



**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**FACULTAD 3**

**GRUPO DE INVESTIGACIÓN AIRI**

**TRABAJO PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS  
INFORMÁTICAS**

**Algoritmo para la restauración de imágenes oblicuas de opacidad de la cápsula  
posterior provenientes de la lámpara de hendidura.**

**AUTORES**

Rosaibis Torres Ruisanchez

Manuel Alejandro Zaldivar Batista

**TUTORES**

Ing. Michel Álvarez Cancio

Ing. Michel Sariol Fernández

La Habana, junio de 2017

“Año 59 de la Revolución”

## DECLARACIÓN DE AUTORÍA

Por este medio se declara que somos los únicos autores de este trabajo y se autoriza a la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estime pertinente con este trabajo.

Para que así conste se firma la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del 2017.

---

Firma del Autor

(Rosaibis Torres Ruisanchez)

---

Firma del Autor

(Manuel Alejandro Zaldivar  
Batista)

---

Firma del Tutor

(Ing. Michel Álvarez Cancio)

---

Firma del Tutor

(Ing. Michel Fernández  
Sariol)

## **DATOS DE CONTACTO**

### **Ing. Michel Sariol Fernández (msariol@uci.cu):**

Graduado de Ingeniería en Ciencias Informáticas en el año 2015. Ha participado en varios eventos internacionales asociados a la Matemática Numérica y la vinculación de las TIC en la enseñanza de las Matemática. Matriculado en la maestría de ciencias matemáticas de la Universidad de la Habana.

### **Ing. Michel Alvarez Cancio (mcancio@uci.cu):**

Graduado (con honores) de Ingeniería en Ciencias Informáticas en el año 2014. Mención CITMA de “estudiante investigador” en el 2015. Premio del rector a mejor recién graduado en adiestramiento en el 2016. Coordinador principal del grupo de investigación AIRI. Tiene varias publicaciones y presentaciones en eventos sobre procesamiento digital de imágenes y reconocimiento de patrones. Matriculado en la maestría en Señales y Sistemas de la Universidad Central de las Villas. Profesor del departamento de Programación de la Facultad 3 de la Universidad de las Ciencias Informáticas.

## AGRADECIMIENTOS

*Rosaibis Torres Ruisanchez*

*A mis padres y a mi Mamoti por amarme, apoyarme y educarme sin condiciones y sin pretextos, como pretendo educar algún día a mis propios hijos, porque sin ustedes yo no fuese quien soy.*

*A mi tío Jhunion por ser como un hermano para mí y a pesar de su carácter, ser conmigo, diferente como con todos los demás.*

*A mi abuela Zenaida, por ser su nieta preferida y hacerme los mejores chistes del mundo aunque hoy por su condición quizás no se acuerde de quien soy.*

*A Diany, mi mejor amiga por 11 años, y más que eso, la hermana que no tuve, por enseñarme que no importa la distancia, ni las creencias ni nada lo suficientemente fuerte para terminar una amistad de verdad.*

*A Jessica y Adrián por ser mi apoyo cuando más lo necesité.*

*A mis amigas la flaca, por estos 9 años, Elizabet, mi arisca preferida, mi Mami Chuly por estar siempre ahí, Yisel mi torpecita, Elaine por pocas veces quedarse sin palabras.*

*A mi tutor por soportarme hasta en su casa y por adoptarme y ayudarme sin conocerme ni necesitar nada de mí, mil gracias.*

*A mi compañero de tesis, de seis años, de risas, de enojos, de boberías, de cosas importantes, de amistad y muchísimo cariño, sin ti por supuesto que no estuviese dando estos agradecimientos hoy.*

*A mi mejor amigo, mi novio, mi compañero por 11 meses, quien me enseñó el valor del amor, la confianza y sobre todo el respeto, quien me dio fuerzas, ánimos y más que suficiente amor cuando intenté rendirme. Porque esta tesis hubiese sido imposible sin ti. Te amo.*

## DEDICATORIA

*Rosaibis Torres Ruisanchez*

*A mis padres, mi familia, mi tutor, mi compañero de tesis y al amor de mi vida, gracias por enseñarme, amarme y apoyarme en este viaje, los amo.*

*Manuel Alejandro Zaldivar Batista*

*A mi padre Jesús, aunque ya no este físicamente conmigo nunca he dejado de sentir su presencia. Fue el guía y ejemplo de estos años de estudio y la razón por la que hoy soy ingeniero, la estrella a la que siempre uno quiere llegar.*

*A mi madre María Teresa que siempre, aunque sin saberlo me dio la fuerza que necesité cuando no la encontraba en ningún lugar, por inculcar en mí valores que solo pocos tienen, eres mi persona favorita.*

*A mi abuela materna Gertrudis que no la tengo físicamente a mi lado, pero es lo más importante de mi vida en todos los tiempos.*

*A mi hermana Daymara para que le sirva de ejemplo y de impulso en su carrera, que sepa que no hay triunfos sin esfuerzo, ¡mucho esfuerzo!!!*

## RESUMEN

El presente trabajo tiene como objetivo desarrollar un algoritmo de restauración de imágenes digitales con iluminación oblicua provenientes de la lámpara de hendidura. Se realizó un estudio de los aspectos relacionados con las operaciones de catarata, específicamente de la opacidad de la cápsula posterior, que constituye la complicación posoperatoria tardía más frecuente tras una intervención quirúrgica. Se abordan los problemas que presentan los especialistas en la obtención de un diagnóstico cuando la imagen oblicua resultante de la lámpara de hendidura sufre un proceso de degradación. Por lo que se propone un algoritmo para la restauración de las imágenes oblicuas para facilitar el diagnóstico a través de la observación de las mismas. Este algoritmo utiliza una red neuronal basada en el método de *Backpropagation* convirtiéndose en un ciclo iterativo de dos procesos: *backward* y *forward*, que simulan una restauración y una degradación por cada iteración. Esta red neuronal minimiza una función de coste definida por el regulador de variación total, sin requerir entrenamiento previo de la red. Como resultado se obtiene una imagen restaurada proporcionando al especialista lo necesario para diagnosticar correctamente a sus pacientes. Para la validación se utilizaron un conjunto de métricas, demostrando la mejoría de la imagen recuperada y el aumento de sus detalles.

Palabras clave: *Backpropagation, redes neuronales, restauración de imágenes digitales, opacidad de la cápsula posterior, variación total*

## ÍNDICE

Introducción.....	10
Capítulo 1: Fundamentación teórica de la investigación.....	14
1.1 Opacidad de la cápsula posterior.....	14
1.2 Lámpara de hendidura.....	14
1.3 Convolución.....	14
1.3.1 Deconvolución.....	15
1.4 Función dispersora de puntos (PSF).....	15
1.4.1 Modelos de PSF.....	15
1.4.1.1 Emborronamiento por movimiento lineal uniforme.....	15
1.4.1.2 Emborronamiento por desenfoque uniforme.....	16
1.4.1.3 Emborronamiento uniforme bidimensional.....	16
1.4.1.4 Emborronamiento por turbulencia atmosférica.....	16
1.5 Restauración de imágenes.....	17
1.5.1 Proceso de restauración de imágenes.....	18
1.5.2 Necesidad de regularización.....	19
1.5.2.1 Métodos de regularización directa.....	20
1.6 Soluciones existentes.....	21
1.6.1 Algoritmo Lucy-Richardson.....	21
1.6.2 Algoritmo Separación de Hoja Cero.....	21
1.6.3 Técnica de identificación de borrosidad a priori haciendo uso del dominio de frecuencia de ceros.....	22
1.6.4 Algoritmo VCG (Validación Cruzada Generalizada).....	24
1.6.5 Técnicas de restauración con restricciones no paramétricas determinísticas.....	25
1.6.6 Dificultades encontradas.....	27
1.7 Red neuronal.....	28
1.7.1 Ventajas que ofrecen las redes neuronales.....	28
1.7.2 Elementos básicos que componen una red neuronal.....	29
1.7.3 Topología de las redes neuronales.....	29
1.7.3.1 Algoritmo de retropropagación del error.....	30
1.8 Metodología, herramientas y técnicas.....	30

1.8.1 Metodología de desarrollo .....	30
1.8.2 Herramienta de modelado .....	32
1.8.3 Entorno de desarrollo integrado (IDE) .....	32
1.9 Conclusiones del capítulo .....	33
Capítulo 2: Análisis y diseño de la solución .....	34
2.1 Propuesta de solución .....	34
2.2 Funcionalidades.....	37
2.3 Lista de reserva del producto.....	37
2.4 Historia de usuario .....	38
2.5 Fase de planificación .....	39
2.5.1 Estimación de esfuerzo .....	39
2.5.2 Plan de iteraciones.....	40
2.5.3 Plan de entrega .....	40
2.6 Fase de diseño .....	41
2.6.1 Tarjetas de Clases-Responsabilidad-Colaboración .....	41
2.6.2 Estándares de codificación.....	42
2.7 Conclusiones del capítulo .....	42
Capítulo 3: Resultados y Validación .....	43
3.1 Resultados.....	43
3.1.1 Validación de los resultados .....	44
3.2 Fase de prueba.....	46
3.2.1 Pruebas unitarias .....	47
3.2.2 Prueba del camino básico .....	47
3.2.3 Pruebas de aceptación .....	49
3.3 Conclusiones del capítulo .....	51
Conclusiones.....	52
Recomendaciones.....	53
Referencias bibliográficas .....	54
Anexos .....	58

## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Funcionalidades del sistema. ....	37
<b>Tabla 2.</b> Historia de usuario HU 5. ....	39
<b>Tabla 3.</b> Plan de esfuerzo por HU. ....	40
<b>Tabla 4.</b> Plan de iteraciones.....	40
<b>Tabla 5.</b> Plan de entrega de las iteraciones. ....	41
<b>Tabla 6.</b> Tarjeta CRC de la clase RNAParaBID. ....	41
<b>Tabla 7.</b> Resultados de las métricas SSIM, PSNR y SNR. ....	46
<b>Tabla 8.</b> Prueba de caja blanca.....	48
<b>Tabla 9.</b> Caso de prueba de aceptación HU 5.....	51
<b>Tabla 10.</b> Historia de usuario HU 1 .....	58
<b>Tabla 11.</b> Historia de usuario HU 2 .....	58
<b>Tabla 12.</b> Historia de usuario HU 3 .....	58
<b>Tabla 13.</b> Historia de usuario HU 4 .....	59
<b>Tabla 14.</b> Historia de usuario HU 6 .....	59
<b>Tabla 15.</b> Historia de usuario HU 7 .....	59
<b>Tabla 16.</b> Historia de usuario HU 8 .....	60
<b>Tabla 17.</b> Historia de usuario HU 9 .....	60
<b>Tabla 18.</b> Historia de usuario 10 .....	60
<b>Tabla 19.</b> Tarjeta CRC de la clase SerieD.....	61

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Proceso completo de adquisición y restauración de imágenes. ....	18
<b>Figura 2.</b> Método iterativo de la deconvolución ciega. ....	26
<b>Figura 3.</b> Ejemplo de una red neuronal totalmente conectada. ....	29
<b>Figura 4.</b> Estrella que representa al proyecto según el método de Boehm y Turner. ....	32
<b>Figura 5.</b> Esquema de la propuesta de solución. ....	35
<b>Figura 6.</b> Imagen borrosa e imagen restaurada. ....	43
<b>Figura 7.</b> Imagen borrosa e imagen restaurada ....	43
<b>Figura 8.</b> Imagen borrosa e imagen restaurada ....	44
<b>Figura 9.</b> Imagen original e imagen emborronada.....	45
<b>Figura 10.</b> Imagen original e imagen emborronada.....	45
<b>Figura 11.</b> Imagen original e imagen emborronada.....	45
<b>Figura 12.</b> Resultados de las pruebas de aceptación. ....	50

## Introducción

Durante la historia de la humanidad el hombre ha intentado mantener y mejorar sus condiciones de vida a través del conocimiento del mundo que se encuentra a su alrededor, ha decidido imponerse al ambiente en el que habita valiéndose para ello de la ciencia. Actualmente resulta evidente la relación absoluta de la sociedad con la ciencia y la tecnología, pues estos dos elementos han desempeñado un papel importante en el desarrollo de múltiples actividades y en la mayoría de los sectores sociales, como son: la educación, la industria y la medicina, entre otros.

En la medicina el uso de las tecnologías actuales ha creado las condiciones para que se pueda incrementar en gran medida el conocimiento de patologías para la investigación médica, de igual forma son un componente esencial en la planificación de diagnósticos y tratamientos de enfermedades. En el campo de la medicina hacer diagnósticos utilizando imágenes es invaluable, y para ello se recurren a modalidades de adquisición de imágenes, como son: la Lámpara de Hendidura o Biomicroscopio, la Mamografía Digital y la Resonancia Magnética por mencionar algunas, para luego aplicar métodos de restauración de imágenes y así tratar de erradicar enfermedades como la catarata, el cáncer de mama, etc.

La catarata es la principal causa de ceguera y se estima que afecta a unas 20 millones de personas. Debido al envejecimiento de la población, la ceguera por catarata aumentará a unas 50 millones de personas al año 2050, a menos que se realice una intervención agresiva operando al triple de lo actual. En el mundo ha aumentado la productividad quirúrgica, pero a la vez se ha generado una alta tasa de malos resultados visuales de la cirugía. En la India, del 15% al 25% de los pacientes operados de catarata ven menos de 5/50 con la corrección disponible, en China, casi 40% de los ojos tuvieron malos resultados y la situación en África pudiera ser similar (Barría, 2012).

En el año 2013, la Organización Mundial de la Visión, en su Asamblea Anual aprobó el plan de acción global 2014-2018, que tiene como objetivo reducir el impedimento visual por un 25%. En términos generales, la cobertura de cirugía de catarata se ha mantenido relativamente baja y varía entre una región a otra. El aumento en las tasas de cirugía de catarata no siempre resulta en una reducción de las prevalencias de ceguera por catarata. La prevalencia de ceguera por catarata solamente se reducirá cuando se operen a los pacientes que tienen agudezas visuales (Barría, 2012).

Por ello la creación de sistemas integrales de salud ocular con centros de entrenamiento tendría un gran impacto en la situación de la catarata en América Latina y en el mundo. Las estrategias implementadas por la Organización Mundial de la Salud, su escritorio regional la Organización Panamericana de Salud, y la Agencia Internacional para la Prevención de la Ceguera, han contribuido de una manera efectiva a reducir la prevalencia de ceguera y aumentar los servicios oculares. El liderazgo en oftalmología de la Academia Americana de Oftalmología y de la Asociación Panamericana de Oftalmología, han unido esfuerzos a través de programas educativos y talleres

de destrezas e intercambio de conocimientos, para mejorar el entrenamiento de los oftalmólogos durante los años de formación en sus residencias de post-grado y también en el ámbito de la educación continuada de los profesionales de América Latina y el mundo (Barría, 2012).

La catarata es una opacidad del cristalino del ojo, evitando que llegue totalmente la luz que viaja a los ojos a través de la pupila y se enfoca sobre la retina. Esta condición de opacidad del lente ocular es la causante de pérdida total o parcial de la vista (Boyd, 2014).

Las operaciones de catarata consisten en técnicas quirúrgicas basadas en incisiones autosellables sin suturas, de mediano y pequeño tamaño. La cirugía se realiza de forma ambulatoria con un seguimiento de consultas postoperatorias (Pérez Candelaria, 2016). La opacidad de la cápsula posterior es actualmente uno de los aspectos más importantes en la cirugía de catarata de los tiempos modernos. Sigue siendo la complicación posoperatoria tardía más frecuente tras la cirugía de catarata asociada con disminución de la agudeza visual, deterioro de la sensibilidad al contraste y problemas de deslumbramiento, que conllevan importantes repercusiones sociales, médicas y económicas.

Con el objetivo de evaluar la opacidad de la cápsula posterior se utiliza la lámpara de hendidura que está diseñada para que los ojos revelen una historia que podría indicar la presencia de varios tipos de enfermedades oculares y posibles problemas de visión. Las pruebas de lámpara de hendidura son formas de amplificar lo que sucede en la superficie del ojo, en la parte anterior, en el interior y, lo que es más importante, lo que sucede en la parte posterior del ojo, es decir, en la retina (Transitions Optical Inc, 2012). La lámpara de hendidura proporciona como resultado imágenes que no son de alta calidad, consecuencia de la iluminación oblicua con la que se obtienen estas.

En Cuba durante dos años consecutivos (2013-2014), se logró superar la tasa ideal propuesta por la Organización Mundial de la Salud (OMS), de 3 000 cirugías por cada millón de habitantes (33 000 mil cirugías por año), al realizarse 35 339 intervenciones en el año 2013 (tasa 3 166) y 37 544 en el año 2014 (tasa 3 349). Esta última cifra constituye un récord histórico en el país. También es válido destacar, que el Centro de Microcirugía Ocular del Instituto Cubano de Oftalmología es líder de la cirugía de catarata en el país y, sin lugar a dudas, en buena parte del mundo. Desde su inauguración se han operado en esta institución 265 101 pacientes; en los últimos cinco años se han efectuado 43 674 cirugías de cataratas, y se han alcanzado cifras récord: 9 523 en el año 2013 y 11 501 en el año 2014 (Rios Caso, 2015).

Además, la mayor de las Antillas, creó un proyecto llamado Misión Milagro con el objetivo de mejorar la ceguera prevenible, donde entran varias patologías, siendo la más frecuente e importante la catarata. Con este programa se les devuelve la visión a muchas personas afectadas en el mundo, que no tienen recursos económicos para costearse una operación, tratándose que tuviera medicina privada (Pérez Candelaria, 2016).

En Julio del 2004 comienza en el país la Misión Milagro, devolviéndole la visión a miles de pacientes de muchos países de América Latina y el Caribe, entre los que se encuentra Venezuela, México, Bolivia, Barbados y Nicaragua (Pérez Candelaria, 2016).

Hasta la fecha se han operados alrededor de 65 000 pacientes de diferentes países con una participación importante de Latinos y del Área Caribe entre los que se encuentran: Panamá, Honduras, Bolivia, Brasil, Paraguay, Uruguay, Argentina y México (Pérez Candelaria, 2016).

Actualmente en Cuba para establecer un diagnóstico de la opacidad en los pacientes operados de catarata se utiliza el software PANDOC, que provee al oftalmólogo de una herramienta por medio de la cual este es capaz de cuantificar numéricamente y detectar diferencias de opacidad (a veces imperceptibles para el ojo humano), para lograr una evaluación objetiva del grado de la misma. Para ello el especialista debe identificar manualmente los puntos que considera opacos, valoración que depende mucho de su nivel de experticia, y se desea evitar este grado de subjetividad. Dentro de las imágenes usadas en los pacientes operados de cataratas se encuentran las imágenes en iluminación oblicua provenientes de la lámpara de hendidura.

La precisión del resultado de la cuantificación está sujeta a los parámetros de calidad de la imagen, ya que esta, en varias ocasiones se ven afectadas por elementos como desenfoque y emborronamiento propios del ambiente y las tecnologías de donde fue tomada la misma. Resaltando como necesidad la obtención de imágenes con poca afectación de emborronamiento y desenfoque para el análisis por parte del software.

Después de analizar la situación problemática se identifica el siguiente **problema a resolver**: ¿Cómo atenuar los niveles de emborronamiento en las imágenes oblicuas provenientes de la lámpara de hendidura? El **objeto de estudio** correspondiente al presente problema: Procesamiento de imágenes digitales. Se define como **campo de acción** asociado a este objeto de estudio: Restauración de imágenes oblicuas provenientes de la lámpara de hendidura. Para solucionar el problema identificado se plantea el siguiente **objetivo general**: Desarrollar un algoritmo para la restauración de imágenes oblicuas provenientes de la lámpara de hendidura.

Para dar cumplimiento al objetivo general planteado se definen los siguientes **objetivos específicos**:

1. Analizar los principales elementos relacionados con la restauración de imágenes digitales.
2. Analizar algoritmos desarrollados para la restauración de imágenes digitales.
3. Implementar un algoritmo para la restauración automática de detalles en imágenes oblicuas provenientes de la lámpara de hendidura.
4. Validar la propuesta de solución.

Teniendo en cuenta el problema a resolver se formuló la siguiente idea a defender:

**Idea a defender:** Con el desarrollo de un algoritmo para la restauración de imágenes oblicuas provenientes de la lámpara de hendidura, se atenuarán sus niveles de emborronamiento.

Para el desarrollo del algoritmo se utilizaron métodos de investigación. A continuación, se listan los mismos:

### **Métodos Teóricos**

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

**Histórico-Lógico:** Gracias a este método se realizó un estudio con el fin de conocer la evolución y desarrollo de las técnicas existentes en la actualidad enfocadas a la validación de la restauración en imágenes médicas.

### **Métodos Empíricos**

**Medición:** Este método se aplicará de manera continua durante todo el proceso investigativo, observando los resultados de los algoritmos que se van a utilizar y así decidir los que mejor se adecuan al proceso.

**Experimento:** Este método permitió probar las distintas técnicas y algoritmos para la validación de la restauración en imágenes médicas.

El documento está estructurado en tres capítulos, siendo estos:

**Capítulo 1:** *Fundamentación teórica de la investigación*, incluye un análisis y revisión del estado del arte, a nivel internacional y nacional, de algoritmos de restauración de imágenes. Se hace una descripción de las tendencias, técnicas, tecnologías y herramientas usadas para el desarrollo del algoritmo.

**Capítulo 2:** *Análisis y diseño de la solución*, tiene como objetivo describir las actividades desarrolladas durante todo el proceso de análisis y diseño. Se detalla la propuesta de solución y se describe su arquitectura. Los artefactos que plantea la metodología escogida contribuyeron al desarrollo de las nuevas funcionalidades agregadas.

**Capítulo 3:** *Validación de la propuesta de solución*, donde se desarrollan las pruebas realizadas al algoritmo, propuestas por la metodología de desarrollo escogida, además se le aplican un conjunto de métricas para realizar la validación de los resultados.

# Capítulo 1: Fundamentación teórica de la investigación

En este capítulo se ofrece un marco conceptual sobre el problema a resolver con el objetivo de profundizar en la investigación. Además, se abordan conceptos, algunos de los cuales son: Opacidad de la cápsula posterior, Lámpara de Hendidura, Restauración de imágenes y Redes neuronales. Se utiliza el método de Boehm y Turner para identificar el enfoque más adecuado en la selección de la metodología. Se exponen y explican el lenguaje de programación, herramientas y tecnologías a utilizar en el desarrollo del algoritmo justificando cada una de ellas.

## 1.1 Opacidad de la cápsula posterior.

Después de una cirugía de catarata, puede ocurrir con el paso del tiempo que la membrana donde se apoya el lente intraocular se opacifique y el paciente manifieste disminución de la visión o halos alrededor de las luces (Dr. Jhon Jairo Aristizábal, 2016). En el campo de la oftalmología esto es conocido como OCP y constituye la complicación tardía más importante en la cirugía de cataratas en los tiempos modernos (López, et al., 2010).

## 1.2 Lámpara de hendidura

La lámpara de hendidura es un equipo oftalmológico que se utiliza para tomar imágenes médicas del ojo humano. Cuando un oftalmólogo va a examinar a un paciente con la lámpara de hendidura el equipo proyecta un haz de luz que llega hasta el fondo del ojo y permite al oftalmólogo observar todas las partes del ojo humano. La lámpara provoca en las imágenes un efecto llamado imágenes de *Purkinje* que no es más que los reflejos que en ese momento está viendo el ojo y las cuales valen para que el oftalmólogo pueda definir qué región del ojo está dañada, por ejemplo, la afectación del cristalino en las cataratas.

Con el fin de detectar la OCP se toman imágenes en retroiluminación provenientes de la lámpara de hendidura, dichas imágenes pueden presentar borrosidad, ruido, difracción, o un canal de la observación fuera de control. Para la restauración de detalles en estas imágenes se decide analizar los algoritmos de restauración en imágenes digitales.

## 1.3 Convolución

La operación convolución es una operación sobre un vecindario en el cual cada píxel de salida es una suma ponderada de los píxeles vecinos, consiste en la aplicación de un filtro o máscara de un tamaño dado (3x3, 5x5,...), píxel a píxel, operando con los valores de la imagen en el vecindario de cada píxel y con los pesos o coeficientes de ponderación que conforman el filtro, los cuales se eligen en función del objetivo del mismo. La matriz de pesos es llamada núcleo de convolución, también conocido como filtro. A nivel espacial en la convolución el *kernel* se transforma y las filas se convierten en columnas, pero a nivel de frecuencia es una multiplicación punto a punto. Se evalúa el píxel central de la ventana de acuerdo con los píxeles de alrededor y sus valores de ponderación. Cuando un nuevo valor es así calculado, se desplaza la ventana sobre el siguiente píxel, realizando la misma operación. Este proceso de evaluar la vecindad ponderada del píxel se denomina

“convolución bidimensional”, y a la matriz de filtro se le conoce como “*kernel* de convolución” (Ruiz Fernández, 2009).

### 1.3.1 Deconvolución

La deconvolución es la técnica inversa a la convolución la cual se encarga de hallar cómo era una imagen original luego que ha sido degradada de alguna forma (por el sistema de observación) (Baume, 2014) y puede clasificarse en:

- ✓ Deconvolución clásica: Busca un estimado de la imagen real asumiendo que la borrosidad es conocida (Campisi, et al., 2001).
- ✓ Deconvolución a ciegas: Encuentra una imagen degradada sin que se requiera determinación a priori de los sistemas asociados de la función dispersora de puntos (PSF) o usando solamente información parcial sobre la PSF (Jiang, et al., 2003).

### 1.4 Función dispersora de puntos (PSF)

Cuando se toma una fotografía y ocurre un difuminado, la intensidad de cada píxel en la imagen real se propaga sobre sus píxeles vecinos. Como ilustración visual a esto, se supone que la imagen real es totalmente negra excepto por un píxel de alta intensidad, que se llamará punto fuente. La función que describe el difuminado producido desde el punto fuente se denomina función dispersora de puntos o por sus siglas en inglés PSF (Montenegro Vargas, 2010).

#### 1.4.1 Modelos de PSF

A continuación se resumen los emborronamientos más comunes a partir del estudio que han realizado diferentes artículos de restauración de los modelos de PSF que mejor representan las degradaciones en los diferentes campos de aplicación (Banham, 1994) (Gennery, 2013) (Lee, 2000).

Dado que las imágenes real y observada son funciones reales y positivas, consecuentemente la PSF también lo es. Además, ha de tenerse en cuenta que la degradación en un sistema de captación de imágenes es una operación pasiva que ni absorbe ni genera energía sobre la imagen, es decir, toda la energía que acumula un punto de la imagen original es conservada tras pasar el filtro de la PSF. Esto se traduce formalmente en que la ecuación de la PSF sea definida de la siguiente manera:

$$\sum_{i=0}^{M_1-1} \sum_{j=0}^{M_2-1} h(i, j) = 1 \quad (1)$$

##### 1.4.1.1 Emborronamiento por movimiento lineal uniforme

Son varios los fenómenos de degradación que producen la función de PSF de movimiento: traslaciones, rotaciones y cambios de escala. Sin embargo, el más común de estos efectos es el correspondiente únicamente a la traslación (movimiento lineal uniforme). Así pues, considerando una escena que se desplaza respecto a la cámara a una velocidad constante  $v$ , con un ángulo  $\phi$  respecto al eje horizontal y durante un período de tiempo  $T$ , se define la longitud del movimiento como  $d = vT$  y la PSF se representa por:

$$h(i, j) = \begin{cases} \frac{1}{d}, & \text{si } \sqrt{i^2 + j^2} \leq \frac{d}{2} \\ 0, & \text{en el resto} \end{cases} = -\tan \phi \quad (2)$$

Esta expresión asume que todos los objetos de la escena se están moviendo a la vez. En el caso de que alguno se mueva de forma independiente respecto al resto de la escena, el modelado del emborronamiento da lugar a una PSF variante espacial.

#### 1.4.1.2 Emborronamiento por desenfoque uniforme

El desenfoque de una imagen captada por una cámara se modela en cada punto por el denominado círculo de confusión, cuyo radio depende de la longitud focal, la apertura de la lente, la longitud de onda de la luz y la distancia entre la cámara y el objeto. Si el grado del desenfoque (radio del círculo) es grande respecto a las longitudes de onda de la luz que atraviesan la cámara, se utiliza una aproximación geométrica que da lugar a una distribución uniforme dentro del círculo de confusión. En estas condiciones, la PSF se define como:

$$h(i, j) = \begin{cases} \frac{1}{\pi R^2}, & \text{si } \sqrt{i^2 + j^2} \leq R \\ 0, & \text{en el resto} \end{cases} \quad (3)$$

Donde  $R$  corresponde al radio de la PSF. Este modelo asume que la distancia del objeto desde la cámara es constante en toda la imagen, lo que le permite ser una PSF invariante espacial.

#### 1.4.1.3 Emborronamiento uniforme bidimensional

Esta forma de emborronamiento supone una aproximación a la degradación por desenfoque uniforme y se emplea en un gran número de simulaciones sobre restauración de imágenes. Su expresión viene dada por:

$$h(i, j) = \begin{cases} \frac{1}{d^2}, & \text{si } -\frac{d}{2} \leq i, j \leq \frac{d}{2} \\ 0, & \text{en el resto} \end{cases} \quad (4)$$

#### 1.4.1.4 Emborronamiento por turbulencia atmosférica

La degradación debida a los cambios en las condiciones de refracción de la atmósfera tiene una importante influencia en la calidad de las imágenes para teledetección. Este tipo de emborronamiento sin embargo no admite un modelo determinista por su dependencia respecto a factores tales como la temperatura, el tiempo de exposición o las condiciones meteorológicas de la adquisición. Si se considera una exposición prolongada, lo cual suele ser frecuente en las observaciones astronómicas, una buena aproximación de la turbulencia atmosférica es una PSF gaussiana que responde a la expresión:

$$h(i, j) = \frac{1}{2\pi\sigma_R^2} \exp\left\{-\frac{i^2+j^2}{2\sigma_R^2}\right\} \quad (5)$$

Donde  $\sigma_R^2$  controla la severidad de la degradación.

## 1.5 Restauración de imágenes

El proceso de observación de las imágenes no es perfecto a medida que se presenten fenómenos tales como borrado, ruido u otra degradación en el proceso de obtención y almacenamiento de las mismas. Por lo que se define que la restauración de imágenes es el arte de mejorar la calidad de una imagen, compensando o eliminando las distorsiones y el deterioro de la imagen (Triana, y otros, 2013). La clave para ello está en resolver un problema inverso donde es apropiado incorporar información a priori de la imagen a recuperar en el proceso de restauración.

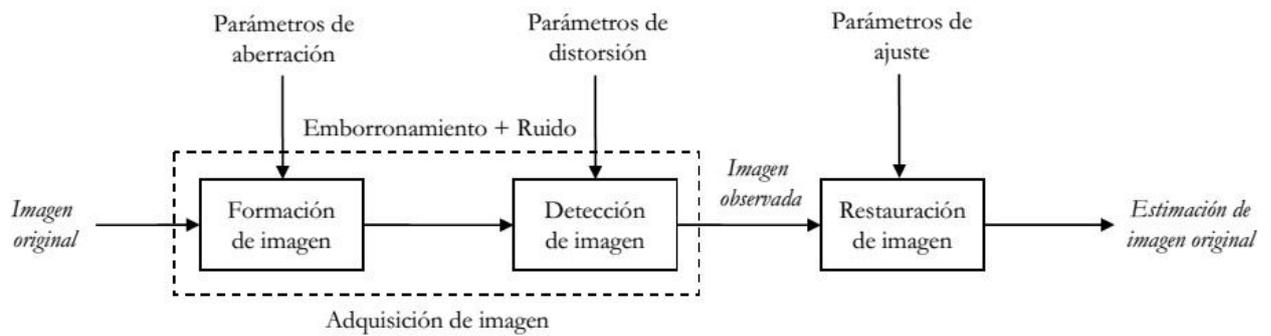
Se pueden distinguir dos fuentes de degradación:

- ✓ Emborronamiento (*blur*): Debido al movimiento relativo entre el detector y el objeto, turbulencia atmosférica, imágenes fuera de foco, humedad, variación de la temperatura, entre otros.
- ✓ Ruido (*noise*): Debido a diversas causas como los granos en una fotografía o ruido electrónico (ruido Johnson) y ruido de cuantización en sistemas digitales.

Sin importar el tipo de degradación presente en la imagen es posible pretender solucionarlo a partir de una única toma o múltiples imágenes, siendo realizada la recuperación de la información en ambos casos, mediante la operación de deconvolución, pues el emborronamiento suele representarse como un operador lineal de convolución que se invierte durante la restauración (Santiago Cabello, 2011).

Invertir el proceso de degradación introduce un número importante de problemas. En primer lugar, la inversión directa del proceso implica que el ruido sea amplificado en el resultado y, en el caso que el proceso no sea invertible, son varias las soluciones posibles. Por otro lado, la imagen observada es una versión truncada de la escena, así pues, se requiere información en los límites de la imagen para poder realizar la inversión. Finalmente, todas estas opciones asumen tener un conocimiento preciso del modelo de degradación de la imagen, lo cual no suele ser cierto ya sea porque se desconozca, no sea fácilmente accesible, o se conozca solo en términos probabilísticos.

La Figura 1 representa los diferentes parámetros que forman parte de un proceso de adquisición y, en consecuencia, que son necesarios para modelar la degradación de una imagen. Dicho modelo depende de elementos como la función característica del detector, u otros parámetros deterministas como los parámetros de distorsión (p.ej., ganancia, cuantificación) y los parámetros de aberración (p.ej., ancho de banda). Una vez obtenido este modelo es necesario ajustarlo a las condiciones reales de experimentación y, en definitiva, se requiere una calibración posterior de los datos con otros parámetros de ajuste.



**Figura 1.** Proceso completo de adquisición y restauración de imágenes.

Para conseguir una buena estimación de la imagen original, los algoritmos de restauración necesitan incorporar conocimiento a priori sobre los tres elementos principales del modelo: el emborronamiento, el ruido y la propia imagen original. La dependencia que presenta la restauración a toda esta información a priori es muy acusada y, de hecho, pequeñas variaciones respecto a los valores reales suponen resultados de restauración muy alejados de su óptimo.

Existen varios métodos de restauración de imágenes que se encargan de estimar la imagen original a partir de los problemas que se puedan presentar, como el emborronamiento. Para darle solución a esto se han planteado diversas soluciones, dentro de las cuales se encuentra la utilización de redes neuronales, donde usualmente se toman cada una de las neuronas de entradas de la misma como el nivel de intensidad de cada píxel de la imagen original (Santiago Cabello, 2011) (Sun, 2000) (Wang, y otros, 2005) (Wu, y otros, 2007).

### 1.5.1 Proceso de restauración de imágenes

El proceso de restauración de imágenes equivale a la resolución de un problema inverso en el que se ha de obtener el vector  $\hat{x}$ , conocida la imagen observada  $y$ , además de la matriz del sistema de captación  $H$

$$\hat{x} | y = H\hat{x} + n \quad (6)$$

La inversión directa de la matriz no es suficiente para resolver esta ecuación, pues la presencia del ruido estocástico origina una situación de mal condicionamiento (Santiago Cabello, 2011).

Un problema está bien condicionado, dado los criterios de Hadamard, si cumple que (Engl, y otros, 1996) (Hansen, 1998):

1. Para cualquier dato, existe una solución. (Existencia)
2. La solución es única. (Unicidad)
3. La solución depende de forma continua de los datos. (Estabilidad)

En primer lugar, la presencia de ruido puede suponer que no exista ninguna  $\hat{x}$  que resuelva la ecuación con igualdad matemática. Seguidamente aparece el problema de unicidad de la solución donde el conjunto nulo de  $H$  representa las imágenes de entrada que producen una salida nula, si

dicho espacio no es vacío entonces existen componentes de la imagen original que no son observadas en  $y$ . En estos casos, son infinitas las posibilidades de  $\hat{x}$  que producen la misma imagen observada y se necesita un criterio para elegir la “correcta” para el problema inverso.

El tercer criterio respecto a la estabilidad es el más crítico, puesto que la solución ha de mantenerse inalterada frente a las perturbaciones en la observación de la imagen, o lo que es lo mismo, a la presencia del ruido.

Por último, la ecuación sólo representa la solución teniendo en cuenta la imagen observada  $y$ , pero no dice nada en lo que se refiere a conocimiento a priori de la solución. Por lo general, cierta información a priori estará disponible a la hora de resolver el problema de restauración y, por tanto, se requiere algún medio de incluirla en la solución. La regularización se emplea como instrumento matemático para resolver todos estos problemas que originan el mal condicionamiento de la restauración de imágenes (Santiago Cabello, 2011).

### 1.5.2 Necesidad de regularización

El propósito de esta técnica matemática es incluir información a priori que permita estabilizar la imagen restaurada frente a la magnificación del ruido.

La regularización matemática se puede generalizar como un problema alternativo a la restauración original que cumple con los tres elementos necesarios para ser bien condicionado, donde se establece una función de coste  $\mathcal{C} : \mathbb{R}^L \rightarrow \mathbb{R}$  que cumple con la siguiente estructura (Hansen, 1998):

$$\mathcal{C}(x) = \mathcal{D}(x) + \lambda \mathcal{R}(x) \quad (7)$$

Donde  $\mathcal{D}(x)$  es el término de datos,  $\mathcal{R}(x)$  es el término de regularización y  $\lambda$  es el parámetro de regularización tal que  $\lambda \geq 0$ .

Analizando cada elemento por separado:

**$\mathcal{D}(x)$ :** Es el término de datos que se encarga de asegurar la fidelidad de la imagen estimada a los datos de la imagen observada.

**$\mathcal{R}(x)$ :** Es el término de regularización que se construye usando información a priori sobre la imagen original y favorece típicamente la aparición de imágenes restauradas con propiedades de suavizado. También se conoce como el término de penalización o la norma de suavizado, y se encarga de penalizar las altas frecuencias en la solución y dejar pasar sin regularizar sólo las bajas frecuencias.

**$\lambda$ :** Es el parámetro de regularización que determina la influencia de ambos términos, controlando el impacto de los datos y el correspondiente a la información a priori. El valor de este parámetro es crítico para conseguir un buen equilibrio entre la estabilidad de la solución (propagación del ruido) y la calidad de los datos. De hecho, si se asocia un valor de este parámetro demasiado alto, se obtiene una solución sobre-regularizada, donde el suavizado en la imagen restaurada es excesivo;

por el contrario, si el parámetro es demasiado bajo se llegaría a la misma solución inestable que la conseguida por mínimos cuadrados en presencia de ruido. Por ello, se tiene que llegar a un valor de compromiso.

### 1.5.2.1 Métodos de regularización directa

Reciben este nombre los métodos de restauración cuya solución se obtiene resolviendo la función de optimización  $\mathcal{C}(x)$ . Existen dos métodos de regularización directa: regularización Tikhonov y regularización por variación total.

#### Regularización Tikhonov

Este método en particular se basa en la función de coste (Santiago Cabello, 2011):

$$\mathcal{C}(x) = \left\{ \begin{array}{l} \mathcal{D}(x) = \frac{1}{2} \| y - Hx \|_2^2 \\ \lambda \mathcal{R}(x) = \frac{\lambda}{2} \| Dx \|_2^2 \end{array} \right\} \quad (8)$$

Donde  $D \in \mathbb{R}^{U \times L}$  representa la matriz de convolución de un filtro paso alto  $d \in \mathbb{R}^U$ , que se elige normalmente como el operador Laplaciano. El hecho de incluir este tipo de regularización implica que la solución tenga una limitada energía de alta frecuencia, dado que son las bajas frecuencias las que pasan a la solución sin ser regularizadas. Como resultado, se obtiene una imagen restaurada en la que los bordes de la escena pierden nitidez, cuanto mayor sea la regularización. Esta última situación es la que motiva la aplicación de otras regularizaciones no cuadráticas.

#### Regularización por variación total (TV)

Como alternativa a la norma de *Tikhonov*, la regularización por TV propuesta por *Rudin, Osher y Fatemi* (Osher, y otros, 1992) se ha utilizado mayoritariamente en las últimas publicaciones de restauración (Wu, y otros, 2007) (Babacan, y otros, 2009) (Oliveira, y otros, 2009). En este caso, el problema de restauración se define por una nueva función de coste:

$$\mathcal{C}(x) = \left\{ \begin{array}{l} \mathcal{D}(x) = \frac{1}{2} \| y - Hx \|_2^2 \\ \lambda \mathcal{R}(x) = \lambda \text{TV}(x) \end{array} \right\} \quad (9)$$

Donde TV ( $\cdot$ ) simboliza el operador de variación total.

El operador TV es una integración de todos los gradientes de una función en cualquier punto de su dominio. Y dado que el gradiente es una medida de la variación de la función en un punto, entonces la variación total representa la cantidad total de cambio que contiene la función.

La razón de utilizar el operador TV como término de regularización es gracias a su propiedad de incorporar los gradientes más acusados y variables en la solución. Pensando en imágenes, esto supone que la imagen restaurada además de evitar las oscilaciones producidas por el ruido, también mantiene la información de los bordes de los objetos de la escena. Los resultados comparativos entre los métodos de *Tikhonov* y TV son muy visibles desde el punto de vista subjetivo y han

convertido a la regularización por variación total en la herramienta elegida para la restauración de imágenes reales (Chan, y otros, 2005).

## 1.6 Soluciones existentes

A continuación, se ofrece una serie de algoritmos de restauración que se encargan de darle solución a problemas de degradación en imágenes digitales.

### 1.6.1 Algoritmo Lucy-Richardson

Una técnica popularmente utilizada es el algoritmo de doble iteración desarrollado por Holmes, basado en el algoritmo EM (Esperanza - Maximización). El algoritmo EM es un algoritmo de restauración de imágenes y ha sido ampliamente utilizado en muchas aplicaciones bajo diferentes nombres como "Lucy" y "Richard", etc. En el caso de un modelo de imagen lineal invariante al espacio  $g(x) = p(x) \otimes \lambda(x)$ , donde  $g(x)$  es el dato observado,  $p(x)$  caracteriza el sistema de imagen y su PSF, y  $\lambda(x)$  es la imagen original, el algoritmo EM itera de la siguiente manera (Jiang, et al., 2003):

$$\text{➤ } \lambda_{k+1}(x) = \lambda_k(x) \cdot p(-x) \otimes (g(x)/(p(x) \otimes \lambda_k(x)))$$

Puesto que la imagen original y la PSF son simétricas en  $g(x) = p(x) \otimes \lambda(x)$ , se obtiene el siguiente algoritmo de iteración doble intercambiando la imagen y la PSF:

$$\text{➤ } \lambda_{k+1}(x) = \lambda_k(x) \cdot p_k(-x) \otimes \left( \frac{g(x)}{p(x) \otimes \lambda_k(x)} \right)$$

$$\text{➤ } p_{k+1}(x) = \frac{p_k(x)}{\Lambda} \cdot \lambda_k(-x) \otimes \left( \frac{g(x)}{p(x) \otimes \lambda_k(x)} \right)$$

Donde  $\Lambda = \sum g(x)$ .

Aunque la convergencia del algoritmo EM está bien establecida, la convergencia del algoritmo de iteración doble no está garantizada, por eso la mayor desventaja de la deconvolución ciega del tipo de doble iteración es que la solución depende de los valores iniciales (Jiang, et al., 2003).

### 1.6.2 Algoritmo Separación de Hoja Cero

El método de Hojas Cero ha recibido atención porque proporciona información valiosa sobre el problema de la deconvolución ciega en múltiples dimensiones. Lane y Bates (Lane, et al., 1987) han demostrado que cualquier imagen degradada  $g$ , formada por la convolución de varios componentes individuales  $f_1, f_2 \dots f_n$ , con soporte compacto, es automáticamente deconvolucionable siempre que su dimensión sea mayor que uno. Su argumento descansa en las propiedades analíticas de la Transformación Z (ZT) en múltiples dimensiones. Los ceros de la ZT de los componentes  $K$ -dimensionales  $f_i$  son casi siempre continuos y se encuentran en una hipersuperficie  $(2K-2)$ -dimensional. Las hojas cero pueden muy raramente poseer singularidades. Al separar estas hipersuperficies, se reconocen los componentes individuales hasta un factor de escala complejo (Kundur, et al., 1996). A continuación, la descripción del algoritmo:

**I. Asociación y agrupación de raíces polinomiales.**

1. Se fija  $z_1 = z_0$  (se suele elegir  $z_1 = 1$ ). Las raíces del polinomio resultante en  $z_2$  se calculan con un solucionador de raíz estándar. Las raíces se denotan  $Z_i(z_0)$ ,  $i = 1, 2, \dots, N_r$  donde  $N_r$  es el número total de raíces.
2. Las raíces se dividen en dos grupos,  $S_1$   $S_2$  donde cada raíz continua perteneciente al mismo factor ZT,  $Z_1(z_0)$ , se asigna a  $S_1$ .

Para cada par de raíces ( $Z_i(z_0)$ ,  $Z_j(z_0)$ ) para  $i$  distinto de  $j$ :

- a) La variable compleja  $z_1$ , se mueve a lo largo de una trayectoria matemáticamente derivada que hace que  $Z_i(z_0)$  y  $Z_j(z_0)$  se fusionen en puntos comunes.
- b) En el punto donde el par de raíces se fusionan, se realiza una prueba de análisis para determinar si las raíces pertenecen o no al mismo factor ZT. Se agrupan en  $S_1$  y  $S_2$  en consecuencia.

**II. Seguimiento de raíz y generación de factores FT.**

1. La ubicación de las raíces en  $S_1$  y  $S_2$  se rastrea a la vez que  $z_1$  se mueve secuencialmente a puntos discretos en el círculo de la unidad.
2. En cada punto del círculo unitario, se genera una columna de cada matriz de factor FT.

**III. I y II se repiten cambiando las funciones de  $z_1$  y  $z_2$ .**

**IV. Matriz del Factor FT Ecuación y Deconvolución.**

1. Las dos matrices de factor FT resultantes son escaladas y fusionadas.
2. La estimación de la imagen  $\hat{f}(x, y)$  y la PSF estimada  $\hat{h}(z, y)$  se generan tomando la transformada rápida de Fourier inversa (FFT) de las matrices de factor FT.

La técnica de separación de hoja cero es útil para proporcionar información sobre el problema de la deconvolución ciega. Basado en el concepto de cero láminas, se ha demostrado que el procesamiento de imágenes en tres o más dimensiones es un problema sobre-determinado si la imagen o la amplitud de Fourier se miden a la tasa de *Nyquist*. El método de hoja cero también se puede utilizar para deconvolucionar simultáneamente más de dos componentes y puede determinar el número de señales irreducibles de las que se compone una señal dada. Aunque el concepto de hojas cero es útil conceptualmente, existen muchos inconvenientes prácticos a este acercamiento de la deconvolución ciega. Básicamente, el algoritmo es altamente sensible al ruido, el algoritmo más rápido hasta la fecha tiene una complejidad computacional de  $O(Ng^4)$ , donde  $Ng$  es el número de píxeles de la imagen borrosa, y el algoritmo es propenso a la inexactitud numérica para grandes tamaños de datos (Kundur, et al., 1996).

**1.6.3 Técnica de identificación de borrosidad a priori haciendo uso del dominio de frecuencia de ceros**

Los métodos de identificación de desenfoque a priori realizan una deconvolución ciega identificando la PSF antes de la restauración. Esta clase de técnicas hace suposiciones sobre las características

de la PSF, tales como la simetría, y la disponibilidad de una forma paramétrica conocida del desenfoque. Son la clase más simple de métodos de deconvolución ciega a implementar y tienen requisitos computacionales bajos. Son aplicables a situaciones en las que se sabe que la imagen verdadera posee características especiales, o bien cuando se sabe que la PSF es de una forma paramétrica especial (Kundur, et al., 1996). A continuación, la descripción de la técnica:

- I. La imagen se divide en marcos más pequeños (posiblemente superpuestos), cada uno de los cuales es lo suficientemente grande como para contener la PSF. Cada subsección se muestra en ventanas para reducir los efectos de borde. El siguiente modelo contiene una aproximación:

$$\text{➤ } g_i(x, y) = f_i(x, y) * h(x, y) + n_i(x, y), i = 1, 2 \dots N_p$$

Donde  $N_p$ , es el número total de subsecciones particionadas. La presencia de ruido aditivo se incluye en el modelo.

- II. Se identifican los nullos del dominio de frecuencia de  $h(x, y)$ .

1. El espectro de potencia de cada subsección  $g_i(z, y)$  definido como:

$$\text{➤ } \Phi_{g_i}(u, v) = |F\{g_i(x, y)\}|^2$$

$$\text{➤ } = \Phi_{f_i}(u, v)|H(u, v)|^2 + \Phi_{n_i}(u, v) \quad i = 1, 2 \dots N_p$$

$\Phi_{f_i}(u, v)$  y  $\Phi_{n_i}(u, v)$  son los espectros de potencia de  $f_i(x, y)$ , y  $n_i(x, y)$ , respectivamente, y  $F\{\cdot\}$  representa el transformador de Fourier rápido (FFT).

2. Se calcula el promedio de los espectros de potencia.

$$\text{➤ } \Phi_g(u, v) = \sum_{i=1}^{N_p} \Phi_{g_i}(u, v)$$

Esta cantidad promedio conserva el sabor de  $|H(u, v)|$ , puesto que está presente en cada una de las sub-secciones promediadas. La contribución de  $f_i(x, y)$  y  $n_i(x, y)$ , por otra parte, variará de sección a sección, resultando en una contribución mucho menor a la media. Los valores "pequeños" de  $\Phi_g(u, v)$  corresponden fuertemente a los ceros de  $|H(u, v)|$ .

3. El spectrum de potencia definido como:

$$\text{➤ } c_g(p, q) = F\{\log(\Phi_g(u, v))\}$$

Es calculado. Inmersiones periódicas en la amplitud de  $\Phi_g(u, v)$  da a lugar a grandes picos negativos en  $c_g(p, q)$ . Por ejemplo, en movimiento lineal, las inmersiones de PSF son periódicas y separadas  $1/d$ . Esto resulta en un pico negativo en  $c_g(p, q)$  a una distancia  $d$  del origen. El ángulo de la espiga denota la dirección del desenfoque de movimiento. La simetría circular de la PSF desenfocado resulta en anillos de espigas en el cepstrum, que reflejan la naturaleza aproximadamente radialmente periódica de sus ceros del dominio de la frecuencia.

- III. Utilizando la posición de los picos de la cantidad  $c_g(p, q)$ , se identifica el tipo de desenfoque (fuera de foco o desplazamiento lineal de cámara) y se calcula el valor del parámetro correspondiente utilizando una inspección visual o un procedimiento de umbral.
- IV. La imagen verdadera se deconvoluciona utilizando la PSF estimada con uno de los métodos clásicos de restauración lineal de imágenes.

La principal limitación es que se requiere conocer una forma paramétrica de la PSF. Además, para aplicaciones como rayos X o imágenes astronómicas, donde la PSF es a menudo gaussiana, no existen ceros de dominio de frecuencia de  $H(u, v)$ . Para tales situaciones, debe usarse otro método de deconvolución ciega (Kundur, et al., 1996).

#### 1.6.4 Algoritmo VCG (Validación Cruzada Generalizada)

La deconvolución ciega utilizando métodos de estimación de parámetros ARMA, propone modelar la imagen verdadera como un proceso autorregresivo bidimensional (AR) y la PSF como un proceso de media móvil bidimensional (MA). Basándose en estos modelos, la imagen borrosa resultante se representa como un proceso de media móvil autorregresiva (ARMA). La identificación de los parámetros ARMA permite identificar la verdadera imagen y PSF (Kundur, et al., 1996).

Los métodos existentes de esta clase difieren en cómo se calculan los parámetros ARMA. Se han propuesto técnicas basadas en estadísticas de segundo orden, como la estimación de máxima verosimilitud (ML), las redes neuronales y la validación cruzada generalizada (VCG), siendo esta última la que mejores resultados arroja en las simulaciones (Kundur, et al., 1996). A continuación, la descripción del algoritmo:

- I. Definiciones:
  1. Parámetros de  $\theta_i$   $\{\{h, l, m\}, \{a(l, m)\}, \alpha\}$ , alfa: imagen, desenfoque y regularización en la i-ésima iteración
  2.  $\hat{f}_k$ : "imagen restaurada", utilizando todos los píxeles en g excepto el k-ésimo píxel
  3.  $E^k(\hat{f}_k, \theta)$ : error de estimación
  4.  $V(\theta)$ : error de validación
  5.  $N_g$ : número de píxeles en g
- II. Inicialización (i = 0):
  1. Establece valores iniciales para  $\theta_i = \{\{h(l, m)\}, \{a(l, m)\}, \alpha\}$  usando información a priori sobre la PSF o imagen verdadera o inspeccionando la imagen borrosa.
  2. Fije tolerancia  $\delta > 0$ .
- III. En la iteración (i):  $i = 0, 1, 2, \dots, N_g$ 
  1. Si  $i = 0$ , pase a 2. De lo contrario, utilice el procedimiento de búsqueda desarrollado por Powell para actualizar  $\theta_i$  de  $\theta_{i-1}$
  2. En iteración (k):  $k = 0, 1, 2, \dots, N_g$

El siguiente criterio se minimiza con respecto a  $\hat{f}$ : para obtener la "imagen de restauración",  $\hat{f}_k$ . Todos los píxeles de  $g$ , pero el  $k$ th, se utilizan para el procedimiento de restauración. A y H se dan a partir del valor actual de  $\theta_i$ :

$$\blacktriangleright E^k(\hat{f}, \theta_i) = \frac{1}{N_g} \sum_{j=1}^{N_g} (g_j - \{H\hat{f}\}_i)^2 + \alpha [(I - A)\hat{f}]^2$$

Donde  $\alpha$  es el parámetro de regularización utilizado para prevenir la amplificación del ruido durante el procedimiento de restauración, y el subíndice  $j$  representa el  $j$ -ésimo elemento del vector correspondiente.

3. Utilizando  $\hat{f}_k$  para  $k = 1, 2, \dots, N_g$ , el error de validación se calcula para determinar la precisión de la estimación de parámetros  $\theta_i$ . El error de validación está dado por:

$$\blacktriangleright V(\theta_i) = \frac{1}{N_g} \sum_{k=1}^{N_g} (g_k - [H\hat{f}_k(\theta_i)]_k)^2 w_{kk}(\theta_i)$$

Donde  $w_{kk}(\theta_i)$  son pesos utilizados para facilitar el cálculo numérico.

4. Si  $\|\theta_i - \theta_{i-1}\| < \delta$ , vaya al paso IV. De lo contrario, vaya al paso 1.

- IV. Si  $\theta_i$  es una buena estimación de H, A, y  $[H\hat{f}_k(\theta_i)]_k$  entonces debe ser una buena aproximación a  $g_k$  promediada sobre todos los valores de  $k$ . Por lo tanto, el parámetro  $\theta_i$ , que produce el mínimo de  $V(\theta_i) = \frac{1}{N_g} \dots$ , es seleccionado como el parámetro de VCG.

La mayor limitación de este método es que el criterio VCG de la ecuación apropiada se vuelve insensible a los cambios en los parámetros individuales de  $\theta$  cuando el número total de parámetros es grande. Además, existe la posibilidad de que los métodos puedan converger a mínimos locales (Kundur, et al., 1996).

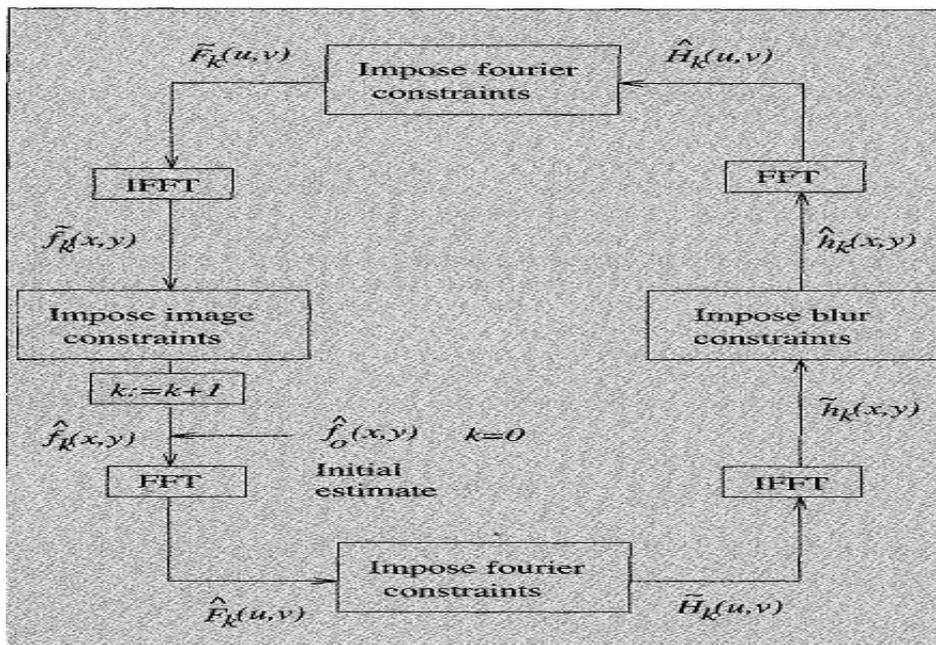
### 1.6.5 Técnicas de restauración con restricciones no paramétricas determinísticas

En contraste con los métodos anteriores, los algoritmos de esta clase no asumen modelos paramétricos para la imagen o el desenfoque. Las limitaciones determinísticas como no negatividad, soporte finito conocido, y la existencia de bordes invariantes de desenfoque se asumen para la imagen verdadera. Una serie de técnicas de deconvolución ciega para las imágenes caen en esta clase, que incluyen el algoritmo iterativo de deconvolución ciega (IBD), el algoritmo de recocido simulado de McCallum (SA), el algoritmo de filtrado inverso recursivo de restricciones y no restrictivas de soporte (NAS-RIF), y el algoritmo de superresolución ciega, entre otros. Los métodos son iterativos y estiman simultáneamente los píxeles de la imagen verdadera y la PSF (o su inversa). Las limitaciones en la imagen verdadera y PSF se incorporan en un criterio de optimalidad que se minimiza utilizando técnicas numéricas (Kundur, et al., 1996).

## Algoritmo IBD

El método iterativo de deconvolución ciega (IBD) es el método más popular en esta clase. Además de los supuestos anteriormente expuestos, el método requiere que la PSF sea no negativa con soporte finito conocido (Kundur, et al., 1996).

El método general hace uso del algoritmo de transformada rápida de Fourier (FFT). La estructura básica del algoritmo se presenta en la figura 2. La estimación de la imagen se denota por  $f(x, y)$ , la estimación de PSF por  $h(x, y)$ , y la imagen linealmente degradada por  $g(x, y)$ . Las mayúsculas representan las versiones transformadas rápidas de Fourier de las señales correspondientes. Los subíndices indican el número de iteración del algoritmo (Kundur, et al., 1996).



**Figura 2.** Método iterativo de la deconvolución ciega.

Después de una suposición inicial aleatoria para la imagen verdadera, el algoritmo alterna entre los dominios de imagen y Fourier, imponiendo restricciones conocidas en cada uno. Las restricciones se basan en la información disponible sobre la imagen y PSF. Las restricciones de dominio de imagen se pueden imponer reemplazando píxeles de valor negativo dentro de la región de soporte con píxeles cero y no cero fuera de la región de soporte con el valor de píxel de fondo. La restricción de dominio de Fourier implica la estimación de la PSF (imagen) utilizando la FFT de la imagen degradada y la imagen (PSF) estimación. Es decir, en la  $k$ -ésima iteración,

$$\triangleright \tilde{H}_k(u, v) = \frac{G(u, v) \hat{F}_{k-1}^*(u, v)}{|\hat{F}_{k-1}(u, v)|^2 + \alpha / |\tilde{H}_{k-1}(u, v)|^2}$$

$$\triangleright \tilde{F}_k(u, v) = \frac{G(u, v) \hat{H}_{k-1}^*(u, v)}{|\hat{H}_{k-1}(u, v)|^2 + \alpha / |\tilde{F}_{k-1}(u, v)|^2}$$

Donde  $(\cdot)^*$  denota el conjugado complejo de  $(\cdot)$ . La constante real  $a$  representa la energía del ruido aditivo y está determinada por el conocimiento previo del nivel de contaminación acústica, si está disponible. El valor de  $a$  debe elegirse cuidadosamente para una restauración confiable. El algoritmo se ejecuta para un número determinado de iteraciones, o hasta que las estimaciones comienzan a converger. El método IBD es popular por su baja complejidad computacional. Se han sugerido muchas implementaciones diferentes de este algoritmo básico. Se diferencian en sus suposiciones sobre la imagen verdadera y PSF, y cómo estas suposiciones se imponen en la imagen y los dominios de Fourier. Otra ventaja de esta técnica es su robustez al ruido a causa de los filtros tipo Wiener. La robustez al ruido se refiere a la capacidad del algoritmo para suprimir la amplificación de ruido que resulta de la naturaleza mal planteada del problema de restauración (Kundur, et al., 1996).

El principal inconveniente del método IBD es su falta de fiabilidad. La unicidad y las propiedades de convergencia son, por el momento, inciertas. Además, la restauración es sensible a la estimación inicial de la imagen, y el algoritmo puede mostrar inestabilidad (Kundur, et al., 1996).

### **Algoritmo SA**

El algoritmo de recocido simulado (SA) de McCallum hace las mismas suposiciones sobre la PSF que el método IBD. Esto implica la minimización de la siguiente función de coste multimodal:

$$\triangleright J(\hat{f}(x, y), \hat{h}(x, y)) = \sum_{v(x, y)} [\hat{f}(x, y) * \hat{h}(x, y) - g(x, y)]^2$$

Se supone que la imagen y la PSF son positivas con soporte finito conocido. Utilizando estas restricciones para  $\hat{f}(x, y)$  y  $\hat{h}(x, y)$ , se emplea un procedimiento de recocido simulado para la minimización de  $J$  con respecto a  $\{\hat{f}(x, y)\}$  y  $\{\hat{h}(x, y)\}$ . En el recocido simulado, las estimaciones de los parámetros de la función de coste son iterativamente variados para minimizar globalmente. Los valores de los parámetros son perturbados aleatoriamente. Si la perturbación reduce  $J$ , entonces se acepta; si aumenta  $J$ , entonces se acepta con probabilidad  $p = \exp(-\Delta J / T_k)$ , donde  $\Delta J$  es el cambio en la función de coste debido a la perturbación, y  $T_k$  es el parámetro de temperatura. A medida que progresan las iteraciones, se reduce el parámetro de temperatura  $T_k$  (es decir, se reduce  $p$ ) (Kundur, et al., 1996).

La principal desventaja de este algoritmo es que la convergencia al mínimo global de la función de coste es lenta. La velocidad de convergencia del algoritmo depende en gran medida de la rapidez con que se reduce  $T_k$ . Para imágenes de tamaño realista, el algoritmo es demasiado lento para producir una buena solución (Kundur, et al., 1996).

### **1.6.6 Dificultades encontradas**

Después de un análisis de las soluciones existentes se encontraron las siguientes dificultades:

- No utilizan el método de regularización total, del cual fueron explicados en la sección 1.5.2 las potencialidades que brinda la regularización en la restauración de imágenes digitales.
- No tienen tolerancia a fallos ya que no son capaces de corregir una estimación incorrecta.
- Tienen lenta velocidad de convergencia, lo que se traduce en altos tiempos de ejecución.

Como alternativa a estas dificultades se propone utilizar una red neuronal la cual utiliza en su implementación el método de regularización total.

## 1.7 Red neuronal

Existen numerosas formas de definir a las redes neuronales; desde las definiciones cortas y genéricas hasta las que intentan explicar más detalladamente qué son las redes neuronales. Por ejemplo (Matich, 2001):

1. Una nueva forma de computación, inspirada en modelos biológicos.
2. Un modelo matemático compuesto por un gran número de elementos procesales organizados en niveles.
3. Un sistema de computación compuesto por un gran número de elementos simples, elementos de procesos muy interconectados, los cuales procesan información por medio de su estado dinámico como respuesta a entradas externas.
4. Las redes neuronales artificiales son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico.

### 1.7.1 Ventajas que ofrecen las redes neuronales

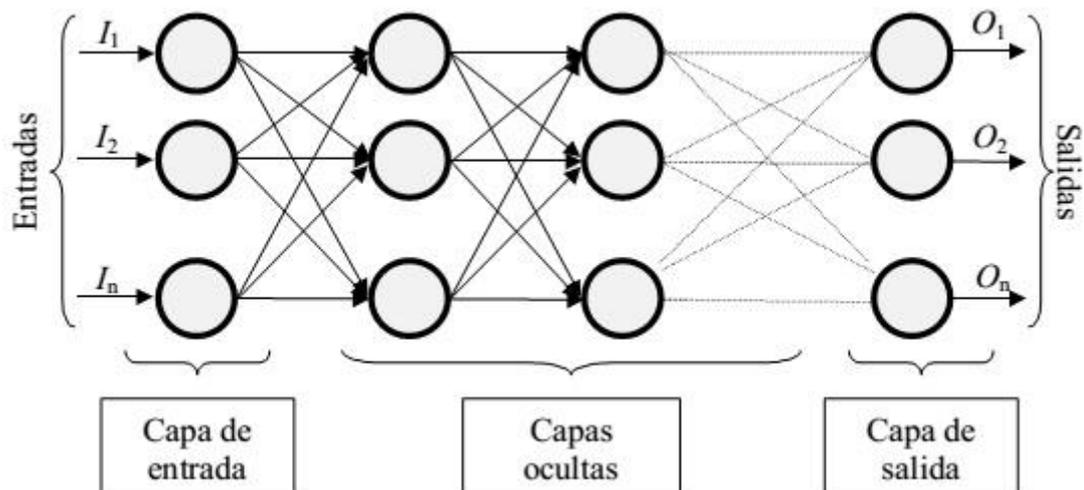
Debido a su constitución y a sus fundamentos, las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro. Por ejemplo, son capaces de aprender de la experiencia, de generalizar de casos anteriores a nuevos casos, de abstraer características esenciales a partir de entradas que representan información irrelevante, etc. Esto hace que ofrezcan numerosas ventajas y que este tipo de tecnología se esté aplicando en múltiples áreas. Entre las ventajas se incluyen (Matich, 2001):

1. **Aprendizaje adaptativo.** Capacidad de aprender a realizar tareas basadas en un entrenamiento o en una experiencia inicial.
2. **Auto-organización.** Una red neuronal puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje.
3. **Tolerancia a fallos.** La destrucción parcial de una red conduce a una degradación de su estructura; sin embargo, algunas capacidades de la red se pueden retener, incluso sufriendo un gran daño.

4. **Operación en tiempo real.** Los cálculos neuronales pueden ser realizados en paralelo; para esto se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.
5. **Fácil inserción dentro de la tecnología existente.** Se pueden obtener chips especializados para redes neuronales que mejoran su capacidad en ciertas tareas. Ello facilitará la integración modular en los sistemas existentes.

### 1.7.2 Elementos básicos que componen una red neuronal

A continuación, se muestra un esquema de una red neuronal:



**Figura 3.** Ejemplo de una red neuronal totalmente conectada.

La misma está constituida por neuronas interconectadas y dispuestas en tres capas. Los datos ingresan por medio de la “capa de entrada”, pasan a través de la “capa oculta” y salen por la “capa de salida”. Cabe mencionar que la capa oculta puede estar constituida por varias capas. (Matich, 2001) Las neuronas de la capa de entrada actúan de forma diferente a las demás neuronas, estas se encargan de recibir la información o patrones de entrada y transmiten o propagan dicha información a todas las neuronas de la siguiente capa. La capa final o capa de salida de la red proporciona al entorno exterior la respuesta de la red neuronal para cada uno de los patrones de entrada. En las capas ocultas se lleva a cabo un procesamiento no lineal de los patrones que reciben.

### 1.7.3 Topología de las redes neuronales

La topología o arquitectura de una red neuronal consiste en la organización y disposición de las neuronas en la misma, formando capas o agrupaciones de neuronas más o menos alejadas de la entrada y salida de dicha red. En este sentido, los parámetros fundamentales de la red son: el número de capas, el número de neuronas por capa, el grado de conectividad y el tipo de conexiones entre neuronas (Matich, 2001).

- **Redes monocapa:** En las redes monocapa, se establecen conexiones entre las neuronas que pertenecen a la única capa que constituye la red. Las redes monocapa se

utilizan generalmente en tareas relacionadas con lo que se conoce como auto asociación (regenerar información de entrada que se presenta a la red de forma incompleta o distorsionada).

- **Redes multicapa:** Las redes multicapas son aquellas que disponen de un conjunto de neuronas agrupadas en varios (2, 3, etc.) niveles o capas. En estos casos, una forma para distinguir la capa a la que pertenece una neurona, consistiría en fijarse en el origen de las señales que recibe a la entrada y el destino de la señal de salida. Normalmente, todas las neuronas de una capa reciben señales de entrada desde otra capa anterior (la cual está más cerca a la entrada de la red), y envían señales de salida a una capa posterior (que está más cerca a la salida de la red). A estas conexiones se las denomina conexiones hacia adelante o *feedforward*. Sin embargo, en un gran número de estas redes también existe la posibilidad de conectar la salida de las neuronas de capas posteriores a la entrada de capas anteriores; a estas conexiones se las denomina conexiones hacia atrás o *feedback*. Estas dos posibilidades permiten distinguir entre dos tipos de redes con múltiples capas: las redes con conexiones hacia adelante o redes *feedforward*, y las redes que disponen de conexiones tanto hacia adelante como hacia atrás o redes *feedforward/feedback*.

#### 1.7.3.1 Algoritmo de retropropagación del error

El algoritmo de retropropagación del error, o por su nombre en inglés *Backpropagation* es el método de entrenamiento más utilizado en redes *feedforward*. Es un método de aprendizaje supervisado de gradiente descendente, en el que se distinguen claramente dos fases: primero se aplica un patrón de entrada, el cual se propaga por las distintas capas que componen la red hasta producir la salida de la misma. Esta salida se compara con la salida deseada y se calcula el error cometido por cada neurona de salida. Estos errores se transmiten hacia atrás, partiendo de la capa de salida, hacia todas las neuronas de las capas intermedias. Cada neurona recibe un error que es proporcional a su contribución sobre el error total de la red. Basándose en el error recibido, se ajustan los errores de los pesos sinápticos de cada neurona (Bertona, 2005).

### 1.8 Metodología, herramientas y técnicas

A fin de garantizar a lo largo de todo el proceso de desarrollo, que el algoritmo cumpla con los objetivos que fueron planteados a desarrollar, se utilizaron un conjunto de técnicas y herramientas, la cuales se describen a continuación.

#### 1.8.1 Metodología de desarrollo

Una metodología es una recopilación de técnicas, procedimientos, herramientas y documentos auxiliares que sirve de ayuda a los desarrolladores de software en la realización de nuevas aplicaciones. En un proyecto de desarrollo, define quién debe hacer qué, cuándo y cómo hacerlo. Es un proceso, puede seguir uno o varios modelos de ciclo de vida e indica cómo hay que obtener

los distintos productos parciales y finales en el desarrollo de un software, por lo que se procede a analizar los diferentes modelos de metodologías existentes para proyectos (Vique, 2012).

Para seleccionar el enfoque, metodología y prácticas más adecuadas a utilizar en el proceso de desarrollo de software, se aplicó el método Boehm y Turner.

### **Método Boehm y Turner**

Caracterizar el proyecto de software a partir de 5 criterios y estima cuan ágil o prescriptivo debería ser el enfoque a utilizar, estos son: tamaño del equipo, criticidad del producto, dinamismo de los cambios, cultura del equipo y personal con que se cuenta. Cada uno de esos criterios tiene elementos que lo discriminan y por tanto se tienen en cuenta a la hora de seleccionar uno u otro enfoque (Boeras Vázquez, et al., 2012).

El enfoque prescriptivo, denominado en algunas bibliografías como tradicional o pesado, busca la estructura, orden y consistencia del proyecto de desarrollo de software en cuestión. Se les llama prescriptivos porque prescriben un conjunto de elementos del proceso (acciones, tareas, productos de trabajo, mecanismos de control y aseguramiento de la calidad). Además, definen la forma en que los elementos del proceso mencionados anteriormente deben relacionarse entre sí.

El enfoque ágil, llamado también como enfoque ligero se centra en los miembros del equipo y su interacción, en la entrega rápida de versiones de software funcional, en la colaboración constante del cliente y la facilidad para manejar los cambios, dándole menor importancia a las herramientas, documentación, la formalidad y planificación exhaustiva del proceso.

Para la selección del valor que se ubicará en cada eje (uno para cada criterio) de la estrella se debe tener en cuenta el comportamiento de estos criterios en el proyecto. En lo sucesivo se describe cada uno:

**Tamaño:** Este criterio se utiliza para representar el número de personas involucradas en el proyecto. Pueden tenerse en cuenta el nivel de complejidad que pueda presentarse en la comunicación entre los miembros del proyecto y los costos que pueden provocar cambios esperados.

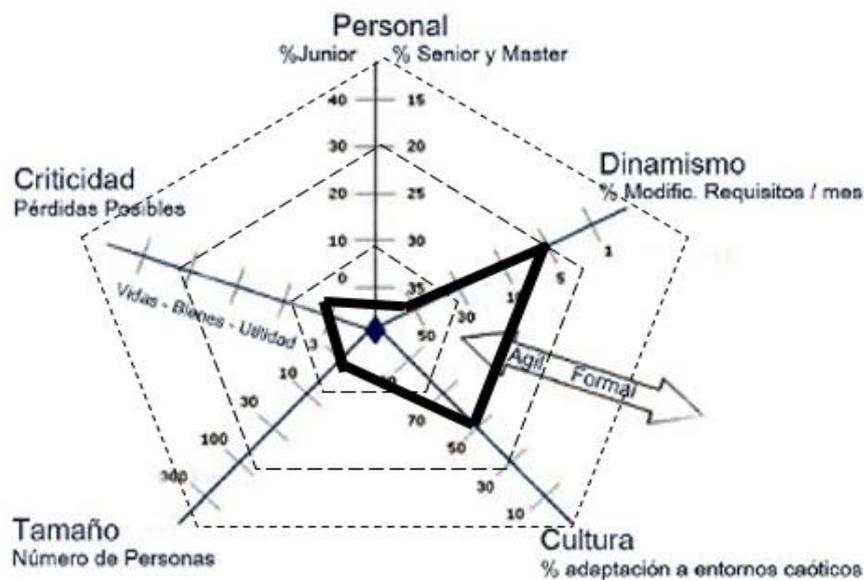
**Criticidad:** Se utiliza para evaluar la naturaleza del daño ocasionado por defectos que no hayan sido detectados al producto. Su evaluación puede ser cualitativa.

**Dinamismo:** Representa la rapidez con la que pueden estar cambiando los requerimientos del proyecto.

**Personal:** Representa la proporción del personal con experiencia alta, media y baja. Los métodos orientados al plan no se ven afectados negativamente por este factor pues no interesa el nivel de experiencia con la que cuenten los miembros del equipo.

**Cultura:** Las organizaciones y las personas que relaciona el proyecto pueden depender de la confianza o de la relación contractual. Esto refleja el nivel de ceremonia necesario y aceptado: documentación, control, formalismo en las comunicaciones.

La figura muestra una representación de la estrella de Boehm y Turner para el proyecto de desarrollo de software:



**Figura 4.** Estrella que representa al proyecto según el método de Boehm y Turner.

Luego de aplicar este método, se decide escoger un enfoque ágil para el desarrollo de la investigación, pues la estrella resultante está dentro de las líneas discontinuas que representan este enfoque.

### Metodología a utilizar

Como guía para lograr un correcto proceso de desarrollo de la propuesta de solución se decide emplear la metodología Programación Extrema (XP, *eXtreme Programming*), de enfoque ágil, cuyo objetivo es conseguir la máxima satisfacción del cliente en forma rápida y eficiente ante los cambios de requisitos.

#### 1.8.2 Herramienta de modelado

Visual Paradigm es una herramienta de diseño para proyectos software. Soporta multitud de estándares de modelado como UML, SysML, ERD, DFD, BPMN. Es una herramienta que facilita y agiliza el desarrollo de los proyectos basados en experiencias de usuario apoyándose en la identificación de los casos, requisitos o flujos de acontecimientos (Visual Paradigm Inc, 2014).

#### 1.8.3 Entorno de desarrollo integrado (IDE)

Un IDE es un editor y un compilador, junto con muchas características adicionales destinadas a brindar comodidad y velocidad al programador (Novara, 2010).

## **Matlab R2014b**

Matlab es una herramienta matemática que brinda la posibilidad de un entorno de desarrollo integrado o entorno de escritorio afinado para la exploración, el diseño y la solución de problemas de manera iterativa, visualizar datos gráficamente y crear propios, además de aplicaciones para ajustar curvas, clasificar datos, analizar señales, y muchas otras tareas relacionadas con dominios concretos (MathWorks Inc, 2017).

### **Justificación a la selección de Matlab R2014b**

Se ha decidido utilizar esta herramienta, pues con adecuadas optimizaciones es rápido en ejecución y garantiza alta precisión. Además, posee un amplio soporte matemático, teniendo la posibilidad de uso de precisión extendida en cálculos y está disponible en diferentes plataformas como Windows y Unix.

### **1.9 Conclusiones del capítulo**

En este capítulo fue realizado un estudio de la restauración de imágenes digitales, así como de diferentes algoritmos utilizados para estimar la imagen original a partir de una versión degradada de la misma. Después de un estudio de algunos de estos sistemas de restauración se expusieron algunas deficiencias encontradas y se propone una red neuronal que implemente el regularizador por variación total. Se seleccionó Matlab como IDE de desarrollo y Visual Paradigm como herramienta para el modelado. Se definieron las herramientas y tecnologías para dar solución al problema planteado, teniendo en cuenta las características de estas. Además, se explicó el método de Boehm y Turner, escogiéndose XP como la metodología de desarrollo a utilizar.

## Capítulo 2: Análisis y diseño de la solución

Este capítulo tiene como objetivo describir las actividades desarrolladas durante todo el proceso de análisis y diseño. Se detalla la propuesta de solución y se describe su arquitectura. Los requisitos funcionales y no funcionales, conjuntamente con los artefactos que plantea la metodología escogida que contribuyeron al desarrollo de las nuevas funcionalidades agregadas.

### 2.1 Propuesta de solución

Para establecer la propuesta de solución se seleccionó una red neuronal artificial (RNA) con el objetivo de restaurar detalles en imágenes oblicuas provenientes de la lámpara de hendidura. Esta describe un algoritmo en el cual la capa de entrada de la red está compuesta por  $L$  neuronas cuyas entradas  $Y_1, Y_2, \dots, Y_L$  son los  $L$  píxeles de la imagen de entrada. Estas neuronas se conectan a tres capas ocultas en las que se calcula la función de activación.

En la  $m$ -ésima iteración de la red la capa de salida estará compuesta por  $L$  neuronas cuyas salidas  $X^m_1, X^m_2, \dots, X^m_L$  son respectivamente los  $L$  píxeles de una aproximación  $X^m$  a la imagen restaurada, siguiendo la lógica del algoritmo *Backpropagation*. Después de  $m$  iteraciones, la red neuronal produce la imagen restaurada final tal que  $X=X^m$ . Por otro lado, la RNA consta de tres capas ocultas, donde la primera capa oculta tiene un total de 10 neuronas, la segunda capa tiene 5 neuronas y la tercera tiene 2 neuronas.

El proceso de funcionamiento de la red consiste en dos procesos repetidos por cada iteración  $m$ :

- *Forward*: la entrada  $Y$  se procesa hacia la derecha de la red hasta conseguir el cálculo de la función de activación.
- *Backward*: con la salida de la función de activación anterior, se aplica el regularizador por variación total, se calcula el vector gradiente local y la función de coste obteniendo como resultado una imagen estimada en cada iteración.

Por tanto, la RNA describe un algoritmo iterativo que simula por cada iteración tanto una aproximación al proceso de degradación (*backward*) como a la solución de restauración (*forward*), de tal forma que los resultados se van adaptando progresivamente a su valor óptimo según la función de coste. Conviene destacar que la entrada a la RNA para cualquier iteración es siempre la imagen degradada  $Y$ , sin necesitar ningún tipo de entrenamiento previo de la red.

Con este algoritmo se persigue que los especialistas tengan una visión con más calidad de las imágenes de los pacientes operados de cataratas. Se seleccionó una secuencia de pasos que responden al funcionamiento del algoritmo, estos son mostrados a continuación en un diagrama de actividades.

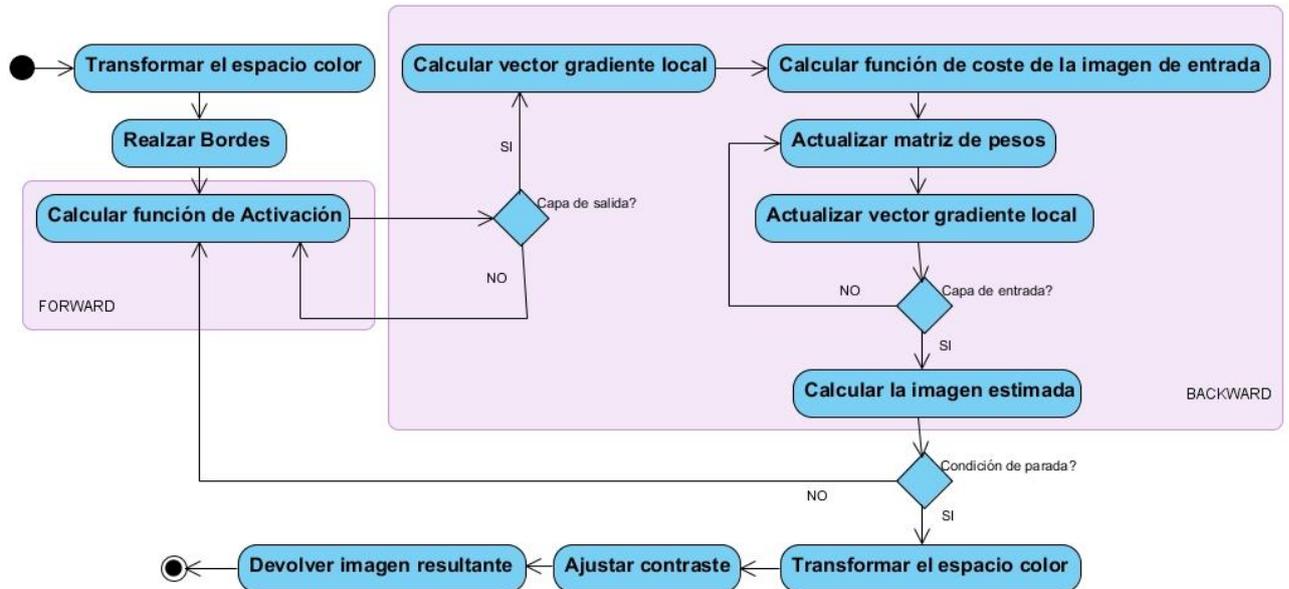


Figura 5. Esquema de la propuesta de solución.

### Paso 1. Transformar el espacio color

En el algoritmo implementado se trabaja con las intensidades de las imágenes y para esto se utiliza el espacio de color *ybcrc*, se define el color en términos de un componente de luminancia y dos de crominancia, luego se transforma la imagen del espacio de color *rgb* a *ybcrc*. El espacio de color *rgb* representa una composición del color en términos de los colores primarios con los que se forma (rojo, verde y azul) y está basado en la síntesis aditiva, es decir, la mezcla por adición de los tres colores luz primarios.

### Paso 2. Realzar bordes

Como primer valor de la imagen estimada  $\hat{x}$  se toma la imagen degradada con un realce de bordes, para lo que se utiliza la función de Matlab “*imsharpen*”.

### Paso 3. Calcular función de activación

El cálculo de la función de activación se va a realizar de la siguiente manera:

$$\varphi\{v\} = \frac{1}{1+e^{-v}} \quad (10)$$

Donde  $v$  es la multiplicación de la matriz de pesos sinápticos por los píxeles de la imagen borrosa de entrada, quedando de la siguiente forma:

$$v = W * Y_L \quad (11)$$

### Paso 4. Calcular vector gradiente local

El vector gradiente local se va a calcular de la siguiente manera:

$$\delta^{(m)} = \varphi'\{v^{(m)}\} \cdot (-H_a^T e^{(m)} + \lambda D_a^T \Sigma_\beta(\hat{x}^{(m)})r^{(m)}) \quad (12)$$

Donde  $H_a^T$  y  $D_a^T$  son matrices de convolución con filtro de paso alto definidas previamente,  $\Sigma_\beta(x^m)$  matriz diagonal de la imagen estimada, los valores de esta matriz a su vez son matrices formadas por los píxeles de la imagen de entrada.  $\lambda$  es el regularizador de variación total,  $e^m$  y  $r^m$  son términos de error y regularización en la capa de salida después de  $m$  iteraciones y se van a calcular de la siguiente manera:

$$e^{(m)} = y_{tru} - trunc\{H_a \hat{x}^{(m)}\} \quad (13)$$

$$r^{(m)} = trunc\{D_a \hat{x}^{(m)}\} \quad (14)$$

Donde  $y_{tru}$  es la convolución truncada de la imagen estimada del paso anterior con  $H_a$  y el otro término es la convolución truncada de la imagen estimada en ese paso con  $H_a$

### Paso 5. Calcular función de coste de la imagen de entrada

La función de coste de la imagen de entrada se calcula de la siguiente manera:

$$C(\hat{x}^{(m)}) = \frac{1}{2} \|e^{(m)}\|_2^2 + \lambda Q_{TV}(\hat{x}^{(m)}) \quad (15)$$

Donde  $Q_{TV}$  es un maximizador cuadrático de la función de coste y se va a calcular de la siguiente manera:

$$Q_{TV}(x) = x^T D^T \Sigma_\beta(x) D_x + K \quad (16)$$

Donde  $K$  es una constante insignificante.

### Paso 6. Actualizar matriz de pesos

La matriz de pesos se va a actualizar de la siguiente manera:

$$W_i^{(m+1)} = W_i^{(m)} - \eta \delta_i^{(m)} (q_{i-1}^{(m)}) \quad (17)$$

Donde  $\eta$  es el factor de aprendizaje el cual va a tomar valor 1 o 2 dependiendo del tamaño de la imagen. El término  $q_{i-1}^{(m)}$  es igualado a la función de activación, por tanto, tiene el mismo valor.

### Paso 7. Actualizar vector gradiente local

La actualización del vector gradiente local se va a realizar de la siguiente manera:

$$\delta_i^{(m)} = \varphi'\{v_i^{(m)}\} \cdot ((W_{i+1}^{(m)})^T \delta_{i+1}^{(m)}) \quad (18)$$

Donde el resultado es la multiplicación punto a punto entre la función de activación y la traspuesta de la matriz de pesos multiplicada con el vector gradiente local anterior.

### Paso 8. Calcular la imagen estimada

La imagen estimada es el resultado del promedio de las estimaciones de cada una de las imágenes de las capas ocultas más la función de coste.

### **Paso 9. Transformar de espacio color**

Al haber utilizado para la estimación la componente lumínica del espacio de color *ycbcr* se vuelve a unificar el mismo con sus componentes cromáticas *cb* y *cr*, recuperándose posteriormente la imagen en formato a color *rgb*.

### **Paso 10. Ajustar contraste**

Para una mejor visualización del resultado se aplica una expansión de histograma donde primeramente se obtienen los límites del mismo, mediante la función “*stretchlim*”, para después expandir el histograma con la función “*imadjust*”.

### **Paso 11. Devolver imagen resultante**

La imagen obtenida después de aplicar el proceso de restauración es mostrada.

## **2.2 Funcionalidades**

Las funcionalidades del sistema son las capacidades o condiciones que el sistema debe cumplir (Pressman, 2007). A continuación, se muestran las funcionalidades:

No.	Funcionalidades
<b>F1</b>	<b>Cargar imagen.</b>
<b>F2</b>	<b>Estimar imagen emborronada.</b>
<b>F2.1</b>	Realizar transformación de espacio color.
<b>F2.2</b>	Realizar iteraciones por la red neuronal artificial.
<b>F2.3</b>	Calcular el vector gradiente local.
<b>F2.4</b>	Calcular imagen estimada.
<b>F2.5</b>	Realizar ajuste de contraste.
<b>F3</b>	<b>Mostrar imagen restaurada.</b>

**Tabla 1.** Funcionalidades del sistema.

## **2.3 Lista de reserva del producto**

Las listas de reserva del producto en una aplicación son muy importantes ya que son las cualidades que debe poseer para un correcto funcionamiento. A continuación, se definen los siguientes requisitos:

- **Funcionamiento:**
- **Hardware:**

2 GB de RAM, Microprocesador Core i3 de tercera generación, espacio disponible en disco duro de 5 GB.

- **Software:**

Sistema Operativo Windows 8.1 o superior, Matlab en su versión 14.

## 2.4 Historia de usuario

Uno de los artefactos generados por la metodología XP son las historias de usuarios (HU), utilizadas como herramientas para dar a conocer los requerimientos del sistema al equipo de desarrollo. Son pequeños textos para describir una actividad que realizará el software. Se puede considerar que estas juegan un papel similar a los casos de uso en otras metodologías, pero en realidad son muy diferentes porque solo muestran la silueta de una tarea a realizarse (Pressman, 2007).

Las HU se clasifican según:

- **La prioridad en el negocio:**

- ✓ Alta: Se le otorga a las HU que resultan funcionalidades fundamentales en el desarrollo del sistema, a las que el cliente define como principales para el control integral del sistema.
- ✓ Media: Se le otorga a las HU que resultan para el cliente como funcionalidades a tener en cuenta, sin que estas tengan una afectación sobre el sistema que se esté desarrollando.
- ✓ Baja: Se le otorga a las HU que constituyen funcionalidades que sirven de ayuda al control de elementos asociados al equipo de desarrollo, a la estructura y no tienen nada que ver con el sistema en desarrollo.

- **El riesgo en su desarrollo:**

- ✓ Alta: Cuando en la implementación de las HU se consideran la posible existencia de errores que conlleven a la inoperatividad del código.
- ✓ Media: Cuando pueden aparecer errores en la implementación de la HU que puedan retrasar la entrega de la versión.
- ✓ Baja: Cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto.

Las HU son representadas mediante tablas divididas por las siguientes secciones:

- ✓ Número: esta sección representa el número, incremental en el tiempo, de la historia de usuario que se describe.
- ✓ Nombre de Historia de Usuario: identifica la HU que se describe entre los desarrolladores y el cliente.
- ✓ Modificación de Historia de Usuario Número: sección que representa si la HU se le realizó alguna modificación con respecto al estado anterior.
- ✓ Usuario: Los programadores responsables de la historia de usuario.
- ✓ Iteración asignada: número de la iteración donde va a desarrollarse la HU.

- ✓ **Prioridad en negocio:** se le otorga una prioridad (Alta, Media, Baja) a las HU de acuerdo a la necesidad de desarrollo.
- ✓ **Riesgo en Desarrollo:** se le otorga una medida de (Alto, Medio, Bajo), a la ocurrencia de errores en el proceso de desarrollo de la HU.
- ✓ **Puntos Estimados:** es el tiempo estimado en semanas que se demorará el desarrollo de la HU.
- ✓ **Puntos Reales:** representa el tiempo que se demoró en realidad el desarrollo de la HU.
- ✓ **Descripción:** breve descripción de la HU.
- ✓ **Observaciones:** señalamiento o advertencia del sistema.

Las HU también son utilizadas para estimar el tiempo que el equipo de desarrollo tomará para realizar las entregas. En una entrega se puede desarrollar una o más HU, esto depende solo del tiempo que demore la implementación de cada una de las mismas (Pressman, 2007).

Historia de Usuario	
<b>Número:</b> HU 5	<b>Nombre Historia de Usuario:</b> Calcular función de activación.
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Medio	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo el cálculo de la función de activación, este es el principal paso del proceso <i>forward</i> con el cual su valor va a ser utilizado para la actualización de los demás parámetros del algoritmo.	
<b>Observaciones:</b>	

**Tabla 2.** Historia de usuario HU 5.

## 2.5 Fase de planificación

En esta fase el cliente establece la prioridad de cada HU, y teniendo en cuenta esto los programadores realizan una estimación de esfuerzo necesario para cada una de ellas. Se realizan acuerdos sobre el material a entregar en la primera iteración y en correspondencia se genera un cronograma junto al cliente (Pressman, 2007).

### 2.5.1 Estimación de esfuerzo

La medida utilizada para la estimación del esfuerzo asociado a la implementación es el punto. Un punto equivale a una semana ideal de programación, y esta semana equivale a 5 días laborables. Esta medida puede tomar valores de 1 a 4 puntos.

No.	Historia de usuario	Punto de estimación
1.	Cargar imagen.	1
2.	Estimar imagen emborronada.	17
3.	Mostrar imagen restaurada.	1

**Tabla 3.** Plan de esfuerzo por HU.

### 2.5.2 Plan de iteraciones

Para cada entrega fueron escogidas las HU, teniendo en cuenta el orden definido. Este plan define las historias de usuario que deben ser implementadas en cada iteración y las fechas de liberación. La funcionalidad 2 con su subconjunto de funcionalidades se desglosa en las historias de usuario de la 2 a la 9. A continuación, se muestran dos iteraciones y el número de historias por cada una.

Iteración	Historias de usuario	Duración total (semanas)
1	Cargar imagen.	8
	Convertir imagen de rgb a ybcr.	
	Convertir imagen de ybcr a rgb.	
	Calcular pesos sinápticos de la RNA.	
	Calcular función de activación.	
2	Calcular regularizador de la variación total.	11
	Calcular vector gradiente local.	
	Realizar expansión del histograma.	
	Calcular imagen estimada.	
	Mostrar imagen restaurada.	

**Tabla 4.** Plan de iteraciones.

### 2.5.3 Plan de entrega

En el momento en que culmina la elaboración de las HU, se inicia el proceso de creación de un plan de entrega. El cual tiene como objetivo fundamental la obtención por parte de los programadores de una estimación detallada del período de tiempo que deben tener en cuenta para la implementación.

Artefacto	Iteración	Entrega
HU1	1	8 de marzo
HU2		

HU3		
HU4		
HU5		
HU6	2	12 de junio
HU7		
HU8		
HU9		
HU10		

**Tabla 5.** Plan de entrega de las iteraciones.

## 2.6 Fase de diseño

En este epígrafe se hace alusión a los principales elementos que componen el diseño del sistema propuesto, donde siguiendo los planteamientos de la metodología XP este debe ser un diseño simple y de fácil interacción con el usuario, basándose fundamentalmente en el desarrollo de las tarjetas Clases, Responsabilidad y Colaboración (CRC). También en este epígrafe se detallan los patrones de diseño utilizados en la solución del algoritmo.

### 2.6.1 Tarjetas de Clases-Responsabilidad-Colaboración

Las tarjetas de Clase-Responsabilidad-Colaboración (CRC) permiten ver las clases no como un depósito de datos, sino que permiten conocer el comportamiento de cada una en un alto nivel. La metodología XP estipula su uso como un artefacto obligatorio durante el desarrollo de un proyecto, debido a los beneficios que aportan a los desarrolladores. A continuación, se muestra la tarjeta CRC de la clase RNAParaBID (Martínez, 2011).

Nombre de la clase: RNAParaBID	
Responsabilidades	Clases Relacionadas
Realizar la transformación espacio color.	-rgb2ycbr.
Realizar forward.	-imsharpen, floor.
Realizar backward.	-fspecial, imfilter, serieD, median, size.
Realizar pos procesado.	-adapthisteq, imadjust.

**Tabla 6.** Tarjeta CRC de la clase RNAParaBID.

### **2.6.2 Estándares de codificación**

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiese escrito todo el código de una sola vez. Al comenzar un proyecto de software, se establece un estándar de codificación para asegurarse de que se trabaje de forma coordinada. Cuando el proyecto de software incorpore código fuente previo, o bien cuando realice el mantenimiento de un sistema de software creado anteriormente, el estándar de codificación debería establecer cómo operar con la base de código existente (Pressman, 2007).

- Utiliza los comentarios para explicar qué hace el código.
- Usa nombres significativos para las clases, métodos, propiedades y variables.
- Los métodos de clases, se define que la primera letra debe ser minúscula, si son varias palabras se debe de intercalar entre minúsculas y mayúsculas.
- Se define una declaración por línea.
- Se evita asignar el mismo valor a varias variables en la misma sentencia, pues dificulta su lectura.

### **2.7 Conclusiones del capítulo**

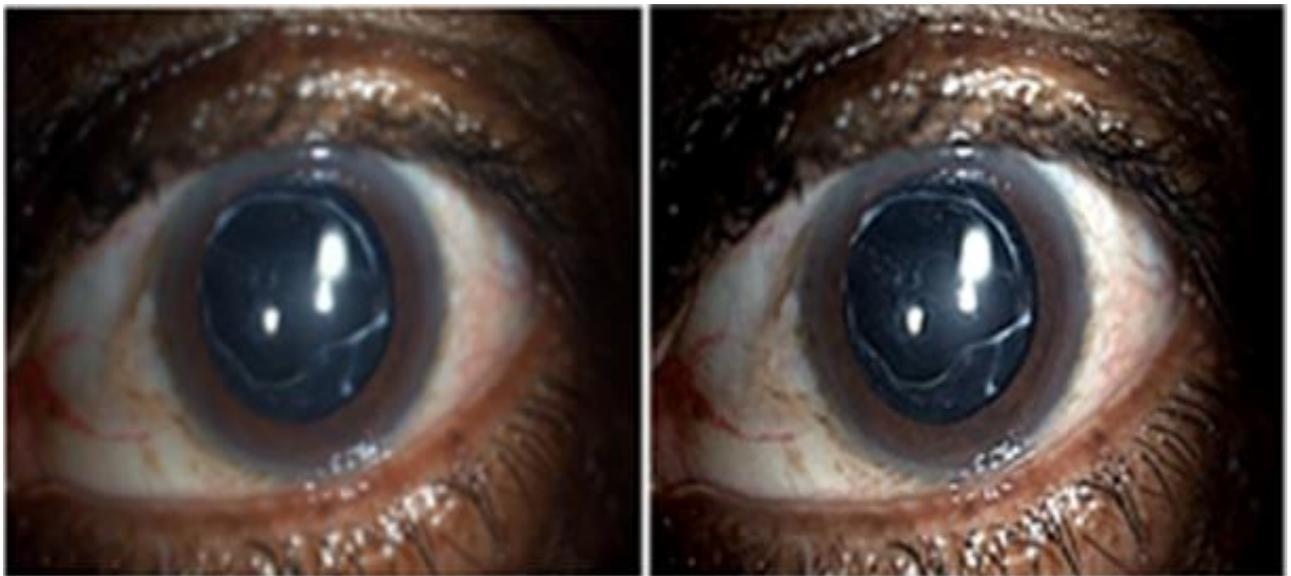
En este capítulo se abordaron los aspectos fundamentales del análisis y diseño de la propuesta de solución. El levantamiento de los requerimientos del sistema permitió determinar las funcionalidades básicas a desarrollar durante el proceso. Se definieron 10 HU para implementarse en 2 iteraciones. Se identificaron las tarjetas CRC necesarias para la implementación de las funcionalidades, las cuales serán desarrolladas haciendo uso de los patrones de diseño.

## Capítulo 3: Resultados y Validación

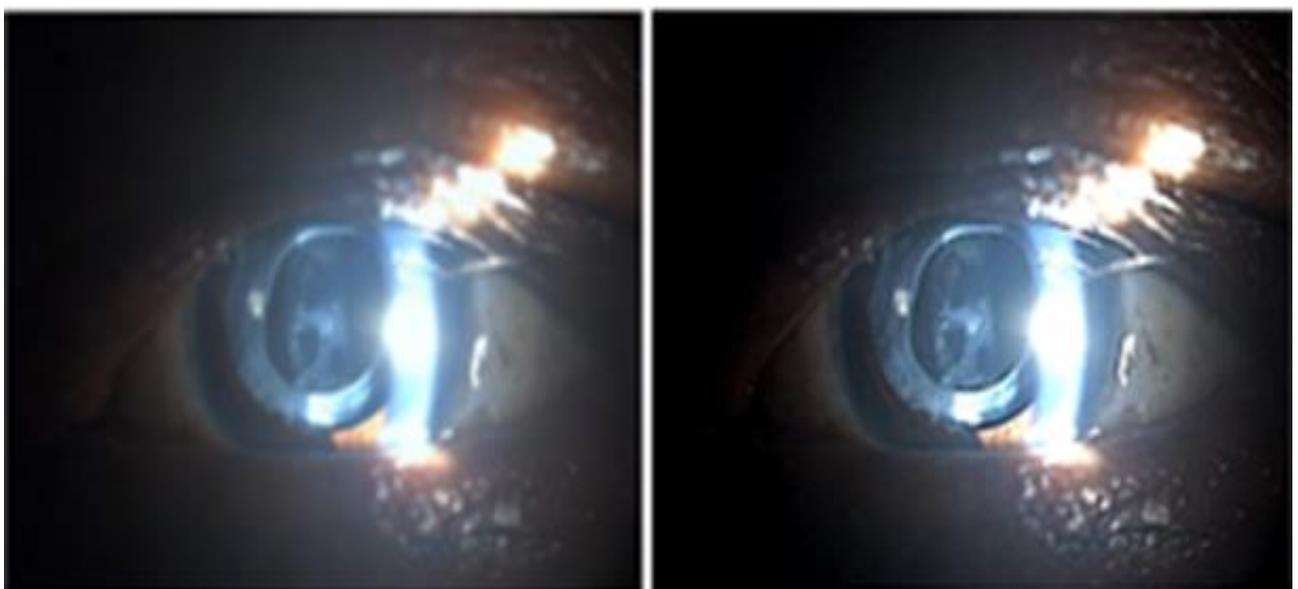
En el siguiente capítulo se desarrollan las pruebas realizadas al algoritmo propuestas por la metodología de desarrollo escogida XP, que se dividen en dos grupos: pruebas unitarias y pruebas de aceptación y por consecuente los resultados obtenidos, además se le aplican un conjunto de métricas para realizar la validación de los resultados.

### 3.1 Resultados

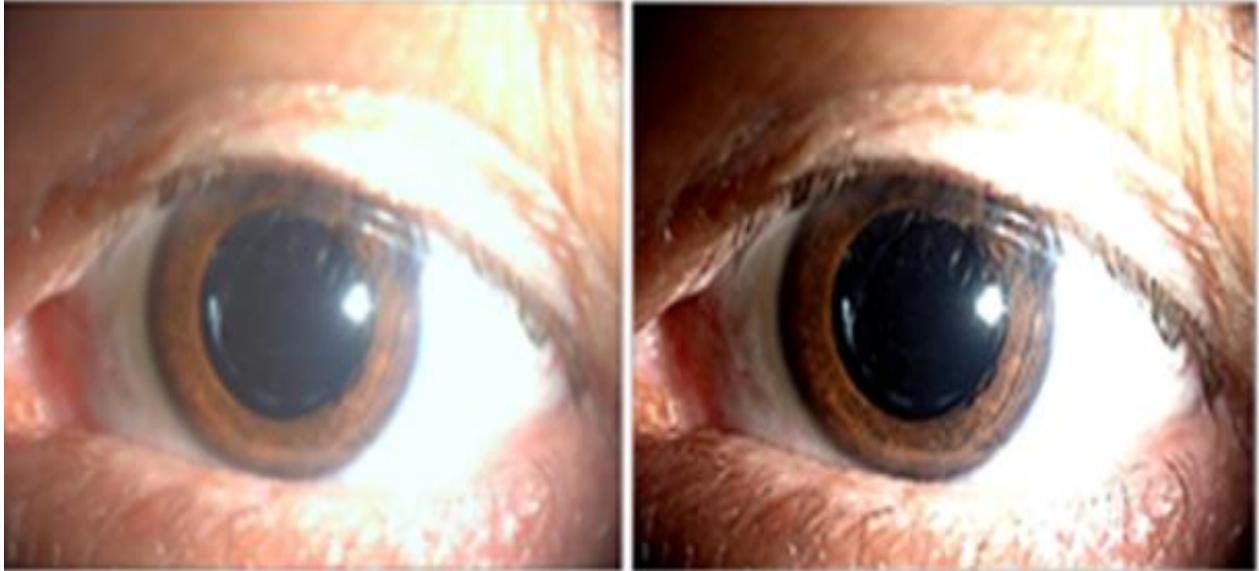
El algoritmo fue probado en 12 imágenes oblicuas, provenientes de pacientes los cuales han sido operados de cataratas y poseen opacidad en la capsula posterior. A continuación, se muestran los resultados obtenidos de aplicar este algoritmo a 3 imágenes provenientes de estos pacientes, mostrándose de izquierda a derecha, la imagen borrosa y la imagen restaurada (Figuras 6,7 y 8).



**Figura 6.** Imagen borrosa e imagen restaurada.



**Figura 7.** Imagen borrosa e imagen restaurada



**Figura 8.** Imagen borrosa e imagen restaurada

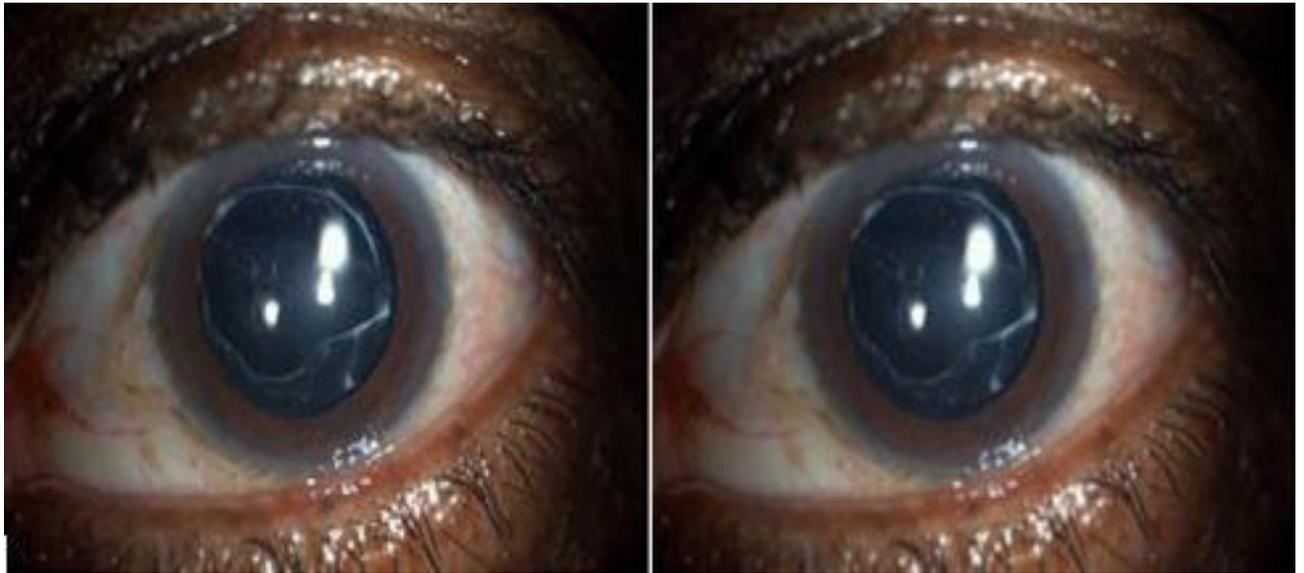
### 3.1.1 Validación de los resultados

Para la validación de los resultados se les aplicó el algoritmo a las 12 imágenes oblicuas de opacidad de la cápsula posterior tomadas a pacientes operados de cataratas, con que cuenta el grupo d investigación AIRI. A las mismas se le calcularon las siguientes métricas:

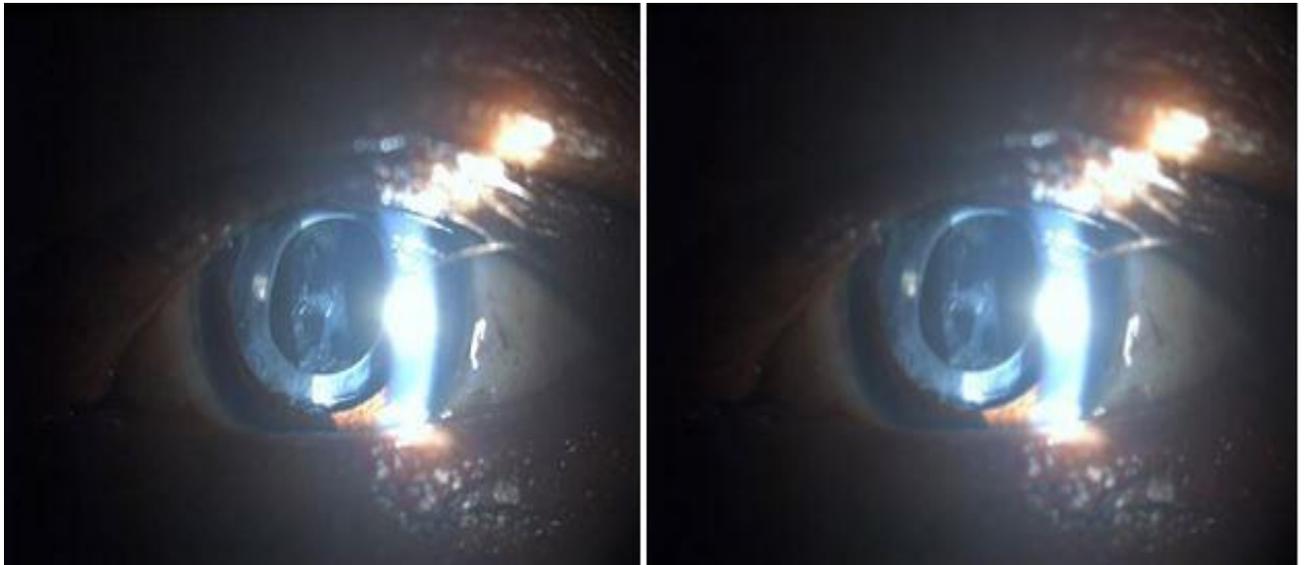
- Índice de similitud estructural (SSIM)
- Relación señal-ruido (SNR)
- Pico de relación señal-ruido (PSNR)

La métrica SSIM se basa en que las estructuras de los objetos de las escenas son independientes de la iluminación, de forma que la influencia de esta debe de ser aislada en el cálculo de la calidad de la imagen (Silvestre Blanes, y otros, 2007). Por lo que dicha métrica calcula el índice de similitud estructural de la imagen restaurada con respecto a la imagen original, este índice se expresa en valores entre 0 y 1, a medida que el valor se acerque más a 1, más similitud existe entre estas imágenes.

Para ello a las 12 imágenes de OCP se les aplica un emborronamiento artificial generando un *kernel* de paso bajo gaussiano. Un filtro paso bajo gaussiano consiste en la eliminación o atenuación de las altas frecuencias de la imagen (Ruiz Fernández, 2009). Para ello, bastará definir una máscara gaussiana de 32x32 con varianza 2, de forma que al aplicarla sobre el espectro de frecuencias mantenga los valores bajos, situados en el centro de la imagen, y elimine los altos. En la figura 9, 10 y 11 se muestran los resultados de convolucionar esta máscara gaussiana con las 3 imágenes oblicuas de OCP.



**Figura 9.** Imagen original e imagen emborronada



**Figura 10.** Imagen original e imagen emborronada



**Figura 11.** Imagen original e imagen emborronada

PSNR es una expresión de la relación entre el máximo valor posible (potencia) de una señal y la potencia de distorsión del ruido que afecta la calidad de su representación, mientras que SNR es un término técnico usado para caracterizar la calidad de la imagen detectada por un sistema de medición (Silvestre Blanes, y otros, 2007), por lo cual, la métrica SNR, calcula la relación señal-ruido entre la imagen restaurada y la imagen original, este valor mientras mayor sea, menor nivel de ruido presentará la imagen restaurada y la métrica PSNR es el pico de esa relación.

A continuación, se muestra una tabla con los resultados de estas métricas en las imágenes a las cuales se les aplico el algoritmo, en el caso del PSNR y el SNR se muestra de la siguiente manera: valor entre la imagen original y la imagen restaura/ valor entre la imagen borrosa y la imagen original.

Imágenes	SSIM	PSNR	SNR
Imagen 1	0.7664	16.2840/14.9832	13.0228/10.3863
Imagen 2	0.8732	19.3174/16.8364	11.8226/9.3981
Imagen 3	0.7010	27.2086/24.3893	17.9468/13.3832
Imagen 4	0.8289	21.7124/18.9361	11.4186/10.2849
Imagen 5	0.8144	18.5100/16.2387	13.7483/11.8935
Imagen 6	0.8543	25.2738/22.9036	16.9630/12.7832
Imagen 7	0.7791	17.4682/15.3517	14.9264/11.9832
Imagen 8	0.7927	19.2537/15.7101	15.2198/11.2046
Imagen 9	0.8823	22.0935/18.5502	18.4983/16.0821
Imagen 10	0.8490	20.0173/17.3751	15.4098/13.7302
Imagen 11	0.7973	21.0386/18.7639	16.0933/13.9838
Imagen 12	0.8767	18.9390/15.7252	17.0383/14.9263
Promedio	0.8179	20.5930/17.9802	15.1756/12.5032

**Tabla 7.** Resultados de las métricas SSIM, PSNR y SNR.

Después de aplicadas las métricas se pudo evidenciar que el valor de similitud estructural entre la imagen restaurada y la imagen original es de aproximadamente un 82%. Los valores de PSNR y SNR arrojados por las métricas demuestran que los valores entre la imagen original y la imagen restaurada son mayores que los valores entre la imagen borrosa y la imagen original, evidenciándose que la imagen restaurada presenta menor ruido que la imagen borrosa.

### 3.2 Fase de prueba

La realización de pruebas permite comprobar la eficacia de un sistema, estas son responsables de la verificación del cumplimiento de los objetivos trazados en la etapa de implementación. Con las pruebas se reduce el número de errores durante la implementación, el tiempo entre la introducción de estos en el sistema y su detección; son las encargadas de aumentar la seguridad y de evitar efectos colaterales no deseados a la hora de realizar modificaciones en la aplicación (Gutiérrez, y otros, 2011).

Uno de los pilares de la eXtreme Programming es el proceso de pruebas. XP anima a probar constantemente tanto como sea posible. Esto permite 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. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones. XP divide las pruebas del sistema en dos grupos: 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 diseñadas por el cliente final (Gutiérrez, et al., 2011).

### 3.2.1 Pruebas unitarias

Para las pruebas unitarias se utilizó el tipo de prueba caja blanca, en donde se realiza un examen minucioso de los detalles procedimentales, comprobando los caminos lógicos del programa, los bucles y condiciones, y examinado el estado del programa en varios puntos. Además, garantizan que (Durán, 2015):

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo.
- Se utilizan las decisiones en su parte verdadera y en su parte falsa.
- Se ejecuten todos los bucles en sus límites.
- Se utilizan todas las estructuras de datos internas.

### 3.2.2 Prueba del camino básico

La prueba de caja blanca realizada a la propuesta de solución fue la prueba del camino básico, a partir del cálculo de la complejidad ciclomática del algoritmo a ser analizado. Para realizarla se deben enumerar las sentencias de código y a partir de ahí elaborar el grafo de flujo de esta funcionalidad. Se definieron una serie de pasos a seguir (Durán, 2015):

1. **Notación del grafo de flujo:** usando el código como base se realiza la representación del grafo de flujo, mediante una sencilla notación. Cada construcción estructurada tiene su correspondiente símbolo.
  - **Nodo:** a cada círculo denominado nodo, representa una o más sentencias procedimentales.
  - **Arista:** las flechas del grafo de flujo, denominadas aristas, representan el flujo de control y son análogas a las flechas del diagrama de flujo.
  - **Región:** las áreas delimitadas por aristas y nodos se denominan regiones.
2. **Complejidad ciclomática:** es una métrica que proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado define el número de caminos independientes del conjunto básico de un programa. Esto indica el límite superior para el número de pruebas que se deben realizar, para asegurar que se ejecuta cada sentencia al menos una vez. Se utilizó la siguiente forma:  $V(G)$ , de un grafo de flujo  $G$  se define como:  $V(G)=A-N+2$ , donde  $A$  es el número de aristas del grafo de flujo y  $N$  es el número de nodos.

3. **Determinar un conjunto básico de caminos linealmente independientes:** el valor de  $V(G)$  es el número de caminos linealmente independientes de la estructura de control del programa.
4. **Obtención de casos de prueba:** se realizan los casos de pruebas que forzarán la ejecución de cada camino del conjunto básico.

Esta técnica se aplicará al método Cálculo\_de\_Oe de la clase RNAparaBID, a continuación, se muestra el fragmento de código al que se le aplica la técnica del camino básico y se le calcula la complejidad ciclomática:

Prueba de caja blanca	
Probado por: Manuel A. Zaldivar Batista	
<p>Código al que se le aplica:</p> <pre><code>%calculo de Oe e=aux-Xest; 1 eMedia=median(median(e)); 1  auxSuma=0; 1 [n,m]=size(Xest); 1 for(i=1:n) 2     for(j=1:m) 3         ek=aux(i,j) - Xest(i,j); 4         resta=ek-eMedia; 4         auxSuma=auxSuma+ (resta^2); 4     end 5 end 6  Oe=sqrt((1/n*m)*(auxSuma)); 7</code></pre>	<p>Representación en grafo del flujo</p>
<ol style="list-style-type: none"> <li>1. Complejidad ciclomática:</li> <li>2. <math>V(G) = (\text{Cantidad de Aristas} - \text{Cantidad de Nodos}) + 2 = (10 - 7) + 2 = 5</math></li> </ol>	<p>Caminos independientes:</p> <ol style="list-style-type: none"> <li>1. 1-2-3-4-5-6-7</li> <li>2. 1-2-7</li> <li>3. 1-2-3-2-7</li> <li>4. 1-2-3-4-5-6-2-7</li> <li>5. 1-2-3-4-5-3-2-7</li> </ol>

**Tabla 8.** Prueba de caja blanca.

A continuación, se muestra el caso de prueba para el camino básico 1:

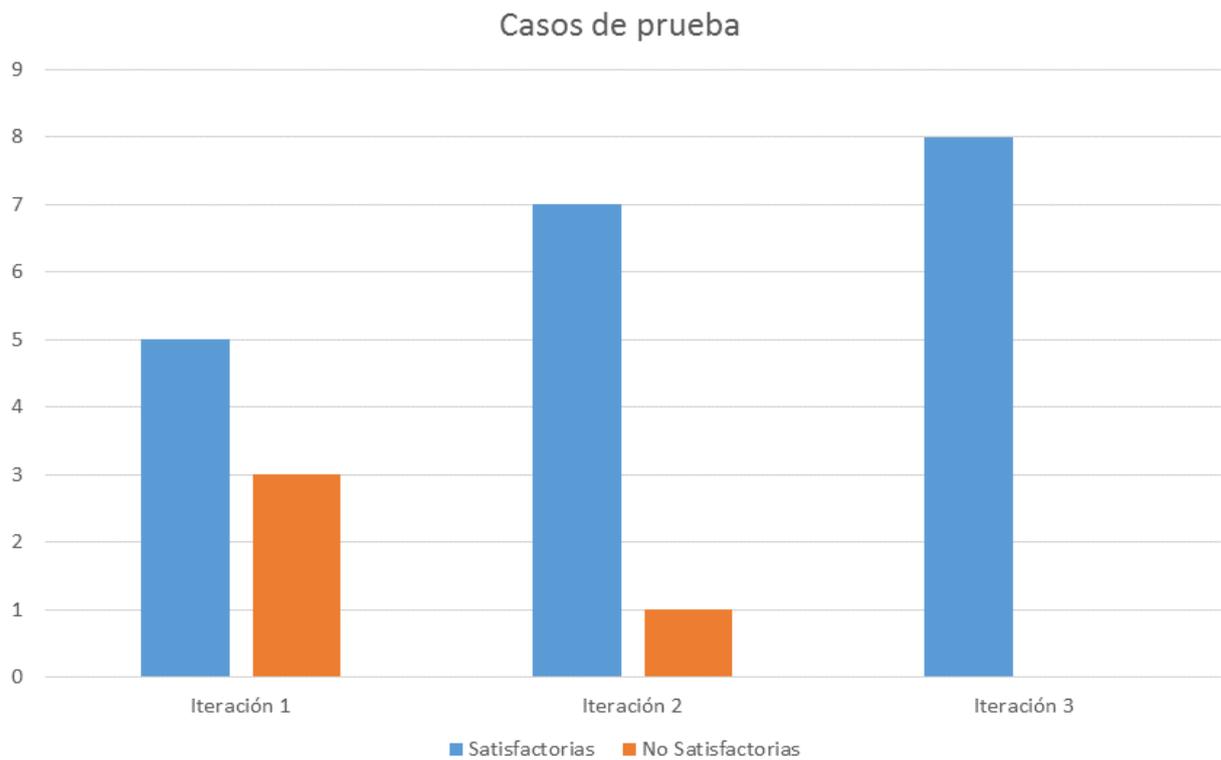
Entrada	La media entre la imagen borrosa menos la imagen estimada en ese paso.
Resultados esperados	El valor $O_e$ el cual es utilizado posteriormente para el cálculo del regularizador por variación total.
Condiciones	La imagen estimada tiene ser hallada a partir de la borrosa de entrada.

La técnica del camino básico se le aplicó al método Cálculo\_de\_ $O_e$  donde se realizaron 5 iteraciones al fragmento de código, en las cuales no se detectaron errores.

### 3.2.3 Pruebas de aceptación

Estas pruebas intentan encontrar casos de prueba en que el módulo no se atiene a su especificación. Se enfocan especialmente en los módulos que se relacionan con la interfaz de usuario ya que no requieren el conocimiento de la estructura interna del programa para su puesta en marcha. Las historias de usuario están asociadas a las pruebas de aceptación, o pruebas funcionales, estas se realizan en esta etapa del proyecto y en ellas se describen las posibles formas de utilización del software. Las pruebas de aceptación no solo validan la transformación de una entrada en una salida, sino que validan una característica completa (Pressman, 2007).

Para las pruebas de aceptación se diseñaron 8 casos de prueba de los cuales para una primera iteración se obtuvieron 5 de forma satisfactoria y 3 no satisfactoria. Para una segunda iteración se obtuvieron 7 casos de pruebas satisfactorios y solamente 1 no satisfactorio. Y para una tercera iteración se obtuvieron los 8 casos de pruebas satisfactorios, resultados mostrados en la gráfica siguiente:



**Figura 12.** Resultados de las pruebas de aceptación.

A continuación, se muestra uno de los casos de prueba diseñados para las pruebas de aceptación:

Caso de prueba de aceptación
Código: HU_5   Historia de Usuario: 5
Nombre: Calcular función de activación.
Descripción: La presente historia de usuario tiene como objetivo el cálculo de la función de activación, este es el principal paso del proceso <i>forward</i> con el cual su valor va a ser utilizado para las actualización de los demás parámetros del algoritmo.
Condiciones de Ejecución: Se tiene que haber tomado cada píxel de entrada como una neurona y haberse multiplicado por la matriz de peso.
Entrada/Pasos de Ejecución: <ul style="list-style-type: none"> <li>• Se toma cada píxel de entrada como neurona de entrada a la red</li> <li>• Se toma la matriz de peso y se multiplica por cada neurona de entrada</li> <li>• Se calcula la función de activación por cada capa.</li> </ul>

<p>Resultado Esperado:</p> <p>Se obtiene el valor de la función de activación por cada capa.</p>
<p>Evaluación de la Prueba: Prueba satisfactoria.</p>

**Tabla 9.** Caso de prueba de aceptación HU 5.

### 3.3 Conclusiones del capítulo

Durante el desarrollo del capítulo se realizaron pruebas para validar las funcionalidades del algoritmo propuesto, tales como las pruebas unitarias en las cuales no se detectaron errores y las pruebas de aceptación las cuales luego de 3 iteraciones se corrigieron las 4 no conformidades detectadas, lo que significa, que las historias de usuarios implementadas cumplen con los requisitos y necesidades del cliente. También se aplicaron una serie de métricas, donde SSIM evidenció que existe un 82% de similitud estructural entre la imagen restaurada y la original, y PSNR y SNR que los valores entre la imagen original y la imagen restaurada son mayores que los valores entre la imagen borrosa y la imagen original respectivamente, demostrando que la imagen restaurada presenta menor ruido que la imagen borrosa.

## Conclusiones

Al término de la investigación se puede afirmar que se ha cumplido con el objetivo general propuesto, así como con los objetivos específicos, obteniéndose las siguientes conclusiones:

- ✓ Fue realizado un estudio de los principales problemas que posee la realización de un diagnóstico, por parte de un especialista, a los pacientes operados de catarata, a partir de la observación de las imágenes con iluminación oblicua provenientes de la lámpara de hendidura, el cual evidenció la necesidad de aplicarle un proceso de restauración a dichas imágenes.
- ✓ De los algoritmos de restauración de imágenes estudiados, se determinó que poseen deficiencias en términos de tiempo de ejecución, la tolerancia a fallos y que no explotan el uso de un regularizador.
- ✓ El desarrollo de un algoritmo de restauración de imágenes utilizando una red neuronal a partir del método *Backpropagation*, permitió obtener una función de activación adecuada en el proceso de *forward*, la cual fue usada en la estimación de la imagen en el *backward*.
- ✓ En la validación se obtuvieron altos valores de similitud estructural (82%), los valores de las métricas SNR y PSNR fueron superiores en las imágenes restauradas que en las borrosas. Por lo que se evidenció que la imagen restaurada es semejante con la original y que presenta menor nivel de ruido que la borrosa.

## Recomendaciones

Para futuras investigaciones se recomiendan las siguientes acciones:

- ✓ Probar la utilización de otras condiciones de parada del algoritmo que no sean la cantidad de veces que se repite el proceso de restauración. Como pudiera ser el encontrar el valor mínimo de la función de coste.
- ✓ Realizar experimentaciones sobre otro número de capas ocultas y arquitecturas de la red.
- ✓ Valorar el uso de otros regularizadores para mejorar y complementar la restauración final.

## Referencias bibliográficas

- Babacan, S D, Molina, R and Katsaggelos, A K. 2009.** *Variational Bayesian blind deconvolution using a total variation prior.* s.l. : IEEE Transactions on Image Processing, 2009. 12-26.
- Banham, M R. 1994.** *Wavelet-Based Image Restoration Techniques.* Illinois : s.n., 1994.
- Barría, Fernando. 2012.** *Visión 20. Visión 20.* [Online] marzo 10, 2012. [Cited: mayo 24, 2017.] <https://vision2020la.wordpress.com/2012/03/30/monitoreo-de-cirugia-de-catarata-una-prioridad-en-ofthalmologia/>.
- Baume, G.L. 2014.** *Astronomía Observacional: Procesamiento de Imágenes.* 2014.
- Bertona, Luis Federico. 2005.** *Entrenamiento e redes neuronales basados en algoritmos evolutivos.* Buenos Aires : s.n., 2005.
- Boeras Vázquez, Mairelys, et al. 2012.** *Aplicando el método de Boehm y Turner.* La Habana : s.n., 2012.
- Boyd, Kierstan. 2014.** American Academy of Ophthalmology. [Online] marzo 3, 2014. <http://www.aao.org/salud-ocular/enfermedades/que-son-las-cataratas>.
- Campisi, Patrizio and Egiazarian, Karen. 2001.** *Blind image deconvolution: theory and applications.* 2001.
- Chan, T, et al. 2005.** Recent developments in total variation image restoration. *Handbook of Mathematical Models in Computer Vision.* s.l. : Springer Nature, 2005.
- Dr. Jhon Jairo Aristizábal. 2016.** [Online] 2016. [Cited: noviembre 17, 2016.] <http://www.jjaristizabal.com/condiciones-del-ojo/opacidad-de-capsula-posterior>.
- Durán, Pedro. 2015.** 6th International Conference on Model and Data Engineering. *6th International Conference on Model and Data Engineering.* [Online] noviembre 21, 2015. [Cited: abril 11, 2016.] <http://indalog.ual.es/MEDI2016/HOME.html>.
- Engl, H W, Hanke, M and Neubauer, A. 1996.** *Regularization of Inverse Problems.* s.l. : Kluwer Academic Publishers, 1996.
- Filter, Eduardo Lauder. 2012.** *Descripción, comparación y ejemplos de uso de la funciones del toolbox de procesamiento digital de imágenes de Matlab.* Madrid : s.n., 2012.
- Gamma, Erich. 2011.** *Patrones de Diseño. Elementos de software orientado a objetos reutilizables.* Madrid : s.n., 2011.
- Gennery, D B. 2013.** Determination of optical transfer function by inspection Determination of optical transfer function by inspection. *Journal of the Optical Society of America.* 2013, Vol. 63, 1571-1577.

- Gutiérrez, Javier José, et al. 2011.** *Pruebas del Sistema en Programación Extrema*. Sevilla : s.n., 2011.
- Hansen, P C. 1998.** *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. s.l. : SIAM, 1998.
- Hernández López, Eymard. 2010.** *Un acercamiento a la deconvolución ciega usando el algoritmo de Lucy-Richardson*. Iztapalapa, México : s.n., 2010.
- Jiang, Ming and Wang, Ge. 2003.** Development of blind image deconvolution and its applications. *X-Ray Science and Technology*. 2003.
- Jiménez, Javier. 2014.** *NetBeans IDE entorno de desarrollo para lenguajes como Java PHP C/C++ Groovy*. 2014.
- Kundur, Deepa and Hatzinakos, Dimitrios. 1996.** *Blind Image Deconvolution*. 1996.
- Lane, R G and Bates, R HT. 1987.** *Automatic multidimensional deconvolution*. 1987.
- Lee, H C. 2000.** *Review of image-blur models in a photographic system using the principle of optics*. s.l. : Optical Engineering, 2000. 405-421.
- López, Iván Hernández, et al. 2010.** *Estrategias de prevención de la opacidad de la cápsula posterior*. La Habana : s.n., 2010.
- Martínez, Francisco Javier. 2011.** *Guía de construcción de software en java con patrones de diseño*. Oviedo : s.n., 2011.
- MathWorks Inc. 2017.** MathWorks. [Online] 2017. [Cited: junio 16, 2017.] <https://es.mathworks.com/products/matlab/features.html>.
- Match, Damián Jorge. 2001.** *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Rosario : s.n., 2001.
- Ming, Jiang and Ge, Wang. 2003.** *Development of blind image deconvolution and its applications*. Beijing : s.n., 2003.
- Montenegro Vargas, Albert Ferney. 2010.** *Regularización de problemas inversos e imágenes borrosas*. Bogotá : s.n., 2010.
- Novara, Ing. Pablo. 2010.** *Introducción a las Herramientas de desarrollo*. Santa Fe : Universidad Nacional del Litoral, 2010.
- Oliveira, J., Bioucas-Dias, J. and Figueiredo, M. 2009.** *Adaptive total variation image deblurring: a majorization-minimization approach*. 2009. 2479-2493.
- Osher, S, Rudin, L and Fatemi, E. 1992.** *Nonlinear total variation based noise removal algorithms*. s.l. : Physica D, 1992. 259-268.

- Padilla, Carmen. 2016.** Instituto Cubano de Oftalmología Ramón Pando Ferrer. [Online] noviembre 9, 2016. <http://www.pando.sld.cu/general.html>.
- Parrallel, Serge Gregory. 1999.** *Logic Programming in PARLOG*. 1999.
- Pérez Candelaria, Eneida. 2016.** Instituto Cubano de Oftalmología Ramón Pando Ferrer. [Online] noviembre 9, 2016. <http://www.pando.sld.cu/servicio%20catarata.html>.
- Pressman, Roger S. 2007.** *Pressman*. 2007. Vols. IV-V.
- . **2007.** *Pressman*. 2007. Vol. VI.
- Rios Caso, Rinaldo. 2015.** Revista Cubana de Oftalmología. *Revista Cubana de Oftalmología*. [Online] abril 2015. [Cited: mayo 24, 2017.] [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S0864-21762015000200001](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21762015000200001).
- Ruiz Fernández, Luis Ángel. 2009.** *La transformada de Fourier. Aplicación al filtrado de imágenes*. Ingeniería Cartográfica, Geodesia y Fotogrametría., Universidad de Valencia. Valencia : s.n., 2009.
- Santiago Cabello, Miguel Ángel. 2011.** *Restauración de imágenes con desensibilización de estimaciones*. Madrid : s.n., 2011.
- Silvestre Blanes, Javier and Gorricho, Juan Luis. 2007.** *Técnicas de Evaluación de la Calidad de la Imagen. Tendencias y Métricas basadas en bordes*. Telemática, Instituto Tecnológico de Informática. Cataluña : s.n., 2007. Proyecto avalado por el MCYT (Ministerio de Ciencia y Tecnología de España).
- Sun, Yang. 2000.** *Hopfield neural network based algorithms for image restoration and reconstruction – Part I: algorithms and simulations*. s.l. : IEEE Transactions on Signal Processing, 2000. 2119-2131.
- Transitions Optical Inc. 2012.** Aprende Sobre Anteojos. [Online] 2012. [Cited: noviembre 14, 2016.] <http://www.aprendasobreanteojos.com/my-visit/vision-testing/slit-lamp-exam.aspx>.
- Triana, Juan Gabriel, Romero, Liliana Constanza and Roldán, Diego Gerardo. 2013.** *Restauración digital de imágenes mediante ecuaciones diferenciales parciales*. Bogotá : s.n., 2013. 511-518.
- Vique, Robert Ramírez. 2012.** *Métodos para el desarrollo de aplicaciones móviles*. Barcelona : s.n., 2012.
- Visual Paradigm Inc. 2014.** Visual Paradigm. [Online] 2014. [Cited: noviembre 17, 2016.] [www.visual-paradigm.com](http://www.visual-paradigm.com).
- Wang, J, Liao, X and Yi, Z. 2005.** *Image restoration using hopfield neural network based on total variational model*. Berlín : Springer Berlin, 2005. 735-740.
- Wu, Y -D, et al. 2007.** *Variational PDE based image restoration using neural network*. 2007. 85-93.



# Anexos

## Anexo 1: Historias de usuario

Historia de Usuario	
<b>Número:</b> HU 1	<b>Nombre Historia de Usuario:</b> Cargar imagen.
<b>Usuario:</b> Manuel A. Zaldívar	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo cargar la imagen degradada.	
<b>Observaciones:</b>	

**Tabla 10.** Historia de usuario HU 1

Historia de Usuario	
<b>Número:</b> HU 2	<b>Nombre Historia de Usuario:</b> Convertir imagen de rgb a ycbcr.
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo transformar al espacio color ycbcr para así trabajar con las intensidades de la imagen.	
<b>Observaciones:</b>	

**Tabla 11.** Historia de usuario HU 2

Historia de Usuario	
<b>Número:</b> HU 3	<b>Nombre Historia de Usuario:</b> Convertir imagen de ycbcr a rgb.
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La presente historia de usuario tiene como objetivo recuperar la imagen al formato de color rgb.	
<b>Observaciones:</b>	

**Tabla 12.** Historia de usuario HU 3

Historia de Usuario	
<b>Número:</b> HU 4	<b>Nombre Historia de Usuario:</b> Calcular pesos sinápticos de la RNA.
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 3
<b>Riesgo en Desarrollo:</b> Medio	<b>Puntos Reales:</b> 2
<b>Descripción:</b> La presente historia de usuario tiene como objetivo calcular los pesos sinápticos de la imagen para ser utilizados en el proceso iterativo de restauración del algoritmo.	
<b>Observaciones:</b>	

**Tabla 13.** Historia de usuario HU 4

Historia de Usuario	
<b>Número:</b> HU 6	<b>Nombre Historia de Usuario:</b> Calcular regularizador de la variación total.
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 3
<b>Riesgo en Desarrollo:</b> Medio	<b>Puntos Reales:</b> 3
<b>Descripción:</b> La presente historia de usuario tiene como objetivo el cálculo de este parámetro que es utilizado en el cálculo del vector gradiente local y la función de coste.	
<b>Observaciones:</b>	

**Tabla 14.** Historia de usuario HU 6

Historia de Usuario	
<b>Número:</b> HU 7	<b>Nombre Historia de Usuario:</b> Calcular vector gradiente local.
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 3
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 3
<b>Descripción:</b> La presente historia de usuario tiene como objetivo calcular el vector gradiente, para esto se aplica la formula descrita en la propuesta de solución en el paso 4.	
<b>Observaciones:</b>	

**Tabla 15.** Historia de usuario HU 7

Historia de Usuario
---------------------

<b>Número:</b> HU 8	<b>Nombre Historia de Usuario:</b> Realizar expansión del histograma.	
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 2	
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 2	
<b>Riesgo en Desarrollo:</b> Medio	<b>Puntos Reales:</b> 2	
<b>Descripción:</b> La presente historia de usuario tiene como objetivo realizar la expansión de los pixeles de la imagen.		
<b>Observaciones:</b>		

**Tabla 16.** Historia de usuario HU 8

Historia de Usuario		
<b>Número:</b> HU 9	<b>Nombre Historia de Usuario:</b> Calcular imagen estimada.	
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 2	
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 3	
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 2	
<b>Descripción:</b> La presente historia de usuario tiene como objetivo la estimación de la imagen que se devolverá.		
<b>Observaciones:</b>		

**Tabla 17.** Historia de usuario HU 9

Historia de Usuario		
<b>Número:</b> HU 10	<b>Nombre Historia de Usuario:</b> Mostrar imagen restaurada.	
<b>Usuario:</b> Manuel A. Zaldivar	<b>Iteración Asignada:</b> 2	
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1	
<b>Riesgo en Desarrollo:</b> Bajo	<b>Puntos Reales:</b> 1	
<b>Descripción:</b> La presente historia de usuario tiene como objetivo mostrar al usuario los resultados obtenidos en el proceso de restauración.		
<b>Observaciones:</b>		

**Tabla 18.** Historia de usuario 10

## Anexo 2: Tarjeta CRC

Nombre de la clase: SerieD	
Responsabilidades	Clases Relacionadas
Calcular el parámetro MDiag	-size, diag,
Calcular el parámetro DelthaBeta	-diag, zeros
Calcular el parámetro Dv, Dh	-size
Calcular el parámetro sumaV, sumaH	-size

**Tabla 19.** Tarjeta CRC de la clase SerieD.