla y posteriormente comenzar a hacer el tracking de nuevo (Bahadori, Iocchi, Leone, Nardi y Scozzafava, 2005).

El tracking de objetos es en general un problema en el que se presentan bastantes dificultades, que pueden ser causadas por movimientos abruptos del objeto, pérdidas en la información causadas por la proyección del mundo 3D en una imagen 2D, cambios en la forma de objetos de estructura no rígida o articulada (por ejemplo, la forma de una persona cambia constantemente mientras va caminando), oclusiones parciales o totales por otros objetos, cambios en la iluminación, entre muchos otros (Yilmaz, Javed y Shah, 2006). El tracking de una persona se complica cuando el sensor también está en movimiento, ya que el fondo cambia constantemente, como sucede en el caso de un robot seguidor.

Reconocimiento de gestos

Esta habilidad consiste en reconocer gestos específicos realizados por humanos, por ejemplo, gestos de apuntar, saludar ondeando las manos, hacer el gesto *thisbig* (un gesto paramétrico que consiste en utilizar ambas manos para indicar el tamaño de una cosa), hacer el gesto *dunno* (que consiste en formar una W con los brazos poniendo las palmas de ambas manos hacia arriba para expresar ignorancia acerca de algo), entre muchos otros (Bennewitz, Axenbeck, Behnke y Burgard, 2008).

Este es un problema de visión computacional que requiere la detección de personas y además la identificación y tracking de sus partes (brazos, cabeza, torso, etc.) para poder reconocer si está haciendo un gesto específico. En el caso del robot seguidor que resuelve la tarea *Follow Me*, éste debe ser capaz de reconocer gestos por parte del usuario para indicarle que se detenga.

Navegación en un ambiente dinámico

La navegación de robots móviles involucra varios problemas: la construcción de mapas (generar una representación interna del lugar a través del cual se desplazará el robot), la autolocalización (la capacidad del robot de determinar su ubicación actual con respecto a algún marco de referencia), la planeación de rutas (elegir el camino por el que el robot se desplazará para llegar a determinado lugar) y la evasión de obstáculos (evitar colisionar con obstáculos conocidos que ya estaban en el mapa, o con obstáculos desconocidos que aparecieron en el ambiente después de la generación del mapa o que están en movimiento) (Becker, Meirelles y Perdigão, 2006).

Para la prueba *Follow Me* el robot no cuenta con un mapa del lugar antes de que empiece la prueba y no necesita generar uno, pues en ningún momento se requiere regresar a un punto determinado por el que ya había pasado. Tampoco se necesita autolocalización. La planeación de rutas y la evasión de obstáculos si se deben considerar, ya que el robot no debe chocar con objetos que forman parte del escenario, incluyendo a las personas.

2.4 Trabajo previo en robots seguidores

Existe mucha información referente a proyectos relacionados con robots que tienen la habilidad de seguir a humanos. La literatura relacionada con robots seguidores está orientada en su mayoría al problema del reconocimiento de personas en tiempo real, proponiendo técnicas que involucran distintos tipos de sensores y algoritmos. El análisis de algunos aspectos de la interacción humano-robot al realizar la tarea de seguimiento también es un tema que puede ser encontrado, pero existe menor cantidad de información en comparación con la relacionada al problema de reconocimiento de personas. Cómo integrar la habilidad de seguimiento con otras habilidades del robot (por ejemplo, entender comandos hablados, comando gestuales, etc.), también es un tema del que hay poca información.

Esta sección se dividirá en tres partes. En la primera de ellas, se pretende mostrar un panorama acerca de los distintos tipos de robots seguidores existentes. Posteriormente, se abordarán los aspectos relacionados con las diversas técnicas utilizadas para el reconocimiento de personas, y finalmente se mostrarán algunos aspectos de la interacción humano-robot en la actividad de seguimiento que han sido objeto de estudio.

2.4.1 Clasificando a los robots seguidores

Ante la gran cantidad de robots seguidores que se han implementado, es necesario establecer un criterio para saber de qué tipo de robot seguidor estamos hablando. El gran problema es que existen algunos robots seguidores que son muy simples y otros que son demasiado complejos. Pongamos el caso de un robot seguidor muy sencillo: un robot seguidor de luz, el cual puede ser guiado por un humano utilizando una lámpara.

Un robot seguidor de luz puede construirse de una manera muy fácil, utilizando una base mecánica sencilla con dos motores, pilas, un circuito simple que consta de un par de foto resistencias y algunos transistores y resistencias extras. Lo único que hace este robot es avanzar hacia donde haya más luz. Este robot seguidor realiza una tarea sumamente sencilla en comparación con la tarea *Follow Me*.

En este trabajo se propone una clasificación de robots seguidores dependiendo de la dificultad de la tarea que deben resolver. Para ello se proponen una serie de preguntas cuyas respuestas pueden ser *sí* o *no*:

1. ¿El robot sigue a una persona que porta un señuelo?
2. ¿El robot conoce a la persona antes de que empiece la tarea?
3. ¿El robot conoce el mapa del lugar antes de que empiece la tarea?
4. ¿El robot realiza el seguimiento en un ambiente estático?
5. ¿El robot realiza únicamente la actividad de seguimiento?
6. ¿El robot realiza el seguimiento en un lugar donde no hay otras personas además del guía?
7. ¿El robot realiza el seguimiento en un lugar con iluminación controlada?
8. ¿El robot realiza la actividad de seguimiento sobre una superficie lisa con pocas irregularidades?

Con respecto a la primera pregunta, algunos robots seguidores requieren que el guía porte un señuelo, por ejemplo, un guante formado con un cuadriculado similar al de un tablero de ajedrez, con el objetivo de que un algoritmo de visión computacional busque en la escena ese patrón. Otro ejemplo de señuelo para un robot seguidor se da cuando el guía porta un transmisor que manda al ambiente señales de radio o infrarrojas y el robot cuenta con un sensor capaz de detectar la dirección actual del transmisor, y por lo tanto, puede seguir a la persona que lo porta (Gigliotta, Caretti, Shokur y Nolfi, 2003). Es claro que cuando un guía porta un buen señuelo, la tarea de seguimiento se simplifica.

La segunda pregunta se refiere a si el robot conoció a la persona fuera de línea, es decir, antes de que el robot comience a funcionar ya cuenta con información acerca de la persona que seguirá. Por ejemplo, el robot puede tener una serie de fotos del rostro de esa persona, o información sobre los colores de su playera. Tener esta información simplifica la tarea de seguimiento, pues tener una clara descripción de la persona a seguir facilita su identificación al momento de estar realizando el tracking. La tarea de seguimiento se complica cuando la persona a la que seguirá el robot es un usuario totalmente desconocido y debe aprender sus características en línea.

La tercera pregunta plantea si el robot cuenta con un mapa del lugar. Algunos robots seguidores basan su funcionamiento en el hecho de que poseen un mapa detallado del lugar en donde se pueden desplazar (por ejemplo, un mapa de la casa). Este mapa contiene información sobre ubicación de los objetos, lo cual ayuda enormemente a facilitar la navegación por el lugar. Sin embargo, la desventaja de usar un mapa es que el robot está condicionado a sólo poder funcionar correctamente en ese lugar.

La cuarta pregunta se enfoca a si el ambiente donde se desplazará el robot es fijo o no. Si es fijo, significa que durante la tarea de seguimiento no habrá objetos que cambien de lugar (por ejemplo, personas caminando en el ambiente). La tarea de seguimiento se complica cuando el ambiente es dinámico, y el robot debe estar alerta no sólo de la persona que sigue, sino también de los demás objetos que cambian de lugar para evitar chocar con ellos.

La quinta pregunta se refiere a si la única tarea que realiza el robot es seguir a una persona. Son pocas las fuentes de información que indican que aparte de estar caminando detrás del usuario, el robot está haciendo otra cosa, por ejemplo, analizar gestos del usuario, mantener una conversación con él, buscar objetos en forma paralela a que sigue al usuario, etc.

La sexta pregunta cuestiona sobre la cantidad de personas que habrá en el ambiente. No es lo mismo que un robot siga a una persona en un lugar donde sólo está el guía, a un robot que realiza la misma actividad en un pasillo de un supermercado por donde pasa mucha gente, lo que complica el reconocimiento de una persona específica.

La respuesta a la séptima pregunta depende del lugar donde se lleva a cabo la actividad. Realizar la actividad de seguimiento en un cuarto cerrado donde hay luz artificial controlada es mucho más fácil que hacerlo en el exterior, pues la luz solar puede ser un gran problema para muchos de los dispositivos sensores.

La octava pregunta se enfoca a la irregularidad del terreno sobre el que avanza el robot. Si el suelo es liso y hay bordes mínimos, como en el interior de una casa, la tarea resulta más sencilla que pedirle a un robot que siga a otra persona a través de un bosque, por ejemplo. Las irregularidades en el suelo provocan vibraciones en el robot que pueden generar problemas en el reconocimiento de personas.

Con esta serie de preguntas se puede notar claramente cómo la descripción de la tarea es muy importante al hablar de robots seguidores. Un robot seguidor puede ser tan sencillo o tan complicado como se desee, y eso depende de la tarea que vaya a desempeñar. En el caso de la tarea *Follow Me* se tiene una respuesta negativa a las primeras seis preguntas planteadas, como se puede ver en la Tabla 2.2. Entre mayor número de respuestas negativas se tengan, la tarea a resolver es más difícil.

Pregunta		Respuesta
1	¿El robot sigue a una persona que porta un señuelo?	No
2	¿El robot conoce a la persona antes de que empiece la tarea?	No
3	¿El robot conoce el mapa del lugar antes de que empiece la tarea?	No
4	¿El robot realiza el seguimiento en un ambiente estático?	No
5	¿El robot realiza únicamente la actividad de seguimiento?	No
6	¿El robot realiza el seguimiento en un lugar donde no hay otras personas además del guía?	No
7	¿El robot realiza el seguimiento en un lugar con iluminación controlada?	Sí
8	¿El robot realiza la actividad de seguimiento sobre una superficie lisa con pocas irregularidades?	Sí

Tabla 2.2 Determinación de la dificultad de la tarea *Follow Me* de *RoboCup@Home*

2.4.2 Reconocimiento de personas en robots móviles

El reconocimiento de personas es un tema de investigación que ha sido objeto de numerosos trabajos de investigación y existe una inmensa cantidad de información al respecto. En sistemas no móviles, existe una gran cantidad de técnicas para el reconocimiento de personas. La detección de personas se resuelve fácilmente por medio de la substracción del fondo. La localización de personas se realiza aplicando métodos de tracking, como Filtros de Kalman o el Algoritmo de Condensación, a la secuencia de imágenes del video. Para la identificación se utilizan medidas biométricas del rostro, la voz de la persona, la ropa que lleva puesta, entre otras. Sin embargo, en sistemas móviles, hay pocas técnicas que pueden ser aplicadas en forma exitosa debido al ruido adicional y la incertidumbre inherente al hecho del que el sistema opera en un ambiente complejo y dinámico cuyo fondo está cambiando constantemente (Cielniak y Duckett, 2004). Esta sección describirá únicamente algunas de las técnicas empleadas para el reconocimiento de personas que han sido utilizadas en robots móviles que realizan la tarea de seguir a una persona.

Cámara

Una de las técnicas más básicas en robots seguidores es utilizar una cámara como sensor y después utilizar un algoritmo para identificar a las personas. Schlegel, Illman, Jaberg, Schuster y Wörz (1998) utilizan en su robot una cámara para percibir el ambiente y posteriormente analizan la imagen mediante histogramas de color y modelos de contorno. Sidenbladh, Kragić y Christensen (1999) utilizan la información de color que provee una cámara para detectar la piel de un humano y poder localizar la cabeza, de modo que el robot debe moverse para lograr que la parte superior de la persona quede en el centro de la imagen captada por la cámara.

Sin embargo, cualquier método basado en cámara está propenso a las variaciones de color del background y a las condiciones de iluminación como resultado del cambio de posición del robot mientras se mueve. Si se están identificando rostros, se tiene como desventaja que el usuario debe ver al robot para que lo siga (Gockley, Forlizzi y Simmons, 2007).

Láser

Muchos robots seguidores utilizan como sensor uno o varios *Laser Range Finders*[5] y después utilizan un algoritmo de detección de piernas para detectar las extremidades inferiores de

[5] Un LRF es un dispositivo que utiliza un rayo láser para determinar la distancia a un objeto. El principio de su funcionamiento consiste en enviar un pulso y medir el tiempo que tarda en regresar después de ser reflejado por el objeto. En base a ese tiempo es posible calcular la distancia a la que se encuentra el objeto (Miller, Vandome y McBrewster, 2010).

un humano. Shaker, Saade y Asmar (2008) construyeron un robot seguidor combinando el uso de láser, el algoritmo de detección de piernas y un algoritmo de inferencia basado en reglas para compensar el ruido y los falsos negativos que ocurren al detectar las piernas del guía.

Utilizando láser, también es posible detectar personas mediante la técnica de mapas de diferencia. El robot debe aprender el mapa del lugar antes de que haya personas. Posteriormente, compara el mapa actual (en donde puede haber personas) con el mapa original, y calcula la diferencia entre ambos para verificar si hay algo nuevo. Un cambio significativo en el nuevo mapa puede corresponder a una persona (Montemerlo, Pineau, Roy, Thrun y Verma, 2002). La desventaja de esta técnica es que el robot debe recibir el mapa del lugar, o tener tiempo suficiente para explorar el lugar y realizar el mapa antes de que entren las personas. Además, el robot debe conocer su propia localización.

Otra opción es combinar el uso de láser y visión. Stückler *et. al.* (2011) proponen un robot de servicio que utiliza LRFs para detectar candidatos a personas, localizarlas y rastrearlas. Utilizando imágenes con cámara, se hace una verificación detectando rostros y la parte superior del cuerpo. Kleinehagenbrock *et. al.* (2002) y Belloto y Hu (2007) proponen una combinación similar de técnicas que consiste en detectar piernas con el láser, y detectar caras con la cámara. Kovilarov y Sukhatme (2006) combinan una cámara omnidireccional y un láser para que un robot pueda reconocer y seguir a personas en un lugar abierto. Zivkovic y Kröse (2008) utilizan en su robot láser para detección de piernas y cámaras para detectar torso y rostro.

Zheng y Meng (2009) construyeron un robot seguidor que utiliza láser, visión y además se auxilia de los sonares. Fritsch, Kleinehagenbrock, Lang, Fing y Sagerer (2004) diseñaron un robot seguidor que combina detección de piernas con láser, detección de rostro y torso por visión y detección de habla mediante micrófonos.

Cámaras estereográficas

Otra forma de reconocer personas es utilizar la información 3D que se puede obtener utilizando cámaras estereográficas. Beymer y Konolige (2001) proponen un robot seguidor que basa su funcionamiento en estos dispositivos, analizando la escena 3D para buscar la parte inferior de una persona (un par de cilindros verticales que se unen en la parte superior por una elipse). Kwon, Yoon, Byung y Kak (2005) implementaron un robot que también utiliza visión estéreo y proponen un algoritmo para seguir personas utilizando dos cámaras móviles independientes sin calibrar.

Mendez, Muñoz y Morales (2010) proponen la combinación de cámaras estereográficas con modelos probabilísticos de piel. Lo primero que hace el sistema es obtener una imagen de disparidad utilizando una cámara stereo, permitiendo calcular la distancia a la que se encuentran los objetos. En la imagen de disparidad se hace una segmentación de objetos, y a cada uno de ellos se les aplica un modelo de contorno semielíptico para verificar si la forma corresponde a la

de un humano. En adición, se emplea un modelo probabilístico para la detección de piel que fue obtenido mediante una máquina de aprendizaje, tomando como muestras regiones de imágenes correspondientes a piel de humanos de diferentes razas y edades. Muñoz-Salinas, Aguirre, García-Silvente y Gonzáles (2005) plantean otra combinación posible utilizando visión estéreo con reconocimiento de rostros.

Cámaras térmicas

Usar cámaras térmicas es otra opción para reconocer a una persona, y aprovecha el hecho de que los seres humanos tenemos niveles de temperatura distintos al de otros objetos del medio ambiente. Una de las ventajas es que los datos del sensor no dependen de las condiciones de luz y las personas pueden ser detectadas incluso en la oscuridad (Treptow, Cielniak y Duckett, 2006).

Cielniak y Duckett (2004) proponen un sistema de reconocimiento de personas para un robot móvil que combina información térmica y de color. En este sistema, se obtiene una imagen térmica a la cual se le aplican técnicas de análisis de imagen para segmentar a las personas. Esta información es usada para segmentar las regiones correspondientes en la imagen de color usando una transformación afín para resolver la correspondencia de la imagen entre las dos cámaras. Después de la segmentación, la región de la imagen que contiene a la persona es dividida en secciones que corresponden a la cabeza, el torso y los pies.

2.4.3 Interacción humano-robot y aplicaciones en la actividad de seguimiento

Existen algunas investigaciones acerca de la forma en que el robot seguidor maneja el espacio que le rodea. Una de las principales preocupaciones es el hecho de que el robot se mantenga a una distancia segura respecto al guía, y que la velocidad a la que se aproxime sea adecuada con el objetivo de no lastimar a nadie (Shaker *et. al.*, 2008). El camino que sigue un robot al seguir a una persona es otro tema de interés, pues se ha propuesto que los robots deben moverse de una manera que los humanos puedan predecir y entender, con el objetivo de que las personas tengan confianza hacia los robots, se sientan cómodos con su presencia y no vayan a ser lastimadas por algún choque durante la actividad de seguimiento (Gockley *et. al.*, 2007).

También se ha hablado de medidas de seguridad que deben tener en cuenta los robots seguidores, entre ellas: mantenerse cerca del guía para evitar que una tercera persona se cruce, contar con mecanismos para ser detenidos en caso de emergencia, aceptar órdenes del guía sólo si se encuentra a determinada distancia, tener señales luminosas para advertir cosas, entre otras (Dessimoz y Gauthey, 2010).

Una interesante aplicación es un robot enfermero que puede entablar con un humano la actividad de seguimiento, pero fungiendo el rol de guía. Se utilizó para asistir a personas de la tercera edad con limitaciones físicas y cognitivas. El robot sirve como acompañante para guiarlos a

un departamento del hospital en el que recibirían atención. Este robot interactúa con el usuario, platicándole del clima o recordándole su próxima cita médica (Montemerlo *et. al.*, 2002). Otra aplicación de la habilidad de seguimiento es un prototipo de robot que asiste a personas en el aeropuerto, cargando su equipaje y caminando detrás de ellos (Nuñez, García, Onetto, Alonzo y Tosunoglu, 2010).

Después de mostrar este panorama acerca de la investigación en torno a robots seguidores, se puede concluir que todavía existen muchas cosas que mejorar, tanto en algoritmos particulares para habilidades específicas (por ejemplo, el reconocimiento de personas) como en la integración del seguimiento con otras actividades que pueda realizar el robot. En el siguiente capítulo se expondrá el marco de trabajo desde el cual se atacará el problema del diseño e implementación de un robot seguidor capaz de ejecutar la prueba *Follow Me* en forma completa.

Capítulo 3

Modelos de Diálogo y Arquitectura Cognitiva Orientada a la Interacción

En el capítulo anterior se presentó la descripción de la tarea que se pretende resolver. En este momento es claro que se desea implementar en un robot móvil un sistema de seguimiento de personas que permita la interacción con el usuario mediante diversas modalidades de entrada y salida. También se identificaron las habilidades que requiere el robot para cumplir la tarea.

El problema ahora es cómo lograr integrar y coordinar todas estas capacidades del robot para tener como resultado un sistema capaz de ejecutar la tarea en forma completa. Lograr lo anterior es un verdadero reto, y se requiere una metodología adecuada que permita desarrollar sistemas interactivos multimodales.

En este capítulo se presenta un marco conceptual y un entorno de programación que permite "la especificación declarativa de sistemas interactivos complejos con entrada y salida multimodal, incluyendo lenguaje hablado, visión computacional y comportamiento de motor", y que se basa en el uso de Modelos de Diálogo y de una Arquitectura Cognitiva (Pineda *et. al.*, 2010: 20).

La sección 3.1 del capítulo introduce y formaliza la noción de Modelos de Diálogo. Posteriormente en la sección 3.2 se explica una Arquitectura Cognitiva Orientada a la Interacción que está íntimamente vinculada con la idea de Modelos de Diálogo. En la sección 3.3 se detalla la forma en que esta Arquitectura Cognitiva ha sido implementada computacionalmente utilizando Agentes Distribuidos. Por último, en la sección 3.4 se mencionan algunas aplicaciones construidas utilizando el marco de trabajo propuesto en este capítulo.

3.1 Modelos de diálogo

"La interacción humano-computadora sigue protocolos que involucran la expresión e interpretación de intenciones, y la ejecución de acciones que satisfacen esas intenciones" (Pineda, 2008: 1). Una muestra de ello se da en la prueba *Follow Me*, cuando el usuario le dice al robot que se detenga. El usuario expresa la intención haciendo un gesto y el robot debe interpretar lo que su interlocutor le está tratando de decir para después ejecutar la acción de detener el movimiento de sus motores.

El principio central del enfoque descrito en este capítulo es que "la interpretación de las expresiones que representan intenciones en diálogos multimodal, así como las acciones realizadas por los agentes como consecuencia de entender dichas intenciones, son procesos dependientes del contexto". Las conversaciones en las que se busca resolver una tarea "siguen un protocolo esquemático que puede ser construido como secuencias de situaciones conversacionales". "Dentro del contexto de dichos protocolos sólo un número muy limitado de intenciones son significativas en relación al contexto en una situación dada, y sólo un pequeño conjunto de acciones son relevantes para lograr los objetivos de la tarea en esa situación particular" (*Ibídem*: 2). En los siguientes párrafos se formalizará esta metodología presentando las ideas y definiciones más importantes.

"Un modelo de diálogo es una especificación abstracta de un protocolo conversacional" y "es definido en términos de situaciones conversacionales". "Una situación es una abstracción del estado informacional del sistema definido en términos de las expectativas del sistema, las acciones que necesitan ser realizadas cuando una de esas expectativas es satisfecha, y la situación a la que se llega cuando la correspondiente acción es realizada" (Pineda *et. al.*, 2010: 21-22).

Por ejemplo, al inicio de la competencia *Follow Me*, el robot está en la situación "esperando la indicación de seguir a una persona". Dado el contexto, el robot tiene como expectativa únicamente recibir una instrucción del operador indicándole que le siga. Que el usuario le de al robot la instrucción de detenerse, no sería una expectativa en esta situación dado que estaría fuera de contexto en esta etapa inicial de la prueba. Cuando la expectativa es satisfecha, entonces el robot realiza una serie de acciones (en este caso, decirle al operador "párate enfrente de mi para poder verte") y pasa a la nueva situación "esperando a identificar una persona".

"Las expectativas pueden ser intencionales (por ejemplo expresadas por el interlocutor) o naturales (por ejemplo eventos que pasan en el mundo). En caso de que la información de entrada no pueda ser interpretada como una expectativa, un modelo de diálogo de recuperación puede ser invocado [...] situando al agente computacional en la conversación de nuevo" (*Ibídem*: 21).

"La especificación de expectativas y acciones es abstracta en relación a la modalidad en la cual una intención es expresada". Por ejemplo, en la tarea *Follow Me*, cuando el robot se encuentra en la situación "siguiendo a la persona", una de las expectativas es que el operador le

de la instrucción de detenerse. Esta expectativa puede ser expresada por el usuario ya sea en lenguaje hablado o mediante un gesto con los brazos (*Ibídem*: 21).

La especificación de expectativas y acciones también es abstracta en relación a la expresión concreta o mensaje de entrada. En la misma situación del ejemplo mostrado en el párrafo anterior, el usuario puede indicarle al robot que se detenga mediante expresiones como "detente", "espera un momento", "no te muevas", entre otras (*Ibídem*: 21).

"Los modelos de diálogo tienen una representación gráfica donde las situaciones son representan mediante nodos y las relaciones entre las situaciones son representadas mediante arcos dirigidos. Cada arco tiene una etiqueta de la forma α:β, donde α corresponde a la expectativa (ya sea intencional o natural) y β corresponde a la acción compleja realizada por el sistema cuando α es satisfecha en el mundo en la situación actual s_i. Como resultado de realizar esa acción, el sistema se mueve a la situación s_j que es señalada por el arco" (*Ibídem*: 22). Lo anterior se muestra gráficamente en la siguiente figura.

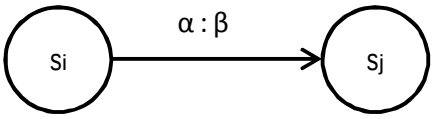

Figura 3.1 Representación gráfica de Modelos de Diálogo

Las expectativas pueden ser vacías o concretas. Si una situación s_i tiene una expectativa vacía, denotada con el símbolo ε, debe tener un único arco de salida, por lo que la correspondiente acción se ejecutará de manera determinista. Las expectativas concretas son intenciones esperadas que pueden ser representadas por una etiqueta constante o por un predicado aterrizado (*Ibídem*: 22).

Las acciones también pueden ser vacías o concretas. Una acción vacía implica que al cumplirse la expectativa se pasará a la situación correspondiente sin realizar acción alguna. Las acciones concretas pueden ser representadas por etiquetas constantes o predicados aterrizados. Una acción compleja β que se realiza cuando se cumple una determinada expectativa está constituida de una secuencia de acciones básicas y se le llama estructura retórica, siguiendo las ideas de la *Rethorical Structure Theory* (RST)[6] de Mann y Thompson (1988). Las acciones básicas

[6] La RST fue concebida originalmente como un modo de describir a un texto, identificando su estructura jerárquica y describiendo las relaciones entre partes del texto en términos funcionales. Una relación se sostiene entre dos partes del texto que no se sobreponen, a una se le llama núcleo y a la otra satélite. Un ejemplo de relación es la Evidencia, que se puede explicar de la siguiente manera: un satélite de Evidencia tiene como intención incrementar la confianza del lector en el núcleo (Mann y Thompson, 1988). En la actualidad esta teoría se ha aplicado con éxito a otras áreas de la ciencia como Lingüística Computacional, Análisis del Discurso, Análisis de Diálogos, entre otros (Taboada y Mann, 2005).

que conforman un acto retórico se pueden presentar en cualquiera de las modalidades de salida del sistema. Estas acciones pueden tener una representación externa (una acción del motor o generar un mensaje mediante el sintetizador de voz), o pueden ser internas (como razonar, hacer un cálculo o resolver un problema) (*Ibídem*: 22-23).

También es posible acceder a la historia de la interacción. "Cuando los arcos de una gráfica han sido recorridos, todas las expresiones representando expectativas y acciones quedan aterrizadas. Por consiguiente, la secuencia de situaciones con las etiquetas de los arcos recorridos en una interacción particular constituye el discurso o contexto anafórico[7] de esa interacción". Esta información se encuentra "disponible para la resolución de dependencias anafóricas que dependan del contexto del discurso" (*Ibídem*: 23).

Además, "las expectativas y las acciones pueden ser especificadas por predicados que incluyan variables libres o funciones proposicionales". A las variables libres se les asigna un valor del mundo a través de la percepción. Por ejemplo, en la figura 3.2 se muestra un modelo de diálogo donde la expectativa y la acción son funciones proposicionales. En este caso, en la situación s_i, se espera que el usuario diga su nombre. Una vez que el usuario proporciona su nombre (puede ser mediante lenguaje hablado diciendo "Me llamo David" o "Soy David") se asigna el valor entregado por el usuario a la variable (en este caso a la variable *x* se le asigna David). Finalmente, al ejecutarse la acción, el sistema ya cuenta con el valor de *x* y puede generar un mensaje como "Hola David, gusto en conocerte" (*Ibídem*: 23).

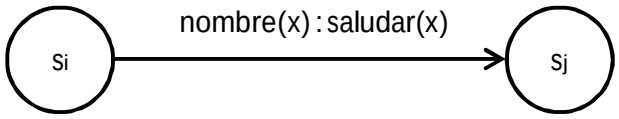

Figura 3.2 Ejemplo de una especificación a través de funciones proposicionales

"Las situaciones también son objetos paramétricos, de modo que los argumentos de una situación se pueden enlazar con los valores tomados por los argumentos de las expectativas y las acciones. Este es un mecanismo adicional que permite el flujo de información a través de los arcos de la gráfica a lo largo de la interacción" (*Ibídem*: 23).

Las acciones, expectativas y situaciones siguientes se pueden modelar en términos de funciones explícitas, como se ilustra en la figura 3.3. Estas funciones tienen como principal argumento la historia de la interacción. Cuando se llega a una situación que tiene una especificación funcional, lo primero que se hace es evaluar la función y después los arcos son

[7] La anáfora es un mecanismo mediante el cual un elemento del discurso remite a otro que apareció con anterioridad. Por ejemplo, si alguien dice: "El joven traía un periódico. Lo estuvo leyendo mientras llegaba el autobús". En este caso, *lo* es una referencia anafórica a *un periódico*.

interpretados utilizando los correspondientes valores de la función. La notación para la definición funcional de la situación siguiente se muestra en la figura 3.3, donde el punto *h* representa a la función, y sus posibles valores se representan con arcos punteados (*Ibídem*: 23-24).

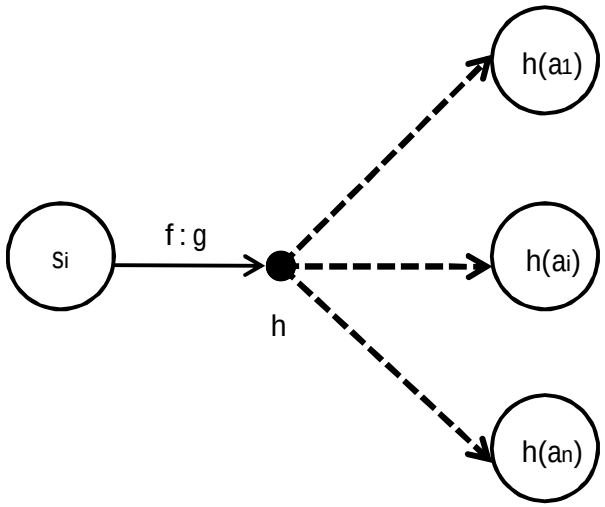

Figura 3.3 Representación funcional de expectativas, acciones y transiciones

Existen tipos de situaciones de acuerdo a la modalidad perceptual a través de la cual la información de entrada es reconocida e interpretada. Los tipos básicos de modalidades de entrada son "escuchar" y "ver", aunque se pueden incluir otras modalidades que reciban información del láser, de botones, etc. También es posible especificar situaciones multimodales, como una situación de "escuchar y ver" (*Ibídem*: 25).

Además existen situaciones del tipo recursivo, que corresponden a un modelo de diálogo completo. "Cuando se llega a una situación de este tipo, el modelo de diálogo actual se mete en una pila, y el modelo de diálogo embebido se carga para su interpretación desde su estado inicial. [...] Cuando un modelo de diálogo llega a una situación final, el modelo de diálogo que está en el tope de la pila, si lo hay, se saca de la pila y se continúa con la interpretación de la situación que sigue a la situación recursiva actual. Cada arco de salida de una situación recursiva tiene una etiqueta de continuación [...] en el lugar que corresponde a la expectativa α [...] Las situaciones finales tienen un único arco de salida con una etiqueta de continuación que indica que el modelo de diálogo ha terminado en esa situación particular y no en otra". Este mecanismo computacional que incluye la posibilidad de situaciones recursivas es llamado *Functional Recursive Transition Networks* (*F-RTN*) (*Ibídem*: 25).

3.2 Arquitectura Cognitiva Orientada a la Interacción

Un enfoque fundamental en ciencias cognitivas es el hecho de que un sistema inteligente no es completamente homogéneo, sino que consiste de un conjunto de subsistemas funcionales, o módulos, que cooperan para lograr el procesamiento de información y el comportamiento inteligente (Stillings *et. al.*, 1995: 16). La arquitectura cognitiva se refiere a la forma en que el sistema está construido en término de esos módulos.

"Una arquitectura cognitiva es un sistema que integra percepción, pensamiento y acción, donde el conocimiento específico de la tarea y el dominio pueden variar, pero las estructuras computacionales y los procesos permanecen constantes" (Pineda *et. al.*, 2011: 2).

La investigación en arquitecturas cognitivas es interdisciplinaria, abarcando las áreas de Inteligencia Artificial, Psicología Cognitiva y Neurobiología. En las últimas décadas, muchas arquitecturas cognitivas se han propuesto, basadas en diferentes enfoques y metodologías, y abordando de formas diversas los componentes cognitivos (percepción, memoria, resolución de problemas, razonamiento, aprendizaje, etc.). El objetivo final de las arquitecturas cognitivas, es la construcción de sistemas coherentes que exhiban de manera robusta un amplio rango de funciones en diferentes conjuntos de problemas (Chong, Tan y Ng, 2007).

En los siguientes párrafos se describirá una Arquitectura Cognitiva Orientada a la Interacción, que está centrada en la idea de modelos de diálogo, los cuales permiten modelar la interacción entre el agente computacional y el mundo. En la figura 3.4 se muestra esta arquitectura. El ciclo de interacción principal involucra a los siguientes módulos: Reconocimiento → Interpretación → Modelos de diálogo → Especificación → Renderización.

La arquitectura incorpora una memoria semántica y una memoria perceptual (esta separación es parecida a la distinción entre memoria semántica y memoria episódica que se utiliza en Psicología cognitiva y Neuropsicología[8]). La memoria semántica almacena los conceptos usados en un dominio y tarea particular. El conocimiento almacenado es proposicional y es independiente de la modalidad. En la implementación actual se utilizan cláusulas de Prolog como representación lógica. Por otra parte, la memoria perceptual almacena asociaciones entre imágenes internas o perceptos específicos de una modalidad, y sus correspondientes interpretaciones o significados (Pineda *et. al.*, 2011: 2).

[8] Tulving (1972) hace una distinción entre memoria episódica y memoria semántica. La memoria episódica almacena información acerca de eventos temporales y la relación espacio-temporal entre ellos. Un evento perceptual puede ser almacenado en el sistema episódico únicamente en términos de sus propiedades perceptibles, y siempre es almacenado en términos de su referencia autobiográfica con respecto al resto del contenido en el sistema episódico. Por otra parte, la memoria semántica es la memoria necesaria para el uso del lenguaje, y es un tesauro mental que organiza el conocimiento que una persona posee acerca de las palabras y otros símbolos verbales, sus significados y referentes, las relaciones entre ellos, y acerca de las reglas, fórmulas y algoritmos para la manipulación de esos símbolos, conceptos y relaciones. La memoria semántica no registra propiedades perceptibles de las entradas, sino más bien almacena referentes cognitivos de las señales de entrada. El contenido de la memoria semántica es independiente de los parámetros espacio-temporales.

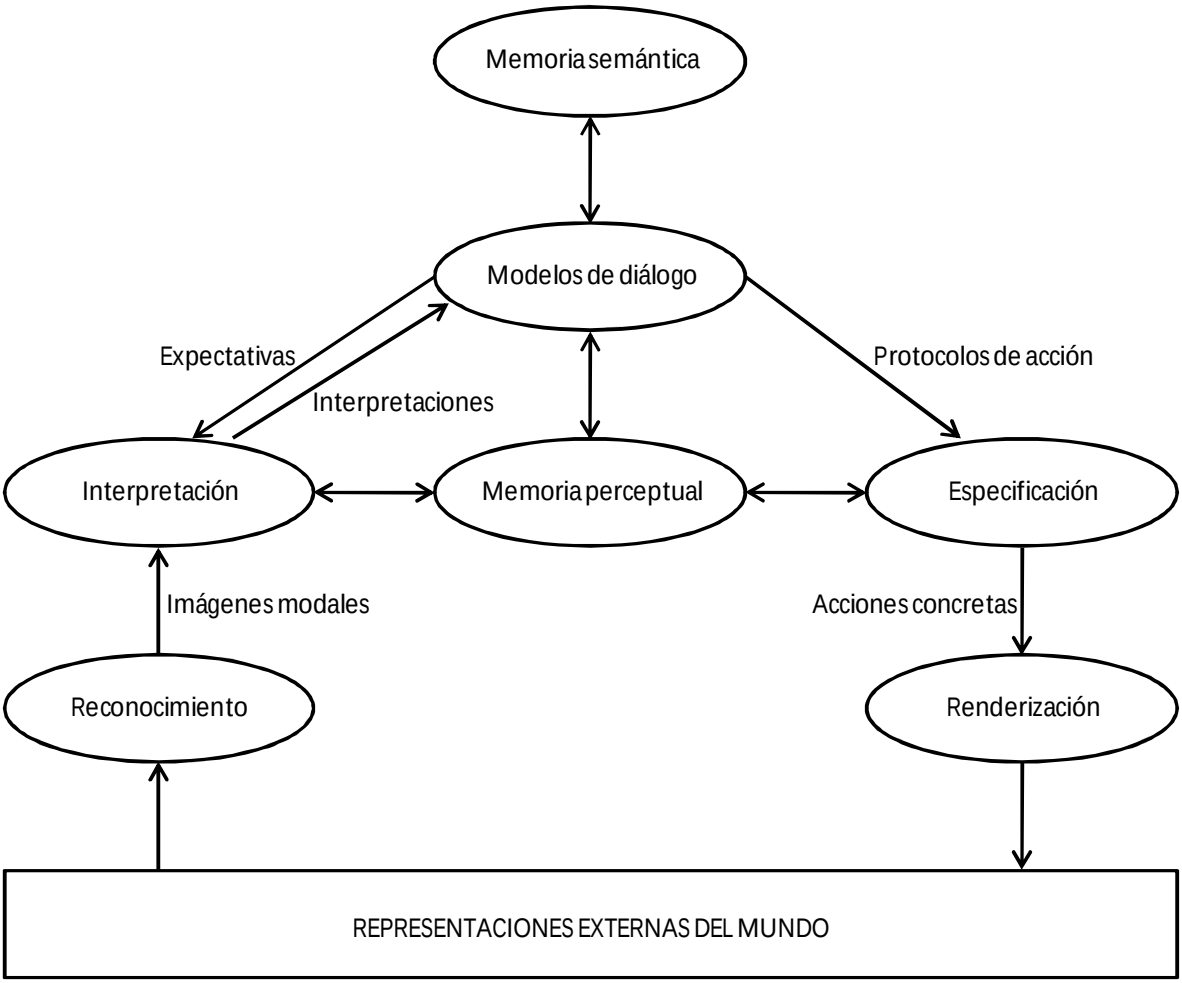

Figura 3.4 Arquitectura Cognitiva Orientada a la Interacción (IOCA)

El reconocimiento es el proceso que transforma la información de entrada proveniente del mundo externo en una imagen. Una imagen es un conjunto de información codificada en un formato particular y que almacena las características sensibles del estímulo externo que se le presenta al sistema. La modalidad es el formato en que está almacenada la información de una imagen (*Ibídem*: 2). Por ejemplo, la presencia de una persona puede ser detectada a través del canal visual, pero la imagen correspondiente a esa persona puede estar almacenada en diferentes formatos, pudiendo ser un conjunto de vectores SIFT[9] que describen los puntos clave de la persona, un conjunto de histogramas con información referente al color en determinadas regiones de la fotografía de la persona, o cualquier otro conjunto de descriptores obtenidos por alguna técnica de visión computacional.

"Las imágenes internas codifican el correspondiente patrón externo en forma independiente de su significado". Además, no es necesario que las imágenes almacenen toda la información del estímulo externo, sino que pueden ser estructuras con la menor información posible (*Ibídem*: 3). Por ejemplo, en el caso del reconocimiento de personas, no es necesario almacenar toda la información referente a texturas, color, bordes, etc., sino que basta con información de sus puntos más importantes, como es el caso de los descriptores SIFT.

"Además, los patrones representados a través de imágenes internas pueden ser dinámicos y evolucionar en el tiempo y en el espacio, por ejemplo, el patrón visual de un gesto físico, como «alto», puede ser codificado como un Modelo Oculto de Markov" (*Ibídem*: 3).

La interpretación consiste en asignar significados a las imágenes internas. Este es un proceso dependiente del contexto que toma en cuenta las expectativas del sistema presentes en la situación y que están especificadas en los modelos de diálogo. Los significados de las imágenes internas se representan en formato proposicional, en forma independiente de la modalidad. El proceso de la interpretación utiliza a la memoria perceptual y "realiza un emparejamiento cualitativo entre las imágenes obtenidas por los dispositivos de reconocimiento y las imágenes en la memoria perceptual, las cuales están almacenadas en la misma modalidad" (*Ibídem*: 3).

Después de la interpretación, los Modelos de Diálogo son el siguiente paso en el ciclo principal de la arquitectura. Asociados a ellos se encuentra el Manejador de Diálogo, que permite interpretar la especificación de la estructura de la tarea (que se encuentra en los respectivos modelos de diálogo), y relaciona las expectativas y las interpretaciones con las respectivas

[9] SIFT (Scale-Invariant Feature Transform) es un algoritmo que permite extraer características invariantes distintivas de una imagen y que pueden ser usadas para realizar emparejamientos entre distintas vistas de un objeto o escena. Estas características son invariantes a la escala y rotación de la imagen, y además son robustas ante distorsiones afines, cambios en el punto de vista 3D, adición de ruido y cambios en la iluminación. Las características obtenidas por SIFT tienen varias propiedades en común con las respuestas de las neuronas del cortex temporal inferior que son usadas por primates para el reconocimiento visual de objetos. Las características se detectan mediante un filtrado por etapas que identifica los puntos estables en el espacio-escala. Los puntos clave de la imagen representan gradientes de la imagen borrosa en múltiples planos de orientación y en múltiples escalas. Después de aplicar el algoritmo a una imagen, se obtiene como resultado un conjunto de *keypoints*, cada uno de los cuales es un vector de 128 elementos, que corresponden a un arreglo de 4 x 4 histogramas de 8 barras cada uno (Lowe, 1999), (Lowe, 2004).

acciones. En este nivel, las interpretaciones y las acciones se encuentran en un formato proposicional que es independiente de las modalidades de entrada y salida. Finalmente, en el módulo de especificación los protocolos de acción provenientes de los modelos de diálogo se renderizan en los dispositivos de salida del sistema (*Ibídem*: 3-4).

Para ejemplificar el ciclo de interacción principal de la arquitectura se considerará la siguiente situación. Un robot móvil ha pedido al usuario que se pare enfrente de él para poder verlo con el propósito de empezar a seguirlo. El usuario se pone enfrente del robot, y a continuación se describe lo que sucede en cada uno de los módulos de la arquitectura (ver la Figura 3.5 para complementar la explicación):

- Reconocimiento.- en este caso, el robot utilizará como dispositivos de reconocimiento una cámara y un láser. Aplicará un algoritmo de segmentación de personas basado en la sustracción del fondo. En caso de detectar a una persona, se aplicará el algoritmo SIFT sobre el segmento correspondiente a la persona, y almacenará la información de los *n* puntos característicos en la matriz P_1. Esta matriz está formada por *n* renglones, cada uno de los cuales contiene los 128 elementos de uno de los vectores SIFT. Esta matriz es la imagen interna.
- Interpretación.- en esta situación, desde el modelo de diálogo se indica al módulo de interpretación que la expectativa es ver a una persona. Esto se hace mediante la función proposicional *persona(x)*. Una vez que en el módulo de interpretación se tiene una matriz correspondiente a una persona, entonces se avisa al modelo de diálogo que la expectativa se ha satisfecho. En este caso, se envía la interpretación *persona(p1)*, donde la variable *x* ha dejado de ser libre y ahora tiene el valor *p1*, el cual sirve como un identificador para la matriz P_1 que describe a una persona específica, es decir, la que fue reconocida en ese momento. La imagen P_1 se puede dejar almacenada en la memoria perceptual para su uso futuro. Quizá en una situación posterior sea necesario verificar si el robot ya conocía a una persona *m*, y para ello hacer una comparación entre la imagen P_m y las imágenes P_1, P_2, … , P_{m-1} almacenadas en la memoria perceptual, y utilizando un algoritmo de emparejamiento ver si la imagen P_m corresponde a una de ellas.
- Modelos de diálogo.- una vez que el manejador de diálogo recibe la expectativa *persona(p1)*, envía la acción *seguir(p1)* al módulo de especificación, y se pasa a la siguiente situación.
- Especificación.- la acción *seguir(p1)* se debe especificar en forma total antes de enviarla al siguiente módulo. Esta acción implica enviar un mensaje al usuario indicándole que ya lo reconoció y que puede empezar a caminar. Posteriormente el robot debe dar inicio con el seguimiento de la persona, lo que implica calcular su posición, para ello puede hacer uso de la imagen P_1 almacenada en la memoria perceptual. Una vez que ha calculado la distancia y el ángulo, se calcula el movimiento rotacional y traslacional necesario para que el robot se ubique a una distancia correcta

del usuario. El resultado de esta especificación son las acciones concretas que el sistema debe realizar y que se envían al siguiente módulo.

- Renderización.- los dispositivos de salida ejecutan las acciones que se les indican, en este caso el sintetizador de voz del robot produce el mensaje "Perfecto, ya te vi. Ahora puedes empezar a caminar y yo te seguiré", mientras que los motores realizan los movimientos necesarios en forma coordinada para girar y avanzar.

En este momento surge una pregunta, ¿es necesario que todas las acciones del sistema siempre sean resultado de recorrer todo el ciclo de interacción principal de la arquitectura? Existen situaciones en las que no es necesario hacerlo, sino que basta con una respuesta reactiva del sistema. Imaginemos el caso de un robot que está siguiendo a una persona, dado que se encuentra en un ambiente dinámico es posible que un obstáculo inesperado surja de repente, por ejemplo que una segunda persona aparezca y quede demasiado cerca del robot. El sistema debe reaccionar en forma rápida para evitar colisionarse con la segunda persona, pero no es necesario que reconozca que es una persona ni que haga una interpretación, sino que basta con que se detenga o modifique su trayectoria para no chocar. Este ejemplo muestra la necesidad de que en la arquitectura existan Sistemas Reactivos Autónomos (ARSs por sus siglas en inglés).

El uso de ARSs permite lidiar con eventos inesperados del mundo o interacciones espontáneas de los interlocutores, los cuales no están contemplados en el modelo de diálogo. Los ARSs relacionan en forma directa a la información proveniente de los dispositivos de reconocimiento con los dispositivos de renderización. Utilizar ARSs requiere de un nuevo módulo llamado Coordinador. Como su nombre lo indica, este módulo coordina a los Modelos de Diálogo con los ARSs (*Ibídem*: 8-9). En la Figura 3.6 se muestra una Arquitectura Cognitiva Orientada a la Interacción que incluye ARSs.

El modelado de comportamientos reactivos en robots no es una tarea sencilla. Laureano-Cruces, De Arriaga-Gómez y Sánchez (2001) proponen una metodología para la creación de arquitecturas multiagentes reactivas mediante un análisis cognitivo de la tarea a resolver, y que da inicio con el análisis del comportamiento de una persona al ejecutar la tarea, incluyendo las acciones, decisiones, estructuras y procesos necesarios para completarla. El resultado de dicho análisis se transfiere al robot que emulará el comportamiento del humano.

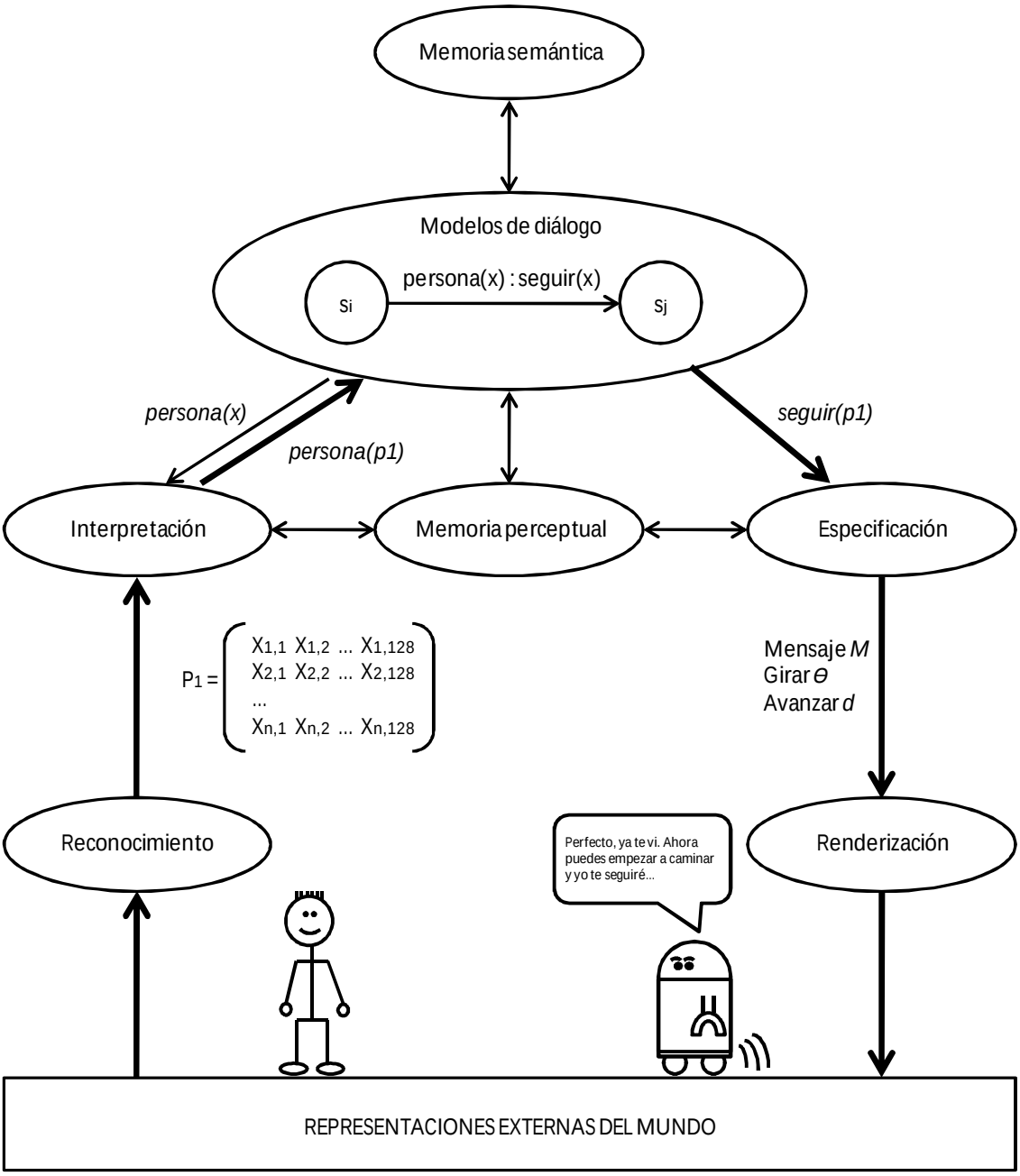

Figura 3.5 Ejemplo del ciclo de interacción principal de IOCA

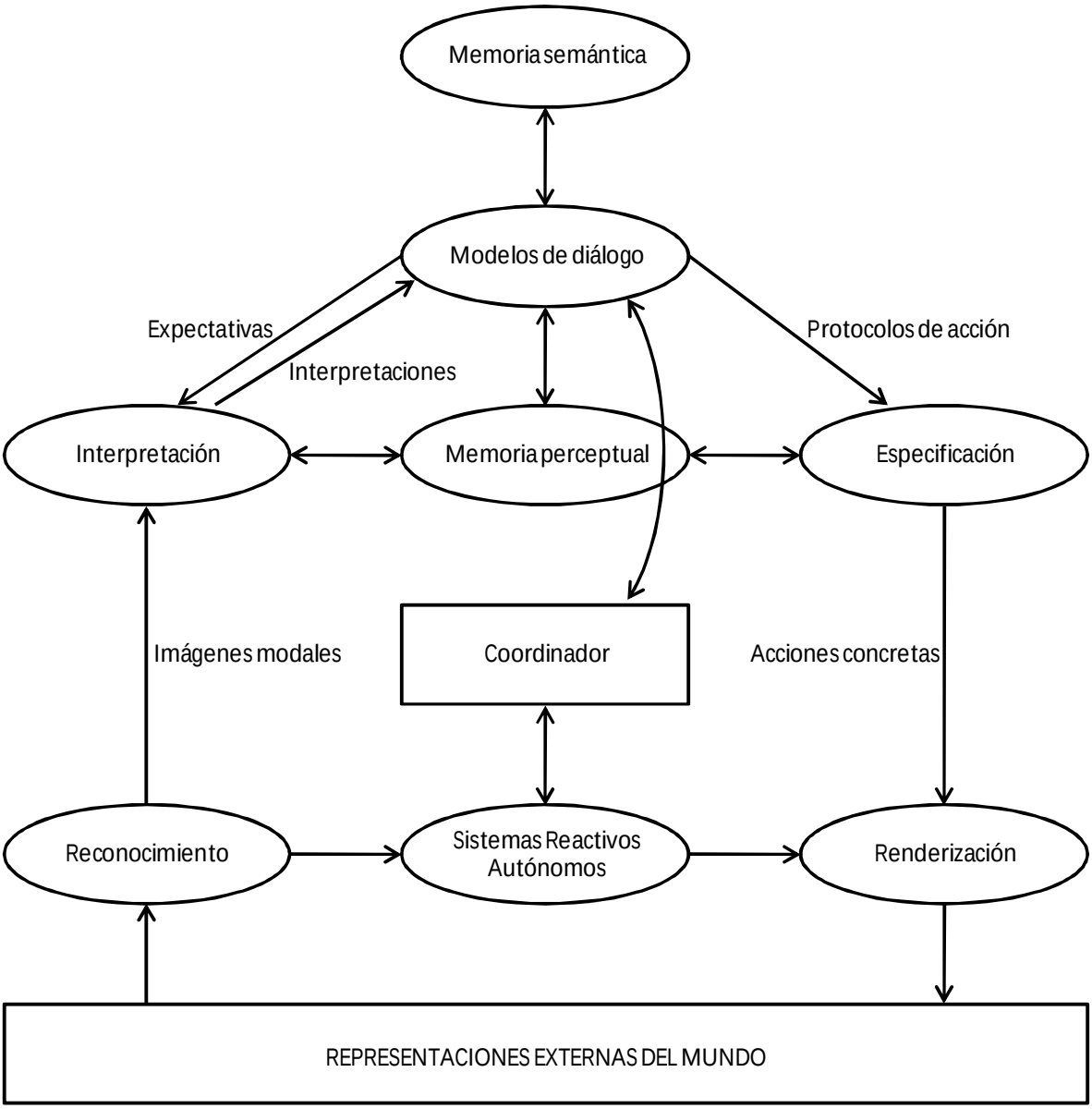

Figura 3.6 IOCA con capacidades reactivas

3.3 Implementación computacional de IOCA

Para la implementación computacional se utiliza Open Agent Architecture (OAA)[10], que es un marco de trabajo para la construcción de sistemas multiagentes. En el contexto de OAA, un agente es un proceso que tiene un conjunto de servicios que puede proveer. Un sistema construido en OAA es un conjunto de agentes, conceptualizados como una comunidad dinámica, donde cada uno de los agentes contribuye con sus servicios a la comunidad.

Cuando uno de los agentes necesita un servicio, en vez de invocar una subrutina específica de otro agente (como sucedería en la Programación Orientada a Objetos), el agente envía una expresión de alto nivel para indicar lo que necesita. Esta petición se encuentra en un lenguaje específico llamado ICL (Interagent Communication Language) y llega al Facilitador, que es un agente especializado cuya función es verificar qué agentes de la comunidad están disponibles y son capaces de resolver la petición, y posteriormente administrar la interacción entre los agentes para resolverla (Martin, Cheyer y Moran, 1999).

El uso de OAA tiene varias ventajas, entre ellas el hecho de que los procesos que forman parte del sistema pueden estar definidos en diferentes lenguajes de programación (como Prolog, Java, C, C++, entre otros), y estar en diferentes sistemas operativos (Linux, Unix, Windows, Solaris, etc.). Además, los agentes pueden residir en diferentes computadoras, permitiendo la creación de sistemas multiagentes distribuidos. Otra de las características de OAA es que los agentes pueden ser añadidos o reemplazados en tiempo de ejecución.

La Arquitectura Cognitiva descrita en la sección 3.2 está implementada haciendo uso de OAA. El agente principal corresponde al Manejador de Diálogo, el cual es un programa intérprete construido en Prolog. El Manejador de Diálogo interpreta los protocolos conversacionales de los modelos de diálogo (cuando están representados en forma de código). Este programa es un coordinador de alto nivel de la percepción del sistema con sus respectivas acciones. Además realiza un seguimiento del contexto conversacional dinámico, el cual es necesario para realizar interpretaciones y ejecutar acciones que dependan de los eventos comunicativos previos y de las acciones en la conversación actual (Meza *et. al.*, 2010).

Subordinados al agente del Manejador de Diálogo, se encuentra un conjunto de agentes encargados de la percepción y comportamiento del sistema. El número de agentes subordinados depende de la aplicación a construir. Por ejemplo, un sistema podría contar con un agente encargado de visión, un agente encargado del reconocimiento de la voz, un agente encargado de la navegación del sistema, etc. En la Figura 3.7 se muestra la implementación de la Arquitectura Cognitiva utilizando agentes.

[10] http://www.ai.sri.com/~oaa/

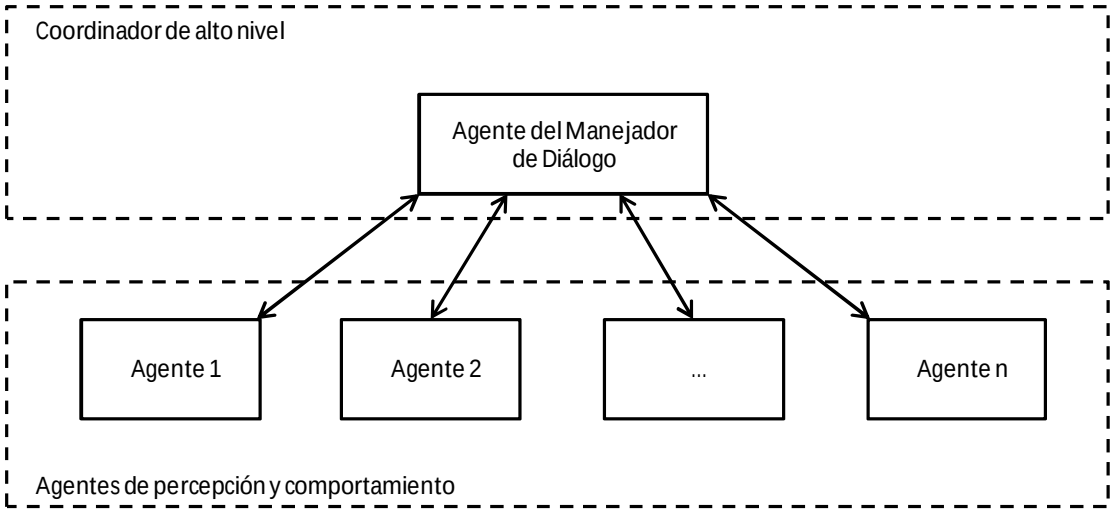

Figura 3.7 Implementación computacional de IOCA utilizando OAA

3.4 Aplicaciones

El marco de trabajo descrito en este capítulo ha permitido el desarrollo de diversas aplicaciones. Por ejemplo, el sistema "Adivina la carta", que es un sistema conversacional en Español hablado con capacidades de visión incorporadas, el cual se encuentra en un *stand* permanente del museo de ciencias Universum de la UNAM y que juega con los visitantes. En esta aplicación, niños de entre 10 y 14 años de edad tratan de adivinar una carta elegida por el sistema en forma aleatoria de un conjunto de diez cartas con temas astronómicos (como el sol, un telescopio, entre otros). Para lograr ganar, el usuario puede realizar hasta cuatro preguntas acerca de las características de la carta (por ejemplo, *¿es rojo?* o *¿parece un plato?*), y al terminar debe mostrar frente a la cámara la carta que piensa que fue elegida por el sistema (Meza *et. al.*, 2010).

También se ha desarrollado un robot guía que es capaz de conducir una sesión de posters a través de Español hablado. El robot es capaz de navegar hasta el lugar donde están los posters, identificarlos visualmente, y explicar secciones de ellos indicadas por el usuario mediante un gesto de apuntar (Avilés *et. al.*, 2010).

Otro sistema implementado sobre un robot móvil es el juego de Marco Polo. En este juego participan dos sujetos, uno de ellos asume el rol de ciego (cerrando o vendándose los ojos) y debe tratar de encontrar al otro únicamente guiándose con el sonido. La persona que es seguida debe gritar "Marco", y el que no puede ver debe responder "Polo" y tratar de moverse al lugar de donde provino el sonido del otro jugador. En esta aplicación, el robot toma el papel del sujeto ciego y es capaz de detectar la dirección del usuario y orientarse hacia él (es decir, estar cara a cara) en un ambiente auditivo complejo, usando sólo voz y un sistema de tres micrófonos. Esta funcionalidad está integrada utilizando modelos de diálogo y una arquitectura cognitiva (Rascón, Avilés y Pineda, 2010).

Haciendo uso de Modelos de Diálogo y de una Arquitectura Cognitiva fue posible la implementación sobre un robot móvil de otras de las pruebas de la competencia *RoboCup@Home* (Pineda, 2011). Una de ellas, llamada *Robot Inspection and Poster Session*, consiste en que un robot móvil debe ser capaz de registrarse por sí mismo para participar en la competencia. Para hacerlo, debe entrar en una habitación y buscar una mesa en la que se encuentra uno de los jueces. El robot se debe acercar a la mesa y presentarse con el juez, posteriormente debe entregarle un folder que contiene la forma de inscripción, y finalmente despedirse y retirarse del cuarto.

Go Get It! es otra de las pruebas de *RoboCup@Home* que fue modelada bajo este enfoque. En ella, el robot debe entrar al escenario y recibir una instrucción del juez que le indica el cuarto al que debe dirigirse (por ejemplo, el juez le puede decir "Ve a la cocina" o "Ve a la sala"). El robot debe ir a esa habitación y buscar un objeto (que se encuentra en una lista predefinida que recibe antes de la prueba). Una vez que lo identifica debe indicar que ya lo vio, para posteriormente tomarlo. Cuando ya tiene al objeto, se debe dirigir a la salida y dejar el escenario.

Otra de las pruebas desarrolladas fue *Who Is Who*, en la cual el robot entra a la arena de la competencia y espera a que dos personas desconocidas se presenten ante él diciendo su nombre. Las dos personas se van a otra habitación, una de ellos se sienta y la otra se queda parada. En la habitación también hay dos personas que son miembros del equipo del robot, una parada y otra sentada, y a quienes ya conoce con anterioridad. Además, en la misma habitación está parada una quinta persona que es totalmente desconocida. El robot recibe la instrucción de iniciar la búsqueda y se dirige a la habitación. Cuando detecta a una persona, debe encararla e identificarla, diciendo el respectivo nombre (con excepción de la persona desconocida a quien debe indicarle que no la conoce). El robot debe tratar de identificar correctamente al mayor número de personas y salir de la arena antes de que se termine el tiempo asignado a esta prueba.

Las aplicaciones listadas en esta sección son un claro ejemplo de que los Modelos de Diálogo y el uso de una Arquitectura Cognitiva permiten el diseño de una rica variedad de sistemas interactivos multimodales. Este marco de trabajo resulta idóneo para desarrollar el sistema del robot seguidor que puede resolver la prueba *Follow Me*. En el siguiente capítulo se procederá con la especificación de la tarea haciendo uso de Modelos de Diálogo.

Capítulo 4

Especificación de la tarea *Follow Me* mediante Modelos de Diálogo

Una vez descrita la tarea a ser implementada en el capítulo dos, y de haber presentado el marco metodológico en el capítulo tres, es momento de empezar con la explicación acerca de cómo lograr pasar de la descripción de la prueba plasmada en el manual de reglas a un sistema real implementado sobre un robot móvil.

El primer paso, y sin duda el más importante, consiste en modelar la estructura de la tarea. Este capítulo presenta los modelos de diálogo que describen la prueba, teniendo como resultado una descripción funcional de *Follow Me* que es totalmente independiente de los aspectos algorítmicos y de implementación.

Cada uno de los modelos de diálogo es mostrado en su representación gráfica, acompañado de una explicación sobre su funcionamiento y de un ejemplo de la interacción entre humano y robot correspondiente a la parte de la tarea que se está modelando. Como complemento, es posible encontrar en el Apéndice A los respectivos modelos de diálogo representados en forma de código.

En la sección 4.1 se presentan en forma breve algunas notas generales sobre los modelos de diálogo que se muestran en este capítulo. Primero se explica el modelo de diálogo principal en la sección 4.2, el cual contiene cinco situaciones recursivas que serán explicadas posteriormente, y que corresponden a las distintas fases de la prueba *Follow Me*: inicio de la prueba (sección 4.3), oclusión temporal (sección 4.4), rastreando a lo lejos (sección 4.5), reconociendo al usuario (sección 4.6) y línea de meta (sección 4.7). Finalmente, en la sección 4.8 se explica una rutina de recuperación utilizada en varios de los modelos anteriores y que se invoca cuando el robot pierde al usuario.

4.1 Generalidades

Los modelos de diálogo que especifican la tarea *Follow Me* son resultado de un cuidadoso análisis y una comprensión profunda de los requerimientos descritos en el manual de reglas de *RoboCup@Home*. El primer paso que se realizó fue el diseño de estos modelos de diálogo en representación gráfica, y posteriormente se codificaron para que pudieran ser interpretados por el manejador de diálogo.

Las situaciones se nombran de acuerdo a su tipo. A continuación se muestra una lista de la nomenclatura utilizada para los tipos de situaciones presentes en los modelos de diálogo de *Follow Me*:

- *si* .- situación inicial (*initial situation)*, representa el punto de partida de la tarea o subtarea modelada y sólo puede haber una en cada modelo o submodelo de diálogo. En todos los casos mostrados es del tipo neutral, pero puede ser de cualquier otro tipo.
- *n* .- situación neutral (*neutral*), implica que no hay una modalidad de entrada para esta situación, ya que la expectativa que se tiene es vacía.
- *ls* .- situación de escuchar (*listening*), está constituida por expectativas que corresponden a actos del habla por parte del interlocutor.
- *v* .- situación de ver (*seeing*), cuyas expectativas corresponden a eventos que el sistema puede percibir mediante el canal visual.
- *c* .- situación de botón (*click*), en las que se espera que el usuario presione un botón específico.
- *R* .- situación recursiva (*recursive*), que corresponde a un submodelo de diálogo completo.
- *fs* .- situación final (*final*), son los puntos en donde termina la tarea o subtarea. Puede haber más de una en cada modelo o submodelo de diálogo.

Esta nomenclatura no es obligatoria, pero se adopta para facilitar la comprensión de los modelos de diálogo. En caso de que en un modelo de diálogo exista más de una situación del mismo tipo, después de las letras que corresponden al tipo de situación, se añade una cadena de símbolos como identificador alfanumérico. Por ejemplo, si en un modelo de diálogo hay varias situaciones del tipo escuchar, podríamos llamarlas *ls*, *ls1*, *ls2*, *ls3*, *… , lsa* , *lsb*, *ls_nombre*, etc.

4.2 Modelo de diálogo principal

A manera de recordatorio, la tarea *Follow Me* consiste en que una persona se debe presentar ante un robot y pedirle que la siga. Una vez que el robot ha terminado la calibración respectiva, le indica a la persona que puede empezar a caminar. El robot debe seguir a la persona a través de una ruta desconocida, y debe superar cuatro eventos:

- Oclusión temporal.- una segunda persona se cruza entre el robot y el usuario.
- Rastreando a lo lejos.- el usuario le dice al robot que se detenga un momento, y camina tres metros hacia enfrente. El robot debe poder identificar al usuario y acercarse a él.
- Reconociendo al usuario.- el usuario le pide al robot que se pare y se esconde de él. Posteriormente regresa acompañado de otra persona, y ambas se ponen enfrente del robot, desafiando la capacidad de reconocer cual era la persona a la que seguía originalmente.
- Línea de meta.- el robot debe cruzarla antes de que el tiempo se agote.

Este recordatorio de la descripción de la tarea se hace con el objetivo de enfatizar las partes que la conforman y que será de gran utilidad para entender la manera en que se diseñaron los modelos de diálogo que se presentarán en las siguientes secciones. Para una descripción más detallada de cada una de las fases consultar la sección 2.2 de este trabajo.

El modelo de diálogo principal, mostrado en la Figura 4.1, está diseñado de modo que capture la estructura global de la tarea, teniendo como constituyentes principales situaciones recursivas dedicadas a cada una de las partes importantes de la prueba, como es el caso de la presentación entre el robot y el usuario y los cuatro eventos que el robot debe superar en forma secuencial.

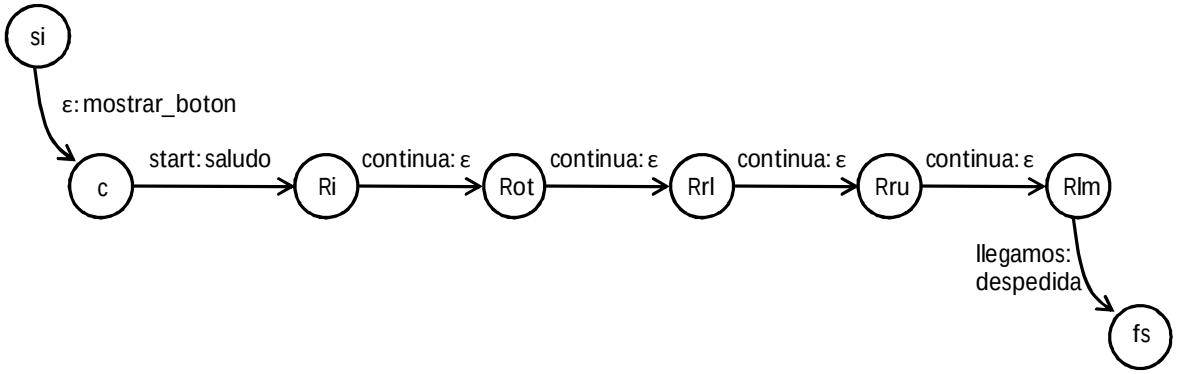

Figura 4.1 Modelo de diálogo principal

La situación inicial es *si* y la situación final es *fs*. Dado que la expectativa de *si* es vacía, siempre se ejecuta de manera determinista el acto retórico *mostrar_boton* y se pasa a la situación *c.* El acto retórico *mostrar_boton*, consiste en enseñarle al usuario un botón e indicarle que lo debe presionar para que la prueba de inicio. La situación *c* es de tipo clic, teniendo como expectativa *start* (que se cumple cuando el usuario da clic sobre el botón de inicio), y teniendo como acto retórico un saludo del robot para presentarse con el usuario.

A partir de aquí, da inicio una secuencia de situaciones recursivas, cada una de las cuales corresponde a una fase de la tarea *Follow Me*:

- *Ri* .- situación recursiva de inicio (presentación entre robot y usuario).
- *Rot* .- situación recursiva de oclusión temporal (checkpoint 1 de la prueba).
- *Rrl* .- situación recursiva de rastreando a lo lejos (checkpoint 2 de la prueba).
- *Rru* .- situación recursiva de reconociendo al usuario (checkpoint 3 de la prueba).
- *Rlm* .- situación recursiva de línea de meta (checkpoint 4 de la prueba).

Para todas las situaciones recursivas, excepto para *Rlm*, se espera como expectativa a la etiqueta de continuación del submodelo de diálogo, la cual es *continua*, y no se ejecuta ninguna acción. La situación *Rlm*, espera como expectativa la etiqueta de continuación *llegamos*, y el acto retórico consiste en despedirse del usuario, indicando que la prueba ha finalizado.

En la Tabla 4.1, se muestra la interacción entre el usuario *H* y el robot *R* correspondiente al modelo de diálogo principal en el sistema ya implementado. Para ver el modelo de diálogo principal codificado consultar el Apéndice A, sección 1.

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
1	R	Dice: “Presiona el botón verde cuando quieras empezar”	Lenguaje hablado	
		Muestra en su display un botón verde	Lenguaje gráfico	
2	H	El usuario presiona el botón verde	Lenguaje táctil	*start*
3	R	Dice: “Hola a todos, mi nombre es Golem”	Lenguaje hablado	
…	…	…	…	…
m	R	Dice: “Es tiempo de decir adiós. Nos vemos después”	Lenguaje hablado	

Tabla 4.1 Ejemplo de interacción entre humano (*H*) y robot (*R*)

4.3 Submodelo de diálogo Inicio

El primer gran reto que enfrenta el robot es conocer a la persona a ser seguida durante el transcurso de la tarea. Para ello debe realizar una oferta indicando que está esperando a que alguien le solicite ser seguido. Una vez que alguien se acerque y le pida que lo siga, el robot le dará determinadas instrucciones para poder conocerlo (por ejemplo, le puede pedir que se pare justo enfrente de él, que levante las manos, etc.). Uno de los problemas que puede ocurrir en esta fase de presentación es el hecho de que el usuario no obedezca correctamente las indicaciones del robot (por ejemplo, que no se coloque en el lugar que el robot le indicó), en cuyo caso el robot debe repetir la indicación o explicar la instrucción de una manera más clara. En la Figura 4.2 se muestra el modelo de diálogo que corresponde a este momento de la prueba.

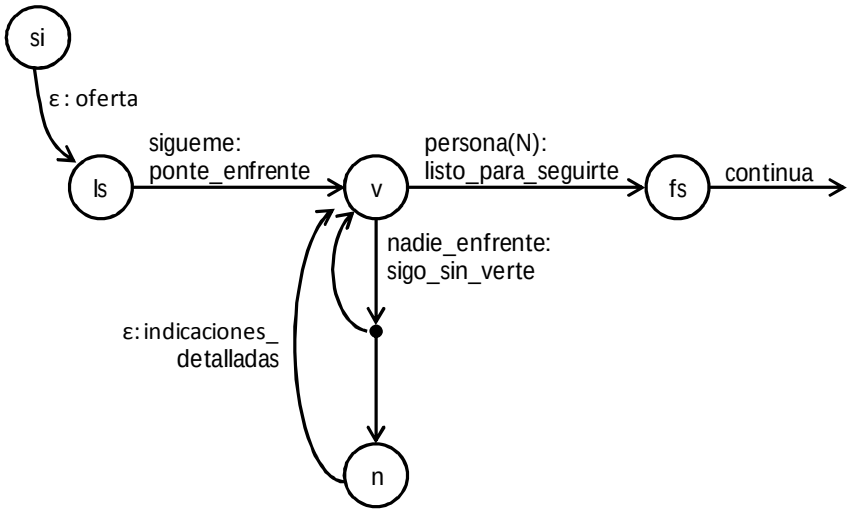

Figura 4.2 Submodelo de diálogo Inicio

Al inicio del modelo de diálogo se presenta la acción *oferta*, en la cual el robot indica al usuario que tiene la capacidad de seguir a una persona y que está en espera de que alguien le de la instrucción para empezar con la tarea. En la situación *ls* se tiene come expectativa recibir un comando hablado, el cual puede ser una expresión como “Sígueme”, “Ven conmigo”, “Vamos”, entre otras.

En la situación *ls* además de la expectativa *sígueme*, es necesario considerar la posibilidad de errores del sistema, entre ellos que no entienda lo que el usuario dice o que no esté escuchando. Aunque no se hace explícito en la representación gráfica del modelo de diálogo, en el modelo codificado se consideran estos posibles errores del sistema, y cuando suceden se informa al usuario enviando un mensaje de error (consultar Apéndice A para ver el código respectivo).

Una vez que la expectativa *sígueme* se satisface, el sistema solicita al usuario que se ponga enfrente para que pueda verlo y se pasa a la situación *v*, en la cual hay dos posibilidades: ver a una persona o no ver a nadie. Cada vez que la expectativa *nadie_enfrente* se cumple, el sistema avisa al usuario que no lo está viendo y evalúa una función para determinar cuál es la situación siguiente. En esta función se está tomando como argumento a la historia de la interacción, y cuenta el número de veces que se ha obtenido la expectativa *nadie_enfrente* en la situación *v*. Si esto ocurre tres veces se pasa a la situación *n*, en la que se tiene como acto retórico una explicación más detallada al usuario de lo que debe hacer para que el sistema pueda verlo (consultar Apéndice A para ver el código de esta función).

Finalmente, cuando en la situación *v* se cumple la expectativa *persona(N)*, donde a la variable *N* se le asigna un identificador numérico que proviene del módulo de interpretación de la Arquitectura Cognitiva Orientada a la Interacción, el sistema informa al usuario que ya puede empezar a caminar. Como parte de este acto retórico, el sistema de navegación se activa y comienza a seguir a la persona.

En la tabla 4.2 se muestra un ejemplo de interacción entre humano y robot correspondiente a esta fase de la prueba, en la cual podemos ver la forma en que se recupera el sistema cuando las hipótesis de reconocimiento son incorrectas. En el turno *n+1* el sistema no entiende lo que el humano le dijo, por lo que en el turno *n+2* le solicita que repita de nuevo el comando hablado. En el turno *n+5*, a pesar de que el humano está enfrente del robot, el sistema no logra identificarlo en forma correcta, por lo que en el turno *n+6* el sistema le pide que espere un momento. Estas indicaciones por parte del sistema ayudan a que la interacción sea más natural y fluida.

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
…	…	…	…	…
n	R	Dice: “Estoy listo para seguir a alguien. Cuando quieras empezar dime Sígueme”	Lenguaje hablado	
n+1	H	Dice: “Muy bien robot, sígueme”	Lenguaje hablado	*noEntendi*
n+2	R	Dice: “No te puedo escuchar bien, podrías repetirlo”	Lenguaje hablado	
n+3	H	Dice: “Sí, ven conmigo”	Lenguaje hablado	*sígueme*
n+4	R	Dice: “Muy bien. Podrías pararte enfrente de mi a seis pies de distancia, haciendo el gesto de la tarjeta”	Lenguaje hablado	
n+5	H	Se aleja del robot y hace el gesto indicado en una tarjeta	Lenguaje gestual	*nadie_enfrente*
n+6	R	Dice: “Espera un momento, sigue haciendo el gesto”	Lenguaje hablado	
n+7	H	Continúa haciendo el gesto	Lenguaje gestual	*persona(1)*[11]
n+8	R	Dice: “Perfecto, ya te vi. Ahora puedes empezar a caminar y yo te seguiré”	Lenguaje hablado	
		Empieza a seguir a la persona	Acción motora	
…	…	…	…	…

Tabla 4.2 Ejemplo de interacción entre humano (*H*) y robot (*R*) en la parte inicial de la prueba

[11] En este ejemplo de interacción y en los que se presentan en las siguientes secciones se supondrá que el módulo de interpretación de la arquitectura cognitiva le asignó al usuario el identificador numérico ‘1’.

4.4 Submodelo de diálogo Oclusión temporal

El siguiente evento que debe superar el robot seguidor es una oclusión temporal. El robot tiene que estar siguiendo al usuario teniendo en cuenta que en cualquier momento una segunda persona se atravesará entre ellos. El robot debe ser capaz de reanudar la actividad de seguimiento una vez que la segunda persona se ha quitado de en medio. Varios problemas pueden presentarse en esta parte de la prueba, entre ellos, el hecho de que el sistema no se percate nunca de que ocurrió la oclusión temporal, o que al quitarse la segunda persona el sistema no identifique de nuevo al usuario. En estos casos, será necesario incluir rutinas de recuperación que permitan al robot restablecer contacto con el usuario. El modelo de diálogo mostrado en la Figura 4.3 corresponde a esta parte de la prueba y considera los problemas antes mencionados.

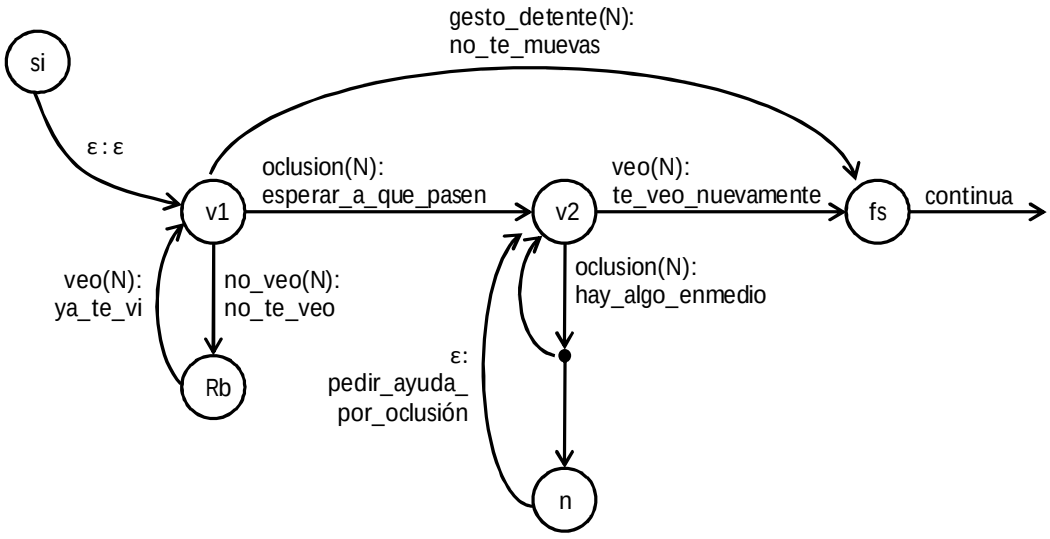

Figura 4.3 Submodelo de diálogo Oclusión Temporal

En la situación *v1* el robot está siguiendo a la persona y se tienen como expectativas tres posibles eventos visuales. El primero de ellos es el hecho de que el usuario sea perdido por alguna razón, por ejemplo, que camine demasiado rápido y se salga del campo visual del sistema, o que exista algún error en el reconocimiento de la persona provocado por la iluminación. En este caso, se cumple la expectativa *no_veo(N)*, por lo que el robot se detiene y envía un mensaje avisando que ha dejado de ver al usuario y pasa entonces a la situación recursiva *Rb*, que corresponde a una rutina de recuperación que termina hasta que el robot vuelve a ver al usuario. Una vez que se obtiene la etiqueta de continuación *veo(N)*, el robot avisa al usuario que ya lo vio de nuevo y empieza a seguirlo otra vez.

Otra de las expectativas de *v1* es *oclusion(N)*, la cual se satisface cuando una segunda persona aparece en la escena y pasa entre el usuario y el robot. Cuando esto sucede, el robot se detiene y avisa que se ha percatado de que alguien está cruzando entre ellos y entonces se prosigue con la situación *v2*. Existe la posibilidad de que la segunda persona cruce demasiado rápido entre el usuario y el robot, y por lo tanto el sistema no se percate del momento en que la oclusión ocurre. En caso de suceder esto, el modelo de diálogo se mantendrá en la situación *v1* siguiendo a la persona hasta que el sistema detecte el gesto del usuario indicando al robot que se detenga (lo cual forma parte de la siguiente fase de la prueba). Al satisfacerse la expectativa *gesto_detente(N)*, el robot se detiene y pide al usuario que se mantenga haciendo el gesto y se va al estado final *fs*.

En la situación *v2* se tiene como expectativa ver nuevamente a la persona. Ya que el robot empezó a decir un mensaje justo cuando detectó que la segunda persona estaba cruzando, se espera que una vez que termine de decirlo la segunda persona ya se haya quitado de en medio. En este caso, será satisfecha la expectativa *veo(N)* y el robot seguirá de nuevo al usuario indicándole que la otra persona ya no le estorba en su camino. En caso de que la segunda persona cruce muy lento, o se quede parada justo entre el usuario y el robot, se volverá a satisfacer la expectativa *oclusion(N)*, y el sistema indicará mediante un mensaje que la persona desconocida sigue obstaculizando el seguimiento. Si varias veces se repite la expectativa *oclusion(N)*, se envía un mensaje pidiendo ayuda al usuario, indicándole que la persona desconocida ha permanecido demasiado tiempo entre ellos y que posiblemente haya un error.

En la Tabla 4.3 se presenta un ejemplo de interacción entre el usuario *H* y el robot *R*. En esta ocasión aparece una segunda persona *S* cuyas acciones afectan a la interacción entre *H* y *R*. En el turno *o* el sistema deja de reconocer al usuario por alguna razón, por ejemplo, un cambio drástico en la iluminación de la nueva posición del usuario, obteniendo como hipótesis de reconocimiento *no_veo(N)*, lo que provoca que en el turno *o+1* el robot se detenga y solicite ayuda entrando a una rutina de recuperación que empieza en el turno *o+2* y termina en el turno *p-1*. Para ver esta rutina de recuperación consultar en la sección 4.8 el modelo de diálogo Buscar. En el turno *p* el sistema reconoce de nuevo al usuario, y lo empieza a seguir de nuevo a partir del turno *p+1*.

En el turno *p+2* ocurre algo interesante, pues la acción depende de dos personas. En este caso, *H* sigue caminando y repentinamente una segunda persona *S* se involucra. A diferencia de lo que debería de ocurrir en la prueba *Follow Me*, en este ejemplo *S* se queda parado entre *H* y *R*, por lo que en el turno *p+4* se vuelve a obtener como hipótesis *oclusion(N)*.

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
…	…	…	…	…
o	H	Sigue caminando	Acción motora	*no_veo(1)*
o+1	R	Se detiene	Acción motora	
		Dice: " ¿A dónde te has ido?. No puedo verte"	Lenguaje hablado	
…	…	…	…	…
p	H	Está detenido y se balancea ligeramente para que el robot lo vuelva a detectar	Acción motora	*veo(1)*
p+1	R	Dice: "Ya te vi. Eso es bueno porque ya estaba espantado"	Lenguaje hablado	
		Empieza a seguir a la persona	Acción motora	
p+2	H	Sigue caminando	Acción motora	*oclusion(1)*
	S	Se atraviesa entre H y R	Acción motora	
p+3	R	Dice: "Alguien está pasando entre tu y yo, así que me detendré por un momento"	Lenguaje hablado	
		Se detiene	Acción motora	
p+4	H	Se detiene	Acción motora	*oclusion(1)*
	S	Se queda parada entre H y R	Acción motora	
p+5	R	Dice: "Todavía hay alguien entre nosotros"	Lenguaje hablado	
p+6	H	Sigue detenida	Acción motora	*veo(1)*
	S	Camina y se quita de entre H y R	Acción motora	
p+7	R	Dice: "Te veo nuevamente, caminemos"	Lenguaje hablado	
		Empieza a seguir a la persona	Acción motora	
…	…	…	…	…

Tabla 4.3 Ejemplo de interacción entre humano (*H*) y robot (*R*) en el checkpoint Oclusión Temporal

4.5 Submodelo de diálogo Rastreando a lo lejos

El siguiente suceso importante en la prueba *Follow Me* se da cuando el usuario le pide al robot que se detenga mediante un comando gestual. Al recibir esta instrucción, el robot se parará y esperará diez segundos inmóvil para que el usuario tenga oportunidad de alejarse caminando tres metros hacia enfrente. Una vez que el tiempo termina, el robot tiene que identificar al usuario a la distancia y acercarse a él. En la Figura 4.4 se presenta el modelo de diálogo correspondiente a esta fase.

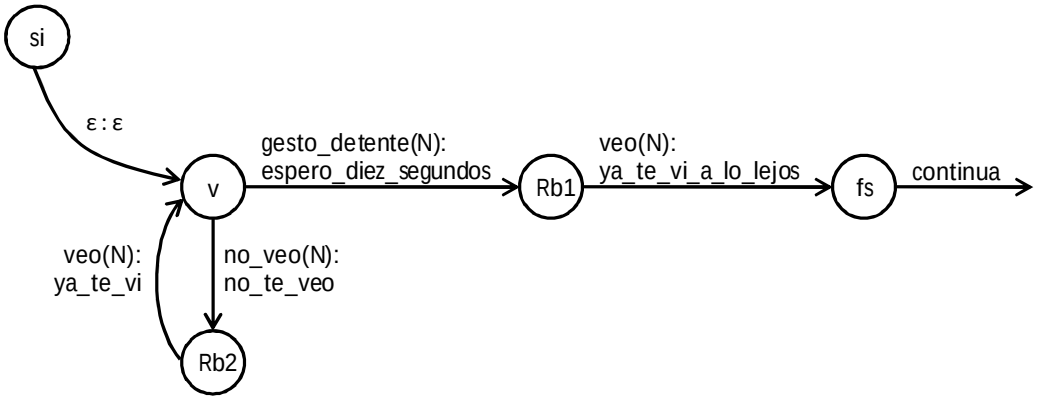

Figura 4.4 Submodelo de diálogo Rastreando a lo lejos

En la situación *v* el robot se encuentra siguiendo a la persona. Hay dos expectativas, una de ellas es *no_veo(N)*, que provoca un comportamiento del sistema similar al de la expectativa *v1* del modelo de diálogo Oclusión temporal (sección 4.4), pasando a una situación recursiva que corresponde a una rutina de recuperación para buscar al usuario. La otra expectativa de *v* es *gesto_detente(N)*, la cual se cumple cuando el sistema identifica un gesto específico de la persona a la que está siguiendo. Acto seguido, el robot le dirá al usuario que se detendrá por diez segundos, tiempo en el cual el usuario debe caminar hacia enfrente.

Es importante señalar la posibilidad de que desde el modelo de diálogo correspondiente a la fase anterior (Oclusión temporal), el sistema haya identificado el gesto del usuario pidiendo al robot que se detenga. Eso no provoca ningún conflicto, ya que cuando se identifica el gesto desde la fase anterior, el sistema solicita al usuario que se mantenga haciendo el gesto, por lo que al llegar a la situación *v* de este diálogo el usuario seguirá haciéndolo.

La situación *Rb1* corresponde a una situación recursiva en la que se invoca al mismo submodelo de diálogo que *Rb2*. El robot se mantendrá detenido hasta que encuentre de nuevo al usuario, y cuando lo haga le dirá que ya lo ha visto y se acercará a él. En la Tabla 4.4 se muestra un ejemplo de interacción entre el robot y el usuario en este momento de la prueba.

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
…	…	…	…	…
q	H	Se detiene	Acción motora	
		Dice al robot que se detenga mediante un gesto	Lenguaje gestual	*gesto_detente(1)*
q+1	R	Se detiene	Acción motora	
		Dice: "Veo un gesto. Esperaré diez segundos mientras te alejas de mí. Uno, dos, tres, cuatro, cinco, seis, siete, ocho, nueve, diez. El tiempo se acabó"	Lenguaje hablado	
…	…	…	…	…
r	H	Está parado tres metros enfrente del robot.	Acción motora	*veo(1)*
r+1	R	Dice: "Te veo. Me acercaré a ti, espera un momento"	Lenguaje hablado	
		Empieza a seguir a la persona	Acción motora	
…	…	…	…	…

Tabla 4.4 Ejemplo de interacción entre humano (*H*) y robot (*R*) en el checkpoint Rastreando a lo lejos

4.6 Submodelo de diálogo Reconociendo al usuario

El siguiente evento al que se enfrenta el robot se da cuando el usuario le pide nuevamente que se detenga, pero en esta ocasión para esconderse. Posteriormente regresa acompañado de una persona desconocida y los dos se ponen frente al robot, quien debe identificar cuál era la persona que seguía originalmente. Este submodelo, mostrado en la Figura 4.5, corresponde a este checkpoint de la prueba.

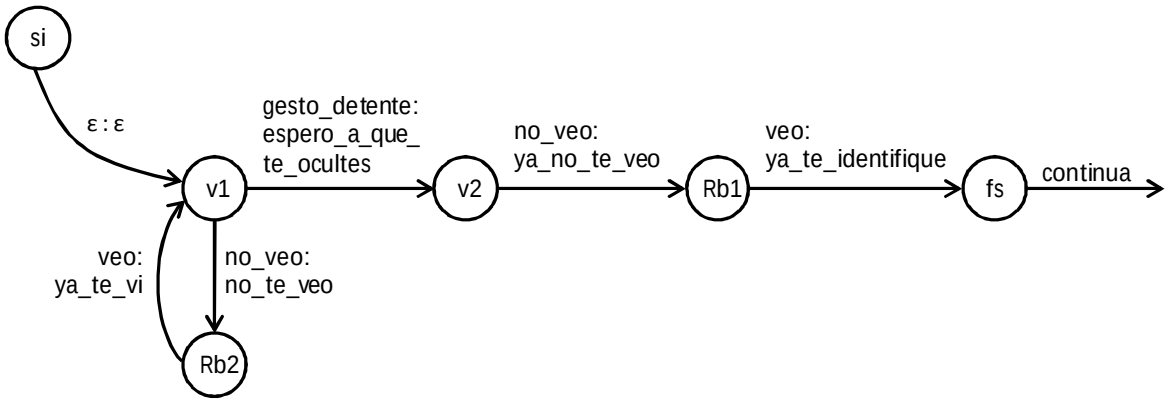

Figura 4.5 Submodelo de diálogo Reconociendo al usuario

La situación *v1* es similar a la del mismo nombre en el modelo de diálogo Rastreando a lo lejos (sección 4.5). La única diferencia es que cuando se satisface la expectativa *gesto_detente(N)*, el robot le dice al usuario que se detendrá y que esperará a que se oculte. En la situación *v2* el robot permanecerá sin moverse esperando a que el usuario salga de su campo visual, y cuando esto sucede se cumple la expectativa *no_veo(N)* y el robot avisa que ya no está viendo al usuario. En la situación recursiva *Rb1* se invoca al submodelo de diálogo Buscar, y el robot está en espera de que la persona a la que seguía originalmente aparezca de nuevo. Una vez que las dos personas se paran enfrente del robot, el sistema identifica cual es la correcta y entonces la expectativa *veo(N)* se cumple. El robot avisa que ya identificó al usuario y empieza a seguirlo de nuevo. En la Tabla 4.5 se muestra un ejemplo de interacción entre el usuario *H* y el robot *R*, considerando a la persona desconocida *D* cuya presencia busca dificultar la interacción en esta etapa de la prueba.

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
…	…	…	…	…
s	H	Se detiene	Acción motora	
		Dice al robot que se detenga mediante un gesto	Lenguaje gestual	*gesto_detente(1)*
s+1	R	Se detiene	Acción motora	
		Dice: "Así que quieres que me detenga de nuevo. Está bien. Ahora te puedes ocultar de mi"	Lenguaje hablado	
s+2	H	Se esconde del robot	Acción motora	*no_veo(1)*
s+3	R	Dice: "Estas escondido muy bien. No puedo verte. Esperaré aquí hasta que regreses"		
…	…	…	…	…
t	H	Está parado frente al robot	Acción motora	*veo(1)*[12]
	D	Está parado frente al robot	Acción motora	
t+1	R	Dice: "Te he identificado. Me acercaré a ti"	Lenguaje hablado	
		Empieza a seguir a la persona	Acción motora	
…	…	…	…	…

Tabla 4.5 Ejemplo de interacción entre humano (*H*) y robot (*R*) en el checkpoint Reconociendo al usuario

[12] En la Tabla 4.3 (Oclusión temporal), las hipótesis de reconocimiento de los turnos en los que participaban dos humanos dependían de la acción de ambos, pues para considerar a un evento como oclusión se requiere a un oclusor y a un ocluido. En esa misma fase, en el turno *p+6* la hipótesis *veo(N)* dependía de que el oclusor se quitara y el ocluido siguiera en su posición. Pero en la tabla 4.5, la obtención de la expectativa *veo(N)* depende sólo de *H*, ya que aunque *D* no haga la acción correspondiente, es decir, no se pare enfrente del robot, el sistema intentará reconocer a *H*. Por esta razón se mantiene la línea divisoria horizontal en la hipótesis de reconocimiento del turno *t*, indicando que la hipótesis sólo depende de *H*.

4.7 Submodelo de diálogo Línea de meta

La última parte de la prueba *Follow* Me consiste en que el robot debe cruzar la línea de meta. En la Figura 4.6 se muestra el último de los modelos recursivos presentes en el modelo de diálogo principal y tiene por objetivo lograr que el robot siga al usuario hasta llegar al lugar en que termina la prueba y asegurarse de que ha terminado la tarea preguntándole al usuario si ya la ha cruzado.

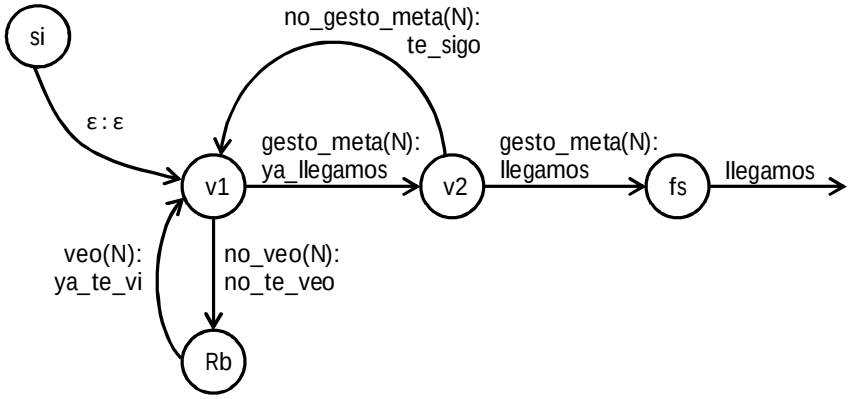

Figura 4.6 Submodelo de diálogo Línea de meta

De acuerdo a la descripción de la prueba *Follow Me*, bastaría con que el robot se mantuviera siguiendo al usuario indefinidamente hasta llegar a la línea de meta, por lo que este modelo de diálogo podría constar únicamente de la situaciones *si*, *v1* y *Rb*. Sin embargo, para que el final de la prueba no fuera tan abrupto, se incluyó la opción de que el usuario le indique al robot mediante un gesto específico que ya cruzó la línea de meta.

Cuando en *v1* se detecta el gesto de meta, el robot informa al usuario que está viendo un gesto indicando que la prueba ha terminado, pero le pide que se mantenga haciendo el gesto con el objetivo de verificar. Si en la situación *v2* el sistema vuelva a ver el gesto de meta, entonces la tarea termina y se manda un mensaje de despedida. En caso contrario, el robot sigue caminando detrás del usuario. En la tabla 4.6 se muestra un ejemplo de interacción en este checkpoint.

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
…	…	…	…	…
u	H	Se detiene	Acción motora	
		Dice al robot que se detenga mediante un gesto	Lenguaje gestual	*gesto_meta(1)*
u+1	R	Se detiene	Acción motora	
		Dice: "Veo que estás haciendo un gesto. Si ya llegamos a la línea de meta mantén tus manos arriba"	Lenguaje hablado	
u+2	H	Deja de hacer el gesto	Acción motora	*no_gesto_meta(1)*
u+3	R	Dice: "Tus manos están abajo. Sigamos caminando."	Lenguaje hablado	
		Empieza a seguir a la persona	Acción motora	
u+4	H	Dice al robot que detenga mediante un gesto	Lenguaje gestual	*gesto_meta(1)*
u+5	R	Se detiene	Acción motora	
		Dice: "Veo que estás haciendo un gesto. Si ya llegamos a la línea de meta mantén tus manos arriba"	Lenguaje hablado	
u+6	H	Sigue haciendo el gesto	Lenguaje gestual	*gesto_meta(1)*
u+7	R	Dice: "Estoy feliz porque hemos llegado"	Lenguaje hablado	
…	…	…	…	…

Tabla 4.6 Ejemplo de interacción entre humano (*H*) y robot (*R*) en el checkpoint Línea de meta

4.8 Submodelo de diálogo Buscar

Este submodelo de diálogo es invocado en varias ocasiones dentro de los modelos de diálogo descritos anteriormente. El objetivo es buscar al usuario, para lo que se vale de una serie de mensajes hablados que tienen por objetivo motivar al interlocutor para que coopere con la recuperación del sistema cuando lo ha perdido de vista. El modelo de diálogo se muestra en la Figura 4.7 y está conformado por situaciones del tipo visual, cada una de las cuales tiene las expectativas *veo(N)* y *no_veo(N)*. Cada vez que se cumple la expectativa *no_veo(N)*, se envía un mensaje al usuario. En la Tabla 4.7 se muestra un breve ejemplo de la interacción que se suscita cuando este submodelo de diálogo es invocado.

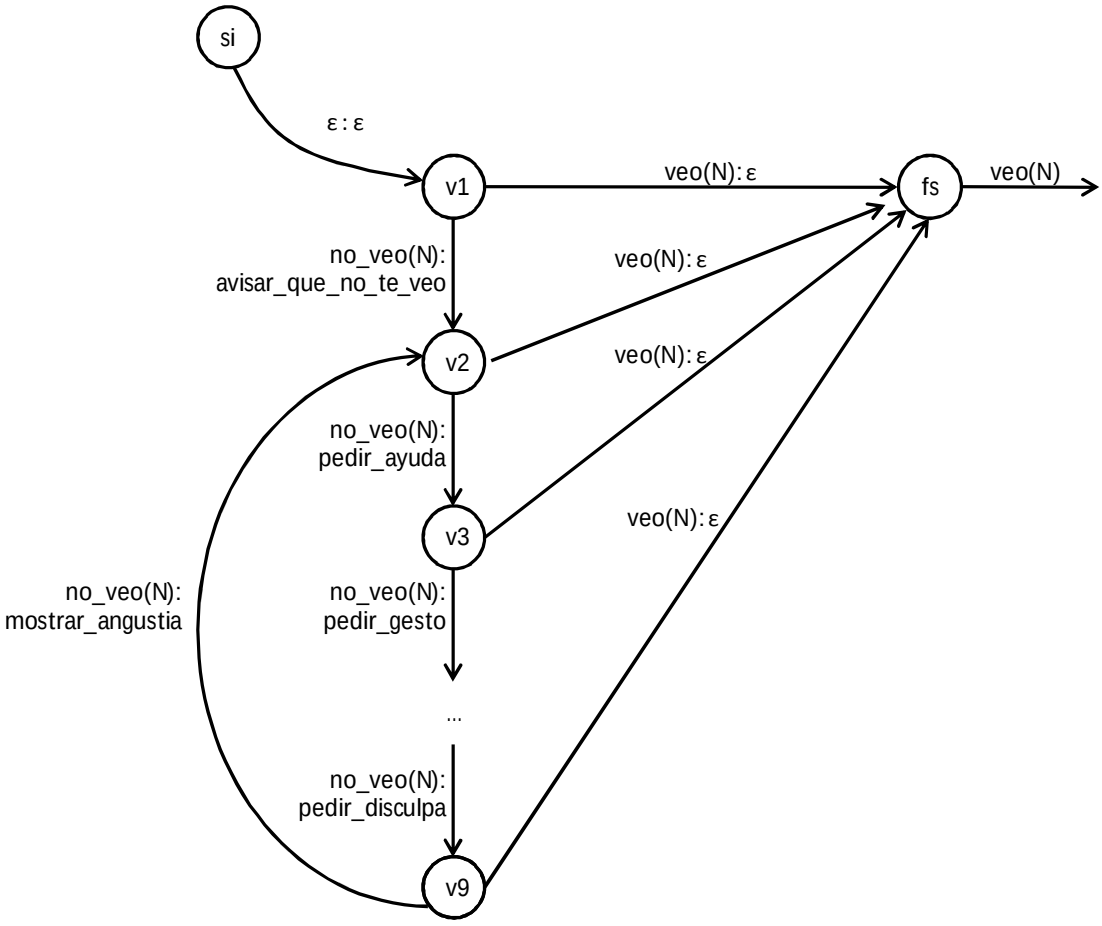

Figura 4.7 Submodelo de diálogo Buscar

Turno	Emisor	Acción	Modalidad de la acción	Hipótesis de reconocimiento del robot
…	…	…	…	…
v	H	Está detenido	Acción motora	*no_veo(1)*
v+1	R	Dice: “Espera un momento. Te perdí”	Lenguaje hablado	
v+2	H	Sigue detenido	Acción motora	*no_veo(1)*
v+3	R	Dice: “Necesito ayuda porque no te puedo encontrar. Te podrías poner enfrente de mi”.	Lenguaje hablado	
v+4	H	Se mueve ligeramente para colocarse justo enfrente del robot	Acción motora	*veo(1)*
…	…	…	…	…

Tabla 4.7 Ejemplo de interacción entre humano (*H*) y robot (*R*) en la rutina de recuperación Buscar

Una vez que todos los modelos de diálogos han quedado definidos, se tiene un panorama completo que permitirá identificar qué agentes de percepción y comportamiento se necesitarán para la implementación de la tarea *Follow Me*. En el siguiente capítulo, se explicarán a detalle estos agentes y la manera en que fueron construidos.

Capítulo 5

Agentes de percepción y comportamiento para la prueba *Follow Me*

En el capítulo anterior se presentaron los modelos de diálogo para un robot seguidor de personas capaz de ejecutar en forma completa la prueba *Follow Me*. Además de tener modelada la estructura de la tarea, ahora es posible identificar los agentes que se necesitan implementar y lo que se espera de cada uno de ellos.

La implementación de la prueba en un robot móvil fue posible gracias a reutilizar muchos de los recursos de los proyectos DIME (Diálogos Inteligentes Multimodales en Español)[13] y GOLEM (Navegación en un robot móvil mediante información visual y de lenguaje natural)[14], entre ellos el manejador de diálogo, el agente de entendimiento del habla, el agente de navegación y el agente de despliegue de botones gráficos. Sin embargo, para la prueba *Follow Me*, se necesitó crear un agente de visión especializado en el reconocimiento de personas. En la Figura 5.1 se muestran a los agentes involucrados en el sistema.

La sección 5.1 está dedicada a explicar detalladamente la forma en que funciona el agente de visión. Posteriormente, en la sección 5.2 se explican en forma breve los demás agentes involucrados en el sistema. Finalmente en la sección 5.3 se brindan los detalles técnicos relacionados a la implementación del sistema sobre un robot móvil.

[13] http://leibniz.iimas.unam.mx/~luis/DIME/
[14] http://golem.iimas.unam.mx/home

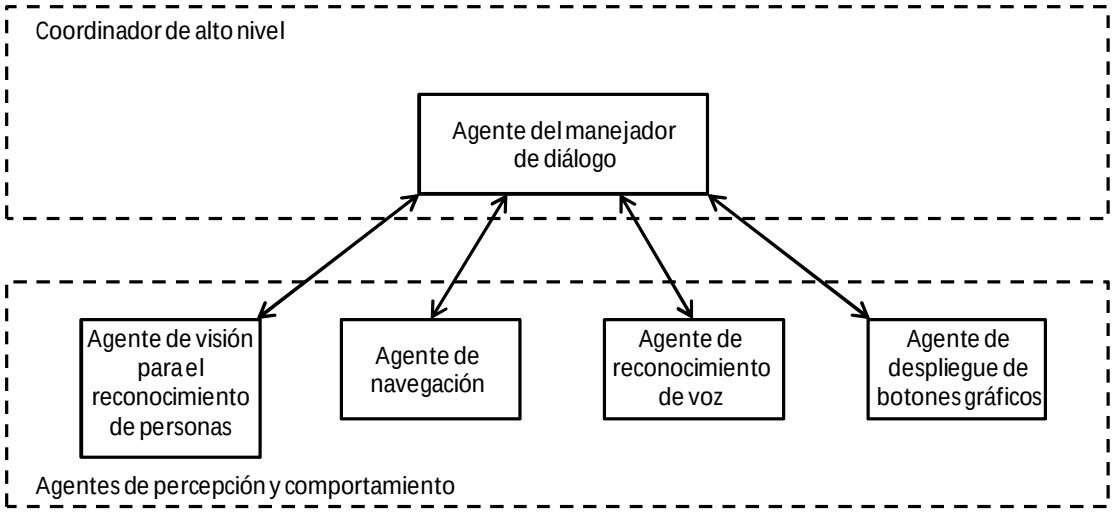

Figura 5.1 Agentes que forman parte del sistema para resolver la tarea *Follow Me*

5.1 Agente de visión para el reconocimiento de personas

Este agente de percepción tiene como objetivo encargarse de todas las situaciones del tipo ver que se presentan en los modelos de diálogo de la tarea *Follow Me*. Es importante destacar que todas estas situaciones involucran el reconocimiento de una persona en la escena tridimensional. En la sección 2.3 se identificaron y explicaron las habilidades que necesita un robot seguidor para poder resolver en forma completa las diversas fases de la prueba, algunas de las cuales están relacionadas con el reconocimiento de personas en la escena tridimensional:

- Reconocimiento de personas:
  - Detección de personas
  - Localización de personas
  - Identificación de personas
- Tracking de personas
- Reconocimiento de gestos

Se puede hacer una especificación puntual acerca de lo que se espera del agente de visión para el reconocimiento de personas, el cual debe ser capaz de:

1. En caso de aún no tener operador, percatarse cuando un usuario se presenta para convertirse en el guía.
2. Realizar el tracking de la persona.
3. En caso de seguir identificando al usuario en la escena, determinar su posición.
4. Poder determinar en cualquier instante si el usuario está o no en la escena. En caso de no identificar al usuario, determinar si se debe a una oclusión temporal.
5. Poder determinar en cualquier instante si el usuario está realizando un gesto específico.

A continuación se describirá la manera en que funciona el agente de visión que se construyó, el cual es capaz de resolver la lista de especificaciones anteriores. Se recomienda consultar la Figura 5.2 para complementar la explicación. Cuando en los modelos de diálogo se presenta una situación del tipo ver, el agente del manejador de diálogo envía al Facilitador de OAA una petición que incluye como parámetro la lista de las *n* expectativas de la situación actual. El agente de visión para el reconocimiento de personas está diseñado para poder resolver estas peticiones, así que el agente Facilitador lo elegirá como el encargado de resolver la petición en cuestión. En la Figura 5.2 el Facilitador de OAA no se muestra explícitamente, pero debe quedar claro que la interacción entre el agente del manejador de diálogo y el agente de visión está siendo administrada por él.

Una vez que el agente de visión recibe la petición con la lista de expectativas, tiene que elegir una de ellas (la que se cumple) e indicar su decisión al manejador de diálogo mediante un mensaje. La expectativa elegida por el agente de visión es la interpretación que le está dando a la imagen proveniente de un proceso encargado del reconocimiento de personas en la escena real. El criterio de decisión referente a la expectativa que se cumple no es trivial y se explica a detalle más adelante. Además, es importante señalar que esta decisión no es instantánea, sino que el agente puede esperar cierto tiempo antes de enviar su interpretación al agente del manejador de diálogo.

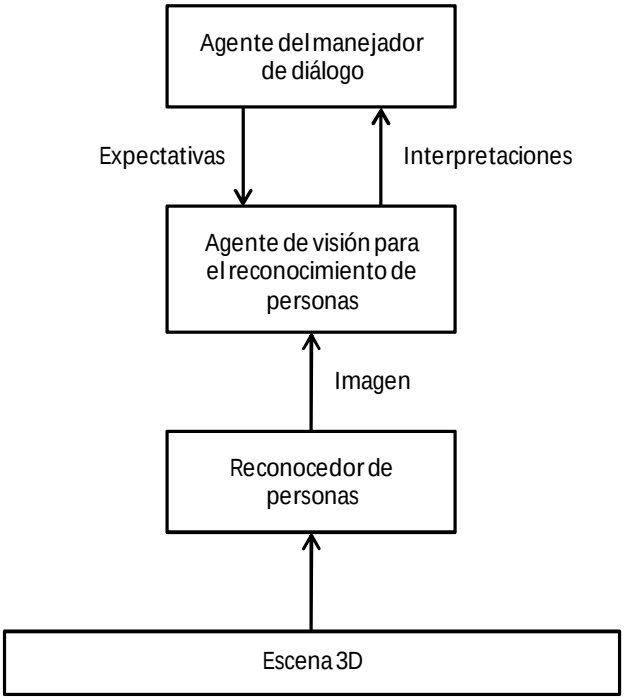

Figura 5.2 Interacción del agente de visión con otros procesos del sistema

Si se compara la Figura 5.2 con la Figura 3.6 (IOCA con capacidades reactivas) salta a la vista la correspondencia entre los módulos de la arquitectura cognitiva y los agentes y procesos involucrados en la implementación computacional del sistema, donde el reconocedor de personas corresponde al módulo de Reconocimiento, el agente de visión corresponde al módulo de Interpretación, y finalmente, el agente del manejador de diálogo corresponde al módulo de Modelos de diálogo.

Para ejemplificar la interacción entre el agente de visión con los demás módulos se supondrá que el robot está en la situación *v* del submodelo de diálogo Rastreando a lo lejos (Figura 4.4), en la cual el robot está siguiendo al usuario *N* y se tienen como posibles expectativas que el usuario haga un gesto al robot para indicarle que se detenga (*gesto_detente(N)*) o que el robot pierda de vista al usuario (*no_veo(N)*). El agente del manejador de diálogo envía una petición al agente de visión que incluye una lista con ambas expectativas. Cuando el agente de visión recibe la lista, verifica los elementos que contiene y analiza la imagen actual I_t proveniente del reconocedor de personas. El análisis de la imagen I_t es acorde a la lista de expectativas recibidas, y en este caso se pueden generar tres posibles respuestas:

1. El usuario desapareció (que corresponde a la expectativa *no_veo(N)*).
2. El usuario está haciendo el gesto para detener al robot (que corresponde a la expectativa *gesto_detente(N)*).
3. Ninguna de las anteriores (es decir, se sigue viendo al usuario y no está haciendo el gesto para detener al robot).

Es importante señalar que estas respuestas son mutuamente excluyentes, por lo que el sistema sólo puede obtener una de ellas al analizar la imagen I_t. (1) y (2) se excluyen mutuamente, ya que si no se está viendo al usuario, es imposible que se detecte que está haciendo un gesto. (1) y (3) lo son porque no tiene sentido no ver al usuario y al mismo tiempo verlo. En forma similar, (2) y (3) no cobran sentido pues en una el usuario está haciendo el gesto y en la otra no.

Si se obtiene la primera o la segunda respuesta, el análisis termina y se envía un mensaje al agente del manejador de diálogo que contiene la correspondiente interpretación de la imagen (*no_veo(id)* o *gesto_detente(id)*), donde la variable *N* ha dejado de ser libre y ahora tiene asignado el valor *id*, que corresponde al identificador del usuario que se está siguiendo. Pero si la respuesta es la tercera, que significaría que ninguna de las expectativas de la situación se ha satisfecho, será necesario analizar la imagen siguiente I_{t+1}. Este análisis se repetirá *k* veces hasta que en la imagen I_{t+k} una de las expectativas sea satisfecha.

5.1.1 Reconocedor de personas

El programa encargado de generar las imágenes que utiliza el agente de visión está codificado en lenguaje C++ y se auxilia de *Open Natural Interaction* (OpenNI)[15]. OpenNI es un framework de código abierto que define APIs[16] para escribir aplicaciones que utilicen Interacción Natural. El término Interacción Natural se refiere a una interacción en la que el humano utiliza voz y comandos gestuales para dar instrucciones.

La interacción de OpenNI con otros elementos del sistema se ilustra en la Figura 5.3, en donde se pueden observar tres capas. La capa inferior corresponde a los dispositivos de hardware que funcionan como sensores, pudiendo ser sensores 3D (tridimensionales), cámaras RGB (video en color), cámaras IR (infrarrojas) o dispositivos de audio (un micrófono o un arreglo de micrófonos). La capa intermedia corresponde a OpenNI, y provee las interfaces de comunicación que interactúan con los sensores y los componentes middleware[17]. Finalmente, la capa superior corresponde a software que implementa aplicaciones haciendo uso de OpenNI.

OpenNI tiene cuatro componentes middleware, los cuales son:

- Middleware para el análisis de cuerpo completo.- identifica cuerpos humanos en la escena y genera información de ellos (centro de masa, orientación, identificación de puntos específicos del cuerpo como la cabeza, el cuello, los hombros, etc.).
- Middleware para el análisis de puntos de la mano.- se encarga de localizar las manos de las personas e indicar mediante puntos el centro de la palma o las puntas de los dedos.
- Middleware para la detección de gestos.- se encarga de identificar gestos predefinidos (como agitar la mano) y alertar a la aplicación cuando suceden.
- Middleware para el análisis de la escena.- analiza la escena y permite separar el foreground (objetos de la escena) del background (fondo de la escena). También permite obtener coordenadas específicas e identificar en forma individual figuras de la escena.

[15] http://www.openni.org/

[16] Las APIs (Application Programming Interfaces) proveen una abstracción de un problema y especifican cómo los usuarios deben interactuar con los componentes de software que implementan la solución a ese problema. Estos componentes se encuentran generalmente distribuidos como una librería de software, permitiendo que sean usados en múltiples aplicaciones (Reddy, 2011).

[17] Funcionalmente, el término middleware abarca a todo aquel software que habilita la interconectividad y operatividad de aplicaciones, sistemas y dispositivos. El middleware se puede ver como una capa de abstracción entre el sistema operativo y las aplicaciones (Lerner, Vanecek, Vidovic y Vrsalovic, 2000).

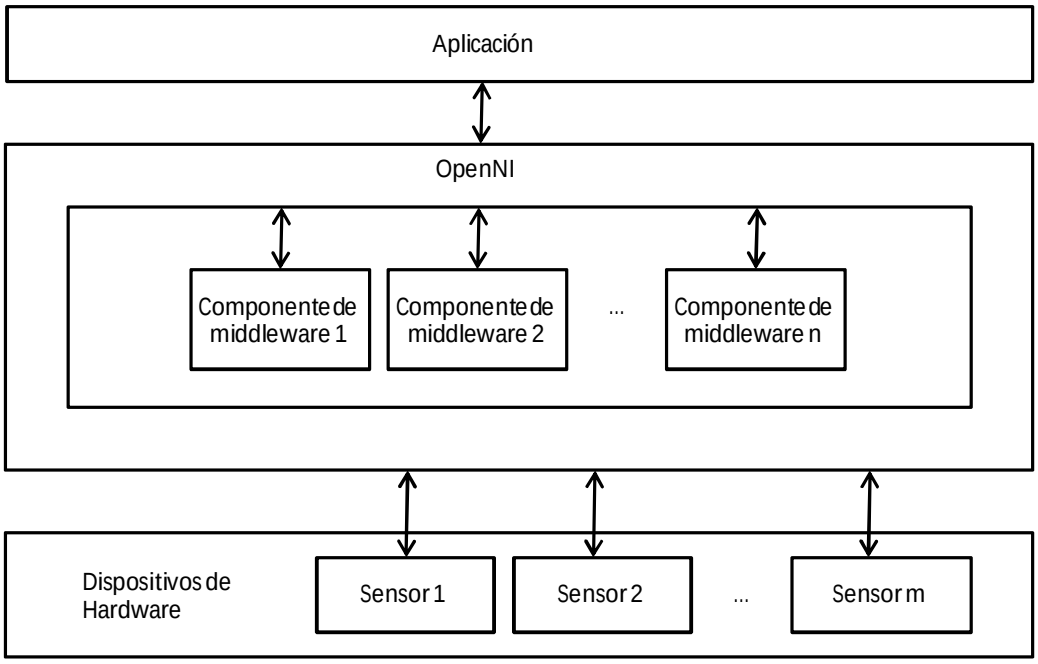

Figura 5.3 Funcionamiento de OpenNI

El funcionamiento de OpenNI da inicio obteniendo datos de la escena mediante sus dispositivos de hardware. Dependiendo de la aplicación que se le vaya a dar, esta información se puede almacenar en los siguientes formatos:

- Mapa de color.- Es una imagen digital en donde cada pixel está representado por un valor RGB.
- Mapa de profundidad.- Es una imagen digital en donde cada pixel está representado por la distancia al sensor.
- Mapa infrarrojo.- Es una imagen digital en donde cada pixel representa el brillo de ese pixel en escala de grises.

Posteriormente, el middleware se utiliza para procesar los datos obtenidos de los sensores en alguno de los formatos anteriores y obtener información de nivel más alto, que puede ser entendida y utilizada por la aplicación.

OpenNI trabaja utilizando nodos de producción, que son conjuntos de componentes que tienen un rol productivo en el proceso de creación de datos. Cada nodo de producción encapsula la funcionalidad que se relaciona con la generación de un tipo específico de dato, y puede proveer este dato a la aplicación o a otro nodo de producción.

Entre las aplicaciones de código abierto que provee OpenNI se encuentra una llamada *user tracker*, cuyo objetivo es realizar el tracking de una figura humana en tercera dimensión. Para complementar la siguiente explicación consultar la Figura 5.4. El primer paso consiste en un nodo

de producción de OpenNI llamado *depth generator*, que se encarga de leer periódicamente los datos de un sensor de profundidad y generar un mapa con la información obtenida.

Un segundo nodo de producción de OpenNI llamado *user generator* lee constantemente los mapas de profundidad que el nodo de producción *depth generator* le proporciona, y los analiza para generar como resultado datos acerca de cuerpos humanos en la escena. Este nodo de producción está en espera de detectar objetos en el mapa de profundidad cuya forma corresponda a la de un cuerpo humano. Cuando se detecta a un posible cuerpo humano, el nodo de producción le asigna un identificador numérico único y lo considerará a partir de ese momento como un usuario.

Una vez que un usuario ha sido detectado, el nodo de producción realiza el tracking de esa persona. Si en el mapa de profundidad M_t el nodo *user generator* tiene detectados n usuarios, al recibir el siguiente mapa de profundidad M_{t+1}, se buscará la nueva posición de cada uno de esos n usuarios. Para cada usuario U_i presente en M_t, se realiza una búsqueda en M_{t+1} para encontrar cuál de los objetos presentes en el nuevo mapa corresponde a U_i. Es posible que del instante t al instante $t+1$ el usuario U_i haya realizado algún movimiento, por ejemplo, moverse a la derecha, moverse a la izquierda, acercarse, alejarse, agacharse un poco, levantar los brazos, etc., lo que implica que en M_{t+1} se debe buscar al objeto que se parezca más a la descripción de U_i en M_t. Si la forma de U_i en M_t es muy diferente de la forma que toma en M_{t+1} es posible que ocurran errores de identificación. En la Figura 5.5 se muestra un pequeño ejemplo que ilustra el tracking realizado por el nodo *user generator* en una secuencia de mapas de profundidad.

Se debe aclarar que en el ejemplo anterior, el tracking de los usuarios se explica en forma cualitativa (se movió a la derecha, se alejó, etc.), pero en realidad este tracking se realiza en forma cuantitativa mediante las coordenadas del centro de masa del usuario. El centro de masa para un humano erguido y con los brazos a los costados se encuentra aproximadamente en un punto interno del sistema digestivo a la altura del ombligo.

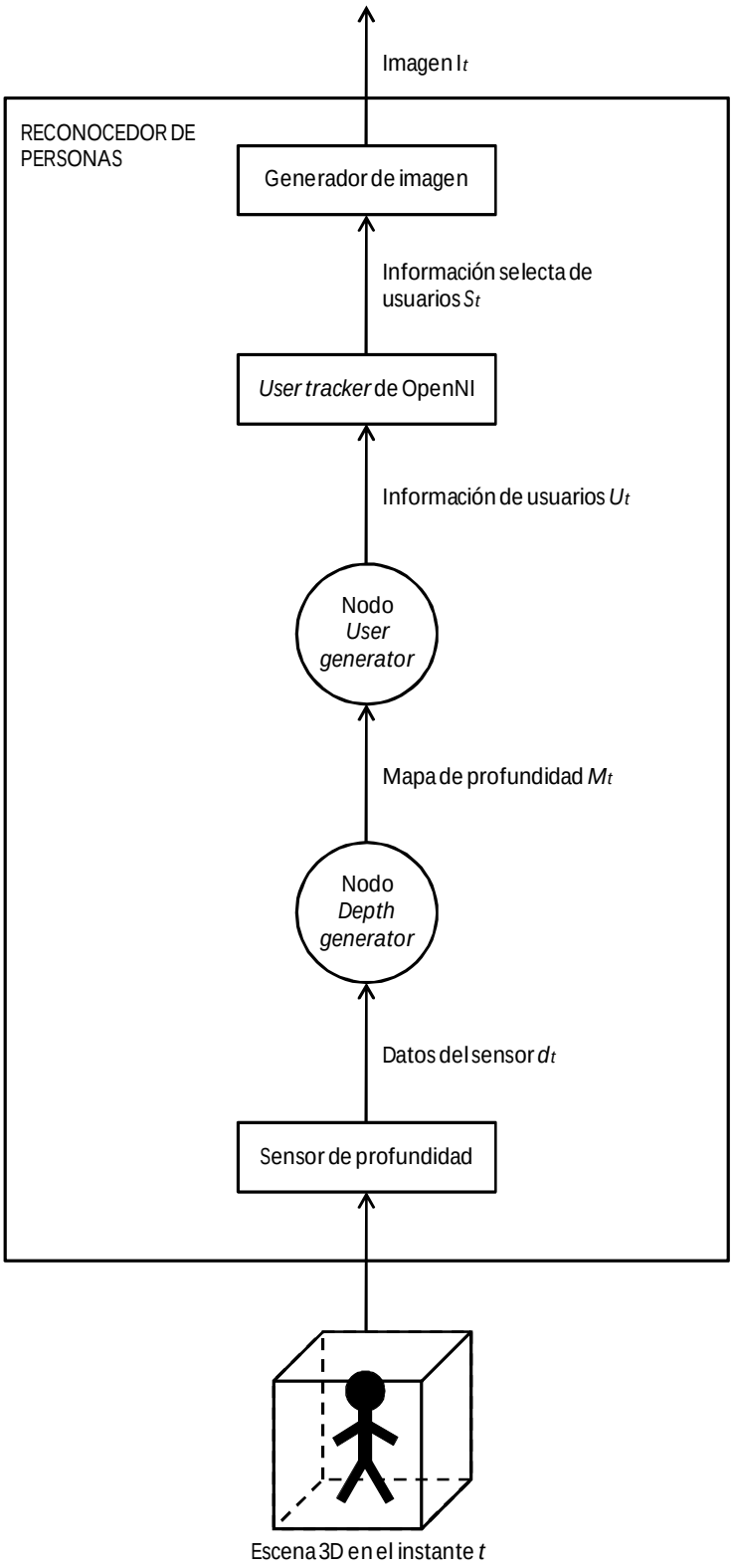

Imagen 5.4 Funcionamiento del reconocedor de personas

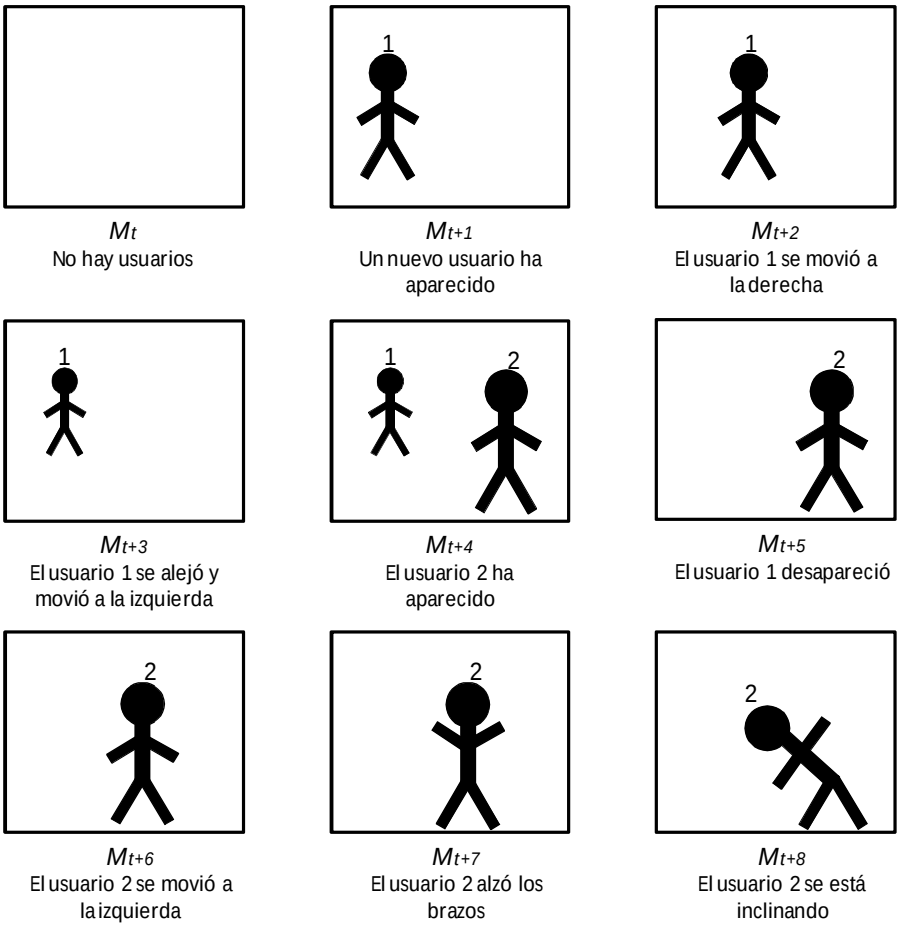

Figura 5.5 Tracking de personas realizado por el nodo *user generator*

El tracking de usuarios realizado por el nodo de producción es muy robusto, y permite seguir el rastro del usuario aún cuando éste cambie de posición (se agache, se pare de manos), sea ocluido por otro objeto de la escena, e incluso permite que el usuario salga temporalmente de la escena. Sin embargo, es posible que ocurran los siguientes errores de tracking:

- Perder a un usuario.- Esto puede suceder cuando el usuario realizó un movimiento muy brusco, o cuando en la escena ocurren demasiados cambios (por ejemplo, un cambio drástico en el background). Cuando esto sucede, el nodo de producción envía un mensaje a la aplicación indicando que ha perdido a determinado usuario.
- Confundir a un usuario.- Es posible que cuando se está siguiendo a un usuario, en algún momento se asigne su identificador en forma equivocada a otro objeto de la escena. Por ejemplo, en ocasiones cuando dos usuarios se cruzan y uno de ellos es ocluido por el otro, es posible que se intercambien los identificadores numéricos de los usuarios. Otro ejemplo es cuando un usuario se acerca demasiado a un objeto,

como un muro, y el nodo le asigna el identificador al objeto, dejando de seguir al verdadero usuario.

- Identificar a un falso usuario.- Ocurre cuando hay objetos cuyas dimensiones son similares a las de un humano, y el nodo piensa que se trata de un humano. Por ejemplo, un robot, una columna, un ventilador, etc.

Además de realizar el tracking de los usuarios, el nodo de producción analiza en cada escena la pose del usuario. Cuando un usuario realiza una pose especial, llamada pose Ψ (*psi*), que se ilustra en la Figura 5.6, el nodo asume que el usuario desea que haga un análisis de su esqueleto. A partir del momento en que el nodo detecta la pose Ψ en un usuario, se realizará un tracking no sólo de su centro de masa, sino de los puntos clave de su cuerpo. Estos puntos se enlistan a continuación y se muestran también en la Figura 5.6:

- Cabeza (*Ca*)
- Cuello (*Cu)*
- Hombro izquierdo (*Hi*)
- Hombro derecho (*Hd*)
- Codo izquierdo (*Coi*)
- Codo derecho (*Cod*)
- Mano izquierda (*Mi*)
- Mano derecha (*Md*)
- Abdomen (*Ab*)
- Cadera izquierda (*Cai*)
- Cadera derecha (*Cad*)
- Rodilla izquierda (*Ri*)
- Rodilla derecha (*Rd*)
- Pie izquierdo (*Pi*)
- Pie derecho (*Pd*)

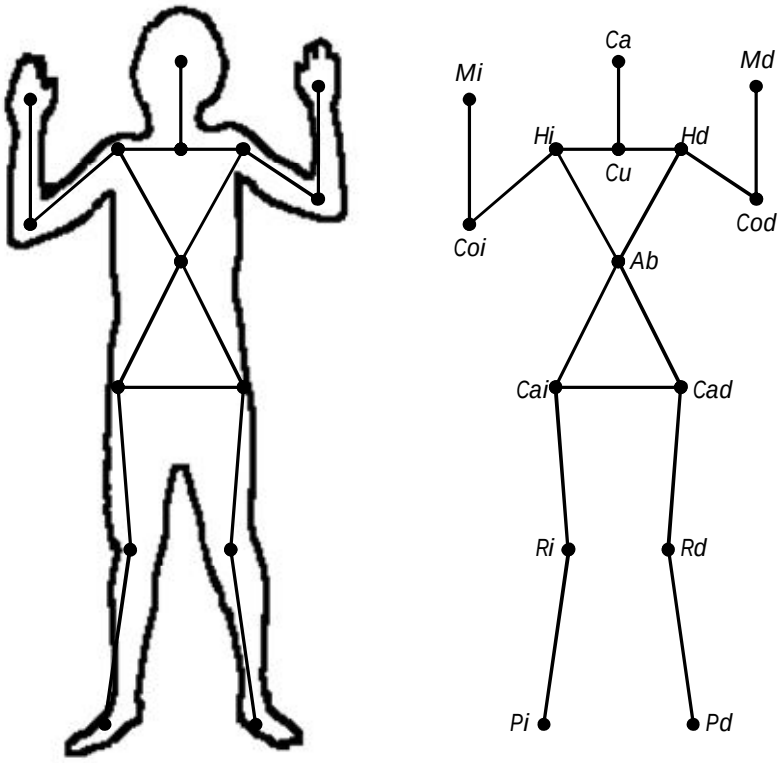

Figura 5.6 Pose Ψ y puntos claves del cuerpo en el esqueleto generado por OpenNI

Cada uno de los puntos anteriores tiene asociado una coordenada en el espacio tridimensional, correspondiente a su ubicación en la escena 3D. En el sistema de coordenadas rectangulares utilizado por el *user tracker* de OpenNI, el origen *O* corresponde al lugar donde se encuentra el sensor de profundidad, el eje horizontal es *X*, el eje vertical es *Y*, y la profundidad corresponde al eje *Z*. En la Figura 5.7 se muestra un ejemplo en donde el centro de masa *C* del usuario tiene como coordenada (x_1 , y_1, z_1), *d* es la distancia entre *O* y *C*, y *Θ* es el ángulo director con respecto al eje *Z* que forma la recta que pasa por los puntos *O* y *C*.

No toda la información que tiene el *user tracker* es necesaria para el agente de visión. El generador de imagen es un programa que se encarga de enviar en forma concreta la información que los agentes del sistema necesitan. En la siguiente sección se hablará sobre la forma en que lo hace.

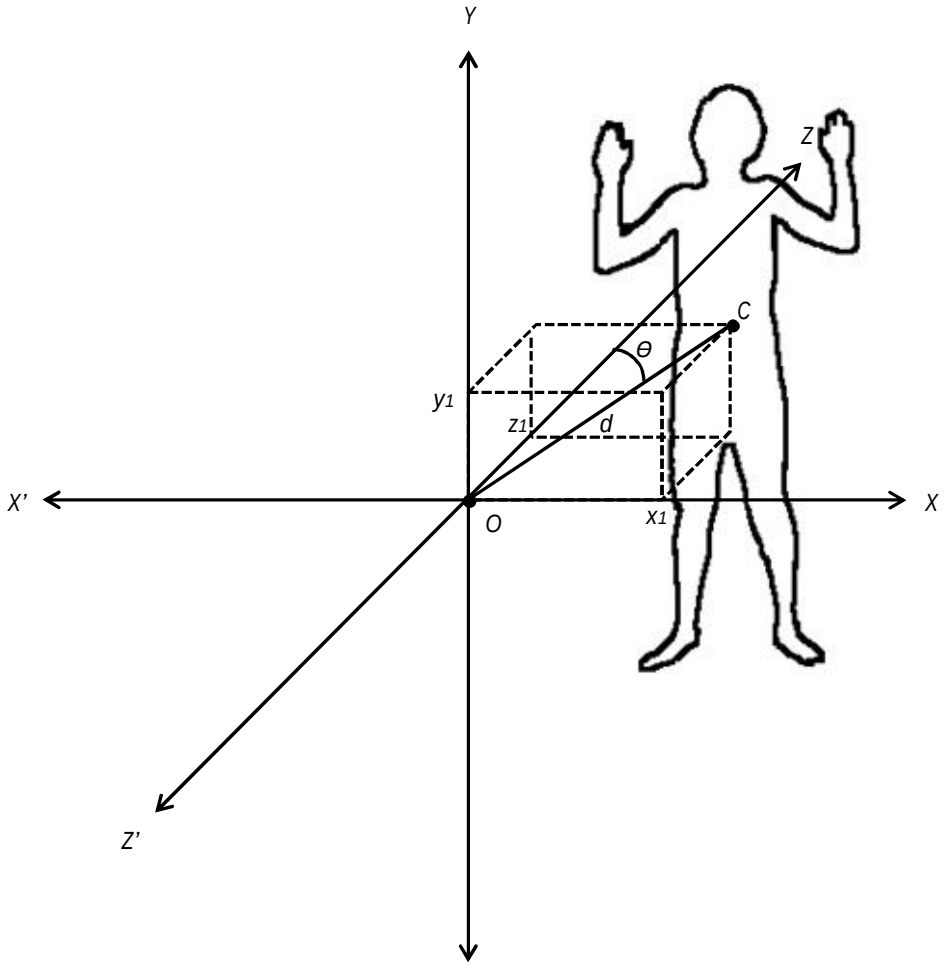

Figura 5.7 Sistema de coordenadas rectangulares utilizado por el *user tracker*

5.1.2 Generación de la imagen I_t

Como se mencionó al final de la sección anterior, el reconocedor de personas tiene que generar la imagen que será enviada a los agentes del sistema. Tanto el agente de visión como el agente de navegación requieren información contenida en esta imagen.

La imagen I_t es un vector formado por los siguientes elementos:

- f, es el identificador numérico del usuario al que se está siguiendo.
- n, es el número de usuarios presentes en la escena.
- d, es la distancia de O a C_f, donde C_f es el centro de masa del usuario f.
- Θ, es el ángulo director con respecto al eje Z que forma la recta que pasa por O y C_f.
- g, es una bandera que indica si el usuario f está haciendo un gesto particular.
- x_1 , y_1 , z_1 , … , x_n , y_n , z_n , son los centros de masa de los n usuarios de la escena.

La decisión de cuál de los usuarios es al que se debe seguir consiste en una estrategia derivada de la descripción de la tarea *Follow Me*. De acuerdo con el manual de reglas, existe una fase de la prueba en la que el operador se debe presentar con el robot. En esta fase, el robot tiene oportunidad de pedirle al usuario que ejecute determinadas acciones con el fin de conocerlo. La estrategia consiste en que el robot solicitará al usuario que se pare justo enfrente de él a una distancia de seis pies. Una vez que el usuario está en la posición correcta, el sistema detecta que se trata de una persona. Sin embargo, esto no es suficiente, pues es posible que haya más personas alrededor de la zona donde se ejecutará la prueba, o puede ser que algún objeto del ambiente sea detectado en forma errónea como una persona. Por esta razón, el robot le solicita al usuario que haga un gesto.

El gesto que se le pide al usuario consiste en alzar las manos, y manteniendo una flexión de los brazos similar a la pose Ψ, mover los antebrazos en forma simétrica, primero hacia afuera y luego hacia adentro, como si estuviera diciendo "hola" al robot. Este gesto se ilustra en la Figura 5.8. El objetivo de pedirle ese gesto al usuario es que la pose Ψ sea detectada en algún momento del gesto, y entonces se obtenga el esqueleto del usuario. Detectar la pose Ψ no es algo instantáneo, y requiere que el usuario haga la pose de la manera correcta. Por esta razón, pedirle al usuario que agite sus manos como si dijera "hola", provoca que el usuario haga la misma pose con ligeras variaciones en cada uno de los mapas de profundidad, por lo que la rapidez de detectar el esqueleto aumenta en comparación de un usuario que se mantiene inmóvil haciendo la pose Ψ.

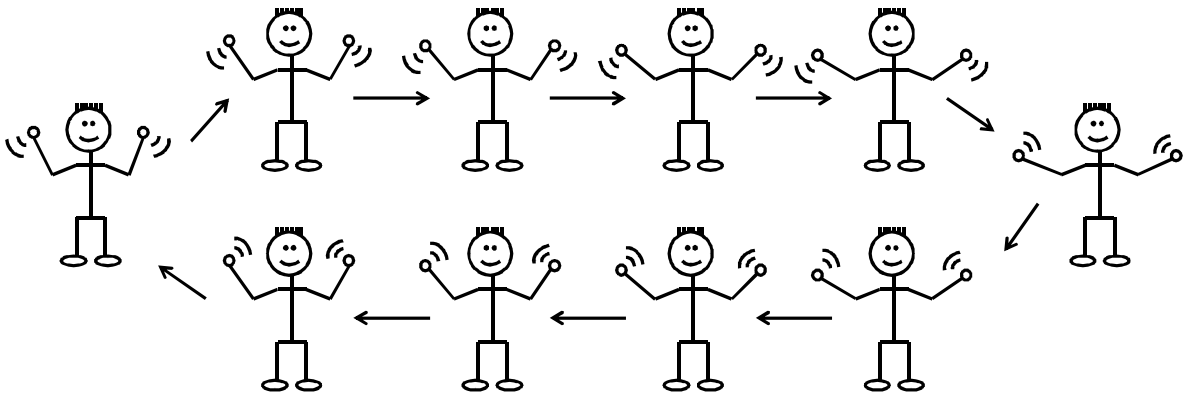

Figura 5.8 Gesto Hola Ψ

Una vez que el esqueleto de un usuario ha sido identificado, su identificador numérico se almacena en *f*, por lo que se le considerará la persona a la que debe seguir el robot. Hacer esto tiene dos grandes ventajas:

1. En el momento en que el robot debe conocer al usuario, estará en espera de reconocer la pose Ψ. Aunque haya más personas en la escena, ninguno de ellos estará haciendo el gesto, por lo que es imposible que el sistema se equivoque y empiece a seguir a otra persona. Si otro de los usuarios de la escena está haciendo la pose y provoca que el robot la identifique a ella como la persona a la que debe seguir, se puede pedir que la prueba reinicie, pues de acuerdo a las reglas de *Follow Me*, ninguna persona debe interferir en el desarrollo de la prueba, y hacer la pose sería algo que claramente interfiere con la calibración del robot.
2. En caso de perder al usuario durante el trayecto, el robot puede pedir ayuda solicitando que el gesto sea repetido nuevamente. Esta ventaja queda fuera de la tarea *Follow Me*, pues no se permiten este tipo de ayudas al robot por parte del operador, sin embargo, en un robot seguidor tener la oportunidad de recuperarse es algo indispensable.

El gesto para pedirle al robot que se detenga consiste en alzar ambas manos. A diferencia del gesto "Hola Ψ", el reconocimiento del gesto "Alto" es más rápido, pues esta vez no se está esperando a identificar una pose específica, sino que se espera a que se cumpla una condición geométrica de algunos puntos clave del esqueleto del usuario. Esta condición consiste en que se cumplan dos cosas: primero, que la altura de la mano izquierda (M_{iy}) sea mayor que la altura del hombro izquierdo (H_{iy}), y segundo, que la altura de la mano derecha (M_{dy}) sea mayor que la altura del hombro derecho (H_{dy}). En la Figura 5.9 se ilustra esta condición.

La condición del gesto "Alto" no es tan rígida como la pose Ψ, sólo basta con que cada mano esté por arriba de su respectivo hombro (Ver Figura 5.10). La desventaja del gesto "Alto" es que se requiere tener el esqueleto del usuario. Por esta razón se da como instrucción al usuario hacer el gesto "Hola Ψ" en caso de detectar que el gesto "Alto" no está funcionando. Cuando el sistema detecta alguno de ambos gestos, pone en verdadero a la bandera *g* de la imagen I_t.

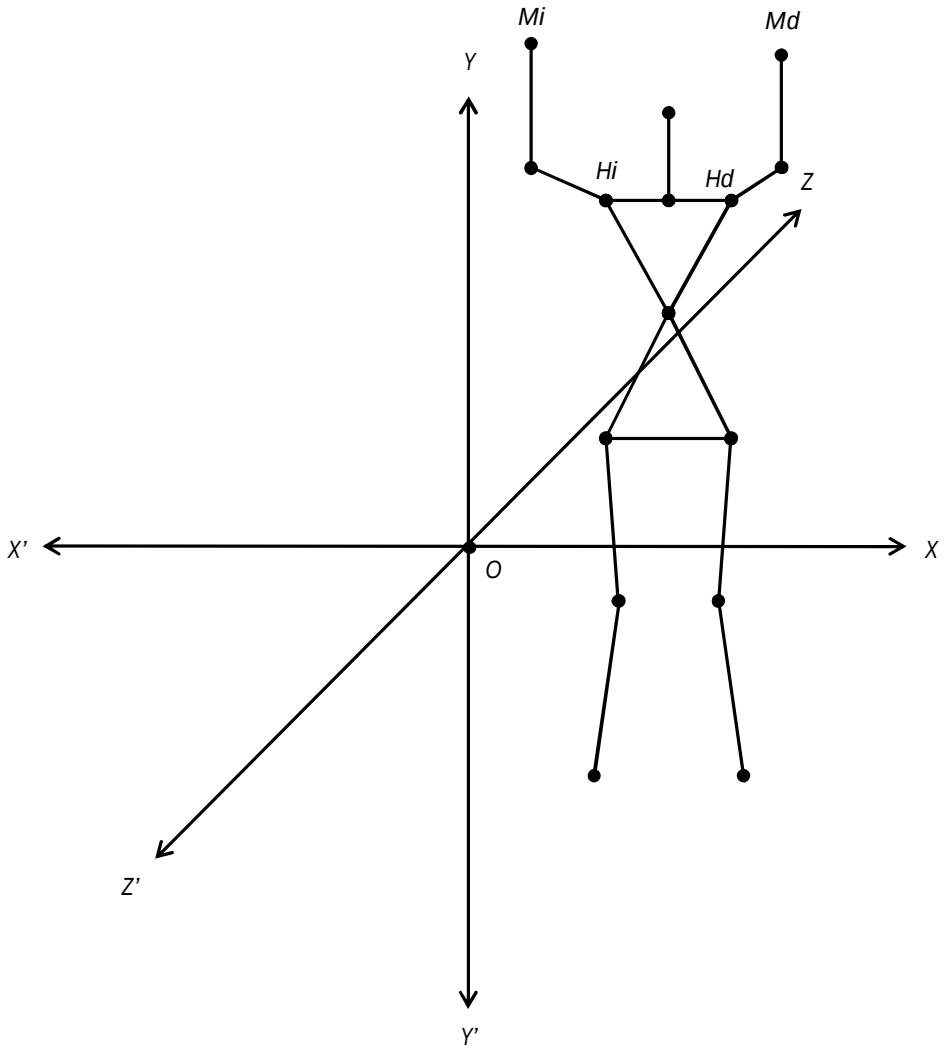

Figura 5.9 Esqueleto de un usuario haciendo la pose “Alto”

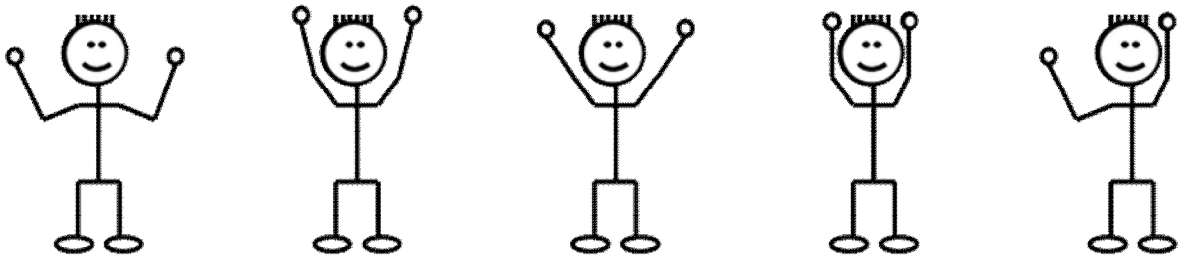

Figura 5.10 Diferentes formas de hacer el gesto Alto

Si el centro de masa del usuario *f* se encuentra en el punto $C\ (x_1, y_1, z_1)$, la distancia *d* del sensor ubicado en $O\ (0, 0, 0)$ al centro de masa (ver Figura 5.7) se calcula fácilmente utilizando la siguiente fórmula:

$d^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2$ *...(1) distancia en el espacio entre dos puntos*

Ya que el segundo punto corresponde al origen, el cálculo se reduce al uso de la siguiente fórmula:

$d^2 = x_1^2 + y_1^2 + z_1^2$ *...(2) distancia en el espacio de un punto al origen*

El ángulo Θ corresponde al ángulo director con respecto al eje *Z* formado por la línea que cruza al origen *O* y al centro de masa *C* del usuario *f* (Ver Figura 5.7). Para calcularlo se utiliza la siguiente fórmula:

$\cos \Theta = (z_1 - z_2) / d$ *...(3) coseno director con respecto al eje Z de una recta en el espacio dados dos puntos*

Nuevamente, como el segundo punto corresponde al origen la fórmula se reduce de la siguiente manera:

$\cos \Theta = z_1 / d$ *...(4) coseno director con respecto al eje Z de una recta en el espacio que pasa por el origen dado uno de sus puntos*

Una vez que se han realizado todos los cálculos, se tienen todos los valores de la imagen I_t, que corresponde al vector $\boldsymbol{I_t} = (f, n, d, \Theta, g, x_1, y_1, z_1, \ldots, x_n, y_n, z_n)$, el cual es enviado al agente de visión para el reconocimiento de personas y al agente de navegación.

5.1.3 Esquema general de decisión del agente de visión

Como se explicó al inicio de la sección 5.1, el agente de visión recibe la lista de expectativas de una determinada situación por parte del agente del manejador de diálogo y tiene que elegir una de ellas e indicar su decisión mediante un mensaje (Ver Figura 5.2). La toma de esta decisión requiere de dos cosas:

1. La lista de expectativas, ya que ellas determinan las posibles respuestas que puede dar el agente de visión.
2. Las imágenes provenientes del reconocedor de personas.

El sistema de decisión está diseñado bajo un esquema general que permite contemplar todos los posibles casos que se requieren en las situaciones visuales de los modelos de diálogo de la tarea *Follow Me*. Este esquema se muestra en la Figura 5.11.

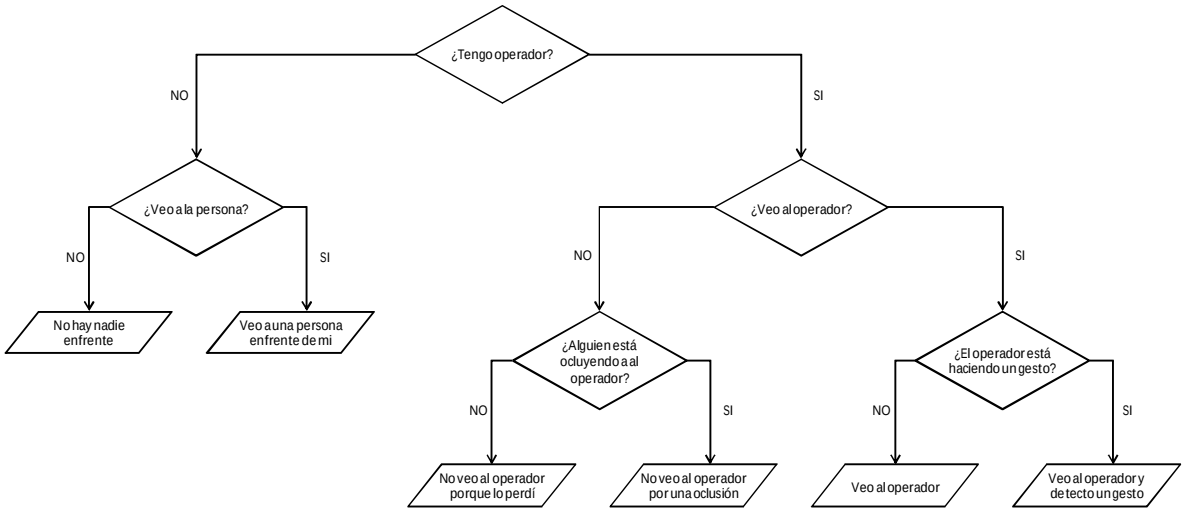

Figura 5.11 Esquema general de decisión del agente de visión

¿Tengo operador?

Lo primero que se tiene que analizar es si ya se tiene un operador, es decir, si alguien ya le solicitó al robot que lo siguiera. Esto se hace verificando el valor que tiene *f* en I_t. En caso de que el valor de *f* sea igual a cero, significa que nadie se ha presentado con el robot, y por lo tanto el sistema está en espera de conocer a una persona. Por otra parte, si el valor de *f* es diferente de cero, significa que se tiene el identificador numérico de un usuario a quien se está siguiendo, y por lo tanto el sistema está en espera de reconocer a una persona.

¿Veo a la persona?

En caso de que se tenga que conocer a una persona, se tiene que preguntar si se está viendo a una persona. Nuevamente se verifica el valor de *f*, en caso de que sea cero otra vez significa que no hay nadie en la escena cuyo esqueleto haya sido detectado. Si el valor de *f* es diferente de cero en esta segunda verificación significa que el esqueleto de una persona ha sido identificado y por lo tanto esa es la persona a la que el robot debe seguir.

¿Veo al operador?

En caso de que se tenga que reconocer a una persona (es decir, cuando ya se conoce al operador), se debe preguntar si el sistema ve al operador. Ver al operador implica que se saben las coordenadas de su centro de masa. Esta información también está disponible en la imagen I_t. Si alguno de los valores del centro de masa del usuario *f* no se encuentran dentro del rango de la Tabla 5.1, significa que hay un error en el tracking del centro de masa del usuario, y por lo tanto no

se está viendo al usuario. La elección del rango de valores permitidos depende de la capacidad de detección del sensor de profundidad específico que se vaya a utilizar para la implementación.

Coordenada	Valor mínimo (milímetros)	Valor máximo (milímetros)
x_f	-5000	5000
y_f	-5000	5000
z_f	1	5000

Tabla 5.1 Rango de valores válidos para el centro de masa del usuario *f*

¿Alguien está ocluyendo al operador?

Cuando alguna de las coordenadas del centro de masa del usuario *f* está fuera del rango válido, significa que se ha perdido de vista al usuario. Esto puede ocurrir por dos razones, la primera porque ha ocurrido un error de identificación, y la segunda porque hay un objeto que ocluye al usuario. En este segundo caso, es posible detectar que se trata de una oclusión si el objeto que ocluye a *f* es otra persona. El agente considerará que se trata de una oclusión cuando se cumplen las siguientes condiciones:

1. Se detecta que en el instante *t* el centro de masa del operador *f* se vuelve repentinamente *(0, 0, 0)*, siendo que en el instante *t-1* los valores de las coordenadas del centro de masa de *f* se encontraban dentro del rango válido.
2. Hay otro usuario *p* en la escena en el instante *t* cuyo centro de masa se encuentra más cerca del origen de lo que se encontraba el centro de masa del usuario *f* en el instante *t-1*.
3. El ángulo director con respecto al eje *Z* del usuario *p* en el instante *t* es aproximadamente igual al ángulo director con respecto al eje *Z* del usuario *f* en el instante *t-1*. Estos ángulos se considerarán aproximadamente iguales si la diferencia entre ellos es menor a un umbral de tolerancia μ[18].

En la Figura 5.12 se presenta un ejemplo de una oclusión temporal detectada por el agente de visión. Las imágenes mostradas son las proyecciones del espacio 3D sobre el plano XZ. Supongamos que en el instante *t-q* se está viendo al operador *f*, lo que implica que los valores de las coordenadas de su centro de masa se encuentran dentro del rango válido. En el instante *t-1* todavía se tienen valores de coordenadas válidos para el centro de masa *f*, pero hay un segundo usuario *p* en la escena. En el instante *t*, el usuario *p* cambia de posición, poniéndose entre el sensor de profundidad ubicado en el punto *O* y el centro de masa de *f*, lo que provocará que las coordenadas del centro de masa de *f* se vuelvan *(0, 0, 0)*. Cuando el agente detecta que esto sucede, se calcula la distancia del usuario *p* al sensor *O* en el instante *t*, y si es menor que la

[18] En la implementación computacional del agente, el valor del umbral de tolerancia μ es de dos grados.

distancia del operador *f* al sensor *O* en el instante t-1, entonces se estará cumpliendo la segunda condición para determinar que se trata de una oclusión temporal. Finalmente, se calcula el ángulo director con respecto al eje *Z* del usuario *p* en el instante *t* y se compara con el ángulo director con respecto al eje *Z* del usuario *f* en el instante *t-1*, y ya que $\Theta_p \approx \Theta_f$, entonces la última condición se cumple y el agente identifica que la desaparición del usuario fue a causa de la oclusión por otra persona. El usuario *p* puede permanecer ocluyendo al operador *f* durante varias imágenes, hasta que en el instante *t+r* el usuario *p* se mueve y permite que el sensor identifique de nuevo a *f*, es decir, que su centro de masa nuevamente tenga coordenadas válidas.

¿El operador está haciendo un gesto?

Finalmente, en el esquema general de decisión mostrado en la Figura 5.11, se tiene el caso en el que ya se conoce al operador y el agente lo está viendo. Se hace la pregunta de si el usuario *f* está haciendo un gesto, y para responder esta pregunta bastará con verificar la bandera *g* de la imagen I_t. Si *g* tiene valor de cero, significará que sólo se está viendo al usuario *f*, pero si *g* tiene valor de uno, entonces se está viendo al usuario *f* y además está haciendo un gesto. En la Tabla 5.2 se muestra un resumen puntual de ésta condición junto con las anteriores.

Condición	¿Qué se debe cumplir para que la respuesta sea afirmativa?
¿Tengo operador?	$f \neq 0$
¿Veo a la persona?	$f \neq 0$
¿Veo al operador?	$-5000 \geq x_f \leq 5000$ $-5000 \geq y_f \leq 5000$ $1 \geq z_f \leq 5000$
¿Alguien está ocluyendo al operador?	$-5000 \geq x_f[t-1] \leq 5000$ $-5000 \geq y_f[t-1] \leq 5000$ $1 \geq z_f[t-1] \leq 5000$ $x_f[t] = 0$ $y_f[t] = 0$ $z_f[t] = 0$ $d_p[t] < d_f[t-1]$ $\lvert \Theta_p[t] - \Theta_f[t-1] \rvert < \mu$
¿El operador está haciendo un gesto?	$g = 1$

Tabla 5.2 Resumen de condiciones del esquema general de decisión

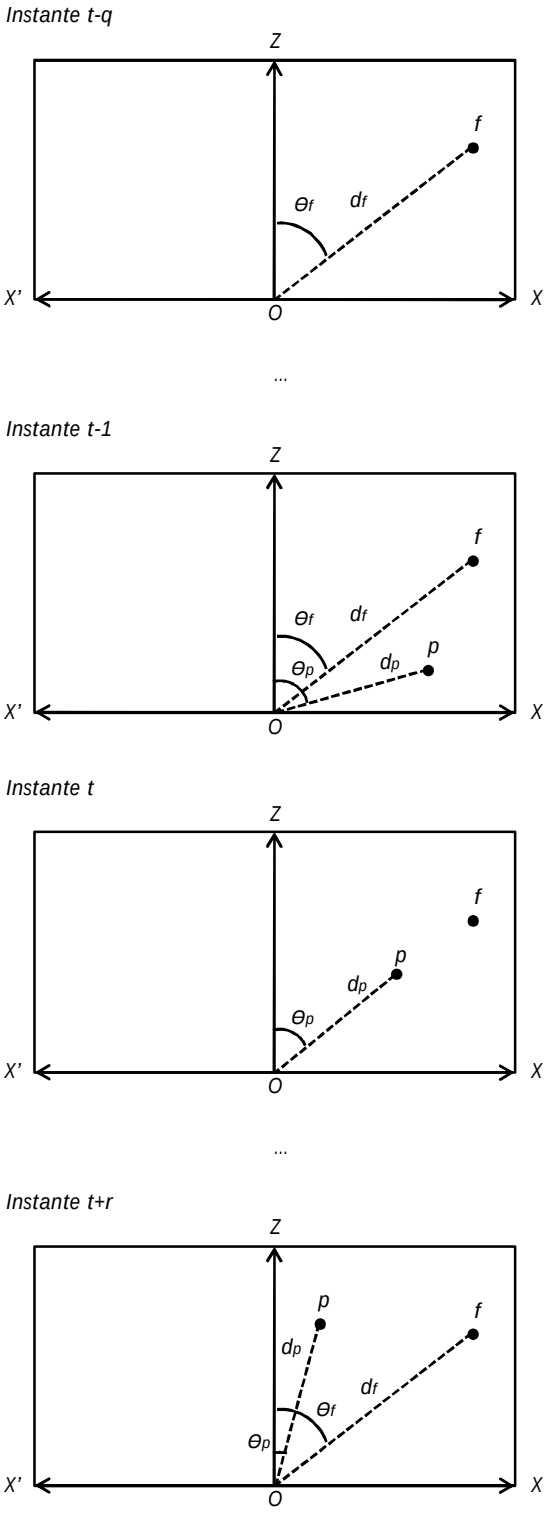

Figura 5.12 Ejemplo de oclusión temporal

Desventajas de la implementación computacional del esquema general de decisión

El esquema general de decisión permite entregar siempre una respuesta, sin embargo existe todavía un gran problema. El agente del manejador de diálogo no está esperando cualquier respuesta, sino que requiere una específica que forme parte de la lista de expectativas que le envió al agente de visión. Entonces, es necesario verificar que la respuesta obtenida forme parte de la lista de expectativas. De no ser así, será necesario esperar la siguiente imagen I_{t+1} y hacer de nuevo el respectivo análisis.

Usar el esquema general de decisión en la implementación computacional implica hacer muchos cálculos. Por ejemplo, si en una situación las expectativas son únicamente ver a la persona (*veo(N)*) o no ver a la persona (*no_veo(N)*), para llegar a la respuesta no sería necesario verificar si alguien está ocluyendo al usuario o si el usuario está haciendo un gesto, pues ello tendría un costo computacional debido a los cálculos que se deben hacer. En este caso bastaría con verificar el centro de masa del usuario *f* y verificar si sus valores están dentro del rango de valores válidos.

El esquema general de decisión descrito no está implementado en el agente de visión como si fuera un módulo, sino que es un modelo que sirve como guía para construir sistemas de decisión específicos a cada lista de expectativas que se presentan en las diferentes situaciones. Este es un esquema de razonamiento basado en casos que simplifica demasiado el número de verificaciones que se deben realizar y aprovecha la lista de expectativas de la situación actual.

5.1.4 Esquemas específicos de decisión del agente de visión

Teniendo como guía el esquema general del proceso de toma de decisión y utilizando de la misma manera en que se explicaron sus bloques de decisión (los rombos de la Figura 5.11), a continuación se explicarán algunos casos representativos que se presentan en los modelos de diálogo de la prueba *Follow Me*.

Caso 1 persona(N)/nadie_enfrente

Este conjunto de expectativas corresponden a la situación *v* del submodelo de diálogo Inicio (Figura 4.2), en donde el sistema aún no ha conocido a ninguna persona y se tienen como expectativas *persona(N)* (es decir, ver a alguien) y *nadie_enfrente* (es decir, no hay ninguna persona enfrente). Cuando el agente visual recibe una lista que contenga a estas expectativas, tiene que decidir entre una de ellas. Como se trata de un caso específico, el agente de visión no tiene que hacer la verificación *¿Tengo operador?* del esquema general de decisión, sino que se asume que no lo conoce. Entonces, la primer verificación que hace es *¿Veo a la persona?*.

Los esquemas específicos de decisión se pueden modificar a conveniencia. Por ejemplo, en este caso no es conveniente que si no se ve a una persona se indique inmediatamente al agente del manejador de diálogo que la expectativa que se satisface es *nadie_enfrente*, ya que en el modelo de diálogo se especifica que al cumplirse la expectativa *nadie_enfrente* se debe enviar un mensaje al usuario indicando que no lo ha detectado. Este mensaje puede convertirse en algo molesto al usuario si lo repite una y otra vez, por lo que es conveniente que el agente de visión dedique cierto tiempo a buscar a la persona antes de decir que no la ve. Por esta razón, se incluye un contador *c* del número de veces que el agente ha verificado si ve a la persona. Si después de c_{max} intentos se sigue sin ver a la persona, entonces el agente indica que la expectativa que se cumple es *nadie_enfrente*[19]. En la Figura 5.13 se muestra el esquema específico del proceso de decisión en forma de diagrama de flujo.

Caso 2 oclusion(N)/no_veo(N)/gesto_detente(N)

En la situación v_1 del submodelo de diálogo Oclusión temporal (Figura 4.3), el robot se encuentra siguiendo a la persona y tiene como expectativas que una segunda persona se cruce entre ellos provocando una oclusión temporal (*oclusion(N)*), perder al usuario por algún error en el sistema de reconocimiento (*no_veo(N)*) o que el usuario le indique que se detenga mediante un gesto (*gesto_detente(N)*). A diferencia del caso anterior, este análisis requiere más elementos de la estructura del esquema general del proceso de toma de decisión.

El primer bloque de decisión es *¿Veo al operador?*. Si la respuesta es negativa, el siguiente bloque de decisión es *¿Alguien está ocluyendo al operador?*. Una respuesta negativa significa que la expectativa *no_veo(N)* se satisface, y una respuesta positiva significa que la expectativa *oclusión(N)* es la que se cumple.

Por otra parte, si en el bloque de decisión *¿Veo al operador?* la respuesta fue positiva, el siguiente bloque de decisión es *¿El operador está haciendo un gesto?*. Si la respuesta es positiva, entonces la expectativa que se satisface es *gesto_detente(N)*. Sin embargo, dado que en este caso ver a la persona no está en la lista de expectativas, el bloque de salida al que conduce la rama negativa del bloque de decisión *¿El operador está haciendo un gesto?* se omite. Sin embargo, si se llega a un resultado negativo en este último bloque de decisión, ninguna expectativa se ha satisfecho, por lo que es necesario volver a repetir el proceso con la información de la siguiente imagen I_{t+1}. En la figura 5.14 se muestra el diagrama de flujo final para este caso.

[19] El número de intentos máximos c_{max} en la implementación del agente está calculado para dar un tiempo aproximado de entre 20 y 30 segundos antes de que el sistema indique que no ha visto al usuario. Este tiempo varía dependiendo de las características de la computadora donde se corra el agente y de la carga de trabajo en el momento en que el agente se ejecuta.

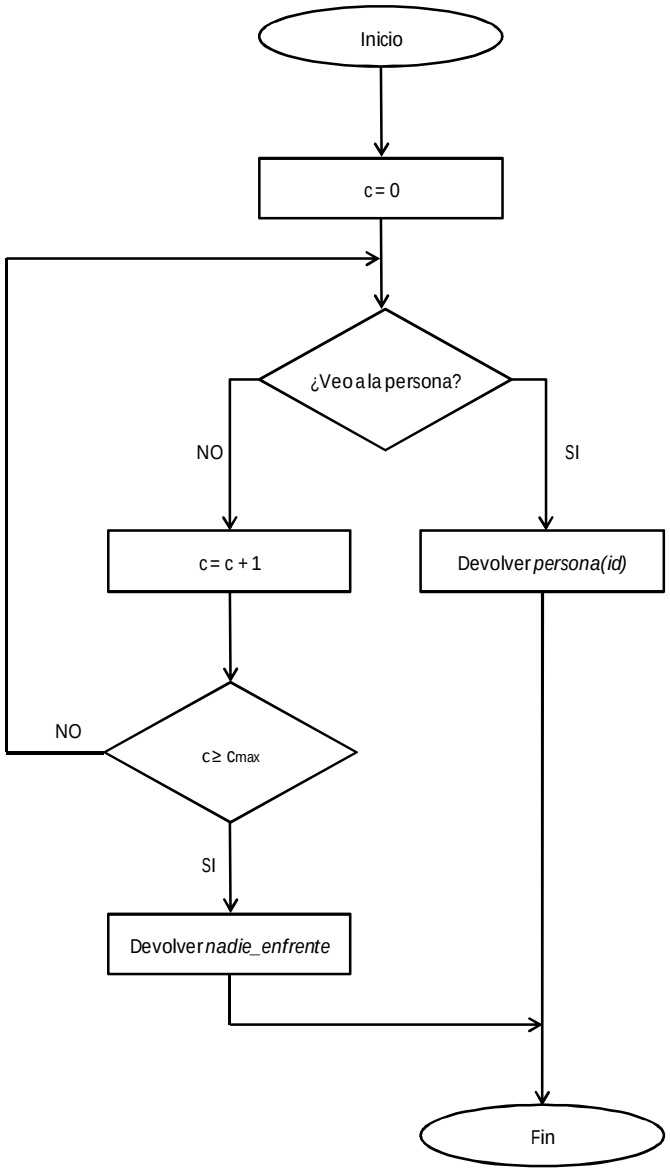

Figura 5.13 Esquema específico de decisión para el caso *persona(N)/nadie_enfrente*

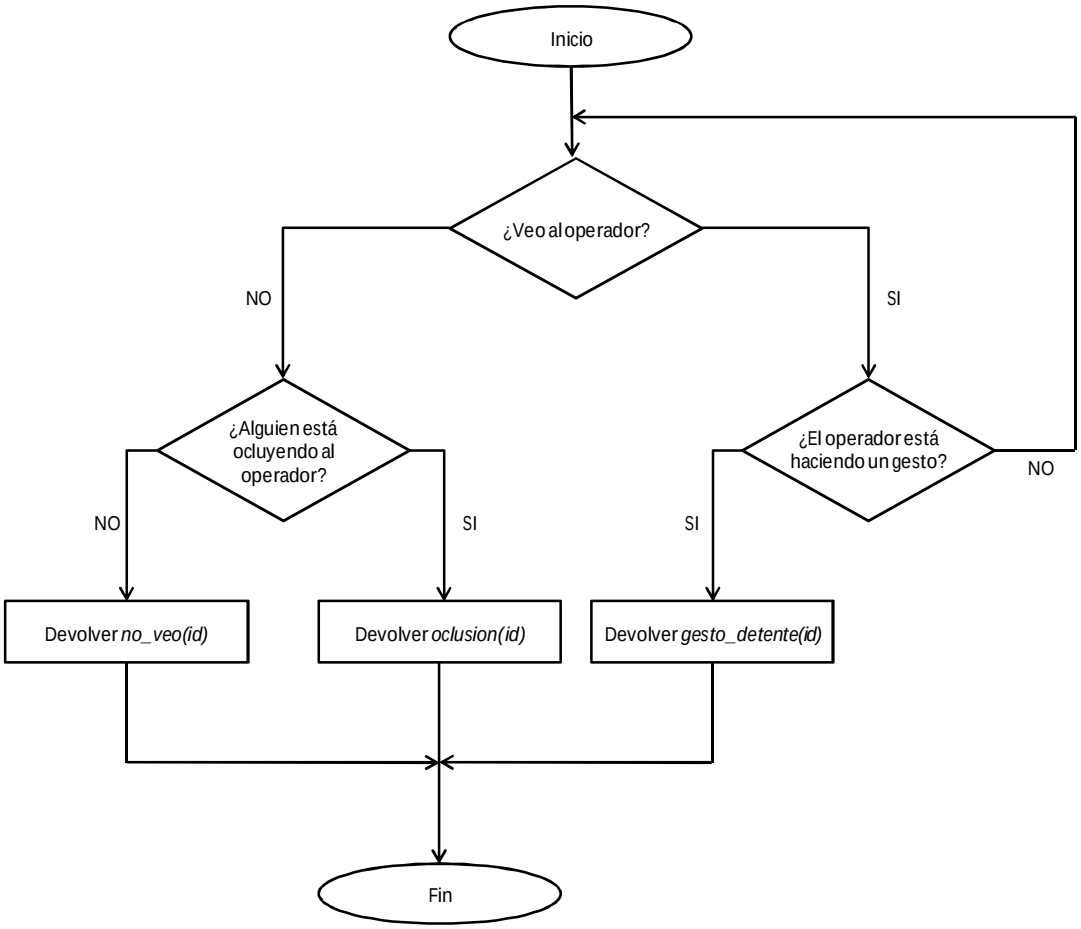

Figura 5.14 Diagrama de flujo para el caso *oclusión(N)/no_veo(N)/gesto_detente(N)*

Caso 3 no_veo(N)/gesto_detente(N)

Esta lista de expectativas se presenta en varias situaciones de los modelos de diálogo de *Follow Me*. Por ejemplo, en la situación v_1 del submodelo de diálogo Reconociendo al usuario (Figura 4.5), el robot se encuentra siguiendo a una persona y se tienen como expectativas que el usuario le pida al robot que se detenga mediante un gesto con sus manos (*gesto_detente(N)*) o que el robot pierda de vista a la persona que está siguiendo (*no_veo(N)*). En caso de perder al usuario, en este momento no importa la razón por la que lo perdió, por esta razón no es necesario verificar si se trata de una oclusión temporal o no.

En este caso, si el bloque de decisión *¿Veo al operador?* tiene una respuesta negativa, automáticamente se considerará que la expectativa que se satisface es *no_veo(N)*. Pero si la respuesta es positiva, entonces se pasará al bloque de decisión *¿El operador está haciendo un gesto?*. Para este último bloque, si la respuesta es positiva, la expectativa satisfecha será *gesto_detente(N)*, pero si la respuesta es negativa se tiene como resultado que ninguna

expectativa se ha satisfecho, y es necesario analizar la siguiente imagen I_{t+1}. El diagrama de flujo para este caso se muestra en la Figura 5.15.

Como se pudo notar con estos tres casos, el esquema general de decisión sirve como una plantilla en la que basta con identificar los bloques de decisión que se requieren para un caso específico, y de esta manera generar la solución de un caso particular.

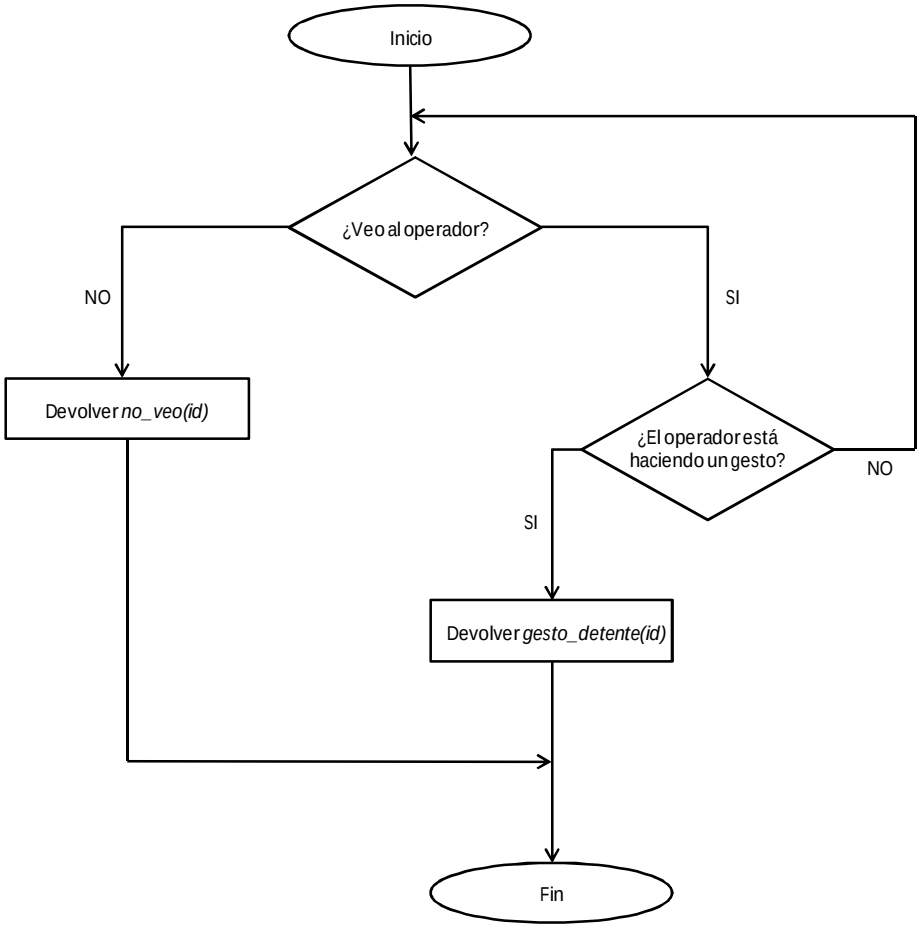

Figura 5.15 Diagrama de flujo para el caso *no_veo(N)/gesto_detente(N)*

5.2 Otros agentes del sistema

En esta sección se pretende explicar en forma general el funcionamiento de los otros agentes que están involucrados en el sistema y que fueron tomados de los proyectos DIME y GOLEM.

Agente de navegación

Este agente de comportamiento se encarga de coordinar los movimientos de los motores para que el robot se desplace hasta una determinada posición. Lo primero que hace es tomar la imagen I_t proveniente del reconocedor de personas y verificar la posición del usuario al que está siguiendo, la cual está determinada por los valores *d* y *Θ* del vector I_t. El agente calcula el punto *P* hasta el cual debe moverse el robot de manera que exista una distancia de separación entre él y la persona. Lo siguiente que hace es planear la ruta que seguirá hasta llegar a *P*, lo cual no necesariamente es una línea recta, ya que debe evitar chocar con los objetos que forman parte del ambiente.

El seguimiento motor de la persona (es decir, caminar detrás de ella) por parte del agente de navegación está funcionando como un Sistema Reactivo Autónomo (ver Figura 3.6), ya que las acciones que realiza no necesitan que se recorra todo el ciclo de interacción principal de la Arquitectura Cognitiva. El Coordinador es un semáforo que habilita o deshabilita al seguimiento de la persona, dependiendo de las acciones provenientes de los modelos de diálogo y que pueden ser *seguir* (habilitar el sistema de seguimiento motor) o *detener* (deshabilitar el sistema de seguimiento motor).

Agente de reconocimiento de voz

Este agente se encarga de interpretar lo que hablan los usuarios. También se encarga de la síntesis de voz. Por esta razón, se trata de un agente de percepción y de comportamiento.

Agente de despliegue de botones gráficos

Este agente se encarga de mostrar un botón en la pantalla (el botón de inicio) y detectar cuando ha sido presionado.

5.3 Detalles técnicos de la implementación

El robot móvil sobre el que se implementó la tarea de seguimiento es llamado "Golem-II+". En esta sección se describirá el hardware y el software que lo integran.

Hardware

- La base es un robot PeopleBot™ (de Mobile Robots Inc.), el cual tiene:
  - Tres arreglos de sonares, cada uno con ocho sensores
  - Dos sensores infrarrojos
  - Un Gripper 2-DOF con break beam
  - Dos break beams verticales
  - Dos arreglos de bumpers protectivos, cada uno con cinco sensores
  - Micrófonos gemelos y bocinas
  - Una Computadora interna VersaLogic EBX-12
- Una Laptop Dell Latitude E6400
- Un Láser Hokuyo UTM-30LX
- Un Kinect Sensor
- Tres micrófonos omnidireccionales Shure Base
- Un micrófono direccional condensador M-Audio
- Una interface de sonido externa M-Audio Fast Track
- Dos bocinas de dos canales de 3.5 pulgadas Infinity

Sofwtare que controla los dispositivos del robot

- Computadoras: el sistema operativo de la computadora interna es Debian Etch 4.0, y el de la computadora externa es Ubuntu 10.04
- Gripper, IR, sonares, bumpers, breakbeams: Librerías de Player/Stage
- Láser: Librerías de Player/Stage y Gearbox
- Kinect: OpenNI y Open Kinect.
- Tarjeta de sonido: M-Audio Fast Track Ultra

Software de los módulos del sistema

- Manejador de diálogo: Sicstus Prolog V. 3.12
- Visión: OpenNI, Open Kinect, OpenCV
- Audio: JACK y PulseAudio
- Reconocimiento de voz: Sphinx 3
- Sintetizador de voz: Festival TTS

- Navegación: Librerías de Player/Stage
- Comunicación entre agentes: OAA
- Agentes: C++, Java

Una vez que se ha explicado la forma y los recursos que se utilizaron para implementar el robot seguidor, es momento de evaluar su funcionamiento. El siguiente capítulo estará destinado a evaluar este sistema de seguimiento de personas sobre un robot móvil.

Capítulo 6

Evaluación del sistema

Una vez que el sistema ha sido implementado en un robot móvil es momento de evaluar su funcionamiento. El problema que surge en este punto es qué se debe evaluar y cómo se debe hacerlo. Se proponen dos formas para evaluar el sistema: evaluar la tarea por partes y evaluar la tarea como un todo. La primera forma permite identificar a detalle las partes de la tarea en donde existe mayor posibilidad de errores con el objetivo de realizar mejoras a futuro. La idea consiste en ejecutar varias veces partes pequeñas de la tarea y registrar en bitácoras de acción si el sistema funciona de manera correcta. En la segunda forma, se pide al usuario que realice la tarea completa y al final se le entrevista, permitiendo de esta manera obtener respuestas globales sobre la experiencia del usuario al utilizar el sistema.

La forma en que se dividirá la tarea para la evaluación corresponde a las habilidades que tiene el robot seguidor implementado. En la sección 6.1 se evalúa la presentación entre el robot y el usuario, en la sección 6.2 la capacidad de resolver oclusiones temporales, en la sección 6.3 qué tan bien puede rastrear un usuario a lo lejos, en la sección 6.4 el reconocimiento del usuario una vez que lo ha perdido de vista, y en la sección 6.5 el entendimiento del robot de que ha llegado a la línea de meta. En la sección 6.6 se presentan los resultados del funcionamiento al probarlo en una competencia real. Finalmente, en la sección 6.7 se evalúa al sistema en forma completa midiendo la satisfacción del usuario tras haber realizado la tarea *Follow Me*.

6.1 Evaluación de la presentación entre usuario y robot

El objetivo de esta prueba es evaluar algunos aspectos de la interacción ocurrida en la fase de presentación entre el robot y el usuario. En esta parte de la tarea, el robot se presenta e indica que está listo para seguir a una persona y se pone en espera de que alguien se acerque y se lo solicite. Cuando el humano le da la instrucción en forma hablada (diciéndole "ven conmigo", "sígueme" o una frase similar), el robot le pide al usuario que se ponga enfrente de él realizando un gesto específico (el gesto "Hola Ψ"). Una vez que el robot identifica a una persona que está realizando ese gesto, le indica que empezará a seguirlo y empieza a caminar detrás de él.

En la Tabla 6.1 se muestra una bitácora de interacción que un evaluador va llenando mientras un usuario está interactuando con el robot en esa fase de la prueba. Se muestran los resultados (frecuencias o promedios dependiendo de la pregunta) después de haber hecho un total de diez experimentos. Ya que los eventos están encadenados, si en una pregunta se obtiene como respuesta No, automáticamente implica que el experimento ha fracasado y se detiene en ese momento. Por esa razón, las frecuencias en las preguntas Sí/No se presentan en el formato *f/e*, donde *f* es la frecuencia de veces que se obtuvo esa respuesta de un total de *e* experimentos, y donde el valor de *e* para la pregunta *n* es el número de veces que se obtuvo un Sí en la pregunta *n-1*.

Estos experimentos, al igual que todos los que se presentan en este capítulo con excepción del presentado en la sección 6.6, se hicieron en un laboratorio escolar en momentos en que había poco ruido ambiental y un nivel de iluminación adecuado (se eligieron horas en las que el sol no incidía directamente sobre el área de trabajo, de modo que la iluminación del lugar era de 500 lx ± 15%).

Como resultado de este experimento se puede señalar que los momentos de la interacción en donde existe mayor posibilidad de error son los relacionados al reconocedor de voz y al reconocedor de personas. En el caso del reconocedor de voz, el usuario tiene que repetir varias veces la instrucción ya que el sistema no entiende en forma correcta lo que se le dice, lo cual se debe a que el usuario en ocasiones no pronuncia el comando en forma correcta o lo hace a una distancia poco adecuada (muy cerca o muy lejos del micrófono). De los diez experimentos realizados, el promedio del número de veces que el usuario tuvo que repetir al sistema el comando hablado fue 3.4 veces.

En el caso del reconocedor de personas, el sistema tarda bastante tiempo en identificar el gesto que está realizando el usuario, lo cual sucede porque en ocasiones la persona no tiene una postura adecuada para realizar el gesto, o no se pone a la distancia que le indicó el robot. De las diez veces que se realizó el experimento, el tiempo promedio que tardó el robot en identificar el gesto "Hola Ψ" del usuario fue de 13.4 segundos.

Pregunta		Sí	No	¿Qué se está midiendo?
1	¿El robot saludó e indicó que estaba esperando a que alguien le dijera "Sígueme"?	10/10	0/10	Desempeño del sintetizador de voz
2	¿El robot entendió el comando hablado que le dio el usuario?	10/10	0/10	Desempeño del reconocedor de voz
3	¿El robot dio las instrucciones al usuario para que se parara enfrente de él e hiciera el gesto "Hola Ψ"?	10/10	0/10	Desempeño del sintetizador de voz
4	¿El robot identificó el gesto "Hola Ψ" del usuario?	10/10	0/10	Desempeño del agente de visión (reconocimiento de gestos)
5	¿El robot anunció al usuario que había identificado su gesto y que estaba listo para seguirlo?	10/10	0/10	Desempeño del sintetizador de voz
6	¿El robot comenzó a caminar detrás del usuario?	9/10	1/10	Desempeño del agente de navegación

Tabla 6.1 Evaluación de la presentación entre el robot y el usuario

6.2 Evaluación de resolución de oclusiones temporales

Esta prueba tiene por objetivo evaluar lo que sucede ante una oclusión temporal. Como condición inicial de cada experimento, el robot y el usuario ya se han presentado y el robot se encuentra siguiéndolo. El robot se encuentra en una situación de seguir que tiene como expectativa una posible oclusión temporal. El usuario caminará en línea recta una distancia pequeña y se detiene. En este momento inicia el experimento.

El robot debe detenerse manteniendo una distancia *d* de aproximadamente 160 centímetros al usuario. En seguida, una segunda persona pasa entre el robot y el usuario, caminando lentamente (con una velocidad aproximada de medio metro por segundo) y cruzando en forma perpendicular el punto medio de la línea formada por el robot y el usuario (ver Figuras 6.1 y 6.2). El robot debe detectar la oclusión temporal e indicarle al usuario que se percató de que una segunda persona está pasando entre ellos. Mientras dice lo anterior, la segunda persona ya se ha alejado lo suficiente y el usuario debe ser identificado de nuevo por el agente de visión. En caso de ser identificado correctamente, el robot le dirá que lo ve nuevamente y que puede empezar a caminar de nuevo, de lo contrario dirá un mensaje diciendo que ya no lo ve. En la Tabla 6.2 se muestran los resultados obtenidos tras realizar diez veces el experimento.

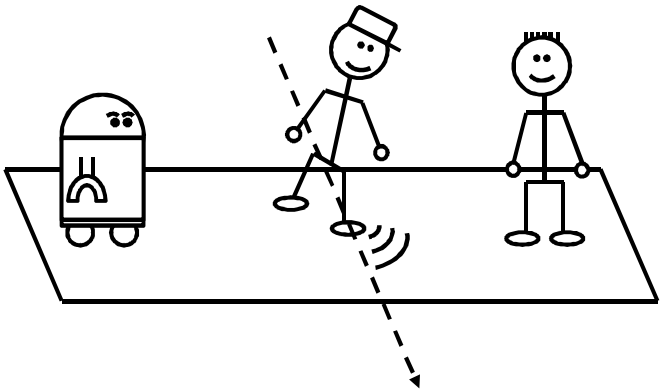

Figura 6.1 Trayectoria de la persona que causa la oclusión temporal

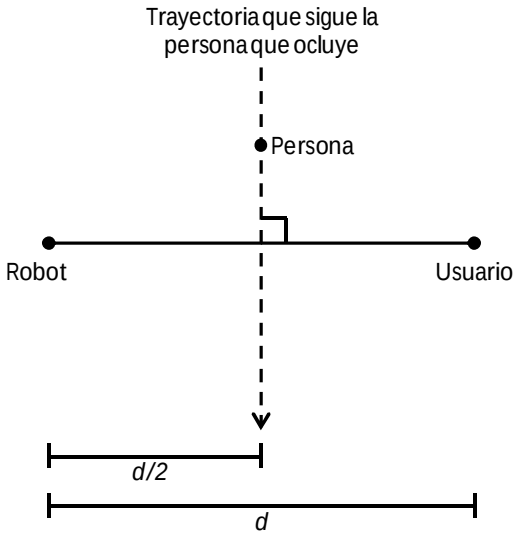

Figura 6.2 Oclusión temporal vista desde arriba

Pregunta		Sí	No	¿Qué se está midiendo?
1	¿El robot se detuvo manteniendo una distancia de 160 centímetros al usuario?	10/10	0/10	Desempeño del agente de navegación
2	¿El robot seguía identificando al usuario después de detenerse?	9/10	1/10	Desempeño del agente de visión (reconocimiento de personas)
3	¿El robot detectó que una segunda persona ocluyó al usuario?	9/9	0/9	Desempeño del agente de visión (detección de oclusiones temporales)
4	¿El robot anunció al usuario que había detectado una oclusión temporal?	9/9	0/9	Desempeño del sintetizador de voz
5	¿El robot identificó de nuevo al usuario?	8/9	1/9	Desempeño del agente de visión (reconocimiento de personas)

Tabla 6.2 Evaluación de la resolución de oclusiones temporales

Los resultados del experimento son satisfactorios, aunque el éxito en identificar de nuevo al usuario (pregunta número 5 de la Tabla 6.2) depende en gran medida de los parámetros elegidos para realizar el experimento, los cuales son la distancia entre el robot y el usuario, el lugar en donde cruza la segunda persona, y la velocidad con que cruza. Consultar la sección 3 del Apéndice B para ver una discusión detallada acerca de estos parámetros.

6.3 Evaluación de la identificación de un usuario a lo lejos

Este experimento tiene como objetivo medir qué tan bien puede resolver el sistema la situación en la que un usuario se aleja demasiado. Como condición inicial de cada experimento, el robot y el usuario ya se han presentado y el robot se encuentra siguiéndolo. El robot se encuentra en una situación de seguir que tiene como expectativa un posible gesto indicándole que se detenga. El usuario caminará en línea recta una distancia pequeña. En este momento inicia el experimento.

El usuario realiza el gesto para pedirle al robot que se detenga. El robot debe identificar el gesto y detenerse manteniendo una distancia de aproximadamente 100 centímetros al usuario (esta vez le conviene estar más cerca del usuario). En seguida, debe avisarle al usuario que esperará en ese lugar sin moverse durante diez segundos. Acto seguido, el usuario debe caminar en línea recta tres metros hacia enfrente. Pasados los diez segundos, el robot debe indicar si identificó al usuario a lo lejos, o si lo perdió de vista. En la Tabla 6.3 se muestran los resultados obtenidos tras ejecutar diez veces este experimento. Los errores en este experimento ocurrieron en el agente de visión, existiendo problemas en el reconocimiento de gestos y personas.

Pregunta		Sí	No	¿Qué se está midiendo?
1	¿El robot identificó el gesto "Alto" del usuario?	9/10	1/10	Desempeño del agente de visión (reconocimiento de gestos)
2	¿El robot se detuvo manteniendo una distancia de 100 centímetros al usuario?	9/9	0/9	Desempeño del agente de navegación
3	¿El robot seguía identificando al usuario después de detenerse?	9/9	0/9	Desempeño del agente de visión (reconocimiento de personas)
4	¿El robot anunció al usuario que esperaría en esa posición durante diez segundos?	9/9	0/9	Desempeño del sintetizador de voz
5	¿El robot identificó de nuevo al usuario?	8/9	1/9	Desempeño del agente de visión (reconocimiento de personas)

Tabla 6.3 Evaluación de la identificación de un usuario a lo lejos

6.4 Evaluación de la identificación de un usuario que salió de la escena temporalmente

En este experimento se pretende medir lo que sucede cuando un usuario sale del campo de visión del robot, espera un tiempo afuera, y regresa a la escena acompañado de otra persona, con el objetivo de probar si el robot es capaz de identificar al usuario original nuevamente. Como condición inicial de cada experimento, el robot y el usuario ya se han presentado y el robot se encuentra siguiéndolo. El robot se encuentra en una situación de seguir que tiene como expectativa un posible gesto indicándole que se detenga. El usuario caminará en línea recta una distancia pequeña. En este momento inicia el experimento.

El usuario realiza el gesto para pedirle al robot que se detenga. El robot debe identificar el gesto y detenerse manteniendo una distancia de aproximadamente 140 centímetros al usuario. En seguida, el robot le avisa al usuario que esperará a que se oculte de él. El usuario debe salir de la escena, y cuando el robot deja de verlo da un aviso por medio de voz. El usuario espera 15 segundos, y regresa a la escena junto con otra persona. Ambas personas se ponen enfrente del robot a una distancia de metro y medio, guardando entre ellas una distancia de un metro. El robot debe identificar cuál de ellos era el usuario correcto. En la Tabla 6.4 se muestran los resultados obtenidos tras ejecutar diez veces este experimento. Los errores en este experimento se dan nuevamente cuando el sistema trata de identificar al usuario, sin embargo cabe añadir como dato interesante que en todos los casos en que se equivocó en la identificación del usuario, el sistema consideró que el usuario original nunca regresó a la escena, y la equivocación no se debió a identificar erróneamente a la otra persona como el usuario original.

Pregunta		Sí	No	¿Qué se está midiendo?
1	¿El robot identificó el gesto "Alto" del usuario?	9/10	1/10	Desempeño del agente de visión (reconocimiento de gestos)
2	¿El robot se detuvo manteniendo una distancia de 140 centímetros al usuario?	9/9	0/9	Desempeño del agente de navegación
3	¿El robot seguía identificando al usuario después de detenerse?	8/9	1/9	Desempeño del agente de visión (reconocimiento de personas)
4	¿El robot anunció al usuario que iba a esperar a que se ocultara?	8/8	0/8	Desempeño del sintetizador de voz
5	¿El robot se percató cuando el usuario salió de su campo de visión?	8/8	0/8	Desempeño del agente de visión (reconocimiento de personas)
6	¿El robot anunció que ya no veía al usuario?	8/8	0/8	Desempeño del sintetizador de voz
7	¿El robot identificó de nuevo al usuario?	4/8	4/8	Desempeño del agente de visión (reconocimiento de personas)

Tabla 6.4 Evaluación de la identificación de un usuario que salió de la escena temporalmente

6.5 Evaluación de la llegada a la línea de meta

Con este experimento se busca evaluar qué tan bien puede el robot entender que ha llegado a la línea de meta y que es momento de detenerse y despedirse. Como situación inicial de cada experimento, el robot y el usuario ya se han presentado y el robot se encuentra siguiéndolo. El robot se encuentra en una situación de seguir que tiene como expectativa un posible gesto indicándole que se detenga. El usuario caminará en línea recta una distancia pequeña. En este momento inicia el experimento.

El usuario realiza el gesto para pedirle al robot que se detenga. El robot debe identificar el gesto y detenerse manteniendo una distancia de aproximadamente 140 centímetros al usuario. En seguida, el robot le avisa al usuario que detectó un gesto y le pide que mantenga las manos arriba para verificarlo. El usuario debe mantener las manos arriba, y cuando el robot detecta de nuevo el gesto le indica que la tarea ha terminado y se despide. En la Tabla 6.5 se muestran los resultados obtenidos tras ejecutar diez veces este experimento. Los resultados de este experimento fueron muy satisfactorios.

Pregunta		Sí	No	¿Qué se está midiendo?
1	¿El robot identificó el gesto "Alto" del usuario?	10/10	0/10	Desempeño del agente de visión (reconocimiento de gestos)
2	¿El robot se detuvo manteniendo una distancia de 140 centímetros al usuario?	10/10	0/10	Desempeño del agente de navegación
3	¿El robot seguía identificando al usuario después de detenerse?	10/10	0/10	Desempeño del agente de visión (reconocimiento de personas)
4	¿El robot anunció que había detectado un gesto y pidió al usuario que mantuviera las manos arriba?	10/10	0/10	Desempeño del sintetizador de voz
5	¿El robot identificó de nuevo el gesto "Alto" del usuario?	9/10	0/10	Desempeño del agente de visión (reconocimiento de gestos)
6	¿El robot avisó que la actividad había terminado y se despidió del usuario?	9/9	0/9	Desempeño del sintetizador de voz

Tabla 6.5 Evaluación de la llegada a la línea de meta

Como conclusión de esta evaluación realizada por partes se puede observar que la mayor cantidad de fallas son provocadas debido a errores en la identificación de personas por parte del agente visual. Es necesario buscar un conjunto de algoritmos más robustos para disminuir la cantidad de errores.

Como se puede notar en esta evaluación, aún tomando fragmentos muy pequeños de la tarea, la complejidad del sistema es muy grande pues son muchos los puntos en donde el sistema puede fallar, y el fallo en resolver un pequeña parte de la tarea repercute en todas las siguientes.

Con el objetivo de hacer uniforme la serie de eventos en cada experimento, al momento de un fallo se suspendió el registro de los siguientes. Sin embargo, es importante señalar que en casi todos los fallos, el sistema es capaz de recuperarse pidiendo ayuda al usuario. Por ejemplo, si el agente de visión no identifica al usuario, el sistema le pedirá que se ponga de nuevo enfrente y que haga el gesto "Hola Ψ" para identificarlo de nuevo y continuar siguiéndolo.

6.6 Evaluación del sistema en la competencia *RoboCup@Home 2011*

El sistema completo se probó en la competencia internacional *RoboCup@Home 2011* realizada en Estambul, Turquía. Esta evaluación en un torneo real permitió medir el funcionamiento del robot seguidor en un ambiente más complejo que el interior de un laboratorio escolar.

El robot seguidor realizó la prueba *Follow Me* siendo guiado por una persona totalmente desconocida. Este operador no había visto con anterioridad el funcionamiento del robot, pero conocía a detalle la estructura de la tarea, es decir, sabía que debía empezar por pedirle al robot que lo siguiera, que debía seguir las instrucciones del robot mientras lo estaba conociendo, que al empezar a caminar habría ciertos eventos que el robot debía superar, etc. Lo que el operador no conocía era la forma específica en la que se debía comunicar con el robot. Al operador se le explicó de manera detallada (sin exceder los dos minutos que se disponían para esta parte de la prueba) la forma en que debía darle instrucciones al robot, es decir, se le explicaron los comandos verbales y gestuales que el robot podía recibir. Una vez que la prueba *Follow Me* dio inicio, los jueces vigilaron que el robot ejecutara la tarea de la manera correcta, y registraron en una hoja de evaluación el puntaje del robot que obtenía en cada checkpoint de la prueba. Esta hoja de evaluación se presenta en el Apéndice C, Figura C.1.

El escenario donde se realizó la prueba fue un corredor de superficie e iluminación regular, sin embargo había mucha gente alrededor mientras se realizaba la prueba, algunas de ellas mirando el funcionamiento de la prueba, y otras caminando a los costados, debido a que ese lugar era el acceso hacía una zona importante de la competencia.

Un total de 19 equipos participaron en este Stage de la competencia, y sólo 17 presentaron a su robot para competir en la tarea *Follow Me*. En la Tabla 6.6 se muestran los resultados obtenidos por los diferentes equipos que participaron en el Stage 1 de *RoboCup@Home 2011*. Es interesante resaltar que ningún equipo obtuvo la calificación máxima de 1100 puntos. El robot seguidor correspondiente a este trabajo de investigación pertenece al equipo mexicano Golem.

Los resultados de esta prueba fueron malos, ya que el robot obtuvo una calificación de cero puntos en los dos intentos. En el primer intento, el robot entendió cuando el operador le dio la instrucción “sígueme”, y le pidió al usuario que se pusiera enfrente de él a una distancia de seis pies haciendo el gesto “Hola Ψ” para identificar el esqueleto de la persona. El problema que surgió fue el hecho de que el gesto no resultó natural para el operador y no lo ejecutó de la manera correcta, por lo que el robot no lo reconocía. Dado que pasaron casi dos minutos sin que el sistema lograra reconocer el esqueleto de la persona a seguir se decidió cancelar este intento y reiniciar la prueba.

Equipo	País	Puntos acumulados en *Follow Me*	Posición final en la competencia
NimbRo	Alemania	1000	1
WrightEagle	China	1000	2
RobotAssist	Australia	1000	4
Robo-Erectus	Singapur	650	8
b-it-bots	Alemania	300	3
ZJUPanda	China	300	7
ToBI	Alemania	100	5
Markovito	México	100	12
MRL	Irán	75	6
homer	Alemania	75	11
PUMAS	México	0	9
Sourena	Irán	0	13
TU Eindhoven	Países Bajos	0	14
Golem	México	0	15
eR@sers	Japón	0	16
HomeBreakers	Chile	0	17
BORG	Países Bajos	0	18
Radical Dudes	Francia	-	10
CAMBADA	Portugal	-	19

Tabla 6.6 Resultados de la prueba *Follow Me* en *RoboCup@Home 2011*

En el segundo intento, el robot reconoció de una manera rápida el esqueleto de la persona y comenzó a seguirla. Sin embargo, en el momento de la oclusión temporal el robot perdió de vista al usuario original, lo que se debió a errores en el sistema de visión. El robot pidió ayuda al usuario (le pidió que se moviera un poco porque lo había perdido de vista) y el usuario, tratando de colaborar con el robot, lo hizo en forma errónea levantando las manos sin que el robot se lo solicitara. El robot identificó al usuario de nuevo, pero al mismo tiempo identificó que le estaba haciendo un gesto, y le dijo que esperaría diez segundos para que se alejara de él, es decir, en vez de seguirlo nuevamente, se pasó al segundo checkpoint de la competencia. Por esta razón, los jueces dictaminaron que la calificación obtenida era de cero puntos, ya que el robot, a pesar de seguir realizando la tarea, no lo había hecho en forma fiel al protocolo dictaminado por la prueba. El error del operador al levantar las manos en un momento inadecuado no se tomó en cuenta ya que, de acuerdo a los jueces, el robot no debía pedir este tipo de ayudas.

A partir de esta experiencia hay una serie de reflexiones importantes que se pueden hacer. La primera de ellas se refiere a la actitud colaborativa del usuario. En todos los experimentos realizados en las secciones anteriores, el usuario que operaba el robot tenía en mentalidad tratar de lograr que el sistema funcionara y colaboraba al máximo para tratar de lograr el objetivo de la prueba. En el caso de esta competencia, podemos considerar que el usuario no colaboró al

máximo con el robot, y esto se debió probablemente a que no entendió de manera clara las instrucciones que se le dieron al principio de la prueba. Además, los gestos no le resultaron tan naturales, pues le costaba trabajo tratar de hacerlos. Durante el desarrollo de la prueba, el usuario manifestaba confusión ante las instrucciones que el robot le daba, y parte de ello se debía que no entendía lo que el robot decía, debido a la gran cantidad del ruido ambiental que provocaba que la voz del robot se oyera débilmente.

Los errores que causaron el fallo del sistema se deben a una mala comunicación entre el robot y el usuario. Es necesario tomar en cuenta esto, y buscar estrategias que garanticen que el usuario logre entender más fácil lo que el robot le solicita, y que los gestos sean más naturales para que cualquier usuario los pueda realizar de manera sencilla. También es necesario robustecer los algoritmos utilizados en materia de visión computacional. Estos aspectos se tomarán en cuenta para la próxima participación en el concurso *RoboCup@Home 2012*, el cual se realizará en México.

Después de la experiencia obtenida en la participación en Estambul, se hicieron varias mejoras al sistema, las cuales consistieron en incluir mayor número de rutinas de recuperación en los modelos de diálogo. El sistema con las mejoras incluidas se sometió a una nueva evaluación, en esta ocasión para evaluar su funcionamiento en forma completa.

6.7 Evaluación de la satisfacción del usuario al utilizar el sistema

Para evaluar la satisfacción de los usuarios al utilizar el robot seguidor se utilizará un cuestionario similar al de PARADISE (Paradigm for Dialogue System Evaluation), que es un marco de trabajo para la evaluación de sistemas de diálogo. El modelo PARADISE propone que el desempeño puede ser correlacionado con un criterio externo significativo tal como la usabilidad. También propone que el objetivo principal de un sistema es maximizar la satisfacción del usuario (Walker, Litman, Kamm y Abella, 1997).

El cuestionario original de PARADISE (Walker, Litman, Kamm y Abella, 1998) está compuesto de ocho preguntas, donde cada una de ellas enfatiza un aspecto de la experiencia del usuario al interactuar con el sistema. En la Tabla 6.7 se muestran los rubros que evalúa el cuestionario, con un ejemplo de cómo plantear la pregunta al usuario.

Para la evaluación del robot seguidor se utilizará un cuestionario que contempla los rubros de PARADISE, pero se añadirán varias preguntas. Esto se hace debido a que se trata de evaluar un sistema interactivo multimodal que no sólo utiliza el lenguaje hablado para establecer la interacción entre usuario y máquina, sino que se involucran acciones motoras y lenguaje gestual para lograr cumplir el objetivo de la tarea. Además, se añaden algunas preguntas para profundizar más en la experiencia del usuario con respecto a qué tan cómodo se sentía con la forma en que el robot le seguía (qué tan cerca caminaba del usuario y a qué velocidad se aproximaba). En la Tabla 6.8 se propone un cuestionario de satisfacción del usuario para la prueba *Follow Me*, donde los rubros y sub-rubros indican lo que se está midiendo con cada pregunta.

Rubro	Pregunta
Desempeño del sintetizador de voz	¿Entendiste lo que el sistema te decía?
Desempeño del reconocedor de voz	¿El sistema entendió lo que tú le decías?
Facilidad de la tarea	¿Era fácil comunicarle al sistema lo que deseabas?
Ritmo de la interacción	¿Fue adecuado el ritmo de la interacción con el sistema?
Habilidad del usuario	¿Sabías qué decirle al sistema en cada momento de la tarea?
Respuesta del sistema	¿El sistema se tardaba en responderte?
Comportamiento esperado	¿El sistema funcionó como te lo imaginabas?
Uso futuro	¿Volverías a utilizar el sistema?

Tabla 6.7 Cuestionario de satisfacción del usuario de PARADISE

Estas preguntas utilizan escala de Likert, y cada una de ella tiene un conjunto de tres o cuatro posibles respuestas que puede dar el usuario entrevistado. Por ejemplo, para la pregunta *¿Entendiste lo que el robot pronunciaba?*, las posibles respuestas son *Nunca*, *Pocas veces*, *Muchas veces* y *Siempre*. Este cuestionario se aplicó a diez personas que utilizaron el robot seguidor. Las personas elegidas para la evaluación nunca antes habían utilizado el sistema. En el Apéndice C, Figura 2, se muestra el formato de evaluación de satisfacción al usuario que se utilizó para este experimento.

Para cada usuario se siguió el procedimiento que se describe a continuación:

1. Se explica al usuario cuál es el objetivo de la tarea y la forma en que le puede dar instrucciones al robot.
2. El usuario realiza la tarea *Follow Me*. En caso de que el sistema falle, la prueba se reinicia (sólo una vez). El tiempo máximo del usuario utilizando al robot es de 8 minutos, y a diferencia de la evaluación en la sección 6.6, el usuario debe ayudar en cualquier instante al robot. Se mide el puntaje que obtiene el robot en cada intento de acuerdo a la Tabla 2.1[20].
3. Al finalizar, se aplica al usuario el cuestionario de satisfacción.

En la Tabla 6.9 se presentan los resultados de la evaluación a cinco usuarios. En cada una de las preguntas, se resalta con negritas la respuesta que fue más frecuente. En la Tabla 6.10 se muestran los puntajes obtenidos por cada uno de los usuarios al realizar la prueba. Estos puntajes se obtuvieron utilizando la hoja de evaluación del Apéndice C, Figura C.1.

[20] En caso de reiniciar la prueba, si el puntaje del segundo intento es menor que el del primer intento, el puntaje final se obtiene promediándolos. En caso contrario, el puntaje final es el obtenido en el segundo intento.

Rubro	Sub-rubro	Pregunta	
Facilidad para entender al sistema	Desempeño del sintetizador de voz	1	¿Entendiste lo que el robot pronunciaba?
	Claridad de las instrucciones	2	¿Entendiste lo que el robot te pedía cuando te daba instrucciones?
Entendimiento del sistema	Desempeño del reconocedor de voz	3	¿El robot entendió las instrucciones que le diste en forma hablada?
	Desempeño del reconocedor de gestos	4	¿El robot entendió las instrucciones que le diste por medio de gestos?
Facilidad de la tarea	Percepción del usuario sobre la facilidad para terminar la tarea	5	¿Qué tan fácil fue terminar la tarea?
	Percepción del usuario sobre la facilidad para cumplir las instrucciones del sistema	6	¿Qué tan fácil fue cumplir las instrucciones que el robot te dio?
	Percepción del usuario sobre la facilidad para realizar los gestos	7	¿Qué tan fácil fue realizar los gestos para darle instrucciones al robot?
Ritmo de la interacción		8	¿Qué te pareció el ritmo de la interacción?
Habilidad del usuario		9	¿Sabías qué hacer en cada momento de la tarea?
Respuesta del sistema	Respuesta a comandos verbales	10	¿El robot se tardaba en entender cuando le dabas instrucciones en forma hablada?
	Respuesta a comandos gestuales	11	¿El robot se tardaba en entender cuando le dabas instrucciones por medio de gestos?
	Respuesta al movimiento del usuario	12	Cuando el robot te estaba siguiendo, ¿se tardaba en reaccionar cuando tú te movías de lugar?
Comportamiento esperado		13	¿El robot funcionó como te lo imaginabas?
Uso futuro		14	¿Volverías a dejar que el robot te siguiera?
Comodidad del usuario	Distancia entre robot y usuario	15	¿Qué te pareció la distancia a la que se mantenía el robot detrás de ti mientras te seguía?
	Velocidad del robot	16	¿Qué te pareció la velocidad del robot mientras te seguía?

Tabla 6.8 Cuestionario de satisfacción del usuario después de utilizar el robot seguidor

Pregunta		Respuestas			
1	¿Entendiste lo que el robot pronunciaba?	Nunca	Pocas veces	Muchas veces	Siempre
		0	0	0	**5**
2	¿Entendiste lo que el robot te pedía cuando te daba instrucciones?	Nunca	Pocas veces	Muchas veces	Siempre
		0	0	2	**3**
3	¿El robot entendió las instrucciones que le diste en forma hablada?	Nunca	Pocas veces	Muchas veces	Siempre
		0	0	**3**	2
4	¿El robot entendió las instrucciones que le diste por medio de gestos?	Nunca	Pocas veces	Muchas veces	Siempre
		1	0	1	**3**
5	¿Qué tan fácil fue terminar la tarea?	Muy difícil	Difícil	Fácil	Muy fácil
		1	0	**3**	1
6	¿Qué tan fácil fue cumplir las instrucciones que el robot te dio?	Muy difícil	Difícil	Fácil	Muy fácil
		0	1	**2**	**2**
7	¿Qué tan fácil fue realizar los gestos para darle instrucciones al robot?	Muy difícil	Difícil	Fácil	Muy fácil
		0	**2**	1	**2**
8	¿Qué te pareció el ritmo de la interacción?	Muy lento	Lento	Rápido	Muy rápido
		0	0	2	**3**
9	¿Sabías qué hacer en cada momento de la tarea?	Nunca	Pocas veces	Muchas veces	Siempre
		0	0	1	**4**
10	¿El robot se tardaba en entender cuando le dabas instrucciones en forma hablada?	Era muy lento	Era lento	Era rápido	Era muy rápido
		0	1	**3**	1
11	¿El robot se tardaba en entender cuando le dabas instrucciones por medio de gestos?	Era muy lento	Era lento	Era rápido	Era muy rápido
		1	1	**2**	1
12	Cuando el robot te estaba siguiendo, ¿se tardaba en reaccionar cuando tú te movías de lugar?	Era muy lento	Era lento	Era rápido	Era muy rápido
		0	0	1	**4**
13	¿El robot funcionó como te lo imaginabas?	Mucho peor	Peor	Mejor	Mucho mejor
		0	0	1	**4**
14	¿Volverías a dejar que el robot te siguiera?	No	Tal vez no	Tal vez si	Sí
		0	0	0	**5**
15	¿Qué te pareció la distancia a la que se mantenía el robot detrás de ti mientras te seguía?	Muy lejos	La distancia era adecuada		Muy cerca
		0	**5**		0
16	¿Qué te pareció la velocidad del robot mientras te seguía?	Muy lenta	La velocidad era adecuada		Muy rápida
		1	**4**		0

Tabla 6.9 Resultados de la evaluación de satisfacción del usuario

Usuario	Intento 1	Intento 2	Puntaje Final
1	1000	No hubo	1000
2	0	0	0
3	350	500	500
4	700	No hubo	700
5	100	1000	1000
Promedio			640

Tabla 6.10 Puntaje obtenido por los usuarios que realizaron la evaluación del sistema

Como resultado de esta evaluación se puede concluir que los usuarios manifiestan en general opiniones positivas acerca de la experiencia de utilizar el robot, aún cuando la tarea no se completa con éxito. A modo de trabajo futuro se pueden realizar evaluaciones más profundas en las que se busque establecer una correlación entre el nivel de éxito de la tarea y la satisfacción del usuario.

Como resultado de este capítulo se tienen una serie de experimentos distintos que se propusieron y ejecutaron para evaluar el funcionamiento del robot seguidor. De las evaluaciones presentadas en este capítulo, la única que permite comparación con otros robots seguidores es la expuesta en la sección 6.6, sin embargo el resultado negativo que se obtuvo también estuvo influenciado por errores humanos generados por la presión de la competencia. La manera más correcta de seguir evaluando este sistema es medirlo nuevamente en futuras competencias de robótica, con otros robots seguidores que realicen la misma tarea y en el mismo ambiente. Comparar el desempeño general del sistema con otros trabajos de investigación publicados en artículos o capítulos de libros no es algo viable pues, como se explicó en el capítulo 2, no todos los robots seguidores realizan la misma tarea. Finalmente, después de haber evaluado el sistema en funcionamiento, es momento de presentar las conclusiones de este trabajo de investigación.

Capítulo 7

Conclusiones

En este último capítulo se presentan las conclusiones y resultados obtenidos a partir de esta tesis, en la que se implementó en un robot móvil la habilidad de seguimiento de personas haciendo uso de modelos de diálogo y de la Arquitectura Cognitiva Orientada a la Interacción. Este sistema cuenta con los elementos necesarios para resolver la tarea *Follow Me* del concurso *RoboCup@Home*.

Lo primero que se realizó fue analizar la naturaleza de la actividad de seguimiento cuando es realizada por los seres humanos, y se explicó que cuando dos humanos tratan de resolver esta tarea se genera un diálogo práctico entre ellos que involucra distintas conductas comunicativas verbales y no verbales. Entre más de estas conductas comunicativas se incorporen a un robot seguidor, la interacción humano-máquina resultará más real, sin embargo, ante la gran cantidad de posibles conductas que se suscitan cuando un humano sigue a otro, se eligió a la prueba *Follow Me* como una pauta para delimitar qué conductas y habilidades se integrarían en el robot seguidor que resulta de este proyecto de investigación.

Se hizo una revisión del trabajo previo en torno a robots seguidores con el objetivo de entender la complejidad de la tarea a resolver y de tener ideas acerca de cómo este problema ha sido atacado por diversos grupos de investigación y las aplicaciones que se le ha dado. Como resultado de esta parte de la investigación se determinó que a pesar de la gran cantidad de información respecto a robots seguidores de personas, no se puede comparar fácilmente un trabajo con otro debido a que las dificultades de la tarea que realizan son diferentes. Por esta razón, en esta tesis se propuso clasificar a los robots seguidores dependiendo del nivel de dificultad de la tarea que realizan, y para ello se plantearon una serie de criterios para discernir las diferencias entre ellos.

El siguiente paso fue modelar la estructura de la tarea. Para ello, se analizó cuidadosamente la descripción de la prueba *Follow Me* y se hicieron los modelos de diálogo que capturaban el protocolo conversacional. Como resultado, se obtuvo una descripción funcional de la tarea, dejando a un lado los aspectos algorítmicos e implementacionales para una fase posterior de diseño del sistema. A pesar de que la tarea que se está modelando es muy compleja, la descripción funcional resultante mediante modelos de diálogo es muy compacta, así como muy flexible en caso de realizar alguna modificación.

Una de las características de la prueba *Follow Me*, es el hecho de que las fases están claramente definidas, por lo que se tiene una descripción muy clara de la tarea. Tener el contexto

para cada situación de una manera tan explícita amerita utilizar una metodología que lo aproveche al máximo, y los modelos de diálogo lo hicieron. Supongamos que en vez de haberse aprovechado la descripción de la prueba se hubiera creado un modelo de diálogo con una única situación de seguir, en la cual el sistema tiene como expectativas todas las acciones que podría hacer el usuario y todos los eventos que podrían ocurrir. En este modelo hipotético de robot seguidor genérico, en cualquier momento el usuario podría pedirle al robot que se parara mediante un gesto, el usuario podría ocultarse del robot, una segunda persona podría cruzar entre el robot y el usuario generando una oclusión temporal, etc. El modelo de diálogo resultante sería mucho más pequeño que los presentados en este trabajo de investigación, pero existiría como gran desventaja una interacción pobre entre robot y usuario. Esto sucedería porque el sistema dejaría de ubicarse en qué punto de la tarea está. Además, las rutinas de recuperación dejarían de ser robustas para una situación particular y se volverían también genéricas. Por ejemplo, si este sistema detectara que ha perdido de vista al usuario, no sabría si la persona lo hizo intencionalmente (porque se está ocultando del robot, algo que ocurre en la fase Reconociendo al usuario de la prueba *Follow Me*) o lo perdió de vista por un error en el reconocedor de personas.

Los modelos de diálogo construidos están orientados a que el robot pueda realizar la tarea *Follow Me* con éxito, pero no se limitan a ello. La primera diferencia radica en que si la prueba fracasa por alguna razón, el robot pedirá ayuda al usuario para poder continuar realizando la tarea. Aunque esto no se permite (aunque tampoco se prohíbe explícitamente) en el reglamento de la prueba *Follow Me*, se implementaron rutinas de recuperación en las que se pide ayuda al humano (por ejemplo, el robot pide que se vuelva a poner enfrente y que le haga el gesto "Hola Ψ"). Esto se hizo con el objetivo de que la habilidad de seguimiento no quede enmarcada sólo dentro de la tarea *Follow Me*, sino que pueda ser utilizada para otras aplicaciones.

Otra diferencia es la manera en que termina la tarea. En la prueba *Follow Me* basta con que el robot cruce la línea de meta y con eso concluye la actividad. En este robot seguidor se incorporó un modelo de diálogo en el cual se continúa siguiendo al usuario hasta que reciba la orden de detenerse por medio de un gesto. Una vez que el robot se da cuenta que el usuario le ha indicado que la actividad ha terminado, se despide y se pone en espera de un nuevo usuario.

Una vez obtenidos los modelos de diálogo fue posible identificar en forma sencilla qué agentes de percepción y comportamiento debían formar parte del sistema, así como las funciones que debían tener. Algunos de estos agentes fueron tomados de proyectos previos y se les realizaron las modificaciones pertinentes para que cumplieran las funciones específicas que este sistema demandaba. Sin embargo, para esta tarea, se construyó un agente de visión especializado en el reconocimiento de personas.

El agente de visión implementado tiene un funcionamiento basado en casos, los cuales son determinados por la lista de expectativas que se tienen en una situación específica de los modelos de diálogo. Guiándose con un esquema general de decisión propuesto en este proyecto y que contempla todos los posibles casos, el agente contiene esquemas de decisión específicos a un conjunto de expectativas en una situación particular, y en base a la información obtenida de un

módulo encargado específicamente del reconocimiento de personas y gestos (que utiliza como base para su construcción al *user tracker* de OpenNI), elige cuál es la expectativa que se cumple.

Es importante señalar que los modelos de diálogo fueron diseñados de modo que el agente de navegación funcionara como un Sistema Reactivo Autónomo. De esta manera se logró coordinar en forma efectiva la navegación y la visión en el robot seguidor utilizando la Arquitectura Cognitiva Orientada a la Interacción.

Una vez que todos los agentes se integraron en el sistema de seguimiento de personas sobre el robot móvil, se hicieron evaluaciones sobre su funcionamiento, de donde se obtuvieron resultados interesantes. La primera evaluación que se realizó fue por partes, ejecutando pequeños fragmentos de la tarea *Follow Me* para detectar los puntos débiles del sistema. Como resultado de esta evaluación se determinó que es necesario mejoras en reconocimiento de personas, gestos y voz. También se hizo un estudio detallado acerca del *user tracker* de OpenNI, para medir qué tan robusto es en el reconocimiento de personas y gestos (Apéndice B).

Utilizar el *user tracker* de OpenNI como base para el reconocedor de personas resultó una excelente opción para la detección y localización de personas, sin embargo, en materia de identificación de personas no es lo suficientemente robusto, específicamente cuando ocurre una oclusión temporal y los casos en que la persona sale de la escena y regresa instantes después. Además, cuando el sensor de profundidad está sobre el robot móvil, la vibración y el cambio de background que ocurren cuando el robot se mueve, incrementan bastante los errores de identificación. Esto no quiere decir que se deba descartar a OpenNI para la construcción de aplicaciones móviles que requieran el reconocimiento de personas, sino más bien se debe añadir una técnica paralela para compensar las posibles fallas mediante una reconfirmación en el proceso de identificación de personas. Ya que el *user tracker* utiliza la información proveniente del sensor de profundidad, sería bueno complementar con alguna técnica de visión computacional que utilice información de color proveniente de una cámara RGB.

La siguiente evaluación que se hizo al sistema fue en la competencia *RoboCup@Home 2011*. Aunque el resultado fue negativo, la experiencia de poner a prueba al robot en condiciones más difíciles (mayor número de personas presentes en el ambiente, un usuario de otra nacionalidad, presión por el hecho de estar en competencia, varios jueces evaluando al robot, etc.) permitió identificar algunos problemas que no se habían contemplado con anterioridad.

Como resultado de la experiencia anterior, se llegó a la conclusión de que el reconocimiento de gestos es otro asunto que debe ser mejorado en este robot seguidor. Aunque se cumple con el objetivo de que el robot entienda un gesto del usuario, los gestos tardan bastante en ser identificados y además no resultan naturales para todas las personas. Como trabajo futuro se pretende incorporar gestos cuya realización por parte del usuario sea menos forzada y más espontánea. Sin embargo, esta tarea se complica debido a que el concepto de naturalidad de un gesto es relativo, y lo que para algunas personas puede resultar muy natural, para otras personas puede resultar totalmente extraño. Este problema se agrava si consideramos

que la prueba *Follow me* se da dentro de un concurso internacional, donde personas de diferentes partes del mundo participan y la determinación de la naturalidad de un gesto puede tener enormes brechas entre personas de distintas culturas.

Otro resultado de la experiencia en el concurso *RoboCup@Home 2011* está relacionado con la importancia de la actitud colaborativa del usuario. En esta evaluación, el usuario no interactuó correctamente con el robot (pues no obedeció correctamente las instrucciones cuando el robot pidió ayuda), y esto se debió a que el usuario no entendió lo que el robot le pidió y a que, por el hecho de estar en condiciones de competencia, se tenía que seguir fielmente el protocolo establecido en la prueba *Follow Me*. Esto permite reflexionar en torno a la importancia que tiene la disposición del usuario de colaborar con el robot para cumplir juntos una tarea.

Cuando dos humanos establecen en forma consciente una tarea de seguimiento y ambos tienen como objetivo llegar juntos a determinado lugar, se trata de un trabajo en equipo. El seguidor irá todo el tiempo prestando atención a los movimientos del guía con el objetivo de no perderlo. Pero también el guía está atento de los movimiento de la otra persona, por ejemplo, si el guía se percata que el otro se ha quedado muy atrás, entonces se detendrá temporalmente o disminuirá su velocidad, esperando a que la otra persona se ponga a la par.

En la actividad de seguimiento, perder a una persona no es dejarla de ver, sino la imposibilidad de lograr establecer de nuevo comunicación con ella para seguir caminando juntos. Por ejemplo, imaginemos que dos personas se encuentran en un concierto, en el cual hay poca iluminación, mucho ruido y demasiada gente. Supongamos que uno de ellos le pide al otro que lo siga, y en el trayecto, pierden contacto visual entre la multitud de gente. Perder a una persona implica que no hay manera de recuperarse, pero en este caso, ambos pueden hacer cualquier tipo de intento para restablecer la comunicación entre ellos y poder reencontrarse. Gritar “aquí estoy”, alzar la mano o mandarle un mensaje por celular indicando un punto de encuentro, son intentos por lograr restablecer el contacto para darle continuidad a la actividad de seguimiento.

La actividad de seguimiento para los humanos, en el sentido de ir en compañía de, requiere una responsabilidad de ambos individuos involucrados en la interacción para lograr el objetivo común, es decir, llegar juntos a un determinado lugar. En el caso de los robots de servicio, el hecho de pedir ayuda al humano que está siguiendo no tendría por qué ser algo negativo, pues los humanos se ayudan unos a otros.

En el caso del robot móvil implementado, el éxito de la tarea depende de que el usuario realmente esté involucrado en la actividad y asuma cierta responsabilidad. Si el usuario sigue caminando sin fijarse si el robot efectivamente va detrás de el, o si ignora las instrucciones que el robot le da cuando está pidiendo ayuda porque lo ha perdido, la tarea está destinada al fracaso. El éxito de la tarea está directamente relacionado con qué tanto el usuario trata de ayudarlo.

Por otra parte, no podemos culpar de todo a los usuarios. En muchas ocasiones en que el usuario no proporciona ayuda al robot no es porque no quiere, sino porque no sabe cómo hacerlo. Esto se puede deber a que no entendió con claridad las instrucciones del robot o las

malinterpretó. En el diseño de este tipo de sistemas se debe buscar un punto intermedio, en el cual suponemos que el usuario no es un experto y puede no entender las instrucciones a la primera, pero sí debe tener una actitud colaborativa para ayudar en todo lo posible al robot, pues de lo contrario, el objetivo de la tarea no se cumplirá. Si pretendemos que los robots interactúen con los humanos de una manera natural, es indispensable que haya un esfuerzo también por parte de los usuarios, y no dejar toda la carga al robot, pues así no es como funcionan las cosas en la comunicación humana. Esta problemática podría ser objeto de un análisis más profundo utilizando la teoría de las estructuras de obligaciones y aterrizaje común aplicadas a diálogos prácticos (Pineda *et. al.*, 2006 ; Pineda, Estrada, Coria y Allen, 2007).

Después de la prueba realizada en Estambul se identificaron algunos puntos débiles del sistema, los cuales fueron mejorados mediante la incorporación de rutinas de recuperación más robustas en los modelos de diálogo, mismos que han seguido en un proceso de refinamiento continuo. De igual manera, se han hecho mejoras en los algoritmos y herramientas que utilizan los agentes del sistema. Por ejemplo, en la versión actual del agente de visión para el reconocimiento de personas se han incorporado nuevas versiones del *user tracker* de OpenNI, que han mejorado en forma considerable el reconocimiento de personas y gestos.

La última evaluación que se hizo al sistema tenía como objetivo medir la satisfacción del usuario al utilizar el robot, y se obtuvieron resultados buenos en general. El hecho de que el robot expresara lo que estaba viendo (por ejemplo, cuando decía "Alguien está pasando entre tu y yo"), cuando pedía ayuda para recuperarse ("Necesito ayuda porque no te puedo encontrar"), o cuando le decía al usuario con anticipación lo que haría ("Me acercaré a ti, espera un momento"), resultaban momentos gratificantes para los usuarios, porque sentían que el robot realmente les estaba prestando atención. El robot seguidor pudo haberse implementado de manera que los diálogos fueran mínimos e incluso nulos, pero esto equivale a decir que el robot camina detrás del usuario. Haber introducido estos diálogos permitió una interacción más interesante y creativa que hizo a las personas tener una mejor experiencia al utilizar el robot seguidor, y sentir que realmente los acompañaba.

Como trabajo futuro, se pretende darle continuación a este trabajo de investigación con mira a presentarlo en la competencia *RoboCup@Home 2012*, que se realizará en México. Son varios los frentes desde los cuales se puede continuar con este proyecto, entre ellos:

- Hacer un análisis más profundo de la forma en que interactúa el robot con el usuario durante la actividad de seguimiento, con el objetivo de mejorar los diálogos y conductas del robot para lograr una interacción más realista.
- Incorporar al robot la capacidad de realizar y entender mayor número de conductas no verbales. Actualmente el robot es capaz de entender gestos del usuario, pero se podría hacer una investigación más profunda acerca de otro tipo de comportamientos no verbales (como los señalados en el capítulo 1).
- Mejorar o reemplazar los algoritmos y herramientas utilizados por los distintos agentes del sistema, para obtener un mejor rendimiento del sistema.

- Integrar la habilidad de seguimiento dentro de tareas aún más complejas.

El hecho de que un robot pueda resolver por completo la tarea *Follow Me* no implica que esté listo para superar los retos del mundo real. Todavía quedan muchas cosas por mejorar e investigar antes de que los robots puedan estar dentro de un hogar realizando las tareas que los humanos les piden y nos puedan seguir el paso por un mundo tan complejo como el que habitamos.

Apéndice A

Modelos de Diálogo codificados

Este Apéndice tiene como objetivo mostrar al lector parte de los modelos de diálogo codificados. Por cuestiones de espacio, sólo se mostrarán los modelos más representativos, que incluyen los diversos tipos de situaciones, la forma de manejar errores del sistema (situación *ls* de la sección A.2) y funciones que evalúan la historia de la interacción (situación *v* de la sección A.2). Los puntos suspensivos (…) indican que se ha omitido una parte del código.

A.1 Modelo de diálogo principal

```
diag_mod(main,
[
	[
		id ==> is,
		type ==> neutral,
		out_pairs ==> [
			empty:mostrar_boton => c
		]
	],

	[
		id ==> c,
		type ==> click,
		out_pairs ==> [
			start:saludo => ri
		]
	],

	[
		id ==> ri,
		type ==> recursive(inicio),
		out_pairs ==> [
			continua:empty => rot
		]
	],

	…

	[
		id ==> fs,
		type ==> final,
		out_pairs ==> [
			salida:empty => empty
		]
	],
```

```
        [
                id ==> unexpected_speech_act,
                type ==> error,
                out_pairs ==> [
                        In_Trans => Previous_Situation
                ]
        ]
],

%Actos retoricos
[

        [
                id ==> mostrar_boton,
                rht_acts ==> [
                        caption(mostrar_boton),
                        putButton('start','boton_verde',300,350),
                        putImage('bck','fondo',0,0)
                ]
        ],

        [
                id ==> saludo,
                type ==> entradda,
                rht_acts ==> [
                        initSpeech,
                        caption(saludo)
                ]
        ],

        …

        [
                id ==> error(Concept),
                type ==> correction,
                rht_acts ==> [
                        error(Concept)
                ]
        ]
]
).
```

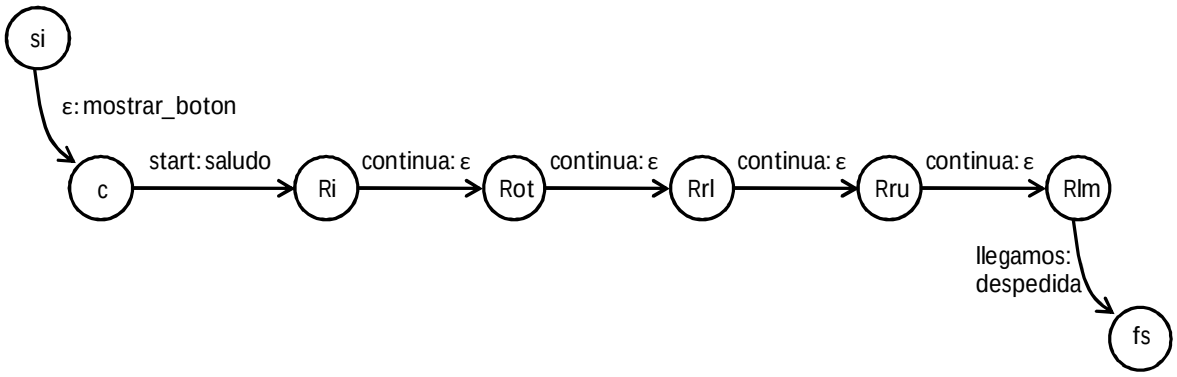

Figura A.1 Modelo de diálogo principal

A.2 Submodelo de diálogo inicio

```
diag_mod(inicio,
[
	[
		id ==> is,
		type ==> neutral,
		out_pairs ==> [
			empty:oferta => ls
		]
	],

	[
		id ==> ls,
		type ==> listening,
		out_pairs ==> [
			sigueme: ponte_enfrente => v
		],
		error_pairs ==> [
			noEntendi: error_message => ls,
			nada: error_message_2 => ls
		]
	],

	[
		id ==> v,
		type ==> seeing,
		out_pairs ==> [
			persona(N): listo_para_seguirte => fs,
			nadie_enfrente: sigo_sin_verte =>
			apply(check_size_pattern,[ID:(_,v,nadie_enfrente:__=>_),3,n,v])
		]
	],

	[
		id ==> n,
		type ==> neutral,
		out_pairs ==> [
			empty:indicaciones_detalladas => v
		 ]
	],

	[
		id ==> fs,
		type ==> final,
		out_pairs ==> [
			continua:empty => empty
		]
	],

	 [
		id ==> unexpected_speech_act,
		type ==> error,
		out_pairs ==> [
			In_Trans => Previous_Situation
		]
	 ]
],

%Actos retoricos
[

	[
		id ==> oferta,
		type ==> juego,
		rht_acts ==> [
			caption(oferta)
```

```
                ]
        ],

        …

        [
                id ==> error_message(X),
                rht_acts ==> [caption(X)]
        ]
]
).
```

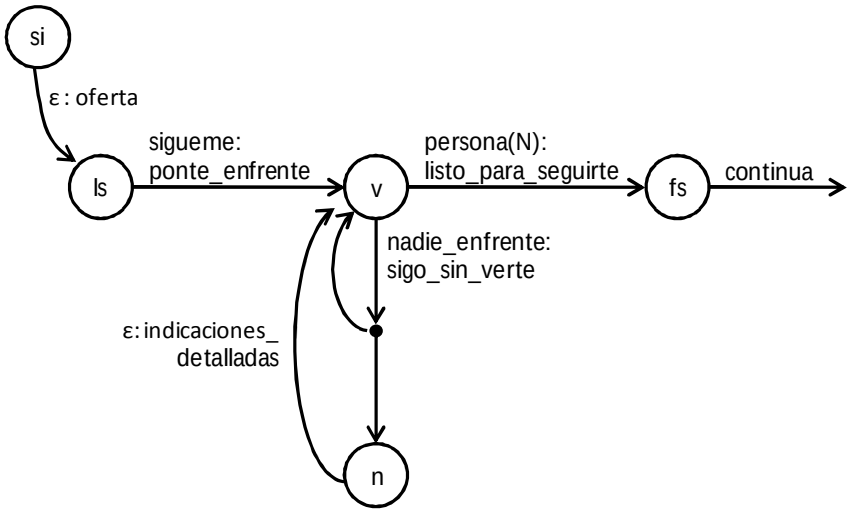

Figura A.2 Submodelo de diálogo Inicio

Apéndice B

Caracterización del *user tracker* de OpenNI v. 1.0.0.25

El objetivo de este Apéndice es presentar una serie de experimentos que se realizaron para evaluar el funcionamiento del *user tracker* de OpenNI, con el propósito de determinar qué tan conveniente es su uso para los diversos retos que demanda la prueba *Follow Me*. Estos resultados fueron de gran ayuda para la elección de algunos de los parámetros del robot seguidor, por ejemplo, la distancia más conveniente a la que el robot se debe mantener detrás del usuario para identificar gestos.

Como se explicó en el capítulo 5, el *user tracker* de OpenNI es uno de los bloques para la construcción del reconocedor de personas que suministra datos al agente de visión que utiliza el robot seguidor. Sin embargo, este bloque puede ser reemplazado en caso de encontrar otro conjunto de algoritmos que resuelvan mejor la tarea del reconocimiento de personas. Para este trabajo se eligió el *user tracker* de OpenNI porque los resultados obtenidos fueron satisfactorios, aunque no perfectos.

Como se explicó en el Capítulo 2, el reconocimiento de personas involucra tres problemas: la detección, la localización y la identificación de personas. Los experimentos realizados estuvieron ideados para medir qué tan bien son resueltos estos problemas por OpenNI. También se añade el problema de reconocimiento de gestos como un asunto a evaluar en este Apéndice.

Todos los experimentos se realizaron en un laboratorio escolar. El sensor de profundidad se encontraba a una altura de 130 centímetros del piso, y apuntaba hacia un área despejada en el que las personas se podían mover libremente. Al fondo y a los costados había objetos como sillas, mesas, computadoras, ventanas, etc. La iluminación era artificial y los experimentos se realizaron durante el día en horas en que la luz del sol no incidiera en forma directa por las ventanas (el nivel de iluminación del lugar era de 500 lx ± 15%).

B.1 Detección de personas

La detección de personas consiste en determinar si un objeto de la escena es una persona. Para el primer experimento se pidió a una persona que se parará enfrente del sensor a una determinada distancia. El *user tracker* se inicializaba y se le avisaba a la persona, la cual debía moverse un paso a la izquierda, luego un paso a la derecha y luego alzar los brazos. El objetivo era provocar que la persona cambiara de posición. Esta serie de movimientos se debía realizar en menos de diez segundos. Si después de este tiempo no se detectaba a la persona se consideraba

como un error. En la Tabla B.1 se muestran los resultados de este experimento variando la distancia a la que se encuentra la persona con respecto al sensor, y haciendo diez pruebas para cada una de esas distancias.

Distancia (cm)	Aciertos	Errores
30	0	10
60	0	10
90	7	3
120	10	0
150	10	0
180	10	0
210	10	0
240	10	0
270	10	0
300	10	0
330	10	0
360	9	1
390	9	1
420	6	4
450	0	10

Tabla B.1 Detección de personas a diferentes distancias

Como resultado de este experimento se puede concluir que hay una buena detección de personas mientras se encuentren a una distancia mayor de un metro y menor a cuatro metros. Es necesario que las personas se muevan un poco para poder ser detectadas, pues si se mantienen totalmente estáticas, el *user tracker* no las detecta.

Uno de los problemas del *user tracker* es que frecuentemente entrega falsos positivos. Esto ocurre cuando el sensor se mueve de posición mientras el *user tracker* está en funcionamiento. Muros, puertas, ventiladores, etc. son detectados erróneamente como si fueran personas.

B.2 Localización de personas

La localización de personas consiste en determinar la posición en el espacio en que una persona se encuentra. El experimento consistió en inicializar el *user tracker* y pedirle a una persona que entrara en la escena para que fuera detectada. Una vez detectada, la persona se puso en distintos lugares del escenario, sin salirse del área en que es detectada correctamente. En cada lugar se registró del *user tracker* la coordenada en la que calculaba se encontraba el centro de

masa del usuario. Posteriormente se obtuvo la medida del centro de masa del usuario en forma manual utilizando una cinta métrica.

Sea $C_u = (x_u , y_u , z_u)$ la coordenada del centro de masa calculada por el *user tracker*, y sea $C_m = (x_m , y_m , z_m)$ la obtenida en forma manual. Suponiendo que la coordenada obtenida en forma manual es correcta, se calculará el error de la medida hecha por el *user tracker* como la distancia entre C_u y C_m de la siguiente manera:

$$d^2 = (x_m - x_u)^2 + (y_m - y_u)^2 + (z_m - z_u)^2 \quad \text{...(1) distancia en el espacio entre dos puntos}$$

Se hicieron un total de diez mediciones, y en promedio se obtuvo como error una distancia de 15.48 cm. Se puede considerar que la localización de personas realizada por el *user tracker*, sin bien no es del todo exacta, es suficiente para el robot seguidor implementado en este trabajo.

B.3 Identificación de personas

La identificación de una persona consiste en determinar si la persona que se está detectando es una persona en particular que ya se había detectado con anterioridad. El primer experimento que se realizó consistió en pedir a una persona (a quien llamaremos usuario) que entrara a la escena. Una vez que había sido detectado por el *user tracker*, ese usuario recibe un identificador numérico. A continuación se pide a una segunda persona que pase entre el sensor y el usuario, cruzando de manera perpendicular la línea que une al sensor con el usuario (ver Figura B.1). Una vez que la persona termina de cruzar, se verifica si el *user tracker* sigue detectando al usuario. Si el usuario tiene asignado el mismo identificador numérico, entonces ha sido identificado correctamente, de lo contrario se considerará como un error.

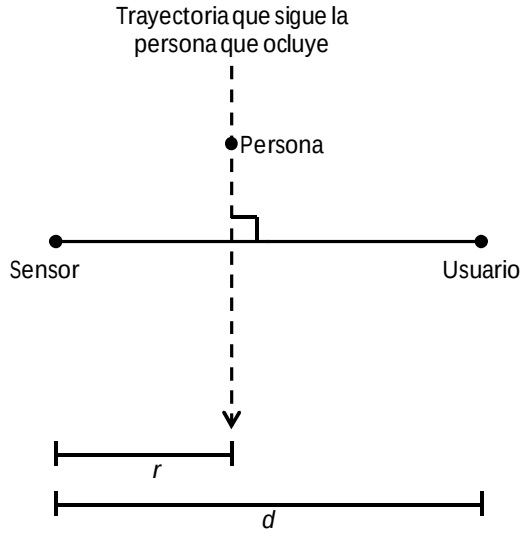

Figura B.1 Forma en la que debe cruzar la persona que causa la oclusión (vista superior)

Sea *d* la distancia entre el sensor y el usuario, y sea *r* la distancia entre el sensor y el punto de cruce de la persona sobre la línea que une al sensor con el usuario. En el primer experimento se varió la distancia *d*, con el objetivo de ver cual es la distancia más conveniente que debe mantenerse entre el usuario y el sensor. En la Tabla B.2 se muestran los resultados obtenidos. La persona cruzó en el punto medio de la línea que une al sensor con el usuario (es decir, $r = d/2$) y lo hizo a una velocidad de aproximadamente medio metro por segundo. Para cada valor de *d* se hicieron diez experimentos. Como resultado se puede concluir que entre mayor distancia haya entre el usuario y el sensor, es más probable que el usuario sea identificado correctamente tras una oclusión temporal provocada por otra persona.

d (cm)	Aciertos	Errores
100	0	10
110	1	9
120	7	3
130	6	4
140	8	2
150	9	1
160	10	0
170	9	1
180	10	0
190	10	0
200	9	1
210	9	1
220	10	0
230	10	0
240	10	0

Tabla B.2 Identificación de personas tras oclusión temporal variando *d*

En el siguiente experimentó se fijó el valor de *d* en 160 cm (ya que fue la distancia mínima entre el robot y el usuario del experimento anterior en la que no hubo ningún error). En este experimento lo que se varía es la distancia *r* con el objetivo de ver los efectos de que la persona cruce ya sea muy cerca del sensor o muy cerca del usuario. Nuevamente la velocidad de la persona al cruzar es de aproximadamente medio metro por segundo. En la Tabla B.3 se muestran los resultados obtenidos, haciendo diez experimentos para cada valor de *r*. Como resultado se puede concluir que entre más cerca pase la persona del sensor, mayor probabilidad de error habrá en la identificación del usuario. Si la persona pasa muy cerca del usuario, también puede provocar errores en su identificación.

r (cm)	Aciertos	Errores
20	0	10
30	1	9
40	1	9
50	3	7
60	2	8
70	9	1
80	10	0
90	10	0
100	10	0
110	9	1
120	10	0
130	10	0
140	7	3

Tabla B.3 Identificación de personas tras oclusión temporal variando *r*

En el siguiente experimento se varió la velocidad *v* con la que cruza la persona. En este experimento *d* = 160 cm y *r* = 80 cm. En la Tabla B.4 se muestran los resultados tras realizar diez experimentos para cada valor de *v*. Se puede concluir que entre más lento cruce la persona, mayor probabilidad de éxito habrá en la identificación del usuario.

	v (cm/s)	Aciertos	Errores
Muy lento	12.5	10	0
Lento	25	10	0
Normal	50	9	1
Rápido	100	4	6
Muy rápido	200	1	9

Tabla B.4 Identificación de personas tras oclusión temporal variando *v*

El siguiente experimento consistió en medir qué tanto puede el *user tracker* identificar a una persona que se salió de la escena y después regresa. El primer paso es pedir a un usuario que entre a la escena y esperar a que sea detectado y reciba su identificador numérico. Posteriormente, se le pide al usuario que salga de la escena y espere fuera un total de quince segundos antes de regresar a la escena. Si el *user tracker* le asigna el mismo identificador se considerará como un acierto. Este experimento se repitió un total de 50 veces y se obtuvieron un 56% de aciertos y un 44% de errores.

En un segundo experimento muy similar al anterior, se introdujo la variante de que al regresar el usuario a la escena, una segunda persona entre junto con el, con el objetivo de medir qué tanto afecta la presencia de una nueva persona en escena al intentar identificar a la original. Este experimento se repitió un total de 50 veces y se obtuvieron un 48% de aciertos y un 52% de errores. Es interesante señalar que en sólo 2 ocasiones el sistema identificó erróneamente a la segunda persona como si fuera el usuario original. En todos los demás errores el sistema consideró que el usuario nunca regresó a la escena.

B.4 Reconocimiento de gestos

El experimento tiene como objetivo medir qué tan bien reconoce el *user tracker* al gesto "Hola Ψ". Se pide al usuario que entre a escena y una vez que está justo enfrente del sensor empiece a realizar el gesto. Se mide el tiempo que tarda en ser reconocido el gesto, si pasan más de 30 segundos y no se ha detectado se considerará como un error. En este experimento se varía la distancia *d* entre el usuario y el sensor. En la Tabla B.5 se muestran los resultados obtenidos al realizar diez pruebas para cada valor de *d*. La última columna muestra el tiempo promedio *t* en que se tardó en detectar el gesto en los casos considerados como aciertos. Como resultado se puede concluir que la persona no debe estar muy cerca del sensor para que su gesto pueda ser identificado, lo cual se debe a que entre más cerca esté, sus brazos levantados quedan fuera de la escena.

d (cm)	Aciertos	Errores	*t* (seg)
100	0	10	-
110	0	10	-
120	0	10	-
130	1	9	14.00
140	7	3	17.85
150	8	2	19.00
160	7	3	8.57
170	7	3	8.71
180	9	1	14.22
190	7	3	18.57
200	8	2	16.25

Tabla B.5 Reconocimiento del gesto "Hola Ψ"

El segundo experimento consiste en medir qué tan bien reconoce el *user tracker* al gesto "Alto". Se pide al usuario que entre a la escena y que haga el gesto "Hola Ψ" hasta que sea reconocido y entonces el *user tracker* empiece a realizar el tracking de su esqueleto. Una vez que

se tiene identificado el esqueleto, se pide a la persona que realice el gesto "Alto". Este experimento se realiza de igual manera variando la distancia *d*. Los resultados se muestran en la Tabla B.6. En esta ocasión no se mide el tiempo, pues en todos los casos es menor de un segundo. Como resultado de este experimento se puede concluir que la detección del gesto "Alto" es muy robusta y rápida, sin embargo es dependiente de que se esté realizando el tracking del esqueleto de la persona. La distancia a partir de la cual el gesto comienza a ser identificado depende de la altura de la persona, pues el gesto será identificado siempre y cuando los brazos levantados estén dentro de la escena sensada.

d (cm)	Aciertos	Errores
100	0	10
110	0	10
120	0	10
130	0	10
140	10	0
150	10	0
160	10	0
170	10	0
180	10	0
190	10	0
200	10	0

Tabla B.6 Reconocimiento del gesto "Alto"

B.5 Conclusión

Como resultado de esta serie de experimentos se puede concluir que el *user tracker* de OpenNI es una muy buena herramienta para la detección de personas, su localización y el reconocimiento de gestos. Sin embargo, en la identificación de persona existen serios problemas, específicamente cuando la persona sale de la escena.

Todos los experimentos que se realizaron fueron con el sensor en una posición fija. Debe tomarse en cuenta que cuando el sensor está en movimiento (como cuando está arriba de un robot móvil) la cantidad de errores aumenta considerablemente.

El *user tracker* puede resolver la mayoría de los retos planteados por la prueba *Follow Me*, con excepción de la fase en la que un usuario sale de escena y regresa acompañado de una segunda persona, pues sólo la mitad de las veces el usuario será identificado correctamente.

Apéndice C

Formatos de evaluación de la prueba *Follow Me*

Score Sheet

Test: Follow Me

Team name: .....

Referee name: .....

The maximum time of the test is *8 minutes*, including calibrating on the operator.

Action	Score:	1st try	2nd try
Checkpoint 1: Temporary occlusion			
Successfully resume following the operator	*100*	.....	.....
Checkpoint 2: Tracking from the distance			
Understanding the user command	*50*	.....	.....
Bonus for using a gesture	*150*	.....	.....
Waiting a minimum of 10 seconds and then *finding and following the operator*	*100*	.....	.....
Checkpoint 3: Recognize owner			
Understanding the user command	*50*	.....	.....
Bonus for using a gesture	*100*	.....	.....
Recognizing and following the operator after he returns	*300*	.....	.....
Checkpoint 4: Finish line			
Crossing the finish line	*300*	.....	.....
No touching			
Reaching all 4 checkpoints and *not having touched any object or human* in the scenario	*100*	.....	.....
Special penalties & bonuses			
Not attending	*-500*	.....	.....
Outstanding performance	*100*	.....	.....
Total score	*1100*	.....	.....

Remarks:

....

Date & time *Referee* *Team leader*

RoboCup@Home Forms & Score Sheets / Final version for RoboCup 2011 Istanbul (Revision: 1643)

Figura C.1 Hoja de evaluación de la prueba *Follow Me* de *RoboCup@Home 2011*

Nombre del usuario: _____ Edad: ____ Sexo: _____

Ocupación: _____ Fecha: ____/_____/____

Pregunta		Respuestas			
1	¿Entendiste lo que el robot pronunciaba?	Nunca	Pocas veces	Muchas veces	Siempre
2	¿Entendiste lo que el robot te pedía cuando te daba instrucciones?	Nunca	Pocas veces	Muchas veces	Siempre
3	¿El robot entendió las instrucciones que le diste en forma hablada?	Nunca	Pocas veces	Muchas veces	Siempre
4	¿El robot entendió las instrucciones que le diste por medio de gestos?	Nunca	Pocas veces	Muchas veces	Siempre
5	¿Qué tan fácil fue terminar la tarea?	Muy difícil	Difícil	Fácil	Muy fácil
6	¿Qué tan fácil fue cumplir las instrucciones que el robot te dio?	Muy difícil	Difícil	Fácil	Muy fácil
7	¿Qué tan fácil fue realizar los gestos para darle instrucciones al robot?	Muy difícil	Difícil	Fácil	Muy fácil
8	¿Qué te pareció el ritmo de la interacción?	Muy lento	Lento	Rápido	Muy rápido
9	¿Sabías qué hacer en cada momento de la tarea?	Nunca	Pocas veces	Muchas veces	Siempre
10	¿El robot se tardaba en entender cuando le dabas instrucciones en forma hablada?	Era muy lento	Era lento	Era rápido	Era muy rápido
11	¿El robot se tardaba en entender cuando le dabas instrucciones por medio de gestos?	Era muy lento	Era lento	Era rápido	Era muy rápido
12	Cuando el robot te estaba siguiendo, ¿se tardaba en reaccionar cuando tú te movías de lugar?	Era muy lento	Era lento	Era rápido	Era muy rápido
13	¿El robot funcionó como te lo imaginabas?	Mucho peor	Peor	Mejor	Mucho mejor
14	¿Volverías a dejar que el robot te siguiera?	No	Tal vez no	Tal vez si	Sí
15	¿Qué te pareció la distancia a la que se mantenía el robot detrás de ti mientras te seguía?	Muy lejos	La distancia era adecuada		Muy cerca
16	¿Qué te pareció la velocidad del robot mientras te seguía?	Muy lenta	La velocidad era adecuada		Muy rápida
Puntuación en el primer intento:					
Puntuación en el segundo intento (si lo hubo):					
Puntaje final:					

Figura C.2 Formato de evaluación de satisfacción al usuario

Referencias bibliográficas

Allen, James F.; Byron, Donna K.; Dzikovska, Myroslava; Ferguson, George; Galescu, Lucian y Stent, Amanda. (2001). "Toward Conversational Human-Computer Interaction". En *AI Magazine*, 22(4), pp. 27-38.

Angelo, Joseph A. (2007). *Robotics: a reference guide to the new technology.* Greenwood Publishing Group, Estados Unidos.

Ambros, Veronika. (2010). "How Did the Golems (and Robots) Enter Stage and Screen and Leave Prague?". En Cornis-Pope, Marcel y Neubauer, John. *History of the literary cultures of East-Central Europe, junctures and disjunctures in the 19th and 20th centuries, Volume IV: Types and stereotypes*. John Benjamins Publishing Company, Amsterdam, pp. 308-320.

Avilés, Héctor; Alvarado-Gonzáles, Montserrat; Venegas, Esther; Rascón, Caleb; Meza, Ivan V. y Pineda, Luis. (2010). "Development of a Tour-Guide Robot Using Dialogue Models and a Cognitive Architecture". En *Advances in Artificial Intelligence – Iberamia 2010.* Springer, Alemania, pp. 512-521.

Bahadori, S.; Iocchi L.; Leone, G. R.; Nardi, D. y Scozzafava, L. (2005). "Real-Time People Loalization and Tracking Through Fixed Stereo Vision". En *Innovations in Applied Artificial Intelligence. 18th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert* Systems. Springer-Verlag Berlin Heidelberg, pp. 44-54.

Becker, Marcelo; Meirelles, Dantas Carolina y Perdigão, Macedo Weber. (2006). "Obstacle Avoidance Procedure for Mobile Robots". En *ABCM Symposium Series in Mechatronics*, Vol. 2, pp. 250-257.

Belloto, Nicola y Hu, Huosheng. (2007). "People tracking with a mobile robot: a comparison of Kalman and Particle Filters". En *Proceedings of the 13th IASTED International Conference Robotics and Applications*. Alemania, pp. 388-393.

Bennewitz, Maren; Axenbeck, Tobias; Behnke, Sven y Burgard Wolfram. (2008). "Robust Recognition of Complex Gestures for Natural Human-Robot Interaction". En *Proc. of the Workshop on Interactive Robot Learning at Robotics: Science and System Conference (RSS)*.

Beymer, David y Konolige, Kurt. (2001). "Tracking People from a Mobile Platform". En *IJCAI-2001 Workshop on Reasoning with Uncertainty in Robotics*, Estados Unidos.

Cielniak, Grzegorz y Duckett, Tom. (2004). "People Recognition by Mobile Robots". En *Journal of Intelligent and Fuzzy Systems*, 15, pp. 21-27.

Chong, Hui-Qing; Tan, Ah-Hwee y Ng, Gee-Wah. (2007). *Integrated Cognitive Architectures: A Survey*. En *Artificial Intelligence Review*, 28, pp. 103-130.

Dessimoz, Jean-Daniel y Gauthey, Pierre-François. (2010). "Domestic Service Robots in the Real World: the Case of Robots Following Humans". En *Proceedings of SIMPAR 2010 Workshops*, Alemania, pp. 217-228.

Freedman, Jeri. (2011). *Robots Through History*. The Rosing Publishing Group, Estados Unidos.

Fritsch, J.; Kleinehagenbrock, S.; Lang, S.; Fink, G. A. y Sagerer, G. (2004). "Audiovisual Person Tracking with a Mobile Robot". En *Proc. International Conference on Intelligent Autonomous Systems"*, pp. 898-906.

Gigliotta, Onofrio; Caretti, Massimiliano; Shokur, Solaiman y Nolfi, Stefano. (2005). "Toward a Person-Follower Robot". En *Proceedings of the Second RoboCare Workshop*, pp. 65-68.

Gockley, Rachel; Forlizzi, Jodi y Simmons, Reid. (2007). "Natural Person-Following Behavior for Social Robots". En *Proceedings of the ACM/IEEE international Conference on Human-Robot interaction*. pp. 17-24.

International Organization for Standardization. (1994). *ISO 8373:1994 Manipulating industrial robots – Vocabulary*.

Jurafsky, Daniel y Martin, James H. (2009). *Speech and Language Processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall, Estados Unidos.

Kleinehagenbrock, M.; Lang, S.; Fritsch, J.; Lömker, F.; Fink, G. A. y Sagerer G. (2002). "Person Tracking with a Mobile Robot based on Multi-Modal Anchoring". En *Proc. IEEE Int. Workshop on Robot and Human Interactive Communication*, Berlin, pp. 423-429.

Kovilarov, Marin y Sukhatme Gaurav. (2006). "People tracking and following with mobile robot using an omnidirectional camera and a laser". En *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Estados Unidos, pp. 557-562.

Krueger, Julianne. (2005). *Nonverbal Communication*. GRIN Verlag, Alemania.

Kwon, Hyukseong; Yoon, Youngrock; Byung, Park Jae y Kak, Avinash C. (2005). "Person Tracking with a Mobile Robot using Two Uncalibrated Independently Moving Cameras". En *Proceedings of the 2005 International Conference on Robotics and Automation*, España, pp. 2877-2883.

Laureano-Cruces, Ana Lilia; De Arriaga-Gómez, Fernando y Sánchez, María García-Alegre. (2001). "Cognitive task analysis: a proposal to model reactive behaviours". En *Journal of Experimental & Theoretical Artificial Intelligence*, 13:3, pp. 227-239.

Lehmann, Charles H. (2004). *Geometría Analítica*. Editorial Limusa, México.

Lerner, Michah; Vanecek, George; Vidovic, Nino y Vrsalovic, Dado. (2000). *Middleware Networks. Concepts, Design and Deployment of Internet Infrastructure*. Kluwer Academic Publishers, Estados Unidos.

Lowe, David G. (1999). "Object Recognition from Local Scale-Invariant Features". En *International Conference on Computer Vision*, Corfu, Greece, pp. 1150-1157.

Lowe, David G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints". En *International Journal of Computer Vision*, 60, 2, pp. 91-110.

Martin, David L.; Cheyer, Adam J. y Moran, Douglas B. (1999). "The Open Agent Architecture: A Framework for Building Distributed Software Systems". En *Applied Artificial Intelligence*, vol. 13, no. 1-2, pp. 91-128.

Mann, William C. y Thompson, Sandra A. (1988). *Rethorical Structure Theory: Towards a functional theory of text and organization*, Text 8(3), pp. 243-281.

Méndez-Polanco, José Alberto; Muñoz-Meléndez, Angélica y Morales-Manzanares, Eduardo F. (2010). "Detection of Multiple People by a Movile Robot in Dynamic Indoor Environments". En *Advances in Artificial Intelligence – Iberamia 2010.* Springer, Alemania, pp. 522-531.

Meza, Ivan; Pérez, Elia; Salinas, Lisset; Aviles, Hector y Pineda, Luis A. (2010). "A Multimodal Dialogue System for Playing the Game 'Guess the card' ". En *Procesamiento de Lenguaje Natural*, 44, pp. 131-138.

Meza, Ivan V.; Salinas, Lisset; Venegas, Esther; Castellanos, Hayde; Chavarría, Alejandra y Pineda, Luis A. (2010). "Specification and Evaluation of a Spanish Conversational System Using Dialogue Models". En *Advances in Artificial Intelligence – Iberamia 2010.* Springer, Alemania, pp. 346-355.

Miller, Frederic P.; Vandome, Agnes F. y McBrewster John. *Laser Rangefinder*. (2010). VDM Publishing House Ltd., 76 pp.

Montemerlo, Michael; Pineau, Joelle; Roy, Nicholas; Thrun, Sebastian y Verma, Vandi. (2002). "Experiences with a mobile robotic guide for the elderly". En *Proceedings of the National Conference of Artificial Intelligence*. pp. 587-592.

Muñoz-Salinas, Rafael; Aguirre, Eugenio; García-Silvente, Miguel y Gonzáles, Antonio. (2005). "People Detection and Tracking Through Stereo Vision for Human-Robot Interaction". En *Lectures Notes on Artificial Intelligence*, 3789, Springer, Alemania, pp. 337-346.

Nuñez, César; García, Alberto; Onetto, Raimundo; Alonzo, Daniel y Tosunoglu, Sabri. (2010). "Electronic Luggage Follower". En *Florida Conference on Recent Advances in Robotics 2010*, Florida.

OpenNI organization. (2011). *OpenNI User Guide*. En http://www.openni.org/documentation.

Pineda, Luis A.; Castellanos, Hayde; Coria, Sergio; Estrada, Varinia; López, Fernanda; López, Isabel; Meza, Ivan; Moreno, Iván; Pérez, Patricia y Rodríguez, Carlos. (2006). "Balancing Transactions in Practical Dialogues". En CICLing 2006, *Lecture Notes in Computer Science*, Springer-Verlag, Berlin Heidelberg, pp. 331-342.

Pineda, Luis A.; Estrada, Varinia M.; Coria, Sergio R. y Allen, James F. (2007). "The obligations and common ground structure of practical dialogues". En *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*. Vol 11, No. 36, pp. 9-17.

Pineda, Luis A. (2008). "Specification and interpretation of multimodal dialogue models for human-robot interaction". En *Artificial Intelligence for Humans: Service Robots and Social Modeling*. SMIA, México, pp. 33-50.

Pineda, Luis A.; Meza, Ivan V. y Salinas, Lisset. (2010). "Dialogue Model Specification and Interpretation for Intelligent Multimodal HCI". En *Advances in Artificial Intelligence – Iberamia 2010.* Springer, Alemania, pp. 20-29.

Pineda, Luis A. (2011). *The Golem Team, RoboCup@Home 2011*. En http://golem.iimas.unam.mx/.

Pineda, Luis A.; Avilés, Héctor H.; Meza, Ivan V.; Gershenson, Carlos; Rascón, Caleb; Alvarado, Montserrat; Salinas, Lisset. (2011). "IOCA: An Interaction-Oriented Cognitive Architecture". Submiteed.

Rascón, Caleb; Avilés, Hector y Pineda, Luis A. (2010). "Robotic Orientation towards Speaker for Human-Robot Interaction". En En *Advances in Artificial Intelligence – Iberamia 2010.* Springer, Alemania, pp. 10-19.

Reddy, Martin. (2011). *API design for C++*. Elsevier, Estados Unidos.

RoboCup, RoboCup@Home. (2011a). *RoboCup@Home. Forms & Score Sheets*. En http://www.ai.rug.nl/robocupathome/.

RoboCup, RoboCup@Home. (2011b). *RoboCup@Home. Rules & Regulations*. En http://www.ai.rug.nl/robocupathome/.

Shacker, Samir; Saade, Jean J. y Asmar, Daniel. (2008). "Fuzzy Inference-Based Person-Following Robot". En *International Journal of Systems Applications, Engineering & Development.* Issue 1, Volume 2, pp. 29.34.

Schlegel, C.; Illman J.; Jaberg, K.; Shuster, M. y Wörz, R. (1998). "Vision based person tracking with a mobile robot". En *Proceedings of the Ninth British Machine Vision Conference*, Reino Unido, pp. 418-427.

Sidenbladh, H.; Kragić, D. y Christensen, H. I. (1999). "A Person Following Behaviour for a Mobile Robot". En *IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 670-675.

Stillings, Neil A.; Weisler, Steven E.; Chase, Christopher H.; Feinstein, Mark H.; Garfield, Jay L. y Rissland, Edwina L. (1995). *Cognitive science: an introduction*. MIT Press, Estados Unidos.

Stückler, Jörg; Dröschel, David; Gräve, Kathrin; Holz, Dirk; Schreiber, Michael y Behnke, Sven. (2011). *NimbRo@Home 2011 Team Description*. En *http://www.nimbro.net/*.

Taboada, Maite y Mann, William C. (2005). "Applications of Rethorical Structure Theory". *Discourse Studies*, Text 8(4), pp. 567-588.

Treptow, André; Cielniak, Grzegorz y Duckett, Tom. (2006). "Real-Time People Tracking for Mobile Robots using Thermal Vision". En *Robotics and Autonomous System*. 54(9), p. 729-739.

Tulving, Endel. (1972). "Episodic and Semantic Memory". En Tulving, Endel y Donaldson, Wayne. *Organization of Memory*. Academic Press, Nueva York, pp. 381-403.

United Nations Economic Commission for Europe, International Federation of Robotics. (2005). *World Robotics 2005: Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investment*. United Nations Publications, Ginebra.

Walker, Marilyn A.; Litman, Diane J.; Kamm, Candace A. y Abella, Alicia. (1997). "PARADISE: A general framework for evaluating spoken dialogue agents". En *Proceedings of the 35th Annual Meeting of ACL*. Madrid, pp. 271-280.

Walker, Marilyn A.; Litman, Diane J.; Kamm, Candace A. y Abella, Alicia. (1998). "Evaluating spoken dialogue agents with PARADISE: Two case studies". En *Computer Speech and Language*, 12(3), pp. 141-168.

Xian-yi, Cheng y De-shen, Xia. (2005). "RoboCup is a Stage which Impulse the Research of Basic Technology in Robot". En Kordic, Vedran; Lazinica, Alexander y Merdan, Munir. *Cutting edge robotics.* Pro literatur Verlag, Croacia, pp. 433-446.

Yilmaz, Alper; Javed, Omar y Shah, Mubarak. (2006). "Object Tracking: A Survey". En *ACM Computing Surveys*, Vol. 38, No. 4, Article 13.

Zivkovic, Zoran y Kröse, Ben. (2008). "People Detection using Multiple Sensors on a Mobile Robot". En *Unifying Perspectives in Computational and Robot Vision*, Springer, pp. 1-15.

Zheng, Yuhua y Meng, Yan. (2009). "Real-Time People Tracking and Following Using a Vision-Controlled Mobile Robot". En *New Research in Robot Vision*, NOVA Publications.