



Universidad de las Ciencias Informáticas

Facultad 3

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

Autores: Thais Alonso Tarafa

Alexei Massabeaut Villafruela

Tutores

Ing. Michel Álvarez Cancio

Lic. Reyder Cruz de la Osa

La Habana, 2016

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos 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 _____.

Firma del autor

Firma del autor

Firma del tutor

Firma del tutor

Agradecimientos

Agradezco en primer lugar a mis padres por ser lo mejor que tengo y por su dedicación y amor hacia mí. Los quiero mucho.

Agradecer a mi tío Kike que ha sido un padre para mí desde siempre, a mis abuelas que son unas luchadoras y me han contagiado su espíritu y a mi familia en general que siempre me ha apoyado.

Gracias a todas las personas que me han ayudado a crecer como persona en esta Universidad, y con las que he pasado momentos muy divertidos, estas son mis amigas las venenosas Yami , Yesi,,y Jenni la sobreviviente del 51101, mis amigos el Poli y Erlis que me han acompañado desde primer año, los quiero gracias por aguantarme. Por ultimo quiero agradecer a mis compañeros de aula, insoportables por momentos, pero muy lindas personas.... muy en el fondo algunos ya todas las personas que de una forma u otra me ayudaron y me apoyaron, a todos ellos **MUCHAS GRACIAS.** *Thais*

Quiero agradecer en primer lugar a mi mamá, que es lo más grande que tengo en la vida, que la quiero mucho, gracias por apoyarme siempre en mis decisiones, esas cosas son las que la hacen que para mí seas lo mejor. Te quiero mami.

Agradecer a Enrique Fabas que ha sido para mí como un padre; por aconsejarme y siempre estar ahí dándome consejos. Esta es una de las tantas formas que le demuestro que también le quiero y que siempre pienso en él.

A mi tía Maritza, a mi abuela, mi primo Michel, mi prima Anaiza y mi tía Lucia.

Agradecer a mi compañera de tesis por saber aguantarme durante el desarrollo de esta investigación.

Gracias a todos mis amigas y amigos, en especial al Robert, al Moro, al Carlo "Helado", al Kelly, a Yoandry "el PC", a Ge, al Mizra, a Raciél, al JeanKa "El amigo imaginario", a Neysi, a Mebys, a Karla, Dailet, al Javi, al Pache, a Yordan, a Cesar, Maceo, a la gente del gym y demás compañeros por los momentos y bromas compartidas durante estos años.

A todas las personas que de una forma u otra me ayudaron y me apoyaron, a todos ellos **MUCHAS GRACIAS.** *Alexei*

Dedicatoria

Quiero dedicarle este logro especialmente a mi abuelo “Betico”, que seguro le hubiese gustado verme graduarme de Ingeniero. Alexei

Dedico esta tesis especialmente a mis padres y a mi tío kike por enseñarme a nunca rendirme ante los obstáculos de la vida.

Thais

RESUMEN

El iris humano presenta una serie de patrones únicos en cada persona. El mismo refleja mediante marcas, síntomas presentes en el cuerpo. Estas marcas son estudiadas por la Iridología, rama de la medicina natural que plantea realizar un diagnóstico de posibles padecimientos mediante el análisis del iris. En la mayoría de los casos el análisis de estas marcas se realiza utilizando imágenes de iris obtenidas usando un iridoscopio y analizando la imagen de manera manual. Las imágenes que se obtienen del iridoscopio pertenecen al espectro de luz visible y suelen tener información no deseada o ruido, ocasionando deficiencias en el diagnóstico.

El presente trabajo se ha realizado con el propósito de desarrollar una herramienta informática que permita segmentar y normalizar la región del iris, como primera fase del desarrollo de una solución informática futura para Iridología. La solución está basada en el uso de técnicas de procesamiento digital de imágenes, las cuales fueron implementadas mediante el uso de tecnologías de código abierto, cumpliendo con las políticas de soberanía tecnológica establecidas en el país. La herramienta obtenida constituye un avance en las investigaciones de tratamiento de imágenes de iris humano en espectro visible, al proponer una combinación de técnicas que reducen el ruido en el proceso de segmentación de iris.

Palabras claves: Iris humano, espectro visible, procesamiento digital de imágenes, segmentación.

ÍNDICE

INTRODUCCIÓN.....	11
--------------------------	-----------

CAPÍTULO 1: Fundamentación teórica.	14
---	-----------

1.1 <i>Introducción.....</i>	14
------------------------------	----

1.2 <i>Modelo Conceptual</i>	14
------------------------------------	----

1.2.1 <i>Iris Humano.</i>	14
--------------------------------	----

1.2.3 <i>Iridología.</i>	14
-------------------------------	----

1.2.4 <i>Iridoscopio HSK -9918U.</i>	14
---	----

1.2.5 <i>Imagen Digital.</i>	15
-----------------------------------	----

1.2.6 <i>Imágenes obtenidas con luz infrarroja.....</i>	15
---	----

1.2.7 <i>Imágenes de iris capturadas bajo el espectro de luz visible.....</i>	15
---	----

1.2.8 <i>Procesamiento Digital de Imágenes.</i>	16
--	----

1.3 <i>Ruido en imágenes.....</i>	16
-----------------------------------	----

1.3.1 <i>Ruido de sal y pimienta o ruido aleatorio.....</i>	17
---	----

1.3.2 <i>Ruido dependiente de la señal o ruido gaussiano.....</i>	17
---	----

1.3.3 <i>Reducción de ruido.....</i>	18
--------------------------------------	----

1.4 <i>Realce y Mejora o Filtrado.....</i>	18
--	----

1.4.1 <i>Técnicas de pre-procesamiento de las imágenes de iris.....</i>	18
---	----

1.4.2 <i>Filtros de convolución o Filtros lineales.....</i>	19
---	----

1.4.3 <i>Filtro de la media.</i>	20
---------------------------------------	----

1.4.4 <i>Filtro de la mediana.</i>	21
---	----

1.4.5 <i>Histogramas.....</i>	23
-------------------------------	----

1.4.6 <i>Morfología Matemática. Operaciones de Minkowsky.....</i>	24
---	----

1.5 <i>Técnicas de mapeo de bordes.....</i>	26
---	----

1.5.1 <i>Operadores de gradiente común.....</i>	26
---	----

1.5.2 <i>Canny</i>	27
--------------------------	----

1.5.3 <i>Algoritmo Iterativo.....</i>	27
---------------------------------------	----

1.6 <i>Técnicas de segmentación de imágenes de iris.</i>	28
---	----

1.6.1 <i>Operador integro-diferencial.</i>	28
---	----

1.6.2 <i>Transformada Circular de Hough.....</i>	29
--	----

1.7 <i>Normalización del iris.....</i>	29
--	----

1.8 <i>Metodología de Desarrollo.....</i>	30
---	----

1.9 <i>Herramientas y lenguajes usados en el desarrollo.</i>	32
---	----

1.10	<i>Conclusiones del capítulo</i>	34
CAPÍTULO 2: Propuesta de Solución		34
2.1	<i>Introducción</i>	34
2.2	<i>Descripción general de la propuesta</i>	35
2.3	<i>Fase I: Planificación del Proyecto</i>	40
2.2.1	<i>Historias de Usuario</i>	41
2.2.2	<i>Requisitos no funcionales del sistema</i>	41
2.2.3	<i>Plan de Iteraciones</i>	42
2.2.4	<i>Plan de entrega</i>	42
2.3	<i>Fase II: Diseño del Sistema</i>	43
2.3.1	<i>Tarjetas Clase - Responsabilidad – Colaborador</i>	43
2.3.2	<i>Patrones de diseño de software</i>	44
2.4	<i>Fase III: Codificación</i>	44
2.4.1	<i>Estándares de codificación</i>	45
2.4.2	<i>Tareas de Programación</i>	45
2.5	<i>Conclusiones del capítulo</i>	46
CAPÍTULO 3: Resultados y validación del sistema		46
3.1	<i>Introducción</i>	46
3.2	<i>Validación del diseño</i>	46
3.2.1	<i>Relaciones entre clases (RC)</i>	46
3.2.2	<i>Tamaño Operacional de Clases (TOC)</i>	48
3.3	<i>Pruebas de software</i>	50
3.4	<i>Métrica de evaluación de la calidad de la segmentación</i>	55
3.5	<i>Pruebas de Ejecución</i>	57
3.5	<i>Aporte de la investigación</i>	58
3.6	<i>Conclusiones del capítulo</i>	58
Conclusiones generales		59
Recomendaciones		60
Bibliografía		61

ÍNDICE DE FIGURAS

Figura 1 : Proceso de convolución de píxeles (Nube, 2010)..... 19

Figura 2 : Banda basura para una máscara de 3x3 (Cepero Ruz, y otros, Mayo 2010).20

Figura 3 : Máscaras más utilizadas en el filtro de la media (Tristan, 2011).21

Figura 4 :a) Imagen con ruido gaussiano b) Imagen resultante del algoritmo de la media con una máscara de 3x3 c) Imagen resultante del algoritmo de la media con una máscara de 5x5 (Cepero Ruz, y otros, Mayo 2010)..... 21

Figura 5 : Proceso del algoritmo de la mediana (Nube, 2010).22

Figura 6 : Máscara cuadrada, en X y en cruz (Cepero Ruz, y otros, Mayo 2010).22

Figura 7 :a) Imagen con ruido aleatorio b) Imagen luego de aplicar algoritmo de la mediana (Cepero Ruz, y otros, Mayo 2010)..... 23

Figura 8 : Ejemplo de Histograma 23

Figura 9 : Ecuación del Histograma (León, Moya, & Catalán, 2005)..... 24

Figura 10: Resultado de las operaciones de erosión, dilatación, apertura y cierre (Benalcazar Palacios, 2008).26

Figura 11 : Localización de párpados, iris y pupila mediante el uso del operador integro diferencial. 29

Figura 12 : Normalización del iris (Caicedo Marmolejo, y otros, 2012)..... 30

Figura 13 : Diagrama de Procesos de Negocio. 36

Figura 14 : Mirilla para la ubicación del iris 37

Figura 15: Imagen Original e imagen en escala de grises. 38

Figura 16: Imagen en escala de grises e imagen con filtro erosión. 38

Figura 17: Imagen con filtro de erosión e imagen con filtro clahe. 38

Figura 18: Imágenes Binarizada de pupila e iris. 39

Figura 19: Imagen con mapa de bordes de pupila e iris..... 39

Figura 20: Resultado de aplicar la transformada circular de Hough para la detección de iris y pupila. 40

Figura 21 : Imagen de iris normalizado. 40

Figura 22 Representación en (%) de los resultados de la aplicación de la métrica RC. 48

Figura 23 Representación en (%) de los resultados de la aplicación de la métrica TOC.....	50
<i>Figura 24 : Código fuente del método binarizaciónIris ().</i>	51
<i>Figura 25 : Grafo de flujo asociado al método binarizaciónIris ().</i>	52

ÍNDICE DE TABLAS

Tabla 1 : Máscaras asociadas a los operadores de gradiente común	27
Tabla 2: Selección de las técnicas de filtrado de imágenes	37
Tabla 3 : HU5 Implementar la técnica de filtrado erosión.....	41
Tabla 4 : Plan de entrega.	43
Tabla 5: Tarjeta CRC de la clase Mediana	44
Tabla 6: Tarea de Programación 1	46
Tabla 7 : Rango de valores para medir la afectación de los atributos de calidad (RC).....	47
Tabla 8 : Rango de valores para medir la afectación de los atributos de calidad (TOC).	49
Tabla 9 : Caso de prueba (CP) para la HU Obtener la Imagen del iridoscopio HSK-9918U.....	55
Tabla 10 : Resultados Generales de la métrica de calidad de segmentación	57
Tabla 11 : Tiempo Promedio de Ejecución de la herramienta.....	58

INTRODUCCIÓN

En el presente siglo la humanidad ha sido testigo del creciente desarrollo de la tecnología y sus diversas aplicaciones en beneficio de la sociedad. En este sentido se aprecian los avances en la investigación del iris humano (Caicedo Marmolejo, y otros, 2012) (Finamore, y otros, 2011) (Tomé González, 2008).

El iris es un tejido de colores en forma de anillo alrededor de la pupila, a través de la cual entra la luz. Este contiene un amplio patrón de surcos, crestas y tramas pigmentadas. La superficie del iris está compuesta por dos regiones, la parte central denominada zona pupilar y la exterior denominada zona ciliar, delimitadas a la vez por el collarete que es el borde entre ambas regiones (Tejedor Gómez, 2009).

La textura pigmentada del iris es aleatoriamente determinada durante el desarrollo fetal del ojo, esta es diferente entre personas (con cualquier grado de parentesco) y su apariencia puede cambiar debido a enfermedades o padecimientos (Tejedor Gómez, 2009). Esta condición de cambio es estudiada en la actualidad por los especialistas de Iridología. La Iridología es un método para el diagnóstico de tendencias o alteraciones funcionales orgánicas, basado en el análisis de los signos que se encuentran en el iris como son: cambios de coloración, alteraciones del tejido, relieves, surcos, bandas y la zona en la que se encuentran (Salles LF, 2008).

El análisis iridológico permite reconocer y diferenciar los signos presentes en cada iris. El primer paso del análisis es obtener la imagen del ojo humano, para ello actualmente se utiliza una cámara digital o aparatos ópticos adaptados como el iridoscopio. Como segundo paso el especialista realiza un análisis de la imagen, según su experiencia en la materia. Luego realiza preguntas al paciente para conocer los síntomas que presenta y posteriormente realiza su diagnóstico. Este procedimiento es lento y en ocasiones no eficiente debido a que todos los especialistas no tienen la misma experiencia.

Existen diversos tipos de iridoscopio utilizados en el proceso de obtención de la imagen, para la siguiente investigación se utilizará el iridoscopio HSK-9918U. Este dispositivo fue donado al grupo de investigación de Inteligencia Artificial (AIRI) de la Universidad de las Ciencias Informáticas, para realizar investigaciones con imágenes de iris pertenecientes al espectro de luz visible.

Actualmente una de las líneas investigativas del grupo AIRI (*Artificial Intelligence Research and Innovation*) de la Universidad de las Ciencias Informáticas pretende desarrollar una herramienta informática que permita la detección de marcas en imágenes del iris en espectro visible.

Las imágenes capturadas por el iridoscopio pertenecen al espectro de luz visible y se caracterizan por tener mucho ruido, reflejos especulares, interferencia de párpados y pestañas que dificultan el proceso de diagnóstico. Estas características pueden ser minimizadas mediante un proceso de

tratamiento digital de imágenes para posteriormente aplicar técnicas de segmentación de iris para la extracción y normalización de este.

Una vez culminada la etapa de análisis se dio paso a la próxima etapa de desarrollo donde se identifica el siguiente **problema a resolver**: ¿Cómo disminuir el ruido generado en imágenes capturadas bajo el espectro de luz visible durante el proceso de extracción de iris?

Se identifica como **objeto de estudio**: Segmentación de imágenes de iris, enfocado en el **campo de acción**: Segmentación de imágenes de iris capturadas bajo el espectro visible.

Se plantea la siguiente **idea a defender**, si se desarrolla una herramienta informática para la segmentación entonces se disminuye el ruido en imágenes de iris capturadas bajo el espectro de luz visible.

Para dar solución al problema planteado se define el **objetivo general**: Desarrollar una herramienta informática que disminuya el ruido, para la segmentación de la región del iris en imágenes del espectro visible.

Objetivos específicos:

1. Elaborar el marco teórico de la investigación mediante el estudio de las técnicas actuales para identificar tendencias y adoptar posiciones al respecto.
2. Diseñar un software para la segmentación de imágenes de iris tomadas con el iridoscopio HSK-9918U.
3. Implementar los elementos definidos en el diseño del software.
4. Validar los resultados obtenidos verificando la efectividad en la detección de iris y pupila.

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

- **Analítico-Sintético**: Se utiliza con el objetivo de analizar las teorías, documentos e información referente al tema en cuestión, para luego extraer los elementos más importantes que se relacionan con la segmentación de imágenes.
- **Inductivo – deductivo**: Para luego de inducir una serie de conocimientos referentes a la visualización médica, poder arribar a razonamientos que conlleven a la deducción de conocimientos que puedan ser aplicables al problema a tratar en particular.
- **Análisis histórico-lógico**: Permitirá analizar y estudiar la trayectoria y evolución de los algoritmos de segmentación, para así poder contar con una noción de cuán desarrollado está el tema a nivel global y tomar la decisión de cuál o cuáles desarrollar.

-
- **Experimento:** Método empírico mediante el cual se realizarán experimentos y pruebas de los principales algoritmos de pre-procesamiento y de segmentación de imágenes médicas 2D, para examinar los resultados en busca de escoger los más adecuados.
 - **Observación:** Mediante este método empírico se observarán los resultados obtenidos en la caracterización e identificación de los principales algoritmos utilizados, para poder decidir luego cuál o cuáles serán más adecuados.

La presente investigación está estructurada por los siguientes capítulos; a continuación, se describe el objetivo principal de cada uno de ellos:

Capítulo 1: Fundamentación teórica.

En este capítulo se realiza un análisis de los diferentes conceptos relacionados con el problema identificado, así como la explicación breve de las técnicas que se utilizan en la actualidad para el tratamiento de imágenes digitales.

Capítulo 2: Propuesta de Solución

En este capítulo se propone una solución al problema planteado. Se explican las razones que derivan en la selección de herramientas y tecnologías a utilizar durante el desarrollo de la misma. Además se desarrollan las tres primeras fases de la metodología seleccionada con los artefactos que genera.

Capítulo 3: Resultados y Validación del sistema. Se muestra el conjunto de pruebas realizadas a la herramienta y los resultados obtenidos.

CAPÍTULO 1: Fundamentación teórica.

1.1 Introducción

En este capítulo se expondrán de manera sintetizada los elementos básicos teóricos, de las técnicas de procesamiento, filtrado y mejora de imágenes de iris. También se explicarán brevemente las técnicas de segmentación y normalización de iris. Este capítulo estará enfocado en tres elementos importantes en el tratamiento de imágenes digitales, “Obtención de imágenes de iris”, “Pre-Procesamiento de las imágenes” y “Segmentación”.

1.2 Modelo Conceptual

En este epígrafe se exponen una serie de conceptos que ayudarán a un mejor entendimiento del tema.

1.2.1 Iris Humano.

El iris del ojo humano es la membrana coloreada y circular del ojo que separa la cámara anterior de la cámara posterior; posee una apertura central de tamaño variable que comunica las dos cámaras y se denomina pupila. La zona blanca que se encuentra alrededor se denomina esclerótica (Finamore, y otros, 2011).

El iris está constantemente activo, permitiendo a la pupila dilatarse (midriasis) o contraerse (miosis), de acuerdo a la intensidad que posea la fuente luminosa. Su textura pigmentada es aleatoriamente determinada durante el desarrollo fetal del ojo, y su apariencia es diferente entre personas (con cualquier grado de parentesco) y relativamente constante aunque puede cambiar debido a enfermedades (Tejedor Gómez, 2009).

1.2.3 Iridología.

La Iridología es el diagnóstico de condiciones médicas por medio del estudio del iris. Según este estudio, es posible diagnosticar y prevenir enfermedades observando el color, la texturaymotas de pigmentación en las diferentes partes del ojo que corresponden a una parte específica del cuerpo (Benavides, y otros, 2015). La observación de las lesiones, cambios de color y marcas que aparecen en el iris están relacionadas con reflejos de los daños orgánicos, alteraciones bioquímicas y enfermedades. Este método no debe ser utilizado como único medio diagnóstico, sino como parte del examen físico, complementado con la entrevista médica, la revisión de la historia clínica personal y familiar y la realización de exámenes de laboratorios, los cuales aumentan su efectividad (Alonso Dominguez, 2014).

1.2.4 Iridoscopio HSK -9918U.

El iridoscopio HSK-9918U es un dispositivo utilizado por los especialistas de Iridología para capturar imágenes del ojo humano y analizar las marcas presentes en el iris. Este dispositivo es

uno de los que comercializa la compañía china HSK Medical con fines médicos y fue cedido al grupo de investigación AIRI para realizar la presente investigación.

Este dispositivo posee las siguientes características (hskmedical, 2013):

1. Permite ajustar el brillo, enfoque, y estabilidad de los colores con un interruptor presente en el mango del mismo.
2. Captura imágenes en formato 640x480 px.
3. Posee dos luces Led que iluminan el ojo para tener una imagen con mayores detalles y dos botones que permiten seleccionar si se utiliza uno de las luces de tipo Led o ambas.

1.2.5 Imagen Digital.

Una imagen digital es un arreglo bidimensional de píxeles, donde el valor de cada píxel se representa mediante una función f , donde $f(x, y)$ 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 (Cepero Ruz, y otros, Mayo 2010).

1.2.6 Imágenes obtenidas con luz infrarroja.

La radiación infrarroja es un tipo de radiación electromagnética de mayor longitud de onda que la luz visible. Los rayos infrarrojos se utilizan en los equipos de visión nocturna cuando la cantidad de luz visible es insuficiente para ver los objetos (Finamore, y otros, 2011).

Los sistemas de reconocimientos de iris, funcionan en general, con imágenes obtenidas utilizando una luz en el espectro cercano al infrarrojo (en el rango de 700-900 nm). Con este tipo de iluminación se pueden apreciar de mejor manera los patrones que conforman el iris. Sin embargo, un exceso de este tipo de luz puede traer daños a la salud humana (Osorio Roig, y otros, 2015).

1.2.7 Imágenes de iris capturadas bajo el espectro de luz visible.

La mayor cantidad de trabajos relevantes publicados se ha centrado en la segmentación del iris en ambientes controlados, en estos las personas tienen que estar cerca de la cámara y mirar a la misma durante muy pocos segundos. Precauciones adicionales deben ser adoptadas para evitar lastimar el ojo humano debido a la excesiva iluminación infrarroja (Osorio Roig, y otros, 2015). Por lo tanto, la investigación sobre el reconocimiento de iris se ha ampliado en nuevas direcciones; como son el reconocimiento de iris desde la distancia (Tomé González, 2008), durante movimiento (Doraijaj, y otros, 2005), y bajo el espectro de longitud de onda visible (Osorio Roig, y otros, 2016).

Las cámaras de alta resolución disponibles en el mercado, hoy pueden capturar imágenes de iris de alta resolución bajo la luz visible. Cuando las imágenes son capturadas bajo el espectro de luz visible, se hace más complejo su procesamiento de segmentación; ya que se agudizan más los ruidos, los cuales pueden dificultar la fase de detección de bordes interior y exterior del iris, ejemplo de estos ruidos pueden ser las reflexiones especulares que son fantasmas de luces reflejados en la zona pupilar o incluso en el propio borde entre la pupila y el iris (Osorio Roig, y otros, 2016). Otro de los problemas que se refleja en la segmentación bajo este espectro está relacionado con la pigmentación del iris, debido a que una de sus moléculas llamada eumelanina permite agravar más los ruidos en el ojo, si este es bien fotografiado bajo estas condiciones (Osorio Roig, y otros, 2016).

En general el espectro de luz visible presenta muchas ventajas con respecto a la luz infrarroja en cuanto a afectaciones de la salud humana. En cuanto a deficiencias se tiene la degradación de la calidad de la imagen; debido a que la captura en espectro de luz visible provoca que aumenten más los ruidos sobre la imagen (Osorio Roig, y otros, 2016).

1.2.8 *Procesamiento Digital de Imágenes.*

El procesamiento de imágenes tiene que ver con la adquisición, transmisión, procesamiento y representación de las imágenes. Las técnicas de proceso de imágenes se utilizan para mejorar la apariencia visual de las imágenes para un observador y para preparar convenientemente el contenido fotográfico de cara a la percepción por parte de máquinas. El proceso digital de imágenes se puede dividir en las siguientes áreas (Gonzalez, y otros, 2002):

Adquisición o captura: se ocupa de los diferentes caminos para la obtención de imágenes; por ejemplo, utilizando cámaras digitales o digitalizando imágenes analógicas (fotografías).

Realce y mejora: técnicas que se usan para mejorar la apariencia visual de las imágenes o para recuperar o restaurar las imágenes degradadas.

Segmentación: se ocupa de la división de las imágenes en regiones o áreas significativas.

Extracción de características: se ocupa de la detección y localización de entidades geométricas simples y complejas. Desde entidades simples como líneas y puntos hasta geometrías complejas como curvas y cuádricas.

1.3 Ruido en imágenes.

El ruido en una imagen es la información no deseada que contamina la imagen o la variación en el nivel de gris que sufre un píxel debido a la aportación lumínica de la escena. Las imágenes pueden contener ruidos provocados por fuentes ruidosas, como son sensores ópticos, eléctricos, mecanismos de apertura en cámaras fotográficas, y también debido a la transmisión de dichas

imágenes a través de un canal físico (Hernández, 2002). Existen dos tipos de ruido en imágenes, el ruido sal y pimienta y el ruido gaussiano.

1.3.1 Ruido de sal y pimienta o ruido aleatorio

Consiste en un gran número de ocurrencias aleatorias de píxeles completamente blancos y completamente negros. Es un ruido que aparece muchas veces producido por interferencias atmosféricas o por acciones hechas por el humano. Corresponde a la siguiente fórmula (Cepero Ruz, y otros, Mayo 2010):

$$\bullet \quad f_{sp} = \begin{cases} f(x,y) & r < l_1 \\ 255 & l_1 < r < l_2 \\ 0 & r > l_2 \end{cases} \quad (1)$$

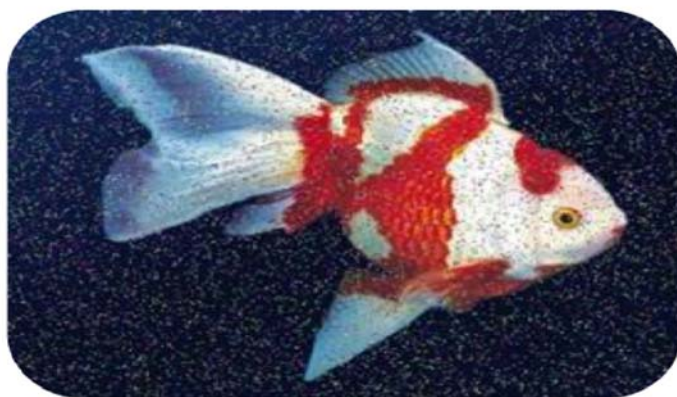


Figura 1. Imagen con ruido Aleatorio (Cepero Ruz, y otros, Mayo 2010).

El ruido tipo sal y pimienta (Salt and Popper) es un tipo especial de ruido impulsivo, que presenta un punteado característico que lo hace fácilmente identificable y que se elimina mediante el empleo del filtro de mediana. (Cepero Ruz, y otros, Mayo 2010) (Hernández, 2002).

Se caracteriza por la aparición de píxeles con valores arbitrarios normalmente detectables porque se diferencian mucho de sus vecinos más próximos. La distribución viene dada por:

$$\bullet \quad \begin{cases} & \text{si } r(x,y) < p/2 \\ L - 1 & \text{si } p/2 \leq r(x,y) < p \\ f(x,y) & \text{si } r(x,y) \geq p \end{cases} \quad (2)$$

1.3.2 Ruido dependiente de la señal o ruido gaussiano

Este tipo de ruido es producido generalmente por los propios sensores fotoeléctricos o también debido al grano de la película de una fotografía o cinta de video. Es un ruido dependiente de la señal cuya densidad de probabilidad responde a la Distribución Gaussiana (Cepero Ruz, y otros, Mayo 2010)

El ruido Gaussiano tiene un efecto general en toda la imagen, es decir, la intensidad de cada píxel de la imagen se ve alterada en cierta medida con respecto a la intensidad en la imagen original. Por el contrario, se observa que el ruido impulsivo tiene un efecto más extremo sobre un subconjunto del total de píxeles de la imagen. Un tanto por ciento de los píxeles de la imagen toma arbitrariamente el valor extremo 0 o 255. (Cepero Ruz, y otros, Mayo 2010)

1.3.3 Reducción de ruido.

La reducción de ruido es el proceso al que se somete una imagen con la finalidad de reducir la información ruidosa que esta presenta. En la literatura consultada no se conoce forma automatizada de detectar el ruido presente en una imagen. Solo se logra mediante la apreciación visual del ojo humano. El mismo trata métodos para resaltar o suprimir, de forma selectiva, información contenida en una imagen a diferentes escalas espaciales, para destacar algunos elementos de la imagen, o también para ocultar valores anómalos (Hernández, 2002).

1.4 Realce y Mejora o Filtrado.

En esta etapa se utilizan métodos para resaltar o suprimir, de forma selectiva información contenida en una imagen a diferentes escalas espaciales, para destacar algunos elementos de la imagen, o también para ocultar valores anómalos. El rol fundamental de este paso es la transformación más adecuada de la imagen facilitando los posteriores procedimientos en el análisis de la escena. Esta etapa comprende varios tipos de procedimientos como son el mejoramiento y restauración. El proceso de filtrado consiste en la aplicación a cada uno de los píxeles de la imagen de una matriz de filtrado de tamaño $N \times N$ (generalmente de 3×3 , aunque puede ser mayor) compuesta por números enteros y que genera un nuevo valor mediante una función del valor original y los de los píxeles circundantes (Márquez Álvarez, y otros, 2009) (Gonzalez, y otros, 2002).

1.4.1 Técnicas de pre-procesamiento de las imágenes de iris.

Las imágenes obtenidas a partir de los diferentes dispositivos de captura, son imágenes que se encuentran en el espectro visible y en formato 2D. Con el objetivo de delimitar cada estructura de interés o disminuir el ruido presente en la imagen se utilizarán filtros según el resultado que se desea obtener.

Los filtros que se utilizan actualmente son los de paso bajo (suavizan la imagen), de paso alto (aumentan el contraste), los filtros direccionales (detectan en la imagen estructuras que siguen una determinada dirección) y los de detección de bordes (permiten identificar y aislar objetos con propiedades homogéneas dentro de la imagen) (Tristan, 2011).

Existen diferentes algoritmos dedicados a reducir el ruido en las imágenes digitales. Entre los más frecuentes se encuentran: (Cepero Ruz, y otros, Mayo 2010)

- **Filtros lineales:** Convolución de una imagen con una máscara de convolución.
- **Filtros no lineales:** Operación no lineal con los píxeles del entorno de vecindad.

1.4.2 Filtros de convolución o Filtros lineales.

Se denomina convolución a una función, que, de forma lineal y continua, transforma una señal de entrada en una nueva señal de salida.

$$g(i, j) = \sum_{k=-n/2}^{n/2} \sum_{l=-n/2}^{n/2} h(k, l) f(i - k, j - l) \quad (3)$$

Para cada píxel en la imagen de entrada la máscara se ubica en el borde superior de este.

- Los valores de cada píxel en la imagen de entrada se multiplican por los pesos de la máscara correspondiente, todos estos valores se suman.
- El píxel central del entorno de vecindad toma el valor de esta suma.
- Este proceso se repite para toda la imagen, dando como salida una imagen nueva.

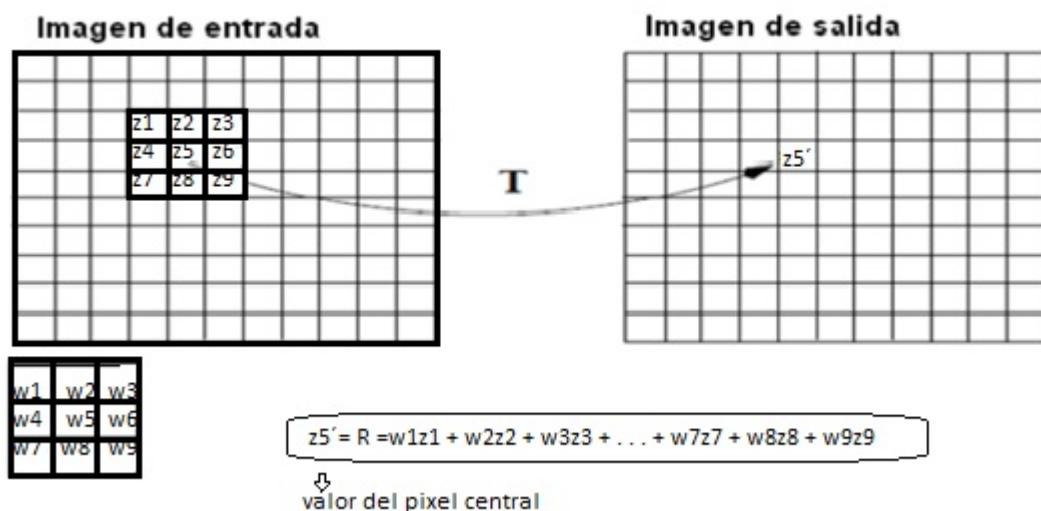


Figura 1 : Proceso de convolución de píxeles (Nube, 2010).

El proceso de convolución presupone un problema, debido a que la máscara inicialmente se sale de la imagen. Esto es solucionado con la llamada "banda basura". Para una máscara de tamaño $n \times n$, donde n es un número impar, la banda basura es $(n-1)/2$. Al aplicar el algoritmo con una máscara de tamaño 3×3 se empezaría por el píxel (1,1) de la imagen de entrada (Cepero Ruz, y otros, Mayo 2010).

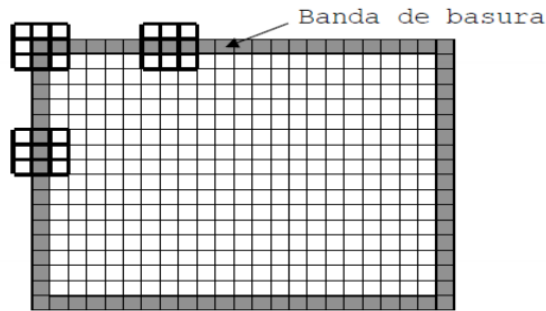


Figura 2 : Banda basura para una máscara de 3x3 (Cepero Ruz, y otros, Mayo 2010).

El resultado del proceso de convolución es un filtrado paso bajo, donde el ruido es eliminado con más o menos intensidad según el tamaño de la máscara. La aplicación de la máscara sobre la imagen suaviza las transiciones más bruscas como pueden ser los bordes de los objetos. Este efecto ocurre debido a que algunas regiones de la imagen poseen componentes en alta frecuencia, que son eliminados tras realizar la operación tipo kernel paso bajo.

No obstante, las técnicas de reducción de ruido presentan varios inconvenientes, en ocasiones se observa una especie de pérdida de resolución de la imagen y/o reducción del contraste.

La reducción de ruido es mayor cuanto más grande sea el orden de la ventana (máscara). Por el contrario, la resolución de la imagen (calidad) disminuye con dicho orden (Márquez Álvarez, y otros, 2009).

Estos tipos de métodos tienen como problema un alto coste computacional, debido al gran número de cálculo matemático y los tiempos de computación que necesita para llegar a un resultado.

1.4.3 Filtro de la media.

Consiste en sustituir cada píxel por la media aritmética de los puntos que tiene alrededor (incluido el mismo). Se toma una ventana de $N \times N$ puntos, donde el punto a sustituir es el central, se toma la media de la suma de los valores de los píxeles presentes en la ventana y se sustituye el píxel en la nueva imagen por el valor obtenido. El efecto del filtro aumenta a medida que lo hace el tamaño de la ventana y los resultados que produce son equivalentes a una difuminación de la imagen, produciendo un suavizado de los gradientes en toda la imagen. Es de gran utilidad para eliminar ruido producido por un muestreo deficiente o una transición ruidosa. (Cepero Ruz, y otros, Mayo 2010)

Es bueno saber que la media del ruido en un entorno de vecindad es cero, si calculamos la media eliminamos el ruido. La imagen es constante en un entorno de vecindad, por lo que su valor es igual a la media. Pero esta característica no se cumple en los bordes. Se logra haciendo convolucionar la imagen con alguna de las siguientes máscaras:

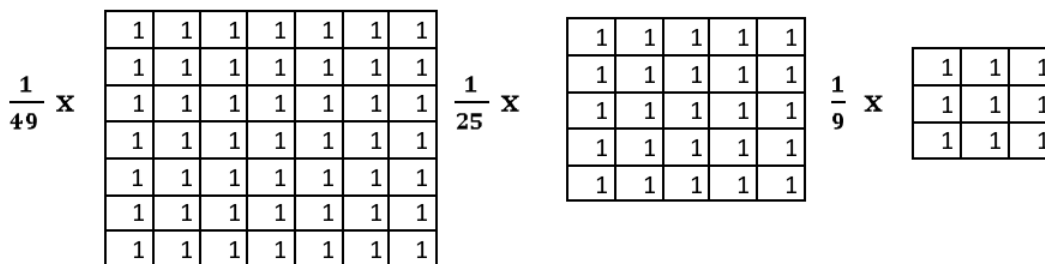


Figura 3 : Máscaras más utilizadas en el filtro de la media (Tristan, 2011).

También se puede implementar promediando los valores del entorno de vecindad. Este algoritmo es útil en la reducción de ruido gaussiano y aleatorio, aunque es más apreciable en el primer caso. Debido a sus características es susceptible a difuminar los bordes y a disminuir la nitidez, con la consecuente pérdida de detalles. Mientras mayor sea la cantidad de valores a promediar más ruido se reducirá, pero también habrá una mayor pérdida de detalles. En las siguientes imágenes se observa lo anteriormente planteado.

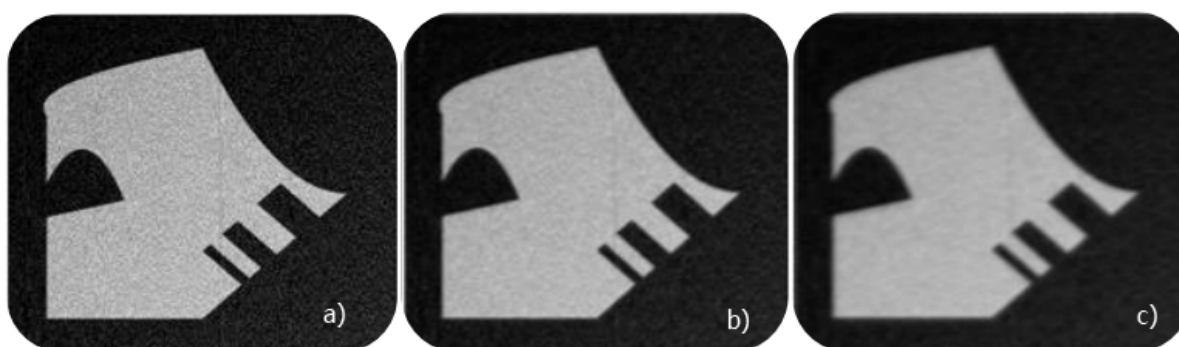


Figura 4 :a) Imagen con ruido gaussiano b) Imagen resultante del algoritmo de la media con una máscara de 3x3 c) Imagen resultante del algoritmo de la media con una máscara de 5x5 (Cepero Ruz, y otros, Mayo 2010).

1.4.4 Filtro de la mediana.

Una de las técnicas no lineales más utilizadas para el procesamiento de imágenes, en cuanto a la reducción de ruido, es el filtrado de la mediana. La idea es que, en un conjunto de píxeles cercanos, valores muy alejados del resto, serán valores que corresponden a píxeles ruidosos. La operación consiste en analizar una matriz de píxeles de $N \times N$ y reemplazar el píxel central por el valor de la mediana de todos ellos. Y por supuesto repetir el algoritmo recorriendo toda la imagen (Cepero Ruz, y otros, Mayo 2010).

Aplicando este método se deben escoger ventanas de $N \times N$ con N impar, para tener bien diferenciado el píxel central. Tamaños habituales son ventanas de 3x3, de 5x5 y de hasta 7x7. La elección del tamaño de ventana es importante, puesto que un valor pequeño puede no eliminar bien el ruido, mientras que un valor demasiado alto es capaz de distorsionar la imagen. Para la

elección de la ventana, en definitiva, no existe una regla fija, sino que se trata de escoger el orden que mejor resultados dé con una determinada imagen. En general, se suele decir que un tamaño de ventana es bueno si el número de píxeles ruidosos dentro de la ventana es menor que la mitad de píxeles de la ventana (Tristan, 2011).

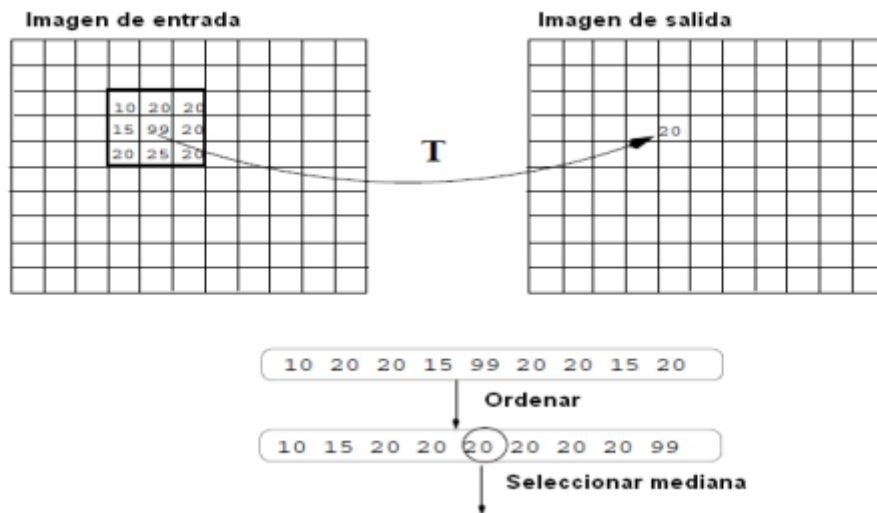


Figura 5 : Proceso del algoritmo de la mediana (Nube, 2010).

Este filtro responde mucho mejor ante escalones de señal. O sea, un filtro de suavizado paso bajo es capaz de distorsionar un borde muy pronunciado, mientras que el filtro de mediana respeta los bordes. Elimina casi por completo el ruido de tipo impulsivo de las imágenes, mientras que un filtro de suavizado lineal no lo elimina, sino que lo difumina: por ejemplo, un punto blanco sobre fondo negro se convertiría en varios puntos gris oscuro.

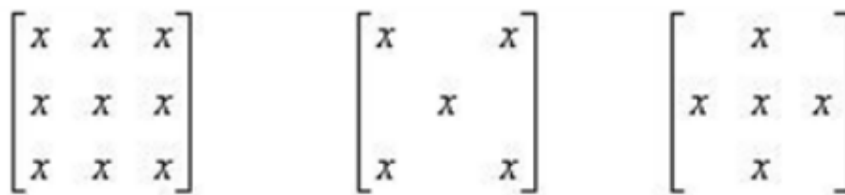


Figura 6 : Máscara cuadrada, en X y en cruz (Cepero Ruz, y otros, Mayo 2010).

La ventana cuadrada es la que mayor grado de reducción de ruido provoca, pero también es la que más fácilmente distorsiona la imagen.

- La ventana en cruz es buena para imágenes con muchos bordes.
- La ventana en equis es un término medio entre las dos anteriores.

El filtro de la mediana es útil a la hora de eliminar manchas localizadas de una imagen fotográfica, por ejemplo, deteriorada con el tiempo. También sirve en imágenes capturadas de la televisión con "nieve": ese molesto efecto que aparece si hay interferencias en el aparato de televisión, o bien si la relación señal a ruido no es muy buena. En ese caso de nuevo el filtro de la mediana es capaz

de eliminar esas manchas de una forma eficiente sin afectar demasiado al resto de la imagen. (Cepero Ruz, y otros, Mayo 2010)



Figura 7 :a) Imagen con ruido aleatorio b) Imagen luego de aplicar algoritmo de la mediana (Cepero Ruz, y otros, Mayo 2010).

1.4.5 Histogramas.

El histograma es una representación gráfica de los niveles de grises definido: $[0, L-1]$ en forma de barras, donde la superficie de cada barra es proporcional a la frecuencia de los valores representados. En el eje vertical se representan las frecuencias, y en el eje horizontal los valores de los niveles de grises, normalmente, señalando las marcas de clase, es decir, la mitad del intervalo en el que están agrupados los datos (Serra, y otros, 2005).

Las técnicas de modificación del histograma de una imagen son útiles para aumentar el contraste de imágenes con histogramas muy concentrados, ya sean imágenes oscuras como claras (Serra, y otros, 2005). En la figura 8 se puede observar la representación gráfica de los niveles de gris de una imagen.

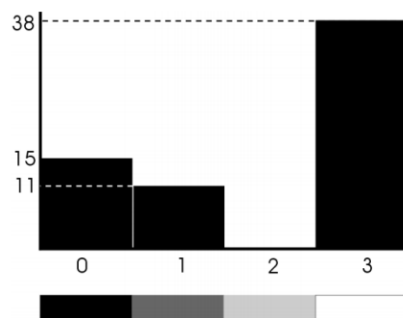


Figura 8 : Ejemplo de Histograma

1.4.5.1 Ecuación del histograma.

Es una forma de manipulación de histograma que reduce automáticamente el contraste en las áreas muy claras o muy oscuras de una imagen. También expande los niveles de gris a lo largo de todos

los intervalos. Consiste en una transformación no lineal que considera la distribución acumulativa de la imagen original, para generar una imagen resultante cuyo histograma será aproximadamente uniforme (Felip León, y otros, 2005).

La ecualización del histograma es un método que consiste en la variación del histograma de una zona de una imagen (o de la imagen entera) a otra de otra forma (Felip León, y otros, 2005).

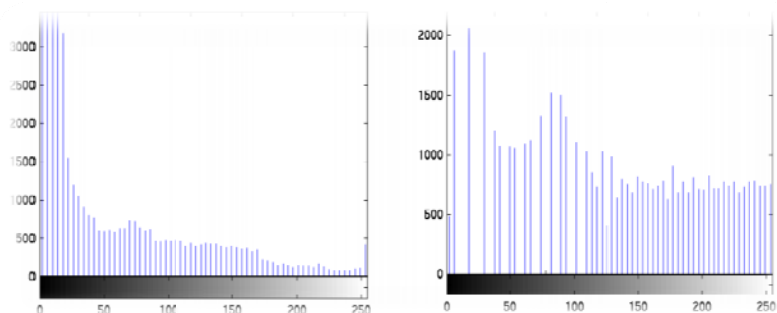


Figura 9 : Ecualización del Histograma (León, Moya, & Catalán, 2005)

1.5.5.1.2 *Ecualización de Histograma Adaptativo.*

Con el objetivo de reducir el tiempo de ejecución de la ecualización de histograma, se realiza esta técnica solo en algunos píxeles (con sus respectivas regiones de contexto). Los factores que influyen en la reducción del tiempo de ejecución son el número de divisiones en la imagen. Cuanto mayor sea el tamaño de la región del contexto se obtendrán por un lado menores tiempos de ejecución y a su vez menos realce de los detalles (Felip León, y otros, 2005).

Las vecindades de la zona local son combinadas utilizando interpolación bilineal para eliminar artificialmente propiedades inducidas. El contraste, especialmente en áreas homogéneas, puede ser limitado para evitar incrementar cualquier ruido que se pueda presentar en la imagen (Fuente-Bermudez, y otros, 2015).

Este método es muy bueno para imágenes que posean bajo contraste, su aplicación permite aumentar el contraste de la imagen de tal forma que se observen detalles que antes no se observaban.

1.4.6 *Morfología Matemática. Operaciones de Minkowsky.*

La morfología matemática es una herramienta para extraer componentes de un objeto de interés, útiles para representar y describir la forma de una región, tales como fronteras y esqueletos. La morfología matemática se ha utilizado con gran éxito en el procesamiento de imágenes. El lenguaje utilizado es la teoría de conjuntos, dado que una imagen binaria se representa por medio de un conjunto (Diaz de León, et al., 2009).

Las operaciones básicas de la teoría de conjuntos asociadas a una imagen son:

Unión: dado un conjunto X , existe otro conjunto cuyos miembros son exactamente los miembros de los miembros de X (Climent Vidal, 2005).

La **intersección** de un conjunto no vacío es el máximo conjunto que está contenido a cada uno de los conjuntos que le pertenecen o, lo que es equivalente, es el extremo inferior, respecto de la inclusión, de los conjuntos del conjunto en cuestión (Climent Vidal, 2005).

El **complemento** de un conjunto $A \subseteq X$ respecto a X (conjunto universo) o simplemente complemento, denotado por A^c , se define como el conjunto formado por aquellos elementos de X que no pertenecen a A (Díaz de León, et al., 2009).

Las bases teóricas de la morfología matemática se deben al científico alemán nacido en Rusia, Hermann Minkowski (1864-1909). Estas bases teóricas se centran en la suma de Minkowski y la resta de Minkowski (Díaz de León, et al., 2009).

La **suma de Minkowski** de $A \subseteq X$ y $B \subseteq X$, denotada por $A \oplus B$ es el conjunto que resulta de sumar cada elemento de A con cada elemento de B . A la suma de Minkowski también se le conoce como dilatación de un conjunto A por un conjunto B (Díaz de León, et al., 2009).

Sean $A \subseteq X$ y $B \subseteq X$ dos conjuntos. La **erosión** de A por B , denotada por $A \ominus B$, se define como la **resta de Minkowski** de A y B (Díaz de León, et al., 2009).

Las operaciones de Minkowski dan lugar a las operaciones básicas de la morfología matemática (iniciada por George Matheron a mediados de los 60's), la dilatación y la erosión. A partir de estas dos operaciones se crean las demás operaciones morfológicas (Díaz de León, et al., 2009).

Según (Platero Dueñas, 2009) la **erosión** es el resultado de comprobar si el elemento estructurante B está completamente incluido dentro del conjunto. Cuando esto no ocurre, el resultado de la erosión es el conjunto vacío.

La aplicación de esta transformación elimina grupos de píxeles de menor tamaño que el elemento estructurante y su resultado es uno si el elemento estructurante queda incluido dentro del subconjunto de la imagen binaria analizada y es cero cuando no está totalmente incluido en el subconjunto (Benalcazar Palacios, 2008).

La dilatación es la operación dual de la erosión. Su aplicación añade todos los puntos del fondo que tocan el borde de un objeto, es decir, rellena contrastes en los que no quepa el elemento estructurante. El resultado de la aplicación de esta operación es uno si el elemento estructurante contiene algún píxel dentro del subconjunto de la imagen binaria analizada y es cero cuando no encuentra ningún píxel en dicho subconjunto (Benalcazar Palacios, 2008).

En un gran porcentaje de procesos de análisis de conjuntos (imágenes binarias) normalmente se realizan procesos iterativos del tipo erosión seguida de una dilatación y una dilatación seguida de

una erosión. Estos procesos se denominan apertura y cierre respectivamente, y aunque se podría asumir que en ambos procesos el resultado es el mismo, esto no ocurre (Díaz de León, et al., 2009).

Los efectos de una apertura sobre una región de interés son (Díaz de León, et al., 2009):

- Se eliminan islas de tamaño menor al elemento de estructura.
- Se eliminan picos o cabos más delgados que el elemento de estructura.
- Se rompen istmos cuya anchura sea menor que el diámetro del elemento de estructura.

Los efectos de un cierre sobre una región de interés son (Díaz de León, et al., 2009):

- Se rellenan los largos o los huecos de tamaño menor al elemento de estructura.
- Se rellenan rajaduras o golfos más delgados que el elemento de estructura.
- Se funden estrechos cuya anchura sea menor al diámetro del elemento de estructura.

El efecto de las operaciones de erosión, dilatación, apertura y cierre pueden ser observados en la imagen siguiente (Gonzalez, et al., 2002):

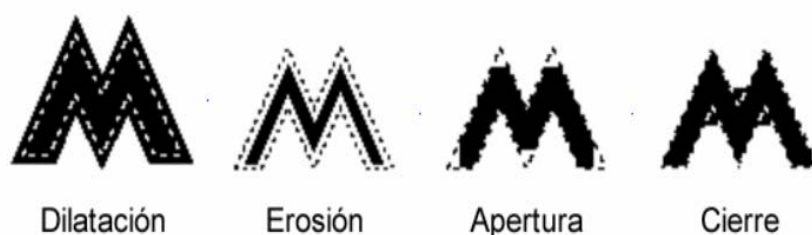


Figura 10: Resultado de las operaciones de erosión, dilatación, apertura y cierre (Benalcazar Palacios, 2008).

La aplicación de filtros morfológicos a imágenes binarizadas y en niveles de grises es de gran impacto en las etapas de procesado, de segmentación y postprocesado o en fases de mayor información visual. Estas transformaciones son usadas para resaltar o remover selectivamente las estructuras u objetos de una imagen de forma tal que se reduzca la información a procesar en las siguientes fases. Los filtros morfológicos tienen múltiples aplicaciones en la restauración de imágenes, detección de bordes, análisis de texturas, aumento del contraste y hasta en la compresión de imágenes (Díaz de León, y otros, 2009).

1.5 Técnicas de mapeo de bordes

Los bordes contienen mucha información de la imagen, estos determinan la posición de los objetos, su forma y tamaño. Los bordes se encuentran en zonas de la imagen donde el nivel de intensidad cambia bruscamente, cuanto más rápido se produce el cambio de intensidad es más fuerte el borde.

1.5.1 Operadores de gradiente común

Los operadores de gradiente común encuentran bordes horizontales y verticales. En esta clasificación se encuentran los operadores de Prewitt, Sobel, Roberts y Frei-Chen. En estos operadores la detección de bordes se realiza en dos pasos, en el primero se aplica una máscara para buscar bordes horizontales, y en el segundo paso se buscan los verticales. El resultado final es la suma de ambos. En la siguiente imagen se observan las máscaras de convolución asociadas a los operadores de Prewitt, Sobel, Roberts y Frei-Chen:

	Roberts	Prewitt	Sobel	Frei-Chen
Máscaras para bordes horizontales	$\begin{matrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$	$\begin{matrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{matrix}$	$\begin{matrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{matrix}$	$\begin{matrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & 1 \end{matrix}$
Máscaras para bordes verticales	$\begin{matrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$	$\begin{matrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{matrix}$	$\begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$	$\begin{matrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -2 & -\sqrt{2} & -1 \end{matrix}$

Tabla 1 : Máscaras asociadas a los operadores de gradiente común

1.5.2 Canny

El operador de detección de bordes Canny fue desarrollado en 1986 basado en un algoritmo de múltiples fases para detectar un amplio rango de bordes. Este detector cuenta con tres fases que se detallan a continuación (Tejedor Gómez, 2009):

- Obtención del gradiente.
- Supresión no máxima al resultado del gradiente.
- Cierre de contornos abiertos.

Obtención del gradiente: en esta etapa se realiza un suavizado a la imagen basado en un filtro gaussiano y se obtiene el gradiente de la imagen suavizada.

Supresión no máxima al resultado del gradiente: se seleccionan dos valores umbrales, uno superior y otro inferior. Los píxeles cuyo valor de la magnitud del gradiente está por encima del umbral superior se consideran como partes de bordes y se les llama bordes fuertes, mientras que los que tienen una respuesta por debajo del umbral inferior no se consideran bordes. Los que están entre ambos valores se les llama bordes débiles.

Cierre de contornos abiertos: el algoritmo incorpora como bordes aquellos bordes débiles que están conectados a bordes fuertes, es decir, que están dentro de un entorno 3x3.

1.5.3 Algoritmo Iterativo

Utiliza como parámetro de entrada una imagen binarizada. Su funcionamiento consiste en recorrer de forma iterativa todos los píxeles de la imagen con valor 0, y de estos píxeles los que tengan un vecino 255 serán considerados como borde asignándole el valor 255, todos los demás píxeles serán llevados a color 0.

1.6 Técnicas de segmentación de imágenes de iris.

Consiste en la separación del iris del resto de las partes del ojo, la cual incluye la detección de bordes y eliminación de párpados, pestañas y reflexiones especulares (Osorio Roig, y otros, 2016).

La segmentación es el proceso mediante el cual la imagen se divide en sus partes constituyentes; es decir, es la separación de los objetos o regiones de interés que se encuentran dentro del registro y en el respectivo posterior descarte de los elementos que no contribuyen en cuanto a información relevante o son considerados como ruido.

1.6.1 Operador integro-diferencial.

Propuesto en 1992, el operador fue creado tomando como premisa que la pupila e iris tienen forma circular. Este funciona con tres parámetros fundamentales: el radio, y las coordenadas x_0 y y_0 del centro del iris y su ecuación es la siguiente [5][6]:

$$\max_{(r,x_0,y_0)} \left| G_\sigma(r) * \frac{\partial}{\partial r} \phi_{r,x_0,y_0} \frac{I(x,y)}{2\pi r} ds \right|_{(5)}$$

G_σ : Función de suavizado

$I(x, y)$: Representa la imagen del ojo

r : Radio del límite límbico del ojo

x_0 : Coordenada x en la imagen del centro del límite límbico del ojo

y_0 : Coordenada y en la imagen del centro del límite límbico del ojo

El operador busca el camino circular en el que hay cambio máximo de los valores de píxel, por la variación del radio r y el centro (x_c, y_c) y la posición del contorno circular, para ello utiliza la ecuación descrita anteriormente [5]. Este método se aplica de manera iterativa reduciendo progresivamente la función de suavizado con el fin de lograr la localización exacta de la pupila y del iris. También puede ser utilizado para la localización de párpados, el resultado de su aplicación se puede observar en la siguiente figura:

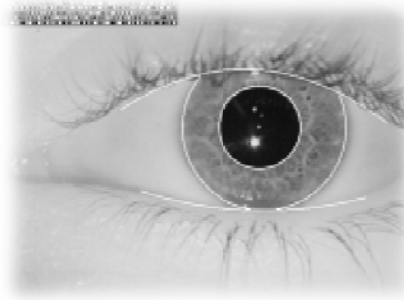


Figura 11 : Localización de párpados, iris y pupila mediante el uso del operador integro diferencial.

1.6.2 Técnicas que mejoran la efectividad del método (Tomé González, 2008):

- Ecuación del histograma: se mejora el contraste entre cada región del ojo, contribuyendo así a la correcta aplicación del algoritmo de segmentación (Tomé González, 2008).
- Binarización: La aplicación de un umbral en una imagen antes de la ejecución del operador permite optimizar el contraste entre las regiones. Este proceso tiene como desventaja: es muy dependiente del umbral elegido, sin embargo presenta una ventaja importante: la inexistencia de parámetros definidos por el usuario para la tarea de segmentación (Tomé González, 2008).

1.6.2 Transformada Circular de Hough.

La transformada de Hough es un algoritmo de *Computervision* estándar que se puede utilizar para determinar los parámetros de objetos geométricos simples, como líneas y círculos, presente en una imagen. La transformada de Hough circular puede emplearse para deducir el radio y el centro de coordenadas de las regiones de la pupila y el iris (Tejedor Gómez, 2009).

El procedimiento que sigue el algoritmo es el siguiente: en primer lugar, se genera un mapa de bordes mediante el cálculo de los primeros derivados de los valores de intensidad en una imagen del ojo y a continuación se aplica un método de umbralización al resultado. Desde el mapa de bordes, se emiten votos en el espacio Hough para cada círculo existente por cada punto del borde. Al final del proceso se selecciona el círculo más votado.

Los parámetros x_c y y_c , centros de coordenadas, y el radio r , son capaces de definir cualquier círculo según la ecuación.

$$(7)$$

1.7 Normalización del iris.

La normalización se basa en el modelo laminar propuesto por Daugman. Este proceso implica "desenvolver" la imagen del iris detectado, es decir, convertir la imagen del iris de coordenadas cartesianas a polares. Para este proceso se considera al centro de la pupila como punto de referencia, y se utiliza una fórmula de re-mapeo para realizar la conversión de coordenadas. El procedimiento consiste en la selección de una cantidad determinada de puntos a lo largo de cada

línea radial, que es conocido como resolución radial, y así mismo determinar la cantidad de líneas radiales que van a través de la región del iris, conocida como resolución angular (Caicedo Marmolejo, y otros, 2012).

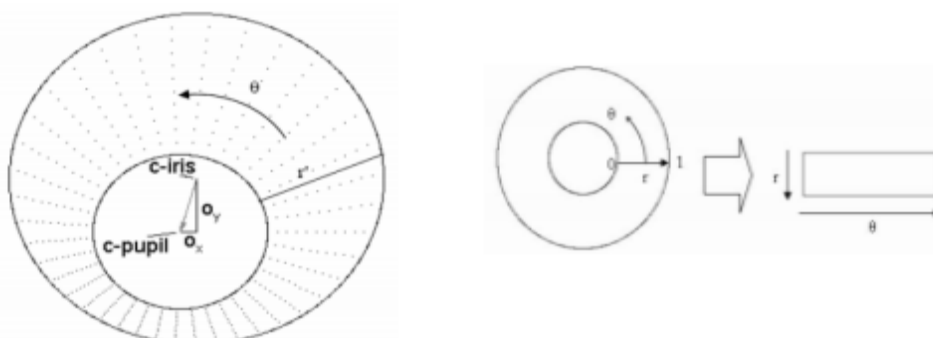


Figura 12 : Normalización del iris (Caicedo Marmolejo, y otros, 2012).

Dado que la pupila no siempre posee el mismo centro que el iris, se necesita una fórmula de reasignación para cambiar la escala de los puntos alrededor del círculo, dicha fórmula está dada por:

$$r' = \sqrt{\alpha\beta} \pm \sqrt{\alpha\beta^2 - \alpha - r_1^2} \quad (6)$$

$$\alpha = o_x^2 + o_y^2 \quad (7)$$

$$\beta = \cos(\pi - \arctan\left(\frac{o_y}{o_x}\right) - \theta) \quad (8)$$

Donde el desplazamiento del centro de la pupila con respecto al centro del iris está dado por o_y y o_x , y así mismo r' es la distancia entre el borde de la pupila y el iris dado un ángulo θ , r_1 es el radio del iris (Caicedo Marmolejo, y otros, 2012).

1.8 Metodología de Desarrollo

Según (Pedreira, 2007) una metodología de desarrollo de software es “un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software”. Una metodología es la que define Quién debe hacer Qué, Cuándo y Cómo.

Seleccionar una metodología apropiada, resulta en ocasiones un factor esencial para obtener un software con la calidad esperada. En la bibliografía consultada no se conoce la existencia de una metodología global aplicable a todos los proyectos y su selección depende de las características del mismo (Pressman, 2002).

El grupo de investigación AIRI definió como metodología de desarrollo XP debido a que:

- El equipo de desarrollo, es de dos personas.
- Se necesitan versiones funcionales de la solución a corto plazo.

-
- La dificultad para un equipo de desarrollo pequeño para adoptar una metodología robusta a causa de la cantidad de documentación generada y la alta resistencia a los cambios durante el desarrollo.
 - La existencia de la presencia constante del cliente en el proceso de desarrollo.

La metodología de software XP, programación extrema, está comprendida dentro de las metodologías de desarrollo de software ágiles y trata de reducir la complejidad de software orientando el trabajo directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción, teniendo como principal peculiaridad la presencia, a tiempo completo en el desarrollo (Penadés, 2006), de un representante del cliente. Esta metodología se basa en la comunicación, la claridad y la reutilización continua de código. Tiene como objetivo fundamental la satisfacción del cliente.

De manera general se puede expresar que es recomendable usar XP debido a que los requisitos tienen altas probabilidades de cambiar con el tiempo (porque el cliente no tiene claro lo que quiere, o porque el cambio de requisitos está ligado al dominio del problema a resolver), el proyecto cuenta con un grupo pequeño de programadores.

Características de XP (Beck, 1999):

- XP es una metodología “liviana” que no tiene en cuenta la utilización de elaborados casos de uso, la exhaustiva definición de requerimientos y la generación de una extensa documentación.
- XP tiene asociado un ciclo de vida y es considerado a su vez un proceso.
- La tendencia de entregar software en espacios de tiempo cada vez más pequeños con exigencias de costos reducidos y altos estándares de calidad.
- XP define Historias de Usuario como base del software a desarrollar, estas historias las escribe el cliente y describen escenarios sobre el funcionamiento del programa, a partir de las historias de usuario y de la arquitectura definida se crea un plan de liberaciones entre el equipo de desarrollo y el cliente.

XP consta de cuatro fases (Beck, 1999):

- Planificación
- Diseño
- Codificación
- Pruebas

1.9 Herramientas y lenguajes usados en el desarrollo.

En el momento de desarrollar un producto de software es necesario aplicar una metodología que guíe el proceso de desarrollo de dicho software para que reúna todos los requisitos y calidad esperados por el cliente.

Las herramientas garantizan la velocidad de respuesta y estabilidad de los resultados que se esperan obtener de un sistema. La combinación de diferentes herramientas permite soportar el proceso de desarrollo de software.

1.9.1 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 (Suárez, 2015).

Para el desarrollo de esta aplicación el grupo de investigación AIRI definió como lenguaje de programación Java, por ser un lenguaje puro Orientado a Objetos, lo que permite gran reusabilidad; independencia de la plataforma, significa que programas escritos en el lenguaje Java pueden ejecutarse igualmente en cualquier tipo de hardware. Además, posee un recolector automático de basura (*garbage collector*) lo cual independiza al programador de tener que administrar memoria solicitada dinámicamente de forma manual (Suaza, 2015).

Dentro de las características fundamentales de Java se encuentran (Fernández, 2004):

- 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.

1.9.2 Entorno integrado de software.

Como entorno integrado de desarrollo (IDE) se utilizó Netbeans, ya que es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. Es un producto libre y gratuito sin restricciones de uso. (Fernández, 2004). Además es un entorno de desarrollo disponible para varios sistemas operativos como Windows, Mac, Linux y Solaris (Fernández, 2004). NetBeans consta de un IDE de código abierto con gran diversidad de funciones escrito con el lenguaje de programación Java y una plataforma para aplicaciones de cliente enriquecido que se puede utilizar como marco genérico para crear cualquier tipo de aplicación, tanto empresariales como de escritorio y móviles.

1.9.3 Visual Paradigm for UML

Es una herramienta muy completa, fácil de usar y con soporte multiplataforma. Su uso es sencillo para la creación de todo tipo de diagramas UML, para los que dispone de un número considerable de estereotipos, permitiendo mayor entendimiento de los mismos (INEI, 2015). Según lo especificado en (Visual Paradigm, 2000) permite visualizar el flujo central detallado de cada proceso mediante diagramas, posibilitando la obtención de los mismos definidos por la metodología escogida, entre ellos el diagrama de clase y el modelo de datos.

Algunas características que presenta son (Visual Paradigm, 2000):

- Navegación intuitiva entre la escritura del código y su visualización.
- Generador de informes en formato PDF/HTML.
- Documentación automática Ad-hoc.
- Interoperabilidad con modelos UML.

Se opta por utilizar la herramienta CASE en su versión 5.0 para trabajar durante todo el desarrollo de la aplicación existente.

1.9.4 Matlab

Utiliza su propio lenguaje de computación técnica de alto nivel (lenguaje M) y un entorno interactivo para desarrollo de algoritmos, visualización de datos, análisis de datos y cálculo numérico. Con MATLAB, se pueden resolver problemas de cálculo técnico más rápido que con otros lenguajes de programación tradicionales, tales como C, C++ y FORTRAN. (García de Jalón, y otros, diciembre 2005)

Puede ser usado en una amplia gama de aplicaciones que incluyen procesamiento de señales e imágenes, comunicaciones, diseño de sistemas de control, sistemas de prueba y medición,

modelado y análisis financiero y biología computacional. Los conjuntos de herramientas complementarios (colecciones de funciones de MATLAB para propósitos especiales, que están disponibles por separado) amplían el entorno de MATLAB permitiendo resolver problemas especiales en estas áreas de aplicación (García de Jalón, y otros, diciembre 2005). Se definió la utilización de este lenguaje en su versión 2013 para el proceso de prueba de algoritmos.

1.9.5 *Toolbox de procesamiento digital de imágenes de Matlab*

Está formada por un conjunto de funciones específicas que amplían las capacidades de Matlab para el desarrollo de aplicaciones y algoritmos en el campo del procesado y análisis digital de imágenes (Laorden Fiter, 2012).

1.10 Conclusiones del capítulo.

En el presente capítulo se definieron los conceptos fundamentales para entender el objetivo de la investigación en curso, así como las diferentes etapas presentes en el procesamiento de imágenes digitales y algunas de las técnicas actualmente usadas en este proceso. Como resultado de esta fase de la investigación se concluye que:

1. La revisión bibliografía evidenció que la investigación de imágenes de iris de espectro visible, es una línea de investigación reciente y en constante desarrollo, principalmente en el contexto de los sistemas de reconocimiento del iris.
2. Con el análisis de los principales conceptos relacionados con el objeto de estudio y su interrelación, se logró profundizar en la comprensión del problema planteado en la investigación.
3. El estudio de las herramientas mencionadas en la bibliografía consultada y el enfoque utilizado por cada una de ellas, permitió seleccionar las técnicas de realce y mejora: Erosión, Mediana, Escala de grises y Ecuilización de histograma adaptativo.
4. Las herramientas y tecnologías definidas por el grupo AIRI para el desarrollo de la herramienta permitirán obtener un software que segmente y normalice el iris de manera eficiente.

CAPÍTULO 2: Propuesta de Solución.

2.1 Introducción.

En este capítulo se describen los aspectos fundamentales de la solución propuesta de acuerdo a las fases de la metodología XP, teniendo en cuenta las funcionalidades del sistema, los patrones de diseño y los estándares de codificación.

2.2. Descripción general de la propuesta.

El proyecto “PANIRIS” tiene como objetivo construir una aplicación basada en el procesamiento de imágenes digitales. Este es un proyecto que sigue una de las líneas investigativas actuales del Cenatav (Centro de aplicaciones de Tecnologías Avanzadas). El proyecto consta de tres actividades fundamentales: Obtención de la imagen, Pre-procesamiento de la imagen, Segmentación y Normalización del iris.

El primer paso es obtener la imagen, este proceso puede ser realizado utilizando el dispositivo de captura del proyecto o cargando una imagen desde la computadora. Como segunda etapa se realiza un pre-procesamiento de la imagen obtenida en el primer paso, debido a que la calidad de la imagen influye directamente en la eficiencia del resultado final del proyecto. En esta etapa la imagen es sometida a un conjunto de filtros (Erosión, Mediana, CLAHE) con el objetivo de reducir al mínimo el ruido, mejorando visiblemente su calidad. Consecuentemente se obtiene una imagen con muy poco ruido y con un buen contraste. Como tercera etapa la imagen es sometida a un conjunto de técnicas (Transformada de Hough para el reconocimiento del iris y pupila) de segmentación, esta etapa se divide en las siguientes sub-etapas: (1) la binarización de la imagen, utilizando dos umbrales, uno para la pupila y el otro para el iris, (2) el mapeo de bordes y (3) la segmentación de la imagen. Obteniendo como resultados de esta fase una imagen con las regiones de pupila e iris señaladas. Como cuarta instancia se utiliza el resultado de la segmentación del iris y se normaliza. Como salida de este proceso se obtiene una imagen que contiene el iris normalizado. Como resultado de realizar estas fases se genera una imagen del iris normalizado que es almacenada en formato “png” o “bmp” en la computadora. Con la obtención de la imagen final es posible realizar un proceso de identificación de marcas en el iris y realizar análisis iridológicos sobre los mismos.

Para una mejor comprensión de lo anteriormente explicado se tiene el siguiente Diagrama de Procesos de Negocios, utilizándose el mismo por su fácil modelación e interpretación de los pasos que el sistema emplea.

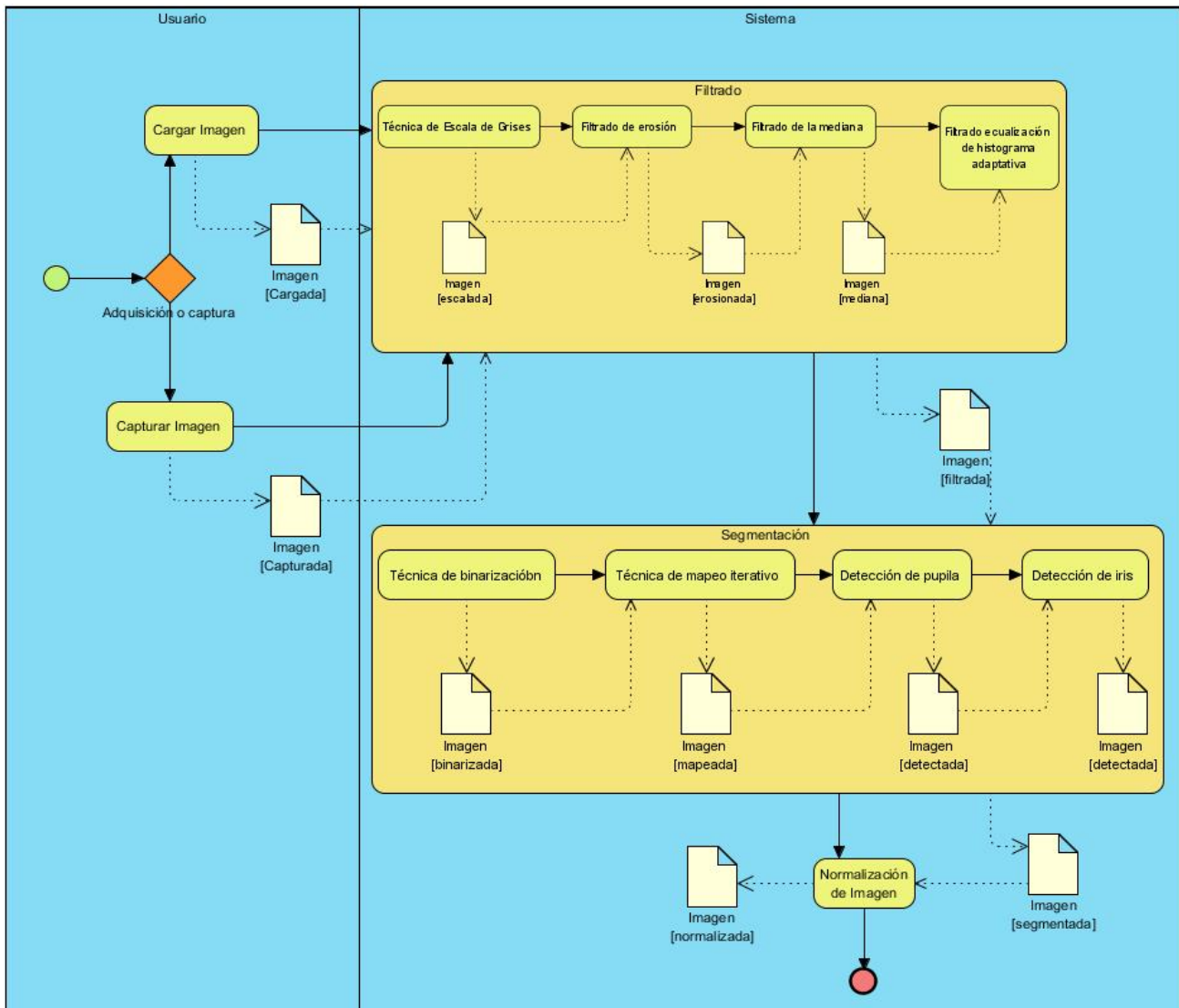


Figura 13 : Diagrama de Procesos de Negocio.

Captura de la Imagen

La imagen es tomada por el especialista, haciendo uso del iridoscopio conectado a una computadora que tiene el software que se encarga de realizar el resto de pasos. Para obtener la imagen del iridoscopio se utiliza la biblioteca **WebcamCapture v.0.3.10**, que permite utilizar el iridoscopio como una webcam.

Con el objetivo de optimizar la búsqueda del iris y la pupila se implementó en la pantalla de captura una mirilla que indica la posición donde debe estar ubicado el ojo.

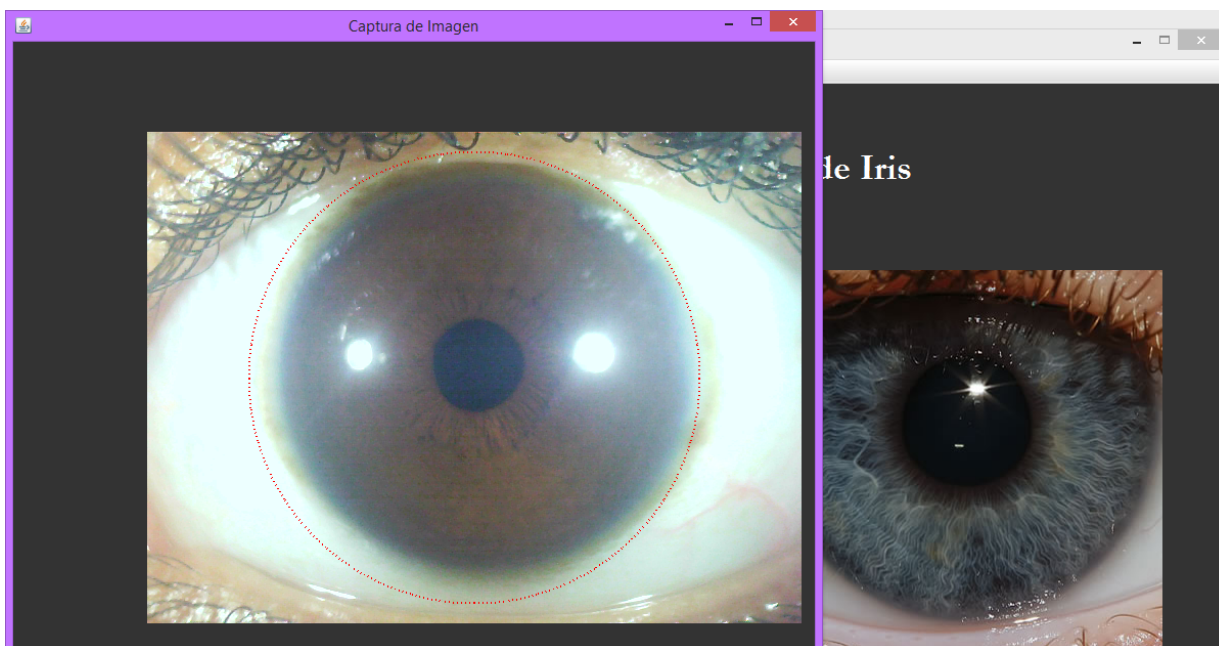


Figura 14 : Mirilla para la ubicación del iris

Selección de las técnicas de pre-procesado de imágenes.

Luego de realizar un estudio de las técnicas de realce y mejora de imágenes digitales, se escogieron los métodos más utilizados: Erosión, Gaussiano, Ecuilización de Histograma Adaptativo, Mediana, Media y Binarización con umbral óptimo para realizar pruebas en Matlab y escoger los de mejores resultados. Para la selección de las mejores técnicas de pre-procesamiento se realizó una tabla comparativa tomando como puntos a evaluar la eliminación del ruido sal y pimienta, su efectividad al hacerlo y la complejidad del mismo. A continuación se muestra esta comparación:

	Erosión	Mediana	Gaussiano	Clahe	Media
Entradas	Tamaño de Elemento Estructurante(EE)=20	máscara 3x3	Tamaño de máscara 3x3 , sigma =0.5	ClipLimit=0.1 Alpha=0.05	máscara 3x3
Eliminación de ruido (Sal y pimienta).	No	Si	No	Si	Si
Efectividad.	Media	Alta	Baja	Alta	Baja
Complejidad	Baja	Media	Media	Media	Media

Tabla 2: Selección de las técnicas de filtrado de imágenes

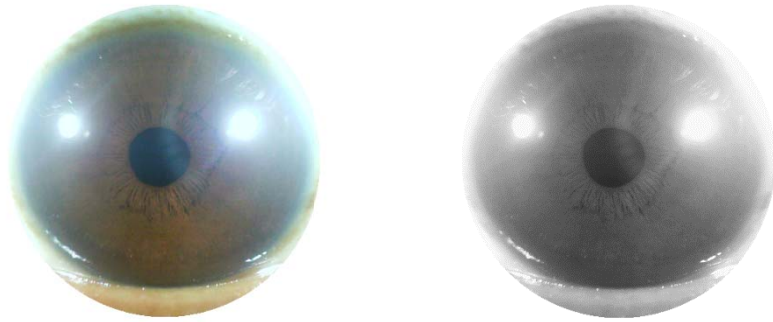


Figura 15: Imagen Original e imagen en escala de grises.

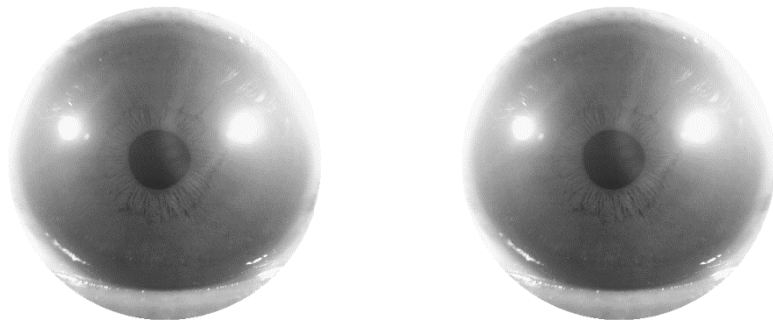


Figura 16: Imagen en escala de grises e imagen con filtro erosión.

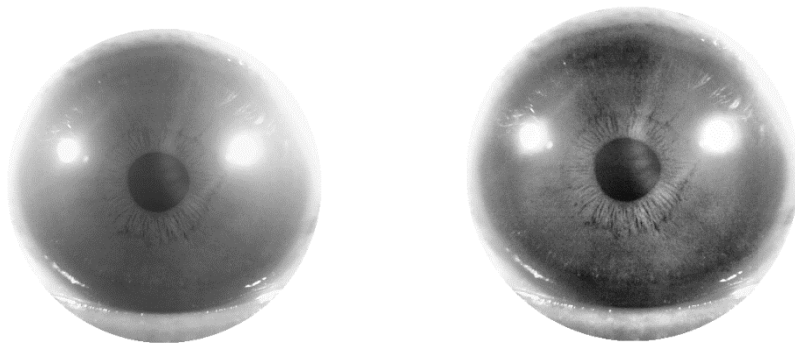


Figura 17: Imagen con filtro de erosión e imagen con filtro clahe.

En el caso de la técnica de Binarización se probó usar el umbral óptimo generado por el algoritmo de Otsu, pero este no obtuvo buenos resultados. Debido a esto se realizó un proceso de prueba de la técnica de binarización asignando umbrales de forma manual para la binarización de iris y pupila comprobando los resultados para cada caso. Definiéndose un umbral de 80 para la pupila y uno de 210 para el iris.

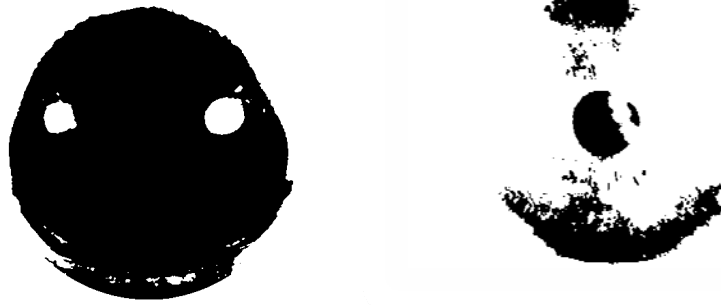


Figura 18: Imágenes Binarizada de pupila e iris.

Detección de bordes de la imagen.

Luego de realizar un estudio de los algoritmos basados en la primera derivada, Canny, e Iterativo para detección de bordes se escogió el algoritmo iterativo, este es muy efectivo en imágenes binarias y posee un tiempo de respuesta rápido. Funciona recorriendo todos los píxeles de la imagen con valor 0, y de estos píxeles los que tengan un vecino de valor 255 será considerado un borde asignándole el valor 255 (valor asociado al color blanco), Todos los demás píxeles serán llevados a color 0 (negro).



Figura 19: Imagen con mapa de bordes de pupila e iris.

Selección de la técnica de segmentación.

La imagen proveniente del iridoscopio HSK-9918U presenta una estructura variada, mucho ruido y reflejos especulares debido a los led del iridoscopio. Las técnicas basadas en fronteras no necesitan umbrales, segmentan las imágenes con algoritmos que utilizan los bordes (cambios de intensidades) como único requisito de entrada. Debido a la forma circular del iris y la pupila resulta útil aplicar un algoritmo basado en detección de bordes circulares. Entre los consultados en la literatura se destaca la Transformada Circular de Hough por ser muy útil en ambientes ruidosos como es el caso de las imágenes de iris de espectro visible.

Aplicación del algoritmo Transformada Circular de Hough

Este algoritmo depende de un buen detector de bordes para realizar sus cálculos y comparaciones. Es el algoritmo de segmentación más utilizado en la literatura consultada para la detección de formas circulares y el resultado de haberlo aplicado a las imágenes obtenidas se muestra en la

imagen 21 (Tejedor Gómez, 2009) (Industrial., Septiembre de 2005). El algoritmo funciona recorriendo todos los píxeles de valor 255 de la imagen, para cada uno de ellos traza círculos tangentes en los 360 grados. Los círculos analizados poseen un radio variable de entre 55 y 100 píxeles. De estos círculos los que cumplan que el centro del eje numérico está contenido dentro del mismo, son adicionados a una lista en la cual, si ya está incluido el círculo, se incrementan sus contadores en 1. Posteriormente se recorre esta lista sacando el círculo con mayor cantidad de repeticiones (votos), siendo este en la mayor cantidad de casos de prueba el que limita las regiones de iris y pupila (Tomé González, 2008).



Figura 20: Resultado de aplicar la transformada circular de Hough para la detección de iris y pupila.

Normalización.

Para aplicar esta técnica se define un área con el ancho de la imagen a devolver, definida por la resta del radio del iris y el radio de la pupila. Para calcular el largo de la imagen se obtiene el valor del área de la circunferencia y dadas las dimensiones largo y ancho se construye una imagen vacía, donde se guardara el iris expandido. Luego de definir estos parámetros el algoritmo funciona iterando por cada píxel de borde de la circunferencia del iris, pintando en la imagen vacía la línea que define el radio y el ángulo en cada iteración.



Figura 21 : Imagen de iris normalizado.

2.3. Fase I: Planificación del Proyecto

Según (Penadés, 2006) en esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de

desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema.

2.2.1 Historias de Usuario.

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible (Penadés, 2006). Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas. A continuación se describe una de las HU definidas, las demás se encuentran descrita en el Anexo 1.

Historia de usuario	
Número:5	Nombre: Implementar la técnica de filtrado erosión
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados:1	Iteración asignada:2
Descripción: <ul style="list-style-type: none"> Se aplica el filtro erosión a la imagen obtenida luego de aplicar el filtro de escala de grises 	

Tabla 3 : HU5 Implementar la técnica de filtrado erosión.

2.2.2 Requisitos no funcionales del sistema.

Los requisitos no funcionales son requisitos enfocados en las propiedades y características que hacen al producto atractivo, usable, rápido o confiable. No son parte de la razón fundamental del producto pero si son necesarios para hacer funcionar el producto de la manera deseada.

El equipo de desarrollo es el encargado de capturar estos requisitos a partir del intercambio de información con el cliente. A continuación, se presentan los requisitos no funcionales definidos:

Requisitos de Hardware y de Software:

Estación de trabajo:

- Hardware:
 - 1Gb de memoria RAM como mínimo
 - Procesador con frecuencia mínima de 1.6 GHz.
- Software:
 - Máquina virtual de java instalada versión 8

Requisitos de interfaz: el sistema ofrece una interfaz amigable y fácil de operar. Representando de modo simple la solución que se quiere alcanzar con el sistema.

2.2.3 Plan de Iteraciones.

En el plan de iteraciones se especifican las HU a implementar en cada iteración del sistema, estableciéndose tres iteraciones para la realización del sistema en coordinación con el cliente.

- Iteración 1: tiene como objetivo la implementación de las HU: 1, 2, 3, 4 y 14 que son las HU relacionadas con la implementación de métodos para guardar las imágenes capturadas, así como la conexión del iridoscopio con la estación de trabajo.
- Iteración 2: tiene como objetivo la implementación de las HU: 5, 6, 7 y 8 que son las HU que determinan la implementación de algoritmos de filtrado de la imagen.
- Iteración 3: tiene como objetivo la implementación de la HU: 9, 10, 11, 12 y 13 que son las HU relacionadas con los métodos de segmentación aplicados en la aplicación, así como el método guardar imagen resultante.

2.2.4 Plan de entrega.

El plan de entregas es el cronograma que establece cuales historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas (Grupo ISSI, 2003). Tiene como objetivo definir el número de liberaciones que se realizarán en el transcurso del proyecto y las iteraciones que se requieren para desarrollar cada una.

Se definió realizar tres entregas, estimando duraciones de 3,7 y 8 semanas respectivamente. En la siguiente tabla se muestran la fecha, estimación en semanas de cada entrega y las HU de usuario que serán implementadas en cada iteración para mostrar al cliente.

No	Historia de usuario	Estimación (semana)	Fecha(inicio-fin)
1	Obtener la Imagen de la estación de trabajo. Conectar el sistema con el iridoscopio HSK-9918U. Obtener la Imagen del iridoscopio HSK-9918U. Cancelar Imagen. Implementar la técnica escala de grises.	3	26/2/2016- 18/3/2016
2	Implementar filtrado de la mediana. Implementar filtrado de erosión. Implementar la técnica de filtrado ecualización de histograma adaptativo. Implementar la técnica de binarización.	7	18/03/2016- 6/04/2016

3	Implementar la técnica de mapeo de bordes iterativa. Implementar la Transformada Circular de Hough para la detección de la pupila. Implementar la Transformada Circular de Hough para la detección del iris. Implementar la técnica de normalización de iris. Guardar imagen.	8	6/04/2016-1/06/2015
---	---	---	---------------------

Tabla 4 : Plan de entrega.

2.3 Fase II: Diseño del Sistema

La metodología XP sugiere que se deben realizar diseños simples y sencillos. Los diseños deben ser fácilmente entendibles para lograr una implementación más rápida (Beck, 1999).

2.3.1 Tarjetas Clase - Responsabilidad – Colaborador.

Las tarjetas Clase - Responsabilidad - Colaborador (CRC) representan una entidad del sistema, a la cual asignar responsabilidades y colaboraciones. El formato físico de las tarjetas CRC facilita la interacción entre los participantes del proyecto, en sesiones en las que se aplican técnicas de grupos como tormenta de ideas o juego de roles, y se ejecutan escenarios a partir de la especificación de requisitos, historias de usuarios o casos de uso. De esta forma, van surgiendo las entidades del sistema junto con sus responsabilidades y colaboraciones (Casas, y otros, 2009).

En la metodología XP el proceso de diseño es iterativo. Las tarjetas no se crean en un mismo tiempo, se van desarrollando en la medida que se ejecuten las iteraciones del sistema y se le añaden responsabilidades y colaboradores según sea necesario. A continuación se muestran las clases de diseño identificadas por el equipo de desarrollo, así como un ejemplo de las tarjetas CRC, las restantes se encuentran en el Anexo 2 al final del documento.

Las tarjetas CRC están compuestas por tres campos, clase, responsabilidades y colaboradores:

- **Clase:** nombre de la clase que se está modelando.
- **Responsabilidades:** las responsabilidades de una clase son las acciones que conocen y realizan, sus atributos y métodos.
- **Colaboradores:** los colaboradores de una clase son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades.

A continuación, se realiza la descripción de la estructura de la tarjeta CRC de la clase Mediana. Esta clase contiene tres métodos, y colabora con la clase ImagenApi para realizar el método aplicar.

Mediana	
Atributos:	
Nombre	Descripción
radio	Tipo int
Responsabilidades:	
Nombre	Collaborator
aplicar()	ImagenApi
getRadio()	
setRadio()	

Tabla 5: Tarjeta CRC de la clase Mediana

2.3.2 Patrones de diseño de software.

En la investigación y desarrollo del proyecto actual se usaron los siguientes patrones:

Creador: asigna la responsabilidad de crear un objeto de la clase *Imagen_api* a la clase *Clase*, por ello el patrón sugiere que dicho modelo es la idónea para asumir la responsabilidad de crear la instancia de *Imagen_api*.

Controlador: la clase *C_principal* le asigna la responsabilidad de realizar la operación de filtrado de la mediana a la clase *Mediana*, por tal motivo *C_principal* es la que conlleva a cumplir este patrón.

Experto: se aplica en las clases que tienen la información suficiente para hacer lo que se les pide, prueba de la utilización de este patrón está en la clase *Circulo*, la cual contiene todos los atributos y métodos para figurar un circulo.

Bajo Acoplamiento: es el patrón que se refiere a la mínima dependencia entre clases. Se utilizó debido a la importancia que se le atribuye a realizar un diseño de clases independientes que puedan soportar los cambios de una manera fácil y que a su vez permitan la reutilización. Así se evidencia en el método *aplicar* de la clase *Mediana* donde solo necesita instanciar una sola vez la clase *Imagen_api* para hacer responsabilidades.

Alta cohesión: la clase *C_principal* asigna ciertas responsabilidades a la clase *Clase* y la misma a la clase *Imagen_api*, de tal forma que se delegue cierta cantidad de carga a otras clases y a la misma vez se tenga una relación entre las mismas.

2.4 Fase III: Codificación.

La codificación debe hacerse atendiendo a estándares de codificación. El uso de estándares mantiene el código consistente y facilita su comprensión y escalabilidad.

2.4.1 Estándares de codificación

Son de gran utilidad e importancia para los programadores, debido a que su definición ayuda a entender el código mucho más rápido y en profundidad. Además permite realizar un buen mantenimiento al software y en caso de distribuir el código se hace necesario que el código tenga la misma calidad de cualquier otro producto en el mercado.

Existen diversos tipos de convenciones, entre las más comunes se encuentran: las de comentario, declaración de clases, sentencias y de nombre (Hommel, 1999).

En el presente trabajo se utilizaron las siguientes convecciones:

- Todas las nomenclaturas a utilizar se definirán en idioma español, las únicas en otro idioma son las que provienen de bibliotecas importadas al proyecto.
- Los nombres de los métodos serán con minúscula, en caso de ser un nombre compuesto las siguientes palabras se escribirán con mayúscula.
- Los identificadores para las variables y los parámetros serán con letras en minúsculas y en caso de ser un nombre compuesto las siguientes palabras se escribirán con mayúscula.
- Los nombres de variables o funciones deben ser lo suficientemente descriptivos, sin exceder de 30 caracteres.
- Todas las funciones deben tener comentarios explicando que realiza cada una de ellas.
- Los nombres de las clases serán con mayúscula, en caso de ser un nombre compuesto las siguientes palabras se escribirán de igual forma.

2.4.2 Tareas de Programación.

Las tareas de programación le permiten a los desarrolladores obtener un nivel de detalle más avanzado por las HU. A continuación se describe la tarea de programación 3, que tiene como objetivo implementar la HU:

Tarea de Programación	
Número de tarea:1	Historia de Usuario(No.1): Conectar el sistema con el iridoscopio HSK-9918U
Nombre de tarea: Cargar Imagen del iridoscopio	
Tipo de tarea: Desarrollo	Puntos Estimados:3
Fecha de inicio:17/2/2016	Fecha de Fin:1/3/2016
Programador responsable: Thais Alonso Tarafa	

Descripción: este requisito debe permitir utilizar una biblioteca para cargar la imagen desde el iridoscopio.

Tabla 6: Tarea de Programación 1

2.5 Conclusiones del capítulo

En este capítulo se propuso una aplicación para realizar la segmentación y normalización de iris. Luego de presentada la propuesta se concluye que:

1. El empleo de la metodología XP en función de describir el proceso de desarrollo de la herramienta permitió realizar un trabajo organizado y estructurado, generando los artefactos de cada una de sus fases.
2. El uso de estándares de codificación y patrones de diseño permitió obtener un diseño estándar y organizado que será de fácil comprensión para otros desarrolladores.
3. Las historias de usuario definidas por el cliente permitieron realizar los planes de iteraciones y entrega donde se estimó el tiempo que requiere la implementación de cada una, y qué historias de usuario forma parte de cada entrega.

CAPÍTULO 3: Resultados y validación del sistema

3.1 Introducción.

En este capítulo se realiza la validación y verificación de la solución propuesta mediante las pruebas de caja blanca, caja negra y pruebas de aceptación.

3.2 Validación del diseño.

Para comprobar la calidad del diseño se emplearon las técnicas del diseño Relaciones entre Clases(RC) y Tamaño Operacional de Clases (TOC).

3.2.1 Relaciones entre clases(RC).

La métrica RC está dada por el número de relaciones de uso de una clase con otra. El primer paso es evaluar un conjunto de atributos de calidad entre los que se encuentran el acoplamiento, complejidad de mantenimiento y reutilización de cada clase (**Pressman, 2002**).

A continuación se explican los pasos que se llevaron a cabo para aplicar la métrica:

1. Determinar la cantidad de relaciones de uso (CRU) que poseen las clases a medir.
2. Calcular el promedio de las CRU.

3. Teniendo en cuenta los valores antes obtenidos se determina la incidencia de los atributos de calidad en cada una de las clases, según los criterios expuestos en la tabla 6.

Atributos de calidad	Clasificación	Criterio
Acoplamiento	Ninguna	CRU = 0
	Baja	CRU = 1
	Media	CRU = 2
	Alta	CRU > 2
Complejidad de mantenimiento	Baja	CRU ≤ Promedio
	Media	Promedio < CRU ≤ 2* promedio
	Alta	CRU > 2* promedio
Reutilización	Baja	CRU > 2* promedio
	Media	Promedio < CRU ≤ 2* promedio
	Alta	CRU ≤ Promedio
Cantidad de pruebas	Baja	CRU ≤ Promedio
	Media	Promedio < CRU ≤ 2* promedio
	Alta	CRU > 2* promedio

Tabla 7 : Rango de valores para medir la afectación de los atributos de calidad (RC).

En la siguiente figura se muestra el resultado de la aplicación de la métrica RC

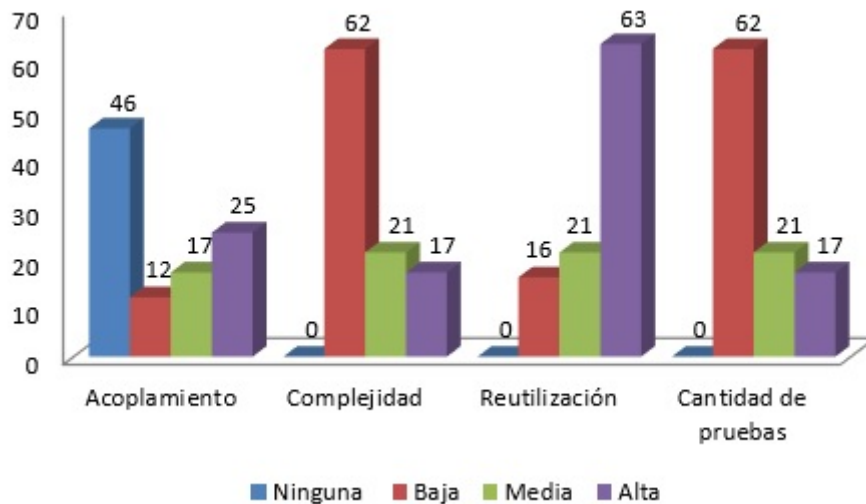


Figura 22 Representación en (%) de los resultados de la aplicación de la métrica RC.

Acoplamiento: según los resultados que se muestran, el 46% de las clases no posee relaciones de uso por lo que la mayoría de las clases no tienen valores de acoplamiento, validando una realización correcta del diseño.

Complejidad de mantenimiento: según los resultados que se muestran en la figura anterior, el 62% de las clases se comportan de forma satisfactoria pues, son de fácil soporte.

Reutilización: según los resultados que se mostraron en la figura, el 63% de las clases tiene un alto grado de reutilización.

Cantidad de pruebas: luego de aplicar la métrica se obtuvo que el 62% de las clases poseen un bajo grado de esfuerzo a la hora de realizar cambios, rectificaciones y pruebas desoftware.

Según lo analizado anteriormente, los valores de RC se comportan de forma satisfactoria. Estos resultados expresan que las clases del diseño de la herramienta presentan bajo acoplamiento, la complejidad de mantenimiento y la cantidad de pruebas son bajas y en consecuencia el grado de reutilización es alto.

3.2.2 Tamaño Operacional de Clases (TOC).

La métrica TOC fue aplicada a cada una de las clases del diseño con el objetivo de medir la calidad de las mismas con respecto a su grado de responsabilidad, complejidad de implementación y reutilización (**Pressman, 2002**).

A continuación se explican los pasos que se llevaron a cabo para aplicar la métrica:

1. Cálculo del umbral. El umbral se toma del tamaño general de una clase que se determina sumando todas las operaciones que posee.
2. Calcular el promedio de los umbrales.
3. Teniendo en cuenta los valores antes obtenidos se determina la incidencia de los atributos de calidad en cada una de las clases, según los criterios expuestos en la tabla 7.

Atributos de calidad	Clasificación	Criterio
Responsabilidad	Baja	Umbral <= Promedio
	Media	Promedio < Umbral <= 2* promedio
	Alta	Umbral > 2* promedio
Complejidad de implementación	Baja	Umbral <= Promedio
	Media	Promedio < Umbral <= 2* promedio
	Alta	Umbral > 2* promedio
Reutilización	Baja	Umbral > 2* promedio
	Media	Promedio < Umbral <= 2* promedio
	Alta	Umbral <= Promedio

Tabla 8 : Rango de valores para medir la afectación de los atributos de calidad (TOC).

En la siguiente figura se muestra el resultado de la aplicación de la métrica TOC.

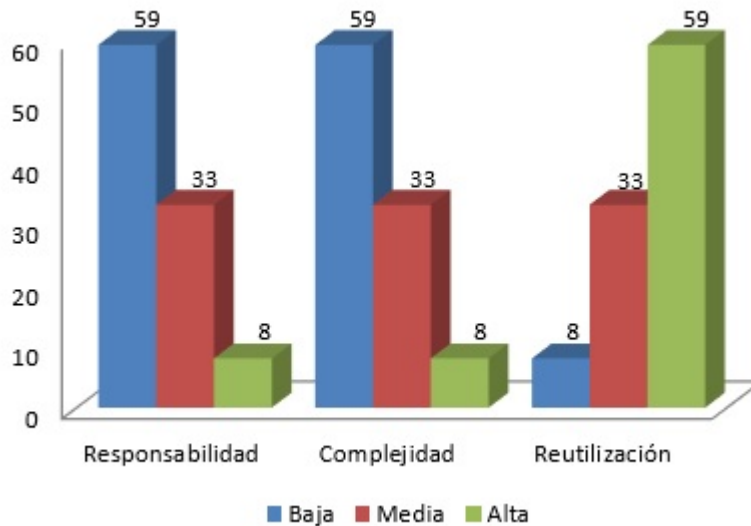


Figura 23 Representación en (%) de los resultados de la aplicación de la métrica TOC.

Responsabilidad: luego de aplicar la métrica se obtuvieron resultados satisfactorios que reflejan una responsabilidad baja con un valor del 59%.

Complejidad de Implementación: después de haberse realizado la medición de la métrica, este parámetro obtuvo resultados positivos debido a que la complejidad de las clases es baja en un 59%.

Reutilización: se obtuvieron valores que según muestra la gráfica de la figura anterior se comporta en un nivel alto con un 59%.

Los resultados arrojados por la métrica TOC expresan que las clases del diseño de la herramienta presentan una elevada reutilización y baja complejidad y responsabilidad. Por lo que se concluye que los resultados obtenidos en esta métrica son positivos.

3.3 Pruebas de software.

La definición del conjunto de pruebas para probar la eficiencia del software, es tan importante como elegir una metodología apropiada para el desarrollo de un producto software. Uno de los pilares de la metodología XP es el proceso de pruebas, permitiendo aumentar la calidad de los sistemas, reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección (Gutiérrez, y otros, 2005).

XP propone dos tipos de pruebas: pruebas unitarias, encargadas de verificar el código y diseñadas por los programadores, y pruebas de aceptación o pruebas funcionales, destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñada por el cliente final. En el caso específico del componente propuesto, solo se utilizarán las pruebas unitarias y las pruebas de evaluación de la calidad de segmentación (Rodríguez, 2011).

3.3.1 Pruebas de unidad.

Seguidamente se describirá la realización 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.

3.3.1.1 Técnica de Camino Básico

La técnica de camino básico se le aplicará al método de binarizaciónIris de la clase Cprincipal, por ser este uno de los pasos más importantes durante el desarrollo de la aplicación. A continuación se muestra los pasos para emplear dicha técnica.

```
public BufferedImage binarizacionIris(BufferedImage temp) {
    //Variables que almacenarán los píxeles
    System.out.println("Binarizacion");
    int mediaPixel;
    Color colorAux;

    //Recorremos la imagen pixel a pixel
    for (int i = 0; i < temp.getWidth(); i++) {

        for (int j = 0; j < temp.getHeight(); j++) {
            //Almacenamos el color del píxel
            colorAux = new Color(temp.getRGB(i, j));
            mediaPixel = (int) ((colorAux.getRed() + colorAux.getGreen() + colorAux.getBlue()) / 3);
            if ((mediaPixel >= 210)) {
                temp.setRGB(i, j, (255 << 16) | (255 << 8) | 255);
            }
            else {
                temp.setRGB(i, j, (0 << 16) | (0 << 8) | 0);
            }
        }
    }

    //Retornamos la imagen
    return temp;
}
```

Figura 24 : Código fuente del método binarizaciónIris().

El grafo de flujo asociado al método binarizaciónIris() es el siguiente:

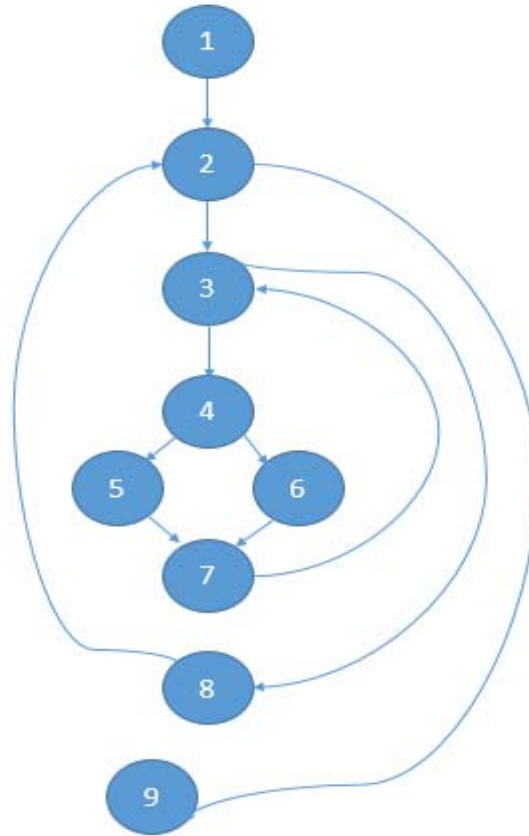


Figura 25 : Grafo de flujo asociado al método binarizaciónIris().

Después de haber realizado el grafo, se calcula la complejidad ciclomática por tres fórmulas distintas, las cuales deben dar el mismo resultado para comprobar que el cálculo sea correcto.

Complejidad ciclomática.

La complejidad ciclomática es una métrica de software extremadamente útil, pues proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado como complejidad ciclomática, define el número de caminos independientes del conjunto básico de un programa y nos da un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecute cada sentencia al menos una vez.

$$V(G) = \text{Regiones} = 4$$

$$V(G) = \text{Aristas} - \text{Nodos} + 2 = 11 - 9 + 2 = 4$$

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

Caminos independientes determinados.

A continuación, se muestran los caminos independientes determinados a partir de los valores obtenidos de la complejidad ciclomática, los mismos deben estar constituidos como mínimo por una arista que no haya sido recorrida anteriormente a la definición del camino.

Camino 1: 1-2-9

Camino 2: 1-2-3-8-2-9

Camino 3: 1-2-3-4-5-7-3-8-2-9

Camino 4: 1-2-3-4-6-7-3-8-2-9

Luego de haberle aplicado el método del camino básico a la funcionalidad anterior se comprobó que cada sentencia es ejecutada al menos una vez. El valor calculado como complejidad ciclomática definió el número de caminos independientes del conjunto básico (Pressman, 2005), lo que facilitó conocer que se deben realizar como máximo cuatro pruebas.

3.3.1.2 Casos de prueba para la prueba de caja blanca.

El próximo paso es ejecutar los casos de pruebas para cada camino y se compara con los resultados esperados, verificando que las instrucciones se ejecuten por lo menos una vez. A continuación se muestra el caso de prueba para el camino básico 3, en el [Anexo 3](#) se muestran los restantes casos de prueba.

Caso de prueba para el camino básico 3	
Descripción	Comprueba si la matriz de píxeles tiene un ancho y alto válido. Para después según la condición pintarla de blanco.
Condiciones de ejecución	<ul style="list-style-type: none">• Que la matriz de píxeles tenga un alto (Height) y un ancho (Width) válidos.
Entrada	BufferedImage temp.
Resultado esperado	Una imagen donde el valor de los píxeles que cumplan la condición será cambiado a 255.

Tabla 32: Casos de prueba para el camino 3.

3.3.1.3 Prueba de Caja Negra.

Según (Pressman, 2005), *Las pruebas de caja negra*, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software.

Los casos de prueba de la caja negra pretenden demostrar que:

- Las funciones del software son operativas.
- La entrada se acepta de forma correcta.
- Se produce una salida correcta.
- La integridad de la información externa se mantiene.

Resultados de la prueba de caja negra.

La siguiente grafica muestra los resultados obtenidos en las pruebas realizadas al sistema durante cada iteración; en búsqueda de no conformidades de funcionalidad, redacción e interfaz.

Obteniendo un resultado satisfactorio durante la última iteración en un total de 25 pruebas.

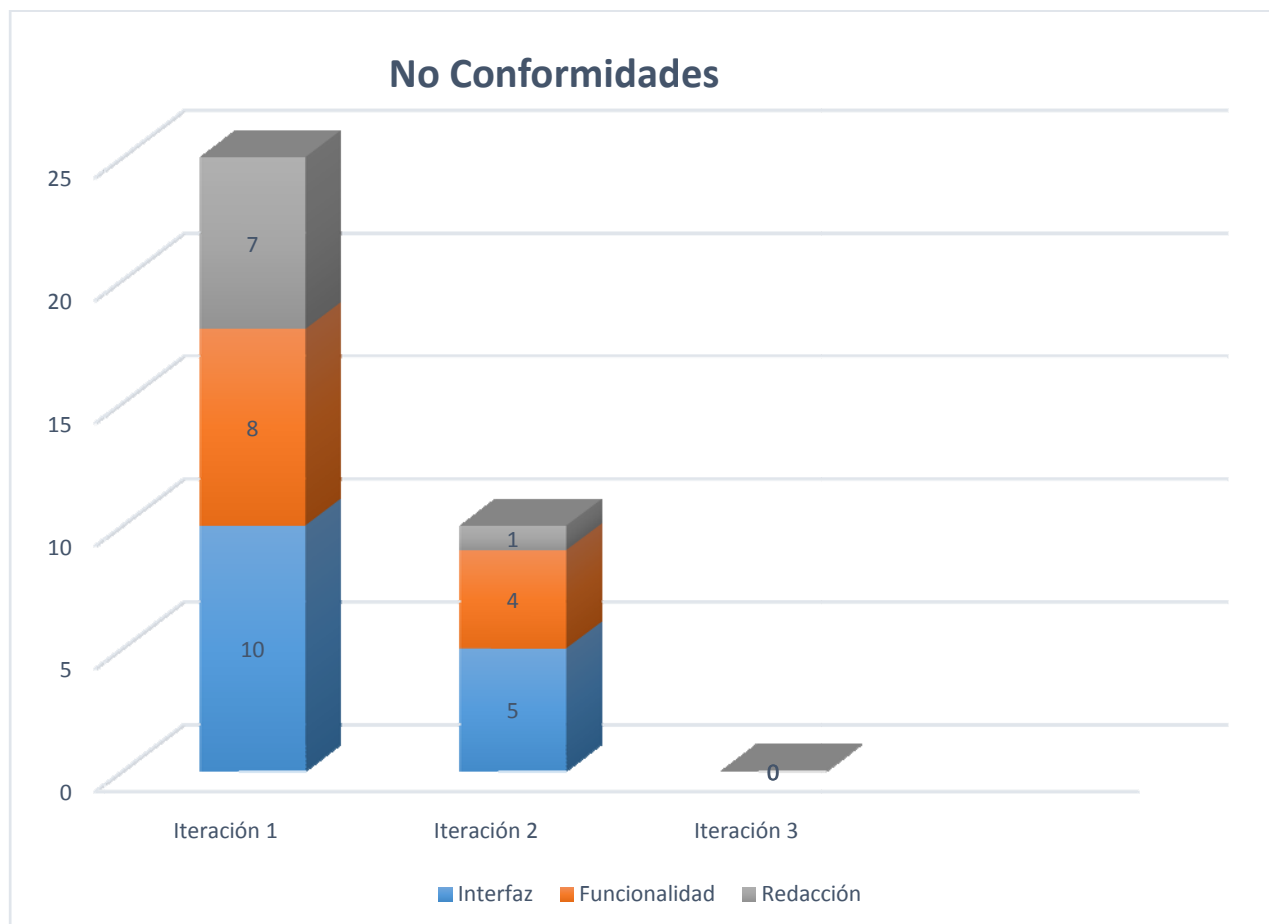


Figura 26: Gráfica de las No conformidades.

3.3.2 Prueba de aceptación.

Las pruebas de aceptación cumplen con el objetivo de indicarnos cuando las funcionalidades de una iteración han sido completadas exitosamente. La validación del software se consigue mediante una serie de pruebas de caja negra que demuestran la conformidad del cliente con los requisitos. Una vez que se procede con cada caso de prueba de validación, puede darse una de las siguientes condiciones:

1. Las características de funcionamiento o de rendimiento están de acuerdo con las especificaciones y son aceptables.
2. Se descubre una desviación de las especificaciones y se crea una lista de deficiencias.

Para la descripción de estas pruebas se presenta el diseño del caso de prueba correspondiente a la historia de usuario ``Obtener la Imagen del iridoscopio HSK-9918U`` que incluye funcionalidades críticas para el sistema. La siguiente tabla muestra los resultados satisfactorios obtenidos en la puerta de aceptación ``Obtener la Imagen del iridoscopio HSK-9918U``.

Caso de prueba de aceptación		
Código de caso de prueba: 1	Nombre historia de usuario: Implementación de la técnica Transformada Circular de Hough para la detección del iris	
Nombre de la persona que realiza el caso de prueba: Thais Alonso Tarafa		
Descripción de la prueba: revisar a través de la herramienta el correcto funcionamiento del RF Detectar Iris		
Condiciones de ejecución: se debe tener una imagen de mapa de bordes previamente cargada.		
Entrada/Pasos de ejecución		Resultados esperados:
Acción:	Entrada:	
	Imagen	
		El sistema debe mostrar un conjunto de imágenes con el resultado del proceso de segmentación completo.
Evaluación de prueba: Satisfactoria		

Tabla 9 : Caso de prueba (CP) para la HU Obtener la Imagen del iridoscopio HSK-9918U.

Se realizó un encuentro con el Ing. Michel Álvarez Cancio, coordinador e investigador del Grupo de Investigación AIRI de la facultad 3, con el objetivo de revisar la herramienta, teniendo en cuenta los CP definidos. Los resultados obtenidos fueron satisfactorios, avalados por el especialista entrevistado como un aporte importante para el desarrollo del tratamiento de imágenes de iris en Cuba. Al finalizar el encuentro se generó el Acta de Aceptación del Producto (ver Anexo 5).

3.4 Métrica de evaluación de la calidad de la segmentación.

La evaluación de la segmentación es una tarea compleja que involucra a varias fases en el proceso y se utiliza para juzgar el rendimiento de los algoritmos de segmentación (Rodríguez, 2011). En (Liu, y otros, 1999) se identifica un protocolo de evaluación del rendimiento compuesto de tres elementos esenciales: adquisición del terreno de verdad, un procedimiento de cotejo y la definición de métricas cuantitativas.

Los métodos de evaluación de segmentación pueden ser clasificados en métodos analíticos y métodos empíricos. Los métodos analíticos tratan los algoritmos de segmentación directamente y están dirigidos a examinar su principio de funcionamiento, mientras que los métodos empíricos juzgan la imagen segmentada de acuerdo a un criterio predefinido o comparando con una imagen de referencia, para evaluar indirectamente el rendimiento de los algoritmos. La evaluación empírica es prácticamente más eficaz y actualmente es más utilizada que la evaluación analítica (Rodríguez, 2011).

Con el objetivo de evaluar la calidad de la segmentación del sistema, se utilizó la métrica de calidad definida en (Osorio Roig, y otros, 2015), la cual se compone de dos fórmulas para calcular la precisión de segmentación de iris y pupila, y una fórmula general para calcular la detección efectiva general dadas las formulas anteriores, estas fórmulas se presentan a continuación:

Diferencia de cuadrados externa

$$ESD = \frac{\sum(DAS-TAS)}{\sum DAS*100} \quad (9)$$

Donde ESD es la diferencia entre el área del cuadrado óptimo que contiene el contorno real y el área del contorno detectado. Las variables DAS y TAS representan el área cuadrada tangencial detectada y el área del contorno real respectivamente. Esta métrica es utilizada específicamente para calcular la precisión del iris (Osorio Roig, y otros, 2015).

Diferencia de cuadrados interna

$$ISD = \frac{\sum(TAS-DAS)}{\sum TAS*100} \quad (10)$$

Donde ISD es la diferencia entre el área del cuadrado óptimo que contiene el contorno real y el área del contorno detectado. Esta métrica es utilizada específicamente para la precisión de la pupila (Osorio Roig, y otros, 2015).

Detección efectiva general

$$GFT = 100 - \left[\frac{1}{2}(ESD + ISD) \right] \quad (11)$$

Donde GTF representa el porcentaje general teniendo en cuenta la diferencia de cuadros externa y la detección efectiva general (Osorio Roig, y otros, 2015).

Con el objetivo de evaluar la calidad de la segmentación del sistema se seleccionaron 10 imágenes tomadas del iridoscopio HSK-9918U para realizar la métrica propuesta en (Rodríguez, 2011). Estas

imágenes muestran el ojo humano en diferentes ángulos, algunas lo presentan de frente y otras hacia los lados, y todas en general presentan distintos niveles de ruido.

Para la aplicación de la métrica de calidad se realizó el siguiente procedimiento:

1. Se señalaron de manera manual las áreas de iris y pupila para cada imagen.
2. Se realizó un método para contar la cantidad de píxeles en las áreas seleccionadas en el paso anterior, identificando esos valores como área real detectada (TAS).
3. Se procesó cada imagen en la herramienta, contando en cada caso la cantidad de píxeles presentes en las áreas localizadas, seleccionándose estos valores como área cuadrada tangencial detectada (DAS) para pupila e iris respectivamente.
4. Luego de obtener los valores TAS y DAS para el iris se calcula la diferencia de cuadrados externa (ESD), con la fórmula propuesta en (Rodriguez, 2011) , utilizada específicamente para calcular la precisión con que se detecta el iris.
5. Se calcula la diferencia de cuadrados interna (ISD), para calcular la precisión con que se detecta la pupila.
6. Como último paso de esta métrica se calcula la detección efectiva general (GFT), porcentaje general de segmentación teniendo en cuenta los valores ISD y ESD.

A continuación se muestra una tabla con los valores obtenidos en cada etapa de la métrica y el resultado general:

$\sum (DAS - TAS)$	$\sum DAS * 100$	ESD	$\sum (TAS - DAS)$	$\sum TAS * 100$	ISD	GFT
-3	3534900	8.4868	1	481235	2.07800	
					Porcentaje	94.7%

Tabla 10 :Resultados Generales de la métrica de calidad de segmentación

La aplicación de la métrica propuesta dio como resultado un 94.7% de la calidad de segmentación, evidenciando que el ruido de la imagen se minimizó lo suficiente para segmentar el iris y la pupila con efectividad.

3.5 Pruebas de Ejecución

Con el objetivo de probar el tiempo promedio de ejecución del sistema, se realizaron un conjunto de pruebas en varias estaciones de trabajo con distintas prestaciones. En la tabla se observan las características de las estaciones de trabajo y los resultados de la aplicación de las pruebas con 10 imágenes.

Estación de Trabajo	1	2	3	4	5	6	7	8	9	10	Media
Intel Centrino 2 núcleos CPU 1.6 2GB RAM	24	21	27	25	27	24	28	24	26	20	24.6

Intel Core i3 4005U CPU 1.70 4GB RAM	12	15	16	17	14	12	16	18	14	11	12.8
Intel Core i7 -26700qm CPU 2.2 8 GB RAM	10	7	11	9	14	8	12	13	11	15	10.7

Tabla 11 : Tiempo Promedio de Ejecución de la herramienta

3.5 Aporte de la investigación.

La herramienta obtenida constituye un aporte a la medicina cubana, teniendo como enfoque la rama de la Iridología. La misma hace un despeje para otra posible versión del PANIRIS, que emplee los métodos de la cuarta etapa del procesamiento digital de imágenes, extracción de características; dando así la posibilidad de detectar marcas y lesiones localizadas en el iris.

3.6 Conclusiones del capítulo

En el presente capítulo se abordaron los temas referentes a la fase de pruebas y validación del sistema, concluyendo que:

1. La aplicación de métricas de diseño, demostró que las clases del diseño poseen bajo acoplamiento, que existe además una baja responsabilidad y complejidad de implementación y una alta reutilización en el diseño propuesto.
2. Con la validación realizada a la solución a partir de pruebas de aceptación y de pruebas unitarias empleando los métodos de caja negra y caja blanca, se pudo constatar después de 3 iteraciones, que el sistema quedo libre de no conformidades, por lo que el cliente puede utilizarlo en un entorno real.
3. La aplicación de la métrica de validación de la calidad de segmentación permitió validar que la solución obtenida presenta altos valores de efectividad en imágenes de iris en espectro visible.

Conclusiones generales

Luego de haber finalizado la investigación se concluye:

- Con el análisis de los principales conceptos relacionados con las técnicas de realce, mejora y segmentación de iris, se logró profundizar en la comprensión del problema planteado por la investigación.
- El estudio de las técnicas y herramientas de la literatura científica consultadas, permitió seleccionar las técnicas de realce y mejora: Erosión, Mediana, Escala de Grises, y Ecuilización de histograma adaptativo y las técnicas mapeo de bordes iterativo y transformada circular de Hough como la combinación de técnicas que más se ajustan a los requerimientos de la investigación.
- El diseño del software permitió definir las responsabilidades de cada integrante de la investigación y los patrones de diseño utilizados en el desarrollo de la herramienta.
- La aplicación de las métricas de diseño, validación de la calidad de segmentación y pruebas de software demostraron que la solución obtenida tiene altos niveles de efectividad y calidad segmentando imágenes de iris pertenecientes al espectro visible.

Recomendaciones.

- Incorporar a la herramienta nuevas funcionalidades que permitan un sistema de reconocimiento de patrones de iris orientado a la biometría.

Bibliografía

- Márquez Álvarez, Yaimara y Salazar Reyes, Licel . 2009.** *Propuesta de un filtro para la eliminación del ruido causado por la iluminación en el Sistema de visión de un robot móvil.* Ciudad de La Habana : s.n., 2009.
- Alexander, Christopher. 1977.** *A Pattern Language: Towns, Buildings, Construction.* 1977.
- Alonso Dominguez, Hector. 2014.** *Iridología.Guía de tratamiento integral.* Managua : s.n., 2014. 978-99964-0-365-1.
- Beck, K. 1999.** 1999, Computer, Vol. 32(10), págs. 70-77.
- Benalcazar Palacios, Freddy Geovanny. 2008.** Aplicación de Morfología Matemática difusa al filtrado de Imágenes de Resonancia Magnética. 2008.
- Benavides, Mireille y González, Alejandra. 2015.** 2015. XXVI CONGRESO DE INVESTIGACIÓN CUAM- ACMor. pág. 2.
- Bowyer Kevin W., Hollingsworth Karen, Flynn Patrick J. 2008.** San Francisco : Elsevier, 2008, Computer Vision and Image Understanding, Vol. 110, págs. 281-307. 1077-3142.
- Caicedo Marmolejo, Luis Felipe y Chamorro Carvajal, Carlos Eduar. 2012.** *Implementación y evaluación de algoritmos de identificación por reconocimiento del iris ocular.* Santiago de Cali : s.n., 2012.
- Calabria, Luis y Píriz, Pablo. 2003.** *Metodología XP.* Universidad ORT Uruguay : Editorial: Universidad ORT, 2003.
- Camus, T A y Wildes, R. 2002.** 2002, 16th International Conference on Proceedings, págs. 389-394.
- Casas, Sandra y Reinaga, Héctor. 2009.** *Aspectos Tempranos: un enfoque basado en Tarjetas CRC.* Argentina : Sandra Casas y Héctor Reinaga, 2009.
- Cepero Ruz, Raciél y Cabrera Rodríguez, Claudia. Mayo 2010.** *Implementación de algoritmos de reducción de ruido en las imágenes.* Ciudad de La Habana : s.n., Mayo 2010.
- Climent Vidal, J. 2005.** *Teoría de Conjuntos.* Valencia : Universidad de Valencia, Departamento de Lógica y Filosofía de la Ciencia, 2005.
- Daugman, J. 2007.** 2007, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 37, págs. 1167–1175.
- Diaz de León, Juan Santiago, y otros. 2009.** 48, Medellin : s.n., 2009, Revista Facultad de Ingeniería Universidad de Antioquia, págs. 216-227. ISSN:0120-6230.

Doraijaj, V, Schmid, N Z y Fahmy, G. 2005. 2005, IEEE International Conference on Image Processing, págs. 11-14.

Felip León, Xavier , y otros. 2005. *MODIFICACIÓN DEL HISTOGRAMA DE UNA IMAGEN.* 2005.

Fernández, Oscar Belmonte. 2004. *Introducción al lenguaje de JAVA. Una guía básica.* 2004.

Finamore, Maria Laura y Devincenzi, Juan Alberto . 2011. *Sistema de reconocimiento biométrico en ambientes de alta seguridad.* s.l. : Universidad Nacional de la Plata, 2011.

Fuente-Bermudez, S.L., Orlena-Rodriguez, F. J. y González-Barbosa, J. J. 2015. *DETECCIÓN DE MASAS Y ASIMETRÍAS EN MAMOGRAFÍAS CON DISTINTOS PATRONES PARENQUIMALES, UTILIZANDO ENTROPÍA DE TSALLIS.* 2015.

Gallo Sánchez, María Belén . 2013. *SEGMENTACIÓN DIGITAL DE IMÁGENES MÉDICAS PARA EL MEJORAMIENTO DEL DIAGNÓSTICO DE ANOMALÍAS EN LOS CENTROS RADIOLÓGICOS DE LA CIUDAD DE AMBATO.* Ambato : s.n., 2013.

Gamma, Erich,Helm, R., Johnson, R., & Vlissides, J. 1995. *Design Patterns: Elements of.* s.l. : Addison-Wesley Professional, 1995.

García de Jalón, Javier y Rodríguez, José Ignacio. diciembre 2005. *Aprenda Matlab 7.0. Como si estuviera en primero.* Madrid : s.n., diciembre 2005.

Gil, G. D., Arias Figueroa, D., Gimson, L., Ramírez, J., & Silvera, J. A. 2012. 2012. XIV Workshop de Investigadores en Ciencias de la Computación.

Gonzalez, Rafael C y Woods, Richard E. 2002. *Digital Image Processing. Digital Image Processing.* New Jersey : Prentice-Hall, 2002.

GONZALEZ, Rafael C y WOODS, Richard E. 2002. *Digital image processing.* 2002.

Grupo ISSI. 2003. *Metodologías Ágiles en el Desarrollo de Software.* Alicante : s.n., 2003.

Gutiérrez, J J, y otros. 2005. *PRUEBAS DEL SISTEMA EN PROGRAMACIÓN.* s.l. : Department de Lenguajes y Sistemas Informáticos, 2005.

Hernández, U.M. 2002. *Reducción de ruido en una imagen digital.* 2002.

Hommel, Scott. 1999. *Convenciones del código Java para el lenguaje de programación.* 1999.

hskmedical. 2013. HSK Medical Technology. [En línea] Iridology.com.cn, 2013. [Citado el: 25 de abril de 2016.] http://www.hskmedical.com/sdp/233709/4/pd-971512/5661102/Iridology_Camera_HSK-9918U_2012_Best_Selling.html. 05147324.

Industrial., Ing. en Automatización y Control. Septiembre de 2005. s.l. : Universidad Nacional de Quilmes, Septiembre de 2005.

INEI. 2015. academia. *Herramientas Case.El mejor soporte para el proceso de desarrollo de software.* [En línea] 2015. [Citado el: 12 de 4 de 2015.] http://www.academia.edu/4513393/Libro_HERRAMIENTAS_CASE.

Laorden Fiter, Eduardo. 2012. *Descripción ,comparación y ejemplos de uso de las funciones de la toolbox de procesamiento digital de imágenes de MATLAB.* 2012.

Liu, W y Dori, D. 1999. 1999, págs. 97-106.

Malfará, Dayvis, y otros. 2006. *Testing en eXtreme Programming.* 2006.

Martínez Juan, Francisco Javier. 2007. *Guía de construcción de software en java con patrones de diseño.* 2007.

Nube, Paralelo. 2010. Procesamiento de Imágenes Digitales. [En línea] 2010. [Citado el: 18 de 1 de 2016.] <http://atenciolpdi.blogspot.com/2010/07/filtros-y-ruidos-unidad-2.html>.

Osorio Roig, Daile y Garea Llano, Eduardo. 2015. 2015. Métodos de segmentación de iris bajo el espectro de luz visible.

Osorio Roig, Dailé y Garea Llano, Eduardo. 2016. 2142, s.l. : CENATAV, 2016. 2072-6287.

Pedreira, O., Piattini, M., Luaces, M. R., & Brisaboa, N. R. 2007. 2007, Innovación, Calidad e Ingeniería del Software, pág. 21.

Penadés, Patricio Letelier y M^a Carmen. 2006. *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP).* Camino de Vera s/n, 46022 Valencia : Universidad Politécnica de Valencia, 2006.

Platero Dueñas, Carlos. 2009. Apuntes de Visión Artificial. s.l. : Dpto.Electronica,Automática e Informática Industrial, 2009, págs. 171-198.

Pressman, Roger S. 2005. *Ingeniería del Software. Un enfoque práctico 6ta edición.* Nueva York : McGraw-Hill, 2005. 0072853182.

Pressman, Roger S. 2002. *Ingeniería del Software. Un Enfoque Práctico. Quinta Edición.* España : McGraw-Hill, 2002.

Procesamiento Global empleando la Transformada de Hough. Industrial., Ing. en Automatización y Control. Septiembre de 2005. s.l. : Universidad Nacional de Quilmes, Septiembre de 2005.

Rodriguez, Jose Luis Gil. 2011. *Evaluación de la calidad en la segmentación de imágenes.* Centro de Aplicaciones y tecnologías Avanzadas. La Habana : s.n., 2011. pág. 50, Reporte Técnico. 2072-6287.

Romo Gonzales, Ana Eugenia. 2005. *Reconocimiento de imágenes mediante redes neuronales artificiales aplicado a la iridología.* México : CINVESTAV-IPN, 2005.

Salles LF, Silva MJ. 2008. 2008, Esc Enferm USP, págs. 596-600.

Serra, Jean y Angulo, Jesus. 2005. *Segmentación de Imágenes en Color utilizando Histogramas Bi-Variabes en Espacios Color Polares Luminancia/Saturación/Matiz.* 2005.

Suárez, María Lorena. 2015. Competencias en TIC: Colección de Fasículos Digitales. *Cuaderno 1: Introducción a la programación y sus lenguajes.* [En línea] 2015. [Citado el: 5 de 4 de 2015.] http://competenciastic.educ.ar/pdf/lenguajes_de_programacion_1.pdf.

Suaza, K. V., García, J. J. T., & Jaramillo, C. M. Z. 2015. 2015, Cuaderno Activa, págs. 41-53.

Tejedor Gómez, Jesús. 2009. *Análisis Comparativo de Algoritmos en Segmentación de Iris.* Madrid : s.n., 2009.

Tomé González, Pedro. 2008. *Reconocimiento Automático de patrones de iris.* Madrid : s.n., 2008.

Tristan, Dr. Rubén Wainschenker Mg. Ing. José María Massa Mg. Ing. Paula. 2011. 2011.

Visual Paradigm. 2000. Visual Paradigm International. *Visual Paradigm.* [En línea] 2000. [Citado el: 4 de 12 de 2014.] <http://www.visual-paradigm.com>.

Zhu, H., Hall, P. A., & May, J. H. 1997. *Software unit test coverage and adequacy. Acm computing surveys.* 1997. págs. 366-427.

