

**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS  
AVANZADOS DEL INSTITUTO POLITÉCNICO NACIONAL**

UNIDAD ZACATENCO

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

SECCIÓN BIOELECTRÓNICA

**“Software para la estimación y registro de  
los parámetros de movimiento en el entrenamiento  
de cirujanos laparoscopistas”**

TESIS

Que presenta

**Ing. Ricardo Mejía Iñigo**

Para obtener el grado de

**Maestro en Ciencias**

**en la Especialidad de  
Ingeniería Eléctrica**

Director de la Tesis:  
Dr. Arturo Minor Martínez



*“Creo que la explicación más simple es que no existe un Dios; nadie creó el universo y nadie dirige nuestro destino. Esto me lleva a una comprensión profunda que probablemente no existe un cielo ni tampoco una vida después de la muerte. Tenemos esta única vida para apreciar el gran diseño del universo y por eso, estoy muy agradecido.” Stephen Hawking.*



*Este trabajo está dedicado a mi familia y a mi abuela Martha; cuyo amor, ejemplo y compañía persisten en mi cada paso que doy día a día.*



# Agradecimientos

A mi familia por darme la confianza de iniciar esta aventura profesional, por apoyar mis decisiones y mis condiciones siempre y en todo lugar: su amor y compañía me hacen dar lo mejor de mi para demostrar lo orgulloso y afortunado que me siento porque estén conmigo.

A mi abuela Martha, quien partió durante mi estancia en la maestría. Tu vives en mi corazón, tu recuerdo y ejemplo permanecen conmigo en cada nuevo proyecto que emprendo.

Al CINVESTAV; muy especialmente al Dr. Arturo Minor Martínez, por procurar siempre de mi bienestar y ayudarme a explotar mis habilidades para ser propositivo e innovador ante las necesidades cambiantes de esta sociedad tan tecnológica y humanista en la que vivimos.

Doy gracias al equipo de trabajo del Laboratorio 5 de la sección de Bioelectrónica junto a José Rosas, Irwin Rosales y Paty Peña quienes trabajaron duramente para disponer del equipo y los medios necesarios para concluir este proyecto de investigación. De la misma forma a Fernando Pérez, Gabriel Vázquez, Judith Guzmán, Néstor Bustos, Rocío Cruz y todos aquellos compañeros de trabajo que hicieron mi estancia en el laboratorio una experiencia agradable.

A Andrea Figueiras, Antonio Valencia, Denisse Trejo, Manuel Pineda, Pedro Alonso, Adriana Esparza, Anayeli Carrasco y todos mis amigos que siempre cuidaron de mi condición en las mejores y peores circunstancias. Gracias a su atención, cuidados, amistad y apoyo me han permitido seguir adelante para procurar mi bienestar y concluir mi posgrado.

---

A Michele Pla, Araceli Santana, Ismael Cosme, Jhonny Aguilar, Cotzarelli Trinidad, Kaled Martínez, Brian Ramírez y el resto de compañeros de generación que siempre compartieron los momentos, tanto profesionales como personales, y que me permitieron ser parte de su vida durante estos años.

A todos mis amigos de la Universidad Autónoma del Estado de México que a través de sus consejos y su ayuda me permitieron culminar este trabajo y me animaron a concluir con éxito mi maestría; muy en particular a Jessica Christel por ser mi ayuda y consejo en la realización de todo proyecto.

A Pamela Alejandra Morales Vilchis, por su infinito amor y soporte durante mi estancia en el Distrito Federal. Gracias a su enorme paciencia, por todos los increíbles experiencias que vivimos y por siempre preocuparte por mi sin importar la distancia.

A CONACYT y todos los mexicanos contribuyentes que a través de sus impuestos me permitieron estudiar el posgrado. Me comprometo a continuar con mi formación como hombre de bien y retribuir a México su inversión para convertirlo en un mejor país.



# Índice general

<b>Agradecimientos</b>	<b>I</b>
<b>Resumen</b>	<b>XIII</b>
<b>Abstract</b>	<b>XV</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Objetivo de la tesis y metodología utilizada . . . . .	2
1.2. Estructura de la tesis . . . . .	3
<b>2. Antecedentes</b>	<b>5</b>
2.1. Historia de la cirugía . . . . .	5
2.1.1. La cirugía en las primeras culturas . . . . .	5
2.1.1.1. Primeras culturas en la cirugía . . . . .	6
2.1.1.2. La cirugía durante la edad media . . . . .	7
2.1.2. La revolución quirúrgica . . . . .	9
2.1.2.1. Anestesia . . . . .	11
2.1.2.2. Procesos de esterilización para cirugía . . . . .	11
2.1.3. Cirugía en el siglo XX . . . . .	13
2.2. Endoscopía . . . . .	14
2.2.1. Historia de la endoscopía . . . . .	15
2.2.2. Tipos de procedimientos endoscópicos . . . . .	15
2.2.3. Cirugía mínimamente invasiva . . . . .	16
2.2.3.1. Laparoscopía . . . . .	17
2.3. Entrenadores laparoscópicos . . . . .	20
2.3.1. Tipos de entrenadores . . . . .	20
2.3.1.1. Entrenadores de realidad virtual . . . . .	21
2.3.1.2. Entrenadores de caja . . . . .	21
2.3.2. Métodos de entrenamiento y evaluación . . . . .	23

---

2.3.2.1. Tipos de pruebas para la evaluación motriz . . . . .	24
2.3.3. Trabajo en el laboratorio 5 de la sección de Bioelectrónica . . . . .	25
2.4. Descripción del método propuesto . . . . .	26
2.5. Resumen . . . . .	27
<b>3. Detección y rastreo del instrumento</b>	<b>29</b>
3.1. Arquitectura del entrenador . . . . .	29
3.2. Disposición de las cámaras . . . . .	30
3.3. Localización del instrumento . . . . .	31
3.3.1. Procesamiento de imágenes . . . . .	34
3.3.1.1. Corrección de la distorsión de la imagen . . . . .	35
3.3.1.2. Transformación del espacio de color <b>RGB</b> a <b>HSV</b> . . . . .	36
3.3.1.3. Segmentación en color por umbralización . . . . .	39
3.3.1.4. Manejo de la oclusión entre marcadores . . . . .	40
3.3.2. Determinación de la posición <b>3D</b> . . . . .	43
3.3.2.1. Modelo geométrico de la escena . . . . .	43
3.3.2.2. Modelo paramétrico para la transformación de <b>2D</b> a <b>3D</b> . . . . .	50
3.3.2.3. Estimación de los parámetros de transformación . . . . .	52
3.3.2.4. Calibración del sistema . . . . .	52
3.3.3. Estimación de las coordenadas <b>3D</b> . . . . .	54
3.3.4. Manejo de la aparición/desaparición de marcadores . . . . .	55
3.3.5. Registro en archivo . . . . .	56
3.4. Resumen . . . . .	57
<b>4. Sistema de registro</b>	<b>59</b>
4.1. Antecedentes del software . . . . .	59
4.2. Software de aplicación . . . . .	60
4.2.0.1. Procesamiento de imágenes . . . . .	60
4.2.0.2. Entorno de desarrollo . . . . .	61
4.2.0.3. Base de datos . . . . .	62
4.2.0.4. Visualización . . . . .	65
4.3. Estimación de parámetros de movimiento . . . . .	66
4.3.1. Parámetros de movimiento . . . . .	66
4.3.2. Parámetros seleccionados para la aplicación . . . . .	69
4.4. Corrección de ángulos . . . . .	70
4.5. Diseño de la aplicación . . . . .	70
4.5.1. Descripción del modelo . . . . .	72

---

4.5.1.1.	Tabla ‘cirujano’ . . . . .	72
4.5.1.2.	Tabla ‘tareas’ . . . . .	73
4.5.1.3.	Tabla ‘parametros’ . . . . .	74
4.5.2.	Navegación de la vista y su descripción funcional . . . . .	75
4.5.2.1.	Ventana de Splash . . . . .	76
4.5.2.2.	Ventana de Login . . . . .	77
4.5.2.3.	Ventana de Nuevo Usuario . . . . .	78
4.5.2.4.	Ventana de Configuración . . . . .	79
4.5.2.5.	Ventana de Inicio . . . . .	80
4.5.2.6.	Ventana de Tareas . . . . .	82
4.5.2.7.	Ventana de Parámetros de Registros . . . . .	84
4.6.	Resumen . . . . .	86
<b>5.</b>	<b>Resultados</b>	<b>87</b>
5.1.	Metodología de análisis de resultados para posicionamiento . . . . .	87
5.1.1.	Discusión de la técnica de posicionamiento . . . . .	89
5.2.	Metodología de análisis de resultados para tiempo de muestreo . . . . .	92
5.2.1.	Discusión los tiempos de muestreo . . . . .	93
5.3.	Análisis de los parámetros de movimiento del instrumento . . . . .	94
5.3.1.	Discusión de la estimación de parámetros . . . . .	94
5.4.	Resumen . . . . .	95
<b>6.</b>	<b>Conclusiones y trabajo futuro</b>	<b>97</b>
6.1.	Conclusiones . . . . .	97
6.2.	Contribuciones . . . . .	98
6.3.	Trabajo futuro . . . . .	99
<b>Anexo:</b>	<b>Código fuente del programa de MATLAB para estimación de parámetros</b>	<b>101</b>
<b>Referencias</b>		<b>107</b>



# Índice de figuras

2.1. Imagen de un cráneo trepanado en el lóbulo parietal. Imagen tomada de Martínez et al. [30]. . . . .	6
2.2. Galeno atendiendo un paciente. Imagen tomada de Martínez et al. [30]. .	7
2.3. Inmovilización para tratamiento de fracturas y luxaciones. Imagen tomada de Martínez et al. [30]. . . . .	8
2.4. <i>'El hombre de Vitruvio'</i> : dibujado por Leonardo da Vinci, donde se describía las proporciones del cuerpo humano a través de una circunferencia y un cuadrado. . . . .	9
2.5. Dibujo de la anatomía muscular humana. Imagen tomada de A. Vesalio [58]. . . . .	10
2.6. W. T. Morton utiliza éter como anestésico en una demostración. Imagen tomada de Martínez et al. [30]. . . . .	12
2.7. Joseph Lister. Imagen tomada de Townsend [54]. . . . .	13
2.8. Conductor lumínico de Bozzini, con diversos espejos. Imagen tomada de A. Sepulveda [49]. . . . .	16
2.9. Imagen de una intervención laparoscópica. Imagen tomada de E.M. Targarona [52]. . . . .	19
2.10. Ejemplos de sistemas entrenadores de realidad virtual. En la imagen se muestran (a) MIST-VR, (b) LapMentor, (c) LapSim y (d) ProMIS. . . . .	22
2.11. Ejemplos de sistemas entrenadores de caja. . . . .	23
2.12. Conjunto de pruebas propuestas en el Departamento de Ginecología en el Centro Médico Universitario de Leiden [12]. Las pruebas consisten en (a) transferencia de un hilo, (b) estiramiento de una liga, (c) formar una letra con una perla y (d) corte de un círculo de caucho. . . . .	25
3.1. Cámara Philips SPC2050NC. Imagen adquirida del folleto del producto [45]. . . . .	30

---

3.2. Colocación de las cámaras con respecto a la estructura física del entrenador. Imagen tomada de F. Perez [43]. . . . .	31
3.3. Diseño estándar de un instrumento laparoscópico completo. . . . .	32
3.4. Imagen de diferentes instrumentos laparoscópicos: (a) tracción, (b) corte, (c) electrocauterización, (d) sutura y (e) disección. Imagen tomada de S. Martínez [30]. . . . .	33
3.5. Vistas de las pinzas de acuerdo a la cada cámara. Se muestran las capturas desde (a) la cámara superior y (b) la cámara lateral. Entrenador de la tesis de doctorado de F. Pérez. . . . .	34
3.6. Ejemplo de la distorsión radial con respecto al centro. Imagen tomada de G. Bradski [7]. . . . .	35
3.7. Corrección de la imagen: (a) imagen original y (b) imagen corregida de acuerdo a los parámetros mostrados. . . . .	36
3.8. Representación geométrica del espacio de color <b>RGB</b> . Imagen adquirida por medio de Color Inspector [3]. . . . .	37
3.9. Representación geométrica del espacio de color <b>HSV</b> . Imagen tomada de <b>Microsoft</b> [33]. . . . .	38
3.10. Etapas del algoritmos de fusión de regiones. En la imagen se muestran (a) las regiones desconexas, (b) la topología de estrella después de unir cada región con el origen, (c) el contorno correspondiente al convex hull y (d) el centroide final de la región corregida. . . . .	42
3.11. Imagen de un cubo posicionado en un punto de origen, resultado de aplicar la transformación mostrada en la ecuación 3.10. Imagen tomada de F. Hanisch [23]. . . . .	45
3.12. Imagen de un cubo trasladado a lo largo del eje $X$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23]. . . . .	46
3.13. Imagen de un cubo rotado $32^\circ$ sobre el eje $Z$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23]. . . . .	46
3.14. Imagen de un cubo escalado con $s_x = 1$ , $s_y = 3$ y $s_z = 2$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23]. . . . .	47
3.15. Imagen de un escalamiento uniforme de un cubo con $s = 2$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23]. . . . .	48

---

3.16. Imagen de un shearing hacia el eje $Y$ con $ShY_2 = 0.5$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23]. . . . .	49
3.17. Representación del modelo de cámara pinhole. Imagen tomada de D.A. Forsyth [19]. . . . .	49
3.18. Disposición de los puntos en el espacio de coordenadas real del sistema.	53
3.19. Representación de los planos visuales de cada cámara con respecto al sistema coordinado del entrenador. Imagen tomada de F. Perez [43]. . .	54
3.20. Algoritmo para la aparición/desaparición de marcadores. . . . .	56
3.21. Vista de la cámara lateral y su división a la mitad de la imagen. Se puede observar que los puntos de inserción son altamente visibles al insertar un instrumento. Entrenador de la tesis de doctorado de F. Pérez. . . . .	57
4.1. <i>Intellisense</i> desplegando los métodos y atributos para una estructura <code>IplImage*</code> de <b>OpenCV</b> . . . . .	62
4.2. Estructura general de un sistema cliente-servidor. Imagen tomada de A. Silberschatz [50]. . . . .	63
4.3. Localización geométrica de las posiciones del trócar con respecto al espacio de referencia. Entrenador de la tesis de doctorado de F. Pérez. .	71
4.4. Diagrama de clases de la clase BD. . . . .	72
4.5. Diagrama relacional de la base de datos para las tres tablas dentro del sistema. . . . .	73
4.6. Vista de la ventana de Splash. . . . .	76
4.7. Vista de la ventana de Login. . . . .	77
4.8. Vista de la ventana de Nuevo Usuario. . . . .	78
4.9. Vista de la ventana de Configuración de Cámaras. . . . .	80
4.10. Vista de la ventana del Menú de Inicio. . . . .	81
4.11. Vista de la ventana de Tareas. . . . .	82
4.12. Vista del resumen de parámetros al terminar la tarea. . . . .	83
4.13. Vista de la ventana de Parámetros de Registros. . . . .	84
4.14. Despliegue de la gráfica para el parámetro <i>percepción de profundidad</i> de la tarea de transferencia. En la pantalla se muestran las opciones de la gráfica para la librería <b>ZedGraph</b> . . . . .	85
5.1. Imagen de los instrumentos de evaluación para la versión anterior del software. Imagen tomada de F. Pérez [43]. . . . .	88

---

5.2. Gráfica del recorrido del cubo de pruebas para dimensiones de 5, 6 y 8 cm. . . . .	91
5.3. Oclusión de los marcadores de ambos instrumentos en su parte media. .	93



# Índice de tablas

2.1. Procedimientos endoscópicos comunes, instrumento, punto de inserción y cavidad de estudio. . . . .	17
2.2. Principales modelos y características de simuladores de realidad virtual.	22
2.3. Principales modelos y características de simuladores de realidad virtual.	23
3.1. Umbrales correspondientes al rango de valores en <b>OpenCV</b> y en el rango de valores real para el espacio de color <b>HSV</b> . . . . .	39
3.2. Coordenadas del mundo real con sus correspondientes coordenadas de proyección sobre la imagen. . . . .	53
4.1. Métricas propuestas por S. Cotin [13]. . . . .	67
4.2. Métricas propuestas por M. Chmarra y los años correspondientes. . . . .	68
4.3. Métricas propuestas por diversos autores en años recientes. . . . .	69
5.1. Coordenadas de prueba de cubo expresadas en milímetros. Tabla tomada de F. Pérez [43]. . . . .	88
5.2. Intervalos de distancias (cm.) al variar la posición del centroide a lo largo de diferentes límites de la imagen. Los valores del rango de distancia corresponden al incremento máximo y mínimo de distancias, que pertenecen al primer y último punto del intervalo del eje de variación.	90
5.3. Métricas con respecto el tiempo de muestreo para diferentes condiciones de oclusión. . . . .	93
5.4. Errores porcentuales para cada uno de los parámetros de movimiento estimados. . . . .	95



# Resumen

El desarrollo tecnológico dentro del área hospitalaria ha conllevado a la creación de nuevos métodos de intervención quirúrgica tales como la cirugía endoscópica. Este tipo de métodos ha sido ampliamente aceptado debido a las ventajas que presentan, como la rápida recuperación y mínima invasión del cuerpo.

La laparoscopia es uno de los procesos endoscópicos más ampliamente conocidos para este tipo de cirugías. Su aprendizaje requiere afrontar ciertos retos para cirujanos novatos como son: el entrenamiento previo a intervención en quirófano, la adaptación a los instrumentos y al entorno visual del espacio de trabajo.

El entrenamiento requiere la ejecución de tareas específicas bajo la supervisión de cirujanos laparoscopistas expertos, quienes evalúan a través de calificaciones subjetivas. Por esto, es necesaria la implementación de nuevos equipos que pueda evaluar su desempeño de manera objetiva, que permitan entrenar a los nuevos cirujanos en condiciones similares a un quirófano y que facilite un seguimiento de su avance.

El presente proyecto propone la implementación de un sistema basado en software para registrar los parámetros de movimiento que realizan los instrumentos de cada cirujano dentro de un entrenador de caja. El entrenador utiliza dos cámaras web colocadas en configuración ortogonal que, a través de procesamiento de imágenes, detectan la posición real de dos instrumentos distintos mediante marcadores de color y registran su posición en un archivo de texto; el archivo es procesado al terminar la tarea para estimar los parámetros de movimiento y guardarlos en una base de datos. Los datos pueden ser recuperados para desplegarse en gráficas y tablas según fecha y hora para auxiliar a la visualización del avance del cirujano.

El sistema provee los componentes visuales para configurar la disposición de las cámaras, estimación de parámetros de movimiento, manejo de cuenta, actualización de la información del usuario y métodos de seguridad para evitar la alteración de los datos.



# Abstract

Technological development for hospital areas has carried out the creation of new surgical intervention methods; for example: surgical endoscopy. This kind of methods has been widely accepted because of the advantages they implicate, such as early recovery time and minimal invasion of the body.

Surgical laparoscopy is one of the most widely known endoscopic processes for this kind of surgeries. Skill learning involves facing some challenges for novice surgeons; for instance, training before interventions in operating room, familiarization with the usage of the instruments and the adaptation to the visual environment of the workspace.

Surgeon training involves the execution of specific task under supervision of expert laparoscopist surgeons, who evaluate these task using subjective scores. For this, it is needed the implementation of new equipments in order they can provide an objective assessment for their performance, where novice surgeons can practice under similar conditions to an operating room and allow easier monitoring of their improvement.

This project propose the implementation of a software-based system to register instruments motion parameters of each surgeon within a box trainer. Laparoscopic trainer uses two cameras placed in orthogonal configuration that, via image processing, uses color markers to detect real-world position of both of the instruments and record their position in a text file. The file is processed at the end of the task to estimate motion parameters and insert them in the database. Data retrieval allow to display the information in graphics and tables, according to date and time, to help visualization of surgeon progress.

The system provides visual components to configure camera disposition, motion parameters estimation, user-account management, user information updates, and security measures to avoid data corruption.



# Capítulo 1

## Introducción

Los procedimientos endoscópicos son técnicas que permiten realizar observaciones en el interior del cuerpo de un paciente mediante un instrumento llamado endoscopio. Existen diferentes procesos endoscópicos específicos dependiendo de las cavidades u órganos que se desean observar; siendo uno de los más comunes la laparoscopia.

La laparoscopia es un proceso quirúrgico endoscópico mínimamente invasivo que se ha popularizado por ser una opción alternativa a la cirugía abierta con ventajas como: disminución de complicaciones post-operatorias en el paciente y una menor permanencia intrahospitalaria. Sin embargo, su enseñanza requiere afrontar ciertos retos para el cirujano novato como lo son el entrenamiento previo a intervenciones en quirófano, la adaptación a la manipulación del instrumental, el reducido espacio operatorio y la pérdida de profundidad debido al uso del laparoscopio.

Existen diferentes equipos que permiten el entrenamiento de cirujanos novatos a través de sistemas de realidad virtual o entrenadores físicos: muchos de ellos evalúan el desempeño según la capacidad o incapacidad del aprendiz de realizar tareas específicas; pese a esto, no proporcionan métricas que permitan evaluar la mejora gradual de las habilidades quirúrgicas. Otros sistemas proporcionan métricas de desempeño para evaluar la habilidad del cirujano, pero utilizan calificaciones de tareas muy específicas del equipo o cuentan con sensores invasivos que modifican la variable que desean medir.

El presente trabajo de tesis propone un software que integra un sistema de visión para la detección de las pinzas laparoscópicas mediante una configuración ortogonal de cámaras y calcula parámetros de movimiento de cada pinza a través de un entrenador de caja. El fin último del trabajo es desarrollar una aplicación que permita el registro de los distintos parámetros de movimiento y que facilite la evaluación por

parte de los instructores a través de una mejor visualización del avance para observar el desempeño del cirujano a lo largo de las pruebas.

## 1.1. Objetivo de la tesis y metodología utilizada

El objetivo del presente trabajo de investigación es el desarrollo del software para la detección y rastreo de las pinzas a través de una configuración ortogonal de cámaras web para la estimación y registro de parámetros de movimiento a lo largo de diferentes tareas.

Las etapas del sistema de rastreo se resumen en los siguientes puntos:

- Cada cirujano crea una cuenta de usuario para registrarse en el sistema de evaluación para efectuar alguna de las tareas disponibles.
- El cirujano inicia sesión y configura las cámaras para determinar la distribución de estas.
- Una vez configurado el sistema, se da inicio a la tarea de entrenamiento deseada.
- Las imágenes de las cámaras son obtenidas a partir de una configuración ortogonal de cámaras web.
- Las posición de las pinzas son localizadas dentro la imagen a través de la segmentación por color de un marcador verde y un marcador azul, cada uno colocado en el extremo cada una de las pinzas.
- Los marcadores son procesados para unificar las regiones cuando existe oclusión entre las pinzas.
- Las coordenadas de los marcadores son introducidas al sistema de ecuaciones para encontrar su posición real en el espacio del entrenador.
- Las coordenadas son registradas en un archivo de texto plano de acuerdo al punto de inserción.
- El archivo de coordenadas es post-procesado para obtener los parámetros de movimiento de cada mano para cada tarea efectuada y son insertados a una base de datos.



- Los resultados son mostrados en forma de gráficas de modo que permitan a un cirujano experto observar la evolución de los parámetros con respecto a cada tarea.

## 1.2. Estructura de la tesis

Esta tesis está organizada en 6 capítulos incluyendo el presente. El capítulo 2 describe la historia de la cirugía y el estado del arte para el entrenamiento de cirujanos laparoscopistas y discute algunos trabajos relevantes publicados en la literatura. El capítulo 3 describe el proceso de detección y rastreo de los instrumentos a través de un modelo matemático de la cámara pinhole y procesamiento de imágenes. El capítulo 4 detalla la infraestructura de software necesaria y las especificaciones de diseño y navegación para el sistema de evaluación llamado *EndoEval*. El capítulo 5 muestra los resultados obtenidos para las diferentes características de evaluación como la resolución, tiempo de muestreo y la diferencia de los parámetros de movimiento estimados por el sistema y los obtenidos mediante **MATLAB**. Finalmente, el capítulo 6 desglosa las conclusiones derivadas del presente trabajo, así como los principales aportes y sugiere varias líneas de trabajo futuro para extender este proyecto.



# Capítulo 2

## Antecedentes

En este capítulo se describen los antecedentes de la cirugía (sec. 2.1, los axiomas fundamentales de la cirugía y la descripción del desarrollo de la endoscopía (sec. 2.2) hasta los procedimientos laparoscópicos (sec. 2.2.3.1).

Se incluyen en este apartado los últimos desarrollos de entrenadores laparoscópicos (sec. 2.3), así como las técnicas utilizadas para el sensado del instrumento (sec. 2.3) y las pruebas utilizadas para su evaluación (sec.2.3.2.1). De la misma forma, se describen los trabajos realizados en el laboratorio 5 de Bioelectrónica (sec.2.3.3) y la propuesta de trabajo para este proyecto (sec. 2.4).

### 2.1. Historia de la cirugía

La cirugía viene de los vocablos griegos *κεῖρ* (*keír*) que significa “mano” y *εργον* (*érgon*) que significa “obra”. Se define como la parte de la medicina que se propone curar las enfermedades, heridas o deformidades por medios manuales, con aparatos, o mediante operaciones que requieren el empleo de instrumentos cortantes [17].

#### 2.1.1. La cirugía en las primeras culturas

La cirugía ha tenido una presencia importante desde las primeras culturas como un procedimiento que ha ayudado a mejorar la supervivencia de las civilizaciones; esto ha permitido adquirir un mejor conocimiento de la estructura y funcionamiento del cuerpo a partir de los diversos descubrimientos realizados por varios investigadores médicos.

### 2.1.1.1. Primeras culturas en la cirugía

Los primeros indicios encontrados de la cirugía fueron encontrados en 1930 por el profesor Zolleki, quien descubrió en los montes Zagros (Irak) los restos de nueve hombres de Neandertal que datan de hace 45000 años: uno de ellos mostraba señales de un brazo derecho amputado [30, p. 1]. Estos hallazgos nos indican que la cirugía, aunque inició como una práctica quirúrgica artesanal y sin ningún conocimiento o método científico, resolvía una necesidad biológica necesaria para la supervivencia de las primeras especies de seres humanos. Sin embargo, desde la edad antigua algunas culturas empezaron a utilizar otro tipo de cirugías como un tipo de intervención para enfrentar algunas patologías.

Los egipcios fueron la primera civilización en realizar intervenciones documentadas. Imhotep, el mayor terapeuta del antiguo egipto, creó tres papiros: de Ebers, de Brusca y de Smith. En Ebers se encuentran descritas algunas técnicas terapéuticas, curaciones para llagas, quemaduras y tumores de cuello; en Smith se hallan diversos casos clínicos documentados minuciosamente. El embalsamamiento y trepanaciones post-mortem (Fig. 2.1) permitió a esta cultura obtener un amplio conocimiento de la anatomía humana [25, p. 9].



**Figura 2.1:** Imagen de un cráneo trepanado en el lóbulo parietal. Imagen tomada de Martínez et al. [30].

En Grecia, Hipócrates (460-356 a.C.) fue el principal representante de la medicina y la cirugía, aportando un compilado de 72 libros y una documentación bastante amplia para la cirugía referente a articulaciones, fracturas, heridas de cabeza, hemorroides y fístulas. Su trabajo indicaba que todas las enfermedades tienen una causa natural, por lo cual tomó un enfoque más científico y refutó totalmente que las patologías fueran una condición mística. Es conocido como el “padre de la medicina”.

La medicina en Grecia tuvo tanto auge que algunos de sus principales exponentes

fueron llamados por el Imperio Romano para tratar heridas sufridas en la guerra (Fig. 2.2). Galeno, médico griego, sirvió como parte de los *Valetudinaria* (hospitales de campaña) y los *Nosocomio* (hospitales para pobres), aportando conocimientos sobre la anatomía y la fisiología del sistema circulatorio [30, p. 3].



**Figura 2.2:** Galeno atendiendo un paciente. Imagen tomada de Martínez et al. [30].

Para la llegada del siglo *I* d.C., India desarrolló instrumentos quirúrgicos para la realización de rinoplastías (operación de nariz) como parte de cirugías reconstructivas. Su educación era constituida a través de la enseñanza en plantas, animales y finalmente en pacientes. De la misma forma, efectuaron intervenciones en fracturas, heridas, abscesos, tumores superficiales, hernias, cesáreas y perineo.

En América, las principales culturas en la cirugía fueron los aztecas y mayas en México, y los incas en Perú. Entre sus aportes se encuentran la inmovilización para tratar las fracturas (Fig. 2.3), el cual se asimilaba al enyesado actual; de la misma forma, utilizaron cabellos para realizar suturas y la herbolaria para la analgesia; por ejemplo el peyote.

En la cirugía, los aztecas y mayas empleaban cuchillos de obsidiana para tratar abscesos y mastitis (inflamación de las glándulas mamarias) a través del drenado y cubrimiento; también operaciones oculares utilizando espinas de maguey. Como parte de ritos religiosos, se realizaban circuncisiones y mutilaciones a lo largo del pene para aquellos que entrarían a la casa sacerdotal [30, p. 4].

#### **2.1.1.2. La cirugía durante la edad media**

Durante la Edad Media ocurrió una decadencia en el desarrollo de la cirugía médica a causa del cristianismo, ya que la iglesia condenaba la disección de cadáveres para su



**Figura 2.3:** Inmovilización para tratamiento de fracturas y luxaciones. Imagen tomada de Martínez et al. [30].

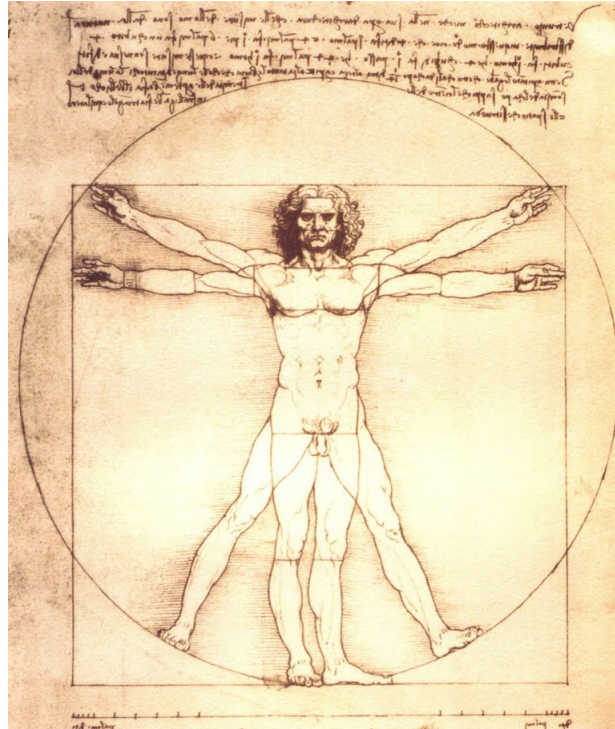
estudio. Por esto, las únicas personas a las que era permitido realizar procedimientos de cuidado era a los barberos, quienes además de cortar cabello se les era permitido realizar apertura para los abscesos, cauterizaciones y curar heridas como cortaduras, fracturas y luxaciones.

Luego de la época de oscurantismo en el conocimiento, la época del Renacimiento tuvo como principal objetivo el recuperar el espíritu científico a través de la observación y el deseo de saber. Artistas tales como Miguel Ángel y Leonardo da Vinci realizaron dibujos de la anatomía humana que describía con gran detalle la escala y arquitectura del cuerpo humano (Fig. 2.4).

Andrés Vesalio se tituló como médico y trabajo para el emperador Carlos V. Publicó su obra llamada "*Tabulae anatomicae sex*" en donde corrigió varios de los diseños realizados por Galeno. De la misma forma, en su obra "*Humani Corporis Fabrica*" hizo un compendio de ilustraciones coloreadas a mano [58], con seis partes dedicadas respectivamente a huesos, músculos, venas, arterias, sistema nervioso órganos abdominales, corazón, pulmones; mientras que la séptima fue exclusivamente dedicada al cerebro (Fig. 2.5).

Ambroise Paré (1510-1590) fue un reconocido miembro de la época renacentista que trabajó como cirujano militar de los ejércitos franceses. Las necesidades de guerra requerían la disminución con contención de las hemorragias durante las batallas; a lo que Paré sustituyó el uso común de aceite hirviendo para la cauterización de heridas por técnicas que empleaban yema de huevo y aceites de rosas, en combinación con técnicas de ligaduras para vasos [54, p. 4].

Aunque la anatomía proveyó de una descripción precisa de los órganos, su ubicación, morfología y disposición, no se tenía una comprensión amplia de su



**Figura 2.4:** 'El hombre de Vitruvio': dibujado por Leonardo da Vinci, donde se describía las proporciones del cuerpo humano a través de una circunferencia y un cuadrado.

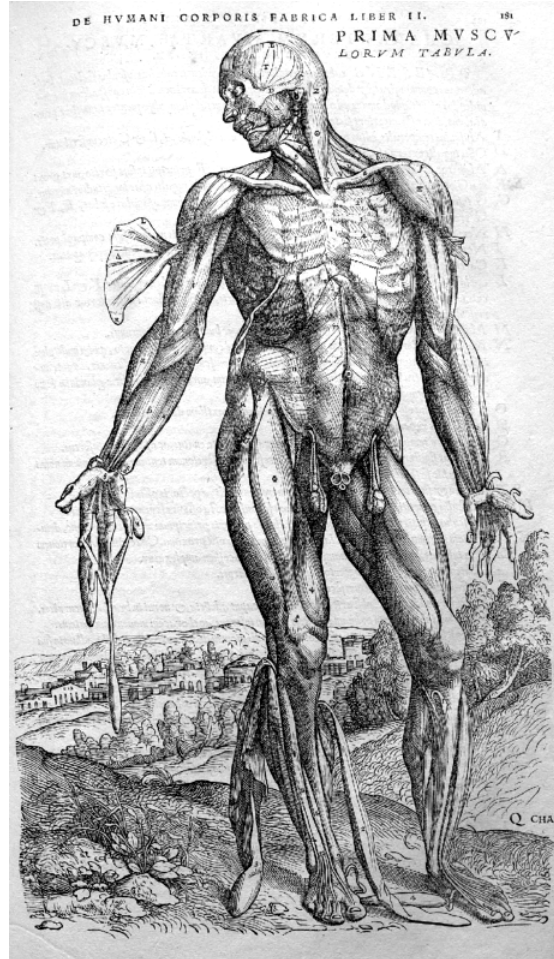
fisiología, es decir, del funcionamiento de cada uno de ellos.

El cirujano inglés, William Harvey (1578-1657), fue uno de los principales investigadores con respecto a la fisiología circulatoria en su obra *Exercitatio anatomica de motu cordis et sanguinis in animalibus* (1628) en donde describió la anatomía y fisiología de aurículas, ventrículos, válvulas cardíacas y arteria pulmonar. En dicho trabajo, describió el ciclo de la circulación mayor, desde la salida de la sangre del corazón hacia la aorta y su regreso por la aurícula derecha [30, p. 7].

En la bacteriología y protozoología, Leeuwenhoek (1632-1723) fue el principal exponente gracias a la construcción de más de 200 microscopios, con los cuales pudo descubrir diversos microorganismos como eritrocitos, espermatozoides y ciliados. De esta manera, se tuvo una clara diferenciación con respecto a las unidades celulares observadas de acuerdo a la funcionalidad del tejido o del fluido de interés.

### 2.1.2. La revolución quirúrgica

La evolución de los descubrimientos médicos a lo largo de la historia de la humanidad pueden observarse sustanciales; sin embargo, fueron avances que



**Figura 2.5:** Dibujo de la anatomía muscular humana. Imagen tomada de A. Vesalio [58].

permitieron conocer descriptiva y funcionalmente el cuerpo humano, aunque no se desarrollaron cuidados para tratar efectivamente las patologías, avances técnicos nuevos o métodos de intervención que fueran efectivos de acuerdo a lo establecido por el método científico. No fue sino hasta finales del siglo XIX, que se establecieron los requisitos fundamentales para considerar que una intervención quirúrgica fuera una opción terapéutica viable.

Los cuatro axiomas clínicos fundamentales [54] son los siguientes:

1. El conocimiento de la anatomía humana.
2. Métodos de control para hemorragias y conservación de las condiciones internas para el intercambio del cuerpo con su ambiente (homeostasis).
3. El uso de anestésicos para llevar a cabo los procedimientos sin dolor.



#### 4. Caracterización de las infecciones y de los procesos de esterilización para la cirugía.

Aunque los dos primeros requisitos quedaron resueltos dentro de los primeros apartados (sección 2.1.1.1 y sección 2.1.1.2), fue hasta finales del siglo XIX para hallar una primera aproximación a los dos restantes.

##### **2.1.2.1. Anestesia**

Uno de los principales problemas para los cirujanos era realizar operaciones sin dolor, esto debido a la inexistencia de anestesia en los orígenes de las primeras intervenciones quirúrgicas. En esos tiempos, los cirujanos tenían que preocuparse más por la velocidad para concluir el procedimiento que por la calidad de su trabajo y los pacientes posponían las cirugías debido al dolor que implicaba.

Los primeros anestésicos fueron los narcóticos o soporíferos, tales como el hachís, mandrágora y el opio; sin embargo, estos no eran suficientes para inhibir el dolor que sufría el paciente. No fue hasta que en 1830 se empezó a conocer el uso de cloroformo, el éter y el óxido nitroso que eran utilizados como gases químicos recreativos entre algunos habitantes debido a los efectos inhalantes producidos luego de su inhalación [54, p. 4].

La primer intervención con anestésicos fue hecha por William T. G. Morton en 1846 utilizando éter sulfúrico a un paciente para extirpación de un pequeño tumor vascular del cuello sin dolor (Fig. 2.6). Esto permitió tener una gran aceptación por parte de los médicos para emplear anestesia inhalada y, aunque facilitó en sobremanera el manejo de pacientes, aún se necesitaba resolver los cuidados higiénicos de los quirófanos para que la cirugía fuera utilizada como procedimiento de rutina.

##### **2.1.2.2. Procesos de esterilización para cirugía**

Aunque la anestesia sirvió como una excelente herramienta para la cirugía, no tuvo tanta importancia como las técnicas de esterilización para los quirófanos ya que, una cirugía sin anestesia podía implicar complicaciones de dolor al paciente, una cirugía sin medios de esterilización tendría como posibles consecuencias la muerte [54, p. 5].

En el siglo XIX se buscaron métodos para controlar la proliferación de microorganismos, estableciendo avances en los dos procesos más importantes: *antisepsia* y *asepsia*.

La *antisepsia* se define como los métodos o procedimientos para disminuir la población de microorganismos de superficies orgánicas. La *asepsia*, por su parte,



**Figura 2.6:** W. T. Morton utiliza éter como anestésico en una demostración. Imagen tomada de Martínez et al. [30].

son aquellos métodos o procedimientos para preservar la ausencia absoluta de microorganismos y sus esporas (conocido como *asepsia*) [30, p. 16].

**Antisepsia** Médicos como Oliver Holmes (1804-1894), profesor de anatomía en Boston, pensaba que los médicos eran los medios de transporta para enfermedades contagiosas; Semmelweis (1818-1865) constató esto al observar que la fiebre puerperal era transmitida a través del contacto de los médicos con cadáveres o individuos infectados y luego con pacientes no infectados; surgiendo el principio del lavado preoperatorio del cirujano.

Spallanzani hizo investigaciones para refutar la teoría de la generación espontánea, teoría que suponía que los microorganismos surgían a partir de una combinación de materiales orgánicos y/o inorgánicos. Louis Pasteur (1822-1895) confirmó los estudios de Spallanzani a través de demostrar que los fermentos eran seres vivos y que, evitando cualquier contacto con el aire, podría preservarse por un tiempo indefinido [30, p.14].

A finales del siglo XIX, el médico inglés Joseph Lister (1827-1912) continuó con los trabajos de Pasteur para demostrar que la atmósfera era el medio de transporte de gérmenes que causaban la putrefacción y, así, sustituir la esterilización por temperatura por algún método químico. A través de estudios sobre cicatrización, determinó un tratamiento con ácido fénico y aceite fénico y, posteriormente, empleó los mismos componentes para cirugías, precediendo un lavado de herida y los instrumentos como primer método de antisepsia [54, p. 5].



*Figura 2.7: Joseph Lister. Imagen tomada de Townsend [54].*

**Asepsia** Aunque los métodos propuestos por Lister eran efectivos, muchos de sus procedimientos eran muy técnicos y complicados para ser utilizados por los cirujanos. En combinación con los métodos, muchos médicos manifestaban diversas molestias por el uso de los químicos utilizados para la esterilización. Sin embargo, diversos grupos quirúrgicos germánicos reemplazaron la antisepsia química por el calor para esterilizar la ropa e instrumentos de los médicos y así evitar las molestias químicas. De esta forma, la antisepsia fue reemplazada por la asepsia, que proveía de mejores resultados ante las infecciones.

Halsted, en 1885, sugirió reemplazar los procesos antisépticos de la piel por el uso de guantes de caucho esterilizables, lo cual disminuía los tiempos de cirugía de manera significativa. Al final, se adoptó este método y es el que aún persiste, exigiendo que el médico nunca debería tocar el instrumental.

### **2.1.3. Cirugía en el siglo XX**

Durante el siglo XX, diversos especialistas se dieron cuenta de la necesidad de crear nuevos métodos instrumentos que facilitaran el diagnóstico, tratamiento, intervención y cuidado de los pacientes. Principalmente, tuvieron gran empuje aquellos descubrimientos que mejoraron la observación de los médicos hacia el interior del cuerpo humano.

La imagenología es una de las áreas médicas que provee varias técnicas de imagen

utilizadas para investigar sistemas específicos, fueron ampliamente desarrollados. Algunos de los métodos de diagnóstico por imagen se encuentran:

- Radiografía: utiliza rayos X, los cuales son generados al aplicar un potencial a un tubo genera una radiación que es incidida hacia un paciente para observar la disposición de huesos y otros tejidos internos.
- Tomografía: esta modalidad emplea varios radiadores de rayos X para obtener imágenes de corte seccional del cuerpo humano, las cuales son reconstruidas a través de una computadora de manera que las anatomía pueda ser interpretada con mayor facilidad.
- Ultrasonografía: es una técnica de diagnóstico que utiliza ondas sónicas de alta frecuencia para generar una imagen, la cual es construida a través del tiempo que tardan los ecos en retornar a la fuente.
- Imágenes de resonancia magnética: esta técnica utiliza el principio de que un campo magnético aplicado externamente causa que los protones en el tejido se alineen en dirección del campo. Al detener el pulso, los protones regresan al equilibrio y producen una señal de radiofrecuencia, la cual puede ser transformada en imágenes por computadora.
- Endoscopía: este tipo de observaciones se hacen a través de un endoscopio, el cual es una cámara con un tubo que es ingresada por medio de cavidades o incisiones para la observación del interior del cuerpo humano.

El uso de métodos de imágenes han permitido mejorar el diagnóstico sin la necesidad de recurrir a la intervención quirúrgica mayor para comprender las patologías de una o más cavidades del cuerpo.

## 2.2. Endoscopía

La endoscopía es un proceso que permite observar el interior del cuerpo utilizando un endoscopio, el cual consiste en una sonda con una cámara rígida o flexible en un extremo de un tubo alargado y delgado. El médico endoscopista manipula el paso de la cámara a través de incisiones o cavidades del cuerpo para observar los órganos internos y, en ocasiones, para procesos quirúrgicos diversos [31].

El examen con un endoscopio es diferente a los estudios por imágenes, como las radiografías y tomografías por computadora debido a que estos últimos capturan

imágenes del interior del cuerpo sin la necesidad de ingresar instrumentos o dispositivos dentro del mismo.

### **2.2.1. Historia de la endoscopía**

La observación del interior de las cavidades humanas ha sido empleada desde la edad antigua por los babilónicos, quienes utilizaban un tubo con espejos para identificar hemorragias dentro del cuello uterino de modo que se utilizara la reflexión óptica para observar el cuerpo. Sin embargo, Caessare Aranzi fue el primero en utilizar luz artificial mediante una botella de vidrio esférica para reflejarlos en el interior de las fosas nasales [49].

En 1804, Philipp Bozzini propuso un diseño de un aparato que consta de una óptica, una fuente lumínica y una parte mecánica que se adapta a la abertura corporal que se desea explorar, el cual constaba de dos partes:

1. Un contenedor de la fuente lumínica (luz de vela) para la óptica
2. Dispositivo mecánico en forma de tubo o conducto, de diversas formas y tamaños para introducirse por las distintas cavidades del cuerpo.

El dispositivo incluía un diseño para ajustar la vela y siempre iluminara a la misma altura pese a su consumo [49], además de varios lentes reemplazables según la parte del cuerpo deseada a observarse por el médico (Fig. 2.8). Este dispositivo se convirtió en el primer endoscopio de uso.

Con el desarrollo de la tecnología, se mejoraron los métodos de observación a través del uso de equipos tales como las cámaras de video y las pantallas para que los médicos pudieran tener una mejor percepción visual; además de la variedad de endoscopios.

### **2.2.2. Tipos de procedimientos endoscópicos**

Hay diferentes tipos de endoscopios: la mayoría de ellos son tubos huecos y delgados que permiten al médico observar el interior del cuerpo; la mayoría tiene una fuente de luz y algunos cuentan con una pequeña cámara de video en el extremo, la cual transmite imágenes a una pantalla de computadora. Algunos tubos de las cámaras son rígidos, mientras que otros son flexibles.

Los endoscopios pueden variar en longitud y forma: existe un endoscopio nuevo que es lo suficientemente pequeño como para ser tragado, el cual



**Figura 2.8:** Conductor lumínico de Bozzini, con diversos espejos. Imagen tomada de A. Sepulveda [49].

transmite inalámbricamente las imágenes. Cada tipo de endoscopio está diseñado especialmente para observar cierta parte del cuerpo. Dependiendo del área del cuerpo que se desea examinar, el endoscopio puede introducirse a través de diversas cavidades o pequeñas incisiones en la piel. El procedimiento quirúrgico y el endoscopio reciben un nombre particular de acuerdo a su punto de inserción.

Algunos de los procedimientos endoscópicos más comunes [51] son:

### 2.2.3. Cirugía mínimamente invasiva

El término *cirugía mínimamente invasiva* o también conocido como *cirugía de mínimo abordaje* se utiliza para nombrar le conjunto de técnicas diagnósticas y terapéuticas que por visión directa, endoscópica o algún otro método de imagen, utiliza vías naturales o incisiones mínimas para introducir herramientas y actuar en diferentes partes del cuerpo [21].

En su generalidad, los procesos endoscópicos implican la observación del cuerpo; sin embargo, algunos tipos de cirugía emplean los métodos de observación a manera de poder maniobrar instrumental quirúrgico dentro del cuerpo sin la necesidad del uso de cirugía abierta (cirugía mayor). Uno de los procedimientos quirúrgicos de mínima invasión más conocidas es la laparoscopia [21].

Nombre del procedimiento	Tipo de endoscopio	Punto de inserción	Parte del cuerpo a observar
Artroscopía	Artroscopio	Cortes en la piel	Articulaciones
Broncoscopía (flexible)	Broncoscopio	Boca o nariz	Traquea y bronquios
Colonoscopia, endoscopia inferior	Colonoscopio	Ano	Colon e intestino grueso
Cistoscopia, cistouretoscopia	Cistoscopio	Uretra	Vejiga
Enteroscopia	Enteroscopio	Boca o ano	Intestino delgado
Esofagogastroduodenoscopia, endoscopia superior, panendoscopia, gastroscopia	Esofagogastroduodenoscopio	Boca	Esófago, estómago y duodeno
Histeroscopia	Histeroscopio	Vagina	Dentro del útero
Laparoscopia, endoscopia peritoneal	Laparoscopio	Cortes en el abdomen	Cavidad abdominal y pélvica
Laringoscopia	Laringoscopio	Boca o nariz	Laringe
Mediastinoscopia	Mediastinoscopio	Cortes encima del esternón	Mediastino (espacio entre pulmones)
Proctosigmoidoscopia	Sigmoidoscopio (flexible)	Ano	Recto y colon sigmoide
Toracoscopia, pleuroscopia	Toracoscopia	Cortes en el pecho	Espacio entre pulmones y pared torácica

**Tabla 2.1:** Procedimientos endoscópicos comunes, instrumento, punto de inserción y cavidad de estudio.

### 2.2.3.1. Laparoscopia

La cirugía laparoscópica viene del término “laparoscopia” que se origina del griego “*laparos*” (flanco o porción de abdomen comprendido entre las costillas y la cadera) y “*skopein*” (examinar) y se refiere al examen visual de la cavidad abdominal por medio

del laparoscopia (endoscopio iluminado) [52].

Un proceso quirúrgico laparoscópico se constituye en los siguientes pasos elementales [16]:

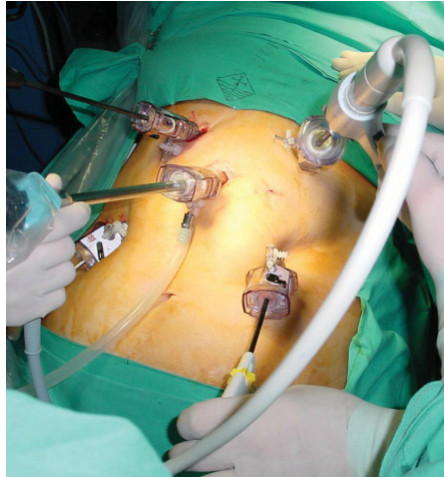
1. **Neumoperitoneo:** se le conoce así a la expansión del volumen intra-abdominal mediante insuflamiento de gas para la inserción de instrumental y manipulación de los órganos, para lo que se requiere que se mantenga cierta presión dentro del abdomen con respecto a su entorno. La insuflación de un gas inerte en la cavidad peritoneal se realiza a través de una cánula con protección para evitar el daño visceral. Sin el neumoperitoneo es imposible operar, porque no habría espacio para el desplazamiento de los instrumentos y manipulación de los órganos.
2. **Instalación de trócares:** se realiza la inserción de las cánulas o trócares en el abdomen del paciente. El primer trocar es generalmente instalado por el ombligo a través de una incisión radial o en el fondo de este. Luego de esto, debe hacerse presión suave y constante, con movimientos rotatorios de la mano hasta activar el sistema de protección que tienen los trócares.
3. **Procedimiento quirúrgico:** se realiza primero una observación de análisis y por cuadrantes en sentido de las manecillas del reloj, iniciando a partir de las 12. De ahí se ubican los trócares en la posición adecuada para insertar los instrumentos según el tipo de intervención.
4. **Exsuflación y retiro de los instrumentos:** una vez terminada la cirugía, se limpia todo líquido y se remueve el gas remanente; los instrumentos son extraídos, los trócares retirados y se suturan las incisiones hechas para su inserción.

El proceso requiere la inserción de una cánula a través de la pared abdominal, para lo que se necesita la distensión de la pared abdominal con un gas y un conjunto de entradas de pequeño diámetro (Fig. 2.9) por donde se introduce el instrumental para la observación y manipulación de los órganos [59].

Para llevar a cabo una intervención laparoscópica son necesarios un conjunto de instrumentos [30] tales como:

- Laparoscopia: el sistema óptico que obtiene la imagen anatómica del área a operar y la transmite por medio de una cámara; cuenta con una fuente de luz.





**Figura 2.9:** Imagen de una intervención laparoscópica. Imagen tomada de E.M. Targarona [52].

- **Monitor:** son pantallas a color que generalmente tienen dimensiones de 20 a 25 pulgadas, con una resolución igual o mayor a la de la cámara para conservar la nitidez de la imagen.
- **Cámara:** el dispositivo debe tener una buena resolución para mejorar la visión del cirujano y que permita realizar acercamientos para determinadas regiones. El equipo debe de encontrarse esterilizado para la inserción en el cuerpo. Durante la operación debe ser manipulado por un ayudante que será guiado por las instrucciones del cirujano.
- **Fuente de luz:** las fuentes de luz se clasifican en calientes y frías, y su clasificación depende de la temperatura que genera la incidencia de la luz en un tejido humano. De este modo, se utilizan fuentes de luz fría tales como halógeno y xenón.
- **Insuflador:** internacionalmente se utiliza el dióxido de carbono debido a ser un gas fisiológicamente inerte y fácilmente desechable por el organismo en caso de que este llegara a ser absorbido por el cuerpo.

La laparoscopia ha tenido un gran auge debido a su versatilidad de emplearse tanto como una técnica de diagnóstico así como quirúrgica de mínima invasión. Este método es el principal para la realización de colecistectomía, coledoscopías, mayoría de cirugías biliares, apendicectomías, hernias inguinales, laparoscopia diagnóstica, cirugía ginecológica, linfomas, tumores benignos, entre otros [16, p. 48] [49].

Las cirugías mínimamente invasivas presenta un conjunto de ventajas con respecto a la cirugía abierta, debido a que reduce la respuesta inflamatoria del sistema, disminuye el dolor post-operatorio del paciente, minimiza complicaciones en la herida, mejora la estética de la cicatriz y reduce la estancia intrahospitalaria [21]; sin embargo, la laparoscopia presenta un mayor número de complicaciones al cirujano, como las dificultades de adaptación de la percepción espacial en 3D (vista humana) a 2D (monitor), pérdida de la profundidad, difícil manejo del sangrado, es técnicamente más exigente, requiere de un aprendizaje gradual, mayor fatiga y el riesgo de contaminación tumoral [5, 6, 21, 52].

Un cirujano mal entrenado es considerado un factor de riesgo para una cirugía laparoscópica [37, p. 5]; esto debido a que la cirugía endoscópica requiere conocer las herramientas, el ajuste de la visión del campo operatorio en una pantalla y la conservación de la homeostasis del cuerpo [49].

## 2.3. Entrenadores laparoscópicos

Las preparación de un cirujano laparoscopista novato presentan nuevos obstáculos en su entrenamiento como es la superación del inicio de la curva de aprendizaje, presiones económicas, de tiempo y el correcto aprendizaje de las habilidades operativas para evitar poner en riesgo a los pacientes [9, 11, 39]. Sin embargo, en algunos lugares como Europa, la implementación de la legislación Europea (Directiva de Tiempo de Trabajo Europeo, en inglés: *European Work Time Directive*), reduce el tiempo de horas de entrenamiento; esto implica que el método de entrenamiento debería proveer los máximos resultados para el aprendizaje del usuario [8].

El desarrollo de la tecnología ha facilitado el diseño de diversos equipos que permiten el entrenamiento del cirujano previo a una intervención quirúrgica.

### 2.3.1. Tipos de entrenadores

Existen distintos tipos de entrenadores laparoscópicos para cirujanos; cada uno con características particulares pero que generalmente pueden clasificarse en dos tipos:

- Entrenadores de caja (*box-trainers*)
- Entrenadores de realidad virtual

### 2.3.1.1. Entrenadores de realidad virtual

Los **entrenadores de realidad virtual** utilizan un manipulador, que usualmente posee algún tipo de dispositivo sensible por el sistema, el cual puede ser posicionado en cualquier parte del área de trabajo y que permiten calificar diversos entrenamientos independientes de tareas quirúrgicas en un escenario simulado de cirugía.

Los sistemas simuladores de realidad virtual ofrecen varias ventajas, tales como el entrenamiento en ambientes controlados y que no requiere la presencia de un médico supervisor para la realización de las tareas. El sistema provee al usuario de retroalimentación mediante calificaciones numéricas por objetivos (tareas); sin embargo, presenta la desventaja de tener un alto costo [8], así como que algunos cirujanos no consideran necesario un entorno simulado para aprender [41].

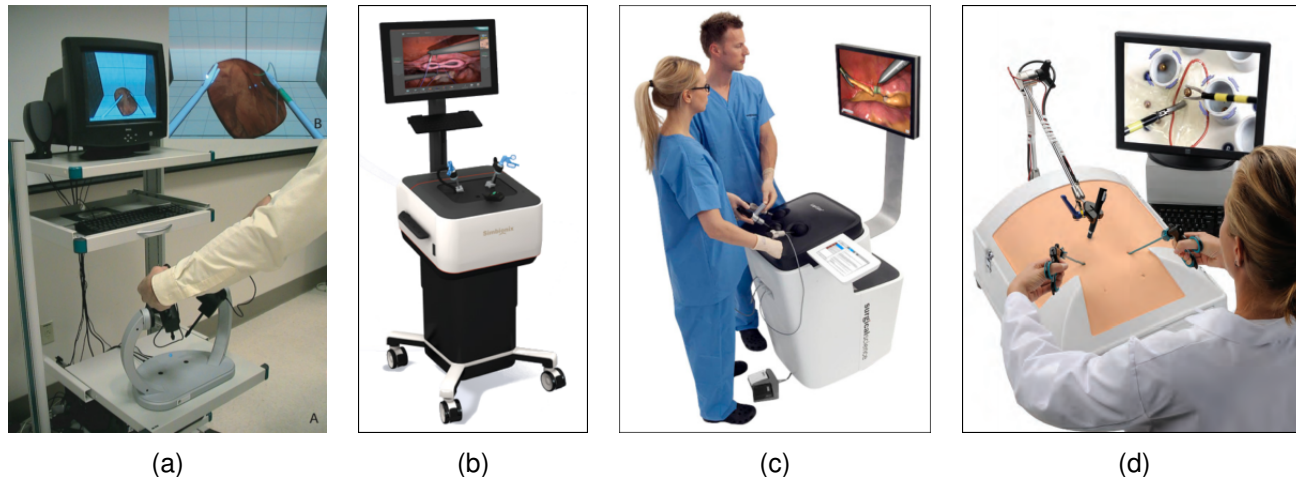
Existen diversos equipos de entrenamiento de realidad virtual, como son:

- **MIST-VR**: diseñado por *The Royal College of Surgeons of England* para evaluar la destreza manual de los cirujanos (Fig. 2.11.a). Tiene un espacio de trabajo de  $10\text{ cm}^3$  y permite simular tareas de colecistectomía y adaptar instrumentos para otros procedimientos, como artroscopios o endoscopios [60].
- **LapMentor**: es un sistema de entrenamiento laparoscópico basado en realidad virtual desarrollado por *Simbionix Corp* (Fig. 2.11.b). El software ofrece soluciones para tareas como agarre, corte, manipulación a dos manos, electrocauterio y transferencia de objetos.
- **ProMIS**: es un simulador de realidad aumentada desarrollado por *Haptica*(Fig. 2.11.c). El entorno tiene un maniquí en forma de torso y se manejan las tareas de localización, coordinación, posicionamiento de objetos, manipulación de tejido y disección.

En la tabla se mencionan algunos de los sistemas más relevantes de realidad virtual:

### 2.3.1.2. Entrenadores de caja

Los **entrenadores de caja** proveen un contenedor que simula la cavidad de cirugía endoscópica y que incluye una cámara que puede ser conectada a una computadora o monitor. En estos equipos, el instrumento puede ser localizado dentro del espacio de trabajo pero requiere de un sistema de detección; el cual puede usar una técnica de



**Figura 2.10:** Ejemplos de sistemas entrenadores de realidad virtual. En la imagen se muestran (a) MIST-VR, (b) LapMentor, (c) LapSim y (d) ProMIS.

Simulador	Módulo de aprendizaje	Escenarios	Entorno	Retroalimentación de fuerza	Referencias
MIST-VR	Habilidades básicas Nefrectomía	Tareas simples Procedimental	No anatómico Anatómico	No Sí	[20]
LapMentor	Básico/esencial/sutura Bilis/Hernia/Ventral/Gástrico/Ginecología	Tareas simples y avanzadas Procedimental	No anatómico Anatómico	Sí -	[61]
LapSim	Habilidades básicas Colecistomía/Ginecología/Apéndice	Tareas simples y avanzadas Procedimental	Ambos Anatómico	Soportado	[18]
ProMIS	Habilidades básicas Procedimental	Tareas simples Procedimientos	No anatómico Simulador híbrido	Interacción realista -	[56]

**Tabla 2.2:** Principales modelos y características de simuladores de realidad virtual.

rastreo de tipo electromagnéticas, mecánicas, ultrasónicas u ópticas [41]. Este tipo de sistemas, se solicita que el cirujano realice tareas, de los cuales se obtiene un registro

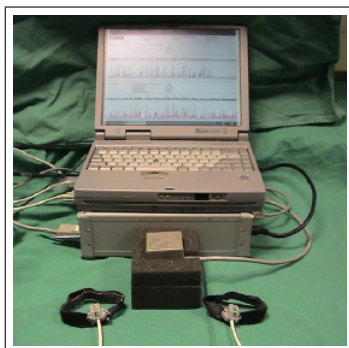
La ventaja de estos entrenadores es la facilidad de utilizar cualquier tipo de pinza laparoscópica, tener una mejor háptica y un menor costo; con la desventaja de que frecuentemente, la técnica utilizada para la localización del instrumento no es tan precisa y a veces requiere de un supervisor para el entrenamiento, complicando su evaluación motriz [8, 39, 41].

A continuación, se resumen algunos entrenadores de caja:

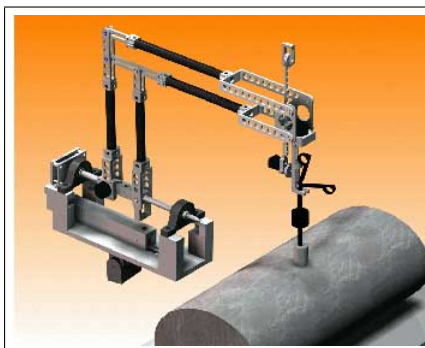
- **ICSAD:** (Imperial College Surgical Assessment Device) es un equipo que combina un sistema electromagnético de rastreo que se sujeta a cada mano del cirujano. El sistema recopila las coordenadas cartesianas con resolución de 1 mm. a una velocidad de 20 Hz [14].
- **BlueDRAGON:** este sistema utiliza dos mecanismos pasivos de cuatro barras sujetas a las herramientas de endoscopia. Los mecanismos transmiten la

rotación y el posicionamiento de la herramienta y son adquiridos mediante sensores de posición, sensores de fuerza en tres ejes y sensores de contacto [48].

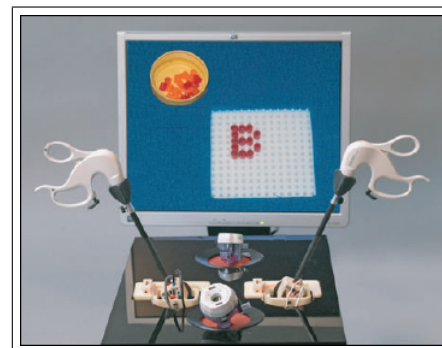
- **TrEndo**: es un sistema de seguimiento para distintos instrumentos a través de un gimbal de dos ejes y sensores de ratón óptico con alta resolución, que puede adquirir datos a una velocidad de 100 Hz [11].



(a) MIST-VR



(b) LapMentor



(c) LapSim

**Figura 2.11:** Ejemplos de sistemas entrenadores de caja.

En la tabla se muestran algunas características de estos equipos

Sistema entrenador	Sistema de rastreo	Área de aplicación	Métricas registradas	Portabilidad	Referencias
ICSAD	Electromagnético	Entrenador, Quirófano	Movimiento de manos	Sí	[14]
BlueDRAGON	Mecánico	Entrenador, Animales	Movimiento, Fuerza	No	[48]
TrEndo	Óptico	Entrenador, Realidad Virtual	Movimiento	Sí	[11]

**Tabla 2.3:** Principales modelos y características de simuladores de realidad virtual.

El potencial de los entrenadores ha demostrado una fácil transferencia de las habilidades adquiridas en la simulación hacia el quirófano [8]; sin embargo, el entrenamiento y la evaluación del residente debe ser cuidadosamente vigilada de manera que el entrenador permita medir los parámetros útiles para evaluar su desempeño [39].

### 2.3.2. Métodos de entrenamiento y evaluación

Durante los últimos años se han evaluado diferentes métodos de entrenamiento que permitan al cirujano novato lograr un mejor aprendizaje: algunos van desde evaluar las

habilidades de utilizar únicamente un sólo tipo de simuladores (realidad virtual o de caja) con respecto a la combinación de ambos [8], mientras otros autores revisan si el entrenamiento debe involucrar el uso inicial de un tipo de entrenador para, después de cierto periodo de práctica, migrar hacia el uso del otro tipo [9]. De la misma forma, se ha sugerido cambiar los tiempos de entrenamiento para que el residente cuente con un tiempo que le permita descansar y asimilar las habilidades recién practicadas [29]. En cualquier caso, es necesario el uso de métricas que permitan realizar un monitoreo objetivo de la evolución de sus habilidades para poder evaluar la mejora del cirujano.

Las métricas que se utilizan se pueden clasificar en 3 tipos:

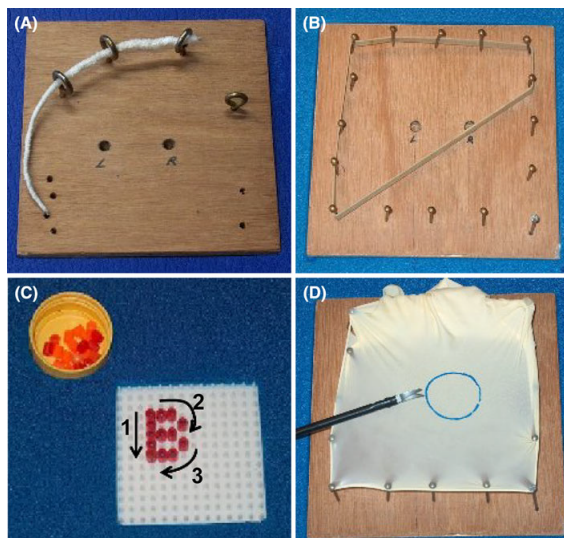
1. Parámetros de análisis de movimiento: cuantitativamente evalúan la calidad de algún parámetro de movimiento con respecto a alguna cualidad en su desplazamiento (velocidad, distancia recorrida, suavidad, etc.)
2. Tiempo: recorrido de la pinza desde el inicio de la tarea hasta el cumplimiento del objetivo o hasta la expiración del tiempo límite para su realización.
3. Evaluación del desempeño de la tarea: brindan una calificación cuantitativa de que tan bien fue realizado el objetivo. Algunas tareas establecen un tiempo límite para realizar la tarea.

De las tres categorías generalmente se excluye la evaluación de la tarea como parámetro generalizable de habilidades, ya que esta depende del tipo de tarea a realizar y de como sea evaluada por el investigador que la propone [8, 29]. De esta manera, los parámetros de movimiento y el tiempo tienden a ser las características principalmente empleadas para evaluar durante el entrenamiento [11, 42].

### **2.3.2.1. Tipos de pruebas para la evaluación motriz**

El conjunto de pruebas tiene como fin evaluar la realización de tareas básicas dentro del quirófano, las cuales van desde transferencias, manipulación de tejido, cortes, nudos, cauterización, entre otros. Los materiales para realizar las pruebas son definidos por cada autor y deben de permitir fácilmente la transferencia de las habilidades a un procedimiento médico (Fig. 2.12).

Independientemente de los sistemas entrenadores, existen algunos protocolos que proponen tareas y tiempos límites para la ejecución de la tarea; por ejemplo el *Sistema Inanimado para el Entrenamiento y Evaluación de Habilidades Laparoscópicas McGill* (en inglés, *McGill Inanimate System for Training and Evaluation of Laparoscopic Skills: MISTELS*). Este protocolo se ha convertido casi en un estándar para muchas pruebas



**Figura 2.12:** Conjunto de pruebas propuestas en el Departamento de Ginecología en el Centro Médico Universitario de Leiden [12]. Las pruebas consisten en (a) transferencia de un hilo, (b) estiramiento de una liga, (c) formar una letra con una perla y (d) corte de un círculo de caucho.

de entrenamiento; sin embargo, el conjunto de pruebas sólo utiliza parámetros de tiempo y calificaciones subjetivas para cada tarea y sin proponer algún parámetro de medición objetivo para la realización de la tarea [57].

Por lo anterior, decimos que los entrenadores laparoscópicos deben de proveer de un espacio de trabajo cómodo para el cirujano, el cual permita transferir sus habilidades fácilmente del entrenador al quirófano. Para evaluar las aptitudes del aprendiz, es necesario un sistema de sensado que permita realizar el registro de los movimientos ejecutados durante la tarea de práctica y en donde los instrumentos de medición no afecten la ejecución del cirujano novato.

Una vez calculados los parámetros de movimiento, el software debe ser capaz de post-procesar las señales adquiridas durante la prueba para obtener parámetros objetivos que califiquen el desempeño del cirujano y manifiesten una mejoría en la realización de los ejercicios.

### 2.3.3. Trabajo en el laboratorio 5 de la sección de Bioelectrónica

En el laboratorio se han desarrollado dos equipos basados en los entrenadores de caja . En el primero se tiene un rastreo de la posición a través de un conjunto de acelerómetros que se encontraban en las inserciones del equipo; el cual permitía una localización de la pinza, pero implicaba una mayor factor de fricción al manipular los instrumentos [47].

Un segundo trabajo elaborado como proyecto de doctorado se basó en el uso de procesamiento de imágenes para medir la posición utilizando como unidades de referencia los píxeles en los que se encontraba el instrumento; el cual fue complementado después con un sistema de análisis con el que se calificaba el desempeño de la tarea [28].

En su última versión, se propuso un sistema de rastreo para las pinzas basado en una distribución de cámaras ortogonales. Este proyecto utiliza marcadores de color en la punta de cada instrumento para su identificación. Las imágenes adquiridas son procesadas y, a través de un sistema de triangulación, permite calcular la posición 3D y los ángulos de cada instrumento y guardarlos en un archivo de texto.

El proyecto en su estado del arte presenta los siguientes detalles:

- La técnica para calcular ángulos es susceptible a ruido.
- La aplicación no permite manejar oclusiones de los marcadores de color de cada pinza, de modo que si la visibilidad de los marcadores se obstruye, se podría ocasionar errores en la detección.
- Para calcular parámetros a partir de los registros es necesario un post-procesamiento del archivo de texto, lo que requiere de una mayor inversión de tiempo y complica la evaluación de los cirujanos novatos por parte de los médicos instructores.
- No se tiene un registro de historial por usuarios para cada practicante, de modo que para evaluar a cada cirujano se requiere coleccionar los archivos de texto con las coordenadas en una misma carpeta.

## 2.4. Descripción del método propuesto

El presente proyecto propone la mejora de la aplicación del último software del entrenador del laboratorio a manera de ofrecer solución a las deficiencias que presentaba.

Se propone diseñar una aplicación que permita:

- Obtener métricas confiables para posición y ángulos, el cual sea capaz de soportar la oclusión de las pinzas.
- Llevar un control de registros basado en el nombre del usuario, fechas y hora de los entrenamientos.



- Realizar el post-procesamiento de los registros para obtener un conjunto de parámetros de movimiento para cada mano y que permita almacenarlos en una base de datos, de manera que permita una visualización de la evolución de los datos.

## **2.5. Resumen**

Este capítulo describió el desarrollo de la medicina, sus principios fundamentales y sus avances tanto científicos como tecnológicos para la intervención quirúrgica de diversos tipos. Particularmente, se describió el proceso e infraestructura necesaria de la cirugía laparoscópica y los medios de entrenamiento para esto.

Se describieron también algunos de los sistemas y metodologías más utilizados para entrenar a los cirujanos novatos. De ellos, se describieron las características de los principales sistemas y se proponen las características de nuestro proyecto.



# Capítulo 3

## DetECCIÓN Y RASTREO DEL INSTRUMENTO

En este capítulo se describe la estructura del entrenador de caja para conocer las cualidades de nuestro espacio de trabajo (sec. 3.1). A continuación se describe la disposición y las características de las cámaras montadas en base a una configuración ortogonal dentro del entrenador (sec. 3.2).

Se describe todo el procesamiento de imágenes (sec. 3.3), tanto la corrección, umbralización y detección de la posición en la imagen del objeto. Entre los pasos subyacentes, se menciona la técnica para el manejo de oclusiones en la parte media de los marcadores (sec. 3.3.1.4). Una vez obtenida la correcta posición en la imagen se menciona todo el desarrollo para conseguir el modelo matemático para encontrar la ecuación para hallar las coordenadas **3D** (sec. 3.3.3).

Finalmente, se describen los métodos de solución para cuando los instrumentos se intercambian en los puntos de inserción (3.3.4) y la estructura de los datos en el archivo de registro.

### 3.1. Arquitectura del entrenador

El diseño del entrenador está basado en los primeros prototipos del proyecto, que consistía en una cavidad semicilíndrica con cuatro puertos de acceso para los instrumentos que simulan un espacio abdominal en quirófano, el cual rodeaba una base plana que se empleaba como piso para colocar algunos objetos para su manipulación [47]. El diseño también incluía una cámara móvil que era usada como la cámara insertada a través de la pared abdominal, la cual permite enfocar los objetos en el interior del entrenador y proyectarlo a una pantalla. Este primer proyecto utilizaba acelerómetros para medir el movimiento de los instrumentos.

Conforme el entrenador siguió bajo revisión, se modificaron los métodos mecánicos de sensado por el uso de cámaras para adquirir imágenes, las cuales permitían identificar la posición de los objetos de la tarea y poder dar una calificación de acuerdo a su posición [28]. Para esto, se tuvo que mejorar la óptica de la cavidad interna del entrenador a través de la instalación de leds para mejorar la iluminación para la captura de las cámaras.

Para la última versión del entrenador se conservan el diseño, los puntos de acceso, la disposición de la cámara y la instalación de los leds para poder mejorar la visibilidad de los objetos en el interior. El espacio de trabajo dispone que los objetos de trabajo se deben colocar distalmente de la óptica, de esta manera se forma un diamante virtual de trabajo, semejante al recomendado en las cirugías de la especialidad [34, 35].

### 3.2. Disposición de las cámaras

El sistema entrenador cuenta con dos cámaras web marca *Philips*® modelo *SPC2050NC* con una resolución de video de 2 MPíxeles y velocidad de hasta 90 **FPS**, con el objetivo de adquirir un alto rango de muestreo para cada toma.

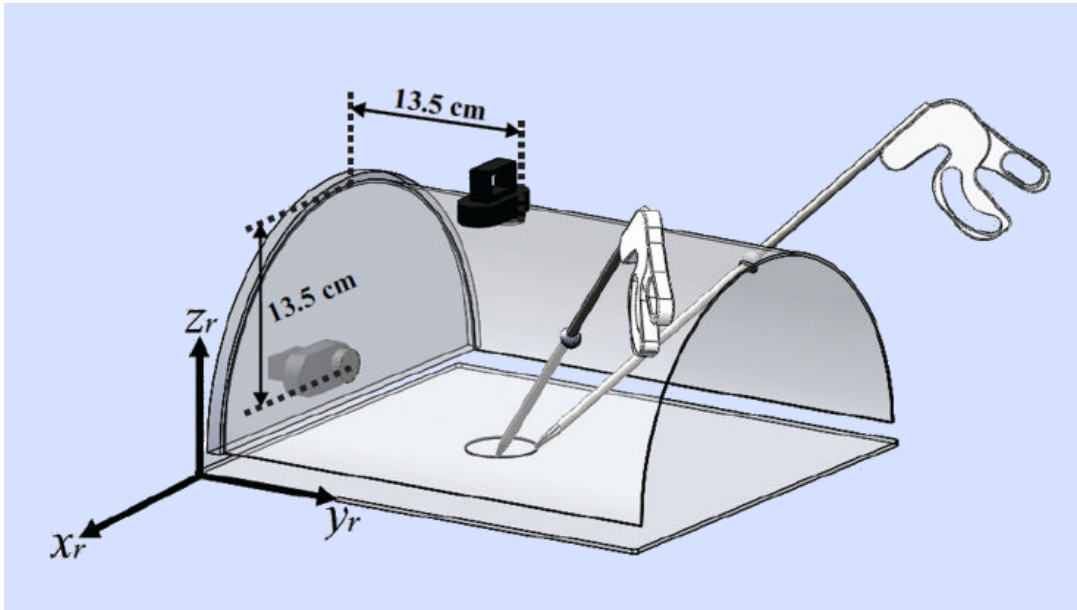


**Figura 3.1:** Cámara Philips SPC2050NC. Imagen adquirida del folleto del producto [45].

Las cámaras se encuentran instaladas en disposición ortogonal de la siguiente manera:

1. La primera se encuentra montada sobre la pared superior a una distancia de 13.5 cm. de la pared posterior.
2. La segunda se encuentra montada sobre la pared posterior a una distancia de 13.5 cm. de la pared superior.

El sistema propone un sistema de referencia cartesiano de 3 dimensiones referenciado de acuerdo a un punto de origen ubicado dentro del entrenador. La distribución de las cámaras y el sistema coordenado del entrenador se pueden observar en la Fig. 3.2.



**Figura 3.2:** Colocación de las cámaras con respecto a la estructura física del entrenador. Imagen tomada de F. Perez [43].

El propósito de las cámaras es capturar la posición del instrumento a partir de las imágenes adquiridas por cada cámara en el espacio visual de trabajo; de modo que la primera cámara captura los movimientos realizados en el plano  $XY$ , mientras que la segunda cámara captura los movimientos en el plano  $XZ$ .

### 3.3. Localización del instrumento

En laparoscopia se utilizan diversos instrumentos que permiten tareas quirúrgicas dentro del cuerpo, los cuales son reutilizables luego de un proceso de esterilización [37].

Un instrumento laparoscópico generalmente se compone de tres elementos:

1. **Mango:** parte de donde el cirujano realiza la sujeción y manejo del instrumento aplicando fuerza sobre el mecanismo del mango. Existen diferentes tipos de mangos de acuerdo a la tarea que se desea realizar

2. **Vástago:** es un eje de aproximadamente 30 cm. de largo que transmite la fuerza aplicada sobre el mango para manipular el movimiento de la punta.
3. **Punta:** existen diferentes arquitecturas de la punta de acuerdo al tipo de tarea que se desea realizar. La punta es la parte de la pinza que realiza la manipulación quirúrgica.

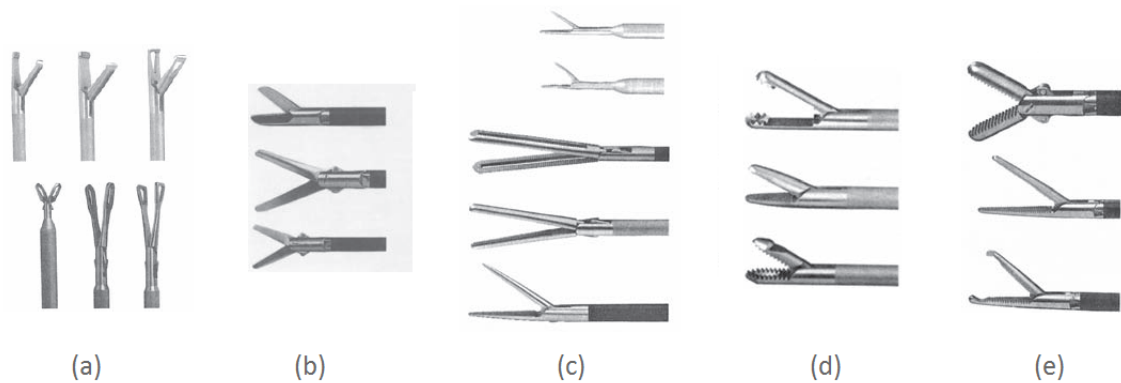
Generalmente la pinza obedece un diseño de tijeras para la parte del mango con una disposición de  $90^\circ$  (Fig 3.3); sin embargo, la punta y el mango del instrumento depende mucho de la tarea que se desea realizar con la pinza.



*Figura 3.3: Diseño estándar de un instrumento laparoscópico completo.*

Las principales tareas realizadas con las pinzas son:

- **Sostén o tracción:** sirven para sujetar, mover o inmovilizar los órganos y tejidos que se encuentra bajo intervención.
- **Corte:** son pinzas que generalmente tienen punta y que permiten separar los tejidos. Generalmente tienen una punta con forma curva.
- **Electrocauterización:** son equipos eléctricos que permiten la sutura del tejido a través de cauterización, el cual es inducido mediante una corriente eléctrica aplicada sobre el instrumento.
- **Sutura:** la sutura se efectúa de la misma forma que la cirugía abierta, pero utilizando instrumentos que permitan una mejor sujeción de las agujas para la realización de nudos intracorpóreos.
- **Disección:** este tipo de dispositivo pueden utilizarse como sujetadores del tejido y además conectarse a las unidades electroquirúrgicas para realizar la disección y separación.



**Figura 3.4:** Imagen de diferentes instrumentos laparoscópicos: (a) tracción, (b) corte, (c) electrocauterización, (d) sutura y (e) disección. Imagen tomada de S. Martínez [30].

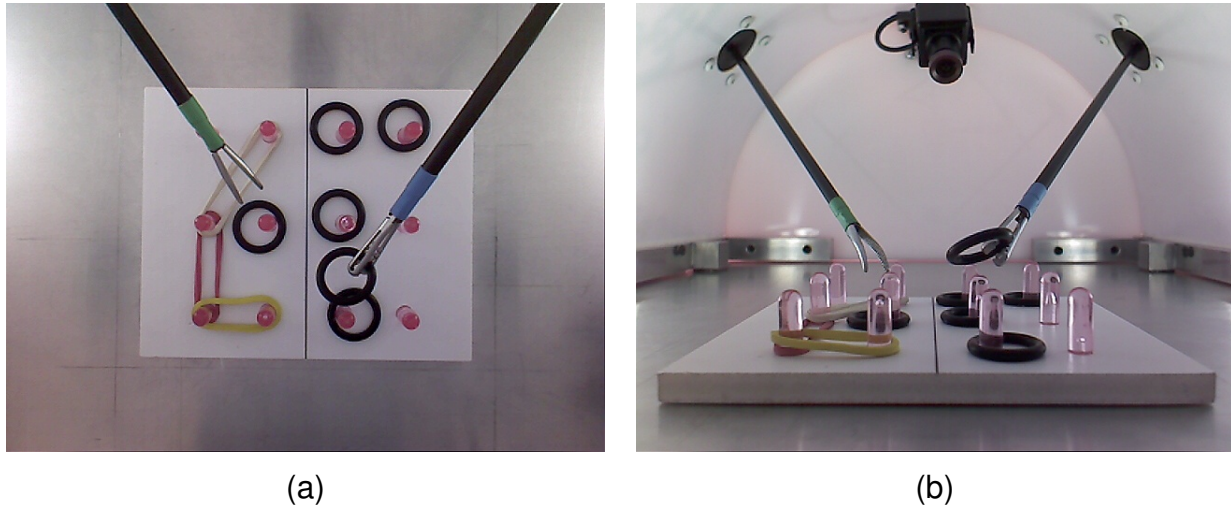
En la Fig. 3.4 se pueden ver la disposición de la punta para cada tipo de instrumento:

La necesidad de llevar a cabo procedimientos con mayor complejidad ha implicado el diseño de nuevos instrumentos que faciliten su uso y permitan un funcionamiento ergonómico [52]. Con la amplia gama de instrumentos existentes, fue necesario proponer un sistema flexible y fácilmente integrable al mismo; por esto, se propuso integrar marcadores de color en el extremo del vástago de la pinza, cada uno de distinto color e identificar su posición a través de técnicas de procesamiento de imágenes.

El marcador consiste en un pedazo de thermofit de dos colores, uno verde y uno azul (Fig. 3.5). Según cada tarea, cada pinza iniciará con un marcador diferente para identificarlo y localizarlo a través del uso de las imágenes, de modo que se obtenga una mejor medición de los parámetros de interés sin afectar el desempeño del cirujano, así como ocurría con los acelerómetros de versiones pasadas.

A manera de lograr la localización de cada instrumento empleamos un conjunto de técnicas de procesamiento de imágenes, los cuales se pueden resumir como:

1. Corrección de la distorsión de la imagen
2. Transformación del espacio de color **RGB** a **HSV**
3. Segmentación en color por umbralización
4. Manejo de la oclusión entre marcadores
5. Detección de la posición **2D** en las imágenes
6. Determinación de la posición **3D**



**Figura 3.5:** Vistas de las pinzas de acuerdo a la cada cámara. Se muestran las capturas desde (a) la cámara superior y (b) la cámara lateral. Entrenador de la tesis de doctorado de F. Pérez.

### 3.3.1. Procesamiento de imágenes

El procesamiento de imágenes se define al procesar los píxeles de una imagen digital por medio de una computadora [22, p. 23].

El procesamiento de imágenes es una respuesta a los tres principales problemas de imágenes [44, p. 1], que son:

- **Digitalización y codificación de la imagen:** facilita la transmisión, impresión y almacenamiento de las tomas.
- **Mejoramiento de la imagen:** mejorar la visualización de la imagen para facilitar la interpretación de su contenido.
- **Segmentación de la imagen:** separación de las zonas homogéneas de acuerdo a algún criterio visual.

El procesamiento de imágenes se realiza a través de la conexión de ambas cámaras a una computadora, de modo que un software de procesamiento de imágenes nos permita adquirir la información a través de la cámara y procesar cada toma para identificar la posición del instrumento.

El software utilizado para realizar el procesamiento de las tomas es **OpenCV**, el cual consiste en un conjunto de librerías de visión por computadora de código abierto. Las librerías se encuentran escritas en lenguaje **C** y **C++** que puede ejecutarse sobre diversos sistemas operativos tales como Linux, Windows y MacOS [7].

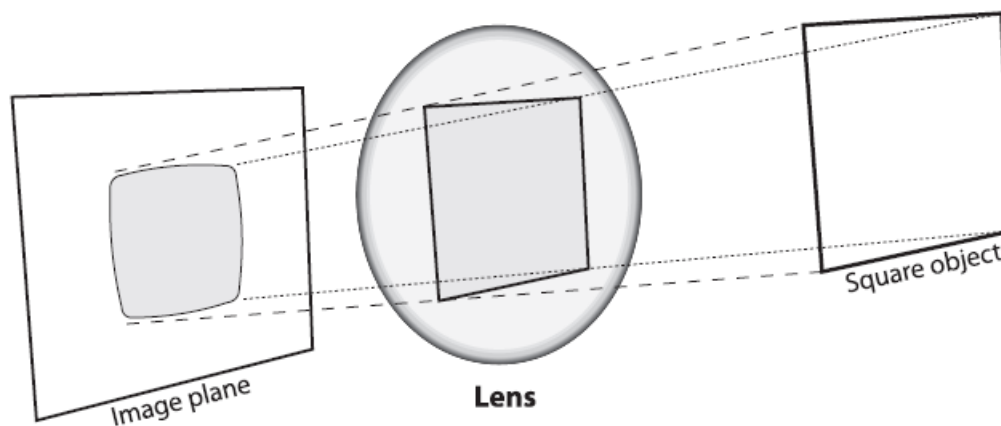


Las librerías de **OpenCV** fue diseñado para permitir la eficiencia de cómputo con un fuerte enfoque en las aplicaciones de tiempo real [7], facilitando más de 500 funciones para el procesamiento. Debido a esto, el software se decidió desarrollar en **C** para dar continuidad al software ya existente. La versión elegida de la librería fue **OpenCV 2.1**, ya que no requiere gran espacio de instalación y contiene las funciones necesarias para poder identificar los marcadores de los instrumentos.

El procesamiento de la imagen involucra la inicialización de las cámaras en modo de captura de video, con el objetivo de que la adquisición de las coordenadas sea en línea y procurando que la estimación se haga en tiempo real.

### 3.3.1.1. Corrección de la distorsión de la imagen

Un problema común de las cámaras es la distorsión radial; un efecto ocurrido en la imagen debido a que la forma del lente causa la refracción de la toma y afecta incidencia planar de la imagen hacia el sensor de la cámara (Fig. 3.6). Este tipo de distorsión se considera 0 en el centro de la toma e incrementa conforme la posición del pixel se encuentra más a la periferia de la toma [7].

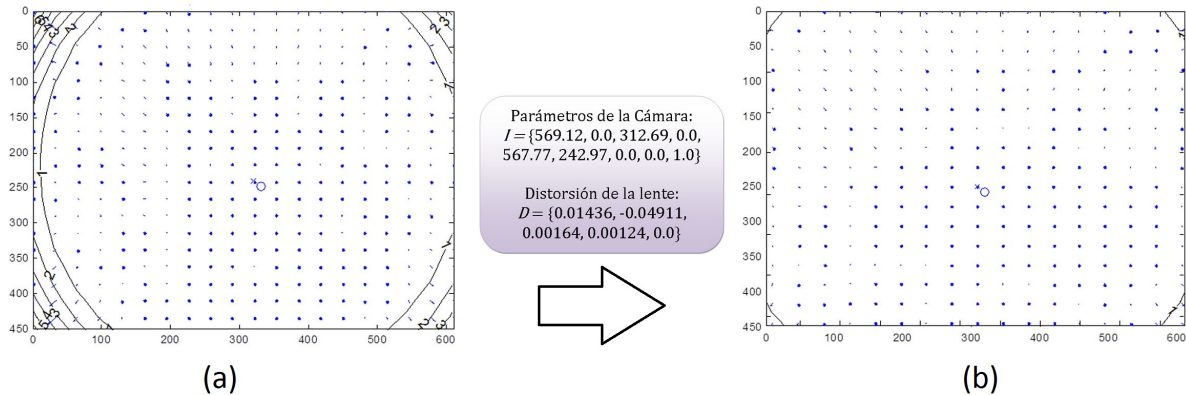


**Figura 3.6:** Ejemplo de la distorsión radial con respecto al centro. Imagen tomada de G. Bradski [7].

A manera de cuantificar y compensar el efecto de la distorsión en nuestro modelo de cámaras, se realizó una calibración de imagen para ambas cámaras utilizando un patrón de tablero de ajedrez. Con esta referencia óptica, se encuentran los parámetros de distorsión y otros parámetros causados por la forma de manufactura de la cámara (extrínsecos), que son necesarios para corregir la perturbación en la imagen.

Aunque la distorsión que presenta la cámara es mínima, es necesaria su corrección para evitar que los defectos de la cámara afecten la linealidad de la imagen y los

próximos procesamientos de la imagen. En la Fig. 3.8 se muestran los parámetros intrínsecos y extrínsecos de la cámara, junto con un ejemplo de la corrección para la distorsión de la imagen.



**Figura 3.7:** Corrección de la imagen: (a) imagen original y (b) imagen corregida de acuerdo a los parámetros mostrados.

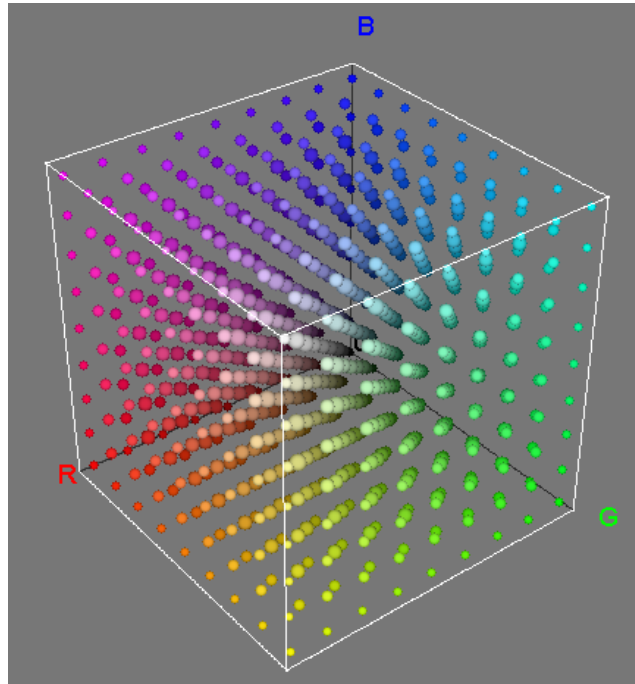
### 3.3.1.2. Transformación del espacio de color RGB a HSV

Una vez realizada la corrección de la distorsión de la cámara requerimos encontrar los marcadores de color dentro del campo visual de la cámara; por lo que es necesario encontrar espacio de color que permita identificar el color de manera inequívoca para evitar los errores de detección.

El espacio de color es una representación geométrica de los componentes de color que conforman la imagen y que cuentan con una representación vectorial debido a su naturaleza multicomponente [27].

La representación de color en medios digitales generalmente utiliza el espacio de color **RGB**, el cual se describe como un espacio cartesiano de 3 dimensiones en donde la posición en el espacio depende de la intensidad en cada uno de sus componentes: **R**, **G** y **B**; sin embargo, esta representación no es perceptualmente uniforme. Es deseable que el espacio de color al cual se desea realizar la transformación cumpla con dicha condición ya que se brinda una mejor representación de como el ser humano percibe el color [55].

El espacio de color a utilizar es **HSV** debido a su naturaleza de perceptualmente uniforme [2]. En el espacio de color, sus componentes son descritos en términos de tono (**H**, del inglés: *hue*), saturación (**S**, del inglés: *saturation*) y valor (**V**, del inglés: *value*).



**Figura 3.8:** Representación geométrica del espacio de color **RGB**. Imagen adquirida por medio de Color Inspector [3].

El componente H se puede calcular de la siguiente forma:

$$H = \begin{cases} \theta : B \leq G \\ 360^\circ - \theta : B > G \end{cases} \quad (3.1)$$

donde:

$$\theta = \arccos \left[ \frac{\frac{1}{2}[(R - G) + R - B]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right] \quad (3.2)$$

El componente **S** se calcula mediante:

$$S = 1 - 3 \frac{\min(R, G, B)}{R + G + B} \quad (3.3)$$

Finalmente, el componente de valor se obtiene de

$$V = \max(r, g, b) \quad (3.4)$$

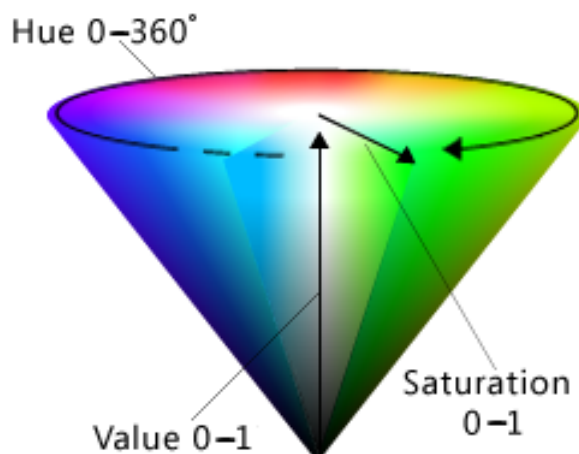
en donde los componentes **r**, **g** y **b** se estiman mediante las fórmulas del espacio de color normalizado **rgb** como:

$$\begin{aligned} r &= \frac{R}{R + G + B} \\ g &= \frac{G}{R + G + B} \\ b &= \frac{B}{R + G + B} \end{aligned} \tag{3.5}$$

De esta manera se obtienen tres componentes de color, donde los valores de cada componente tienen los siguientes rangos de valores:

$$\begin{aligned} H &\in [0^\circ, 360^\circ] \\ S &\in [0, 1] \\ V &\in [0, 1] \end{aligned} \tag{3.6}$$

El espacio de color **HSV** representa geoméricamente un cono como el que se muestra en la Fig. 3.9, en la que se observa el cambio de posición de acuerdo al valor de cada uno de sus componentes.



**Figura 3.9:** Representación geométrica del espacio de color **HSV**. Imagen tomada de **Microsoft** [33].

### 3.3.1.3. Segmentación en color por umbralización

Se mencionó en la sección 3.3 que cada instrumento cuenta con un marcador de color, ya sea verde o azul. Aunque las tonalidades de los marcadores son consistentes ante la vista humana debemos de entender que el color puede percibirse diferente dependiendo de algunos factores como:

- La calidad, marca y modelo de la cámara.
- La iluminación del espacio de trabajo.
- El contraste de colores entre el marcador y el ambiente.
- El ruido visual del entorno.

Para afrontar este problema, se utilizó un rango de valores en el espacio **HSV** para distinguir los marcadores con respecto del entorno de trabajo.

**OpenCV** cuenta con funciones para transformar las imágenes del espacio de color original **RGB** al **HSV**. Sin embargo, el rango de valores es distinto a los mostrados en la ecuación 3.6 debido a que **OpenCV** utiliza una representación de 8 bits por componente de espacio de color(24 bits para **RGB**) y lo mismo sucede con el espacio **HSV**. De esta forma, cada componente sólo puede almacenar valores enteros en un rango de  $[0, 255]$ , por lo que el componente **H** se divide entre 2 para tener valores de  $[0, 179]$ , mientras que los componentes **S** y **V** se multiplican por 255 para ocupar todo el rango de valores para 8 bits.

Los rangos de colores utilizados para cada marcador fueron los siguientes:

Umbral	Opencv			Reales		
	H	S	V	H	S	V
Inferior (verde)	38	45	58	76°	0.1765	0.2275
Superior (verde)	85	255	255	170°	1.0	1.0
Inferior (azul)	95.525	75	70.25	191.05°	0.2941	0.2755
Superior (azul)	120	255	255	240°	1.0	1.0

**Tabla 3.1:** Umbrales correspondientes al rango de valores en **OpenCV** y en el rango de valores real para el espacio de color **HSV**.

Una vez realizada la umbralización se obtienen dos imágenes binarias, una para el marcador azul y otra para el marcador verde. En ellas se puede observar ruido visual que debe ser removido a través de algún tipo de operaciones para imágenes binarias.

El método empleado es un post-procesamiento morfológico para disminuir el ruido remanente de sal y pimienta, el cual es referido como puntos blancos en regiones negras (sal) y puntos negros en regiones blancas (pimienta) [38, p. 90].

Para este proyecto, los puntos son eliminados utilizando dos erosiones y cuatro dilataciones sobre cada imagen binaria, todas las sobre la misma imagen y usando un elemento estructurante cuadrado de  $3 \times 3$  centrado sobre cada pixel de interés [15].

#### **3.3.1.4. Manejo de la oclusión entre marcadores**

En la sección 3.3.1.3 se nombraron los pasos para obtener una imagen binaria para cada color, en donde sólo debería existir un cuerpo binario para cada marcador de modo que, a través de la localización del centro de cada marcador en la imagen binaria, se podría construir un sistema de transformación geométrica para convertir las coordenadas de los dos puntos en **2D** (coordenadas del centro del marcador para cada imagen) a una coordenada espacial en **3D**.

El problema que presenta este tipo de técnica es que, en ocasiones, las pinzas pueden llegar a obstruirse visualmente entre ellas durante ciertas tareas; por ejemplo, las tareas de corte y sutura generalmente requieren el intercambio de posición para sujetar la aguja quirúrgica o manipular el tejido a diseccionar.

Existen tres tipos de oclusiones potenciales en el espacio de trabajo:

1. *Oclusión total*: la oclusión total implica que los objetos de la tarea o los mismos instrumentos bloqueen totalmente la visibilidad de los marcadores de la otra pinza. En este tipo de oclusión no se puede registrar el desplazamiento.
2. *Oclusión parcial*: la oclusión parcial implica que alguna parte exterior del marcador no sea visible para una o más cámaras, lo que implica la modificación del tamaño del marcador y, por lo tanto, de la posición del centro del mismo.
3. *Oclusión y división de la región*: es un tipo de oclusión parcial en donde el objeto causante de la obstrucción visual cruza por en medio de la región a detectar, lo que genera dos o más regiones para un mismo marcador.

El software ofrece una solución para la mayoría de las oclusiones previamente expuestas; principalmente para resolver las oclusiones con divisiones múltiples.

**Algoritmo para el manejo de regiones** El procesamiento propuesto para evitar los casos de *oclusión y división de región* utiliza un conteo y fusión para todas las regiones contenidas en cada imagen binaria.

La metodología ofrece soluciones cuando en la imagen:

1. Existe una sola región
2. Existen dos o más regiones
3. No existe ninguna región detectada

**Detección de una región** Una vez que la umbralización se ha realizado y sólo se encuentra una región, se asume que esa región pertenece al marcador correspondiente. Lo ideal es hallar el valor de la coordenada de la imagen que corresponde al centroide de la región del marcador.

El método para calcular la posición de la coordenada en una imagen de tamaño  $M \times N$  es a través de los *momentos 2D de orden  $(p+q)$*  [22, p. 862] y que se define como:

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x, y) \quad (3.7)$$

en donde  $p = 0, 1, 2, \dots$  y  $q = 0, 1, 2, \dots$  son enteros, en donde las coordenadas del centroide serían calculadas como:

$$\bar{X} = \frac{m_{10}}{m_{00}} \quad \text{y} \quad \bar{Y} = \frac{m_{01}}{m_{00}} \quad (3.8)$$

Al utilizar una imagen binaria, la expresión  $f(x, y)$  se reduce a ceros y unos. De esta forma, la expresión  $m_{00}$  se puede expresar como el total de pixeles que componen la región de interés.

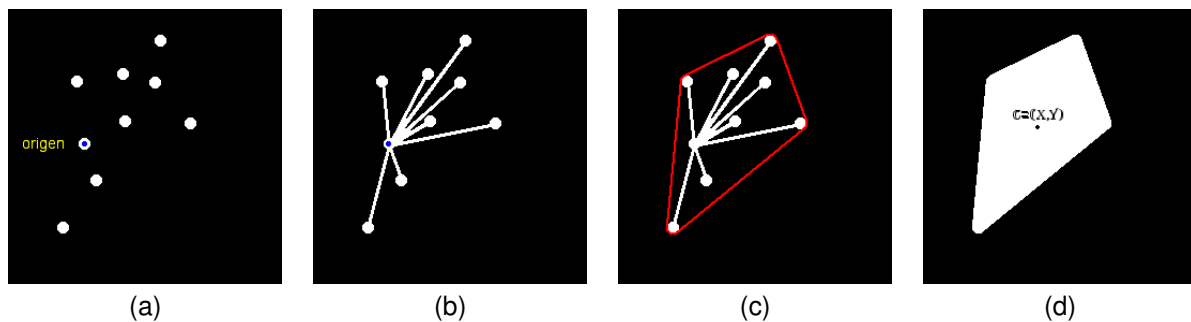
**Detección de dos o más regiones** Para el caso en el que dos o más regiones existan en la imagen binaria se realiza el mismo procedimiento hecho para la detección de una región, solo que se repite para todas las regiones existentes de manera que se obtenga un vector de coordenadas del mismo tamaño que el número de regiones. Una vez obtenido el vector de centroides, es necesario fusionar las regiones desconexas para que, al procesar nuevamente la superficie, se obtenga un sólo centroide.

Para la fusión es requerido enumerar las regiones y empezar a estimar cada par de coordenadas de sus centroides, en donde la primer región será guardada

como un punto de origen y este punto se utilice para combinar el resto de las regiones (Fig. 3.10.(a)).

Las regiones subsecuentes serán fusionadas mediante el trazo de líneas blancas desde el origen hacia el centroide del resto de las regiones. De esta forma, se obtiene una topología tipo estrella luego de conectar todas las regiones subsecuentes (Fig. 3.10.(b)). Un problema notable es que el marcador tiene forma rectangular y convexa por lo que, aunque la fusión de las regiones nos proporciona solo una región, sigue existiendo una distorsión de la forma y, por lo tanto, de la posición del centroide de la nueva región.

El método usado para recuperar la forma original del marcador es a través de la operación *convex hull*. Un *convex hull* se define como una operación que permite convertir cualquier región en convexa; en donde se define región convexa como aquella en la que trazada una recta entre dos puntos que pertenecen a una región, todos los puntos de la recta pertenecen a la región [22, p. 669]. El procedimiento es análogo a colocar una liga alrededor de todas las regiones de la imagen binaria (Fig. 3.10.(c)).



**Figura 3.10:** Etapas del algoritmos de fusión de regiones. En la imagen se muestran (a) las regiones disconexas, (b) la topología de estrella después de unir cada región con el origen, (c) el contorno correspondiente al *convex hull* y (d) el centroide final de la región corregida.

Una vez obtenida la región convexa correspondiente a todas las regiones, utilizamos el mismo procedimiento para identificar el centroide en una región (sección 3.3.1.4); al final sólo tendremos la coordenada final de la región corregida (Fig. 3.10.(d)).

**Detección de ausencia de regiones** Para el caso en el que no se encuentre ninguna región en la imagen, ya sea por alguna oclusión total o porque la pinza o instrumento se encuentre fuera del campo de vista de la cámara, se propone conservar el valor de las últimas coordenadas de detección que haya sido percibida por las cámaras. De este modo, se evita una falsa suposición de cambio y se asume que



el instrumento se queda estático en la misma posición.

### **3.3.2. Determinación de la posición 3D**

La obtención de los centroides proveen dos pares de coordenadas en la imagen para cada marcador: las coordenadas  $(x, y)$  en la imagen captada por la cámara superior y las coordenadas  $(x, y)$  en la imagen captada por la cámara lateral. Estos puntos, aunque brindan información espacial dentro de una proyección bidimensional (imagen) no nos proveen información de la posición dentro del plano de trabajo **3D**.

Mediante el entendimiento del modelo matemático a través del cual las cámaras capturan las imágenes y la disposición de las cámaras, es posible construir un sistema de transformación para convertir ambos pares de puntos **2D** en un arreglo de coordenadas **3D**.

#### **3.3.2.1. Modelo geométrico de la escena**

Para entender los parámetros necesarios para convertir las coordenadas obtenidas a través de las imágenes en coordenadas tridimensionales, es necesario comprender cuales son los factores que afectan tanto la escena como el equipo de captura.

En este apartado se mencionan algunas de las condiciones geométricas más importantes a considerar para construir un sistema geométrico de transformación.

**Transformaciones afines de los objetos en la escena** En matemáticas, cualquier punto en el espacio puede ser representado en su forma matricial a través de un vector de coordenadas y, como vector, tiene todas las propiedades dadas en un espacio vectorial tales como: *producto escalar*, *producto punto* y *suma y resta de vectores*. El problema que presenta el espacio vectorial es que para representar un vector se necesita un origen, es decir, un punto no es visto como un punto sino como un vector que va desde el origen hacia dicho punto.

A manera de resolver la necesidad de un punto de origen en el espacio vectorial, se necesita emplear un espacio y las operaciones necesarias que faciliten la representación de puntos para realizar transformaciones en dicho espacio; a este tipo de espacios se llaman *espacios afines* [4, p. 32].

Un espacio afín es un espacio lineal que, como su nombre lo indica, mantiene la afinidad entre los puntos, rectas y planos que existen en la escena, por lo que para conservar la linealidad tanto en la captura como en la escena se deben disminuir cualquiera de los factores, tanto internos como externos, que ocasionen la distorsión

de los datos. Por esta misma razón es que fue necesaria una compensación inicial a través de la estimación de los parámetros de distorsión de la cámara (sec. 3.3.1.1) y así conservar las propiedades del espacio.

En un espacio afín se pueden realizar las siguientes transformaciones:

- **Traslación:** implica el desplazamiento de cada uno de los puntos con respecto a los valores de un nuevo punto.
- **Rotación:** permite realizar una transformación de giro con respecto a alguno de los ejes del plano.
- **Escalamiento:** logra un escalamiento de los puntos, de modo que la distancia entre ellos sea mayor o menor. Este escalamiento puede ser para uno o varios componentes.
- **Shearing:** es una transformación que involucra el sesgo o inclinación de uno de los extremos hacia alguno de los ejes coordenados.

Cada punto utiliza una representación en forma de renglón aumentado, lo que significa que utiliza los tres componentes dimensionales de las coordenadas (en caso de **3D**:  $(x, y, z)$ ) pero se le adiciona un componente extra con el valor de uno. De esta forma, las estructuras aumentadas son  $n + 1$  dimensionales, donde  $n$  es el número de dimensiones espaciales a representar. Esta característica de los espacios afines proporciona una independencia del punto al origen.

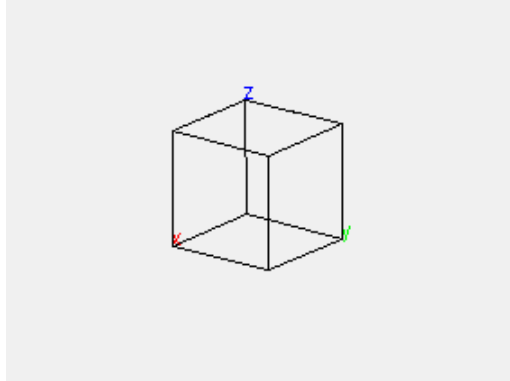
De la misma manera, existe una matriz de  $n + 1$  dimensional que permite realizar las transformaciones a través de un producto matricial, el cual se representa como:

$$\overline{\mathbf{X}'} = T \overline{\mathbf{X}} \tag{3.9}$$

En forma matricial, los valores de la matriz  $T$  que permiten que  $\overline{\mathbf{X}'} = \overline{\mathbf{X}}$ ; es decir, que no realicen ninguna transformación se expresan en la ecuación 3.10 como:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{3.10}$$

A través de la modificación de los valores originales de la matriz  $T$  podemos realizar las transformaciones previamente mencionadas en este apartado. El resultado deja los puntos en la misma posición con respecto a su origen (Fig. 3.11).



**Figura 3.11:** Imagen de un cubo posicionado en un punto de origen, resultado de aplicar la transformación mostrada en la ecuación 3.10. Imagen tomada de F. Hanisch [23].

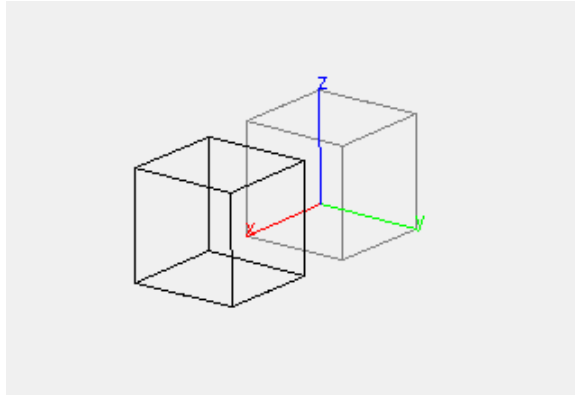
**Traslación** La traslación implica el desplazamiento de cada uno de los puntos con respecto a los valores de un nuevo punto.

El nuevo punto tiene coordenadas  $P = (x_n, y_n, z_n)$  por lo que, si queremos modificar la matriz  $T$  para incluir el desplazamiento en los puntos originales utilizamos la siguiente expresión

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & x_n \\ 0 & 1 & 0 & y_n \\ 0 & 0 & 1 & z_n \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (3.11)$$

La traslación resulta a cada componente de cada punto el valor de  $P$ , resultando en un movimiento con respecto al origen (Fig. 3.12).

**Rotación** Esta operación realiza un giro de los puntos con respecto a alguno de los ejes coordenados del plano; de modo que para cada eje coordenado existe una matriz de transformación que llamaremos  $R_x$ ,  $R_y$  y  $R_z$  que utiliza un ángulo  $\theta$  para

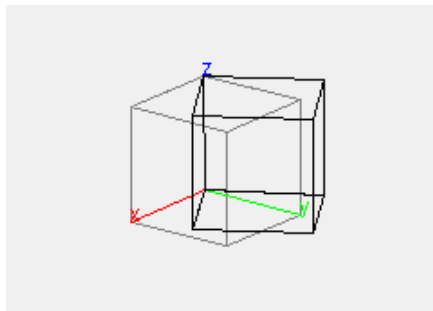


**Figura 3.12:** Imagen de un cubo trasladado a lo largo del eje  $X$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23].

realizar la rotación mediante las siguientes matrices:

$$R_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad R_y = \begin{pmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad R_z = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.12)$$

La rotación se efectúa definiendo el valor del ángulo  $\theta$  y el eje de rotación a través de la elección de la matriz apropiada, utilizando dicha matriz en lugar de la matriz  $T$  (Fig. 3.13).



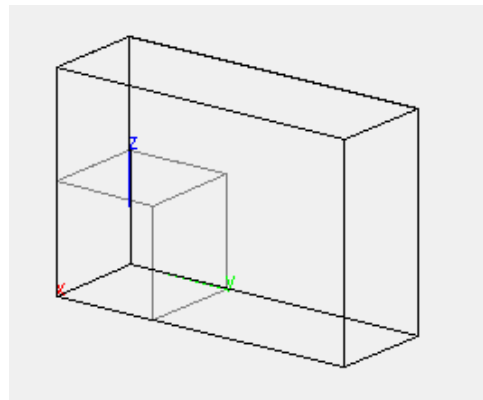
**Figura 3.13:** Imagen de un cubo rotado  $32^\circ$  sobre el eje  $Z$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23].

**Escalamiento** El escalamiento logra un aumento o disminución de los puntos y la distancia entre ellos sea mayor o menor. El escalamiento puede estar dado para uno o varios componentes.

El escalamiento puede darse independientemente hacia cada eje a través de tres factores de escala ( $s_x, s_y, s_z$ ) que intervienen en la matriz de transformación como:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (3.13)$$

Cada factor de escala puede ser diferente, lo que manifestaría una prolongación de la distancia distinta entre los puntos de acuerdo al componente en el que se encuentren (Fig. 3.14).



**Figura 3.14:** Imagen de un cubo escalado con  $s_x = 1$ ,  $s_y = 3$  y  $s_z = 2$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23].

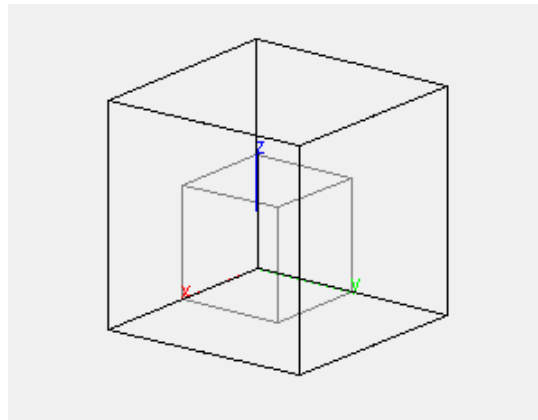
Si a la matriz se le desea aplicar los mismos coeficientes de escalamiento para todos los componentes entonces se habla de un *escalamiento uniforme* y se puede modificar un factor común de escalamiento, el cual está dado por el último elemento de la matriz. Cuando el valor de este componente es igual a 1 no se genera ningún escalamiento; si su valor aumenta se produce un aumento en la escala, mientras que si su valor disminuye también la escala lo hará.

La matriz correspondiente se puede manejar con un factor único  $s$  de escalamiento

a través de la siguiente representación matricial:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{s} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (3.14)$$

De esta manera controlamos el escalamiento de todos los componentes mediante un sólo factor (Fig. 3.15).



**Figura 3.15:** Imagen de un escalamiento uniforme de un cubo con  $s = 2$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23].

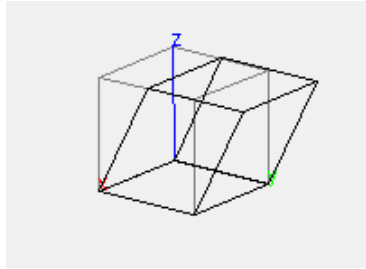
**Shearing** Es una transformación que involucra el sesgo o inclinación de uno de los extremos hacia alguno de los ejes coordenados; esto es producido por la modificación de cualquiera de los valores que no se encuentran en la diagonal y que tampoco se localizan en el último renglón o fila de la matriz.

La matriz de *shear* puede entonces modificarse en cualquiera de los dos coeficientes existentes para cada componente de la siguiente manera:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & \text{ShX}_1 & \text{ShX}_2 & 0 \\ \text{ShY}_1 & 1 & \text{ShY}_2 & 0 \\ \text{ShZ}_1 & \text{ShZ}_2 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (3.15)$$

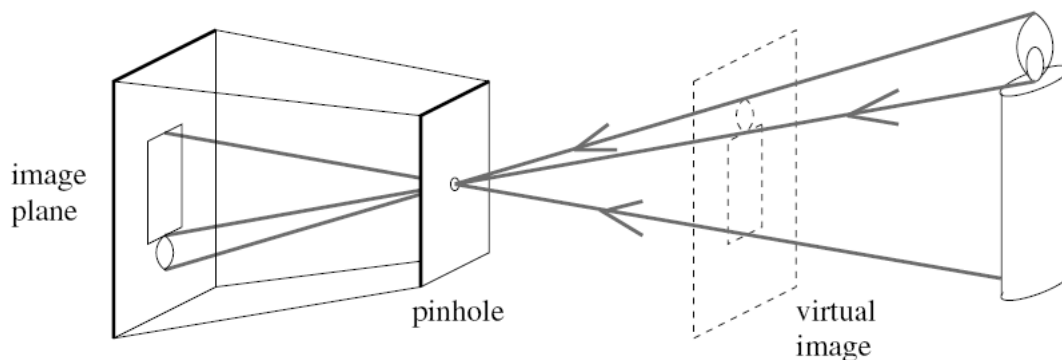
El shearing puede aplicarse para más de un componente, incrementando el tamaño

de acuerdo a los planos y los valores elegidos (Fig. 3.16).



**Figura 3.16:** Imagen de un shearing hacia el eje  $Y$  con  $ShY_2 = 0.5$ : se muestra en color negro el resultado de la transformación y en color gris las coordenadas originales. Imagen tomada de F. Hanisch [23].

**Modelo de cámara pinhole** Las cámaras son dispositivos de captura que también siguen un modelo matemático llamado *modelo de cámara estenopeica* o *modelo de cámara pinhole*. En este modelo, se tiene un contenedor con forma de caja y con un pequeño agujero en el centro de una de las paredes de la caja: el agujero de la caja permite el paso de la luz reflejada por los objetos frente a ella y se proyectará dentro de la pared opuesta a la tapa perforada de manera invertida.



**Figura 3.17:** Representación del modelo de cámara pinhole. Imagen tomada de D.A. Forsyth [19].

En una cámara, el agujero por donde se recibe la luz se llama obturador y se abre rápidamente para capturar los haces de luz de la escena que se encuentra frente a ella; además de integrar un lente antes de la llegada de la luz al obturador para ingresar a la cámara. Pese a los factores que incluye el modelo de cámara, este sigue un modelo de proyección llamado *perspectiva de pinhole* o *perspectiva central* [19, p. 4].

Sean  $(X, Y, Z)$  un conjunto de coordenadas de un punto en la escena **3D**, las coordenadas de proyección en la imagen como  $(x, y, z)$  y  $z = d$  (la distancia del

obturador al plano de proyección) se tiene que la proyección está dada por:

$$\begin{aligned}x &= \lambda X \\y &= \lambda Y \\d &= \lambda Z\end{aligned}\tag{3.16}$$

que se puede reducir como:

$$\begin{aligned}x &= d \frac{X}{Z} \\y &= d \frac{Y}{Z}\end{aligned}\tag{3.17}$$

La ecuación 3.17 denota la ausencia de profundidad debido a su naturaleza bidimensional de la imagen, lo que demuestra la importancia de la cámara superior; ya que, sin ella, no tendríamos alguna referencia de la posición de profundidad con respecto al escenario.

### 3.3.2.2. Modelo paramétrico para la transformación de 2D a 3D

A través de la matriz de transformación se puede describir cualquier tipo de transformación afín sufrida por los objetos en la escena como traslación, rotación, escalamiento y shearing. Varias transformaciones pueden combinarse en una sola matriz a través del producto de las matrices correspondientes a cada transformación que se desea hacer. El vector resultante tiene un tamaño de  $n + 1$ , donde  $n = 3$  puesto que  $n$  le corresponde el valor del número de dimensiones del espacio **3D**. En los modelos anteriores, el resultado de la transformación geométrica tiene el mismo tamaño que el vector aumentado de coordenadas  $(x, y, z, 1)^T$ .

Debido a que la matriz de transformación es cuadrada, se puede calcular la inversa de la matriz a manera de hallar los valores originales de las coordenadas previos a la transformación. Sin embargo, se sabe que las imágenes son proyecciones de una escena real sobre un plano y que implica la pérdida del componente espacial sobre el cual no se realiza la proyección: una proyección sobre el plano  $XY$  hace que se pierda información del componente  $Z$ . Para resolver la pérdida de información debido a la proyección, debemos analizar las ecuaciones de perspectiva que modelan la adquisición de la imagen.

Sea  $\bar{c}_n$  un vector aumentado con las coordenadas en la imagen de la cámara  $n$ ,



entonces

$$\bar{\mathbf{c}}_n = d \begin{pmatrix} x_n \\ y_n \\ 1 \end{pmatrix} \quad (3.18)$$

en donde  $d$  es el factor de profundidad en la proyección para la captura de la imagen.

De la misma forma, podemos proponer un vector aumentado  $\bar{\mathbf{x}}$  como el conjunto de coordenadas del mundo real como

$$\bar{\mathbf{x}} = \begin{pmatrix} x_R \\ y_R \\ z_R \\ 1 \end{pmatrix} \quad (3.19)$$

lo que significa que debe de existir una matriz de parámetros  $P$  de  $(3 \times 4)$  que incluye todas las transformaciones afines y de proyección que ocurren en las coordenadas **3D** y las convierta a un sistema de coordenadas **2D**.

La matriz  $P$  se expresa de la siguiente manera:

$$\bar{\mathbf{c}}_n = P \bar{\mathbf{x}} \quad (3.20)$$

que en su forma matricial

$$\begin{pmatrix} d x_n \\ d y_n \\ d \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \begin{pmatrix} x_R \\ y_R \\ z_R \\ 1 \end{pmatrix} \quad (3.21)$$

Al separar la matriz  $P$  en renglones, denotados como  $\bar{\mathbf{P}}_r$  y siendo  $r$  es el número

de renglón, podemos separar cada componente

$$\begin{aligned} d x_n &= \overline{P_1} \bar{x} \\ d y_n &= \overline{P_2} \bar{x} \\ d &= \overline{P_3} \bar{x} \end{aligned} \tag{3.22}$$

que al sustituir  $d$  por la parte derecha de la ecuación resulta en un sistema de dos ecuaciones

$$\begin{aligned} x_n \overline{P_3} \bar{x} &= \overline{P_1} \bar{x} \\ y_n \overline{P_3} \bar{x} &= \overline{P_2} \bar{x} \end{aligned} \tag{3.23}$$

### 3.3.2.3. Estimación de los parámetros de transformación

Las fórmulas utilizadas en la ecuación 3.26 nos permiten establecer una relación entre las coordenadas del mundo real  $\bar{x}$  y las coordenadas bidimensionales en la imagen  $\overline{c_n}$ . Al obtener un conjunto de valores de la coordenada real y sus respectivas mediciones en el plano de la imagen, podríamos hallar los valores de cada uno de los elementos de la matriz  $P$ .

La matriz de proyección requiere la estimación de los coeficientes de  $P$  a excepción de  $p_{34}$ , ya que corresponde al factor de escalamiento uniforme de la matriz. Dado que en nuestro problema no requerimos un escalamiento uniforme, se establece el factor de escala como  $p_{34} = 1$  para disminuir el número de coeficientes a 11.

Mediante 6 puntos distintos con sus coordenadas en el mundo real y sus correspondientes coordenadas de imagen podemos usar las dos ecuaciones 3.26 para armar un sistema de 11 ecuaciones para resolver todos los valores faltantes de la matriz  $P$ ; sin embargo, requerimos estimar una matriz  $P$  para cada una de las cámaras ya que cada una tiene sus propios parámetros de proyección. Aunque este proceso duplica el número de parámetros a encontrar y se incrementa a 22, se pueden utilizar los mismos puntos usados para estimar la primera matriz y sólo cambiar por los valores de las coordenadas correspondientes en la imagen de la segunda cámara.

### 3.3.2.4. Calibración del sistema

Para calibrar el sistema en el proyecto se tuvieron que proponer 6 puntos de coordenadas reales junto con sus coordenadas correspondientes para cada imagen

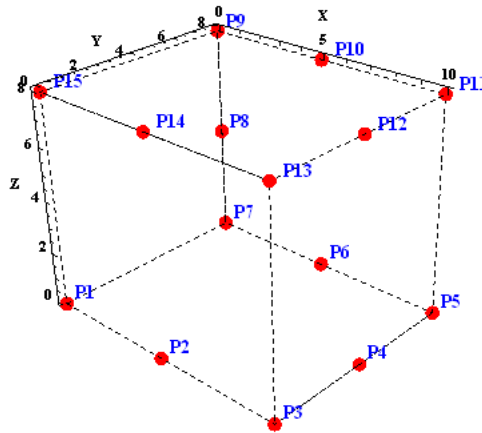
de la cámara. Para esto, se procuró utilizar los extremos de todo el espacio visual de ambas cámaras y medir su posición real para obtener los puntos de calibración.

Los puntos sugeridos se muestran en la tabla 3.2.

No. de punto	Posición real			Cámara superior		Cámara lateral	
	$x_R$	$y_R$	$z_R$	$x_1$	$y_1$	$x_2$	$y_2$
1	0	0	0	39.04344417	17.28372686	142.4419323	107.8126532
2	5	0	0	314.5985962	18.27905792	303.5886591	109.1138705
3	10	0	0	601.4410372	16.52387279	465.8986876	109.2613732
4	10	4	0	600.799064	237.7844026	509.6033629	73.08330565
5	10	8	0	600.4410372	463.385662	586.5552029	16.88395617
6	5	8	0	312.7330942	462.7408408	295.2195706	16.69130633
7	0	8	0	39.30208037	463.3138899	16.5245129	16.87242153
8	0	8	4	122.9240729	389.2384427	15.38675285	239.4462442
9	0	8	8	170.8837392	350.0142653	16.02994372	463.5964445
10	5	8	8	325.4902838	351.4596751	291.0831218	463.0059151
11	10	8	8	484.463924	350.3573507	582.2005942	464.8668852
12	10	4	8	485.986409	226.7163752	510.2982989	408.681097
13	10	0	8	486.6859821	98.77456851	466.1411254	369.2916083
14	5	0	8	327.9531434	98.66758661	305.5925177	368.1522535
15	0	0	8	171.5010377	99.29228233	145.0338672	367.3358638

**Tabla 3.2:** Coordenadas del mundo real con sus correspondientes coordenadas de proyección sobre la imagen.

Los puntos fueron sugeridos a razón que se ocupara puntos que otorgaran información de profundidad y apertura sobre todos los ejes coordenados del espacio visual.



**Figura 3.18:** Disposición de los puntos en el espacio de coordenadas real del sistema.

Los puntos utilizados para la calibración del proyecto fueron el 1, 4, 6, 8, 11 y 14.

### 3.3.3. Estimación de las coordenadas 3D

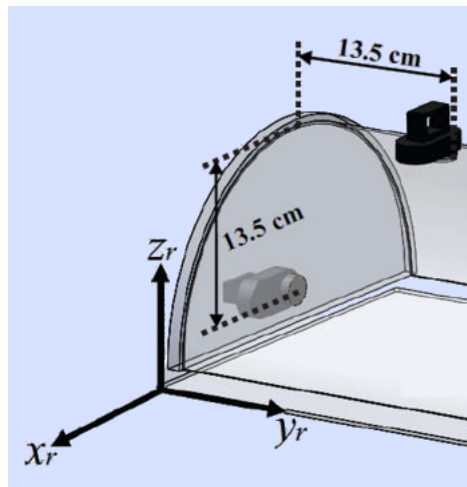
Una vez estimados los coeficientes de las matrices  $P$  y  $P'$ , donde  $P$  corresponde a la cámara superior y  $P'$  a la cámara lateral (Fig. 3.2), se puede utilizar la ecuación 3.26 con respecto a cada una de las cámaras para encontrar el vector de coordenadas originales  $\bar{x}$ .

Reescribiendo la ecuación 3.26 de acuerdo a cada cámara tenemos que:

$$\begin{aligned} x_1 \overline{P_3} \bar{x} &= \overline{P_1} \bar{x} \\ y_1 \overline{P_3} \bar{x} &= \overline{P_2} \bar{x} \\ x_2 \overline{P'_3} \bar{x} &= \overline{P'_1} \bar{x} \\ y_2 \overline{P'_3} \bar{x} &= \overline{P'_2} \bar{x} \end{aligned} \tag{3.24}$$

lo cual conforma un sistema de 4 ecuaciones. Sin embargo, sólo requerimos tres ecuaciones para encontrar los valores de las coordenadas  $(x_R, y_R, z_R)$ ; de manera que podemos descartar alguna de las ecuaciones para evitar que el sistema sea sobredeterminado.

La cámara superior recibe una proyección del plano  $XY$  mientras que la cámara lateral solo percibe el plano  $XZ$ , por lo que podemos utilizar ambas ecuaciones de la primera cámara y sólo utilizar una ecuación de la segunda cámara de modo que excluyamos el componente redundante de la imagen.



**Figura 3.19:** Representación de los planos visuales de cada cámara con respecto al sistema coordenado del entrenador. Imagen tomada de F. Perez [43].

Si se observa la Fig. 3.19, se puede apreciar que el componente  $x_1$  obtenido de la imagen de la cámara superior concuerda con el componente  $x_2$  obtenido de la imagen

de la cámara lateral, por lo que podríamos excluir la tercera ecuación para así formar el sistema de solución como:

$$\begin{aligned} x_1 \overline{P_3} \bar{x} &= \overline{P_1} \bar{x} \\ y_1 \overline{P_3} \bar{x} &= \overline{P_2} \bar{x} \\ y_2 \overline{P'_3} \bar{x} &= \overline{P'_2} \bar{x} \end{aligned} \tag{3.25}$$

de donde se puede obtener una ecuación para cada componente  $(x_R, y_R, z_R)$  y estimar así la posición real de la pinza.

### **3.3.4. Manejo de la aparición/desaparición de marcadores**

El algoritmo original considera que un marcador de color siempre permanece en la misma mano, por lo que si un cirujano realiza tareas que involucran el intercambio de pinzas podría ocasionar un error en las métricas ya que el software no estaba preparado para el cambio de la posición de referencia. En este sistema se propone un algoritmo para evitar errores en el registro de los datos.

El medio para detectar si el instrumento es substraído es a través de la cámara lateral, ya que esta tiene visibilidad de los puntos de inserción de los instrumentos que se encuentran fuera del rango de visión de la cámara superior. Mediante la determinación de las condiciones adecuadas, se puede establecer si los instrumentos cambian su punto de inserción.

El algoritmo propone evaluar estados para cada captura de la cámara, por lo que se utilizan dos variables booleanas llamadas `no_instrumento` que indica si no se detectó un instrumento en la toma pasada y `vizq_ader` que indica que la pinza con el verde se encuentra sujeta por la mano izquierda, mientras que la otra se encuentra a la derecha. El algoritmo de decisión se puede ver en la Fig. 3.20.

Lo primero que verifica el algoritmo es que la cámara sea la cámara lateral, ya que es la única que puede decidir si los instrumentos se encuentran en el espacio de trabajo o no. En caso de que no detecte ningún objeto entonces colocará la bandera `no_instrumento` como verdadero.

Si un instrumento aparece entonces entrará a evaluar si es que el instrumento que existe en el campo de vista es verde o azul y dependiendo de su posición asignará el valor correspondiente a la bandera `vizq_ader`. Esto es intuitivamente lógico, ya que los instrumentos no pueden insertarse ambos en el mismo hueco, por lo que es necesario que primero desaparezcan los marcadores para que puedan cambiar de posición.

```

SI (Cam==CAM_LAT)
  SI((No mVerde) Y (No mAzul))
    no_instrumento = true;
  SINO
    SI(no_instrumento = true)
      SI(Hay Verde)
        SI(x_mVerde>320)
          Vizq_ader = true;
        SINO
          Vizq_ader = false;
        FINSI
      SINO
        SI(x_mAzul<320)
          Vizq_ader = true;
        SINO
          Vizq_ader = false;
        FINSI
      FINSI
    no_instrumento = false;
  FINSI
FINSI
FINSI

```

**Figura 3.20:** Algoritmo para la aparición/desaparición de marcadores.

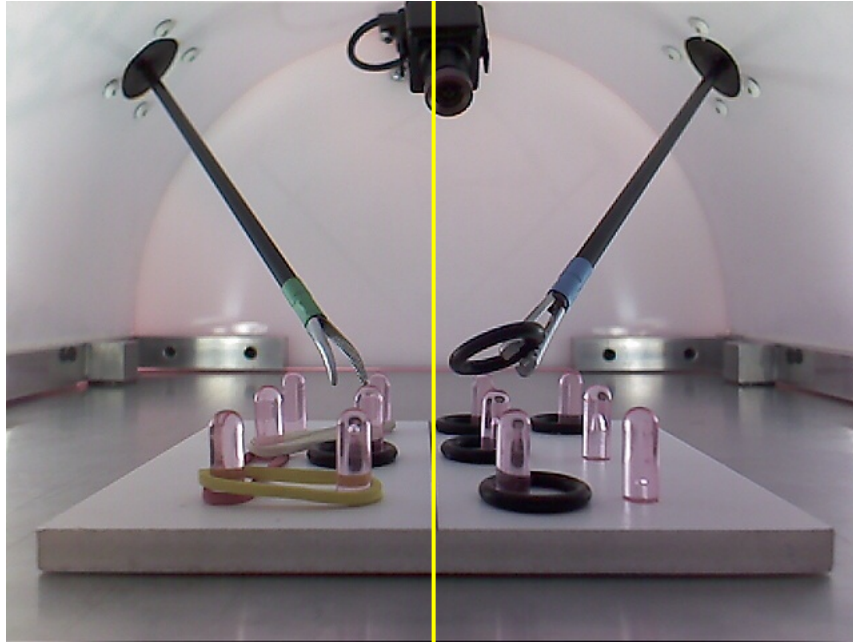
En los valores de decisión utilizamos 320 porque dicho valor corresponde a la mitad de la imagen, por lo que si aparece en alguna de los dos lados de la imagen de la cámara lateral entonces automáticamente se estará determinando la posición para dicho marcador (Fig. 3.21).

### 3.3.5. Registro en archivo

Las coordenadas de cada pinza son guardadas en un archivo de texto plano en donde se separa cada componente mediante comas y tiene la siguiente disposición

$$x_{R.izq}, y_{R.izq}, z_{R.izq}, x_{R.der}, y_{R.der}, z_{R.der}, \theta_{XY_{izq}}, \theta_{XZ_{izq}}, \theta_{XY_{der}}, \theta_{XZ_{der}}, t_{muestra} \quad (3.26)$$

en donde  $R.izq$  corresponde al componente de la posición real de la mano izquierda,  $R.der$  corresponde al componente de la posición real de la mano derecha,  $\theta$  muestra el ángulo con respecto a cada plano de proyección para ambas manos y  $t_{muestra}$  indica el tiempo en el que se adquirieron las coordenadas luego de iniciar el registro de posiciones.



**Figura 3.21:** Vista de la cámara lateral y su división a la mitad de la imagen. Se puede observar que los puntos de inserción son altamente visibles al insertar un instrumento. Entrenador de la tesis de doctorado de F. Pérez.

Al finalizar el registro de la imagen correspondiente, se procede a procesar las nuevas tomas de cada cámara y adquirir mediante los mismos pasos las coordenadas de ubicación de cada instrumento junto con el tiempo en el que fue adquirido para incluirlo en el archivo de registros.

### **3.4. Resumen**

En este apartado se describió la infraestructura y arquitectura del entrenador para la colocación de las cámaras en posición ortogonal. De la misma manera, se desarrollo todo el modelo matemático hasta llegar al modelo de cámara pinhole, de donde se partió para obtener un sistema de ecuaciones que permitieran obtener la coordenadas **3D**.

La descripción del procesamiento de imágenes se describe paso a paso y se menciona los métodos de solución para resolver el problema del intercambio de posición de los instrumentos para finalmente escribir el archivo con los datos correspondientes.





# Capítulo 4

## Sistema de registro

En el capítulo 3 fueron descritos los detalles físicos y algorítmicos del entrenador de caja, pero no se detallaron las herramientas de desarrollo que se utilizarán para la implementación del software.

En el presente capítulo se describen las restricciones que las versiones anteriores del software tenían (sec. 4.1) y las soluciones que el nuevo sistema propone. Posteriormente, se describen las herramientas de software a utilizar para el desarrollo de la aplicación (sec. 4.2). De igual forma, se plantea el uso de parámetros de movimiento y la descripción y selección de aquellos que han sido más ampliamente aplicados dentro del área de análisis de movimiento (sec. 4.3).

### 4.1. Antecedentes del software

La última versión del sistema de registro era capaz de realizar las tareas que fueron descritas en el capítulo 3; sin embargo, el programa presentaba algunos inconvenientes como:

- El software estaba desarrollado sobre *Eclipse IDE* bajo un entorno de consola, de modo que cada registro requería una nueva ejecución del programa, por lo que no se podía mantener un uso ininterrumpido de la aplicación.
- La identificación de la cámara superior y lateral se realizaba de acuerdo al orden de conexión de los puertos USB, de manera que si en algún momento las cámaras se conectaban en desorden, pudiera entonces ocurrir un cambio en el sistema coordinado del equipo.

- No se tiene un registro de historial por usuarios para cada practicante, por lo que evaluar a cada cirujano requiere coleccionar los archivos de texto con las coordenadas en una misma carpeta.
- El software no tiene manera de hacer un post-procesamiento a los registros luego de la ejecución de la tarea para obtener métricas que cuantitativamente fueran más descriptivas: sólo refleja el tiempo y la posición en un archivo y es necesario su procesamiento con otro software como *MATLAB* o *Excel*.

El software propuesto para este proyecto no sólo procura resolver las insuficiencias de la última versión del software y generar una aplicación usable, sino el proporcionar un framework que sugiera prácticas, mediciones y despliegues de información que permitan un entrenamiento supervisado eficiente y controlado por el usuario, por lo que es necesario la infraestructura tecnológica y el software necesario que disminuya el tiempo de desarrollo.

## 4.2. Software de aplicación

El programa propuesto debe de brindar la posibilidad de resolver los problemas mencionados en la sección 4.1 y promover nuevas funcionalidades que ayuden a mejorar la visualización y el seguimiento de la evolución del aprendiz conforme realiza prácticas con el manejo de los instrumentos. Para alcanzar estos objetivos, se mencionan las herramientas usadas para el proyecto.

El software utiliza los siguientes componentes de software:

- Librerías de procesamiento de imágenes
- Librerías de base de datos
- Librerías de visualización
- Entorno de desarrollo

### 4.2.0.1. Procesamiento de imágenes

A manera de conservar la consistencia con la última versión del programa y evitar problemas de compatibilidad, se utiliza **OpenCV 2.1** como parte de las librerías de procesamiento de imágenes. Esta versión sigue la misma firma de funciones tal como

se encuentra documentado por Bradski [7] y es fácilmente configurable para trabajar con distintos sistemas operativos y ambientes de desarrollo.

El lenguaje para la programación seguirá siendo **C** para la declaración de variables y el manejo de funciones, pero para permitir la generación de componentes independientes se incluyeron patrones de desarrollo generalmente usados en programación orientada a objetos. Este modelo permite separar la estructura de las funciones y la funcionalidad en archivos distintos, por lo que el desarrollo manejará este método de trabajo.

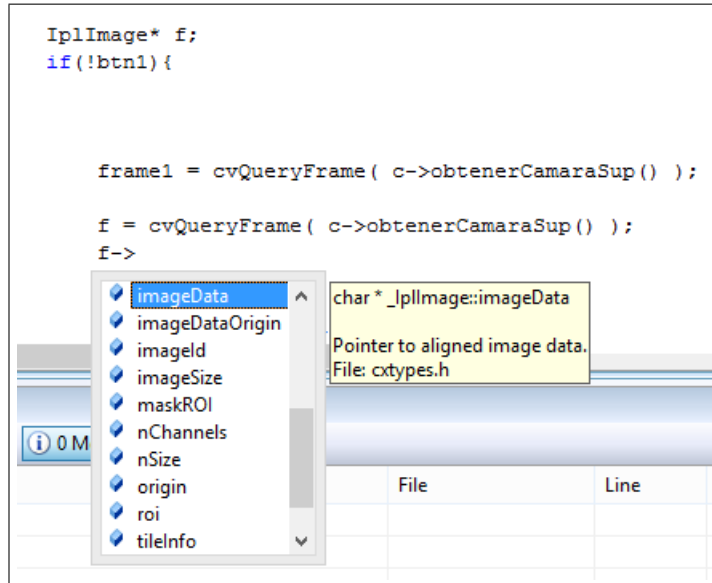
#### 4.2.0.2. Entorno de desarrollo

**OpenCV 2.1** tiene dos métodos principales de configuración:

1. *Compilación de código fuente*: en este modo se descarga el código fuente que incluyen todas las funciones, declaraciones y variables; estos archivos deben de ser compilados en algún entorno de desarrollo que incluya un compilador de **C/C++** para generar las librerías, archivos de configuración y ejecutables necesarios para el uso de las funciones. Este proceso toma tiempo y conduce a que el software deba incluir todo el código fuente de **OpenCV**.
2. *Instalación de las librerías para Visual Studio 2008*: en este método se descarga un archivo ejecutable para *Windows* que instala el código fuente, las librerías y algunos ejemplos en una carpeta dentro del sistema. Estas librerías ya se encuentran compiladas y permiten su inclusión en proyectos dentro de diferentes versiones de *Visual Studio*, no sólo la versión 2008.

Por facilidad, se empleó el segundo método ya que disminuye el tiempo y la complejidad de configuración. El archivo de instalación “*OpenCV-2.1.0-win32-vs2008.exe*” se puede descargar de SourceForge, en donde se encuentran los ejecutables y la documentación sobre la versión [26].

El entorno de desarrollo elegido fue *Visual Studio 2008*, debido a su ambiente con un editor gráfico de formas que agiliza el diseño y la programación de las ventanas que componen la interfaz bajo distintos lenguajes, como **Visual C#**, **Visual Basic** y **Visual C++**. Además, brinda algunas otras funcionalidades como lo es *Intellisense* que lista todos los métodos, atributos, eventos y la documentación de uso de cualquier componente que sea incluido. Al utilizar **OpenCV** con *Intellisense*, se agiliza la búsqueda de métodos y propiedades de cualquier variable manejada y agiliza el tiempo de desarrollo (Fig. 4.1).



**Figura 4.1:** Intellisense desplegando los métodos y atributos para una estructura *IplImage\** de **OpenCV**.

Aunque existen versiones más recientes de *Visual Studio* (*VS2010*, *VS2012* y *VS2014*), se eligió la 2008 debido a que las versiones posteriores presentan problemas de compatibilidad entre *Intellisense* y el lenguaje de programación **C/C++**, lo que complica la visualización de las variables y métodos.

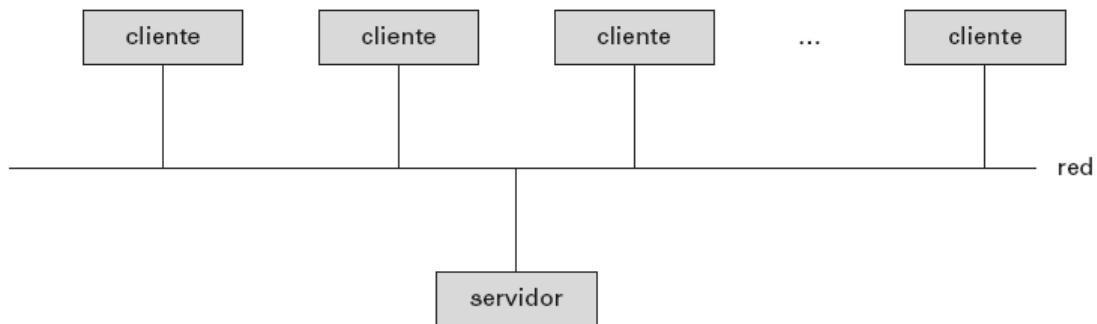
#### 4.2.0.3. Base de datos

El sistema tiene como objetivo recolectar información de manera ordenada sobre las tareas realizadas por el cirujano aprendiz, para lo que es necesario tener un software de registro que nos permita recuperar, insertar y modificar información de los usuarios; el medio más apropiado es a través de un *sistema gestor de base de datos*.

Un *sistema de base de datos* se define como la colección de datos interrelacionados, estructurados y organizados, y el conjunto de programas para acceder a ellos; estos elementos en combinación disminuyen la dificultad de acceso y redundancia, además de que proveen aislamiento, mejor acceso e integridad de la información [46, p. 7].

Generalmente los sistemas gestores de bases de datos en su arquitectura más básica de cliente-servidor requiere que un servidor o mainframe de base de datos (Fig. 4.2), el cual debe de satisfacer las peticiones de los clientes o terminales que se encuentren conectados al servidor a través de red a manera de conservar una

estructura de datos centralizada [50].



**Figura 4.2:** Estructura general de un sistema cliente-servidor. Imagen tomada de A. Silberschatz [50].

El software de base de datos depende de diversos programas y servicios incluidos por defecto en los sistemas gestores de bases de datos más comunes, tales como son **Oracle**, **MySQL** o **PostgreSQL**; lo que implica que el servidor tenga que disponer gran cantidad de recursos de memoria y procesador. Esto acarrea un problema debido a que se desea que la computadora central disponga de los recursos necesarios debido a la alta carga de trabajo y memoria que implica el procesamiento de imágenes.

Es necesario contar con un sistema de base de datos minimalista que consuma pocos recursos de cómputo, con requisitos mínimos de instalación y de configuración y permita la portabilidad del software.

**Librería de base de datos (SQLite)** Existe una gran diversidad de sistemas gestores de bases de datos que permiten gran portabilidad para diversos entornos de desarrollo; sin embargo la librería de bases de datos elegida para satisfacer las necesidades de desarrollo fue **SQLite 3**.

**SQLite 3** es una librería que implementa un motor de base de datos y que se encuentra liberada bajo "*licencia de dominio público*"; por lo tanto, es gratis para su descarga y uso en sistemas de aplicación, ya sean de carácter público, comercial o privado.

El sistema cuenta con las siguientes características:

- *Autocontenido*
- *Minimalista*
- *No requiere de un servidor*

- *No necesita configuración*
- *Transaccional*

**Autocontenido** Diversos sistemas de bases de datos utilizan diversos archivos para controlar la estructura, datos, operaciones, respaldos de la información y sus correspondientes procesos; a diferencia de **SQLite**, pues este no utiliza procesos separados para controlar los datos, ya que todo es gestionado por la misma librería. Más aún, tanto los datos como la estructura de la base se encuentra escrito directamente en un sólo archivo: tablas, índices, triggers y vistas.

**Minimalista** La escritura de un sólo archivo que contenga la información de toda la base de datos sigue el mismo principio de autocontenido y de diseño minimalista, ya que no se requiere de recursos adicionales para controlar la información.

La simplicidad del diseño también se encuentra relacionada directamente con la compacidad del sistema gestor, ya que la base de datos con todas sus características funcionales pueden ser contenidas en menos de *500KiB*, dependiendo de la plataforma de desarrollo. Si se desactivan algunas características opcionales, el tamaño de la librería de **SQLite** se puede reducir hasta *300KiB*.

**No requiere de un servidor** La librería no requiere de instalación dentro de la computadora, ya que se utiliza como una librería tipo in-process; es decir, que se abre, manipula y cierra de acuerdo a las necesidades de consulta del usuario a través de una aplicación. De la misma manera, tampoco requiere de ningún tipo de comunicación vía red para crear una arquitectura de cliente-servidor, ya que está implementado para utilizarse en sólo un equipo para su uso.

**No necesita configuración** La base de datos no sigue procesos de otros sistemas gestores como el inicio, paro o configuración; puede ser utilizada una vez creadas las tablas a utilizar, ya que no necesita la creación de usuarios o password para su modificación. Del mismo modo, tampoco utiliza archivos de configuración para controlar las modificaciones de la base debido a que todo está autocontenido en el archivo de datos.

En *Visual Studio 2008*, el único requisito necesario es la inclusión del archivo de librería y un archivo de declaraciones dentro del proyecto de desarrollo.

**Transaccional** Las bases de datos deben de procurar que los datos mantengan las características de atomicidad, consistencia, aislamiento y durabilidad para la estructura de los mismos. Es por esto que **SQLite** implementa transacciones que mantienen la información inclusive si el programa que maneja la base es interrumpido.

**Librería de base de datos** La base de datos se puede descargar del sitio **SQLite** en donde se encuentran las librerías para *Windows* para 32 y 64-bits (*sqlite-dll-win32*). Dentro se encuentran las definiciones (archivo *.DEF*) necesarias para generar la librería estática (archivo *.LIB*); una vez generada, se incluye este último para poder acceder a la base de datos.

Para poder incluir todas las variables y funciones de la librería dentro de las sugerencias de *Intellisense* y poder utilizar la librería, se requiere incluir el archivo previamente compilado *sqlite.lib* y el archivo de cabecera *sqlite3.h* [1, p. 22].

#### 4.2.0.4. Visualización

La idea de utilizar un entorno gráfico como *Visual Studio 2008* es, no sólo mejorar la usabilidad del programa a través de componentes visuales, sino que estos auxilien a la labor de los instructores a visualizar mejor los datos para evaluar el avance de los aprendices.

El *GUI* de *Visual Studio 2008*, por defecto contiene un componente llamado *Chart* para visualización de diagramas tales como diagramas de cajas o dispersión, gráficas de barras, stock, de dona, pirámide, de líneas o de pay [32]. Sin embargo, el componente se encuentra muy poco documentado, además de que existen pocos ejemplos para el manejo y manipulación de datos en el lenguaje **C/C++**; por esta razón, se eligió utilizar otra librería de visualización que facilite el manejo de datos y la exportación de gráficas.

**Librería de visualización (ZedGraph)** **ZedGraph** es un conjunto de clases escritas en **C#** para la creación de gráficas de diferentes tipos, altamente configurable y sencillo de usar. Las clases incluyen código para elegir el rango apropiado de escala y los tamaños de los incrementos para los valores de datos que se van a graficar. Adicionalmente, este tipo de contenedor para gráficos tiene la posibilidad de arrastrar, hacer zoom, restaurar el tamaño original, desplegar los datos para ciertos indicadores y poder exportar los gráficos a modo de imagen dentro del portapapeles.

La librería puede incluirse fácilmente dentro del entorno de desarrollo *Visual Studio* para lenguajes tales como **C/C++** y **Visual Basic**.

### **4.3. Estimación de parámetros de movimiento**

El desarrollo de distintos tipos de entrenadores laparoscópicos ha permitido a los cirujanos practicar diversas tareas que refuerzan las habilidades necesarias para convertirse en laparoscopistas expertos. Sin embargo, la cuestión para el desarrollo de mejores modelos educativos es encontrar cuáles son los factores que promueven el aprendizaje de manera más ágil.

Existen dos puntos principales sobre los cuales se supone reside el aprendizaje:

1. El conjunto de tareas que son ejecutadas por el aprendiz que le permiten mejorar su nivel de comfort y confianza en el entorno de trabajo de laparoscopia.
2. La evaluación de las tareas de acuerdo a su desempeño en la realización y el alcance de los objetivos.

#### **4.3.1. Parámetros de movimiento**

La creación de entrenadores de realidad virtual conllevó a crear ambientes animados que permiten simular tareas con objetivos específicos dentro del escenario. Como refuerzo al aprendizaje, las tareas incluían un criterio de evaluación de acuerdo a los resultados finales de la tarea.

La evaluación se consideraba como subjetiva al diseñador del entrenador, ya que los resultados provistos eran muy particulares del ambiente y los instrumentos utilizados dentro de la tarea; esto no permitía comparar los resultados con otros equipos que no siguieran los mismos parámetros de evaluación. Esta razón fue por la que muchos sistemas de entrenamiento utilizaron como único parámetro de comparación el tiempo que le tomaba al individuo en concluir la tarea [8, 9, 53, 57].

La necesidad de desarrollar un nuevo conjunto de métricas tenía que satisfacer la evaluación, no sólo el resultado, sino también del desempeño durante la realización, los factores de éxito o falla del desempeño y facilitar la identificación y cuantificación de la diferencia entre expertos y novatos [13].

La primer aproximación para el diseño de las métricas fue el movimiento del instrumento, ya que reflejan la adaptación al campo visual-espacial que sufre el



cirujano. Sin embargo, también se observó que no debe descartarse el tiempo de la tarea, ya que este refleja uno de los principales parámetros de éxito para la cirugía [13]

Cotin fue el primero en sugerir estimaciones para métricas de movimiento (Tabla 4.1), la cuales fueron probadas en el sistema de *Virtual Laparoscopic Interface* que tenía la capacidad de percibir cinco grados de libertad para estimar los parámetros.

Parámetro	Fórmula	Descripción
Tiempo (Time)	$P1 = T$	El tiempo total requerido para realizar la tarea (ya sea concluida o no).
Longitud de la trayectoria (Path Length)	$P2 = \int_0^T \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2} dt$	Es la distancia de la curva descrita por la punta del instrumento a lo largo del tiempo.
Suavidad del movimiento (Motion Smoothness)	$P3 = \frac{\sqrt{\frac{1}{2} \int_0^T \left(\frac{d^3x}{dt^3}\right)^2 dt}}{T}$	La suavidad se describe como el cambio de aceleración y es mínima en movimientos suaves. Debido a que su valor aumenta junto con el tiempo, se normaliza utilizando la división entre T
Percepción de profundidad (Depth perception)	$P4 = \text{no definida}$	Se mide como la distancia total que el instrumento viaja a lo largo de su eje.
Respuesta a la orientación (Response orientation)	$P5 = \sqrt{\int_0^T \frac{d\theta^2}{dt} dt}$	La cantidad de rotación a lo largo del eje del instrumento. Sirve para ilustrar la habilidad de colocar la herramienta en la orientación apropiada.

**Tabla 4.1:** Métricas propuestas por S. Cotin [13].

Los parámetros de movimiento fueron popularizados para comparar diversos equipos; muchos autores inclusive usaron los mismos parámetros, mientras que algunos fueron modificados para proponer nuevas métricas para comparar con las ya existentes y elegir aquellos que resultaban más discriminantes.

En el 2007, Chmarra incluyó dos modificaciones del parámetro de suavidad del movimiento para medirlo tridimensionalmente y a lo largo de su eje utilizando el equipo TrEndo cuando él o un asistente realizan la sujeción de la cámara [10]. Posteriormente,

propuso algunos otros parámetros que combinaban el movimiento angular sobre diferentes planos para obtener métricas basadas en volumen.

Parámetro	Fórmula	Descripción	Año
Suavidad del movimiento <b>3D (3D Motion Smoothness)</b>	$\sqrt{\frac{1}{2} \int_0^T \left( \left( \frac{d^3x}{dt^3} \right)^2 + \left( \frac{d^3y}{dt^3} \right)^2 + \left( \frac{d^3z}{dt^3} \right)^2 \right) dt}$	La suavidad se describe como el cambio de aceleración y es mínima en movimientos suaves. Debido a que su valor aumenta junto con el tiempo, se normaliza utilizando la división entre T	2007 [10]
Suavidad del movimiento <b>1D a lo largo del eje (3D Motion Smoothness along the Axis)</b>	$\sqrt{\frac{1}{2} \int_0^T \left( \frac{d^3h}{dt^3} \right)^2 dt}$	Parámetro basado en la tercer derivada de la posición del instrumento moviéndose a lo largo de su eje (dirección $h$ ).	2007 [10]
Área angular (Angular area)	$AA=(Max(\alpha)-Min(\alpha))(Max(\beta)-Min(\beta))$	Parámetro relacionado con las distancias entre las posiciones más lejanas mientras se realiza la tarea, donde $\alpha$ es la posición angular del tercer grado de libertad del instrumento y $\beta$ es la posición angular en el cuarto grado de libertad.	2010 [12]
Volumen (Volume)	$V = AA(Max(h) - Min(h))$	Relación de distancias entre la posición más lejana del instrumento con el área angular, donde $h$ es la distancia a lo largo del eje.	2010 [12]

**Tabla 4.2:** Métricas propuestas por M. Chmarra y los años correspondientes.

Otros autores en años recientes utilizaron los parámetros propuestos por los autores anteriores y sugirieron algunos otros que incluían cualidades como la destreza ambidiestra, medidas promedio y el análisis de como está compuesto el movimiento.

El método de trabajo para realizar las métricas en una tarea era a través de mediciones individuales para cada mano; sin embargo, algunos autores proponen el uso del promedio de las mediciones de ambas manos para obtener una sola métrica.

Parámetro	Fórmula	Descripción	Referencia
Destreza ambidiestra (Bimanual dexterity)	$\frac{\sum_{n=1}^N (V_L(n) - \bar{V}_L) (V_R(n) - \bar{V}_R)}{\sqrt{\sum_{n=1}^N (V_L(n) - \bar{V}_L)^2 + \sum_{n=1}^N (V_R(n) - \bar{V}_R)^2}}$	La habilidad del participante de controlar dos instrumentos al mismo tiempo medida de acuerdo a la velocidad de la punta de los instrumentos, donde $v$ es la velocidad de los instrumentos y $\bar{v}$ la velocidad media	[24]
Número de submovimientos (Number of Submovements)	$NoS$	Un submovimiento es definido por un movimiento de la punta del instrumento menor a 15 mm/s.	[24]
Velocidad promedio (Average Velocity)	$\frac{1}{T} \int_0^T \frac{d r(t) }{dt}$	La velocidad promedio de la velocidad de la punta del instrumento	[24, 40]
Aceleración promedio (Average Acceleration)	$\frac{1}{T} \int_0^T \frac{d^2 r(t) }{dt}$	La velocidad promedio de la velocidad de la punta del instrumento	[40]
Tiempo muerto (Idle Time)	$\frac{ \mathfrak{S} }{T} : \mathfrak{S} = \{t \in (0, \dots, T)   \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2} \leq 5\}$	Porcentaje de tiempo donde el instrumento se considera sin movimiento	[40, 41]
Economía de movimiento (Economy of Movements)	$EOM = \frac{ShortestDistance}{TotalDistance}$	La razón entre la distancia más corta para una tarea con respecto a la distancia total recorrida	[41]
Longitud angular (Angular Length)	$AL = \int_0^T \sqrt{\left(\frac{d\alpha}{dt}\right)^2 + \left(\frac{d\beta}{dt}\right)^2} dt$	El cambio total de los ángulos en la punta del instrumento.	[24]

Tabla 4.3: Métricas propuestas por diversos autores en años recientes.

### 4.3.2. Parámetros seleccionados para la aplicación

Los parámetros de movimiento elegidos para ser usados dentro de la aplicación son:

1. Tiempo
2. Longitud del recorrido (*Path Length*)
3. Percepción de profundidad (*Depth Perception*)
4. Suavidad del movimiento (*Motion Smoothness*)
5. Area Angular (*Angular Area*)
6. Volumen (*Volume*)
7. Velocidad Promedio (*Average Velocity*)

8. Aceleración Promedio (*Average Acceleration*)

9. Longitud Angular (*Angular Length*)

Estos parámetros fueron elegidos debido a que se adecúan a las mediciones que puede efectuar el sistema (posiciones y ángulos con respecto al punto de inserción) y no implican algún criterio subjetivo particular. Por ejemplo, el porcentaje de submovimientos requiere un valor de distancia mínima para determinar que el instrumento se encuentra estático.

Estos parámetros son implementados dentro del archivo `Parametros` a manera de función para que, dentro del sistema, se puedan recuperar las métricas obtenidas.

## 4.4. Corrección de ángulos

Como parte de las mejoras propuestas para el sistema de registro, se propuso mejorar la obtención de ángulo del instrumento con respecto al plano de referencia del entrenador. La versión anterior incluía una determinación de ángulos de acuerdo a los momentos centrales de su forma, de modo que si en algún instante el marcador obstruido visualmente entonces se podría acarrear una desviación en el ángulo posicional.

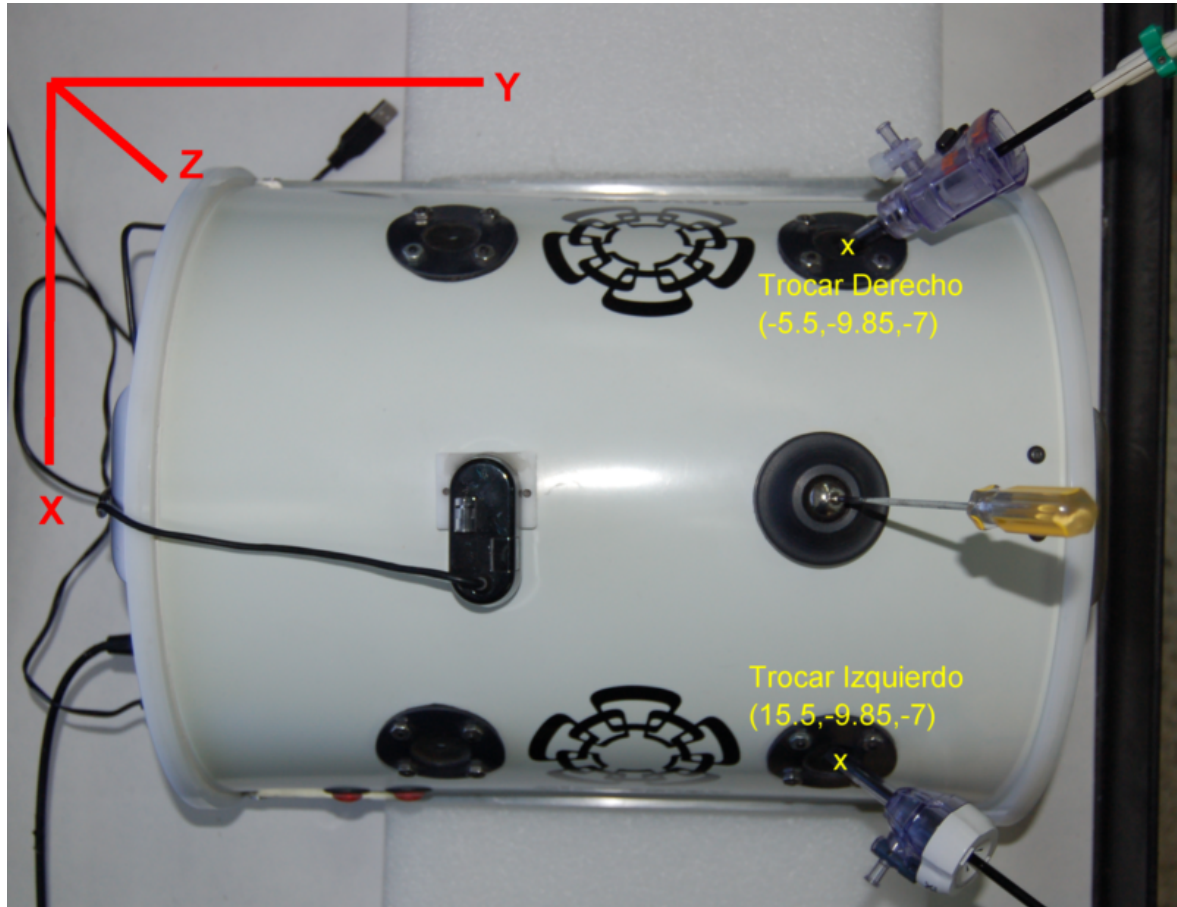
La manera de resolver este conflicto es localizando la entrada de los trócares en el espacio geométrico de referencia del entrenador de caja ( $\bar{T} = (x_T, y_T, z_T)$ ). Al obtener este punto, podemos trazar un vector cuyo origen se encuentre en el trocar y su extremo en la coordenada como:

$$\bar{V} = (x_R - x_T, y_R - y_T, z_R - z_T) \quad (4.1)$$

Una vez obtenido este vector, se puede obtener el ángulo con respecto a cualquiera de los planos mediante una proyección geométrica. Este tipo de estimación es apropiada para calcular ángulos y también desplazamientos axiales a lo largo del eje del instrumento.

## 4.5. Diseño de la aplicación

El modelo-vista-controlador es un patrón de desarrollo de software en el cual divide la parte del modelo de negocios de la interfaz visual (vista) y los intercomunica a través



**Figura 4.3:** Localización geométrica de las posiciones del trocar con respecto al espacio de referencia. Entrenador de la tesis de doctorado de F. Pérez.

de un controlador.

La parte de vista involucra el diseño de la interfaz que manipula directamente el usuario, la presentación de los datos y los controles que facilita el lenguaje de programación en el cual se está desarrollando; ya sea una aplicación de escritorio, web o móvil.

El modelo involucra un gestor de datos; es decir, el software y la estructura de datos que permite manejar la información relevante para quien diseña la aplicación. Esta se comunica a la vista mediante un controlador, que es quien atiende los eventos y actividades que realiza el usuario para enviar o recabar información a la parte del modelo.

En el proyecto se muestra que la parte del modelo es manipulada por la base de datos de **SQLite**, mientras que la vista y el controlador es implementada por el entorno de desarrollo de *Visual Studio*; por lo que podremos dividir la descripción de cada una de las partes por separado.

### 4.5.1. Descripción del modelo

Al arranque de la aplicación se inicia una clase llamada BD, la cual contiene los métodos apropiados para insertar, recuperar y borrar los datos de base. Al arranque de la aplicación se ejecuta el método de inicialización que permite verificar la existencia de las tablas del sistema y, en caso de no existir, crearlas. La base de datos es generada en un archivo con nombre BD.DB, que contiene toda la estructura y registros del sistema al iniciar el programa.

En la Fig. 4.4 se puede ver los métodos y atributos de la clase BD; que es la interfaz que permite la comunicación con la base de datos.

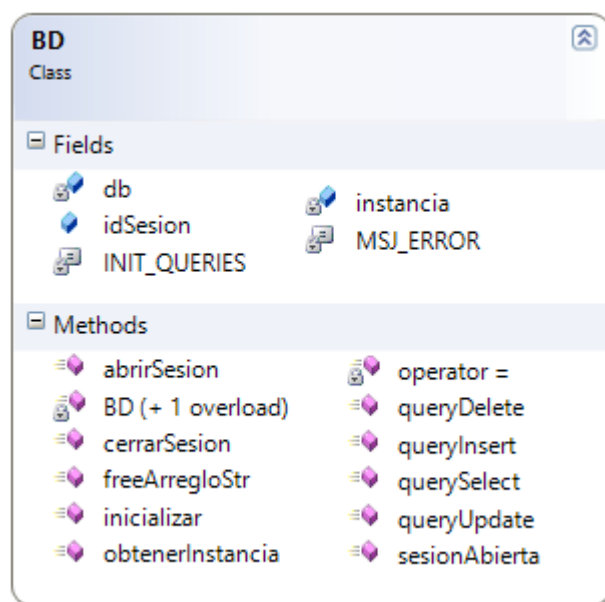


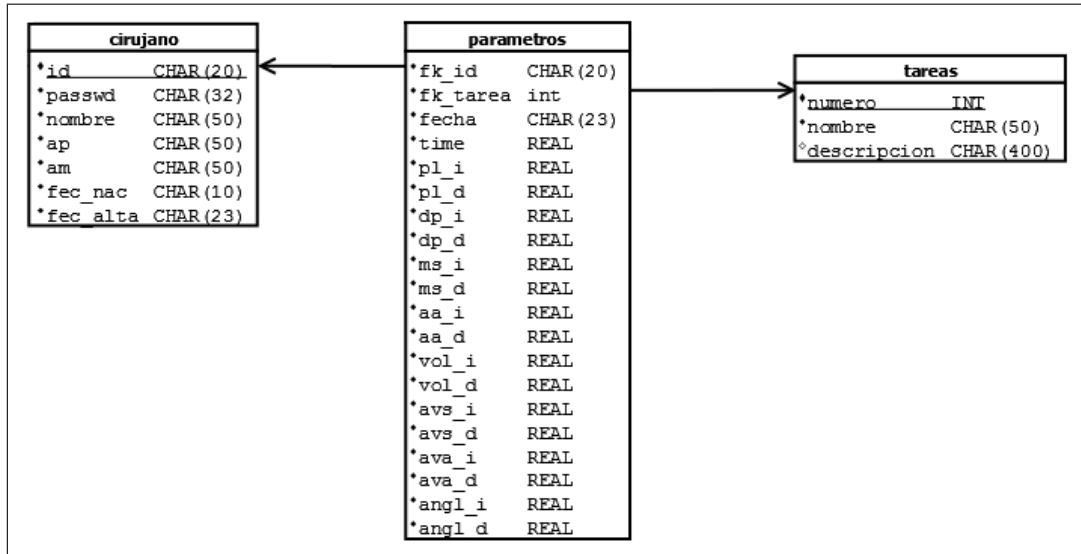
Figura 4.4: Diagrama de clases de la clase BD.

El modelo de la base de datos se compone de tres tablas creadas dentro del sistema de gestión de **SQLite** para el registro de las tareas, los usuarios y los parámetros de cada tarea efectuada. Las tablas emplean los tipos de datos permitidos por el sistema gestor, que en su mayoría son de tipo `double` y `char`.

En la Fig. 4.5 se muestra el diagrama relacional que se compone de las tres tablas del sistema: *cirujano*, *parametros* y *tareas*. Cada una tiene los campos necesarios para relacionar las tablas entre si y los datos de interés.

#### 4.5.1.1. Tabla 'cirujano'

La tabla de cirujano contiene los siguientes campos:



**Figura 4.5:** Diagrama relacional de la base de datos para las tres tablas dentro del sistema.

1. id: es un campo que tiene longitud máxima de 20 caracteres, el cual sirve para identificar al cirujano a través de un nombre clave.
2. passwd: es la clave que permite el ingreso al usuario, el cual es almacenado a través de una cadena hexadecimal de 32 caracteres, resultado del sistema de hash **MD5**.
3. nombre: el primer nombre de la persona, con un máximo de 50 caracteres.
4. ap: el apellido paterno, con un máximo de 50 caracteres.
5. am: el apellido materno, con un máximo de 50 caracteres.
6. fec\_nac: la fecha de nacimiento en formato YYYY-MM-DD.
7. fec\_alta: la fecha y hora de registro de alta en el sistema, el cual tiene el formato YYYY-MM-DD HH:mm:ss.sss, donde la hora se almacena como hora en formato de 24 horas, minutos, segundos y milisegundos.

#### 4.5.1.2. Tabla 'tareas'

La tabla de cirujano contiene los siguientes campos:

1. numero: el número de tarea corresponde a la clave numérica dada dentro del sistema para identificar la tarea a realizar.

2. *nombre*: es una cadena de máximo 50 caracteres para ingresar el nombre de la tarea.
3. *descripcion*: una cadena de máximo 200 caracteres para escribir un resumen de las actividades o mencionar el modo de como se debe realizar la tarea.

El programa cuenta con una etapa de inicialización de la base de datos, el cual incluye instrucciones para ingresar 3 tareas predeterminadas que son:

1. Transferencia
2. Sutura
3. Corte

De esta manera, si la base de datos no se encuentra o no está inicializada, se contará al menos con estas tres tareas iniciales.

### 4.5.1.3. Tabla 'parametros'

La tabla de parámetros tiene como objetivo el registrar algunos de los parámetros de movimiento más ampliamente usados. Estos se encuentran relacionados con las tareas y el usuario correspondientes y asignarles el tiempo de ejecución de la tarea.

En la aplicación se declararon los siguientes campos:

1. *fk\_id*: es el id correspondiente al usuario que se encuentra realizando la tarea. Tiene el mismo tamaño que el id en la tabla *cirujano*.
2. *fk\_tarea*: es la clave primaria de la tarea que se está realizando. Se guarda el número correspondiente a la tarea.
3. *fecha*: la fecha y hora del término del registro del cirujano en el sistema, el cual tiene el formato *YYYY-MM-DD HH:mm:ss.sss*, donde la hora se almacena como hora en formato de 24 horas, minutos, segundos y milisegundos.
4. *time*: el tiempo total transcurrido para la realización de la tarea.
5. *pl\_i,pl\_d*: es el *path length* o longitud del recorrido para la mano izquierda y derecha.
6. *dp\_i,dp\_d*: es el *depth perception* o percepción de profundidad para la mano izquierda y derecha.



7. *ms\_i,ms\_d*: es el *motion smoothness* o suavidad del movimiento para la mano izquierda y derecha.
8. *aa\_i,aa\_d*: es el *angular area* o área angular de toda la trayectoria de la mano izquierda y derecha.
9. *vol\_i,vol\_d*: es el volumen de la mano izquierda y la mano derecha.
10. *avs\_i,avs\_d*: es el *average speed* o velocidad promedio de la mano izquierda y derecha.
11. *ava\_i,ava\_d*: es el *average acceleration* o aceleración promedio de la mano izquierda y derecha.
12. *angl\_i,angl\_d*: es el *angular length* o longitud angular de la mano izquierda y derecha.

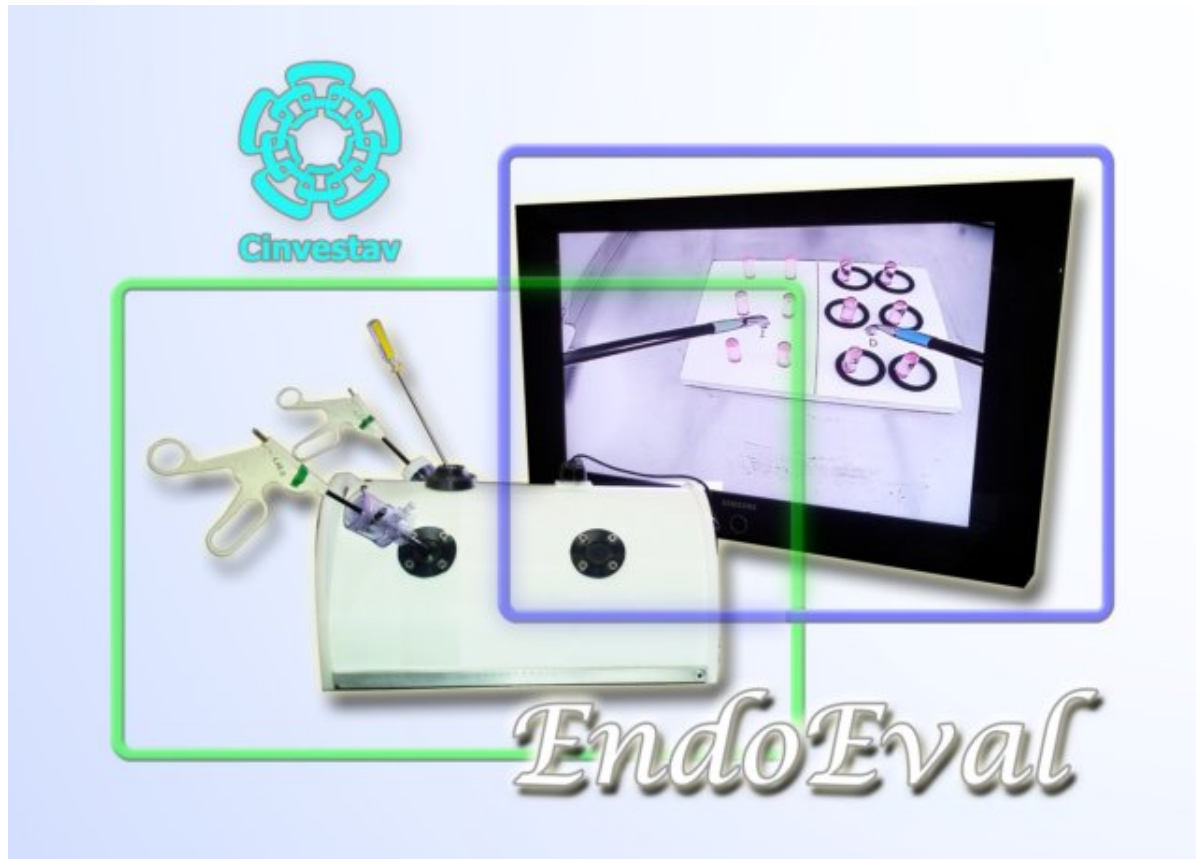
#### 4.5.2. Navegación de la vista y su descripción funcional

La aplicación se compone de los siguientes componentes de programación:

1. *10 formas de Windows*: constituyen el sistema visual que maneja un sistema de navegación a través de la apertura y cierre de cada una de las interfaces dependiendo de la tarea que se desee realizar.
2. *2 clases estáticas*: estas clases se encargan del manejo de las cámaras y de la base de datos desde cualquiera de las formas previamente mencionadas, de modo que sirvan de manera global a la aplicación.
3. *2 funciones en C*: estas dos funciones se encargan de la encriptación del password de los usuarios y la estimación de los parámetros de movimiento.

Los componentes se encuentran integrados para facilitar el desarrollo de nuevas funcionalidades y, a través de ellos, manipular los instrumentos de captura y programación como las cámaras, base de datos e interfaces visuales.

A manera de entender el funcionamiento del programa, en este apartado se describirán el orden de aparición de cada una de las ventanas y las funcionalidades que contiene cada forma para así explicar el manejo del programa.



*Figura 4.6: Vista de la ventana de Splash.*

#### 4.5.2.1. Ventana de Splash

La ventana de splash es una ventana que se despliega al inicio de la aplicación que contiene el logo y nombre comercial del programa: **EndoEval**. La razón de este nombre es debido a que tiene como objetivo evaluar las habilidades motrices del cirujano al momento de realizar tareas endoscópicas.

La imagen del logo tiene como objetivo mostrar los elementos que forman parte de las tareas laparoscópicas y resaltar el entrenador de caja diseñado dentro del laboratorio. A su alrededor, se muestra un recuadro verde y uno azul, que representan el tono de los marcadores utilizados para la realización de las tareas.

**Características de la ventana de splash** La ventana de splash no tiene bordes para brindarle una mayor vista a la imagen del logo. Esta ventana se abre al iniciar el programa y automáticamente se cierra luego de 5 segundos; sin embargo, puede cerrarse antes dando un click sobre la imagen.

Una vez finalizada la ventana de splash, se despliega la ventana de

login (sección 4.5.2.2).

#### 4.5.2.2. Ventana de Login



**Figura 4.7:** Vista de la ventana de Login.

La ventana de login tiene como objetivo permitir el ingreso a la aplicación a usuarios registrados a través de un identificador y un password (Fig. 4.7).

En la pantalla se muestra un logo con forma circular y las cajas de texto para ingresar el identificador y el password. En caso de no contar con un nombre de usuario, en esta misma ventana se puede acceder a los componentes que nos permitan abrir la ventana para ingresar los datos de nuevo usuario.

En la ventana también se incluye la función de configurar la disposición de las cámaras web, de modo que al iniciar la aplicación ya se cuente con las cámaras listas para su uso.

**Características de la ventana de Login** En la pantalla principal contiene dos campos de texto: uno para el identificador del usuario y el otro para el ingreso del

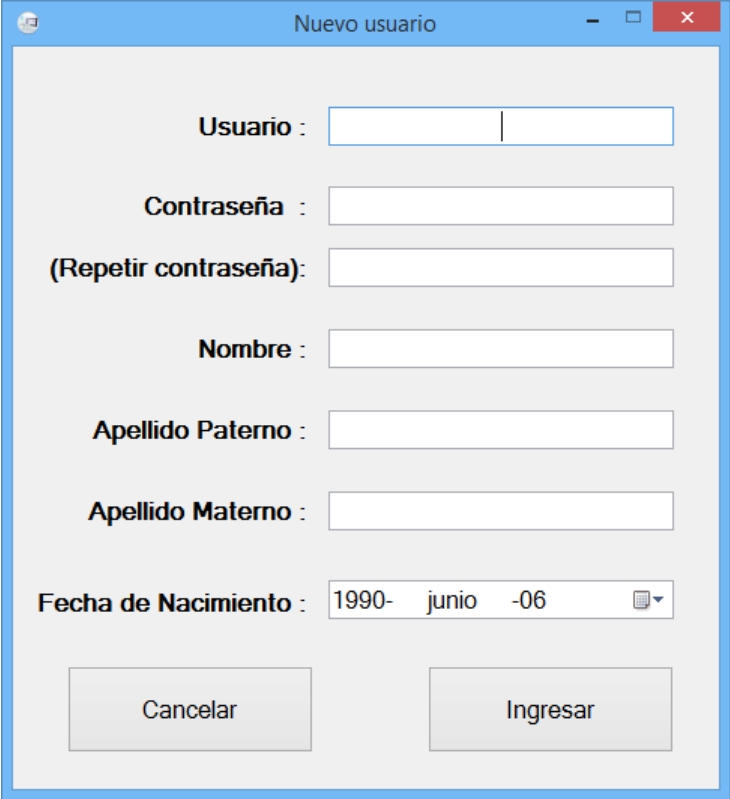
password. En la parte inferior se encuentra el botón para acceder al sistema una vez llenados los campos anteriores.

En caso de no poseer una cuenta en el sistema se puede dar click en la etiqueta de vínculo debajo del botón, el cual abre una ventana para crear un nuevo usuario.

En la barra de menús de la forma se encuentran las siguientes opciones:

- Usuarios: se encuentra una única opción llamada Agregar usuario, que despliega la forma para ingresar los datos necesarios para dar un usuario de alta (sección 4.5.2.3).
- Cámaras: en el menú se halla la opción Configurar cámaras, que muestra la ventana para la configuración de las cámaras de acuerdo a su colocación en el sistema (sección 4.5.2.4).
- Acerca de: despliega un resumen del proyecto de tesis.

#### 4.5.2.3. Ventana de Nuevo Usuario



The image shows a software window titled "Nuevo usuario". It contains a form with the following fields and controls:

- Usuario :** A text input field.
- Contraseña :** A text input field.
- (Repetir contraseña):** A text input field.
- Nombre :** A text input field.
- Apellido Paterno :** A text input field.
- Apellido Materno :** A text input field.
- Fecha de Nacimiento :** A date picker control showing "1990- junio -06".
- Cancelar** and **Ingresar** buttons at the bottom.

*Figura 4.8: Vista de la ventana de Nuevo Usuario.*

La forma incluye las cajas de texto necesarias para ingresar los datos del usuario y validar que sean correctos y los botones para introducir los datos al sistema o cancelar la creación de la nueva cuenta (Fig. 4.8).

La forma permite la validación de los datos, a manera que no se dupliquen nombres de usuario dentro de la base de datos y que los campos como usuario, nombre o apellido no sean nulos.

**Características de la ventana de Nuevo Usuario** Para guardar el usuario nuevo se debe verificar que los campos sean válidos para su almacén. La forma solicita los datos de nombre de usuario, contraseña, nombre, apellido paterno, apellido materno y fecha de nacimiento; los datos deben de cumplir las condiciones establecidas en la sección 4.5.1.1. Al terminar de ingresar los datos, se da click en el botón *Ingresar* para dar de alta los datos dentro de la base; de modo contrario, se puede dar click en *Cancelar* para abortar la operación.

Esta misma forma es utilizada para la opción de Actualizar datos, la cual se describe más adelante.

#### **4.5.2.4. Ventana de Configuración**

En la ventana de configuración de cámaras se muestran dos paneles en donde se despliegan las imágenes que son capturadas por cada número de cámaras(Fig. 4.9). El número de cámara es establecido por el control up-down que se encuentra en la parte inferior.

El objetivo de esta forma es visualizar cada número de cámara para hallar una correspondencia para indicar cual es la cámara superior y cual es la cámara lateral y establecer la configuración para ser usadas dentro del programa.

**Características de la ventana de Configuración** Los paneles de la forma contienen las leyendas para indicar cual es la cámara superior y cual es la cámara lateral. Los botones que se localizan en la parte inferior de cada panel permiten mostrar en el panel la cámara cuyo número (de acuerdo a la numeración de cámaras de **OpenCV**) se encuentra en el control up-down.

Para detener la visualización, se da click en el mismo botón y se cambia el número de cámara hasta encontrar la cámara correspondiente a la posición; las dos cámaras deben de estar abiertas y con número de cámaras diferentes.



**Figura 4.9:** Vista de la ventana de Configuración de Cámaras.

Para finalizar la configuración se da click en el botón Establecer cámaras para indicar que la configuración ha finalizado.

#### 4.5.2.5. Ventana de Inicio

Al ingresar al sistema vemos que el usuario puede elegir entre realizar alguna de las tareas predeterminadas por el sistema o analizar los datos de tareas anteriores; ambas tareas se encuentran en la pantalla principal de la ventana. Sin embargo, podemos utilizar la barra de menús para modificar los datos o dar de baja la cuenta.

**Características de la ventana de Inicio** En la pantalla se muestran dos actividades principales para el manejo de las prácticas de los cirujanos:

- Ejecución de tareas de práctica: en la primer caja de grupo se encuentran tres botones que corresponden a las tareas de transferencia, corte y sutura, que son tareas que se encuentran propuestas dentro del protocolo *MISTELS* [57].



**Figura 4.10:** Vista de la ventana del Menú de Inicio.

Para iniciar la tarea sólo debe dar click sobre el botón correspondiente, previa configuración de las cámaras; en caso de no estar configuradas, se desplegará la pantalla de configuración (sección 4.5.2.4).

- Análisis de los datos de prácticas anteriores: el botón de Registros y gráficas despliega una nueva ventana que contiene todas las ejecuciones del practicante y los parámetros correspondientes.

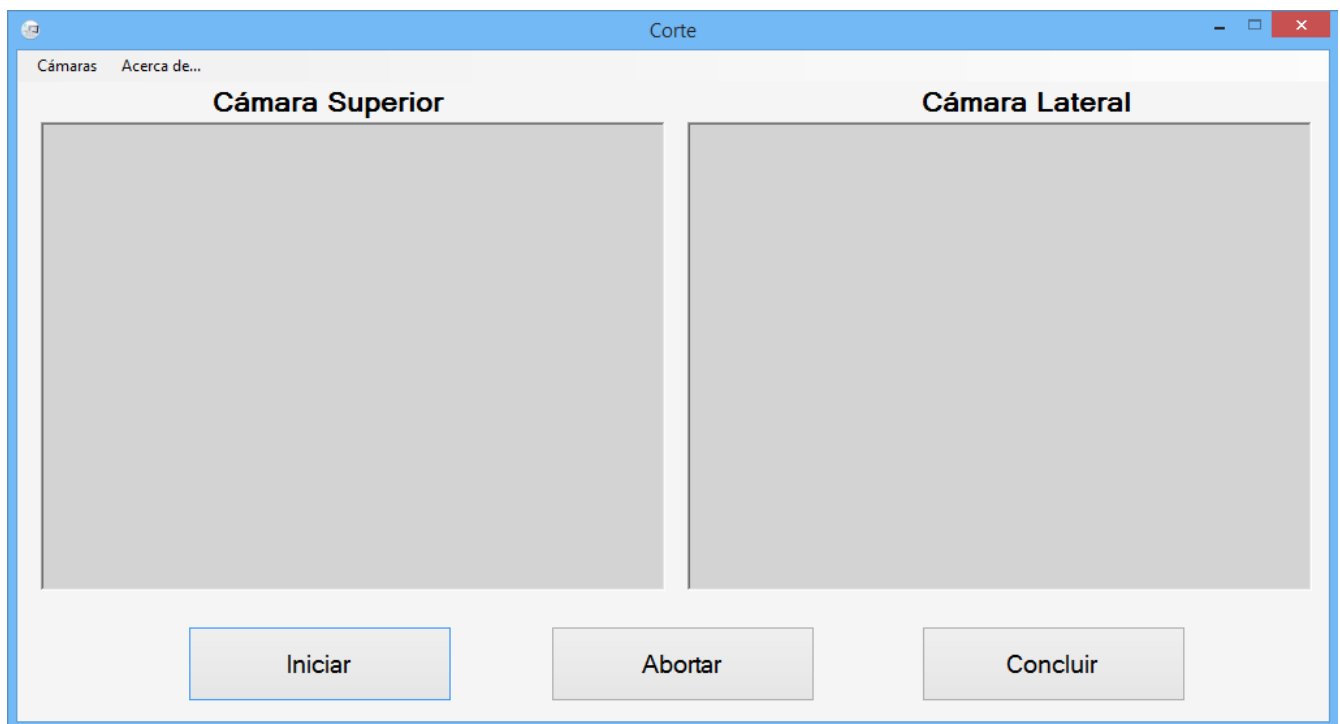
Para ayudar al cirujano novato a entender la tarea, se muestra en el botón una foto del escenario de ejemplo para que el practicante tenga una mejor referencia visual del espacio de trabajo. De la misma forma, al dejar el cursor sobre el botón se despliega un mensaje a manera de descripción del método de ejecución.

En el menú se muestran las siguientes opciones:

- Usuarios
  - Actualizar datos: despliega la forma de Nuevo Usuario (sección 4.5.2.3) pero con los datos del usuario actual para así modificarlos. Para actualizarlo requiere ingresar el password del usuario.

- Dar de baja mi cuenta: esta acción borra todos los datos de usuario y parámetros de las tareas ejecutadas por el cirujano. Para dar de baja el usuario requiere ingresar su password.
- Cámaras: contiene la opción Configurar cámaras como se describió en la sección 4.5.2.4.
- Acerca de: datos del equipo de desarrollo.

#### 4.5.2.6. Ventana de Tareas



**Figura 4.11:** Vista de la ventana de Tareas.

En la ventana se presenta un diseño similar a la configuración de cámaras, con la diferencia que los paneles son más grandes; esto sirve para observar la captura de cada una de las cámaras y mostrar algunos otros marcadores dentro de la aplicación.

Debajo de los paneles se encuentran botones para finalizar el registro de acuerdo a si se concluyó satisfactoriamente la tarea o si se desea cancelar el registro.

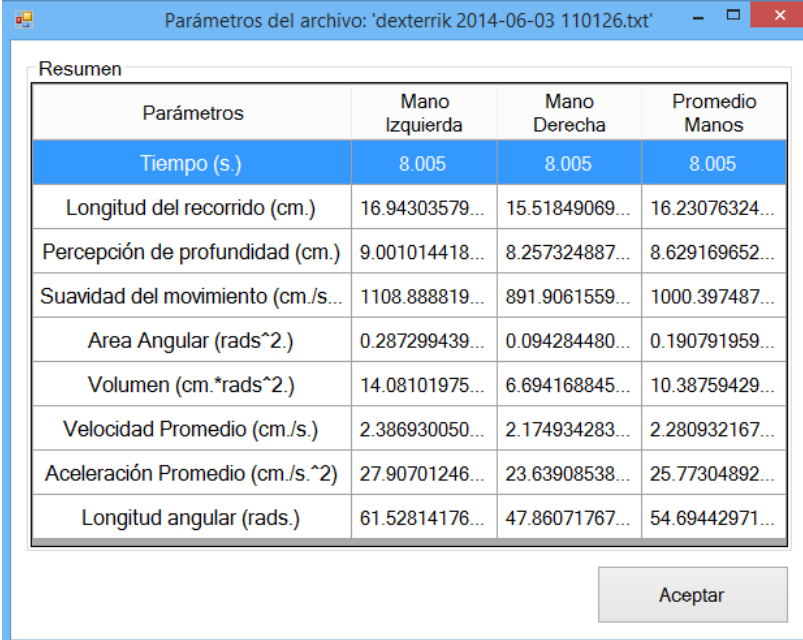
**Características de la ventana de Tareas** Los paneles se encuentran vacíos durante el inicio de la aplicación y sólo serán modificados hasta que se de click en el botón de



iniciar. Cuando se inicia el registro, cada p nel se actualizar  constantemente con la  ltima toma recuperada de la c mara correspondiente.

El registro se puede finalizar de dos formas:

- Bot n Concluir: al dar click en el bot n concluir se detiene la captura de las c maras, se cierra el archivo de registro y se llama a la funci n leerDatosArch que se encuentra en el archivo Parametros.cpp. En este archivo se implementan las funciones necesarias para recuperar los datos desde el archivo, obtener los par metros de movimiento e ingresarlos a la base de datos. Al finalizar se despliega una ventana con los par metros correspondientes a la tarea (Fig. 4.12).
- Bot n Abortar: termina la captura de las c maras, pero a diferencia del bot n Concluir, este borra el archivo de registros, de modo que no es procesado para estimar las par metros de movimiento.



Par�metros	Mano Izquierda	Mano Derecha	Promedio Manos
Tiempo (s.)	8.005	8.005	8.005
Longitud del recorrido (cm.)	16.94303579...	15.51849069...	16.23076324...
Percepci�n de profundidad (cm.)	9.001014418...	8.257324887...	8.629169652...
Suavidad del movimiento (cm./s...)	1108.888819...	891.9061559...	1000.397487...
Area Angular (rads <sup>2</sup> .)	0.287299439...	0.094284480...	0.190791959...
Volumen (cm.*rads <sup>2</sup> .)	14.08101975...	6.694168845...	10.38759429...
Velocidad Promedio (cm./s.)	2.386930050...	2.174934283...	2.280932167...
Aceleraci�n Promedio (cm./s. <sup>2</sup> )	27.90701246...	23.63908538...	25.77304892...
Longitud angular (rads.)	61.52814176...	47.86071767...	54.69442971...

**Figura 4.12:** Vista del resumen de par metros al terminar la tarea.

En el men  se muestran las siguientes opciones:

- C maras: contiene la opci n Configurar c maras como se describi  en la secci n 4.5.2.4.
- Acerca de: datos del equipo de desarrollo.



Figura 4.13: Vista de la ventana de Parámetros de Registros.

#### 4.5.2.7. Ventana de Parámetros de Registros

La forma tiene como objetivo combinar las librerías de base de datos y visualización para permitir a los cirujanos instructores el evaluar el desempeño y avance de los aprendices.

En la pantalla se visualizan las tareas, las fechas y los parámetros obtenidos para cada registro de manera separada (Fig. 5.1). Al seleccionar alguno de los parámetros disponibles se puede desplegar la gráfica correspondiente a cualquiera de los parámetros disponibles.

**Características de la ventana de Parámetros de Registros** La forma se constituye de componentes tipo tabla para separar los tres ámbitos que sirven para clasificar la actividad del cirujano:

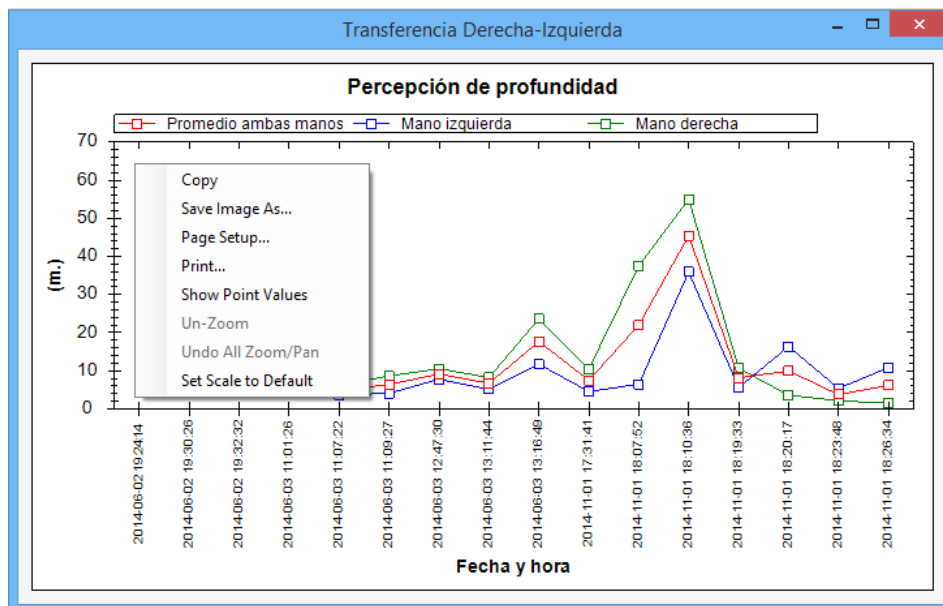
1. *Tipo de tarea:* son las tres tareas que permite el sistema que es transferencia, corte y sutura. Al seleccionar el renglón correspondiente a cualquiera de ellas, se actualizará el campo de fecha y de parámetros automáticamente.
2. *Fecha/hora:* al existir al menos una ejecución de la tarea, este componente despliega las fechas y hora en la que se ejecutó el tipo de tarea seleccionada. Al cambiar el renglón seleccionado se desplegará en la tabla de parámetros los valores de los parámetros para la fecha y hora seleccionados. Los datos pueden ser borrados a partir de los ítems de la barra de menús.

3. *Parámetros de movimiento*: es una tabla que se compone de 4 columnas que son el nombre y unidades del parámetro, el valor para la mano izquierda, el valor de la mano derecha y el promedio de ambas manos.

En la última tabla, sólo se puede seleccionar un sólo renglón; esto con el objetivo de elegir sólo un parámetro de interés que puede ser graficado a través de dar click en el botón que dice *Mostrar gráfica del parámetro seleccionado*. Al dar click en este, se despliega una ventana con la gráfica del parámetro para dicha tarea en todas sus ejecuciones.

La gráfica muestra el parámetro seleccionado y todas las fechas en las que se ha ejecutado: si el parámetro es *tiempo* sólo se muestra una línea roja que une todos los puntos correspondientes al valor de cada ejecución; si el parámetro es otro, se muestra en color rojo el promedio de ambas manos, en color verde el parámetro correspondiente a la mano derecha y en color azul el de la mano izquierda.

La gráfica de los parámetros se despliega utilizando la librería **ZedGraph** que, además de permitir la visualización de los datos, muestra otras opciones como lo son la visualización de las abscisas y ordenadas de cada punto, la ampliación y reducción del rango de datos, impresión de la imagen y la copia de la imagen en el portapapeles (Fig. 4.14).



**Figura 4.14:** Despliegue de la gráfica para el parámetro percepción de profundidad de la tarea de transferencia. En la pantalla se muestran las opciones de la gráfica para la librería **ZedGraph**.

En caso de requerir copiar más de un sólo renglón de la tabla de parámetros se puede seleccionar la casilla de verificación que se encuentra abajo de dicha tabla.

Esta permite seleccionar varias casillas de la tabla y así copiarlas a otra fuente como una hoja de cálculo o una bitácora de texto; sin embargo, esto deshabilita la graficación del parámetro seleccionado.

En el menú se muestran las siguientes opciones:

- **Registros:** contiene la opción **Borrar registro** que elimina todos los parámetros correspondientes a la fecha y hora que se encuentra seleccionada en la segunda tabla. Para borrar es necesario introducir el password del usuario.
- **Acerca de:** datos del equipo de desarrollo.

## 4.6. Resumen

En este apartado se describieron los alcances y limitaciones del último software de evaluación, a lo cual se sugirieron mejoras para incrementar la funcionalidad del programa como un sistema de entrenamiento.

Después fueron descritas las librerías necesarias para el procesamiento de imágenes, base de datos y visualización de los registros de manera adecuada. Al revisar que los componentes previos son compatibles con *Visual Studio 2008* se pudo consolidar una estación de trabajo completa.

Del mismo modo, fueron revisados distintos parámetros de movimiento y fueron elegidos los más adecuados para el entrenador de caja. Las características de los parámetros es que permiten obtener métricas de acuerdo a los ángulos y posiciones, los cuales pueden ser obtenidos a partir de nuestro sistema de triangulación. Posteriormente se describió la corrección correspondiente a los ángulos del instrumento para mejorar las métricas que incluyen el manejo de ángulos y posiciones axiales.

Finalmente se describieron las partes del modelo y la vista de la aplicación a través de un recorrido sobre la distribución de las tablas y las funciones de las ventanas de navegación. Así, el usuario puede llegar a entender el manejo del programa sin necesidad de detalle técnico.

# Capítulo 5

## Resultados

En este capítulo se presentan los resultados de evaluar los métodos de detección como de rastreo de instrumentos, así como del sistema para la estimación de parámetros de movimiento para algunas pruebas.

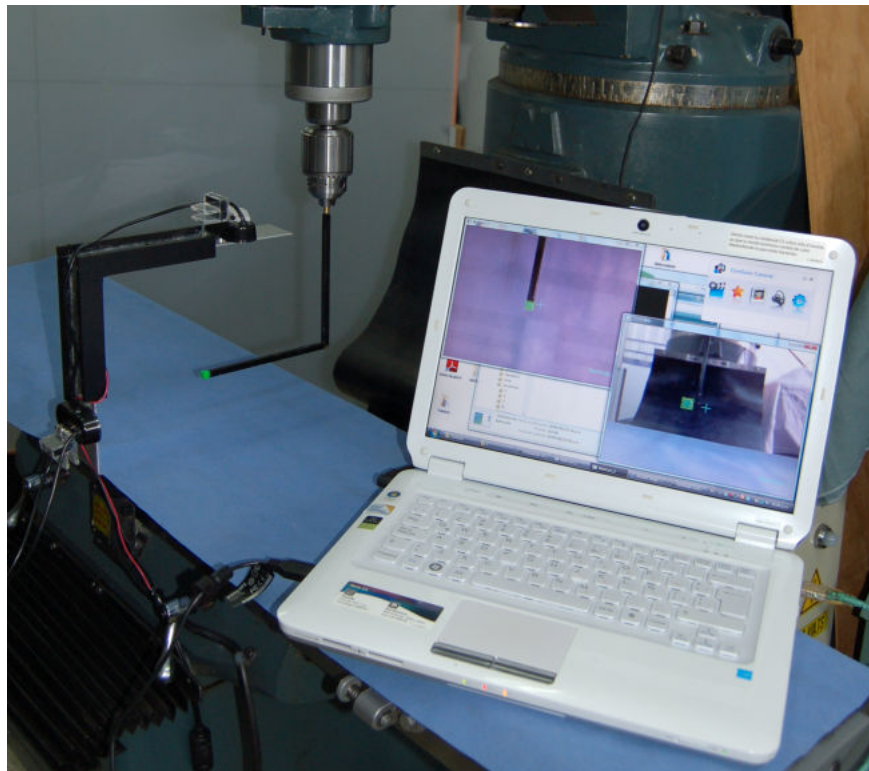
Inicialmente se hace una comparación de la evaluación de posicionamiento con respecto a previas verificaciones físicas con respecto al modelo matemático evaluado en la computadora(sec. 5.1) de acuerdo a su posición e cambio. Posteriormente, se evalúan los tiempos de muestreo para cuando el algoritmo de manejo de oclusiones es ejecutado para diversos casos de obstrucción visual (sec. 5.2).

Finalmente, se realiza una comparación de los parámetros de movimiento escogidos en sec. # estimados en el lenguaje **C/C++** con respecto a los obtenidos en **MATLAB** y se realiza la discusión de resultados para ambas pruebas (sec. 5.3).

### 5.1. Metodología de análisis de resultados para posicionamiento

En el trabajo de F. Pérez se había empleado una fresadora para realizar el desplazamiento de un marcador verde sobre cada eje de desplazamiento evaluar un cubo imaginario de  $5 \times 5 \times 5$  *cm*. La evaluación tuvo como objetivo el comparar puntos de referencia físicos con las coordenadas otorgadas por el modelo geométrico de cámara [43].

Los estudios realizados indicaron que el instrumento cuenta con una resolución de  $.616$  *mm*. en cada uno de sus ejes, que el modelo no presenta no linealidades y que los errores eran mínimos para los grados de libertad en los que se realizó la evaluación (Tabla 5.1). Sin embargo, la prueba considera puntos dentro el espacio



**Figura 5.1:** Imagen de los instrumentos de evaluación para la versión anterior del software. Imagen tomada de F. Pérez [43].

dentro del campo visual y no las fronteras visuales del dispositivo.

Puntos	Coordenadas 3D reales	Coordenadas 3D estimadas por el sistema	Error $\pm$ Desv. Estándar
Centro	(45.0,45.0,45.0)	(44.907, 45.077, 45.412)	0.429 $\pm$ 0.251
Vértice 1	(20.0, 20.0, 20.0)	(20.077, 19.803, 20.406)	0.491 $\pm$ 0.140
Vértice 2	(70.0, 20.0, 20.0)	(69.601, 20.330, 19.945)	0.568 $\pm$ 0.081
Vértice 3	(70.0, 70.0, 20.0)	(70.186, 69.959, 19.050)	0.974 $\pm$ 0.089
Vértice 4	(20.0, 70.0, 20.0)	(20.604, 70.232, 19.710)	0.722 $\pm$ 0.056
Vértice 5	(20.0, 70.0, 70.0)	(19.327, 70.842, 71.487)	1.859 $\pm$ 0.191
Vértice 6	(70.0, 70.0, 70.0)	(69.335, 70.752, 70.856)	1.337 $\pm$ 0.051
Vértice 7	(70.0, 20.0, 70.0)	(68.847, 20.622, 70.151)	1.377 $\pm$ 0.033
Vértice 8	(20.0, 20.0, 70.0)	(18.996, 20.295, 70.821)	1.330 $\pm$ 0.023
Error promedio			1.009 $\pm$ 0.101

**Tabla 5.1:** Coordenadas de prueba de cubo expresadas en milímetros. Tabla tomada de F. Pérez [43].

A manera de corroborar los resultados de manera analítica se decidió utilizar las ecuaciones del modelo matemático para observar el comportamiento de las coordenadas **3D** estimadas cuando las posiciones en la imagen del marcador se localizan en la frontera del área visual de la cámara; para esto se introdujeron las ecuaciones para estimar los valores de  $\bar{x}$  y se introdujeron distintos valores de coordenadas para el centroide del marcador.

La evaluación consiste en la modificación de las variables  $x_1$ ,  $y_1$  y  $y_2$  debido a que, como se mencionó en la sec. 3.3.3, el componente  $x_2$  no participa dentro del sistema para la resolución de las coordenadas **3D**. De los tres componentes, dos se les asignó un valor en la frontera de su posición y el tercer punto era evaluado a lo largo de todos los valores posibles (desde 0 al tamaño de dicha dimensión en la imagen); una vez calculadas las coordenadas **3D**, se calculaba la distancia a partir del punto actual con respecto al siguiente (por ejemplo:  $x_1 = 0$ ,  $y_1 = 0$  y  $y_2$  toma valores de  $[0,480]$ ).

La tabla 5.2 muestra todas las combinaciones realizadas para la caracterización del sistema: el eje dimensional que cuenta con un intervalo en vez de un valor único es el que fue utilizado para iterar el extremo del cubo visual. La columna **Rango de distancia** muestra un rango en donde el primer elemento es la distancia de la primera iteración con su punto subsecuente, mientras que el último valor del rango es la distancia **3D** de la posición correspondiente al penúltimo punto de iteración con la correspondiente al último punto del eje de evaluación.

Para verificar el trabajo se realizó un estudio para cubos de mayores dimensiones y centrados sobre el mismo origen: los cubos utilizados poseen dimensiones de  $5\text{ cm.}$ ,  $6\text{ cm.}$  y  $8\text{ cm.}$  Los resultados del recorrido fueron graficados en **MATLAB** y se muestran en la Fig. 5.2.

### 5.1.1. Discusión de la técnica de posicionamiento

Las mediciones efectuadas a través del desplazamiento de los marcadores mediante el uso de una máquina fresadora demuestra un error medio de posicionamiento de  $1\text{ mm.}$  y una resolución de  $.616\text{ mm.}$  en la dirección de cada eje; lo que no implica un error significativamente grande con respecto a las métricas que se desean adquirir del espacio de trabajo.

El trabajo menciona que el sistema se encuentra ausente de no linealidades; lo que es discutible debido a la discrepancia que existe con respecto a los errores obtenidos en la fresadora con respecto a las distancias mínimas que nos provee la prueba del modelo geométrico al iterar sobre la frontera visual de las cámaras. Este error se

Prueba	$x_1$	$y_1$	$x_2$	$y_2$	Promedio $\pm$ Desv. Est.	Rango de distancia
1	0	0	–	[0 – 480]	$0.0433 \pm 0.0075$	[0.0322, 0.0583]
2	0	480	–	[0 – 480]	$0.0188 \pm 0.0042$	[0.0274, 0.0128]
3	640	0	–	[0 – 480]	$0.0398 \pm 0.0061$	[0.0306, 0.0518]
4	640	480	–	[0 – 480]	$0.0179 \pm 0.0040$	[0.0263, 0.0121]
5	0	[0 – 480]	–	0	$0.0170 \pm 0.0035$	[0.0120, 0.0241]
6	0	[0 – 480]	–	480	$0.0396 \pm 0.0076$	[0.0549, 0.0285]
7	640	[0 – 480]	–	0	$0.0164 \pm 0.0032$	[0.0117, 0.0230]
8	640	[0 – 480]	–	480	$0.0369 \pm 0.0067$	[0.0503, 0.0270]
9	[0 – 640]	0	–	0	$0.0125 \pm 0.0020$	[0.0123, 0.0136]
10	[0 – 640]	0	–	480	$0.0414 \pm 0.0012$	[0.0441, 0.0400]
11	[0 – 640]	0	–	0	$0.0176 \pm 0.0030$	[0.0183, 0.0169]
12	[0 – 640]	480	–	480	$0.0300 \pm 0.0060$	[0.0314, 0.0286]
Distancia promedio					$0.0276 \pm 0.0046$	

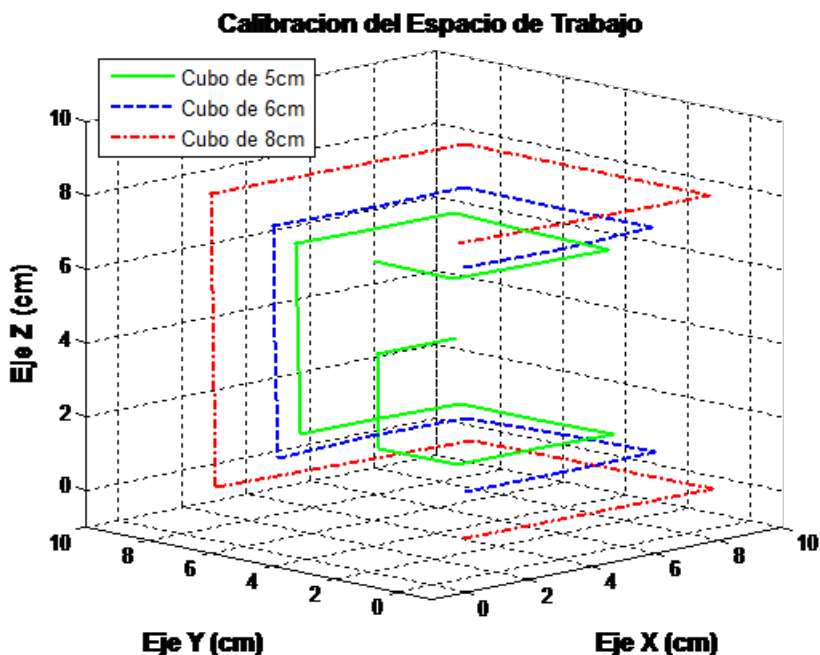
**Tabla 5.2:** Intervalos de distancias (cm.) al variar la posición del centroide a lo largo de diferentes límites de la imagen. Los valores del rango de distancia corresponden al incremento máximo y mínimo de distancias, que pertenecen al primer y último punto del intervalo del eje de variación.

supone puede ser provocado por la diferencia entre la disposición ideal de las cámaras (configuración ortogonal a 15 cm. de un vértice común) y la configuración real en el entrenador de caja, ya que esta última puede verse afectada por la distancia o ligeras desviaciones del ángulo distinta a los 90°.

La prueba con el cubo de 5 cm. en comparación con el modelo ideal presenta ligeras no linealidades que, por muy pequeñas, reflejan una perturbación mayor al momento de trabajar con infraestructuras reales. Más aún, estas distorsiones se verán magnificadas cuando el espacio de trabajo de los marcadores se aleja de ambas cámaras.

En el recorrido mostrado en la Fig. 5.2 se puede percibir que en realidad el sistema si presenta no linealidades; sin embargo, la distorsión no resulta muy significativa ya que el espacio de trabajo es igual al cubo de mayor dimensión (8 cm. de lado), por lo





**Figura 5.2:** Gráfica del recorrido del cubo de pruebas para dimensiones de 5, 6 y 8 cm.

que cualquier error podría reflejarse en forma porcentual como:

$$ErrorPorcentual = \frac{ErrorMedio}{Magnitud} = \frac{1 \text{ mm.}}{80 \text{ mm.}} \times 100 \% = 1.25 \% \quad (5.1)$$

La Fig. 5.2 en los recorridos para el cubo de mayor dimensión visualmente refleja una no linealidad de los ejes, lo cual difiere de lo reportado. Sin embargo, se puede ver que muchas de las distorsiones lineales ocurren cuando la posición de los instrumentos se alejan de ambas cámaras, como se puede ver en la prueba 1 y 10 de la tabla 5.2; esto debido al modelo de cámara pinhole.

Por lo anteriormente reportado se sugiere un especial cuidado de que la distancia y el ángulo de la estructura del montaje de la cámara se haga como lo propuesto en el modelo geométrico o, como solución, utilizar un sistema que incluya un proceso de recalibración para cuando el usuario así lo requiera.

## 5.2. Metodología de análisis de resultados para tiempo de muestreo

En los documentos referentes al sistema de visión no se encuentran reportados los tiempos de muestreo para el registro de tareas; por lo cual se decidió evaluar esta característica para distintos casos en los que los tiempos de procesamiento pudieran variar.

El procesamiento de imágenes para cada toma en general consta de las mismas etapas para la detección del instrumento. La única circunstancia que adiciona un procesamiento adicional es sólo cuando la pinza se encuentra ocluida visualmente en su parte media.

Para evaluar los tiempos de adquisición se utilizaron dos instrumentos laparoscópicos, cada uno con un marcador de distinto color. Estos instrumentos se colocaron dentro del entrenador para registrar tres condiciones distintas para los marcadores:

- Los dos instrumentos se encuentran dentro del entrenador sin oclusión
- Uno de los marcadores se encuentra ocluido en su parte media, mientras que el otro es visualmente accesible.
- Ambos marcadores de los instrumentos se encuentran ocluidos en su parte media.

Para ocluir el marcador se utilizó una pieza de metal color grisáceo que corta ambos instrumentos por la parte central, obligando a que el programa ejecute el proceso para la fusión de las regiones del mismo color (Fig. 5.3).

Se realizaron tres registros, uno para cada condición: todos los registros fueron hechos aproximadamente durante 60 segundos para recopilar información más confiable. Los resultados de los tiempos y frecuencia de muestreo se resumen en la Tabla 5.3.

Los registros fueron adquiridos en una computadora con procesador Intel Core i3-M350 con velocidad de 2.27 GHz, con 6.00 GB de memoria RAM y con sistema operativo *Windows 8.1 Pro 32-bits*, en donde el único proceso activo era aplicación de *EndoEval*.



**Figura 5.3:** Oclusión de los marcadores de ambos instrumentos en su parte media.

Característica del muestreo	Promedio $\pm$ Desv. Estándar del tiempo	Frecuencia de muestreo
Adquisición de ambos instrumentos sin oclusión	0.115865 $\pm$ 0.00515 s.	8.6307 Hz.
Adquisición de ambos instrumentos con corrección por oclusión de un marcador	0.11886 $\pm$ 0.00630 s.	8.4132 Hz.
Adquisición de ambos instrumentos con corrección por oclusión de dos marcadores	0.12325 $\pm$ 0.00589 s.	8.1136 Hz.

**Tabla 5.3:** Métricas con respecto el tiempo de muestreo para diferentes condiciones de oclusión.

### 5.2.1. Discusión los tiempos de muestreo

En los tiempos de muestreo vemos que el principal factor que afecta el tiempo es la fusión de los marcadores cuando estos se encuentran ocluidos, y más cuando la oclusión ocurre para ambos marcadores ya que se reduce 0.5 Hz. al muestreo obtenido cuando no existen obstáculos visuales. Pese a esto, los tiempos de muestreo no reflejan un cambio altamente notable en las tres condiciones pues se conserva una frecuencia mínima de 8 Hz.

Podríamos considerar los tiempos de muestreo como estables porque, inclusive para las tareas que involucran el cruce entre ellas, generalmente sólo ocurre la

obstrucción de una de ellas, pero no de ambas. Así, podemos suponer que los tiempos de muestreo oscilen sólo entre los tiempos para los primeros dos casos.

### 5.3. Análisis de los parámetros de movimiento del instrumento

Los parámetros de evaluación estimados por el software requieren compararse con alguna otra aplicación que nos permita manejar variables numéricas de manera muy precisa; de este modo, nosotros podremos establecer un valor de referencia con cuál comparar los parámetros obtenidos por nuestro sistema y verificar que tan similares son nuestras estimaciones según los parámetros de interés.

Para evaluar los parámetros, se diseñó una función en **MATLAB 2012RA** que en su entorno incluye diversas funciones para el procesamiento matemático y gran capacidad para el almacenamiento de variables numéricas. La función llamada `Params` recibe la matriz de datos y la posición de la pinza (“LG” o “RB” si el marcador verde se encuentra en la pinza izquierda, o cualquier otro valor para el caso contrario) y regresa diversos valores, como los valores de profundidad con respecto a su punto de inserción, ángulos de acuerdo a los planos  $XY$ - $XZ$  y los parámetros de movimiento que se desean calcular: el código se muestra en el anexo al final de esta tesis.

Para el análisis se realizaron 7 movimientos para ambas manos, lo cual dio un total de 14 mediciones por cada parámetro. Se obtuvieron las métricas para cada archivo mediante el sistema y la función de **MATLAB**; los resultados fueron medidos mediante la fórmula del error porcentual, en donde se consideró al valor de la función como valor real y a las métricas otorgadas por el sistema como valor aproximado.

Los resultados fueron clasificados por tarea y se muestran en la Tabla 5.4.

#### 5.3.1. Discusión de la estimación de parámetros

La tabla 5.4 demuestra que el error entre los resultados obtenidos entre **MATLAB** y la aplicación programada en **C/C++** no demuestran grandes diferencias y dando hasta 4 cifras porcentuales de precisión.

En ocasiones, la migración de un lenguaje de programación a otro implica la modificación en la precisión de los resultados debido a que cada uno tiene distintos manejos para variables de punto flotante. En este caso particular se alcanzó el objetivo de tener valores cercanos entre ambos lenguajes, lo cual se esperaba debido a que

Parámetro (unidades)	Error medio $\pm$ Desv. Estándar
Tiempo (s.)	0 $\pm$ 0 %
Longitud del recorrido (cm.)	0.0000000109 $\pm$ 0.0000000134 %
Percepción de profundidad (cm.)	0.0000000218 $\pm$ 0.0000000235 %
Suavidad del movimiento (cm./s. <sup>3</sup> )	0.0000000085 $\pm$ 0.0000000079 %
Area Angular (rads <sup>2</sup> .)	0.0000140032 $\pm$ 0.0000183003 %
Volumen (cm. * rads <sup>2</sup> .)	0.0000062919 $\pm$ 0.0000144020 %
Velocidad Promedio (cm./s.)	0.0000009930 $\pm$ 0.0000014959 %
Aceleración Promedio (cm./s. <sup>2</sup> )	0.0000001508 $\pm$ 0.0000002131 %
Longitud angular (rads.)	0.0000003345 $\pm$ 0.0000004945 %

**Tabla 5.4:** Errores porcentuales para cada uno de los parámetros de movimiento estimados.

**MATLAB** es un entorno programado bajo el lenguaje **C** [36]; lo que indica que las capacidades de las variables deberían ser similares.

La ventaja de utilizar el sistema *EndoEval* es que dispone de un entorno amigable para los cirujanos novatos que además facilita la estimación de parámetros para la evaluación y seguimiento de estos conforme la práctica lo permite. Más importante aún, facilita la exportación de datos a través de archivos independientes que permiten ampliar las métricas a otros lenguajes y entornos, como lo puede ser **MATLAB**.

## 5.4. Resumen

En este apartado se describieron las características a evaluar como el posicionamiento, el tiempo de muestreo y los parámetros de movimiento obtenidos a través del sistema.

Se abordó la resolución física y la resolución ideal que debería de tener el sistema y se describió como mínimo el error que presenta el sistema para pequeñas desviaciones de la configuración ideal. Sin embargo, se evaluó por primera ocasión el tiempo de muestreo dentro para diversas condiciones que pueden ligeramente afectar la frecuencia de trabajo.

Finalmente, se reportaron las diferencias obtenidas mediante la estimación de parámetros a través de **MATLAB** con los obtenidos mediante el sistema *EndoEval*, mostrando un error casi nulo para todos sus parámetros.



# Capítulo 6

## Conclusiones y trabajo futuro

Este último capítulo está dividido en (1) las conclusiones obtenidas del presente proyecto de investigación para el sistema *EndoEval*, (2) se enuncian las contribuciones más significativas en el área tecnológica y hospitalaria, finalmente, (3) se proponen formas en la que este trabajo puede ser extendido.

### 6.1. Conclusiones

Esta tesis utiliza un entrenador de caja para la detección y rastreo de instrumentos laparoscópicos a través de un procesamiento de imágenes y un conjunto marcadores de color.

El proyecto adiciona funcionalidades a la última versión del software, como la inclusión de una técnica que evita la separación de regiones cuando estas son ocluidas visualmente en su parte media por otro objeto; lo cual proporciona una mejor estabilidad al momento de identificar la posición del centroide del objeto. También se agregó un nuevo método de estimación para los ángulos del instrumento de acuerdo al punto de inserción del trocar, lo que brinda un menor ruido que la técnica anterior que empleaba los momentos de la imagen para hallar la orientación.

Anteriormente se requería que el usuario iniciara con el color de un marcador de un lado específico para evitar la confusión del sistema, ya que sólo distinguía por color y no por el lado de la mano en la que se encontraba. Esta condición fue corregida y mejorada, de modo que el software permite la extracción de los instrumentos para intercambiarlos de posición sin generarse confusión durante su aparición/desaparición.

Las condiciones de evaluación demuestran que el sistema tiene una alta resolución para la captura de la posición **3D**; sin embargo, requiere que el sistema cumpla

adecuadamente con las características de ángulo y distancia para maximizar la precisión del instrumento.

El sistema provee con una interfaz que permite la recopilación y visualización de datos para distintas ejecuciones de tareas, lo que brinda mayor orden y mejor seguimiento del avance de los cirujanos. La generación de archivos separados de registro facilita la ampliación del análisis de parámetros motrices mediante otras herramientas de análisis: como hojas de cálculo o sistemas de procesamiento.

El estudio de validación con las métricas obtenidas por el sistema y las estimadas por medio de **MATLAB** demostraron una mínima desviación, demostrando que el software es capaz de proveer métricas con igual grado de precisión a un software matemático para el procesamiento de datos. El postprocesamiento efectuado por el sistema disminuye el tiempo necesario para el análisis de las señales del sistema y facilita el seguimiento de los practicantes de manera inmediata mediante el uso de parámetros de movimiento.

Finalmente, se describe que todos los objetivos planteados al inicio de este documento fueron alcanzados de manera satisfactoria, obteniendo como producto un software funcional que ayuda al manejo y seguimiento de los cirujanos según sus habilidades de movimiento durante la ejecución de las tareas preestablecidas.

## 6.2. Contribuciones

La tesis describe un sistema que integra librerías de visualización y de base de datos para el registro, estimación y seguimiento de la evolución de los parámetros de movimiento.

El software permite recopilar los datos de tareas anteriores para trazar en gráficas que muestran el avance o retroceso de los practicantes con respecto a diversos parámetros de movimiento evaluados durante la realización de las tareas. Este tipo de herramientas facilita la tarea del instructor al resumir los datos mediante un modelo visual más intuitivo.

El sistema es el primero en su tipo dentro del laboratorio del equipo de trabajo y propone soluciones para el seguimiento de médicos en práctica, lo que proyecta nuevos desarrollos y ajustes al sistema actual para mejorar la calificación de los estudiantes. El equipo, en combinación con el sistema, proveen un sistema integral de entrenamiento para los nuevos cirujanos laparoscopistas.



### 6.3. Trabajo futuro

A través de un análisis de las capacidades que posee el sistema ya descrito, existen limitaciones dentro de la funcionalidad del dispositivo. Algunas de las restricciones deben de ser trabajadas y mejoradas para proveer una mejor implementación del sistema en campo.

Como trabajo futuro se propone:

- Proveer una nueva de configuración y número de cámaras que pueda reducir el error de posicionamiento ocurrido en esta configuración cuando los puntos se alejaban del campo de vista de ambas cámaras.
- Utilizar marcadores con un diseño distinto a los segmentos cilíndricos; por ejemplo, diseños en forma de espiral o segmentos de diferentes proporciones que permitan obtener información del movimiento angular del instrumento sobre su propio eje.
- Implementar los algoritmos para imágenes de mayor resolución, a manera de mejorar la resolución del instrumento. De la misma forma, utilizar cámaras que manejen el protocolo **USB 3.0** para evitar que las imágenes afecten la velocidad de muestreo.
- Incluir nuevos parámetros de movimiento y añadir al sistema una función para realizar estudios comparativos entre usuarios y evaluar el desempeño y/o la diferencia entre expertos y novatos.
- Agregar una funcionalidad de autocalibración al sistema, a manera de que cualquier perturbación posicional en la o las cámaras pueda ser remediada inmediatamente.
- Extender el uso de *EndoEval* no sólo para sistema operativo *Windows*, sino también a otros como *Linux* o *MacOS*.
- Ampliar los alcances del sistema a otros departamentos médicos para evaluar otras habilidades quirúrgicas para cirujanos novatos.



# Anexo: Código fuente del programa de MATLAB para estimación de parámetros

```
1 function [parameters] = Params( matrizParams)
2 %PARAMETROS función para estimar parámetros de movimientos de EndoEval
3 %   MatrizParams - es el archivo de texto que contiene el registro de todas
4 %                   las posiciones y ángulos en cada tiempo de muestreo.
5 %   Los valores de retorno son:
6 %   - parameters: son los parámetros de movimiento estimados por la función
7 %
8
9 % length es el número de registros que tiene el archivo
10 length = size(matrizParams,1);
11
12 %posición del TROCAR DERECHO
13 Trocar_R = [-5.5,-9.85,-7];
14 %posición del TROCAR IZQUIERDO
15 Trocar_L = [15.5,-9.85,-7];
16
17 %se extraen la columna de tiempo
18 time_records = matrizParams(:,11);
19
20 % el tiempo transcurrido es el máximo tiempo reportado por la columna
21 % de registros de tiempo
22 TIME = max(time_records);
23
24 m_L = matrizParams(:,1:3);
25 m_R = matrizParams(:,4:6);
26
27 % inicializa las variables
```

```
28 distances_R = zeros(length-1,1);
29 distances_L = zeros(length-1,1);
30
31 % -> calculo de las distancias discretas entre puntos
32 for i = 2 : length;
33     distances_L(i-1) = pdist( [m_L(i-1,:) ; m_L(i,:)], 'euclidean');
34     distances_R(i-1) = pdist( [m_R(i-1,:) ; m_R(i,:)], 'euclidean');
35 end
36 % <-
37
38 % vectores de diferencias de tiempo que sirven para calcular las derivadas
39 time_inc = time_records(2:end)-time_records(1:end-1);
40
41 % se suma la distancia total recorrida entre los puntos y se obtiene
42 % la longitud del recorrido
43 PATH_LENGTH_R = sum(distances_R);
44 PATH_LENGTH_L = sum(distances_L);
45
46 % -> cálculo de ángulos
47 % inicializamos variables para calcular los ángulos
48 % son los componentes vectoriales del vector que va desde el trocar a
49 % la punta de la pinza para cada mano
50 a_PosR = zeros(length,3);
51 a_PosL = zeros(length,3);
52
53 % ángulos calculados para cada plano de proyección y para cada mano
54 aXY_R = zeros(length,1);
55 aXZ_R = zeros(length,1);
56
57 aXY_L = zeros(length,1);
58 aXZ_L = zeros(length,1);
59
60 % magnitud es la magnitud vector posicionado desde el trocar a la punta
61 magnitud_R = zeros(length,1);
62 magnitud_L = zeros(length,1);
63
64 % axial dist es la diferencia de distancia de dos vectores consecutivos
65 % en tiempo
66 axial_dist_R = zeros(length-1,1);
67 axial_dist_L = zeros(length-1,1);
68
69 for i = 1 : length
70     % se calculan los componentes del vector desde el punto de
71     % inserción del trocar hasta la posición de la punta de la pinza
```

```

72     a_PosR(i,:) = m_R(i,:) - Trocar_R;
73     a_PosL(i,:) = m_L(i,:) - Trocar_L;
74
75     % se obtienen los vectores de proyección para cada plano (xy y xz)
76     A_1_R = a_PosR(i,[1 2]);
77     A_1_L = a_PosL(i,[1 2]);
78     B_1_R = [1 0];
79     B_1_L = [-1 0];
80
81     A_2_R = a_PosR(i,[1 3]);
82     A_2_L = a_PosL(i,[1 3]);
83     B_2_R = [1 0];
84     B_2_L = [-1 0];
85
86     % calculamos el producto punto para cada vector de referencia
87     % y después su ángulo coseno correspondiente
88     aXY_R(i) = dot( A_1_R,B_1_R ) / ( norm(A_1_R)*norm(B_1_R) );
89     aXY_R(i) = acos( aXY_R(i) );
90
91     aXZ_R(i) = dot( A_2_R,B_2_R ) / ( norm(A_2_R)*norm(B_2_R) );
92     aXZ_R(i) = acos( aXZ_R(i) );
93
94     aXY_L(i) = dot( A_1_L,B_1_L ) / ( norm(A_1_L)*norm(B_1_L) );
95     aXY_L(i) = acos( aXY_L(i) );
96
97     aXZ_L(i) = dot( A_2_L,B_2_L ) / ( norm(A_2_L)*norm(B_2_L) );
98     aXZ_L(i) = acos( aXZ_L(i) );
99
100    % la magnitud nos ayudará a calcular la norma del vector
101    magnitud_R(i) = norm( a_PosR(i,:) );
102    magnitud_L(i) = norm( a_PosL(i,:) );
103
104    % para después medir la distancia axial entre vectores adyacentes
105    % en tiempo
106    if(i > 1)
107        axial_dist_R(i) = pdist( [magnitud_R(i); magnitud_R(i-1)] );
108        axial_dist_L(i) = pdist( [magnitud_L(i); magnitud_L(i-1)] );
109    end
110 end
111
112 % calculamos la percepción de profundidad
113 DEPTH_PERCEPTION_R = sum(axial_dist_R);
114 DEPTH_PERCEPTION_L = sum(axial_dist_L);
115

```

```
116 % -> CALCULO DEL SMOOTHNESS
117 % aquí empiezan las derivadas
118
119 % primer derivada
120 deriv1_L = distances_L ./ time_inc;
121 deriv1_R = distances_R ./ time_inc;
122
123 % velocidad promedio de cada mano
124 AVG_SPEED_R = mean(abs(deriv1_R));
125 AVG_SPEED_L = mean(abs(deriv1_L));
126
127 % segunda derivada
128 deriv2_L = diff(distances_L) ./ (time_inc(1:end-1) .* time_inc(2:end));
129 deriv2_R = diff(distances_R) ./ (time_inc(1:end-1) .* time_inc(2:end));
130
131 % tercera derivada
132 deriv3_L = diff(distances_L,2) ./ (time_inc(1:end-2) .* time_inc(2:end-1) .* time_inc(3:end));
133 deriv3_R = diff(distances_R,2) ./ (time_inc(1:end-2) .* time_inc(2:end-1) .* time_inc(3:end));
134
135 % ACELERACIÓN PROMEDIO
136 AVG_ACCEL_R = mean(abs(deriv2_R));
137 AVG_ACCEL_L = mean(abs(deriv2_L));
138
139 % SUAVIDAD SIN NORMALIZAR
140 J_R1 = sqrt( sum(deriv3_R .^ 2) / 2 );
141 J_L1 = sqrt( sum(deriv3_L .^ 2) / 2 );
142
143 % SUAVIDAD NORMALIZADA
144 MOTION_SMOOTHNESS_R = J_R1 / TIME;
145 MOTION_SMOOTHNESS_L = J_L1 / TIME;
146
147 % incrementos de ángulo para el plano XY
148 anglesdiff_xy_L = aXY_L(2: end) - aXY_L(1 : end- 1);
149 anglesdiff_xy_R = aXY_R(2: end) - aXY_R(1 : end- 1);
150 % incrementos de ángulo para el plano XZ
151 anglesdiff_xz_L = aXZ_L(2: end) - aXZ_L(1 : end- 1);
152 anglesdiff_xz_R = aXZ_R(2: end) - aXZ_R(1 : end- 1);
153
154 % distancia euclidiana para ambos planos
155 angdist_R = sqrt((anglesdiff_xy_R.^2) + (anglesdiff_xz_R.^2));
156 angdist_L = sqrt((anglesdiff_xy_L.^2) + (anglesdiff_xz_L.^2));
157
158 % LONGITUD ANGULAR
159 ANGULAR_LENGTH_L = sum(angdist_L);
```

```

160 ANGULAR_LENGTH_R = sum(angdist_R);
161
162 % ÁREA ANGULAR
163 ANGULAR_AREA_R = max(aXY_R) - min(aXY_R) * max(aXZ_R) - min(aXZ_R);
164 ANGULAR_AREA_L = max(aXY_L) - min(aXY_L) * max(aXZ_L) - min(aXZ_L);
165
166 % VOLUMEN
167 VOLUME_R = ANGULAR_AREA_R * ( max(magnitude_R) - min(magnitude_R));
168 VOLUME_L = ANGULAR_AREA_L * ( max(magnitude_L) - min(magnitude_L));
169
170 % se crea la matriz de retorno
171 parameters = zeros(9,2);
172
173 parameters(1,1) = TIME;
174 parameters(1,2) = TIME;
175
176 parameters(2,1) = PATH_LENGTH_L;
177 parameters(2,2) = PATH_LENGTH_R;
178
179 parameters(3,1) = DEPTH_PERCEPTION_L;
180 parameters(3,2) = DEPTH_PERCEPTION_R;
181
182 parameters(4,1) = MOTION_SMOOTHNESS_L;
183 parameters(4,2) = MOTION_SMOOTHNESS_R;
184
185 parameters(5,1) = ANGULAR_AREA_L;
186 parameters(5,2) = ANGULAR_AREA_R;
187
188 parameters(6,1) = VOLUME_L;
189 parameters(6,2) = VOLUME_R;
190
191 parameters(7,1) = AVG_SPEED_L;
192 parameters(7,2) = AVG_SPEED_R;
193
194 parameters(8,1) = AVG_ACCEL_L;
195 parameters(8,2) = AVG_ACCEL_R;
196
197 parameters(9,1) = ANGULAR_LENGTH_L;
198 parameters(9,2) = ANGULAR_LENGTH_R;
199
200 % y formateamos su impresión
201 fprintf(1, '\t \t \t Parametros de movimiento\n');
202 fprintf(1, 'Parametro          \t          L          \t          R          \t units\n');
203 fprintf(1, 'TIME                        \t % 15.8f \t % 15.8f \t (s.) \n',

```

```
204 parameters(1,1),parameters(1,2));
205 fprintf(1, 'PATH LENGTH      \t % 15.8f \t % 15.8f \t (cm.) \n',
206 parameters(2,1),parameters(2,2));
207 fprintf(1, 'DEPTH PERCEPTION \t % 15.8f \t % 15.8f \t (cm.) \n',
208 parameters(3,1),parameters(3,2));
209 fprintf(1, 'MOTION SMOOTHNESS \t % 15.8f \t % 15.8f \t (cm/s^3) \n',
210 parameters(4,1),parameters(4,2));
211 fprintf(1, 'ANGULAR AREA      \t % 15.8f \t % 15.8f \t (rads^2) \n',
212 parameters(5,1),parameters(5,2));
213 fprintf(1, 'VOLUME              \t % 15.8f \t % 15.8f \t (mm*rads^2) \n',
214 parameters(6,1),parameters(6,2));
215 fprintf(1, 'AVERAGE SPEED      \t % 15.8f \t % 15.8f \t (cm./s.) \n',
216 parameters(7,1),parameters(7,2));
217 fprintf(1, 'AVG. ACCELERATION \t % 15.8f \t % 15.8f \t (cm./s.^2) \n',
218 parameters(8,1),parameters(8,2));
219 fprintf(1, 'ANGULAR LENGTH     \t % 15.8f \t % 15.8f \t (rads.) \n',
220 parameters(9,1),parameters(9,2));
221
222 end
```



# Referencias

- [1] G. Allen and M. Owens. *The Definitive Guide to SQLite*. Apress, 2da. edition, 2010.
- [2] María E. Barilla-Pérez. *Colour-based Texture Image Segmentation*. PhD thesis, University of Birmingham, 2008.
- [3] K. U. Barthel. 3D color inspector/color histogram. Online, sept. 2007.
- [4] M. Berger. *Geometry I*. Springer, 1987.
- [5] R. Berguer, S. Gerber, G. Kilpatrick, and D. Beckley. An ergonomic comparison of in-line vs pistol-grip handle configuration in a laparoscopic grasper. *Surgical Endoscopy*, 12(6):805–808, junio 1998.
- [6] R. Berguer, M. Remler, and D. Beckley. Laparoscopic instruments cause increased forearm fatigue: A subjective and objective comparison of open and laparoscopic techniques. *Minimally Invasive Therapy & Allied Technologies*, 6(1):36–40, 1997.
- [7] G. Bradski and A. Kaehler. *Learning OpenCV*. O'Reilly, 1a. edition, 2008.
- [8] W. M. Brinkman, S. Y. Havermans, S. N. Buzink, S. M. B. I. Botden, J. J. Jakimowicz, and B. C. Schoot. Single versus multimodality training basic laparoscopic skills. *Surgical Endoscopy*, 26(8):2172–2178, agosto 2012.
- [9] S. Chin-Tan, N. Marlow, J. Field, W. Altree, M. Babidge, P. Hewett, and G. J. Maddern. A randomized crossover trial examining low- versus high-fidelity simulation in basic laparoscopic skills training. *Surgical Endoscopy*, 26(11):3207–3214, noviembre 2012.
- [10] M. Chmarra, W. Kolkman, F. Jansen, C. A. Grimbergen, and J. Dankelman. The influence of experience and camera holding on laparoscopic instrument movements measured with the trendo tracking system. *Surgical Endoscopy and Other Interventional Techniques*, 21:2069–2075, mayo 2007.

- 
- [11] M. K. Chmarra, N. H. Bakker, C. A. Grimbergen, and J. Dankelman. TrEndo, a device for tracking minimally invasive surgical instruments in training setups. *Sensors and Actuators A: Physical*, 126(2):328–334, febrero 2006.
- [12] M.K. Chmarra, S. Klein, J.C.F. De-Winter, F.W. Jansen, and J. Dankelman. Objective classification of residents based on their psychomotor laparoscopic skills. *Surgical Endoscopy*, 24(5):1031–1039, mayo 2010.
- [13] S. Cotin, N. Stylopoulos, M. Ottensmeyer, P. Neumann, D. Rattner, and S. Dawson. Metrics for laparoscopic skills trainers: The weakest link! *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2002*, 2488:35–43, 2002.
- [14] V. Datta, S. Mackay, M. Mandalia, and A. Darzi. The use of electromagnetic motion tracking analysis to objectively measure open surgical skill in the laboratory-based model. *American College of Surgeons*, 193(5):479–485, noviembre 2001.
- [15] E. R. Davies. *Computer & Machine Vision: Theory, Algorithms, Practicalities*. Academic Press, 4a edition, 2012.
- [16] F. de Medicina, E. Académico Profesional de Medicina Humana, and Dep. Académico de Cirugía. Cirugía I: cirugía general. Online, 1999.
- [17] Selecciones del Readers Digest. *Gran Diccionario Enciclopédico Ilustrado*. Readers Digest, 1er edition, 1986.
- [18] J. R. Eriksen and T. Grantcharov. Objective assessment of laparoscopic skills using a virtual reality stimulator. *Surgical Endoscopy And Other Interventional Techniques*, 19(1216), 2005.
- [19] D. A. Forsyth and J. Ponce. *Computer Vision, A Modern Approach*. Pearson, 2a edition, 2012.
- [20] A. Gallagher, Lederman. A., and K. McGlade. Discriminative validity of the minimally invasive surgical trainer in virtual reality (mist-vr) using criteria levels based on expert performance. *Surgical Endoscopy And Other Interventional Techniques*, 18(660), 2004.
- [21] M. García-Berro. El futuro de la cirugía mínimamente invasiva. tendencias tecnológicas a medio y largo plazo, noviembre 2004.

- [22] R. C. González and R. E. Woods. *Digital Image Processing*. Prentice-Hall, 3a edition, 2007.
- [23] F. Hanisch. Affine and perspective transformations. Online, 1997.
- [24] E. F. Hofstad, C. Våpenstad, M. Chmarra, Langø, E. Kuhry, and R. Mårvik. A study of psychomotor skills in minimally invasive surgery: what differentiates expert and nonexpert performance. *Surgical Endoscopy and Other Interventional Techniques*, 27:854–863, 2013.
- [25] J. M. Izquierdo-Rojo. *Lecciones de Neurocirugía*. Universidad de Oviedo, 1992.
- [26] A. Kamaev, A. Alekhin, A. Shishkov, A. Smorkalov, E. Talanin, G. Bradski, K. Korniyakov, V. Eruhimov, V. Pisarevsky, and A. Pavlenko. OpenCV: Open source computer vision library. Online, abril 2010.
- [27] A. Koshkan and M. Abidi. Detection and classification of edges in color images. *IEEE Signal Processing*, 22:64–73, enero 2005.
- [28] D. Lorias-Espinoza. *Sistema biomédico de entrenamiento y evaluación de las habilidades del cirujano laparoscopista*. PhD thesis, Bioelectrónica. Departamento de Ingeniería Eléctrica. CINVESTAV-IPN, noviembre 2007. Tutor: PhD Arturo Minor Martínez.
- [29] S. Mackay, P. Morgan, V. Datta, A. Chang, and A. Darzi. Practice distribution in procedural skills training. *Surgical Endoscopy And Other Interventional Techniques*, 16(6):957–961, 2002.
- [30] S. Martínez-Dubois. *Cirugía: Bases del conocimiento quirúrgico y apoyo en trauma*. McGraw-Hill, 4th edition, 2009.
- [31] MedlinePlus. Endoscopía. Online, noviembre 2012.
- [32] Microsoft. Chart types: Visual Studio 2012. Online, agosto 2012.
- [33] Microsoft. Color. Online, 2014.
- [34] A. Minor-Martínez, A. C. Kalach, and D. Lorias-Espinoza. Millimetric laparoscopic surgery training on a physical trainer using rats. *Surgical Endoscopy*, 22(1):249–249, enero 2008.

- 
- [35] A. Minor-Martínez and D. Lorias-Espinoza. Novel laparoscopic home trainer. *Surgical laparoscopy, endoscopy & percutaneous techniques*, 17(4):300–302, agosto 2007.
- [36] C. Moler. The growth of MATLAB and the MathWorks over two decades. *The MathWorks News&Notes*, enero 2006.
- [37] A. Najmaldin and P. Guillou. *A Guide to Laparoscopic Surgery*. Blackwell Science Ltd, 1998.
- [38] M. S. Nixon and A. S. Aguado. *Feature Extraction and Image Processing*. Newnes, 1a edition, 2002.
- [39] E. Nugent, N. Shirilla, A. Hafeez, D. S. O’Riordain, O. Traynor, A. M. Harrison, and P. Neary. Development and evaluation of a simulator-based laparoscopic training program for surgical novices. *Surgical Endoscopy*, julio 2012.
- [40] I. Oropesa, P. Sánchez-González, M. Chmarra, P. Lamata, A. Fernández, J. A. Sánchez-Margallo, F. Willem-Jansen, J. Dankelman, F. M. Sánchez-Margallo, and E. J. Gómez. EVA: Laparoscopic instrument tracking based on endoscopic video analysis for psychomotor skill assessment. *Surgical Endoscopy And Other Interventional Techniques*, oct. 2012.
- [41] I. Oropesa, P. Sánchez-González, P. Lamata, M. D. Chmarra, J. B. Pagador, J. A. Sánchez-Margallo, F. M. Sánchez-Margallo, and E. J. Gómez. Methods and tools for objective assessment of psychomotor skills in laparoscopic surgery. *Journal of Surgical Research*, 171:81–95, 2011.
- [42] S. Payandeh, A. J. Lomax, J. Dill, C. L. Mackenzie, and C. G. Cao. On defining metrics for assessing laparoscopic surgical skill in a virtual training environment. *Studies in health technology and informatics*, 334:334–340, 2002.
- [43] F. Pérez, H. Sossa, R. Martínez, D. Lorias, and A. Minor. Video-based tracking of laparoscopic instruments using an orthogonal webcams system. *International Journal of Medical, Health, Pharmaceutical and Biomedical Engineering*, 7(8):129 – 132, 2013.
- [44] M. Petrou. *Image Processing: The Fundamentals*. John Wiley & Sons, 1999.
- [45] Philips. Webcam pro for laptops SPC2050NC. Online, marzo 2010.

- [46] M. Ramos-Martín, A. Ramos-Martín, and F. Montero-Rodríguez. *Sistema Gestores de Bases de Datos*. McGraw-Hill, 2006.
- [47] S. F. Rivera-Ortega. Sistema de evaluación 3D del entrenamiento laparoscópico. Master's thesis, Bioelectronic. Electrical Engineering Department. CINVESTAV-IPN, diciembre 2008. Tutor: PhD Arturo Minor-Martínez.
- [48] J. Rosen, J. D. Brown, M. Barreca, L. Chang, B. Hannaford, and M. Sinanan. The Blue DRAGON—a system for monitoring the kinematics and the dynamics of endoscopic tools in minimally invasive surgery for objective laparoscopic skill assessment. *Studies in health technology and informatics*, 85:412–418, 2002.
- [49] A. Sepúlveda, C. Lizana, K. Semm, and K. Dubois. *Cirugía laparoscópica avanzada*, volume II. CIRUGIA LAPAROSCOPICA, 2007.
- [50] A. Silberschatz, H. F. Korth, and S. Sudarshan. *Fundamentos de Bases de Datos*. McGraw-Hill, 4a edition, 2002.
- [51] American Cancer Society. ¿Qué es endoscopía? Online, abril 2013.
- [52] E. M. Targarona. *Cirugía Endoscópica*. Arán Ediciones, 2004.
- [53] M. Tokunaga, M. Okajima, H. Egi, M. Yoshimitsu, D. Sumitani, T. Kawahara, M. Hattori, and H. Ohdan. The importance of stressing the use of laparoscopic instruments in the initial training for laparoscopic surgery using box trainers: A randomized control study. *Journal of Surgical Research*, pages 1–8, 2011.
- [54] C. M. Jr. Townsend, R. D. Beauchamp, B. M. Evers, and K. L. Mattox. *Sabiston. Tratado de cirugía: Fundamentos biológicos de la práctica quirúrgica moderna*. Elsevier, 2013.
- [55] A. Trémeau, S. Tominaga, and K. N. Plataniotis. Color in image and video processing: most recent trends and future research directions. *J. Image Video Process.*, 2008:1–26, 2008.
- [56] K. R. Van-Sickle, D. A. McClusky, and A. G. Gallagher. Construct validation of the promis simulator using a novel laparoscopic suturing task. *Surgical Endoscopy And Other Interventional Techniques*, 19(1227), 2005.
- [57] M. C. Vassiliou, G. A. Ghitulescu, L. S. Feldman, D. Stanbridge, K. Leffondré, H. H. Sigman, and G. Fried. The mistels program to measure technical skill in

---

laparoscopic surgery : evidence for reliability. *Surgical Endoscopy*, 20(5):744–747, mayo 2006.

- [58] A. Vesalius. *De humani corporis fabrica*. Culture et Civilisation, 1543.
- [59] G. A. Vilos, A. Ternamian, J. Dempster, and P. Y. Laberge. Laparoscopic entry: a review of techniques, technologies, and complications. *Journal of Obstetrics and Gynaecology Canada : JOGC*, 29(5):433–465, 2007.
- [60] M. S. Wilson, A. Middlebrook, and C. Sutton. MIST VR: a virtual reality trainer for laparoscopic surgery assesses performance. *Annals, The Royal College of Surgeons of England*, 79(6):403–404, noviembre 1997.
- [61] S. Yamaguchi, K. Konishi, and T. Yasunaga. Construct validity for eye-hand coordination skill on a virtual reality laparoscopic surgical simulator. *Surgical Endoscopy And Other Interventional Techniques*, 21(2253), 2007.