Seguimiento Monocular 3D Para Rehabilitación

por

Ariel Molina Rueda

Tesis sometida como requisito parcial para obtener el grado de

MAESTRO EN CIENCIAS EN LA ESPECIALIDAD DE COMPUTACION

en el

Instituto Nacional de Astrofísica, Óptica y Electrónica

Julio 2008

Tonantzintla, Puebla

Supervisada por:

Dr. Enrique Sucar Succar, INAOE
Dr. Leopoldo Altamirano Robles, INAOE

Resúmen

En este trabajo de tesis se muestra el resultado del desarrollo de un sistema de seguimiento monocular tridimensional de objetos que puede funcionar incluso con cámaras (webcams) de bajo costo. El sistema desarrollado es capaz de detectar y seguir objetos de color, para ello requiere una muestra previa, a la cual se le calcula un histograma de color. Después el histograma es usado para realizar una retroproyección sobre las imágenes provenientes de la cámara. El proceso de retroproyección da como resultado mapas de probabilidad que indican la probabilidad de que el histograma de la muestra esté o no presente. Luego los mapas de probabilidad son insertados en una rejilla de un Filtro de Ocupación Bayesiano (BOF).

La retroproyección es insertada en la rejilla BOF la cual registra la probabilidad de que cada una de las celdas se encuentre "ocupada". Sobre la rejilla se aplica posteriormente un algoritmo de agrupamiento para encontrar cúmulos de celdas con altas probabilidades de ocupación. Finalmente los cúmulos con altos índices de ocupación son tratados como objetos en la escena a los cuales se les da seguimiento. En este trabajo sólo se da seguimiento a un solo objeto por lo que sólo se toma en cuenta el objeto con mayor certidumbre. La detección de la profundidad se hace mediante el conocimiento del tamaño del objeto y de la dicretización hecha por la rejilla BOF.

Se hicieron experimentos para medir la presición en el seguimiento. Se comparó la detección de profundidad con un sistema estéreo y se obtuvo que el sistema monocular tiene un desempeño similar pero sin la complejidad del sistema estéreo, ya que el sistema monocular no requiere de calibración de cámara ni del uso de sistemas de soporte especiales. Finalmente se integró el sistema de seguimiento monocular en un sistema de rehabilitación mediante gestos y se comenzaron pruebas clínicas preliminares.

Abstract

In this thesis, there are shown the results of the development of a three dimensional monocular tracking system that detects objects and can work even with low cost webcams, the system has application for in house and self directed rehabilitation. The developed system is able to detect and track color objects. In order to do it, the system requires a sample from which is obtained a color histogram. Then the color histogram is used to compute a backprojection over every image captured from the camera. The backprojection process gives a probability maps for every frame. Those maps indicate the presence or absence probability of the object. Then, the probability maps are fed to a Bayesian Occupancy Filter (BOF).

The backprojection is inserted into the BOF grid, which in turn records the probability of each cell being "occupied". Afterwards a clustering algorithm is applied over the grid to find clusters of highly occupied cells. At the end, the highly occupied clusters are marked as objects on the scene and are tracked in 2D. In this work we focus on tracking one object, so only the object with the highest certainty is tracked and the rest (if any) are dismissed. The depth detection is archieved via the knowledge of the real size of the tracked object and the fact that relative depths can be easily measured by using only one camera, also the discretization is important for the depth detection and some factors obtained via the BOF.

Experiments were conducted to measure the precision of the tracking. The system was compared with a stereo system and similar performance was measured. The system has the advantage of less implementation complexity since it does not require camera calibration nor special mounting hardware as the stereo system does. It can be used with typical webcams.

The system was integrated into a rehabilitation system named Gesture Therapy with promising results so pilot clinical tests were started.

Agradecimientos

- A CONACYT y a INAOE.
- A mi asesor Enrique Sucar, y a mi coasesor Leopoldo Altamirano.
- A Juan Manuel Ahuactzin, Kamel Mekhnacha, Emmanuel Mazer, Ronan Le Hy, Samir Ouifi y al resto del equipo de Probayes.
- A Luis, Mao y Thiago.
- A Lucy in the Sky.
- This work is partially supported by the BACS project 6th Framework Programme of the European Commission contract number: FP6-IST-027140, Action line: Cognitive Systems.

Agradezco a mi familia: Erick, Anhuar, Edith, Esther, Alfredo y a Piña.

Y también, por supuesto, a Karina.

ÍNDICE GENERAL

1. INTRODUCCIÓN

1.1. Motivación

La visión computacional tiene una amplia variedad de aplicaciones. Se puede aplicar para interfaces humano-máquina, realidad virtual, animación y captura de movimiento, incluyendo aplicaciones de realidad aumentada o seguimiento pasando por toda una lista de otras aplicaciones interesantes. Una de estas aplicaciones es la que se investiga en esta tesis: el seguimiento visual del movimiento, enfocado a rehabilitación de extremidades de personas que han sufrido accidentes cerebrovasculares.

En México se estima una cifra de alrededor de 200,000 personas cada año que sufren enfermedades cerebrovasculares (comúnmente conocidas como embolias cerebrales). Aproximadamente el 80 % de las personas que sobreviven a un derrame pierden la habilidad en el movimiento del brazo y de la mano. También es conocido que los sistemas hospitalarios tienen cada vez más presiones presupuestales, lo que los obliga a reducir los tratamientos y terapias de rehabilitación vitales para la correcta rehabilitación de los pacientes. La consecuencia es que los pacientes son enviados a casa de manera prematura algunas veces sin hacer ningún tipo de terapia. Los pacientes que intentan rehabilitarse por si mismos, pueden no hacer los ejercicios de manera adecuada debido a que no tienen preparación ni guías adecuadas como las que un fisioterapeuta puede dar, no logran avances significativos. En otros casos simplemente por pereza o falta de motivación los pacientes no realizan ejercicio alguno. El contratar un fisioterapeuta profesional no es una opción viable

para muchos de ellos por el alto costo de las sesiones terapéuticas.

Para que los sistemas de terapia autodirigida en casa lleguen a mayor cantidad de pacientes, se requiere que éstos tengan precios accesibles. Así que se requiere investigación en sistemas de rehabilitación personal de bajo costo para que de esta manera una mayor cantidad de usuarios puedan rehabilitarse en sus casas.

El presente trabajo de tesis es un avance en el área de visión computacional aplicado a reducción del precio de los sistemas rehabilitación. Se propone un sistema de rehabilitación basado en un sistema de visión por computadora. Para hacerlo simple y fácil de usar se propone usar una única cámara. De esta manera se reduce la complejidad de otros sistemas previos que también han sido basados en cámaras y se reduce a su vez el costo asociado.

Para lograr un sistema de rehabilitación de bajo costo, los retos computacionales a resolver son varios. Se requiere hacer seguimiento tridimensional que sea robusto. El sistema propuesto hará uso de una única cámara, por lo cual hay que resolver el problema de seguimiento con detección de profundidad de manera monocular. También el seguimiento debe ser tolerante a las diversas condiciones de iluminación, adicionalmente el sistema completo debe ser de fácil instalación y uso.

Diseñar un sistema para el seguimiento del movimiento humano no es una tarea trivial. Existen varias dificultades [22] [35], incluyendo ambigüedades en la profundidad, deformidades en la apariencia, complejidad en los modelos cinemáticos y oclusiones. Para simplificar los problemas del seguimiento del movimiento humano, la mayoría de los algoritmos de seguimiento emplean modelos tridimensionales de la forma de la persona, otros emplean múltiples cámaras para mejorar la robustez. Los modelos de la forma del sujeto varían desde un simple modelo de estructura de alambre [5], hasta modelos volumétricos sofisticados [6] [12] [23].

1.2. Antecedentes

Ha habido algunos avances en la investigación en sistemas de rehabilitación. Algunos sistemas han sido desarrollados, varios de ellos funcionan con sistemas de visión. Los sistemas de seguimiento visual existentes se pueden clasificar en dos categorías principales:

- **Sistemas de seguimiento visual basados en marcas**. Esta es una técnica que usa sensores ópticos (eg. cámaras), para el seguimiento del cuerpo humano, cuya imagen es capturada colocando identificadores en las articulaciones del cuerpo. Este tipo de sistemas ha atraído la atención de investigadores en la ciencia médica, los deportes y la ingeniería. Su principal desventaja es el uso de marcas que impiden la movilidad de algunas partes del cuerpo humano, así como su alto costo y su manejo especializado.

- **Sistemas de seguimiento visual libres de marcas**. Esta es una técnica que utiliza cámaras de video convencionales para la captura del movimiento humano. Estos sistemas son capaces de superar algunos problemas de oclusión y solo se concentran en los contornos, bordes o características relevantes del cuerpo humano. Sin embargo, requieren de un alto costo computacional.

Los métodos estándares para el análisis clínico del movimiento humano son los de seguimiento basados en marcas. Existen varios sistemas comerciales de captura del movimiento humano basados en marcas que pueden ser empleados para el seguimiento del movimiento de los pacientes, como son Qualisys y CODA [2][1]. Sin embargo, además de las dificultades del calibrado tanto de las cámaras como de las marcas, estos sistemas son demasiado costosos para ser usados por los pacientes en su casa, y muy complicados para que el fisioterapeuta interprete los resultados de seguimiento del paciente.

El programa de rehabilitación requiere de un sistema de seguimiento visual que sea de bajo costo, con cierta precisión y que pueda ejecutarse en tiempo real. Tanto los métodos de seguimiento visual basados en marcas y los métodos libres de marcas sólo pueden cumplir partes de estos requerimientos. En otras palabras, los sistemas de seguimiento basados en marcas pueden proporcionar la precisión de seguimiento requerida, pero son demasiado costosos. Por otro lado, los sistemas de seguimiento libres de marcas son relativamente baratos, pero su robustez y precisión necesitan ser mejoradas. Por lo tanto, es necesario crear sistemas de seguimiento visual que tomen las ventajas de ambos métodos de seguimiento.

Tao y otros [37] proponen un algoritmo de seguimiento basado en color para capturar el movimiento de partes del cuerpo humano enfocado en el proceso de rehabilitación en casa. Diferentes cintas de color son colocadas en las articulaciones de interés del cuerpo y seguidas en una secuencia de video. El rendimiento de su sistema es comparado con el sistema de seguimiento comercial Qualisys [2]. De manera similar Zhou y otros [61], proponen un sistema de seguimiento visual en tiempo real para la captura del movimiento del brazo de un individuo. Su sistema integra técnicas de visión computacional y sensores de inercia para seguir el brazo en un espacio 3D; consideran que el uso de dos fuentes de información reduce el problema de oclusión presente en los sistemas que sólo utilizan técnicas de visión.

Reinkensmeyer y otros [31] [34], desarrollaron el sistema T-WREX (siglas en inglés provenientes de *Therapy Wilmington Exosqueleton*) enfocado al proceso de rehabilitación después de un derrame cerebral. El sistema consiste en un dispositivo robótico que es fijado al brazo del paciente para poder medir su movimiento en un espacio tridimensional. La información de movimiento del brazo es enviado al software T-WREX para interactuar con un ambiente virtual, llevando a cabo diferentes tareas diseñadas para imitar situaciones de la vida real orientadas a la rehabilitación. Sin embargo, la manufactura

del dispositivo robótico tiene un alto costo, limitando así su accesibilidad. Además, el uso de un dispositivo fijado al brazo del paciente impide ejecutar movimientos de forma natural. Es evidente, de acuerdo a los trabajos de investigación recientes, el interés que existe en desarrollar sistemas computacionales que ayuden en el proceso de rehabilitación en pacientes que han sufrido un derrame cerebral. El uso de sistemas de seguimiento visual ofrece una alternativa para el proceso de rehabilitación, ya que son relativamente baratos y no se requieren mecanismos especiales para su uso. El uso de sistemas de visión estéreo no es lo más adecuado debido a la necesidad de calibrar ambas cámaras y de mantenerlas alineadas con precisión mediante soportes especiales. El uso de un sistema monocular por el contrario no requiere de ninguna calibración de la cámara y tampoco de soportes especiales y puede funcionar en las posiciones típicas en que se usan las cámaras *web*.

Por lo tanto, consideramos que es una gran motivación participar en el desarrollo de estos sistemas.

1.3. Objetivos

El objetivo general de esta tésis es el del desarrollo de un sistema de visión que provea de seguimiento monocular. El seguimiento debe ser tal que permita recuperar la posición 3D de un objeto, además el sistema debe integrarse a un sistema para rehabilitación. Para ello el sistema debe tener las siguientes características:

1. **Usar solo una cámara**. El sistema debe usar para su funcionamiento una única cámara, es decir, debe ser monocular. Esto reduce la complejidad. No se debe requerir calibración de cámara ni se requieren soportes especiales.

2. **Robustez**. Se debe tener un sistema que sea capaz de sobrellevar diversas condiciones de iluminación. El sistema de seguimiento debe también

superar movimientos bruscos de manera eventual. El sistema debe también superar problemas de oclusión y recuperarse rápidamente ante ellos. Finalmente debe superar de manera aceptable las salidas de cuadro, es decir, cuando el objeto rastreado sale del rango de visibilidad de la cámara y posteriormente regresa, inclusive si los puntos de salida y luego reentrada son distintos.

3. **Reconocimiento 3D**. Se debe tener un sistema de reconocimiento y seguimiento para un punto en la extremidad del paciente, se debe lograr saber en qué lugar se encuentra la mano en todo momento. Se deben superar los problemas para detección de profundidad de manera que una sola cámara sea suficiente detectar y hacer seguimiento en 3D.

4. **Cámara de bajo costo**. El sistema de seguimiento debe ser funcional con cámaras de bajo costo. Debe ser posible su uso con cámaras web convencionales que pueden encontrarse en cualquier tienda de accesorios de cómputo.

5. **Fácil operación**. El sistema debe ser de fácil instalación y fácil operación. En la parte mecánica ésto se cubre al usar una única cámara y al usar una cámara convencional. Además se simplifica al no requerir soportes especiales, las cámaras pueden estar en las posiciones típicas de una cámara web. En la parte de *software* se debe cuidar que el sistema de reconocimiento tenga una interfaz de usuario lo suficientemente amigable al usuario como para no presentar una curva de aprendizaje demasiado pronunciada. Para eso se hará uso de la plataforma de terapias basada en el *software* T-WREX desarrollado por Reinkensmeyer [34]. El sistema de seguimiento 3D monocular estará integrado en T-WREX.

1.4. Resultados

El sistema desarrollado consiste de un módulo de visión que recibe imágenes de una cámara montada en el monitor de una computadora personal. El módulo de visión procesa cada una de las imágenes que recibe para detectar la presencia de un objeto, enseguida da seguimiento al objeto haciendo además un estimado de la profundidad.

La detección del objeto se realiza mediante la técnica de retroproyección de histogramas usando una muestra del objeto con antelación. La retroproyección se usa como mapa de probabilidades para alimentar una rejilla que calcula índices de ocupación mediante el un Filtro de Ocupación Bayesiano.

Posteriormente se detectan cúmulos de celdas con áltos indices de ocupación sobre la rejilla del Filtro de Ocupación Bayesiano y son tomados como potenciales objetos en la escena, pero sólo el cúmulo con mayor certidumbre es tomado como un objeto. Conociendo las medidas reales del objeto y la discretización hecha se logra una estimación de la profundidad. En capítulos posteriores se detalla tanto la discretización como la evaluación de la precisión en la detección.

El sistema no requiere calibración de cámara ni soportes especiales para el correcto funcionamiento y puede funcionar en las posiciones típicas de una cámara web. Esto hace que la instalación y uso sea mas amigable al usuario y además amplía las aplicaciones potenciales.

Después de hacer las pruebas para medir la precisión en el seguimiento y en la estimación de profundidad, el sistema se integró al *software* de terapias *Gesture Therapy* (Terapia por Gestos) mediante el uso de bibliotecas dinámicas (o DLL del inglés *Dynamic Link Library*). Se obtuvieron resultados prometedores en pruebas preliminares con el sistema de terapias, a decir de los expertos medicos involucrados, por lo que se han iniciado pruebas clínicas.

1.5. Contribuciones

Ésta tesis aporta las siguientes contribuciones:

- Se diseñó y desarrollo un sistema de seguimiento visual que es capaz de detectar posiciones 3D de un objeto utlizando una sola cámara.
- Se integró el sistema monocular desarrollado en una aplicación para terapias, útil para rehabilitación de brazos.

1.6. Organización del Documento

El resto del documento esta organizado de la siguiente manera:

- CAPÍTULO 2: REHABILITACIÓN. En este capítulo se habla de trabajos anteriores a esta tesis cuyos enfoques son relacionados hacia la rehabilitación mediante el uso de diversos sistemas. Estos sistemas van desde aquellos que funcionan a través de un navegador de internet, hasta otros que funcionan con cámaras y sistemas que se apoyan de un robot para ayudar al paciente a realizar tareas.
- CAPÍTULO 3: PROCESAMIENTO VISUAL. Se hace una introducción y un repaso de los algoritmos de procesamiento visual relevantes para esta tesis. Se habla del proceso de segmentación y seguimiento y la manera en que se han venido atacando ambos problemas a lo largo de los años. Enseguida se hace un repaso de los algoritmos de detección de objetos. Se describe el proceso para segmentación y en su caso el proceso de seguimiento.
- CAPÍTULO 4: SEGUIMIENTO Y ESTIMACIÓN DE PROFUNDIDAD. En este capítulo se comienza a adentrar en el trabajo realizado en esta tesis. Se describe el sistema de detección de objetos usado, se

describen además los algoritmos de seguimiento que fueron implementados. Adicionalmente se describe la manera en que se usó la información del tamaño de los objetos para obtener datos acerca de la profundidad. Con la información de la profundidad es como se logra una aproximación de la detección 3D. Al final del capítulo se describe la manera en que este proceso de detección y seguimiento en 3D fue integrado al sistema T-WREX.

- CAPÍTULO 5: EXPERIMENTOS. En este capítulo se presentan los datos que fueron colectados durante la fase de experimentación para el seguimiento 3D. Se presentan además datos acerca de reconstrucción de trayectorias que fueron capturados durante las pruebas. También se muestran datos y gráficos de superficies 3D que fueron reconstruidas usando el sistema desarrollado. Finalmente se presenta un apartado para describir la integración con *Gesture Therapy*, que es una parte del sistema T-WREX. Se muestran gráficos y datos que muestran el desempeño del sistema desarrollado funcionando integrado a *Gesture Therapy*.

- CAPÍTULO 6: CONCLUSIONES. Se describen las conclusiones a que este trabajo ha llegado. Se mencionan también algunas alternativas para trabajo futuro.

- APÉNDICE A: DESCRIPCIÓN TÉCNICA DEL SISTEMA MONOCULAR. En este apéndice se presentan todos los datos técnicos del funcionamiento del sistema monocular. Se presentan datos del lenguaje de programación, la manera en que está organizado el software. Se presentan las funciones de las Librerías Dinámicas (DLL). Se incluye el código fuente y se especifica la manera en que se puede usar explicando ejemplos simples. Se detallan las librerías necesarias, incluyendo la manera de obtenerlas.

2. REHABILITACIÓN

En este capítulo se describen algunos sistemas de rehabilitación que han sido desarrollados en los últimos años. Estos sistemas tienen el mismo propósito principal, y es la rehabilitación de pacientes que han sufrido de embolia cerebral. Se presentan desde sistemas de terapia basados en telerehabiltación via internet que usan palancas de juegos originalmente diseñadas para simuladores de vuelo con retroalimentación, sistemas basados en exoesqueletos, hasta sistemas que son meramente evaluativos y son únicamente testigos de la rehabilitación.

Todos estos sistemas hacen uso de diversas herramientas computacionales para hacer la interacción humano-máquina. Algunos sistemas dependen de dispositivos especiales y otros hacen uso de visión por computadora.

2.1. Java Therapy

La gran mayoría de los sistemas de rehabilitación usan interacción física para lograr las metas terapéuticas [10], [9], [33]. Sin embargo, el contacto físico entre instrumentos robóticos de terapia y el paciente, o el movimiento asistido de las extremidades del paciente en terapia puede crear algunas preocupaciones de seguridad, y dichas preocupaciones están bien fundamentadas. El sistema *Java Therapy* [33] minimiza las preocupaciones de seguridad al reducir al máximo el movimiento asistido.

El sistema *Java Therapy* es un sistema económico para telerehabilitación de brazo y mano. El paciente es dirigido hacia un sitio web en donde el sistema

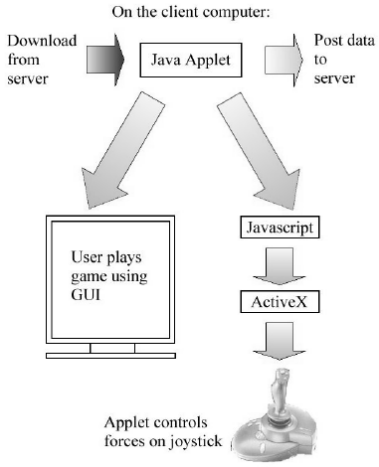

Fig. 2.1: Flujo de Procesamiento de Java Therapy. El usuario accede a la página y un subprograma (o *applet*) de Java es descargado. El *applet* recibe información de movimiento de una palanca de juegos y puede retroalimentar la palanca para que se oponga o ayude al movimiento. Al final de las terapias los datos son guardados en el servidor remoto para mantener estadísticas de los avances.

Java Therapy se encuentra instalado. El sitio web de *Java Therapy* contiene una gama de actividades evaluativas para rehabilitación. Estas actividades son registradas por el sistema el cual provee luego de una retroalimentación cuantitativa del desempeño del paciente, lo cual permite al paciente y a sus terapeutas tener una guía del progreso de la rehabilitación.

El sistema de terapia remota Java Therapy se basa en palancas de juegos con retroalimentación (*force feedback*). En la terapia, la palanca usada puede tanto oponerse al movimiento como ayudar en el movimiento del paciente. La diferencia con los sistemas robóticos es que éste usa únicamente la PC como instrumento para la rehabilitación sin tener demasiado contacto con los miembros afectados del paciente. Se puede ver un diagrama del funcionamiento en la Fig. 2.1.

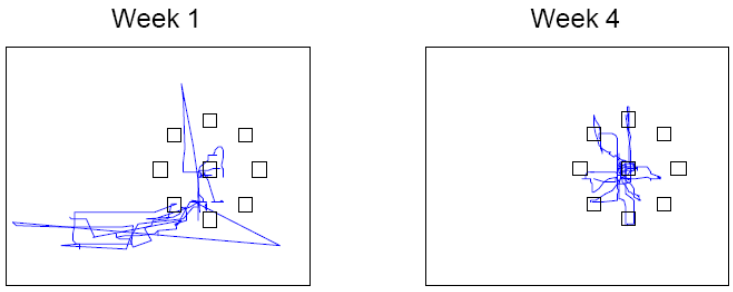

Fig. 2.2: Trayectorias en uno de las pruebas de *Java Therapy*. Se puede ver la mejora entre la primera semana (izquierda) y luego de un mes de uso (derecha). El usuario debía moverse entre el centro y los cuadros exteriores. Note la mayor precisión lograda luego de usar el sistema.

El advenimiento de la tecnología de retroalimentación (*force feedback*) fue lo que propició el desarrollo del sistema *Java Therapy* conjuntamente con la ubicuidad y el fácil acceso remoto a Internet. Los sistemas comerciales de retroalimentación diseñados principalmente para juegos pueden medir el movimiento de las personas y pueden también aplicar fuerzas durante el movimiento. En su mercado principal, los juegos, sirven para dar realismo a la escena, pero usados en terapias pueden ser usados para estimular el sentido del tacto y el movimiento. Por lo anterior, los sistemas de retroalimentación pueden aplicar patrones de movimientos terapéuticos de fuerzas a la mano y brazos de los pacientes. Los sistemas de retroalimentación también tienen el añadido de ser mucho más económicos que los sistemas robóticos y pueden ser usados en mayor medida.

Los componentes mecánicos del sistema *Java Therapy*, consisten en un clip de sujeción, un soporte de brazo, una base especial para el sistema y la palanca de retroalimentación. Todos los componentes mecánicos pueden ser instalados en una mesa de tamaño regular. El *software* consiste en el sitio Web con un programa escrito en lenguaje Java (`http://www.javatherapy.com`)

el cual se encarga de llevar registro de los avances de la rehabilitación.

Muchos usuarios con impedimentos de movimiento de brazos y manos fueron tratados durante la etapa de experimentación de *Java Therapy* mostrando, en general, mejoría en su movimiento, se puede notar la mejoría de un paciente en la Fig. 2.2.

2.1.1. Limitaciones

El sistema *Java Therapy* está limitado en cuanto a que sólo puede rehabilitar parte de la mano y músculos con los que se realizan movimientos de la palanca de juegos. Adicionalmente está limitado en cuando a la inmersión y la interacción del paciente con los juegos de terapias, ya que son muy simples y poco creativos, lo que puede repercutir en el entusiasmo y motivación del paciente.

De manerá técnica el sistema también tiene desventajas en cuanto a crecimiento o expansión, ya que el sistema está desarrollado para funcionar dentro de un navegador de internet y ello reduce sus posibilidades de acceder a todo el potencial de recursos de una computadora. El sistema se ve restringido por los recursos (memoria, acceso a hardware, y otros) que el navegador de internet pueda ofrecerle. Además podría no ser compatible con futuros navegadores, y dada la rapidez con la que evoluciona dicho software puede convertirse en una preocupación.

2.2. T-WREX

Existen estudios que indican que la razón por la que las personas no tienen iniciativa para comenzar una terapia, y asimismo la deserción en el seguimiento de terapias luego de comenzarlas es debido a la falta de motivación [25]. En el sistema original Therapy-WREX (o T-WREX) [34] se provee al usuario de una motivación para su rehabilitación mediante juegos.

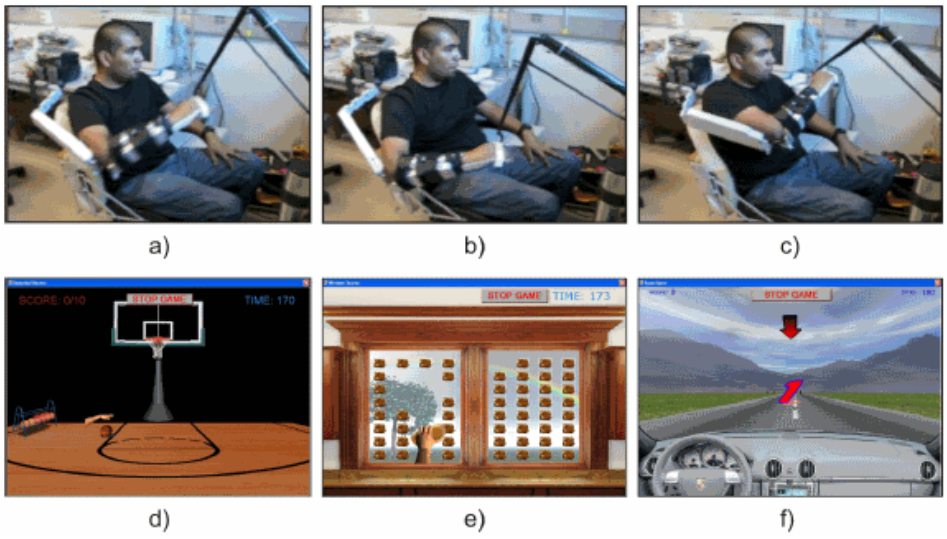

Fig. 2.3: Paciente usando el sistema T-WREX. **Arriba (a, b y c).** Uso del sistema de soporte al brazo. **Abajo (d, e y f).** Tres juegos para terapias: juego de baloncesto, juego de limpieza de vidrios y juego de carrera de autos.

T-WREX se apoya en un sistema de exoesqueleto con ayudas gravitatorias pasivas para el paciente, de tal manera que no tenga que mover todo el peso de sus propias extremidades.

El sistema T-WREX permite a los individuos con severos daños practicar movimientos de brazos sin necesidad de la asistencia o supervisión contínua de un terapeuta. Las partes mecánicas consisten de una órtosis que asiste al movimiento dentro de un gran espacio de movimiento, tiene también un sensor de presión que detecta la presión de la mano y en la parte de *software* hay un programa que simula diversas actividades.

La órtosis es una versión de adultos de el Wilmington Robotic Exoskeleton (WREX), el cual es un mecanismo de 5 grados de libertad que balancea el peso del brazo usando bandas elásticas.

Se ha completado diversas etapas en el desarrollo de T-WREX [31] [34], en las cuales se ha demostrado en estudios las capacidades del sistema.

En un estudio se demostró que los sujetos con embolia cerebral cuyos movimientos de brazo habían sido severamente reducidos en un ambiente de gravedad normal, podían realizar movimientos para alcanzar objetos y para hacer dibujos mientras se encontraban usando el sistema T-WREX. También se ha demostrado que al ejercitar los brazos afectados de cinco sujetos con T-WREX durante un periodo de ocho semanas el movimiento sin asistencia mejoró pues se registró una mejora en la puntuación Fugl-Meyer [3] (5 puntos ± 2 SD, el cambio medio en el movimiento para alcanzar objetos fue de 10 %, $p < 0{,}001$). La puntuación de la escala Fugl-Meyer representa un índice especial para pacientes de embolia. Está diseñada para medir el funcionamiento motriz, balance, sensación y el funcionamiento de articulaciones en pacientes que ha sufrido embolias. Se aplica en ambientes clínicos y en investigación para determinar la severidad del padecimiento, para describir la recuperación motriz y para planeación de terapias y tratamientos.

Estos resultados mostraron la viabilidad en la automatización de la rehabilitación terapéutica, incluso en casos severos, usando un sistema de asistencia pasiva de gravedad, un sensor de presión y un sistema simple de realidad virtual que simula diversas actividades mediante juegos.

Mientras el usuario realiza estos juegos está obligando a sus extremidades a alcanzar movimientos similares a los que una terapia puede ofrecer. Se intenta con T-WREX mejorar la movilidad de los brazos de manera significativa sin la ayuda de un terapeuta. Además se han reportado buenos resultados con respecto a la motivación de los pacientes. Se reporta que los pacientes prefieren fuertemente el uso de T-WREX que la terapia autónoma con ejercicios auto dirigidos.

Se sigue investigando acerca de la rehabilitación automatizada en el Instituto de Rehabilitación de Chicago para establecer la utilidad terapéutica y las condiciones de seguridad del dispositivo, antes de poder comercializarlo para uso casero.

2.2.1. Limitaciones

El sistema T-WREX tiene la principal desventaja en el alto costo que supone, por lo cual es útil únicamente para hospitales. El alto costo es resultado de la implementación del sistema del exoesqueleto. Difícilmente una persona puede comprarse el sistema completo incluyendo el exoesqueleto, y aunque lo hiciere sería una inversión económicamente fuerte.

Se puede contar también como limitación el que el sistema requiera de un exoesqueleto, lo que lo hace un sistema aparatoso y de difícil instalación.

2.3. *Gesture Therapy*

En recientes trabajos se extendió el sistema de T-WREX para hacerlo más amigable y menos intrusivo intentando encaminarlo hacia la comercialización casera sin preocupaciones de seguridad. Se reemplazó la órtosis WREX con un sistema de rastreo mediante el uso de visión estéreo *Gesture Therapy* [27]. El no usar ningún método de asistencia activa al movimiento de los brazos diluye las preocupaciones de seguridad del paciente.

Mediante el uso de dos cámaras (sistema estéreo) y una computadora, la mano del usuario es detectada y rastreada en una sucesión de imágenes para obtener sus coordenadas 3D en cada imagen. Luego esta información es enviada al modulo de *Gesture Therapy* del sistema T-WREX. El proceso conlleva varias etapas:

1. **Calibración**. Para tener una estimación precisa de la posición 3D de la mano en el espacio, el sistema estéreo se calibra. La calibración consiste en obtener los parámetros intrínsecos de la cámara (longitud focal, tamaño de pixel) y los parámetros extrínsecos (posición y orientación).

   Los parámetros intrínsecos se obtienen mediante un patrón de referencia tipo tablero de ajedrez que se coloca enfrente de cada una de las

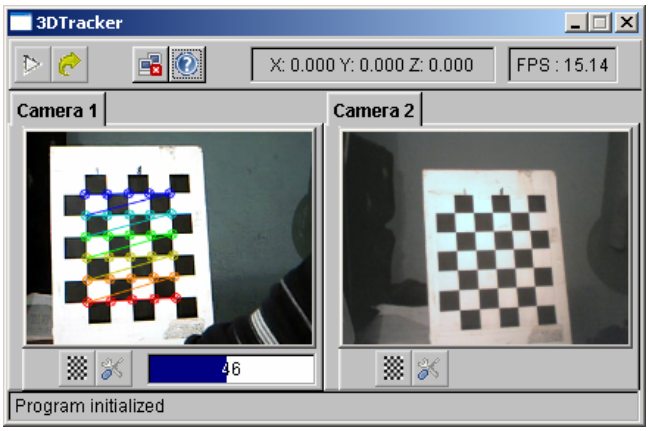

Fig. 2.4: Patrón de calibración usado para obtener los parámetros intrínsecos de las cámaras.

dos cámaras, ver Fig. 2.4.

Los parámetros extrínsecos se obtienen insertando las coordenadas en las cuales está localizada la cámara de acuerdo a un sistema de referencia fijo.

2. **Segmentación.** La mano del paciente se localiza en la imagen inicial combinando información de color y movimiento. El color de la piel es una buena pista de las regiones potenciales en donde se encuentra el rostro y mano de los pacientes. El procedimiento es entrenar un clasificador bayesiano con miles de muestras de piel, las cuales se representan como valores de pixel en el espacio de color HSV. También se hace segmentación basada en movimiento y se usan ambos para inicializar la posición de la mano.

3. **Rastreo.** Luego de tener la aproximación inicial de la posición de la mano en ambas cámaras se hace rastreo de la mano. El rastreo se basa en el algoritmo CAMSHIFT [8]. El cual usa la información de color obtenida mediante la segmentación para rastrear el objeto, en este caso

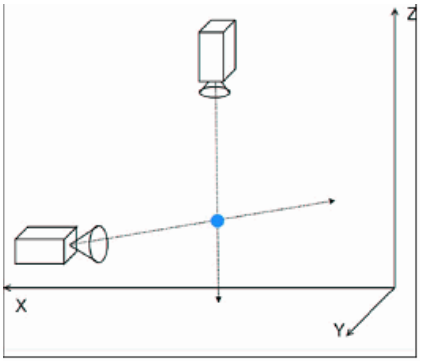

Fig. 2.5: Estimación de la posición 3D de la mano en el sistema *Gesture Therapy* mediante el uso de 2 cámaras.

la mano. Se puede leer una descripción con más detalle de CAMSHIFT y de MEANSHIFT en el capítulo 3 de esta tesis.

4. **Reconstrucción 3D.** Basados en las coordenadas 2D del punto central de la región que se detecta en cada imagen proveniente de la cámara, se obtienen las coordenadas 3D de la siguiente manera. Para cada imagen se toma en cuenta una línea desde la región detectada que pasa por el centro de la lente de la cámara en cuestión, para esto se toman como base en los parámetros intrínsecos y extrínsecos.

   Al tener dos cámaras se tienen dos líneas y su intersección provee de una aproximación de las coordenadas 3D, Fig. 2.5.

El sistema de visión estéreo de *Gesture Therapy* logra procesar un promedio de 15 cuadros por segundo.

Se realizó un estudio clínico con el sistema estéreo de *Gesture Therapy* [27]. En dicho estudio realizado por el Instituto Nacional de Neurología y Neurocirugía (INNN) en la Cd. de México se probó el sistema con un paciente antes de realizar pruebas médicas de mayor escala, intentando anticipar problemas potenciales. Se obtuvo experiencia en el uso de la tecnología sobre pacientes reales en ambientes y condiciones médicas reales.

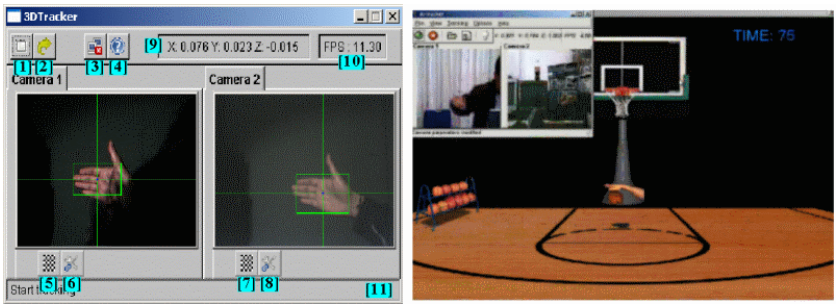

Fig. 2.6: **Izquierda**: *Gesture Therapy* estéreo durante la detección de piel, note el recuadro donde se ha detectado una mano. **Derecha**: *Gesture Therapy* estéreo durante la detección y mientras se ejecuta un juego útil para rehabilitación, el juego de baloncesto donde se puede usar la mano detectada para tomar la pelota y encestar.

Un paciente diagnosticado con embolia isquémica y hemipáresis izquierda con una evolución de 4 años fue evaluado con la escala Fugl-Meyer [3] al inicio y al final del estudio. El paciente usó el sistema estéreo de *Gesture Therapy* durante 6 sesiones de entre 20 y 45 minutos cada una. El principal objetivo de los ejercicio era el control de la porción distal de la extremidad superior y de la mano. El sujeto realizó pre ejercicios para estiramiento, relajación y contracción de dedos, flexores y extensores de muñeca. El paciente realizó diversos ejercicios simulados en el ambiente virtual con grado de dificultad cada vez mas alto.

Después de 6 sesiones el paciente había incrementado su capacidad de extender y flexional de manera voluntaria la muñeca; el paciente incrementó el uso de su extremidad usándola para cerrar puertas.

Basados en los resultados del estudio piloto, la evaluación del terapeuta y la evaluación del paciente se tienen indicios que el sistema *Gesture Therapy* ha dado resultados alentadores. El paciente recuperó en cierto grado la movilidad y el control, los cuales se vieron reflejados de manera positiva en su vida.

2.3.1. Limitaciones

El sistema de visión estéreo requiere la calibración de ambas cámaras, lo cual debe hacerse de manera cuidadosa. Además el sistema requiere un soporte metálico en el cual se colocan las cámaras, el soporte mide aproximadamente $1{,}5 \times 1{,}5$ metros de base por 2 metros de altura, lo que lo hace estorboso. Luego de instaladas, se mide la rotación de las cámaras y la posición 3D con respecto a un eje establecido. Sin la calibración el sistema no puede funcionar. Si el sistema es movido se requiere recalibración, lo cual obliga a mantener el sistema en condiciones especiales para no ser movido accidentalmente.

En esta tésis se sustituyó el sistema estéreo de *Gesture Therapy* por el sistema monocular desarrollado, logrando simplificarlo y facilitar su uso.

2.4. Otros Sistemas

Un robot puede usar un sistema de interacción con humanos puede ser usado para interacción con el usuario y ayudarlo en los ejercicios de recuperación de miembros afectados [18] mediante la observación, programación de recordatorios y análisis de los movimientos. En este sistema la observación y análisis de los movimientos del humano son la herramienta básica para la exitosa interacción.

En [4] se describe un robot móvil que ayuda a la rehabilitación del paciente vigilándolo, incitando a realizar ejercicios y dando recordatorios. El robot vigila la actividad del usuario y le ayuda a mantener un programa de rehabilitación. Este robot está dirigido en especial a las personas que han sufrido una embolia cerebral. En ese trabajo se investigó el rol que juega la personalidad de un robot para la asistencia sin intervención en el proceso terapéutico. Se enfocó en la personalidad introvertida o extrovertida de un robot y del usuario. Los resultados experimentales muestran los primeros intentos

para descubrir las relaciones en las personalidades. También se hicieron intentos para averiguar el tipo de terapias que pueden conseguirse si el robot se ajusta a la personalidad del paciente.

Este tipo de robots son testigos de la rehabilitación, se dedican a monitorizar la actividad y a sugerir la realización de terapias. No asisten en ningún momento la realización y sólo pueden parcialmente averiguar si el usuario está realizando cierta actividad.

En [30] se propuso, en lugar de comparar directamente los movimientos de los gestos, el entrenar un modelo basado en las trayectorias para hacer la comparación. Se usaron Modelos Ocultos de Markov [28] (*o por sus siglas HMM, del inglés Hidden Markov Models*) para calificar los gestos realizados. Un *HMM* es entrenado basado en modelos de referencia para un gesto correcto. Enseguida, muestras del gesto que se quiere evaluar se usan para entrenar un segundo HMM. Finalmente, ambos HMM son comparados y se califica el gesto mediante métricas (Levinson, Kullback-Leibler y Porikli). Un sistema estéreo similar al de [27] fue usado. Ambas cámaras se colocaron de manera ortogonal al sujeto que realiza los gestos.

En el trabajo se usan marcadores de diferente color en las principales articulaciones del sujeto: hombro, codo y muñeca. Usando detección de profundidad basada en estéreo para cada uno de ellos se obtiene una aproximación de la profundidad.

Los resultados se compararon con escalas que se se usan en terapias, en particular el índice de motricidad y la escala Fugl-Meyer. Los experimentos fueron realizados en el Instituto Nacional de Neurología y Neurocirugía en la Cd. de México. Debido a los prometedores resultados se dedujo que el sistema podría ser útil para dar retroalimentación a un paciente que está en rehabilitación. El sistema tiene las mismas limitaciones que el sistema estéreo para *Gesture Therapy*, ya que se requiere el mismo soporte de las cámaras y el mismo tipo de calibración.

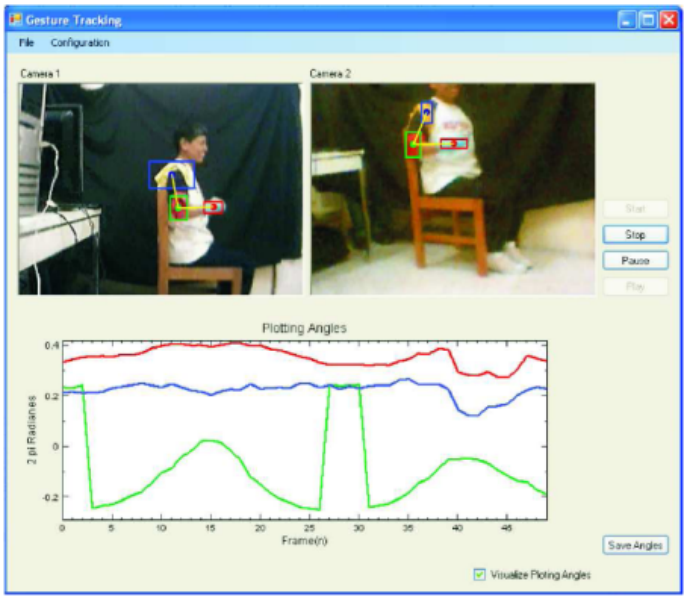

Fig. 2.7: Sistema de rastreo 3D de articulaciones.

2.5. Conclusiones

En este capítulo se ha dado un breve panorama de varios sistemas de rehabilitación que se han venido desarrollando por diversos autores. Algunos funcionan de manera remota a traves de internet, otros son exoesqueletos que proporcionan ayuda al paciente y otros más son sistemas de evaluación de movimiento. Uno de ellos resalta en particular, *Gesture Therapy*, que usa un par de cámaras para visión estéreo, al respecto, esta tesis simplifica el uso de *Gesture Therapy* al integrar el sistema monocular desarrollado, logrando una estimación robusta de la profundidad sin necesidad de calibración ni soportes especiales.

En el siguiente capítulo se dará un repaso de las técnicas de procesamiento visual usadas para segmentación y seguimiento de objetos. Se hace énfasis en las que fueron de utilidad para lograr el seguimiento monocular.

3. PROCESAMIENTO VISUAL

En este capítulo se detallan las herramientas de procesamiento visual que son útiles para detección de objetos. Se explica que son los modelos de color, su aplicación a la segmentación de imágenes y algunos algoritmos de seguimiento de objetos.

3.1. Modelos de Color

Un modelo de color es un modelo matemático abstracto que describe la forma en que los colores pueden ser representados como tuplas de números, generalmente se usan tres o cuatro valores o "componentes" de color. Cuando el modelo de color es asociado a una descripción precisa de la manera en que los componentes deben ser interpretados (i.e. condiciones para verlo y otros), el resultado es llamado un espacio de color.

Existen diversos modelos de color, en esta tesis se usó el modelo HSV debido a que permite separar de manera eficiente la información de color, llamada 'tonalidad', sin ser afectada por variaciones en la luminosidad del color.

3.1.1. Modelo de color HSV

El llamado modelo HSV (del inglés *Hue, Saturation and Value*) define un modelo de color en términos de sus componentes constituyentes en coordenadas cilíndricas. Este modelo de color, debido a la manera en que representa

el color, permite discernir de manera eficiente los colores puros sin distraerse por variaciones de luminosidad.

- TONALIDAD. El tipo de color (como rojo, azul o amarillo). Se representa como un grado de ángulo cuyos valores posibles van de 0 a 360 (aunque para algunas aplicaciones se normalizan del 0 al 100 %). Cada valor corresponde a un color. Ejemplos: 0 es rojo, 60 es amarillo y 120 es verde.

- SATURACIÓN. Se representa como la distancia al eje de brillo negro-blanco. Los valores posibles van del 0 al 100 %. A este parámetro también se le suele llamar "pureza" por la analogía con la pureza de excitación y la pureza colorimétrica. Cuanto menor sea la saturación de un color, mayor tonalidad grisácea habrá y más decolorado estará. Por eso es útil definir la insaturación como la inversa cualitativa de la saturación.

- VALOR DEL COLOR. El brillo del color. Representa la altura en el eje blanco-negro. Los valores posibles van del 0 al 100 %. El cero siempre es negro. Dependiendo de la saturación, 100 podría ser blanco o un color más o menos saturado.

El modelo de color HSV puede verse como un cono invertido de acuerdo a la Fig. 3.1. En la figura se nota el uso de coordenadas cilíndricas (r, θ, z) en la representación del modelo HSV, siendo el cono centrado en el origen, el ángulo θ representa el Hue, la distancia desde el origen r representa la Saturación y la coordenada z el Valor. Cuando lo que interesa es rastrear el color del objeto, la coordenada más importante es la θ, que representa la tonalidad o *Hue*. Mediante la elección de un pequeño conjunto de valores contiguos del Hue se puede seleccionar un color puro de una imagen, sin distracciones por variaciones en la luminosidad.

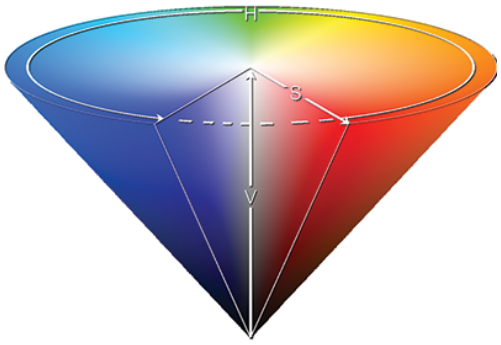

Fig. 3.1: El modelo HSV puede verse como un cono invertido y usando coordenadas cilíndricas. El valor angular (θ) representa la tonalidad (H), el radio (r) representa a la saturación (S) y el valor (V) está representado por la altura (z).

El modelo HSV fue creado en 1978 por Alvy Ray Smith [20]. Se trata de una transformación no lineal del espacio de color RGB. Para definir la transformación de color de RGB hacia HSV primero definimos M y m de la siguiente manera:

$$M = Max(R, G, B) \tag{3.1}$$

$$m = Min(R, G, B) \tag{3.2}$$

Enseguida la obtención de H, S y V se hace usando las siguientes transformaciones:

$$H = \begin{cases} 60 \times \frac{G-B}{M-m} + 0 & \text{si } M = R \text{ y } G \geq B \\ 60 \times \frac{G-B}{M-m} + 360 & \text{si } M = R \text{ y } G < B \\ 60 \times \frac{B-R}{M-m} + 120 & \text{si } M = G \\ 60 \times \frac{R-G}{M-m} + 240 & \text{si } M = B \\ \text{No definido} & \text{si } M = m \end{cases} \tag{3.3}$$

$$S = \begin{cases} 0 & \text{si } M = 0 \\ 1 - \frac{m}{M} & \text{si } M \neq 0 \end{cases} \tag{3.4}$$

$$V = M \tag{3.5}$$

3.2. Detección

Diversas técnicas se han desarrollado a lo largo de los años para intentar dar seguimiento a objetos de color [19]. Algunos de ellos buscan detectar rostros [24]. Hay algunos métodos muy elaborados donde rastrean contornos con *snakes* y algunos otros usan *eigenespacios* o hipótesis estadísticas, entre ellos hay algunos que hacen convoluciones de imagen con detectores de características.

3.2.1. Convolución

La convolución de una muestra de color sobre una imagen da como resultado un mapa de probabilidades que se representa mediante una matriz que puede verse como una imagen en escala de grises. Entre más alta la probabilidad de que el color esté presente mayor es el valor de la entrada en la matriz, lo que se traduce en la imagen en pixels con mayor "cantidad" de blanco. Al final se tiene como resultado una imagen en escala de grises (Fig. 3.2) que puede ser tratada con técnicas de umbralización, erosión y dilatación para eliminar el posible ruido.

El ruido se puede eliminar hasta cierto punto, sin embargo no se pueden eliminar otros objetos que se encuentren en la escena y que tengan colores que correspondan a los colores representados en la muestra. Lo anterior implica que el color debe seleccionarse de manera adecuada.

Una convolución es una integral que expresa la cantidad de "*superposición*" de una función g mientras se desliza sobre otra función f. Es decir, una convolución "*superpone*" una función sobre otra y mide los resultados.

De manera abstracta, una convolución se define como un producto de funciones $f, g \in \mathbb{R}^n$. La convolución de las funciones f y g sobre un rango finito $[0, t]$ esta definida por:

$$f * g \equiv \int_0^t f(\tau) g(t - \tau) d\tau \tag{3.6}$$

donde $f * g$ (a veces denotado por $f \otimes g$) denota la convolución de f y g.

Cuando se usan funciones discretas para aplicaciones computacionales, la función f es representada por el histograma de una muestra de color. La función g es un cuadro (frame) tomado de la cámara, representado también por el histograma de color asociado a cada pixel. Entonces la convolución de la muestra (f) sobre el cuadro (g) a lo largo de todos los pixeles da como resultado un mapa de distribución de que tan bien la muestra se parece al los pixeles. Esto es lo que en términos computacionales se conoce como la retroproyección, es decir el resultado de la convolución es una función de la distribución de la muestra de color sobre el cuadro. De esta manera podemos calcular la probabilidad de que el color se encuentre sobre ciertas zonas.

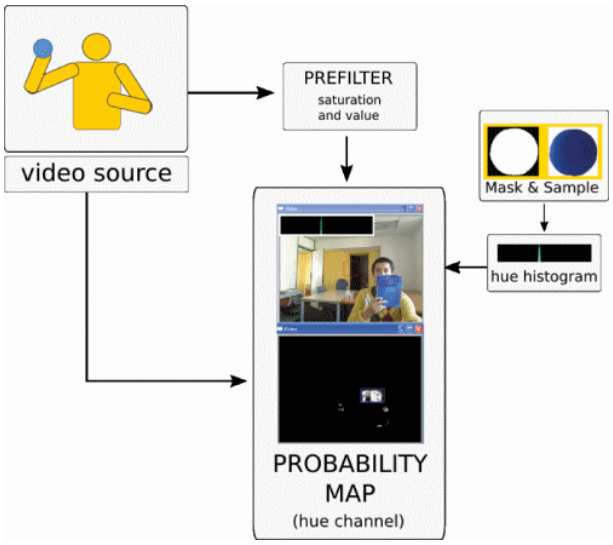

Fig. 3.2: Convolución de una muestra de color sobre una imagen. Se obtiene una imagen desde alguna fuente, en este caso un video. Con antelación se ha tomado una muestra del objeto de color a seguir a la cual se le calcula un histograma. Se obtiene una retroproyección o mapa de probabilidades al hacer una convolucion del histograma de la muestra sobre la imagen. Mientras más blanco, mayor probabilidad de que el color esté presente.

3.3. Seguimiento

3.3.1. Blobs

La mayoría de las formas de seguimiento involucran como primer paso la segmentación del fondo de la escena para detectar objetos que se mueven con respecto a ésta. El resultado de este procesamiento inicial es que, para cada imagen, se obtiene un conjunto de regiones etiquetadas. Estas regiones etiquetadas iniciales son comúnmente llamadas *blobs*. La detección de *blobs* en visión se refiere a aplicar operaciones enfocadas a segmentar la imagen y detectar regiones en la imagen que son de interés especial, los *blobs*. Usualmente se detectan mediante cambios en las propiedades de los elementos de la imagen, como el brillo, contraste, saturación de color, etc; dependiendo del tipo de modelo de color asociado a la imagen.

Es importante mencionar que hay terminología inexacta en cuando a los términos *blob detection, blob extraction, region labeling, connected-component labeling, blob discovery* y *region extraction*. Todos parecen referirse a la técnica de seguimiento por *blobs*, aunque a veces tienen definiciones ligeramente distintas dependiendo de los autores. Algunos de ellos incluso colisionan con definiciones de otras áreas.

El establecer etiquetas consistentes para los *blobs* es una tarea que se resuelve por lo regular en la etapa de seguimiento de objetos. Se pueden distinguir diferentes formas de seguimiento, dependiendo del nivel semántico de las entidades a dar seguimiento y la cantidad de conocimiento que se tiene del objeto en específico. Algunos algoritmos intentan seguir objetos muy específicos en situaciones muy definidas, lo cual facilita la tarea.

El seguimiento de *blobs* se aplica en el nivel semántico más bajo posible; es decir, en el sentido de que establecen relaciones temporales entre características segmentadas de una imagen sin el uso de información dependiente del dominio. Los algoritmos de seguimiento por *blobs* están en el segundo

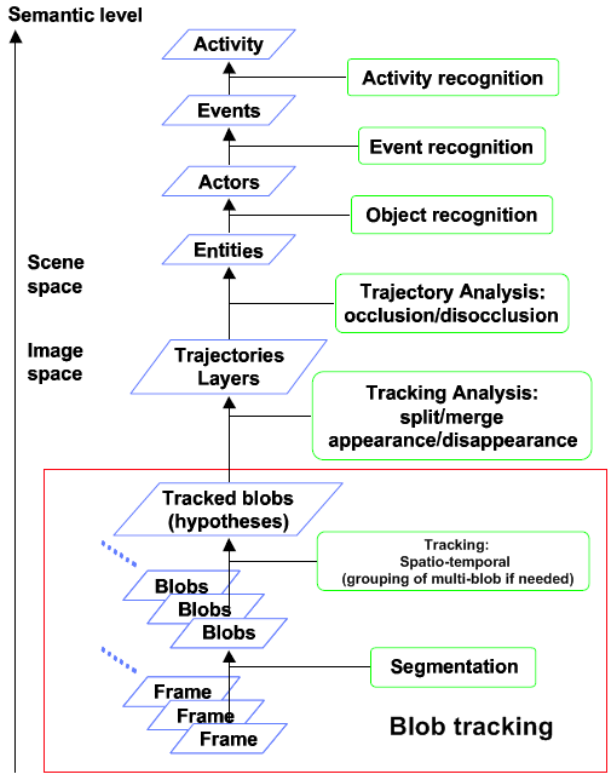

Fig. 3.3: Niveles semánticos. Hasta abajo está el nivel semántico que representa las imágenes, mientras mas arriba el nivel, mayor relación con el mundo real se tiene. El seguimiento por *blobs* se a realiza en el nivel mas bajo, justo después de la segmentación [21].

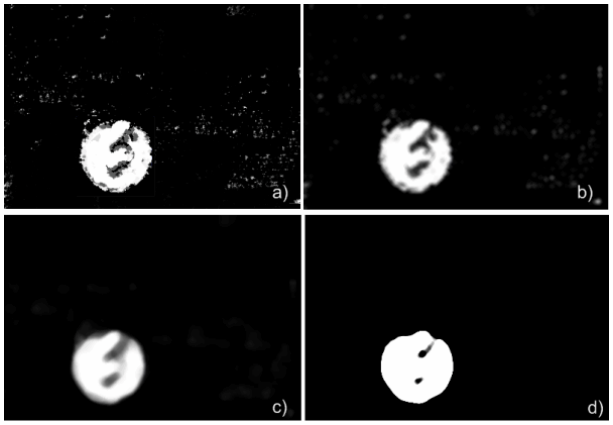

Fig. 3.4: Procesamiento de una imagen para obtener un *blob*. *a)* Imagen original, la mancha redonda imperfecta representa un objeto redondo. Note el ruido. *b)* Procesamiento de la imagen original mediante un algoritmo de eliminación de ruido. Note que no se pudo eliminar el ruido por completo. *c)* Procesamiento mediante algoritmo que calcula las medianas de los valores. Note que un poco más de ruido fue eliminado. *d)* Procesamiento final mediante umbralización. El ruido desapareció por completo y se tiene una única mancha. Los huecos en la mancha pueden ser eliminados mediante algoritmos para crecimiento de región.

escalón en la escalera semántica, justo después de la segmentación inicial. En particular el concepto de "*capa*" y por lo tanto el concepto de *oclusión* son irrelevantes. Vea la Fig. 3.3.

Sin el concepto de *oclusión*, los *blobs* pueden aparecer y desaparecer, separase y unirse (en un espacio bidimensional representado por la imagen). La aparición y desaparición de los *blobs* puede ser causada por ruido, por reflexiones y sombras. Por lo anterior la única manera de usar algoritmos de seguimiento basados en *blobs* es en condiciones quasi-ideales en las cuales se tiene cierto grado de certeza que al finalizar el proceso inicial de segmentación sólo se obtendrá un único blob y un bajo nivel de ruido.

Existen técnicas de procesamiento de imágenes que son de utilidad en la eliminación de ruido, algunas otras son útiles para evitar la separación de un *blob* cuando la detección sufre distorsión.

Cuando se tienen imágenes de calidad suficiente, es decir, con poco nivel de ruido y se tiene una segmentación buena del objeto a seguir, se puede aplicar un simple seguimiento de *blobs* mediante el siguiente algoritmo.

- **Preprocesar imagen**. Se prepara la imagen para la detección de blobs, tal vez haciendo convoluciones o filtrados de color previos. Se aplican técnicas de procesamiento de imágenes para suavizar o marcar bordes, para eliminar ruido y para aplicar umbrales.
- **Detectar blob en la imagen**. Se detecta el *blob* en la imagen.
- **Actualizar coordenadas**. Mediante alguna convención de asignación de coordenadas se miden y actualizan las coordenadas del *blob*.
- **Leer otra imagen y repetir**. Se repite en el siguiente cuadro, y de esta manera se obtiene el seguimiento.

Los sistemas de seguimiento basados en *blobs* tienen la ventaja de ser muy simples pues no se requiere ningún otro procesamiento. Sin embargo, tienen la seria desventaja de que requieren de una segmentación muy precisa de la imagen, que en la mayoría de los casos no es posible.

Los sistemas de seguimiento por *blobs* se pueden usar en ambientes muy controlados como en sistemas robóticos industriales de aplicaciones específicas en los cuales se tiene garantía de que las imágenes recibidas serán siempre muy homogéneas y las condiciones de iluminación serán siempre ideales.

3.3.2. MEAN-SHIFT

El algoritmo llamado *Mean-shift* es un importante algoritmo de agrupamiento originalmente propuesto por Fukunaga y Hostleter [26], al cual

llamaron "*valley-seeking procedure*". A pesar de su excelente desempeño fue prácticamente olvidado hasta que Cheng [40] lo extendió y lo introdujo a la comunidad de análisis de imágenes. Recientemente Comanciu y Meer [14] [15] lo aplicaron a la segmentación y seguimiento. DeMenthon [16] tuvo éxito al aplicarlo a la segmentación espacio-temporal de secuencias de video en un espacio de características de 7 dimensiones.

Mean-shift es esencialmente un análisis de puntos de datos basado en características, las cuales requieren un estimador no paramétrico del gradiente de la densidad en el espacio de características. Algunas ventajas del espacio de características son la representación global de los datos originales y la excelente tolerancia al ruido [32]. Cuando una función de densidad en el espacio de características tiene picos y valles es deseable acotar puntos de datos en cúmulos de acuerdo a los valles de las densidades del punto, porque tales acotamientos son mapeados a fronteras de segmentación mucho más naturales. Se puede ver un ejemplo de segmentación en la Fig. 3.5.

El procedimiento de *Mean-shift* consiste de dos pasos: (i) la estimación del gradiente de la función de densidad, y (ii) el uso de los resultados para formar cúmulos. El gradiente de la función de intensidad es estimado por un estimador no paramétrico [32]. Entonces, comenzando de cada punto muestra el algoritmo *Mean-shift* iterativamente encuentra un camino a lo largo de la dirección del gradiente alejándose de los valles y aproximándose al pico más cercano. El procedimiento estándar de *Mean-shift* es tomar siempre la pendiente más alta (*Steepest Ascent Hill Climbing*) para escalar los picos y encontrar puntos estacionarios. Se puede ver una imágen a la cual se le aplicó segmentación usando *Mean-shift* en la Fig. 3.5.

Analizando Mean-shift

Dados n puntos de datos, $x_1, x_2, \ldots, x_n$ en el espacio d-dimensional R^d, el *kernel estimador de densidad* con función de kernel $K(x)$ y una ventana de ancho h está dado por [32][17][29]:

$$\hat{f}_n(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{3.7}$$

Donde el kernel d-variado $K(x)$ es no negativo y su integral da uno. Una clase de kernels ampliamente usada son los radialmente simétricos:

$$K(x) = c_{k,d} k \left|\left|x\right|\right|^2 \tag{3.8}$$

donde la función $k(x)$ es llamada el $perfil$ del kernel, y la constante $c_{k,d}$ es una constante de normalización que hace que $K(x)$ se integre a 1. El estimador de densidad (3.7) se puede entonces reescribir como:

$$\hat{f}_{h,k}(x) = \frac{c_{k,d}}{nh^d} \sum_{i=1}^{n} k\left(\left|\left|\frac{x - x_i}{h}\right|\right|^2\right) \tag{3.9}$$

donde $c_{k,d}$ es la constante de normalización. Dos kernels comúnmente usados aquí son el kernel de Epanechikov:

$$K(x) = \begin{cases} \frac{1}{2} c_d^{-1}(d+2)(1 - ||x||^2) & 0 \leq ||x|| \leq 1 \\ 0 & ||x|| > 1 \end{cases} \tag{3.10}$$

y el kernel multivariado Gaussiano:

$$K_N(x) = (2\pi)^{-d/2} e^{\frac{-||x||^2}{2}} \tag{3.11}$$

En el procedimiento *Steepest Ascent Hill Climb*, el cual es el estándar original de *Mean-shift* se requiere una estimación del gradiente:

$$\nabla \hat{f}_{h,K}(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (x_i - x)\, g\left(\left|\left|\frac{x - x_i}{h}\right|\right|^2\right) \tag{3.12}$$

$$= c_{k,g} \hat{f}_{h,G}(x) \left[\frac{\sum_{i=1}^{n} x_i g(\left|\left|\frac{x-x_i}{h}\right|\right|^2)}{\sum_{i=1}^{n} g(\left|\left|\frac{x-x_i}{h}\right|\right|^2)} - x\right] \tag{3.13}$$

donde $g(x) = -k'(x)$ la cual puede ser en su caso usada para definir un kernel $G(x)$. El kernel $K(x)$ es llamado *la sombra* de $G(x)$ [40]. $\hat{f}_{h,G}(x)$ es la estimación de la densidad con el kernel G. $c_{k,G}$ es el coeficiente de normalización. El último término es el *Mean-shift*:

$$m(x) = \left[\frac{\sum_{i=1}^{n} x_i g(\left|\left|\frac{x-x_i}{h}\right|\right|^2)}{\sum_{i=1}^{n} g(\left|\left|\frac{x-x_i}{h}\right|\right|^2)} - x\right] \tag{3.14}$$

el cual es proporcional al gradiente de densidad normalizado y siempre apunta hacia la dirección cuya pendiente es máxima sobre la función de densidad, es decir, hacia la "punta" de la colina. El algoritmo estándar de *mean-shift* iterativamente hace lo siguiente:

- Calcular el vector *mean-shift*: $m(x^k)$
- Actualizar la posición actual $x^{k+1} = x^k + m(x^k)$

hasta que alcanza un punto estacionario, el cual es el candidato a ser el centro del cúmulo.

En la siguiente sección se detallan algoritmos de seguimiento, CAMSHIFT y el Filtro de Ocupación Bayesiano.

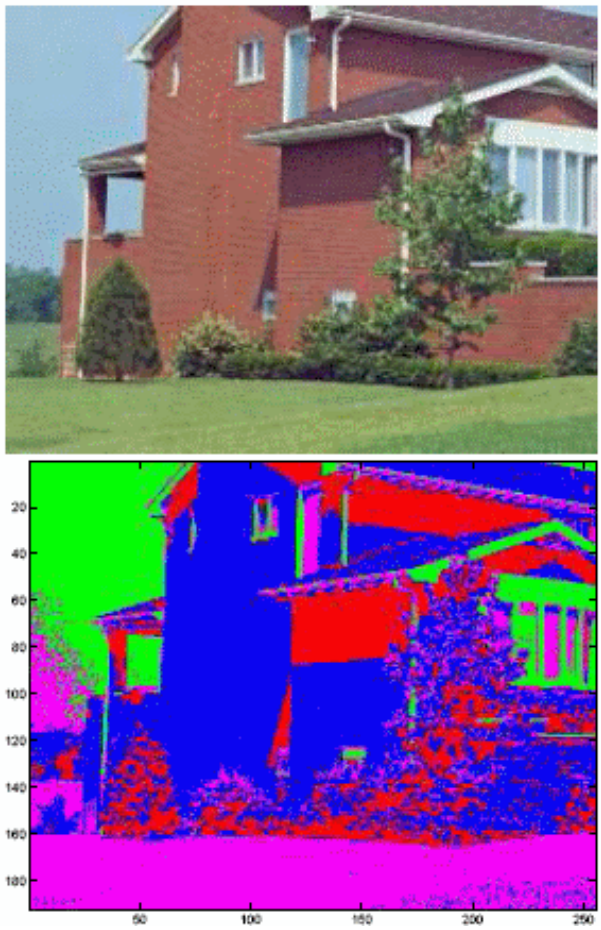

Fig. 3.5: Ejemplo de segmentación de una imagen usando *Mean-shift* [38]. La imagen a segmentar se muestra arriba. Abajo se muestra la imagen segmentada, note que regiones "similares" han sido pintadas de un solo color para hacer visible la segmentación.

3.3.3. CAMSHIFT

CAMSHIFT [8] es un algoritmo de seguimiento de objetos simple y eficiente en cuestión de recursos computacionales, usa *Mean-shift* internamente. A pesar de su simplicidad se puede ver que da seguimiento en dos dimensiones de manera precisa, semejante a otros sistemas de seguimiento [8] más sofisticados. CAMSHIFT también da seguimiento de manera parcial en ambientes ruidosos.

El algoritmo CAMSHIFT requiere compensar el ruido y los distractores (aquellos objetos lejanos de la región de interés) en la imagen. CAMSHIFT usa *Mean-shift* como una técnica no paramétrica para subir por gradientes de densidad y encontrar cimas de colinas. Aunque Mean-shift no fue ideado originalmente como algoritmo de seguimiento, es muy efectivo.

Ya que el algoritmo *Mean-shift* opera sobre distribuciones de probabilidad, se requiere transformar las imágenes que recibe a una representación de distribución de probabilidad. Para ello se usan histogramas de color y convoluciones de los histogramas sobre la imagen, con lo que se obtiene un mapa de probabilidad. Es sobre estos mapas de probabilidad donde opera aplicando iterativamente el algoritmo *Mean-shift*, se logra obtener un escalado de colinas sobre los valores del mapa de probabilidad.

Adicionalmente el algoritmo Mean-shift fue modificado para adaptarse dinámicamente a las condiciones cambiantes de los mapas de probabilidad pues se trata de seguimiento de objetos en movimiento sobre video.

Se puede ver un diagrama del funcionamiento de CAMSHIFT en la Fig. 3.6 que ha sido tomado de [8] donde se expone por primera vez CAMSHIFT. Si se inicializa la ventana de CAMSHIFT a todo el cuadro, CAMSHIFT converge hacia el centro de masa, usualmente es el objeto (*blob*) más grande. Puede ver un ejemplo de la convergencia de CAMSHIFT en la Fig. 3.7.

El procedimiento que realiza CAMSHIFT para el seguimiento es aplicar de manera iterativa el algoritmo *Mean-shift* que calcula centros de masa y

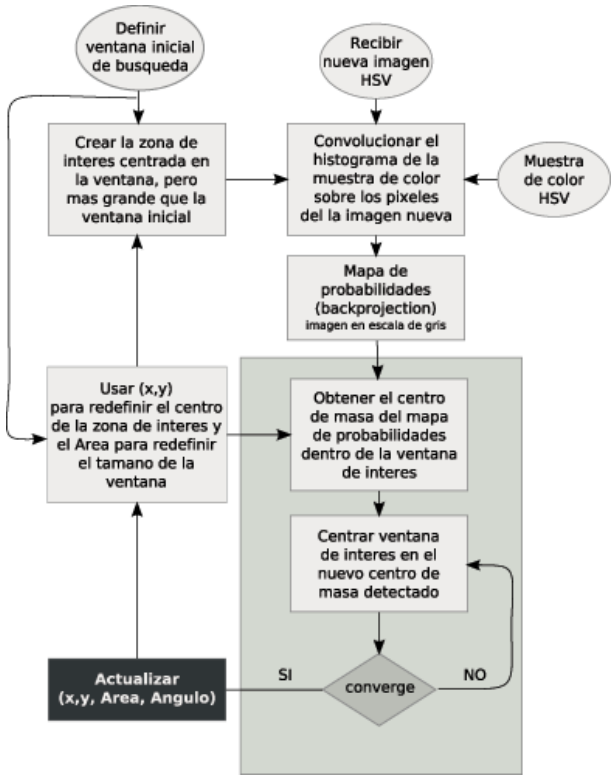

Fig. 3.6: Funcionamiento de CAMSHIFT. Se define una zona inicial de búsqueda la cual será reducida y movida de lugar mediante varias iteraciones. Se usa una convolución de un histograma de color para transformar la imagen en un mapa de probabilidades que se representa como una imagen en escala de grises, de manera iterativa se encuentra el pixel con mayor valor usando *Mean-shift*. Luego de varias iteraciones el algoritmo converge, aunque también se le puede programar un número de iteraciones máximo. CAMSHIFT también puede lograr una estimación del ángulo basado en los gradientes calculados a los valores de pixel del mapa de probabilidades.

ajustando la ventana de búsqueda. CAMSHIFT itera hasta que el centro de masa converge, o hasta que la diferencia entre el anterior y el nuevo centro de masa calculado es menor a cierto umbral. También se tiene la posibilidad de limitar a CAMSHIFT a un número máximo de iteraciones y con esto se limita el tiempo de convergencia.

CAMSHIFT también provee del área del objeto que está siguiendo, y una aproximación del ángulo del objeto. En este trabajo se usa el área del objeto para posteriormente calcular la profundidad y con esto obtener la coordenada z. El otro dato, el ángulo, no es de utilidad en este trabajo ya que se utiliza una esfera y como se dijo la proyección de una esfera sobre un plano es siempre un circulo, entonces el ángulo podría ser confundido fácilmente, además el ángulo no provee de ninguna pista adicional para el cálculo de la profundidad.

El orden de complejidad de CAMSHIFT es $O(\alpha N^2)$ donde α es una constante y la imagen se supone cuadrada de $N \times N$. α está influenciada por parámetros internos de CAMSHIFT como el cálculo de momentos.

3.3.4. Filtro de Ocupación Bayesiano BOF

Representación Mediante Rejillas

En los sistemas de navegación robóticos se ha representado el ambiente en forma de rejillas [11] [36] [39]. Esta representación se ha usado de manera extensa para hacer mapas de lugares cerrados y se ha hecho usando rejillas de 2 dimensiones. La finalidad es calcular la probabilidad de que cada celda esté "llena" o "vacía" usando los datos provenientes de un sensor. Para evitar la explosión combinatoria se asume que las celdas son independientes una de la otra y por lo tanto se tratan como variables aleatorias independientes.

Recientemente, las rejillas de ocupación (*occupancy grids*) se han adaptado para hacer seguimiento multi objetivo [39] para dar seguimiento a varios

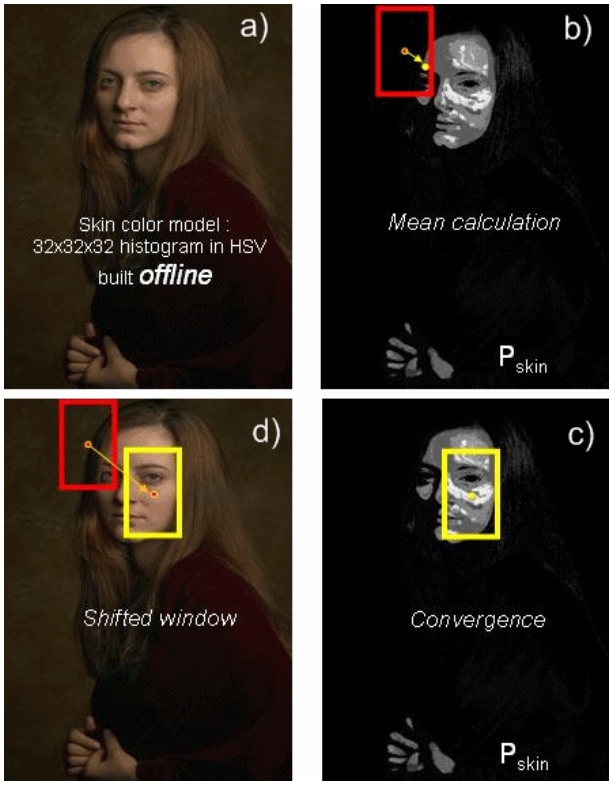

Fig. 3.7: Proceso del algoritmo CAMSHIFT. En sentido horario, a) Imagen original, se obtiene un histograma de color en base a los colores de la piel. b) Se aplica el histograma a la imagen mediante convolución y se obtiene una retroproyección, que es un mapa de probabilidades donde se usa *Mean-shift*. Se puede ver la ventana con que se inicializa CAMSHIFT. c) Ventana luego de converger al realizar varias iteraciones. d) La ventana inicial y la ventana final sobrepuestas en la imagen original.

objetos en movimiento. En estos enfoques se se aplican algoritmos de agrupación espacio-temporal a los mapas espacio-temporales y con ellos se logra la detección y el seguimiento.

En la Fig. 3.8 se ejemplifica una rejilla similar a la que se usa en los Filtros de Ocupacion Bayesianos [13]. Luego de tener una rejilla se pueden realizar algoritmos de agrupamiento sobre las celdas con mayores valores de ocupación. Vea la Fig. 3.9 para un ejemplo.

Filtro de Ocupación Bayesiano

En el Filtro de Ocupación Bayesiano [13] (o *BOF*, de su nombre en inglés *Bayesian Occupancy Filter*) es un enfoque que toma en cuenta el historial de observaciones del sensor para lograr estimaciones más robustas en ambientes que cambian constantemente. Al usar el historial *BOF* puede superar las oclusiones y otros problemas de detección en el sensor.

Los filtros de Bayes resuelven el problema de estimar de la secuencia de estados $x^k, k \in N$ de un sistema dado por:

$$x^k = f^k(x^{k-1}, u^{k-1}, w^k) \tag{3.15}$$

donde f^k es una función de transición (posiblemente no lineal), u^k es una variable de control (por ejemplo la velocidad o la aceleración) para el sensor, lo cual nos permite estimar su movimiento entre el tiempo $k-1$ y el tiempo k; y w^k es el ruido del proceso. Esta ecuación describe un proceso de Markov de orden uno.

Sea z^k la observación del sensor del sistema en un tiempo k. El objetivo de filtrar es estimar recursivamente x^k a partir de las mediciones del sensor:

$$z^k = h^k(x^k, v^k) \tag{3.16}$$

h^k es una función (posiblemente no lineal) y v^k es el ruido de las mediciones.

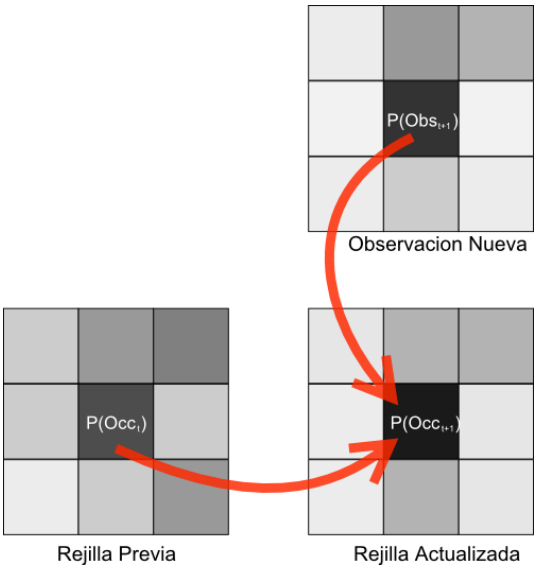

Fig. 3.8: Funcionamiento de una rejilla de un Filtro de Ocupación Bayesiano: versión simplificada de una rejilla de 3×3. En esta figura los valores de probabilidad de cada celda han sido representados por niveles de gris, más negro es más alto. **Arriba**: Cada observación se representa como una rejilla nueva con probabilidades asociadas para cada celda. **Izquierda**: La rejilla previa también contiene representación de ocupación. **Abajo, Derecha**: La rejilla previa y la nueva observación se usan para calcular la nueva rejilla. La nueva rejilla es calculada celda a celda actualizando el valor $P(Occ_{t+1}|P(Occ_t)P(Obs_{t+1}))$. Las celdas son consideradas independientes una de la otra para evitar una explosión combinatoria. Note en la figura que la celda central ha sido actualizada y en la rejilla actualizada tiene un valor más alto (más oscuro). Note también que la celda inferior izquierda fue actualizada con un valor más bajo (más blanco), debido a que la observación indica una menor probabilidad de ocupación.

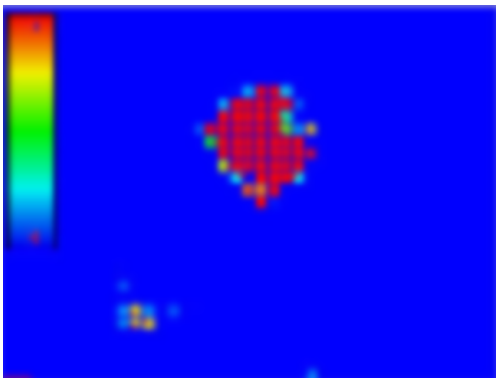

Fig. 3.9: Rejilla de BOF con representación de colores térmicos, la escala está indicada en el lado izquierdo. Note un cúmulo integrado por muchas celdas con áltos índices de ocupación, compuesto de celdas rojas; note un pequeño cúmulo en la parte inferior el cual tiene poco índice de ocupación, compuesto por unas pocas celdas amarillas. La certidumbre del cúmulo se calcula usando los índices de ocupación de las celdas que lo componen. Cabe mencionar que en esta etapa BOF no reporta ninguna información de objetos, sólamente reporta información acerca de la rejilla, los cúmulos sobre la rejilla y la velocidad con la que las celdas de un cúmulo se mueven. Posteriormente se extraen objetos a partir de los cúmulos.

Esta función modela la incertidumbre en la medición de z^k de un estado x^k de un sistema.

En otras palabras, la finalidad del filtrado es estimar recursivamente la distribución de probabilidad $P(X^k|Z^k)$, lo cual es conocido como la distribución "*a posteriori*". En general, esta estimación se hace en dos etapas: *predicción* y *estimación*:

- PREDICCIÓN. La finalidad de esta etapa es calcular un estimado *a priori* del estado del objeto que se está siguiendo. Este estimado es conocido como la distribución *a priori*.
- ESTIMACIÓN. La finalidad de esta etapa es calcular la distribución posterior del estado del objeto seguido. Para ello se usa el estimado *a priori* y la medición del sensor.

Existen soluciones exactas para la propagación recursiva de la densidad posterior, pero sólo son para casos especiales y restrictivos. En particular el filtro de Kalman es una solución óptima cuando las funciones f^k y h^k son lineales y los ruidos w^k y v^k son Gaussianos. Pero en general, las soluciones no pueden ser determinadas analíticamente de manera eficiente y por lo regular se tiene que recurrir a soluciones aproximadas obtenidas de manera numérica.

En este caso, el estado del sistema está dado por el estado de ocupación de cada celda de la rejilla y las condiciones requeridas para ser capaz de aplicar las soluciones exactas (tales como Kalman) no siempre se satisfacen. En el filtro *BOF* el proceso de predicción y estimación se hace de manera independiente en cada celda de la rejilla.

Extracción de objetos a partir del BOF

El Filtro de Ocupacion Bayesiano (BOF) da resultados satisfactorios pero para lograr abstracción a nivel de objeto para análisis posteriores se debe

hacer una extracción de objetos y un seguimiento. La extracción de objetos se hace después de la creación de la rejilla, usándola como base. El proceso se logra en varias etapas:

1. **Extracción de objetos**. El primer paso es un proceso de segmentación que localiza regiones individuales en el ambiente. BOF provee de una estimación coherente de índices de ocupación y de velocidad, así que una simple segmentación basada en discontinuidades de cúmulos es suficiente para lograr una segmentación. La similaridad entre dos cúmulos cercanos se calcula como la inversa de distancia euclideana entre ellos en el espacio definido por la posición, índice de occupación y velocidad. Cúmulos con alta similaridad son agrupados para formar un único cúmulo.

2. **Filtrar con Kalman** A nivel de objetos se usa un filtro de Kalman para manejar el movimiento de cada región ocupada. El movimiento se modela para tener una velocidad constante. La velocidad es observable, gracias a la información que BOF provee.

Luego de la extracción de objetos BOF es capaz de reportar posicion 2D de cada uno de los objetos que se les da seguimiento. También BOF puede reportar las velocidades de los objetos y la varianza en el tamaño de ellos que más adelante veremos como se usa para detección de profundidad.

3.4. Conclusiones

Se han dado a lo largo de este capítulo diversos algoritmos para segmentación, y seguimiento. Se explicó el modelo de color HSV y la razón por la que se usó en esta tesis. También se dieron algunos detalles de algoritmos importantes de segmentación y seguimiento como *Mean-shift* y CAMSHIFT. Luego se explicó a grandes rasgos el funcionamiento de una Rejilla de Ocupación Bayesiana. Es importante mencionar que ninguno de los algoritmos presentados tiene detección de profundidad. En el siguiente capítulo se explica la manera en que se utilizaron y la manera en que se logró la estimación de profundidad utilizando algunos datos obtenidos a través de un Filtro de Ocupación Bayesiano.

4. SEGUIMIENTO Y ESTIMACIÓN DE PROFUNDIDAD

En este capítulo se explica a detalle todo el proceso de seguimiento, el cual está conformado por las siguientes etapas:

- Se realiza una segmentación del objeto a seguir usando el color mediante el uso de retroproyección de histogramas.
- Se inserta la retroproyección en un Filtro de Ocupación Bayesiano (BOF). El BOF representa la escena como una rejilla donde cada celda tiene un índice de ocupación asociado.
- Usando información de los niveles de ocupación del BOF se hace obtiene la posición 2D del cúmulo de celdas contiguas con índices de ocupación más altos.
- Usando información del tamaño y varianza de los cúmulos se estima la profundidad.

Enseguida se explica como funciona el algoritmo con el que se hace el seguimiento y la detección de profundidad. Finalmente se da una justificación formal de la manera en que se detecta la profundidad y la razón por la cual podemos ignorar la calibración de la cámara.

Se explicará a continuación la manera en que se logra la segmentación inicial de las imágenes recibidas de la cámara. Después se explica el algoritmo de seguimiento y al final se da una explicación de la manera en que detecta la profundidad.

4.1. Segmentación

En este trabajo se usa una esfera de color, y se usa un método similar al usado en [7] para la detección de esferas. Ya que se eligió usar una esfera completamente azul y sin marcas, el problema en principio es detectar objetos de color, en particular una esfera.

Para resolver el problema de detección de objetos de color, convertimos los cuadros recibidos de la cámara del espacio de color *Red, Green, Blue* (RGB) al espacio *Hue, Saturation, Value* (HSV) [20].

Con antelación al rastreo se hace una selección adecuada de una muestra del color a seguir. A partir de la muestra se obtiene un histograma de color exclusivo del canal *Hue*. Se hace una convolución de este histograma para obtener la retroproyección, para cada cuadro. La retroproyección puede verse como una imagen en escala de grises del mismo tamaño que el cuadro donde fue extraída, siendo oscuros los valores bajos y blancos los valores altos. La imagen resultante de la reotroproyección luego se filtra de ruido usando técnicas simples de erosión en las cuales se eliminan cúmulos (*clusters*) de pixeles tamaño pequeño (en una imagen de 320×340 consideramos a un cúmulo como pequeño si es de menos de 80 pixeles). En este momento se tiene una detección de color adecuada y resistente a cambios ligeros de iluminación. Sin embargo la detección rara vez es perfecta y hay algunas zonas adicionales de la imagen que también son detectadas y que debemos eliminar o de alguna manera ignorar para tener un seguimiento más robusto.

La proyección de una esfera en el plano del sensor de la cámara, es decir en un plano 2D, es siempre un círculo, lo cual nos evita problemas de rotación en cualquier eje, y al final nos facilita en un proceso posterior la tarea de encontrar el radio del la esfera.

La convolución se aplica a todos las cuadros (*frames*) recibidos de la cámara. Por lo cual se tiene un flujo de mapas de probabilidad, uno por cada

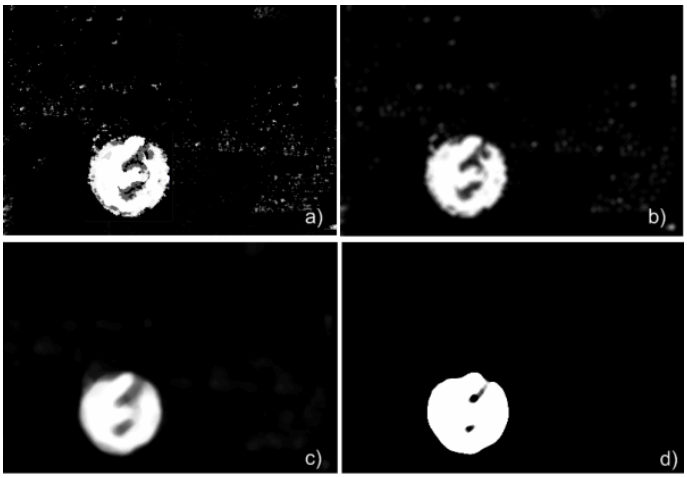

Fig. 4.1: Técnicas para eliminación de ruido. *a)* Imagen original, la mancha redonda imperfecta representa un objeto redondo. Note el ruido. *b)* Procesamiento de la imagen original mediante un algoritmo de eliminación de ruido. Note que no se pudo eliminar el ruido por completo. *c)* Procesamiento mediante algoritmo que calcula las medianas de los valores. Note que un poco más de ruido fue eliminado. *d)* Procesamiento final mediante umbralización. El ruido desapareció por completo y se tiene una única mancha. Los huecos en la mancha pueden ser eliminados mediante algoritmos para crecimiento de región.

cuadro. Estos mapas representan las probabilidades de encontrar objetos del color representado por el histograma que se usó para la convolución (los procedimientos de convolución fueron explicados en un capítulo anterior).

Antes de hacer procesamiento posterior se aplica un filtro de paso bajo con el cual se reduce considerablemente el ruido. Vea la Fig. 4.1 para un ejemplo de la técnica de reducción de ruido.

El segundo problema de la detección de la esfera es el seguimiento del movimiento. Una vez obtenidas las aproximaciones de la posición se debe mantener un rastreo sobre las coordenadas 2D de la posición de la esfera y calcular una estimación de la profundidad.

A continuación se explica el algoritmo usado para el seguimiento del objeto y mas adelante la manera en que se detecta la profundidad junto con su justificación.

4.2. Algoritmo de Seguimiento

El rastreo en 2D está basado en un filtro Bayesiano. Inicialmente, el objeto a seguir se captura (en este caso una esfera) y se obtiene un histograma de color de la imagen capturada. Luego se aplica el siguiente algoritmo a cada cuadro proveniente de la cámara:

1. Calcular el histograma del cuadro recibido usando el histograma previamente capturado del objeto. La retroproyección del histograma representa un mapa de probabilidades. El mapa de probabilidades está representado como una imagen en escala de grises.

2. Procesar la retroproyección tomada como imagen en escala de grises con un filtro de paso bajo para eliminar ruido.

3. Alimentar la retroproyección a un Filtro de Ocupación Bayesiano (BOF)

a) La imagen en escala de grises se usa como entrada (observaciones) al BOF, el cual divide la imagen en un conjunto de celdas (BOF *grid*). El BOF asigna una distribución de probabilidad a cada celda la cual representa un índice de ocupación. Las celdas con consideradas independientes para evitar una explosión combinatórica.

b) Actualizar el índice de ocupación de cada celda basándose en la retroproyección y adicionalmente el índice de ocupación previo.

c) Usar un algoritmo de agrupamiento sobre las celdas para rastrear cúmulos de celdas con altos índices de ocupación (i.e. con distribuciones de probabilidad de valores altos). Estas celdas se consideran ocupadas.

d) Reportar la posición (x, y) del centroide del cúmulo junto con el nivel de certidumbre y la varianza del tamaño del objeto.

4. Si el nivel de certidumbre del cúmulo (*cluster*) está sobre un cierto umbral, entonces marcar el cúmulo como un objeto detectado. Almacenar la posición (x, y).

5. Usar el tamaño del cúmulo y la varianza del mismo para estimar el tamaño del objeto.

   Reescalar el valor de acuerdo a una medición del objeto real y de la rejilla de celdas, la cual es de un tamaño considerablemente menor que la resolución de la imágen. De esta manera se logra un estimado de la profundidad, z, del objeto.

6. Reportar la existencia de un objeto y su posición en el espacio, (x, y, z).

En la siguiente sección se da una justificación formal de la detección de la profundidad y se muestra cómo se puede evitar la calibración de la cámara.

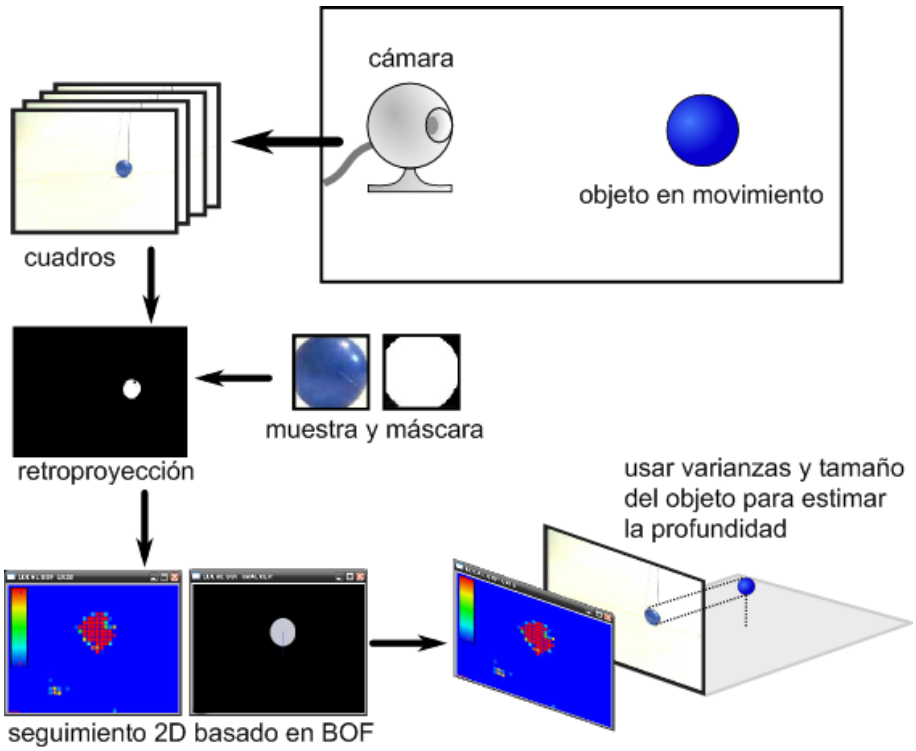

Fig. 4.2: Diagrama de bloques del sistema de seguimiento 3D monocular. Las imágenes recibidas por la cámara son proyecciones bidimensionales sobre el plano del sensor. Entonces, usando retroproyección de color y un Filtro de Ocupación Bayesiano (BOF) se obtiene seguimiento 2D. La rejillas de ocupación del BOF (BOF *grid*) se usan para obtener una aproximación del tamaño del objeto en unidades de celdas de la rejilla. Conociendo el tamaño real del objeto y la discretización hecha por la rejilla BOF es posible realizar la estimación de profundidad.

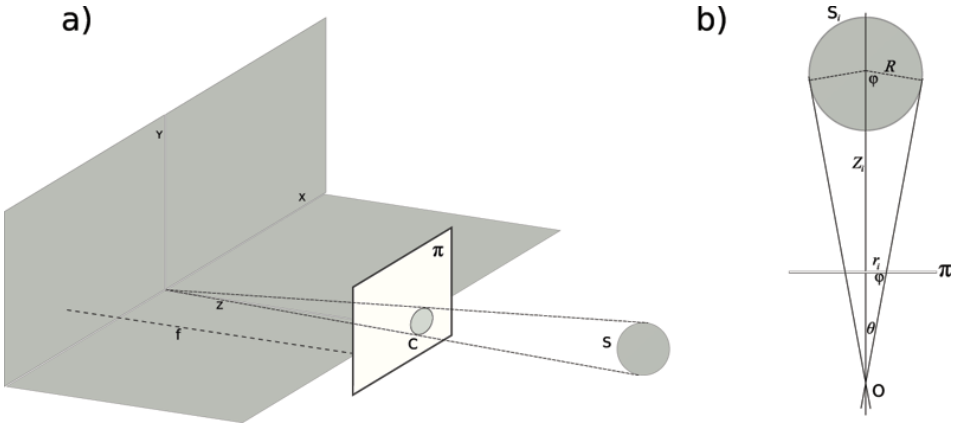

Fig. 4.3: **a)**Proyección de una esfera S sobre un plano π, el plano π representa el plano del sensor de la cámara. **b)** Vista simplificada de la esfera y el plano π.

4.3. Detección de Profundidad

Considere la Fig. 4.3 en b), la cual muestra una sección de a). El origen de las coordenadas está en O, donde el centro de la esfera intersecta al plano π y proyecta un círculo de radio r_i. Consideremos el caso de una cámara calibrada de tal manera que podemos conocer los parámetros intrínsecos, en particular, la longitud focal f, la relación de aspecto s_x/s_y y el centro óptico (u_0, v_0) se asumen conocidos. Despreciaremos la distorsión del sistema óptico.

La proyección de la esfera en el plano retinal π se puede ver en la Fig. 4.3$-b$. La esfera tiene un radio conocido R. Denotemos al centro de la esfera como:

$$P_i = (X_i, Y_i, Z_i) \tag{4.1}$$

Para cada posición de la esfera S, la proyección de S sobre el plano retinal π produce un círculo al que denotamos como C centrado en (x_i, y_i) con radio r_i. Haremos la convención de usar letras mayúsculas para las coordenadas del

sistema referencial de la cámara, y letras minúsculas para las coordenadas del plano π. Los símbolos (u_i, v_i) denotaran coordenadas en pixels sobre π.

Entonces a partir del modelo de proyección:

$$x_i = \frac{fX_i}{Z_i} = -s_x u_i - u_0 \tag{4.2}$$

$$y_i = \frac{fY_i}{Z_i} = -s_y v_i - v_0 \tag{4.3}$$

Luego, de las relaciones:

$$sen(\theta) = \frac{R}{Z_i} \tag{4.4}$$

$$tan(\theta) = \frac{r_i}{f} \tag{4.5}$$

Derivamos que:

$$cos(\theta) = \frac{fR}{Z_i r_i} \tag{4.6}$$

Hacemos una sustitución de 4.4 y 4.6 en $sen^2\theta + cos^2\theta = 1$ y tomando la solución positiva resulta:

$$Z = R\sqrt{1 + \frac{f^2}{r_i^2}} \tag{4.7}$$

Asumiremos que las esferas están lo suficientemente lejanas de el plano π, lo cual en la práctica es lo que se presenta. En este caso, la proyección sobre π corresponde a un círculo, denotemos su centro como P_i. Bajo estas *suposiciones débiles* de perspectiva el centro de la esfera S puede ser aproximado a partir del centro y radio de su círculo proyectado C.

Sea $P_j = (X_j, Y_j, Z_j)$ un punto sobre la superficie de S cuyo punto proyectado es (x_j, y_j) en la circunferencia C. Puesto que (x_i, y_i) es el centro de C, su radio puede ser expresado de la siguiente manera:

$$r^2 = (x_j - x_i)^2 + (y_j - y_i)^2 \tag{4.8}$$

Y de manera análoga, el radio R de S se puede expresar como

$$R^2 = (X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2 \tag{4.9}$$

Aplicando las ecuaciones (4.2) y (4.3) para obtener X_i, X_j, Y_i, Y_j y sustituyéndolas en (4.9) se obtiene que

$$R^2 = \left(\frac{Z_j x_j - Z_i x_i}{f}\right)^2 + \left(\frac{Z_j y_j - Z_i y_i}{f}\right)^2 + (Z_j - Z_i)^2 \tag{4.10}$$

Note que $Z_i > 0$, así que se toma la solución positiva,

$$Z_i = fR/r_i \tag{4.11}$$

De esta última relación, es interesante notar que para dos posiciones distintas de la esfera S con respecto al eje Z, por ejemplo Z_1 y Z_2. La relación de profundidades relativas.

$$\frac{Z_1}{Z_2} = \frac{fR/r_1}{fR/r_2} = \frac{r_2}{r_1} \tag{4.12}$$

Es decir, que se pueden detectar profundidades relativas sin requerir de los parámetros intrínsecos de la cámara. Esto es un resultado interesante que se puede intuir fácilmente, pero es mejor demostrarlo de manera formal. En la implementación se usa la conclusión de que no se requieren los parámetros intrínsecos y que las profundidades relativas se pueden obtener de manera directa, y se obtienen medidas relativas de profundidad. Las medidas relativas de profundidad aunadas a una simple calibración de los puntos *adelante* y atras hace que el sistema sea ajustable a diversas profundidades. Lo anterior trae como beneficio adicional que el sistema se pueda adaptar a diferentes pacientes con diferentes rangos de movilidad, ya que éstos pueden hacer uso del sistema de acuerdo a sus capacidades y niveles de rehabilitación simplemente recalibrando.

4.4. Conclusiones

En este capítulo se ha descrito la manera en que el sistema desarrollado logra hacer la segmentación, el seguimiento en x y y y la estimación de profundidad z basada en datos obtenidos a partir del Filtro de Ocupación Bayesiano. También se dió una justificación de la razón por la cual es posible ignorar la calibración de la cámara.

En el siguiente capítulo se presentan datos experimentales y la comparación del sistema monocular con el sistema estéreo. También se presentan algunos experimentos de la integración con el *software* de rehabiltiación *Gesture Therapy*.

5. EXPERIMENTOS

En este capítulo se muestran varios resultados experimentales del sistema monocular. Primero se muestra la comparación del sistema con un sistema estéreo y los datos que fueron colectados, esto con el fin de tener una comparativa de su precisión. Además se presentan resultados de las pruebas de robustez del sistema de seguimiento monocular. Al final se muestran resultados del funcionamiento del sistema monocular integrado al *software* de terapias *Gesture Therapy*.

5.1. Precisión en seguimiento 3D

Para evaluar la precision del seguimiento monocular 3D lo comparamos con un sistema estéreo. Las pruebas fueron desarrolladas en un ambiente de interiores con luz artificial; se hizo una grabación para cada una de las dos cámaras manteniendo el video sincronizado. Esta grabación se hizo con cámaras calibradas debido a los requerimientos del sistema estéreo. Sin embargo, para todas las pruebas posteriores con el sistema monocular esta calibración fue desechada.

Luego de la grabación, uno de los videos se usó para experimentar con el sistema monocular y compararlo con el sistema estéreo. El sistema de coordenadas usadas fue tal que el origen estaba en el centro de una de las dos cámaras y esa fue precisamente la que se usó para el sistema monocular. Este arreglo nos permitió simplificar el sistema ya que no se requería de un reajuste de coordenadas y el sistema podría funcionar de la manera en que

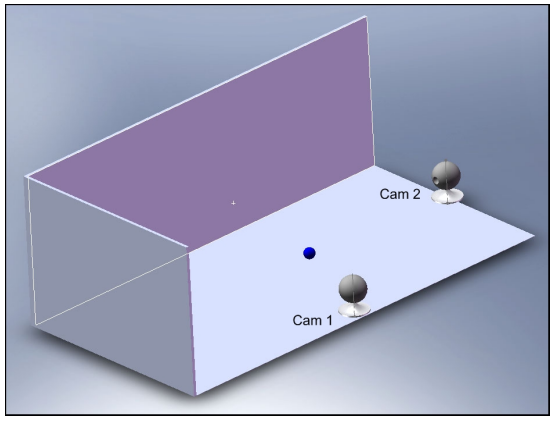

Fig. 5.1: Arreglo experimental para la comparación con el sistema estéreo. El video grabado por la cámara indicada como "Cam 1" fue usado como datos de entrada para el sistema monocular. La cámara marcada como "Cam 2" se usa como cámara adicional para el sistema estéreo. El punto azul en el centro de la figura es el blanco al que le da seguimiento, el blanco fue movido en el área dentro del rango visual de ambas cámaras.

fue diseñado. Se puede ver un diagrama del arreglo experimental en la Fig. 5.1.

Se usó el sistema estéreo como sistema de referencia para medir la precisión del sistema monocular. Varias trayectorias fueron realizadas y grabadas usando una esfera de color azul. Tanto el sistema estéreo como el monocular usaron el método de retroproyección de histogramas de color para hacer la segmentación de la imagen y la detección de la esfera azul.

Al finalizar la grabación de los videos, el video grabado con la cámara marcada como "Cam 1" en la Fig. 5.1 fue usado como datos de entrada para el sistema monocular. El sistema monocular entonces procesó el video y generó como salida las coordenadas 3D estimadas de la esfera.

La Fig. 5.2 muestra los resultados de una secuencia de video de 700 cuadros donde se compara el sistema de seguimiento estéreo con el sistema

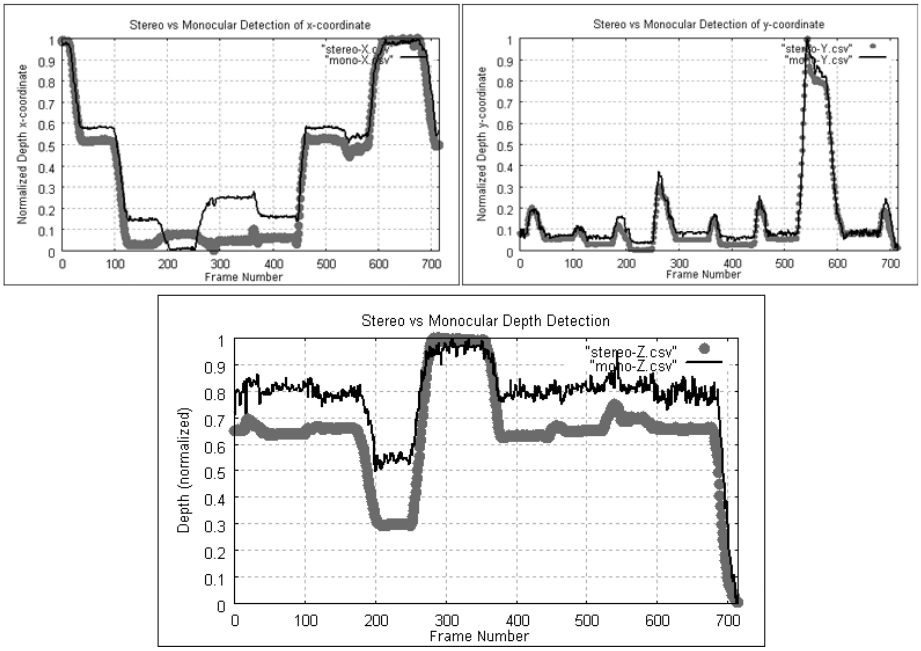

Fig. 5.2: Comparación de la detección 3D del sistema estéreo y el monocular. **Arriba**: Detección de las coordenadas x y y de ambos sistemas. Para la coordenada x existe una ligera diferencia en los cuadros 100–450 debido a que el objeto estaba demasiado cerca del límite del campo de visión de la cámara. **Abajo**: Comparación de la estimación de profundidad del sistema estéreo con el sistema monocular. Note que los cambios de profundidad son detectados de manera precisa y en el momento adecuado. Existe una ligera diferencia debido al escalado distinto entre ambos sistemas. Los datos para x y y en el sistema monocular fueron capturados mientras se aplicaba un filtro de Kalman. Para z no hubo filtrado, se muestran los datos brutos. Vea la Fig. 5.3 para la comparación de la detección de profundidad con los datos filtrados.

de seguimiento monocular 3D, se muestran gráficas de x, y y z. Los 700 cuadros corresponden a aproximadamente 23.3 segundos de video grabados a 30 cuadros por segundo. La computadora usada para los experimentos es una PC con procesador Intel Pentium 4 Prescott a 2.9Ghz de un solo núcleo, con 512 megabytes de RAM. Todos los videos fueron grabados a una resolución de 320x240 pixeles. Es importante mencionar que el video no fue procesado en el momento de la grabación sino de manera posterior debido a la necesidad de rastrear conjuntamente con el sistema estéreo y el monocular, pues en caso que se desee hacer rastreo de video en vivo, ambos sistemas requieren acceso exclusivo a las cámaras, cosa que no era posible.

En la secuencia de 700 cuadros, se movió un objeto en una escena, los datos de la escena fueron normalizados para convertirlos de dimensiones físicas a valores en el rango $[0 \ldots 1]$ de tal forma que en la gráfica se muestran datos adimensionales, lo mismo para la Fig 5.3.

Luego de hacer el procesamiento se obtuvo que el sistema de seguimiento monocular logró procesar el video en 20.1 segundos, lo que equivale a aproximadamente 34.8 cuadros por segundo. Lo anterior permite que el sistema de seguimiento monocular pueda funcionar en tiempo real a 30 cuadros por segundo teniendo en cuenta que la captura del video es un proceso que consume pocos recursos y que puede despreciarse la carga adicional que pueda tener. El sistema estéreo con el que se hizo la comparacón tenia una velocidad reportada de 15 cuadros por segundo.

Podemos apreciar en la Fig. 5.2 una similitud en las trayectorias estimadas por ambos sistemas de seguimiento. En la estimación de profundidad existe una diferencia debido al escalamiento del sistema monocular distinto al del sistema estéreo, debido a la manera relativa en que el sistema monocular calcula las profundidades. Ésta puede ser corregida con una simple calibración de profundidad del sistema monocular. Sin embargo para muchas aplicaciones no se requiere una estimación de la profundidad absoluta, tal es el caso del

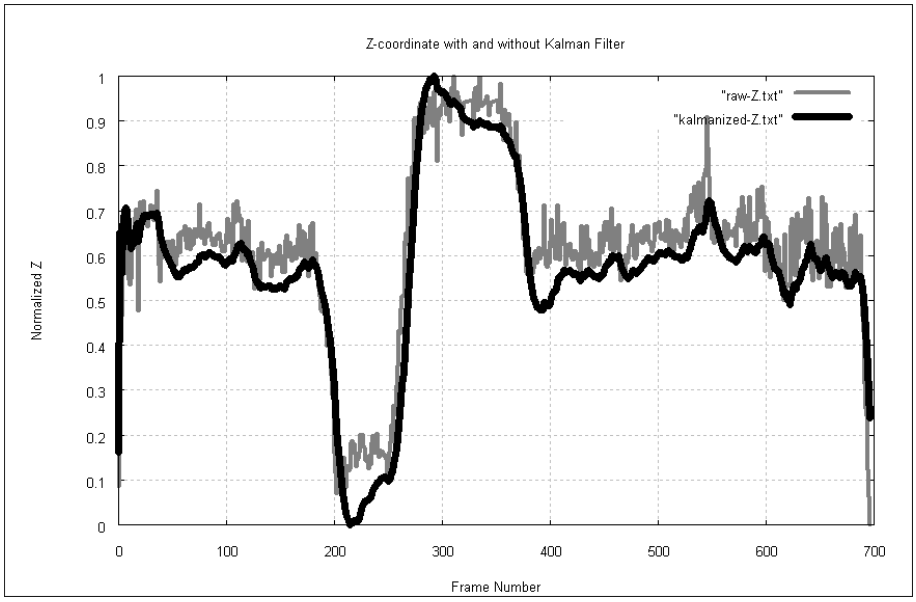

Fig. 5.3: Comparación de la detección de la profundidad con los datos en bruto sin filtrar (raw-z.txt en gris) y con los datos filtrados mediante un filtro de Kalman (kalmanized-z.txt en negro). Se puede observar una reducción significativa del ruido en la señal filtrada. No se puede dar una conclusión adicional acerca de las ligeras vibraciones restantes pues no se tuvo instrumental para medir el temblor natural de la mano.

software para rehabilitación *Gesture Therapy* (Terapia por Gestos).

Luego de obtener un estimado de la profundidad, z, ésta se procesa con un filtro de Kalman para reducir el ruido. En la Fig. 5.3 se muestra la diferencia en la detección de la profundidad de los datos en bruto obtenidos del sistema monocular y de los obtenidos mientras se aplica un filtro de Kalman.

5.1.1. Robustez en el seguimiento

El seguimiento está basado en el Filtro de Ocupación Bayesiano para seguimiento 2D, entonces hereda su robustez. Para validar la robustez se probaron diversos aspectos: (i) oclusiones temporales, (ii) movimientos *rápi-*

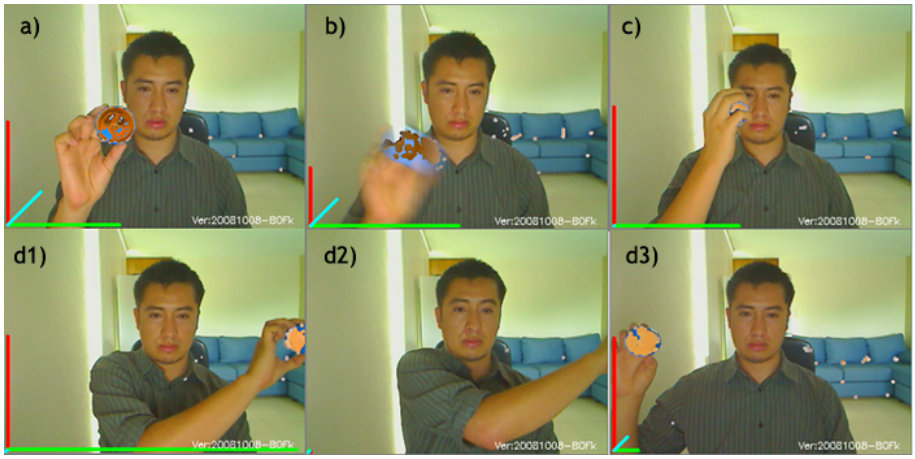

Fig. 5.4: Robustez del seguimiento monocular 3D. **a)** Condiciones de seguimiento normales. El color detectado aparece en video inverso. Note un poco de ruido debido al color azul del sofá que aparece atrás de la escena. Note también la detección imperfecta de la esfera azul. Adicionalmente note las 3 líneas en la esquina inferior izquierda, éstas líneas representan una pista visual de la detección de coordenadas (x, y, z). **b)** Detección difícil debido a movimientos rápidos de la esfera. Incluso cuando se hacen movimientos rápidos, el seguimiento puede lograrse. Estos movimientos pueden incluso ser de lado a lado del cuadro y el seguimiento siempre se recupera. Como se aprecia en la imagen, en movimientos rápidos y de corta longitud, la estimación 3D aun funciona. **c)** Detección difícil debido a oclusión. En oclusión parcial temporal el seguimiento 3D puede fallar momentáneamente o reportar una profundidad distinta, pero la detección se recupera tan pronto como la oclusión finalice. En la imagen mostrada, la detección 2D aun funciona. **d1) a d3)** Recuperándose de una salida de cuadro. En **d1** el objeto se mueve fuera el rango visual y eventualmente desaparece por el lado derecho del cuadro. Entonces en **d2**, el seguimiento está completamente perdido debido a que no hay objeto en la escena; note que el ruido es la única detección pero el sistema no confunde al ruido con objetos. Finalmente, en **d3**, el objeto aparece de nuevo en la escena entrando por el lado izquierdo del cuadro, y el sistema de seguimiento se recupera incluso cuando el lugar por el que entró no es el mismo que el lugar por donde salió.

dos, (iii) sacar el objeto del campo de visión. En general, si el seguimiento se pierde, se recupera de inmediato como se ilustra en los ejemplos de la Fig. 5.4 donde se observan algunas situaciones extremas donde el seguimiento puede ser muy difícil de lograr. También se observan los casos donde el objeto sufre de oclusiones y cuando el objeto sale del campo de visión, regresando por un lugar distinto al que salió.

Cuando el objeto se pierde por completo, como es el caso de oclusiones totales prolongadas y salidas de campo de vision, el seguimiento se recupera en general luego de aprox. 60 cuadros (equivalente a 2 segundos), a partir de los cuales continúa reportando las coordenadas 3D.

5.2. *Gesture Therapy*

Se integró el sistema de seguimiento monocular en el software de terapias por gestos *Gesture Therapy*, parte del sistema *T–WREX*. Se sustituyó un sistema anterior [27] que usaba seguimiento estéreo.

Gesture Therapy permite a los individuos que han sufrido de accidentes cerebrovasculares practicar movimientos de brazo, tanto en casa de manera autónoma, como en una clínica bajo supervisión mínima y con interacción periódica de un terapeuta. Hace uso de un ambiente virtual para facilitar el entrenamiento de movimientos repetitivos, estos movimientos simulan actividades de la vida cotidiana [31, 34].

5.2.1. *Integración con Gesture Therapy*

Gesture Therapy usa un sistema modular para la implementación de mecanismos de entrada o dispositivos de interfaz humana. De esta manera para integrar un nuevo dispositivo de interfaz humana lo único que hay que hacer es crear un módulo nuevo o biblioteca que será llamado cuando *Gesture Therapy* sea ejecutado. *Gesture Therapy* se ejecuta sobre *Windows*, por lo

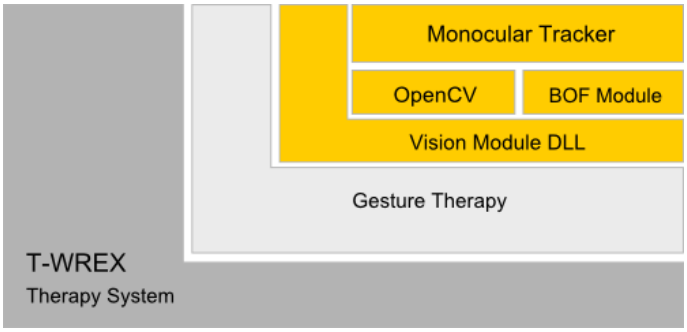

Fig. 5.5: Diagrama de integración del sistema de rastreo monocular con *Gesture Therapy*, el sistema monocular se integra a través de una biblioteca de vínculos dinámicos (DLL). T-WREX es un sistema padre que contiene como subprograma a *Gesture Therapy* y *Gesture Therapy* a su vez carga la DLL con la cual es posible dar seguimiento a los objetos y por ende usar los objetos como dispositivo de interfaz humana entre el usuario y *Gesture Therapy*. El módulo de visión utiliza biblitecas de OpenCV y de BOF.

que los módulos toman la forma de bibliotecas de vínculos dinámicos.

Se implementó una biblioteca de vínculos dinámicos (o DLL por sus siglas del inglés: *Dynamic Link Library*), la cual puede ser cargada por *Gesture Therapy*. De esta manera *Gesture Therapy* puede leer datos acerca del movimiento directamente del sistema de seguimiento monocular a través de la DLL. En la Fig. 5.5 se puede ver un diagrama que explica de manera gráfica la manera en que el sistema monocular está integrado en T-WREX y en *Gesture Therapy*.

Una pelota de color se usa como objeto a rastrear en la mano del usuario. Durante la terapia las coordenadas 3D son enviadas al simulador de tal manera que el paciente puede interactuar con el ambiente virtual. Durante la interacción el usuario mueve su brazo desarrollando diferentes tareas. Las tareas están orientadas a reproducir situaciones de la vida real, por lo cual

están orientadas a rehabilitación.

Se hicieron experimentos con el sistema *Gesture Therapy* y el seguimiento funciona muy bien, permitiendo al usuario realizar los juegos de una manera muy precisa. En el futuro cercano se comenzaran las pruebas clínicas con pacientes. En la Fig. 5.6 se muestran algunos ejemplos de los diferentes juegos de rehabilitación.

5.3. Conclusiones

En este capítulo se mostraron los resultados de los experimentos con el sistema monocular desarrollado. Se comparó el sistema con un sistema de visión estéreo y se mostró un desempeño similar. Se mostró también la integración con el *software* de terapias por gestos *Gesture Therapy*. Y se explicó la manera en que el sistema monocular fue integrado.

En el próximo capítulo se comentaran los logros obtenidos, explicando formas de mejorar el sistema.

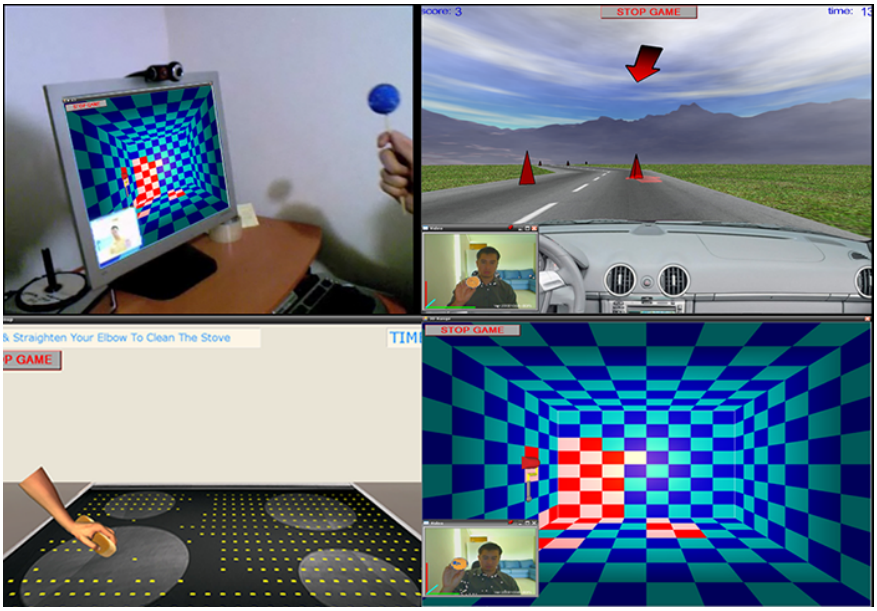

Fig. 5.6: Uso del sistema de seguimiento monocular con *Gesture Therapy*. **Arriba-Izquierda**: Computadora con el software de terapias *Gesture Therapy* y el sistema se seguimiento monocular 3D, la cámara aparece sobre el monitor y se muestra la esfera. El sistema de seguimiento monocular permite al usuario tener movimientos finos. **Arriba-Derecha**: El juego de carreras de autos permite al usuario ejercitar movimientos de brazo laterales para poder controlar el automóvil sin salirse de la pista. **Abajo-Izquierda**: El juego de limpieza de la estufa. Este juego es útil para ejercitar movimientos de estirar y encoger el brazo. El sistema monocular provee de una estimación de profundidad excelente. **Abajo-Derecha**: En el juego de pintar un cubo por dentro el sistema de seguimiento monocular desarrollado muestra un seguimiento (x, y) y estimación (z) de profundidad robustos, lo cual permite al usuario poder pintar cuadros específicos del cubo.

6. CONCLUSIONES Y TRABAJO FUTURO

6.1. Conclusiones

El desarrollo de ésta tesis se enfocó en resolver el problema de seguimiento monocular con estimación de profundidad. Este problema en específico no ha tenido muchos avances en los ultimos años, tal vez por la generalización del uso de sistemas estéreos con los cuales muchas aplicaciones pueden resolverse de manera satisfactoria. Sin embargo, el simplificar la detección para que pueda funcionar de manera robusta con una única cámara trae grandes beneficios y aumenta el espectro de aplicaciones potenciales, sacándolo del contexto de laboratorio y atrayéndolo más hacia el uso generalizado. Entre las aplicaciones potenciales de un sistema monocular están los usos en realidad virtual y realidad aumentada, interfaces humano-máquina (útiles para navegación en escritorio o juegos) o aplicaciones de seguimiento de extremidades para reabilitación médica como es el caso de este trabajo.

El aporte principal de esta tésis es el sistema de seguimiento monocular 3D. El sistema funciona mediante la recolección de imágenes de una cámara web que posteriormente son alimentadas a una Rejilla de Ocupación Bayesiana (BOF). Sobre la rejilla se realiza agrupamiento de celdas con altos índices de ocupacion y los cúmulos con áltos indices de certidumbre son tratados como objetos. La detección de profundidad se hace mediante el conocimiento del objeto, y de la rejilla BOF.

Se mostró que el sistema desarrollado es robusto tanto a oclusiones como salidas del campo de visión de la cámara e incluso puede rastrear en

situaciones difíciles como es el caso de objetos que se mueven muy rápidamente. El sistema desarrollado se recupera casi de inmediato (aprox. 2 seg.) de cualquier situación extrema que pueda hacer que el seguimiento se pierda.

Otro aporte importante fue la simplificación del sistema de terapias por gestos, *Gesture Therapy*. El sistema usaba un sistema estéreo que requería la calibración de ambas cámaras y el uso de sistemas mecánicos para fijarlas, además requería de la exacta medición de los puntos y ángulos en los cuales se colocaban las cámaras con respecto a un sistema coordenado fijo. En éste aspecto el sistema monocular funcionó de manera sobresaliente, mostrando en los experimentos un desempeño en el seguimiento similar al que el sistema estéreo puede proveer. En breve, el sistema monocular integrado al sistema de terapias y entrará en la fase de estudios clínicos en el Instituto Nacional de Neurología y Neurocirugía (INNN), parte del sector salud de México. Con las pruebas clínicas y la eventual implementación de este sistema para rehabilitación, se cumple una de las metas que cualquier científico debe aspirar, el obtener frutos de la investigación que tengan un impácto social en beneficio de la población.

6.2. Puntos débiles del sistema

Durante los experimentos, el sistema de seguimiento monocular funcionó de manera adecuada cuando se tenía una correcta muestra del objeto. En caso de no tener una muestra adecuada, la segmentación no se logra adecuadamente, y ya que es la base de todo el proceso de seguimiento, sin una segmentación buena el seguimiento no puede funcionar bien. Sin embargo, siempre puede tomarse una nueva muestra en caso de que la muestra anterior no haya sido lo suficientemente buena. Además una vez que se obtiene una muestra adecuada, ésta puede servir para toda la sesión. Existen dos casos donde se requiere tomar una nueva muestra: (i) cuando se utiliza un obje-

to distinto y, (ii) cuando las condiciones de iluminación cambian de manera notable y alteran el color del objeto tal y como lo percibe el sensor de la cámara.

Para detectar la profundidad, el sistema usa el tamaño de la rejilla de BOF y otros datos del cúmulo de celdas que representan el objeto. En oclusiones parciales, el tamaño del cúmulo es detectado de manera parcial, por lo que la detección de profundidad podría ser estimada de manera errónea. Éste problema se soluciona tan pronto la oclusión termina. Para el *software* de rehabilitación el manejo de oclusiones parciales no es una preocupación ya que por la manera en que se realizan los ejercicios no se espera tener oclusiones.

6.3. Trabajo futuro

Durante el desarrollo de la tesis y al estar trabajando en el perfeccionamiento de la robustez del sistema, se hizo palpable que el eslabón débil no era ya el seguimiento, sino la segmentación. Se puede continuar en el futuro investigando en la forma de perfeccionar la segmentación. Algunas ideas que han surgido para mejorarla son el agregar transformadas de Hough para el caso de detectar círculos, o incluso el uso de texturas a la par de una transformada SIFT (del inglés *Scale-invariant feature transform*). El mejorar la segmentación incrementaría inmediatamente la robustez del sistema.

Las perspectivas a futuro de este sistema son enormes, puesto que las cámaras son cada vez más populares y cada vez más dispositivos electrónicos las incluyen. Sólo por mencionar algunos dispositivos se pueden nombrar: celulares, agendas personales (PDA), computadoras portátiles y ultraportátiles. Todo estos dispositivos son potenciales plataformas para el uso de esta tecnología y existen en la actualidad millones de ellos siendo usados por personas en todo el mundo. En éstos dispositivos el sistema monocular tiene una enorme ventaja sobre otros sistemas.

ÍNDICE DE FIGURAS

REFERENCIAS

[1] Coda software. *http://www.charndyn.com*.

[2] Qualisys software. *http://www.qualisys.com*.

[3] Fugl-Meyer A. R., Jaasko L., Leyman I., Olsson S., and S. Steglind. The post-stroke hemiplegic patient: a method for evaluation of physical performance. *Scand J Rehabil Med*, 7:13–31, 1975.

[4] Tapus Adriana, Tapus Cristian, and Maja J Mataric. Hands-off therapist robot behavior adaptation to user personality for post-stroke rehabilitation therapy. *IEEE International Conference on Robotics and Automation*, 2007.

[5] A. Bharatkumara, K. Daigle, M. Pandy, Q. Cai, and J. Aggarwal. Lower limb kinematics of human walking with the medial axis transformation. *Proc. of IEEE Workshop on Non-Rigid Motion*, pages 70–76.

[6] AM. Black, Y. Yaccob, A. Jepson, and D. Fleet. Learning parameterized models of image motion. *Proc. of CVPR*, pages 561–567.

[7] Derek Bradley and Gerhard Roth. Natural interaction with virtual objects using vision-based six dof sphere tracking. In *ACE '05: Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 19–26, New York, NY, USA, 2005. ACM.

[8] Gary R. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, (Q2):15, 1998.

[9] B.R. Brewer, R. Klatzky, and Y. Matsuoka. Feedback distortion to overcome learned nonuse: A system overview. *IEEE Engineering in Medicine and Biology*, 3:1613–1616, 2003.

[10] Burgar C, Lum P, Shor P, and van der Loos H. Development of robots for rehabilitation therapy: The palo alto va/standford experience. *Journal of Rehabilitation Research and Development*, 2002.

[11] Cheng Chen, Christopher Tay, Kamel Mekhnacha, and Christian Laugier. Dynamic environment modeling with gridmap: a multiple-object tracking application. In *Proc. of the Int. Conf. on Control, Automation, Robotics and Vision*, December 2006.

[12] G. K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. *ACM SIGGRAPH*, pages 77–84.

[13] C. Coue, T. Fraichard, P. Bessiere, and E. Mazer. Multi-sensor data fusion using bayesian programming: an automotive application. In *International. Conf. Intelligent Robots and Systems*, 2003.

[14] Comaniciu D. and Meer P. Mean shift analysis and applications. *Proc. International Conference on Computer Vision*, pages 1197–1203, 1999.

[15] Comaniciu D. and Meer P. Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[16] DeMenthon D. Spatio-temporal segmentation of video by hierarchical mean shift analysis. *Statistical Methods in Video Processing Workshop*, 2002.

[17] Scott D.W. *Multivariate Density Estimation: Theory, Practical, and Visualization*. John Wiley and Sons, 1992.

[18] J Eriksson, M. Matarić, and C Winstein. Hands-off assistive robotics for post-stroke arm rehabilitation. *Int. Conf. on Rehabilitation Robotics*, pages 21–24, 2005.

[19] P. Fieguth and D. Terzopoulos. Color based tracking of heads and other mobile objects at video frame rates, 1997.

[20] J. D. Foley, A. van Dam, Feiner S. K., and J. F. Hughes. *Computer graphics: principles and practice*. Addison-Wesley Longman Publishing Co., 2 edition, 1990.

[21] Alexandre R.J. Francois. Real-time multi-resolution blob tracking. Technical report, Institute for Robotics and Intelligent Systems University of Southern California, 2004.

[22] Sidenbladh H. *Probabilistic Tracking and Reconstruction of 3D Human Motion in Monocular Video Sequences*. PhD thesis, Dept. of Numerical Analysis and Comp. Sci., 2001.

[23] N. R. Howe, M. E. Leventon, and W. T. Freeman. Bayesian reconstruction of 3d human motion from single-camera video. Technical report, A Mitsubishi Electric Research Laboratory (MERL), 1999.

[24] M. Hunke and A. Waibel. Face locating and tracking for human-computer interaction, 1994.

[25] Kiratli J. Telehealth technologies for monitoring adherence and performance of home exercise programs for persons with spinal cord injury: Tele-exercise. *Exercise and Recreational Technologies For People With Disabilities: State of the Science*, 2006.

[26] Fukunaga K. and Hostetler L. D. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Information Theory*, 21:32–40, 1975.

[27] Sucar L.E. and Azcárate G. Gesture therapy: A low-cost vision-based system for motor rehabilitation after stroke. 2007.

[28] Rabiner L.R. A tutorial on hidden markov models an selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, 1989.

[29] Wand M.P. and Jones M. C. *Kernel Smoothing*. Chapman and Hall, 1995.

[30] G. Eliezer Quintana. Calificación de gestos terapéuticos del brazo humano con modelos ocultos de markov. Master's thesis, Instituto Nacional de Astrofísica, Óptica y Electrónica, 2007.

[31] Sanchez R., Reinkensmeyer D., Shah P., and Liu J. Monitoring functional arm movement for home-based therapy after stroke. *Proceedings of the 2004 Engineering in Medicine and Biology Society Meeting, IEEE*, 14(1-5):4787–4790, 2004.

[32] Duda R. O., Hart P. E., and D. G. Stork. *Pattern Classification*. John Wiley and Sons, 2000.

[33] David J. Reinkensmeyer, Clifton T. Pang, Jeff A. Nessler, and Chris C. Painter. Java therapy: Web-based robotic rehabilitation. *IEEE Transactions on Neural Science and Rehabilitation Engineering*, 10(2):102–108.

[34] R J. Sanchez, J Y. Liu, R Smith S Rao, P Shah, T Rahman, S C. Cramer, J E. Bobrow, and D J. Reinkensmeyer. Automating arm movement training following severe stroke: Functional exercises with quantitative feedback in a gravity-reduced environment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(3):378–389, 2006.

[35] C. Sminchisescu. PhD thesis, National Politechnique de Grenoble (INRIA), 2002.

[36] Christopher Tay, Kamel Mekhnacha, Cheng Chen, Manuel Yguel, and Christian Laugier. An efficient formulation of the bayesian occupation filter for target tracking in dynamic environments. *International Journal Of Autonomous Vehicles*, To Appear Spring, 2007. (Accepted) To be published.

[37] Tao Y. and Hu. H. Building a visual tracking system for home-based rehabilitation. *Proc. of the 9 Chinese Automation and Computing Society Conference*, pages 343–448, 2003.

[38] Changjiang Yang, Ramani Duraiswami, Daniel DeMenthon, and Larry Davis. Mean-shift analysis using quasi-newton methods. Technical report, Perceptual Interfaces and Reality Laboratory, University of Maryland,.

[39] Manuel Yguel, Christopher Tay, Kamel Mekhnacha, and Christian Laugier. Velocity estimation on the bayesian occupancy filter for multi-target tracking. Technical report, INRIA, 2006.

[40] Cheng Yizong. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799, 1995.


INAOE


INAOE