



**UNIVERSIDAD DEL MAR
CAMPUS PUERTO ESCONDIDO**

RESTAURACIÓN DE IMÁGENES CON DEGRADACIÓN POR
BORROSIDAD EMPLEANDO TÉCNICAS DE
PROCESAMIENTO DIGITAL DE IMÁGENES

T E S I S

QUE PARA OBTENER EL TÍTULO DE
LICENCIADA EN INFORMÁTICA

PRESENTA
PAULA FRANCO SANTOS

DIRECTOR DE TESIS
M. EN C. JOSÉ EDGAR LARA RAMÍREZ

Dedicatoria

*Dedico esta tesis a **Dios**, que me ha dado la vida y por permitirme terminar este trabajo, porque me ha dado grandes bendiciones que han coadyuvado a la conclusión de este gran proyecto, porque sé que sola no hubiera tenido la fuerza necesaria para continuar luchando contra todo obstáculo que a lo largo de la carrera se ha presentado.*

*A mis padres **Vicencio Franco Venegas** y **Epifanía Santos Colmenares** que me han brindado todo su apoyo, amor y consejos en cada momento, porque gracias a su guía he llegado a realizar uno de los anhelos más grandes de la vida, terminar mis estudios profesionales; de todo corazón deseo expresarles que mis logros han sido también suyos.*

*A mis hermanos **Armando, Rogelio, Victoria y Antonia** por haberme brindado su apoyo incondicional, comprensión y confianza.*

Agradecimientos

Agradezco a la Universidad del Mar campus Puerto Escondido por haberme abierto las puertas, porque sé que con su ayuda el camino fue menos difícil. Por su gran apoyo al facilitarme los medios suficientes para terminar la carrera, además de brindarme el financiamiento de gran parte de mis estudios, otorgándome las becas de colegiatura y alimentación desde el primer año que me dieron la oportunidad de formar parte de sus filas.

Al Programa Nacional de Becas para la Educación Superior (PRONABES) porque su existencia motivó mis deseos de superación y fue una gran contribución para poder continuar con mis estudios. Además por el financiamiento de esta tesis de licenciatura mediante la concesión de la beca de titulación a través del programa becanet - superior.

Desde luego un agradecimiento a mis maestros, ya que sus enseñanzas me han dado una inmensa satisfacción día a día, porque por ustedes he alcanzado todos los logros que hasta ahora he tenido, ya que adquirí sus sabios consejos, compartí un valioso e interesante proceso de enseñanza-aprendizaje y por su capacidad para guiar mis ideas.

Un especial agradecimiento a mis padres por su constante apoyo por lo cual les estaré eternamente agradecida, sé que esos cimientos que han puesto en mí, han sido la base para llegar hasta donde hoy estoy.

Gracias a mi director de tesis M. en C. José Edgar Lara Ramírez por aceptarme para realizar esta tesis bajo su dirección, la ayuda, confianza, paciencia y sus consejos en mi trabajo han sido un aporte invaluable no solamente en el desarrollo de este trabajo, sino también en mi formación como profesional.

Por sus observaciones y consejos agradezco a los revisores de este trabajo: M. en C. Isidro Moctezuma Cantorán, M. en C. Jorge Ochoa Somuano, M. en C. José Francisco Delgado Orta y M. en C. Manuel Alejandro Valdés Marrero.

Resumen

En este trabajo de tesis se realizó un estudio sobre la restauración digital de imágenes con degradación por borrosidad, mediante la utilización de técnicas de procesamiento digital de imágenes, dentro de los algoritmos que aplican este tipo de técnicas se encuentran los siguientes: Filtro de Wiener, Filtro Inverso y el criterio de Minimización de rizado; para poder implementarlos se creó una aplicación con el uso de la herramienta Matlab. El objetivo fue disminuir o remover los efectos que degradan una determinada imagen y de esta forma obtener imágenes de mejor calidad. El proceso de restauración utilizado en este trabajo de tesis, requiere el conocimiento de los parámetros de ruido que causan la degradación (ángulo y distancia hacia donde se efectúa la borrosidad); aun, cuando los valores de estos parámetros son desconocidos, es posible obtener una aproximación de ellos.

Se procesaron los resultados y con base en la evaluación que se hizo después de aplicar el filtro de mediana con los tres métodos de restauración, se concluyó que el filtro de Wiener es más eficiente, con un porcentaje de error de 6.47%, mientras que al aplicar el filtro inverso el porcentaje de error fue más alto, de 13.50%, y con el criterio de minimización de rizado se tuvo 8.28%. Los porcentajes de error se obtienen al aplicar la fórmula del cálculo del error relativo, tomando como denominador el punto medio del rango del nivel de intensidad que es 128. De acuerdo a estos resultados, se recomienda el uso del filtro de Wiener porque presenta un porcentaje de error menor, en el caso de usar los filtros restantes es recomendable aplicar el filtrado de mediana para reducir el ruido sal y pimienta que pueda contener la imagen y así lograr la disminución del porcentaje de error. En la mayoría de los casos de estudio evaluados, se encontró que el ruido sal y pimienta causa un aumento en el porcentaje de error en los filtros inverso y minimización de rizado, al aplicar el filtro de mediana se consiguió reducir este ruido, sin embargo no se logró que estas dos técnicas obtuvieran mejores resultados.

Abstract

This research project entails a digital restoration project of images degraded by blurring, using digital image processing techniques that include the following algorithms: Wiener filter, inverse filtering, and ripple minimization; to implement them, it was used the Matlab tool. The objective was to reduce or remove the effects that degrade a given image obtaining images of better quality. The restoration process requires knowledge of noise parameters that cause degradation (angle and distance to where the blurring takes effect); however, when the values of these parameters are unknown it is possible to approximate them.

After the results were processed, an evaluation was made after applying the median filtering using the above-mentioned methods of restoration and the following conclusion was reached: The Wiener filter is more efficient than the others filters, with an error percentage of 6.47 %. While applying the inverse filter, the percentage was the highest, 13.50 %, and the ripple minimization technique error was 8.28 %. The error percentages are obtained applying the formula for calculating the relative error, using as a denominator the midpoint of the range of intensity level is 128. In agreement to these results, it is recommended to use the Wiener filter because it present a minor error percentage, in case of using the others filters, it is necessary to apply the median filtering to reduce the salt and pepper noise contained in the image, to decrease the error percentage. In most case studies evaluated, it was found that the salt and pepper noise causes an increase in the error percentage, the filters inverse and ripple minimization, to apply the median filter to reduce this noise, however does not possible that these two techniques to obtain better results.

CONTENIDO

LISTADO DE FIGURAS.....	viii
LISTADO DE TABLAS	xi
LISTADO DE ECUACIONES	xii
LISTADO DE CÓDIGO FUENTE.....	xiii
GLOSARIO DE TÉRMINOS	xiv
CAPÍTULO 1. INTRODUCCIÓN	1
CAPÍTULO 2. ANTECEDENTES	5
2.1. Estado del arte y trabajos relacionados.....	5
2.2. Justificación	14
2.3. Planteamiento del problema.....	15
2.4. Objetivos.....	16
2.5. Alcances y límites del estudio	16
CAPÍTULO 3. MARCO TEÓRICO	19
3.1. Procesamiento digital de imágenes.....	19
3.2. Formación de una imagen en el sistema de visión humano.....	23
3.3. Pixel	25
3.4. Imagen digital	26
3.5. Imagen de color RGB	29
3.6. Imagen a escala de grises.....	30
3.7. Histograma de una imagen	31
3.8. Filtrado de la mediana.....	31

3.9. Degradación de imagen.....	32
3.10. Modelo de degradación.....	34
3.11. Restauración digital de imagen.....	35
3.12. Dominio espacial y dominio de la frecuencia.....	35
3.13. Transformada de Hough	37
3.14. Transformada de Fourier	39
3.15. Métodos de restauración	43
3.16. Filtro de Wiener	43
3.17. Filtro inverso.....	45
3.18. Criterio minimización de rizado	46
3.19. Teorema de No Free Lunch	47
3.20. Matlab	48
 CAPÍTULO 4. DESARROLLO DEL TEMA.....	 50
4.1. Adquisición y características de las imágenes.....	51
4.2. Aplicación de ruido a las imágenes	54
4.3. Implementación de los algoritmos de restauración.....	56
4.4. Realización de pruebas y mejoras.....	58
4.5. Evaluación y comparación de resultados.....	67
 CAPÍTULO 5. CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS.....	 82
 ANEXO A. MANUAL DE USUARIO.....	 85
 ANEXO B. CÓDIGO FUENTE DE LA APLICACIÓN.....	 97
 ANEXO C. CONTENIDO DEL CD.....	 107
 REFERENCIAS	 110

LISTADO DE FIGURAS

Figura 2.1.	Restauración de la imagen del planeta Saturno.....	6
Figura 2.2.	Resultados de degradación de una imagen.....	8
Figura 2.3.	Resultados de la restauración en 6 imágenes de prueba.....	9
Figura 2.4.	Resultados del desempeño de los métodos evaluados.....	11
Figura 2.5.	Restauración de la imagen de las placas de un vehículo en movimiento.....	12
Figura 2.6.	Resultados de la restauración de una imagen utilizando los métodos: Multigrid, Gauss Seidel, Jacobi y Gradiente.....	14
Figura 3.1.	Etapas fundamentales del procesamiento digital de imágenes.....	20
Figura 3.2.	Operación básica de un dispositivo CCD.....	21
Figura 3.3.	Formación de la imagen en el ojo humano.....	24
Figura 3.4.	Analogía entre el ojo humano y la cámara fotográfica.....	25
Figura 3.5.	Representación de un pixel.....	26
Figura 3.6.	Notación de una imagen digital.....	27
Figura 3.7.	Ejemplos de los efectos de la cuantificación.....	28
Figura 3.8.	Ejemplos de los efectos de muestreo.....	29
Figura 3.9.	Composición de matrices de una imagen en color RGB, en Matlab...	29
Figura 3.10.	Imagen de prueba y su histograma correspondiente.....	31
Figura 3.11.	Proceso para obtener la mediana usando un vecindario de 3 x 3.....	32
Figura 3.12.	Resultado del filtro de la mediana.....	32
Figura 3.13.	Representación de la función de probabilidad del ruido uniforme.....	33
Figura 3.14.	Modelo de degradación.....	34
Figura 3.15.	Tratamiento de la información de una imagen en el dominio de la frecuencia.....	36
Figura 3.16.	Representación gráfica de la transformada de Hough.....	37
Figura 3.17.	Representación de la ecuación paramétrica de una recta.....	38
Figura 3.18.	Ejemplo de la transformada de Hough en una imagen.....	38
Figura 3.19.	Ejemplo de la transformada de Fourier en una imagen.....	40
Figura 3.20.	Ejemplo de la restauración de una imagen digital usando filtro	

	Wiener.....	44
Figura 3.21.	Ejemplo de la restauración de una imagen digital usando filtro inverso.....	45
Figura 3.22.	Restauración de una fila f	47
Figura 4.1.	Etapas del trabajo desarrollado.....	51
Figura 4.2.	Imágenes originales de prueba.....	53
Figura 4.3.	Muestra de imagen en RGB y su resultado a escala de grises.....	54
Figura 4.4.	Imágenes con degradación por borrosidad.....	55
Figura 4.5.	Resultados al aplicar la transformada de Hough.....	57
Figura 4.6.	Imágenes restauradas, donde el ángulo equivale a 0°	59
Figura 4.7.	Imagen restaurada con el filtro de minimización de rizado donde el ángulo es distinto a 0°	60
Figura 4.8.	Imágenes restauradas, donde el ángulo es distinto a 0°	60
Figura 4.9.	Imagen restaurada a color.....	63
Figura 4.10.	Imagen de prueba para evaluar con diferentes parámetros de ruido...	63
Figura 4.11.	Resultados de restauración digital a una sola imagen de prueba.....	64
Figura 4.12.	Imágenes restauradas con valores aproximados.....	65
Figura 4.13.	Imágenes restauradas con valores aproximados con los tres filtros propuestos.....	66
Figura 4.14.	Imagen de prueba para métodos de evaluación.....	68
Figura 4.15.	Datos de la imagen cuadros.....	69
Figura 4.16.	Matriz de resultados de la resta de imágenes original y restaurada....	69
Figura 4.17.	Matriz de porcentajes de error.....	70
Figura 4.18.	Matriz de porcentajes de error a partir del punto medio.....	71
Figura 4.19.	Gráficas de porcentajes de error, donde el ángulo equivale a 0°	71
Figura 4.20.	Gráficas de porcentajes de error, donde el ángulo es distinto a 0°	72
Figura 4.21.	Histograma de imagen con los distintos métodos de restauración.....	80
Figura A.1.	Cambio de ruta, para poder llevar a cabo la ejecución de la aplicación.....	86
Figura A.2.	Archivo principal.m.....	86

Figura A.3.	Ventana de bienvenida de la aplicación.....	87
Figura A.4.	Ventana del filtro de Wiener.....	88
Figura A.5.	Ventana del filtro inverso.....	88
Figura A.6.	Ventana del filtro del criterio de minimización de rizado.....	89
Figura A.7.	Ventana para la búsqueda de archivo que se desea abrir.....	91
Figura A.8.	Visualización de la imagen en la aplicación.....	91
Figura A.9.	Resultado de la conversión de la imagen “Trees” a escala de grises...	92
Figura A.10.	Resultado de la aplicación de ruido a la imagen.....	92
Figura A.11.	Resultado de la restauración de la imagen borrosa.....	93
Figura A.12.	Histograma de la imagen original y restaurada.....	93
Figura A.13.	Resultado del porcentaje de error de la imagen.....	94
Figura A.14.	Gráficas de los resultados de porcentajes de error, respecto a un punto medio (128 píxeles).....	94
Figura A.15.	Resultados de la medición de las imágenes de prueba y original.....	94
Figura A.16.	Visualización del cálculo del ángulo para una imagen borrosa.....	95
Figura A.17.	Resultados de la medición de las imágenes de prueba y original.....	96
Figura C.1.	Estructura del contenido del CD.....	107
Figura C.2.	Archivos fuente de la aplicación.....	108
Figura C.3.	Documento de tesis en formato pdf.....	108
Figura C.4.	Estructura de la carpeta imágenes de prueba.....	108

LISTADO DE TABLAS

Tabla I	Mediciones de imágenes con los tres filtros de restauración.....	76
Tabla II	Tiempos aproximados del proceso de restauración.....	77
Tabla III	Promedio de los porcentajes de error.....	78
Tabla IV	Promedio general de los porcentajes de error de la tabla III.....	78
Tabla V	Resultados de la evaluación a una sola imagen con diferentes parámetros de degradación.....	79
Tabla VI	Promedio general de los porcentajes de error de la tabla V.....	80
Tabla VII	Funciones y herramientas de la ventana principal.....	87
Tabla VIII	Funciones y herramientas de las ventanas de los filtros.....	89

LISTADO DE ECUACIONES

Ecuación 1	Ecuación para calcular el ISNR.....	10
Ecuación 2	Sistema de ecuaciones.....	13
Ecuación 3	Notación matricial de una imagen digital.....	27
Ecuación 4	Fórmula para obtener el número de niveles de cuantificación.....	28
Ecuación 5	Ecuación para convertir una imagen a escala de grises.....	30
Ecuación 6	Ecuación que permite convertir de RGB a YIQ.....	30
Ecuación 7	Función de densidad de probabilidad.....	34
Ecuación 8	Relación entrada-salida del modelo de degradación.....	34
Ecuación 9	Expresión para denotar técnicas que pertenecen al dominio espacial..	36
Ecuación 10	Ecuación paramétrica de una recta.....	38
Ecuación 11	Transformada de Fourier de una función $f(x)$ para funciones continuas.....	40
Ecuación 12	Ecuación de la inversa de Fourier para funciones continuas.....	40
Ecuación 13	Transformada discreta de Fourier.....	41
Ecuación 14	Inversa de la transformada discreta de Fourier.....	41
Ecuación 15	Transformada de Fourier de una imagen $f(x, y)$ de tamaño $M \times N$	41
Ecuación 16	Inversa de la transformada de Fourier de una imagen $f(x, y)$ de tamaño $M \times N$	42
Ecuación 17	Ecuación para definir el error cuadrático medio.....	44
Ecuación 18	Fórmula para representar el filtro inverso.....	45
Ecuación 19	Ecuación matricial para aplicar degradación a imágenes.....	46
Ecuación 20	Fórmula para obtener el error relativo.....	68
Ecuación 21	Ecuación para obtener el error relativo a la máxima diferencia.....	68
Ecuación 22	Ecuación para obtener el porcentaje de error si se toma el punto medio.....	70

LISTADO DE CÓDIGO FUENTE

Código 1	Método abrirImagen().....	98
Código 2	Método guardarImagen().....	98
Código 3	Método convertirEscalaGris().....	99
Código 4	Método agregarRuido().....	99
Código 5	Método restaurarImagenWiener().....	100
Código 6	Método restaurarImagenInverso().....	100
Código 7	Método restaurarImagenRizado().....	101
Código 8	Método rest().....	102
Código 9	Método calcularAngulo().....	103
Código 10	Método hough().....	104
Código 11	Método calcularHistograma().....	104
Código 12	Método aplicarMediana().....	105
Código 13	Método calcularPorcentajeError().....	105
Código 14	Método editAnguloRes().....	106

GLOSARIO DE TÉRMINOS

BORDE	Es un área donde existe un cambio importante de intensidad o de iluminación en la imagen, un ejemplo es la región donde termina un objeto y comienza otro con niveles de gris distintos (Sucar 2008).
BORROSA	Impreciso. Que no se ve o no se distingue claramente (Lara 2000).
BRILLO	Indica la intensidad de luz emitida sobre un área específica (Somolinos 2002).
CONVOLUCIÓN	La convolución es un operador matemático que transforma dos funciones f y g en una tercera función que en cierto sentido representa la magnitud en la que se superponen f y una versión trasladada e invertida de g . En la convolución el valor de un pixel de salida se calcula mediante la suma ponderada de los pixeles vecinos. Dentro del campo del procesamiento de imágenes, la convolución se realiza entre una imagen f y una matriz llamada máscara para filtrar g , la convolución de f y g se denota $f * g$. (Esqueda & Palafox 2005).
DEGRADACIÓN	Reducir o desgastar las cualidades inherentes a alguien o algo (RAE 2001). De lo cual se deduce que degradación en una imagen es la disminución gradual de cualidades o características de la imagen.
DIGITAL	Cuando se hace referencia a esta palabra se indica que opera mediante la conversión a dígitos o a unidades discontinuas de los datos o las señales que maneja, por ejemplo las computadoras (Lara 2000).

FILTRO	Un filtro es una ecuación matemática que permite modificar el valor de un pixel según los valores de los pixeles contiguos. El filtrar una imagen consisten en aplicar una transformación para obtener una nueva imagen de forma que ciertas características son acentuadas o disminuidas (Sucar & Gómez 2001).
LUMINANCIA	Magnitud que expresa el flujo luminoso en una dirección determinada, por unidad de ángulo sólido y por unidad de área proyectada de la superficie radiante sobre el plano normal a la dirección de radiación (RAE 2001).
LUMINOSIDAD	Es el brillo de una zona respecto a otra zona blanca de la imagen (Somolinos 2002).
PROCESAMIENTO	Aplicación sistemática de una serie de operaciones sobre un conjunto de datos, generalmente por medio de máquinas, para explotar la información que estos datos representan (Lara 2000).
PSF	Del inglés <i>point spread function</i> , función de la extensión del punto. Representa la función de emborronamiento bidimensional del sistema de captación de imágenes, que generalmente es variante por traslaciones (Banham & Katsaggelos 1997).
RIZADO	El factor de rizado se define como la relación entre la tensión pico-pico de la señal rectificada (diferencia entre la fluctuación máxima y el mínimo alrededor del valor medio) y la componente continua de la misma (GTE 2004). El rizado corresponde a las ondulaciones o fluctuaciones que se presentan en una señal, en este caso señal de una imagen.

TONO Indica si un área parece similar al rojo, amarillo, verde, azul, o a una proporción de ellos (Somolinos 2002).

CAPÍTULO 1. INTRODUCCIÓN

La base del presente proyecto es el procesamiento digital de imágenes y su línea de investigación está enfocada a la restauración de imágenes digitales. Dado que en este trabajo la materia prima son imágenes, es necesario empezar definiendo qué es una imagen y más específicamente qué es una imagen digital.

Una imagen es una representación o ilustración de algo que puede ser una cosa, un objeto, una persona, o incluso un lugar, el contenido de una imagen es información que describe al objeto, y se presenta a las personas para que puedan visualizarlo y generar una interpretación, formando así sus propias conclusiones sobre ciertos fenómenos en distintas ciencias. Según González & Woods (1992), una imagen digital se define como una representación en dos dimensiones del mundo visual, corresponde a una matriz de valores que indican la intensidad o brillantez del objeto. Los seres humanos tienen la tendencia de realizar una representación visual de las acciones que realizan; de los sentidos que poseen,

el de la visión es el más desarrollado, éste es el medio por el cual el organismo capta la información.

De acuerdo a los estudios realizados por Antonio López Alemany, publicados por Ortega (2007), las nuevas tecnologías de visión han hecho que el 85% de la información que el ser humano recibe sea a través de los ojos, y esto sigue aumentando a medida que la tecnología avanza, lo que constituye un reto en el futuro, ya que gran parte de esta información son imágenes.

La computadora es una herramienta creada por el hombre con el fin de almacenar y procesar información, mucha de esa información actualmente la encontramos en forma de imágenes digitales.

En los últimos años el manejo de imágenes digitales se ha convertido en un tema de interés general, ya existen áreas del conocimiento en que las imágenes constituyen la materia prima para la extracción de información, esto hace que el tratamiento de las imágenes sea algo inherente al desarrollo tecnológico y científico.

Existen ciertas ramas dentro de la computación encargadas de estudiar el tratamiento de este tipo de imágenes, de las cuales se pueden destacar:

Visión por computadora: Sucar (2008) define visión como el mecanismo que permite reconocer y percibir objetos en el ambiente. En un ser vivo es el cerebro quien le da un significado a la información que se obtiene a través de la vista, analiza las imágenes de lo particular a lo general. Desde el punto de vista informático, la visión por computadora es un área de la inteligencia artificial que trata de estudiar los procesos de la visión humana con la finalidad de entenderlos y así poder generar técnicas de conocimiento que ayudarán a construir máquinas que tengan características y comportamiento similar al de los seres humanos, las cuales podrán percibir e interpretar lo que ven.

La visión por computadora tiene como objetivo extraer, clasificar y medir características de una imagen para su descripción e interpretación por la computadora. Dentro de las numerosas aplicaciones que existen de la visión por computadora, se mencionan las siguientes áreas de aplicación: procesos industriales, percepción remota, robótica, interpretación de escritura, reconocimiento de personas y apoyo al diagnóstico médico. La diferencia con el procesamiento digital de imágenes son los resultados, en el área de visión por computadora los resultados son descriptivos, es decir permiten

determinar la identidad y localización de objetos en una imagen, mientras en el procesamiento de imágenes los resultados son imágenes.

Procesamiento digital de imágenes: es una disciplina que cuenta con diferentes técnicas, las cuales permiten procesar imágenes por medio de una computadora (Castleman 1996).

En el área de procesamiento de imágenes, a menudo se encuentran imágenes que no tienen la calidad adecuada, debido a que existe una degradación o ruido en ellas, situación que impide obtener información fidedigna. En ocasiones, el contenido en las imágenes distorsionadas es muy valioso y no hay manera de poder acceder a ella sin realizar una restauración de la imagen, situación que se complica debido al desconocimiento de las herramientas que permiten realizar la restauración de imágenes.

La importancia del procesamiento digital de imágenes radica en que cada día más áreas del conocimiento hacen uso de imágenes como una fuente importante de información. Entre las más conocidas se encuentran la astronomía, biología, medicina, criminalística, historia y robótica.

Los esfuerzos de este trabajo de tesis están enfocados a eliminar el efecto de borrosidad que produce el movimiento en la cámara al tomar una fotografía y recuperar la imagen real a partir de la imagen distorsionada.

En este trabajo se usaron diversos algoritmos de restauración de imágenes, los cuales se implementaron en computadora. Entre los algoritmos empleados está el filtro de Wiener, el filtro inverso y el criterio de minimización de rizado. De cada uno se realiza una evaluación de los resultados obtenidos.

Entre las herramientas utilizadas está el software Matlab, el cual permite la manipulación de datos matriciales, que es la forma en la que están estructuradas las imágenes al ser manipuladas por una computadora y además cuenta con funciones apropiadas para el procesamiento de imágenes.

Organización del documento de tesis

Esta tesis está estructurada en cinco capítulos: la introducción, antecedentes, marco teórico, desarrollo del tema y el último capítulo son las conclusiones, recomendaciones y trabajos

futuros. La introducción es a la que en este momento se le está dando lectura, a continuación se describen de manera general los cuatro capítulos restantes y los anexos.

Capítulo 2. Se describen los estudios que se han realizado en el campo de la restauración digital, mostrando que han sido diversas disciplinas las que han utilizado técnicas que forman parte de este campo de investigación. Se presentan los objetivos que se alcanzaron bajo ciertas limitaciones y los alcances que tiene la tesis.

Capítulo 3. Contiene el marco teórico, que es la base para la realización del proyecto. La investigación de los conceptos básicos es fundamental, permite conocer a detalle el tema del procesamiento digital del cual forma parte la restauración digital de imágenes, ofreciendo una definición clara de los términos que se utilizarán a lo largo del desarrollo de este documento de tesis.

Capítulo 4. Describe las técnicas empleadas para darle solución a la restauración de imágenes digitales con degradación por borrosidad que presentan algunas imágenes. También, se explica cómo funcionan algunos de los métodos más utilizados que son el filtro de Wiener, el criterio de minimización de rizado y el filtro inverso. Por último, se detallan las pruebas realizadas junto a los resultados obtenidos.

Capítulo 5. Se resumen las conclusiones a las que se llegaron a lo largo del proyecto y se señalan posibles líneas de trabajos futuros que podrían extender los resultados obtenidos en el presente trabajo.

Anexo A. Contiene una descripción sobre el uso de la aplicación creada para realizar el estudio sobre la restauración digital de imágenes.

Anexo B. Se muestra el código utilizado para crear la aplicación, escrito en el lenguaje de Matlab.

Anexo C. Se da una breve descripción del contenido del CD.

Después de haber descrito la panorámica general del tema de restauración digital de imágenes, se procederá a dar a conocer la información necesaria para comprender este trabajo de tesis a partir de los estudios previos que se han realizado.

CAPÍTULO 2. ANTECEDENTES

En este capítulo se describen los trabajos que se han realizado acerca del tema de la restauración de imágenes digitales, esto consiste en ubicar y comprender el trabajo de tesis tomando en cuenta estudios similares que se han hecho anteriormente, en los que se podrá verificar que la restauración digital es un tema primordial en diversas disciplinas científicas. Se plantea la forma en que se han realizado y cuál ha sido la solución.

2.1. Estado del arte y trabajos relacionados

Algunos estudios realizados sobre la restauración digital de imágenes han sido de gran utilidad para la obtención de imágenes de mejor calidad en diferentes áreas del conocimiento e investigación. En este trabajo de tesis se realiza un análisis sobre algoritmos de restauración aplicados a imágenes que contengan degradación por borrosidad.

El estudio de restauración de imágenes en astronomía empleando una perspectiva bayesiana, desarrollado por Molina *et al.* (2001), fue realizado para darle solución al problema de degradación que presentaban las imágenes tomadas por el telescopio Hubble antes que se detectara el error en el espejo principal. El problema del espejo se debía a la aberración esférica que era causada por una curvatura errónea del mismo, producida por un pulido realizado por un instrumento defectuoso, lo que producía imágenes sometidas a un fenómeno de borrosidad. El hecho de que cada minuto de observación costara cien mil dólares, dio pauta para que se proporcionara más impulso a la investigación sobre la restauración de las imágenes digitales.

El objetivo principal de este estudio fue presentar los métodos más utilizados para restaurar imágenes en astronomía óptica, en especial el uso de métodos bayesianos sobre las imágenes. Los métodos bayesianos se distinguen de los otros porque incluyen información espacial específica acerca de los objetos presentes en la imagen, el teorema de Bayes se utiliza para conocer la probabilidad *a posteriori* de cierta variable de interés dado un conjunto de hallazgos, en otras palabras la estimación bayesiana debe indicar las imágenes que tienen mayor probabilidad de parecerse o ser la imagen original, de todo el conjunto de imágenes posible que se ha creado al someter la imagen degradada a cambios (aplicación de técnicas de suavizado, mejora de imagen, etc.), lo que se busca es obtener una función que permita estimar o aproximar, la solución que se está buscando. Las imágenes enviadas por el telescopio antes de que fuera solucionado el problema de su espejo principal, sirvieron como una excelente recopilación de imágenes para realizar pruebas de este método.

Un ejemplo de la aplicación de los métodos bayesianos, es la utilizada en una imagen del planeta Saturno tomada desde la Tierra, la cual se restauró mediante el método de Bayes. En la figura 2.1 se muestran los resultados obtenidos.

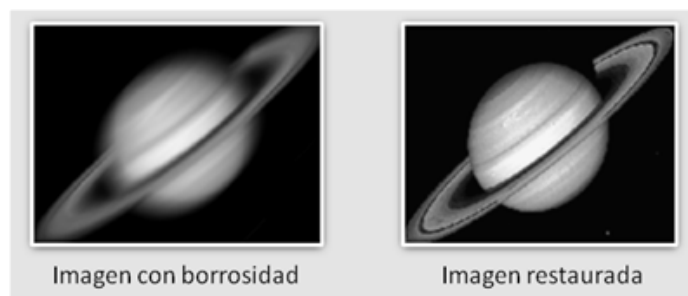


Figura 2.1. Restauración de la imagen del planeta Saturno.

El hacer este estudio, demostró que no sólo es necesario restaurar las imágenes tomadas desde la tierra, sino también las que han sido enviadas desde el espacio.

Muchas de las aplicaciones astronómicas involucran el estudio de la restauración digital de imágenes. Esta área sirve como punto de partida y apoyo a proyectos que requieren conocer del tema, es por ello que en este trabajo fue necesario conocer los inicios de la restauración digital para tener un panorama más amplio de la clasificación de las técnicas además de los problemas relacionados con ella.

Otro de los estudios realizados en la restauración digital de imágenes astronómicas es el realizado por Banham & Katsaggelos (1997), donde mencionan que las imágenes que se obtuvieron de las diversas misiones espaciales (*Ranger*, *Lunar Orbiter* y *Mariner*) estaban sujetas a diversas degradaciones, como las causadas por las vibraciones de los equipos o la incapacidad de los astronautas para mantenerse quietos mientras tomaban las fotografías en un ambiente sin gravedad. La pérdida de información debida a la mala calidad de las imágenes generó problemas económicos; se ha estimado que las 22 imágenes tomadas durante el vuelo de *Mariner IV* a Marte en 1964, costaron diez millones de dólares.

Para darle solución a este problema se tuvo que combinar la aproximación basada en el campo del procesamiento de la señal (la aproximación de los ingenieros) y la aproximación basada en modelos estadísticos, en particular en los campos aleatorios de Markov, la llamada aproximación bayesiana o estadística. El estudio mostró que los dos son perfiles de un mismo problema, cuya visualización conjunta proporciona una imagen más nítida y con mejor calidad. Los campos aleatorios proporcionan medidas de probabilidad sobre un dominio de definición que tenga relaciones de tipo espacial o temporal, para este caso se estudiaron las relaciones de tipo espacial, esto es porque para poder determinar una función de estimación apropiada, que permita obtener la imagen restaurada llamada f en el modelo de degradación, es necesario que a la imagen degradada denominada g se le someta a cambios, estos cambios van a generar diversas imágenes, es necesario que sólo se consideren aquellas imágenes que tengan mayor probabilidad de ser f . De manera que cuando se realice un cambio de un valor de un punto f , no sólo se toma en cuenta el valor en sí, sino que se consideran los valores de sus vecinos.

Este trabajo ayudó a que muchas personas se interesaran en estudiar, mejorar y crear algoritmos para la restauración digital de imágenes (Banham & Katsaggelos 1997). El estudio realizado hace uso de ciertos métodos; al investigar sobre ellos se encontró que el filtro inverso es de gran utilidad para este trabajo de tesis en el que se requieren estudiar algoritmos de restauración, porque es aplicable a los casos donde existe ruido por movimiento del sistema de captación, la única desventaja es que el ruido debe ser muy pequeño.

En el departamento de ingeniería informática, de la Universidad de Santiago de Chile, se realizó un trabajo de restauración de imágenes usando el criterio de minimización de rizado (Mery & López 2003), donde el problema consistió en encontrar la fila o columna de la imagen original no degradada a partir de la matriz de degradación y la fila o columna de la imagen degradada.

Por esta razón se presentó una metodología para restaurar imágenes degradadas por un movimiento horizontal puro, lo que significa que se utiliza para una degradación correspondiente a un movimiento lineal uniforme de n píxeles.

En la figura 2.2 se muestra un ejemplo que ilustra este proceso de degradación, para una imagen (fig. 2.2a) donde n es equivalente a 10 píxeles (fig. 2.2b), 20 píxeles (fig. 2.2c) y 30 píxeles (fig. 2.2d).

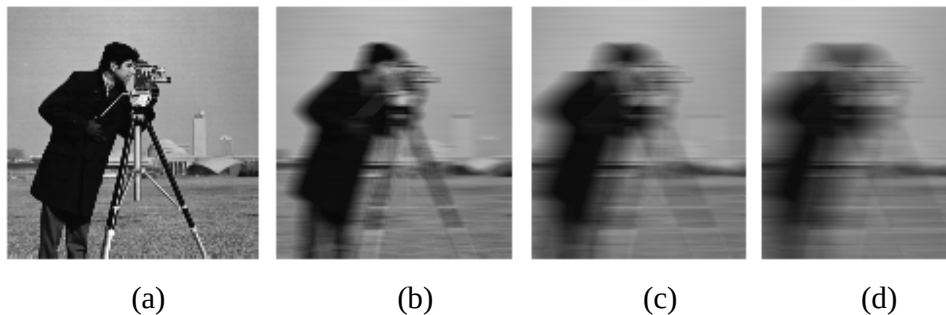


Figura 2.2. Proceso de degradación de una imagen. a) imagen original. Imágenes degradadas por medio de un movimiento horizontal uniforme: b) 10 píxeles, c) 20 píxeles y d) 30 píxeles.

Debido a que se observó que un conjunto de soluciones no deseadas presentan grandes oscilaciones, se planteó darle solución con el proceso de restauración aplicando una minimización de rizado, por lo que se realizó una evaluación con distintos criterios de minimización de rizado según costo computacional y desempeño en la restauración de

imágenes con y sin ruido. El criterio de minimización de rizado consiste en disminuir las oscilaciones debido a la degradación y porque existen muchas soluciones en las cuales se debe tomar en cuenta la que tenga menores picos en su respuesta.

Se efectuó una etapa de pruebas con 6 imágenes en blanco y negro, todas con dimensiones de 256 x 256 píxeles, y se simularon degradaciones con 6 movimientos horizontales y 6 niveles de ruido gaussiano, realizando pruebas con y sin discretización de la imagen degradada, cada una de las imágenes fue restaurada usando 10 métodos distintos (transformada discreta de Fourier, transformada discreta de cosenos, método de Laplace, filtro laplaciano de Gauss, filtro paso alto, método de la inversa, etc.) que fueron evaluados según desempeño y tiempo de ejecución. Los métodos fueron implementados en una computadora Intel Pentium IV a 1500-MHZ de velocidad, el software utilizado fue Matlab versión 6.1.

Los resultados mostraron que la solución basada en una optimización directa presenta una buena alternativa debido a un tiempo de ejecución bajo. En la figura 2.3 se muestran las imágenes originales empleadas, se aprecia también los resultados de aplicar el filtro pseudo-inversa que es un método de restauración aplicado a ruido gaussiano.

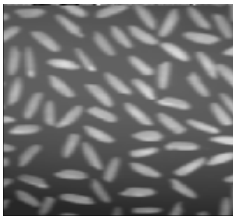
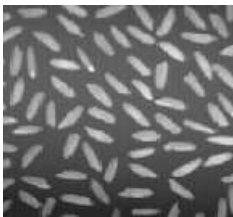
n	Imagen original	Imagen degradada	Imagen restaurada
3 píxeles			
10 píxeles			
20 píxeles			

Figura 2.3. Resultados de la restauración en 6 imágenes de prueba.



Figura 2.3. Resultados de la restauración en 6 imágenes de prueba (continuación).

Para realizar la evaluación del desempeño de los métodos de restauración se utilizó la fórmula del mejoramiento de la relación señal/ruido (ISNR, por sus siglas en inglés), la cual se representa por la ecuación 1.

$$ISNR = 20 \log_{10} \left(\frac{\|F - G\|}{\|F - F'\|} \right) = 20 \log_{10} \left(\frac{\sum_{i=1}^L \sum_{j=1}^N (F_{ij} - G_{ij})^2}{\sum_{i=1}^L \sum_{j=1}^N (F_{ij} - F'_{ij})^2} \right), \quad (1)$$

donde F representa a la imagen original, G la imagen degradada, la imagen restaurada será denotada por F' . Si el ISNR es grande significa que la restauración es buena, y si tiende a infinito quiere decir que F y F' son iguales. Este criterio de medición sólo puede ser aplicado a problemas de restauración en los que la imagen degradada ha sido simulada pues se necesita contar con la imagen original. En la imagen se muestran dos gráficas, la figura 2.4a representa la medición del ISNR en cada uno de los métodos, la figura 2.4b es la desempeño en cuanto a tiempo de ejecución, en ambas evaluaciones el mejor desempeño fue obtenido por el método de la pseudo-inversa, aunque a simple vista en la gráfica es difícil determinar cuál de ellos es el mejor.

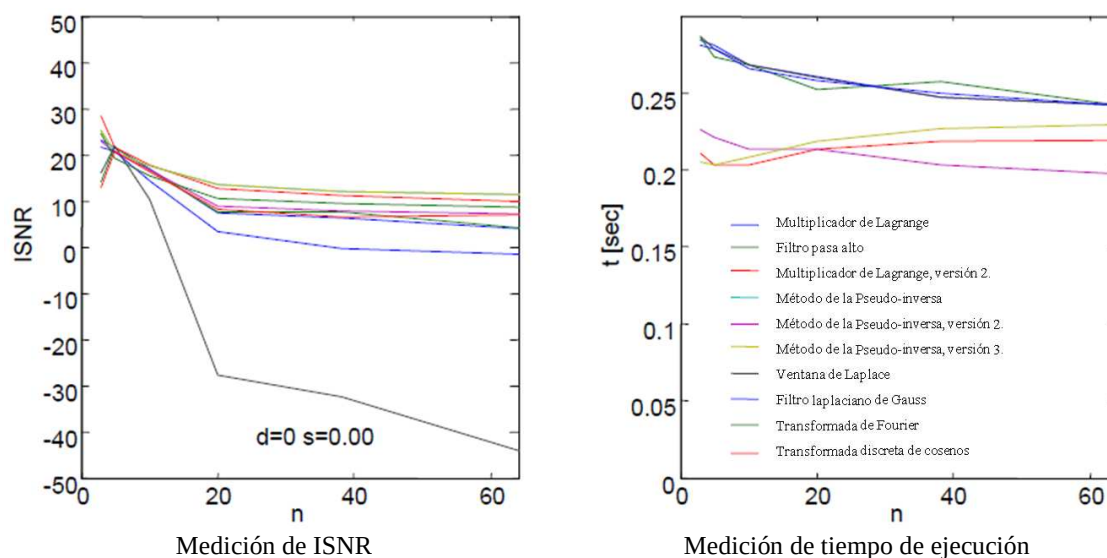


Figura 2.4. Resultados del desempeño de los métodos evaluados.

En este trabajo se utilizó el método de minimización de rizado para la restauración de imágenes con muy buenos resultados y, como ya se ha hecho mención, este método es utilizado en esta tesis, confirmando que los resultados obtenidos después de llevar a cabo el proceso de la restauración digital de imágenes son satisfactorios.

En el Departamento de Ciencia Forense de la Universidad de Policía Central en Taiwán se realizaron estudios sobre la restauración digital en imágenes forenses (Wen 2002), ya que en el campo de la ciencia forense hay muchos datos en imágenes penales, incluidas las impresiones dactilares, fotos de escena, cámaras de vigilancia y cintas de video. Las pruebas pueden convertirse en el punto clave para resolver crímenes. Sin embargo, las pruebas pueden estar contaminadas por el ruido (borrosidad en fotografías debido al movimiento de objetos como: automóviles, delincuentes, entre otros) de las escenas de los crímenes, y llegar a estar distorsionadas o deformadas. En este caso, las pruebas no podrán utilizarse y se puede perder la oportunidad de detener a los delincuentes. El objetivo principal de las técnicas de restauración es mejorar la calidad de una imagen degradada. Los fenómenos de degradación se pueden presentar por movimiento de la imagen, la turbulencia atmosférica, borrosidad, fuera de foco y ruidos electrónicos borrosos.

En el proceso de restauración de imágenes se usa la función de la extensión del punto (PSF) para modelar el tipo de ruido que presenta la imagen, sin embargo en las

imágenes forenses el obtener un PSF correcto hace difícil el proceso de la restauración de la imagen, en este trabajo se revisan varias PSF a las cuales se les busca una solución, de los resultados experimentales se podrá obtener que imágenes son mejoradas con que PSF y los filtros utilizados.

Se hicieron dos etapas de prueba, la primera consintió en hacer pruebas con una sola imagen, se agregó ruido a la imagen, incluyendo el ruido gaussiano, uniforme, por turbulencia atmosférica y por movimiento.

Los métodos propuestos que se han implementado para restaurar las imágenes son el filtro inverso y el filtro de Wiener. El filtro inverso es muy eficiente si se sabe el modelo que causa la degradación.

Posteriormente se aplicaron los métodos a un caso real para resolver el caso de la obtención de placas de un vehículo en movimiento, obteniendo la imagen restaurada de la imagen con ruido, como lo muestra la figura 2.5.



Figura 2.5. Restauración de la imagen de las placas de un vehículo en movimiento.

Esta investigación es de gran ayuda ya que los resultados que se obtuvieron al aplicar el algoritmo de filtro de Wiener fueron exitosos, tomando en cuenta de que si no existiera la imagen original es posible percibir el contenido de la imagen y serviría para el propósito por el cual fue tomada, que es la identificación de placas de un vehículo, por lo que es necesario evaluar con otros algoritmos las mismas imágenes. La aplicación desarrollada en esta tesis hace uso de este algoritmo de restauración y los resultados que se obtuvieron fueron satisfactorios.

En la Universidad Tecnológica de la Mixteca se realizó la tesis “Técnica multigríd en la solución del problema de restauración digital” (Núñez 2007), donde se planteó el problema que debido al crecimiento de equipos digitales nuevos y más potentes, los cuales permiten capturar imágenes de alta resolución, trae consigo que las imágenes sean más grandes y que los algoritmos que en un principio funcionaban sin dificultad con ellas, ahora se consideran lentos. Por ello es necesario desarrollar otros algoritmos con distintos

enfoques, que permitan seguir con el procesamiento digital de imágenes de forma fluida y sin complicaciones. Aún cuando este problema ya ha sido solucionado por diversos métodos, como lo son Gauss Seidel, gradiente conjugado y Jacobi entre otros, resulta que tienen como principal desventaja el ser muy lentos. Estos métodos buscan resolver el sistema de ecuaciones mostrado en (Ec. 2), los procesos son iterativos y se realizan hasta que convergen,

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & & A_{2n} \\ \vdots & & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}, \quad (2)$$

donde A es una matriz cuadrada, b es un vector conocido y x es un vector desconocido. Las variables que se representan con estos vectores son: A imagen degradada, x vector de la imagen degradada, b es el vector de la imagen restaurada.

De ahí que este trabajo se hizo con el objetivo de aplicar la técnica multigrid combinando estimación bayesiana y campos aleatorios markovianos, para hacer la reconstrucción de una imagen que ha sido degradada, con la finalidad de realizar la restauración digital más rápida.

Al aplicar la técnica multigrid se observó que superó a los algoritmos Gauss Seidel, Jacobi y Gradiente, tanto en la evaluación de eficiencia como en el costo computacional. La técnica multigrid sólo trabaja con matrices cuadradas que sean múltiplos de dos, lo que hace que sea deficiente respecto a otros algoritmos que sí trabajan con imágenes de cualquier dimensión. Se realizaron varias pruebas con distintas imágenes a escala de grises, utilizando el software llamado “ImageRestore” que fue desarrollado por el autor del trabajo antes mencionado, lo que le permitió comparar los resultados de aplicar los algoritmos multigrid, Jacobi, Gauss Seidel y Gradiente a cada una de ellas. A las imágenes se añadió el 20% de ruido uniforme, que es más difícil de quitar que el de “sal y pimienta”, mediante el programa de edición de imágenes Photoshop.

Los resultados presentados en la figura 2.6 son para una imagen de 512 x 512 píxeles, se puede observar claramente que la mejor solución en cada algoritmo tarda más en comparación con la técnica multigrid.

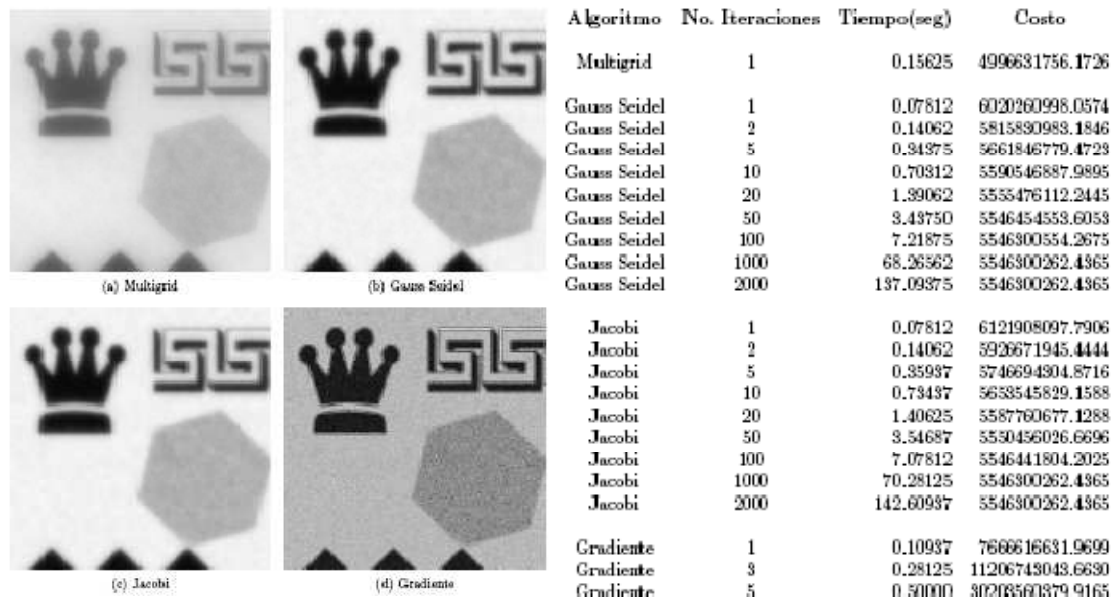


Figura 2.6. Resultados de la restauración de una imagen utilizando los métodos: Multigrid, Gauss Seidel, Jacobi y Gradiente.

Al utilizar la técnica multigrid para la solución del problema de restauración digital se realizaron diversas comparaciones con los algoritmos Gauss Seidel, Jacobi y Gradiente, por lo que se encuentra cierta similitud con la investigación propuesta en el presente documento, sólo que en este caso se investigaron y aplicaron otros algoritmos de procesamiento digital de imágenes, los cuales son el filtro de Wiener, el filtro inverso y el criterio de minimización de rizado.

2.2. Justificación

En la actualidad, los equipos de software y hardware para el tratamiento de imágenes son más accesibles (Domínguez 1996), lo que ha permitido que en los últimos años se pueda aplicar la restauración de imágenes digitales en diversas disciplinas, tales como: astronomía, medicina, ingeniería, criminología, biología, sistemas de información geográfica, microscopía, etc. El procesamiento digital de imágenes brinda las posibilidad de hacer investigación en estas áreas sin necesidad de hacer grandes inversiones en equipos muy sofisticados, siendo el procesamiento digital de imágenes una de las más explotadas recientemente.

Muchas veces, el repetir un proceso para la adquisición de una imagen involucra una cantidad grande de procesos que pueden ser costosos, tanto económica como computacionalmente; o que puede ser difícil de que ocurran otra vez, por lo que es necesario tener mecanismos para aprovechar más la imagen y obtener una mayor y mejor calidad en la información.

Mediante la restauración de imágenes por computadora es posible obtener información relevante de la escena captada por una cámara. La restauración de imágenes sirve para mejorar una imagen con fines interpretativos, al aplicar la restauración se podrá extraer más información para una toma de decisiones.

Con el fin de disminuir o eliminar los defectos por borrosidad que produce el tomar imágenes en movimiento, surgió la necesidad de crear una aplicación que permitió la obtención de imágenes de mayor calidad, donde se utilizaron diversos algoritmos de procesamiento digital de imágenes que fueron implementados en computadora, los cuales buscan cómo obtener la imagen original a partir de la imagen observada, con el objetivo de extraer propiedades y características que ayuden a ampliar el análisis que esté realizando la persona.

2.3. Planteamiento del problema

La imagen captada por un dispositivo electrónico, óptico o electro-óptico puede presentar degradaciones producidas por ruidos en el sensor, borrosidad debida, por ejemplo, a un mal enfoque de la cámara, al movimiento de la cámara o a turbulencias atmosféricas (Pérez 2005). Hay que considerar que aunque se pudiera volver a realizar el proceso de captura de la imagen en repetidas ocasiones, no siempre se obtendrá la misma imagen con las mismas características. La degradación en una imagen va influir en la confiabilidad de la imagen en la investigación científica, debido a que ayuda al descubrimiento de nuevos sucesos, como lo es el descubrimiento de nuevos cuerpos celestes, la toma de decisiones, como encontrar al o los culpables de un delito en la ciencia forense, detección de objetos captados por un robot móvil, etc.

La metodología a seguir y con la cual se buscó dar solución a este problema consta de los siguientes pasos:

- Dominio del problema.
- Adquisición de imágenes.
- Aplicación de ruido a las imágenes.
- Implementación de los algoritmos de restauración utilizando Matlab.
- Realizar pruebas y mejoras con los algoritmos.
- Evaluación y comparación de los resultados.

2.4. Objetivos

Objetivo general

Realizar y evaluar la restauración de imágenes con degradación por borrosidad, con la finalidad de obtener imágenes que contengan información de mayor calidad, empleando técnicas de procesamiento digital de imágenes.

Objetivos específicos

- Buscar y estudiar diferentes técnicas y algoritmos para la restauración digital de imágenes con degradación por borrosidad.
- Implementar algoritmos para restaurar imágenes degradadas con ruido por borrosidad.
- Analizar y evaluar los resultados obtenidos en la implementación de los algoritmos que permiten la restauración digital de imágenes con degradación por borrosidad.

2.5. Alcances y límites del estudio

Este proyecto de tesis consiste en la restauración de imágenes con degradación por borrosidad, lo que indica que sólo es aplicable en los casos donde existe borrosidad, aunque para este trabajo solo se abarca la degradación ocasionada por movimiento lineal uniforme.

Alcances

El realizar un estudio de la restauración digital de imágenes mediante el desarrollo de una aplicación ha ayudado a obtener los siguientes alcances:

- Los algoritmos utilizados son filtro de Wiener, criterio de minimización de rizado y filtro inverso.
- El estudio ha permitido comprobar que para los casos elegidos en esta tesis, el filtro Wiener es más eficiente a los demás filtros.
- La aplicación del filtro de la mediana ha ayudado a reducir el porcentaje de error, en los casos donde se utiliza el filtro inverso y el criterio de minimización de rizado.
- La aplicación permite guardar el resultado de la imagen después de haber realizado el proceso la restauración.
- Se pueden comparar los resultados gráficamente y generar a partir de ellos ciertas conclusiones.

Limitaciones

Los límites que tiene el estudio realizado son:

- Los resultados son aplicables a imágenes en escala de grises.
- Cuando no se conocen los parámetros de degradación o distancia y son calculados por la aplicación, los resultados son aproximados.
- El criterio de minimización de rizado es aplicable a imágenes cuya degradación sea por movimiento horizontal o vertical puro.
- La degradación de las imágenes debe ser invariante; esta limitante es propia de cada uno de los algoritmos de restauración.

Estas limitaciones pueden ser resueltas de la siguiente manera:

- Creación de resultados para cada componente de color RGB y su posterior integración en una sola imagen, esto sería utilizando el código de matlab que se tiene para imágenes en escala de grises, solo que ahora habría que generarlo para cada componente.

- Investigación y creación de algoritmos que realicen el cálculo de los parámetros de ruido y distancia más exactos.
- Estudio e implementación de nuevos algoritmos que permitan manejar el parámetro del ángulo de borrosidad.
- Debido a que la degradación debe ser invariante en cada algoritmo: Wiener, inverso y minimización de rizado, se deben crear o investigar algoritmos que permitan manejar degradaciones variantes no uniformes.

A medida que transcurre el tiempo se puede ir innovando en la creación de algoritmos que permitan realizar una mejor restauración en imágenes y donde sean utilizados procesos menos complejos.

Una vez establecida la información necesaria de los estudios sobre de la restauración de imágenes digitales realizados por diversos investigadores, se prosigue a detallar los conceptos teóricos acerca del tema, que sirven como fundamento para abordar el problema planteado en este trabajo de tesis.

CAPÍTULO 3. MARCO TEÓRICO

En este capítulo se presentan las bases teóricas necesarias para el buen entendimiento de los conceptos y técnicas relacionadas con la restauración digital de imágenes presentados en este trabajo de tesis.

3.1. Procesamiento digital de imágenes

El procesamiento digital de imágenes (PDI) es un área que cuenta con diversas técnicas y procesos que permiten la manipulación y análisis de imágenes a través de una computadora.

El PDI hace transformaciones de imágenes a otras imágenes, con el objetivo de hacer más fácil la recuperación de información para ayudar en la interpretación humana (Escalera 2001).

La evolución histórica sobre los trabajos relacionados al estudio sobre imágenes parte desde el sistema conocido como “Bartlane”, a principios de la década de 1920. El nombre del sistema surge en honor al nombre de sus inventores Harry Guy Bartholomew y

Robert McFarlane. El sistema contenía una de las primeras técnicas del tratamiento de imágenes, que consistía en el mejoramiento de fotografías periodísticas que eran enviadas por un cable submarino entre Londres y Nueva York. Mediante el cable se hacían transmisiones de impulsos eléctricos codificados en 5 niveles de brillo. En 1929 ya se trabajaba con 15 niveles de brillo distintos (Esqueda & Palafox 2005). Los problemas para mejorar la calidad visual, que comúnmente se presentaban en esa época, estaban relacionados con la selección de procedimientos de impresión y la distribución de los niveles de brillo (González & Woods 1992).

Con el paso del tiempo continuaron las investigaciones, las cuales tuvieron un gran crecimiento durante la década de los 60's por el nacimiento del procesamiento digital de imágenes, esto se debió a los adelantos en cuanto a hardware y lenguajes de programación, motivo por el cual se consolidó como disciplina científica. Los primeros estudios que se realizaron fueron a imágenes provenientes del espacio durante las misiones *Ranger* y *Mariner* en el año de 1964, posteriormente tuvo aplicaciones en la medicina y otras áreas científicas.

El procesamiento digital de imágenes actualmente es la base de una creciente variedad de áreas de investigación que incluyen diagnóstico médica, percepción remota, exploración espacial, visión por computadora, microscopía, entre otras. Aunque pareciera que no existe una relación entre estas ciencias y el tratamiento de imágenes, los problemas que se presentan con imágenes normalmente requieren de algoritmos capaces de identificar ciertas características que resultan importantes para la interpretación y el análisis humano.

En el modelo presentado en la figura 3.1, se muestran las etapas fundamentales que se llevan a cabo durante el procesamiento digital de imágenes.

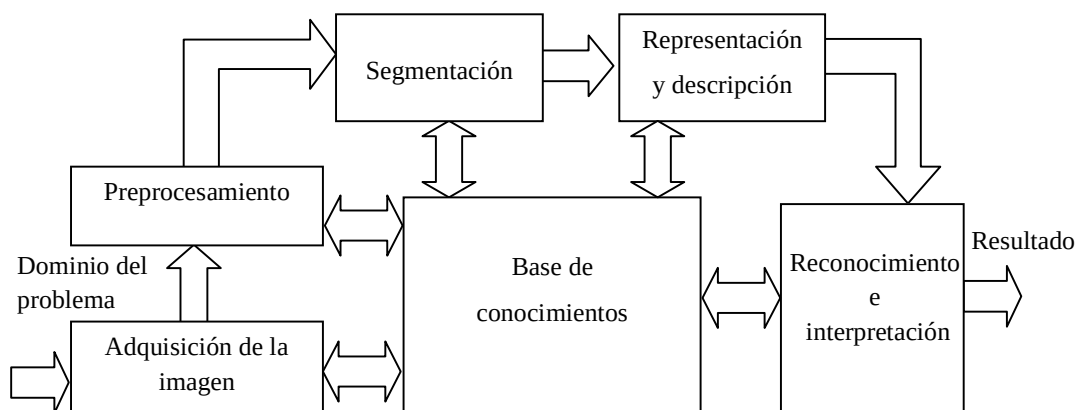


Figura 3.1. Etapas fundamentales del procesamiento digital de imágenes (González & Woods 1992).

A continuación se hará una breve descripción de cada uno de ellas.

Adquisición de la imagen: es la primera etapa del procesamiento de imágenes. Es una etapa fundamental, es la encargada de todo el proceso de formación y captación de imágenes que son el elemento primordial del procesamiento digital de imágenes. Consiste en adquirir una imagen para digitalizarla, es decir, capturar los datos visuales mediante sensores. Existen diferentes medios para obtener una imagen digital, los más comunes son el escáner y la cámara digital. Estas herramientas se basan en un dispositivo llamado sensor CCD (*Charge - Coupled Device*, dispositivo de cargas interconectadas), que está formado por receptores fotosensibles llamados fotodiodos; el CCD recibe la luz de la imagen y mediante el proceso de reflexión o transmisión, integra en un tiempo definido la cantidad de luz que llega a él, formando así un arreglo de CCD's, los cuales fueron inventados y patentados por los laboratorios Bell en el año de 1969 para un programa espacial de la NASA (Russ 2006). Para que la computadora pueda leer los datos de la imagen, los CCD necesitan de un convertidor analógico digital (ADC), el cual se encarga de convertir los datos de cada pixel en datos digitales binarios (Somolinos 2002).

La operación básica de un CCD es ilustrada en la figura 3.2. Los fotodiodos están acomodados por columnas, así que por cada columna se transfiere la cantidad de luz que acumula cada fotodiodo a una fila de lectura, para formar un segmento que corresponde a un pixel.

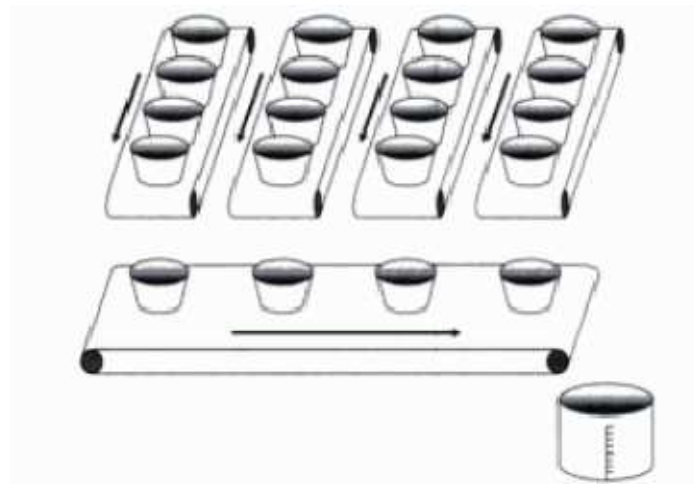


Figura 3.2. Operación básica de un dispositivo CCD (Russ 2006).

Preprocesamiento de imágenes: en este paso se utilizan diversas técnicas para mejorar la imagen, como son realzar el contraste o restaurar la imagen. En ocasiones,

durante el proceso de adquisición de la imagen, se puede incurrir en la formación de cierta degradación debido a la contaminación por ruido o por alguna acción no deseada por la cámara o el objeto captado. Esta degradación provoca que el contenido de la información de la imagen sea alterada, influyendo en la calidad de la imagen, por lo que es necesario aplicar técnicas que permitan el mejoramiento de la imagen, así como la reducción de ruido. Además permite mejorar ciertas propiedades, como color, contraste y estructura.

El trabajo de tesis que se está presentando abarca las dos etapas del procesamiento digital de imágenes descritas anteriormente, siendo en esta última donde se contempla el tema de restauración digital de imágenes, sin embargo es importante mostrar la descripción de las etapas restantes.

Segmentación: se utiliza para crear partes de una imagen, esto es, extraer formas especiales como objetos o palabras. El objetivo de la aplicación de la segmentación es determinar las estructuras o subestructuras que corresponden a la imagen de un objeto de interés. La segmentación es muy útil para las etapas de descripción, reconocimiento e interpretación. Es considerada como un proceso de alto nivel ya que es una tarea que implica un grado más avanzado de procesamiento. En procesamiento de imágenes los métodos de segmentación se basan en dos corrientes.

La primera consiste en métodos basados en discontinuidades que utilizan algoritmos de detección de bordes, la segunda está basada en similitud de la información para realizar agrupamientos que formen los objetos, como son algoritmos de umbrales y crecimiento de regiones (Chacón & Molina 2007).

Representación y descripción: la representación de una imagen implica dos posibilidades: la primera consiste en tomar en cuenta las características externas, el objetivo principal se centra en los rasgos de forma, los bordes son un claro ejemplo de ello, la segunda es en términos de las características internas, esto es eligiendo los píxeles que comprenden la región (González & Woods 1992). La descripción o selección trata de extraer rasgos básicos para diferenciar una clase de objetos con otros, dentro de la zona de la representación elegida; se hace con la finalidad de asignar etiquetas a los objetos, por ejemplo, tamaño y forma. Una característica es un atributo usado para tomar decisiones respecto a objetos en la imagen. Algunos atributos son naturales y se definen mediante la

apariciencia de manera visual en la imagen, en tanto otros son el resultado de operaciones realizadas a la imagen (Russ 2006).

Reconocimiento e interpretación: el reconocimiento es el proceso para identificar un objeto basándose en la información que proveen sus descriptores. La interpretación es la asignación de un significado a un objeto reconocido. Por ejemplo, determinar mediante el color si una fruta está madura o no.

El procesamiento digital de imágenes generalmente tiene dos tipos de aplicaciones, según Chacón & Molina (2007).

- a) Mejorar una imagen, la cual será posteriormente interpretada por una persona dependiendo el área de estudio. Mejorar no siempre significa generar una imagen mejor para la apreciación de una persona, sino es generar una imagen que sea más apropiada para resolver el problema con el cual se trabaja.
- b) Procesamiento de una imagen para realizar análisis de escena por medio de una máquina, es decir, percepción autónoma.

3.2. Formación de una imagen en el sistema de visión humano

La adquisición de una imagen está a cargo de algún transductor o conjunto de transductores que, mediante la manipulación de la luz o alguna otra forma de radiación que es emitida o reflejada por los cuerpos, logran formar una representación del objeto dando lugar a la imagen. Los transductores registran la radiación que ha interactuado con ciertos objetos físicos. Ejemplos de transductores son: el ojo humano, sensores de cámara fotográfica o de vídeo, tomógrafos, etc.

La visión en los humanos permite interpretar y percibir el mundo que lo rodea. El ojo del ser humano es el órgano capaz de informar al cerebro respecto de la luz que lo ilumina para que realice el proceso de la visión. Como lo muestra la figura 3.3, para la formación de una imagen en el ojo humano, existe la participación de diferentes elementos que realizan diversos procesos.

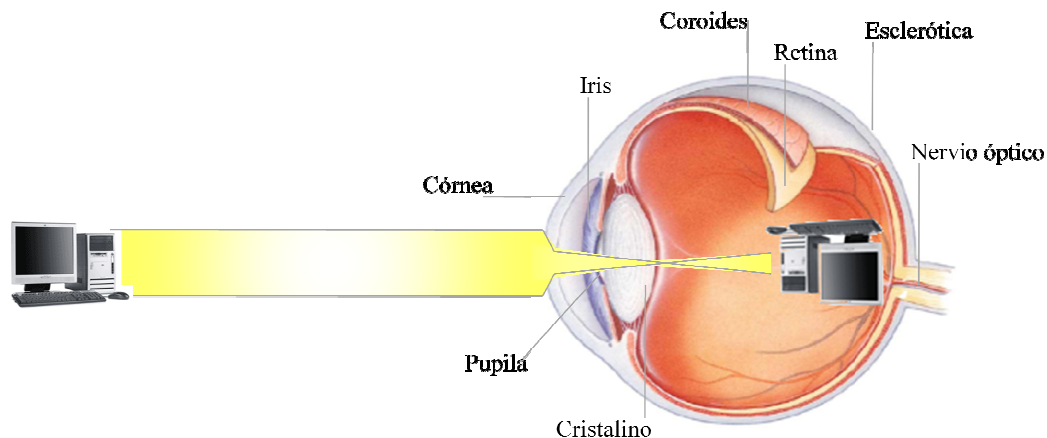


Figura 3.3. Formación de la imagen en el ojo humano.

La luz penetra en el ojo a través de una abertura llamada pupila, la cual se encuentra rodeada por el iris quien se encarga de abrirla o cerrarla de forma gradual para controlar la cantidad de luz que recibe el ojo, el cual se contrae por un arco reflejo si hay mucha luz, o se dilata si hay poca (Pratt 2007).

La luz atraviesa el cristalino e incide en la retina situada en el fondo del ojo. Las células fotosensibles de la retina generan señales eléctricas que recorren el nervio óptico y llegan al cerebro. Ajenos al cristalino están los músculos ciliares, que modifican su curvatura acomodándolo para la visión próxima o lejana, de manera que los rayos procedentes de los distintos puntos del objeto converjan en único punto de la retina. Cuando se mira un objeto, los rayos luminosos reflejados por él, penetran en el ojo a través del cristalino y éste modifica su curvatura. Los rayos que emergen de la parte superior del objeto se proyectan en el interior de la retina y forman una imagen invertida. Los seres humanos no ven imágenes invertidas, esto sucede porque no miran la imagen sobre su propia retina, el cerebro es quien se encarga de interpretar la orientación correcta.

La retina se integra de dos tipos de células sensibles de la luz: Los conos, encargados de percibir los colores. Algunos son más sensibles al rojo, verde o azul ya que mediante la combinación de estos colores llamados primarios se puede percibir toda la gama de colores. Los conos son aproximadamente entre 6 y 7 millones, se localizan en la parte central de la retina, llamada fovea y reciben este nombre por su forma. Los bastones actúan cuando no hay mucha luz ya que son sensibles a la intensidad, permiten ver cuando hay muy poca luz,

se calcula que existen de 75 a 150 millones distribuidos sobre la superficie de la retina (Chacón & Molina 2007).

En ambos casos la excitación luminosa provoca una descomposición de sustancias fotosensibles que componen la púrpura retiniana, actuando sobre los conos y bastones, los cuales transmiten la sensación a través de las diferentes capas de la retina y del nervio óptico, hasta el cerebro.

La analogía entre el ojo humano y la cámara fotográfica es muy coherente con la realidad, puede ser vista en la figura 3.4.

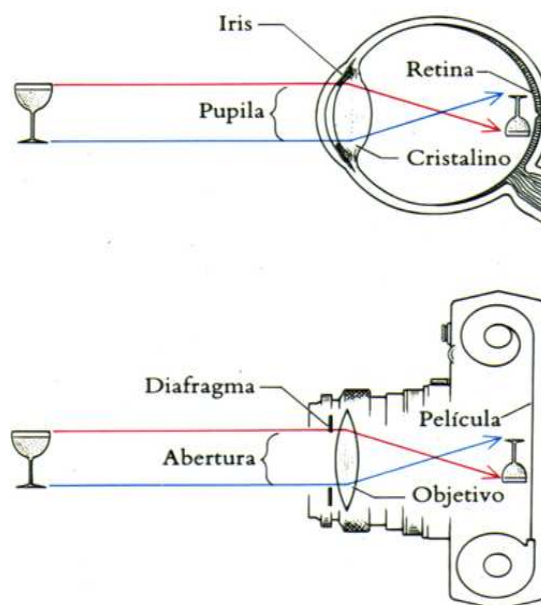


Figura 3.4. Analogía entre el ojo humano y la cámara fotográfica (Domínguez 1996).

El funcionamiento de una cámara fotográfica ha sido comparado con el de un ojo humano, aunque sólo logra cumplir con las funciones básicas que éste realiza. El cristalino que es equivalente a la lente de la cámara, se encuentra en una posición fija, no se mueve hacia adelante o atrás como en la cámara. El diafragma equiparado con la pupila funciona en un rango menor de intensidades de iluminación que el ojo.

3.3. Pixel

Un pixel (px) es el elemento más esencial en una imagen digital, es acrónimo del inglés *picture element* que significa “elemento de la imagen”. Se refiere al elemento más pequeño

que compone a una imagen y corresponde a las unidades de brillo y color de la imagen. Los pixeles se representan como pequeños cuadrados o rectángulos. La imagen completa es la unión de un conjunto de pixeles (González & Woods 2008). En la figura 3.5 se muestra el ejemplo de cómo se representa un pixel en una imagen.

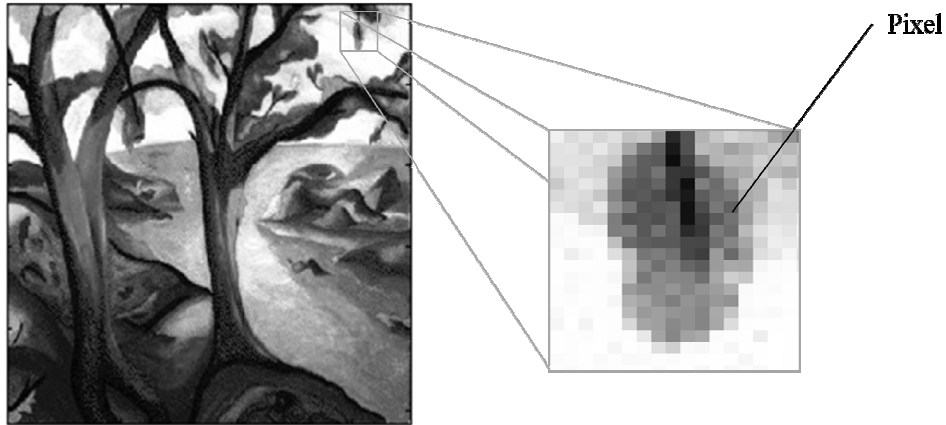


Figura 3.5. Representación de un pixel (Pérez 2005).

El número de niveles de gris y las dimensiones de la matriz (filas x columnas) condicionan la capacidad de resolución de la imagen digital. La resolución se define como el número de pixeles que se muestran por una unidad de medida.

La calidad de una imagen también depende de la resolución que tenga el dispositivo que la capta. El número de pixeles que contenga una imagen obedece a cuántos pixeles utilice el sensor CCD de la cámara para captar la imagen.

3.4. Imagen digital

Cuando una imagen se representa como una matriz de valores, que representan la intensidad o brillantez del objeto, forma lo que se conoce como imagen digital, que es la estructura con la cual procesa las imágenes una computadora. Las imágenes digitales se obtienen al tomar fotos electrónicas de una escena o al escanear fotografías o documentos.

En la figura 3.6 se muestra la representación digital en forma de matriz de una imagen. Una imagen escrita en forma de matriz con $\mathbf{M} \times \mathbf{N}$ elementos se denota como $f(x, y)$, donde cada elemento que integra la matriz recibe el nombre de pixel. El pixel $[0, 0]$ se localiza en la esquina superior izquierda de la imagen (González & Woods 2008).

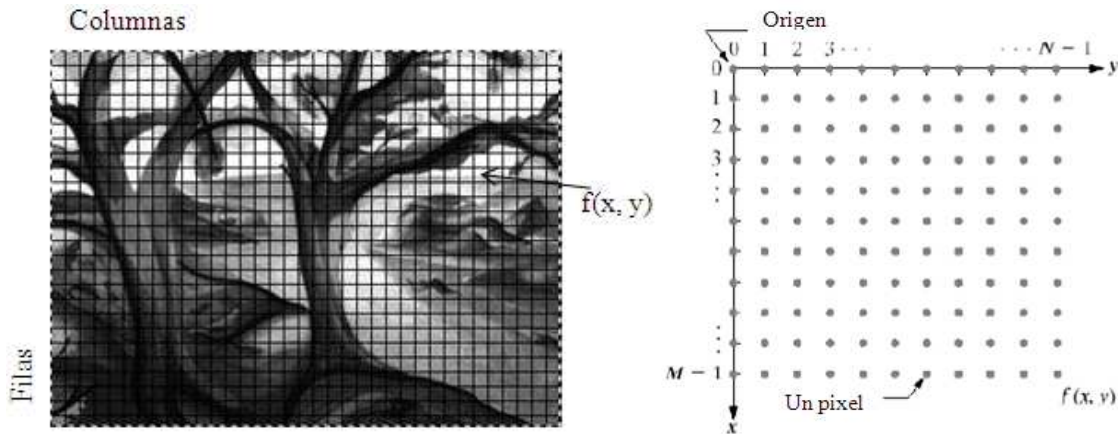


Figura 3.6. Notación de una imagen digital (González & Woods 1992).

La notación matricial de una imagen digital puede observarse en la ecuación 3.

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0, N-1) \\ f(1,0) & f(1,1) & \cdots & f(1, N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1, N-1) \end{bmatrix}, \quad (3)$$

donde $f(x, y)$ es una función de la imagen. Generalmente hace referencia a la notación matemática de la imagen que se compone de dos variables espaciales (x, y) que son valores enteros que especifican la fila y columna en la matriz de valores de pixeles. La f representa el nivel de brillo o intensidad de la imagen en las coordenadas (x, y) .

El utilizar una imagen digital genera una serie de ventajas por las muchas posibilidades de manipulación que ofrece como son: modificar el brillo, el color, se puede rotar, duplicar, aumentar, de hecho puede transmitirse a grandes distancias en pocos segundos.

Para poder trabajar con números en la computadora, el valor de cada pixel es cuantificado a códigos binarios enteros positivos, por la razón de que el brillo no puede ser negativo. Cuantificar es la conversión que sufre la amplitud de la señal analógica, generándose el concepto de nivel de gris o intensidad (Somolinos 2002), proceso a través del cual se le asigna un valor discreto a la muestra de una señal continua. La cuantificación asocia a cada pixel con una cantidad discreta, proporcional a la intensidad de radiación correspondiente a la coordenada espacial (x, y) . La cuantificación afecta directamente la cantidad de los niveles de gris en la imagen.

El número de niveles de cuantificación se calcula con la ecuación 4.

$$L = 2^B, \quad (4)$$

donde L es el número de niveles de gris y B representa el número de bits que se necesitan para cada muestra. Pratt (2007) menciona que para tener una imagen que sea de buena calidad para el ojo del ser humano es necesario tener como mínimo 64 niveles de gris. Por ello, para tener una apreciación muy real se usa una cuantificación estándar a 8 bits que equivale a 256 niveles de gris. Un ejemplo de una imagen cuantificada se muestra en la figura 3.7.



Figura 3.7. Ejemplos de los efectos de la cuantificación (Somolinos 2002).

Muestreo es el proceso que toma muestras periódicas de la amplitud de una determinada señal continua, es aquí donde se lleva a cabo una transformación del dominio continuo al discreto (Somolinos 2002). El proceso de muestreo realiza una partición del espacio en $M \times N$ píxeles, asignando a cada uno un par de coordenadas en el espacio bidimensional (x, y) . La resolución está definida entonces por la cantidad de píxeles por unidad de área.

Para tener una definición más clara, muestreo es la conversión que sufren las dos dimensiones espaciales de la señal analógica (horizontal y vertical) y que generan la noción de pixel. El número de muestras que se tomen en el sentido horizontal de la imagen equivale al número de columnas de píxeles almacenadas, así como el número de muestras tomadas en sentido vertical corresponde al número de filas.

En la figura 3.8 se puede observar un ejemplo de muestreo a partir de una imagen de prueba, en la cual se redujo a 8 y 4 veces más pequeños que la imagen original, en ambas direcciones espaciales.

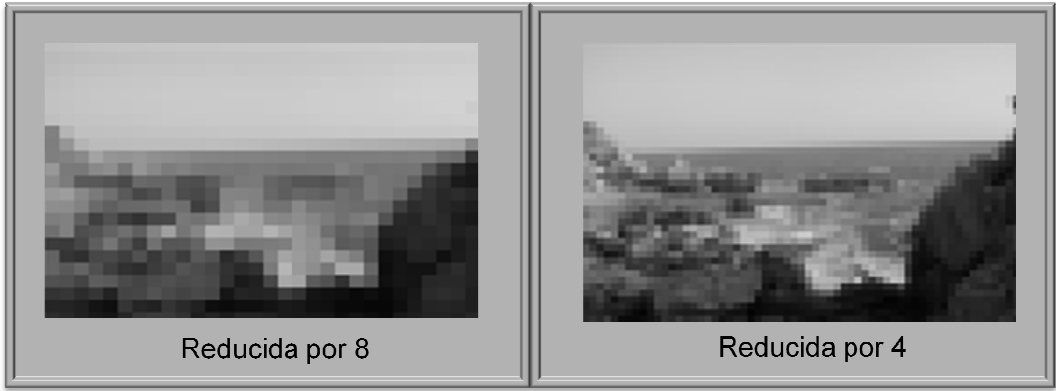


Figura 3.8. Ejemplos de los efectos de muestreo (Somolinos 2002).

3.5. Imagen de color RGB

Las imágenes a color también son conocidas como monocromáticas de tres bandas, donde cada banda representa un color diferente. Estas imágenes contienen 24 bits por pixel.

Una imagen de color RGB viene definida por tres matrices de tamaño $M \times N$, de manera a que un pixel corresponde a una posición específica en las mismas y tiene asociado una tripleta de valores, (r, g, b: red, green, blue, respectivamente) que indican la tonalidad correspondiente a cada uno de los colores rojo, verde y azul.

En la figura 3.9 se puede ver una imagen a color y los diferentes valores de cada pixel en los diferentes canales de la imagen.

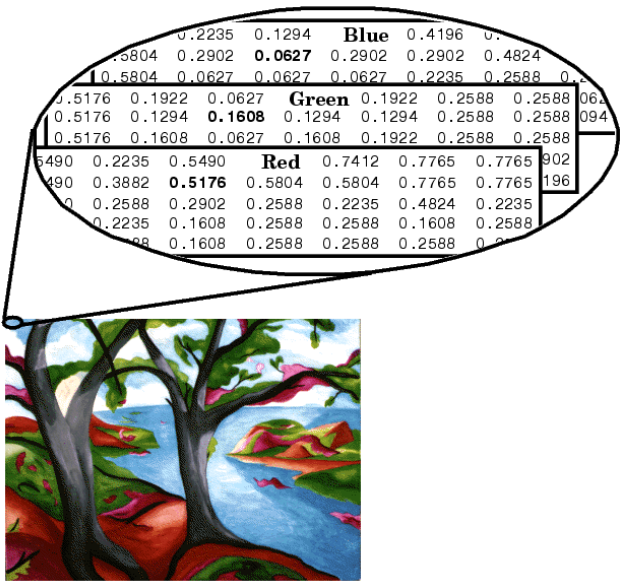


Figura 3.9. Composición de matrices de una imagen en color RGB, en Matlab (González, Woods & Eddins s.f.).

3.6. Imagen a escala de grises

Las imágenes a escala de grises reciben el nombre de monocromáticas o imágenes de un sólo color (Chacón & Molina 2007). Representan la brillantez de la imagen, contienen información de intensidad no de color. Contienen datos de 8 bits, lo que quiere decir que se tienen 256 niveles de brillantez, donde 0 representa el negro y 255 al blanco, del 1 al 254 son los diferentes niveles de gris. Son muy utilizadas porque es más rápido el procesamiento al aplicar algoritmos y técnicas de procesamiento.

En Matlab se utiliza el comando `rgb2gray(RGB)`, el cual convierte los valores de una imagen RGB a escala de grises, eliminando el matiz y la información de saturación, conservando la luminancia.

La fórmula que utiliza es la siguiente (Ec. 5):

$$0.2989 * R + 0.5870 * G + 0.1140 * B \quad (5)$$

Los valores utilizados son los que utiliza el modelo YIQ (Y = luminancia, I = fase, Q = cuadratura), el cual es utilizado por la TV abierta a color del estándar NTSC (Comisión Nacional de Sistemas de Televisión). Este modelo es compatible con la TV blanco/negro. El sistema NTSC fue desarrollado en Estados Unidos a mediados de 1940 y se emplea en la actualidad en la mayor parte de América y Japón (González & Woods 2008).

El modelo YIQ es muy utilizado, ya que es la extensión que permite explotar la característica del sistema visual humano, el cual tiene una fuerte equivalencia a un sensor azul, verde y rojo.

Y o luminancia: es el único componente utilizado por los televisores blanco y negro. Además, es el componente que utiliza Matlab para obtener una imagen a escala de grises a partir de una en RGB.

I y Q o crominancia: está formada por la diferencia entre la luminancia con el rojo (R-Y) y el azul (B-Y). La ecuación 6 que permite convertir de RGB a YIQ (Acharya & Ray 2005) se demuestra con la siguiente estructura:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.212 & -0.523 & 0.311 \end{pmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (6)$$

3.7. Histograma de una imagen

El histograma de una imagen es una función que representa el número total de píxeles que la imagen tiene en cada nivel de intensidad (eds Passariello & Mora 1995). Es usado tanto de forma cualitativa como cuantitativa. El histograma es representado por un gráfico, el cual es bidimensional y puede proporcionar información acerca del brillo y contraste de una imagen. La figura 3.10 muestra un ejemplo del histograma de una imagen.

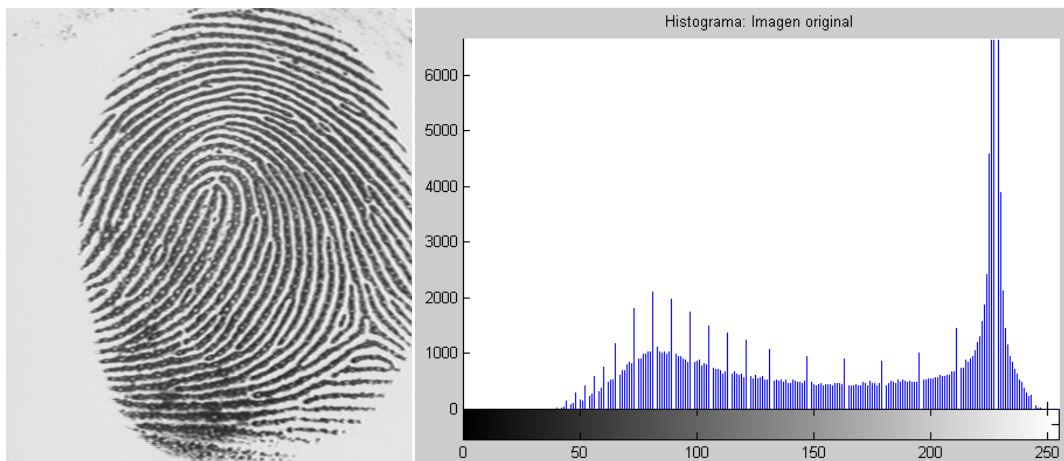


Figura 3.10. Imagen de prueba y su histograma correspondiente.

3.8. Filtrado de la mediana

Método desarrollado por Turkek y Pratt a finales de los años setenta, y partió de la idea de conseguir el suavizado de imágenes. De acuerdo a Somolinos (2002), esta técnica consiste en generar la nueva imagen denominada $g(x, y)$ con el fin de encontrar la mediana de los valores que se encuentran en el entorno de vecindad del punto (x, y) en la imagen original llamada $f(x, y)$.

La mediana tiene como función hacer que los puntos con intensidades muy distintas se asemejen a sus vecinos, es muy efectivo para eliminar píxeles cuyo valor es muy diferente al resto de sus vecinos. Se utiliza para eliminar ruido en las imágenes.

En un entorno de vecindad 3×3 como el que utiliza la función `medfilt2` de la herramienta Matlab, para obtener la mediana primeramente se ordenan los píxeles de manera ascendente a los niveles de gris, después se elige el que está en la quinta posición. Éste será el valor que reemplazará al que contiene el píxel $[i, j]$, desde luego se tendrá que

repetir el algoritmo recorriendo toda la imagen. En la figura 3.11 se da un ejemplo de este proceso.

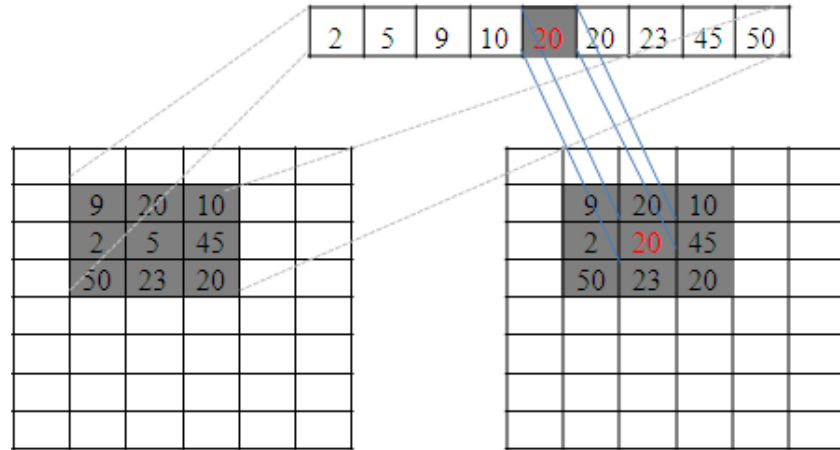


Figura 3.11. Proceso para obtener la mediana usando un vecindario de 3 x 3.

La mediana es un filtro no lineal debido a que la operación que se realiza no es una convolución, sino que aplica un algoritmo u operación a la matriz de píxeles.

Un ejemplo de la aplicación del filtro de mediana a una imagen a escala de grises se muestra en la figura 3.12.

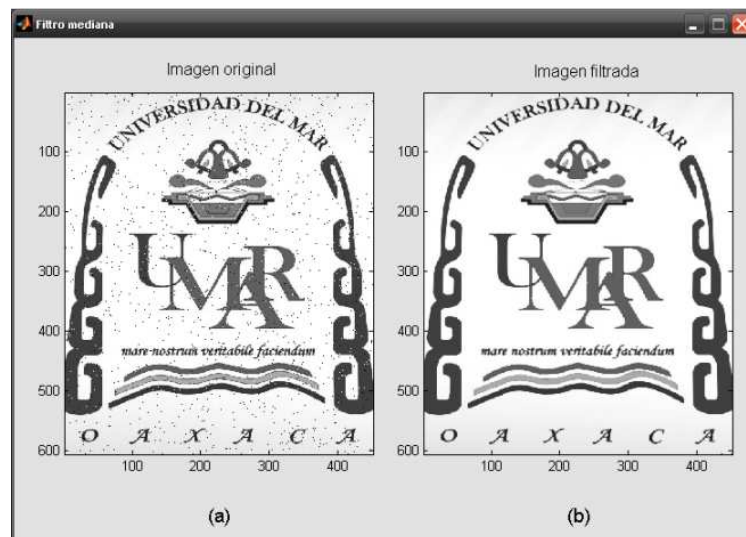


Figura 3.12. Resultado del filtro de la mediana. (a) Imagen original. (b) Resultado al aplicar filtro mediana.

3.9. Degradación de imagen

El objetivo de la restauración digital de imágenes es hallar cómo era una imagen original luego que ha sido degradada de alguna forma (por el sistema de observación).

Según Baume (2007) la degradación puede ser causada por dos razones:

Borrosidad (blur): Debido al movimiento relativo entre el detector y el objeto, turbulencia atmosférica o imágenes fuera de foco.

Ruido (noise): Debido a diversas causas, como los granos en una fotografía o ruido electrónico.

El ruido es considerado como una información indeseable que genera contaminación en una imagen, su origen procede de una gran variedad de fuentes, generalmente se obtiene durante el proceso de adquisición de la imagen (Pajares & De la Cruz 2008).

Dentro de los tipos de ruido se encuentra:

Ruido gaussiano: Produce pequeñas variaciones en la imagen. Se debe al ruido de los digitalizadores; perturbaciones en la transmisión.

Ruido sal y pimienta: Existen píxeles que toman valores muy altos o bajos con respecto al de sus vecinos, se caracteriza entonces porque el pixel toma un valor máximo, causado por una saturación en el sensor o mínimo si se ha perdido su señal. Por lo general contiene ocurrencias aleatorias de valores de intensidades blanco y negro.

Ruido impulsivo: contiene solamente ocurrencias aleatorias de valores de intensidades de color blanco.

Ruido uniforme: El ruido uniforme, sigue una función de densidad de probabilidad (Fig. 3.13), denotada por la ecuación 7.

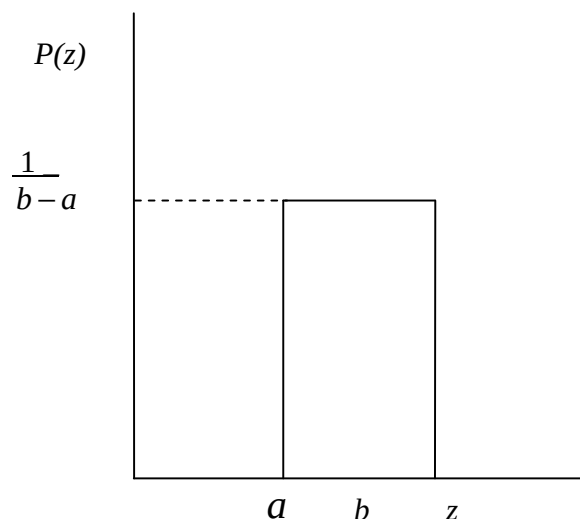


Figura 3.13. Representación de la función de probabilidad uniforme.

$$p(z) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq z \leq b \\ 0 & \text{de otra forma} \end{cases}, \quad (7)$$

donde z es un número aleatorio que sigue a la distribución mostrada en la figura 3.13, este valor se genera para cada pixel de la imagen, a es un número mayor que cero y b es un número entero positivo mayor o igual al valor de a . Los límites a y b corresponden a un rango determinado por el usuario, lo que indica que z tendrá valores que van de a hasta b . el valor generado se añade al valor original del pixel.

El ruido uniforme comúnmente se emplea para degradar imágenes que se utilizan en la evaluación de algoritmos de restauración de imágenes digitales.

3.10. Modelo de degradación

En la figura 3.14 se presenta el diagrama del modelo de degradación (González & Woods 1992) que se lleva a cabo en las imágenes digitales.

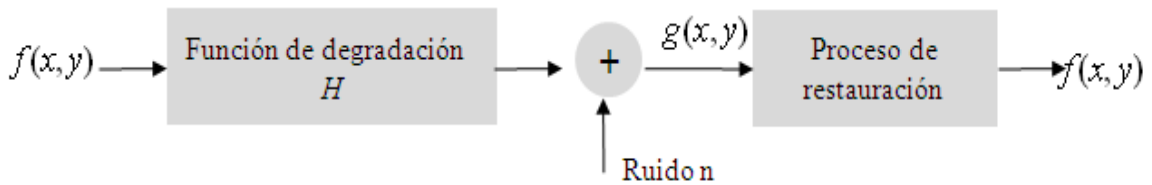


Figura 3.14. Modelo de degradación.

El proceso de degradación está modelado por una función H , que junto a un término aditivo de ruido n actúa sobre una imagen de entrada $f(x, y)$ para producir una imagen degradada $g(x, y)$. La restauración digital de imágenes puede definirse como el proceso de obtener una aproximación de $f(x, y)$, a partir de $g(x, y)$ y el conocimiento de la degradación ocasionado por la función H . Por lo regular, el conocimiento de n se limita a información de naturaleza estadística. La relación entrada-salida del modelo de degradación se define como (Ec. 8):

$$g(x, y) = H[f(x, y) + n(x, y)] \quad (8)$$

La degradación de una imagen puede ser “espacialmente invariante” o “espacialmente variante”. La primera afecta a todos los píxeles de igual forma en una imagen digital, mientras que la segunda no afecta a todos los píxeles por igual, por lo que es mucho más difícil de modelar que la degradación invariante.

3.11. Restauración digital de imagen

La restauración es un proceso que intenta recuperar o reconstruir un objeto, con lo cual se deduce que la restauración digital de imágenes consiste en recuperar una determinada imagen logrando la reducción o eliminación de las degradaciones en las que se incurrieron mientras la imagen digital fue obtenida. Según Pajares & De la Cruz (2008) se puede considerar a la restauración como un proceso, el cual mediante algún conocimiento *a priori* del fenómeno de la degradación intenta recuperar o reconstruir una imagen digital. Estas degradaciones incluyen borrosidad o empañamiento ocasionado por sistemas ópticos, movimiento de la imagen, así como ruido por fuentes electrónicas y fotométricas.

Mediante la restauración de imágenes por computadora es posible obtener información relevante de la escena captada por una cámara. La restauración de imágenes sirve para mejorar una imagen con fines interpretativos, al aplicar la restauración se podrá extraer más información para una toma de decisiones.

A partir del modelo de degradación, González & Woods (1992) definen la restauración como el proceso de adquirir de manera aproximada $f(x, y)$, a partir de $g(x, y)$ y del conocimiento de la degradación en la forma del operador H . Se ha de suponer que el valor de $n(x, y)$ se limita a una información que es de naturaleza estadística.

3.12. Dominio espacial y dominio de la frecuencia

Existen dos dominios en los cuales se hace el mejoramiento de una imagen: el dominio espacial y el dominio de la frecuencia.

El dominio espacial se refiere al propio plano de la imagen, en donde las técnicas que pertenecen a esta categoría se basan en la manipulación directa de los píxeles de la imagen y se denotan por la ecuación 9.

$$g(x, y) = T[f(x, y)] \quad (9)$$

donde $f(x, y)$ es la imagen de entrada, $g(x, y)$ es la imagen de salida y T un operador sobre f , definido sobre una vecindad (x, y) . El proceso utiliza sólo los píxeles alrededor de la vecindad (x, y) .

Los métodos de restauración de imágenes digitales se originaron a partir del estudio de los conceptos manejados en el dominio de la frecuencia (Pajares & de la Cruz 2008), de ahí la importancia de conocer el significado de éste.

En una imagen el dominio de la frecuencia se utiliza para conocer donde se encuentran los distintos rangos de frecuencias. Frecuencia es la cantidad de veces que ocurre un determinado suceso periódico en cierta unidad de tiempo.

El dominio de la frecuencia se basa en la aplicación de la transformada de Fourier sobre la imagen capturada, lo cual genera una nueva imagen, a la cual se le aplican los filtros y, si es necesario, se regresa al dominio espacial aplicando la transformada inversa de Fourier.

De acuerdo a la figura 3.15, la imagen original $I[x, y]$ se transforma en una imagen, $I[w1, w2]$ en el dominio de la frecuencia mediante la transformada de Fourier $\mathcal{T}$. El operador de la transformada $T[\cdot]$ ahora trabaja con la información de frecuencia contenida en $I[w1, w2]$ y genera una nueva imagen, $I'[w1, w2]$ en el dominio de la frecuencia. Por último la imagen $I'[x, y]$ es convertida al dominio espacial mediante la transformada inversa de Fourier $\mathcal{T}^{-1}$ (Chacón & Molina 2007).

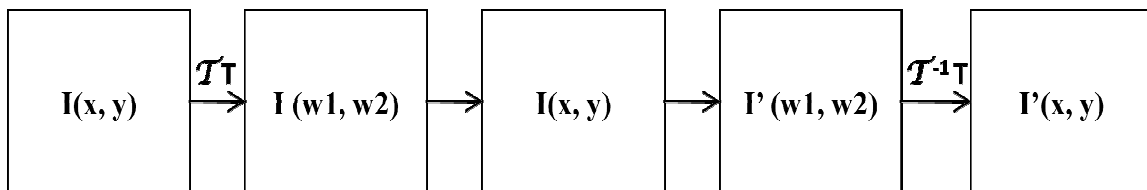


Figura 3.15. Tratamiento de la información de una imagen en el dominio de la frecuencia.

Transformar es hacer que una cosa cambie o sea distinta, pero sin alterar sus características esenciales.

Una transformada es la operación matemática que convierte una función de un dominio a otro, codifica información de una imagen de forma completa que posteriormente puede ser reconstruida utilizando un método inverso. Puede definirse como el mapeo a

partir de un sistema de coordenadas a otro. Por ejemplo, la rotación de una imagen es una transformada, este sistema es distinto al sistema original, pero cada coordenada en la imagen original tiene su correspondiente valor en la imagen rotada.

3.13. Transformada de Hough

La transformada de Hough fue nombrada así en honor a su creador Paul Hough que la propuso en el año de 1962. Es una técnica que permite identificar características de forma particular dentro de una imagen, básicamente son líneas, rectas y círculos. Esta transformada busca formas geométricas en toda la imagen, encontrando los parámetros de aquella que contenga más puntos en ella (Sucar & Gómez 2003).

Para este trabajo sólo se realizó la identificación de líneas porque la degradación por borrosidad para este caso de estudio es ocasionada por un movimiento lineal uniforme, por ello es necesario explicar cómo funciona la transformada de Hough para esta función.

De forma gráfica la transformada de Hough se representa de la siguiente manera (Fig. 3.16).

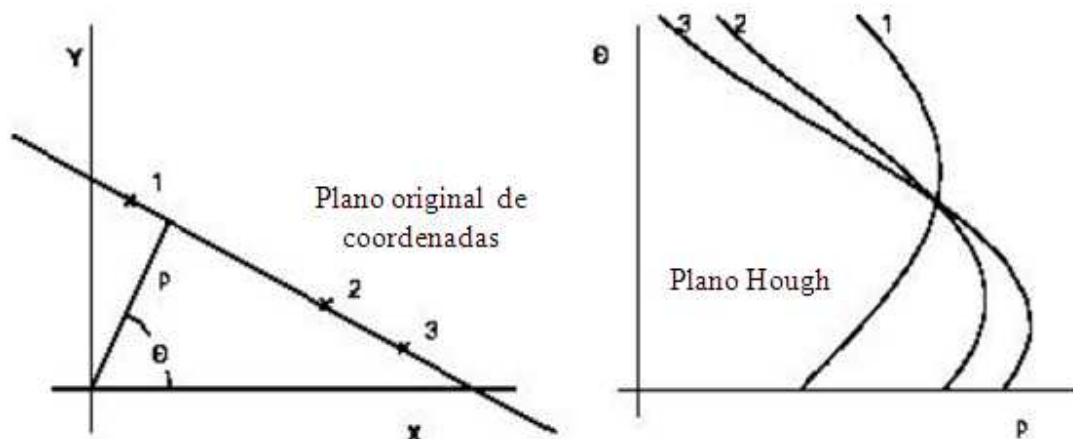


Figura 3.16. Representación gráfica de la transformada de Hough.

Los puntos 1, 2 y 3 son los puntos que pertenecen a la recta, ρ es la distancia al origen de la recta y θ es el ángulo de ρ con respecto al eje x . Cualquier línea recta en una imagen es representada en un punto simple (ρ_1, θ_1) en el espacio paramétrico. Los puntos de la intersección de las curvas en el espacio paramétrico corresponden a los parámetros de las posibles rectas que se encuentran en la imagen.

La ecuación para describir una línea recta es la notación mostrada en la figura 3.17:

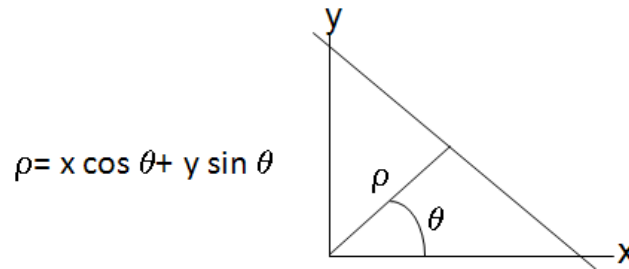


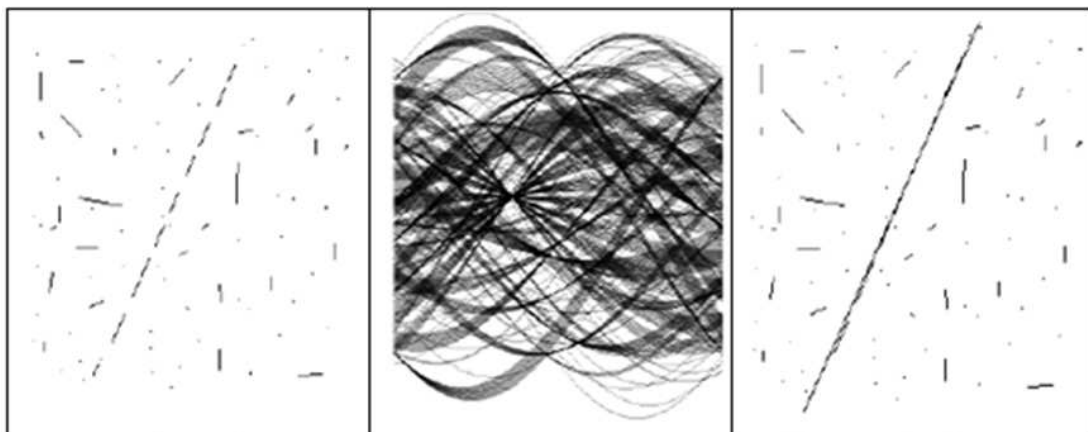
Figura 3.17. Representación de la ecuación paramétrica de una recta.

Donde ρ es la longitud y θ es el ángulo de ρ con respecto al eje x . Todos los puntos que pertenezcan a la misma recta coincidirán en las componentes θ y ρ . Se considera que los puntos pertenecientes a la línea, son aquellos cuyo valor sea más alto de un valor prefijado. Para cada punto (x_i, y_i) detectado se calcula la curva, usando la ecuación 10.

$$\rho = x_i \cos \theta + y_i \sin \theta, \quad (10)$$

donde θ y ρ son variables y representan todas las rectas que pasan por el punto (x_i, y_i) . Al final aquellos valores más repetidos coinciden con la recta que contiene el mayor número de intersecciones.

En la figura 3.18 se muestra un ejemplo de la transformada de Hough. Donde de la imagen original (Fig. 3.18a), se pasa al espacio paramétrico (Fig. 3.18b) y donde existe el mayor número de intersección de líneas se encontrará la mejor recta (Fig. 3.18c).



(a) Imagen Original

(b) Espacio paramétrico(θ, ρ)

(c) Superposición de la mejor recta encontrada(θ, ρ)

Figura 3.18. Ejemplo de la transformada de Hough en una imagen (Sucar & Gómez 2003).

Descripción del algoritmo de la transformada de Hough

Los pasos (Rain, Kasturi & Schunck 1995) para llevar a cabo el algoritmo de la transformada Hough son:

- 1.- Discretizar el espacio paramétrico (θ , ρ).
- 2.- Asumir que cada celda en el espacio paramétrico es un acumulador. Inicializar todas las celdas a cero.
- 3.- Para cada punto (x , y) en el espacio de la imagen, incrementar a 1 cada uno de los acumuladores que satisface la ecuación, esto es:

Hacer $\rho = x \cos \theta_d(h) + y \sin \theta_d(h)$.

Encontrar el intervalo K que sea más cercano al ρ obtenido.

Incrementar la matriz de acumuladores.

- 4.- Encontrar el máximo en el arreglo de acumuladores correspondiente a los parámetros del modelo de instancias.
- 5.- Con ρ y θ encontrados, se describe la línea en forma polar.

3.14. Transformada de Fourier

Aunque existen diversos tipos de transformadas, una de las más utilizadas es la transformada de Fourier que convierte coordenadas espaciales en frecuencias. Es una técnica esencial para la restauración digital de imágenes. Fue el físico, Joseph Fourier quien desarrolló una representación de funciones basada en la frecuencia, a lo cual se denominó transformadas de Fourier.

Es el método para moverse dentro y fuera del dominio de la frecuencia. El hecho de que una imagen se pueda convertir en una representación del dominio de la frecuencia implica que la imagen puede contener información de alta frecuencia o de baja frecuencia. Por ejemplo, si el nivel gris de una cierta porción de una determinada imagen cambia lentamente a través de las columnas, entonces sería representado en el dominio de la frecuencia como función del seno o del coseno teniendo una frecuencia baja. El contar una imagen en el dominio de la frecuencia ayuda a la mejora, restauración, codificación y descripción de las imágenes. Cualquier señal periódica puede representarse por una suma de señales en senos y cosenos con diferente amplitud, frecuencia y fase. Por lo tanto, si se

conoce la información en frecuencia contenida en la imagen es posible diseñar los filtros necesarios para la restauración de la imagen. La representación gráfica de la transformada de Fourier es un diagrama, denominado espectro de Fourier, donde se representa la amplitud y frecuencia de cada una de los componentes seno y coseno determinados. En la figura 3.19 se presenta un ejemplo.

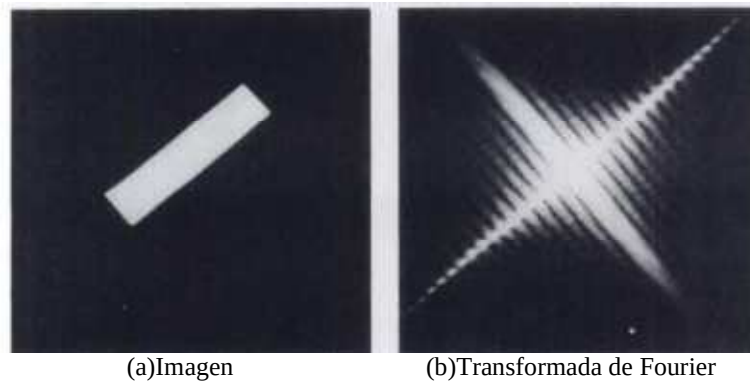


Figura 3.19. Ejemplo de la transformada de Fourier de una imagen (González & Woods 1992).

Las ecuaciones 11, 12, 13, 14, 15 y 16 referentes a la transformada de Fourier están descritas en Acharya & Ray (2005).

La transformada de Fourier de una función $f(x)$ describe la cantidad de cada término de frecuencia que debe ser agregada para describir la forma de $f(x)$ y ésta se encuentra dada por la ecuación 11:

$$F(w) = \int_{-\infty}^{+\infty} f(x) \exp[-j2\pi wx] dx, \quad (11)$$

en donde:

X = Tiempo.

W = Frecuencia en hertz.

$f(x)$ = Señal de prueba.

$\exp[-j2\pi wx]$ = Función base de la transformación, donde $j = \sqrt{-1}$.

$F(w)$ = Espectro de la función de la frecuencia de W .

La ecuación 12 muestra la inversa de Fourier:

$$f(x) = \int_{-\infty}^{+\infty} F(w) \exp[j2\pi wx] dw \quad (12)$$

Ésta permite recuperar la señal original.

Transformada discreta de Fourier

Las fórmulas anteriores representan la transformada de Fourier para funciones continuas, pero es necesario definir la representación de éstas como funciones discretas, por lo que la transformada discreta de Fourier (DFT, por sus siglas en inglés) se define como (Ec. 13):

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp\left[-j \frac{2\pi ux}{N}\right], \quad (13)$$

en donde:

N = Número de muestras de $f(x)$.

$f(x)$ = Señal de prueba discreta con índice x . Los valores discretos que puede tomar x van desde 0, 1, 2, ... $N-1$.

$\exp\left[-j \frac{2\pi ux}{N}\right]$ = Función base de la transformación para señales discretas.

$f(u)$ = Espectro en función de la frecuencia discreta (con índice u).

Su correspondiente ecuación inversa (Ec. 14):

$$f(x) = \sum_{u=0}^{N-1} F(u) \exp\left[j \frac{2\pi ux}{N}\right] \quad (14)$$

Por lo que se puede observar, la integral se convierte en la suma de todos los puntos muestreados.

Las transformadas anteriores son para uso en una sola dimensión, pero como las imágenes son representaciones bidimensionales, es necesario conocer las fórmulas que se aplican en estos casos.

La transformada de Fourier de una imagen $f(x, y)$ de tamaño $M \times N$ de 0 a $M-1$ y 0 a $N-1$, viene dada por la ecuación 15:

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp\left[-j2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)\right], \quad (15)$$

en donde:

N y M = Representan el tamaño de las muestras $f(x, y)$, para imágenes corresponde a los valores de la dimensión.

$f(x, y)$ = Señal de prueba discreta. Los valores discretos que puede tomar X van desde 0, 1, 2, ... $N-1$ y para y desde 0, 1, 2, ... $M-1$.

$\exp\left[-j \frac{2\pi ux}{N}\right]$ = Función base de la transformación para señales discretas.

$f(u)$ = Espectro en función de la frecuencia discreta (con índice u).

La ecuación inversa de la DFT bidimensional se define con la ecuación 16.

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp\left[j2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)\right] \quad (16)$$

Transformada rápida de Fourier

La transformada rápida de Fourier (FFT, por sus siglas en inglés) es un algoritmo que calcula de manera más rápida la transformada discreta de Fourier. Reduce el tiempo de cálculo de la transformada discreta, esto es, el número de operaciones, gracias a trucos de programación, ya que calcula algunos de los valores por adelantado y fuera del ciclo en el programa, además de utilizar técnicas matemáticas. El implementar la transformada discreta involucra una gran cantidad de sumas y multiplicaciones complejas que es proporcional a N^2 . La FFT reduce este número aproximadamente a $N \cdot \log_2 N$, por ejemplo para una imagen de 1024 x 1024 píxeles $N = 1024$, se tendría:

$N^2 = 1,048,576$ operaciones complejas. Mientras que con FFT se tendría,

$N \cdot \log_2 N = 10,240$ operaciones complejas.

La reducción de operaciones se debe a que el término $\exp[j2\pi ux/M]$ puede ser almacenado en una tabla para aplicaciones subsecuentes; por tal razón, la multiplicación de u de X en este término no se contabiliza normalmente como parte de la implementación. Además con la FFT se mejora la precisión de los cálculos, pues al tener que realizar menos operaciones, se reducen los errores de redondeo debidos a la limitación del tamaño de las palabras por parte del equipo de cómputo (Pratt 2007).

3.15. Métodos de restauración

Los métodos de restauración a los que hace mención este trabajo se realizan en el dominio de la frecuencia, por lo que una vez que el modelo de degradación ha sido terminado se debe extender con ceros hasta el tamaño de la imagen, antes de obtener su transformada de Fourier, esto con el fin de lograr que la convolución lineal y circular sean iguales (González & Woods 2008). Aunque la restauración no pueda realizarse de manera perfecta, las técnicas de restauración permiten mejorar la calidad de la información contenida en una imagen.

3.16. Filtro de Wiener

El filtro de Wiener fue propuesto por Norbert Wiener durante la década de 1940, publicado en 1949. Es un filtro lineal de tiempo discreto. Su origen se remonta a principios de la Segunda Guerra Mundial, ya que en este tiempo Norbert estaba realizando trabajos sobre un proyecto de tipo militar, el cual consistía en diseñar un controlador que dirigiera el fuego antiaéreo con base en información que proveía el radar. El problema era que el controlador tenía que predecir el movimiento del objetivo utilizando información con ruido que proporcionaba el radar. La solución que encontró Wiener fue hallar el menor error cuadrático medio en función de la correlación (Molina *et al.* 2001).

También conocido como mínimos cuadrados, consisten en estimar una imagen de tal forma que la diferencia entre ésta y la imagen original sea mínima, en el sentido de mínimos cuadrados. Para hacerlo de forma rápida, se utiliza el algoritmo de la transformada rápida de Fourier. El filtro de Wiener trabaja en los casos donde el ruido ha llegado a ser significativo, este ruido se presenta cuando el movimiento relativo entre la cámara y la escena original es grande.

Los filtros Wiener son filtros lineales adaptativos (adaptación del comportamiento acorde a las características de la vecindad) basados en las características de varianza locales de la imagen. Los filtros Wiener suavizan gradualmente cambiando áreas de una imagen donde el ruido es muy aparente, pero manteniendo áreas donde los detalles están presentes y el ruido es menos aparente.

El filtro Wiener mantiene bordes y otras partes de alta frecuencia de una imagen. Se adapta a la varianza local de la imagen. Donde ésta es mayor, el filtro realiza un suavizado menor. Donde la varianza es menor, realiza un mayor suavizado.

El filtro Wiener se adapta a la varianza local de la imagen. Donde ésta es mayor, el filtro Wiener realiza un suavizado menor. Donde la varianza es menor, realiza un mayor suavizado, manteniendo bordes y otras partes de alta frecuencia de una imagen.

Error cuadrático medio es el estimador que indica el valor esperado del error al cuadrado, donde “error” hace referencia a la cantidad por la cual la imagen estimada es diferente a la imagen original. Esta diferencia sucede cuando se lleva a cabo el proceso de restauración de imágenes, debido a que existen parámetros que no se conocen.

El error cuadrático medio es definido por la ecuación 17:

$$e^2 = E \left\{ \left[f(x, y) - \hat{f}(x, y) \right]^2 \right\}, \quad (17)$$

en donde:

e^2 = Error cuadrático medio.

$E\{\}$ = Valor esperado del argumento.

$f(x, y)$ = Matriz que contiene los datos de la imagen digital.

$\hat{f}(x, y)$ = estimado de la imagen original f .

La figura 3.20 muestra un ejemplo de la restauración de una imagen digital usando el filtro de Wiener.



Figura 3.20. Ejemplo de la restauración de una imagen digital usando filtro Wiener (Pratt 2007).

3.17. Filtro inverso

El filtro inverso es una de las técnicas desarrolladas para llevar a cabo la restauración de imágenes digitales. La principal característica de este filtro es que permite obtener una primera aproximación sobre la imagen original para imágenes que se encuentran borrosas.

La ecuación 18 describe el filtro inverso:

$$F'(u, v) = \frac{G(u, v)}{H(u, v)}, \quad (18)$$

en donde:

$F'(u, v)$ = Estimación de la imagen original.

$H(u, v)$ = Es la transformada de Fourier de la función de degradación.

$G(u, v)$ = Transformada de Fourier de la imagen degradada.

Este método obtiene como resultado un estimado de la imagen original dividiendo la transformada de la imagen degradada $G(u, v)$ entre la transformada de la función de la degradación conocida $H(u, v)$. En la mayoría de los casos, la función de degradación se obtiene mediante observación y experimentación.

El filtro inverso, según González & Woods (1992), es un método de restauración de imágenes. Su nombre proviene de considerar $H(u, v)$ como una función o filtro que multiplica a $F(u, v)$. Este filtro es utilizado cuando la degradación es muy pequeña. En la figura 3.21 se muestra un ejemplo de la restauración de una imagen digital usando el filtro inverso.



Figura 3.21. Ejemplo de la restauración de una imagen digital usando filtro inverso (Pratt 2007).

3.18. Criterio minimización de rizado

Los autores Mery & López (2003) definen que el criterio de minimización de rizado consiste en disminuir las oscilaciones que existen en una imagen causadas por la degradación. Debido a que existen muchas soluciones, se debe tomar en cuenta la que tenga menos picos en su respuesta.

El criterio de minimización de rizado es aplicable a un movimiento horizontal puro, lo que indica que el valor del ángulo de borrosidad es 0° y sólo se puede utilizar donde la degradación de una imagen sea lineal uniforme. Para aplicar la degradación se hace uso de una convolución, la cual se define por la siguiente ecuación matricial que se representa en la ecuación 19.

$$g = f * h = \begin{bmatrix} h_1 & \dots & h_n & 0 & 0 & 0 & 0 \\ 0 & h_1 & \dots & h_n & 0 & 0 & 0 \\ \vdots & & & & \vdots & & \\ 0 & 0 & \dots & 0 & h_1 & \dots & h_n \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_M \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_N \end{bmatrix}, \quad (19)$$

donde f representa una fila de la imagen original, h es la respuesta al impulso de la degradación y g representa una fila de la imagen degradada. La degradación corresponde a un movimiento lineal uniforme de n pixeles, los elementos de la matriz de convolución son constantes. Por ejemplo, para una fila $H = (h_1, h_2, h_3, h_4 \dots h_n)$ donde $h_n = 1/n$, los elementos de h deben cumplir con la condición $\sum h_n = 1$ para que la imagen degradada no sea ni más oscura, ni más clara que la imagen original. En el desplazamiento horizontal se realiza la transpuesta para pasar los elementos a una fila.

Este criterio fue establecido a partir de un conjunto de soluciones para el proceso de degradación, en los cuales se realiza una evaluación de distintos criterios de minimización de rizado según costo computacional y desempeño en la restauración de imágenes con y sin ruido.

En la figura 3.22 se muestran las gráficas de rizado para una fila con degradación $n = 2$. Es posible observar que en la solución propuesta los rizados son más notorios, motivo por el cual este método busca disminuir este rizado.

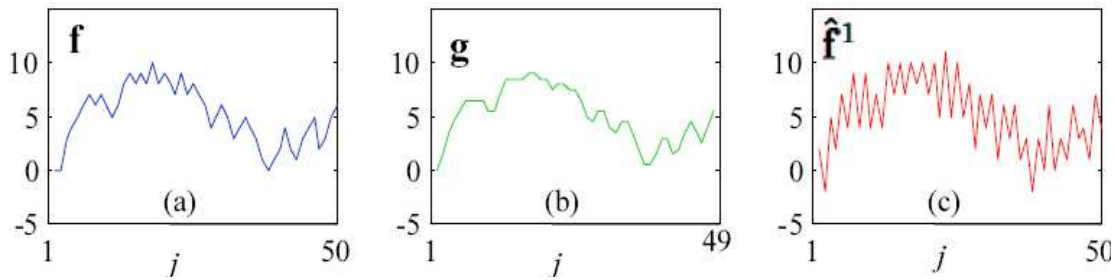


Figura 3.22. Restauración de una fila f : a) imagen original. b) fila degradada con $n = 2$. c) posible solución.

La restauración para minimizar los rizados consiste en recuperar de la mejor manera posible la fila de la imagen original. Para llevar a cabo esta operación existen algunos modelos matemáticos que permiten restaurar la imagen, los cuales se basan en minimizar el rizo, ya sea con respecto a la fila de la imagen degradada o con respecto a los armónicos de las gráficas de la fila.

Existen dos formas de lograr la reducción de estos rizados: los criterios de similitud y los criterios de frecuencia. Los primeros pueden ser utilizados para restaurar una imagen que ha sido degradada por un proceso de degradación. Los criterios de similitud se basan en la idea que dos señales son similares si éstas se correlacionan entre sí; las métricas de correlación, la covarianza y el coeficiente de correlación sirven para evaluar la similitud entre dos señales. Los segundos establecen criterios que tienen como finalidad reducir el rizado de la señal restaurada a partir de la reducción de sus armónicos.

La correlación es una operación parecida a la convolución, en la cual el valor de un pixel de salida se calcula como la suma ponderada de los pixeles vecinos. La diferencia es que la matriz o máscara, en el caso de la correlación llamada núcleo o *kernel* de correlación, no se rota durante el cálculo (Esqueda & Palafox 2005). La correlación permite encontrar el parecido entre pixeles de una imagen, se dice que los pixeles están correlacionados si sus valores son iguales o parecidos. La correlación es muy utilizada en el reconocimiento de patrones.

3.19. Teorema de No Free Lunch

Los algoritmos de restauración utilizados en este trabajo de tesis han sido evaluados desde distintas perspectivas, lo que generó que el desempeño de estos fuera diferente en algunos

casos, al observar esta reacción fue interesante investigar la existencia de un fundamento teórico que respaldara dicho comportamiento, al hacer esta actividad se encontró el teorema denominado “No Free Lunch”. La traducción del nombre en español prácticamente es: “No hay almuerzo gratis”, que deriva del hecho de que no se puede obtener algo por nada y cuyo término se hizo muy popular en ese tiempo, siendo el motivo por el cual se le dio tal nombre.

Este teorema fue creado por David Wolpert y William Macready en 1995, donde básicamente explica que: *“el rendimiento de todos los algoritmos de optimización amortizado sobre todas las funciones objetivo siempre es igual (en espacios discretos)”*. (Wolpert & Macready 1997).

El espacio discreto contiene un número finito de puntos muestrales es decir, es aquel donde se puede contar el número de posibles resultados, mientras que el espacio continuo contiene una afinidad de puntos muestrales, en otras palabras no se puede enumerar los posibles resultados. De esto se concluye que el tipo de problema que se está resolviendo es continuo ya que no se tiene un dominio finito de posibilidades donde la restauración es satisfactoria.

Lo que significa que un algoritmo va tener campos de aplicación donde su rendimiento es deficiente, de tal forma que si un algoritmo es muy eficiente para una colección de problemas, se mostrará muy ineficiente para otra colección, en otras palabras un algoritmo se comporta en estricta concordancia con la cantidad y calidad del conocimiento específico del problema que incorpora.

3.20. Matlab

La aplicación creada para apoyar las diferentes etapas de este proyecto de tesis fue realizada bajo la plataforma del software Matlab versión 7.0, la cual fue creada el 6 de mayo de 2004, en esta tesis se hizo uso de la versión de prueba. A continuación se da una breve explicación de este software.

Matlab significa *“MATrix LABoratory”*, traducido al español es laboratorio de matrices. Es un lenguaje de alto nivel que incorpora herramientas de cálculo numérico y visualización, de la empresa Mathworks Inc. (González, Woods & Eddins 2009).

Permite la manipulación de datos matriciales, que es la forma en la que están estructuradas las imágenes al ser manipuladas por una computadora, y además cuenta con funciones apropiadas para el procesamiento de imágenes (*Image Processing Toolbox*). El *toolbox* de procesamiento de imágenes contiene un conjunto de funciones de los algoritmos más conocidos para trabajar transformaciones geométricas y manipulación de color que junto con las funciones ya integradas en Matlab permite realizar análisis y transformaciones de imágenes en el dominio de la frecuencia.

Integra un entorno de programación de fácil uso, para el desarrollo de aplicaciones e interfaces gráficas de usuario (GUI), además proporciona unos paquetes de extensión *toolboxes* para aplicaciones específicas. Estos paquetes incluyen librerías de funciones Matlab (*M-files*) que extienden las posibilidades para resolver problemas específicos.

El código para programar es muy parecido a otros lenguajes de programación como C++, Java, etc. Facilita la creación de funciones específicas sin tener la necesidad de escribirlas comando a comando. Los archivos en los cuales se guarda el código tienen extensión .m y es aquí donde se editan las funciones de llamada (callback) de cada uno de los controles que pertenecen a la interfaz gráfica. Otro de los puntos fuertes de Matlab es la creación de gráficos en dos y tres dimensiones.

Una vez que se ha dado a conocer los conceptos teóricos importantes de este trabajo, a continuación se presenta el capítulo cuatro destinado al desarrollo del tema propuesto.

CAPÍTULO 4. DESARROLLO DEL TEMA

En este capítulo se presenta una descripción de los métodos utilizados para la restauración digital de imágenes con degradación por borrosidad empleados en este trabajo de tesis, así como su respectiva evaluación y obtención de resultados.

En la figura 4.1 se muestra un diagrama general con las diferentes etapas del desarrollo de este proyecto de tesis, que son la parte primordial para llevar a cabo la restauración digital de imágenes, de esta forma se puede dar una mejor explicación de la estructura que se siguió al desarrollar este trabajo de tesis.

Cada una de las etapas presentadas en el diagrama, se describen a continuación. En el anexo A se indica como operar el software desarrollado con un ejemplo y en el anexo B se muestran los códigos implementados en Matlab.

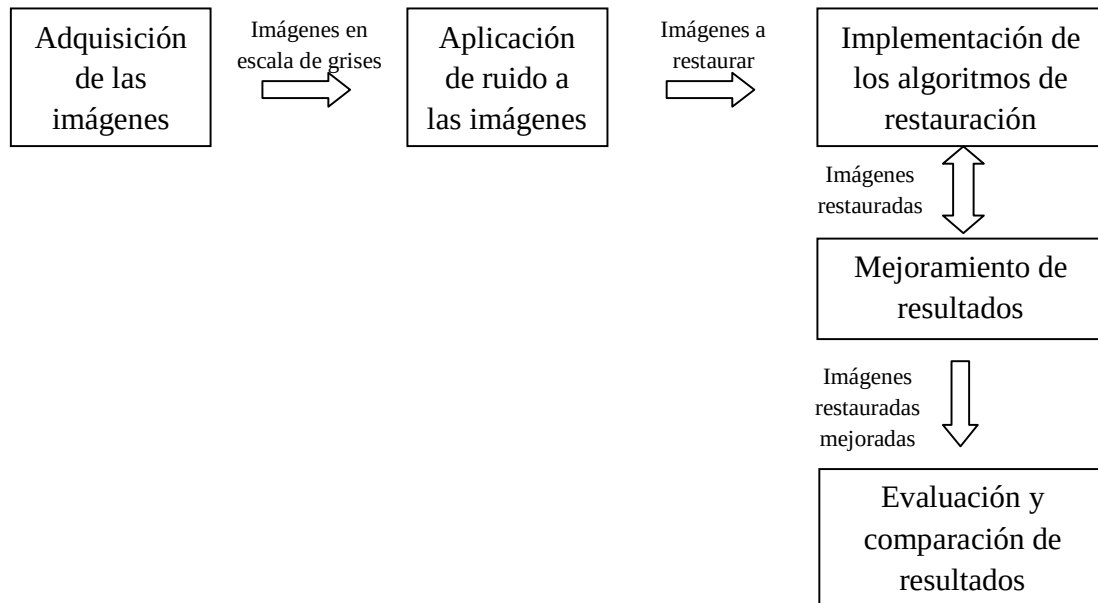


Figura 4.1. Etapas del trabajo desarrollado.

4.1. Adquisición y características de las imágenes

En esta parte se describe la forma en que se obtuvieron las imágenes utilizadas en este proyecto, haciendo mención de las características que debieron reunir.

Se buscaron imágenes en internet de diferentes dimensiones y contraste. En total fueron 15 imágenes de prueba, de las cuales 2 vienen en el *Toolbox Image Processing* de Matlab (Fig. 4.2a, Fig. 4.2g), las figuras 4.2c y 4.2d fueron adquiridas de la página oficial de la NASA (NASA 2008) y 9 de otra página especializada (González, Woods & Eddins s.f.) en el procesamiento de imágenes (Fig. 4.2e, Fig. 4.2h, Fig. 4.2i, Fig. 4.2j, Fig. 4.2k, Fig. 4.2l, Fig. 4.2m, Fig. 4.2n y Fig. 4.2o). El motivo por el cual se hizo uso de estos bancos de imágenes es que son de calidad, esta característica es importante porque el primer campo de acción a evaluar son a imágenes donde se les aplicará cierto ruido y se necesita comparar la imagen original con la imagen restaurada, además de ser de acceso gratuito y se utilizan comúnmente en el procesamiento digital de imágenes, lo que permite realizar una evaluación de resultados más confiable. También se emplearon 2 imágenes que fueron especialmente capturadas para fines de este proyecto (Fig. 4.2b, Fig. 4.2f) con una cámara digital modelo Canon PowerShot A1100 IS que cuenta con una resolución de 12 Mega Píxeles y un sensor CCD de 1/2, 3 pulgadas. Una de las principales características de

estas imágenes es que no se les ha realizado procesamiento alguno. Al unir todas las imágenes se creó un banco de imágenes, utilizado para realizar las diferentes pruebas y evaluaciones de este proyecto.

En la figura 4.2 se puede apreciar de forma visual las imágenes, que son utilizadas para estudiar el proceso de restauración digital así como sus respectivas dimensiones. En la sección donde se lleva la aplicación de ruido se explica porqué se limitó a usar este número de imágenes.

Las imágenes son de distintas dimensiones, lo que permitió estudiar el comportamiento de los métodos de restauración digital en imágenes grandes y pequeñas, además de poder evaluar el factor tiempo en el procesamiento de las imágenes. Otra de las características es que son imágenes de diversas escenas, recordando que la restauración digital es aplicable en diversas áreas del conocimiento y puede llevarse a cabo en imágenes de diversas índoles.

El equipo de cómputo que se utilizó para desarrollar este trabajo de tesis cuenta con las siguientes características:

- PC con procesador Pentium(R) D CPU 3.00GHz.
- Espacio en disco duro de 80Gb.
- 512 Mb de Memoria RAM.
- Windows XP versión profesional.
- Resolución de pantalla de 1024 x 768 pixeles.

Para llevar a cabo el procesamiento de las imágenes se manejó como herramienta la versión 7.0, también llamada *release 14* de Matlab. Con la ayuda de algunas de las librerías que ofrece, se creó una aplicación con interfaz gráfica de usuario (en el anexo A, se da una explicación más detallada), en apartados anteriores se ha detallado una explicación más exhaustiva de esta potente herramienta.

La aplicación creada en Matlab (Fig. A.3, Fig. A.4 y Fig. A.5), permite abrir imágenes que cumplan con el formato .JPEG, .PNG y .TIFF. El primer paso consiste en cargar la imagen para poder visualizarla de forma gráfica (Fig. A.7). Para abrir una imagen se busca la ruta donde ha sido guardada. La información de los pixeles de la imagen es almacenada en una matriz de datos numéricos, que es la forma en la cual se representan las imágenes una vez que han sido digitalizadas.

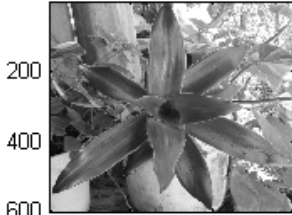
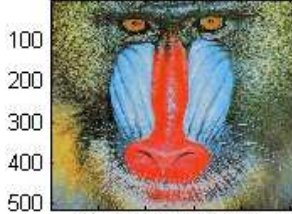
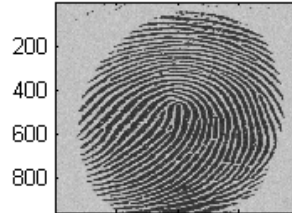
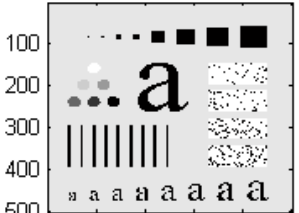
Dimensión: 350 x 258 px  (a)	Dimensión: 600 x 600 px  (b)	Dimensión: 628 x 768 px  (c)
Dimensión: 768 x 766 px  (d)	Dimensión: 512 x 512 px  (e)	Dimensión: 1100 x 1200 px  (f)
Dimensión: 198 x 135 px  (g)	Dimensión: 512 x 512 px  (h)	Dimensión: 402 x 566 px  (i)
Dimensión: 256 x 256 px  (j)	Dimensión: 688 x 688 px  (k)	Dimensión: 600 x 493 px  (l)
Dimensión: 198 x 135 px  (m)	Dimensión: 798 x 958 px  (n)	Dimensión: 350 x 258 px  (o)

Figura 4.2. Imágenes originales de prueba.

Para que la imagen pueda ser procesada es necesario convertirla a escala de grises. Todas las imágenes mostradas en la figura 4.2 se transforman en formato a escala de grises, debido a que el procesamiento de evaluación de los resultados, será más rápido, ya que no se trabaja con las tripletas de color, que es como están constituidas las imágenes RGB. En la figura 4.3 se muestra un ejemplo de cómo sería el resultado, después de convertir una imagen a monocromática.

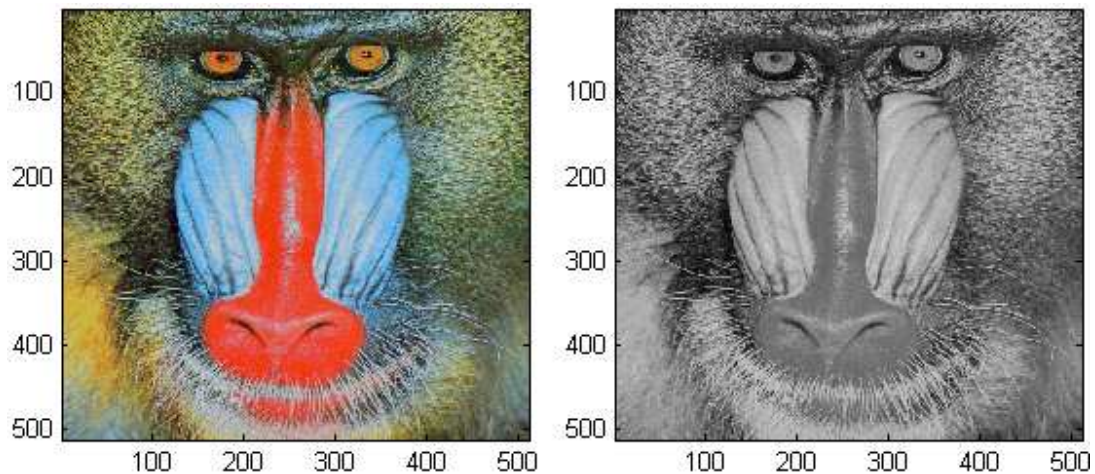


Figura 4.3. Muestra de imagen RGB y su resultado a escala de grises.

Una vez realizado el proceso de conversión a escala de grises, las imágenes están listas para que se pueda aplicar el ruido o bien, si ya son imágenes con borrosidad, aplicar la etapa de la restauración.

4.2. Aplicación de ruido a las imágenes

Del total de las imágenes recopiladas y adquiridas existe un conjunto de imágenes de prueba que no contienen degradación. A partir de este conjunto de imágenes sin degradación se optó por generar un nuevo conjunto de imágenes con degradación por borrosidad, mediante el uso de los parámetros de movimiento. Para esto se usó la función de propagación PSF que corresponde a un movimiento lineal invariante a través de cierta distancia, que equivale al valor de píxeles que se recorre un punto y un ángulo para saber la dirección hacia dónde va efectuar el desplazamiento. Para aplicarle al conjunto de imágenes esta degradación se tiene que convolucionar el filtro con la imagen original. La razón por la

que se aplicó este ruido, es porque corresponde al tipo de degradación que se planteó estudiar desde el inicio de este proyecto.

En la figura 4.4 se muestran los resultados de aplicar degradación por borrosidad a las 15 imágenes de prueba. Esta figura se integra de dos elementos principales: el ángulo en grados sexagesimales ($^{\circ}$) y la distancia de borrosidad medida en píxeles.

Los valores de los ángulos son de 0° , 45° , 90° , 180° y 270° . Se eligieron estos valores para poder estudiar el comportamiento de los filtros con imágenes con borrosidad en diferentes direcciones. Para las distancias se utilizaron valores de 20, 60 y 100 píxeles; todos estos valores deben ser menores a la dimensión de una imagen y como todas cumplen con esta condición se obtuvo un total de 15 imágenes de prueba. Se aclara que no se utilizó ningún criterio para determinar qué parámetros se aplicaban a qué imágenes, buscando generar imágenes con variedad de características y sin ningún tipo de sesgo inductivo.

La figura 4.4g es la más pequeña de las imágenes de prueba, con una dimensión de 198×135 píxeles y debido a esta característica al aplicar la degradación no se puede apreciar a simple vista los objetos que contiene esa escena, mientras que la figura 4.4d, que tiene dimensiones de 768×766 píxeles, se aprecia de forma más clara el contenido de la escena. Con lo que se muestra que el conjunto de imágenes utilizado en este trabajo de tesis, está compuesto por imágenes variadas con diferentes características.

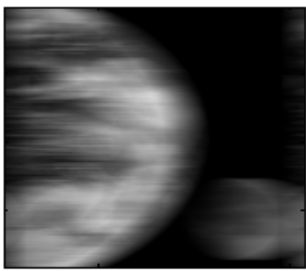
Distancia Ángulo	20 píxeles	60 píxeles	100 píxeles
	 (a)	 (b)	 (c)

Figura 4.4. Imágenes con degradación por borrosidad.

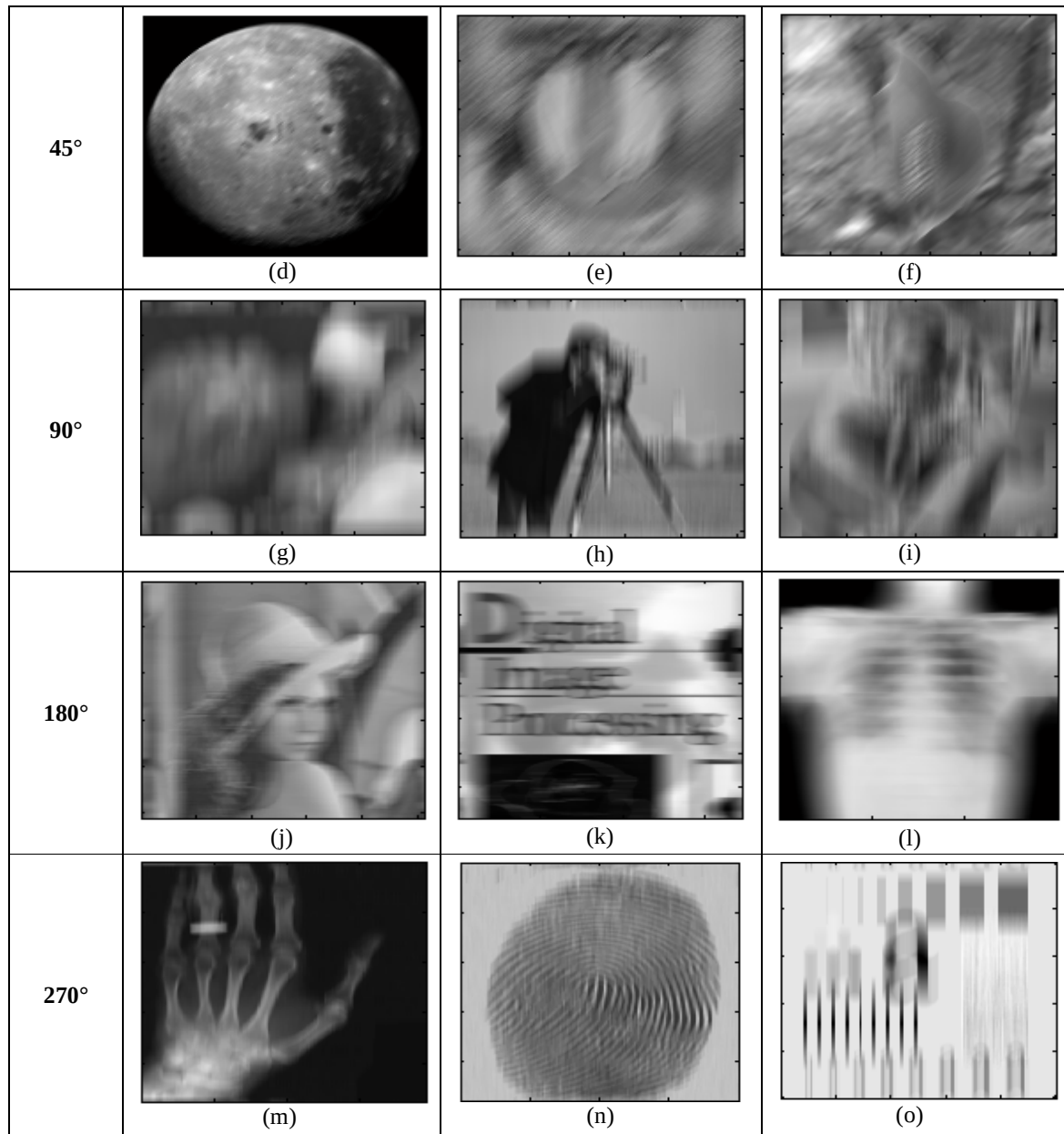


Figura 4.4. Imágenes con degradación por borrosidad (continuación).

4.3. Implementación de los algoritmos de restauración

La siguiente etapa de este trabajo consistió en restaurar las imágenes borrosas. Para esto se utilizaron los métodos de filtro de Wiener, filtro inverso y criterio de minimización de rizado, que fueron los que se plantearon utilizar desde el inicio de este proyecto de tesis, debido a que los dos primeros son los más usados actualmente y el de minimización de

rizado es una nueva propuesta(Mery & López 2003) interesante de evaluar. La implementación de estos algoritmos se muestra en el anexo B de este trabajo de tesis.

Un aspecto muy importante al realizar la restauración de imágenes son los parámetros de degradación o ruido, los cuales se desconocen. Para esto se hizo uso de la transformada de Hough (ver anexo B), que detecta líneas basada en el uso de la transformada de Fourier, para encontrar los posibles segmentos de recta existentes, los cuales indican los posibles ángulos de barrido de la imagen.

El primer paso consiste en pasar la imagen que se desea restaurar al dominio de la frecuencia, esto se logra aplicando la transformada de Fourier.

En la figura 4.5a se muestra una imagen con borrosidad de 45° y una distancia de 60 pixeles, la figura 4.5b muestra el resultado que se obtiene al aplicar la transformada de Fourier y la figura 4.5c es el resultado de aplicarle la transformada Hough.

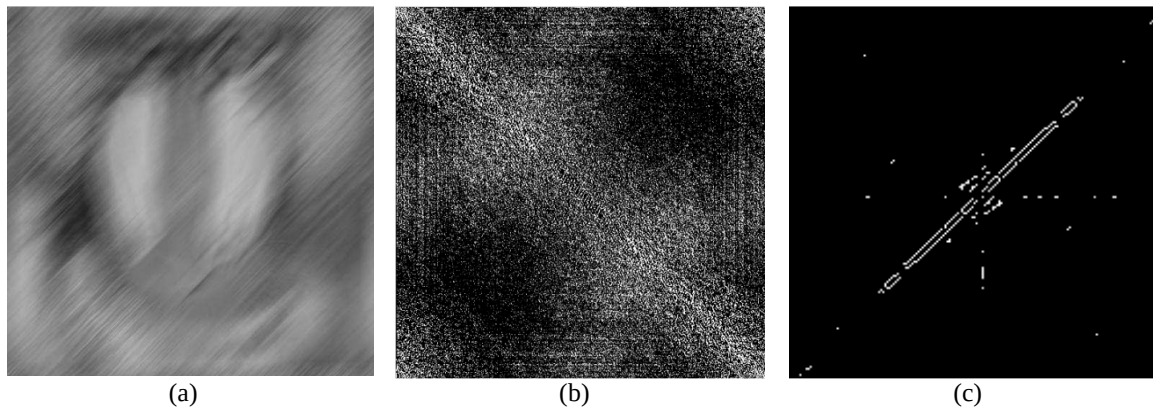


Figura 4.5. Resultados al aplicar la transformada de Hough. a) Imagen con degradación. b) Transformada de Fourier de la imagen borrosa. c) Líneas en la imagen.

Al aplicar la transformada Hough a una imagen se obtiene una estructura de datos que almacena cierta información sobre las líneas detectadas. Debido a que en este proceso se pueden obtener un gran número de líneas, se usó un criterio de selección, en función del número de intersecciones que obtuvo cada punto. Empleando este criterio se extrajeron 10 posibles ángulos de barrido de la imagen.

Adicionalmente, utilizando la transformada de Fourier junto con los ángulos se obtiene una aproximación de la distancia de borrosidad.

Con el ángulo y la distancia obtenida se realiza la restauración de la imagen y se calcula un índice de error respecto a la imagen original empleando diferencia de imágenes al cuadrado, el cual indica los parámetros para la restauración definitiva.

4.4. Realización de pruebas y mejoras

En este apartado se presentan las diferentes pruebas que se realizaron en este trabajo de tesis, para lo cual se utilizó una aplicación creada con la herramienta Matlab. Las pruebas se realizaron con las imágenes que se muestran en la figura 4.2 sólo que ya convertidas a escala de grises y las imágenes distorsionadas que se crearon a partir de ellas (Fig. 4.4), estas pruebas permitirán comparar los resultados de aplicar la restauración con los algoritmos filtro de Wiener, filtro inverso y criterio de minimización de rizado.

Una vez realizado el proceso de restauración por cualquiera de los filtros, a la imagen resultante se le aplicó el método de la mediana, para que los pixeles de intensidad muy diferente con respecto al resto de sus vecinos se homogenicen. Con la aplicación de este método se reduce el ruido sal y pimienta que se origina durante la restauración, el ruido sal y pimienta se presenta en una imagen como pixeles blancos y negros distribuidos por toda la imagen. Se decidió combinar el método de la mediana después de la aplicación de los filtros de restauración, porque de esta forma no se interfiere en la función de propagación que causa la degradación por borrosidad en la imagen, además es en la imagen restaurada donde es posible observar si se tiene presencia del ruido sal y pimienta. En la sección de evaluación y comparación de resultados se muestra qué tanto mejora la imagen restaurada si se le aplica el método de la mediana.

Para el criterio de minimización de rizado es importante aclarar que el ángulo debe de ser de 0° ya que el algoritmo está diseñado para restaurar con esa restricción, por lo que para este algoritmo se utilizan imágenes que cumplan con esa característica. De esta forma se podrá llevar a cabo la restauración y visualizar los resultados con los tres métodos propuestos.

En la figura 4.6 se muestran los resultados de aplicar los filtros de restauración digital ya mencionados a tres imágenes.

Algunas de las imágenes que se obtienen son de buena calidad, ya que es posible identificar visualmente la composición de la escena de la imagen, así como los elementos principales que aparecen en la imagen.

A simple vista, es posible observar que el filtro inverso tiene resultados más deficientes, mientras que los filtros de Wiener y el criterio de minimización de rizado muestran resultados similares, aunque en la imagen que tiene degradación de ángulo 0° y movimiento de 100 píxeles, visualmente se puede ver que la imagen que fue restaurada con el filtro de Wiener, tiene menos residuos de ruido.

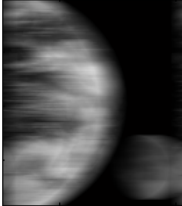
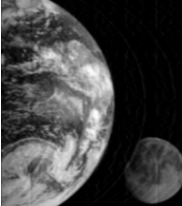
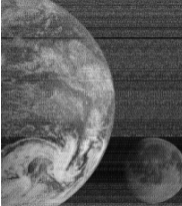
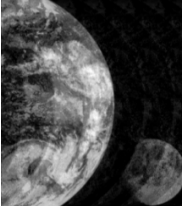
Ángulo Distancia	Imagen original a escala de grises	Imagen degradada	Filtro Wiener	Filtro Inverso	Minimización de rizado
0° 20 px					
0° 60 px					
0° 100 px					

Figura 4.6. Imágenes restauradas, donde el ángulo equivale a 0°.

Como ya se ha mencionado, una de las características del filtro de minimización de rizado es que el ángulo debe ser de 0°, sin embargo en la figura 4.7 se muestra un ejemplo del resultado que se obtiene al aplicar este filtro, para el caso de la imagen que presenta un ángulo de borrosidad de 90° y una distancia de 20 píxeles. Esta prueba se realiza con la finalidad de comprobar tal afirmación.

Distancia y Ángulo	Imagen original a escala de grises	Imagen degradada	Minimización de rizado
90° 20 pixeles			

Figura 4.7. Imagen restaurada con el filtro de minimización de rizado donde el ángulo es distinto a 0°.

En la figura 4.8 se representan las imágenes que se obtuvieron al aplicar el filtro de Wiener y el filtro inverso, para los casos donde el ángulo de borrosidad es distinto a 0°.

Distancia y Ángulo	Imagen original a escala de grises	Imagen degradada	Filtro Wiener	Filtro Inverso
45° 20 pixeles				
45° 60 pixeles				
45° 100 pixeles				

Figura 4.8. Imágenes restauradas, donde el ángulo es distinto a 0°.

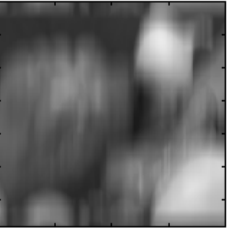
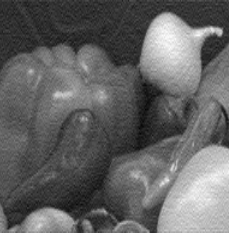
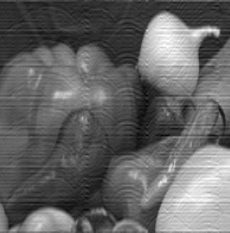
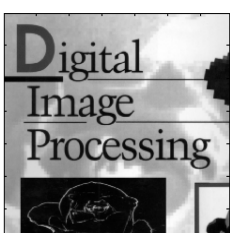
90° 20 píxeles				
90° 60 píxeles				
90° 100 píxeles				
180° 20 píxeles				
180° 60 píxeles				

Figura 4.8. Imágenes restauradas, donde el ángulo es distinto a 0° (continuación).

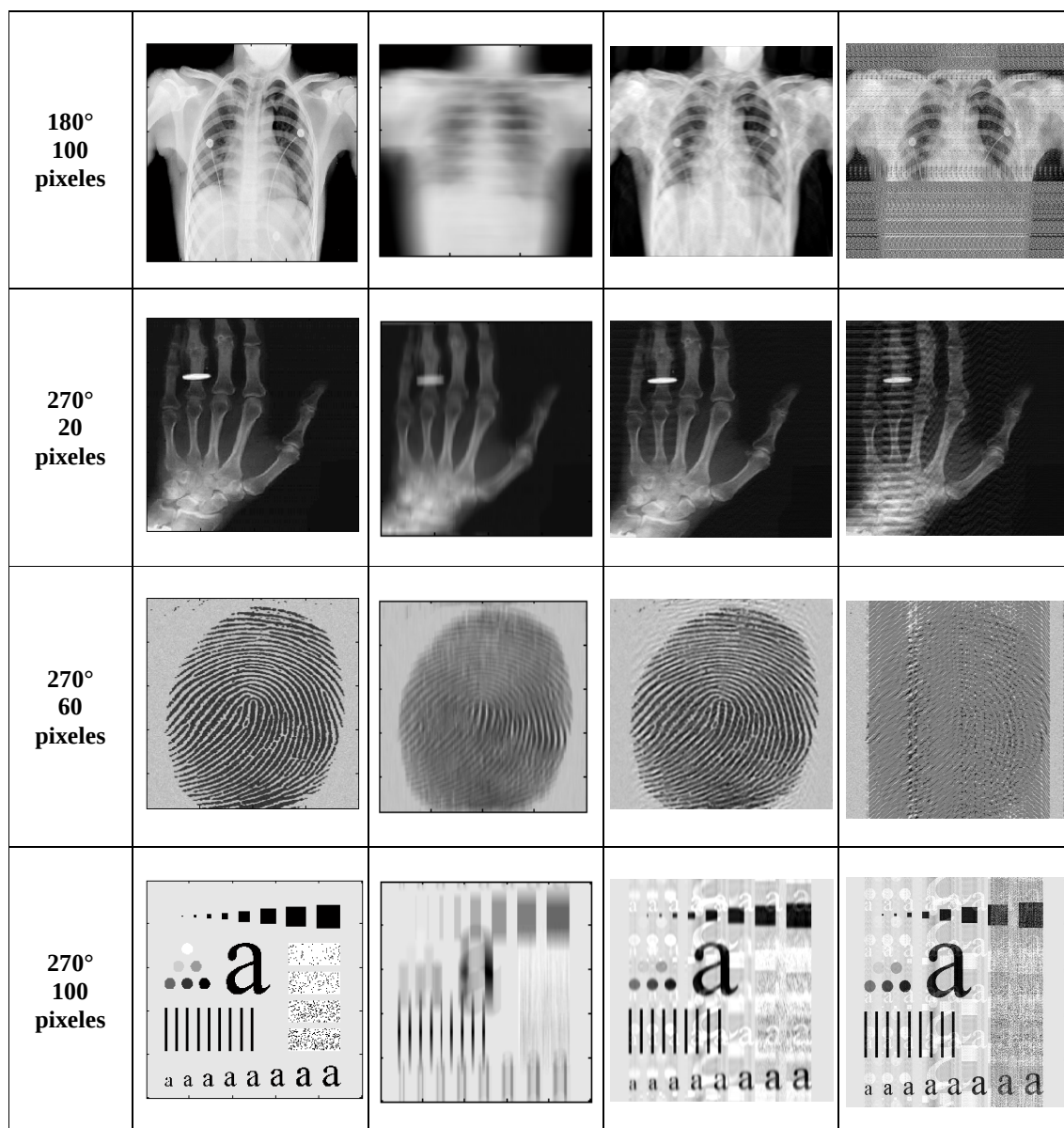


Figura 4.8. Imágenes restauradas, donde el ángulo es distinto a 0° (continuación).

El mejor método depende de la imagen y del tipo de ruido o degradación. El filtro de Wiener suaviza gradualmente la imagen, disminuyendo el brillo de la imagen, y mantiene los detalles presentes en la imagen. Es posible observar a simple vista que produce mejores resultados que el filtro inverso.

En la figura 4.9 se muestra un ejemplo de restauración con imágenes a color, sólo con los filtros de Wiener y filtro inverso; el método de minimización de rizado sólo es aplicable a imágenes a escala de grises, ya que las funciones de degradación son

específicamente a imágenes monocromáticas. Cabe resaltar que aunque no corresponde a las características de las imágenes que un principio se plantearon utilizar, se decidió mostrar estos resultados, porque se muestra que con la aplicación realizada es posible restaurar imágenes a color.

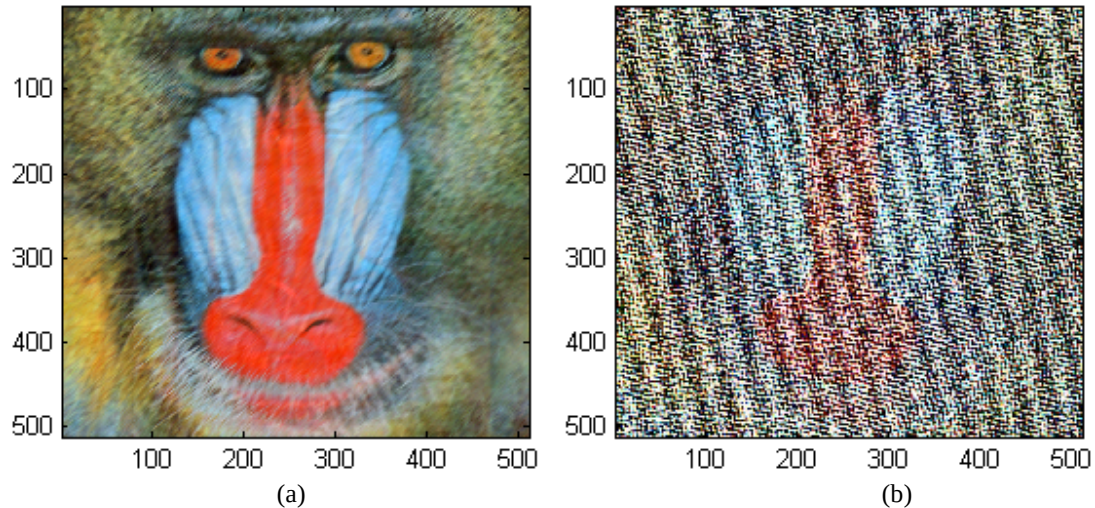


Figura 4.9. Imagen restaurada a color. a) Filtro Wiener. b) Filtro inverso.

En la figura 4.11 se muestra las pruebas realizadas con una sola imagen (Fig. 4.10) a la que se le aplicaron diferentes parámetros de ruido, esto para saber cómo es el comportamiento tanto de la degradación, como de los filtros de restauración.

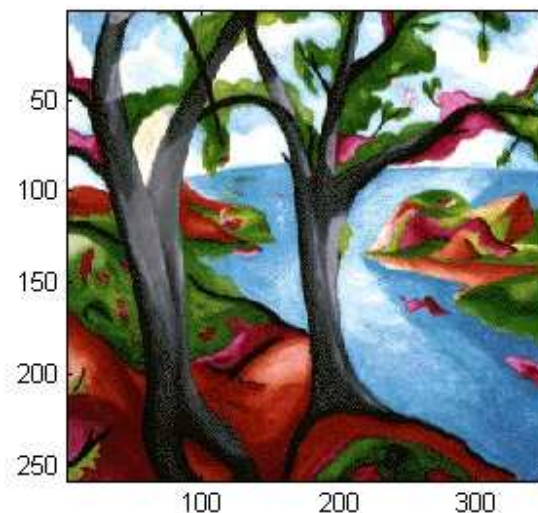


Figura 4.10. Imagen de prueba para evaluar con diferentes parámetros de ruido.

El comportamiento de la degradación se modifica conforme aumenta el valor de los parámetros de borrosidad, las imágenes dan un claro ejemplo de esta característica. Los

filtros también tienden a tener un comportamiento similar, en donde la degradación es más grande es posible observar que las escenas no son tan claras y siguen dejando ruido en la imagen restaurada.

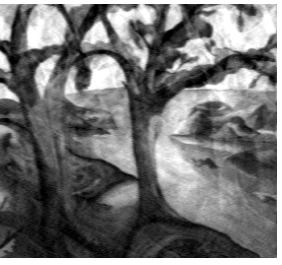
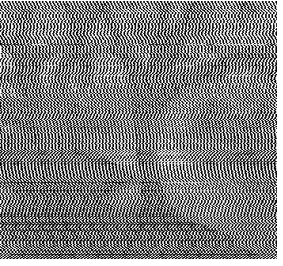
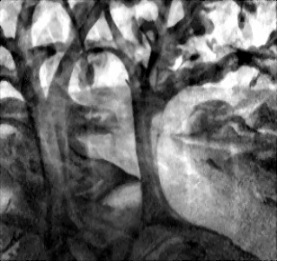
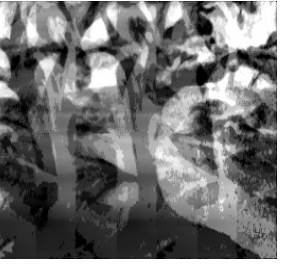
Parámetros de borrosidad Tipo de imagen	0 °,20 pixeles	0 °,60 pixeles	0°,100 pixeles
Imágenes con degradación			
Imágenes restauradas con filtro de Wiener			
Imágenes restauradas con filtro inverso			
Imágenes restauradas con criterio de minimización de rizado			

Figura 4.11. Resultados de restauración digital a una sola imagen de prueba.

Hasta este punto sólo se ha mencionado el uso de imágenes degradadas a partir de una imagen original sin degradación. Para probar la robustez de los algoritmos implementados, se utilizaron los algoritmos en imágenes que ya tenían una degradación de

las que se desconocían sus parámetros de ángulo y distancia de barrido. En la figura 4.12 se muestran los resultados de aplicar cada uno de los filtros de restauración a estas imágenes. En los resultados anteriores se tenía un previo conocimiento sobre la función de degradación, éste fue un motivo por el cual los filtros dieron buenos resultados, pero cuando se desconoce el comportamiento de esta función los resultados varían como en el filtro inverso que arroja resultados de mala calidad, debido a que con este filtro se debe conocer con precisión el fenómeno que causa la distorsión.

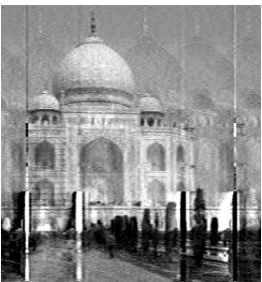
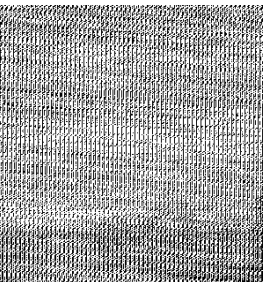
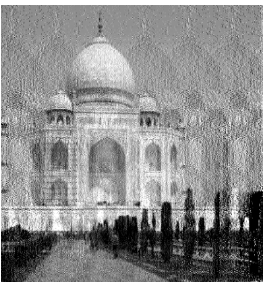
Imagen original borrosa	Imagen restaurada con filtro Wiener	Imagen restaurada con filtro inverso	Imagen restaurada con minimización de rizado
			
			

Figura 4.12. Imágenes restauradas con valores aproximados.

Las imágenes presentadas en la figura 4.13 fueron capturadas con una cámara digital modelo VLUU PL65 / SAMSUNG, estas imágenes tienen la característica de que ya contienen degradación. A cada una se le aplicaron los tres filtros de restauración, tomando en cuenta que se desconoce los parámetros que causan el ruido; los resultados obtenidos son poco eficientes ya que a simple vista no es fácil distinguir el contenido de las escenas en estas imágenes. Esto se explica con el teorema de “No Free Lunch” que indica que si cambiamos el campo de aplicación de un algoritmo, tiende a comportarse de manera diferente, ya que existen ambientes no controlados donde van a existir diversos factores de

ruido que van a propiciar que para el mismo tipo de problemas existen instancias considerablemente distintas que provocan distintos comportamientos de los algoritmos. Lo importante de mostrar estos resultados es que se manifiesta que tiene sentido estudiar bajo qué circunstancias un algoritmo se comporta mejor que otros al solucionar una clase determinada de problemas.

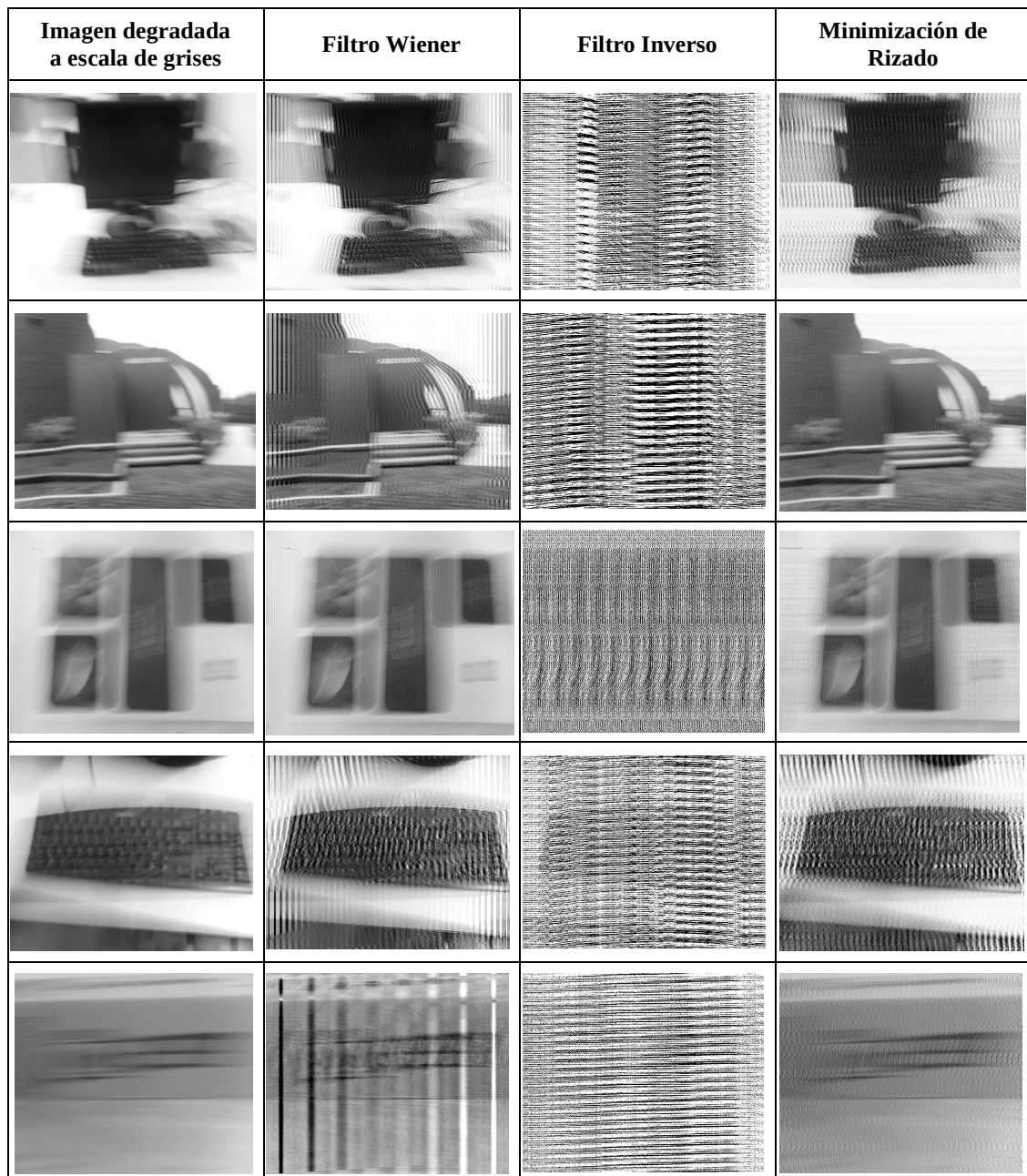


Figura 4.13. Imágenes restauradas con valores aproximados con los tres filtros propuestos.

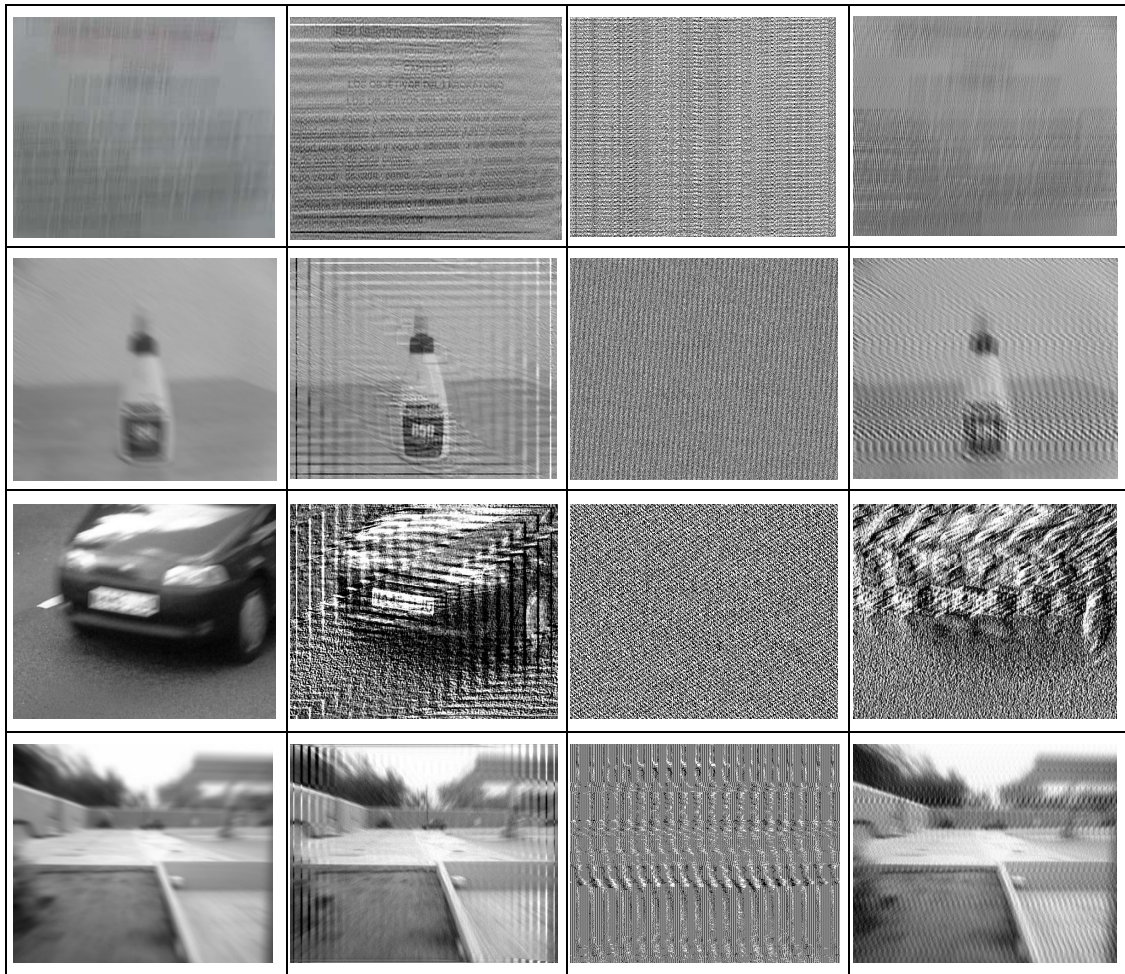


Figura 4.13. Imágenes restauradas con valores aproximados con los tres filtros propuestos (continuación).

4.5. Evaluación y comparación de resultados

Al realizar las pruebas se disponía de las imágenes originales que son imágenes sin ruido; a éstas se les añadió la degradación por borrosidad con la ayuda de los parámetros de ángulo y distancia, que varían en cada imagen; al aplicar la restauración se obtuvieron las imágenes resultantes. Al contar con la imagen original y la imagen resultante se procedió a realizar la comparación entre estas dos imágenes, con la finalidad de apreciar que tan buenos son los métodos de restauración que se aplicaron.

El primero de los métodos a utilizar para evaluar los resultados respecto a la imagen original es a través del cálculo del porcentaje de error. Una forma de conocer el porcentaje de error es aplicando la fórmula del cálculo del error relativo (Ec.20)

$$E_R = \left(\left| \frac{\text{valorExpeimental} - \text{valorTeórico}}{\text{valorTeórico}} \right| \times 100 \right) \quad (20)$$

Debido a que las imágenes de prueba elegidas son de gran dimensión, la explicación de los métodos de evaluación se realiza en una imagen de dimensión de 10 x 10 píxeles. En la figura 4.14 se puede observar la imagen original (Fig. 4.14a) y los resultados al aplicarle tanto ruido (Fig. 4.14b), como el proceso de restauración (Fig. 4.14c).

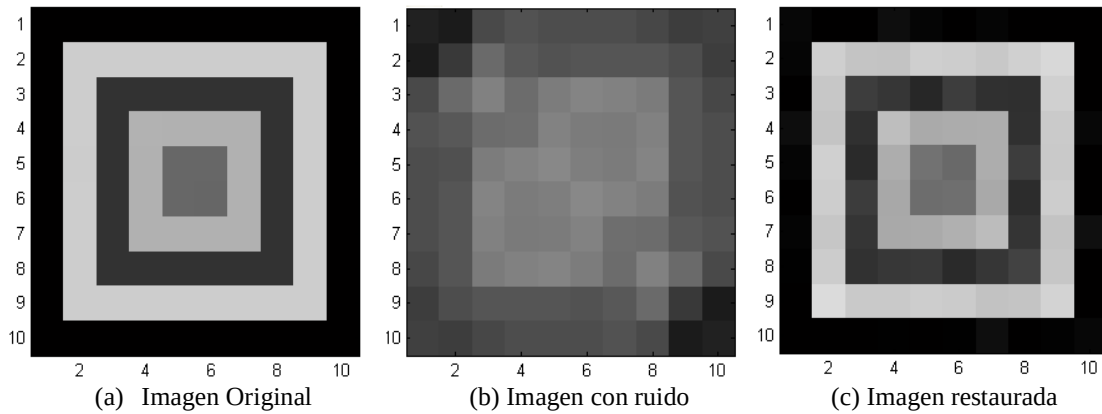


Figura 4.14. Imagen de prueba para métodos de evaluación.

Para fines de este trabajo se modificó la ecuación 14 experimentando en dos casos en particular.

Porcentaje de error a la máxima diferencia

Para el primer caso, la fórmula del error relativo queda expresada como lo muestra la ecuación 21.

$$E_R = \left(\left| \frac{\text{valorExperimental} - \text{valorTeórico}}{\text{valorMáximaDiferencia}} \right| \times 100\% \right) \quad (21)$$

Se toma el valor de la diferencia máxima como el 100%, puesto que al usar la fórmula del error relativo original existen procesos matemáticos donde el valor del píxel es 0 y la división entre 0 no está definida. Por eso se optó en tomar el valor de la máxima diferencia.

A continuación se explica un ejemplo de la aplicación de la fórmula anterior en la imagen de cuadros (Fig. 4.14a), en la figura 4.15 se muestran las matrices de datos correspondientes a la imagen original (Fig. 4.15a) y la imagen restaurada (Fig. 4.15b).

Matriz de datos de la imagen original									
0	0	0	0	0	0	0	0	0	0
0	204	204	204	204	204	204	204	204	0
0	204	50	50	50	50	50	50	204	0
0	204	50	177	176	176	176	50	204	0
0	203	50	176	103	103	176	50	204	0
0	204	50	176	104	102	176	50	204	0
0	204	50	176	176	176	176	50	204	0
0	204	50	50	50	50	50	50	204	0
0	204	204	204	204	204	204	204	204	0
0	0	0	0	0	0	0	0	0	0

(a)

Matriz de datos de la imagen restaurada									
9	5	2	9	7	10	1	0	12	0
5	204	194	197	206	192	183	223	191	14
0	198	54	60	45	51	66	35	233	3
11	203	51	184	167	170	183	49	195	0
0	209	42	169	106	97	164	63	211	0
8	194	63	180	112	118	181	38	206	0
14	175	70	168	157	168	189	44	189	9
0	202	47	43	67	38	55	67	204	0
14	188	218	192	205	200	200	187	203	12
0	20	0	0	10	8	20	1	21	0

(b)

Figura 4.15. Datos de la imagen cuadros. (a) Matriz de imagen original. (b) Matriz de imagen restaurada.

El primer paso a realizar es calcular la resta del valor experimental – valor teórico, para el primer caso sería $|0-9|=9$, y así se procede con todos los elementos hasta obtener todos los valores (Fig. 4.16).

9	5	2	9	7	10	1	0	12	0
5	0	10	7	2	12	21	19	13	14
0	6	4	10	5	1	16	15	29	3
11	1	1	7	9	6	7	1	9	0
0	6	8	7	3	6	12	13	7	0
8	10	13	4	8	16	5	12	2	0
14	29	20	8	19	8	13	6	15	9
0	2	3	7	17	12	5	17	0	0
14	16	14	12	1	4	4	17	1	12
0	20	0	0	10	8	20	1	21	0

Figura 4.16. Matriz de resultados de la resta de imágenes original y restaurada.

El paso número dos es obtener el valor de la máxima diferencia. En este ejemplo es 29. Cada uno de los valores de la matriz mostrada en la figura 4.15 se divide entre el valor de la diferencia máxima y se multiplica por 100. Los valores resultantes son expresados en porcentaje (Fig. 4.17).

31.0%	17.2%	6.9%	31.0%	24.1%	34.5%	3.4%	0.0%	41.4%	0.0%
17.2%	0.0%	34.5%	24.1%	6.9%	41.4%	72.4%	65.5%	44.8%	48.3%
0.0%	20.7%	13.8%	34.5%	17.2%	3.4%	55.2%	51.7%	100.0%	10.3%
37.9%	3.4%	3.4%	24.1%	31.0%	20.7%	24.1%	3.4%	31.0%	0.0%
0.0%	20.7%	27.6%	24.1%	10.3%	20.7%	41.4%	44.8%	24.1%	0.0%
27.6%	34.5%	44.8%	13.8%	27.6%	55.2%	17.2%	41.4%	6.9%	0.0%
48.3%	100.0%	69.0%	27.6%	65.5%	27.6%	44.8%	20.7%	51.7%	31.0%
0.0%	6.9%	10.3%	24.1%	58.6%	41.4%	17.2%	58.6%	0.0%	0.0%
48.3%	55.2%	48.3%	41.4%	3.4%	13.8%	13.8%	58.6%	3.4%	41.4%
0.0%	69.0%	0.0%	0.0%	34.5%	27.6%	69.0%	3.4%	72.4%	0.0%

Figura 4.17. Matriz de porcentajes de error.

Si se calcula el promedio de los porcentajes de error, se obtendrá el porcentaje de error de toda la imagen. La situación de este tipo de error es que dado que está en función de la diferencia máxima, el porcentaje de error es relativo a dicha diferencia. Así que si la diferencia máxima es muy pequeña, puede parecer que el error es muy grande, como en el caso del ejemplo que la diferencia es de veintinueve niveles de intensidad de gris y sin embargo aparece como 100% de error, y si la diferencia máxima fuera grande esos 29 niveles de intensidad de los píxeles podrían representar un porcentaje de error bajo.

Porcentaje de error si se toma el punto medio

El rango del nivel de intensidad de un píxel en una escala de grises empleando 8 bits, que generalmente es la escala utilizada, oscila de 0 a 255, siendo en total 256 valores, al dividir este dato entre 2 se obtiene como resultado 128. En la ecuación 22 ahora la división será entre el valor medio que es 128.

$$E_R = \left(\left| \frac{\text{valorExperimental} - \text{valorTeórico}}{128} \right| \times 100\% \right) \quad (22)$$

Se toman los mismos valores de la matriz de resultados de la resta de imágenes original y restaurada (Fig. 4.15) que se obtuvieron en el paso uno del método de porcentaje de error a la máxima diferencia, y se aplica la división y multiplicación como se muestra en la ecuación 20. El resultado se muestra en la figura 4.18.

7.0%	3.9%	1.6%	7.0%	5.5%	7.8%	0.8%	0.0%	9.4%	0.0%
3.9%	0.0%	7.8%	5.5%	1.6%	9.4%	16.4%	14.8%	10.2%	10.9%
0.0%	4.7%	3.1%	7.8%	3.9%	0.8%	12.5%	11.7%	22.7%	2.3%
8.6%	0.8%	0.8%	5.5%	7.0%	4.7%	5.5%	0.8%	7.0%	0.0%
0.0%	4.7%	6.3%	5.5%	2.3%	4.7%	9.4%	10.2%	5.5%	0.0%
6.3%	7.8%	10.2%	3.1%	6.3%	12.5%	3.9%	9.4%	1.6%	0.0%
10.9%	22.7%	15.6%	6.3%	14.8%	6.3%	10.2%	4.7%	11.7%	7.0%
0.0%	1.6%	2.3%	5.5%	13.3%	9.4%	3.9%	13.3%	0.0%	0.0%
10.9%	12.5%	10.9%	9.4%	0.8%	3.1%	3.1%	13.3%	0.8%	9.4%
0.0%	15.6%	0.0%	0.0%	7.8%	6.3%	15.6%	0.8%	16.4%	0.0%

Figura 4.18. Matriz de porcentajes de error a partir del punto medio.

En la figura 4.19 se exponen las gráficas con los valores de los porcentajes de error respecto al punto medio (128 pixeles), que representan diferentes medidas de desempeño tomadas a los tres métodos de restauración que fueron simulados.

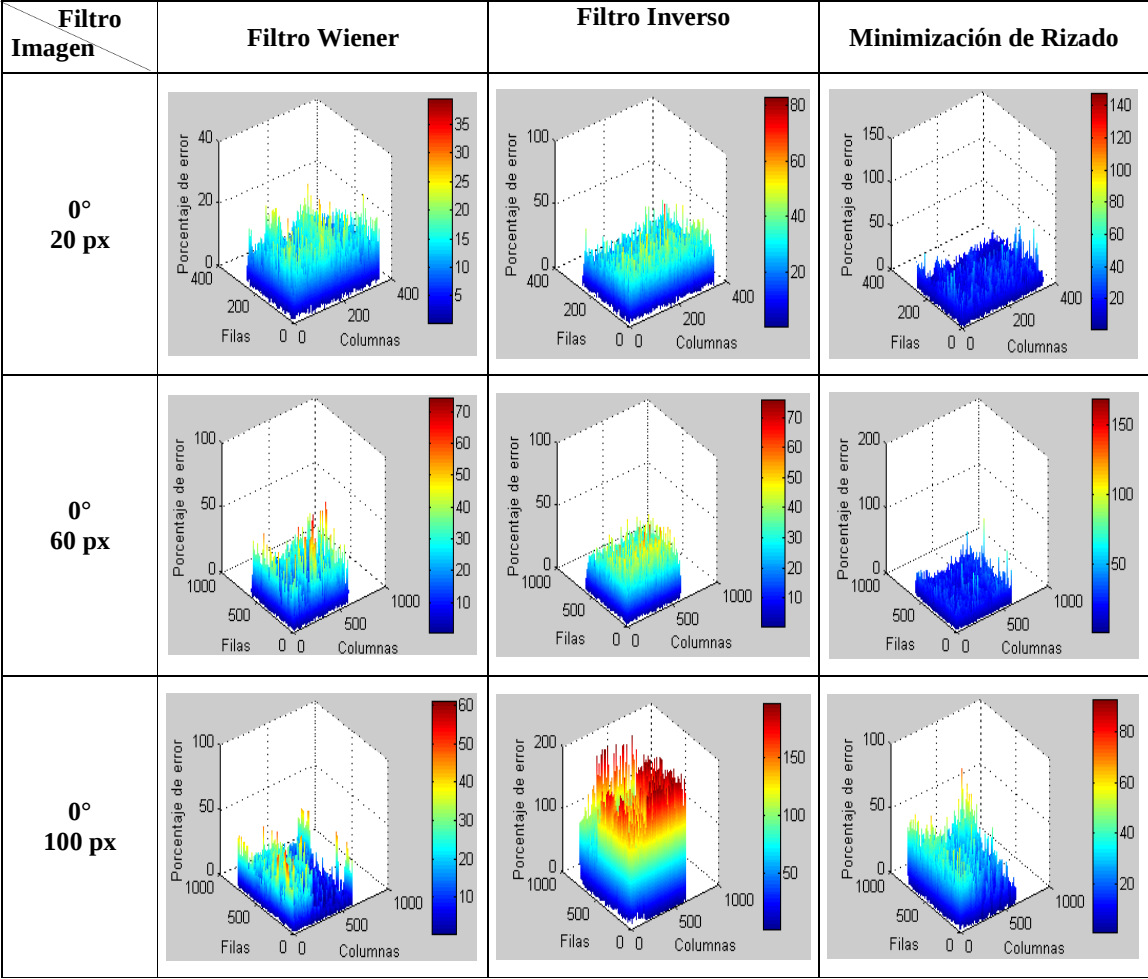


Figura 4.19. Gráficas de porcentajes de error, donde el ángulo equivale a 0°.

Los resultados indican que con el filtro de Wiener se obtienen mejores resultados, porque sus errores más altos en porcentajes son de 35%, 70%, 60%; mientras que en los casos del filtro inverso y el de minimización de rizado los resultados son variables, porque las gráficas muestran que donde se tienen distancias de 20 y 60 pixeles de borrosidad el filtro inverso supera al de rizado, únicamente porque este último maneja valores de mayor porcentaje de error, aunque sólo en algunos puntos de manera esporádica; no así en el filtro inverso, donde es fácil apreciar que los errores tienen una distribución más uniforme; esto se observa porque se tienen mayores áreas de color rojo. Las gráficas muestran que el filtro inverso es más deficiente cuando la distancia de borrosidad es mayor, ya que en la imagen donde se tiene una distancia de borrosidad de 100 pixeles, se aprecia que los resultados son muy significativos.

En la figura 4.20 se muestra la evaluación únicamente con los filtros de Wiener y el filtro inverso.

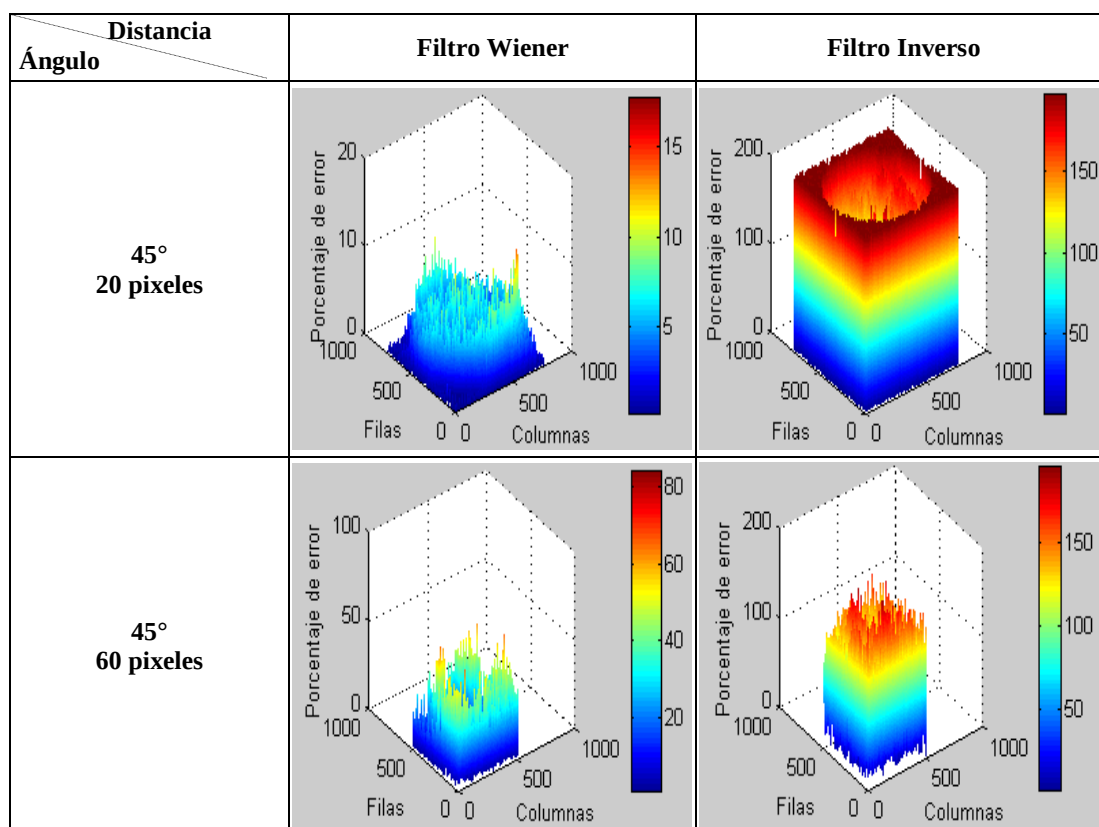


Figura 4.20. Gráficas de porcentajes de error, donde el ángulo es distinto a 0°.

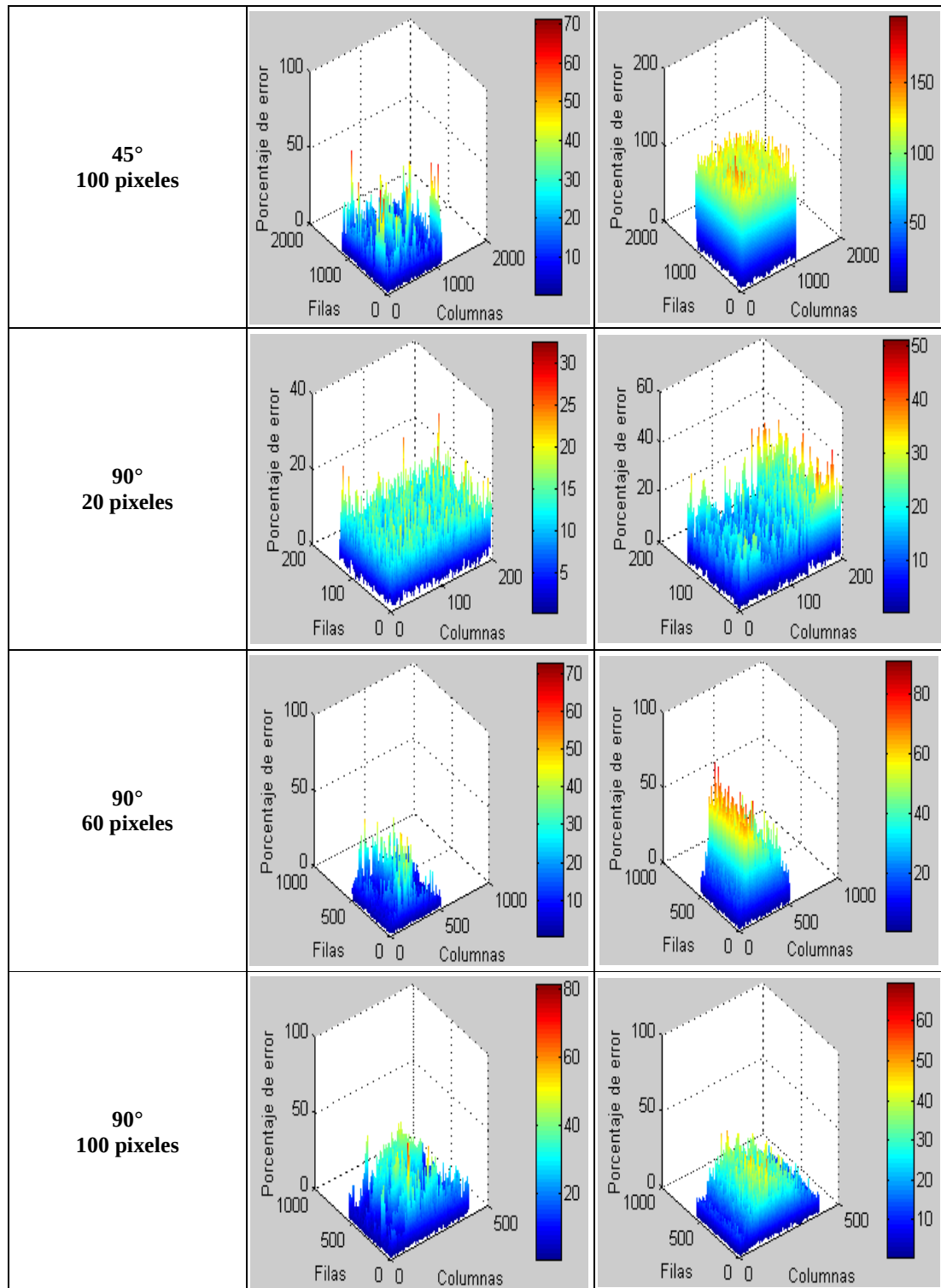


Figura 4.20. Gráficas de porcentajes de error, donde el ángulo es distinto a 0° (continuación).

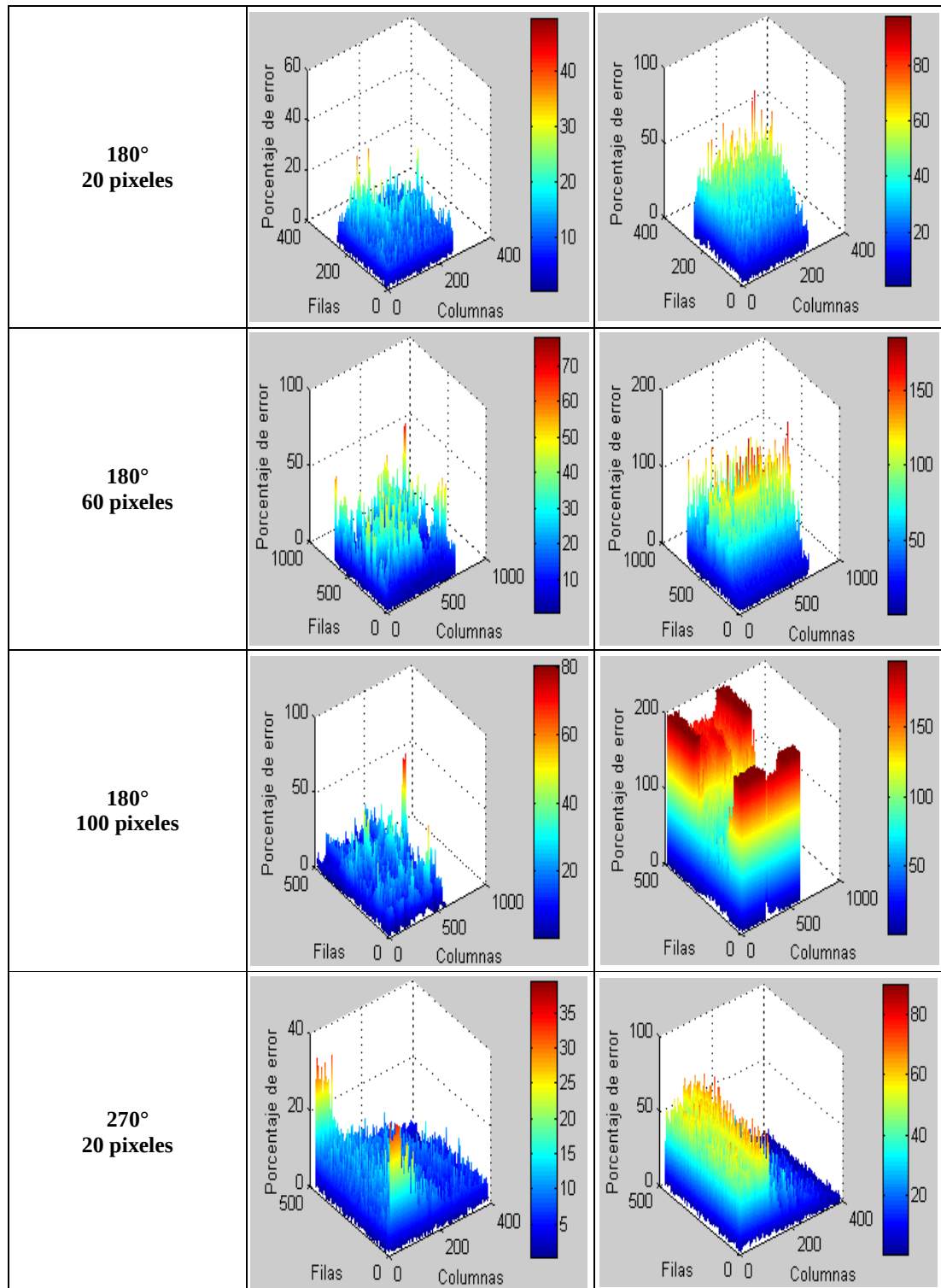


Figura 4.20. Gráficas de porcentajes de error, donde el ángulo es distinto a 0° (continuación).

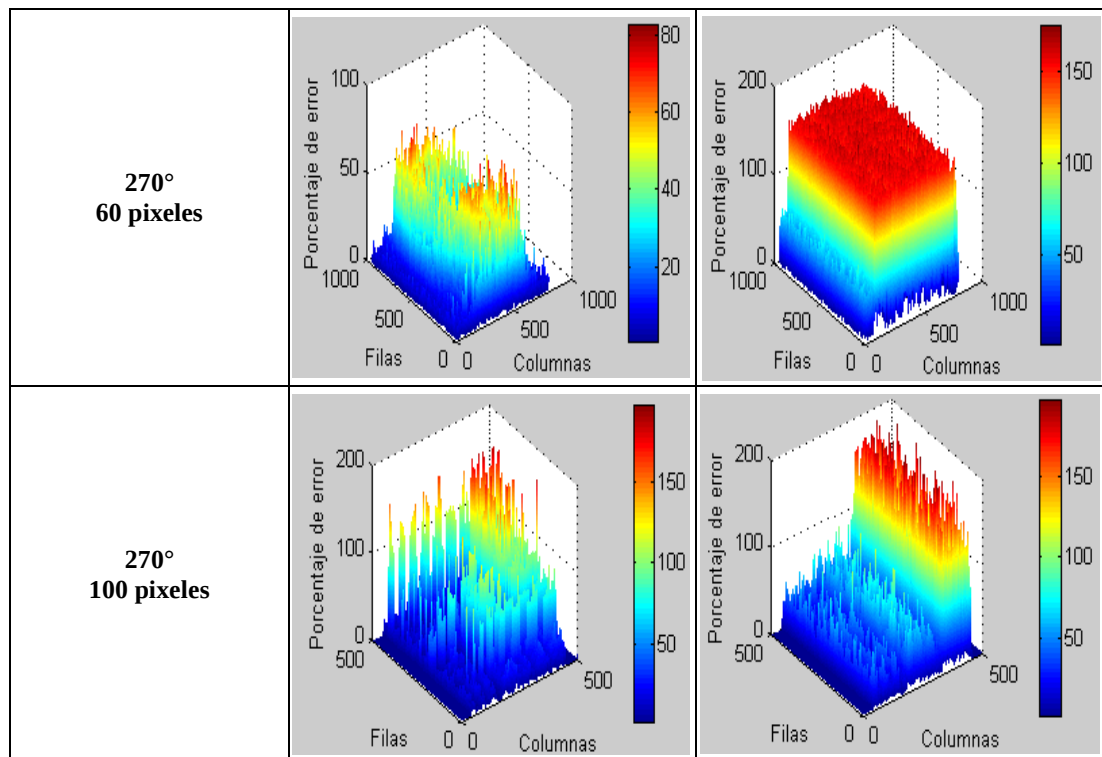


Figura 4.20. Gráficas de porcentajes de error, donde el ángulo es distinto a 0° (continuación).

Las gráficas son muy contundentes, ya que demuestran que el filtro de Wiener tiene menor porcentaje de error en sus imágenes que el filtro inverso.

Para comparar las imágenes restauradas por los métodos filtro de Wiener, filtro inverso y criterio de minimización de rizado se midió el tiempo de restauración, usando las funciones `tic` y `toc` que ofrece Matlab, las cuales trabajan juntas para medir el tiempo de ejecución empleado en cada método, medido en segundos.

En la tabla I, se muestran las mediciones de los resultados para las imágenes que se obtuvieron al aplicar la restauración mediante los tres filtros que se estudiaron, cuando el ángulo de borrosidad es de 0°.

Los tiempos que se tarda en realizar el procesamiento son aproximados, depende del equipo donde se lleven a cabo las pruebas, así como de los procesos que están siendo ejecutados al momento de realizar la restauración. Las pruebas que se realizaron fueron ejecutadas en el mismo equipo de cómputo especificado anteriormente y ejecutando los mismos procesos. Cabe mencionar que ninguno sobrepasa el minuto de procesamiento. El tiempo que se tarda en producir los resultados es más rápido gracias al software de

desarrollo que se utiliza. Como Matlab está destinado para trabajos con matrices, los cálculos son más rápidos.

Tabla I. Mediciones de imágenes con los tres filtros de restauración.

Variable	Parámetros de imagen	Filtro Wiener	Filtro Inverso	Minimización de Rizado
Tiempo	0° 20 pixeles	0.172 s	0.078 s	0.219 s
	0° 60 pixeles	0.719 s	0.172 s	1.125 s
	0° 100 pixeles	1.453 s	0.469 s	1.765 s
	0° 20 pixeles	5.73 %	13.1 %	7.98 %
Promedio de error sin mediana	0° 60 pixeles	7.6%	11.8 %	9.09 %
	0° 100 pixeles	5.03 %	48.7%	9.82 %
	0° 20 pixeles	6.61 %	9.35 %	7.87 %
	0° 60 pixeles	7.7 %	7.55 %	8.18 %
Promedio de error con mediana	0° 100 pixeles	5.1 %	23.6%	8.79 %
	Todas	6.12%	24.53%	8.96%
	Todas	6.47%	13.50%	8.28%
	Todas	6.47%	13.50%	8.28%

Se obtuvo el promedio de error antes y después de haberle aplicado el filtro mediana para cada imagen, posteriormente se hizo el cálculo del promedio general en ambos casos. Al hacer la evaluación y comparación se puede observar que cuando se aplica el filtro mediana al filtro inverso, el promedio de error disminuye de 24.53% a 13.50% , de igual manera si se utiliza el filtro del criterio de minimización de rizado, de 8.79% el promedio de error baja a 8.28%, mientras que al emplear el filtro de Wiener sucede lo contrario, el porcentaje de error aumenta de 6.12% a 6.47% aunque de manera global este filtro es más eficiente porque maneja los valores más pequeños de porcentajes de error. De manera que se llega a la conclusión de que al utilizar el filtro inverso y el criterio de minimización de rizado es conveniente aplicar el filtro de mediana, porque reduce el porcentaje de error.

En la tabla II, no aparece el filtro de minimización de rizado porque los resultados mostrados ya corresponden a imágenes donde el ángulo de borrosidad es distinto a 0° y recordemos que este filtro no aplica a esta característica. Al analizar la tabla II, se obtiene como resultado que el filtro inverso es el más eficiente en términos de tiempo, ya que superó en todas las pruebas al filtro de Wiener. La imagen de mayor dimensión es la que tiene como parámetros de borrosidad 45° , 100 pixeles, pero el procesamiento en el filtro inverso no influyó en el factor tiempo ya que sólo tardó 0.672 segundos, mientras que con el filtro de Wiener se tardó 2.922 segundos aunque esto no influyó para que fuera la imagen que más tiempo se tardó en realizar la restauración.

Tabla II. Tiempos aproximados de procesamiento del proceso de restauración.

Imagen	Filtro de Wiener	Filtro Inverso
45°, 20 px	2.140 s	0.984 s
45°, 60 pixeles	0.657 s	0.188 s
45°, 100 pixeles	2.922 s	0.672 s
90°, 20 pixeles	0.047 s	0.016 s
90°, 60 pixeles	0.625 s	0.203 s
90°, 100 pixeles	1.266 s	0.391 s
180°, 20 pixeles	0.141 s	0.031 s
180°, 60 pixeles	1.390 s	0.125 s
180°, 100 pixeles	0.750 s	0.469 s
270°, 20 pixeles	0.391 s	1.328 s
270°, 60 pixeles	3.562 s	0.125 s
270°, 100 pixeles	0.531 s	0.172 s

La tabla III muestra que el filtro de Wiener presenta porcentajes de error menores al 15%, mientras que el filtro inverso cuenta con valores cercanos al 100%. En el filtro inverso la imagen con parámetros de ruido de 45° y 20 pixeles de distancia de borrosidad es

la que presenta mayor porcentaje de error lo que significa que la restauración no es satisfactoria.

Tabla III. Promedio de los porcentajes de error con ángulo diferente de 0° .

Imagen	Filtro de Wiener sin mediana	Filtro de Wiener con mediana	Filtro Inverso sin mediana	Filtro Inverso con mediana
45°, 20 px	1.77 %	1.55%	90.4 %	73.9%
45°, 60 px	11.1 %	12.2%	76.5%	60%
45°, 100 px	5.76 %	5.76%	45.5 %	22.9%
90°, 20 px	6.39 %	2.93%	7.56 %	4.44%
90°, 60 px	4.93 %	4.68%	13.1%	8.3%
90°, 100 px	8.02 %	9.05%	9.62 %	10%
180°, 20 px	5.61 %	5.08%	16.2 %	9.9%
180°, 60 px	6.37 %	6.28%	22 %	19.8%
180°, 100 px	7.21 %	7.17%	61.6 %	40%
270°, 20 px	3.49 %	2.21%	10.7 %	7.25%
270°, 60 px	12.5 %	12.7%	76 %	65.9%
270°, 100 px	13.9 %	14.1%	21.1 %	14.9%

En la tabla III se puede ver que para los casos del filtro inverso, cuando se les aplica el método de la mediana los porcentajes de error disminuyen, excepto por la imagen con parámetros de borrosidad de 90° y 100 pixeles de distancia, donde el error sin filtro mediana es de 9.62% y crece a 10% donde ya es utilizada la mediana. Mientras, en el filtro de Wiener el comportamiento del porcentaje de error al aplicarle el método de mediana es variado, se tienen ejemplos donde el porcentaje se mantiene (1 caso), sube (4 casos) o baja (7 casos).

En la tabla IV se puede observar el promedio general de los porcentajes de error de la tabla III. El filtro de Wiener es el que dio el menor porcentaje de error y fue en los casos donde no se hizo uso del filtro de mediana.

Tabla IV. Promedio general de los porcentajes de error de la tabla III.

Filtro de Wiener sin mediana	Filtro de Wiener con mediana	Filtro Inverso sin mediana	Filtro Inverso con mediana
7.25%	6.98%	37.52%	28.11%

La tabla V muestra los resultados de la evaluación de las imágenes de la figura 4.10. Cabe resaltar que los resultados son a una misma imagen, sólo cambian los parámetros de error. La observación a las imágenes mostró que conforme se aumentaban los valores de los parámetros de degradación, las imágenes de resultado que se obtenían al aplicarle la restauración tenían más ruido. Los resultados numéricos del porcentaje de error sin filtro mediana muestran que efectivamente éste ha sido el comportamiento, ya que en el filtro de Wiener (ver tabla V) los porcentajes son 5.73%, 7.34% y 11.4% y en el filtro de minimización de rizado son 7.98%, 15.9% y 23.5%; es claro determinar que el porcentaje de error tiende a aumentar conforme los parámetros de ruido aumentan. El filtro inverso es más variable ya que no tiene un comportamiento definido, esto se aprecia en la tabla V donde de 13.1% pasa a 85.1 % y luego baja a 21.1%. Los porcentajes de error con filtro mediana reflejan que para el filtro inverso y el criterio de minimización de rizado los resultados disminuyen, pero no así en el filtro de Wiener donde aumentan, la razón es porque los datos de los pixeles vecinos en este filtro tienden a ser muy diferentes en ciertos casos.

Tabla V. Resultados de la evaluación a una sola imagen con diferentes parámetros de degradación.

Parámetros de degradación	Filtro de restauración	Tiempo de restauración	Porcentaje de error sin filtro mediana	Porcentaje de error con filtro mediana
0 °, 20 px	Filtro de Wiener	0.282 s	5.73 %	6.61 %
0 °, 60 px	Filtro de Wiener	0.297 s	7.34 %	7.59 %
0 °, 100 px	Filtro de Wiener	0.281 s	11.4 %	11.6 %
0 °, 20 px	Filtro Inverso	0.062 s	13.1 %	9.35 %
0 °, 60 px	Filtro Inverso	0.063 s	85.1 %	66.3 %
0 °, 100 px	Filtro Inverso	0.062 s	21.1%	12.8 %
0 °, 20 px	Minimización de rizado	0.5 s	7.98 %	7.87 %
0 °, 60 px	Minimización de rizado	0.344 s	15.9 %	15.5 %
0 °, 100 px	Minimización de rizado	0.328 s	23.5 %	22.9 %

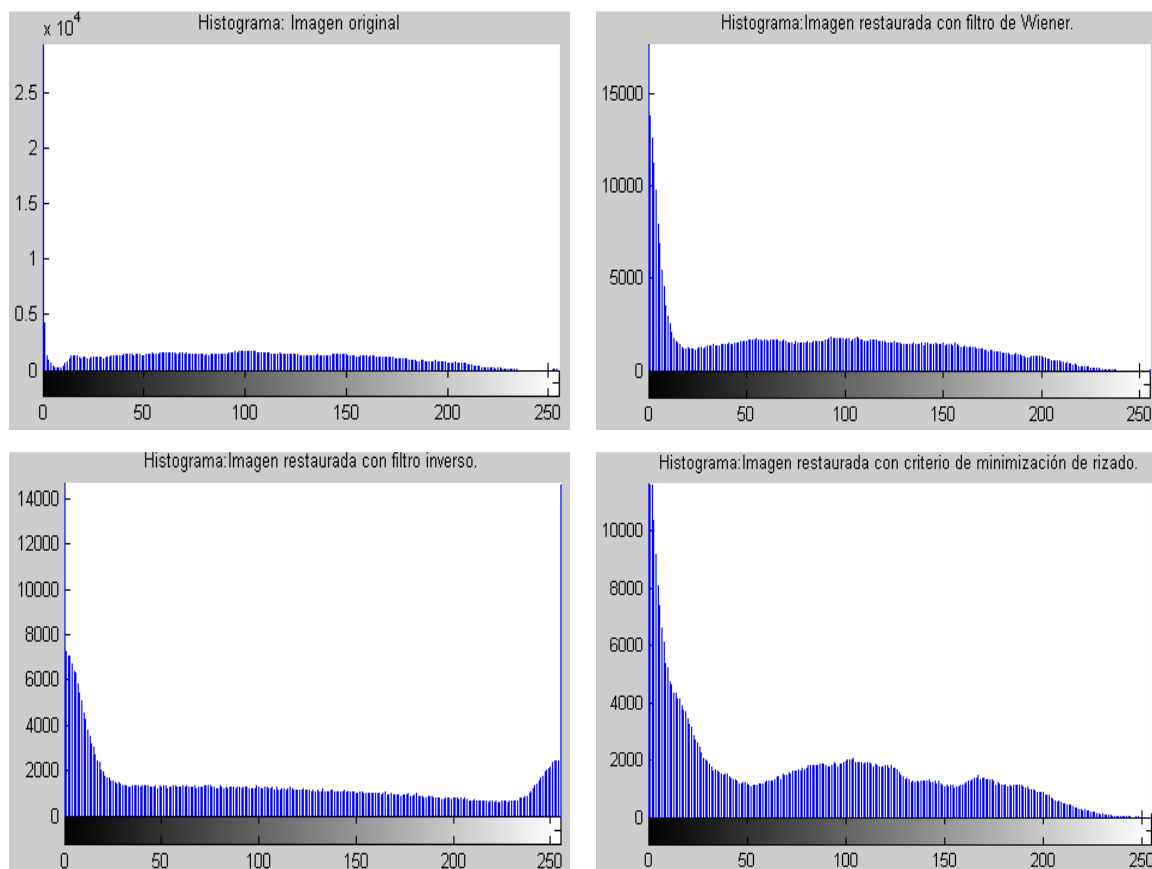
Como la tabla VI resalta, el filtro de Wiener es el que tiene el promedio general de error más bajo, 8.16%, y fue donde no se aplicó el filtro mediana.

Tabla VI. Promedio general de los porcentajes de error de la tabla XIII.

Método	Filtro de Wiener	Filtro Inverso	Criterio de minimización de rizado
sin filtro mediana	8.16%	39.77%	15.79%
con filtro mediana	8.60%	29.48%	15.42%

Se utilizó el histograma para visualizar la distribución de intensidades de la imagen, así fue más fácil apreciar que tan diferente es la combinación en cada método de restauración respecto de la imagen original. En la figura 4.21 se muestra un ejemplo de la imagen de 0° 100 pixeles, para cada uno de los filtros aplicados.

El rango de intensidades difiere en cada filtro, pero en apariencia la forma sigue siendo la misma, es claro determinar que los niveles aumentan en las imágenes restauradas. Esto indica que cuando existe una acumulación en los niveles de gris bajo, entonces la imagen es relativamente oscura y si el histograma tiene una concentración de pixeles con un alto nivel de gris, entonces la imagen es relativamente luminosa.

**Figura 4.21.** Histogramas de imagen con los distintos métodos de restauración.

Tomando en cuenta las pruebas que se realizaron y los resultados obtenidos, finalmente se determinó que el filtro que es mejor para la restauración de imágenes digitales con ruido por borrosidad es el filtro de Wiener, aunque existe una mejora notable en las imágenes restauradas obtenidas al aplicar los demás filtros propuestos. Dependiendo de la finalidad que se tiene para hacer uso de las imágenes en las diversas disciplinas donde es aplicado el proceso de restauración, se recomienda que cuando se conocen los parámetros que causan borrosidad se utilice el filtro Wiener, pero si los parámetros son desconocidos es conveniente utilizar el criterio de minimización de rizado aplicando el filtro mediana. En el siguiente capítulo se dará un análisis más detallado de las conclusiones finales de este trabajo de tesis, así como futuras líneas de investigación.

CAPÍTULO 5. CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS

Conclusiones

En este trabajo se presentaron diferentes técnicas que permiten realizar la restauración de imágenes digitales, de las cuales tres se llevaron a cabo, por ser las más conocidas y porque presentan las características de no requerir tiempo excesivo de procesamiento, además de que son utilizables para los casos donde está presente la degradación por borrosidad, que es uno de los puntos fundamentales de este trabajo. Las técnicas son: el filtro de Wiener, criterio de minimización de rizado y el filtro inverso.

El filtro de Wiener tuvo buenos resultados donde se tenía un conocimiento previo de los parámetros de degradación que causan la borrosidad.

El criterio de minimización de rizado tiene la desventaja de que sólo puede ser aplicado cuando el ángulo de borrosidad es 0° , es decir que el movimiento horizontal sea puro. Los resultados demostraron que la imagen obtenida después de aplicar la restauración

mejora de forma considerable, ya que de forma visual permite percibir las formas de los objetos de la imagen.

El filtro inverso es utilizado cuando el ruido es muy pequeño, si se aplica cuando la degradación es muy grande no muestran buenos resultados. Además, al evaluar con imágenes donde se desconocían los valores de borrosidad que causan la degradación, los resultados no fueron satisfactorios.

Al realizar los cálculos de los porcentajes de error de las imágenes donde se utilizaron los tres filtros de restauración y en los cuales al resultado se le aplicó el filtro mediana se obtuvo que el porcentaje de error para el filtro de Wiener fue de 6.47%, el 13.50% de error para el filtro inverso y el 8.28% de error para el criterio de minimización de rizado; sin aplicar el filtro mediana se obtuvo el 6.12% de error para el filtro de Wiener, mientras que al aplicar el filtro inverso el porcentaje de error fue de 24.53% y finalmente con el criterio de minimización de rizado se tuvo el 8.96%, de lo cual se concluye que el filtro que da mejores resultados es el filtro de Wiener por tener un porcentaje de error menor, además si se aplica el proceso de restauración usando el filtro inverso o el criterio de minimización de rizado es conveniente aplicar al resultado el filtro de mediana debido a que se ve una reducción en el porcentaje de error.

Para llevar a cabo los algoritmos es importante contar con un software potente que sea capaz de realizar diversas operaciones a las imágenes. Matlab es un software que reúne las características fundamentales para el trabajo con imágenes, permite crear aplicaciones donde el usuario define los elementos a ocupar, así como determinar el código de cada uno de ellos, para esto cuenta con un propio lenguaje de programación. La ejecución de las aplicaciones es muy rápida, además de no demorar en la tarea de presentación de resultados.

Recomendaciones

- Cuando se restaura una imagen con el filtro inverso y el criterio de minimización de rizado, aplicar el filtrado de mediana.
- Aplicar el filtro de minimización de rizado cuando se conoce que existe un movimiento horizontal de 0° .

- Utilizar el filtro inverso cuando se tiene un conocimiento aproximado de los parámetros que causan la borrosidad, además de tener en cuenta que se tendrán mejores resultados si la borrosidad presentada en la imagen es pequeña.

Trabajos futuros

El tema de la restauración de imágenes digitales en los últimos años ha estado en constante innovación, ya que cada día se cuenta con nuevos adelantos tecnológicos en cuanto a sistemas de captura y procesamiento de imágenes, por ello respecto a las líneas de trabajo a seguir se recomienda que se explore más el tema, aportando nuevas técnicas de restauración o mejorando las ya existentes.

En este trabajo únicamente se abordó el problema de ruido con movimiento uniforme en distancia y ángulo, pero cuando el movimiento presenta variación en distancia y ángulo el problema se convierte en un problema más complejo digno de ser estudiado. El trabajo a futuro sería estudiar la restauración para los diferentes tipos de degradación, como son: el ruido impulsivo, ruido gaussiano y ruido uniforme.

Resultaría importante aplicar la restauración en imágenes adquiridas en tiempo real, como por ejemplo, una cámara montada sobre un robot que se va moviendo.

Aplicación de los métodos de restauración a un área específica, por ejemplo control de placas de vehículos, trabajo con imágenes médicas o bien en el área biológica para monitoreo de cierto tipo de animales.

Extender el método de minimización de rizado para diferentes tipos de movimiento y no excluirlo para los casos donde sólo sea movimiento horizontal o vertical puro.

Estudiar otras causas que hacen que los filtros: inverso y minimización de rizado no presenten buenos resultados, asimismo, darle solución.

Implementar los métodos en otros lenguajes de programación donde también es factible trabajar con imágenes y ver el comportamiento que éstos presentan, por ejemplo, Java, Microsoft Visual C#, entre otros.

ANEXO A. MANUAL DE USUARIO

En este anexo se da una breve descripción de la aplicación creada en esta tesis, para realizar el estudio sobre la restauración digital de imágenes. La aplicación es de gran ayuda porque permite evaluar los algoritmos: filtro de Wiener, filtro inverso y el filtro del criterio de minimización de rizado.

La aplicación requiere tener instalado el software de Matlab *release* 14, versión 7.0. En el CD adjunto a este trabajo de tesis se encuentra la carpeta con el nombre de “Aplicación”, ésta se copia en la ruta de Matlab donde se guardan los proyectos realizados (C:\MATLAB7\work es la carpeta por defecto).

El primer paso consiste en ejecutar el programa de Matlab, posteriormente se cambia la ruta de trabajo, tal y como lo especifica la figura A.1.

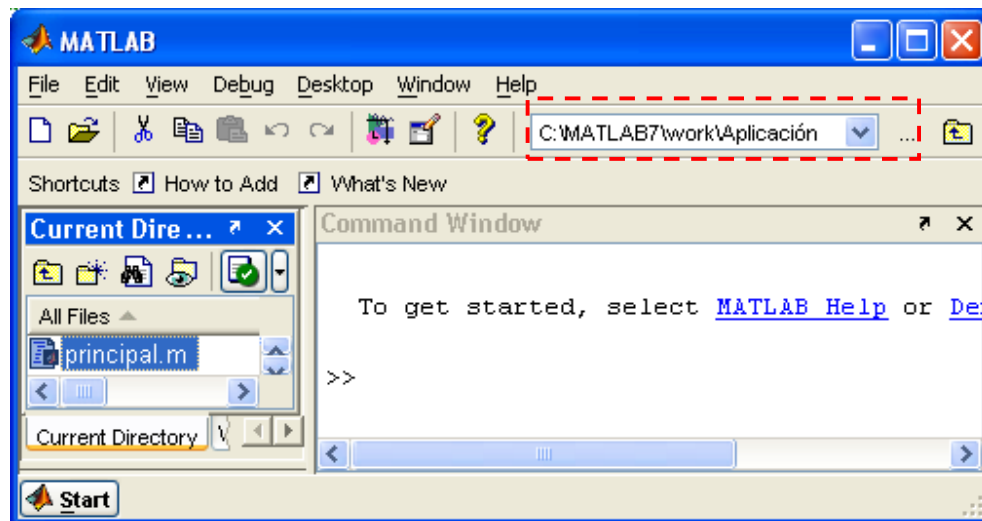


Figura A.1. Cambio de ruta, para poder llevar a cabo la ejecución de la aplicación.

En la parte izquierda de la ventana de Matlab, se muestran todos los archivos que contiene la carpeta, abrir el que tiene el nombre de `principal.m`, esto se logra al dar doble clic sobre él. En la nueva ventana que se muestra (Fig. A.2), presionar la tecla F5 para ejecutar el código fuente de este archivo, de esta manera visualizar de forma gráfica la ventana principal.

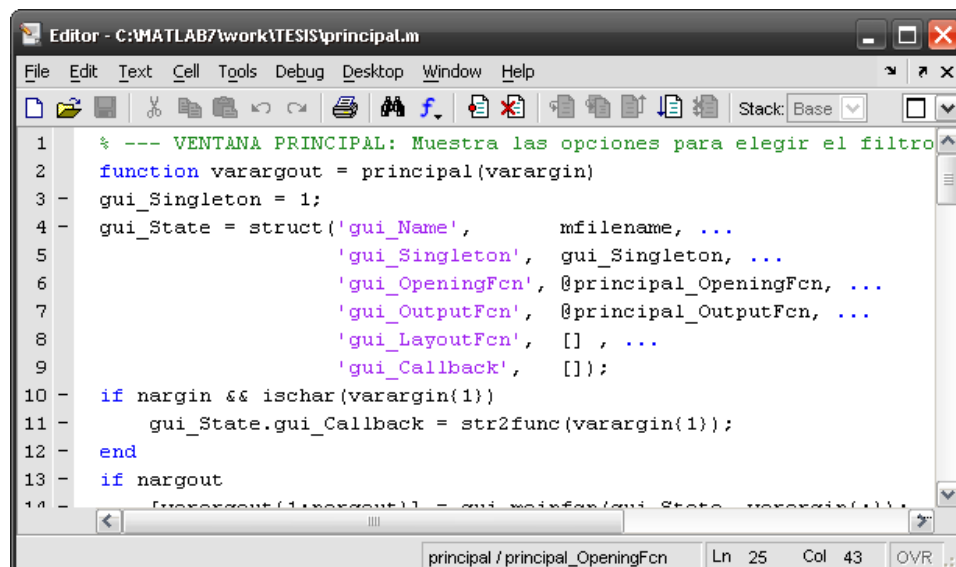


Figura A.2. Archivo `principal.m`.

La interfaz de la ventana principal (Fig. A.3), contiene las opciones de los tres filtros que se utilizan para restaurar las imágenes borrosas y el botón para salir de la

aplicación. Además se muestra el nombre de la institución donde se creó la aplicación, el título del tema de tesis, los logotipos y el nombre de la autora.



Figura A.3. Ventana de bienvenida de la aplicación.

En la tabla VII se muestran las funciones que realizan cada una de las herramientas que presenta la ventana principal.

Tabla VII. Funciones y herramientas de la ventana principal.

Herramienta	Función
Filtro de Wiener	Abre la ventana del filtro de Wiener.
Filtro Inverso	Permite abrir la ventana del filtro inverso.
Minimización de rizado	Abre la ventana del filtro minimización de rizado.
	Sirve para salir de la aplicación

Al seleccionar el botón filtro de Wiener se muestra la ventana de este filtro, como se visualiza en la figura A.4.



Figura A.4. Ventana del filtro de Wiener.

Cuando se da clic en el botón Filtro Inverso se muestra la ventana de este filtro (Fig. A.5).



Figura A.5. Ventana del filtro inverso.

Para acceder a la interfaz del filtro de minimización de rizado, dar clic en el botón Minimización de rizado. En la figura A.6 se presenta el contenido de la ventana referente a este filtro.



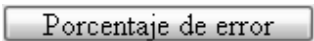
Figura A.6. Ventana del filtro del criterio de minimización de rizado.

Los filtros son muy similares por lo tanto, las secciones de las ventanas no presentan gran variación, las funciones de las herramientas se detallan en la tabla VIII.

Tabla VIII. Herramientas de las ventanas de los filtros.

Icono	Función
	Sirve para regresar a la ventana principal.
	Para guardar una imagen, depende de la sección donde se encuentre, ésta lleva a una nueva ventana donde se busca la ruta en la cual se va a guardar la imagen.
	Se utiliza para abrir una imagen.
☒ Convertir a escala de grises	Al seleccionar la casilla, convierte una imagen a escala de grises.

Tabla VIII. Herramientas de las ventanas de los filtros (continuación).

	Limpia el área de gráficos y de datos.
	Los controles deslizantes permiten seleccionar un valor moviendo el indicador.
	Muestra el anterior de los ángulos calculados.
	Muestra el posterior de los ángulos calculados.
Distancia: Ángulo:	En las cajas de texto se escriben los valores correspondientes a distancia y ángulo, que puede ser para agregar ruido o bien ser valores para restaurar la imagen.
	Calcula el ángulo de borrosidad.
	Calcula la distancia de borrosidad.
	Sirve para agregar ruido a la imagen.
	Sirve para restaurar una imagen donde se conocen los parámetros de borrosidad.
	Sirve para restaurar una imagen donde se desconoce los parámetros que causan la borrosidad.
☒ Aplicar filtro mediana	Al seleccionar la casilla, aplica el filtro mediana a la imagen restaurada.
	Permite obtener el histograma de la imagen original y la imagen restaurada.
	Calcula la diferencia de cuadrados que existe entre la imagen original y restaurada.
	Se utiliza para obtener el porcentaje de error entre imagen original e imagen restaurada.
	Muestra los resultados de tiempo, porcentaje de error y diferencia de cuadrados.

Para entender bien el funcionamiento, se mostrará un ejemplo utilizando el filtro de Wiener. El primer paso es cargar la imagen de prueba, esto se logra al dar clic en el botón **Abrir imagen**, se desplegará una ventana donde el usuario elegirá la ruta del archivo de imagen que desea abrir, además se especifica el tipo de imagen a utilizar. En la figura A.7 se visualiza que el archivo que se decide abrir es “Trees” con formato .jpg.

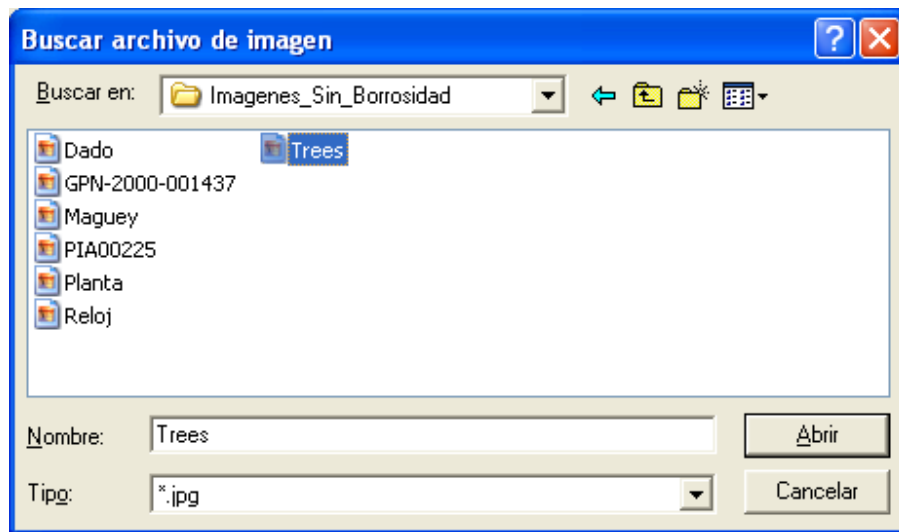


Figura A.7. Ventana para la búsqueda de archivo que se desea abrir.

Cuando se abre la imagen, se muestra en la ventana, visualizando las dimensiones que tiene (Fig. A.8).



Figura A.8. Visualización de la imagen, en la aplicación.

Como ya se han explicado las imágenes a utilizar necesitan estar en formato a escala de grises, para ello se selecciona la casilla **Convertir a escala de**

grises, en la figura A.9 se muestra el ejemplo de la imagen con la cual se está trabajando.



Figura A.9. Resultado de la conversión de la imagen “Trees” a escala de grises.

Después de convertirla a imagen monocromática se agregan los parámetros de ruido, en este caso la distancia será de 51 píxeles y el ángulo de 36°. Una vez indicados estos parámetros se da clic en el botón **Agregar ruido**. Esta parte tiene la opción de guardar la imagen borrosa, para hacerlo se da clic en el icono de guardar, se especifica el nombre del archivo con su respectiva extensión. El resultado de la imagen borrosa de la imagen “Trees” se visualiza en la figura A.10.



Figura A.10. Resultado de la aplicación de ruido a la imagen.

Si se desea cambiar los parámetros, se limpia el gráfico y se especifican nuevamente.

Para restaurar la imagen borrosa se da clic en el botón Restaurar (parámetros conocidos), toma los valores indicados para agregar ruido a la imagen, esta parte también cuenta con la opción para guardar la imagen restaurada, además al resultado se le puede aplicar el filtrado de mediana (Fig. A.11).

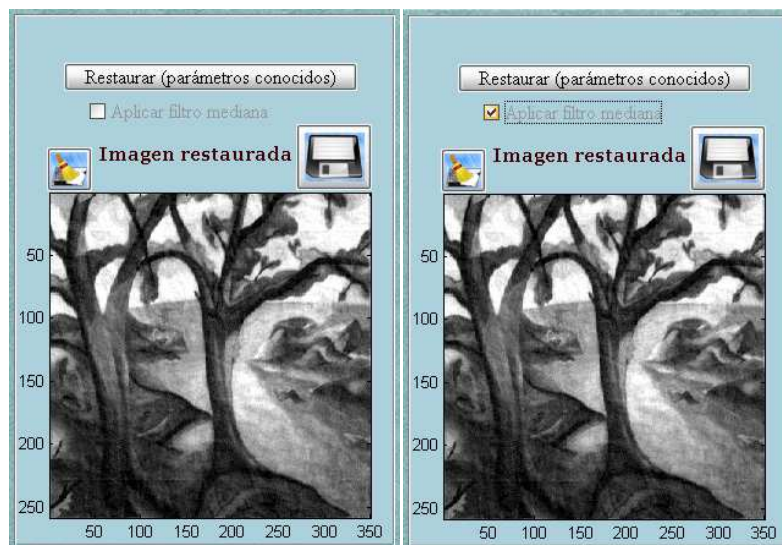


Figura A.11. Resultado de la restauración de la imagen borrosa. (a) Sin la aplicación del filtro mediana. (b) Con la aplicación del filtro mediana.

Los histogramas de la imagen original y restaurada se pueden obtener al ir al apartado de mediciones, sólo se da clic en el botón Histograma. En la figura A.12 se muestra los resultados de la imagen “Trees”.

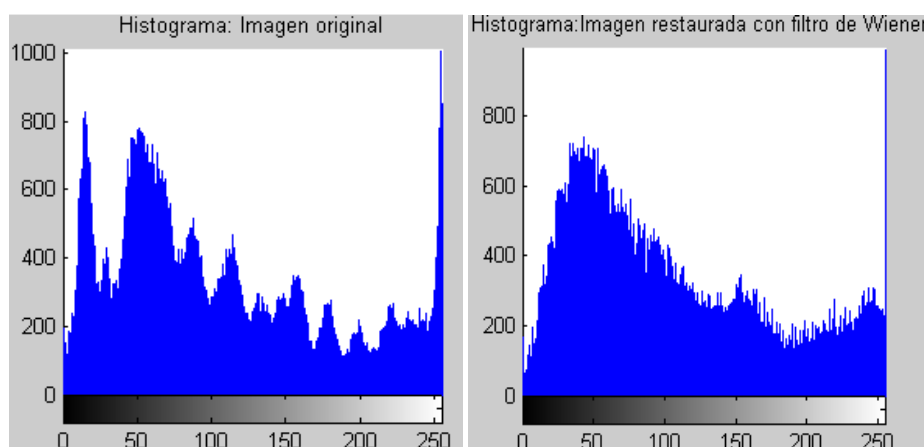


Figura A.12. Histograma de la imagen original y restaurada.

El porcentaje de error de la imagen se calcula al pulsar el botón % de error, el resultado se muestra en una ventana (Fig. A.13).



Figura A.13. Resultado del % de error de la imagen.

Al mismo tiempo se despliegan las gráficas de porcentaje de error (Fig. A.14), evaluadas con respecto al punto medio que es 128 píxeles.

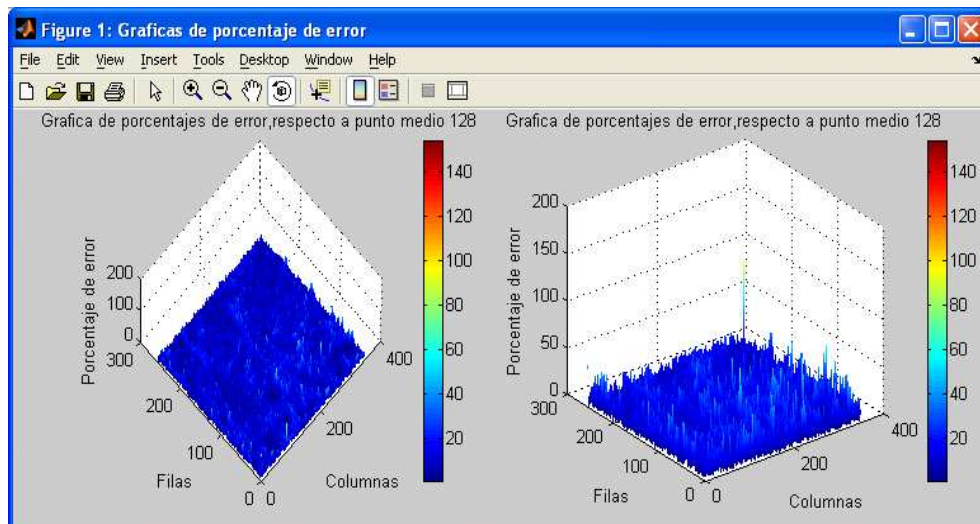


Figura A.14. Gráficas de los resultados de porcentajes de error, respecto a punto medio (128 píxeles).

Los tiempos que tarda el proceso de restauración son desplegados al seleccionar el botón **Mostrar resultados** que se encuentra en la parte de mediciones. En la figura A. 15 se visualiza tales resultados.

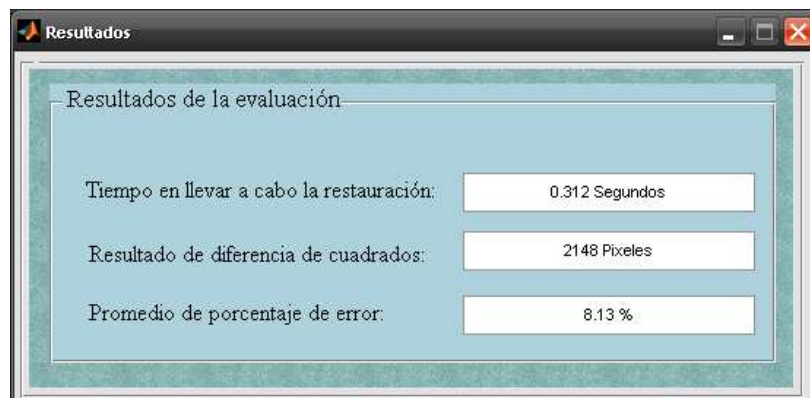


Figura A.15. Resultados de la medición de las imágenes de prueba y original.

El ejemplo anterior es para los casos donde se conocen los parámetros que causan borrosidad, ahora se presenta un ejemplo cuando estos parámetros son totalmente desconocidos. Se aplican los pasos de abrir y convertir la imagen a escala de grises. Si se conoce cuales son los parámetros exactos se escribe en el espacio correspondiente y si no en la parte donde dice Si desconoce los parámetros de borrosidad, se presiona el botón Calcular ángulo, aparecerá una ventana de espera, al terminar mostrará el resultado en la caja de texto del ángulo (Fig. A.16). Después se calcula la distancia y para restaurar la imagen se da clic en el botón Restaurar (parámetros desconocidos).



Figura A.16. Visualización del cálculo del ángulo para una imagen borrosa.

El resultado se visualizará en la parte de imagen restaurada (Fig. 4.17).



Figura A.17. Resultados de la medición de las imágenes de prueba y original.

Después de presentar una descripción sobre el funcionamiento de la aplicación desarrollada en este trabajo de tesis, en el siguiente anexo se da a conocer el código utilizado para crear tal aplicación, empleada para hacer el estudio sobre la restauración digital de imágenes.

ANEXO B. CÓDIGO FUENTE DE LA APLICACIÓN

En este anexo se integra el código fuente utilizado para crear la aplicación empleada para hacer el estudio sobre la restauración de imágenes digitales, están escritos en el lenguaje de Matlab. Una interfaz gráfica de usuario de Matlab (GUIDE) consta de dos archivos: .m y .fig. Los archivos con extensión .m son los que contienen el código correspondiente de los botones de control de la interfaz y los archivos .fig contienen los elementos gráficos.

Código para abrir y guardar una imagen

Para cargar una imagen, se utiliza el comando `imread()`, en este caso se cargará una imagen en formato JPG, TIFF, PNG desde la ruta elegida por el usuario, la imagen se despliega mediante el comando `image()`, tal y como lo muestra el código 1.

Si una variable dentro de la función va ser usada por otra u otras funciones, se tiene que declarar como variable global, para lograr esto se escribe la palabra `global` antes del nombre de la variable.

```
function abrirImagen_Callback(hObject, eventdata, handles)
global imagen;
global imagenOriginal;
colormap(gray(255)); %Especifica que la imagen utilizará el mapa de
color de grises
axes(handles.axesImagenOriginal); %Abre el componente donde se
mostrará la imagen
[nombreArchivo ruta] = uigetfile({'*.jpg'; '*.png'; '*.tif'}, 'Buscar
archivo de imagen');
rutaArchivo=[ruta,nombreArchivo];%Enlaza la ruta y nombre del
archivo
imagen= imread(rutaArchivo);%Abre la imagen desde la ruta indicada
image(imagen); %Muestra la imagen en el componente
imagenOriginal=imagen;
```

Código 1. Método `abrirImagen()`. Abre una imagen, realizando la búsqueda por parte del usuario, concatena la ruta y el nombre del archivo; con la función `imread()` se abre la imagen desde la ruta elegida (continuación).

Para guardar el contenido de una imagen en un archivo se utiliza el comando `imwrite(variable, nombre del archivo)` donde, `variable` es la que contiene la imagen a guardar y `nombre del archivo` es el nombre que indica el usuario con su respectiva extensión, el código 2 es un ejemplo para guardar la imagen del filtro de Wiener.

```
function guardarImagen_Callback(hObject, eventdata, handles)
global imgWiener;
%Se obtiene el nombre y ruta del archivo que indica el usuario
[nombreArchivo ruta] = uiputfile({'*.jpg'; '*.bmp'; '*.tif'},
'Guardar archivo de imagen');
rutaArchivo=[ruta,nombreArchivo];
imwrite(imgWiener, rutaArchivo); %Escribe en el archivo
```

Código 2. Método `guardarImagen()`. Guarda una imagen, a partir de la ruta indicada por el usuario, se concatena la ruta y el nombre del archivo, con la función `imwrite()`, graba el contenido de la imagen en un archivo.

Código para convertir una imagen a escala de grises

Las líneas mostradas en el código 3 se ejecutan cuando el usuario activa la casilla convertir a escala de grises dentro de la interfaz gráfica. La obtención de valores de los componentes se realiza mediante la sentencia `get()`. La función `rgb2gray()` cambia una imagen RGB a escala de grises.

```

function convertirEscalaGrisés_Callback(hObject, eventdata,
handles)
global imagen;
global imagenOriginal;
%Verifica el estado de la casilla para convertir una imagen en
escala de grises
valorCasilla = get(handles.convertirEscalaGrisés,'value');
if valorCasilla == 0 %Regresa la imagen a su color original
    axes(handles.axesImagenOriginal);
    image(imagenOriginal);
    imagen=imagenOriginal;
else %Convierte imagen en escala de grises
    imagenGris=rgb2gray(imagen);
    axes(handles.axesImagenOriginal);
    colormap(gray(255));
    image(imagenGris);
    imagen=imagenGris;
end;

```

Código 3. Método convertirEscalaGrisés(). La función convierte una imagen de color a monocromática, usando la función rgb2gray() cuando la casilla este activa.

Código para agregar degradación por borrosidad a una imagen

Aquí podemos ver el código que agrega degradación por borrosidad a una imagen (código 4). Los valores de distancia y ángulo son indicados por el usuario dentro de la interfaz gráfica, por ser datos de texto se tienen que convertir a numéricos usando el comando str2double().

```

function agregarRuido_Callback(hObject, eventdata, handles)
global imagen;
global psf;
global imgBorrosa;
axes(handles.axes2)
%Obtiene los valores de las cajas de texto del ángulo y distancia
para convertirlos en numéricos
distancia = str2double(get(handles.editDistancia,'string'));
angulo = str2double(get(handles.editangulo,'string'));
%Crea la función de propagación de la borrosidad
psf = fspecial('motion',distancia,angulo);
%Agrega el ruido a la imagen a partir de la función de propagación
creada
imgBorrosa = imfilter(imagen,psf,'circular','conv');
image(imgBorrosa);

```

Código 4. Método agregarRuido(). La función imfilter() filtra la imagen con el filtro de la función de propagación, el resultado será una imagen borrosa.

Código para restaurar una imagen usando filtro de Wiener

Como resalta el código 5 para restaurar la imagen borrosa a partir de los valores de ángulo y distancia conocidos, se utiliza la función `deconvwnr()` que ofrece Matlab.

```
function restaurarImagenWiener_Callback(hObject, eventdata,
handles)
global imagen;
global imgWiener
%Convierte e numéricos los valores de ángulo y distancia
distancia=str2double(get(handles.editDistanciaRest,'string'));
angulo= str2double(get(handles.editAnguloRest,'string'));
psf = fspecial('motion',distancia,angulo); %Genera la función de
propagación de ruido de acuerdo a los parámetros indicados
axes(handles.axes3)
tic;
imgWiener = deconvwnr(imagen,psf); %Restaura la imagen con filtro
Wiener
toc;
tiempoRestauracion=toc
image(imgWiener);
```

Código 5. Método `restaurarImagenWiener()`. La función `deconvwnr()` es la encargada de restaurar la imagen borrosa.

Código para restaurar una imagen usando filtro Inverso

Las siguientes líneas pertenecen al código 6 que se encarga de restaurar la imagen a partir del filtro inverso. Como ya se había mencionado en el tema 3.17 del capítulo 3, para aplicar el filtro inverso se divide la transformada de Fourier de la imagen borrosa entre la transformada de Fourier de la función de extensión del punto, al resultado se aplica la inversa de Fourier con el comando `ifft2()` de Matlab, para obtener la imagen restaurada.

```
function restaurarImagenInverso_Callback(hObject, eventdata,
handles)
global imgBorrosa otf imgInverso distancia angulo imgSinMediana;
axes(handles.axes3);
distancia=distancia-0.2;
%Obtiene la transformada de Fourier de la PSF
psf2=fspecial('motion',distancia,angulo);
otf2=psf2otf(psf2,size(imgBorrosa));
for l = 1:size(otf2,1)
    for m= 1:size(otf2,2);
        if otf2(l,m)==0;
            otf2(l,m)=distancia;
        end
    end
end
end
```

Código 6. Método `restaurarImagenInverso()`.


```

tic;
%Obtiene la transformada de Fourier de la imagen borrosa
p=fft2(imgBorrosa);
%Calcula la división entre las dos transformadas
R=p./otf2;
%Calcula la transformada inversa de Fourier
imgInverso=ifft2(R);
for a = 1:size(imgInverso,1)
    for b = 1:size(imgInverso,2)
        if imgInverso(a,b)>=1
            imgInverso(a,b)=0.999;
        end
        if imgInverso(a,b)<=0
            imgInverso(a,b)=0.001;
        end
    end
end
toc;
%Muestra el resultado de la imagen restaurada
tiempoRestauracion=toc
imwrite(imgInverso, 'C:\MATLAB7\work\TESIS\inverso.jpg');
imgInverso=imread('C:\MATLAB7\work\TESIS\inverso.jpg');
image(imgInverso)
imgSinMediana=imgInverso;

```

Código 6. Método restaurarImagenInverso (), (continuación). Con la función `psf2otf()` se realiza el cálculo de la transformada de Fourier de la función de dispersión de punto, para que después se realice la división entre la transformada de la imagen borrosa que se calcula con la función `fft2()`. Con la aplicación de la función inversa de Fourier `ifft2()` se retorna al dominio del espacio para mostrar la imagen restaurada.

Código para restaurar una imagen usando el criterio de minimización de rizado

Al ejecutar las líneas del código 7 se restaura una imagen a partir del filtro de minimización de rizado.

```

function restaurarImagenRizado_Callback(hObject, eventdata,
handles)
global imgMinimizacionRizado imagenBorrosa imgSinMediana;
global matrizUnos tiempoRestauracion;
axes(handles.axes3)
tic;
%Llamado a la función de restauración que aplica el filtro de
minimización de rizado
imgMinimizacionRizado = rest(imagenBorrosa,matrizUnos);
toc;
tiempoRestauracion=toc
imwrite(imgMinimizacionRizado,
'C:\MATLAB7\work\TESIS\rizado.jpg');
image(imgMinimizacionRizado);

```

Código 7. Método restaurarImagenRizado().

```
imgMinimizacionRizado= imread('C:\MATLAB7\work\TESIS\rizado.jpg');
imgSinMediana=imgMinimizacionRizado;
```

Código 7. Método restaurarImagenRizado(), (continuación). La función restaura la imagen borrosa utilizando el criterio de minimización de rizado, hace el llamado a la función rest(), que realiza todos los cálculos.

Las líneas del código 8 se encargan de restaurar la imagen usando el criterio de minimización de rizado, H es la matriz de degradación y P es la fila de la imagen original, recordemos que este criterio se utiliza para restaurar imágenes degradadas con movimiento lineal.

```
function imagenRestaurada = rest(imagenBorrosa,matrizUnos);
%Determina la longitud del vector para obtener la distancia de
borrosidad
distanciaBorrosidad = length(matrizUnos);
tamano = size(imagenBorrosa,2);
M = tamano + distanciaBorrosidad -1;
H = zeros(tamano,M);
%Genera una matriz, donde se asigna los valores de la distancia de
borrosidad
for i=1:tamano
    H(i,i:iz+distanciaBorrosidad -1) = matrizUnos;
end
lambda = 1e6;
P = [eye(tamano,tamano) zeros(tamano,distanciaBorrosidad -1)];
%Se obtiene la imagen restaurada
W = P-H;
imagenRestaurada = imagenBorrosa*(lambda*H)*
(inv(lambda*H'*H+W'*W))';
```

Código 8. Método rest(). Restaura la imagen borrosa utilizando el criterio de minimización de rizado.

Código para calcular el ángulo de borrosidad

Para calcular el ángulo de borrosidad se utilizan las líneas mostradas en el código 9, para saber que está llevando a cabo el proceso se muestra una barra de progreso esto es con el comando waitbar() dividida en 8 pasos. La imagen borrosa se transforma al dominio de la frecuencia con el comando fft2(), se procesa este espectro para reducir el ruido y de esta forma encontrar las líneas que existen en la imagen, de las cuales con la transformada de Hough se detectan las que tienen mayor presencia. Posteriormente, se busca en el arreglo de acumuladores la que tiene mayor valor, la cual será el posible valor del ángulo en la imagen.

```

function [Angulos] = calcularAngulo(imagenb, barraprogreso)
pasos = 8;
waitbar(1/pasos, barraprogreso);
%Convertir la imagen del dominio espacial al dominio de la
frecuencia
global fbl
fbl = abs(fft2(imagenb));
waitbar(2/pasos, barraprogreso);
%Obtener el logaritmo neperiano de la imagen que está en el dominio
de la frecuencia
lg = log(1+fbl);
lgpow = abs(lg).^2;
waitbar(3/pasos, barraprogreso);
%Se obtiene la transformada inversa de Fourier en dos dimensiones
lgcep = ifft2(lgpow);
waitbar(4/pasos, barraprogreso);
%Encontrando las líneas en la imagen
imglineas = edge(lgcep);
%Si se ha desplazado la transformada para verla mejor en pantalla,
se puede invertir el proceso de la siguiente forma
imglineas = ifftshift(imglineas);
waitbar(5/pasos, barraprogreso);
%Llamando a la transformada de Hough
h = hough(imglineas);
siz = size(imagenb);
rl = ceil(sqrt(siz(1)^2+siz(2)^2));
waitbar(6/pasos, barraprogreso);
%Buscando el valor máximo en el acumulador
maxi = 0;
angulo = 0;
for i = 1:rl
    for j = 1:360
        if h(i,j)>maxi
            maxi = h(i,j);
            angulo = j;
        end
    end
end
waitbar(7/pasos, barraprogreso);
%Almacenamiento del valor en el arreglo
g = 1;
maxarr(g) = angulo;
g = g + 1;
%Posible valor de ángulo
angulos = newarr;

```

Código 9. Método calcularAngulo(). Lleva a cabo el cálculo del ángulo de borrosidad de la imagen borrosa, se procesa la imagen al dominio de la frecuencia de donde se obtienen las líneas con la ayuda de la transformada de Hough se detecta el valor del ángulo aproximado.

Código para realizar la transformada de Hough

El código 10 se ejecuta para el cálculo de la transformada de Hough. Éste acepta una imagen de entrada y retorna un arreglo de acumuladores.

```
function [matrizAcumulador] = hough(imagen)
tamano= size(imagen); %Calcula las dimensiones de la imagen
rl = ceil(sqrt(tamano(1)^2+tamano(2)^2));
matrizAcumulador = zeros(rl,360); %Crea una matriz de ceros
angulo = 360;
%Construyendo el arreglo de acumuladores
for x = 1:tamano(1)
    for y = 1:tamano(2)
        if imagen(x,y)==1
            for angulo = 1:360
                p = round(x*cos(angulo*pi/180) +
                    y*sin(angulo*pi/180));
                if p<1
                    p = 1;
                    matrizAcumulador(p,angulo) =
                        matrizAcumulador(p,angulo);
                else
                    matrizAcumulador(p,angulo) =
                        matrizAcumulador(p,angulo) + 1;
                end
            end
        end
    end
end
end
```

Código 10. Método Hough(). Se obtiene el tamaño de la imagen, posteriormente se calcula la raíz cuadrada de los valores elevados al cuadrado de la dimensión de la imagen, este valor se redondea al entero más próximo con el comando `ceil()`, crea una matriz de ceros, por cada punto (x, y) de la imagen se calcula la ecuación paramétrica de la recta, redondeando al entero más próximo, incrementa la matriz de acumuladores si el valor de la ecuación es mayor o igual a 1.

Código para obtener el histograma de una imagen

Para observar la distribución de las intensidades de gris en la imagen se utiliza el comando `imhist()`. El código 11 muestra el ejemplo del cálculo del histograma usado para el filtro de Wiener.

```
function calcularHistograma_Callback(hObject, eventdata, handles)
global imagen
global imgWiener
%Se obtiene el histograma de la imagen y se muestra de forma
gráfica
figure, imhist(imagen)
title('Histograma: Imagen original');
figure, imhist(imgWiener)
```

Código 11. Método `calcularHistograma()`.

`title('Histograma:Imagen restaurada con filtro de Wiener.');`
Código 11. Método `calcularHistograma()`, (continuación). Muestra el histograma de la imagen original y la imagen restaurada usando el comando `imhist()`.

Código para aplicar filtro mediana

Después de la restauración de la imagen borrosa se puede aplicar el filtro mediana. En Matlab la instrucción empleada para realizar el filtro mediana es `medfilt2()`, en el código 12 se muestra su uso.

```
function aplicarMediana_Callback(hObject, eventdata, handles)
global imagen;
global imgWiener;
global imgSinMediana;
%Verifica el estado de la casilla para aplicar el filtro mediana
valorCasilla = get(handles.aplicarMediana,'value');
if valorCasilla == 1
%Función para aplicar filtro mediana
    imgConMediana = medfilt2(imgWiener);
    imgWiener=imgConMediana;
else
    imgWiener=imgSinMediana;
end;
```

Código 12. Método `aplicarMediana()`. Para aplicar el filtro mediana a una imagen se usa la función `medfilt2()`, se ejecuta cuando el usuario active la casilla de aplicar filtro mediana.

Código para calcular el porcentaje de error

El código 13 realiza las operaciones para calcular el porcentaje de error que existe entre la imagen restaurada y la original, primero se calculan las diferencias entre ambas imágenes para después aplicar la fórmula 21 explicada en el apartado de desarrollo.

```
function calcularPorcentajeError_Callback(hObject, eventdata,
handles)
global imagen imgWiener;
%Se obtiene las diferencias que existen en la imagen original e
imagen restaurada
tamano = size(imagen);
for x = 1:tamano(1)
    for y = 1:tamano(2)
        diferencias(x,y)=abs(double(imgWiener(x,y))-
double(imagen(x,y)) );
    end
end
%Se calcula el porcentaje de error tomando en cuenta el punto medio
for x = 1:tamano(1)
    for y = 1:tamano(2)
```

Código 13. Método `calcularPorcentajeError()`.

```

        porcentajesPuntoMedio128(x,y)=double((diferencias(x,y)
                                                /128)*100);
    end
end
%Se indica los parámetros para mostrar en forma gráfica los
porcentajes de error
subplot(111)
mesh(double(porcentajesPuntoMedio128(:,:,:)));
surf(double(porcentajesPuntoMedio128(:,:,:)),
'FaceColor','interp','EdgeColor','none','FaceLighting','phong');

colorbar
xlabel('Columnas');
ylabel('Filas');
zlabel('Porcentaje de error');
title('Grafica de porcentajes de error,respecto a punto medio 128')

```

Código 13. Método `calcularPorcentajeError()`, (continuación). Muestra las gráficas de los porcentajes de error calculados entre la imagen original y la imagen restaurada respecto al punto medio del rango de valores de intensidad que es 128 píxeles.

Código para agregar valor a un control deslizante

Las líneas del código 14 permiten reflejar el valor de lo que se escribe en las cajas de texto de ángulo o distancia en los controles deslizantes conocidos por su nombre en inglés de *sliders*.

```

function editAnguloRest_Callback(hObject, eventdata, handles)
valorSlider = get(handles.editAnguloRest,'String');%Convierte el
valor de texto a numérico
valorSlider = str2num(valorSlider);
%Valida que el valor del ángulo este en el rango de 0 a 360 grados,
indicando que se refleje este valor en el control deslizante
if (isempty(valorSlider) || valorSlider < 0 || valorSlider > 360)
    set(handles.sliderAngulo,'Value',0);
    set(handles.editAnguloRest,'String','0');
else
    set(handles.sliderAngulo,'Value',valorSlider);
end

```

Código 14. Método `editAnguloRes()`. Muestra el valor del ángulo en el *slider* validado sólo cuando esté en el rango de 0° a 360°, se activa cuando se edita el texto correspondiente al ángulo de restauración.

Una vez presentado el código empleado para crear la aplicación, el siguiente anexo tiene la finalidad de mostrar el contenido del CD que se incorpora junto a este trabajo de tesis.

ANEXO C. CONTENIDO DEL CD

Este anexo da a conocer la estructura de los archivos que contiene el CD que se anexa a este trabajo de tesis.

El contenido del CD de este trabajo de tesis se integra de tres carpetas, las cuales se muestran en la figura C.1.



Figura C.1. Estructura del contenido del CD.

Carpeta Aplicación

Esta carpeta contiene los archivos necesarios para ejecutar la aplicación realizada para efectuar el estudio sobre la restauración de imágenes digitales (Fig. C.2). Dentro de esta carpeta se encuentran los archivos con extensión .m cuyo contenido es el código de la

aplicación, mientras que los .fig son archivos con los datos relativos a la estructura de la interfaz gráfica, la carpeta **Iconos** tiene las imágenes que se utilizaron para la aplicación.

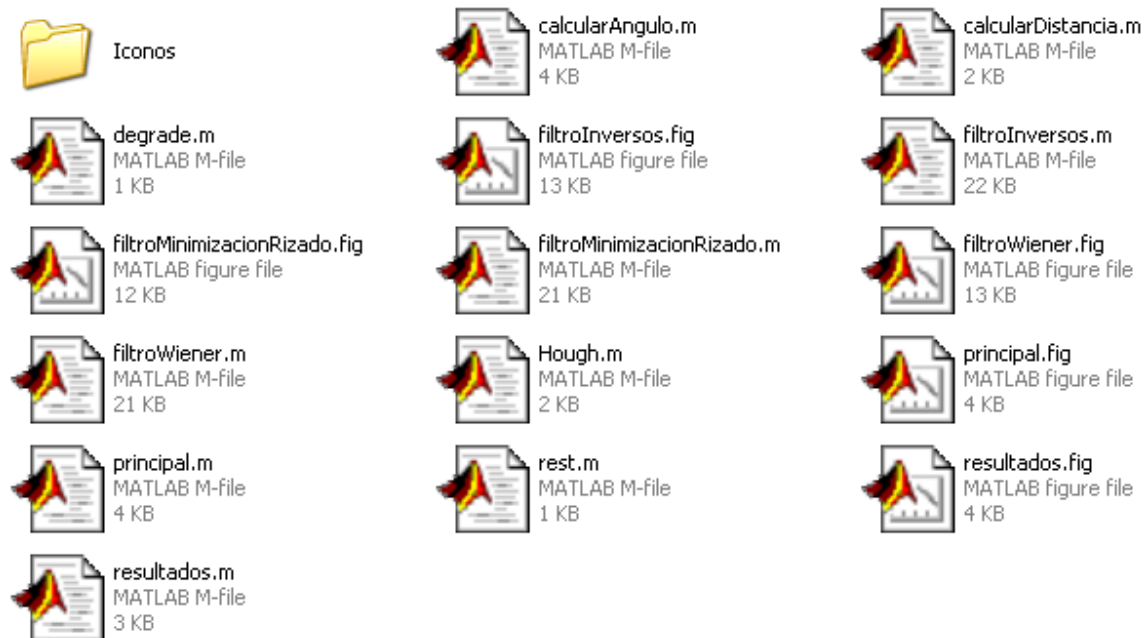


Figura C.2. Archivos fuente de la aplicación.

Carpeta Documento de tesis

La carpeta con el nombre de documento de tesis contiene en formato digital el presente documento de tesis denominado “Tesis restauración imágenes digitales” (Fig. C.3).



Figura C.3. Documento de tesis en formato pdf.

Carpeta Imágenes de prueba

Dentro de esta carpeta se encuentran dos carpetas denominadas: **Imágenes con borrosidad** e **Imágenes sin borrosidad**, el contenido corresponde a las imágenes que se utilizaron en este trabajo de tesis (Fig. C.4).

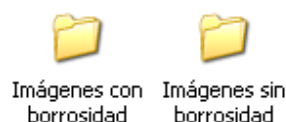


Figura C.4. Estructura de la carpeta imágenes de prueba.

En este anexo se ha mostrado los archivos que contiene el CD. Para finalizar con este trabajo de tesis en la siguiente sección se da a conocer la fuente bibliográfica que se consultó y que respalda este trabajo.

REFERENCIAS

Acharya, T & Ray, AK 2005, *Image Processing: Principles and applications*, Wiley - interscience, Tucson.

Banham, MR & Katssagelos, AK 1997, 'Digital image restoration', *IEEE Signal Processing Magazine*, vol. 14, pp. 24-41.

Baume, GL 2007, *Procesamiento de imágenes, Astronomía Observacional*, Universidad Nacional de La Plata, Facultad de Ciencias Astronómicas y Geofísicas, Buenos Aires.

Castleman, KR 1996, *Digital Image Processing*, Prentice Hall, Upper Saddle River.

Chacón, M & Molina, M 2007, *Procesamiento Digital de Imágenes*, Trillas, México, D. F.

Domínguez, TA 1996, 'Procesamiento Digital de Imágenes', *Perfiles Educativos*, no. 72, pp. 2-11.

Escalera, HA 2001, *Visión por computadora: Fundamentos y Métodos*, Pearson Educación, Madrid.

Esqueda, EJ & Palafox ML 2005, *Fundamentos de procesamiento de imágenes*, Selección anual para el libro universitario, Universidad Autónoma de Baja California, Mexicali.

González, RC & Woods, RE 1992, *Tratamiento digital de imágenes*, Addison-Wesley/Díaz de Santos, New York.

González, RC & Woods, RE 2008, *Digital Image Processing*, Prentice Hall, Upper Saddle River.

González, RC, Woods, RE & Eddins, SL 2009, *Digital Image Processing Using MATLAB*, Gatesmark Publishing, Knoxville.

González, RC, Woods, RE & Eddins, SL s.f., *Image Databases*, Image Processing Place, consultado el 15 diciembre de 2009, <<http://www.imageprocessingplace.com/index.htm>>.

GTE 2004, *Guiones de prácticas de circuitos electrónicos*, Grupo de Tecnología Electrónica, Universidad de Sevilla, consultado el 10 de junio de 2010, <http://www.gte.us.es/ASIGN/CE_2A/practicas/Guiones04_05.pdf>.

Lara, LF 2000, *Diccionario del español usual en México*, Biblioteca Virtual Miguel de Cervantes, consultado el 15 diciembre de 2009, <<http://www.cervantesvirtual.com/FichaObra.html?Ref=3161&portal=0>>.

Mery, D & López, M 2003, *Restauración de imágenes usando el criterio de minimización de rizado en la imagen restaurada*, Universidad de Santiago de Chile, consultado el 20 de abril de 2009, <<http://dccplone.ing.puc.cl/dmery/dmery/repositorio/papers/2003-CLEI.pdf>>.

Molina, R, Núñez, J, Cortijo, FJ & Mateos J 2001, 'Image Restoration in Astronomy, a Bayesian perspective', *IEEE signal processing magazine*, vol. 18, pp. 11-29.

NASA 2008, *Photojournal: NASA's Image Access Home Page*, National Aeronautics and Space Administration, consultado el 20 de enero de 2010, <<http://photojournal.jpl.nasa.gov/>>.

Núñez, AF 2007, 'Técnica Multigrad en la solución del problema de restauración digital', tesis de ingeniería en computación, Universidad Tecnológica de la Mixteca, Huajuapán de León.

Ortega, O 2007, 'Entrevista al doctor Antonio López Alemany- Videojuegos GameOver.es', *GameOver*, consultado el 3 diciembre de 2008,

<<http://www.gameover.es/entrevista/entrevista-al-doctor-antonio-lopez-alemany.html>>.

Pajares, G & De la Cruz JM 2008, *Visión por computador imágenes digitales y aplicaciones*, 2da edn, Alfaomega Grupo Editor, México, D. F.

Passariello, G & Mora F (eds) 1995, *Imágenes médicas: Adquisición, Análisis, procesamiento e Interpretación*, ediciones Equinoccio de la Universidad Simón Bolívar, Baruta.

Pérez, J 2005, *Procesamiento de Imágenes*, Departamento de Lenguajes y Ciencias de la computación, Universidad de Málaga, Málaga.

Pratt, WK 2007, *Digital Image Processing*, John Wiley & Sons, Los Altos.

RAE 2003, *Diccionario de la lengua española*, vigésima segunda edición, Real Academia Española, consultado el 18 noviembre de 2009, <<http://buscon.rae.es/draeI/>>.

Rain J, Kasturi, R & Schunck BG 1995, *Machine Vision*, McGraw-Hill, New York.

Russ, J 2006, *The Image Processing Handbook*, 5ta edn, CRC Press Taylor & Francis Group, New York.

Somolinos, JA 2002, *Avances de Robótica y visión por computador*, Ediciones de la Universidad de Castilla-La Mancha, Murcia.

Sucar, LE 2008, 'Visión: Natural y Artificial', *Komputer Sapiens*, no. 1, pp. 6-10, consultado el 6 agosto de 2009, <<http://ccc.inaoep.mx/~ksapiens/archivos/KS11.pdf>>.

Sucar, LE, Gómez G 2008, *Visión computacional*, INAOE, consultado el 6 agosto de 2009, <<http://ccc.inaoep.mx/~esucar/Libros/vision-sucar-gomez.pdf>>.

Wen, ChY 2002, 'Point Spread functions and their applications to forensic image restoration', *Forensic Science Journal 2002*, vol. 1, pp. 15-26.

Wolpert, DH, Macready, WG 1997, 'No free lunch theorems for optimization', *IEEE Transactions on Evolutionary Computation*, vol. 1, pp. 67-82.