

# Conjunto de métricas para la validación de la segmentación de imágenes médicas.

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias  
Informáticas.

**Autor:**

Dalilis Lidiesky Cisnero  
Cabrera

**Tutores:**

Ing. Michel Alvarez Cancio  
MsC. Dailien Moré Soto

FACULTAD 3

LA HABANA. JUNIO, 2018.

## **Declaración de autoría**

Declaro ser el autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

\_\_\_\_\_

**Dalilis Lidiesky Cisnero Cabrera**

Autor

\_\_\_\_\_

**Ing. Michel Álvarez Cancio**

Tutor

\_\_\_\_\_

**MsC. Dailien Moré Soto**

Tutor

## **Dedicatoria**

*A mi Madre, Padre, Hermano y Abuelos*

## **Agradecimientos**

Quiero agradecer a todos aquellos que durante estos 5, perdón, 7 años me apoyaron y a lo que no también. Le agradezco a mi motor impulsor, la luz de mi vida, la mujer que ha dado todo por mí, la que me enseñó a no rendirme jamás, a luchar por mis sueños, me enseñó que cada tropiezo es una experiencia y que de ella se aprende. Todos se preguntarán ¿De quién hablo? Sí, mi madre **Misleydis Cabrera Alarcón**.

Hoy en día no está a mi lado, ya son muchos años que no la veo, no la abrazo, no le digo lo mucho que la extraño, la falta que me hace, y que mi sueño más anhelado es estar junto a ella. Pero bueno, en la vida no siempre se tiene lo que queremos, y tenemos que aprender a ser fuertes, aunque a veces te derrumbes y sientas que estás sola tienes que seguir adelante. Gracias a Dios, que tengo a mi segunda madre. Gracias por acogirme, por confiar, por enseñarme que si se puede, por ser un ejemplo a seguir. Por ayudarme en todo, por defenderme, cuidarme, apoyarme. Gracias **Dailien Moré Soto**.

Le agradezco a mi padre **Erundo Cisnero Castillo**, por cuidarme, aunque apenas he convivido con él, sé que hoy en día se siente muy orgulloso. Mis amistades más cercanas soben la historia entre mi papá y yo, y el miedo que le tengo. A la única persona que le tengo pánico es a él. Todo tiene una explicación, pero bueno sé que él me quiere mucho ¿cómo no va a quererme si soy su hija y la única?

A mi abuelita que dio su vida por mí, me malcrió mucho, donde quiera que esté yo la amo. A mi hermano, no es fácil tener un hermano chiquito que quiere todo para él, pero yo quería un hermanito, y desde que lo vi por primera vez lo he amado con todo mi corazón.

¡La vida es una caja de sorpresa y cómo da vuelta! ¡Quién lo diría! Pasó de no soportarme, de hablar mal de mí, de no querer que fuera novia de su mejor amigo, a ser más que amigas, hermana. Nos apoyamos mutuamente, nos queremos mucho, bueno déjame hablar por mí, yo la quiero mucho, aunque yo sea insoportable, sí porque son pocas las personas que me aguanta y ella es una de ellas. Me ha ayudado mucho, se ha comportado como la hermana mayor que no tengo, y quiero que sepas que estoy dispuesta a dar la vida por ti mi gran hermana **Arletis Francis Alitimón**.

A mi tutor **Michel Alvarez Cancio**, son muchas las palabras que le diría y lo agradecida que estoy con él, gracias por apostar por mí, por ser de las pocas personas que depositó su confianza en mí. Hoy en días más que mi tutor, eres mi amigo, más que mi amigo eres mi hermano y más que mi hermano te has comportado como un padre, sí, aunque solo me llevas dos años eres más viejo que yo.

Soy de pocas amigas, para no decir que las puedo contar con una sola mano y me sobran dedos. Sé que yo al principio no era de su agrado, ella tampoco del mío o mejor dicho me burlaba de ella, pero ¿De quién no se burla Dalilis y a quién no le da chucho? ya conocen mis características. Le decía antes: “*Bailando*” como la canción de Gente de Zona, pero ¿quién diría que por cosas del destino nos íbamos a convertir en buenas amigas? Gracias **Dayana de la Caridad Capestany** por apoyarme, por estar ahí cuando más lo he necesitado, por soportarme todas las malcriadeces, por confiar en mí.

A lo largo de mi vida he conocido personas muy importantes, especiales, personas que quiero mucho y una de ellas es **Leidanys Martinez Dias**, que me enseñó a ser una mejor persona, a crecer día a día. Sé que muchos me van a criticar y decir: “*Dalilis te pasaste*” pero bueno es mi tesis, son mis agradecimientos y sobre todo, son mis sentimientos.

Gracias a mi otra familia, sí, a mi hermano **Osiel** el venenoso, que a pesar de todo su veneno así mismo lo quiero mucho, a mi abuela **Rosalba**, a mis hermanos pequeños **Estefhan y Elias**, y a mi papá **Polanco** ¡Los quiero mucho!

Y por últimos al resto de mis amigos y amigas, por favor el que se me quede que levante la mano.

## Resumen

La opacidad de la cápsula posterior es actualmente uno de los aspectos más importantes en la cirugía de catarata de los tiempos modernos. Sigue siendo la complicación posoperatoria tardía más frecuente tras la cirugía de catarata asociada con disminución de la agudeza visual, deterioro de la sensibilidad al contraste y problemas de deslumbramiento que conllevan a importantes repercusiones sociales, médicas y económicas. En nuestra Universidad se han desarrollado varios algoritmos para cuantificar la Opacidad de la Cápsula Posterior en imágenes del PENTACAM; ellos han logrado diversos resultados, aunque no se ha concretado cuál identifica mejor este tipo de complicación postoperatoria. En el 2017 se realizó una investigación para lograr el producto de software que permitiera la validación de la segmentación de los tomogramas Scheimpflug del PENTACAM. Pero no se logra implementar algunas técnicas que continúan siendo de interés en el ámbito del procesamiento digital de imágenes y en el grupo de investigación AIRI en lo particular y realizar un análisis estadístico utilizando el software SPSS. Se implementaron las medidas estadísticas de error (Clasificación del Error, Índice Rand Ajustado, Índice de Jaccard, Índice de Czekanowski, Índice de Sokal-Sneath, Índice de Rogers-Tanimoto, Índice de Yule, Índice de Russell-Rao y Índice de Fowlkes-Mallows), medidas de localización (Desajuste de bordes y Tasa de Correctos Equilibrada) y métricas de Hausdorff (Modificación de la Distancia de Hausdorff Normalizada), logrando así una evaluación objetiva del grado de exactitud de la segmentación. Con el agregado de estas nuevas métricas sobre los métodos existentes se pudo constatar la eficacia del método propuesto por Paula Vidal en el 2016.

**Palabras claves:** Validación de la segmentación, medidas estadísticas de error, medidas de localización, métricas de Hausdorff, opacidad de la cápsula posterior.

## ÍNDICE DE CONTENIDO

DEDICATORIA .....	II
AGRADECIMIENTOS.....	III
RESUMEN.....	V
ÍNDICES DE FIGURA.....	8
INTRODUCCIÓN.....	8
CAPÍTULO 1.VALIDACIÓN DE LA SEGMENTACIÓN DE IMÁGENES MÉDICAS: BASES CONCEPTUALES.....	14
1.1 PROCESAMIENTO DIGITAL DE IMÁGENES .....	14
1.1.1 <i>Imagen Digital</i> .....	15
1.1.2 <i>Imágenes médicas</i> .....	15
1.2 SEGMENTACIÓN .....	16
1.2.1 <i>Transformada de Hough</i> .....	17
1.2.2 <i>Detección de bordes</i> .....	17
1.2.3 <i>Operador Sobel</i> .....	18
1.2.4 <i>Operador Canny</i> .....	19
1.2.6 <i>Contornos Activos:</i> .....	21
1.3 VALIDACIÓN DE LA SEGMENTACIÓN.....	22
1.3.1 <i>Medidas del error</i> .....	23
1.3.2. <i>Medidas estadísticas de error.</i> .....	24
1.3.3 <i>Métricas de validación en la segmentación de imágenes</i> .....	24
1.3.4 <i>Clasificación del Error (CE)</i> .....	25
1.3.5 <i>Desajuste de Bordes (EMM)</i> .....	25
1.3.6 <i>Índice Rand Ajustado (ARI)</i> .....	26
1.3.7 <i>Índice de Jaccard (JI)</i> .....	26
1.3.8 <i>Índice de Czekanowski ( Czl)</i> .....	27
1.3.9 <i>Tasa de Correctos Equilibrada (BCR)</i> .....	27
1.3.10 <i>Índice de Sokal-Sneath (SSI)</i> .....	27
1.3.11 <i>Índice de Rogers-Tanimoto (RTI)</i> .....	27
1.3.12 <i>Índice de Yule (YI)</i> .....	27
1.3.13 <i>Índice de Russell-Rao (RRI)</i> .....	28
1.3.14 <i>Índice de Fowlkes-Mallows (FMI)</i> .....	28

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

1.3.14 Modificación de la Distancia de Hausdorff Normalizada (NMHD).....	29
1.5 HERRAMIENTAS USADAS DURANTE EL DESARROLLO .....	30
1.5.1 Metodología de desarrollo de Software.....	30
1.5.3 Entorno integrado de desarrollo.....	33
1.5.4 Sistema Gestor de Base de Datos. ....	34
1.5.6 HERRAMIENTA PARA EL ANÁLISIS ESTADÍSTICO UTILIZANDO SPSS.....	35
1.7 CONCLUSIONES DEL CAPÍTULO.....	36
<b>CAPÍTULO 2. MÉTRICAS BASADAS EN UN GROUND TRUTH PARA LA VALIDACIÓN DE SEGMENTACIÓN DE IMÁGENES MÉDICAS. ....</b>	<b>37</b>
2.1 PROGRAMA ANALIZADOR DE OPACIDAD CAPSULAR.....	37
2.2.1 Interfaz visual.....	37
2.2 PROGRAMA ANALIZADOR DE LA SEGMENTACIÓN DE IMÁGENES .....	39
2.3 DESCRIPCIÓN DE LA SOLUCIÓN .....	43
2.4 FASE DE PLANIFICACIÓN .....	45
2.4.1 Historias de Usuario.....	45
2.4.2 Estimación de esfuerzos por Historias de Usuario .....	46
Índice de Russell-Rao.....	46
2.4.3 Requisitos no funcionales .....	47
2.4.4 Plan de iteraciones .....	47
2.2.5 Plan de entrega .....	49
2.5 DISEÑO DE LA PROPUESTA DE SOLUCIÓN .....	49
2.5.1 Tarjetas CRC.....	49
2.5.2 Patrones de diseño utilizados .....	50
2.6 CONCLUSIONES DEL CAPÍTULO.....	52
<b>CAPÍTULO 3. RESULTADOS Y VALIDACIÓN DEL SISTEMA. ....</b>	<b>53</b>
3.1 ANÁLISIS ESTADÍSTICO DE LOS RESULTADOS EN IMÁGENES DEL PENTACAM .....	53
PRUEBAS PARAMÉTRICAS.....	<b>¡ERROR! MARCADOR NO DEFINIDO.</b>
PRUEBAS NO PARAMÉTRICAS .....	54
TEST DE SUMA DE RANGOS DE WILCOXON .....	54
TEST DE FRIEDMAN .....	55
3.2 PRUEBAS.....	55
3.2.1 PRUEBAS UNITARIAS.....	56
3.2.2 PRUEBAS ACEPTACIÓN .....	59
3.2.3 ANÁLISIS DE LOS RESULTADOS EN LAS IMÁGENES DEL PENTACAM .....	59

3.2.4 ANÁLISIS ESTADÍSTICO .....	69
3.3 CONCLUSIONES PARCIALES.....	72
CONCLUSIONES GENERALES .....	73
RECOMENDACIONES.....	74
REFERENCIAS BIBLIOGRÁFICAS.....	75
ANEXOS.....	84
GLOSARIO DE TÉRMINOS .....	91

## **Índices de Figura**

<b>Figura 1:</b> <i>Imagen Digital</i> .....	15
<b>Figura 2:</b> <i>Imágenes Médicas</i> .....	16
<b>Figura 3:</b> <i>Máscaras binomiales de 3x3, para las direcciones en función de X y Y del gradiente</i> .....	19
<b>Figura 4:</b> <i>Ejemplo de condiciones de comparación de segmentación (a la izquierda la imagen segmentada y a la derecha el Ground Truth).</i> .....	23
<b>Figura 5:</b> <i>Ejemplo de variables de evaluación.</i> .....	24
<b>Figura 6:</b> <i>Interfaz visual de PANDOC</i> .....	38
<b>Figura 7:</b> <i>Interface principal del software PASI</i> .....	39
<b>Figura 8:</b> <i>Ventana de visualización de los resultados de las métricas.</i> .....	41
<b>Figura 9:</b> <i>Imágenes de graficación por métricas</i> .....	42
<b>Figura 10:</b> <i>Diagrama de flujo del sistema.</i> .....	43
<b>Figura 11:</b> <i>Clase Método.</i> .....	51
<b>Figura 12:</b> <i>Clase LabellImage.</i> .....	51
<b>Figura 13:</b> <i>Creación de instancias.</i> .....	52
<b>Figura 14:</b> <i>Algoritmo para repeticiones</i> .....	56
<b>Figura 15:</b> <i>Diagrama de flujo del algoritmo “repeticionesIntensidad”.</i> .....	57
<b>Figura 16:</b> <i>Imágenes del Pentacam segmentadas por los algoritmos propuestos (b,c,d) y la imagen de referencia marcada por los especialistas (a).</i> .....	60
<b>Figura 17:</b> <i>Resultados de la métrica Desajuste de Bordes en los 3 algoritmos.</i> .....	62
<b>Figura 18:</b> <i>Resultados de la métrica Índice Rand Ajustado en los 3 algoritmos</i> .....	62
<b>Figura 19:</b> <i>Resultados de la métrica Índice Jaccard en los 3 algoritmos</i> .....	63
<b>Figura 20:</b> <i>Resultados de la métrica Índice de Czekanowski en los 3 algoritmos.</i> .....	63

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

<b>Figura 21:</b> Resultados de la métrica Tasa de Correctos Equilibrada en los 3 algoritmos. .....	64
<b>Figura 22:</b> Resultados de la métrica del Índice de Sokal-Sneath en los 3 algoritmos.	65
<b>Figura 23:</b> Resultados de la métrica del Índice de Índice de Rogers-Tanimoto en los 3 algoritmos. ....	65
<b>Figura 24:</b> Resultados de la métrica del Índice de Yule en los 3 algoritmos. ....	66
<b>Figura 25:</b> Resultados de la métrica del Índice de Índice de Russell-Rao en los 3 algoritmos. ....	66
<b>Figura 26:</b> Resultados de la métrica del Índice de Fowlkes-Mallows en los 3 algoritmos. .....	67
<b>Figura 27:</b> Resultados de la métrica Modificación de la Distancia de Hausdorff Normalizada en los 3 algoritmos. ....	68
<b>Figura 28:</b> Resultados de la métrica Clasificación del Error en los 3 algoritmos. ....	68

## **INTRODUCCIÓN**

La catarata, un padecimiento que provoca la pérdida total o parcial de la visión y que afecta a miles de personas en todo el mundo [1] [2], no es más que opacidad en el cristalino del ojo, trayendo como consecuencia que la luz se disperse dentro del ojo y no sea posible enfocar en la retina. Esto provoca que el paciente vea imágenes difusas al no poder distinguir cambios de colores y contrastes [3]. La catarata es la causa más común de ceguera tratable con cirugía. Tiene diversas causas, pero se le atribuye mayormente a la edad, acelerando este proceso si el paciente padece de enfermedades como la diabetes o hipertensión. Existen otras enfermedades o condiciones que también propician la aparición de cataratas: la inflamación de ojos, tener antecedentes familiares, el uso prolongado de otros medicamentos, frecuente exposición a la radiación, el tabaquismo, haber sido intervenido quirúrgicamente por otro problema ocular, etc [3].

Tras una intervención quirúrgica el paciente puede recuperar su visibilidad total o parcialmente, pero no en todos los casos la cirugía es un éxito a largo plazo, pues en muchos de ellos el paciente puede presentar complicaciones posoperatorias. La complicación posoperatoria más frecuente a largo o mediano plazo es la Opacidad de la Cápsula posterior (OCP) [4].

La cápsula posterior es una estructura que anatómicamente queda por detrás de la lente intraocular implantada en el paciente operado de catarata. Para garantizar la recuperación visual del paciente es imprescindible la transparencia de esta estructura. Su incidencia en la actualidad se encuentra entre 0,7% y 47,6 % en los primeros cinco años de realizada la cirugía, según un análisis realizado en [5] de 90 estudios publicados, mientras que en Cuba ha llegado a aparecer hasta en el 50% de los casos [5] . En la comunidad médica internacional no existe consenso alguno acerca de la correcta cuantificación de la OCP, se deja a cada especialista la elección de alguna de las técnicas propuestas [6].

Para el diagnóstico de la Opacidad de la Cápsula posterior (OCP) uno de los equipos utilizados es el PENTACAM. Este es un instrumento oftalmológico de alta tecnología desarrollado por la compañía *OCULUS*<sup>1</sup>, capaz de reconstruir imágenes de alta resolución del polo anterior del ojo. Las imágenes son reconstruidas a partir de múltiples

---

<sup>1</sup> Compañía alemana del campo de la óptica y especializada en oftalmología. Promotora de eventos de oftalmología y con gran cantidad de investigadores y desarrolladores de nuevos equipos médicos sobre esta rama. Algunas de sus líneas de productos son: C-Quant · EasyField CenterField 2 · Twinfield 2 · EasyGraph · Keratograph · ImageCam 2 · Pachycam · Oculus PENTACAM, entre muchas otras.

fotografías tomadas mediante una cámara rotacional del sistema Scheimpflug con que cuenta el equipo. Este sistema Scheimpflug es capaz de capturar imágenes en múltiples meridianos (hasta 50 imágenes en 2 segundos) en un único escaneo automatizado. El análisis de las imágenes resultantes es de gran utilidad para la cuantificación objetiva de la OCP. Una ventaja notable que poseen las imágenes provenientes del PENTACAM es la no presencia de destellos de luz que interfieran con el análisis de las mismas. [7]

Las imágenes provenientes del PENTACAM pueden poseer regiones de OCP tan pequeñas que se hace difícil su detección a simple vista, lo que puede traer consigo un análisis incorrecto por parte del especialista. En la actualidad existen diversos software que identifican de forma automática la OCP, destacándose entre ellos el PANDOC, el cual fue desarrollado en el Grupo de Investigación AIRI (Artificial Intelligence: Research and Innovation) de la Universidad de las Ciencias Informáticas (UCI) en colaboración con el Instituto Cubano de Oftalmología (ICO) "Ramón Pando Ferrer".

PANDOC es un software para identificar regiones con OCP, mediante el uso de las imágenes provenientes del PENTACAM, y evaluar su correlación con la graduación subjetiva a través de la lámpara de hendidura. Este software provee al oftalmólogo de una herramienta capaz de detectar diferencias de opacidad, a veces imperceptibles para el ojo humano [8].

Para la identificación automática de la opacidad por el software PANDOC, en el año 2015 se realizó una propuesta fundamentada por un sistema basado en casos (SBC) [9] [10]. En esta propuesta cada caso a identificar era un pixel, donde se analizaban sus características de color y las de sus vecinos, así como también su cercanía con respecto al centro de la estructura ocular.

También se realizó una variante en el 2016 que utilizaba un SBC, pero en esta ocasión con el objetivo principal de minimizar tiempo de ejecución con respecto al SBC anterior. En esta versión se usan como casos, regiones que fueron previamente extraídas usando un algoritmo de componentes conexas [11]. Entre los rasgos de estas regiones seguía estando el de cercanía al centro de la estructura ocular debido a la importancia de este rasgo para la decisión del SBC [12].

En el propio 2016 se propone otra variante utilizando mayormente técnicas de procesamiento digital de imágenes, en esta ocasión buscando los bordes de la capsulorrexis [13], esto bajo la premisa de que toda variación de la imagen de fondo en la región comprendida por la capsulorrexis es OCP. Esta solución está basada en una

combinación de filtros espaciales, la transformada circular de Hough y un algoritmo de contornos activos [13] [14].

Estos algoritmos desarrollados han tenido como objetivo cuantificar OCP en tomogramas Scheimpflug del PENTACAM logrando diversos resultados, principalmente en cuanto a calidad de la segmentación y tiempo de ejecución. Pero, hasta el momento no se ha concretado cuál identifica mejor este tipo de complicación postoperatoria.

En el 2017 se realizó una investigación para lograr el producto de software que permitiera la validación de la segmentación de los tomogramas Scheimpflug del PENTACAM [15]. Este implementaba un conjunto de algoritmos que permitían obtener disímiles variables de validación de segmentación utilizando un Ground Truth. Pero no se logra implementar algunas técnicas que continúan siendo de interés en el ámbito del procesamiento digital de imágenes y en el grupo de investigación AIRI en lo particular.

Dada la situación problemática expuesta anteriormente, se plantea como **Problema a resolver** ¿Cómo validar la segmentación en imágenes del PENTACAM partiendo de un Ground Truth? Se toma como **Objeto de estudio** Procesamiento Digital de Imágenes, enmarcándose en el **Campo de acción:** Validación de la Segmentación de Imágenes Digitales. En aras de dar solución al problema planteado se tiene como **Objetivo general:** Implementar un conjunto de métricas al software PASI para la validación de los algoritmos de segmentación en imágenes de tomogramas Scheimpflug del PENTACAM.

Dado el objetivo general expuesto se describen los siguientes objetivos específicos.

**Objetivos específicos:**

- Elaborar el marco teórico de la investigación mediante el estudio de las técnicas actuales de segmentación digital de imágenes.
- Identificar los requisitos funcionales y no funcionales para el desarrollo de la propuesta de solución.
- Diseñar un módulo para la validación de la segmentación en imágenes del PENTACAM.
- Implementar los elementos definidos en la etapa del diseño.
- Validar los resultados de los algoritmos de segmentación mediante pruebas estadísticas, y validar la implementación con pruebas unitarias y de aceptación.

Teniendo en cuenta el problema a resolver se formuló la siguiente **Idea a Defender:**

## *Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

Con el desarrollo de métricas para validar la segmentación de los algoritmos de detección de opacidad capsular desarrollados en el grupo de investigación AIRI, se mejorarán los resultados de la herramienta PASI en el momento de seleccionar el algoritmo apropiado para medir la incidencia de la opacidad de la cápsula posterior OCP en pacientes sometidos a cirugía de catarata.

Para dar cumplimiento al objetivo de la investigación se trazan las siguientes tareas a cumplir:

### **Tareas a cumplir:**

- Identificación de las diferentes técnicas de procesamiento de imágenes médicas.
- Revisión de las técnicas de banalización en imágenes médicas.
- Revisión de las técnicas de validación de la segmentación en imágenes médicas.
- Estudio de medidas estadísticas para la validación de la segmentación.
- Descripción de los pasos a seguir para el uso de un sistema propuesto.
- Estudio de una metodología para el desarrollo de aplicaciones de este tipo.
- Diseñar la solución propuesta en función de los requisitos especificados.
- Implementación de las técnicas de validación de la segmentación seleccionadas.
- Implementación de las medidas estadísticas para el sistema de validación de la segmentación.
- Implementación del mecanismo de selección y banalización seleccionado.
- Integración al software PASI.
- Validación de los resultados a partir de pruebas estadísticas.
- Validación de la implementación del sistema a partir de la aplicación de pruebas unitarias y funcionales.

Durante el desarrollo de la investigación es necesario utilizar varios métodos científicos, los cuales se mencionan a continuación.

**Experimento:** Permite probar las distintas técnicas y algoritmos para la validación de la segmentación en imágenes médicas.

**Análisis-Síntesis:** Posibilitó la búsqueda y análisis de los algoritmos, teorías y documentos relacionados con el tema y a partir de esto se seleccionaron los elementos claves para el desarrollo de este trabajo.

**Histórico-Lógico:** Contribuyó a la realización de un estudio con el fin de conocer la evolución y desarrollo de las técnicas existentes en la actualidad enfocadas a la validación de la segmentación en imágenes médicas.

## **CAPÍTULO 1. VALIDACIÓN DE LA SEGMENTACIÓN DE IMÁGENES MÉDICAS: BASES CONCEPTUALES.**

En el siguiente capítulo se realizará una disquisición desde el punto de vista teórico del problema general en que se enmarca la investigación. Se abordan los principales conceptos asociados al dominio del problema, como la incidencia de la OCP. También se realiza un estado del arte de las técnicas más usadas en la validación de la segmentación de imágenes médicas, las que ayudarán en la identificación de diferencias en la segmentación automática de las mismas.

### **1.1 Procesamiento digital de imágenes**

El procesamiento digital de imágenes (PDI) es el conjunto de técnicas y procesos para descubrir, resaltar y describir la información contenida en una imagen, usando como herramienta principal un sistema de cómputo, este tiene como objetivo el mejoramiento de la información gráfica para la interpretación humana y el procesamiento de los datos de la escena para la percepción automática por computadora [16]. El PDI consta de 5 etapas las cuales se definen a continuación [17]:

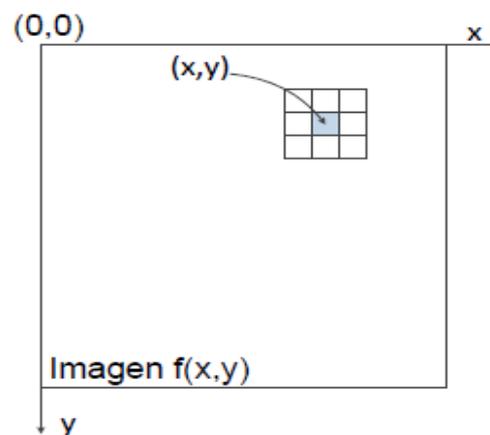
1. **Adquisición de la imagen:** Es el proceso de obtención de una imagen.
2. **Pre-procesamiento:** Incluye técnicas tales como la reducción de ruido y realce de detalles. Se realizan para que el resultado sea más apropiado para una aplicación en particular. Dichas técnicas consisten en la mejora de la nitidez o aclarado de las imágenes desenfocadas, eliminación de ruido, mejora del contraste, mejora del ruido y detención de bordes.
3. **Segmentación de la imagen:** Se realiza con el objetivo de aislar ciertos objetos en una imagen. Dentro de las tareas habituales que se realizan se encuentran Búsqueda y selección de formas determinadas y máscaras de la imagen.
4. **Representación y descripción:** Mediante los procesos de descripción se obtienen características (tamaño, perímetro, etc.) convenientes para diferenciar un objeto de otro.
5. **Reconocimiento e interpretación:** El reconocimiento es el proceso que identifica los objetos y la interpretación le asocia un significado a un conjunto de objetos reconocidos. Generalmente es conveniente agrupar estas áreas de acuerdo con la complicación y el grado de detalles que lleva aparejada su implementación.

Es importante destacar que no existe un método común que logre realizar correctamente el procesamiento de una imagen digital, por lo que se tiene que analizar según las características propias de la imagen en cuestión, cuál de los algoritmos existentes en

cada una de las etapas planteadas responde mejor a nuestros intereses. De todas ellas la segmentación de la imagen digital es la más importante debido a que de ella se obtendrán los objetivos deseados para su posterior análisis, por lo que una correcta selección de los mismos se hace imprescindible. La presente investigación se centra más en la etapa de la segmentación que en el resto por la importancia que conlleva la misma, aunque todas son respetadas como parte del proceso del tratamiento de una imagen digital.

### **1.1.1 Imagen Digital**

Una imagen digital es un arreglo bidimensional de píxeles, donde el valor de cada píxel se representa mediante una función  $f(x,y)$  que representa el nivel de brillantez, color o intensidad de la imagen en tales coordenadas. Por lo tanto, una imagen en blanco y negro puede ser representada por una matriz de dimensión  $M \times N$ , donde  $f(x,y) \in \{0,1\}$ . Además de la representación en blanco y negro, otras opciones son las imágenes en tonos de gris, para las cuales  $f(x,y)$  representa un nivel de intensidad típicamente asociado a un entero entre 0 y 255 y otras más sirven para describir imágenes en color [18].

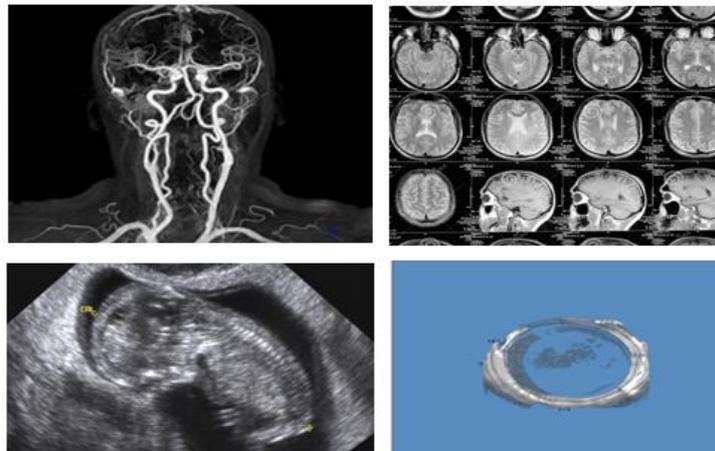


**Figura 1:** Imagen Digital.

### **1.1.2 Imágenes médicas**

El desarrollo y avance de las tecnologías de obtención de imágenes médicas está revolucionando la medicina. Estas imágenes juegan un papel importante en el diagnóstico y tratamiento de muchas enfermedades, debido a que permiten la obtención de información vital del interior del cuerpo humano de forma no invasiva. En consecuencia, la comunidad de analistas de imágenes se ha sumergido en la búsqueda de la solución del problema de extracción de información clínica útil sobre estructuras

anatómicas a partir de imágenes de Tomografía Computarizada (CT – Computer Tomography), Resonancia Magnética (MR – Magnetic Resonance), Ultrasonido y otras modalidades con ayuda de la computadora [19] [20]. Los algoritmos dedicados a la extracción de información a partir de imágenes, son conocidos como “algoritmos de segmentación de imágenes”, y juegan un papel significativo en numerosas aplicaciones biomédicas con tratamiento de imágenes médicas anatómicas ya que ayudan a la toma de decisiones y mejoran la calidad del diagnóstico médico [19].



**Figura 2:** *Imágenes Médicas*

## 1.2 Segmentación

La segmentación en el campo del procesamiento digital de imágenes es el proceso de dividir una imagen digital en varias partes (grupos de píxeles) u objetos, y su objetivo es simplificar y cambiar la representación de una imagen en otra más significativa y más fácil de analizar [21]. En el caso de las imágenes médicas las técnicas de segmentación se utilizan para realizar la identificación de estructuras anatómicas presentes en una imagen, las cuales permiten particionar la imagen en un conjunto no solapado de regiones, cuya unión es la imagen completa [22]

De manera general, los métodos de segmentación se clasifican de la siguiente forma [23]:

- **Métodos basados en la comparación de umbrales en el histograma**, en los cuales se obtiene un umbral de comparación para el agrupamiento de los píxeles.
- **Métodos basados en la detección de discontinuidades**, en donde se calculan las derivadas de la función de la imagen, éstos dividen la imagen a partir de cambios bruscos de los niveles de grises.

- **Métodos basados en la propiedad de similitud de los valores de los niveles de grises**, que permiten la agrupación de puntos a partir de ciertos criterios de homogeneidad.
- **Métodos heurísticos de segmentación**, los cuales se basan en el conocimiento previo de la imagen a segmentar y en la experiencia del observador, ellos incluyen en muchas ocasiones los métodos supervisados de segmentación.

### 1.2.1 Transformada de Hough

La transformada de Hough es un algoritmo usado en reconocimiento de patrones que permite aislar características de forma particular dentro de una imagen. La idea principal es encontrar curvas que puedan ser parametrizadas como líneas rectas, polinomios y círculos, es decir, que si se tiene un segmento de línea, esta se representaría con los parámetros  $p$  y  $\theta$ , donde  $p$  es la distancia entre la línea y el origen, y  $\theta$  es el ángulo del vector desde el origen al punto más cercano de la línea, la ecuación de la recta se podría escribir de la siguiente forma [24]:

$$p = x \cos \theta + y \sin \theta \quad (1)$$

Una de las variantes de la Transformada de Hough es la circular, sirve para la detención de círculos los cuales pueden ser descritos mediante su centro y radio. Para definir la transformada de Hough para el círculo, lo primero que se debe hacer es expresar el círculo mediante la siguiente ecuación:

$$(x - a)^2 + (y - a)^2 = r^2 \quad (1.1)$$

- Donde (a,b) es el centro del círculo y r su radio. Para encontrar circunferencias usando la transformada de Hough, se necesita un acumulador con tres dimensiones.
- Después cada punto en la imagen vota por las circunferencias en los que pudiera estar. Una vez terminado este procedimiento se buscan los picos en el acumulador y con esto se obtienen el radio y el centro de la circunferencia. Si se conociera el radio de antemano, solo se necesitaría un acumulador de dos dimensiones [25].

### 1.2.2 Detección de bordes

La detección de bordes es una parte esencial del procesamiento digital de imágenes. Su función es identificar los límites de las regiones homogéneas en una imagen

basándose en propiedades como la intensidad y la textura. Muchos algoritmos de detección de bordes han sido desarrollados sobre la base del cálculo de la intensidad del vector gradiente el cual en general tiene en cuenta al ruido de la imagen [26] .

Esta técnica de procesamiento de imágenes se utiliza para la búsqueda de los límites de los objetos dentro de la misma. Su funcionamiento se basa en la detección de discontinuidades en los niveles de intensidad o brillo. La detección de bordes se utiliza para la segmentación de imágenes y extracción de datos en áreas como procesamiento de imágenes, visión por computadora, y la visión de máquina [27].

Entre los problemas que presenta la detección de bordes al igual que otras técnicas en el campo del procesamiento digital de imágenes se encuentra el ruido, el cual forma parte de la información no deseada en la imagen en sí, pudiendo tener su existencia diversos orígenes.

Un borde, como parte de una imagen, es considerado como un tipo de discontinuidad, por lo que puede ser detectado usando derivadas de primer y segundo orden. La derivada de primer orden para una imagen es el gradiente, mientras que la segunda derivada tiene una respuesta más fuerte ante detalles como puntos aislados o líneas [28] .

### **1.2.3 Operador Sobel**

Este operador es capaz de detectar los bordes horizontales y verticales de forma separada sobre una imagen en escala de grises. El operador Sobel calcula el gradiente de la intensidad de cada punto en una imagen, dando para cada punto la magnitud del mayor cambio posible, la dirección de éste y el sentido desde los tonos oscuro a claro [29].

El resultado de este operador muestra cómo cambia una imagen de forma abrupta o suave en el punto analizado, y en consecuencia, cuán probable es que este represente un borde en la imagen, así como la orientación a la que tiende ese borde [30] [31].

Se trata de una técnica basada en el gradiente para realizar la detección de bordes, de manera que las variaciones de intensidad en los píxeles de la imagen prevalezcan frente a las zonas de intensidad constante. En función de cual sea la dirección en la que se aplica el gradiente son las máscaras que realizan la detección de bordes (ver Figura 3) [32].

$$G_X = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad G_Y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

**Figura 3:** Máscaras binomiales de 3x3, para las direcciones en función de X y Y del gradiente.

#### **1.2.4 Operador Canny**

El algoritmo de Canny es considerado como uno de los métodos más eficientes para la detección de bordes mediante el empleo de máscaras de convolución, ya que es menos sensible al ruido y es capaz de detectar bordes débiles [33].

En este método se utiliza la primera derivada para la detección de contornos, debido a que toma el valor cero en las regiones donde la intensidad se mantiene invariante y tiene un valor constante en toda la transición de intensidad [34] [35].

Canny utiliza el cálculo de variaciones, que no es más que una técnica que se encarga de encontrar la función que optimiza a su vez a una función dada.

El algoritmo de Canny se divide en diferentes etapas [36]:

1. Suavizado de la imagen.
2. Encontrar gradientes.
3. Suprimir los puntos que no sean máximos.
4. Umbral doble.
5. Seguimiento de bordes por histéresis.

**1-Suavizado de la imagen:** El suavizado de la imagen en el algoritmo de Canny forma parte del preprocesamiento de la imagen. Este operador utiliza un filtro basado en el concepto de la primera derivada gaussiana convolucionando la función con filtros gaussianos unidimensionales. De esta forma se suaviza el ruido presente en la imagen.

**2-Encontrar el gradiente:** Determinando el gradiente de la imagen se puede localizar dentro de la imagen los puntos de mayor intensidad. El borde de una imagen apunta a varias direcciones por lo que este algoritmo trabaja con cuatro filtros para detectar bordes horizontales, verticales y diagonales. Los operadores de detección de bordes como *Roberts*, *Prewitt* y *Sobel* devuelven un valor para la primera derivada en la dirección horizontal y vertical [37] [27] [38].

Las magnitudes del gradiente, las cuales representan los puntos fuertes de los bordes, se pueden determinar mediante la distancia Euclideana y aplicando el teorema de Pitágoras

$$|G| = \sqrt{G_x^2 + G_y^2} \quad (2)$$

donde  $G_x$  y  $G_y$  son los gradientes en las direcciones  $x$  e  $y$  respectivamente.

Para la localización de los bordes se utiliza la ecuación siguiente:

$$\theta = \arctan\left(\frac{|G_y|}{|G_x|}\right) \quad (3)$$

**3-Suprimir los puntos que no sean máximos:** Para suprimir los puntos no máximos se realiza un umbralado, eliminando aquellos píxeles que no tienen una magnitud (módulo del gradiente) alta, marcando los píxeles que resultan ser máximos con un recuadro de borde blanco y obviando el resto.

**4-Umbrales doble:** Al terminar la etapa de supresión los bordes que superen el umbral alto son clasificados como fuertes, los que estén por debajo del umbral bajo son eliminados, mientras que los bordes intermedios se clasifican como débiles. La umbralización con histéresis requiere de dos umbrales, alto y bajo. La suposición de que los bordes deben ser curvas continuas en la imagen nos permite poder descartar algunos píxeles que no forman parte de una línea. Comenzar la umbralización mediante la aplicación de un umbral alto nos permitirá marcar los bordes reales en la imagen.

**5-Seguimiento de bordes por histéresis:** Los bordes fuertes son incluidos también en la imagen final, ya que son interpretados como bordes reales y los bordes débiles solo se adicionan en la imagen final si son conexos con un borde fuerte. Los bordes débiles debido a los bordes verdaderos son mucho más propensos a estar directamente conectados a los bordes fuertes. Sin embargo, los bordes débiles que se deben a variaciones de color estarán distribuidos de forma independiente con respecto a los bordes fuertes [39] [40].

### **1.2.5 Basada en el uso de un umbral**

La segmentación basada en umbral, permite separar un objeto de la imagen del fondo que lo circunda, esta técnica se basa en la comparación de alguna propiedad de una imagen con un umbral fijo o variable, realizando dicha comparación para cada uno de los píxeles que conforman la imagen. Si el valor de la propiedad (intensidad, color, etc.) a medir de un píxel supera el valor de un umbral determinado, entonces el píxel pertenece al objeto, en caso contrario, el píxel pertenece al fondo. Cuando la segmentación se basa en el nivel de gris de la imagen, el valor del nivel de gris de cada píxel debe ser

comparado con el umbral, para decidir si el pixel evaluado pertenece al objeto o al fondo. La imagen resultante, es una imagen binaria donde los pixeles de valor uno, pertenecen al objeto y los pixeles con valor cero, pertenecen al fondo de la imagen [22].

Generalmente la selección del valor del umbral, se realiza a partir del histograma de la imagen. Si una imagen está compuesta de un objeto que aparece sobre un fondo, entonces el histograma será bimodal, es decir, si el objeto es mucho más claro que el fondo, en el histograma aparecerán dos picos, el pico correspondiente al objeto ubicado en los valores de gris más elevados y otro para niveles de gris más bajos, correspondientes al fondo de la imagen [22].

### **1.2.6 Contornos Activos:**

El modelo Snake o modelo contorno activo deformable como también se le conoce, es representado matemáticamente como una curva  $v[s]= [x(s), y(s)]$  que se mueve en un espacio constantemente dentro de un número de iteraciones que se puede interpretar como una secuencia de tiempo. Esta curva está representada de forma paramétrica, teniendo como único parámetro a  $s$ . Este parámetro, que está relacionado con ambas variables en el espacio  $(x, y)$ , representa el espacio que ocupa la curva ubicada en la imagen de operación. Existe otro parámetro que indica la cantidad de iteraciones representada como la secuencia de tiempo, el parámetro  $t$ . Se tiene entonces un modelo definido como una curva  $v [s, t]= [x(s, t), y(s, t)]$

El modelo original está representado como una curva (ecuación 2) parametrizada  $v[s]= [x(s), y(s)]$ ,  $s \in [0, 1]$  que se mueve a través de un dominio espacial buscando minimizar el funcional de energía siguiente [41].

$$E_{snake}^* = \int_0^1 E_{snake}(v(s)) ds \quad (4)$$

El modelo Contornos Activos permite ajustar curvas en una imagen a los bordes de la misma. El funcionamiento de este modelo imita a una serpiente que pasa por una superficie no plana, ajustando su cuerpo a las irregularidades del camino mientras que los contornos activos ajustan las curvas a las irregularidades presentes en la imagen. Aunque los algoritmos que ponen en práctica esta técnica tienen un gran costo computacional, existen variantes que eliminan esta desventaja. Este algoritmo es también muy usado en la segmentación de imágenes médicas con el objetivo de identificar regiones con gran variedad de estructuras [42].

### **1.3 Validación de la segmentación**

En el mundo del procesamiento digital de imágenes médicas el desarrollo de nuevas técnicas de segmentación es un campo de interés. Debido a esto se requieren junto con los algoritmos de segmentación, métodos consistentes con los que poder evaluar y validar el desempeño de los mismos.

La validación es la medición (evaluación cuantitativa) del desempeño de un algoritmo de procesamiento de imágenes. La evaluación del desempeño tiene dos aspectos principales [43]:

- eficiencia computacional
- exactitud del algoritmo.

En esta sección se expondrán algunos de los métodos más usados para la evaluación de la exactitud de un algoritmo de segmentación de imágenes médicas, teniendo en cuenta que solo tiene sentido hablar de la eficiencia computacional cuando la exactitud de dos algoritmos que realizan la misma tarea es comparable. Este tipo de evaluación permitirá hacer afirmaciones del tipo: "El algoritmo A es mejor que el algoritmo B para la detección de la opacidad de la cápsula posterior en pacientes operados de catarata" [43].

Los problemas de segmentación de imágenes involucran la división de una imagen en un conjunto de regiones diferentes, de acuerdo a una medida de similitud determinada. Las métricas de desempeño para este tipo de problemas son generalmente más complejas. Existen dos versiones del problema de segmentación: segmentación genérica de bajo nivel y detección y localización de anomalías. En la segmentación genérica de bajo nivel el objetivo es producir descripciones del contenido de una imagen, mientras que en el problema de detección y localización de anomalías el objetivo es encontrar todas las regiones de una imagen que contengan formas anómalas [43].

Al comparar el resultado de un algoritmo de segmentación con una imagen segmentada de referencia (por ejemplo, la segmentación manual de un experto) se tienen en cuenta las siguientes condiciones de comparación [43]:

**Segmentación correcta de una región:** Esto sucede cuando el solapamiento entre la región segmentada por el algoritmo propuesto y la región correspondiente en la imagen de referencia cumplen con cierto umbral mínimo de similitud.

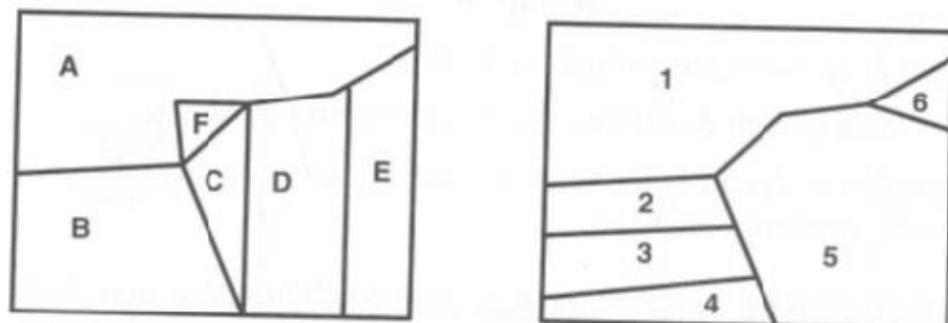
**Sub-segmentación de una región:** La subsegmentación sucede cuando varias regiones de la imagen de referencia corresponden solo a una región de la imagen segmentada por el algoritmo propuesto.

**Sobre-segmentación de una región:** Ocurre cuando varias regiones de la imagen segmentada por el algoritmo propuesto corresponden a una sola región de la imagen de referencia.

**Región faltante:** La región faltante existe cuando una región en la imagen de referencia no corresponde con ninguna región existente en la imagen segmentada por el algoritmo propuesto.

**Región producida por ruido:** Este tipo de regiones aparecen en la imagen segmentada y no corresponden a ninguna región de la imagen de referencia.

Un ejemplo de las condiciones de comparación anteriores se puede ver en la Figura 4 [44].



**Figura 4:** Ejemplo de condiciones de comparación de segmentación (a la izquierda la imagen segmentada y a la derecha el Ground Truth).

Donde:

**A** corresponde a **1** como una instancia de una segmentación correcta.

**5** corresponde a **C, D** y **E** como una sobresegmentación.

**B** corresponde a **2,3** y **4** como una subsegmentación.

**6** es una región perdida.

**F** es una región producida por ruido.

### **1.3.1 Medidas del error**

Las medidas de error son usadas frecuentemente a la hora de diseñar algoritmos de segmentación, clasificación y en la detección de contornos en sistemas de inteligencia artificial. Estas medidas se clasifican en tres grupos [45]:

- Medidas estadísticas de error.
- Medidas de localización.

- Métrica Hausdorff

### 1.3.2. Medidas estadísticas de error.

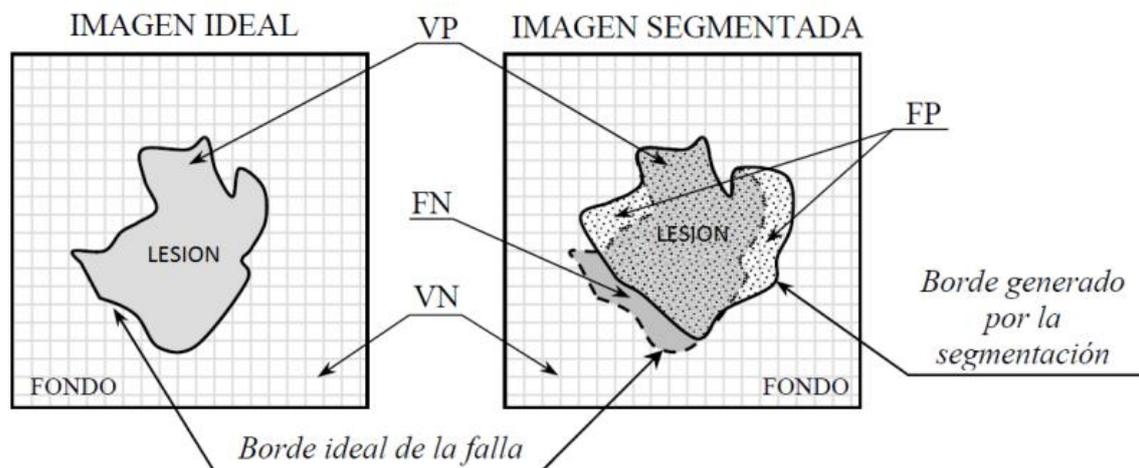
Las medidas estadísticas de error nos darán la medida de la frecuencia con la que se clasifican incorrectamente los píxeles. Esta medida tiene cuatro variables de evaluación definidas a continuación [45] [46]:

**Verdaderos Positivos (VP):** Son los números de píxeles que pertenecen a la imagen clasificados correctamente.

**Verdaderos Negativos (VN):** Son los números de píxeles pertenecientes al fondo clasificados como tales.

**Falsos Positivos (FP):** Son los números de píxeles del fondo clasificados como parte de la imagen.

**Falsos Negativos (FN):** Son los números de píxeles pertenecientes a la imagen clasificados como parte del fondo [47].



**Figura 5:** Ejemplo de variables de evaluación.

### 1.3.3 Métricas de validación en la segmentación de imágenes

La validación consiste en la medición (evaluación cuantitativa) del desempeño de un algoritmo de procesamiento y análisis de imágenes [48]. En esta sección se presentan algunas de las técnicas principales para evaluar la exactitud de un algoritmo de análisis de imágenes sobre imágenes del PENTACAM, que permitirán hacer afirmaciones del tipo:

“El algoritmo A es mejor que el algoritmo B para la detección de opacidad de la cápsula posterior”

### 1.3.4 Clasificación del Error (CE)

Este error cuantifica la segmentación como un proceso de clasificación de píxeles, que refleja el porcentaje de píxeles de fondo clasificados incorrectamente en primer plano y, a la inversa, los píxeles de primer plano erróneamente asignados al fondo. Para un problema de segmentación de dos clases,

$$CE = 1 - \frac{|Bo \cap BT| + |Fo \cap FT|}{|Bo| + |BT|} \quad (5)$$

donde Bo y Fo son el fondo y el primer plano de la imagen original ground-truth ,BT y FT son el fondo y el primer plano de la imagen de prueba, y  $|\cdot|$  es la cardinalidad de los conjuntos. Debido a que el error de clasificación errónea es una medida de disimilaridad o error, entonces, en el caso ideal de coincidencia perfecto,  $CE = 0$  y en los peores casos.  $CE = 1$  [49] [50].

### 1.3.5 Desajuste de Bordos (EMM)

Esta métrica penaliza las discrepancias entre el mapa de borde de la imagen de nivel gris y el mapa de borde obtenido de la imagen de umbral. El EMM se define como:

$$EMM_{\%} = 100 - \frac{100CE}{CE + \omega[\sum_{k \in \{EO\}} \delta(k) + \alpha \sum_{k \in \{ET\}} \delta(k)]} \quad (6)$$

$$, \text{ con } \delta(k) = \begin{cases} |d_k| & \text{si } |d_k| < maxdist \\ |D_{max}| & \text{en otro caso} \end{cases}$$

donde CE es el número de píxeles de borde comunes encontrados entre la imagen ground-truth del suelo y la imagen con umbral, EO es el conjunto de píxeles del borde ground-truth del terreno faltantes en la imagen de umbral, ET es el conjunto de píxeles de borde con umbral en exceso que no tienen lugar en la verdad fundamental,  $\omega$  es la penalización asociada con un píxel de borde original en exceso, y finalmente  $\alpha$  es la relación de las penalizaciones asociadas con un píxel de borde de umbral en exceso a un píxel de borde original en exceso. Aquí  $d_k$  denota la distancia euclidiana del k' píxel de borde sobrante a un píxel de borde complementario dentro de un área de búsqueda determinada por el parámetro *maxdist*. El parámetro habitual es  $maxdist = 0.025Q$ , donde  $Q = M \times N$  es el número de píxeles,  $D_{max} = 0.1Q$ ,  $\omega = 10 / Q$  y  $\alpha = 2$  [51] [52].

### **1.3.6 Índice Rand Ajustado (ARI)**

El ARI ha sido uno de los índices de validación de clúster más exitosos, se recomienda como la opción para medir el acuerdo entre dos particiones en el análisis de clustering, y se puede calcular como:

$$ARI = \frac{4(VP*VN - FP*FN)}{(VP + FP)(FP + VN) + (VP + FN)(FN + VN)} \quad (7)$$

El ARI medirá no solo la separación correcta de elementos pertenecientes a diferentes clases sino también la relación entre elementos de la misma clase. Se presta más atención a la relación entre los elementos que a la relación entre cada elemento y su etiqueta de destino. Resumiendo, el ARI evalúa la capacidad del algoritmo para separar los elementos que pertenecen a diferentes clases. El ARI proporciona una medida estandarizada tal que su valor esperado es cero cuando las particiones se seleccionan al azar y dos cuando las particiones coinciden completamente. En [53] [54] se puede encontrar una explicación amplia sobre el uso de la ARI como parámetro para evaluar la clasificación supervisada.

### **1.3.7 Índice de Jaccard (JI)**

El índice de Jaccard, también conocido como el coeficiente de similitud de Jaccard, ya que mide cuán similares son dos conjuntos, sigue la misma filosofía que el PRI como medida de similitud, excepto que excluye los VN. Se ha definido como:

$$JI = \frac{VP}{VP + FP + FN} \quad (8)$$

Suponiendo que ambos elementos de primer plano (en la imagen segmentada y la de Ground-Truth) son unos, y ambos elementos de fondo son ceros, entonces puede interpretarse como el número de unos compartidos por las variables en las mismas posiciones, que pueden calcularse como la intersección entre estos dos primeros planos, mientras que  $VP + FP + FN$  se puede interpretar como el número total de posiciones donde se producen unos, que se pueden calcular como la unión entre estos dos primeros planos; entonces, el JI también se puede calcular como:

$$JI = \frac{|Fo \cap FT|}{|Fo \cup FT|} \quad (9)$$

Las expresiones anteriores de la JI están entre 0 y 1, correspondiendo los valores más altos a una mejor concordancia entre las formas en la imagen segmentada y ground-truth fundamental, y el valor cero para los peores acuerdos. En algunos entornos es habitual la distancia de Jaccard (JD) como medida de desemejanza; sin embargo,  $JD = 1 - JI$  [55] [56]

### **1.3.8 Índice de Czekanowski (CzI)**

Comúnmente conocido como (coeficiente de similitud-cualitativo), que da el doble de peso a los VP, entonces se usa regularmente con datos de presencia / ausencia en el caso de que haya solo unas pocas coincidencias positivas con respecto al número de desajustes, definidos como: [57] [58]

$$CzI = \frac{2VP}{2VP + FP + FN} = \frac{2|Fo \cap FT|}{|Fo| + |FT|} \quad (10)$$

### **1.3.9 Tasa de Correctos Equilibrada (BCR)**

El BCR es especialmente adecuado para dos clases conjuntos de datos desequilibrados, expresados como: [59] [60]

$$BCR = \frac{0.5VP}{VP + FP} + \frac{0.5VN}{FN + VN} \quad (11)$$

### **1.3.10 Índice de Sokal-Sneath (SSI)**

Similar a la JI, pero destacando los desacuerdos que dieron dos veces el peso a FP + FN, se expresó como: [61] [62]

$$SSI = \frac{VP}{VP + 2(FP + FN)} \quad (12)$$

### **1.3.11 Índice de Rogers-Tanimoto (RTI)**

Similar a la RRI, pero destacando los desacuerdos, se expresó como [63] [64]

$$RTI = \frac{VP + VN}{VP + VN + 2(FP + FN)} \quad (13)$$

### **1.3.12 Índice de Yule (YI)**

$$YI = \left| \frac{VP}{VP + FP} + \frac{VN}{FN + VN} \right| \quad (14)$$

Nota: En Índice de Yule no puede ser usado cuando  $VP + FP = 0$  o  $FN + VN = 0$ . [65] [66]

### 1.3.13 Índice de Russell-Rao (RRI)

Lo mismo PRI y JI, pero considerando el acuerdo en verdaderos positivos, expresados como: [67] [68]

$$RRI = \frac{VP}{VP + FP + FN + VN} \quad (15)$$

### 1.3.14 Índice de Fowlkes-Mallows (FMI)

Cuanto mayor sea el valor de la FMI, mayor será la similitud entre los clústeres y las clasificaciones de referencia:

$$FMI = \frac{VP}{\sqrt{(VP+FP)(VP+FN)}} \quad (16)$$

Se puede ver un ejemplo breve que utiliza la métrica de rendimiento objetivo de referencia completa basada en la tabla de contingencia para evaluar tres procedimientos de umbralización diferentes. La columna a) muestra la imagen del Ground- Truth con 5 × 5 píxeles donde los píxeles blancos son los unos (verdad). Columnas b) a d) son las imágenes obtenidas usando tres métodos diferentes de umbralización. El Método I (columna b) sobresegmenta la imagen en un píxel, el método II (columna c) subsegmenta la imagen en un píxel, y el método III (columna d) obtiene un incorrecto.

a) Ground	b) Method I	c) Method II	d) Method III
ARI	0.9049	0.8963	0.8016
JI	0.8750	0.8571	0.7500
BCR	0.9722	0.9286	0.9008
SSI	0.7778	0.7500	0.6000
RTI	0.9231	0.9231	0.8519
YI	0.9444	0.8571	0.8016
RRI	0.2800	0.2400	0.2400
CZI	0.9333	0.9231	0.8571
FMI	0.9354	0.9258	0.8571
JI Log	0.9464	0.9358	0.8856
RRI Log	0.6382	0.5973	0.5973
CTI	0.6012	0.5891	0.4949

Las penalizaciones por subsegmentaciones son mayores que las obtenidas por las segmentaciones excesivas, excepto para RTI que no puede distinguir estos errores. Además, sanciones por error a) Terreno- b) Método I c) Método II d) Método III las

segmentaciones son mayores que las obtenidas por sobre y segmentaciones, excepto RRI usando versiones lineales y logarítmicas.

Todas estas métricas de desempeño son de alguna manera complementarias entre sí, y deben considerarse juntas para evaluar el rendimiento de un método de umbral determinado. De hecho, la métrica de rendimiento objetivo recomendada para evaluar un procedimiento de umbralización depende con mucho de los significados que pueda tener cualquier error en la segmentación. Un esfuerzo por derivar una medida combinada de las medidas descritas anteriormente, con el intento de simplificar la evaluación de la calidad de la segmentación, podría no dar la mejor aproximación de la medida de calidad general. Esto se debe a que las medidas descritas no son totalmente independientes, comparten una parte de la información y su sensibilidad (pendiente de la función) es diferente.

Algunos autores usan promedios de rango, de modo que, para cada imagen de prueba, clasificaron los algoritmos de segmentación de 1 a TM, donde TM es el número de métodos de segmentación evaluados, según cada criterio por separado. Luego, los rangos (no los puntajes reales) se promedian sobre las imágenes y las métricas de rendimiento objetivo de referencia completa utilizadas. [69] [70]

#### **1.3.14 Modificación de la Distancia de Hausdorff Normalizada (NMHD)**

Evalúa la similitud de forma de las regiones con umbrales con las formas *ground-truth* del suelo mediante la siguiente fórmula

$$NMHD = \frac{1}{1 + 0.2(MHD-1)} \quad (17)$$

donde MHD es la distancia de Hausdorff alterada definida

$$MHD (F_0, F_T) = \frac{1}{|F_0|} \sum_{f_0, F_0} d(f_0, F_T) \quad (18)$$

donde  $d(f_0, F_T)$  denota la distancia euclidiana mínima de cualquier píxel mal clasificado en la imagen con umbral de píxel en la imagen *ground-truth*,  $|F_0|$  es el número de píxeles en primer plano en la imagen *ground-truth*. Para un rendimiento perfecto  $NMHD = 1.25$ , y mientras más mal sea el resultado va a tender a  $NMHD = 0$ .

2. Hay métricas de desempeño de métodos de segmentación que son no referenciadas, también denominadas métricas ciegas ya que no utilizan imagen de referencia. Entre estas, una de las más utilizadas es la medida de uniformidad (UM), también conocida como uniformidad de región [71] [72].

## **1.5 Herramientas usadas durante el desarrollo**

En este epígrafe se describen las tecnologías y herramientas a utilizar durante el desarrollo del software, se exponen las principales características de la metodología de software escogida, así como las ventajas que esta ofrece para este proyecto. Se describe la herramienta para el modelado, el entorno integrado de desarrollo y se argumenta acerca de las características y beneficios que ofrece el lenguaje de programación seleccionado, así como las ventajas que trae consigo el sistema gestor de base de datos propuesto.

### **1.5.1 Metodología de desarrollo de Software**

El proceso de desarrollo de software se define como el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software, teniendo como objetivo la obtención de un producto que cumpla con las expectativas del cliente [73]. Este proceso es una de las tareas más difíciles de la ingeniería; prueba de ello es la existencia de numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte, aquellas propuestas más tradicionales o robustas se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán, o sea se basa en el uso exhaustivo de documentación durante todo el ciclo del proyecto. Por otra parte se encuentran las metodologías ágiles o ligeras las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas [74]. La selección de una metodología apropiada resulta en ocasiones un factor esencial para obtener un software con la calidad esperada.

Luego de la evaluación de las metodologías de desarrollo y el estudio de sus características, etapas de desarrollo y ventajas que posibilitan, se determina utilizar una metodología ágil, dado que la prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software. El cliente es parte del equipo de desarrollo, el equipo de desarrollo es de solo una persona, además de la dificultad para un equipo de desarrollo pequeño el adoptar una metodología robusta a causa de la cantidad de documentación generada y la alta resistencia a los cambios durante el desarrollo, lo cual permitió identificar la Programación Extrema (XP) como una alternativa acertada, además de ser la metodología de desarrollo del grupo de investigación [75] [73].

### **Metodología de software XP**

## *Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

La metodología XP además de ser una de las más conocidas y más usadas de las metodologías ágiles fue seleccionada para guiar el proceso de desarrollo de la presente investigación, debido a que se está en presencia de un proyecto pequeño, de corta duración, con un reducido equipo de desarrollo, donde el cliente forma parte activa del mismo, permitiendo la comunicación y retroalimentación frecuente entre ambas partes. Una de las prácticas más significativas que posee esta metodología es que con ella es posible simplificar el diseño para agilizar el desarrollo, así como facilitar el mantenimiento del sistema. La ejecución de las pruebas unitarias frecuentemente permite descubrir fallos debido a cambios recientes en el código. Es también recomendable utilizar esta metodología debido a que los requisitos tienen altas probabilidades de cambiar con el tiempo [73]. Esta metodología ha sido la utilizada durante el desarrollo del software PANDOC, además, es la metodología definida por el grupo de investigación científica *Inteligencia Artificial, Programación e Innovación* por sus siglas en inglés (AIRI), creador del mismo, lo cual constituye un buen criterio a tener en cuenta siendo el resultado de este proyecto un módulo para el software antes mencionado.

Algunas de las características de XP son [73]:

- Pruebas unitarias: se basa en las pruebas realizadas a los principales procesos con el objetivo de detectar futuros errores.
- Refabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.
- Entrega de software en espacio de tiempo pequeño: La tendencia de entregar software en espacios de tiempo cada vez más pequeños con exigencias de costos reducidos y altos estándares de calidad.

Ventajas que brinda XP [73]:

- Comienza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- El cliente o el usuario se convierte en parte del equipo.

XP consta de cuatro fases para el desarrollo [73] [76]:

## *Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

- 1. Planificación:** Es la fase donde se realiza una recopilación de todos los requerimientos del proyecto, se crean las HU, el plan de iteraciones y el plan de entregas.
- 2. Diseño:** La metodología XP hace especial énfasis en los diseños simples y sencillos, con el objetivo de procurar hacerlo todo lo menos complicado posible para el cliente o usuario, se crean las tarjetas CRC, las cuales definen una clase expresando sus funcionalidades y las otras clases con las que colabora.
- 3. Desarrollo:** Esta fase consiste en establecer una buena comunicación entre el equipo y el cliente, para que los desarrolladores puedan codificar todo lo necesario para el proyecto que se requiere. Además, se definen las tareas de ingeniería para que los desarrolladores tengan una guía para implementar todas las HU.
- 4. Pruebas:** Se basa en la comprobación del funcionamiento de la codificación que se halla implementado, garantizando la evaluación de las distintas tareas en las que ha sido divididas las HU de manera que si se encuentran errores darles solución para prever que no vuelvan a ocurrir.

### **1.5.2 Lenguaje de Programación**

Un lenguaje de programación es un idioma artificial diseñado para expresar instrucciones que pueden ser llevadas a cabo por un ordenador. Puede usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión o como modo de comunicación humana. Permiten especificar de manera precisa sobre qué datos debe operar una computadora, cómo deben ser almacenados o transmitidos y qué acciones debe tomar bajo una gran cantidad de opciones posibles. Todo esto, a través de un lenguaje que intenta ser relativamente próximo al lenguaje humano o natural [77].

Debido a que el software PANDOC está desarrollado en el lenguaje de programación Java y el resultado de este proyecto constituye un módulo de esa aplicación informática se escogió el mismo para desarrollar la solución propuesta. Además, se cuenta con abundante documentación sobre el lenguaje ya que es uno de los más utilizados. A continuación, se mencionan algunas de sus características.

#### **Características fundamentales de Java [78]:**

- **Simple:** elimina la complejidad de otros lenguajes como C y se enfoca en el contexto de los lenguajes orientados a objetos.
- **Robusto:** maneja la memoria de la computadora para que el programador no se tenga que preocupar por ello, además de realizar verificaciones en busca de errores lo mismo en tiempo de compilación que en tiempo de ejecución.
- **Portable:** un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el intérprete de Java, ya que su código compilado es interpretado.
- **Multiproceso:** puede ejecutar diferentes líneas de código al mismo tiempo.
- **Dinámico:** no es necesario que compile todas las clases de un programa para que este funcione. Al efectuar al menos un cambio en alguna de las clases, Java se encarga de realizar un enlace dinámico o una carga dinámica para encontrar las clases.
- **Interpretado:** corre en máquina virtual.

### **1.5.3 Entorno integrado de desarrollo**

Para el desarrollo de la solución, como Entorno de Desarrollo Integrado (IDE, según sus siglas en inglés) se seleccionó el NetBeans 8.2. Es libre, de código abierto y contiene todas las herramientas necesarias para crear aplicaciones profesionales de escritorio, empresariales, web y aplicaciones móviles con la plataforma Java. Es conocido por la integración con el lenguaje Java, facilita el desarrollo utilizando funcionalidades como completamiento de código, coloreo de sintaxis, permite la utilización y edición de los componentes visuales de forma sencilla. Incluye también el control de versiones, lo cual representa una ventaja debido a que permite administrar las diferentes versiones del código fuente. Además es un entorno de desarrollo disponible para varios sistemas operativos como Windows, Mac, Linux y Solaris [78]:

### **1.5.4 Herramienta para el modelado**

Como herramienta para el modelado de la solución se seleccionó Visual Paradigm for UML (VP-UML v8.0), una de las líderes del mercado de las llamadas herramientas de Ingeniería de Software Asistida por Computadora (CASE, según sus siglas en inglés). Esta herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML. El VP-UML en su versión v8.0 soporta los principales estándares de la industria tales como el Lenguaje de Modelado Unificado (UML, según sus siglas en inglés), SysML, BPMN, XMI, entre otros. Ofrece un

conjunto completo de herramientas, brindando a los equipos de desarrollo de software todo lo necesario para la planificación de software, captura de requisitos, planificación de controles y modelado de clases y de datos. Esta herramienta ofrece gran interoperabilidad entre diagramas ya que es capaz de exportar los diagramas de un modelo a otro con mucha facilidad, ahorrando de esta manera tiempo, lo cual es crucial para el desarrollo, además de hacer posible la generación de código Java desde los diagramas [79].

#### **1.5.4 Sistema Gestor de Base de Datos.**

Como sistema gestor de base de datos (SGBD) se utilizó SQLite, el cual es un SGBD relacional que está contenido en una biblioteca en C relativamente pequeña (alrededor de 500 kb). Es un proyecto de dominio público creado por Dr. Richard Hipp. A diferencia de los SGBD cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones [80].

Esto reduce la latencia en el acceso a la base de datos (BD), debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de BD (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host. Este diseño simple se logra bloqueando todo el fichero de BD al principio de cada transacción. Este SGBD dispone de una completa interfaz orientada a objetos, con distintas funciones que facilitan la manipulación de datos y posee funciones muy similares a las que se pueden manejar con MySQL.

Combina el motor y la interfaz del BD en una única biblioteca, y almacena los datos en un único archivo de texto plano. Esto hace que cada usuario pueda crear tantas BD como desee sin la necesidad de la intervención de un administrador de BD que gestione los espacios de trabajo, usuarios y permisos de acceso

La potencia de SQLite se basa fundamentalmente en la simplicidad, lo que hace que no sea una buena solución en entornos de tráfico muy elevado y/o alto acceso concurrente a datos, además encapsula toda la BD en un único fichero. El hecho de almacenar toda la BD en un único archivo, facilita la portabilidad de los datos, y solamente tiene la restricción del espacio de disco asignado.

Se puede utilizar SQLite de dos formas:

- Como gestor de base de datos local en un computador. De esta forma, se pueden gestionar BD con SQLite igual que si se estuviese trabajando con un sistema gestor de base de datos como MySQL, sin necesidad de instalar nada, ya que SQLite se compone de un único archivo ejecutable.
- Como una extensión más de PHP, utilizando las funcionalidades de SQLite configuradas, o bien como módulo de PHP, o como biblioteca; sin necesidad de tener instalado o conectar con un servidor de base de datos. Ofrece una rápida interfaz de la BD almacenada en el archivo de texto plano.

Varios procesos o hilos pueden acceder a la misma BD sin problemas. Varios accesos de lectura pueden ser servidos en paralelo. Un acceso de escritura sólo puede ser servido si no se está sirviendo ningún otro acceso concurrentemente. En caso contrario, el acceso de escritura falla devolviendo un código de error (o puede automáticamente reintentarse hasta que expira un timeout configurable). Esta situación de acceso concurrente podría cambiar cuando se está trabajando con tablas temporales. Sin embargo, podría producirse un deadlock debido al proceso multihilo.

SQLite Admin es un administrador de bases de datos SQLite que básicamente permite [81]:

- Abrir, crear, compactar bases de datos.
- Crear, renombrar y eliminar tablas vacías.
- Crear, renombrar y eliminar columnas.
- Crear, modificar y eliminar registros.
- Encriptar o reencriptar bases de datos.
- Correr sentencias SQL personalizadas.
- Crear y eliminar índices, triggers y cursores.

### **1.5.6 Herramienta para el análisis estadístico utilizando SPSS**

El programa estadístico SPSS (del inglés *Statistical Package for the Social Sciences*) es uno de los programas de mayor uso en los Estados Unidos de Norteamérica, así como en América Latina. Los procedimientos estadísticos que incluye son de mucha utilidad para desarrollar y subsecuentemente analizar bases de datos para aplicaciones prácticas o para diversas necesidades de investigación. SPSS permite manejar bancos de datos de gran magnitud y también efectuar análisis estadísticos muy complejos. [82]

Familiarizarse con las diversas opciones y procedimientos estadísticos de un programa como SPSS permite la realización de varios procedimientos estadísticos, brindando

gráficos, ejemplos de análisis de hipótesis en una interfaz sencilla y con una curva de aprendizaje pequeña.

### **Ventajas y desventajas del programa SPSS**

SPSS 14.0 se desarrolló en una forma que, aparentemente, es fácil de navegar (user-friendly). El programa utiliza una serie de cuadros de diálogo (dialog boxes) que permiten, en forma secuencial, determinar las acciones a tomar y seleccionar aquellos análisis útiles. Sin embargo, si el usuario no tiene experiencia previa utilizando SPSS o si sus conocimientos de estadística no están actualizados, es difícil discernir qué opciones seleccionar. Esta característica del programa puede convertirse en una desventaja e inhibir al usuario. Otro aspecto del programa que puede causar inconvenientes es el hecho de que la mayoría de los reportes de resultados contiene un nivel excesivo de información que, más que aclarar, confunde al usuario. En este libro nuestro objetivo es minimizar el uso de términos técnicos y al mismo tiempo entrenar al usuario para que reconozca los términos estadísticos necesarios y descarte aquellos que no se necesitan para su problema específico. Desgraciadamente, el programa SPSS incluye una gran cantidad de información en “forma automática” (by default) que distrae al usuario.

## **1.7 Conclusiones del capítulo**

Al término de la definición del marco teórico se arribaron a las siguientes conclusiones:

- El software PASI para la validación de la segmentación de los tomogramas Scheimpflug del PENTACAM no cuenta con varias métricas que son importantes para la validación de la segmentación de imágenes.
- Se escoge a Java como lenguaje de programación, NetBeans como IDE, SQLite como SGBD, Visual Paradigm como herramienta para el modelado y se define la metodología Programación Extrema en correspondencia con su utilización por el software PASI. Además del software SPSS para la validación estadística de los resultados.

## **Capítulo 2. Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.**

Este capítulo tiene como objetivo describir las actividades desarrolladas durante todo el proceso de análisis y diseño. Se parte del análisis de los softwares PANDOC y PASI que son base de la propuesta de solución. La misma se detalla y se describe su arquitectura. Los requisitos funcionales y no funcionales, se describe la fase inicial de la metodología de desarrollo utilizada para la implementación de la solución propuesta y se obtienen los artefactos de esta fase, tales como, las Historias de Usuarios, Plan de Iteraciones, Plan de Duración de Iteraciones, Plan de Entregas y tarjetas CRC.

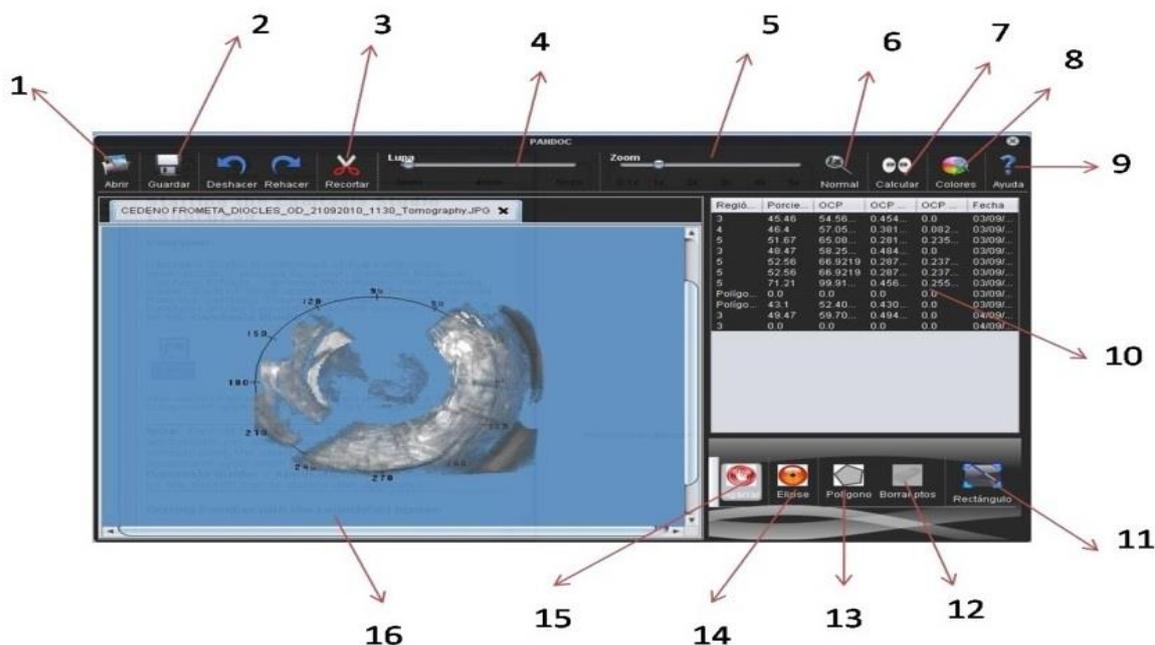
### **2.1 Programa analizador de Opacidad Capsular**

El Programa analizador de Opacidad Capsular (PANDOC) es una herramienta para la cuantificación y análisis de la OCP. Es un software intuitivo con una curva de aprendizaje pequeña, el cual enfoca sus resultados en pos del análisis estadístico que aporten información de alto valor para el estudio y disminución de esta anomalía postoperatoria. Es un software de nuevo tipo, ya que a diferencia del resto toma las imágenes de los tomogramas de Scheimpflug del PENTACAM y brinda datos antes no vistos por la comunidad oftalmológica mundial como la correlación opacidad/transparencia en el centro y la periferia, el mismo comprende funcionalidades automáticas y semiautomáticas orientadas solo al uso de imágenes de tomogramas del PENTACAM, haciendo más intuitiva la interfaz actual y la focaliza en su propósito principal [83].

#### **2.2.1 Interfaz visual**

A continuación, se describen los principales elementos que componen la interfaz visual del software PANDOC y las funcionalidades de los mismos, mostrados en la figura:

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*



**Figura 6:** Interfaz visual de PANDOC.

Para realizar la segmentación de la OCP en imágenes provenientes del PENTACAM dentro del grupo de investigación AIRI se han propuesto 3 algoritmos, en función de obtener mejores resultados en tiempo de ejecución y en calidad de la segmentación:

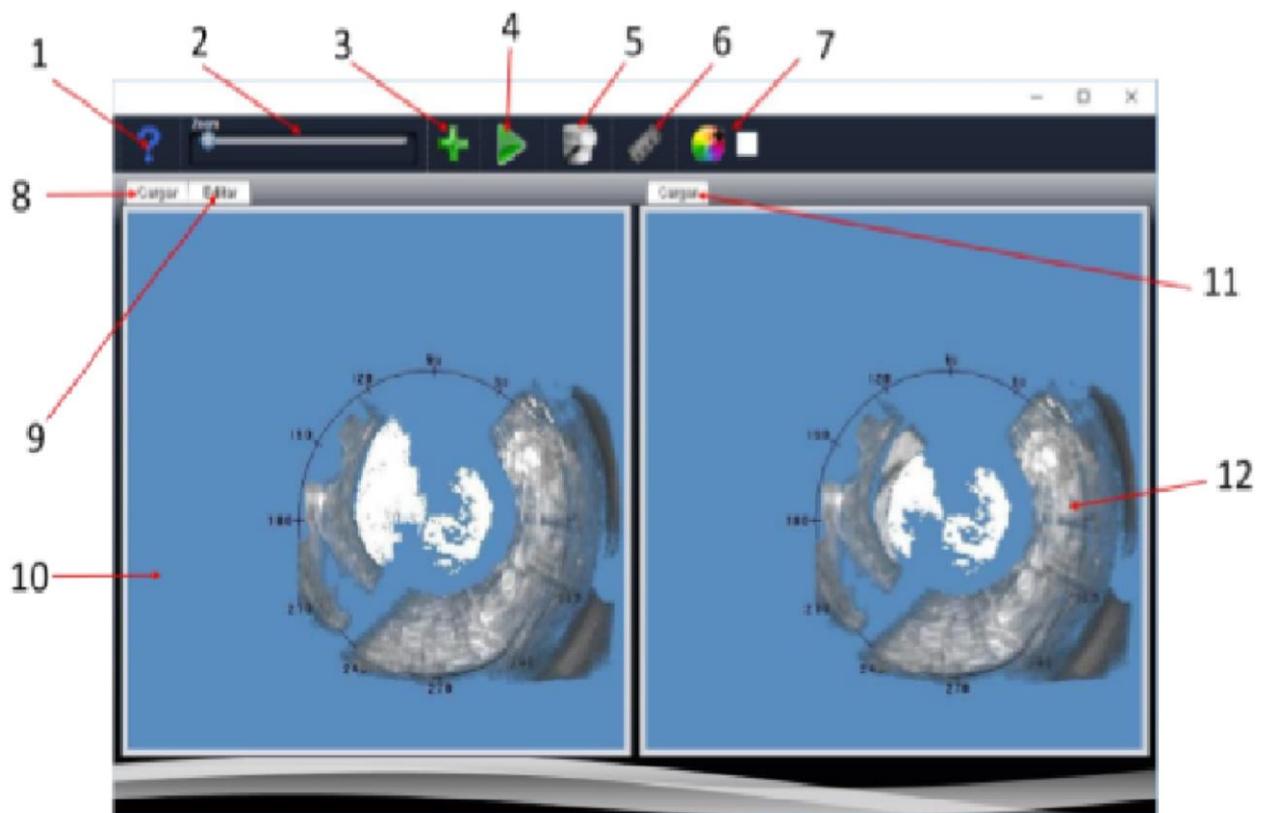
- 1 **Algoritmo para identificar OCP (SBC\_PP):** Segmentación de la OCP mediante Sistema Basado en Casos y algoritmo de trilateración celular, donde cada pixel diferente del fondo entre los 3 y 6 mm del centro de la estructura ocular es un nuevo caso. Los rasgos están basados en el comportamiento de la intensidad del píxel, su cercanía al centro y el comportamiento de sus vecinos. Este algoritmo está formulado fundamentalmente bajo el concepto de que la capsulorrexia solo varía entre esas distancias.
- 2 **Algoritmo para identificar OCP (SBC\_CC):** Segmentación de la OCP mediante sistema basado en casos y algoritmo de componentes conexas, donde cada región extraída que tenga elementos entre los 3 y los 6 mm del centro de la estructura ocular es un nuevo caso. Los rasgos están basados en las características (como valores de intensidad y tamaño) y variaciones de las regiones (como forma), también se tuvo en cuenta la cercanía al centro de la región. Este algoritmo atiende a la formación de la opacidad en el saco posterior, que se van acumulando cada una de las células en grupos (regiones opacas).

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

- 3 **Algoritmo para identificar OCP (TCH\_CA):** Segmentación de la OCP mediante el uso de la transformada circular de Hough y algoritmo de contornos activos, donde primeramente se realiza un conjunto de técnicas de filtrado, para con la transformada circular de Hough detectar la forma circular de la capsulorrexis, para posteriormente usando contornos activos ajustar este círculo a los bordes (contornos) reales de la misma. Esta región comprendida por la capsulorrexis es OCP.

## 2.2 Programa analizador de la segmentación de imágenes

Para determinar cuál de estos algoritmos era más eficiente se realiza un software que aplicaba un conjunto de métricas de validación de la segmentación sobre las imágenes del PENTACAM. Este software era titulado Programa analizador de segmentación de imágenes por sus siglas (PASI).



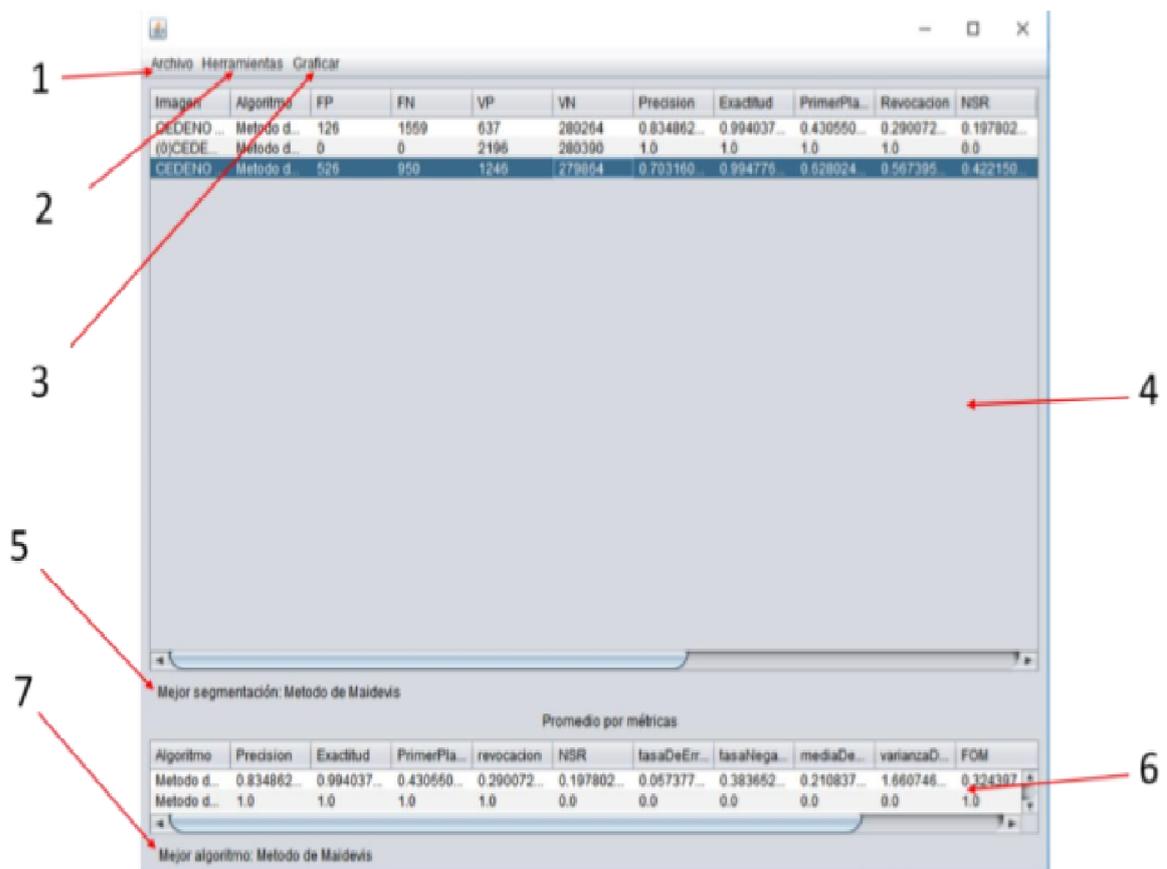
**Figura 7:** Interface principal del software PASI.

1. Muestra la ayuda del software.
2. Permite ampliar la imagen en la proporción seleccionada por el usuario.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

3. Permite insertar un nuevo algoritmo al sistema y consultar la descripción de los mismos.
4. Permite iniciar el cálculo de las métricas.
5. Permite consultar la tabla de resultados de las métricas.
6. Permite ponderar por nivel de importancia para el usuario cada métrica.
7. Permite obtener el color de segmentación de la imagen segmentada automáticamente con el objetivo de que al segmentar manualmente la imagen Ground Truth coincidan los colores de segmentación y se obtengan resultados válidos.
8. Permite cargar la imagen Ground Truth pre-segmentada o segmentarla manualmente.
9. Permite desplegar la ventana de edición de la imagen Ground Truth.
10. Permite visualizar la imagen Ground Truth.
11. Permite cargar la imagen segmentada manualmente.
12. Permite visualizar la imagen segmentada automáticamente.

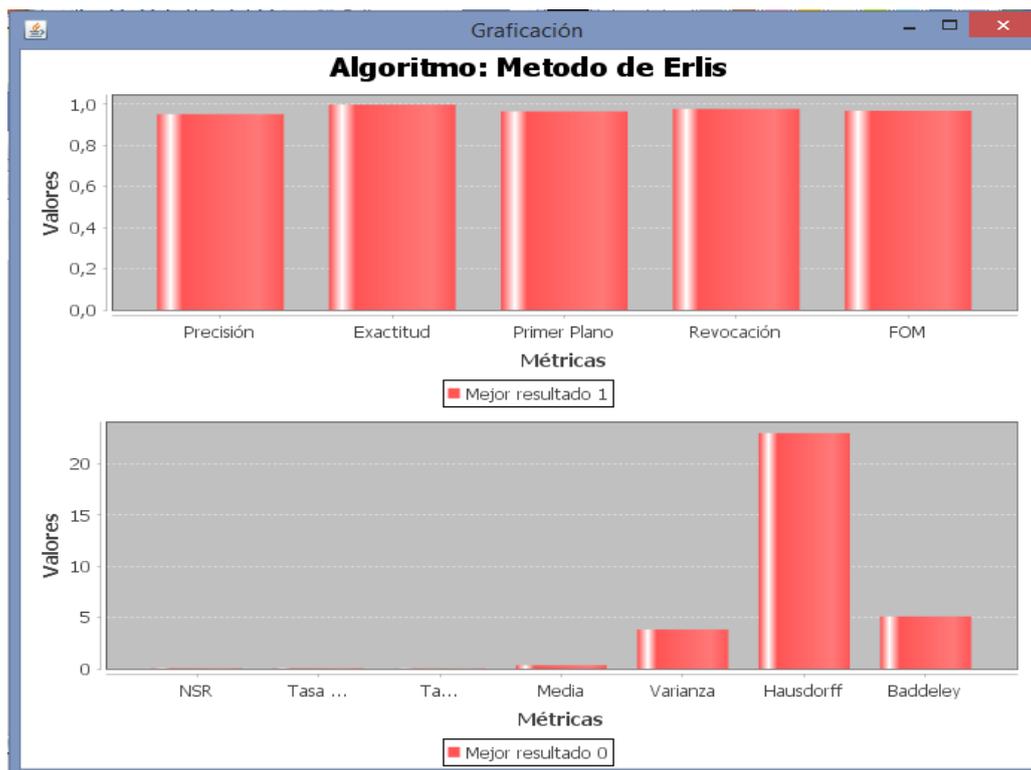
*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*



**Figura 8:** Ventana de visualización de los resultados de las métricas.

1. Permite exportar e importar los resultados de las métricas en un archivo Excel.
2. Permite eliminar la tabla de resultados o una fila seleccionada por el usuario.
3. Permite graficar los resultados de las métricas tras seleccionar una fila, por imagen, por algoritmo, o los de la fila seleccionada.
4. Permite visualizar los resultados de las métricas.
5. Permite visualizar el nombre del algoritmo mejor segmentado.
6. Permite visualizar los promedios por métricas entre los algoritmos probados.
7. Permite visualizar el nombre del mejor algoritmo entre los probados.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*



**Figura 9:** Imágenes de graficación por métricas.

### 2.3 Descripción de la solución

A continuación, se muestra el diagrama de flujo del sistema propuesto y se describen los pasos que sigue el mismo.

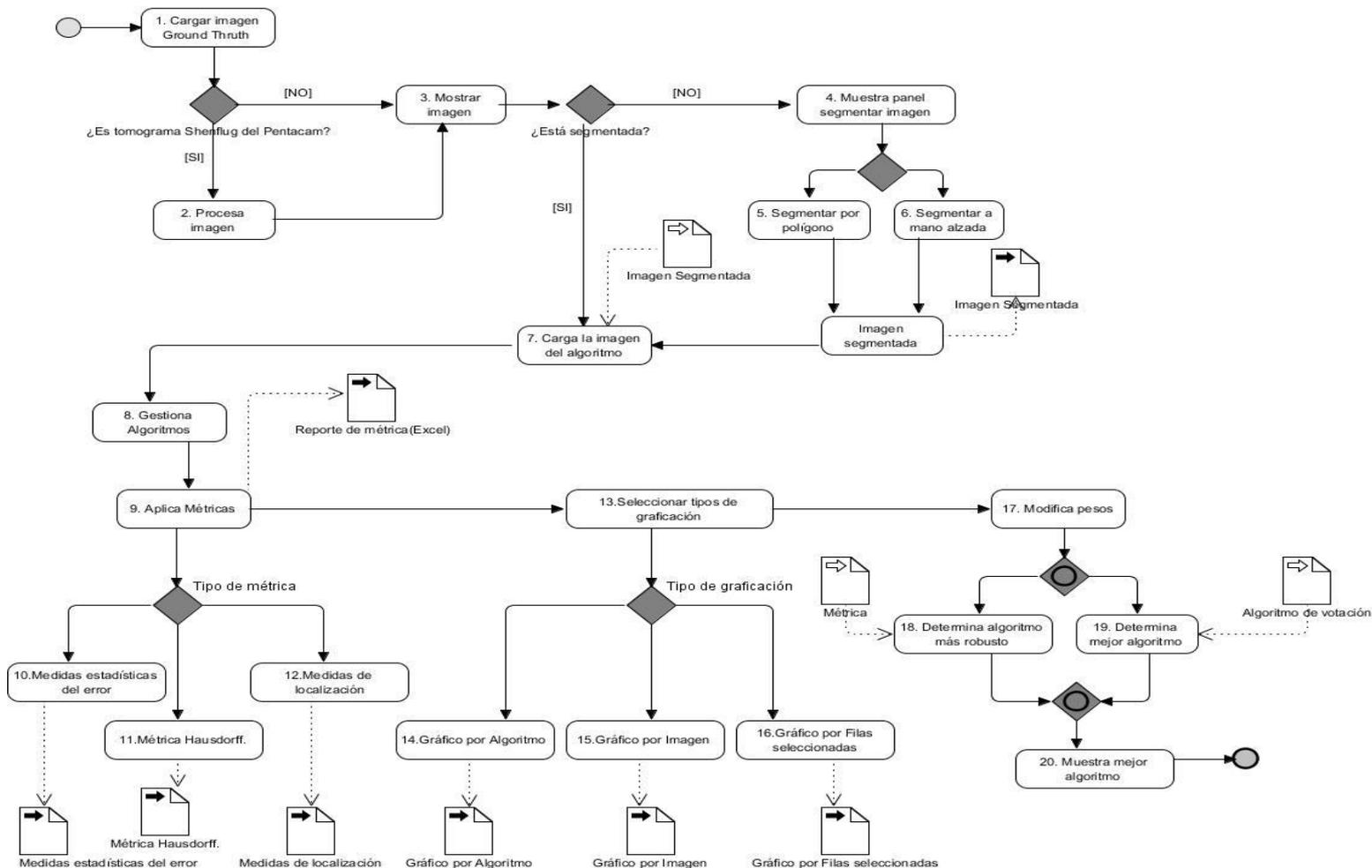


Figura 10: Diagrama de flujo del sistema.

Pasos:

1. Carga la imagen Ground Truth en el formato png y se verifica si es un tomograma Shenflug proveniente del PENTACAM.
2. Recorta la imagen proveniente del PENTACAM de forma automática.
3. Muestra la imagen cargada y el usuario verifica visualmente si está segmentada o no.
4. Muestra el panel de herramientas para realizar la segmentación manual de la imagen.
5. Brinda la posibilidad al usuario de seleccionar la región a segmentar de forma poligonal, punto a punto.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

6. Brinda la posibilidad al usuario de seleccionar la región a segmentar de forma libre, a mano alzada.
7. Carga la imagen del algoritmo segmentada automáticamente y la muestra.
8. Introducen los datos del algoritmo a ser analizado (datos como nombre del algoritmo, nombre del creador, descripción etc.).
9. Aplican sobre la imagen segmentada automáticamente, las medidas de error propuestas las cuales se clasifican en medidas estadísticas de error (Clasificación del Error, Índice Rand Ajustado, Índice de Jaccard, Índice de Czekanowski, Índice de Sokal-Sneath, Índice de Rogers-Tanimoto, Índice de Yule, Índice de Russell-Rao y Índice de Fowlkes-Mallows), medidas de localización (Desajuste de bordes y Tasa de Correctos Equilibrada) y métrica Hausdorff (Modificación de la Distancia de Hausdorff Normalizada).
10. Muestran de forma tabular los resultados de la aplicación de las métricas.
11. Tras mostrar los resultados de las métricas se brinda la posibilidad al cliente de exportarlos como un documento Excel.
12. Tras mostrar los resultados de las métricas se brinda la posibilidad al cliente de mostrarlos de forma gráfica.
13. Al seleccionar el usuario el modo de graficación por imagen el sistema recorrerá todas las imágenes del mismo nombre, calculará el promedio de las métricas entre ellas y mostrará el resultado de forma gráfica.
14. Al seleccionar el usuario el modo de graficación "fila seleccionada" el sistema graficará los valores de cada métrica en esa fila.
15. Al seleccionar el usuario el modo de graficación "por algoritmo" el sistema recorrerá todas las imágenes que utilizan el mismo algoritmo, calculará el promedio de las métricas entre ellas y mostrará el resultado de forma gráfica.
16. En esta opción el usuario puede ponderar cada métrica según el nivel de importancia que el estime para cada métrica con el objetivo de obtener el mejor algoritmo de segmentación.
17. El sistema determinará el algoritmo más robusto calculando la media de todas las métricas entre todas las imágenes y mostrando una tabla con los valores medios de todas las métricas por algoritmo.
18. El sistema determinará el mejor algoritmo calculando la media entre los valores medios de cada algoritmo de segmentación probado apoyándose en un algoritmo de Votación para su selección.
19. El sistema mostrará el nombre del mejor algoritmo al igual que el nombre del algoritmo más robusto (el que mostró mejores resultados en las métricas).

## **2.4 Fase de Planificación**

La metodología XP define como fase inicial del desarrollo de software la planificación. Durante el transcurso de esta etapa se realiza el proceso de identificación y elaboración de las historias de usuario (HU), además el equipo de trabajo se familiariza con las tecnologías y herramientas seleccionadas para el desarrollo. El cliente define el nivel de prioridad con que se deben implementar las HU, así como la estimación del esfuerzo que costará implementar las mismas. El resultado de la presente fase es un plan de entregas donde se realiza una estimación de las versiones que tendrá el producto en su elaboración, de forma tal que sea una guía durante el desarrollo [84].

### **2.4.1 Historias de Usuario**

Uno de los artefactos definidos por la metodología XP son las HU las cuales son utilizadas para especificar las funcionalidades que brindará el sistema. Cada HU es una representación de un requerimiento de software escrito en una o dos frases, utilizando el lenguaje común del usuario. Estas representan una forma rápida de administrar los requerimientos de los usuarios, sin necesidad de elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos [85]. En correspondencia con la prioridad asignada por el cliente a cada HU y teniendo en cuenta la complejidad y riesgo determinado por el programador, se realiza la estimación de cada una de las identificadas.

Como parte de la presente investigación, el empleo de este artefacto fue clave, a continuación (Tabla#1) se muestra uno de los ejemplos generados.

*Tabla 1. Historia de usuario # 5*

Historia de usuario	
<b>Número:</b> 5	<b>Nombre:</b> Calcular Índice de Jaccard(JI)
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b>

**Descripción:** Dada la imagen segmentada y el ground thrust se obtiene el Índice de Jaccard(JI)

**Observaciones:** Deben haberse calculado los cantidades de (VP,VN,FP y FN)

### **2.4.2 Estimación de esfuerzos por Historias de Usuario**

En esta fase se define las prioridades de cada historia de usuario, y consecuentemente se realiza una estimación del esfuerzo necesario de cada una de ellas. Tras la definición de prioridades de cada HU se realiza una estimación del esfuerzo necesario de cada una de ellas, donde la unidad de estimación es el punto, que equivale a una semana ideal de programación. Esta estimación arrojó los resultados siguientes:

*Tabla 2. Puntos de estimación por historias de usuario*

No	Historias de usuario	Puntos de estimación
1	Calcular índice (VP,VN,FP,FN)	2
2	Calcular Tasa de Correctos equilibrada	2
3	Calcular métrica de Clasificación del error	2
4	Calcular índice Desajuste de Bordes	2
5	Calcular índice de similitud de Jaccard	2
6	Calcular índice de Yule	2
7	Calcular índice de Fowlkes-Mallows	2
8	Calcular índice de similitud de Czekanowki	2
9	Calcular índice Rand ajustado	2
10	Índice de Rogers-Tanimoto	2
11	Índice de Sokal-Sneath	2
12	Índice de Russell-Rao	2
13	Modificación de la Distancia de Hausdorff Normalizada	2
14	Determinar mejor segmentación	2
15	Calcular algoritmo más robusto	1

16	Gestionar resultados	1
17	Graficar y exportar resultados	2

### **2.4.3 Requisitos no funcionales**

Los requisitos no funcionales son aquellos que no describen información a guardar, ni funciones a realizar, sino características de funcionamiento que debe cumplir o cumple el producto de software [86]. Estos requisitos tienen lugar por la demanda del sistema de ciertas funciones o cualidades, o porque el cliente desea que ese requerimiento forme parte del producto de software final.

**Requisitos de software:** Se debe tener instalada en la computadora la máquina virtual de Java en su versión 8 o superior.

**Requisitos de hardware:** La computadora deben tener como mínimo las siguientes características:

- 1 GB de memoria RAM.

Más de un núcleo físico con frecuencia mayor de 1.6 GHz

### **2.4.4 Plan de iteraciones**

Una vez definidas las historias de usuarios y estimado el tiempo y el esfuerzo para el desarrollo de cada una de ellas, se propone un plan de iteraciones donde las historias de usuario están contenidas. Una iteración tiene como finalidad implementar las HU que se consideran necesarias atendiendo a su relevancia e impacto. Partiendo de los aspectos tratados anteriormente, la aplicación que se pretende construir se desarrollará en 3 iteraciones, expuestas de forma detallada a continuación:

#### **Iteración 1**

En esta iteración se realizan todas las funcionalidades relacionadas con el cálculo y estimación de las métricas de error y los aciertos entre una imagen segmentada de manera automática y el Ground Truth.

#### **Iteración 2**

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

En esta iteración se realiza el cálculo de los diferentes índices de similitud obtenidos en la literatura entre la imagen segmentada de manera automática y el Ground Truth.

**Iteración 3**

En esta iteración se realizan las HU que se encargan de mostrar un resultado estadístico de forma gráfica, ya que en ellas se implementan las funcionalidades de gestión de resultados, calcular mejor segmentación y calcular algoritmo más robusto, los cuales se generan a través de la media.

*Tabla 3. Plan de duración de las iteraciones*

Iteraciones	Historias de usuario	Duración
Iteración 1	Calcular índice (VP,VN,FP,FN)	10 semanas
	Calcular tasa de Correctos equilibrada	
	Calcular métrica de clasificación del error	
	Modificación de la Distancia de Hausdorff Normalizada	
	Calcular Desajuste de Bordes	
Iteración 2	Calcular índice de similitud de Jaccard	16 semanas
	Calcular índice de Yule	
	Calcular índice de Fowlkes-Mallows	
	Calcular índice de similitud de Czekanowki	
	Índice de Russell-Rao	
	Índice de Rogers-Tanimoto	
	Índice de Sokal-Sneath	
	Calcular índice Rand ajustado	
Iteración 3	Determinar mejor segmentación	6 semanas
	Calcular algoritmo más robusto	
	Gestionar resultados	
	Graficar y exportar resultados	

### **2.2.5 Plan de entrega**

El plan de entregas es el cronograma que define cuáles HU serán agrupadas para conformar una entrega, y el orden en que se realizarán estas [87]. Este plan se encarga de definir el número de liberaciones que se realizarán en el transcurso del proyecto y las iteraciones que se requieren para desarrollar cada una de ellas.

*Tabla 4. Plan de entrega*

	<b>Iteración 1</b>	<b>Iteración 2</b>	<b>Iteración 3</b>
<b>Cantidad de HU</b>	5	8	4
<b>Fecha de entrega</b>	12/01/2018	10/03/2018	27/04/2018

## **2.5 Diseño de la propuesta de solución**

En este epígrafe se hace alusión a los principales elementos que componen el diseño del sistema propuesto, donde siguiendo los planteamientos de la metodología XP este debe ser un diseño simple y de fácil interacción con el usuario, basándose fundamentalmente en el desarrollo de las tarjetas Clases, Responsabilidad y Colaboración (CRC). También en este epígrafe se detallan las tareas generadas por cada HU, así como las tres iteraciones llevadas a cabo en la fase anterior.

### **2.5.1 Tarjetas CRC**

La metodología XP, a diferencia de otras metodologías, en lugar de utilizar diagramas para desarrollar modelos representa las clases mediante tarjetas *Clases, Responsabilidad y Colaboración*. Las tarjetas CRC ayudan a definir actividades durante el diseño del sistema y evitar el enfoque procedimental destacando la orientación a objetos [88]. Al poder trabajar con una metodología basada en objetos estas tarjetas permiten que el equipo de desarrollo completo contribuya también en la tarea del diseño. La estructura física de las tarjetas CRC facilita la interacción entre los integrantes del proyecto, en sesiones en las que se aplican técnicas de grupos como juego de roles o tormenta de ideas y se ejecutan escenarios a partir de la especificación de los requisitos, historias de usuarios o casos de uso. De esta forma, van surgiendo las entidades del sistema junto con sus responsabilidades y colaboraciones [89]. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda y las clases que se implican en cada responsabilidad a la derecha, en la misma línea que su requerimiento correspondiente.

Tabla 5. Tarjeta CRC de la clase NewJFrame

Clase: NewJFrame	
Responsabilidades	Colaboradores
Permite mostrar, editar y mover la imagen proveniente del PENTACAM.	LabelImage, MouseEvent, Point
Realizar el cálculo de las métricas de error.	Metodos
Permite cargar la imagen de un fichero externo y tomar todos sus atributos.	Imagen, LabelImage
Permite a la aplicación conectarse a la base de datos.	ConexionSqlite
Permite exportar los valores resultantes de cada imagen por métrica a un documento Excel	ExportarExcel
Permite importar los valores resultantes de cada imagen de un documento Excel a la aplicación, para su posterior análisis.	ImportarExcel

### 2.5.2 Patrones de diseño utilizados

Los patrones de diseño constituyen una descripción de la interacción entre las clases, con el objetivo de resolver un problema de diseño general en un determinado contexto. El uso de patrones permite la estandarización de la forma en que se realiza el diseño de manera que el código pueda ser reutilizable, y se le pueda dar mantenimiento cuando se requiera.

En este proyecto se utilizaron los siguientes patrones de diseño:

#### GRASP

Los patrones GRASP (Patrones Generales de Software para la Asignación de Responsabilidades) describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones. El nombre se eligió para sugerir la importancia de aprehender (*grasping* en inglés) estos principios para diseñar con éxito el software orientado a objetos. Es necesario elegir cuidadosamente las clases adecuadas y decidir cómo estas deben interactuar, por esta razón resulta de vital

importancia el uso de patrones GRASP. Los patrones utilizados para la asignación de responsabilidades son [90]:

**Expertos:** Este patrón se utiliza con frecuencia en la asignación de responsabilidades; es un principio de guía básico que se utiliza continuamente en el diseño de objeto [90]. El uso de este patrón se evidencia en la clase Método, la cual contiene toda la información referente a todas las métricas.

```
// todos estos metodos trabajan sobre imagenes binarias
public class Metodos {

    //*****Medidas de Localizacion*****
    // tamDeVecindad sera la cantidad de casillas que se buscara el valor (en este caso
    // de izquierda a derecha "si no se encuentra el valor la distancia no se calcula"
    public double mediaDePeliYMalah(int[][] contorno, int[][] imgAProbar) {

        double media = 0;
        int distancia = 0;
        int sumatoria = 0;
        int cardinal = 0;

        for (int i = 0; i < imgAProbar.length; i++) {
            for (int j = 0; j < imgAProbar[0].length; j++) {

                //cuento la cantidad de pixeles de valor 1 de la matriz contorno
                if (contorno[i][j] == 1) {
                    cardinal++;
                }
            }
        }
    }
}
```

**Figura 11:** Clase Método.

**Controlador:** El patrón controlador sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, manejando los eventos de entrada de dicha interfaz [91]. En la aplicación se manifiesta el uso de este patrón con la creación de varios controladores que se encargan de manejar eventos, el controlador “LabelImage” ejemplifica claramente el uso de este patrón.

```

L */
public class labelImage extends JLabel {

    public Graphics2D g2;
    public LinkedList<Point> irrFigure;
    public LinkedList<BufferedImage> historial;
    public float xRect = 0;
    public float yRect = 0;
    public float wRect = 0;
    public float hRect = 0;
    private Rectangle2D rectangulo;
    public int posHist = -1;
    private double scale = 1.0;
    private BufferedImage imgBackground;
    private String nombreImagen;
}
```

**Figura 12:** Clase LabelImage.

**Creador:** Se encarga de guiar la asignación de responsabilidades relacionadas con la creación de objetos. Su intención es encontrar un creador que necesite conectarse al objeto creado en alguna situación [92]. Dentro de la aplicación este patrón se observa en las acciones del controlador.

```
DefaultTableModel tm = new DefaultTableModel();
//son los valores de las columnas de la tabla
tm.addColumn("Algoritmo");
tm.addColumn("Precisión");
tm.addColumn("Exactitud");
tm.addColumn("Primer Plano");
tm.addColumn("Revocación");
tm.addColumn("NSR");
tm.addColumn("Tasa de Error");
tm.addColumn("Tasa Negativa");
tm.addColumn("Media");
tm.addColumn("Varianza");
tm.addColumn("FOM");
tm.addColumn("Hausdorff");
tm.addColumn("Baddeley");

jTable2.setModel(tm);
```

**Figura 13:** Creación de instancias.

**Alta Cohesión:** Propone que la información que almacena una clase debe de ser coherente y debe estar, en la medida de lo posible, relacionada con la clase. Al realizar un cambio en una clase con alta cohesión, todos los métodos que pueden verse afectados, estarán a la vista, en el mismo archivo. Incrementa la claridad, la reutilización y la facilidad de comprensión del diseño [90].

**Bajo Acoplamiento:** Este patrón expresa que entre las clases deberán existir pocas ataduras, es decir, estas estarán lo menos relacionadas posible, de forma tal que, en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases, incrementando la reutilización y disminuyendo la dependencia entre las clases [90].

## 2.6 Conclusiones del capítulo

Después de realizado el diseño de la propuesta de solución se pudieron sacar las siguientes conclusiones:

- El sistema cuenta con 17 HU para implementarse en 3 iteraciones, con una duración de 32 semanas.

- Se propone la utilización de gráficos para la representación visual de los resultados de las métricas.
- El sistema presenta gran dependencia de la clase “Métodos” que implementa los valores de VP, VN, FN y FP; siendo claves estos en la implementación de las métricas propuestas.
- Se definió el uso de los siguientes patrones de diseños: *Expertos, Controlador, Creador, Alta Cohesión y Bajo Acoplamiento*.

### **Capítulo 3. Resultados y validación del sistema.**

En el presente capítulo se describe el análisis estadístico para determinar el algoritmo más eficiente para la detección de la opacidad de la cápsula posterior en imágenes provenientes de tomogramas Scheimpflug del PENTACAM. Además, se realiza la fase de implementación y se detallan las pruebas realizadas al sistema con el fin de validar que este haya sido implementado de acuerdo a las exigencias del cliente y para garantizar la calidad del mismo.

#### **3.1 Análisis estadístico de los resultados en imágenes del PENTACAM**

Las pruebas estadísticas permiten contrastar la veracidad o falsedad de las hipótesis enunciadas desde el punto de vista estadístico. Las cuales se clasifican en dos grandes grupos de pruebas de significación estadística, el referido a las paramétricas y el relacionado con las no paramétricas, con rasgos distintivos que las caracterizan [93].

##### **Pruebas paramétricas**

Las paramétricas exigen, a los datos a los que se aplica, que se cumplan los siguientes requisitos [93]:

- Variable numérica. Que las variables de estudio estén medidas en una escala que sea por lo menos de intervalo.
- Normalidad. Que los valores de las variables sigan una distribución normal, por lo menos, en la población al que pertenece la muestra.
- Homocedasticidad. Que las varianzas de la variable dependiente en los grupos que se comparan sean aproximadamente iguales (homogeneidad de las varianzas).

Cuando los datos cumplen con los requisitos indicados, las pruebas estadísticas paramétricas exhiben su máximo poder. Cuando estas pruebas estadísticas se aplican

a datos que no cumplen al menos uno de los requisitos señalados, pierden parte de su poder. Si no se cumple una de las tres condiciones se consideran las pruebas no paramétricas, las cuales no hacen ningún tipo de suposición acerca de la forma exacta de la población en la que fueron extraídas las muestras.

### **Pruebas no paramétricas**

Los métodos no paramétricos son la manera más directa de solucionar el problema de falta de normalidad. Estos métodos son muy simples de usar y están disponibles en SPSS [94]. Es importante distinguir que las pruebas de hipótesis no paramétricas no contestan a la misma pregunta que las pruebas paramétricas. El test no paramétrico establece la hipótesis en términos de la mediana y el test paramétrico usa la media. Además existen dos grandes tipos de test no paramétricos, los que usan cuentas o números y los que usan rangos [95].

El tamaño de la muestra es de solo 28 elementos, lo cual es la principal razón que dificulta probar la normalidad de los datos, lo que indica el uso de pruebas no paramétricas. En este trabajo se revisó el test de suma de rangos de Wilcoxon [96] y el Test de Friedman [97] como las pruebas a utilizar.

### **Test de suma de rangos de Wilcoxon**

Se tiene una muestra aleatoria de tamaño  $n_1$  de una población, y una segunda muestra aleatoria de tamaño  $n_2$  de otra población. Hay  $n$  observaciones en total, donde  $n = n_1 + n_2$ . Se calcula el rango de las  $n$  observaciones. El test estadístico será la suma  $W$  de los rangos del grupo con menor suma de rangos, este será el estadístico de suma de rangos de Wilcoxon. Si las dos poblaciones tienen la misma distribución continua, entonces  $W$  tiene media:

$$\mu_W = \frac{n_1(n+1)}{2} \quad (19)$$

y desviación estándar:

$$\sigma_W = \sqrt{\frac{n_1 n_2 (n+1)}{12}} \quad (20)$$

Donde  $n_1$  será el tamaño muestral del grupo con menor suma de rangos.

El test de suma de rangos de Wilcoxon rechaza la hipótesis nula de que las dos poblaciones tienen la misma distribución cuando la suma de rangos  $W$  está lejos de su media.

### **Hipótesis para una prueba de signo o una prueba de rango con signo de Wilcoxon**

Las hipótesis que se prueban con la prueba de signo y la prueba de rango con signo de Wilcoxon son las siguientes [98] :

$H_0$ : La probabilidad de una diferencia positiva es igual a la probabilidad de una diferencia negativa.

$H_a$ : La probabilidad de una diferencia positiva no es igual a la probabilidad de una diferencia negativa.

### **Test de Friedman**

La prueba de Friedman es una alternativa no paramétrica a un análisis de varianza de medidas repetidas utilizado para comparar observaciones repetidas sobre los mismos sujetos, a diferencia de las medidas paramétricas ANOVA [98]. Además si la investigación tiene k variables en columnas y n elementos en filas se trata de ordenar cada fila de menor a mayor según las diferentes columnas de 1 hasta k (esto es el rango que ocupa cada variable para ese caso). Si no hay diferencias estadísticamente significativas entre las variables se espera que los rangos estén repartidos en cada columna de manera uniforme y sólo se encontrarán entre las variables pequeñas diferencias debidas al azar.

### **Hipótesis para la prueba de Friedman**

Las hipótesis que se prueban en la prueba de Friedman son las siguientes [98]:

$H_0$ : Las distribuciones son las mismas a través de medidas repetidas.

$H_a$ : Las distribuciones a través de medidas repetidas son diferentes.

## **3.2 Pruebas**

Uno de los pilares de la metodología XP es el proceso de pruebas [99]. Las pruebas de software son un elemento crítico para garantizar la calidad del software y representan una revisión final de las especificaciones, del diseño y de la codificación [100]. XP exhorta constantemente a realizar pruebas tanto como sea posible, permitiendo aumentar la calidad de los sistemas, reduciendo el número de errores no detectados [101]. La metodología XP divide las pruebas del software en dos grupos: pruebas unitarias, encargadas de verificar el código de manera automática y diseñada por los programadores, y pruebas de aceptación o pruebas funcionales, destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida en las historias de usuario diseñadas por el cliente final [102].

### 3.2.1 Pruebas unitarias

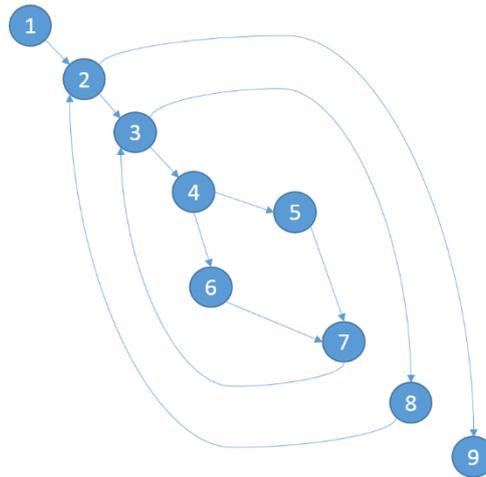
Una prueba unitaria es la verificación de una unidad de código (módulo) determinado dentro de un sistema. Estas pruebas nos aseguran que un módulo determinado cumpla con un comportamiento esperado en forma aislada antes de ser integrado al sistema, es decir, debe aprobar satisfactoriamente todos los casos de prueba definidos. El uso de estas pruebas evita tener que invertir una gran cantidad de horas en sesiones de debugging al momento de integrar el código con el sistema existente. En la metodología XP los programadores deben definir las pruebas unitarias para cada módulo antes de escribir el código, y sólo para aquellas unidades de código donde exista la posibilidad de que puedan fallar es necesario escribir casos de prueba [103].

A continuación, se describirá la confección de las pruebas unitarias realizadas, desglosándola en sus elementos fundamentales: confección del grafo de flujo, cálculo de la complejidad ciclomática, extracción de los caminos independientes, realización de los casos de pruebas y análisis de los resultados obtenidos. A uno de los métodos que se le aplicaron estas pruebas fue al método “repeticionesIntensidad“, el cual nos permite tal cual su nombre indica, contar la cantidad de repeticiones de cierto valor de intensidad. A este se le aplica la métrica de complejidad ciclomática debido a su relevancia para el sistema.

```
public int[][] repeticionesIntensidad(BufferedImage bi,Color color){
    WritableRaster raster = bi.getRaster();
    //Lleno la imagenMatriz
    double h = bi.getWidth();
    double w = bi.getHeight();
    int[][]imagenMatriz=new int[(int) w][(int) h];
    for (int i = 0; i < (int) w; i++) {
        for (int j = 0; j < (int) h; j++) {
            //formato de colores del 0 - 255.
            if (raster.getSample((int) (j), (int) (i), 0)==color.getRed() && raster.getSample((int)
                imagenMatriz[i][j]= 1;
            }
            else{
                imagenMatriz[i][j]= 0;
            }
        }
    }
    return imagenMatriz;
}
```

**Figura 14:** Algoritmo para repeticiones.

Partiendo del fragmento de código mostrado en la Figura 14 se obtuvo el grafo de flujo que muestra la Figura 15.



**Figura 15:** Diagrama de flujo del algoritmo “repeticionesIntensidad”.

Tras definir el grafo de flujo se procede al cálculo de la complejidad ciclomática siendo esta una métrica del software que proporciona una medición cuantitativa de la complejidad lógica de un programa [104]. Para esta operación existen tres vías de solución, las cuales se enuncian a continuación

$$V(G)=(A-N) +2 \quad (21)$$

$$V(G)=P +1 \quad (22)$$

$$V(G)=R \quad (23)$$

La complejidad ciclomática,  $V(G)$ , de un grafo de flujo  $G$ , se define como:

$V(G) = A - N + 2$ , donde  $A$  es el número de aristas del grafo de flujo y  $N$  es el número de nodos del mismo.

La complejidad ciclomática,  $V(G)$ , de un grafo de flujo  $G$ , también se define como:

$V(G) = P + 1$ , donde  $P$  es el número de nodos predicados contenidos en el grafo de flujo  $G$ .

$V(G) = R$ , donde  $R$  es la cantidad total de regiones.

Del grafo de flujo  $G$  se obtiene que:

$$V(G) = (A - N) + 2 = (11 - 9) + 2 = 4$$

$$V(G) = P + 1 = 3 + 1 = 4$$

$$V(G) = R = 4$$

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

Tras aplicar las tres formas para afirmar un resultado estable, se obtiene una complejidad ciclomática  $V(G)= 4$  la cual representa la cantidad de caminos independientes para el grafo de flujo construido para el método “repeticionesIntensidad”, por lo que se definen los 4 caminos que muestra la Tabla 6.

*Tabla 6: Caminos básicos*

Número	Caminos básicos
1	1-2-9
2	1-2-3-8-2-9
3	1-2-3-4-5-7-3-8-2-9
4	1-2-3-4-6-7-3-8-2-9

Luego de tener elaborado el grafo de flujo e identificados los caminos a recorrer, se preparan los casos de prueba que forzarán la ejecución de cada uno de esos caminos. Con el fin de comprobar cada camino, se escogen los datos de manera que las condiciones de los nodos predicado estén adecuadamente establecidas. A continuación, se especifica uno de estos casos de prueba en la Tabla 7.

*Tabla 7: Caso de prueba del camino básico #1*

Caso de Prueba: Camino Básico # 1
Descripción: Este método dada una imagen, devuelve las repeticiones de un valor de intensidad específico por cada canal de color.
Condiciones de ejecución: Se debe haber cargado la imagen antes.
Entradas\Pasos de ejecución: La imagen inicial y el objeto color.
Resultado esperado: La matriz de la imagen con valores 0 y 1, y el resultado del número de coincidencias.
Evaluación de la prueba: Satisfactoria.

Después de realizadas las pruebas unitarias a los principales métodos de la aplicación tal como se mostró en el método de ejemplo, se comprobó que el diseño del software se correspondiera con lo programado o sea que se programó correctamente lo diseñado. Estas pruebas por sí solas no aseguran que las respuestas emitidas por el software, dada una entrada de datos por el usuario, sean las esperadas por los clientes, siendo necesarias las Pruebas de Aceptación.

### 3.2.2 Pruebas aceptación

Cada prueba de aceptación o prueba funcional como también se le conoce está asociada a una funcionalidad del sistema. Estas pruebas se realizan en esta etapa del proyecto describiéndose en ellas las posibles formas de utilización del software. Estas pruebas funcionales no sólo validan la transformación de una entrada en una salida, sino que validan una característica completa [105]. En estas pruebas se indican las posibles respuestas que da el software tras la utilización de cada funcionalidad, así como los posibles mensajes de error, información, o de aceptación que emita cuando se utiliza dicha funcionalidad [106].

Para la descripción de estas pruebas se presenta el diseño del caso de prueba correspondiente a la historia de usuario “Determinar mejor segmentación” que incluye funcionalidades críticas para el sistema. La siguiente tabla muestra los resultados satisfactorios obtenidos en la prueba de aceptación “Determinar mejor segmentación”.

Tabla 8: Caso de prueba de aceptación de la HU #14

Caso de prueba de aceptación	
<b>Código:</b> HU7_P14	<b>Historia de Usuario:</b> 14
<b>Nombre:</b> “Determinar mejor segmentación”	
<b>Descripción:</b> Prueba para la funcionalidad “Determinar mejor segmentación”.	
<b>Condiciones de Ejecución:</b> Se debe tener al menos el resultado de las métricas de dos algoritmos de segmentación.	
<b>Resultado Esperado:</b> El sistema devuelve el nombre de la mejor segmentación.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

### 3.2.3 Análisis de los resultados en las imágenes del PENTACAM

Luego de efectuar las pruebas correspondientes para validar la fiabilidad del sistema, se procedió a realizar la comparación de las métricas de error y los tiempos de ejecución entre los tres algoritmos de segmentación propuestos, con el fin de seleccionar el algoritmo más eficiente para la detección de la opacidad de la cápsula posterior en imágenes provenientes de tomogramas Scheimpflug del PENTACAM. Estas comparaciones fueron ejecutadas sobre un conjunto de 28 imágenes de pacientes con esta complicación postoperatoria por cada uno de los algoritmos propuestos. Para la

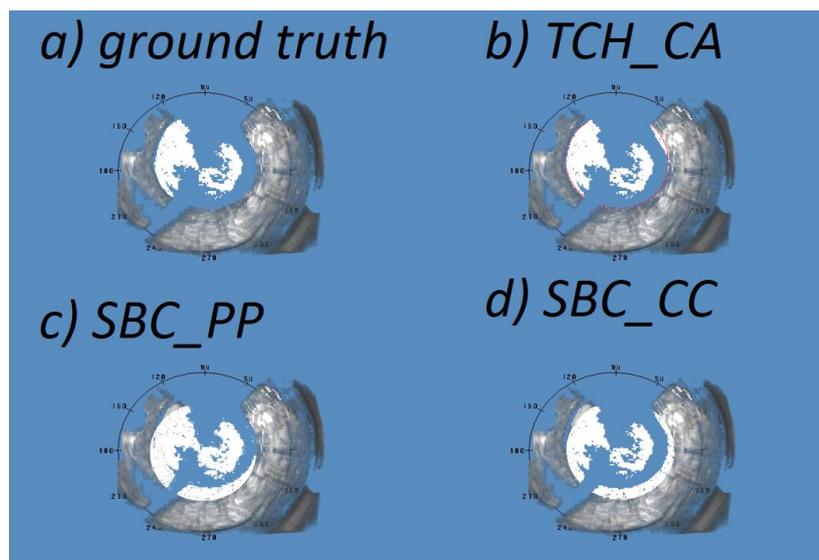
## *Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

validación se usaron las métricas descritas en la sección 1.3.3, estas necesitan un Ground Truth plantilla proporcionada por los especialistas.

Para esta labor se confeccionó un equipo de 3 especialistas del servicio de cataratas del ICO con amplia experiencia en el trabajo con la OCP:

- Dr. Iván Hernández López, Especialista de I grado en Oftalmología, Investigador Auxiliar.
- Dr. Eneida Pérez Candelaria, Especialista de II grado en Oftalmología, Profesor Auxiliar, Investigador Auxiliar.
- Dr. Zucell Veitía Roviroso, Especialista de II grado en Oftalmología, Profesor Auxiliar, Investigador Auxiliar.

Estos se reunieron y marcaron en consenso en las 28 imágenes las regiones que consideraron opacidad. Estas imágenes (Ground Truth) fueron usadas en la validación de la segmentación por los algoritmos propuestos como se muestra en la Figura 16.



**Figura 16:** *Imágenes del Pentacam segmentadas por los algoritmos propuestos (b,c,d) y la imagen de referencia marcada por los especialistas (a).*

En la Figura 16 se puede apreciar que la imagen más semejante al Ground Truth es la imagen resultante del algoritmo TCH\_CA, y que las propuestas SBC\_PP y SBC\_CC identifican como OCP regiones propias de la estructura ocular. Se pudo apreciar lo siguiente:

- En esta imagen el algoritmo TCH\_CA identificó 349 píxeles erróneos como parte de la OCP y dejó de identificar 159 que si eran OCP.

## *Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

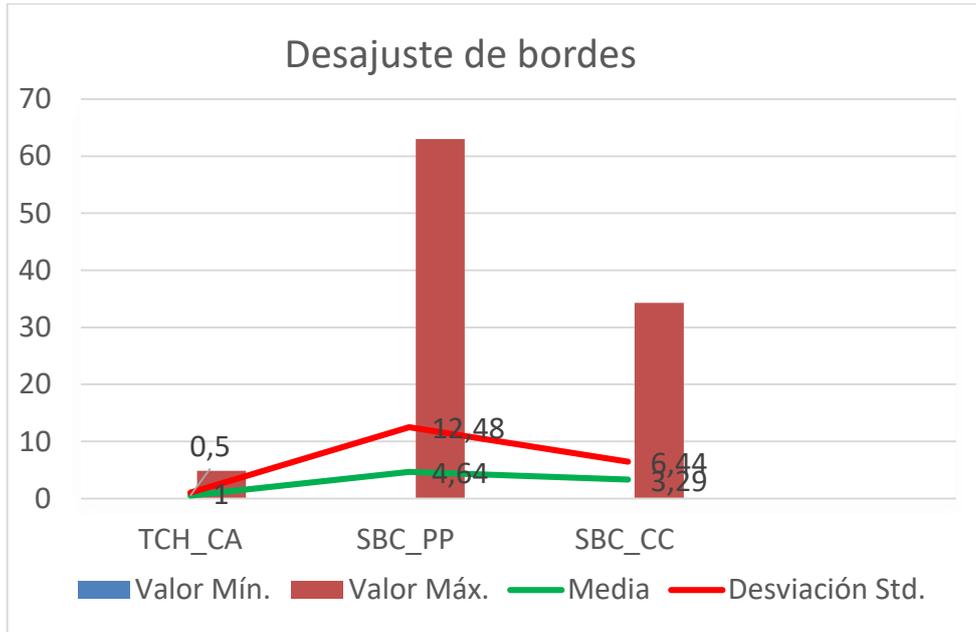
- El algoritmo SBC\_PP identificó 4586 píxeles erróneos como parte de la OCP y dejó de identificar 504 que sí eran OCP.
- El algoritmo SBC\_CC identificó 3008 píxeles erróneos como parte de la OCP y dejó de identificar 638 que sí eran OCP.

Estos resultados son de un total de 282 078 píxeles a clasificar provenientes de una imagen del PENTACAM. En estos valores, se aprecian mejores resultados con el algoritmo TCH\_CA en la identificación de la OCP. Para determinar con mayor exactitud la fidelidad de las segmentaciones, es necesario calcular varias métricas del error (1.3.3).

A continuación, se analizan los resultados (Mínimo Valor, Máximo Valor, Media y Desviación Típica) de las métricas (Clasificación del Error, Desajuste de Bordes, Índice Rand Ajustado, Índice de Jaccard, Índice de Czekanowski, Tasa de Correctos Equilibrada, Índice de Sokal-Sneath, Índice de Rogers-Tanimoto, Índice de Yule, Índice de Russell-Rao, Índice de Fowlkes-Mallows y Modificación de la Distancia de Hausdorff Normalizada) aplicadas a las segmentaciones resultantes de los algoritmos propuestos, así como los tiempos de ejecución de cada uno de ellos (Tabla 9).

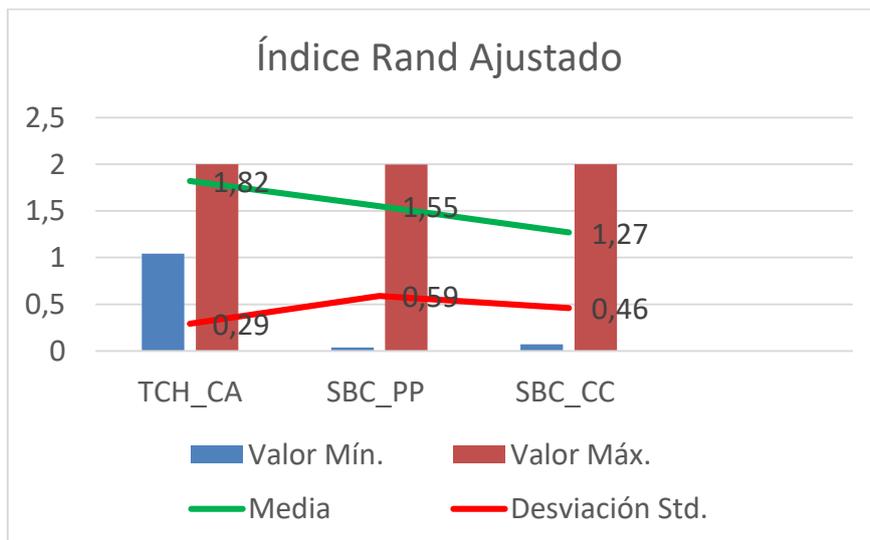
**Desajuste de Bordes** (ecuación 6) es otra de las métricas calculadas. Penaliza las discrepancias entre el mapa de borde de la imagen de nivel gris y el mapa de borde obtenido de la imagen de umbral. Mientras más cercano esté su valor a 1, menor cantidad de elementos del fondo fueron identificados como parte de la enfermedad. Los resultados de la misma sobre los algoritmos son mostrados en la Figura 17, donde se puede apreciar la superioridad en los valores del algoritmo TCH\_CA en comparación con los otros. Los valores de las desviaciones de los algoritmos basados en casos también son mayores, lo cual denota una mayor variación entre los valores de las medias.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*



**Figura 17:** Resultados de la métrica Desajuste de Bordes en los 3 algoritmos.

**Índice Rand Ajustado** (ecuación 7) es otra de las métricas calculadas. Esta tomará valor 2 cuando lo segmentado por el algoritmo coincida completamente con la OCP marcada por los especialistas. Los resultados de esta métrica se pueden apreciar en la Figura 18. En el caso de TCH\_CA es el mejor algoritmo, marcando una media de 1.82 a diferencia de los basados en casos que son los peores, con un valor medio de 1.26.

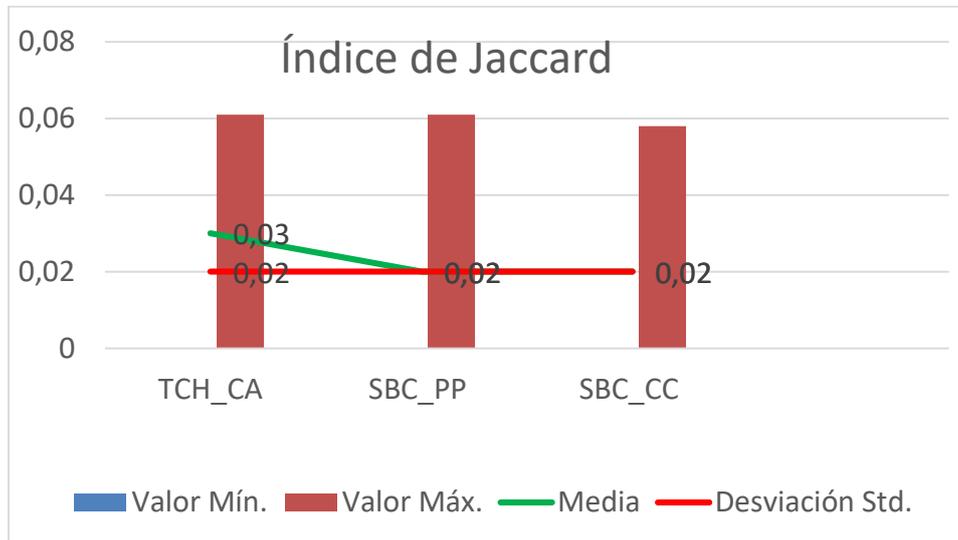


**Figura 18:** Resultados de la métrica Índice Rand Ajustado en los 3 algoritmos.

**Índice de Jaccard** (ecuación 8 y 9) Los resultados de **Índice de Jaccard** se muestran en la Figura 19 donde los valores de las medias se comportan de manera muy similar,

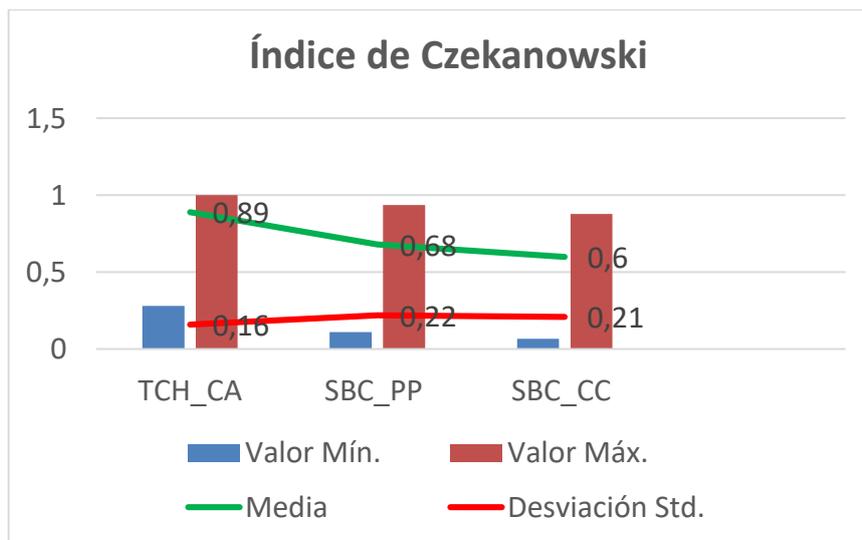
*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

de igual forma sucede con los valores máximos y desviaciones, así como los valores mínimos que toman valor ceros. Los resultados de la misma sobre los algoritmos donde se puede apreciar la mínima superioridad en los valores del algoritmo TCH\_CA en comparación con los otros.



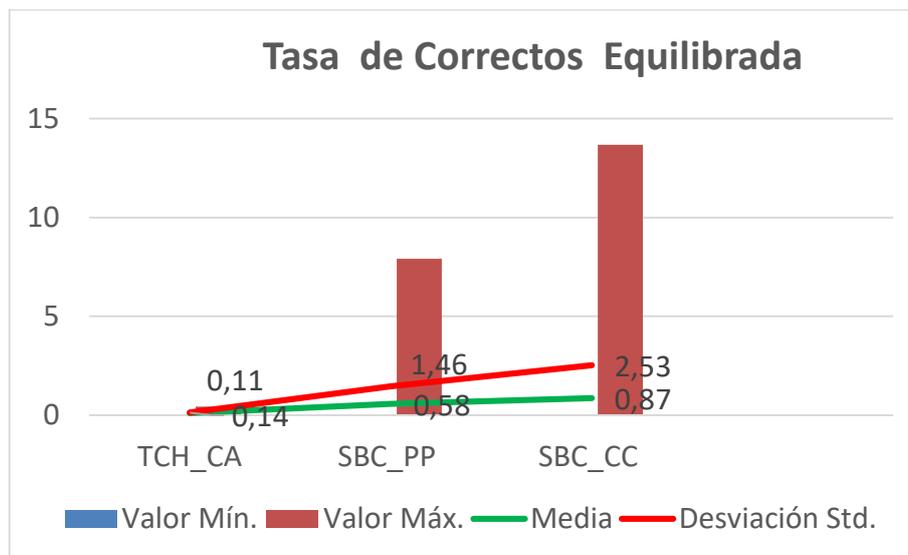
**Figura 19:** Resultados de la métrica Índice Jaccard en los 3 algoritmos

**Índice de Czekanowski** (ecuación 10) Los resultados de esta métrica se pueden ver en la Figura 20. El algoritmo TCH\_CA fue el de mejores resultados medios y menor desviación en comparación con los basados en casos (SBC\_PP y SBC\_CC). Las desviaciones tuvieron valores similares y por encima de 0.16 indicando una variación alta en los resultados.



**Figura 20:** Resultados de la métrica Índice de Czekanowski en los 3 algoritmos.

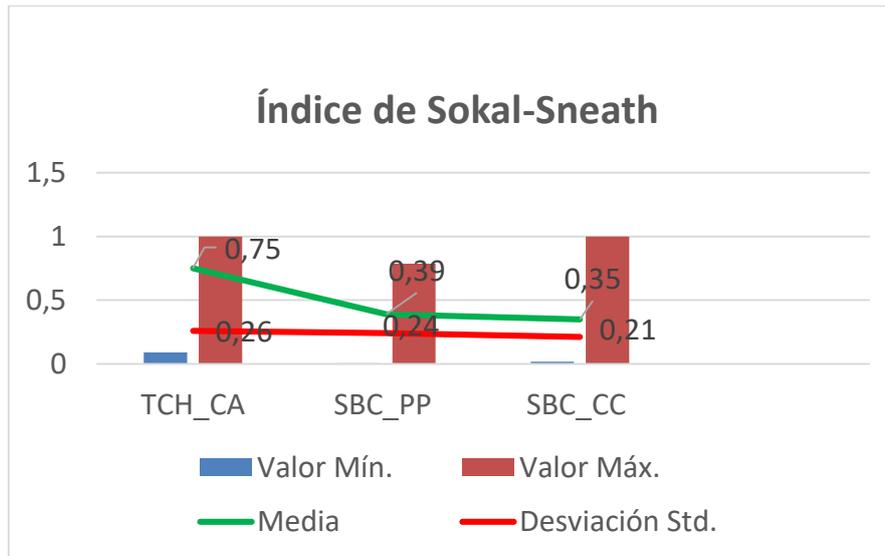
**Tasa de Correctos Equilibrada** (ecuación 11) esta tomará valor 1 cuando lo segmentado por el algoritmo coincida completamente con la OCP marcada por los especialistas. En la Figura 21 se pueden apreciar los resultados de esta medida sobre los algoritmos propuestos. En el caso de TCH\_CA tiene una marcada superioridad en esta medida marcando una media de 0.11 y desviación 0.14 a diferencia de los basados en casos que solo alcanzan un valor medio de 2.57.



**Figura 21:** Resultados de la métrica Tasa de Correctos Equilibrada en los 3 algoritmos.

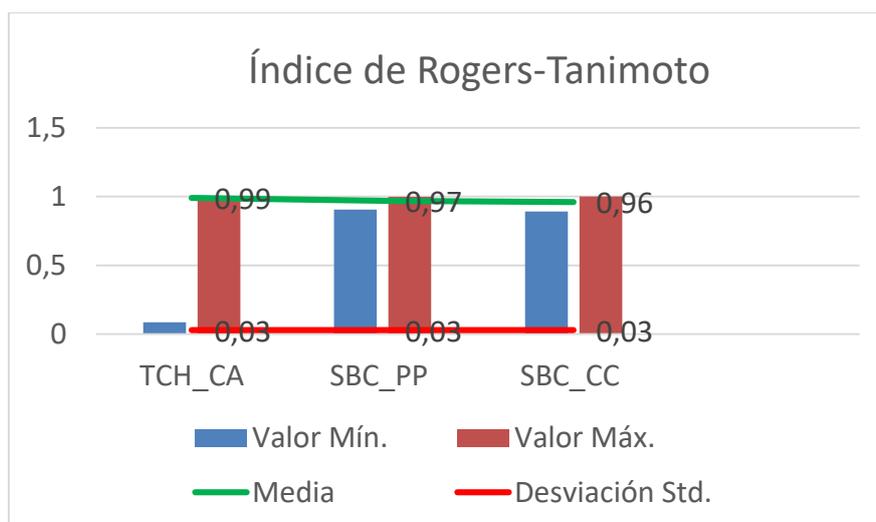
**Índice de Sokal-Sneath** (ecuación 12) es una de las medidas estadísticas del error tomada en cuenta. Esta penaliza las regiones que fueron mal identificadas (FN y FP). Cuando toma valor 1, significa que todas las regiones con opacidad fueron correctamente segmentadas. En la Figura 22 se pueden apreciar los resultados de esta medida sobre los algoritmos propuestos. En el caso de TCH\_CA tiene una marcada superioridad en esta medida marcando una media de 0.75 a diferencia de los basados en casos que solo alcanzan un valor medio de 0.35.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*



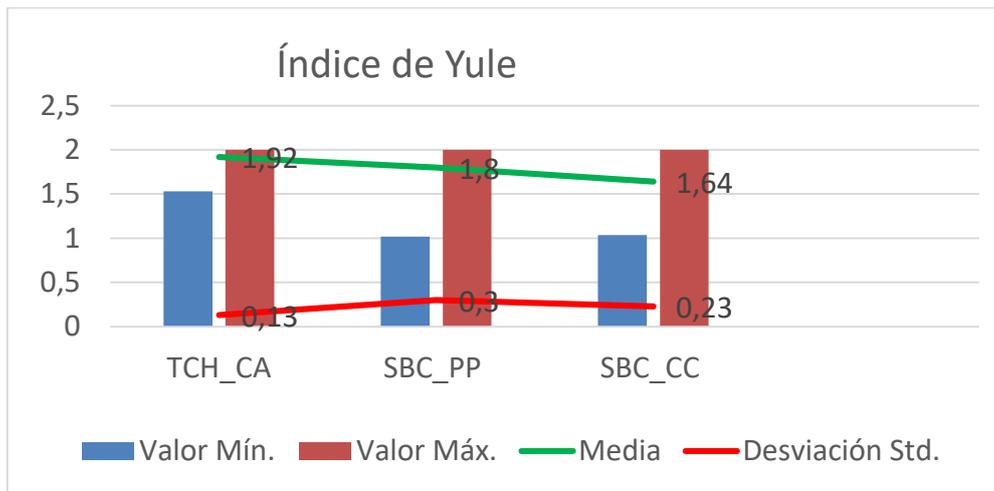
**Figura 22:** Resultados de la métrica del Índice de Sokal-Sneath en los 3 algoritmos.

**Índice de Rogers-Tanimoto** (ecuación 13) es otra de las métricas calculadas. Esta tomará valor 1 cuando lo segmentado por el algoritmo coincida completamente con la OCP marcada por los especialistas. Los resultados de esta métrica (ver Figura 23) son bastante parecidos, aunque sobresale que la media del algoritmo TCH\_CA es aproximadamente 1, lo que representa un alto nivel de coincidencia entre las imágenes de plantilla y las resultantes del algoritmo, aunque los resultados de esta métrica de los algoritmos basados en casos también son elevados obteniendo un valor medio de 0.97 y 0.96 para los algoritmos SBC\_PP y SBC\_CA respectivamente.



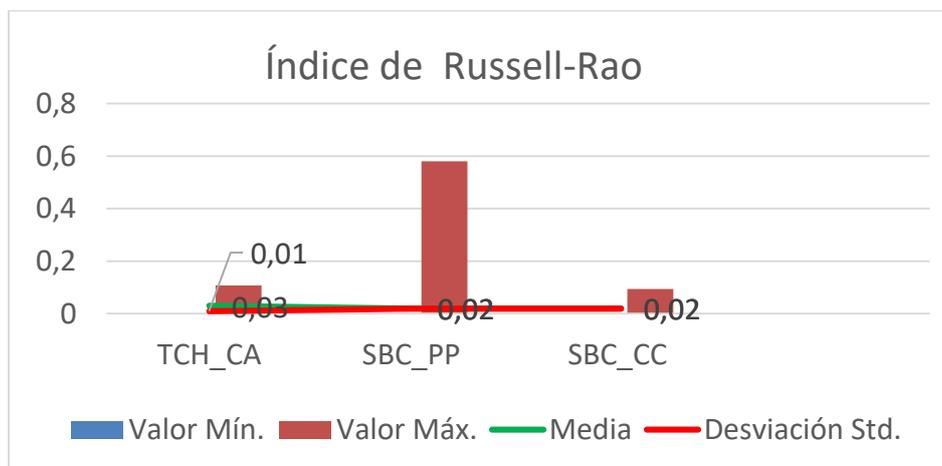
**Figura 23:** Resultados de la métrica del Índice de Índice de Rogers-Tanimoto en los 3 algoritmos.

**Índice de Yule** (ecuación 14) cuando toma valor 2, significa que todas las regiones con opacidad fueron correctamente segmentadas. En la Figura 24 se pueden apreciar los resultados de esta medida sobre los algoritmos propuestos (TCH\_CA, SBC\_PP y SBC\_CA). En el caso de TCH\_CA tiene una marcada superioridad en esta medida marcando una media de 1.92 a diferencia de los basados en casos que solo alcanzan un valor medio de 1.64.



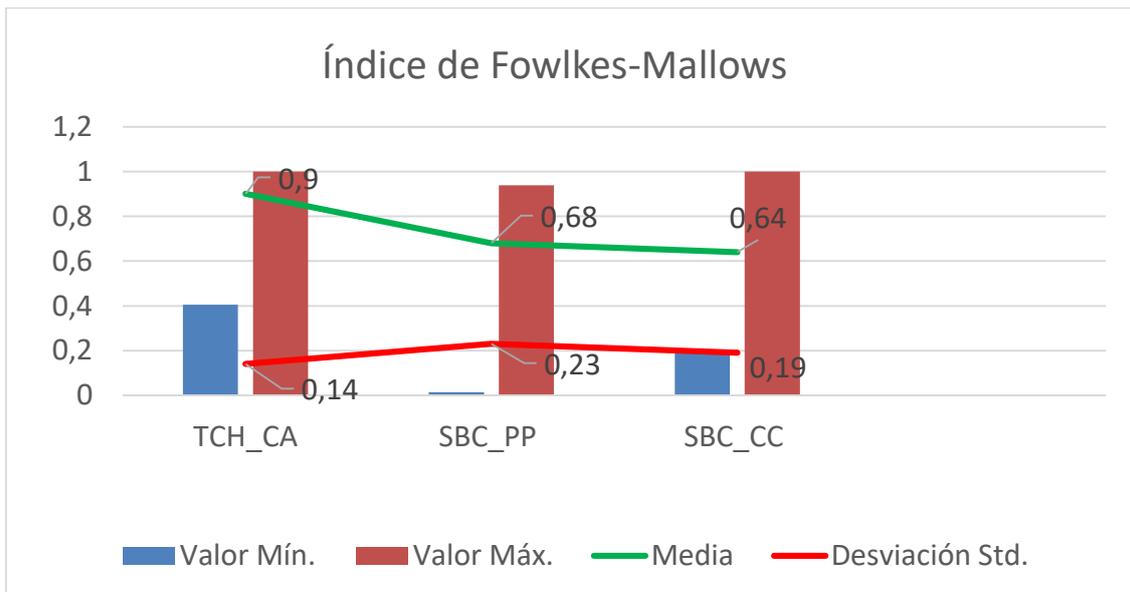
**Figura 24:** Resultados de la métrica del Índice de Yule en los 3 algoritmos.

**Índice de Russell-Rao** (ecuación 15) En el caso de esta métrica los resultados de esta medida son mostrados en la Figura 25. Obtuvo mejores resultados el algoritmo TCH\_CA debido a que mientras más pequeño sea el valor resultante de esta métrica se obtiene peor resultado.



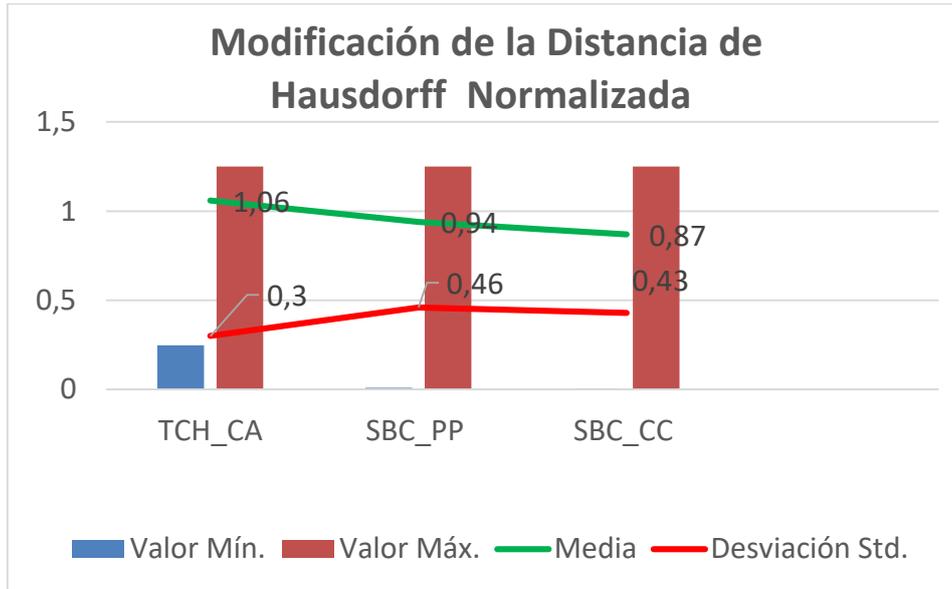
**Figura 25:** Resultados de la métrica del Índice de Índice de Russell-Rao en los 3 algoritmos.

**Índice de Fowlkes-Mallows** (ecuación 16) Esta tomará valor 1 cuando lo segmentado por el algoritmo coincida completamente con la OCP marcada por los especialistas. Los resultados de esta métrica (ver Figura 26) la media del algoritmo TCH\_CA es aproximadamente 57.12 significando esto un alto nivel de coincidencia entre las imágenes de plantilla y las resultantes del algoritmo. Aunque los resultados del algoritmo SBC\_PP también son elevados obteniendo un valor medio de 43.55 y menor valor en la desviación 23.27, siendo el algoritmo SBC\_CC de peor resultado.



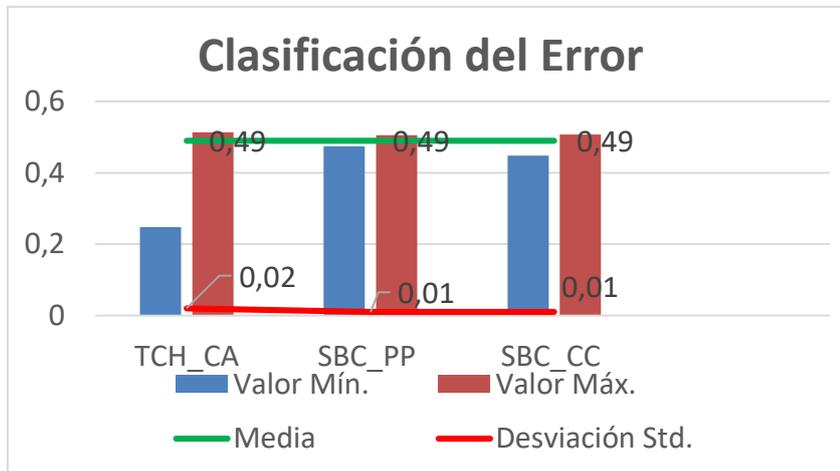
**Figura 26:** Resultados de la métrica del Índice de Fowlkes-Mallows en los 3 algoritmos.

**Modificación de la Distancia de Hausdorff Normalizada** (ecuación 17) esta métrica que devuelven valores de distancia por las diferencias existentes entre la segmentación de los algoritmos y el Ground Truth significando esto que el valor sea 0 en caso de una segmentación perfecta, y mientras mayor sea el valor de la misma, mayor la cantidad de diferencias encontradas.



**Figura 27:** Resultados de la métrica Modificación de la Distancia de Hausdorff Normalizada en los 3 algoritmos.

**Clasificación del Error** (ecuación 5) Los valores de las medias y desviaciones de los tres algoritmos sobre esta métrica se pueden observar en la Figura 28. Los resultados de la media de los tres algoritmos es el mismo.



**Figura 28:** Resultados de la métrica Clasificación del Error en los 3 algoritmos.

Se analizaron distintas medidas estadísticas del error (Clasificación del Error, Desajuste de Borde, Índice de Rand Ajustado, Índice de Czekanowski, Tasa de Correctos Equilibrado, Índice de Sokal-Sneath, Índice de Rogers-Tonimoto, Índice de Yule, Índice de Russell-Rao, Índice de Fowlkes-Mallows y Modificación de la Distancia de Hausdorff Normalizada). En las 12 medidas de calidad de la segmentación calculadas el algoritmo TCH-CA fue el de mejores resultados medios y menor desviación en comparación con

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

los basados en casos (SBC-PP y SBC-CC) esto se puede apreciar mejor en la Tabla 7, donde se muestran los resultados arrojados por cada una de las métricas calculadas y la desviación típica.

Métrica	Mejor Resultado	TCH_CA		SBC_PP		SBC_CC	
		Media	Desv.	Media	Desv.	Media	Desv.
//////////	//////////						
Clasificación del Error	0	0.49	0.022	0.49	0.01	0.49	0.01
Desajuste de Bordes	0	0.5	1	4.64	12.48	3.29	6.44
Índice Rand Ajustado	2	1.82	0.29	1.55	0.59	1.27	0.46
Índice de Jaccard	1	0.03	0.02	0.02	0.02	0.02	0.02
Índice de Czekanowski	1	0.89	0.16	0.68	0.22	0.60	0.21
Tasa de Correctos Equilibrada	0	0.11	0.14	0.58	1.46	0.87	2.53
Índice de Sokal-Sneath	1	0.75	0.26	0.42	0.23	0.32	0.21
Índice de Rogers-Tanimoto	1	0.99	0.03	0.97	0.03	0.96	0.03
Índice de Yule	2	1.92	0.13	1.80	0.30	1.64	0.23
Índice de Russell-Rao	1/VN	0.03	0.03	0.02	0.02	0.02	0.02
Índice de Fowlkes-Mallows	1	0.9	0.14	0.68	0.23	0.64	0.19
Modificación de la Distancia de Hausdorff Normalizada	1.25	1.06	0.3	0.94	0.46	0.87	2.43

**Tabla 9: Valores medios y desviaciones por cada métrica (mejor resultado marcado en verde y peor en rojo).**

Estos resultados deben estar relacionados con el enfoque de los algoritmos, debido a que TCH\_CA busca los bordes de la capsulorrexis para encontrar la OCP. En cambio, los otros suelen “confundir” estos elementos como parte de la OCP.

### 3.2.4 Análisis Estadístico

Con el objetivo de comparar el desempeño de los algoritmos y de corroborar lo expresado anteriormente se procede a realizar un análisis estadístico. Para esto se

selecciona el uso de pruebas no paramétricas debido al tamaño de la muestra (solo 28 imágenes). Se escoge usar el análisis de dos vías de Friedman y como prueba *post-hoc* la prueba de rango de Wilcoxon por pares, ajustándose el nivel de significancia con una corrección de Bonferroni. Esto se aplicó sobre la variable Índice de Yule como medida estadística del error cometido en la segmentación, sobre la variable Desajuste de Bordes como medida de localización y sobre Modificación de la Distancia de Hausdorff Normalizada como una variante de las métricas de Hausdorff. Para realizar todos los cálculos se empleó el software IBM SPSS Statistics 21.

**Tabla 11:** Test de Friedman sobre los resultados de las variables Índice de Yule, Desajuste de Bordes y Modificación de la Distancia de Hausdorff Normalizada.

#	Hipótesis Nula	Test	Sig. (p)	Decisión	Variable
1	Las distribuciones de los algoritmos son las mismas.	Friedman	< 0.000	Rechazar la hipótesis nula.	Índice de Yule
2	Las distribuciones de los algoritmos son las mismas.	Friedman	< 0.000	Rechazar la hipótesis nula.	Desajuste de Bordes
3	Las distribuciones de los algoritmos son las mismas.	Friedman	< 0.000	Rechazar la hipótesis nula.	Modificación de la Distancia de Hausdorff Normalizada

Como se muestra en la Tabla 11 el valor de precisión (p) es menor que 0.05, por lo que se puede confirmar que existen diferencias significativas (en el Índice de Yule, Desajuste de Bordes e Índice de Sokal-Sneath) entre los grupos de algoritmos analizados. Es decir, los resultados de los algoritmos en las variables analizadas cambiaron significativamente.

Se desarrolló entonces una prueba *post-hoc* de rangos con signos de Wilcoxon con el objetivo de comparar dos a dos las muestras relacionadas (algoritmos de segmentación) y determinar entre qué resultados se presentaron las diferencias significativas. Se usó la corrección de Bonferroni para evitar cometer un error de Tipo I (rechazar la hipótesis nula cuando debería aceptarse). El nuevo nivel de significación que se tomó por tanto fue de 0,016 ( $0.05/3=0.0166$ ).

Cuando se analiza los resultados de los algoritmos probados en la investigación, se puede observar en las Tablas 12, 13 y 14 que existen diferencias significativas entre los

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

algoritmos en las variables estudiadas (en el Índice de Yule, Desajuste de Bordes e Índice de Sokal-Sneath), rechazando en todos los casos la hipótesis nula. Dado este resultado se toma la media como medida para evaluar el rendimiento del algoritmo en las métricas aplicadas. Por tanto, se asume que el algoritmo de segmentación de OCP que mejores resultados brinda, dado los valores de las medias, es TCH\_CA.

**Tabla 12:** Prueba por pares de Wilcoxon sobre los resultados de la variable en el Índice de Yule

#	Hipótesis Nula	Test	Sig. (p)	Decisión	Variable
1	La mediana de las diferencias de entre los algoritmos TCH_CA y SBC_CC es igual a 0.	Wilcoxon	.000	Rechazar la hipótesis nula.	en el Índice de Yule
2	La mediana de las diferencias entre los algoritmos TCH_CA y SBC_PP es igual a 0.	Wilcoxon	.002	Rechazar la hipótesis nula.	en el Índice de Yule
3	La mediana de las diferencias de entre los algoritmos SBC_CC y SBC_PP es igual a 0.	Wilcoxon	.001	Rechazar la hipótesis nula.	en el Índice de Yule

**Tabla 13:** Prueba por pares de Wilcoxon sobre los resultados de la variable Desajuste de Bordes.

#	Hipótesis Nula	Test	Sig. (p)	Decisión	Variable
1	La mediana de las diferencias de entre los algoritmos SBC_CC y SBC_PP es igual a 0.	Wilcoxon	.002	Rechazar la hipótesis nula.	Desajuste de Bordes
2	La mediana de las diferencias de entre los algoritmos TCH_CA y SBC_CC es igual a 0.	Wilcoxon	.000	Rechazar la hipótesis nula.	Desajuste de Bordes
3	La mediana de las diferencias de entre los algoritmos TCH_CA y SBC_PP es igual a 0.	Wilcoxon	.000	Rechazar la hipótesis nula.	Desajuste de Bordes

**Tabla 14:** Prueba por pares de Wilcoxon sobre los resultados de la variable Índice de Sokal-Sneath.

#	Hipótesis Nula	Test	Sig. (p)	Decisión	Variable
---	----------------	------	----------	----------	----------

1	La mediana de las diferencias de entre los algoritmos TCH_CA y SBC_CC es igual a 0.	Wilcoxon	.000	Rechazar la hipótesis nula.	Índice de Sokal-Sneath
2	La mediana de las diferencias de entre los algoritmos TCH_CA y SBC_PP es igual a 0.	Wilcoxon	.000	Rechazar la hipótesis nula.	Índice de Sokal-Sneath
3	La mediana de las diferencias de entre los algoritmos SBC_CC y SBC_PP es igual a 0.	Wilcoxon	.001	Rechazar la hipótesis nula.	Índice de Sokal-Sneath

### **3.3 Conclusiones parciales**

Durante el desarrollo del presente capítulo se abordaron los temas referentes a las fases de implementación y pruebas al sistema. Se realizaron las pruebas de aceptación y las pruebas unitarias, presentándose los casos de prueba correspondientes para validar las funcionalidades del software propuesto. Los resultados fueron satisfactorios, lo que significó que las historias de usuarios implementadas cumplen con los requisitos y necesidades del cliente. Con la validación realizada a la solución desarrollada se pudo efectuar la comparación entre los tres algoritmos de segmentación propuestos por el grupo AIRI. Se calcularon medidas de calidad de la segmentación, y se comprobó la existencia de diferencias estadísticamente significativas entre las muestras.

## **CONCLUSIONES GENERALES**

Con el desarrollo de este trabajo se cumplieron todas las actividades propuestas y se respondió a necesidad de buscar una solución al problema general planteado, arribando a las siguientes conclusiones:

- Se implementaron un total de doce métricas al software PASI, de las cuales nueve son estadísticas de error, dos pertenecen a medidas de localización y una métrica Hausdorff permitiendo obtener un análisis más amplio de los resultados de la validación.
- Mediante las pruebas realizadas a un conjunto de 28 imágenes de tomogramas Scheimpflug del PENTACAM se pudo obtener que el algoritmo más estable para la detección de la Opacidad de la Cápsula posterior es el propuesto por Paula Vidal 2016 atendiendo a que brindó los mejores resultados en cuanto a métricas estadísticas de error, medidas de localización y métricas Hausdorff.
- Dados los resultados satisfactorios de las pruebas de software aplicadas, se puede afirmar que se tiene un software en concordancia con lo diseñado y que funciona correctamente.

## **RECOMENDACIONES**

Se recomienda para versiones futuras:

- Implementar otras métricas para la validación de la segmentación de imágenes de manera general.
- Incluir la validación estadística.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] O. M. d. I. Salud., «Prevención de la ceguera,» 1972.
- [2] A. M. Medina Aguaguña, «Actores de riesgo de catarata en el adulto mayor, atendidos en el Hospital Alfredo Noboa Montenegro período enero-septiembre 2015,» 2016.
- [3] A. R y e. al, Revisión sistemática de estudios poblacionales de prevalencia de catarata. Archivos de la Sociedad Española de oftalmología, vol. 81, Ingeniería Mecánica, 2006, pp. 509-516.
- [4] I. Hernández López y e. al., «Estrategias de prevención de la opacidad de la cápsula posterior,» *Revista Cubana de Oftalmología*, vol. 23, pp. 608-623, 2010.
- [5] I. Hernández López y e. al, «Opacidad de la cápsula posterior.,» de *Oftalmología. Criterios y tendencias actuales.*, La Habana, Ciencias Médicas, 2009, pp. 307-317.
- [6] I. Hernández López y e. al, «Estrategias de prevención de la opacidad de la cápsula posterior,» vol. 23, 2010.
- [7] D. Grewal y e. al, «Pentacam tomograms: A Novel Method for Quantification of Posterior Capsule Opacification.,» *IOVS*, vol. 49, nº 5, 2008.
- [8] M. Cancio Alvarez, R. Cruz de la Osa y I. López Hernández, «PANDOC:Software para la Cuantificación Objetiva,» Junio del 2015.
- [9] M. Moreno Rodríguez, «Desarrollo de un sistema basado en casos para la identificación de opacidad de la cápsula posterior en el software PANDOC,» La Habana, Junio del 2015.
- [10] M. I Alvarez Cancio, R. Cruz de la Osa y I. Hernández López, «Informática2016,» 2016.
- [11] C. A. Alonso Peña, P. L. Machado Sanchez y M. Alvarez Cancio, «Sistema de cuantificación de opacidad capsular mediante Componentes Conexas,» Facultad 3, UCI. La Habana, 2016.

- [12] C. A. Alonso Peña, P. L. Machado Sanchez y M. Alvarez Cancio, «Sistema de cuantificación de opacidad capsular mediante componentes conexas,» Facultad 3, UCI. La Habana, 2016.
- [13] E. Paula Vida, M. Álvarez Cancio y R. Cruz de la Osa, «Algoritmo para la identificación de la opacidad de la cápsula posterior en imágenes provenientes del PENTACAM,» Junio ,2016.
- [14] M. Alvarez Cancio, «Algoritmo para la identificación de la opacidad de la cápsula posterior en imágenes provenientes del PENTACAM,» *Revista Cubana de Ciencias Informáticas*, vol. 11, nº págs. 153-167. ISSN:2227., Enero-Marzo de 2017.
- [15] R. H. Rodríguez Espinosa, «Módulo para la validación de la segmentación en imágenes de opacidad de la cápsula posterior provenientes del Pentacam,» La Habana. Junio, 2017..
- [16] G. F. Franklin y J. D. Powell, "Digital Image Processing." Adison Welsey, 1977, p. 770.
- [17] R. C. Gonzalez y R. E. Woods, *Digital Image Processing Second Edition.*, 2002.
- [18] R. Ruz Cepero y C. Rodriguez Cabrera, «Implementación de algoritmos de reducción de ruido en las imágenes,» Ciudad de La Habana, Mayo 2010.
- [19] C. Ernesto, *Métodos de Segmentación de Imágenes Médicas*, Caracas,Venezuela, Marzo 2003, pp. 1316-6239.
- [20] M. Llanes Reyes, «Avances en el diagnóstico por imágenes. A step forward in imaging diagnosis,» *Revista de Ciencias Médicas de Pinar del Río*, vol. 4, 2000.
- [21] N. d. L. S. Palomino, «Revista de Ingeniería de Sistemas e Informática,» vol. 6, nº 2, pp. 9-16, Julio - Diciembre 2009.
- [22] M. C. Elizondo y J. J. Esqueda, *Fundamentos de Procesamiento de Imágenes Digitales*, California: CONATEC, 2002.
- [23] J. Rosell Aldana, «Algoritmo para la detección y realce de bordes en imágenes oblicuas provenientes de las lámparas de hendidura.,» La Habana, Junio, 2016.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

- [24] J. P. Urrea y O. Emmanuel, Implementación de la Transformada de Hough para la detección de líneas para un sistema de visión de bajo nivel, 2004..
- [25] H. A. S. ROJO, «Escuela de Ingeniería Eléctrica».
- [26] T. Hermosilla y e. al., Detección de bordes con precisión subpíxel en imágenes digitales: Interpolación lineal frente a esquemas de tipo no lineal, V. J. d. M. Aplicada, Ed., Valencia, 2006, p. 2006.
- [27] S. S. Al-Amri, N. Kalyankar y S. Khamitkar, «Image segmentation by using edge detection.,» vol. 2, nº 3, 2010.
- [28] V. Valencia Delgado, J. Moreira y P. Chavez, «Implementacion de un algoritmo para la deteccion y conteo de celulas en imagenes microscopicas.,» 2010.
- [29] E. Coto, «Métodos de Segmentación de Imágenes Médicas,» Marzo, 2003.
- [30] H. Spontón y J. Cardelino, «A Review of Classic Edge Detectors,» 2015.
- [31] G. Shrivakshan y D. Chandrasekar, «A Comparison of various Edge Detection Techniques used in Image Processing,» vol. 9, nº 1, September 2012.
- [32] I. B. Hernanz, «Algoritmos de detención de imágenes ruidosas y duplicadas.,» Diciembre 2008.
- [33] M. d. P. Chiriboga Maldonado y A. E. Salcedo Eches, «Implementación de algoritmos para preprocesamiento, detención de bordes y segmentación de imágenes digitales a color utilizando Matlab,» Junio 2003.
- [34] S. Saini, B. Kasliwal y S. Bhatia, «Comparative study of image edge detection algorithms,» 2010.
- [35] G. Shrivakshan y D. Chandrasekar, «A Comparison of various Edge Detection Techniques used in Image Processing,» vol. 9, nº 1, September 2012.
- [36] R. Maini y H. Aggarwal, «Canny Edge Detection,» *International journal of image processing (IJIP)*., vol. 3, nº 1, pp. 1-11, 2009.
- [37] D. Corral Alcón, «Mejora de un algoritmo de segmentación de Iris mediante Contornos Activos.,» Julio 2012.

- [38] G. Shrivakshan y D. Chandrasekar, «A Comparison of various Edge Detection Techniques used in Image Processing,» vol. 9, nº 1, September 2012.
- [39] L. F. Caicedo Marmolejo y C. E. Chamarro Carvajal, «Implementación y Evaluación de algoritmos de identificación por Reconocimiento del Iris Ocular.,» 2012.
- [40] O. R. Vicent y O. Folorunso, «Canny Edge Detection,» pp. 97-107, 2009.
- [41] L. R. Sanchiz Redondo, «Segmentación de Iris mediante Contornos Activos.,» 2011.
- [42] V. Masero Vargas, «Una nueva metodología de segmentación de imágenes basadas en Contornos Activos. Aplicación de la segmentación de imágenes médicas para la reconstrucción tridimensional de estructuras anatómicas.,» 2016.
- [43] F. Arámbula, Análisis de imágenes médicas, 2008.
- [44] J. Fessler, M. Sonka y J. M. Fitzpatrick, «Handbook of Medical Imaging,» vol. 2, pp. 1-70, 2000.
- [45] M. d. M. Pujol López, Incorporación de características en la función de energía para segmentación de imágenes usando campos aleatorios de Markov, 2000.
- [46] N. J. Machado, Segmentación de Melanoma basado en Métodos de Binarización., San Lorenzo, Diciembre, 2014.
- [47] N. J. Machado, Segmentación de Melanoma basado en Métodos de Binarización., San Lorenzo, Diciembre, 2014.
- [48] M. Sonka y J. Fitzpatrick, «Validation of medical analysis techniques.,» de *Handbook of medical image analysis.*, vol. 2, 2000.
- [49] A. Z. Arifin y A. Asano, Image segmentation by histogram thresholding using hierarchical cluster analysis. *Pattern Recognition Letters*, vol. 27, *Pattern recognition letters*, 2006, pp. 1515-1521..
- [50] W. Wang y e. al, Comparison of image segmentation methods in simulated 2D and 3D microtomographic images of soil aggregates., vol. 162, *Geoderma*, 2011, pp. 231-241.

- [51] S.-F. Yang-Mao, Y.-K. Chan y Y.-P. Chu, «Edge enhancement nucleus and cytoplasm contour detector of cervical smear images. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*,» de *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, 2008, pp. 353-366..
- [52] M. Sezgin y B. Sankur, Selection of thresholding methods for nondestructive testing applications. En *Image Processing, 2001. Proceedings. 2001 International Conference on*. IEEE, 2001, pp. 764-767.
- [53] I. Saha y e. al., SVM ensemble fuzzy clustering for satellite image segmentation. *IEEE Geoscience and Remote Sensing Letters*, vol. 9, 2012, pp. 52-55.
- [54] A. A. Taha y A. Hanbury, Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC medical imaging*, vol. 15, 2015, p. 29.
- [55] L. P. Coelho, A. Shariff y R. F. Murphy, «Nuclear segmentation in microscope cell images: a hand-segmented dataset and comparison of algorithms.,» de *En Biomedical Imaging: From Nano to Macro, 2009, 2009*, pp. 518-521..
- [56] C. e. a. CHU, Multi-organ segmentation based on spatially-divided probabilistic atlas from 3D abdominal CT image. En *International Conference on Medical Image Computing and Computer-Assisted Intervention.*, Springer, Berlin, Heidelberg, 2013., pp. 165-172..
- [57] P. M. Yoshioka, Misidentification of the Bray-Curtis similarity index. *Marine Ecology Progress Series*, vol. 368, 2008, pp. 309-310.
- [58] A. Schubert y A. Telcs, A note on the Jaccardized Czekanowski similarity index. *Scientometrics*, 2014, pp. 1397-1399.
- [59] M. L. Goulden y a. et, Sensitivity of boreal forest carbon balance to soil thaw. *Science*, 1998, pp. 214-217.
- [60] K. Asanovic y e. al, The landscape of parallel computing research: A view from Berkeley. Technical Report UCB/EECS-2006-183, EECS Department., University of California, Berkeley, 2006.
- [61] M. Floris y e. al, «A generalizable definition of chemical similarity for read-across.,» *Journal of cheminformatics*, vol. 6, nº 1, p. 39, 2014.

- [62] L. Nieddu y A. Rizzi, Metrics in symbolic data analysis. En *New Developments in Classification and Data Analysis*, Springer, Berlin, Heidelberg, 2005, pp. 71-78.
- [63] O. Arenas y e. al, «Comparison of the Haplotypes of the major Histocompatibility complex in Rat:V.A Serological,Genetic and numerical study of the RT1. A Region Antigens,» *International Journal of Immunogenetics*, vol. 8, nº 4, pp. 307-314, 1981.
- [64] B. Desgraupes, Clustering indices. University of Paris Ouest-Lab ModalX, vol. 1, 2013, p. 34.
- [65] H. Michalopoulou y C. Jacovides, Instability indices for the Cyprus area. *Meteorology and Atmospheric Physics*, 1987, pp. 153-158..
- [66] H. Hubert y e. al, «Comparison of the diagnostic quality of body mass index, waist circumference and waist-to-height ratio in screening skinfold-determined obesity among children,» *Journal of science and medicine in sport*, vol. 12, nº 4, pp. 449-451, 2009.
- [67] K. Virágh, The effect of herbicides on vegetation dynamics: comparison of classifications. *Abstracta Botanica*, 1987, pp. 53-70..
- [68] J. Valente de Olivera, A. Szabo y L. Nunes de Castro, Particle Swarm Clustering in clustering ensembles: Exploiting pruning and alignment free consensus. *Applied Soft Computing*, vol. 55, 2017, pp. 141-153.
- [69] M. Meilá, «Comparing clusterings: an axiomatic view,» de *En Proceedings of the 22nd international conference on Machine learning*, ACM, 2005.
- [70] R. J. Campello, A fuzzy extension of the Rand index and other related indexes for clustering and classification assessment. *Pattern Recognition Letters*, 2007, pp. 833-841..
- [71] O. Jesorsky, K. J. Kirchberg y R. W. Frischholz, «Robust face detection using the hausdorff distance,» de *En International Conference on Audio-and Video-Based Biometric Person Authentication*, Springer, Berlin, Heidelberg, 2001.
- [72] C.-B. Yu y e. al., «Finger-vein image recognition combining modified hausdorff distance with minutiae feature matching. *Interdisciplinary Sciences*,» de *Computational Life Sciences*, 2009, pp. 280-289..

- [73] M. A. Sánchez Mendoza, «Metodologías de desarrollo de software.,» 2004.
- [74] C. J. H, L. Patricio y P. M.Carmen., *Métodologías Ágiles en el Desarrollo de Software.*, Valencia, 2003.
- [75] J. Joskowicz, *Reglas y prácticas en eXtreme Programming.*, 2008.
- [76] D. Bustamante y J. Rodriguez, «Metodología Actual:Metodología XP.,» 2014.
- [77] M. L. Suarez, «Introducción a la programación y sus lenguajes.,» de *Competencias en TIC: Colección de Fasículos Digitales.*, 2015.
- [78] Oracle, «Java,» [En línea]. Available: [http://www.java.com/es/about/..](http://www.java.com/es/about/) [Último acceso: 14 1 2015].
- [79] P. Visual, «Visual Paradigm,» 2016. [En línea]. Available: <https://www.visual-paradigm.com..> [Último acceso: 30 3 2016].
- [80] SQLite, «SQLite,» [En línea]. Available: <http://www.sqlitemanager.org.> [Último acceso: 5 2 2015].
- [81] «Admin, SQLite,» [En línea]. Available: [sqliteadmin.orbmu2k.de.](http://sqliteadmin.orbmu2k.de.) [Último acceso: 10 2 2015].
- [82] D. A. F. C. Y. N. W. d. V. Dra. María Belén Castañeda, *Procesamiento de datos y análisis estadísticos utilizando SPSS*, Porto Alegre,Brasil, 2010, pp. 13-164.
- [83] M. Alvarez Cancio, «PANDOC: Sistema basado en casos para la cuantificación objetiva de la opacidad capsular de un paciente operado de catarata,» La Habana, 2014.
- [84] K. Beck, F. J. T. Z. Martínez y J. R. T. G. MOLINA., «Una explicación de la programación extrema: aceptar el cambio,» 2002.
- [85] K. Beck, F. J. T. Z. Martínez y J. R. T. G. MOLINA., «Una explicación de la programación extrema: aceptar el cambio,» 2002.
- [86] A. Stellman y J. Greene, «Applied Software Project Management,» nº 113, 2005.
- [87] J. Joskowicz, *Reglas y prácticas en eXtreme Programming.*, 2008.
- [88] K. BECK, «Embracing change with extreme programming,» vol. 32, nº 10, pp. 70-77, 1999.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

- [89] S. L. Casas y H. H. Reinaga, «Aspectos tempranos : un enfoque basado en Tarjetas CRC,» vol. 6, nº 1, pp. 85-92, 2009.
- [90] C. Larman, UML y patrones, Introducción al análisis y diseño orientado a objetos, 2da edición, 2003.
- [91] C. Larman, UML y Patrones. Introducción al análisis y diseño orientado a objetos., México, 2005.
- [92] E. Integra., «ERP Integra.,» 2010. [En línea]. Available: [http://erp-integra.com/homepage/index.php?option=com\\_content&view=article&id=29&Itemid=51](http://erp-integra.com/homepage/index.php?option=com_content&view=article&id=29&Itemid=51). [Último acceso: 1 marzo 2018].
- [93] D. G. Valle y D. G. Rafael PhD, Guía de SPSS 22 para desarrollo de trabajos de investigación., Malaga, 2016.
- [94] R. R. Ochoa Torrez, Analisis Estadistico en el SPSS, La Paz, 2014.
- [95] C. L. J. Galindo, Técnicas de investigación en sociedad, cultura y comunicación., Pearson Educación, 1998.
- [96] F. Wilcoxon, «Individual comparisons by ranking methods,» *Biometrics bulletin*, vol. 1, pp. 80-83., 1945.
- [97] M. Friedman, «The use of ranks to avoid the assumption of normality implicit in the analysis of variance,» *Journal of the american statistical association*, vol. 32, pp. 675-701, 1937.
- [98] A. C. Elliott y W. A. Woodward, Statistical analysis quick reference guidebook: With SPSS examples, Sage, 2007, pp. 201-203.
- [99] K. BECK, «Extreme Programming Explained.,» 1999.
- [100] A. Yagüe y J. Garbojosa, «Las pruebas en metodologías ágiles y convencionales: ] papeles diferentes.,» *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*, vol. 3, nº 4, 2009.
- [101] G. J y a. et, «Pruebas del Sistema en Programación Externa,» 2010.  
]
- [102] I. Sommerville, Ingeniería del software. Séptima Edición., Madrid. España:  
] Pearson Educación. S. A., 2005.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

[103 D. Malfará, D. Cukerman, F. Cócaro, J. P. Cassinelli y R. Séttimo, Testing en  
] eXtreme Programming, 2006.

[104 R. Pressman, «Ingeniería del Software un enfoque práctico.,» 2005.  
]

[105 I. Sommerville, Ingeniería del software. Séptima Edición., Madrid. España:  
] Pearson Educación. S. A., 2005.

[106 L. P. P. M. C. Canós José H, «Metodologías Ágiles en el Desarrollo de Software,»  
] 2003.

ANEXOS

**Anexo1. Historias de usuario**

Historia de usuario	
<b>Número:</b> 1	<b>Nombre:</b> Calcular índice (VP,VN,FP,FN)
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular las cantidades de (VP,VN,FP,FN) atendiendo las coincidencias y discrepancias de la imágenes	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 15:HU1 Calcular índice (VP, VN, FP, FN).

Historia de usuario	
<b>Número:</b> 2	<b>Nombre:</b> Calcular Tasa de Correctos equilibrada
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular la Tasa de Correctos equilibrada mediante la fórmula (11).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente	

Tabla 16:HU2 Calcular Tasa de Correctos equilibrada.

Historia de usuario	
<b>Número:</b> 3	<b>Nombre:</b> Calcular métrica de Clasificación del Error
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

<b>Descripción:</b> El sistema debe calcular la métrica de Clasificación del error mediante la fórmula (5).
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.

Tabla 17:HU3 Calcular métrica de Clasificación del Error.

Historia de usuario	
<b>Número:</b> 4	<b>Nombre:</b> Calcular índice Desajuste de Bordes
<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice Desajuste de Bordes mediante la fórmula (6).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 18:HU4 Calcular índice Desajuste de Bordes.

Historia de usuario	
<b>Número:</b> 5	<b>Nombre:</b> Calcular índice de Similitud de Jaccard
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Medio (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de similitud de Jaccard mediante la fórmula (8).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 19:HU5 Calcular índice de Similitud de Jaccard.

Historia de usuario	
<b>Número:</b> 6	<b>Nombre:</b> Calcular índice de Yule
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de Yule mediante la fórmula (14).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 20:HU6 Calcular índice de Yule.

Historia de usuario	
<b>Número:</b> 7	<b>Nombre:</b> Calcular índice de Fowlkes-Mallows
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de Fowlkes-Mallows mediante la fórmula (16).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 21:HU7 Calcular índice de Fowlkes-Mallows.

Historia de usuario	
<b>Número:</b> 8	<b>Nombre:</b> Calcular índice de Similitud de Czekanowki
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de similitud de Czekanowki mediante la fórmula (10).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 22:HU8 Calcular índice de Similitud de Czekanowki.

Historia de usuario	
<b>Número:</b> 9	<b>Nombre:</b> Calcular índice Rand ajustado
<b>Iteración asignada:</b> 2	

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice Rand Ajustado mediante la fórmula (7).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 23:HU9 Calcular índice Rand Ajustado.

Historia de usuario	
<b>Número:</b> 10	<b>Nombre:</b> Calcular índice de Índice de Rogers-Tanimoto
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Medio (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de Rogers-Tanimoto mediante la fórmula (13) (14).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 24:HU10 Calcular índice de Rogers-Tanimoto.

Historia de usuario	
<b>Número:</b> 11	<b>Nombre:</b> Calcular índice de Índice de Sokal-Sneath
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Medio (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de Sokal-Sneath mediante la fórmula (12)	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 25:HU11 Calcular índice de Sokal-Sneath.

Historia de usuario	
<b>Número:</b> 12	<b>Nombre:</b> Calcular índice de Índice de Russell-Rao

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Medio (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular el índice de Russell-Rao mediante la fórmula (15).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 26:HU12 Calcular índice de Russell-Rao.

Historia de usuario	
<b>Número:</b> 13	<b>Nombre:</b> Calcular Modificación de la Distancia de Hausdorff Normalizada
<b>Iteración asignada:</b> 2	
<b>Prioridad en negocio:</b> Medio (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema debe calcular Modificación de la Distancia de Hausdorff Normalizada mediante la fórmula (17).	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 27:HU13 Calcular Modificación de la Distancia de Hausdorff Normalizada.

Historia de usuario	
<b>Número:</b> 14	<b>Nombre:</b> Determinar mejor segmentación
<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semanas
<b>Descripción:</b> El sistema tras tener los resultados obtenidos en las comparaciones entre el valor medio de las métricas por cada algoritmo devuelve el nombre del mejor algoritmo entre ellos atendiendo a los valores de la media.	
<b>Observaciones:</b>	

Tabla 28:HU14 Determinar mejor segmentación.

*Métricas basadas en un Ground Truth para la validación de segmentación de imágenes médicas.*

Historia de usuario	
<b>Número:</b> 15	<b>Nombre:</b> Calcular algoritmo más robusto
<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Media (Alta / Media / Baja)	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> Medio (Alto / Medio / Bajo)	<b>Puntos reales:</b> 1 semanas
<b>Descripción:</b> El sistema debe calcular el algoritmo más robusto.	
<b>Observaciones:</b> El usuario debe entrar el "Ground Truth" y la imagen segmentada automáticamente.	

Tabla 29:HU15 Calcular algoritmo más robusto.

Historia de usuario	
<b>Número:</b> 16	<b>Nombre:</b> Gestionar resultados
<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 1 semanas
<b>Riesgo en desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos reales:</b> 1 semanas
<b>Descripción:</b> El sistema debe permitir al usuario exportar e importar los resultados de las métricas en un Excel, así como también debe permitir eliminar una tabla o una fila seleccionada.	
<b>Observaciones:</b>	

Tabla 30:HU16 Gestionar resultados.

Historia de usuario	
<b>Número:</b> 17	<b>Nombre:</b> Graficar y exportar resultados
<b>Iteración asignada:</b> 3	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 2 semanas
<b>Riesgo en desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos reales:</b> 2 semana
<b>Descripción:</b> El sistema debe permitir al usuario graficar los resultados de las métricas por fila, por imágenes y por algoritmos, así como también debe permitir exportar el grafico como imagen.	

**Observaciones:**

Tabla 31:HU17 Graficar y exportar resultados.

**Anexo2.Tiempo de implementación de las HU**

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Calcular índice (VP,VN,FP,FN)	2	2
Calcular tasa de Correctos equilibrada	2	2
Calcular métrica de clasificación del error	2	2
Modificación de la Distancia de Hausdorff Normalizada	2	2
Calcular Desajuste de Bordes	2	2

Tabla 32: Tiempo de implementación de las HU de la iteración #1.

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Calcular índice de similitud de Jaccard	2	2
Calcular índice de Yule	2	2
Índice de Russell-Rao	2	2
Índice de Rogers-Tanimoto	2	2
Índice de Sokal-Sneath	2	2
Calcular índice de Fowlkes-Mallows	2	2
Calcular índice de similitud de Czekanowski	2	2
Calcular índice Rand ajustado	2	2

Tabla 33: Tiempo de implementación de las HU de la iteración #2.

Historias de usuario	Tiempo de implementación	
	Estimación	Real
Determinar mejor segmentación	2	2
Calcular algoritmo más robusto	2	2
Gestionar resultados	2	2
Graficar y exportar resultados	2	2

Tabla 34: Tiempo de implementación de las HU de la iteración #3

## GLOSARIO DE TÉRMINOS

**Cristalino:** Medio refringente del ojo con alto poder dióptrico encargado de la acomodación (Enfoque para visión cercana o lejana).

**PENTACAM:** Equipo oftalmológico de alta tecnología capaz de reconstruir imágenes tridimensionales de alta resolución del polo anterior del ojo

**Sistema Sheimpflug:** Sistema de cámaras de alta definición acopladas a diferentes equipos médicos como el PENTACAM.

**PANDOC:** Programa analizador de opacidad capsular.

**Capsulorrexia:** Maniobra realizada en la cirugía de catarata donde se crea una abertura circular continua en la cápsula anterior del cristalino.

**Convolución:** Es un operador matemático que transforma dos funciones  $f$  y  $g$  en una tercera función que en cierto sentido representa la magnitud en la que se superponen  $f$  y una versión trasladada e invertida de  $g$ .

**Retina:** Membrana interior del ojo en la cual se reciben las impresiones luminosas que son transmitidas al cerebro; cubre la coroides hasta el iris y está formada esencialmente por expansiones del nervio óptico.

**Pixel:** Unidad básica de una imagen digitalizada en pantalla a base de puntos de color o en escala de grises.