



VICERRECTORÍA DE INVESTIGACIÓN Y POSTGRADO  
DIRECCIÓN DE FORMACIÓN POSTGRADUADA

# SISTEMA ESTEREOSCÓPICO PARA EL SEGUIMIENTO DEL INSTRUMENTAL QUIRÚRGICO EN UN NEURONAVEGADOR

**Tesis presentada en opción al título de  
Máster en Informática Aplicada**

Autores: Ing. Mileydi Moreno Mirabal

Tutores: MSc. Iván Pérez Mallea

DrC. Liesner Acevedo Martínez

**La Habana, 2015**

*... sea bendito el nombre de Dios de siglos en siglos, porque suyos son el poder y la sabiduría. El muda los tiempos y las edades; quita reyes, y pone reyes; da la sabiduría a los sabios, y la ciencia a los entendidos.*

*Dan. 2:20-21*

*A Dios*

---

## Agradecimientos

---

*Doy Gracias Dios, por mi esposo y mi bebé que son muy especiales, por los padres en la Fe y biológicos que me dio, a mis tesistas José Lozano, Alejandro Ravelo y Yuriaski Leyva, mis compañeros de trabajo y a todos los que han contribuido con mi formación académica, al claustro de la maestría, en especial al profesor Dr. Rafael Trujillo y a mis tutores, todos ustedes son partícipes de este resultado.*

---

## Declaración de autoría

---

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales sobre esta, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Ing. Mileydi Moreno Mirabal  
Autor

---

MSc. Iván Pérez Mallea  
Tutor

---

DrC. Liesner Acevedo Martínez  
Tutor

Uno de los componentes principales de un sistema de neuronavegación guiada por imágenes, es el sistema de seguimiento de los instrumentos quirúrgicos. En el año 2012 se desarrolló un prototipo de neuronavegador en la Facultad 5 de la UCI, basado en un sistema de seguimiento monocular con un error de exactitud de  $4.17\text{ mm}$ . Este trabajo se enfoca en la problemática de que el sistema de seguimiento monocular existente en el prototipo de neuronavegador con cámaras de bajo costo, no satisface las necesidades de exactitud requeridas para el seguimiento del instrumental quirúrgico en un sistema de neuronavegación guiada por imágenes. Se plantea como objetivo Implementar un sistema de seguimiento estereoscópico que permita incrementar la exactitud en el seguimiento del instrumental quirúrgico para un sistema de neuronavegación quirúrgica. Se desarrolló un sistema estereoscópico para el seguimiento del instrumental quirúrgico basado en las bibliotecas ArUco, OpenCV, IGSTK y el marco de trabajo Qt. El sistema desarrollado es capaz de adquirir la orientación y posición espacial de varios marcadores visibles para ambas cámaras. Mediante las pruebas realizadas se obtuvo un error de  $1.81\text{ mm}$ , con lo que se demuestra que el uso de las técnicas estereoscópicas mejora considerablemente la exactitud con respecto a la solución monocular. La solución ofrece una exactitud limitada para un contexto real pero permite el desarrollo de funcionalidades para un neuronavegador como una alternativa de bajo costo y que sirva de apoyo a las prácticas docentes en la formación de neurocirujanos.

**Palabras clave:** neuronavegador, sistema de seguimiento, visión estereoscópica, visión por computadoras.

<b>Introducción</b>	<b>1</b>
<b>1 Fundamentación teórica</b>	<b>5</b>
1.1 Cirugía asistida por computadora . . . . .	5
1.2 Navegación guiada por imágenes . . . . .	6
1.3 Componente de seguimiento de un neuronavegador . . . . .	6
1.3.1 Marco estereotáxico . . . . .	7
1.3.2 Brazo mecánico . . . . .	8
1.3.3 Ultrasonido . . . . .	8
1.3.4 Magnetismo . . . . .	8
1.3.5 Ópticos . . . . .	8
1.4 Trabajos relacionados . . . . .	10
1.5 Seguimiento visual de objetos . . . . .	12
1.5.1 Técnicas de detección de objetos . . . . .	13
1.5.2 Técnicas de seguimiento basado en marcadores . . . . .	14
1.6 Visión estereoscópica . . . . .	15
1.6.1 Transformaciones afines y perspectivas . . . . .	15
1.6.2 Disparidad . . . . .	16
1.7 Etapas en el proceso de la visión estereoscópica . . . . .	17
1.7.1 Adquisición de imágenes . . . . .	17
1.7.2 Modelo geométrico de la cámara . . . . .	18
1.7.3 Extracción de características . . . . .	22
1.7.4 Correspondencia . . . . .	23
1.7.5 Determinación de la distancia o triangulación . . . . .	24
1.8 Transformación rígida . . . . .	25
1.9 Proceso de calibración . . . . .	26
1.10 Marco de trabajo . . . . .	29
1.11 Conclusiones parciales . . . . .	30

<b>2 Propuesta de solución</b>	<b>31</b>
2.1 Solución técnica . . . . .	31
2.2 Dispositivo de seguimiento . . . . .	32
2.3 Adquisición de imágenes . . . . .	34
2.4 Calibración estereoscópica . . . . .	35
2.4.1 Distribución del archivo de calibración . . . . .	36
2.5 Rectificación de las imágenes estereoscópicas . . . . .	37
2.6 Detección del marcador . . . . .	38
2.6.1 Obtención de los puntos . . . . .	38
2.7 Triangulación . . . . .	40
2.8 Obtención de la matriz de transformación . . . . .	40
2.8.1 Cálculo de la transformación rígida . . . . .	41
2.9 Arquitectura y diseño . . . . .	42
2.9.1 Arquitectura por capas . . . . .	42
2.9.2 Integración del componente de seguimiento con IGSTK . . . . .	42
2.9.3 Diagrama de clases y patrones de diseño . . . . .	44
2.10 Conclusiones parciales . . . . .	45
<b>3 Análisis y discusión de los resultados</b>	<b>47</b>
3.1 Estimación de la transformación mediante la triangulación . . . . .	47
3.1.1 Descripción del escenario de pruebas . . . . .	48
3.1.2 Análisis de rendimiento . . . . .	48
3.1.3 Análisis de la triangulación . . . . .	48
3.1.4 Análisis de la transformación rígida . . . . .	49
3.2 Resultados experimentales de la triangulación y la estimación de la transformación . . . . .	50
3.3 Seguimiento de la herramienta quirúrgica . . . . .	51
3.3.1 Calibración de la herramienta quirúrgica . . . . .	52
3.4 Conclusiones parciales . . . . .	57
<b>Conclusiones</b>	<b>58</b>
<b>Recomendaciones</b>	<b>59</b>
<b>Glosario</b>	<b>60</b>
<b>Acrónimos</b>	<b>61</b>
<b>Referencias bibliográficas</b>	<b>63</b>

---

## Índice de figuras

---

1.1	Marco estereotáxico . . . . .	7
1.2	Marcador basado en triángulos . . . . .	11
1.3	Marcadores del dispositivo de <i>Claron Technology Inc. s.f.</i> . . . . .	11
1.4	Marcadores del tipo ARTag . . . . .	11
1.5	Correspondencia de un sistema estéreo . . . . .	19
1.6	Sistema de referencia de una cámara . . . . .	20
1.7	Geometría esencial de las imágenes estereoscópicas . . . . .	21
1.8	Etapas en la rectificación . . . . .	22
1.9	Parámetros de calibración de una cámara . . . . .	27
1.10	Distorsiones geométricas . . . . .	28
2.1	Flujo del sistema de seguimiento estereoscópico propuesto . . . . .	33
2.2	Diagrama del dispositivo propuesto . . . . .	33
2.3	Herramienta calibrador estereoscópico . . . . .	36
2.4	Rectificación utilizando OpenCV . . . . .	38
2.5	Proceso de detección de marcadores con ArUco . . . . .	39
2.6	Eliminación de la proyección perspectiva . . . . .	39
2.7	Codificación de los marcadores según ArUco . . . . .	40
2.8	Arquitectura por capas de IGSTK . . . . .	43
2.9	Integración de la solución propuesta con IGSTK . . . . .	44
2.10	Diagrama de clases del diseño de la propuesta de solución . . . . .	44
3.1	Configuración del sistema para tomar las medidas . . . . .	49
3.2	(Raíz del Error Cuadrático Medio (RMS)) de transformación rígida . . . . .	51
3.3	Patrón acoplado que simula la herramienta quirúrgica . . . . .	52
3.4	Pivote de la herramienta de calibración . . . . .	53
3.5	Resultados obtenidos de la calibración de pivote monocular y estereoscópico . . . . .	56
3.6	Herramienta de calibración de pivote utilizando la solución propuesta . . . . .	56

---

## Índice de tablas

---

1.1	Comparación relativa de velocidad entre algoritmos de triangulación . . . . .	24
3.1	Resumen de valores del cálculo de errores de estimación de la transformación . . . . .	50
3.2	Resultados de calibración de pivote con los componentes monocular y la solución propuesta	55

---

## Introducción

---

El proceso de cirugía en ocasiones es complejo, algunas intervenciones pueden durar todo el día y demandar la presencia de varias personas en el quirófano. La planificación de estas cirugías es un proceso delicado tan importante como la ejecución en sí. En ocasiones el cirujano debe abandonar la acción para observar las imágenes médicas y hacerse un mapa mental del área comprometida. El desarrollo de la ciencia y la técnica ha favorecido a esta rama de la medicina, y la cirugía asistida por computadora es una tendencia que se está generalizando en todos los hospitales por las ventajas que aporta. El neuronavegador es una herramienta de asistencia para los neurocirujanos con la cual pueden visualizar las imágenes médicas y observar la posición de instrumentos quirúrgicos dentro del paciente en una cirugía, lo cual es imposible a simple vista. Estos equipos brindan un elevado margen de seguridad a los cirujanos, lo que facilita el tratamiento quirúrgico de lesiones cerebrales. Dada su evolución, existe una alta demanda por parte de los cirujanos y los pacientes, lo cual provoca que cada vez existan menos intervenciones que no usen esta tecnología.

La gran desventaja para la adquisición de estos equipos es su alto costo, las empresas privadas que los desarrollan se aprovechan de la suficiencia de esta rama elitista de la medicina y proveen un conjunto completo de hardware y software, que muchos países subdesarrollados no son capaces de adquirir. El precio en general es debido al software y no tanto al hardware.

El desarrollo alcanzado en las ciencias informáticas en Cuba, posibilita la construcción de un sistema de neuronavegación, que podrá distribuirse por los hospitales del país y en las colaboraciones médicas. Por las carencias económicas existentes, es necesario desarrollar el sistema con materiales económicamente viables y médicamente fiables.

En virtud de lo antes expuesto y debido a la necesidad de utilizar sistemas de navegación guiados por imágenes, la Clínica Central Cira García ha propuesto a la Universidad de las Ciencias Informáticas la fabricación de un sistema de neuronavegación guiada por imágenes. Esta decisión está sustentada por la experiencia de algunos neurocirujanos en hospitales europeos, que utilizan estos sistemas como procedimiento de rutina para brindar a los pacientes servicios de elevada calidad.

Un neuronavegador tiene tres componentes fundamentales. El primero se encarga de visualizar las imágenes médicas tomadas por una tomografía, radiología u otra técnica, y brindar un mecanismo de navegación por ellas en el software. El segundo componente, el cual es objeto de la investigación en cuestión, es el sistema de seguimiento tridimensional<sup>1</sup>. Este componente tiene la tarea de rastrear la posición y orientación espacial del instrumento quirúrgico usado por el cirujano, tanto fuera como dentro de las estructuras anatómicas del paciente. Por la complejidad que representa la zona en la que operan los neurocirujanos, existen probabilidades de generar secuelas e incluso causar la muerte del paciente. Por tanto el componente de software debe ser robusto y ofrecer valores de seguimiento con exactitud. El componente de seguimiento debe brindar al sistema de neuronavegación las coordenadas del instrumento con la mayor aproximación posible a las coordenadas reales. Además, esto se debe hacer en un tiempo interactivo cercano al tiempo real. El tercer componente es el encargado de tomar la información tridimensional adquirida por el componente de seguimiento y correlacionarlas con las imágenes médicas (ENQUOBAHRIE y Ziv YANIV, 2009).

El seguimiento necesita acoplar marcas a los instrumentos quirúrgicos para la navegación intraoperatoria, las cuales pueden ser de naturaleza diferente. Según (MARTÍNEZ MAÑAS, 2004), con sistemas de brazos mecánicos en combinación con marcos estereotáxicos, se puede obtener una alta exactitud hasta alcanzar valores de 1.665 mm. Sin embargo, con los sistemas sin marco la exactitud disminuye y puede llegar a alcanzar valores de hasta 4.8 mm, probados con sistemas comerciales de campo magnético. En el caso de los sistemas ópticos como el neuronavegador de BrainLab en la práctica se han alcanzado valores de 4 mm. Este mismo autor indica que siempre es deseable una precisión máxima, aunque existen procedimientos neuroquirúrgicos que no la requieren. La identificación y localización de estructuras cerebrales y óseas es una acción que se realiza al inicio del procedimiento quirúrgico y para esta acción es suficiente una exactitud de 3 a 4 mm. Este valor es superior a lo que pueden lograr la mayoría de los neurocirujanos. Este valor de exactitud es menor a la alcanzada por los sistemas con marco pero se encuentra dentro del rango de los sistemas de neuronavegación.

En el [Centro Entornos Interactivos 3D \(VERTEX\)](#) de la Facultad 5, se tiene experiencia en áreas como la visión por computadoras y se han realizado trabajos que incluyen el seguimiento de marcadores en imágenes. En el año 2012, se realizó un prototipo de neuronavegador en la Facultad 5 (MARTÍNEZ, 2012). El sistema de seguimiento monocular implementado alcanzó una exactitud de 4,17 mm. En esa investigación se utilizó la biblioteca ARTToolKit de (KATO, s.f.) para el componente de seguimiento. Esta, aunque ofrece un modelo de licencia dual, exige el pago de licencias para el entorno académico y comercial. El modelo de licencia ofrecido, es un argumento que impide que esta sea solución viable para el centro VERTEX. Además, el sistema monocular propuesto tiene la limitación de ofrecer un espacio de trabajo reducido debido a que se deben utilizar marcadores de 80 mm para un seguimiento estable. Con estas dimensiones los marcadores pueden salirse de la vista de la cámara, lo cual no ofrece libertad de movimiento para el instrumento quirúrgico que

---

<sup>1</sup>En el contexto de esta investigación cuando se menciona el término seguimiento se hace referencia al *seguimiento tridimensional* en un escenario quirúrgico.

utiliza el cirujano. Sin embargo, con ese prototipo se comprobó que es posible construir sistemas como éste, aunque se recomienda la utilización de otras técnicas para alcanzar mayor exactitud.

Teniendo en cuenta lo anteriormente expuesto, se plantea el siguiente **problema de investigación**: El sistema de seguimiento monocular existente en el prototipo de neuronavegador con cámaras de bajo costo, no satisface las necesidades de exactitud requeridas para el seguimiento del instrumental quirúrgico en un sistema de neuronavegación guiada por imágenes.

La investigación tiene como **objeto de estudio**: seguimiento 3D de objetos basado en visión por computadora, el **campo de acción** queda constituido por: las técnicas de seguimiento 3D de objetos mediante visión estereoscópica.

Para darle solución al problema, se propone el siguiente **objetivo general**: Implementar un sistema de seguimiento estereoscópico que permita incrementar la exactitud en el seguimiento del instrumental quirúrgico para un sistema de neuronavegación quirúrgica.

Para resolver el problema científico y darle cumplimiento al objetivo general, se plantearon los siguientes **objetivos específicos**:

1. Establecer los referentes teóricos de las técnicas de seguimiento tridimensional utilizadas para constituir el marco teórico de la investigación.
2. Caracterizar las diferentes técnicas de seguimiento utilizadas en los neuronavegadores para seleccionar la más conveniente como propuesta de solución.
3. Analizar la reutilización de bibliotecas para el desarrollo del sistema de seguimiento estereoscópico de la propuesta de solución.
4. Implementar un componente de software capaz de realizar el seguimiento tridimensional del instrumental quirúrgico basándose en una técnica previamente analizada.
5. Realizar pruebas para comprobar la exactitud lograda con la solución.

El proceso de investigación y elaboración del presente trabajo involucra los siguientes métodos científicos de investigación:

**Métodos teóricos:**

- *Histórico - lógico*. Se estudian los antecedentes y las tendencias de los sistemas de seguimiento estereoscópicos que existen, además de los conceptos, vocabularios y términos del campo, que contribuyen al mejor entendimiento del trabajo.

- *Analítico - sintético*. Analizar teorías, documentos e información referentes a sistemas de neuronavegación con el objetivo de extraer los elementos esenciales que se relacionan con el objeto de estudio.
- *Inductivo - deductivo*. Luego de adquirir una serie de conocimientos referentes a la geometría que involucra la estereoscopia, se pueden deducir ideas que pueden ser aplicadas en el problema a tratar.

### **Métodos empíricos:**

- Revisión de la bibliografía en todo tipo de fuentes, para conformar el marco teórico de la investigación.
- Entrevistas a personas especialistas en el tema de seguimiento de objetos y estereoscopia, para obtener información sobre el tema de la investigación, así como para la validación de los resultados esperados.

Esta tesis está estructurada de la siguiente forma:

Capítulo §1: Fundamentación teórica. Se indican las bases teóricas fundamentales de los sistemas estereoscópicos. Se refleja el estado del arte del tema de investigación y se explican conceptos y modelos matemáticos necesarios para el desarrollo de estos sistemas. Se identifican los componentes del sistema de seguimiento estereoscópico basado en imágenes que se utilizarán en la solución.

Capítulo §2: Propuesta de solución. Se describe de forma general la solución propuesta como un proceso que transita por varias etapas. Inicialmente se describen las etapas del proceso de seguimiento tridimensional, la configuración del dispositivo de seguimiento y se explican los pasos para calibrar el sistema de visión estereoscópica. Finalmente se describe el diseño de software propuesto para la solución.

Capítulo §3: Análisis y discusión de los resultados. Se describe el seguimiento de los instrumentos quirúrgicos a partir de la solución propuesta en el capítulo anterior. Se describen los resultados obtenidos por el proceso de triangulación y la **RMS** resultante del cálculo de la transformación rígida asociada a los marcadores. Finalmente se hace un análisis de los resultados alcanzados en términos de la exactitud que ofrece la solución propuesta mediante la calibración de pivote de un prototipo de instrumento quirúrgico.

---

## Fundamentación teórica

---

En este capítulo se abordan los principales elementos teóricos de la presente investigación. Para lograr una mejor comprensión del objeto de la investigación se describen los principales componentes de un neuro-navegador, a través de sus conceptos, haciendo énfasis en el seguimiento del instrumental quirúrgico. Se describen las principales estrategias de seguimiento de un neuronavegador, identificadas en la literatura y se profundiza en los dispositivos de seguimiento óptico basados en grabación de video. Se explican los distintos enfoques y técnicas del seguimiento visual de objetos, así como las etapas que conforman un proceso de visión estereoscópica y la estimación de los parámetros intrínsecos de las cámaras, a través de la calibración. Finalmente se definen las herramientas informáticas necesarias para cumplir con el objetivo de la investigación.

### 1.1. Cirugía asistida por computadora

La **cirugía asistida por computadora (CAS)** tiene como objetivo reducir al máximo el trauma terapéutico. Mediante la **CAS** se logra reducir el tiempo de duración de las intervenciones quirúrgicas y mejorar la eficacia, seguridad y exactitud mediante la introducción de técnicas alternativas a las utilizadas en la cirugía tradicional. La **CAS** utiliza herramientas informáticas que le permiten al cirujano incrementar su rendimiento durante el procedimiento quirúrgico, así como en el diagnóstico y planificación de las cirugías. En la actualidad la navegación quirúrgica guiada por imágenes médicas es una variante de la **CAS** y constituye una de las técnicas de última generación usada por cirujanos.

## 1.2. Navegación guiada por imágenes

La navegación, de forma general, es el proceso de determinar y mantener un camino o trayectoria hacia una ubicación u objetivo. En la medicina específicamente en la cirugía, se adopta el término, para expresar la orientación espacial respecto a un volumen anatómico. La navegación guiada por imágenes tiene como objetivo tomar las coordenadas de instrumentos quirúrgicos durante una operación y mostrar su posición relativa en las imágenes médicas del paciente, para guiar al cirujano en zonas de difícil acceso y visibilidad (LARSON; POLLY JR; GUIDERA et al., 2012).

En la actualidad existen sistemas de navegación guiada por imágenes dirigidos a intervenciones quirúrgicas en regiones anatómicas como el abdomen, cadera, rodillas y cráneo. Debido a que esta investigación se enfoca en los sistemas dirigidos a las intervenciones quirúrgicas craneales se hará énfasis en los neuronavegadores, como también son llamados. Es válido destacar que existe una creciente demanda de este tipo de sistemas por los neurocirujanos a nivel internacional.

Un neuronavegador tiene tres componentes fundamentales. El primero se encarga de visualizar las imágenes médicas tomadas por una tomografía computarizada, resonancia magnética, radiología u otra técnica. Además brinda un mecanismo de navegación a través de las imágenes o las regiones anatómicas reconstruidas como un volumen de visualización en el software. El segundo componente, el cual es objeto de la investigación en cuestión, es el sistema de seguimiento tridimensional. Este componente tiene la tarea de rastrear la posición y orientación espacial del instrumento quirúrgico usado por el cirujano, tanto fuera como dentro de las estructuras anatómicas del paciente. Este componente debe ser robusto y exacto, pues debe brindar al sistema una estimación de las coordenadas lo más cercano posible a las coordenadas reales del instrumento que se está siguiendo. Al mismo tiempo, esa estimación debe hacerse en un tiempo interactivo cercano al tiempo real. El tercer componente es el encargado de tomar la información tridimensional adquirida por el componente de seguimiento y correlacionarlas con las imágenes médicas (ENQUOBAHRIE y Ziv YANIV, 2009).

## 1.3. Componente de seguimiento de un neuronavegador

Los componentes de seguimiento (*tracking*), tienen como objetivo la localización y seguimiento tridimensional de objetos. En el campo quirúrgico los objetos rastreados son los instrumentos quirúrgicos usados por los cirujanos. Estos sistemas progresan constantemente en número, fiabilidad y facilidad de uso. Existen diferentes sistemas de este tipo, los cuales se clasifican principalmente en dos grupos de acuerdo a la estrategia que utilizan, los ópticos y los no ópticos. Cada grupo tiene sus ventajas y desventajas. La diferencia radica en que los dispositivos ópticos requieren una línea de visión directa entre emisores y receptores, por lo que

son afectados por la presencia de obstáculos que impiden la visualización del objetivo a seguir ([ibíd.](#)).

Los componentes de seguimiento utilizados tradicionalmente en la neuronavegación suelen ser los marcos estereotáxicos, brazos mecánicos, basados en ultrasonidos, sistemas de campos magnéticos y ópticos que utilizan luz infrarroja, grabación de video o láser. Estos componentes utilizan diferentes tipos de dispositivos que se describen en esta sección.

### 1.3.1. Marco estereotáxico

Este es el primer método de seguimiento utilizado por los neurocirujanos. Es el de mejor influencia y longevidad en la historia del desarrollo de la navegación quirúrgica. Este método consiste en una estructura rígida (marco estereotáxico), que es firmemente anclada al cráneo del paciente, lo que impide el movimiento relativo entre el marco y el paciente (LUNSFORD; MARTINEZ y LATCHAW, 1986).

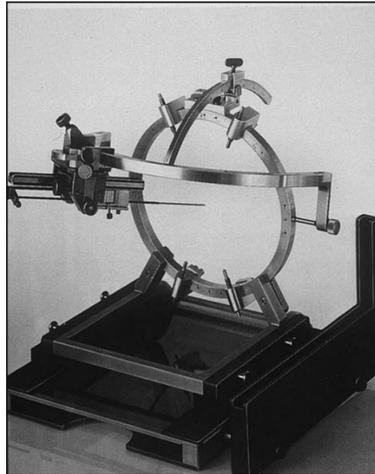


Figura 1.1. Marco estereotáxico.

La principal ventaja de este dispositivo es la exactitud, se han alcanzado niveles aproximados de fracciones de milímetros. No obstante tiene como desventaja la obstrucción del campo quirúrgico por la estructura del marco y los sistemas de guía, solo se puede calcular un número limitado de objetivos quirúrgicos, falta de maniobrabilidad cuando el instrumento avanza en el campo quirúrgico. Por otro lado, el paciente debe someterse al procedimiento de toma de imágenes, transportación y cirugía con el marco fijado al cráneo, por lo que aumenta la invasividad del procedimiento, el tiempo y riesgo en la cirugía.

#### 1.3.2. Brazo mecánico

Este dispositivo consiste en un brazo articulado con medidores de ángulos en las articulaciones, de modo que el componente conoce las coordenadas de un puntero colocado al extremo distal del brazo articulado. Este método mejora la obstrucción del campo quirúrgico causado por el marco estereotáxico, pero su exactitud es limitada comparada con el anterior y es difícil manipular el brazo mecánico en una intervención quirúrgica de estructuras profundas (MARTÍNEZ, 2012).

#### 1.3.3. Ultrasonido

Este tipo de dispositivo utiliza sensores de ultrasonidos. Al instrumento quirúrgico se le fija un emisor de ultrasonido y se colocan sensores en el quirófano que determinan la posición del instrumento por medio de *triangulación* (SEHNERT, 2008). La posición y orientación de los instrumentos se obtiene mediante el cálculo de las diferencias de tiempo que se generan desde que se emite un pulso de ultrasonido en la fuente hasta que se recibe en los sensores. La ventaja de estos dispositivos es que no requieren de una línea de visión directa entre emisores y sensores, pero tiene como inconveniente la susceptibilidad a las interferencias del ruido ambiental y a los cambios de temperatura (BUCHOLZ y SMITH, 1993) en el área de trabajo.

#### 1.3.4. Magnetismo

Estos dispositivos son relativamente nuevos en el campo de las aplicaciones médicas. Su principal ventaja, al igual que en los dispositivos basados en ultrasonidos, es que no requieren de una línea directa de visión, pero tienen como desventaja la susceptibilidad a la distorsión causada por metales cercanos y la exactitud limitada comparada con los dispositivos ópticos (PETERS y CLEARY, 2008). Aunque en la actualidad se han logrado soluciones que atenúan estas interferencias con valores de exactitud inferiores a un milímetro, como en el caso del dispositivo de seguimiento magnético NDI Aurora (*Northern Digital s.f.*).

#### 1.3.5. Ópticos

Los dispositivos ópticos son los que más éxito han tenido en el ambiente clínico (ENQUOBAHRIE y Ziv YANIV, 2009). Estos además de ofrecer alta exactitud y fiabilidad, suelen tener como valor agregado la adaptación a diferentes condiciones de espacio de trabajo, por lo que suelen ser muy flexibles en este sentido. Su uso se ha estandarizado en muchas clínicas, aunque tienen como desventaja que necesitan de una línea de visión limpia de obstáculos. También se aplican en la terapia de radiación para enfermedades de la retina y corrección de movimientos en la reconstrucción tomográfica, donde se necesita una precisión máxima

(MARTÍNEZ, 2012; PETERS y CLEARY, 2008). No obstante los sistemas de seguimiento ópticos utilizan tres estrategias distintas:

- *Basados en infrarrojos.* Estos suelen ser de dos tipos, los activos y los pasivos. Los activos consisten en la ubicación de dos o más cámaras receptoras de luz infrarroja en el quirófano. Las cámaras receptoras detectan la luz infrarroja emitida por diodos emisores de luz (LED) fijados a los instrumentos quirúrgicos, luego por **triangulación** es posible determinar la localización espacial del instrumento (SEHNERT, 2008). Los pasivos funcionan de manera similar, la diferencia radica en que las cámaras infrarrojas ubicadas en el quirófano son las encargadas de emitir la luz y a las herramientas se les fijan marcadores que reflejan la luz infrarroja que reciben, lo que permite conocer su posición mediante la misma técnica que los activos. Un elemento fuerte de este tipo de seguimiento es que tiene la ventaja de que eliminan toda la luz ambiental de otras longitudes de ondas, por tal razón la identificación de los marcadores es más simple y confiable. Un ejemplo de estos es el *Polaris Spectra* de (*Northern Digital s.f.*).
- *Basados en láser.* La posición del cuerpo del paciente y de los instrumentos se estima mediante el muestreo de diferentes haces de luz láser y un foto sensor. Este método no es muy utilizado por el costo que supone la utilización de esta tecnología. Las otras dos estrategias de seguimiento óptico se prefieren porque logran alcanzar resultados de exactitud equivalentes, además de la facilidad de uso y fiabilidad (PETERS y CLEARY, 2008).
- *Basados en grabación de video.* Estos identifican patrones conocidos en secuencias de video tomadas por una o varias cámaras. Ejemplo de un componente comercial de este tipo es el *Claron Tracker* de (*Claron Technology Inc. s.f.*). La proyección perspectiva de los dispositivos usados en sistemas de neuronavegación guiada por imágenes implica una pérdida de información. Por tal motivo es frecuente que posean más de una cámara para realizar cálculos más exactos. Estos dispositivos operan en el espectro de luz visible. La exactitud de estos dispositivos está sujeta a la resolución y a los niveles de ruido de las cámaras utilizadas.

Por las características de todos estos dispositivos, los ópticos basados en luz infrarroja y luz visible son los más producidos por investigadores y empresas dedicadas a este sector, por lo tanto suelen ser más asequibles en el mercado. En general los dispositivos descritos son caros y su precio mínimo ronda los diez mil dólares. A esto se le suma que la mayoría de los fabricantes reconocidos no ofrecen sus soluciones a Cuba en virtud del bloqueo económico y comercial impuesto por los Estados Unidos.

Una ventaja que ofrecen los dispositivos ópticos es que se pueden desarrollar soluciones de prueba con cámaras de bajo costo. Esta afirmación se basa en que las cámaras de bajo costo son compatibles con las cámaras de gama alta, porque la diferencia radica en la resolución y niveles de ruido de las imágenes ofrecidas por

las mismas. En la siguiente sección se abordan algunos trabajos relacionados con prototipos de navegación guiada por imágenes y los sistemas de seguimiento utilizados respectivamente.

### 1.4. Trabajos relacionados

Debido a la demanda que han adquirido los sistemas de neuronavegación guiada por imágenes en los quirófanos, algunas empresas privadas como (*Claron Technology Inc. s.f.*), (*Northern Digital s.f.*) y (*Medtronic s.f.*) proveen esta clase de sistemas. Cada una de estas empresas ofrece su solución de seguimiento óptico con el hardware y su respectiva [Interfaz de programación de aplicaciones \(API\)](#).

Los costos excesivos de estos dispositivos de seguimiento y el conocimiento existente de las técnicas de visión por computadoras han dado lugar al surgimiento de investigaciones a nivel internacional para producirlos. A nivel internacional existen varios trabajos dirigidos a crear prototipos de navegadores guiados por imágenes realizados por grupos de desarrollo que buscan una alternativa a los sistemas de estas empresas.

En la Facultad de Ingeniería de la Universidad de la República de Uruguay se trabajó en el desarrollo de un prototipo de neuronavegador, para reducir los costos de adquisición de un equipo de este tipo en el mercado. En este caso se utilizaron las bibliotecas libres [Imaged-Guided Surgery Toolkit \(IGSTK\)](#), [Insight Segmentation and Registration Toolkit \(ITK\)](#) y [Visualization Toolkit \(VTK\)](#) para la carga, procesamiento y visualización de las imágenes médicas. La diferencia de este trabajo con respecto a la presente investigación radica en que aquí se presenta un neuronavegador como un todo y utilizan un dispositivo de seguimiento óptico profesional. El dispositivo de seguimiento utilizado es el *Polaris Spectra* de (*Northern Digital s.f.*). Este dispositivo cuenta con dos cámaras infrarrojas. Las herramientas que son rastreadas son esferas y discos reflectantes (CARBAJAL; GOMEZ; PEREYRA et al., 2010). El costo de este dispositivo en el mercado supera los diez mil dólares y su venta está prohibida a Cuba.

Otros autores de la misma Universidad de la República diseñaron un sistema de seguimiento basado en marcadores de papel en blanco y negro con un diseño basado en triángulos que cuenta con varias esquinas formando un hexágono (GÓMEZ y RANDALL, 2007). El diseño del marcador se muestra en la figura 1.2. Este trabajo se logró mediante un sistema de visión estereoscópica con dos cámaras que filmaban la escena. En este trabajo no se expresa la posibilidad de generar varios marcadores para el caso de seguir más de una herramienta, aunque es válido destacar que se logró una exactitud con un error de 3.38 mm.

Una solución similar para cirugía maxilofacial fue presentada por (TREVISAN; NEDEL y MACQ, 2008). En este trabajo se utilizó un sistema estereoscópico profesional de *Claron Technology Inc. s.f.* Estos usaban cámaras estereoscópicas y marcadores impresos personalizados de forma circular con cuadros en blanco y negro para detectar las esquinas fácilmente. Es válido destacar que los dispositivos de seguimiento iniciales

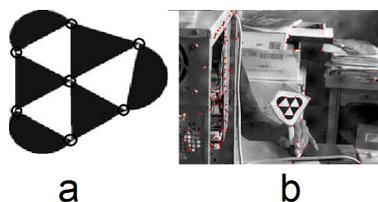


Figura 1.2. Marcador basado en triángulos (a) diseño del marcador, (b) marcador en espacio de trabajo (GÓMEZ y RANDALL, 2007).

de este fabricante eran basados en luz infrarroja y en los últimos 5 años se han dedicado a producir sistemas de seguimiento ópticos basados en luz visible (figura 1.3). Esto se basa en el poder de cálculo que han alcanzado los procesadores y la aplicación de nuevas tecnologías de cálculo como la paralelización que permite hacer cálculos en tiempos interactivos.

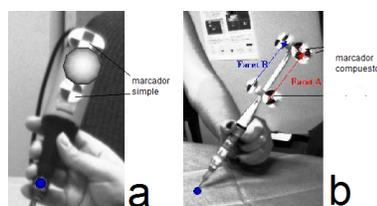


Figura 1.3. Marcadores del dispositivo de *Claron Technology Inc. s.f.* (a) marcador simple (b) marcador multicaras.

Otra variante basada en la detección de marcadores fue propuesta por (NICOLAU; GOFFIN y SOLER, 2005). En ese trabajo los autores presentan un navegador para cirugía laparoscópica. Aunque en este trabajo no se trata de neurocirugía, es válido destacar que emplean la misma estrategia utilizada por (GÓMEZ y RANDALL, 2007). En este caso los marcadores del seguimiento son los de la biblioteca de **Realidad Aumentada (RA) ARTag** (figura 1.4). Para realizar el seguimiento se utilizó un par estereoscópico con dos cámaras a color de resolución 768x576, las cuales se ponían en funcionamiento luego de una etapa de calibración. Los marcadores se utilizaron para detectar sus esquinas y a partir de ese resultado estimar la posición 3D del plano formado por las esquinas a partir de la geometría epipolar. En este trabajo los autores alcanzan una exactitud con un error estimado de 1.6 mm.

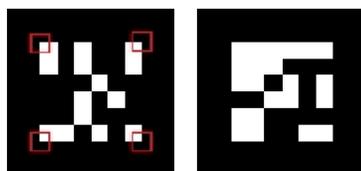


Figura 1.4. Marcadores del dispositivo de ARTag utilizados en (NICOLAU; GOFFIN y SOLER, 2005).

Debido a la factibilidad de la construcción de estos sistemas ópticos, se decidió desarrollar un sistema de seguimiento con una sola cámara en la tesis de grado de (MARTÍNEZ, 2012). Para ese trabajo se utilizó el método de detección y seguimiento de patrones que forman el sistema de marcadores de la biblioteca

ARToolKit, propuestos por (KATO, s.f.). La solución brindada tiene un error estimado de 4.17 mm en la ubicación de los instrumentos. Aunque, carece de la exactitud requerida por las aplicaciones de neurocirugía, los resultados alcanzados demostraron que es posible desarrollar este tipo de sistemas en la [Universidad de las Ciencias Informáticas \(UCI\)](#) con cámaras de bajo costo.

A partir de los trabajos descritos en esta sección, se identificó el uso del seguimiento óptico basado en grabación de video como una de las alternativas viables. Se observó además la visión estereoscópica y el uso de marcadores como las técnicas más usadas en los sistemas de seguimiento alternativos, los cuales utilizan técnicas de visión por computadoras para el seguimiento visual de los objetos en una escena filmada.

De forma general se aprecia en estos trabajos algunos pasos y etapas generales de la visión estereoscópica, aunque los autores no expresan detalles de la implementación de estas estrategias. Por otro lado se aprecia que se alcanza mayor exactitud con un sistema estereoscópico que con uno monocular. Por tal motivo se decide investigar sobre las técnicas de visión por computadoras para realizar el seguimiento visual de objetos mediante visión estereoscópica para darle solución al problema de esta investigación.

### 1.5. Seguimiento visual de objetos

Existen varias técnicas para realizar el seguimiento o seguimiento visual de objetos. Cuando se tiene un video, a diferencia de imágenes aisladas, se desea a veces seguir algunos objetos en particular, que se encuentran en la escena durante el flujo del video. La detección y clasificación de objetos, son pasos que anteceden al seguimiento del objeto en una secuencia de imágenes o fotogramas.

El seguimiento de objetos, se realiza monitorizando los cambios espaciales y temporales durante una secuencia de video, incluyendo su presencia, posición, tamaño, forma y orientación. El seguimiento de objetos es usado en video vigilancia, visión robótica, monitorización de tráfico, animación, deportes como el fútbol y el tenis (ENDRESEN, 2010).

Algunos aspectos a tener en cuenta en el seguimiento de objetos son: la pérdida de información con la proyección perspectiva del mundo real 3D en la imagen 2D, el ruido en la imagen, el movimiento de un objeto difícil de seguir, la oclusión parcial o total del objeto y las estructuras complejas del objeto (PAREKH; THAKORE y JALIYA, 2014). En el artículo de (JACOB y ANITHA, 2012) se comparan varias técnicas de seguimiento de objetos.

Los pasos básicos para el seguimiento de objetos son los siguientes:

1. *Selección de características para el seguimiento.* Seleccionar las características adecuadas es fundamental. La mejor propiedad para las características visuales es la unicidad, así el objeto es distinguido

fácilmente en el entorno. Algunas de las características más usadas son: los puntos de interés, el color, las aristas, el flujo óptico, la textura. Es común, que estas sean elegidas por el usuario en dependencia de la aplicación.

2. *Detección del objeto*. Existen diversas técnicas para detectar objetos. Estas son las basadas en puntos, la diferenciación de marco, flujo óptico, substracción de fondo y segmentación. Estas se explicarán en la siguiente sección.
3. *Seguimiento de objetos*. El objetivo del seguimiento es conocer la trayectoria del objeto en el tiempo, esto se logra con la localización de su posición en cada fotograma del video. Hay dos metodologías distintas para enfocar el problema del seguimiento, de arriba hacia abajo (*hacia adelante*) y de abajo hacia arriba (*de vuelta*). Los métodos de arriba hacia abajo están orientados a encontrar la posición de los objetos en el fotograma actual, usando hipótesis generadas al comienzo del seguimiento basadas en representaciones paramétricas del blanco u objetivo. Estos métodos usan diferentes características de los objetos como color, textura, forma y movimiento. Un método popular en esta categoría es el cambio medio (*mean-shift*). Por otra parte, en el enfoque hacia arriba, los objetos en movimiento son detectados en cada fotograma y se establece una correspondencia con los objetos detectados en fotogramas anteriores. Dentro de esta categoría entran el modelado y substracción de fondo (JALAL y SINGH, 2012).

De los dos enfoques usados en el seguimiento de objetos, el de *arriba hacia abajo* es más seguro y robusto. En el otro enfoque *de abajo hacia arriba* se adquiere la posición del objeto, a partir de su trayectoria por los fotogramas anteriores, este enfoque es usado para rastrear objetos de difícil reconocimiento. Según (ibíd.) puede fallar si hay grandes saltos de los objetos en fotogramas consecutivos. Por esta razón, se decide utilizar el primer enfoque y detectar en cada fotograma el objeto, usando características de fácil detección.

### 1.5.1. Técnicas de detección de objetos

1. *Detector de puntos*. Son usados para encontrar puntos de interés en imágenes. Esta técnica se usa en el contexto de movimiento, estereoscopía y seguimiento. Una característica deseable en un punto de interés es su invarianza a los cambios en la iluminación y al punto de vista de la cámara. Los algoritmos detectores de puntos o esquinas más usados son el operador de interés de Moravec, el detector de puntos de interés de Harris, el detector KLT (Shi-Tomasi) y el detector SIFT (SZELISKI, 2010).
2. *Diferenciación de marco*. La presencia de objetos en movimiento se determina al calcular la diferencia entre dos imágenes consecutivas. Esta técnica ofrece ventajas para una variedad de entornos dinámicos, pero es generalmente difícil de obtener un esquema completo del objeto en movimiento. Se puede

manifiestar el fenómeno *vacío* como consecuencia de una detección inexacta del objeto en movimiento (PAREKH; THAKORE y JALIYA, 2014).

3. *Flujo óptico*. Esta técnica puede obtener una información completa del movimiento, lo representa como una serie de vectores de desplazamiento, que definen el desplazamiento de cada píxel en una región entre dos fotogramas consecutivos. La mayor desventaja de este es la complejidad algorítmica, requiere de más tiempo que otras técnicas y es sensible al ruido, por lo que no es adecuado para ocasiones en que se exige información en tiempo real (JALAL y SINGH, 2012).
4. *Substracción de fondo*. El primer paso en esta técnica es el modelado de fondo, este es el núcleo del algoritmo, se trata de tener un modelo de referencia para poder compararlo en cada fotograma en busca de posibles variaciones, estas variaciones en término de píxeles significan que existen objetos en movimiento. Esta técnica proporciona la información más completa del objeto si el fondo es conocido. Esta técnica tiene dos enfoques: el algoritmo recursivo y el no recursivo, el segundo depende de un *buffer* de almacenamiento (SEN-CHING y KAMATH, 2004).
5. *Segmentación*. Las técnicas de segmentación particionan la imagen en regiones perspectivamente similares. Los dos problemas a tener en cuenta son: el criterio de cuándo una partición es buena y el método para lograr el particionamiento eficiente. Algunos de los métodos son: agrupamiento de desplazamiento medio (*mean-shift clustering*), segmentación de la imagen usando cortes de grafos, y contornos activos (YILMAZ; JAVED y SHAH, 2006).

### 1.5.2. Técnicas de seguimiento basado en marcadores

El uso de marcadores o fiduciaros en una escena, es un modo de simplificar la tarea de detección del objeto. El marcador es un objeto que presenta características conocidas y de fácil rastreo, como son los colores, las aristas y las esquinas. Estas características son fáciles de explotar y confiables para estimar la posición y orientación espacial. Los marcadores se pueden dividir en dos grupos: basados en puntos y extendidos. Los extendidos son muy populares y los más usados son generalmente de forma rectangular, esto reduce el costo computacional cuando son rastreados en una imagen.

Según (MORENO y CRUZ GUEVARA, 2008) existen varias técnicas para realizar el seguimiento utilizando marcadores:

- *Marcadores en forma de puntos*. Estos marcadores han sido usados en la *fotogrametría* por muchos años. Tienen un diseño que permite su rápida y fácil detección. Presentan una apariencia de patrones circulares y son relativamente invariantes a la distorsión perspectiva (*ibíd.*). Su posición en el mundo real es conocida con exactitud, esto se logra de forma manual con láser o un algoritmo llamado Estructura a partir del movimiento (*Structure-From-Motion*). Se diseñan patrones geoméricamente distintos

y una vez que son identificados en una imagen se puede adquirir la posición de la cámara a partir de las correspondencias encontradas. Una desventaja es que este tipo de marcador necesita ser distribuido en varios lugares de la escena y sus posiciones necesitan ser medidas con exactitud para poder extraer la posición de la cámara.

- *Marcadores extendidos.* Estos marcadores cuadrados en blanco y negro son introducidos por (KOLLER; KLINKER; ROSE et al., 1997), contienen pequeños cuadrados para su identificación. Los marcadores rectangulares son usados también y según (FUA y LEPETIT, 2005) se demuestra que es necesario solamente un marcador para estimar la posición de la cámara. Durante el proceso de detección de los marcadores la estimación de la posición se ejecuta en tiempo real, por lo que puede ser usado en cada fotograma dentro del flujo de video.

El seguimiento 3D de los marcadores cuadrados no requiere inicialización manual, su uso se ha extendido porque llevan a una solución robusta y de bajo costo. Existen varias bibliotecas que brindan soporte para marcadores extendidos como ARToolKit (KATO, s.f.), ArTag (FIALA, 2005) y ArUco (AVA, s.f.), por lo que se selecciona esta técnica para realizar el seguimiento en la solución propuesta. Además es una estrategia que se identificó en trabajos relacionados con la visión estereoscópica, como una técnica frecuentemente utilizada en los sistemas de seguimiento.

## 1.6. Visión estereoscópica

La visión estereoscópica es el proceso de entender o analizar superficies visibles de objetos tridimensionales, basado en imágenes. Está basada en los sistemas visuales de los humanos y animales, los cuales prueban que la visión estereoscópica funciona en ambientes complejos. Con el análisis de dos imágenes tomadas de un objeto que conforman un par estéreo, se puede recuperar la información perdida a causa de la proyección perspectiva en el plano bidimensional de la imagen. La proyección perspectiva y la afín son transformaciones que se deben tener en cuenta dentro del proceso de visión por computadoras, porque son la clave para la recuperación de la información espacial de los objetos detectados (ENDRESEN, 2010).

### 1.6.1. Transformaciones afines y perspectivas

Una transformación afín en matemática, se refiere a la transformación que se puede expresar con una multiplicación de una matriz y una adición vectorial. Una transformación afín de un paralelogramo  $ABCD$  es otro paralelogramo  $A'B'C'D'$ , esta transformación mantiene la condición de paralelismo de las rectas, pero puede modificar los ángulos y las longitudes de los segmentos. En la visión por computadoras, la transformación afín se basa en el escalado, rotación y traslación en el espacio bidimensional. Una transformación afín se

puede describir de la siguiente forma:  $x' = Ax$  donde  $A$  es el elemento básico, una matriz de transformación de dimensiones  $2 \times 3$ , que es la encargada de mapear las coordenadas de un punto cuando se le realiza la transformación (DÍAZ, 2014).

$$x' = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{bmatrix} x \quad (1.6.1)$$

Por su parte, la transformación perspectiva está estrechamente ligada a la proyección perspectiva. Esta proyección se pone de manifiesto en el proceso de formación de la imagen, donde se forma una representación bidimensional, a partir de una representación tridimensional del mundo real. En esta conversión se pierde una dimensión.

La forma usual de representar este proceso es con la proyección central, en la cual un rayo de un punto del espacio atraviesa un punto central e intersecta un plano específico llamado plano de imagen (R. HARTLEY y ZISSERMAN, 2003). La transformación perspectiva u homografía determina una correspondencia entre dos figuras geométricas, de forma que a cada uno de los puntos y rectas de una de ellas le corresponden, respectivamente, un punto y una recta de la otra. La matriz  $\tilde{H}$  que representa la transformación perspectiva se expresa en coordenadas homogéneas y es de dimensiones  $3 \times 3$ .

$$\tilde{x}' = \tilde{H}x \quad (1.6.2)$$

La coordenada resultante es homogénea y debe ser normalizada para obtener un resultado no homogéneo.

$$x' = \frac{h_{00}x + h_{01}y + h_{02}}{h_{20}x + h_{21}y + h_{22}} \quad y' = \frac{h_{10}x + h_{11}y + h_{12}}{h_{20}x + h_{21}y + h_{22}} \quad (1.6.3)$$

Muchas veces es necesario reconstruir la dimensión perdida a causa de la proyección perspectiva, esto se puede lograr gracias a la disparidad que se obtiene a partir de dos perspectivas de un mismo punto.

### 1.6.2. Disparidad

Los humanos, y cualquier otro animal que tenga los ojos de tal forma que pueda ver un punto con ambos a la vez, son capaces de percibir profundidad basándose en las diferencias entre las imágenes que captan el ojo izquierdo y el ojo derecho. Esta disparidad o paralaje entre las dos imágenes, es inversamente proporcional a la distancia desde el observador. Medir automáticamente la disparidad, estableciendo correspondencias exactas entre las imágenes es un reto (SZELISKI, 2010).

Cuando el objeto se encuentra relativamente cerca de las cámaras, arroja diferentes posiciones en las fotografías, pero cuando no se detecta ninguna disparidad en el objeto, significa que está a distancia infinita. Esto ocurre generalmente cuando el objeto se encuentra a una distancia lejana de las cámaras y los rayos homólogos son paralelos. Este es el elemento final en el proceso de visión estereoscópica, a partir del cual se puede estimar la profundidad a la que se encuentra un objeto. De esta forma queda por analizar las etapas del proceso de visión estereoscópica para crear un bosquejo general del seguimiento visual de los objetos.

## 1.7. Etapas en el proceso de la visión estereoscópica

La visión estereoscópica se basa en la adquisición de imágenes que forma un par estéreo y la modelación de las cámaras, a través de la estimación de su geometría intrínseca. Una descomposición en mayor detalle del proceso de visión estereoscópica fue realizada por (BARNARD y FISCHLER, 1982). Según esta descomposición, el proceso completo contempla seis pasos principales:

1. Adquisición del par estéreo de imágenes.
2. Modelado de la cámara (geometría de cámaras).
3. Extracción de las características.
4. Correspondencia de las imágenes (características).
5. Determinación de la distancia (triangulación).
6. Interpolación, cuando sea necesaria.

Estos pasos se realizan de forma secuencial en el orden que se presentan. De estos pasos el más complicado es hallar la correspondencia entre conjuntos de píxeles que definen características específicas en las imágenes. Este paso depende en gran medida de la elección de características que representan estos conjuntos de píxeles (MONTALVO, 2010).

### 1.7.1. Adquisición de imágenes

La adquisición de imágenes puede realizarse de diversas formas. Pueden ser tomadas simultáneamente o mediante intervalos de tiempo de una duración determinada. Además las imágenes pueden tomarse desde localizaciones y direcciones ligeramente distintas o totalmente diferentes. Las imágenes pueden tomarse con diferencia temporal, ejemplo de esto, cuando se usa sólo una cámara móvil y se hacen tomas de la

escena desde distintas posiciones, en ese caso influye el momento del día en que fueron capturadas, las condiciones de iluminación y cualquier elemento que haya cambiado la escena (MONTALVO, 2010). En estos casos la reconstrucción no es necesario que se realice en tiempos interactivos. No obstante, en el presente trabajo, pretende detectar con exactitud milimétrica las coordenadas espaciales de un instrumento quirúrgico en tiempos interactivos, por lo que la adquisición debe ser controlada por ciertas restricciones para asegurar la validez de la detección de los instrumentos quirúrgicos. El tiempo de adquisición del par de imágenes debe ser lo más simultáneo posible en cada cámara. Una imagen con al menos un segundo de diferencia con respecto a la anterior, puede asociar un punto distinto en el espacio. Esto puede resultar en una diferencia sustancial entre las coordenadas reales y las estimadas para ese instante de tiempo.

### 1.7.2. Modelo geométrico de la cámara

Un modelo de cámara, son los atributos geométricos y físicos más importantes para la visión estéreo. Este modelo puede tener una componente relativa, la cual relaciona el sistema de referencia de una cámara con el de la otra. Es independiente de la escena y también puede tener una componente absoluta, la cual relaciona el sistema de referencia de una de las cámaras con un sistema de referencia en la escena. El modelo básico utiliza dos cámaras que tienen sus ejes ópticos paralelos, con la distancia que los separa como su línea base. Esto indica que sus ejes ópticos son perpendiculares a la línea base y que sus líneas de exploración o líneas epipolares son paralelas a la línea base.

Las líneas epipolares son líneas que unen la imagen izquierda y la imagen derecha a partir de las proyecciones de un punto. Cualquier punto del espacio tridimensional unido a los dos centros de proyección de las cámaras define un plano, llamado plano epipolar. La intersección de un plano epipolar con el plano de proyección de una cámara define una línea epipolar. Para todos los puntos, cuyas proyecciones izquierdas estén contenidas en una línea epipolar en la imagen izquierda, sus proyecciones derechas deben estar contenidas también sobre la línea epipolar en la imagen derecha, y viceversa (R. HARTLEY y ZISSERMAN, 2003).

El principal problema de las técnicas de visión estereoscópica es encontrar la correspondencia entre los puntos de las imágenes. Se debe efectuar una búsqueda en dos dimensiones, tanto en el eje  $X$  como en el eje  $Y$ . Para reducir la complejidad de este proceso se puede llegar a una búsqueda unidimensional. Basta con situar y orientar las cámaras de forma que sólo exista un desplazamiento horizontal entre ellas, los ejes ópticos deben ser paralelos y los ejes de abscisas (eje  $X$ ) de cada una de las cámaras deben ser coincidentes como sucede en la figura 1.5.

Bajo esta situación, las líneas epipolares que definen el plano epipolar son coincidentes, esto consigue simplificar la búsqueda del emparejamiento al recorrer las imágenes por filas. Con esta geometría se obtiene la denominada restricción epipolar, de manera que, en el sistema de ejes paralelos convencional todos los planos epipolares originan líneas horizontales al cortarse con los planos de las imágenes. En un sistema con

la geometría anterior, se obtiene un valor de disparidad  $d$ , para cada par de puntos emparejados  $P_I(X_I, Y_I)$  y  $P_D(X_D, Y_D)$  dado por  $d = X_I - X_D$ . Con el valor de disparidad para cada punto de la imagen se construye un matriz o mapa de disparidad, en el que cada punto de la imagen contiene su valor de disparidad, y así se puede calcular la profundidad para cada punto de la escena (ibíd.).

En la figura 1.5 se pueden ver todos los elementos comentados. El sistema de referencia absoluto, que pertenece al espacio real tridimensional, está dado por el origen de coordenadas  $O$  y los ejes  $\{X, Y, Z\}$ . El sistema de referencia relativo de la cámara izquierda está dado por el origen de coordenadas  $O_I$  y los ejes  $\{X_I, Y_I, Z_I\}$ , y el sistema de referencia de la cámara derecha tiene origen de coordenadas en  $O_D$  y ejes  $\{X_D, Y_D, Z_D\}$ .

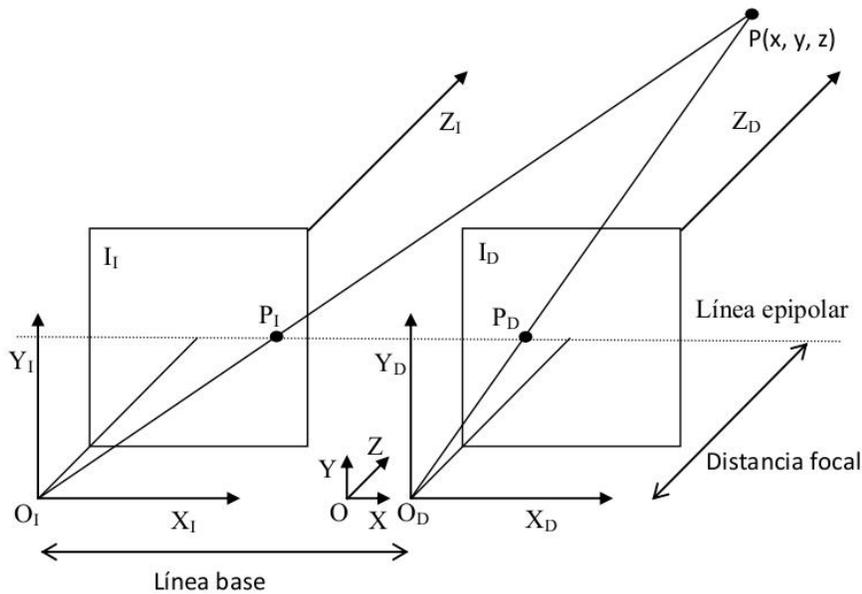


Figura 1.5. Correspondencia de un sistema estéreo (MONTALVO, 2010).

En ambos sistemas, el de la cámara izquierda y el de la derecha, se hace coincidir el centro de proyección óptico con el origen de coordenadas. Según esta notación, todos los elementos que se refieran a la cámara izquierda y se expresen en su sistema de referencia tendrán el subíndice  $I$ , de forma idéntica para la cámara derecha pero con el subíndice  $D$ . El punto  $P$  en la escena tridimensional se proyecta en la imagen izquierda como el punto  $P_I$  y en la imagen derecha tiene como proyección  $P_D$ . Para obtener la proyección de un punto en una imagen se hace pasar un rayo por su centro óptico y por el propio punto, en la intersección de este rayo con el plano imagen se formará la proyección del punto. Los rayos de proyección  $PO_I$  y  $PO_D$  definen el plano de proyección del punto de la escena 3D, (el plano epipolar). Como se puede comprobar en la figura 1.5 se han hecho coincidir los ejes  $X$  de los dos sistemas relativos, así las imágenes estarán en correspondencia y las coordenadas  $Y$  de los puntos  $P_I$  y  $P_D$  serán idénticas. Como las líneas epipolares son paralelas se consigue simplificar la búsqueda. La línea epipolar correspondiente a un punto se puede hallar con las matrices esencial y fundamental.

En la figura 1.6 se muestra una cámara junto a su sistema de referencia. El origen de coordenadas  $O$  coincide con el centro de proyección, y el eje  $Z$  está superpuesto con el eje óptico del sistema.

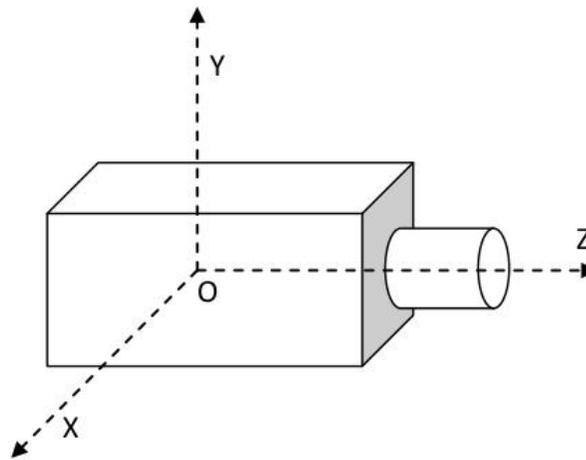


Figura 1.6. Sistema de referencia de una cámara.

A pesar de la teoría expuesta anteriormente, en la realidad muy pocas veces las imágenes que forman el par estereoscópico se encuentran alineadas horizontalmente. Además en el proceso de adquisición se introducen deformaciones en las imágenes que degradarán la exactitud en la medida de profundidad a menos que sean corregidas. Dos ejemplos de estas deformaciones son la distorsión radial y la distorsión tangencial. En este caso, es necesario realizar un proceso previo de calibración de cámaras con el fin de corregir dichas anomalías (ver sección 1.9).

### Matriz esencial y fundamental

Las líneas epipolares pueden ser calculadas mediante el uso de dos matrices. La *matriz esencial*  $E$  contiene la información de la traslación y rotación referente a ambas cámaras en el espacio físico. Según (SZELISKI, 2010) con ella se puede mapear un punto  $x_0$  en la imagen de la izquierda dentro de una línea  $l_1 = Ex_0$  en la imagen derecha. La matriz esencial no contiene la información intrínseca de las cámaras, y relaciona los puntos en coordenadas físicas o de cámara, no en coordenadas de píxeles (BRADSKI y KAEHLER, 2008).

La *matriz fundamental* contiene toda la información geométrica espacial de las cámaras respectivamente. En la práctica, usualmente se trabaja con coordenadas de píxeles. Para encontrar una relación entre un píxel de una imagen y su correspondiente línea epipolar en la otra imagen se debe introducir la información intrínseca de las cámaras. La matriz fundamental es como la esencial pero esta opera en coordenadas de píxeles (ibíd.). Según (SZELISKI, 2010) se puede calcular la línea epipolar  $l_1$  correspondiente al punto  $x_0$  de la imagen izquierda en la imagen derecha con la ecuación  $l_1 = Fx_0$ . En el artículo de (BAILEY y WOLF, 2007) se

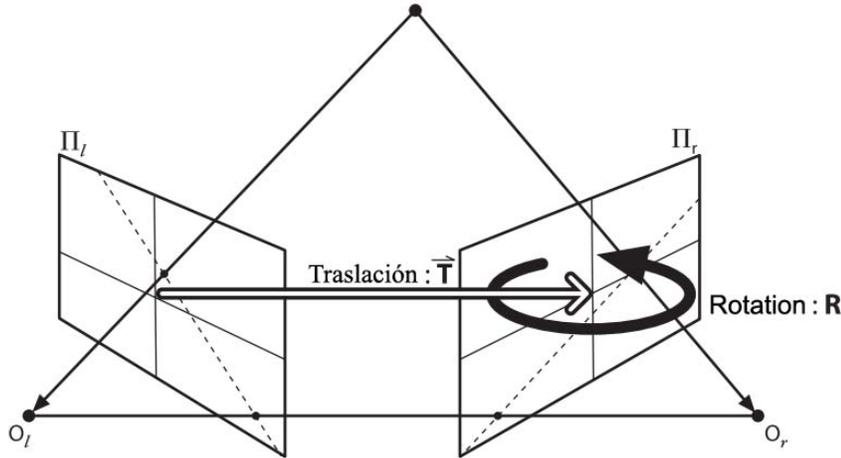


Figura 1.7. Geometría esencial de las imágenes estereoscópicas: se captura en la matriz  $E$ , que contiene toda información de rotación  $R$  y traslación  $T$  que describe la localización de la segunda cámara con respecto a la primera en coordenadas globales (BRADSKI y KAEHLER, 2008).

explica con claridad el propósito de la matriz fundamental.

### Rectificación de las imágenes

Con el uso de la geometría de las cámaras, se puede usar la línea epipolar de cada píxel en una imagen para restringir su búsqueda en la otra imagen. Una estrategia eficiente puede ser obtenida al rectificar las imágenes en un paso previo, de modo que las líneas de búsqueda horizontales correspondientes sean líneas epipolares. Para rectificar las imágenes se utilizan los parámetros obtenidos en la calibración previa, luego cualquier píxel de la imagen izquierda presenta una coordenada "y" idéntica al correspondiente en la imagen derecha (SZELISKI, 2010).

Los pasos para rectificar las imágenes son los siguientes (figura 1.8):

1. Rotar ambas cámaras para que sus ejes  $x$  coincidan.
2. Dado que existe un grado de libertad en la inclinación, las rotaciones más pequeñas que se obtienen se deben utilizar.
3. Para determinar el giro deseado alrededor de los ejes ópticos, el vector de arriba (el eje  $y$  de la cámara) se pone perpendicular a la línea central de la cámara. Esto asegura que las líneas epipolares correspondientes sean horizontales y que la disparidad de los puntos en el infinito es 0.
4. Por último, se escalan las imágenes si es necesario, se magnifica la imagen más pequeña y se reduce al mínimo las últimas distorsiones horizontales.

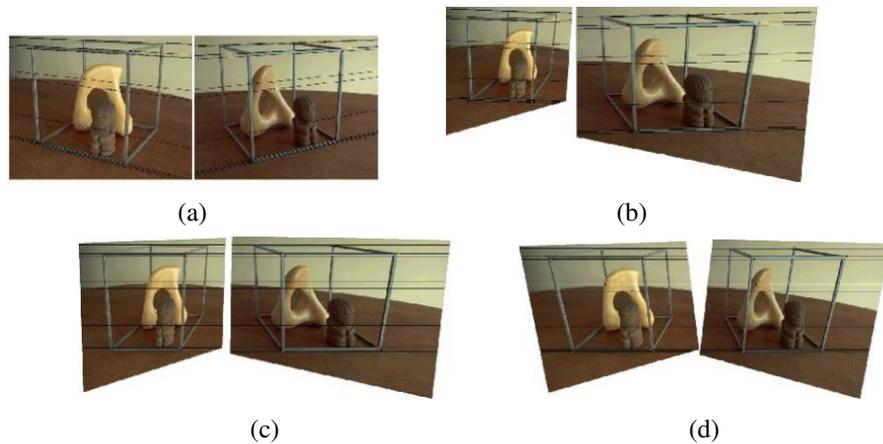


Figura 1.8. Etapas en la rectificación: (a) Par de imágenes originales con varias líneas epipolares; (b) imágenes transformadas, las líneas epipolares quedan paralelas; (c) imágenes rectificadas de manera que las líneas epipolares son horizontales y verticales en correspondencia; (d) rectificación final que reduce al mínimo las distorsiones horizontales (SZELISKI, 2010).

### 1.7.3. Extracción de características

En este paso de la visión estereoscópica, se procede a extraer las características a las que luego, en el próximo paso se le buscará la correspondencia en las imágenes. Por lo tanto, este paso está estrechamente ligado al de correspondencia. Primero se suele decidir qué método utilizar al realizar la correspondencia entre imágenes y según las características que se empleen, éstas serán las que se extraigan de las imágenes. Existen dos clases de técnicas para determinar la correspondencia entre dos imágenes estereoscópicas: las técnicas basadas en el área (*area-based*) y las técnicas basadas en las características (*feature-based*) (MONTALVO, 2010).

Como esta investigación trata dos áreas del conocimiento: *el seguimiento de objetos y la visión estereoscópica*, hay conceptos que se interceptan. Este es el caso de la elección de características visto en la sección 1.5. Los *puntos de interés* son características que presentan las imágenes que identifican una subregión de la imagen unívocamente, estas pueden ser esquinas, formas o regiones de cambio brusco de gradientes (líneas). En esta investigación las características seleccionadas fue una combinación de detección de aristas y esquinas por su bajo coste computacional. Los puntos de interés son invariantes a los cambios de iluminación y a la posición de las cámaras. Cuando sea necesario buscar la correspondencia, solo se buscarán dichos puntos de forma directa en la otra imagen en lugar de buscar por regiones completas, este proceso es llamado *correspondencia escasa*.

Esta decisión está sustentada por la necesidad de obtener una exactitud alta en el cálculo de las posiciones espaciales y porque el diseño de los marcadores supone la aparición de aristas bien definidas en las imágenes. Es preferible concentrarse en unos pocos puntos y refinarlos, que trabajar con zonas más grandes de la imagen.

#### 1.7.4. Correspondencia

En este paso, dado el método seleccionado, que puede ser basado en área (píxeles) o en características, se procede a localizar para un punto en el espacio tridimensional su proyección en cada uno de las imágenes del par estereoscópico. Los métodos basados en características, restringen la búsqueda a un conjunto disperso de características. Se emplean propiedades simbólicas y numéricas de las características, obtenidas a partir de los llamados descriptores. En los métodos basados en área, los elementos a comparar son ventanas de la imagen de dimensión fija y el criterio de semejanza dentro de la región de búsqueda. Dentro de este segundo grupo se pueden distinguir métodos como: la [Suma de diferencias absolutas \(SAD\)](#), la [Suma de diferencias al cuadrado \(SSD\)](#), la [Correlación cruzada normalizada \(NCC\)](#) y el coeficiente de correlación de Pearson. La explicación detallada de estos métodos se encuentran en la siguiente bibliografía: (LÓPEZ-VALLES; FERNÁNDEZ-CABALLERO y FERNÁNDEZ, 2005); (KANADE y OKUTOMI, 1994); (OKUTOMI y KANADE, 1993); (MATTHIES; KANADE y SZELISKI, 1989) y (HIRSCHMULLER; INNOCENT y GARIBALDI, 2002).

Ambos procesos de correspondencia utilizan propiedades de la realidad física, que son formuladas en términos de restricciones. Las restricciones utilizadas se enuncian a continuación:

- *Epipolar*. Las imágenes de una entidad 3D deben proyectarse sobre una línea epipolar. Esta restricción se deriva de la geometría de las cámaras y requiere que las cámaras estén alineadas.
- *Semejanza*. Las dos imágenes de una entidad 3D deben tener propiedades o atributos similares.
- *Unicidad*. Para cada característica en una imagen debe haber una única característica en la otra imagen, salvo que se produzca una oclusión y no haya correspondencia de alguna característica.
- *Orden posicional*. Dadas dos características en una determinada imagen, por ejemplo la izquierda, situada una a la derecha de la otra, esta restricción supone que este orden se mantiene en la imagen derecha para sus respectivas características homólogas.
- *Continuidad de la disparidad*. Asume que las variaciones de disparidad en la imagen son generalmente suaves, si se considera un mapa de disparidad, éste se presenta de forma continuo salvo en unas pocas discontinuidades (PAJARES y CRUZ, 2007; SCHARSTEIN y SZELISKI, 2002).

La correspondencia obtenida puede ser escasa si se seleccionó el tipo basado en características o densa si se escogió el basado en áreas o píxeles. Los puntos de interés a buscar son las esquinas del marcador previamente detectadas sus aristas. De esta forma el algoritmo encargado de buscar la correspondencia no necesita pasar por toda la imagen para obtener el mapa de disparidad completo, sino que solamente busca la correspondencia de los cuatro puntos que definen sus esquinas.

### 1.7.5. Determinación de la distancia o triangulación

Para hallar la posición de un punto en el espacio a partir de un par estereoscópico, es preciso tener la posición de las dos proyecciones del punto, la calibración de las cámaras estereoscópicas y su pose. Este proceso requiere de la intersección de dos rayos en el espacio y es comúnmente llamado **triangulación**. Cuando hay ausencia de ruido, el problema es trivial. Cuando existe ruido en las imágenes los dos rayos generalmente no se interceptan y es preciso hallar una aproximación al mejor punto de intersección.

Para calcular la intersección de los rayos se han propuesto varios métodos. En (R. I. HARTLEY y STURM, 1997) se explican varios algoritmos usados en la **triangulación** y realizan a cada uno de ellos pruebas de eficiencia y el error cometido. Dentro de las pruebas de error se mide el error 3D y 2D para la reconstrucción euclidiana y el error en la reproyección 2D. Además de esto se realizan evaluaciones de los resultados de estos algoritmos en imágenes reales.

En ese trabajo se comparan los algoritmos *polinomial*, *polinomial-abs*, *lineal-valor-propio* (*linear-eigen*), *iterativo-valor-propio* (*iterative-eigen*), *punto medio*, *lineal-LS* (*least squares*) e *iterativo-LS*. Es válido destacar que el valor de comparación arrojado por la ejecución de los algoritmos representa unidades de tiempo relativas (*ibíd.*).

Tabla 1.1. Comparación relativa de velocidad entre algoritmos de triangulación. Se expresan números que representan la diferencia relativa de ejecución entre los algoritmos, los cuales procesan miles de puntos.

<b>Algoritmo</b>	
polinomial-abs	60
polinomial	28
iterativo-valor-propio	10
lineal-valor-propio	6
iterativo-LS	6
lineal-LS	4
punto medio	4

Todos los métodos se desempeñan bien para la reconstrucción euclidiana, en términos de error 3D. El polinomial-abs es ligeramente superior que el polinomial en cuanto al error. Ambos métodos tienen como desventajas que no son especialmente fáciles de generalizar para más de dos imágenes y son más lentos que los otros.

El iterativo-LS es un método eficiente, aunque a veces no converge, es más rápido que el polinomial y es muy cercano a ser invariante a la proyección. Además según estos autores el algoritmo iterativo-LS es ligeramente más rápido que el iterativo-valor-propio.

Según (VITE-SILVA; CRUZ-CORTÉS; TOSCANO-PULIDO y FRAGA, 2007) cuando hay grandes niveles

de ruido en las imágenes, y a su vez en las matrices fundamentales y de proyección, el polinomial-abs no se desempeña bien, incluso el iterativo-LS arroja mejores resultados. Esto es un elemento que se debe tener en cuenta a la hora de decidirse por uno de estos algoritmos.

Por otro lado estos algoritmos se utilizan en un contexto donde se necesita generar una nube de puntos tridimensionales a partir del par estereoscópico. Además el espacio de trabajo en este contexto puede ir desde unos pocos metros de profundidad hasta cientos de metros. En la presente investigación existe una restricción en el espacio de trabajo, el cual usualmente involucra un volumen de un metro cúbico.

Este volumen de trabajo restringido reduce el riesgo de obtener una no convergencia con algún par de puntos con disparidad cero (en el infinito). Además, esto se debe a que el objetivo no es triangular todos los puntos de la escena, sino un subconjunto de puntos en la escena. Este subconjunto de puntos fácilmente identificables puede ser las esquinas de los marcadores. Un elemento determinante es que ninguno de los marcadores estará a una distancia mayor de un metro de las cámaras.

En virtud de las conclusiones obtenidas por los autores (R. I. HARTLEY y STURM, 1997) y (VITE-SILVA; CRUZ-CORTÉS; TOSCANO-PULIDO y FRAGA, 2007), el algoritmo iterativo-LS es una opción elegible para realizar la triangulación en la presente investigación. Esta decisión se debe a que este arroja buenos resultados en ambientes reales donde existe ruido en las imágenes y a las restricciones que impone el contexto de trabajo del neuronavegador. Luego de la triangulación es necesario hallar la transformación rígida correspondiente al conjunto de puntos detectados.

## 1.8. Transformación rígida

El proceso final del seguimiento estereoscópico es el cálculo de la transformación rígida del conjunto de puntos identificados, a partir de sus coordenadas. Luego de realizarse la triangulación de las esquinas de los marcadores en las imágenes, se tiene un conjunto de puntos en el espacio. Estos puntos de forma general están asociados a un marcador que contiene un patrón identificable (sección 1.4). Como el objetivo de un componente de seguimiento es brindar posición y orientación, no basta con conocer la posición de estos marcadores, es decir, las coordenadas  $(x, y, z)$  de cada punto identificado. Es necesario obtener una matriz de transformación que exprese la rotación y traslación del conjunto de puntos asociado a cada marcador. Esta matriz contendrá la información espacial de los marcadores con respecto al sistema de referencia estereoscópico.

Se le llama transformación rígida a la matriz que es capaz de alinear dos conjuntos de puntos en un espacio dado, donde sus pares correspondientes han sido determinados. Se llama *rígida* porque sólo aplicando rotaciones y traslaciones se puede llegar desde una posición inicial hasta una final.

Existen varios algoritmos que calculan los componentes de rotación y traslación de una matriz de transformación. En el trabajo de (LORUSSO; EGGERT y FISHER, 1995) se comparan cuatro algoritmos muy populares que le dan solución al problema. Esos difieren en términos de representación de la transformación y método de solución. En dicho trabajo se utiliza respectivamente los métodos [Descomposición en valores singulares de una matriz \(SVD\)](#), matrices ortogonales (OM), *quaternions* (UQ) y *dual quaternions* (DQ). La comparación presenta los resultados de varios experimentos diseñados para determinar la exactitud en presencia de ruido, estabilidad (*respecto a juegos de datos degenerados*) y tiempo de cómputo.

La comparación realizada por ([ibíd.](#)) determinó que el [SVD](#) provee el mejor promedio de exactitud y estabilidad, pero no es el más eficiente para juegos de datos con grandes cantidades de puntos. El algoritmo que usa *quaternions* es ligeramente más rápido pero está en desventaja en los otros aspectos. Como los conjuntos de puntos que se utilizan en la presente investigación no son grandes, la mejor opción es usar el SVD que provee mejor exactitud y estabilidad. El algoritmo de Kabsch usa la descomposición SVD y es el elegido para la propuesta de solución (KABSCH, 1978).

En álgebra lineal, la descomposición en valores singulares de una matriz real o compleja es una factorización de la propia matriz. Esto tiene muchas aplicaciones en estadística y otras disciplinas. Según (MARTÍNEZ FERNÁNDEZ DE LAS HERAS, 2005), una descomposición SVD de la matriz compleja  $M$  de dimensiones  $m \times n$  es una factorización de la forma  $M = UWV^T$ , donde  $U$  es una matriz unitaria  $m \times m$ ,  $W$  es una matriz rectangular  $m \times n$  con números no negativos en su diagonal, y  $V^T$  (la transpuesta de  $V$ ) es una matriz unitaria  $n \times n$ . Los valores de la diagonal de  $W$  son conocidos como los valores singulares de  $M$ .

## 1.9. Proceso de calibración

La calibración es esencial en cualquier aplicación de visión por computadoras. Su finalidad radica en la obtención de los coeficientes de distorsiones y el establecimiento de la geometría epipolar del sistema estereoscópico. Mediante la calibración se calculan los parámetros intrínsecos y extrínsecos de las cámaras, a partir de un conjunto de puntos de control. Los puntos de control tienen coordenadas tridimensionales bien conocidas, se miden las coordenadas correspondientes en la imagen obtenida con la cámara y finalmente se obtienen los parámetros intrínsecos a partir de sus correspondencias (FONSECA, 2011) y (SAÍZ, 2010).

### Parámetros intrínsecos

- *Distancia focal:  $f$*
- *Desplazamiento del centro de la imagen:  $(C_x, C_y)$*
- *Coficiente de distorsión radial:  $K_1, K_2$*

Cuando se realiza la calibración, se obtiene una matriz intrínseca correspondiente a la cámara con la siguiente distribución:

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (1.9.1)$$

Donde:

- $f_x$  y  $f_y$ : denotan la distancia focal en la dirección  $x$  y la dirección  $y$  respectivamente.
- $c_x$  y  $c_y$ : denotan el punto principal que idealmente debería ubicarse en el centro de la imagen.

En ocasiones estos parámetros son suministrados directamente por el fabricante de la cámara, aunque existen técnicas consolidadas para su determinación. El método de Zhang es el más extendido y se encuentra implementado en la biblioteca de visión por computadoras OpenCV. Esta matriz resultante del proceso de calibración representa la proyección que utiliza la cámara para generar la imagen del mundo real.

En la figura 1.9 se muestran algunas de las representaciones de los parámetros de la cámara más importantes que intervienen en el proceso de adquisición tridimensional (FONSECA, 2011).

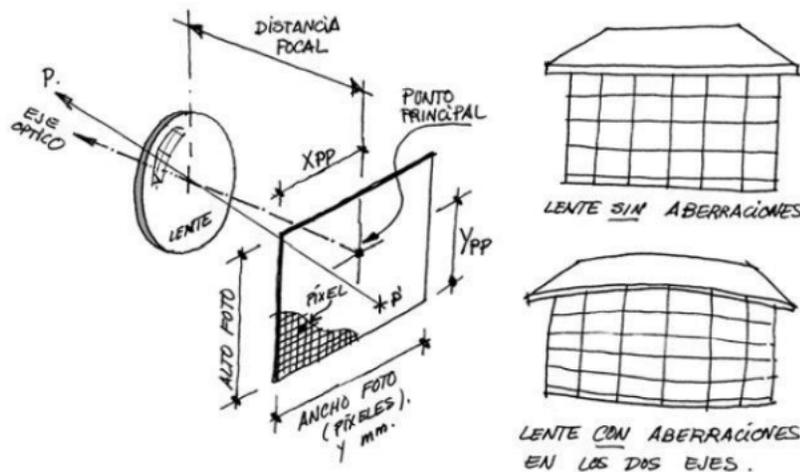


Figura 1.9. Parámetros de calibración de una cámara (FONSECA, 2011).

### Distorsiones

La distorsión geométrica existe por la imposibilidad de construir una cámara ideal. Existen dos tipos de distorsiones, la tangencial debido a la no perpendicularidad del eje óptico del lente con el plano de la imagen, y la radial, que hace que los píxeles se alejen (distorsión de barril) o se acerquen (distorsión de alfiletero, *pincushion distortion*) del centro de la imagen en una cantidad proporcional a la distancia radial. Con la distorsión radial las líneas rectas aparecen curvadas. La distorsión de ojo de pescado se manifiesta en cámaras de un lente más ancho donde se pueden tomar imágenes de hasta 180° de un lado a otro (SZELISKI, 2010).



Figura 1.10. Distorsiones geométricas (SZELISKI, 2010).

### Parámetros extrínsecos

Los parámetros extrínsecos definen la posición y orientación de la cámara con respecto al sistema de referencia de la escena: traslación  $T_{ij}$  y rotación  $R_j$ . Existen muchas técnicas para obtener los parámetros extrínsecos de una cámara, aunque los más habituales son los que utilizan conocimiento específico acerca de la escena (FONSECA, 2011).

Las etapas que sigue el proceso de calibración para llegar a determinar dichos parámetros son:

- *Obtención del modelo matemático.*
- *Obtención de los datos de campo.* En esta etapa se obtienen los puntos 2D de la imagen que provienen de los puntos tridimensionales de las escenas.
- *Determinación de los parámetros tanto intrínsecos como extrínsecos de la cámara.* Una vez conocidos todos los puntos de la escena real y la imagen capturada por la cámara, se procede a resolver las ecuaciones que rigen el modelo que mejor se aproxima a la cámara.

Hasta aquí se describieron los elementos teóricos necesarios para comprender el objeto de estudio de la presente investigación. En la próxima sección se describen las herramientas, tecnologías y lenguajes de programación para llevar a la práctica los elementos teóricos identificados en este capítulo.

## 1.10. Marco de trabajo

Para el desarrollo de un software es necesario definir un conjunto de herramientas, dentro de las que se encuentran los lenguajes de programación, bibliotecas y herramientas auxiliares para la implementación del software. Debido a que existe un prototipo de neuronavegador previamente implementado con el lenguaje C++, este es el lenguaje que se selecciona para implementar la solución de la presente investigación. Además de producir un código optimizado que aumenta el rendimiento de las aplicaciones.

Para la implementación de los algoritmos de visión por computadoras se seleccionó la biblioteca OpenCV. Es una biblioteca de código abierto con licencia BSD escrita en C y C++. Es compatible con Linux, Windows y Mac OS X. OpenCV fue diseñada para eficiencia computacional y aplicaciones de tiempo real. Puede aprovechar los procesadores multinúcleos y las funciones de tarjetas gráficas. Posee toda una infraestructura de visión por computadora con más de 500 funciones incluidas y se perfila como un estándar en el campo de la visión por computadoras (BRADSKI y KAEHLER, 2008).

Para la implementación de las interfaces de usuario se seleccionó el marco de trabajo de Qt en su versión 4.8, porque además de las ventajas que ofrece (descritas en [Qt Cross-platform application s.f.](#)), es el utilizado en el centro VERTEX para el desarrollo de interfaces gráficas de usuario. El [Entorno de desarrollo Integrado \(IDE\)](#) seleccionado fue el Qt Creator por la integración que tiene con el marco de trabajo de Qt.

Para el desarrollo de las funcionalidades relacionadas con la navegación guiada por imágenes se utilizó la [IGSTK](#). Es una biblioteca de C++ de código abierto, desarrollada sobre VTK e ITK que provee los componentes necesarios para el desarrollo de aplicaciones de cirugía guiada por imágenes. Es multiplataforma y usa CMake para gestionar el proceso de compilación. IGSTK soporta la mayoría de los dispositivos de seguimiento existentes en el mercado y provee una vía para extender el soporte a otros tipos de componentes de seguimiento (ENQUOBAHRIE y Ziv YANIV, 2009). Es la biblioteca utilizada para realizar el prototipo existente (MARTÍNEZ, 2012).

Para la detección de los puntos de interés se utilizó la biblioteca ArUco. Esta es una biblioteca minimalista para aplicaciones de [RA](#) basada en OpenCV. Es libre, multiplataforma y provee un conjunto de funcionalidades para detectar patrones conocidos y su posición espacial utilizando una cámara. Los marcadores son cuadrados para facilitar el proceso de detección, además cuentan con un código binario para su identificación que permite el uso de varios de ellos en la escena. Las funcionalidades de detección constituyen el punto de

partida del sistema estereoscópico.

Se seleccionó CMake como herramienta de autoconstrucción, para la simplificación el proceso de compilación y el desarrollo de proyectos en diferentes plataformas de forma transparente. CMake automatiza la generación de archivos para la compilación como *Makefiles* y genera configuraciones para varias plataformas. Es flexible y extensible, cuenta con módulos para OpenCV, IGSTK y Qt (*CMake s.f.*). Además el IDE seleccionado para la implementación de la solución brinda soporte para la compilación con CMake.

### 1.11. Conclusiones parciales

En el presente capítulo se analizaron distintos dispositivos de seguimiento utilizados en los neuronavegadores. Se identificaron los dispositivos ópticos basados en grabación de video de luz visible como la solución más factible para la implementación de la solución por la exactitud que se puede lograr, incluso utilizando cámaras de bajo costo. Se seleccionaron las técnicas de seguimiento visual de objetos basadas en detección de marcadores cuadrados porque brinda una estrategia de seguimiento robusta y eficiente. Se identificó el algoritmo de triangulación iterativo-LS por su buen desempeño en ambientes donde existe distorsión y ruido, además de tener un bajo costo computacional. Se determinó que el algoritmo basado en la descomposición SVD ofrece la mejor opción para hallar la transformación rígida entre los conjuntos de puntos generados en la detección de los marcadores.

---

### Propuesta de solución

---

En este capítulo se describe la solución propuesta en dos etapas. Inicialmente se describe el método de calibración y se explican los pasos para calibrar el sistema de visión estereoscópica. Seguidamente se describe el sistema estereoscópico propuesto, su utilización y las etapas del el componente de seguimiento. Finalmente se describe su integración con la biblioteca IGSTK.

### 2.1. Solución técnica

Teniendo en cuenta los trabajos descritos en la sección 1.4, el seguimiento óptico basado en grabación de video constituye una de las alternativas factibles a los sistemas comerciales existentes en el mercado. Se identificó incluso que las estrategias de seguimiento basadas en dispositivos ópticos se comercializan por fabricantes como *Claron Technology Inc. s.f.* y (*Northern Digital s.f.*). Además, la existencia de soluciones alternativas como las propuestas por (GÓMEZ y RANDALL, 2007) y (NICOLAU; GOFFIN y SOLER, 2005) demuestran la factibilidad de construir sistemas de seguimiento estereoscópicos con cámaras de bajo costo, con resultados cercanos a los ofrecidos por las soluciones comerciales.

Debido a estas razones se decidió implementar el sistema de seguimiento óptico basado en grabación de video en su variante estereoscópica. Para la implementación de la solución se decidió utilizar dos *webcams* para la adquisición de las imágenes. Las etapas que definen la solución son las siguientes:

- *Adquisición.* En esta etapa se adquieren las imágenes de ambas cámaras. La adquisición es una etapa requerida para la calibración y la ejecución del sistema de seguimiento.

- *Calibración.* Etapa previa a la ejecución del seguimiento. En esta etapa se realizan las operaciones necesarias para obtener el modelo geométrico de la cámara: distancia focal, coeficientes de distorsión de las cámaras información extrínseca de las cámaras para conformar el sistema estereoscópico.
- *Rectificación.* En esta etapa se realiza el proceso de rectificación de las imágenes provenientes del sistema estereoscópico (ver sección 1.7.2).
- *Detección de marcadores.* En esta etapa se buscan e identifican los patrones en cada imagen correspondientes a los marcadores distribuidos en la escena. Los marcadores definidos son cuadrados en blanco y negro al estilo de la figura 1.4. La biblioteca ArUco implementa funcionalidades de detección de estos marcadores. Estos son colocados en la herramienta quirúrgica como muestra la figura 3.3.
- *Triangulación.* Una vez que se tienen las posiciones del marcador en cada una de las imágenes (sus respectivas proyecciones), se calculan las posiciones de sus cuatro esquinas en el espacio tridimensional mediante triangulación. Estas posiciones son relativas al eje de coordenadas del sistema estereoscópico (ver sección 1.7.2). El algoritmo utilizado para triangular los puntos es el iterativo-LS (ver epígrafe 1.7.5).
- *Estimación de la transformación.* Por último se halla la matriz de transformación. Es necesario conocer no solo la posición espacial de los marcadores, sino su orientación con respecto a la geometría de las cámaras. Para esto se define una posición canónica del marcador en el centro de coordenadas del sistema, y se calcula la transformación rígida entre los puntos espaciales del marcador hallados con la triangulación y los puntos canónicos (ver sección 1.8).

Estas etapas deben ejecutarse constantemente para hacer el seguimiento continuo a partir del flujo de video, exceptuando la calibración. La calibración se realiza una sola vez y luego la información generada en esa etapa se utiliza en las demás. Es por eso que la calibración es un procedimiento *offline* que se realiza en etapas previas en las aplicaciones de visión por computadoras.

## 2.2. Dispositivo de seguimiento

El componente desarrollado utiliza dos cámaras de bajo costo fabricadas por la compañía (*Logitech International S.A. s.f.*). Las cámaras cuentan con un cable USB 2.0 para la conexión. La resolución es de  $640 \times 480$  píxeles y el zoom se puede ajustar manualmente. Estas cámaras cuentan con una frecuencia de captura estándar de 30 fotogramas por segundo. El dispositivo propuesto como solución está basado en la descripción que aparece en el diagrama de la figura 1.5, el cual queda finalmente como se aprecia en la figura 2.2.

Las variables de la figura 2.2 son:

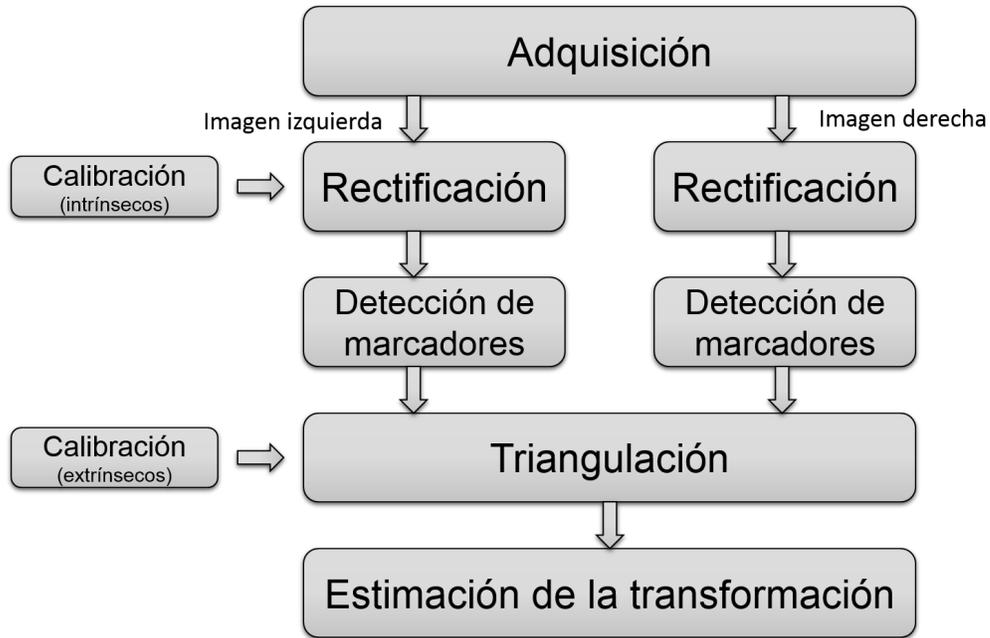


Figura 2.1. Flujo del sistema de seguimiento estereoscópico propuesto.

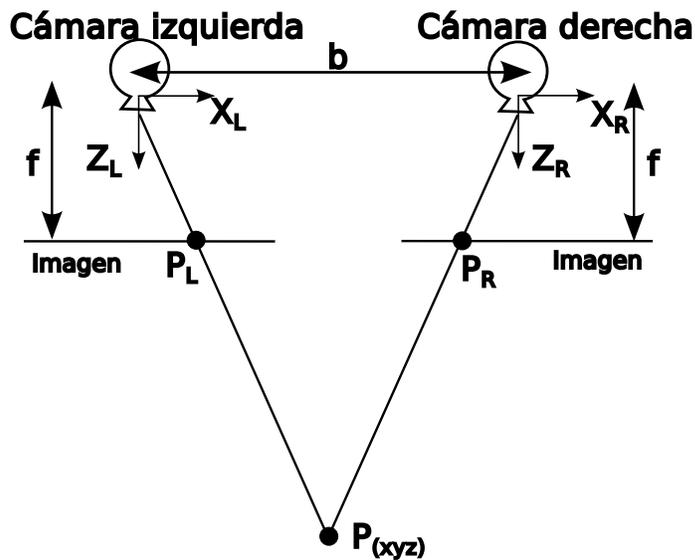


Figura 2.2. Diagrama del dispositivo propuesto.

1.  $b$ : es la línea base o distancia entre las dos cámaras
2.  $f$ : es la distancia focal de la cámara

3.  $X_L$  y  $X_R$ : son las abscisas X del sistema de referencia local de las cámaras izquierda y derecha respectivamente
4.  $Z_L$  y  $Z_R$ : son los ejes ópticos de las cámaras izquierda y derecha respectivamente
5.  $P$ : es un punto en el espacio del mundo real definido por las coordenadas  $(X, Y, Z)$  respecto al sistema de referencia del dispositivo de seguimiento
6.  $P_L$ : es la proyección del punto  $P$  del mundo real en la imagen adquirida por la cámara izquierda
7.  $P_R$ : es la proyección del punto  $P$  del mundo real en la imagen adquirida por la cámara derecha

El flujo del sistema del dispositivo de seguimiento puede apreciarse en la figura 2.1, la cual muestra un diagrama en bloques con las etapas descritas en la sección 2.1.

### 2.3. Adquisición de imágenes

Durante la etapa de adquisición de las imágenes la posición de las cámaras es fundamental. Para la solución propuesta, las cámaras se colocan una al lado de la otra, con una separación de aproximadamente 100 milímetros y sus respectivos ejes ópticos paralelos. De esta forma los ejes ópticos quedan perpendiculares a la línea base (ver sección 1.7.2). Esta separación puede variar en dependencia de la distancia promedio a la que se detectarán los marcadores. Con esta separación se puede realizar el seguimiento a una distancia promedio de 30cm a 3m. Sin embargo, WIDMANN; STOFFNER y BALE, 2009 sugiere que el volumen de trabajo para el seguimiento en la navegación guiada por imágenes, sea aproximadamente de  $1 m^3$ .

Con esta configuración, se hace una aproximación inicial a las restricciones para la correspondencia y se obtiene una disparidad horizontal. Es válido destacar que los ejes ópticos de las cámaras no siempre van a resultar estrictamente paralelos y que siempre va a existir una rotación de las cámaras con respecto a sus respectivos ejes ópticos. Sin embargo, cuando se realiza la configuración de la disposición de las cámaras se deben minimizar estas rotaciones y maximizar la condición paralela de los ejes ópticos. Si se tienen en cuenta estas condiciones iniciales se obtendrá una mejor aproximación a los parámetros intrínsecos y extrínsecos de las cámaras como resultado del proceso de calibración. De esta forma serán corregidos los pequeños errores inevitables.

Otro punto a tener en cuenta es la *sincronización de las cámaras*. En el seguimiento estereoscópico es una condición necesaria que las dos cámaras que filman la escena tomen una imagen simultánea, donde sólo varía el punto de visión desde el que se capturó la escena. Si se diera el caso de un lapso de tiempo entre la toma de la imagen de cada una de las cámaras, aumenta la probabilidad de que no capten la misma escena cuando hay objetos en movimiento.

Esta restricción usualmente se resuelve con una sincronización por *hardware* en las cámaras, para asegurar que se toman dos imágenes de la misma escena en el mismo instante de tiempo. Las cámaras utilizadas (*webcams*) en la presente investigación no cuentan con la posibilidad de sincronización por *hardware*. Por tal motivo es muy difícil que las imágenes sean tomadas en el mismo instante de tiempo. Una estrategia para reducir este efecto negativo es que las peticiones y los fotogramas lleguen de forma concurrente, para eso las cámaras deben contar con su propio bus de conexión [USB](#).

En la solución propuesta la captura de las imágenes se realizó mediante la clase *cv::VideoCapture*, para conectar las cámaras y obtener los fotogramas. La llamada a *VideoCapture::read* encapsula llamadas a las funciones *VideoCapture::grab* (provoca la señal de captura en el dispositivo) y *VideoCapture::retrieve* (recupera la imagen del dispositivo). La llamada a *VideoCapture::retrieve* es un proceso lento porque se debe decodificar la imagen desde el dispositivo para ponerla a disposición de la biblioteca, esto implica que exista un tiempo innecesario hasta la llamada a *VideoCapture::read* de la siguiente cámara.

Para reducir el tiempo de captura entre el par de imágenes, se utilizaron las funciones *VideoCapture::grab* y *VideoCapture::retrieve* en vez de *VideoCapture::read*. Esta es una buena estrategia cuando las cámaras no cuentan con sincronización por *hardware*. Primero se llama la función *grab* para cada cámara, y luego el método más lento *retrieve* que decodifica la imagen. De esta forma ambos fotogramas estarán más cercanos en el tiempo. Luego de la adquisición de las imágenes, se puede pasar a las siguientes etapas de la solución propuesta.

## 2.4. Calibración estereoscópica

La calibración estereoscópica es un procedimiento parecido a la calibración de un sistema monocular (una cámara). Se necesita hacer varias tomas con ambas cámaras de un objeto conocido, el más utilizado y difundido es una rejilla de cuadrados alternados blancos y negros, que es llamado usualmente tablero de ajedrez, este no tiene que tener necesariamente 8 filas y 8 columnas ni ser cuadrado, pero mientras más esquinas posea, se comparan más puntos y se puede llegar a calibraciones más exactas. Según (BRADSKI y KAEHLER, 2008) cualquier tablero asimétrico mayor de 3x3 es válido. Este método es conocido como el método de (ZHANG, 2000) en la literatura y es el implementado en la biblioteca OpenCV.

Para calibrar el sistema se construyó una herramienta de calibración estereoscópica con interfaz visual (ver figura 2.3). La biblioteca de visión por computadora OpenCV contiene funciones para calibrar estos sistemas como *cv::stereoCalibrate* y *cv::stereoRectify*. Existe otra función que es utilizada para hallar las matrices de calibración de un sistema estereoscópico, *cv::stereoRectifyUncalibrated*, pero ésta no utiliza los parámetros intrínsecos de las cámaras y su posición relativa en el espacio, por lo que se optó por utilizar *cv::stereoRectify* en su lugar. Al inicio de la calibración, se deben realizar tomas ubicando el tablero en varias poses para

## 2.4. CALIBRACIÓN ESTEREOSCÓPICA

obtener varias perspectivas y con condiciones lumínicas lo más parecidas posibles. La iluminación global afecta a ambas cámaras por igual. Todas las esquinas deben ser detectadas en cada iteración, para esto se usó el detector de Harris implementado en OpenCV.

Además de buscar los parámetros intrínsecos y extrínsecos de las cámaras, la herramienta encuentra la matriz de rotación  $R$  y el vector de traslación  $T$  de la cámara derecha con respecto a la izquierda. Las matrices esencial y fundamental (ver sección 1.7.2) también son obtenidas, además de las nuevas matrices de reproyección para cada cámara y la matriz de mapeo de disparidad.

Para la solución se utilizó un tablero de  $9 \times 6$  filas y columnas. Con la herramienta se puede definir el tamaño de la casilla, y todas las medidas calculadas corresponderán a esa métrica, es decir, si se determina que el lado de la casilla tiene  $10\text{ cm}$ , entonces todas las unidades estarán en centímetros. Además se agregó la funcionalidad de mostrar el mapa de disparidad y modificar un conjunto de parámetros que intervienen en su representación, con el objetivo de visualizar la calidad de la calibración. Por último se agregaron las funcionalidades de guardar y cargar toda la configuración de la calibración en formato [Extensible Markup Language \(XML\)](#) o [Formato de serialización de datos legible al humano \(YML\)](#).

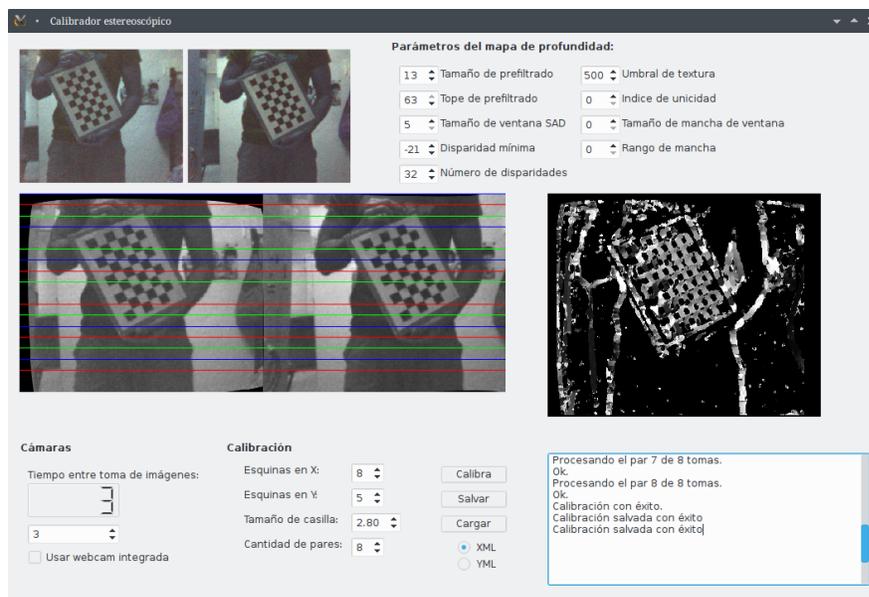


Figura 2.3. Herramienta para la calibración estereoscópica desarrollada.

### 2.4.1. Distribución del archivo de calibración

La herramienta de calibración genera un archivo en formato XML ó YML, a decisión del usuario, con toda la información de calibración de las cámaras. Las etiquetas utilizadas aparecen en el siguiente orden en el archivo de calibración:

1. *frame\_size*: resolución de la imagen.
2. *intrinsics\_left*: matriz intrínseca de la cámara izquierda.
3. *distortion\_left*: coeficientes de distorsión de la cámara izquierda.
4. *intrinsics\_right*: matriz intrínseca de la cámara derecha.
5. *distortion\_right*: coeficientes de distorsión de la cámara derecha.
6. *rotation\_matrix*: matriz de rotación de la cámara derecha respecto a la izquierda.
7. *rotation\_matrix1*: matriz de rotación de la cámara izquierda.
8. *rotation\_matrix2*: matriz de rotación de la cámara derecha.
9. *traslation\_matrix*: matriz de traslación de la cámara derecha respecto a la izquierda.
10. *essential\_matrix*: matriz esencial.
11. *fundamental\_matrix*: matriz fundamental.
12. *P1*: nueva matriz de proyección de la cámara izquierda después de la rectificación.
13. *P2*: nueva matriz de proyección de la cámara derecha después de la rectificación.
14. *Q*: matriz  $4 \times 4$  utilizada para mapear la disparidad y profundidad.
15. *mx1*: mapa para la rectificación de la coordenada  $x$  de la primera imagen.
16. *my1*: mapa para la rectificación de la coordenada  $y$  de la primera imagen.
17. *mx2*: mapa para la rectificación de la coordenada  $x$  de la segunda imagen.
18. *my2*: mapa para la rectificación de la coordenada  $y$  de la segunda imagen.

Cuando se tiene el sistema calibrado no se puede cambiar su configuración, pues las matrices que contienen la información geométrica ya no serían válidas. En ese caso se debe recalibrar todo el sistema. Luego de calibrar el sistema se procede a rectificar las imágenes.

## 2.5. Rectificación de las imágenes estereoscópicas

El objetivo de la rectificación es mantener en una línea epipolar entre los puntos homólogos de cada imagen. Se utilizó la función *cv::initUndistortRectifyMap*, la cual es capaz de hallar una matriz de igual tamaño que

las imágenes a rectificar. Esta matriz es capaz de mapear cada píxel a su nueva posición rectificadora y sin distorsión. Esta función devuelve una nueva imagen rectificadora a partir de la anterior, como si se hubiese tomado con una cámara sin distorsiones.

Estos mapas son utilizados por la función `cv::remap` para rectificar cada imagen adquirida. En la herramienta de calibración implementada se muestran las imágenes rectificadas unidas horizontalmente con líneas epipolares dibujadas (ver figura 2.4). Con las imágenes rectificadas se procede al siguiente paso que es la detección de marcadores.

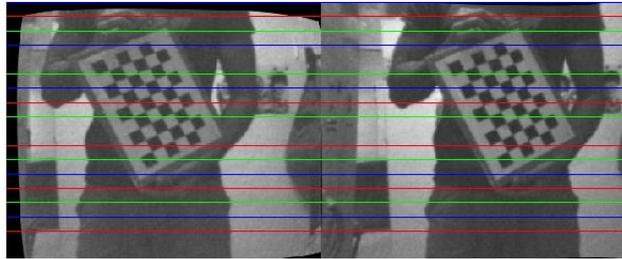


Figura 2.4. Rectificación utilizando OpenCV.

## 2.6. Detección del marcador

La biblioteca ArUco (AVA, s.f.) desarrollada por (AVA s.f.) tiene implementado un algoritmo robusto y fiable de detección de marcadores extendidos o fiduciales mediante un sistema monocular. Este utiliza marcadores de forma cuadrada, que han sido ampliamente difundidos por la facilidad de detección de su forma y sus cuatro esquinas. Los marcadores de ArUco poseen una codificación en su interior que además de servir para hallar la rotación de forma robusta frente a oclusiones, posibilita su identificación, por lo que se pueden utilizar varios en una escena (GARRIDO, 2014).

### 2.6.1. Obtención de los puntos

La detección de marcadores con ArUco se realiza utilizando algunos algoritmos implementados en OpenCV. Estos son los pasos fundamentales que realiza esta biblioteca (BAGGIO, 2012):

1. *Adquisición de la imagen* que contiene los marcadores desde el dispositivo de captura (figura 2.5a).
2. *Aplicación de un filtrado de umbral adaptativo* a la imagen original para binarizarla y eliminar los gradientes para el paso siguiente (figura 2.5b).

3. *Aplicación de un filtrado de detección de aristas*, con el que se detectan los contornos convexos en la imagen binarizada (figura 2.5c).

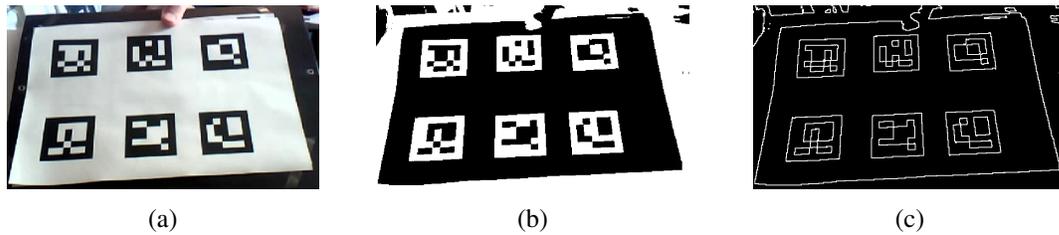


Figura 2.5. Proceso de detección de marcadores con ArUco: a) imagen original, b) aplicación de un filtro para llevar a blancos y negros la imagen (umbral adaptativo), c) detección de contornos (BAGGIO, 2012).

4. *Análisis de las regiones cuadradas* que pueden ser marcadores potenciales. Para esto se le aplican filtros a los contornos, con estos se eliminan los contornos que no tienen cuatro aristas y los que tienen longitudes de aristas menores que las esperadas.
5. *Eliminación de la proyección perspectiva* para identificar el marcador en su forma canónica.

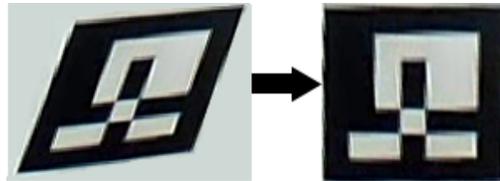


Figura 2.6. Eliminación de la proyección perspectiva.

6. *Se realiza la decodificación y se analiza el código interno* del marcador con una máscara de bits. Si el código es válido se considera que el rectángulo es un marcador.
7. Por último se *refinan las esquinas* para obtener su ubicación con una exactitud de subpíxel, para esto se utiliza la función `cv::cornerSubPix` de OpenCV.

### Codificación del marcador

La codificación del interior del marcador permite el uso de varios marcadores de forma simultánea en la escena, es una versión modificada del código de [Hamming](#). La codificación del patrón está definida por un total de 25 bits divididos en 5 filas de 5 bits cada una, así se obtienen 5 palabras de 5 bits. Cada palabra contiene solo 2 bits de información, el resto es para la detección y corrección de errores. Los 10 bits de información real permite obtener  $2^{10} = 1024$  marcadores diferentes.

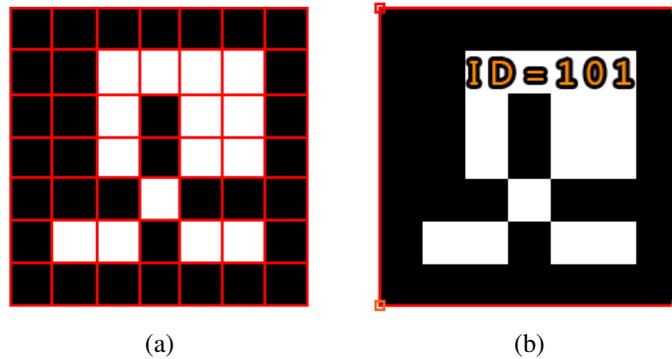


Figura 2.7. Codificación de los marcadores según ArUco: a) diseño del marcador, b) identificación del marcador.

## 2.7. Triangulación

Luego de la detección de los marcadores en las imágenes se tienen los puntos asociados a sus esquinas (figura 2.7b). Seguidamente se hallan las coordenadas espaciales correspondientes a estos puntos. Como se explicó en la sección 1.7.5, se implementó el algoritmo iterativo-LS, al cual se le pasan dos puntos en el espacio bidimensional y devuelve la información correspondiente al punto en el espacio tridimensional. Los puntos en el espacio bidimensional son las coordenadas del punto en cuestión en cada una de las imágenes del par estereoscópico.

OpenCV también provee un método (*cv::triangulatePoints*) para hacer la triangulación, aunque no es el más exacto según los debates de varios investigadores en los propios foros de OpenCV. No obstante se utilizaron ambos métodos y se compararon sus resultados. El proceso de triangulación se realizó para cada una de las esquinas de los marcadores detectados, para luego obtener la transformación de cada uno de los marcadores.

## 2.8. Obtención de la matriz de transformación

Una vez realizada la triangulación, es necesario hallar la matriz de transformación asociada a cada marcador con respecto al sistema de coordenadas de las cámaras (ver sección 1.8). Una de las ventajas de utilizar la visión estereoscópica es que se pueden hallar directamente los puntos 3D, a diferencia de cuando se utiliza solo una cámara. En el caso monocular hay que aplicar ciertos algoritmos para buscar la matriz de transformación a partir de un conjunto de puntos 2D (la proyección) y un conjunto inicial 3D definidos de forma estática. La exactitud se puede ver afectada, puesto que una figura puede tener varias proyecciones perspectivas. Al contrario, si se cuenta con los puntos 3D, el objetivo es hallar una matriz de transformación que sea capaz de alinear un conjunto de puntos  $P_1 \in \mathbf{R}^3$  en otro conjunto  $P_2 \in \mathbf{R}^3$ .

### 2.8.1. Cálculo de la transformación rígida

En el trabajo (SORKINE, 2009) se explica el algoritmo para hallar la transformación rígida que alinea dos conjuntos de puntos utilizando la descomposición SVD. en esta sección se describe el procedimiento para calcular la transformación existente entre los dos conjuntos de puntos.

Dados dos conjuntos de puntos correspondientes  $P = \{p_1, p_2, \dots, p_n\}$  y  $Q = \{q_1, q_2, \dots, q_n\}$  del espacio  $\mathbf{R}^d$  se desea encontrar la transformación rígida que alinea a ambos conjuntos minimizando la función:

$$(R, t) = \sum_{i=1}^n \|(Rp_i + t) - q_i\|^2 \quad (2.8.1)$$

Para dar solución al problema de encontrar la transformación, expresado en la ecuación 2.8.1 se diseñó un algoritmo que cuenta con los siguientes pasos:

1. *Se calculan los centroides* de cada conjunto de puntos. Se decide establecer este centroide como el origen del sistema de referencia local del conjunto de los puntos. De esta forma la transformación se calcula utilizando ese origen. Esta operación elimina la traslación de forma tal que reduce el problema a encontrar la rotación existente entre ambos conjuntos de puntos respecto al centroide.

$$\bar{p} = \frac{\sum_{i=1}^n p_i}{n} \quad \bar{q} = \frac{\sum_{i=1}^n q_i}{n} \quad (2.8.2)$$

2. *Se calculan los vectores centrados.* Se hace una conversión de las coordenadas de los puntos al sistema de referencia local.

$$x_i = p_i - \bar{p}, \quad y_i = q_i - \bar{q}, \quad i = 1, 2, \dots, n. \quad (2.8.3)$$

3. *Se calcula la matriz de covarianza  $d \times d$ .*

$$S = XY^T, \quad (2.8.4)$$

donde X y Y son matrices  $d \times n$  que tienen a  $x_i$  y a  $y_i$  como sus columnas respectivamente.

4. *Se realiza la descomposición de valores singulares  $S = UWV^T$ .* A partir de la descomposición la matriz de rotación sería  $R = UV^T$ . No obstante la matriz de rotación resultante R podría contener potencialmente reflexiones, lo cual es posible conocer si el determinante  $\det(VU^T) = -1$ . Para resolver este problema se introduce una matriz auxiliar D que elimina posibles reflexiones de la rotación.

Luego el cálculo de la matriz de rotación  $R$  con la nueva matriz  $D$  se resuelve con la ecuación 2.8.5:

$$R = VDU^T, \quad D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(VU^T) \end{bmatrix} \quad (2.8.5)$$

5. Finalmente se *calcula la traslación*  $t$  de los puntos.

$$t = \bar{q} - R\bar{p} \quad (2.8.6)$$

Luego de definir la forma en que se calcula la matriz de transformación se procede al diseño e implementación del componente.

## 2.9. Arquitectura y diseño

En esta sección se describen la arquitectura utilizada y el diagrama de las clases que se diseñó, como los elementos fundamentales de la arquitectura y diseño de la solución propuesta.

### 2.9.1. Arquitectura por capas

Una de las técnicas más comunes utilizadas por los arquitectos de software es el patrón por capas. Un sistema en capas permite establecer responsabilidades bien definidas para cada capa. Un estilo común es que las capas inferiores ofrecen visibilidad de sus interfaces a las capas superiores y actúan como una interfaz que encapsula a las capas que le anteceden. Este estilo arquitectónico minimiza las dependencias entre capas, estandariza los servicios y permite la reutilización de funcionalidades (FREEMAN; ROBSON; BATES y SIERRA, 2004). La arquitectura de la solución propuesta se basa en este estilo debido a que está pensada para aplicaciones que utilizan IGSTK como biblioteca de desarrollo. IGSTK propone una arquitectura por capas como se muestra en la figura 2.8 (ENQUOBAHRIE y Ziv YANIV, 2009).

### 2.9.2. Integración del componente de seguimiento con IGSTK

El componente de seguimiento diseñado es compatible con la biblioteca IGSTK. La integración con la biblioteca se realiza al heredar de dos clases: `igstk::Tracker` que es una clase abstracta que actúa como interfaz para diferentes modalidades de seguimiento. La otra clase es `igstk::TrackerTool` que actúa como una clase abstracta de los objetos seguidos o sensores como también se les llama. En el caso de la solución propuesta

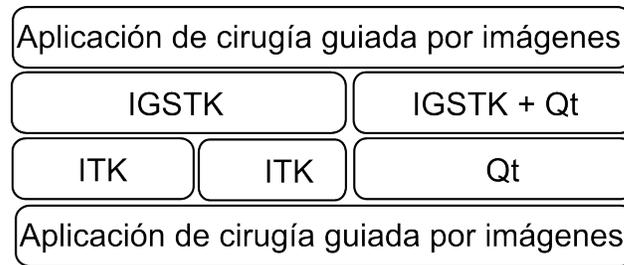


Figura 2.8. Arquitectura por capas de IGSTK (ENQUOBAHRIE y Ziv YANIV, 2009).

los objetos son los marcadores. Este componente además de IGSTK, depende de la biblioteca ArUco para la detección de los marcadores para la detección de los marcadores y esta a su vez se basa en OpenCV para el procesamiento de las imágenes.

Las clases *igstk::Tracker* e *igstk::TrackerTool* son dos componentes fundamentales de IGSTK para utilizar dispositivos de seguimiento. Su diseño se basa en el patrón de diseño *Fábrica abstracta* para crear familias de componentes de seguimiento. Además, sus funcionalidades abarcan el establecimiento, mantenimiento y cierre la comunicación con los dispositivos de seguimiento; adquirir la información del seguimiento, y verificar la validez del dispositivo en todo momento.

Los componentes de IGSTK funcionan como un sistema síncrono, cada componente se ejecuta según su turno. Debido a esto, no existe garantía de que el procesamiento y la adquisición de una posición sean completados antes de que se haga la siguiente medición de posición. Además no se garantiza que la frecuencia del *Tracker* sea similar a la del componente de visualización de imágenes anatómicas del paciente. Por otra parte, cuando un componente envía un evento a otro componente, el segundo componente controla la ejecución del programa hasta que cede el control nuevamente al primero.

Para solucionar esta situación el componente *igstk::Tracker* utiliza un hilo independiente para la comunicación con el dispositivo de seguimiento. Esto evita que el componente tenga que esperar a que se le pase el control de la aplicación para obtener los datos de posición y orientación de las herramientas. Este hilo independiente se utiliza para mantener la comunicación con el dispositivo de seguimiento y almacenar en un *buffer* los datos de medición de las herramientas. Cuando el hilo principal de la aplicación le solicita los datos de medición de las herramientas al componente *Tracker*, este le devuelve las mediciones más recientes del *buffer*. De esta manera se garantiza que no haya retardos en el proceso de obtención de los datos de medición, y que se mantenga una comunicación continua con el dispositivo de seguimiento. La figura 2.9b ilustra el funcionamiento del componente *Tracker* de IGSTK. En la figura 2.9a se muestra como se integra la solución propuesta en la arquitectura de IGSTK.

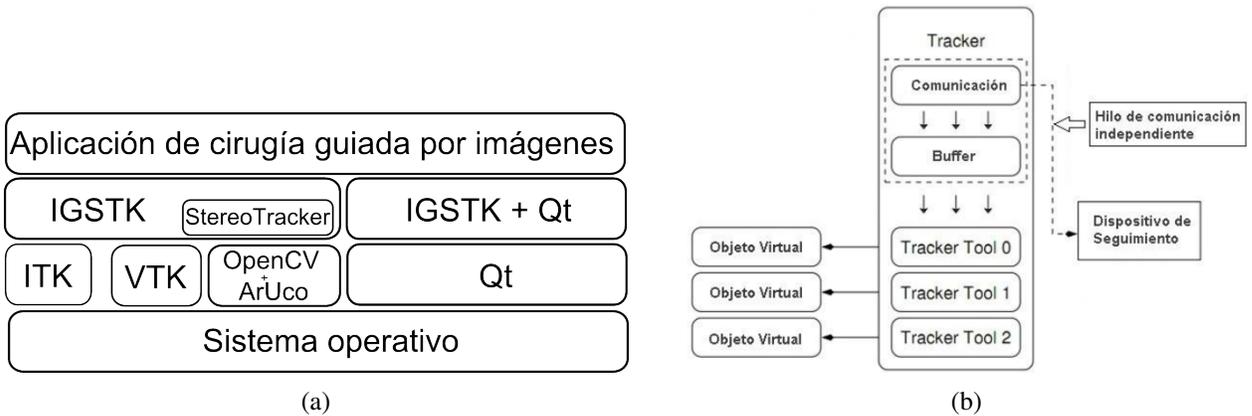


Figura 2.9. Integración de la solución propuesta con IGSTK: a) vista de la arquitectura, b) funcionamiento del *buffer*.

### 2.9.3. Diagrama de clases y patrones de diseño

La integración del componente de software de seguimiento con la biblioteca IGSTK se realizó según el diagrama de clases de la figura 2.10. En este diagrama se representan las clases donde se implementa la solución.

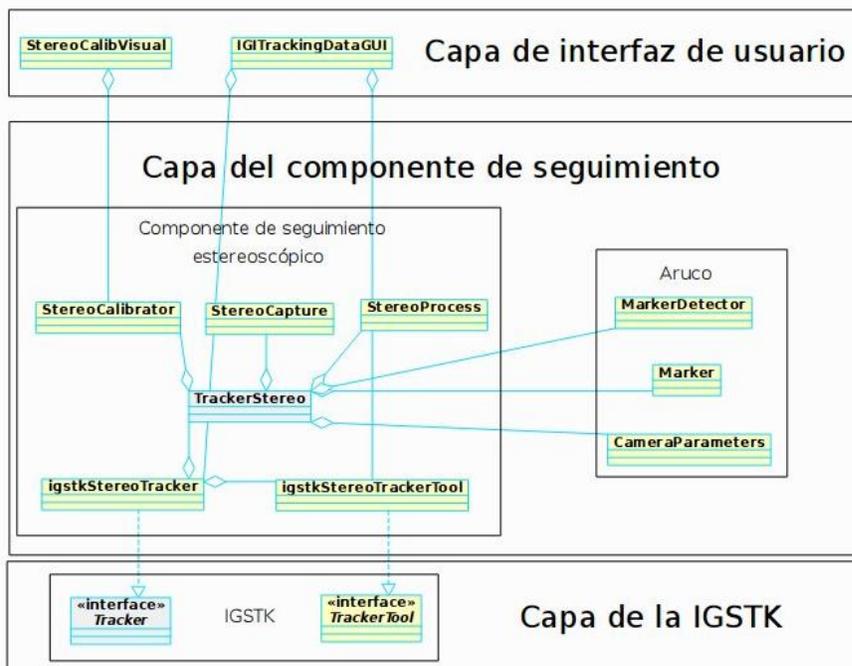


Figura 2.10. Diagrama de clases del diseño de la propuesta de solución.

Los patrones de diseño son considerados soluciones ya probadas a problemas de desarrollo de software sujeto a contextos similares. Los [Patrones de Software de Asignación de Responsabilidad General \(GRASP\)](#), se

consideran una serie de buenas prácticas para el diseño de software (TABARES, 2011).

El patrón *Experto* se aplicó en cada una de las clases que conforman el *TrackerStereo*. Cada una está encargada de una serie de funcionalidades que solo ella conoce, con esto se asegura el encapsulamiento.

El uso del patrón *Creador* se ve de manifiesto en las clases *TrackerStereo* e *igstkStereoTracker*. Estas clases son las encargadas de crear instancias de las subordinadas, no en el otro sentido.

Los patrones *Alta cohesión y bajo acoplamiento* están estrechamente ligados, las responsabilidades de las clases están asignadas, de tal forma que cada una realice actividades específicas y exista la mínima dependencia con otras clases. Esto se aprecia con mayor rigor en las clases *StereoProcess* y *StereoCapture*, que tienen un conjunto de funciones propias de ellas para el procesamiento de las imágenes y el acceso a los dispositivos de captura.

El patrón *Controlador* es utilizado para separar la capa de lógica del negocio de la capa de aplicación. Este se pone de manifiesto en la clase *MainWindow* de la herramienta de calibración, la cual controla todo el proceso de negocio asignando responsabilidades a las clases correspondientes.

Dentro de los patrones [Gang of Four \(GOF\)](#) se usó el *patrón estado* en la implementación de los estados de las clases *igstkStereoTracker* e *igstkStereoTrackerTool*, al mismo tiempo estas se ajustan al patrón *Fabrica abstracta*. Además como se comentó en la sección anterior el concepto de máquina de estado fue introducido por Alan Turing en 1936. Una máquina de estado es definida por un conjunto de estados, entradas y transiciones de un estado a otro. Puede ser determinista si dada una entrada existe solo una transición a un único estado (FREEMAN; ROBSON; BATES y SIERRA, 2004). Esta biblioteca implementa este patrón para mejorar la seguridad y robustez en sus aplicaciones, limita los posibles comportamientos y asegura que el programa se encuentre siempre en una condición válida. La máquina de estados se definió mediante macros de la IGSTK. Una ventaja de este enfoque es la generación de un código válido y robusto.

## 2.10. Conclusiones parciales

En este capítulo se describió el dispositivo estereoscópico de seguimiento y las etapas del proceso seguimiento de la propuesta de solución. El diseño del dispositivo se realizó con dos *webcams* de bajo costo, el cual ilustra el funcionamiento general de un sistema estereoscópico y permitió consolidar los elementos teóricos del seguimiento tridimensional mediante un sistema estereoscópico. Para al puesta en marcha del sistema de seguimiento se implementó una herramienta auxiliar con interfaz gráfica de usuario para la calibración del mismo. Esta herramienta disminuyó considerablemente la complejidad de la interacción del usuario con respecto al método propuesto por OpenCV mediante una consola de comandos. La utilización de la ArUco basada en OpenCV permitió definir un modelo de marcador con una estructura bien conocida que es detecta-

da por un algoritmo robusto y fiable de detección de marcadores y sirvió como una aproximación al modelo de detección que utilizan los sistemas comerciales. Se implementó el algoritmo iterativo-LS porque según el estudio de (R. I. HARTLEY y STURM, 1997) produce resultados más estables y robustos que el método de triangulación propuesto por OpenCV. Finalmente se propuso un diseño de la solución flexible basado en el uso de patrones de diseño como buenas prácticas de programación, que se ajusta a la arquitectura de la biblioteca IGSTK utilizada para la creación del software de neuronavegación.

---

## Análisis y discusión de los resultados

---

En el presente capítulo, se describe el seguimiento de los instrumentos quirúrgicos a partir de la solución propuesta en el capítulo anterior. Se describen los resultados obtenidos por el proceso de triangulación y el valor de la **RMS** resultante del cálculo de la transformación rígida asociada a los marcadores. Finalmente se hace un análisis de los resultados alcanzados en términos de la exactitud que ofrece la solución propuesta mediante la calibración de pivote de un prototipo de instrumento quirúrgico.

### 3.1. Estimación de la transformación mediante la triangulación

El componente de seguimiento de un neuronavegador debe alcanzar la mayor exactitud posible. Este tiene una misión crítica, dado que la vida de los pacientes puede correr peligro en caso de un funcionamiento incorrecto del software. La determinación de la exactitud del seguimiento es una tarea crítica para los sistemas de neurocirugía guiados por imágenes. Una forma convencional de evaluar dicha exactitud es calcular la **RMS** durante la estimación de la transformación de los marcadores (FITZPATRICK; J.B. y MAURER JR, 1998). Sistemas comerciales como el Polaris Vicra de (*Northern Digital s.f.*) pueden alcanzar valores de **RMS** de 0.35 *mm* en el volumen de medición. Para que el cirujano aborde una lesión, debe pasar el instrumento quirúrgico a través de regiones de diferente valor funcional. Una medida con algunos milímetros de error, puede significar el acceso a zonas que no debían ser intervenidas y la ocurrencia de incidentes inesperados.

Por tal motivo, en esta sección se decide realizar un experimento para evaluar la exactitud del método de seguimiento propuesto en la solución. En este experimento se comparan los resultados alcanzados en la estimación de la transformación mediante el método de triangulación propuesto iterativo-LS, y el método

implementado por OpenCV `cv::triangulatePoints`. El objetivo es comprobar cuál de los dos algoritmos ofrece mayor exactitud en la triangulación.

#### 3.1.1. Descripción del escenario de pruebas

La calidad de la estimación de la triangulación se comprobó mediante dos criterios, la velocidad de ejecución de las implementaciones, así como su exactitud. En el experimento se utilizó la configuración estereoscópica descrita en la figura 2.2 con una línea base de 100 mm. Se recomienda que para este tipo de sistemas se utilicen en la medida de lo posible el mismo modelo de cámaras del mismo fabricante. Las dos cámaras utilizadas fueron dos webcams Logitech Quick Chat con una resolución de 640x480 píxeles cada una.

#### 3.1.2. Análisis de rendimiento

Para la evaluación de la velocidad de ejecución de los algoritmos se adquirieron 1000 transformaciones de un marcador mientras se movía por el volumen de visión de las cámaras definido en  $[0, 1000] \times [0, 1000] \times [0, 1000] \text{ mm}^3$ . A cada transformación se le puso una marca de tiempo y se guardó en memoria, para no interferir la ejecución. Al final, los datos adquiridos fueron escritos en un archivo de texto para su posterior análisis. El objetivo es calcular la diferencia de tiempo existente entre transformaciones sucesivas y calcular la cantidad de veces que se estima la transformación por segundo, asociando esta medida al rendimiento tradicional de fotogramas por segundo.

#### 3.1.3. Análisis de la triangulación

Para medir la exactitud de la triangulación se utilizó el tamaño del marcador, las dimensiones para la prueba fueron de  $(48 \times 48 \text{ mm}^2)$ . En este caso se obtienen las esquinas del marcador con la biblioteca ArUco, se estima su posición  $(x, y, z)$  en el espacio utilizando los algoritmos de triangulación y se mide la longitud del lado del marcador a partir de la posición 3D estimada de sus esquinas. Luego se compara la longitud estimada con la longitud conocida para comprobar si existen diferencias y calcular el error de estimación. Este error es conocido en la literatura como *Error de Registro del Seguimiento (TRE)* y está definido por la ecuación 3.1.1, donde  $T_{pconocida}$  es un valor de conocido y  $T_{pestimada}$  es el valor estimado a partir de la triangulación.

$$TRE = \|T_{pconocida} - T_{pestimada}\| \quad (3.1.1)$$

Las mediciones del experimento se realizaron en 3 localizaciones de profundidad conocidas dentro del volumen de visión. La figura 3.1 muestra la configuración de las cámaras y las posiciones en las que se hizo la medición.

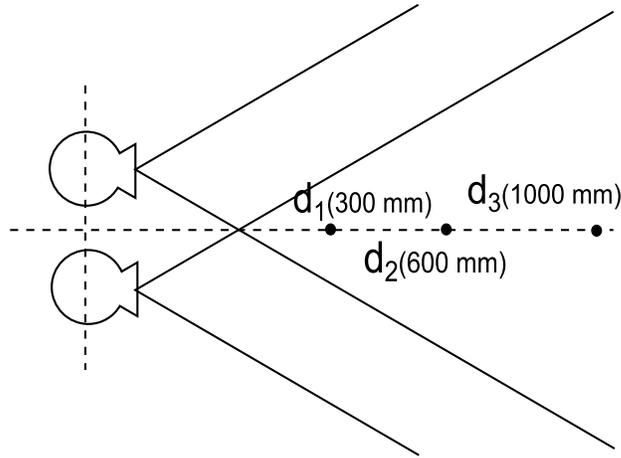


Figura 3.1. Configuración del sistema para tomar las medidas.

### 3.1.4. Análisis de la transformación rígida

Para tener una medida de la exactitud en la estimación de la transformación del componente implementado (ver sección 2.8.1), se calculó el error asociado a la transformación rígida correspondiente a un marcador. La transformación rígida es capaz de alinear dos conjuntos de puntos del mismo espacio, aplicando solamente rotaciones y traslaciones. Teniendo la matriz de rotación  $R$ , el vector de traslación  $t$  y los conjuntos de puntos  $P_a$  y  $P_b$  de  $N$  puntos cada uno, se transforma el conjunto de puntos inicial  $P_a$  en  $P_{a'}$  con la siguiente ecuación:

$$P_{a'} = RP_a + t \quad (3.1.2)$$

El conjunto de puntos  $P_{a'}$  debe estar alineado con  $P_b$ , pues se le aplicó la transformación rígida obtenida. En una situación con distorsión cero, no existiría error, o estaría muy cercano a cero. En este caso el conjunto de puntos asociado al marcador fue hallado aplicando técnicas para su detección en las imágenes distorsionadas, y luego se aplicó la triangulación. El error asociado a la transformación rígida se calcula mediante la ecuación 3.1.3:

$$RMS = \sqrt{\frac{\sum_{i=1}^N (P_{a'i} - P_{bi})^2}{N}} \quad (3.1.3)$$

### 3.2. Resultados experimentales de la triangulación y la estimación de la transformación

Para la selección del método de triangulación, se compararon el método propuesto iterativo-LS, y el método implementado por OpenCV *cv::triangulatePoints*. Luego de adquirirse 1000 capturas de transformaciones del marcador en cada distancia, se calcularon los errores de transformación utilizando ambos métodos. El gráfico 3.2 muestra una comparación donde el error está dado en milímetros y se observa una clara diferencia que favorece al iterativo-LS. En la tabla 3.1 se resumen los valores obtenidos a partir de la evaluación descrita en la sección 3.1. De forma general se puede decir que el error varía de acuerdo a la localización espacial del marcador. El error aumenta mientras mayor es la distancia a la que se encuentra el marcador del sistema estereoscópico y es menor cuando el marcador se encuentra cercano a 300 mm. Este resultado se puede apreciar visualmente en la figura 3.2.

Tabla 3.1. Resumen de valores del cálculo de errores de estimación de la transformación según las localizaciones de la figura 3.1, mediante *cv::triangulatePoints* de OpenCV y el iterativo-LS.

Distancia del sistema	300 mm		600 mm		1000 mm	
	OpenCV	Iter.-LS	OpenCV	Iter.-LS	OpenCV	Iter.-LS
<b>RMS</b> (media) (mm)	0.58	0.34	1.77	0.99	3.64	2.67
<b>RMS</b> (mínimo) (mm)	0.21	0.1	0.22	0.13	0.4	0.52
<b>RMS</b> (máximo)(mm)	2.4	2.75	5.35	3.08	12.76	8.71
Longitud de arista promedio(mm)	48.72	48.17	50.04	48.07	51.92	49.33
Longitud de arista Mínima (mm)	48.01	47.61	47.21	46.31	44.99	44.26
Longitud de arista Máxima (mm)	49.85	49.74	57.46	51.86	74.59	63.31
<b>TRE</b> (mm)	0.72	0.17	2.04	0.07	3.92	1.33

El mismo comportamiento se observa cuando se realiza el cálculo de las dimensiones del marcador a partir de los resultados de triangulación de los algoritmos de OpenCV y el iterativo-LS. El error de la estimación de las dimensiones del marcador aumenta conforme este se aleja del dispositivo de seguimiento. A partir de la tabla 3.2, se aprecia una pequeña diferencia entre los resultados alcanzados por el algoritmo de OpenCV *cv::triangulatePoints* y los ofrecidos por el iterativo-LS. Se observa que los resultados del algoritmo de OpenCV aumentan el error **TRE** en mayor medida que el iterativo-LS, conforme se incrementa la distancia del marcador respecto al sistema estereoscópico.

En la tabla 3.2 se observa mediante la **RMS** que la estimación de la transformación también se ve afectada por los resultados de la triangulación. En la figura 3.2 se observa gráficamente que la **RMS** como consecuencia del algoritmo iterativo-LS muestra valores más bajos que con el algoritmo de OpenCV. La interpretación de estos resultados indica que el método de triangulación basado en la implementación del algoritmo iterativo-LS ofrece mayor exactitud que el que se encuentra implementado en OpenCV. Aunque es válido destacar que en distancias cercanas a 300 mm los resultados tienen diferencias mínimas.

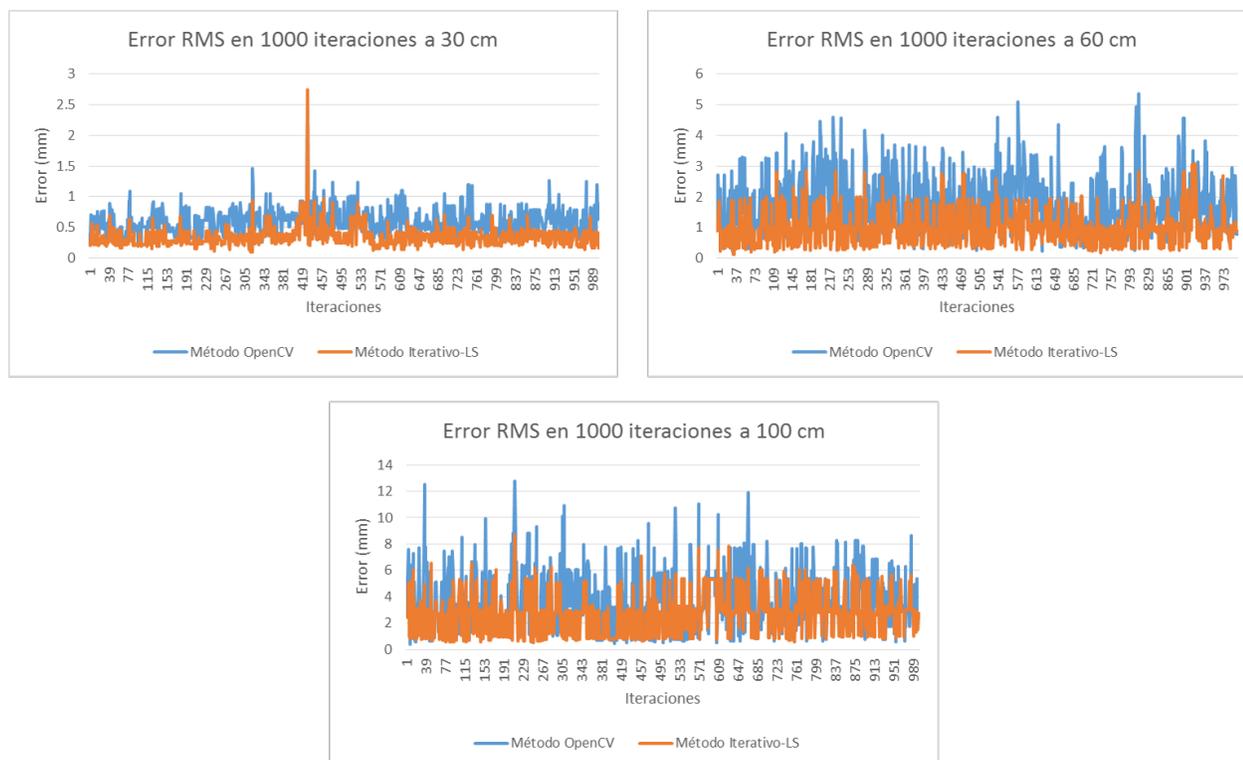


Figura 3.2. RMS a partir de los algoritmos de triangulación `cv::triangulatePoints` de OpenCV y el iterativo-LS.

La frecuencia de ejecución utilizando las dos implementaciones en la triangulación es superior a los 30 Hz, lo cual garantiza que se obtengan valores en tiempos interactivos. De forma general en la tabla 3.1 se observa que mediante el método de triangulación iterativo-LS se obtienen resultados con mayor exactitud, evidenciado en los valores TRE asociados.

### 3.3. Seguimiento de la herramienta quirúrgica

Uno de los pasos críticos en la cirugía guiada por imagen es el registro de imágenes pre-operatorias al sistema de coordenadas del paciente. Varias técnicas de registro han sido desarrolladas para este propósito, y se dividen en dos grandes categorías, basadas en marcos y sin marcos. En una técnica de registro basada en marcos, marcos estereotáxicos son adjuntados al órgano de interés para proporcionar una referencia rígida. Este método de registro es común en neurocirugía. En el registro sin marcos, se utilizan marcas de referencia (fiduciales) tanto en la imagen preoperatoria como en el paciente para calcular los parámetros de transformación que relacionan la imagen preoperatoria y el paciente. Los fiduciales pueden ser puntos o parches de superficie. En IGSTK existe una clase implementada (`igstk::Landmark3DRegistration`) para el registro de fiduciales de un cuerpo rígido 3D basado en puntos. Utilizando un dispositivo de seguimiento o puntero, el

operador identifica los puntos de referencia en la imagen preoperatoria y en el cuerpo del paciente, luego los parámetros de transformación son calculados en base a esos puntos. IGSTK también provee una herramienta para predecir errores en el registro de fiduciales: *igstk::igstkLandmarkRegistrationErrorPredictor*.

#### 3.3.1. Calibración de la herramienta quirúrgica

Para la navegación intraoperatoria se deben acoplar los marcadores a los instrumentos quirúrgicos. Según (MARTÍNEZ MAÑAS, 2004) se pueden alcanzar altos valores de exactitud al utilizar sistemas con brazos mecánicos en combinación con marcos estereotáxicos, con valores de  $1.665\text{ mm}$ . Con los sistemas sin marco la exactitud disminuye y puede llegar a alcanzar valores de hasta  $4.8\text{ mm}$ , probados con sistemas comerciales de campo magnético. En casos de sistemas ópticos como el neuronavegador de BrainLab en la práctica se han alcanzado valores de exactitud de  $4\text{ mm}$ . Este mismo autor indica que siempre es deseable una exactitud máxima, aunque muchos procedimientos neuroquirúrgicos no la requieren. La identificación y localización de estructuras cerebrales y óseas es una acción que se realiza al inicio del procedimiento quirúrgico y para esta acción es suficiente una exactitud de  $3 - 4\text{ mm}$ , la cual es superior a la que pueden lograr la mayoría de los neurocirujanos. Esta exactitud es menor a la alcanzada por los sistemas con marco pero se encuentra dentro del rango aceptado en los sistemas de neuronavegación (*ibíd.*).



Figura 3.3. Patrón acoplado que simula la herramienta quirúrgica.

Los marcadores definidos para el seguimiento se acoplan en el instrumento quirúrgico como se muestra en la figura 3.3. El marcador acoplado es detectado con respecto al sistema de referencia del dispositivo estereoscópico. Sin embargo, se necesita conocer dónde se encuentra posicionada la punta del instrumento con respecto al marcador acoplado para poder deducir su posición física con respecto al sistema de referencia del dispositivo estereoscópico. Estas posiciones relativas son provistas por un paso previo de calibración de la herramienta quirúrgica.

El método de calibración de los instrumentos es el llamado calibración de pivote. El principio básico de este método es mover la herramienta alrededor de un punto estático que actúa como centro de gravedad o pivote y tomar varias posiciones  $G_i$  del marcador acoplado, describiendo una esfera (ver figura 3.4). El centro de la esfera corresponde a la punta del instrumento. Existen varios métodos equivalentes para la estimación de la punta del instrumento: estimar el centro de la esfera a partir del ajuste de las posiciones estimadas a una esfera (NICOLAU; GOFFIN y SOLER, 2005) o resolviendo un sistema de ecuaciones sobredeterminado (BAUER, 2007). El primer método se utiliza cuando solamente se tienen posiciones y el segundo se utiliza cuando se tienen posiciones y orientaciones del marcador en el instrumento.

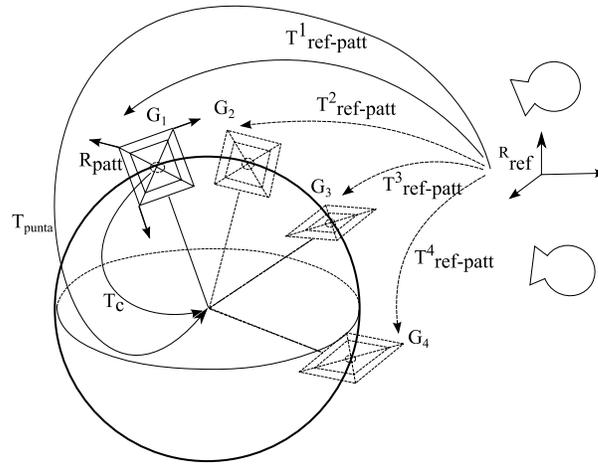


Figura 3.4. Pivote de la herramienta de calibración.

Como en la detección del marcador se obtiene la matriz de transformación  $T_{ref-patti} = [R_i, t_i]$  que contiene una matriz de rotación  $R_i$  y un vector de posición  $t_i$ , entonces se puede componer un sistema de ecuaciones

$$\begin{pmatrix} t_{punta} \\ 1 \end{pmatrix} = \begin{pmatrix} R_i & t_i \\ 0 & 1 \end{pmatrix} \begin{pmatrix} t_c \\ 1 \end{pmatrix} \quad (3.3.1)$$

donde  $t_{punta}$  es la posición de la punta con respecto al sistema de referencia global y  $t_c$  es la transformación con respecto al sistema de referencia del marcador (TUCERYAN; GREER; WHITAKER et al., 1995). Este es el método que emplea IGSTK para hacer la calibración de pivote en su marco de trabajo (ENQUOBAHRIE y Ziv YANIV, 2009). La solución se obtiene con una aproximación por mínimos cuadrados adquiriendo múltiples transformaciones mientras se rota el instrumento quirúrgico sobre su punta, el resultado es el sistema sobredeterminado:

$$\begin{bmatrix} R_0 & -I \\ \vdots & \vdots \\ R_n & -I \end{bmatrix} \begin{bmatrix} t_c \\ t_{punta} \end{bmatrix} = \begin{bmatrix} -t_0 \\ \vdots \\ -t_n \end{bmatrix} \quad (3.3.2)$$

que se resuelve por el método pseudoinverso (STRANG, 2009).

IGSTK implementa este método en tres clases diferentes:

1. *PivotCalibrationAlgorithm*. En esta clase se hace la implementación del método descrito con anterioridad y devuelve la **RMS** de la estimación de la transformación.
2. *PivotCalibration*. Esta clase es la encargada de la adquisición de las transformaciones de los marcadores acoplados al instrumento quirúrgico, utiliza la clase *PivotCalibrationAlgorithm* para calcular los valores de  $t_c$  y  $t_{punta}$
3. *PivotCalibrationFLTKWidget*. Esta es una clase que provee la interfaz gráfica de usuario para la clase *PivotCalibration* y provee una retroalimentación visual del proceso de captura de las transformaciones (barra de progreso) y los resultados de la calibración

IGSTK ofrece la ventaja de que estas clases trabajan con el **API** del *igstk::Tracker* genérico. Por tal motivo brinda las funcionalidades de calibración por pivote para todos los sistemas de seguimiento que se ajusten a este diseño. La solución obtenida en (MARTÍNEZ, 2012) no se utilizó para comprobar los resultados alcanzados, debido a que no se cuenta con dicha solución, además que se utiliza el seguimiento de la biblioteca ARToolKit que exige el pago de licencia para contextos académicos y comerciales, aunque se tuvo en cuenta la **RMS** promedio de 4.17 mm de su solución. En su lugar se utilizó una solución equivalente provista por (GULER y Z. YANIV, 2012), donde los autores proveen utilidades de software con varias aplicaciones hechas con IGSTK. Entre las herramientas provistas se encuentra una utilidad de calibración de pivote con la detección de marcadores provista por la biblioteca ArUco en un sistema monocular. A esta herramienta se le adicionó el componente estereoscópico de seguimiento propuesto por la presente investigación y se realizó el proceso de calibración de pivote. Se utilizó la misma configuración propuesta en la sección 3.1.3 descrita en la figura 3.1.

Se compararon los resultados de calibración de pivote usando el componente estereoscópico propuesto en esta investigación con el sistema monocular propuesto por (ibíd.), a partir de la **RMS** resultante de la estimación de la transformación de la punta. El proceso de calibración de pivote se realizó 7 veces tomando cantidades de posiciones distintas respectivamente. Se utilizó la misma calibración intrínseca de las cámaras para probar ambos sistemas en igualdad de condiciones.

Los resultados obtenidos se resumen en la tabla 3.2, en los cuales se observa de forma general que la solución propuesta estima la transformación de la punta con un error medio de 1.81 *mm* a diferencia de los 4.36 *mm* ofrecidos por el sistema monocular. Existe entre ambos métodos una diferencia de 2.55 *mm*, lo cual aunque es un valor pequeño, es significativo por el contexto en el que se utiliza este tipo de aplicaciones. Comparado con los resultados alcanzados por (MARTÍNEZ, 2012) con valor **RMS** medio de 4.17 *mm*, la solución propuesta ofrece una exactitud significativamente menor con una diferencia de 2.36 *mm*. La interpretación de este resultado indica que la solución propuesta localiza la punta del instrumento quirúrgico con un error de 1.81 *mm*, con las cámaras utilizadas. Es válido destacar que este proceso de calibración debe ser un paso previo a cualquier procedimiento quirúrgico, para conocer en todo momento con qué error se está trabajando. Este error está dado también por los niveles de ruido presentes en las imágenes y la resolución de las cámaras, que dificulta la detección efectiva de las esquinas de los marcadores.

Tabla 3.2. Resultados de calibración de pivote con los componentes monocular (GULER y Z. YANIV, 2012) y la solución propuesta.

No. de muestras	Error de transformación ( <b>RMS</b> )( <i>mm</i> )	
	Monocular	Estereoscópico
300	4.81	1.96
400	5.04	2.19
500	4.36	1.76
600	4.21	1.58
700	4.36	1.66
800	4.06	2.09
900	3.7	1.43
Promedio	4.36	1.81

En la figura 3.5 se muestra gráficamente las mediciones de la calibración del pivote. En esta se observa gráficamente cómo el valor de la **RMS** resultante de la estimación de la transformación de la punta del instrumento mediante la solución propuesta, tiene valores menores al método propuesto por (GULER y Z. YANIV, 2012)

En la figura 3.6 se muestra una imagen de la herramienta de calibración de pivote utilizada para la obtención de los datos. En esta figura *quaternion* indica la rotación estimada, *translation* es la traslación relativa al sistema de referencia del marcador acoplado al instrumento de pivote y *pivot point* es la traslación con respecto al sistema de referencia global del dispositivo de seguimiento. *RMSE* es el valor de la **RMS** resultante de la estimación de la transformación rígida.

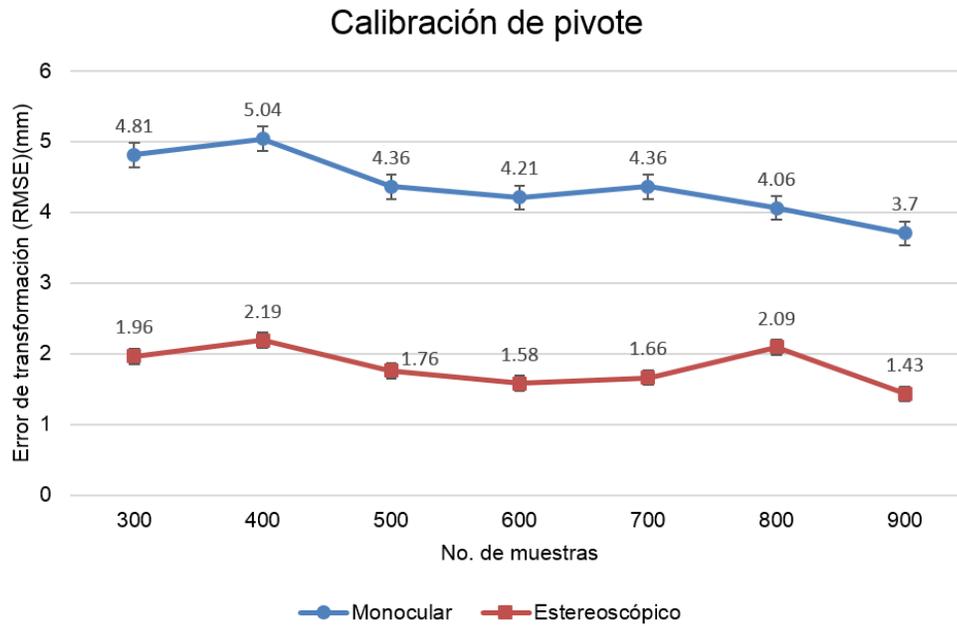


Figura 3.5. Resultados obtenidos de la calibración de pivote mediante el seguimiento monocular (GULER y Z. YANIV, 2012) y estereoscópico.

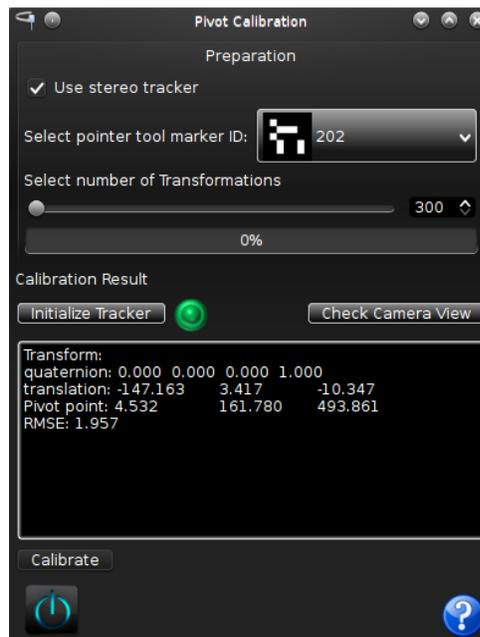


Figura 3.6. Herramienta de calibración de pivote utilizando la solución propuesta.

### 3.4. Conclusiones parciales

En este capítulo se analizaron los resultados obtenidos durante la realización de la presente investigación. Se comprobó que el método de triangulación iterativo-LS seleccionado para la solución propuesta ofrece mejores valores con mayor exactitud que el método *cv::triangulatePoints* propuesto en OpenCV. A partir de la calibración de pivote de un prototipo de instrumento quirúrgico utilizando un sistema monocular y la solución propuesta, se observa que el sistema estereoscópico ofrece resultados con menor error en la calibración. Esto indica que con la configuración utilizada en la solución propuesta se comete un error medio de 1.81 *mm* en la estimación de la transformación de la punta del instrumento. Estos resultados son alentadores, teniendo en cuenta que las cámaras utilizadas tienen una resolución de 640x480 píxeles y ofrecen niveles de ruido que afectan la detección de los marcadores. Esta exactitud puede mejorarse si se utilizan cámaras de mejor calidad, ya que la solución no depende del hardware y sí de la calidad de la imagen.

El componente todavía no ofrece los valores de exactitud adecuados para ser utilizado en ambientes reales de neurocirugía, pero se demostró que el uso de técnicas estereoscópicas es un enfoque acertado. No obstante, con la exactitud obtenida se puede construir un sistema de neuronavegación guiada por imágenes como apoyo a la educación de neurocirujanos y desarrolladores. Para los neurocirujanos como una herramienta de familiarización de los conceptos y procedimientos asociados a la neuronavegación y sus posibles dificultades. Para los desarrolladores como una infraestructura alternativa de bajo costo a los sistemas comerciales, que puede ser utilizada para el desarrollo de funcionalidades de neuronavegadores.

---

## Conclusiones

---

Con la realización de la presente investigación pueden ser enunciadas a las siguientes conclusiones:

- Se desarrolló un componente estereoscópico de seguimiento tridimensional basado en grabación de video, utilizando cámaras de bajo costo. Se utilizó visión por computadoras mediante la detección de marcadores provista por ArUco y se comprobó que el método de triangulación iterativo-LS ofrece resultados superiores al implementado en OpenCV.
- Se desarrolló una herramienta de calibración de cámaras que puede ser utilizada para calibrar otros sistemas de cámaras.
- El diseño flexible permitió integrar la solución propuesta con la biblioteca de desarrollo de aplicaciones de navegación guiada por imágenes IGSTK. Esto permitió la reutilización de las funcionalidades que esta biblioteca posee para el desarrollo de aplicaciones para navegación guiada por imágenes y la reutilización de herramientas de calibración de pivote existentes, para comparar los resultados alcanzados con soluciones provistas por la comunidad científica.
- Los resultados alcanzados mediante la calibración de pivote demostraron que el componente propuesto en esta investigación ofrece resultados con una exactitud con un valor de la **RMS** de  $1.81 \text{ mm}$ . Este resultado indica que se logró incrementar la exactitud de  $4.17 \text{ mm}$  que se había alcanzado hasta el momento, con tiempos de ejecución interactivos en el contexto de la presente investigación. Sin embargo, aunque la exactitud es limitada, la solución propuesta se puede utilizar como una alternativa a los sistemas comerciales por su bajo costo, para el desarrollo de las funcionalidades de un sistema de neuronavegación. Al mismo tiempo puede ser utilizado para desarrollar sistemas de neuronavegación para apoyar la docencia en la formación de neurocirujanos; para la familiarización de los conceptos y procedimientos asociados a la neuronavegación y sus posibles dificultades.

---

## Recomendaciones

---

Esta investigación actúa como punto de partida para futuras investigaciones en el área de la visión estereoscópica, el seguimiento tridimensional de objetos y sistemas de seguimiento para la neurocirugía. Algunas recomendaciones para futuros trabajos son:

- Utilizar el sistema de seguimiento estereoscópico propuesto en otros sistemas de navegación guiada por imágenes.
- Probar el sistema de seguimiento estereoscópico propuesto con cámaras que ofrezcan: bajos niveles de ruido en sus imágenes, mayor definición, mayor resolución y que ofrezcan sincronización por hardware para comparar las diferencias con los resultados obtenidos en esta investigación.

**buffer** Memoria intermedia del sistema. [14](#), [43](#), [44](#)

**fotogrametría** Técnica para determinar las propiedades geométricas de los objetos y las situaciones espaciales a partir de imágenes fotográficas. [14](#)

**Hamming** Código detector y corrector de errores que lleva el nombre de su inventor, Richard Hamming. En los datos codificados se pueden detectar errores en un bit y corregirlos. [39](#)

**SIFT** Transformación de características invariantes a escalado, SIFT por sus siglas en inglés. [13](#)

**triangulación** Hallar la posición de un punto en el espacio interceptando dos rayos correspondientes a dos proyecciones del mismo punto. [8](#), [9](#), [24](#)

**USB** Canal para el traspaso de datos universal. [35](#)

- API** Interfaz de programación de aplicaciones. [10](#), [54](#)
- CAS** cirugía asistida por computadora. [5](#)
- GOF** Gang of Four. [45](#)
- GRASP** Patrones de Software de Asignación de Responsabilidad General. [44](#)
- IDE** Entorno de desarrollo Integrado. [29](#), [30](#)
- IGSTK** Imaged-Guided Surgery Toolkit. [10](#), [29](#), [30](#), [42](#), [43](#)
- ITK** Insight Segmentation and Registration Toolkit. [10](#)
- NCC** Correlación cruzada normalizada. [23](#)
- RA** Realidad Aumentada. [11](#), [29](#)
- RMS** Raíz del Error Cuadrático Medio. [4](#), [47](#), [50](#), [51](#), [54](#), [55](#), [58](#)
- SAD** Suma de diferencias absolutas. [23](#)
- SSD** Suma de diferencias al cuadrado. [23](#)
- SVD** Descomposición en valores singulares de una matriz. [26](#), [41](#)
- TRE** Error de Registro del Seguimiento. [48](#), [50](#), [51](#)
- UCI** Universidad de las Ciencias Informáticas. [12](#)
- VERTEX** Centro Entornos Interactivos 3D. [2](#)

**VTK** Visualization Toolkit. [10](#)

**XML** Extensible Markup Language. [36](#)

**YML** Formato de serialización de datos legible al humano. [36](#)

---

## Referencias bibliográficas

---

- AVA. *ArUco: a minimal library for Augmented Reality applications based on OpenCv*. (Visited on 04/02/2015).  
Url: <http://www.uco.es/investiga/grupos/ava/node/26>.
- AVA. (Visited on 06/02/2015). Url: <http://www.uco.es/investiga/grupos/ava/node/1>.
- BAGGIO, Daniel Lélis. 2012. *Mastering OpenCV with practical computer vision projects*. 2012.
- BAILEY, Brenton y WOLF, Alexander. 2007. Real Time 3D motion tracking for interactive computer simulations. *Unter: http://www3.imperial.ac.uk/pls/portallive/docs/1/20673697.PDF (abgerufen am 21.07.2014)*. 2007, vol. 3, n.º 1, págs. 3-1.
- BARNARD, Stephen T y FISCHLER, Martin A. 1982. Computational stereo. *ACM Computing Surveys (CSUR)*. 1982, vol. 14, n.º 4, págs. 553-572.
- BAUER, Martin. 2007. *Tracking Errors in Augmented Reality*. 2007.
- BRADSKI, Gary y KAEHLER, Adrian. 2008. *Learning OpenCV: Computer vision with the OpenCV library*. 2008.
- BUCHOLZ, Richard D y SMITH, Kurt R. 1993. A comparison of sonic digitizers versus light emitting diode-based localization. *Interactive image-guided neurosurgery*. 1993, págs. 179-200.
- CARBAJAL, Guillermo; GOMEZ, Alvaro; PEREYRA, Gabriela et al., 2010. First neuronavigation experiences in uruguay. En. *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*. 2010, págs. 2317-2320.
- Claron Technology Inc. (Visited on 06/02/2015). Url: <http://www.clarontech.com>.
- CMake. (Visited on 06/20/2015). Url: <http://www.cmake.org/>.
- DÍAZ, Joaquín Alberto. 2014. *Componente de software para Realidad Aumentada, basado en la detección de características naturales en imágenes*. 2014.
- ENDRESEN, Kai Hugo Hustoft. 2010. *Tracking objects in 3D using Stereo Vision*. 2010.
- ENQUOBAHRIE, Andinet y YANIV, Ziv. 2009. *IGSTK: The Book*. 2009.

- FIALA, M. 2005. ARTag, a fiducial marker system using digital techniques. En. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. 2005, págs. 590-596 vol. 2. Url: <http://dx.doi.org/10.1109/CVPR.2005.74>).
- FITZPATRICK, J.M.; J.B., West y MAURER JR, C.R. 1998. Predicting error in rigid-body point-based registration. *IEEE Trans. Med. Imag.* 1998, vol. 14, págs. 694-702.
- FONSECA, Adrián. 2011. *Adquisición de modelos geométricos a partir de objetos del mundo real*. 2011.
- FREEMAN, Eric; ROBSON, Elisabeth; BATES, Bert y SIERRA, Kathy. 2004. *Head first design patterns*. 2004.
- FUA, Pascal y LEPETIT, Vincent. 2005. Vision based 3D tracking and pose estimation for mixed reality. *Emerging Technologies of Augmented Reality Interfaces and Design*. 2005, págs. 43-63.
- GARRIDO, S. Jurado. 2014. Automatic Generation and Detection of Highly Reliable Fiducial Markers Under Occlusion. *Pattern Recogn.* 2014, vol. 47, n.º 6, págs. 2280-2292. Url: <http://dx.doi.org/10.1016/j.patcog.2014.01.005>). ISSN 0031-3203.
- GÓMEZ, Alvaro y RANDALL, Gregory. 2007. *NAVEGACIÓN GUIADA POR IMÁGENES EN NEUROCI-RUGÍA*. 2007.
- GULER, O. y YANIV, Z. 2012. Image-guided navigation: A cost effective practical introduction using the image-guided surgery toolkit (IGSTK). En. *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*. 2012, págs. 6056-6059. Url: <http://dx.doi.org/10.1109/EMBC.2012.6347375>).
- HARTLEY, Richard I y STURM, Peter. 1997. Triangulation. *Computer vision and image understanding*. 1997, vol. 68, n.º 2, págs. 146-157.
- HARTLEY, Richard y ZISSERMAN, Andrew. 2003. *Multiple view geometry in computer vision*. 2003.
- HIRSCHMULLER, Heiko; INNOCENT, Peter R y GARIBALDI, Jon. 2002. Real-time correlation-based stereo vision with reduced border errors. *International Journal of Computer Vision*. 2002, vol. 47, n.º 1-3, págs. 229-246.
- JACOB, Ann Maria y ANITHA, J. 2012. Inspection of Various Object Tracking Techniques. *International Journal of Engineering and Innovative Technology*. 2012, vol. 2, págs. 118-124.
- JALAL, Anand Singh y SINGH, Vrijendra. 2012. The State-of-the-Art in Visual Object Tracking. *Informatica (Slovenia)*. 2012, vol. 36, n.º 3, págs. 227-248.
- KABSCH, W. 1978. A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*. 1978, vol. 34, n.º 5, págs. 827-828. Url: <http://dx.doi.org/10.1107/S0567739478001680>).

- KANADE, Takeo y OKUTOMI, Masatoshi. 1994. A stereo matching algorithm with an adaptive window: Theory and experiment. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 1994, vol. 16, n.º 9, págs. 920-932.
- KATO, Hirokazu. *ARToolKit. Open Source Augmented Reality SDK*. (Visited on 06/02/2015). Url: <http://www.artoolkit.org/>.
- KOLLER, Dieter; KLINKER, Gudrun; ROSE, Eric et al., 1997. Real-time vision-based camera tracking for augmented reality applications. En. *Proceedings of the ACM symposium on Virtual reality software and technology*. 1997, págs. 87-94.
- LARSON, A Noelle; POLLY JR, David W; GUIDERA, Kenneth J et al., 2012. The accuracy of navigation and 3D image-guided placement for the placement of pedicle screws in congenital spine deformity. *Journal of Pediatric Orthopaedics*. 2012, vol. 32, n.º 6, págs. e23-e29.
- Logitech International S.A.* Url: <http://www.logitech.com>.
- LÓPEZ-VALLES, José M; FERNÁNDEZ-CABALLERO, Antonio y FERNÁNDEZ, Miguel A. 2005. Conceptos y técnicas de estereovisión por computador. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*. 2005, vol. 9, n.º 27, págs. 35-62.
- LORUSSO, A.; EGGERT, D. W. y FISHER, R. B. 1995. A Comparison of Four Algorithms for Estimating 3-D Rigid Transformations. En. *Proceedings of the 1995 British Conference on Machine Vision (Vol. 1)*. Birmingham, United Kingdom : BMVA Press, 1995, págs. 237-246. BMVC '95. Url: <http://dl.acm.org/citation.cfm?id=236190.236213>). ISBN 0-9521898-2-8.
- LUNSFORD, L Dade; MARTINEZ, A Julio y LATCHAW, Richard E. 1986. Stereotaxic surgery with a magnetic resonance-and computerized tomography-compatible system. *Journal of neurosurgery*. 1986, vol. 64, n.º 6, págs. 872-878.
- MARTÍNEZ, Fidel. 2012. *Sistema de Registro para software de Navegación Quirúrgica*. 2012.
- MARTÍNEZ FERNÁNDEZ DE LAS HERAS, José Javier. 2005. La descomposición en valores singulares (SVD) y algunas de sus aplicaciones. *Gaceta de la Real Sociedad Matemática Española*. 2005, vol. 8, n.º 3, págs. 796-810.
- MARTÍNEZ MAÑAS, Rosa María. 2004. *Precisión y utilidad de la neuronavegación en la cirugía encefálica*. 2004.
- MATTHIES, Larry; KANADE, Takeo y SZELISKI, Richard. 1989. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*. 1989, vol. 3, n.º 3, págs. 209-238.
- Medtronic*. Url: <http://http://www.medtronic.com>.
- MONTALVO, M. 2010. *Técnicas de visión estereoscópica para determinar la estructura tridimensional de la escena*. 2010.

- MORENO, Mileydi y CRUZ GUEVARA, Ernesto de la. 2008. *Localización de objetos virtuales en el mundo real con técnicas de Realidad Aumentada*. 2008.
- NICOLAU, S. A.; GOFFIN, L. y SOLER, L. 2005. A Low Cost and Accurate Guidance System for Laparoscopic Surgery: Validation on an Abdominal Phantom. En. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*. Monterey, CA, USA : ACM, 2005, págs. 124-133. VRST '05. Url: <http://doi.acm.org/10.1145/1101616.1101642>. ISBN 1-59593-098-1.
- Northern Digital. (Visited on 06/02/2015). Url: <http://www.ndigital.com/medical/products/polaris-family>.
- OKUTOMI, Masatoshi y KANADE, Takeo. 1993. A multiple-baseline stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 1993, vol. 15, n.º 4, págs. 353-363.
- PAJARES, G y CRUZ, JM. 2007. *Visión por Computador: Imágenes digitales y aplicaciones*, RA-MA. 2007.
- PAREKH, Himani S; THAKORE, Darshak G y JALIYA, Udesang K. 2014. A Survey on Object Detection and Tracking Methods. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol.* 2014, vol. 2.
- PETERS, Terry y CLEARY, Kevin. 2008. *Image-guided interventions: technology and applications*. 2008. *Qt Cross-platform application*. (Visited on 06/18/2015). Url: <http://www.qt.io/>.
- SAÍZ, M. 2010. *Reconstrucción tridimensional mediante visión estéreo y técnicas de optimización*. 2010.
- SCHARSTEIN, Daniel y SZELISKI, Richard. 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*. 2002, vol. 47, n.º 1-3, págs. 7-42.
- SEHNERT, Shelly. 2008. *Triangulation*. 2008.
- SEN-CHING, S Cheung y KAMATH, Chandrika. 2004. Robust techniques for background subtraction in urban traffic video. En. *Electronic Imaging 2004*. 2004, págs. 881-892.
- SORKINE, Olga. 2009. *Least-Squares Rigid Motion Using SVD*. 2009.
- STRANG, Gilbert. 2009. *Introduction to Linear Algebra*. 2009.
- SZELISKI, Richard. 2010. *Computer vision: algorithms and applications*. 2010.
- TABARES, Ricardo Botero. 2011. Patrones Grasp y Anti-Patrones: un Enfoque Orientado a Objetos desde Lógica de Programación. *Entre Ciencia e Ingeniería*. 2011, n.º 8, págs. 161-173.
- TREVISAN, Daniela G; NEDEL, Luciana P y MACQ, Benoit. 2008. Augmented vision for medical applications. En. *Proceedings of the 2008 ACM symposium on Applied computing*. 2008, págs. 1415-1419.
- TUCERYAN, M.; GREER, D.S.; WHITAKER, R.T. et al., 1995. Calibration requirements and procedures for a monitor-based augmented reality system. *Visualization and Computer Graphics, IEEE Transactions on*. 1995, vol. 1, n.º 3, págs. 255-273. Url: <http://dx.doi.org/10.1109/2945.466720>. ISSN 1077-2626.

- VITE-SILVA, Israel; CRUZ-CORTÉS, Nareli; TOSCANO-PULIDO, Gregorio y FRAGA, Luis Gerardo de la. 2007. Optimal triangulation in 3D computer vision using a multi-objective evolutionary algorithm. En. *Applications of Evolutionary Computing*. 2007, págs. 330-339.
- WIDMANN, Gerlig; STOFFNER, Rudolf y BALE, Reto. 2009. Errors and error management in image-guided craniomaxillofacial surgery. *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology, and Endodontology*. 2009, vol. 107, n.º 5, págs. 701-715.
- YILMAZ, Alper; JAVED, Omar y SHAH, Mubarak. 2006. Object tracking A survey. *Acm computing surveys (CSUR)*. 2006, vol. 38, n.º 4, págs. 13.
- ZHANG, Z. 2000. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis. Machine Intelligence*. 2000, vol. 22, n.º 11, págs. 1330-1334.