

**DESARROLLO DE HERRAMIENTA SOFTWARE PARA LA RESTAURACIÓN Y LA
REDUCCIÓN DE RUIDO EN IMÁGENES DIGITALES**



JULIO ANDRES MONTERO ROSERO

UNIVERSIDAD DEL CAUCA

FACULTAD DE INGENIERÍA ELECTRÓNICA

PROGRAMA DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y CONTROL

POPAYÁN

2003

**DESARROLLO DE HERRAMIENTA SOFTWARE PARA LA
RESTAURACIÓN Y LA REDUCCIÓN DE RUIDO EN IMÁGENES
DIGITALES**

JULIO ANDRÉS MONTERO ROSERO

**Trabajo de Grado presentado como requisito
parcial para optar al título de Ingeniero
en Electrónica y Telecomunicaciones**

**Director
JUAN FERNANDO FLOREZ MARULANDA
Ingeniero en Electrónica y Telecomunicaciones**

**UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA
PROGRAMA DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES
DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y CONTROL
POPAYÁN**

2003

ANEXOS

ANEXO A INTERFAZ GRÁFICA Y CÓDIGO FUENTE

A Elvia y Julio.

AGRADECIMIENTOS

El autor de este trabajo agradece a Juan Fernando Flórez, Helena España y Luis Montoya cuya colaboración llevo a buen termino este proceso.

A mis amigos que hicieron de la universidad el más grato de los recuerdos.

Y a pastas la muñeca, pilas eveready, café de Colombia y todos mis patrocinadores.

TABLA DE CONTENIDO

	pag
RESUMEN	1
INTRODUCCIÓN	2
1 FORMULACIÓN DEL PROBLEMA EN RESTAURACIÓN DIGITAL DE IMÁGENES	4
1.1 PROCESO DE FORMACION DE UNA IMAGEN	4
1.1.1 Óptica y los sistemas de adquisición de imágenes:	4
1.2 MODELO MATEMATICO DEL SISTEMA DE ADQUISICIÓN DE IMÁGENES	5
1.3 IMPLICACIONES Y CARACTERÍSTICAS DE LOS PROBLEMAS INVERSOS	7
1.4 CLASIFICACIÓN DE LAS TÉCNICAS Y ALGORITMOS DE RESTAURACIÓN DE IMÁGENES.	8
1.4.1 Técnicas de restauración con PSF conocida	10
1.4.2 Técnicas con PSF desconocida (Deconvolución Ciega)	16
2 DECONVOLUCIÓN DE WIENER	19
2.1 PRELIMINARES TEÓRICOS	19

2.1.1	Matrices y vectores	19
2.1.2	Probabilidad y Estadística	22
2.2	DEDUCCIÓN DE LA EXPRESION FINAL DEL FILTRO DE WIENER	25
2.3	MODELO MATEMÁTICO DE LAS FUNCIONES DE EMBORRAMIENTO A SIMULAR	28
2.4	ANALISIS DE LAS CARACTERÍSTICAS DE LA DECONVOLUCIÓN DE WIENER	29
2.5	METRICAS UTILIZADAS PARA MEDIR EL DESEMPEÑO DEL FILTRO.	33
2.6	RESULTADOS	34
3	LA INFERENCIA BAYESIANA Y LOS CAMPOS ALEATORES COMPUESTOS DE GAUSS-MARKOV EN LA REDUCCIÓN DE RUIDO EN IMÁGENES DIGITALES	40
3.1	INFERENCIA BAYESIANA	40
3.2	MODELO DEL RUIDO	41
3.3	MODELO DE LA IMAGEN	42
3.4	CAMPOS ALEATORIOS DE MARKOV	42
3.4.1	Campos aleatorios de Markov y distribución de Gibbs.	43
3.5	FORMA DE LA DISTRIBUCIÓN POSTERIOR	45
3.6	MANEJO DE DISCONTINUIDADES	47

3.6.1	Proceso de línea	47
3.7	DISTRIBUCIÓN A POSTERIORI	48
3.7.1	Los algoritmos para optimización <i>Simulated Annealing</i> y <i>Gibbs Sampling</i>	49
3.7.2	Cálculo del número de operaciones para medir la complejidad de los algoritmos	54
3.7.3	Resultados y comparación de técnicas de reducción de ruido	55
4	CONCLUSIONES	63
4.1	CONCLUSIONES PARA LA DECONVOLUCION DE WIENER	63
4.2	CONCLUSIONES PARA EL MÉTODO BASADO EN CACGM	64
4.3	RECOMENDACIONES	65
5	MODELAMIENTO Y DISEÑO DE LA HERRAMIENTA SOFTWARE	66
5.1	CAPTURA DE REQUERIMIENTOS	66
5.1.1	Requerimientos no funcionales	66
5.1.2	Requisitos funcionales	66
5.2	DESCOMPOSICIÓN EN CASOS DE USO	66
5.2.1	Descripción de los escenarios de casos de uso:	67
5.3	DIAGRAMAS DE COLABORACIÓN	72

5.3.1	Diagrama de colaboración para el caso de uso Abrir imagen	72
5.3.2	Diagrama de colaboración para el caso de uso simular degradación	73
5.3.3	Diagrama de colaboración para el caso de uso Restaurar imagen	73
5.3.4	Diagrama de colaboración para el caso de uso Guardar imagen en disco	74
5.4	DIAGRAMA DE CLASES	75
5.4.1	Descripción del diagrama de clases	76
5.5	INTERFAZ GRÁFICA	79
	BIBLIOGRAFIA	82
	ABREVIATURAS	85

LISTA DE TABLAS

	pag
Tabla 1.1 Clasificación de métodos de restauración digital de imágenes	9
Tabla 2.1 Resultados de la deconvolución de Wiener con emborramiento por movimiento	36
Tabla 2.2 Resultados de la deconvolución de Wiener por emborramiento por turbulencia atmosférica:	37
Tabla 2.3 Resultados de la deconvolución de Wiener por emborramiento por fuera de foco	38

LISTA DE FIGURAS

	pag
Figura 1.1 Sistema óptico simple	4
Figura 2.1 Potencia de la imagen 1D sin degradar	30
Figura 2.2 Magnitud al cuadrado de la función de emborramiento 1D	30
Figura 2.3 Potencia de la señal de ruido de 1D	31
Figura 2.4 Potencia de la señal degradada 1D	31
Figura 2.5 Gráfica del filtro para la deconvolución de Wiener 1D	32
Figura 2.6 Potencia de la señal potencia de ruido constante	32
Figura 2.7 Filtro de Wiener para señal ruidosa sin emborramiento	33
Figura 2.8. (a) imagen original. (b) imagen degradada, BSNR 33 Db. (c) imagen restaurada con la potencia de la imagen y de ruido originales, ISNR 4.73 Db. (d) imagen restaurada con la potencia de la imagen y de ruido estimadas, ISNR 2.05 Db	35
Figura 3.1 a) Sistema de vecinos de 4 nodos. b) Sistema de vecinos 8 nodos	43
Figura 3.2 Cliques de tipo 1, 2, 3 y 4.	44
Figura 3.3 Vecinos del píxel i , y su proceso de línea.	46
Figura 3.4 (a) imagen de prueba, tamaño 32*32 bytes. (b) imagen ampliada ocho veces	56
Figura 3.5 a) Imagen de prueba. b) Imagen con ruido SNR = 11.87314 Db.	57
Figura 3.6 a) Imagen restaurada por CACGM, ISNR = 4.40675 Db; $\sigma_w^2 = 400$, $\alpha = 1$. b) proceso de línea resultante.	57
Figura 3.7 a) Imagen restaurada por CACGM, ISNR = 5.1446 Db; $\sigma_w^2 = 200$, $\alpha = 1$. b) proceso de línea resultante.	58

Figura 3.8 a) Imagen restaurada por CACGM, ISNR = 4.78185 Db; $\sigma_w^2 = 100$, $\alpha = 1$. b) proceso de línea resultante.	58
Figura 3.9 a) Imagen restaurada por CACGM, ISNR = 1.0647 Db; $\sigma_w^2 = 30$, $\alpha = 1$. b) proceso de línea resultante.	59
Figura 3.10 a) Imagen restaurada por CACGM, ISNR = 0.58572 Db; $\sigma_w^2 = 30$, $\alpha = 0.7$. b) proceso de línea resultante.	59
Figura 3.11 a) Imagen restaurada por CACGM, ISNR = 1.60815 Db; $\sigma_w^2 = 30$, $\alpha = 1.5$. b) proceso de línea resultante.	60
Figura 3.12 a) Imagen restaurada por CACGM MSNR = 1.72806 Db; $\sigma_w^2 = 30$, $\alpha = 2$. b) proceso de línea resultante.	60
Figura 3.13 a) Imagen restaurada por CACGM ISNR = 1.72806 Db; $\sigma_w^2 = 30$, $\alpha = 2$. b) imagen restaurada por filtro de Wiener.	61
Figura 3.14 a) Bordes de la imagen degradada. b) bordes de la imagen restaurada por filtro de Wiener.	61
Figura 3.15 a) bordes detectados por CACGM con $\sigma_w^2 = 30$ y $\alpha = 2$. b) bordes de la imagen restaurada por filtro de Wiener.	62
Figura 5.1 Diagrama de caos de uso herramienta para restauración de imágenes digitales	67
Figura 5.2 Diagrama de colaboración para el caso de uso Abrir imagen	72
Figura 5.3 Diagrama de colaboración para el caso de uso Simular degradación	73
Figura 5.4 Diagrama de colaboración para el caso de uso Restaurar imagen	74
Figura 5.5 Diagrama de colaboración para el caso de uso Guardar imagen en disco	74
Figura 5.6 Diagrama de Clases	75
Figura 5.7 GUI Inicial.	79
Figura 5.8 Dialogo_Abrir	79
Figura 5.9 Dialogo_Opciones_Degradacion	80
Figura 5.10 Dialogo Seleccione Técnica de Reducción de Ruido	80

Figura 5.11 Dialogo_Guardar	81
Figura 5.12 GUI con todas las imágenes desplegadas	81

RESUMEN

Los objetivos de este trabajo son: explorar el estado del arte de las técnicas de restauración en imágenes digitales, profundizar teóricamente en dos de sus técnicas representativas e implementar una aplicación software que ejecute una muestra de los métodos estudiados. Para cumplir con estos objetivos se inicia con el modelo matemático de un sistema de captura de imágenes, este modelo representa lo que en el campo matemático se conoce como problemas inversos mal condicionados. Para solucionar este tipo de problemas existe una gran variedad de métodos que se explican brevemente en el capítulo uno.

Una vez explorado el estado del arte se selecciona de este dos métodos de restauración y se los describe en los capítulos dos y tres. En el capítulo dos se describe lo que se conoce como Deconvolución de Wiener el cual es una combinación de filtrado inverso con el filtro de Wiener, este es el estimador lineal de una señal ruidosa que minimiza el error cuadrático medio, el capítulo incluye los preliminares analíticos en álgebra y descomposición matricial y en probabilidad y procesos aleatorios, además de los resultados obtenidos. En el capítulo tres se describe la reducción de ruido por Campos Aleatorios Compuestos de Gauss-Markov; se introducen en este capítulo todos los conceptos de campos aleatorios de Markov, lo relacionado a la inferencia bayesiana y a los algoritmos de muestreo y optimización necesarios para encontrar lo que se conoce como Maximun a Posteriori, que es el criterio con el cual se busca la mejor imagen para solucionar el problema de reducción de ruido. Para terminar se diseña la estructura de la aplicación software de restauración con la ayuda del lenguaje de modelamiento unificado (UML).

Palabras clave: restauración digital de imágenes, ruido blanco gaussiano, problema inverso, convolución, deconvolución, matriz circular, matriz Toeplitz, matriz por bloques, proceso aleatorio, matriz de covarianza, FFT, filtro de Wiener, inferencia bayesiana, maximun a posteriori, campos aleatorios de Markov, vecindad, grafo, clique, proceso de línea, distribución de Gibbs, teorema de Hammersley-Clifford, muestreo de Gibbs, Simulated Annealign.

INTRODUCCIÓN

El campo del procesamiento digital de señales ha sufrido un gran progreso en las últimas décadas gracias al desarrollo de los sistemas digitales, ciencias de la computación, hardware especializado y herramientas matemáticas. Estos desarrollos han facilitado la implementación de complejas transformaciones y operaciones matemáticas para manipular los datos de señales discretas. Es así como, por ejemplo, la aparición del algoritmo de la FFT, a mediados de la década del 40, revoluciona el concepto de procesamiento de señales discretas.

Como una rama de la teoría del procesamiento de señales aparece el Procesamiento Digital de Imágenes. Las imágenes son portadoras de una gran cantidad de información por lo que muchas veces necesitan tratamiento especial para su almacenamiento, transmisión o manipulación; por ejemplo: la gran cantidad de espacio en memoria necesario para su almacenamiento o la gran cantidad de ancho de banda necesario para transmitir las han impulsado el desarrollo de diversos algoritmos de codificación y compresión como el conocido MPEG.

Como cualquier señal de un sistema electrónico, las imágenes pueden ser distorsionadas por ruido o por imperfecciones del sistema y como cualquier señal a estas también se les puede tratar mediante filtros, amplificadores, ecualizadores y otros dispositivos comunes en lo que a tratamiento de señales se refiere.

Profundizando en cuanto a lo que las imágenes ofrecen, se llega al área de conocimiento conocida como Visión Artificial, área que se nutre de disciplinas como el Procesamiento Digital de Imágenes, Ciencias de la Computación y la inteligencia Artificial; todo con el fin de que una máquina pueda entender directamente la información que presenta una escena, tratando de imitar así esta capacidad innata en el hombre. La gran cantidad de ciencias y teorías que convergen a la temática y lo interesante de las implicaciones que sus resultados puedan tener son la principal motivación para escoger al Procesamiento Digital de Imágenes como la temática global para abordar en este proyecto de grado. Para cumplir este propósito se estudiará una de sus aplicaciones específicas: La Restauración de Imágenes Digitales.

El origen de la restauración de imágenes está en la astronomía. Esfuerzos realizados por la antigua Unión Soviética y Los Estados Unidos en las décadas de los 50s y 60s para restaurar las imágenes obtenidas por las misiones espaciales debido a que sufrieron mucha degradación fotográfica como una consecuencia de las dificultades y hostilidades del ambiente en que fueron tomadas. Una referencia más reciente son las imágenes obtenidas por el Telescopio Espacial Hubble, las imágenes obtenidas antes de corregir las imperfecciones del espejo principal necesitaron este tipo de tratamiento. La restauración

ha sido aplicada útilmente también en las imágenes médicas como radiografías, imágenes de resonancia magnética, tomografías, etc. aumentando considerablemente su resolución; se aplica a las películas y filmes deteriorados por el tiempo. Actualmente un área donde la restauración esta siendo ampliamente necesitada es en la codificación y compresión de imágenes y video. La compresión generalmente introduce ruido y pérdida de información debido a los efectos de la cuantificación y las grandes escalas de compresión que generalmente se necesitan para poder transmitir en tiempo real; la restauración se usa como un paso de post-procesamiento para ser ejecutado después de la descompresión. Los procesos de restauración son procesamientos de bajo nivel, esta generalmente no busca extraer información de la imagen, a excepción de los algoritmos que implementan el llamado proceso de línea que logra detectar los bordes en una imagen ruidosa.

El propósito de este trabajo de grado es abordar el estado del arte en la Restauración Digital de imágenes y profundizar en dos de sus técnicas más representativas, para después implementar con base en lo estudiado una aplicación software que ejecute los algoritmos de restauración elegidos. En el primer capítulo de esta Monografía se aborda la formulación del problema de restauración, sus implicaciones matemáticas y se hace una clasificación y listado de los algoritmos más empleados. El segundo y el tercer capítulo desarrollan la parte teórica para la formulación de los algoritmos escogidos que fueron la Deconvolución de Wiener y la reducción de ruido por Campos Aleatorios Compuestos de Gauss-Markov. Para terminar, el último capítulo utiliza las herramientas de diseño Orientado a Objetos como es el lenguaje de modelamiento unificado (UML) para realizar el modelamiento y diseño de la aplicación deseada.

1 FORMULACIÓN DEL PROBLEMA EN RESTAURACIÓN DIGITAL DE IMÁGENES

La restauración digital de imágenes tiene sus orígenes en las misiones espaciales; las fotografías obtenidas sufrieron degradación debido a vibraciones en la maquinaria, giros y temblores en la nave espacial. Debido al gran costo de las misiones espaciales, la pérdida de información en las imágenes resultaba excesivamente costosa. Para abordar este problema algunos algoritmos del Procesamiento Digital de Señales en una dimensión y de la Teoría de Estimaciones entraron a formar parte de lo que actualmente se conoce como Restauración Digital de Imágenes. Con el transcurrir del tiempo, estas técnicas han encontrado aplicación no solo en la astronomía sino también en campos como las imágenes médicas, las ciencias forenses y la codificación y compresión de imágenes y video [1].

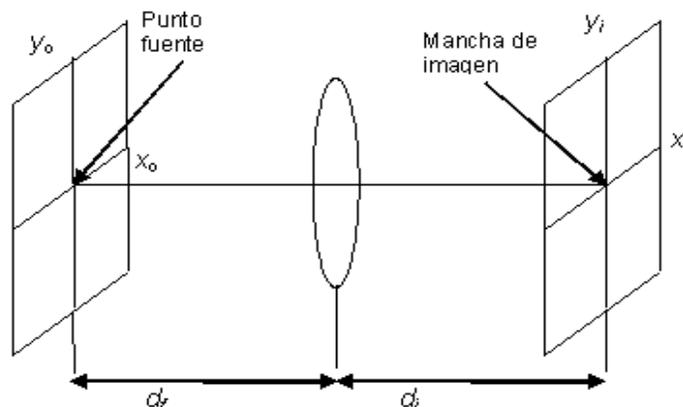
1.1 PROCESO DE FORMACION DE UNA IMAGEN

1.1.1 Óptica y los sistemas de adquisición de imágenes:

Los lentes utilizados para captar la imagen de un objeto producen en estas los efectos de proyección y la degradación debido al fenómeno de la difracción y a la aberración o incorrecta utilización de los lentes. La difracción es una consecuencia de la naturaleza ondulatoria de la luz, la aberración es causa de los errores en los procesos de manufactura de los sistemas ópticos.

Una simple representación gráfica de un sistema óptico es:

Figura 1.1 Sistema óptico simple



Un punto fuente en el plano focal (x_0, y_0) , Figura 1.1, el objeto produce una mancha de imagen en el plano de la imagen. La imagen producida por un punto es llamada PSF por su nombre en inglés point spread function, función de esparcimiento de punto.

Definición 1.1: La PSF de un sistema óptico describe la forma de la imagen producida por una fuente tipo delta de Dirac, es decir puntual, sobre el sensor [2].

Definición 1.2: La OTF, función de transferencia óptica (optical transfer function) es la transformada de Fourier de la PSF.

La PSF es una representación espacial de las características del sistema, mientras que la OTF es una representación en frecuencia.

Las definiciones de la PSF y la OTF son semejantes a las utilizadas en teoría de análisis de circuitos donde para encontrar la respuesta al impulso del sistema, que es la función que caracteriza al circuito, se aplica una señal impulso, delta de Dirac, a la entrada, y la forma de la señal resultante en la salida es la respuesta al impulso del circuito. Esta respuesta es importante debido a que la salida de un circuito cuando se le introduce cualquier señal puede ser encontrada convolucionando su respuesta al impulso con la señal de entrada. Análogamente también existe una forma de análisis frecuencial denominada la función de transferencia que es la transformada de Laplace de la respuesta al impulso. Entre más pequeña la mancha que produce el punto fuente mayor la calidad del lente. El tamaño mínimo que puede tener la mancha se alcanza cuando el sistema, Figura 1.1, está enfocado, es decir cumple la ecuación de Descartes; expresión (1.1):

$$\frac{1}{d_f} + \frac{1}{d_i} = \frac{1}{f} \quad (1.1)$$

Donde f es la longitud focal del lente.

1.2 MODELO MATEMATICO DEL SISTEMA DE ADQUISICIÓN DE IMÁGENES

Si al analizar el sistema este cumple con las propiedades de linealidad e invarianza en el espacio que implica: si el sistema al entrar una imagen $f(x, y)$ produce $g(x, y)$, entonces al ingresar $f(x + x_0, y + y_0)$ la salida debe ser $g(x + x_0, y + y_0)$. Si las dos condiciones anteriores se cumplen el sistema puede describirse por la convolución de la imagen real del objeto con la función PSF, además si hay presencia de ruido el sistema se modela con la expresión:

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x-l, y-m) \cdot f(l, m) dl \cdot dm + \eta(x, y) \quad (1.2)$$

$$g(x, y) = h(x, y) ** f(x, y)$$

Donde:

$g(x, y)$: representa a la imagen obtenida.

$h(x, y)$: representa la PSF

$f(x, y)$: representa la imagen "verdadera"

$\eta(x, y)$: representa el ruido presente.

** simboliza la convolución en dos dimensiones.

El origen del ruido puede ser por fenómenos ambientales y características del sensor, entre otros. Si la imagen es transmitida por algún sistema de telecomunicaciones se debe a el ruido que contamina la transmisión.

Si no se cumplen las propiedades de invarianza espacial y linealidad la expresión (1.2) toma una forma más general:

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y; l, m) \cdot f(l, m) dl \cdot dm + \eta(x, y) \quad (1.3)$$

Si se aplica un proceso de muestreo a le sistema de adquisición de imágenes la expresión (1.3) tiene la siguiente forma matricial:

$$g = H \cdot f + n \quad (1.4)$$

Donde si el muestreo produce una imagen que contiene $N \times N$ píxeles, entonces g , f y n son vectores columna de N^2 elementos que representan la imagen obtenida, la imagen "verdadera" y el ruido respectivamente; H modela el operador integral y la PSF; f y g están ordenados lexicográficamente, es decir, se obtienen al ordenar las filas de la matriz que representa los píxeles de la imagen una después de la otra en un vector columna.

Definición 1.3: " La Restauración Digital de Imágenes es la ciencia de la ingeniería que estudia los métodos usados para recobrar una escena original desde observaciones degradadas"[1].

Más precisamente es encontrar, refiriéndose a la ecuación (1.4), una estimación de f a partir de g , con conocimiento parcial o total de H y algunas características de n ; a este tipo de planteamientos se los conoce como problemas inversos.

1.3 IMPLICACIONES Y CARACTERÍSTICAS DE LOS PROBLEMAS INVERSOS

Cuando se busca la solución de un problema, se desea que ésta exista, sea única y estable; la estabilidad se refiere a que pequeños cambios en el dato de entrada, ruido; produzcan pequeños cambios en el dato de salida. Si estas tres características se cumplen se afirma que el problema está bien condicionado, si alguna no se cumple se dice por el contrario que el problema está mal condicionado [3]. Análíticamente esto implica:

Dado el mapeo $A: X \rightarrow Y$ y su ecuación $A \cdot x = y$ este está bien condicionado si proporciona:

- Existencia $\forall y \in Y, \exists x \in X$ tal que $A \cdot x = y$.
- Unicidad $A \cdot x_1 = A \cdot x_2 \Leftrightarrow x_1 = x_2$
- La estabilidad implica que para el operador A , A^{-1} exista.

En el caso finito discreto donde A es una matriz, $A: \mathfrak{R}^n \rightarrow \mathfrak{R}^n$, es bien condicionado si:

$$A \cdot x = y \text{ bien condicionado} \Leftrightarrow \begin{cases} A^{-1} \text{ exista} \\ \det(A) \neq 0 \\ A \cdot x = 0 \Leftrightarrow x = 0 \end{cases} \quad [4]$$

La componente de ruido en las expresiones (1.3) y (1.4) y el hecho de que la matriz H pueda ser singular conducen a que la solución no sea única, y la existencia de ceros en la OTF conduce a que la solución no sea estable. Por lo tanto la restauración digital de imágenes está en el marco de los problemas inversos mal condicionados. Muchos de los algoritmos y soluciones presentados para la restauración son producto del estudio de este tipo de problemas tanto desde un enfoque de regulación como desde un enfoque bayesiano. La regulación consiste en reformular el problema a partir de imponer restricciones a los datos y características del mismo, por ejemplo restringir la solución solo al conjunto de funciones suaves, garantizando la existencia de las derivadas de primer y segundo orden. El enfoque bayesiano busca aplicar las técnicas de inferencia bayesiana, para introducir información antecedente del problema con la ayuda de modelos estadísticos, para estimar la solución del problema inverso.

1.4 CLASIFICACIÓN DE LAS TÉCNICAS Y ALGORITMOS DE RESTAURACIÓN DE IMÁGENES.

Existen muchos métodos y algoritmos usados en esta área; por ejemplo en [5] propone la siguiente clasificación:

- a) **Directos, recursivos o iterativos:** Se refieren al número de veces que el algoritmo o filtraje se tiene que hacer para lograr la restauración.
- b) **Lineales o no lineales:** Implica que el algoritmo cumpla con la propiedad de linealidad.
- c) **Determinísticos, híbridos o estocásticos:** Si el algoritmo relaciona el resultado con métodos de la teoría de las probabilidades.
- d) **En el dominio del espacio o en el dominio de la frecuencia:** Se refiere a si los datos sufren un cambio de dominio, por ejemplo aplicando la Transformada de Fourier.
- e) **Adaptativos y no adaptativos:** Si el algoritmo relaciona algún criterio en especial para dar los resultados, por ejemplo adaptarse a los criterios de calidad de el ojo.

Otras clasificaciones pueden estar basadas en si el método tiene sus bases en la teoría de regulación o en la de inferencia bayesiana, o si trabajo con función de emborramiento conocida o no conocida, estos últimos llamados de deconvolución ciega. La clasificación propuesta en [5] no es exclusiva lo que implica que un algoritmo o una técnica pueden tener una clasificación como Directa, Lineal, Estocástica en el dominio de la frecuencia y no adaptativa.

En la actualidad existe un gran número de técnicas para la restauración digital de imágenes; a continuación se presenta una tabla de referencia para clasificar algunos métodos y algoritmos, que darán una idea de la temática de la restauración y posteriormente se les explicará con un mayor detalle.

Notación Clasificación:

Directo:	D
Recursivo:	R
Iterativo:	I
Lineal:	L
No lineal:	NL
Determinístico	Dt
Híbrido:	H
En el dominio del Espacio:	S
En el dominio de la frecuencia.	F
Adaptativos:	A
No adaptativos:	NA
Basado en Técnicas de regulación:	TR
Basado en Inferencia Bayesiana:	B
Deconvolución ciega:	DC

Notación métodos:

Filtrado inverso:	M1
Deconvolución de Weiner:	M2
Algoritmo iterativo Básico:	M3
Algoritmo iterativo con re-emborramiento:	M4
Algoritmo Básico con restricciones:	M5
Método de Tikhonov o método de CLS:	M6
Método de Millar:	M7
Método de máxima entropía:	M8
Algoritmo de Richardson_Lucy:	M9
Separación de lámina cero:	M10
Identificación de emborramiento basada en los ceros del dominio de la frecuencia:	M11
Aproximación de máxima verisimilitud:	M12
El método iterativo de deconvolucion ciega (IBD)	M13
Algoritmo de simulated annealing (SA)	M14
Algoritmo nonnegativity and support constraints recursive inverse filtering(NAS-RIF):	M15
Métodos no paramétricos basados en estadísticas de alto orden:	M16

Tabla 1.1 Clasificación de métodos de restauración digital de imágenes

M	D	R	I	L	NL	Dt	H	S	F	A	NA	TR	B	DC
M1	■			■		■			■		■			
M2	■			■			■		■		■	■		
M3			■	■		■		■			■			
M4			■	■							■	■		
M5			■	■							■	■		
M6	■			■							■	■		
M7	■			■		■					■	■		
M8			■		■		■				■		■	
M9			■		■		■				■		■	
M10	■			■					■			■		■
M11	■			■					■			■		■
M12			■											■
M13			■		■		■	■					■	
M14			■	■		■			■			■		
M15			■	■		■			■			■		
M16		■		■		■			■			■		■

La Tabla 1.1 muestra la no exclusividad de las clasificaciones propuestas. Un algoritmo puede tomar características distintivas de los diversos ítems de clasificación. Aunque algunos de los grupos propuestos si son excluyentes entre sí, por ejemplo: los métodos lineales con los no lineales, o los determinísticos con los estocásticos. Finalmente para

generar una clasificación se agrupará a los métodos de restauración en aquellos con PSF conocida y aquellos con PSF no conocida.

1.4.1 Técnicas de restauración con PSF conocida

Filtrado inverso: Conocida la función que describe a la PSF, representada por la letra H ; se utiliza la Transformada de Fourier Discreta en la ecuación (1.4) despreciando el ruido n presente en esta expresión, y del resultado se despeja f que sería la imagen buscada. El procedimiento descrito produce la expresión:

$$\hat{F}(l) = \frac{H^*(l) \cdot G(l)}{|H(l)|^2} \quad (1.5)$$

Donde l representa el par de frecuencias espaciales (u, v) ; las letras mayúsculas cursivas se usan para simbolizar que son resultado de aplicar la Transformada de Fourier Discreta. En este modelo no se ha considerado la componente de ruido, pero es claro que debido a la componente de la magnitud de la OTF en el denominador, los ceros, o sus vecindades, de esta amplificarán el ruido, esta es una de las principales razones de la inestabilidad de la solución del problema de restauración.

Deconvolución Weiner [1]: Es una aproximación de regulación directa que busca que el error cuadrático entre la imagen original y la imagen estimada sea mínimo, esta condición conduce a una expresión de la forma:

$$\hat{F}(l) = \frac{S_{ff}(l) \cdot H^*(l) \cdot G(l)}{S_{ff}(l) \cdot |H^*(l) + S_{mm}(l)|} \quad (1.6)$$

Donde S_{ff} , S_{mm} representan los espectros de potencia de la señal original y del ruido respectivamente. La expresión (1.6) esta sujeta a que tanto la imagen como el ruido sean estacionarios.

Nota. Una versión recursiva del filtro de Weiner es el llamado filtro de Kalman.

Dentro de los algoritmos iterativos se tiene:

Algoritmo iterativo Básico [5]: Si de la expresión (1.4) se considera el ruido igual a cero se llega a la identidad

$$f = f + \beta \cdot (g - H \cdot f) \quad (1.7)$$

Si se aplica a (1.7) el método de aproximaciones sucesivas se llega a

$$f_0 = \beta \cdot g \quad (1.8)$$

$$f_{k+1} = \beta \cdot g + (I - \beta \cdot H) \cdot f_k \quad (1.9)$$

Donde I representa la matriz identidad. Su clasificación, para los grupos propuestos, sería iterativo, no lineal, determinístico, en el dominio del espacio y no adaptativo.

En los algoritmos iterativos un aspecto fundamental son sus condiciones de convergencia, en este caso si H es no negativa el algoritmo converge.

Algoritmo iterativo con re-emborramiento [5]: Para garantizar la no negatividad de H se aplica un filtraje previo con la matriz H , re-emborramiento, lo que produce las expresiones:

$$\begin{aligned} f_0 &= \beta \cdot H^T \cdot g \\ f_{k+1} &= \beta \cdot H^T \cdot g + (I - \beta \cdot H^T \cdot H) \cdot f_k \end{aligned} \quad (1.10)$$

Algoritmo Básico con restricciones [5]: Las restricciones permiten la incorporación de conocimiento antecedente sobre la solución en el proceso de restauración. Estas restricciones, que modelan el conocimiento antecedente de la imagen, se definen como C_h tal que:

$$\tilde{f} = C_H \cdot f \quad (1.11)$$

Por lo tanto se producen las expresiones:

$$f_{k+1} = \beta \cdot H^T \cdot g + (I - \beta \cdot H^T \cdot H) \cdot \tilde{f}_k \quad (1.12)$$

Nota: Para la formulación de los anteriores algoritmos, a excepción de la deconvolución de Wiener, se desprecia la componente de ruido en la ecuación (1.4); esto se constituye en su principal desventaja pues al ser considerado cero, este se amplifica en cada iteración.

Método de Tikhonov o método de CLS (Restricción de mínimo cuadrado) [5]: El problema de la ecuación (1.4) puede ser reemplazado por el problema de búsqueda de mínimos

$$\begin{aligned} \min \|C \cdot f\| \\ \text{sujeto a } \|H \cdot f - g\| = \|n\| \end{aligned} \quad (1.14)$$

Donde n es una estimación de la precisión del dato (ruido) y C es un operador lineal. Aplicando el método de Lagrange resulta:

$$(H^T \cdot H + \lambda \cdot C^T \cdot C) \cdot f = H^T \cdot g \quad (1.15)$$

Donde λ es un multiplicador de Lagrange.

Método de Millar [5]: El problema de resolver la ecuación (1.4) es reemplazado por buscar un vector f que satisfaga las siguientes dos condiciones:

$$\begin{aligned} \|H \cdot f - g\| &\leq \|n\| \\ \|C \cdot x\| &\leq E \end{aligned} \quad (1.16)$$

E : es una constante predefinida.

Su solución es:

$$(H^T \cdot H + \alpha \cdot C^T \cdot C) \cdot g = H^T \cdot g \quad (1.17)$$

donde α es el parámetro de regulación.

$$\alpha = \left(\frac{\|n\|}{E} \right)^2 \quad (1.18)$$

Método de máxima entropía [3]: Este es un claro ejemplo de un método de estimación estadística. El principio de máxima entropía afirma que de todas las posibles soluciones se debe escoger aquella que tiene la máxima entropía.

la entropía de la imagen es de la forma (en la notación discreta en 1-D).

$$S(f) = \sum_i \left[f_i - m_i - f_i \cdot \log \left(\frac{f_i}{m_i} \right) \right] \quad (1.19)$$

donde $f = \{f_i\}$ representa la imagen a ser restaurada y $m = \{m_i\}$ es el antecedente (estimado de la imagen) o el modelo por defecto; S mide la similitud entre f y m .

Para introducir mejoras al modelo se considera la alta correlación que existe entre píxeles de una imagen. Este hecho no es considerado en la expresión (1.19), para resolver el

problema se considera a la imagen f como la convolución de una imagen oculta con una función intrínseca de correlación, procediendo a aplicar a la imagen oculta la definición de entropía de una imagen.

imagen oculta*ICF = imagen visible.

imagen visible**PSF = imagen emborrada.

Para encontrar el máximo de la entropía, puede ser usado el método de Newton que se aplica para encontrar el máximo de una función.

Sea $F(x); x = (x_1, x_2, \dots, x_n)^T$ la función objetivo las iteraciones son actualizadas de acuerdo a

$$x_{k+1} = x_k + \delta_x \quad (1.20)$$

δ_x es calculado resolviendo la ecuación:

$$\nabla \nabla f(x_k) \cdot \delta_x = \nabla f(x_k) \quad (1.21)$$

o de la forma equivalente:

$$\delta_x = (\nabla \nabla f(x_k))^{-1} \cdot \nabla f(x_k) \quad (1.22)$$

Algoritmo de Richardson_Lucy [3]: Este se deriva de la ecuación de imagen y la ecuación de la distribución de Poisson. Es una forma particular del método de máxima verosimilitud.

$$g(i) = \sum_i H(i | i) \cdot f(i) \quad (1.23)$$

g : imagen emborrada sin ruido.

H : PSF.

f : Objeto original.

La posibilidad de obtener un valor determinado n en un píxel cuando el valor promedio esperado es \bar{n} esta dado por la distribución de Poisson:

$$\phi(n | \bar{n}) = \frac{e^{-\bar{n}} \cdot \bar{n}^n}{n!} \quad (1.24)$$

La verosimilitud común L de conseguir el valor observado $d(i)$ en cada píxel dado el valor esperado $g(i)$ es:

$$\begin{aligned}
L &= \prod_i \phi(d(i) | g(i)) \\
L &= \prod_i \frac{e^{-g(i)} \cdot g(i)^{d(i)}}{d(i)!} \\
\ln(L) &= \sum_i [d(i) \cdot \ln g(i) - g(i) - \ln(d(i)!)] \\
\ln(L) &= \sum_i \left[d(i) \cdot \ln \sum_i H(i|i) \cdot f(i) - \sum_i H(i|i) \cdot f(i) - \ln(d(i)!) \right] \tag{1.25}
\end{aligned}$$

La solución de máxima verosimilitud ocurre cuando

$$\begin{aligned}
\frac{\partial \ln(L)}{\partial f(i)} &= 0 \\
0 &= \sum_i \left[\left(\frac{d(i)}{g(i)} - 1 \right) \cdot H(i|i) \right] \tag{1.26}
\end{aligned}$$

De este resultado se deduce la iteración Richardson-Lucy.

$$f(i)_{k+1} = f(i)_k \frac{\sum_i H(i|i) \cdot \frac{d(i)}{g(i)}}{\sum_i H(i|i)} \tag{1.27}$$

Este método es tal vez el más ampliamente usado en restauración de imágenes astronómicas, teniendo como desventaja su pobre respuesta al ruido lo que resulta en tener que detener el algoritmo iterativo antes de su convergencia para evitar la amplificación del ruido.

Métodos Bayesianos [6]: Comenzando con la definición de la función de distribución de probabilidad de observar la imagen g si f fuera la imagen verdadera $P(g|f)$, y utiliza la definición de la distribución de probabilidad $P(f)$ para incorporar información antecedente sobre la estructura esperada de la imagen. El paradigma bayesiano sugiere que la inferencia sobre la f "verdadera" debería estar basada en $P(f|g)$ dada por la regla de Bayes.

$$\begin{aligned}
P(f|g) &= \frac{P(g|f) \cdot P(f)}{P(g)} \\
P(f|g) &\propto P(g|f) \cdot P(f) \tag{1.28}
\end{aligned}$$

Se busca entonces la imagen \hat{f} la cual satisfaga

$$\hat{f} = \operatorname{argmax}(P(g | f) \cdot P(f)) \quad (1.29)$$

La cual es conocida como la estimación MAP (máxima a posteriori) de f . La expresión anterior es equivalente a:

$$\hat{f} = \operatorname{argmin}(-\log(P(f | g)) - \log(P(f))) \quad (1.30)$$

El modelo que se escoge para modelar el ruido, $P(g | f)$, y los antecedente y conocimiento de la imagen original, $P(f)$, tienen fundamental importancia en la solución del problema. Tanto para $P(g | f)$ y $P(f)$ es común usar los modelos.

$$P(g | f) \propto \frac{1}{Z_{\text{noise}}(\beta)} \exp\left(-\frac{1}{2} \beta \cdot \|g - H \cdot f\|^2\right) \quad (1.31)$$

$$P(f) \propto \alpha^{q/2} \cdot \exp\left(-\frac{1}{2} \alpha \cdot S(f)\right) \quad (1.32)$$

donde $S(f)$ es una forma cuadrática no negativa, q el número de valores propios de S , $\beta^{-1} = \sigma_n^2$, $Z_{\text{noise}}(\beta) = (2 \cdot \pi / \beta)^{p/2}$ y alfa una constante, para más detalle ver [7].

Los métodos de máxima entropía y de Richardson-Lucy son también formas de modelar estadísticamente la imagen y el ruido respectivamente.

Además, diferentes técnicas son propuestas para modelar tanto a $P(g | f)$ y $P(f)$ dependiendo del tipo de imagen, aplicación y detectores. Para destacar entre estas técnicas están los llamados campos aleatorios de Markov, MRF, que permiten incluir información sobre las discontinuidades dentro de la imagen. El manejo de las discontinuidades presentes en las imágenes es una de las principales dificultades presentes en las soluciones propuestas para restaurar imágenes, debido a que se asume la condición de suavidad en las reformulaciones del problema inverso.

Por otra parte los métodos de estimación estadística como la máxima entropía, máxima verosimilitud y la inferencia bayesiana pueden ser complementadas por herramientas como *Cross-Validation* que es usada para determinar los hiper parámetros bayesianos, o el punto de parada para la máxima verosimilitud [3].

1.4.2 Técnicas con PSF desconocida (Deconvolución Ciega)

Un problema adicional en la restauración de imágenes se presenta cuando la función PSF es desconocida o parcialmente desconocida, a este tipo de problemas se los conoce como deconvolución ciega. En [8] se encuentra una descripción completa de la deconvolución ciega.

El problema general de la deconvolución ciega se refiere a la tarea de separar dos señales convolucionadas, cuando son parcialmente o no son conocidas.

Características de la deconvolución ciega:

- La imagen y la PSF deben ser irreducibles. Una señal irreducible es aquella que no puede ser expresada como la convolución de dos o mas señales componentes.
- El objetivo de la deconvolución ciega es obtener una versión escalada y desplazada de la imagen original:
$$f = K \cdot f(x - D_x, y - D_y)$$
- El problema de deconvolución ciega en presencia de ruido sigue detenidamente el caso en ausencia de este.

Las formas de aproximarse a la solución son:

- Identificar la PSF separadamente, para usarla después en algún algoritmo común de restauración. La obtención de la PSF y la imagen verdadera son procesos disyuntos.
- Incorpora el proceso de identificación con el de restauración. Esta combinación envuelve estimar simultáneamente la PSF y la imagen verdadera.

Algunos métodos que se utilizan para resolver el problema de la deconvolución son:

Separación de lámina cero: Consiste en factorizar la transformada Z de la imagen degradada, aprovechando las propiedades de la transformada Z multidimensional y el hecho de que la imagen verdadera y la PSF son irreducibles.

Métodos de emborramiento *A priori*: Se basa en que ciertas propiedades de la PSF son conocidas o su forma es conocida. Por ejemplo el fuera de foco.

- **Identificación de emborramiento basada en los ceros del dominio de la frecuencia:** Aprovechando el hecho de que la PSF y la imagen son irreducibles y por lo tanto los ceros en frecuencia de la imagen degradada serán ceros de la imagen verdadera o de la transformada de la PSF y que

la forma de la PSF es conocida; se identificarán los ceros correspondientes a esta parte del sistema y con estos se completará la formulación de la forma de emborramiento.

Métodos de estimación de parámetros ARMA:

Definición 1.5 Un proceso autorregresivo (AR) describe un proceso estocástico que puede ser descrito por una suma pesada de sus valores anteriores y un error de tipo ruido blanco. Un proceso AR de orden q , (AR(q)) se modela mediante la expresión:

$$y_t = \sum_{j=1}^q \phi_j \cdot y_{t-j} + e_t \quad (1.33)$$

Donde y_t es la variable aleatoria que representa el proceso estocástico, ϕ_j el peso de cada valor anterior y e_t el ruido blanco.

Definición 1.6: Un proceso de movimiento medio (AM) describe un proceso estocástico que puede ser descrito por una suma pesada de componentes anteriores de ruido blanco. Un proceso AM de orden p (AM(p)) se modela mediante la expresión:

$$y_t = e_t + \sum_{j=1}^p \theta_j \cdot e_{t-j} \quad (1.34)$$

Donde θ_j es el factor de peso de cada componente de ruido.

Definición 1.9: Un proceso de movimiento medio autorregresivo (ARMA) es una unión de los dos procesos anteriores y se modela por:

$$y_t + \sum_{j=1}^q \phi_j \cdot y_{t-j} = e_t + \sum_{k=1}^p \theta_k \cdot e_{t-k} \quad (1.35)$$

El método consiste en modelar la imagen como un proceso autorregresivo (AR) y a la PSF como un proceso de movimiento medio (MA). Por lo tanto la imagen degradada resultante es un proceso de movimiento medio autorregresivo (ARMA). Identificando los parámetros ARMA permite reconocer la imagen verdadera y la PSF. Dentro de estos métodos se agrupan:

- **Aproximación de máxima verisimilitud.**
- **Aproximación cross-validation generalizada.**

Técnicas de restauración determinística, no paramétrica restringida de imágenes:

No asume parámetros. Utiliza restricciones determinísticas como la no negatividad, soporte conocido finito y la existencia de bordes emborrados invariantes son utilizados para determinar la imagen verdadera. Los métodos son iterativos y estiman los píxeles de la imagen verdadera y la PSF (o su inversa). Dentro de estas técnicas se pueden agrupar:

- **El método iterativo de deconvolucion ciega (IBD)**
- **Algoritmo de simulated annealing (SA).**
- **Algoritmo nonnegativity and support constraints recursive inverse filtering(NAS-RIF).**

Adicionalmente, las técnicas de restauración en su mayoría involucran parámetros en su formulación que deben ser estimados o calculados, este es un problema adicional que hay que resolver y forma otra ramificación de esta temática.

Los métodos anteriormente mencionados son solo una muestra de la gran cantidad de técnicas y algoritmos usados en esta área del procesamiento de imágenes, y se busca al presentarlos dar una idea del estado del arte en lo que concierne a la Restauración de Imágenes Digitales.

Al hacer un recorrido sobre una muestra de las diversas técnicas de restauración existentes se observa la gran cantidad de teorías y herramientas de la que esta área de la ingeniería se nutre, teorías que van desde el procesamiento de señales clásico con sus transformadas de Fourier y Z, teoría de regulación para solucionar los inconvenientes que conllevan que la restauración sea de tipo problema inverso mal condicionado, hasta las técnicas de inferencia estadística que permiten modelar el conocimiento antecedente que sobre las imágenes se tiene.

Los métodos relacionados a la inferencia bayesiana, aunque no son relativamente nuevos están actualmente a la vanguardia en lo que a restauración de imágenes se refiere, debido a los buenos resultados que han producido; por esta razón se desea explorar sus beneficios implementando la reducción de ruido en imágenes basada en los Campos Aleatorios Compuestos de Gauss-Markov que son una muestra clara de esta técnica.

Uno de los primeros métodos de restauración que fue aplicado es la Deconvolución de Wiener, que tiene que ver más con el procesamiento clásico basado en análisis en el dominio de la frecuencia. Este método se escoge para ser implementado debido a que las herramientas matemáticas usadas en su formulación son de mayor conocimiento y dominio.

2 DECONVOLUCIÓN DE WIENER

En este capítulo se profundiza en la formulación del filtro que implementa la deconvolución de Wiener. El filtro de Wiener tiene sus antecedentes históricos en la Segunda Guerra Mundial, donde el ingeniero del mismo nombre lo diseñó con el fin de reducir el ruido presente en las señales de radar.

Para facilitar la deducción de la expresión final que implementa la deconvolución de Wiener se necesitarán conceptos de la teoría de matrices, de probabilidad y estadística y del análisis de Fourier, que serán presentados en la siguiente sección.

2.1 PRELIMINARES TEÓRICOS

2.1.1 Matrices y vectores

Sea A una matriz compleja cualquiera, A puede expresarse:

$$A = URU^* \quad (2.1)$$

Donde $*$ denota transpuesta conjugada U es una matriz unitaria y R es triangular superior, los valores propios de la matriz A conforman la diagonal principal de R . Si A es normal, $A^*A = AA^*$ entonces R es una matriz diagonal, si A es hermitiana, $A = A^*$, los valores propios son reales, si A es hermitiana también es normal [9].

Matriz Toeplitz: Si $T_n = (t_{kj})$ es una matriz $n \times n$ tal que $t_{kj} = t_{k-j}$; entonces la matriz T_n tiene la forma [9]:

$$T_n = \begin{bmatrix} t_0 & t_{-1} & t_{-2} & \cdots & t_{-(n-1)} \\ t_1 & t_0 & t_{-1} & & \\ t_2 & t_1 & t_0 & & \\ \vdots & & & \ddots & \\ t_{n-1} & & & \cdots & t_0 \end{bmatrix} \quad (2.2)$$

Matriz Circular: Es una matriz $n \times n$ $C_{kj} = c_{(j-k) \bmod n}$, las matrices circulares son un tipo especial de matrices Toeplitz. Las matrices circulares son normales $C^*C = CC^*$ Tienen la forma [9]:

$$C = \begin{bmatrix} c_0 & c_1 & c_2 & \cdots & c_{(n-1)} \\ c_{n-1} & c_0 & c_1 & \ddots & \\ \vdots & c_{n-1} & c_0 & \ddots & \vdots c_2 \\ & & \ddots & \ddots & c_1 \\ c_1 & & & c_{n-1} & c_0 \end{bmatrix} \quad (2.3)$$

C tiene la particularidad de que puede ser expresado:

$$C = U\Psi U^* \quad (2.4)$$

Donde U es una matriz unitaria:

$$U = n^{-1/2} \left\{ e^{-2\pi i m k / n}, m, k = 0, 1, \dots, n-1 \right\} \quad (2.5)$$

y Ψ es una matriz diagonal cuyos elementos se calculan mediante la expresión:

$$\psi_m = \left\{ \sum_{k=0}^{n-1} c_k e^{-2\pi i m k / n}, m = 0, 1, \dots, n-1 \right\} \quad (2.6)$$

U será la misma para toda matriz circular de tamaño $n \times n$.

Matriz por bloques Toeplitz: Una matriz por bloques es aquella cuyos elementos son matrices, por lo que una matriz por bloques Toeplitz tendrá la forma:

$$T_n = \begin{bmatrix} T_0 & T_{-1} & T_{-2} & \cdots & T_{-(n-1)} \\ T_1 & T_0 & T_{-1} & & \\ T_2 & T_1 & T_0 & & \vdots_3 \\ \vdots & & & \ddots & \\ T_{n-1} & & \cdots & & T_0 \end{bmatrix} \quad (2.7)$$

Donde T_i es una matriz Toeplitz.

Matriz por bloques circular: Las matrices circulares por bloques también tienen la propiedad de ser normales. Estas tendrán la forma:

$$C = \begin{bmatrix} C_0 & C_1 & C_2 & \cdots & C_{(n-1)} \\ C_{n-1} & C_0 & C_1 & \ddots & \\ \vdots & C_{n-1} & C_0 & \ddots & \vdots C_2 \\ & & & \ddots & C_1 \\ C_1 & & & C_{n-1} & C_0 \end{bmatrix} \quad (2.8)$$

Donde C_i es una matriz circular.

C se puede expresar:

$$C = WRW^T \quad (2.9)$$

Tal que W es una matriz por bloques con la forma, y T denota la transpuesta conjugada:

$$W = \begin{bmatrix} W_{0,0} & W_{0,1} & \cdots & W_{0,n-1} \\ W_{1,0} & W_{1,1} & \cdots & W_{1,n-1} \\ \vdots & & \ddots & \vdots \\ W_{n-1,0} & W_{n-1,1} & \cdots & W_{n-1,n-1} \end{bmatrix} \quad (2.10)$$

$$W_{j,k} = n^{-2} \left\{ e^{-2\pi i j k / n} U; j, k = 0, 1, \dots, n-1 \right\} \quad (2.11)$$

Donde U es la matriz unitaria resultante de la descomposición de una matriz circular de tamaño $n \times n$.

Y R es una matriz diagonal por bloques con la forma:

$$R = \begin{bmatrix} R_0 & 0 & \cdots & 0 \\ 0 & R_1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & R_{n-1} \end{bmatrix} \quad (2.12)$$

Las diagonales de R_i se calculan:

$$r_v = \left\{ \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} F(m, n) e^{-2\pi i \left(\frac{jm + vn}{N} \right)}; v = 0, 1, \dots, n-1 \right\} \quad (2.13)$$

Donde $F(m,n)$ es la matriz que se uso para construir las matrices circulares de la siguiente manera:

$$C_i = \begin{bmatrix} F(i,0) & F(i,1) & \dots & F(i,n-1) \\ F(i,n-1) & F(i,0) & & \vdots \\ \vdots & & \ddots & F(i,1) \\ F(i,1) & \dots & F(i,n-1) & F(i,0) \end{bmatrix} \quad (2.14)$$

Nota [10]:

- Las matrices Toeplitz describen las transformaciones entrada-salida de los sistemas lineales espacialmente invariantes.
- Las matrices Circulares describen las transformaciones entrada-salida de los sistemas lineales espacialmente invariantes periódicos.
- Las matrices Toeplitz y circulares simples representan sistemas unidimensionales, las matrices por bloques sistemas bidimensionales.

2.1.2 Probabilidad y Estadística

Variable aleatoria: Permita ser E un experimento donde todos sus posibles eventos forman el espacio muestral U para el cual la probabilidad $P(e)$ es definida para cada $e \in U$. Una variable aleatoria es una función $X(e)$ que asocia un número real con cada salida $e \in U$. [11]

Esperanza matemática: la esperanza matemática de una variable aleatoria x con probabilidad asociada $p(x)$ esta definida por:

$$E[x] = \sum_{i=1}^n \alpha_i p(\alpha_i) \quad (2.15)$$

Donde α_i y $p(\alpha_i)$ son los valores y las probabilidades que puede tomar x respectivamente

Varianza: la varianza de una variable aleatoria se calcula por la expresión:

$$\sigma^2 = E[(x - E[x])^2] \quad (2.16)$$

Proceso aleatorio:

Función de autocorrelación [12]: Sea $X(t_1)$ y $X(t_2)$ muestras de un proceso aleatorio. La autocorrelación de estas un proceso aleatorio se calcula por la expresión:

$$R_X(t_1, t_2) = E[X(t_1)X(t_2)] \quad (2.17)$$

Proceso estacionario en sentido amplio [13]: Un proceso aleatorio $X(t)$ es estacionario en sentido amplio si:

- $E[X(t)] = \mu$ (independiente del tiempo)
- $R_X(t_1, t_2) = R_X(\tau)$ con $\tau = t_2 - t_1$ (es solo función entre la diferencia de los instantes considerados).

Proceso ergódico [14]: en un proceso aleatorio se pueden calcular dos tipos de promedios: promedio en el tiempo y promedio en conjunto; esperanza matemática. Entonces un proceso aleatorio será ergódico si y solo si:

- El promedio en el tiempo es el mismo para toda la familia de funciones del proceso aleatorio.
- El promedio en conjunto es constante respecto al tiempo.
- El promedio en el tiempo y el promedio en el conjunto son numéricamente iguales.

Promedio en el tiempo: se calcula por la expresión:

$$\mathcal{E}\{x(t)\} = \int_{-\infty}^{\infty} x(t) dt \quad (2.18)$$

Correlación en el tiempo: se calcula por la expresión:

$$R(x(t), y(t)) = \int_{-\infty}^{\infty} x(t)y(t + \tau) dt \quad (2.19)$$

Autocorrelación en el tiempo: se calcula por la expresión:

$$R_x(\tau) = \int_{-\infty}^{\infty} x(t)x(t + \tau) dt \quad (2.20)$$

Los procesos aleatorios tienen la propiedad que su espectro de potencia se puede calcular como la transformada de Fourier de la función de autocorrelación:

$$P_x(\omega) = \mathfrak{F}\{R_x(\tau)\} \quad (2.21)$$

Nota: los conceptos de procesos aleatorios unidimensionales son fácilmente generalizables para dos dimensiones.

Vector aleatorio: es un vector cuyos componentes son variables aleatorias.

$$x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix} \quad (2.22)$$

El vector promedio de x será:

$$E[x] = \begin{bmatrix} E[x_0] \\ E[x_1] \\ \vdots \\ E[x_{n-1}] \end{bmatrix} \quad (2.23)$$

La correlación de dos vectores aleatorios x , y será la matriz:

$$R_{xy} = E[x \cdot y^T] \quad (2.24)$$

Si x e y no son correlacionados R_{xy} es una matriz de ceros.

La auto-covarianza del vector x será la matriz:

$$C_{xx} = E[(x - E[x])(x - E[x])^T] \quad (2.25)$$

Principio de ortogonalidad: para la estimación lineal del error cuadrático medio mínimo $\hat{\theta}$, de θ , el error es ortogonal al dato observado x [15].

$$E[(\theta - \hat{\theta})x^T] = 0 \quad (2.26)$$

Note que 0 es una matriz.

2.2 DEDUCCIÓN DE LA EXPRESIÓN FINAL DEL FILTRO DE WIENER

Considerando el modelo matricial del proceso de observación, captura de una imagen, ecuación (1.4), tenemos.

$$g = H \cdot f + n$$

Se asumirán los siguientes aspectos:

- La expresión (1.4) representará un proceso lineal, periódico e invariante espacialmente, por lo que H será una matriz circular por bloques, que es construida a partir de la función de emborramiento.
- f y n serán procesos bidimensionales estacionarios en sentido amplio, ergódicos y de media cero.
- f y n son procesos no correlacionados.
- n es ruido blanco gaussiano.

De (1.4) se busca encontrar un estimador lineal A , para producir un estimado \hat{f} ; definido por:

$$\hat{f} = A \cdot g \tag{2.27}$$

que minimice[16]: $E\left[\|f - \hat{f}\|^2\right]$

Es decir, se busca la mejor estimación lineal de la imagen original a partir de la imagen observada tal que el error cuadrático medio $E\left[\|f - \hat{f}\|^2\right]$ sea mínimo.

Considerando el principio de ortogonalidad, citado en lo preliminares analíticos, se tiene:

$$E\left[(f - \hat{f}) \cdot g^T\right] = 0 \tag{2.28}$$

$$E\left[f \cdot g^T\right] - A \cdot E\left[g \cdot g^T\right] = 0$$

$$E\left[f \cdot (H \cdot f + n)^T\right] - A \cdot E\left[(H \cdot f + n) \cdot (H \cdot f + n)^T\right] = 0$$

$$E\left[f \cdot f^T \cdot H^T\right] + E\left[n^T\right] - A \cdot \left\{E\left[H \cdot f \cdot f^T \cdot H^T\right] + E\left[n \cdot f^T \cdot H^T\right] + E\left[H \cdot f \cdot n^T\right] + E\left[n \cdot n^T\right]\right\} = 0$$

Debido a que f y n son no correlacionados, su media es cero y que la esperanza es invariante ante las transformaciones lineales. La expresión (2.28) resulta en:

$$\begin{aligned} E[f \cdot f^T] \cdot H^T - A \cdot \{H \cdot E[f \cdot f^T] \cdot H^T + E[n \cdot n^T]\} &= 0 \\ R_{ff} \cdot H^T - A \cdot (H \cdot R_{ff} \cdot H^T + R_{nn}) &= 0 \\ A = R_{ff} \cdot H^T \cdot (H \cdot R_{ff} \cdot H^T + R_{nn})^{-1} & \quad (2.29) \end{aligned}$$

Por lo que la estimación será:

$$\hat{f} = R_{ff} \cdot H^T \cdot (H \cdot R_{ff} \cdot H^T + R_{nn})^{-1} \cdot g \quad (2.30)$$

Se supuso que f y n son procesos estacionarios en sentido amplio, esto conduce a que la correlación de dos píxeles $f_{i,j}$ y $f_{l,m}$ solo depende de la distancia que hay entre las filas y las columnas de estos, es decir cumplen con la condición:

$$R_{ff}((i, j), (l, m)) = R_{ff}(l - i, m - j)$$

El cumplimiento de esta condición hace que las matrices R_{ff} y R_{nn} sean matrices circulares por bloques, que se construyen a partir de las siguientes matrices base:

$$\begin{bmatrix} E[f_{0,0}f_{0,0}] & E[f_{0,0}f_{0,1}] & \cdots & E[f_{0,0}f_{0,n-1}] \\ E[f_{0,0}f_{1,0}] & E[f_{0,0}f_{1,1}] & \cdots & E[f_{0,0}f_{1,n-1}] \\ \vdots & \vdots & & \vdots \\ E[f_{0,0}f_{n-1,0}] & E[f_{0,0}f_{n-1,1}] & \cdots & E[f_{0,0}f_{n-1,n-1}] \end{bmatrix} \quad (2.31)$$

$$\begin{bmatrix} E[n_{0,0}n_{0,0}] & E[n_{0,0}n_{0,1}] & \cdots & E[n_{0,0}n_{0,n-1}] \\ E[n_{0,0}n_{1,0}] & E[n_{0,0}n_{1,1}] & \cdots & E[n_{0,0}n_{1,n-1}] \\ \vdots & \vdots & & \vdots \\ E[n_{0,0}n_{n-1,0}] & E[n_{0,0}n_{n-1,1}] & \cdots & E[n_{0,0}n_{n-1,n-1}] \end{bmatrix} \quad (2.32)$$

Lo que facilita que estas matrices se puedan diagonalizar y los elementos de la diagonal representen el espectro de potencia de las señales f y n que son procesos ergódicos, debido a que se estableció esta característica como una de las hipótesis iniciales. .

Como R_{ff} , R_{mm} y H tienen las mismas dimensiones y son circulares por bloques por lo tanto normales, entonces se pueden expresar mediante un producto con la forma WRW^* donde W es la misma para R_{ff} , R_{mm} y H . Entonces \mathcal{A} , \mathcal{H} , P_f , P_n son las matrices diagonales correspondientes a A , H , R_{ff} , R_{mm} respectivamente.

Por lo tanto si se multiplica por W^T a ambos lados de la expresión (2.30) se obtiene:

$$W^T \cdot \hat{f} = W^T \cdot W \cdot P_f \cdot W^T \cdot (W \cdot \mathcal{H} \cdot W^T)^T \cdot \left(W \cdot \mathcal{H} \cdot W^T \cdot W \cdot P_f \cdot W^T \cdot (W \cdot \mathcal{H} \cdot W^T)^T + W \cdot P_n \cdot W^T \right)^{-1} \cdot g$$

Aprovechando que W es una matriz normal y unitaria, la expresión anterior se simplifica a:

$$\hat{F} = P_f \cdot \mathcal{H}^T (\mathcal{H} \cdot P_f \mathcal{H}^T + P_n)^{-1} \cdot G \quad (2.33)$$

Apoyándose en las propiedades de las matrices diagonales se logra la expresión final para implementar computacionalmente.

$$\hat{F}(u, v) = \frac{P_f(u, v) \cdot H^*(u, v) \cdot G(u, v)}{|H(u, v)|^2 \cdot P_f(u, v) + P_n(u, v)} \quad (2.34)$$

Para construir se asume que se conoce o bien las matrices de correlación de la imagen y el ruido o se conoce sus espectros de potencia. Esta situación no es real y se deberán estimar estos elementos que compone el filtro.

La potencia del ruido, gracias a que se asumió que es blanco gaussiano, se puede estimar; aproximar, con su varianza σ_n^2 . Con el fin de estimar la potencia de la señal se utilizará lo que se conoce como periodograma [1]:

$$P_y^{per}(u, v) = \frac{1}{N^2} [Y(u, v) \cdot Y^*(u, v)] \quad (2.35)$$

Comúnmente se utiliza el espectro de la imagen degradada como estimado del espectro de la imagen original, pero aprovechando que [17]:

$$P_g(u, v) = H(u, v) \cdot P_f(u, v) + P_n(u, v) \quad (2.36)$$

$$P_f(u, v) = \frac{P_g(u, v) - P_n(u, v)}{H(u, v)} \quad (2.37)$$

La expresión (2.37) tiene la desventaja que $H(u, v)$ puede tener ceros, es decir la matriz H es singular, lo que hace que no se pueda implementar. Para la estimación de $P_f(u, v)$ se toma solamente el numerador de (2.37) y se utiliza también la estimación del ruido así:

$$\hat{P}_f(u, v) = P_g^{per}(u, v) - \sigma_n^2 \quad (2.38)$$

Entonces la ecuación (2.34) se transforma en:

$$\hat{F}(u, v) = \frac{\hat{P}_f(u, v) \cdot H^*(u, v) \cdot G(u, v)}{|H(u, v)|^2 \cdot \hat{P}_f(u, v) + \sigma_n^2} \quad (2.39)$$

Cuando el emborramiento no está presente la componente $H(u, v)$ se hace igual a la matriz identidad y (2.39) se reduce a:

$$\hat{F}(u, v) = \frac{\hat{P}_f(u, v) \cdot G(u, v)}{\hat{P}_f(u, v) + \sigma_n^2} \quad (2.39)$$

En la expresión (2.39) se destaca que el filtro encontrado no efectúa ningún tipo de procesamiento, o recuperación sobre la fase de la señal degradada, esto es una gran falencia de la deconvolución de Wiener debido a que en la componente de la fase en el espectro de la señal se encuentra información importante sobre la posición espacial de los detalles en una imagen.

2.3 MODELO MATEMÁTICO DE LAS FUNCIONES DE EMBORRAMIENTO A SIMULAR

Para medir y probar el funcionamiento del filtro se trabajará con las funciones de emborramiento más conocidas que son:

Emborramiento por movimiento [1]: producido por el movimiento relativo del objeto. Se modela:

$$h(i, j) = \begin{cases} \frac{1}{L^2} & \text{si } -\frac{L}{2} \leq i, j \leq \frac{L}{2} \\ 0 & \text{en otro caso} \end{cases} \quad (2.40)$$

Emborramiento por fuera de foco uniforme [1]: pretende modelar el fuera de foco simple como una distribución de intensidad uniforme dentro de un disco circular. Se modela:

$$h(i, j) = \begin{cases} \frac{1}{\pi \cdot R^2} & \text{si } 0 < \sqrt{i^2 + j^2} \leq R \\ 0 & \text{en otro caso} \end{cases} \quad (2.41)$$

Emborramiento por turbulencia atmosférica [1]: Debida a largas exposiciones a los fenómenos atmosféricos. Generalmente se aproxima por:

$$h(i, j) = K \cdot \exp\left(-\frac{i^2 + j^2}{2 \cdot \sigma^2}\right) \quad (2.42)$$

Donde K es una constante que normaliza la expresión y σ^2 mide la severidad del emborramiento.

2.4 ANALISIS DE LAS CARACTERÍSTICAS DE LA DECONVOLUCIÓN DE WIENER

Estudiando las gráficas de las señales y la de las funciones de emborramiento, en el dominio de la frecuencia, que por razones de comodidad y simpleza se limitarán a una dimensión, lo que no cambia las condiciones del filtro, se tratará de extraer algunas de los principales rasgos del filtro estudiado.

Las señales son simuladas por pulsos gaussianos, mientras que la función de emborramiento lo es por un filtro pasa-bajo ideal lo que facilita mucho los cálculos y también la comprensión de los gráficos.

Figura 2.1 Potencia de la imagen 1D sin degradar

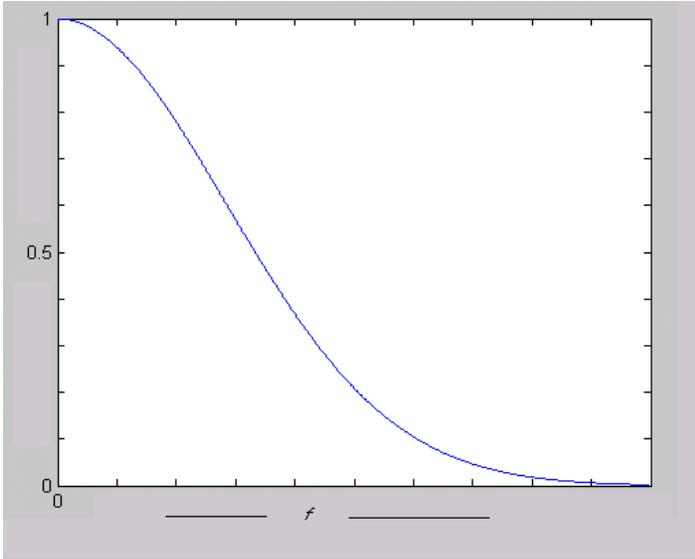


Figura 2.2 Magnitud al cuadrado de la función de emborramiento 1D

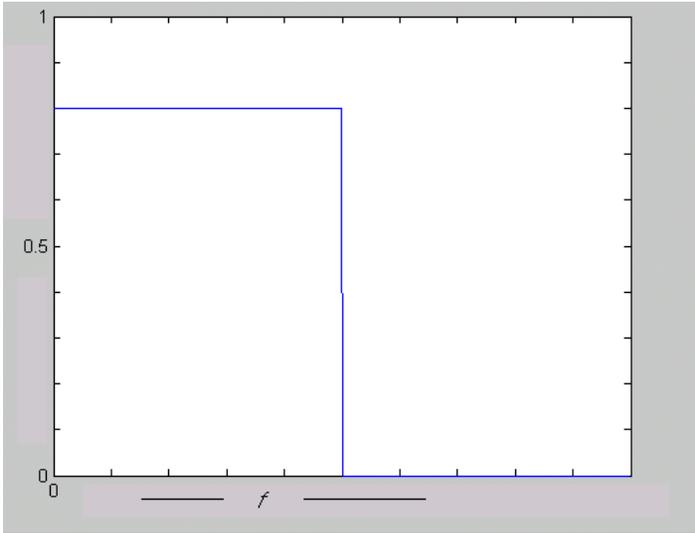


Figura 2.3 Potencia de la señal de ruido de 1D

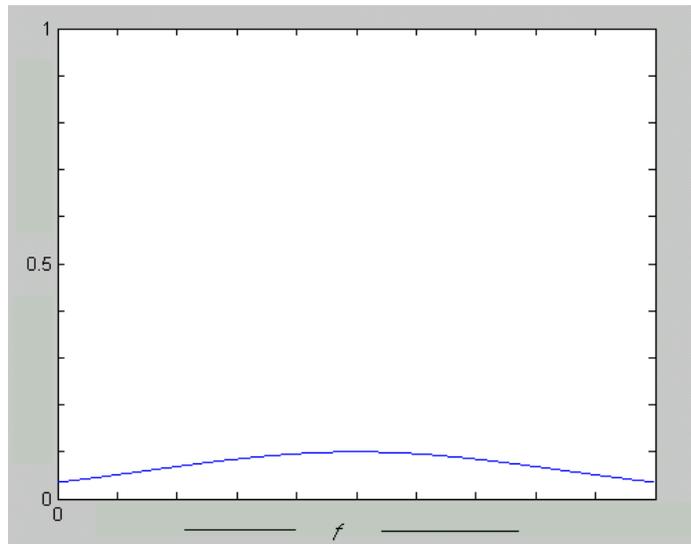
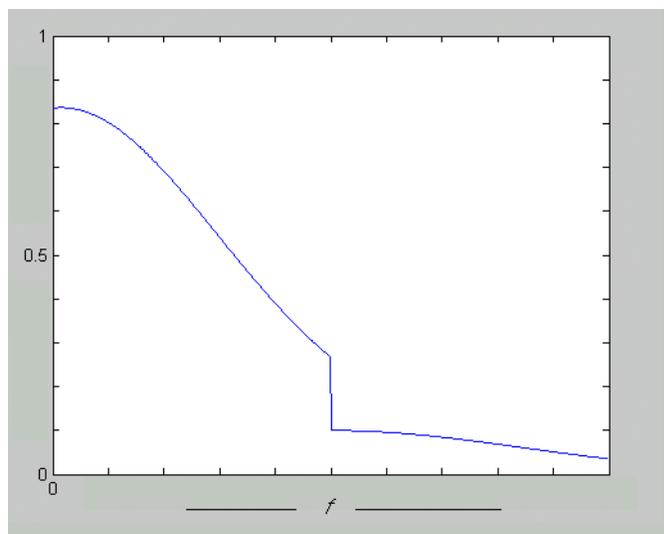
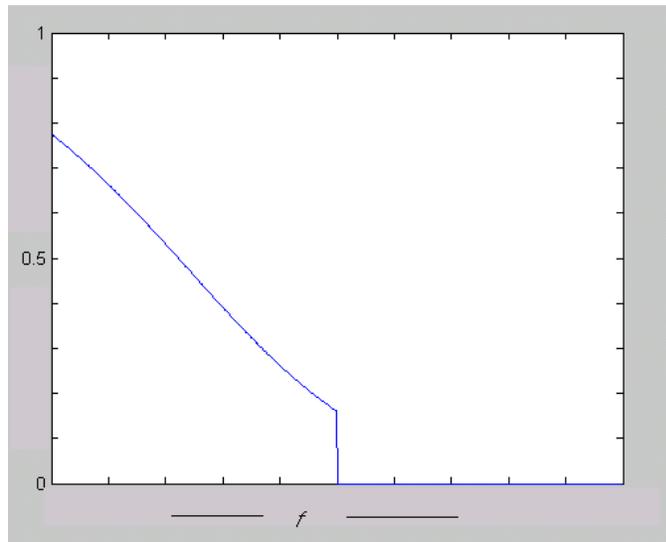


Figura 2.4 Potencia de la señal degradada 1D



Esta señal es obtenida multiplicando la potencia de la señal original con la función de emborramiento y adicionándole la potencia de la señal de ruido. Se destaca que el ruido esta presente en todo el rango de frecuencias y que la señal de salida ha perdido todas las componentes de frecuencia de la imagen original superiores a la frecuencia de corte de la función de emborramiento.

Figura 2.5 Gráfica del filtro para la deconvolución de Wiener 1D



El filtro resultante es cero a partir de la frecuencia de corte de la función de emborramiento, por lo que no es capaz de recuperar ninguna componente de la señal original a partir de esta frecuencia. Este es uno de los principales inconvenientes que posee la deconvolución de Wiener.

Para observar su desempeño frente al ruido se excluye la función de emborramiento y se asume que la potencia del ruido es constante en todo el espectro.

Figura 2.6 Potencia de la señal potencia de ruido constante

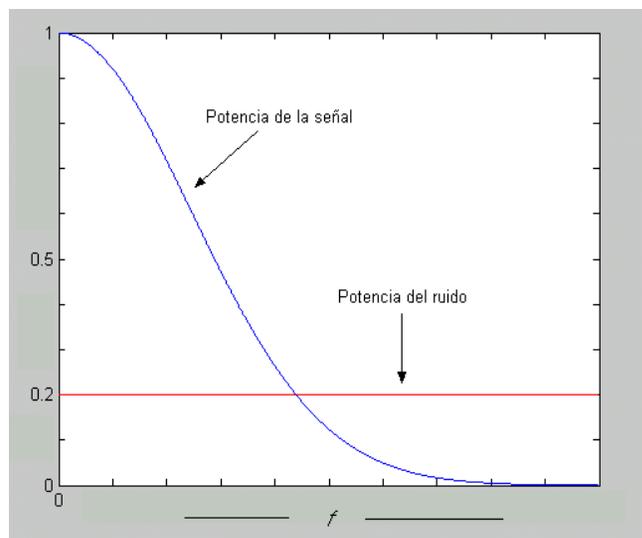
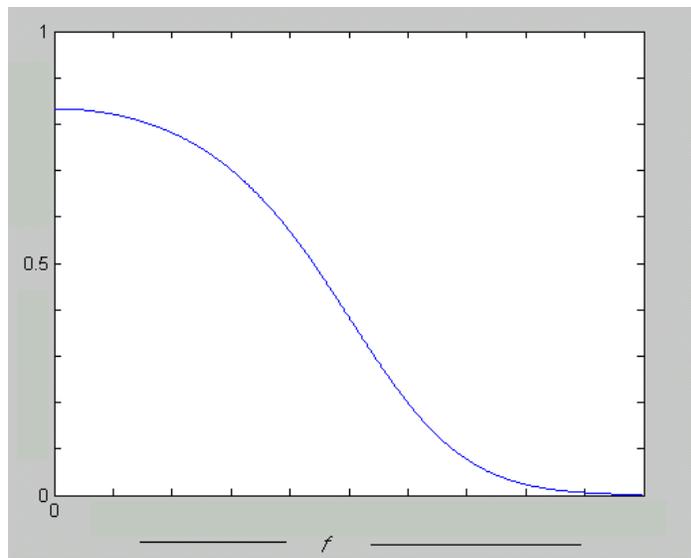


Figura 2.7 Filtro de Wiener para señal ruidosa sin emborramiento



En la última gráfica se observa que en las frecuencias más altas, parte derecha de la figura, tienden a cero, y es en estas frecuencias es donde el ruido supera totalmente la magnitud de la señal, es decir la deconvolución de Wiener elimina aquellas componentes de frecuencia que están totalmente enmascaradas por el ruido.

2.5 METRICAS UTILIZADAS PARA MEDIR EL DESEMPEÑO DEL FILTRO.

En [1] se proponen las siguientes métricas:

Razón señal a ruido emborrada (BSNR): Trata de medir la degradación producida por el emborramiento y el ruido, su valor se calcula mediante la expresión:

$$BSNR = 10 \cdot \log_{10} \left\{ \frac{\frac{1}{M \cdot N} \sum_{i,j} [H \cdot f - \overline{H \cdot f}]}{\sigma_n^2} \right\} \quad (2.43)$$

Donde $\overline{H \cdot f}$ representa el valor esperado o el promedio de $H \cdot f$.

Mejoramiento en relación señal a ruido (ISNR): Trata de medir la eficacia de los algoritmos utilizado, y se define mediante la expresión:

$$ISNR = 10 \cdot \log_{10} \left\{ \frac{\sum_{i,j} [f(i,j) - g(i,j)]^2}{\sum_{i,j} [f(i,j) - \hat{f}(i,j)]^2} \right\} \quad (2.44)$$

Donde \hat{f} representa la imagen restaurada. Esta medida se basa en el error cuadrático medio (MSE). Es aplicable también en la ausencia de emborramiento.

Relación señal a ruido (SNR): Utilizada cuando la imagen degradada no sufrió emborramiento. Se calcula por:

$$SNR = 10 \cdot \log_{10} \left(\frac{\frac{1}{MN} \sum_{i,j} f(i,j)^2}{\sigma_n^2} \right) \quad (2.45)$$

2.6 RESULTADOS

La figura 2.8 presenta el resultado visual para la imagen de prueba cameraman de tamaño 256x256 píxeles con un BSNR de 33 Db. La figura muestra los efectos de estimar el espectro de potencia de f . Hay un evidente deterioro en la imagen de salida cuando se estimando el espectro de potencia de f con respecto a la imagen de salida cuando se usa la verdadero espectro de potencia de f .

Para medir el desempeño del filtro, Deconvolución de Wiener, este se prueba con cada una de las funciones de emborramiento planteadas variando secuencial mente sus parámetros de definición y la varianza del ruido σ_n^2 . Esta evaluación produjo los resultados mostrados en las tablas Tabla 2.1 a Tabla 2.3. Cada cuadro interno en las tablas tiene el valor de las métricas BSNR, ISNR y MSE ordenadas en forma descendente. De los resultados expuestos se destaca el aumento de ISNR a media que σ_n^2 aumenta, y su decrecimiento cuando el grado de emborramiento aumenta.

Figura 2.8. (a) imagen original. (b) imagen degradada, BSNR 33 Db. (c) imagen restaurada con la potencia de la imagen y de ruido originales, ISNR 4.73 Db. (d) imagen restaurada con la potencia de la imagen y de ruido estimadas, ISNR 2.05 Db



(a)



(b)



(c)



(d)

Tabla 2.1 Resultados de la deconvolución de Wiener con emborramiento por movimiento

Des Var	10	20	30	40	50	60	70
200	19.325 2.054 498.001	19.216 1.763 761.283	19.141 1.643 964.787	19.079 1.619 1079.256	19.020 1.676 1153.614	18.964 1.388 1308.101	18.910 1.539 1339.540
400	16.314 2.809 521.735	16.205 2.314 791.122	16.131 2.017 1004.178	16.068 2.020 1114.545	16.010 2.001 1192.620	15.953 1.726 1342.249	15.900 1.874 1377.013
600	14.553 3.444 541.369	14.444 2.810 810.603	14.370 2.417 1029.570	14.307 2.426 1126.824	14.249 2.315 1227.654	14.192 2.057 1364.065	14.139 2.134 1416.492
800	13.304 4.027 554.997	13.195 3.311 816.062	13.121 2.860 1034.718	13.058 2.784 1139.859	12.999 2.655 1243.115	12.943 2.456 1366.095	12.889 2.479 1417.026
1000	12.335 4.463 564.405	12.226 3.740 821.814	12.151 3.233 1039.060	12.089 3.092 1161.010	12.030 2.989 1252.189	11.974 2.806 1367.465	11.920 2.772 1428.646
1200	11.543 4.871 579.593	11.434 4.115 834.390	11.360 3.593 1053.256	11.297 3.434 1170.715	11.238 3.284 1262.909	11.182 3.074 1388.270	11.128 3.099 1425.485
1400	10.874 5.316 585.315	10.765 4.418 855.414	10.690 3.887 1065.984	10.628 3.719 1174.574	10.569 3.575 1277.256	10.513 3.280 1402.800	10.459 3.265 1464.735
1600	10.294 5.658 596.211	10.185 4.749 854.267	10.110 4.215 1063.996	10.048 4.020 1181.125	9.989 3.780 1286.308	9.933 3.607 1388.859	9.879 3.473 1480.944
1800	9.782 5.965 607.955	9.673 5.046 861.390	9.599 4.435 1084.645	9.536 4.290 1181.883	9.477 4.127 1265.430	9.421 3.763 1411.043	9.368 3.843 1458.472
2000	9.325 6.257 613.687	9.216 5.299 873.145	9.141 4.709 1083.579	9.079 4.517 1193.540	9.020 4.311 1296.188	8.964 4.055 1414.614	8.910 4.017 1470.337

Tabla 2.2 Resultados de la deconvolución de Wiener por emborramiento por turbulencia atmosférica:

Turb	100	200	300	400	500	600	700
Var							
200	19.185 1.581 804.591	19.116 1.319 1000.482	19.069 1.190 1115.102	19.032 1.186 1188.058	19.000 1.199 1227.094	18.972 1.241 1272.554	18.947 1.267 1299.409
400	16.175 2.093 839.082	16.106 1.763 1031.015	16.059 1.666 1143.776	16.022 1.594 1203.949	15.990 1.612 1259.515	15.961 1.672 1288.611	15.936 1.682 1312.851
600	14.414 2.606 849.517	14.345 2.261 1044.410	14.298 2.104 1148.823	14.261 2.054 1221.461	14.229 2.068 1264.139	14.201 2.020 1304.346	14.175 2.070 1326.072
800	13.164 3.075 867.345	13.096 2.602 1070.274	13.049 2.455 1175.458	13.011 2.488 1219.361	12.979 2.405 1276.362	12.951 2.412 1301.022	12.926 2.432 1328.748
1000	12.195 3.526 868.764	12.127 3.045 1068.033	12.080 2.893 1164.925	12.042 2.807 1234.170	12.010 2.798 1280.185	11.982 2.768 1312.084	11.957 2.736 1350.438
1200	11.403 3.863 886.965	11.335 3.373 1088.704	11.288 3.178 1182.330	11.250 3.146 1238.283	11.218 3.115 1281.483	11.190 3.103 1321.425	11.165 3.097 1337.372
1400	10.734 4.216 894.036	10.665 3.703 1084.845	10.618 3.510 1191.086	10.581 3.462 1242.878	10.549 3.431 1296.985	10.521 3.372 1324.722	10.496 3.337 1367.837
1600	10.154 4.573 896.147	10.085 4.010 1097.203	10.038 3.847 1180.783	10.001 3.779 1241.053	9.969 3.671 1300.263	9.941 3.661 1319.416	9.916 3.615 1363.068
1800	9.642 4.854 903.882	9.574 4.319 1099.348	9.527 4.095 1199.452	9.489 4.009 1257.306	9.457 3.968 1291.193	9.429 3.906 1337.508	9.404 3.896 1368.346
2000	9.185 5.108 916.586	9.116 4.619 1089.896	9.069 4.352 1199.252	9.032 4.216 1264.285	9.000 4.256 1285.953	8.972 4.173 1325.546	8.947 4.069 1387.800

Tabla 2.3 Resultados de la deconvolución de Wiener por emborramiento por fuera de foco

Rad	10	20	30	40	50	60	70
Var							
200	20.314 2.726 721.265	19.635 1.671 1050.998	19.341 1.513 1218.578	19.175 1.401 1389.713	19.029 1.683 1447.638	18.911 2.153 1443.508	18.826 1.533 1824.040
400	17.304 3.205 742.373	16.625 2.016 1092.916	16.331 1.868 1252.913	16.165 1.702 1430.372	16.018 2.001 1469.573	15.901 2.264 1517.330	15.815 1.778 1859.128
600	15.543 3.617 758.138	14.864 2.464 1097.738	14.570 2.239 1269.610	14.404 2.043 1443.169	14.257 2.246 1501.598	14.140 2.409 1572.203	14.054 1.949 1918.522
800	14.294 3.994 771.058	13.615 2.745 1129.223	13.320 2.639 1277.422	13.154 2.373 1450.923	13.008 2.647 1488.266	12.891 2.640 1608.805	12.805 2.131 1967.047
1000	13.325 4.442 772.829	12.646 3.211 1120.235	12.351 2.874 1291.358	12.185 2.732 1454.367	12.039 2.884 1512.947	11.922 3.009 1586.306	11.836 2.367 1987.253
1200	12.533 4.692 794.041	11.854 3.485 1142.301	11.559 3.210 1306.115	11.394 2.996 1459.189	11.247 3.171 1512.069	11.130 3.141 1631.045	11.044 2.545 2005.166
1400	11.863 5.050 795.684	11.185 3.808 1148.476	10.890 3.510 1314.604	10.724 3.256 1474.205	10.578 3.389 1526.770	10.460 3.410 1624.011	10.375 2.679 2051.483
1600	11.283 5.315 808.357	10.605 4.070 1155.139	10.310 3.763 1316.595	10.144 3.512 1467.726	9.998 3.706 1511.946	9.880 3.599 1640.828	9.795 2.945 2039.833
1800	10.772 5.675 800.407	10.093 4.293 1173.928	9.799 4.020 1315.243	9.633 3.745 1479.441	9.486 3.889 1528.545	9.369 3.760 1653.290	9.283 3.134 2040.210
2000	10.314 5.894 811.366	9.635 4.600 1167.375	9.341 4.253 1326.353	9.175 3.981 1482.950	9.029 4.076 1518.602	8.911 3.947 1658.732	8.826 3.331 2034.303

Se menciona anteriormente que la métrica ISNR aumenta cuando la varianza del ruido aumenta y disminuye cuando el parámetro característico de la función de emborramiento crece; este comportamiento se explica gracias a que la métrica ISNR depende en el numerador de la diferencia al cuadrado entre la imagen degradada y la imagen original, lo cual si la varianza del ruido es grande esta diferencia será grande, mientras que el denominador, que es el error cuadrático medio, será pequeño debido a que el filtro busca su minimización, la combinación de estas dos características explican las variaciones de ISNR debido a su varianza. Los cambios en la métrica producidos por el crecimiento de el

parámetro de la función de emborramiento se deben a que entre mayor sea éste más estrecho es el espectro de la PSF, lo que produce un mayor efecto de filtro pasa bajo y mayor pérdida del detalle en la imagen. Los mejores resultados de mejoramiento aparecen cuando la varianza es máxima y el emborramiento mínimo.

3 LA INFERENCIA BAYESIANA Y LOS CAMPOS ALEATORES COMPUESTOS DE GAUSS-MARKOV EN LA REDUCCIÓN DE RUIDO EN IMÁGENES DIGITALES

La gran mayoría de los algoritmos presentados en el capítulo I pueden ser formulados a partir de la teoría de la inferencia bayesiana, ejemplos típicos de esto son el método de máxima entropía y el método de Richardson-Lucy..

Para profundizar un poco más en el campo de la restauración digital de imágenes el otro método estudiado para implementar se basa en los principios de la Inferencia Bayesiana. Este tipo de inferencia busca incluir información antecedente sobre la imagen que se desea estimar.

3.1 INFERENCIA BAYESIANA

Para generar los modelos que se necesitan en este tipo de inferencia, se comienza por distinguir entre f la imagen 'verdadera', la cual es observada en condiciones de no ruido y no emborramiento y g la imagen observada. Los métodos Bayesianos comienzan por definir una distribución antecedente $P(f)$, donde se incorpora la información de la estructura esperada dentro de la imagen; y $P(g|f)$ que se interpreta como la probabilidad de observar la imagen g siendo f la imagen verdadera. La inferencia Bayesiana afirma que la estimación de f se debe obtener de $P(f|g)$ [6]. Con la ayuda del teorema de Bayes $P(f|g)$ vendrá dada por:

$$P(f|g) = \frac{P(g|f) \cdot P(f)}{P(g)} \quad (3.1)$$

$P(g)$ es la distribución de probabilidad de obtener la imagen g . Esta es independiente de f y por lo tanto no influye en la estimación de la imagen "verdadera".

Se debe encontrar entonces la imagen f más probable dado la imagen observada g . Esto es conocido como maximum a posteriori (MAP). La forma de encontrarlo es minimizar el riesgo Bayesiano dado por:

$$R(\hat{f}) = \int C(\hat{f}, f) \cdot P(f|g) \cdot df$$

Donde g es la imagen observada, $C(\hat{f}, f)$ la función de costo y $P(f|g)$ la distribución posterior [18]

$$C(\hat{f}, f) = 1 - \delta(\hat{f}, f) \quad \delta(\hat{f}, f) = 1 \Leftrightarrow \hat{f} = f$$

Es decir todos los estimados diferentes al original tienen el mismo costo. Por lo tanto el MAP de f esta dado por:

$$\hat{f} = \arg \max_f (P(f | g)) \quad (3.2)$$

De la expresión (3.1) se observa que solo en numerador depende de f por lo que:

$$\hat{f} = \arg \max_f (P(g | f) \cdot P(f)) \quad (3.3)$$

Al observar las anteriores expresiones surge la pregunta: ¿Cuál es la forma de modelar las expresiones de probabilidad que se definen en la inferencia bayesiana? al contestar esta pregunta surgen los diversos métodos y algoritmos, que basados en la expresión (3.3), fueron presentados en el capítulo uno. Resulta claro que tanto para $P(g | f)$ y $P(f)$ se debe formular una expresión concreta que resultará de la forma escogida para modelar a cada una respectivamente.

La precisión de observar g dado f dependerá del ruido presente en la ecuación que modela el proceso de adquisición de la imagen, (1.4), por lo tanto hablar de $P(g | f)$ será equivalente a hablar de la distribución del ruido. $P(f)$ por otra parte modela la información antecedente que se tiene de la forma de f y la elección del modelo dará la forma de $P(f)$.

3.2 MODELO DEL RUIDO

Por simplicidad y conveniencia se supondrá que el ruido es aditivo independiente de la imagen y que su distribución será una gaussiana con media cero y varianza σ_n^2 , donde n es el ruido presente en la ecuación (1.4):

$$P(g | f) \propto \exp\left(\frac{-\|g - H \cdot f\|^2}{2 \cdot \sigma_n^2}\right) \quad (3.4)$$

El distribución gaussiana modela una gran cantidad de procesos de ruido aditivo presentes en práctica y de los cuales se tienen propiedades y características más conocidas.

La elección del modelo del ruido depende mucho del proceso de adquisición, por ejemplo el ruido presente en las cámaras tipo CCD (sensor de dispositivos acoplados por carga) y en las imágenes medicas tomográficas se describe mediante la distribución de Poisson que describe el comportamiento de el ruido producido en estos procesos de captura de imágenes.

3.3 MODELO DE LA IMAGEN

Para describir los antecedentes de la imagen se utilizará la teoría de los Campos Aleatorios de Markov, que utiliza la relación existente entre píxeles vecinos para describir las propiedades locales de la imagen, el comportamiento global, la distribución conjunta, esta dado por la equivalencia Markov-Gibbs que establece el teorema de Hammersley-Clifford [19]. Para facilitar la comprensión del origen de las expresiones que al final serán obtenidas es necesario introducir algunos conceptos teóricos sobre los Campos Aleatorios de Markov.

3.4 CAMPOS ALEATORIOS DE MARKOV

Problema del etiquetado: “Muchos problemas de visión artificial pueden ser considerados como problemas de etiquetado, de manera que la solución al problema es el conjunto de etiquetas asignado a los píxeles de la imagen.” [20]

Este problema puede ser especificado en términos de un conjunto de estados, nodos, y un conjunto de etiquetas que puede ser continuo o discreto.

Sea S un conjunto de m estados, nodos:

$$S = \{1, 2, \dots, m\}$$

Donde $1, 2, \dots, m$ son índices. Una cuadrícula rectangular para una imagen en dos dimensiones de tamaño $n \times n$ puede ser notada por [19]:

$$S = \{(i, j) | 1 \leq i, j \leq n\}$$

Generalmente los estados (i, j) son re-indexados a un solo índice k de tal forma que k toma valores entre $\{1, 2, \dots, m\}$ donde $m = n \times n$. Los estados en S pueden estar homogéneamente espaciados o no esto permite clasificar a S como homogéneo o no homogéneo.

Una etiqueta es un evento que puede suceder en el estado, nodo. Un conjunto de etiquetas puede tomar valores en un intervalo continuo o discreto. Para el caso continuo puede ser la línea real o un intervalo compacto de esta [19].

$$L_c = [X_l, X_h] \subset \mathfrak{R}$$

Para el caso discreto, una etiqueta asume un valor desde un conjunto finito M de etiquetas:

$$L_d = \{1, 2, \dots, m\}$$

El problema del etiquetado consiste en asignar a cada estado de S una etiqueta de L . Cuando todos los estados tienen el mismo conjunto de etiquetas L , el conjunto de todas las configuraciones posibles de etiquetas es el producto cartesiano:

$$\ell = \underbrace{L \times L \times \dots \times L}_{m \text{ veces}} \text{ Donde } m \text{ es el tamaño de } S.$$

3.4.1 Campos aleatorios de Markov y distribución de Gibbs.

De [20] se extraen las siguientes definiciones:

Definición 3.1: Se le llama sistema de vecinos para el conjunto de estados o nodos S a:

$$N = \{N_i / \forall i \in S\}$$

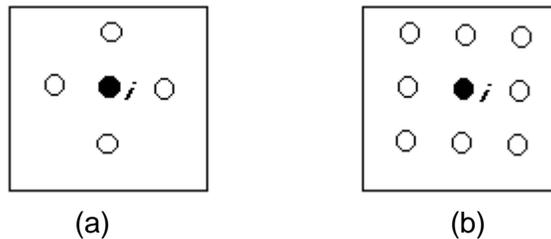
Donde N_i es el conjunto de estados o nodos vecinos a i para los que:

1. $i \notin N_i$
2. $i \in N_j \Leftrightarrow j \in N_i$

N_i es el conjunto de estados dentro de un radio r .

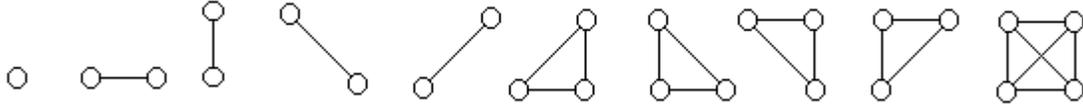
Por ejemplo, en una cuadrícula de dos dimensiones, si $r=1$ una vecindad N_i esta formada por los cuatro nodos que se encuentran inmediatamente arriba, abajo a la derecha y a la izquierda.

Figura 3.1 a) Sistema de vecinos de 4 nodos. b) Sistema de vecinos 8 nodos



Definición 3.2: Un clique c para $\{S, N\}$ se define como un conjunto de S tal que c consiste de un único estado $c = \{i\}$ o un par de estados vecinos $c = \{i, j\}$, o tres estados vecinos $c = \{i, j, k\}$ y así sucesivamente dependiendo del tamaño de la vecindad. Para un sistema de cuatro vecinos los cliques serán hasta de tamaño dos, para uno de ocho vecinos serán hasta de tamaño cuatro.

Figura 3.2 Cliques de tipo 1, 2, 3 y 4.



Se denota el conjunto de cliques de un solo estado, el conjunto de cliques de dos estados, ..., por C_1, C_2, \dots respectivamente donde:

$$C_1 = \{i / i \in S\}$$

$$C_2 = \{(i, j) / j \in N_i \quad i \in S\}$$

$$C_3 = \{(i, j, k) / i, j, k \in S \text{ son vecinos dos a dos}\}$$

El conjunto de todos los cliques para $\{S, N\}$ será entonces $C = C_1 \cup C_2 \cup \dots$ hasta el tamaño de cliques definido por el sistema de vecindades.

Definición 3.3: Sea $F = \{F_1, F_2, \dots, F_m\}$ una familia de variables aleatorias definidas en S , donde cada variable aleatoria F_i toma un valor f_i en L . Se llamará Campo aleatorio a la familia de variables F_i . La notación $F_i = f_i$ significará que el suceso F_i toma el valor f_i .

Definición 3.4: Un suceso conjunto $\{F_1 = f_1, F_2 = f_2, \dots, F_m = f_m\}$, abreviadamente $F = f$, es una realización de F , Se llamará configuración del suceso conjunto F a $f = \{f_1, f_2, \dots, f_m\}$.

Para especificar la densidad de probabilidad se utilizara la letra p minúscula.

Definición 3.5: Una familia de variables aleatorias F se dice que es un Campo Aleatorio de Markov sobre S con respecto a N si y solo si:

1. $p(F = f) > 0 \quad \forall f \in \ell$
2. $p(F_i = f_i / F_j = f_j, j \in S; j \neq i) = p(F_i = f_i / F_j = f_j, j \in N_i)$ (Propiedad Markoviana).

Definición 3.6: Un conjunto de variables aleatorias F es un Campo Aleatorio de Gibbs en S con respecto a N si y sólo si sus configuraciones siguen una distribución de Gibbs, una distribución de Gibbs tiene la siguiente forma de función de densidad:

$$p(f) = \frac{1}{Z} \exp\left(-\frac{1}{T} U(f)\right)$$

Donde:

$U(f)$ = Función de energía.

Z = Constante de normalización

T = Temperatura.

La función de energía $U(f)$ viene dada por:

$$U(f) = \sum_{c \in C} V_c(f)$$

$V_c(f)$ es una función con la propiedad que depende solamente de los f_i de f par los cuales $i \in c$.

Este tipo de distribuciones es común encontrarla para modelar “estados de equilibrio” para sistemas físicos, por ejemplo el ferromagnetismo.

Teorema 3.1: (Hammersley-Clifford) F es un Campo Aleatorio de Markov en S con respecto a N si y solo si F es un campo aleatorio de Gibbs en S con respecto a N .

El anterior teorema permite moverse entre las características locales, especificadas por las funciones potenciales $V_c(f)$, y las características globales expresadas por la distribución conjunta de Gibbs.

3.5 FORMA DE LA DISTRIBUCIÓN POSTERIOR

Debido al conocimiento antecedente sobre la suavidad de la distribución de la luminosidad de los objetos, es posible modelar la distribución de f por un modelo Condicional Auto-Regresivo (CAR) [21]

$$p(f) = \frac{1}{Z_1} \exp\left\{-\frac{f^t \cdot (I - \phi \cdot N) \cdot f}{2 \cdot \sigma_w^2}\right\} \quad (3.5)$$

Donde:

Z_1 es la constante de normalización.

$N_{i,j} = 1$ si las celdas i y j son espacialmente vecinas, cero en otro caso,

ϕ menor o igual que 0.25.

Los parámetros pueden ser interpretados al seguir las expresiones en las distribuciones condicionales:

$$E(f(i) | f(j), j \neq i) = \phi \cdot \sum_{j \text{ vecino de } i} f(j) \quad (3.6)$$

$$\text{var}(f(i) | f(j), j \neq i) = \sigma_w^2$$

La esperanza condicional se halla para los cuatro vecinos de i ver Figura 4.1, y el parámetro σ_w^2 mide la suavidad de la imagen “verdadera”.

Si $i+1$, $i+2$, $i+3$, $i+4$ denotan los cuatro píxeles alrededor de i como en la figura 4.1, y si se asume “corrección toroidal de bordes” lo que implica que los píxeles inferiores son vecinos de los superiores y que los píxeles del extremo derecho son vecinos de los que encuentran en el extremo izquierdo; si se dibujara esta situación tendría forma de un toroide [21]:

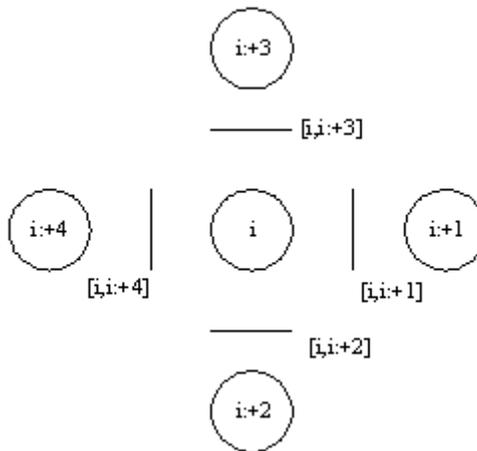


Figura 3.3 Vecinos del píxel i , y su proceso de línea.

$$p(f) = \frac{1}{Z_1} \exp \left\{ - \frac{\sum_i [\phi \cdot (f(i) - f(i+1))^2 + \phi \cdot (f(i) - f(i+2))^2 + (1 - 4 \cdot \phi) \cdot f^2(i)]}{2 \cdot \sigma_w^2} \right\} \quad (3.7)$$

Si se observa la ecuación (3.7) se nota que esta es una distribución de Gibbs, donde el etiquetado es continuo, y los potenciales $V_c(f)$ que contribuyen a la energía total $U(f)$ son del tipo $V_1(f)$ y $V_2(f)$ es decir formados por los cliques de uno y dos píxeles.

3.6 MANEJO DE DISCONTINUIDADES

Resolver la ecuación (1.2), de adquisición, se afirmó anteriormente pertenece a el dominio de los problemas inversos mal condicionados. Para hacer que el problema de restauración esté bien condicionado la teoría de la regulación asume condiciones antecedentes adicionales; una importante asunción es que la imagen sea suave. Desde el punto de vista probabilístico una solución por vía de la regulación corresponde al estimado maximun a posteriori (MAP) de un Campo Aleatorio de Markov (CAM).

Los CAM proveen mas generalidad que la regulación, primero por que permite incluir restricciones distintas a la de la suavidad, y por que se puede definir un sistema de vecinos diferente a los más cercanos [19].

Debido a la estrecha relación entre continuidad y suavidad, si reformulamos el problema e introducimos una expresión, para penalizar las soluciones poco suaves, de la forma

$$\sum_{n=1}^N \lambda_n \cdot \int_a^b g(f^{(n)}(x)) \cdot dx \quad \text{donde } N \text{ es el orden de derivadas más alto a considerar, } \lambda_n \text{ un}$$

factor de peso y $g(f^{(n)}(x))$ es el costo de la irregularidad en $f^{(n-1)}$ y es equivalente a los potenciales antecedentes de cliques para los CAM. La aplicación homogénea o isotrópica de el operador de regulación inevitablemente produce que se sobre elimina las discontinuidades, donde la derivada es infinita. Aquí se esta tratando el problema sin digitalizar la imagen por lo que las expresiones de integración no son incorrectas.

3.6.1 Proceso de línea

Aplicando este modelo se asume que la imagen es suave a trozos por lo cual la condición de suavidad es “apagada” en puntos donde la magnitud de la derivada de la señal excede cierto límite [19]. Para este modelo ya es necesario considerar la cuadrícula, píxeles, mejor que un dominio continuo; se introduce un conjunto de variables binarias $l_i = 0$ o $l_i = 1$ dentro del término $U(f)$ de la distribución de Gibbs.

Para introducir el proceso de línea se reescribe la expresión (3.7) como sigue [21]:

$$\begin{aligned}
 p(f, l) = \frac{1}{Z_1} \exp \left\{ -\frac{1}{2 \cdot \sigma_w^2} \sum_i [\phi \cdot (f(i) - f(i: +1))^2 \times (1 - l([i, i: +1])) + \beta \cdot l([i, i: +1]) \right. \\
 \left. + \phi \cdot (f(i) - f(i: +2))^2 \times (1 - l([i, i: +2])) + \beta \cdot l([i, i: +2]) \right. \\
 \left. + (1 - 4 \cdot \phi) \cdot f^2(i) \right\} \quad (3.8)
 \end{aligned}$$

La expresión (3.8) es un modelo compuesto debido a que incorpora el proceso o campo de intensidad, valores de cada píxel, y al proceso de línea para definir el modelo estadístico de la imagen f . Esta será por consiguiente la distribución antecedente.

Debido a que se incorporó el modelo de proceso de línea, el problema se amplía a encontrar también el campo l es decir:

$$(f, l) = \arg \max_{f, l} (p(f, l | g)) \quad (3.9)$$

3.7 DISTRIBUCIÓN A POSTERIORI

Con los modelos de ruido y la distribución antecedente se llegará a la expresión $p(f, l | g)$ buscada:

$$p(f, l | g) = \frac{p(g | f, l) \cdot p(f, l)}{p(g)}$$

Si se considera las variables g y l independientes, se tiene:

$$\begin{aligned}
 p(f, l | g) &= \frac{1}{Z_2} p(g | f) \cdot p(f, l) \\
 p(f, l | g) &= \frac{1}{Z_2} \exp \left\{ \frac{-\|g - H \cdot f\|^2}{2 \cdot \sigma_n^2} \right. \\
 &\quad - \frac{1}{2 \cdot \sigma_w^2} \sum_i \left[\phi \cdot (f(i) - f(i: +1))^2 \times (1 - l([i, i: +1])) \right. \\
 &\quad \left. + \beta \cdot l([i, i: +1]) + \phi \cdot (f(i) - f(i: +2))^2 \times (1 - l([i, i: +2])) \right. \\
 &\quad \left. + \beta \cdot l([i, i: +2]) + \phi \cdot (f(i) - f(i: +2))^2 \times (1 - l([i, i: +2])) \right. \\
 &\quad \left. + \beta \cdot l([i, i: +2]) + (1 - 4 \cdot \phi) \cdot f^2(i) \right] \left. \right\} \quad (3.10)
 \end{aligned}$$

Si se usa una notación simplificada:

$$p(f, l | g) = \frac{1}{Z_2} \exp[-(U(g | f) + U(f, l))] \quad (3.11)$$

Donde $U(g | f)$ es la contribución del modelo del ruido y $U(f, l)$ del modelo antecedente de f .

Resulta evidente que no hay una expresión directa para encontrar f y l , por otra parte la expresión (3.10) tiene un gran número de máximos locales, esto haría que métodos como el de Newton o del gradiente descendente no lleven a la mejor solución.

Para encontrar el máximo de esta función se usa el algoritmo de *simulated annealing*.

3.7.1 Los algoritmos para optimización *Simulated Annealing* y *Gibbs Sampling*

La distribución de Gibbs se usa para describir la probabilidad de una configuración de acuerdo a su energía. De su expresión general se deduce que el estado más probable es aquel de mínima energía. Para obtener este estado de mínima energía algunos procedimientos de la metalurgia enfrían lentamente los materiales y así los electrones alcanzan un estado de más estabilidad, produciendo metales más duros. El proceso es similar en la formación de cristales; el horario para llevar a cabo el enfriamiento es muy importante, si el cambio en la temperatura es demasiado brusco, el proceso de formado de cristales no formara un único y regular cristal, por el contrario formará muchos pequeños cristales de diferentes formas.

3.7.1.1 *Simulated annealing*

Es un algoritmo que busca simular estas técnicas anteriormente descritas para maximizar, u optimizar cierto tipo de funciones; es ampliamente usado en solución de problemas de orden combinatorial. La ubicación de los dispositivos en un circuito integrado pertenece a este tipo de problemas, por lo que *simulated annealing* es usado para optimizar la ubicación de los dispositivos dentro de las pastillas de silicio.

La forma básica del algoritmo es como sigue [19]:

1. Inicio
2. iniciar T y f
3. repita:
muestreo aleatorio de f desde $p_T(f)$
decrementar T
hasta que $T \rightarrow 0$
4. devuelva f

Se observa que el paso 3.1 necesita de un muestreo aleatorio, es en este punto donde el algoritmo conocido como *Gibbs sampling* entra a funcionar. El *Gibbs sampling* es una versión particular del algoritmo *Metropolis* perteneciente a los algoritmos de cadenas de Markov de Monte Carlo [22].

3.7.1.2 El algoritmo *Gibbs sampling*

Supóngase que se desea extraer una muestra desde la distribución conjunta de $F = \{F_1, F_2, \dots, F_n\}$ dada por $p(f_1, f_2, \dots, f_n)$ donde el rango de F_i puede ser continuo o discreto. El muestreador de Gibbs hace esto al remplazar repetidamente cada componente con un valor elegido desde la distribución condicional sobre todos los otros componentes. El procedimiento para generar $F^{(t)}$ desde $F^{(t-1)}$ se puede expresar como sigue [22]:

1. Elija $F_1^{(t)}$ desde la distribución de F_1 dado $f_2^{(t-1)}, f_3^{(t-1)}, \dots, f_n^{(t-1)}$.
2. Elija $F_2^{(t)}$ desde la distribución de F_2 dado $f_1^{(t)}, f_3^{(t-1)}, \dots, f_n^{(t-1)}$.
- ⋮
- i. Elija $F_i^{(t)}$ desde la distribución de F_i dado $f_1^{(t)}, \dots, f_{i-1}^{(t)}, f_{i+1}^{(t-1)}, \dots, f_n^{(t-1)}$.
- ⋮
- n. Elija $F_n^{(t)}$ desde la distribución de F_n dado $f_1^{(t)}, f_2^{(t)}, \dots, f_{n-1}^{(t)}$.

Se debe notar que el valor de F_{i-1} es usado inmediatamente cuando se elige el próximo valor para F_i

Debido a que el modelo usado para la imagen f es la distribución de Gibbs, y debido al teorema 1 las probabilidades condicionales dependerán solo de los vecinos de cada F_i . Para presentar el teorema que garantiza la convergencia del muestreo de Gibbs se define primero:

D : El conjunto de los estados, nodos, del proceso de línea.

Z : El conjunto de los estados, nodos, del proceso de intensidad, equivalente a la ubicación de cada píxel.

$S = Z \cup D$.

F : Es un campo aleatorio de etiquetado continuo.

f : Es la realización de F .

L : Es un campo aleatorio de etiquetado discreto.

l : Es una realización de L .

$X = (F, L)$.

Ω es el espacio muestral de X tal que $(f, l) \in \Omega$.

$\pi(f, l)$ representa la distribución de Gibbs con respecto a (f, l) .

n_1, n_2, \dots es la secuencia en que los nodos de S son visitados $n_t \in S$.

La evolución de $X^{(t-1)} \rightarrow X^{(t)}$ se expresa en términos matemáticos por:

$$p(X_i^{(t)} = (f_s, l_s), i \in S) = \pi(X_{n_t} = (f_{n_t}, l_{n_t}) | X_i = (f_i, l_i), i \neq n_t) \cdot p(X_i^{(t-1)} = (f_i, l_i), i \neq n_t)$$

Teorema 3.2: Se asume que para cada $i \in S$. La secuencia $\{n_t \in S, t \geq 1\}$ visita a i infinitamente a menudo. Entonces para toda configuración inicial $(\eta_f, \eta_l) \in \Omega$ y cada $(f, l) \in \Omega$:

$$\lim_{t \rightarrow \infty} p(X^{(t)} = (f, l) | X^{(0)} = (\eta_f, \eta_l)) = \pi(f, l)$$

Una demostración que implica algunas limitaciones sobre el espacio muestral Ω es dada por los hermanos Geman en [23], una versión más fuerte de la demostración por Jeng y Woods en [23]. El teorema (3.2) afirma básicamente que después de infinitas evoluciones $X^{(t-1)} \rightarrow X^{(t)}$ las muestras obtenidas tendrán la distribución de probabilidad $\pi(f, l)$.

Por lo tanto es necesario obtener las distribuciones condicionales para cada f_i y l_i . Si la probabilidad conjunta para f , l y g es $p(f, l, g)$, y por la independencia de g y l entonces:

$$p(f, l, g) = p(g | f, l) \cdot p(f, l) = p(g | f) p(f, l) \quad (3.12)$$

$$\begin{aligned} p(l[i, j] | l[k, l], [i, j] \neq [k, l], f, g) &= \frac{p(g | f) \cdot p(f, l)}{\sum_{l_i=0}^1 [p(g | f) \cdot p(f, l)]} \\ &= \frac{\exp(-U(f, l))}{\sum_{l_i=0}^1 \exp(-U(f, l))} \\ &= \frac{1}{Z_3} \exp \left[-\frac{[(f(i) - f(j))^2 \times (1 - l([i, j])) + \beta \cdot l([i, j]))]}{2 \cdot T \cdot \sigma_w^2} \right] \end{aligned} \quad (3.13)$$

donde

$$Z_3 = \exp \left[-\frac{[(f(i) - f(j))^2 + \beta]}{2 \cdot T \cdot \sigma_w^2} \right]$$

Repitiendo el proceso para f_i :

$$p(f_i | f_j, \forall i \neq j, g, l) = \frac{p(g | f) \cdot p(f, l)}{\int_{-\infty}^{\infty} p(g | f) \cdot p(f, l) \cdot df_i}$$

Realizando las respectivas operaciones:

$$p(f_i | f_j, \forall i \neq j, g, l) = \frac{1}{\sqrt{2 \cdot \pi \cdot T \cdot \sigma^{2l[i]}}} \exp \left[\frac{-(f(i) - \mu^{l[i]})^2}{2 \cdot T \cdot \sigma^{2l[i]}} \right] \quad (3.14)$$

Donde:

$$\begin{aligned} \mu^{l[i]} = & \lambda^{l[i]}(i) \phi \sum_{j \text{ vecinos de } i} \frac{f(j) \cdot (1 - l([i, j]))}{nn^{l[i]}(i)} \\ & + (1 - \lambda^{l[i]}(i)) \left(\frac{(H^T \cdot g)(i) - (H^T \cdot H \cdot f)(i)}{c} + f(i) \right) \end{aligned} \quad (3.15)$$

$$\sigma^{2l[i]} = \frac{\sigma_n^2 \sigma_w^2}{nn^{l[i]}(i) \sigma_n^2 + c \sigma_w^2} \quad (3.16)$$

$$nn^{l[i]}(i) = \phi \sum_{j \text{ vecinos de } i} (1 - l([i, j])) - (1 - 4\phi) \quad (3.17)$$

$$c = \sum_i h(i)^2 \quad (3.18)$$

donde $h(i)$ es el operador de convolución.

$$\lambda^{l[i]}(i) = \frac{nn^{l[i]}(i) \cdot \sigma_n^2}{nn^{l[i]}(i) \cdot \sigma_n^2 + c \sigma_n^2} \quad (3.19)$$

En las expresiones anteriores se ha introducido la variable T debido a que ésta permite ejecutar el algoritmo de *simulated annealing*. La expresión (3.14) corresponde a la distribución normal por lo que el algoritmo de restauración se ubica en los Campos Aleatorios Compuestos de Gauss-Markov (CACGM).

La presencia de emborramiento hace que al aplicar *simulated annealing* con las expresiones (3.13) y (3.14) para hacer el muestreo aleatorio no garantice converger a el *maximum a posteriori*. En ausencia de emborramiento el siguiente teorema garantiza la convergencia: y finalmente es la expresión que se implementará en la aplicación.

Teorema 3.3: Si las siguientes condiciones se satisfacen:

1. $|\phi| < 25$.
2. $T(t) \rightarrow 0$ cuando $t \rightarrow \infty$ tal que:
3. $T(t) \geq C_T / \log(1 + k(t))$

Entonces para alguna configuración inicial f_{-1}, l_{-1} se tiene:

$$p(f_t, l_t | f_{-1}, l_{-1}, g) \rightarrow p_0(f, l) \text{ cuando } t \rightarrow \infty$$

Donde $p_0(\dots, \dots)$ es la distribución uniforme de probabilidad sobre la solución MAP, C_T es una constante positiva y $k(t)$ es el número de veces que se ha barrido el campo aleatorio hasta el tiempo t .

Lo que afirma el teorema 3.3 es que *simulated annealing* extrae la solución desde el espacio muestral comprendido por las soluciones MAP, al ser $p_0(f, l)$ uniforme todos los máximos tienen la misma probabilidad de ocurrir, incluyendo el máximo global; es decir si se visita el espacio muestral las suficientes veces se extraerán todos los elementos de este al menos una vez. Entonces cuando el máximo global ocurra el algoritmo no permite un salto a una condición de mayor energía $U(f | g) + U(f | l)$ gracias al continuo decrecimiento de la variable T . Si un máximo local ocurre el algoritmo permite salta al estado de mínima energía siguiente.

Para garantizar por lo menos la convergencia de *simulated annealing* a un mínimo local los autores de [21] modifican la expresión (3.14) para obtener:

$$p(f_i | f_j, \forall i \neq j, g, l) = \frac{1}{\sqrt{2 \cdot \pi \cdot T \cdot \sigma_m^{2l_{t-1}^{[i]}}}} \exp \left[\frac{-\left(f(i) - \mu_m^{l_{t-1}^{[i]}}\right)^2}{2 \cdot T \cdot \sigma_m^{2l_{t-1}^{[i]}}} \right] \quad (3.20)$$

$$\mu_m^{l_{t-1}^{[i]}} = \omega^{l_{t-1}^{[i]}}(i) f_{t_{k-1}}(i) + \left(1 - \omega^{l_{t-1}^{[i]}}(i)\right) \mu^{l_{t-1}^{[i]}} \quad (3.21)$$

$$\sigma_m^{2l_{t-1}^{[i]}} = \left(1 - \left(\omega^{l_{t-1}^{[i]}}(i)\right)^2\right) \sigma^{2l_{t-1}^{[i]}} \quad (3.22)$$

$$\omega^{l_{t-1}^{[i]}} = \left[\sigma_n^2 \left(1 - n \omega^{l_{t-1}^{[i]}}(i)\right) + (1 + c) \sigma_w^2 \right] / \left(\sigma_n^2 + \sigma_w^2 \right) \quad (3.23)$$

Los subíndices t_{k-1} indican que es el valor del barrido inmediatamente anterior y el subíndice $t-1$ que se trabaja con los datos de la última actualización del campo. Con estas modificaciones el algoritmo de *simulated annealing* se describe [21]:

1. Fije $t = 0$ e inicie las configuraciones iniciales de f y l notadas por f_{-1}, l_{-1} , y la temperatura inicial $T(0) = 1$.
2. La evolución del proceso de línea, $l_{t-1} \rightarrow l_t$ se obtiene al muestrear el próximo punto desde un esquema de barrido basado en las funciones de distribución de probabilidad definida por la ecuación (3.13) y manteniendo el resto de los l_{t-1} sin cambiar.
3. haga $t = t + 1$ hasta que el barrido total del campo este completo.
4. La evolución del proceso de intensidad, $f_{t-1} \rightarrow f_t$ por muestrear el próximo valor de la imagen en conjunto desde la función de distribución de probabilidad definida en la expresión 15.
5. Ir a 2 hasta que $t > t_f$, t_f es un entero específico.

En [19] se garantiza la convergencia de este algoritmo formulando el siguiente teorema:

Teorema 3.4: Si las siguientes condiciones se satisfacen:

1. $|\phi| < 25$.
2. $T(t) \rightarrow 0$ cuando $t \rightarrow \infty$ tal que:
3. $T(t) \geq C_T / \log(1 + k(t))$

entonces para alguna configuración inicial f_{-1}, l_{-1} se tiene:

$$p(f_t, l_t | f_{-1}, l_{-1}, g) \rightarrow p_0(f, l) \text{ cuando } t \rightarrow \infty$$

Donde $p_0(\dots, \dots)$ es la distribución cualquiera de probabilidad sobre una solución MAP, C_T es una constante positiva y $k(t)$ es el número de veces que se ha barrido el campo aleatorio hasta el tiempo t . Lo que como el teorema anterior garantiza la convergencia pero a un máximo local, debido a la distribución es arbitraria no se garantiza que al muestrear un número suficiente de veces se extraigan al menos una vez cada elemento del espacio de soluciones MAP.

3.7.2 Cálculo del número de operaciones para medir la complejidad de los algoritmos

Para tener una medida de la complejidad de los algoritmos se expresará el número de operaciones que cada uno realiza de acuerdo al tamaño en píxeles de la imagen. El algoritmo basado en CACGM para actualizar los procesos de línea hace mínimo siete operaciones por cada píxel, como se utilizan dos arreglos del mismo tamaño de la imagen para almacenar tanto las líneas verticales como horizontales el número total de

operaciones para este procedimiento será $7 * 2 * M_f * M_c$ donde M_f y M_c son el número de filas y de columnas respectivamente. Para actualizar los píxeles de la imagen se hacen 20 operaciones por píxel; estas operaciones se hacen por cada iteración. El número de operaciones es:

$$\text{Número operaciones CACGM} = 34 * M_f * M_c * \text{iteraciones} \quad (3.24)$$

Para la deconvolución de Wiener se necesitan dos FFT y una multiplicación compleja punto a punto de la transformada de la imagen degradada con el filtro, cada multiplicación compleja ejecuta 6 operaciones. El número de operaciones es:

$$\text{Número de operaciones Wiener} = 6 * M_f * M_c + 2 * M_f * M_c * \log_2(M_f * M_c) \quad (3.25)$$

Al comparar las dos expresiones y proponer la desigualdad tal que CACGM sea menor que Wiener se tiene que se debe cumplir $N > 2^C$ donde N sería el número de filas o columnas para una imagen cuadrada y C una constante que es directamente proporcional al número de iteraciones de el algoritmo basado en CACGM; el algoritmo *simulated annealing* es de convergencia lenta, y necesita alrededor de cinco mil operaciones para producir una solución, por lo que se debería cumplir por lo menos que $N > 2^{5000}$ esta sería una imagen exageradamente grande. Para una imagen de 256x256 píxeles se ejecutarán 11.141'120.000 operaciones. Con la deconvolución de Wiener se realizan 2'490.368; aproximadamente 4.000 veces menos operaciones que con el algoritmo basado en CACGM.

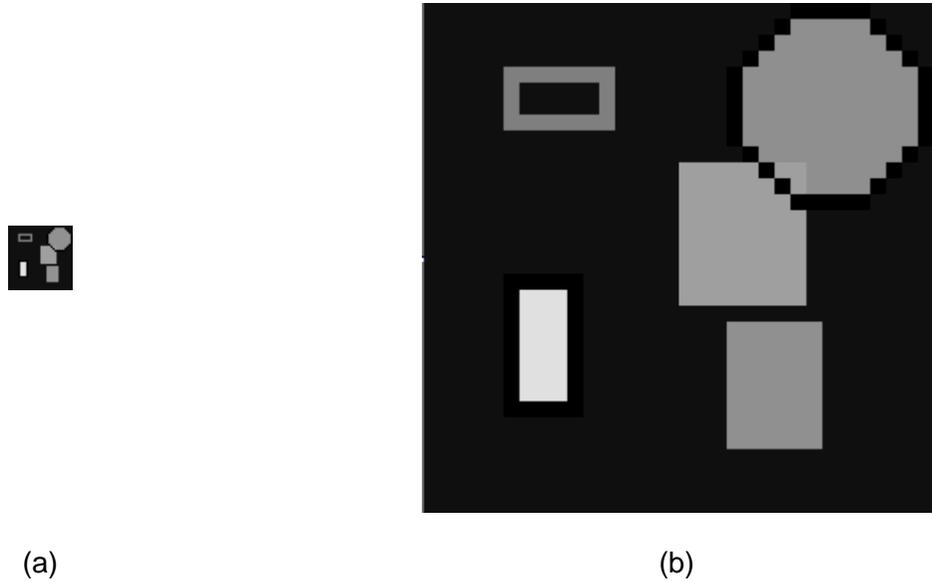
El factor de 34 en la expresión (3.24) depende del número de vecinos de un píxel, en este caso cuatro, por ejemplo si se considera vecinos a los píxeles en dirección diagonal el número de vecinos sería ocho y el factor de 34 se multiplicará al menos por dos.

3.7.3 Resultados y comparación de técnicas de reducción de ruido

A continuación se presentan los resultados del algoritmo para la reducción de ruido basado en CACGM, probando distintos parámetros σ_w^2 y α para su sintonización. Se debe destacar que un mayor valor en la métrica ISNR no es sinónimo de una imagen de mayor calidad según la subjetividad del ojo humano.

Para evadir el gran número de operaciones que una imagen de tamaño por lo menos 256x256 píxeles requiere, se trabajará con una de 32x32 que requiere 174'080.000 lo que es más razonable.

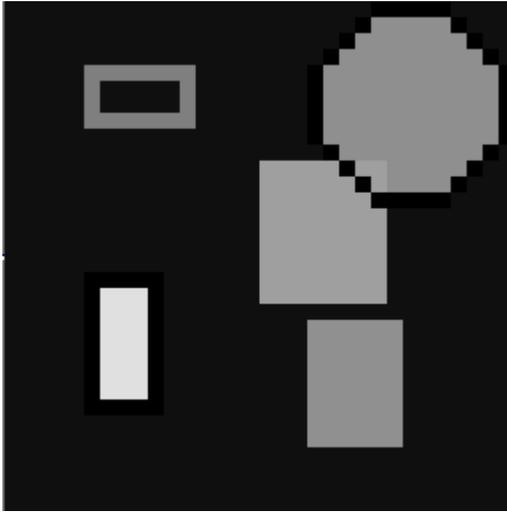
Figura 3.4 (a) imagen de prueba, tamaño 32*32 bytes. (b) imagen ampliada ocho veces



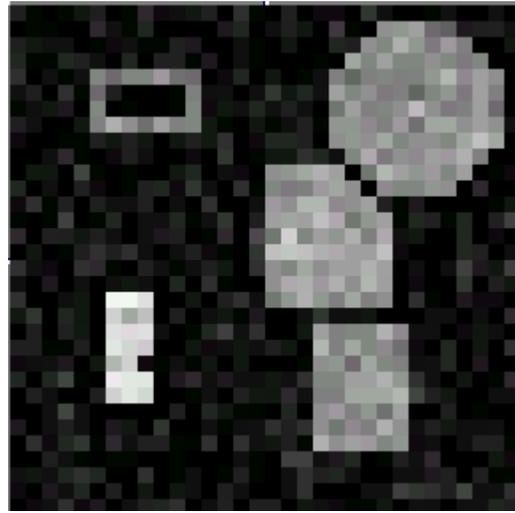
Para captar mejor lo producido por la técnica estudiada las imágenes resultantes, Figura 3.2 a Figura 3.12 todas tendrán una ampliación de 8.

El mejor resultado respecto a la métrica ISNR se tiene cuando $\sigma_w^2 = 200$, ISNR=51446 Db pero desde un punto de vista subjetivo, criterio propio, la mejor restauración se da cuando $\sigma_w^2 = 30$ y $\alpha = 2$ con un ISNR = 1.73806 Db. Se nota en los resultados que a medida que σ_w^2 decrece la imagen restaurada se oscurece; la razón de este fenómeno está en que σ_w^2 es una medida de la "suavidad" de la imagen, puesto que el fondo de esta es negro, una imagen suave no presentaría colores claros en cercanías del negro.

Figura 3.5 a) Imagen de prueba. b) Imagen con ruido SNR = 11.87314 Db.

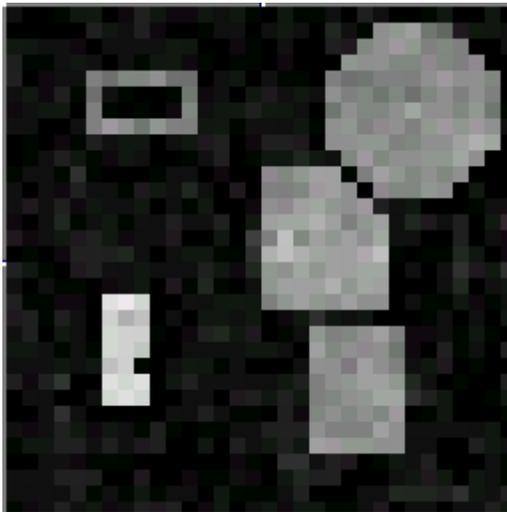


(a)

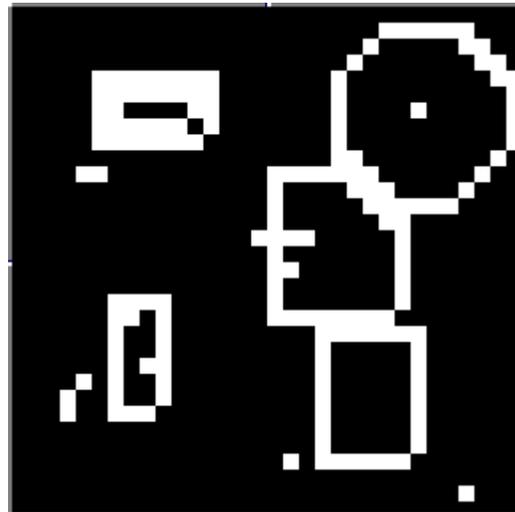


(b)

Figura 3.6 a) Imagen restaurada por CACGM, ISNR = 4.40675 Db; $\sigma_w^2 = 400$, $\alpha = 1$. b) proceso de línea resultante.

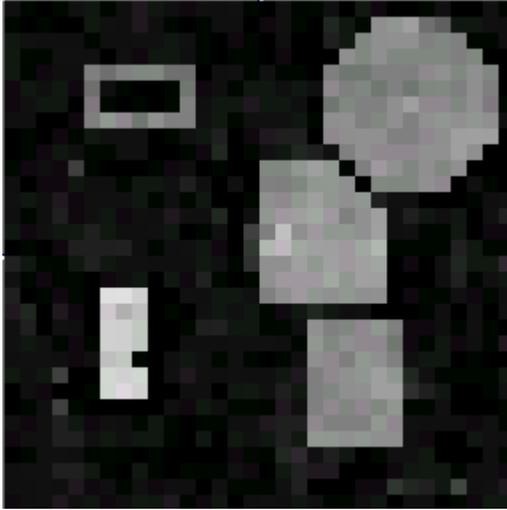


(a)

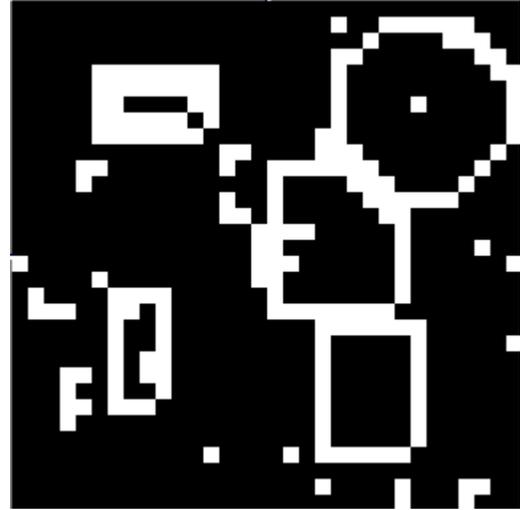


(b)

Figura 3.7 a) Imagen restaurada por CACGM, $\text{ISNR} = 5.1446 \text{ Db}$; $\sigma_w^2 = 200$, $\alpha = 1$. b) proceso de línea resultante.

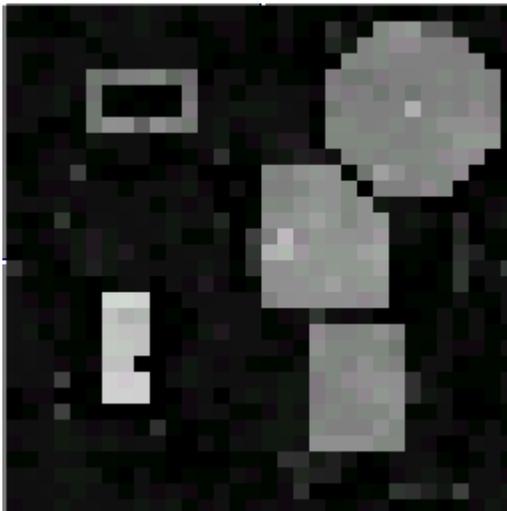


(a)

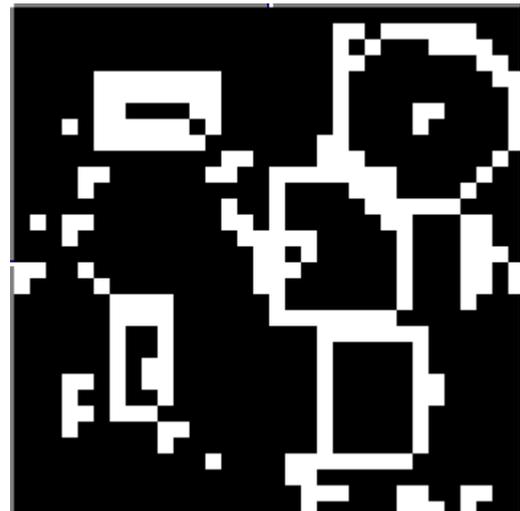


(b)

Figura 3.8 a) Imagen restaurada por CACGM, $\text{ISNR} = 4.78185 \text{ Db}$; $\sigma_w^2 = 100$, $\alpha = 1$. b) proceso de línea resultante.

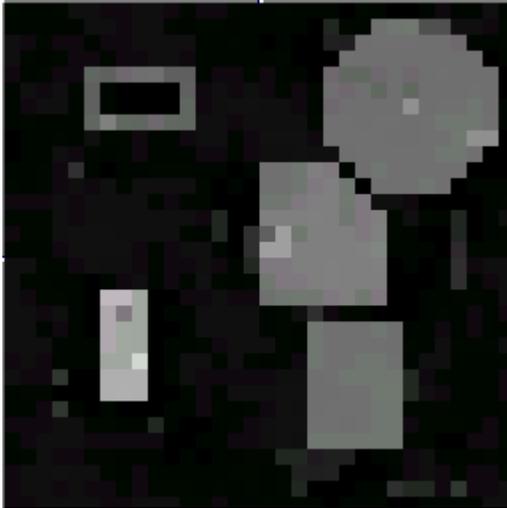


(a)

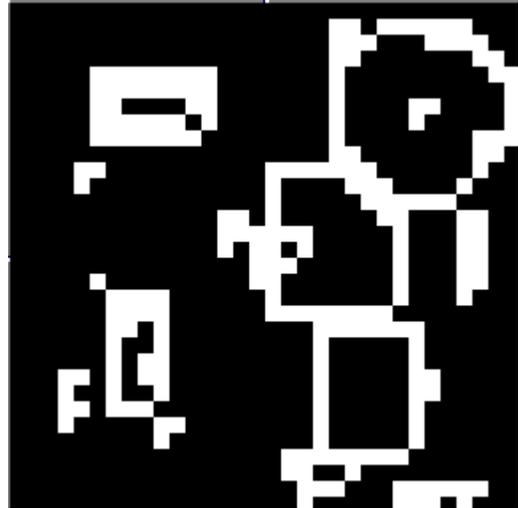


(b)

Figura 3.9 a) Imagen restaurada por CACGM, $\text{ISNR} = 1.0647 \text{ Db}$; $\sigma_w^2 = 30$, $\alpha = 1$. b) proceso de línea resultante.

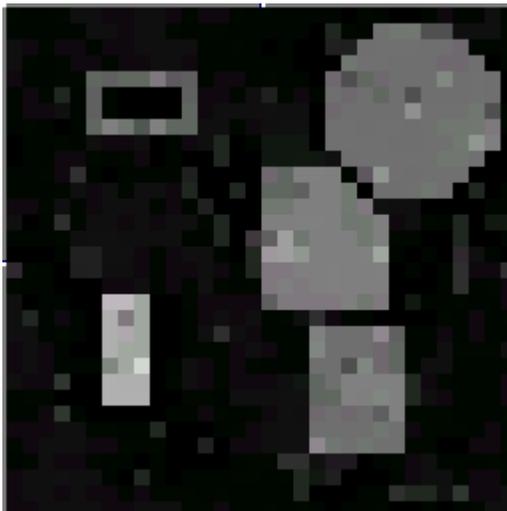


(a)

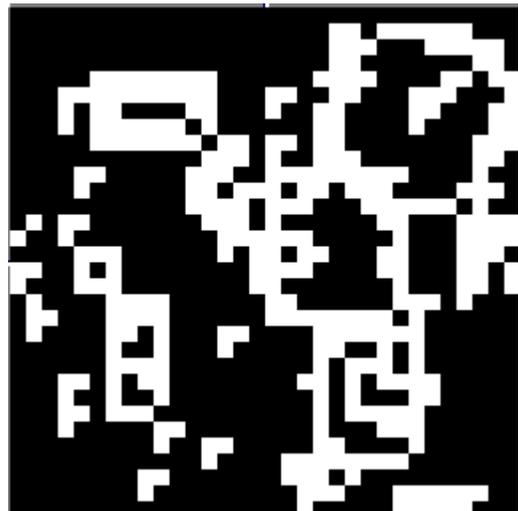


(b)

Figura 3.10 a) Imagen restaurada por CACGM, $\text{ISNR} = 0.58572 \text{ Db}$; $\sigma_w^2 = 30$, $\alpha = 0.7$. b) proceso de línea resultante.

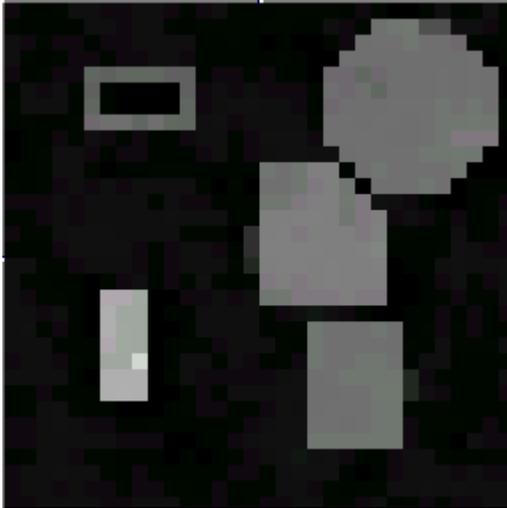


(a)

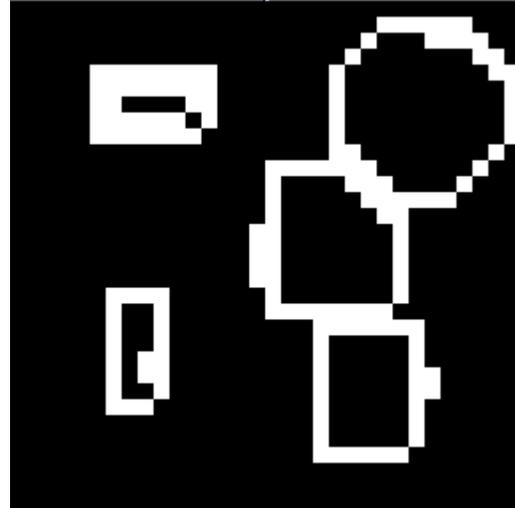


(b)

Figura 3.11 a) Imagen restaurada por CACGM, ISNR = 1.60815 Db; $\sigma_w^2 = 30$, $\alpha = 1.5$.
b) proceso de línea resultante.

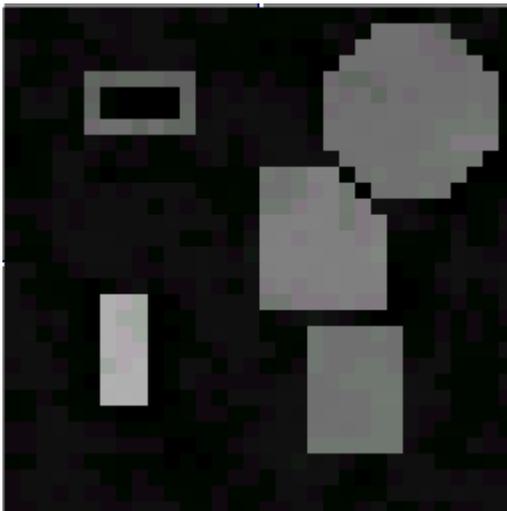


(a)

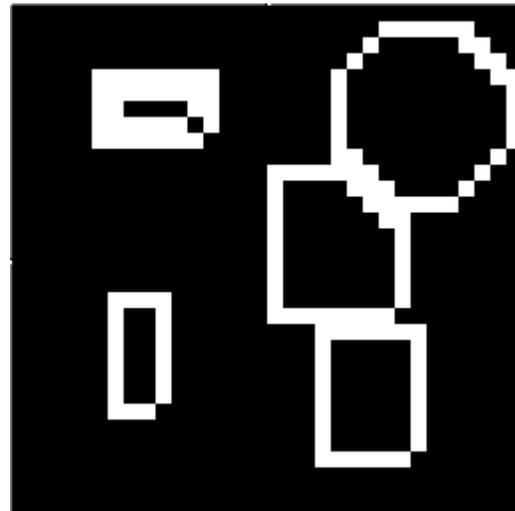


(b)

Figura 3.12 a) Imagen restaurada por CACGM MSNR = 1.72806 Db; $\sigma_w^2 = 30$, $\alpha = 2$.
b) proceso de línea resultante.



(a)

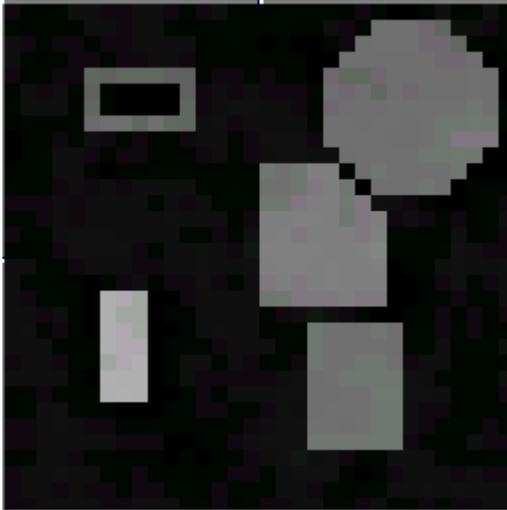


(b)

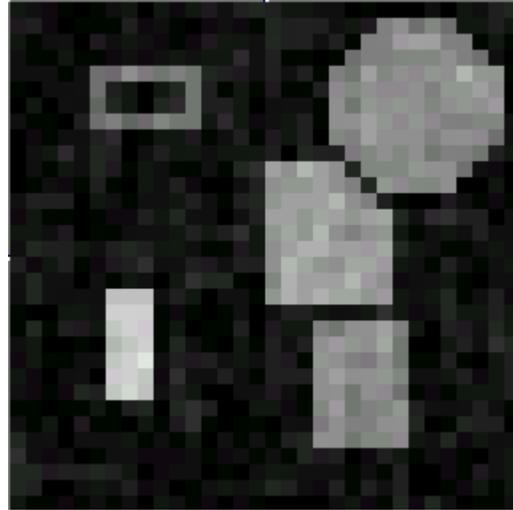
Las anteriores imágenes presentan los diferentes resultados en el proceso de sintonía de los parámetros α y σ_w^2 . El comportamiento de la salida mejora a medida que σ_w^2 se hace

más pequeño y α crece. Lo que significa que se cree que la imagen es suave y se penaliza fuertemente las discontinuidades. La estimación de α y σ_w^2 es heurística.

Figura 3.13 a) Imagen restaurada por CACGM ISNR = 1.72806 Db; $\sigma_w^2 = 30$, $\alpha = 2$. b) imagen restaurada por filtro de Wiener.

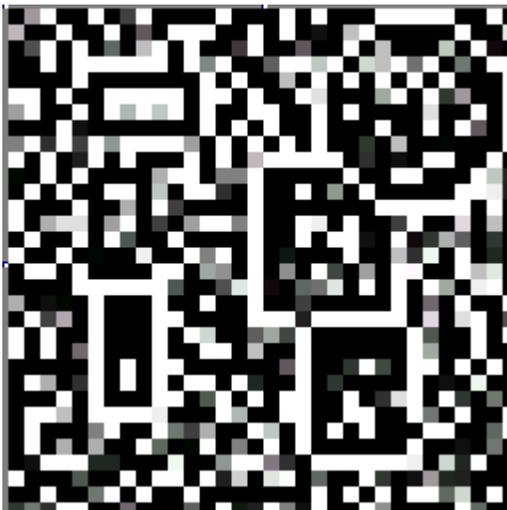


(a)

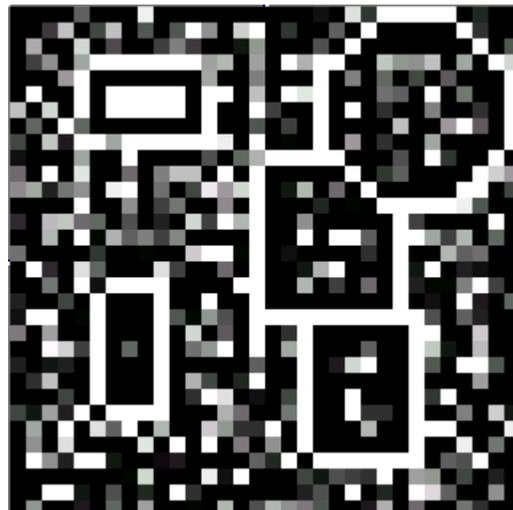


(b)

Figura 3.14 a) Bordos de la imagen degradada. b) bordos de la imagen restaurada por filtro de Wiener.

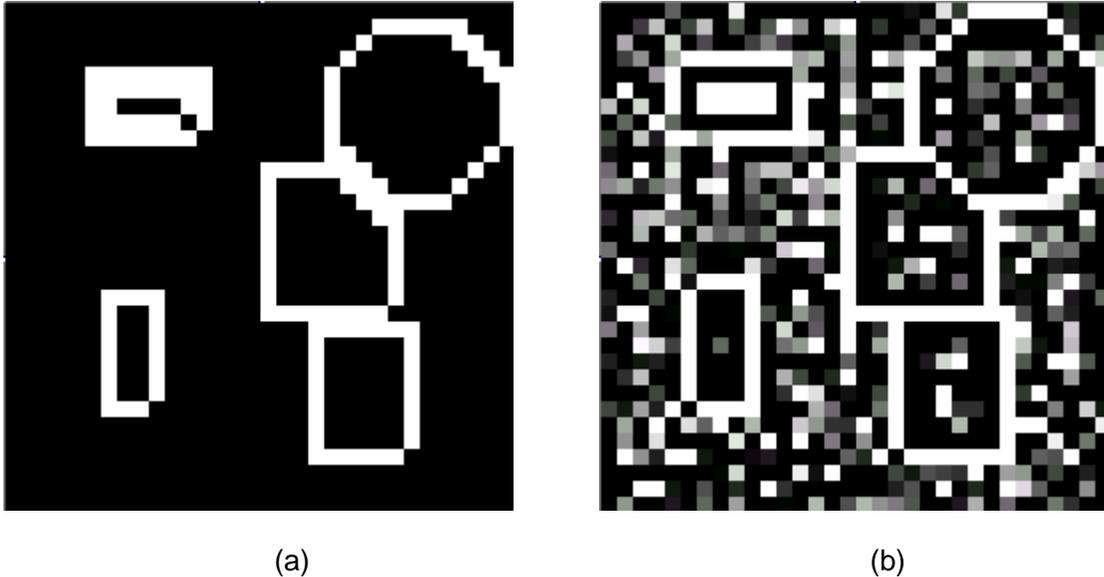


(a)



(b)

Figura 3.15 a) bordes detectados por CACGM con $\sigma_w^2 = 30$ y $\alpha = 2$. b) bordes de la imagen restaurada por filtro de Wiener.



La ventaja evidente de el método basado en CACGM sobre el filtro de Wiener esta en el proceso de línea, las secuelas del ruido en la salida del filtro hacen que la aplicación de detectores de bordes basados en el gradiente, sea poco conveniente debido a su sensibilidad al ruido; por otra parte el proceso de línea produce una muy buena detección de los bordes. La detección de bordes y la segmentación es un proceso fundamental en las aplicaciones de visión artificial.

4 CONCLUSIONES

A continuación se presentan las conclusiones más sobresalientes con respecto a los algoritmos de restauración estudiados. La formulación de estas se deriva de el análisis de las expresiones matemáticas obtenidas, de las gráficas y las tablas mostradas y de los resultados producidos por el software.

Además, se presentan algunas recomendaciones para futuros trabajos que deseen ampliar o seguir con esta línea de trabajo enmarcada en el procesamiento digital de imágenes y la visión artificial.

4.1 CONCLUSIONES PARA LA DECONVOLUCION DE WIENER

De las expresiones, las mediciones y observaciones obtenidas en el capítulo dos al abordar el método convencional de la deconvolución de Wiener se destacan las siguientes características:

- El filtraje no cambia la relación S/N en una frecuencia dada; por lo tanto no puede recuperar componentes de frecuencia que están totalmente enmascaradas por el ruido y como consecuencia las suprime.
- No logran súper-resolución. Esto es no llegan a recuperar componentes de frecuencia que están fuera de banda del sistema de degradación. (Todos los sistemas de adquisición de imágenes son limitados en banda por los límites de difracción de la luz).
- La Deconvolución de Wiener actúa como un filtro pasa-bajas dependiendo de la SNR en la frecuencia (u, v) ;
- Actúa como un filtro pasa-altas dependiendo de la OTF.
- La Deconvolución de Wiener controla el excesivo impulso que podría tener el inverso de la OTF en frecuencias con baja relación S/N, generalmente en altas frecuencias.
- El uso de estimados para la potencia de la imagen y de ruido, causa una pérdida de desempeño, caída de la ISNR, en el filtro que se refleja en la imagen estimada.

4.2 CONCLUSIONES PARA EL MÉTODO BASADO EN CACGM

- La inclusión del proceso de línea hace posible recuperar información de bordes en la imagen restaurada.
- El proceso de línea resulta adecuado debido a que produce la detección de bordes, etapa fundamental de los procesos de visión artificial. En una imagen ruidosa la detección de bordes por métodos convencionales como los basados en el gradiente fallan, por su alta sensibilidad al ruido.
- La sintonización de los parámetros involucrados en la restauración como α y σ_w^2 resultan fundamentales para el éxito del algoritmo. Se elección generalmente se hace mediante algún criterio heurístico.
- Los valores α y σ_w^2 no son constantes para la imagen, estos están estrechamente relacionados entre sí. Además también tienen una significativa dependencia con el valor de σ_n^2 . Igualar σ_w^2 a σ_n^2 y fijar α a uno es un criterio aceptable para sintonizar los parámetros.
- Debido a que el algoritmo de optimización es el *simulated annealing* de convergencia muy lenta el método estudiado no sería apto si se necesitarán resultados en tiempo real.
- El uso de la teoría de los Campos aleatorios de Markov permite un mejor modelamiento de las características locales y globales de una imagen.
- El uso de vecindades más grandes de píxeles mejoraría los resultados, pero incrementaría considerablemente la complejidad de las expresiones y el costo computacional al incrementar el factor numérico que acompaña a la expresión (3.24).
- El *maximun a posteriori* el cual se busca para estimar la imagen “verdadera” es el criterio que se usa para escoger una solución para el problema de restauración de imágenes, la forma como se lo encuentra es lo que puede llegar a constituir un algoritmo de restauración de imágenes..
- El hecho de que la métrica afirme que el grado de restauración ha sido grande, no implica que ante el criterio del observador la imagen tenga un grado alto de reducción de ruido.

- El éxito de asumir el modelo de campos aleatorios compuestos de Gauss-Markov radica en que la imagen sea representada por este modelo. Por ejemplo que el sistema de vecindades describa la relación existente entre píxeles vecinos.
- La ventaja de usar este tipo de modelo matemático es que proporciona la forma de llegar a las expresiones de comportamiento global con la ayuda de características locales de la imagen.

4.3 RECOMENDACIONES

Este trabajo de grado aborda la temática de la restauración digital de imágenes en una forma aún no muy profunda y poco práctica desde el punto de vista comercial, pero sirve de referencia para futuros trabajos de grado que quieran continuar con esta temática, al igual que proporciona una valiosa ayuda a los cursos de Procesamiento Digital de Imágenes en la temática de restauración. Se recomienda encaminar los futuros trabajos, si los hay, hacia la restauración de las imágenes médicas, debido a que en este tipo específico de imágenes se encuentra el verdadero valor práctico de los métodos basados en CACGM. Gracias a que los dispositivos e instrumentos médicos encargados de rayos X, tomografías sufren deterioro y las imágenes que producen pierden calidad y se degradan, restaurarlas es la mejor opción debido al elevado costo de los equipos de instrumentación médica.

Además la restauración no es la única área del procesamiento de imágenes ni de la visión artificial, estas áreas son de gran interés y de continuo desarrollo y es conveniente que los proyectos de La Facultad de Ingeniería Electrónica y Telecomunicaciones de La Universidad del Cauca exploren dichas temáticas.

5 MODELAMIENTO Y DISEÑO DE LA HERRAMIENTA SOFTWARE

Con el fin de facilitar el diseño de las interfaces y el análisis de los requerimientos, se utiliza el lenguaje de modelamiento UML, lenguaje de modelamiento unificado, y sus subproductos como son los diagramas de casos de uso, diagramas de colaboración de clases, y el diagrama de clases como herramienta de apoyo para la construcción, implementación y documentación final de la herramienta software de restauración, debido a su gran desempeño en relación con las metodologías de diseño orientado a objetos. Así se busca lograr una aplicación software que cumpla los objetivos teóricos propuestos y además proporcione una interfaz de usuario amigable.

5.1 CAPTURA DE REQUERIMIENTOS

5.1.1 Requerimientos no funcionales

Se desea una aplicación software capaz de leer una imagen BMP en escala de grises, y que a los datos obtenidos se les filtre con alguna de las siguientes funciones de emborramiento: emborramiento por movimiento, por turbulencia atmosférica y por fuera de foco; además de agregarle ruido blanco gaussiano, esto con el fin de simular el proceso de degradación de una imagen. Por otra parte se debe implementar la deconvolución de Wiener para restaurar la imagen degradada, y el filtro de Wiener o la técnica de reducción de ruido por Campos Aleatorios Compuesto de Gauss-Markov si para degradar la imagen solo se agregó ruido. Para facilitar la interacción con el usuario esta debe poseer una adecuada interfaz gráfica.

5.1.2 Requisitos funcionales

Como requisitos funcionales se tiene:

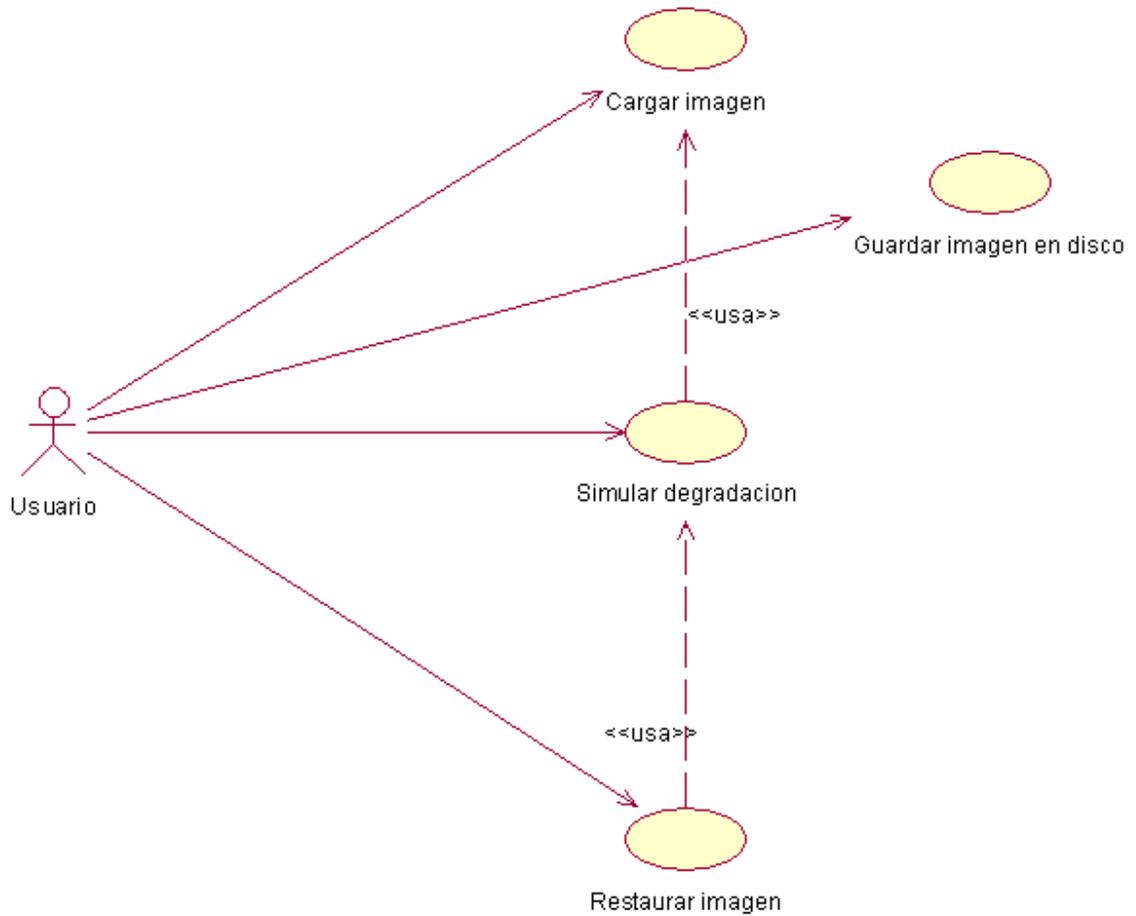
- La aplicación debe ser funcional bajo la plataforma de sistema operativo Linux.
- El código fuente de la aplicación debe estar implementado en el lenguaje C++.

5.2 DESCOMPOSICIÓN EN CASOS DE USO

Al leer los requerimientos no funcionales se concluye que la aplicación tendrá solamente un actor al que se llamará Usuario, que se descompone en cuatro casos de uso principales que son *Cargar imagen*, *Simular degradación*, *Restaura imagen*, *Guardar imagen en disco*. Una descomposición más profunda o con mayor número de casos de

uso no aportara significativamente ni a la comprensión, al diseño ni a la implementación de la aplicación que se desea lograr.

Figura 5.1 Diagrama de caos de uso herramienta para restauración de imágenes digitales



5.2.1 Descripción de los escenarios de casos de uso:

5.2.1.1 Abrir imagen

Iniciador: Usuario

Precondición: ninguna

Flujo de eventos

Flujo principal:

1. El usuario selecciona la opción Abrir desde el menú de Archivo, o el botón Abrir desde la barra de herramientas o presionando las teclas Ctrl-A.
2. Aparece el diálogo de Abrir Archivo donde el usuario ubica el directorio y el archivo a abrir, y luego presiona OK.
3. La aplicación comprueba si es un tipo de archivo válido.
4. Con el path obtenido a través de la ventana de diálogo Abrir la aplicación abre el archivo y despliega la imagen en el área de dibujo etiquetada como Original.

Post condiciones

1. Búfer cargado en memoria con los valores de los pixeles de la imagen, tipo unsigned char.
2. Búfer cargado en memoria con los valores de los pixeles de la imagen, del doble de elementos de búfer anterior y de tipo float.

Flujos alternativos

En 2, si el Usuario elige la opción Cancelar la ventana de diálogo Abrir desaparece y la aplicación retorna a su estado anterior.

En 3, si el tipo de archivo no es válido, se despliega un mensaje de error y se retorna a la ventana de diálogo Abrir.

Excepciones

1. Memoria insuficiente en el sistema.

Recursos especiales utilizados

1. Memoria dinámica

Interfaces de Usuario relacionadas:

- GUI_Inicial. (Contiene menús, barra de herramientas, áreas de dibujo y barra de progreso)
- Dialogo_Abrir (Contiene los botones OK y Cancelar y la forma de acceder a los directorios y archivos.)

5.2.1.2 Simular degradación

Iniciador: Usuario

Precondición: imagen cargada en memoria

Flujo de eventos

Flujo principal

1. El usuario selecciona la opción Simular desde el menú Acciones, o el botón Simular de la barra de herramientas o presiona F7.
2. Aparece la ventana de diálogo de Opciones de Degradación.
3. Si el usuario selecciona la opción Simular emborramiento se habilitan las entradas para esta opción.
4. El usuario debe escoger una función de emborramiento de las opciones señaladas en los radiobuttons.
5. El usuario debe introducir el parámetro de la función de emborramiento y la varianza del ruido en sus entradas respectivas y presiona OK.
6. La aplicación verifica la validez de las entradas.
7. La aplicación efectúa las operaciones para degradar la imagen y la despliega.

Post-condiciones

1. Búfer cargado en memoria con el doble de tamaño de la imagen original con los datos de la imagen degradada.
2. Búfer cargado en memoria con los valores de la función de emborramiento del mismo tamaño de la imagen

Flujos alternativos

En 3, si el Usuario no elige la opción de simular emborramiento deberá solamente entrar la varianza.

En 1, si el Usuario elige Cancelar la aplicación retorna a su estado anterior.

Excepciones:

1. Memoria insuficiente en el sistema.

Recursos especiales utilizados

1. Memoria dinámica

Interfaces de Usuario relacionadas

- GUI_Inicial. (Contiene menús, barra de herramientas, áreas de dibujo y barra de progreso)
- Dialogo_Opciones_Degradacion (Contiene los botones OK y Cancelar, un checkbox, un grupo de radiobutton y dos entradas de texto.)

5.2.1.3 Restaurar imagen

Iniciador: Usuario

Precondición: imagen degradada cargada en memoria

Flujo de eventos

Flujo principal

1. El usuario selecciona la opción Restaurar desde el menú Acciones, o el botón Restaurar de la barra de herramientas o presiona F8.
2. Si en la degradación no se simuló borrado aparece la ventana de Dialogo Seleccione Técnica de Reducción de Ruido.
3. Se hace la selección y se llama algoritmo correspondiente.
4. A medida que el algoritmo trabaja la barra de progreso se actualiza.
5. Al terminar la restauración se actualiza la imagen.

Post-condiciones

1. Búfer cargado en memoria con el doble de tamaño de la imagen original con los datos de la imagen restaurada.

Flujos alternativos

1. No existen

Excepciones

1. Memoria insuficiente en el sistema.

Recursos especiales utilizados

2. Memoria dinámica

Interfaces de Usuario relacionadas

- GUI_Inicial. (Contiene menús, barra de herramientas, áreas de dibujo y barra de progreso)
- Dialogo Seleccione Técnica de Reducción de Ruido (Contiene un botón de OK, y un grupo de radiobuttons.)

5.2.1.4 Guardar imagen en disco

Iniciador: Usuario

Precondición: imagen degradada y restaurada.

Flujo de eventos

Flujo principal

1. El usuario selecciona la opción Guardar desde el menú Archivo, o el botón Guardar de la barra de herramientas o presiona Ctrl-G.
2. Aparece el Diálogo Guardar desde el nombre del archivo. Se presiona OK
3. Se verifica la validez del nombre.
4. Se crea un archivo con el nombre dado y se escriben los datos.

Post-condiciones

1. Archivo creado en formato bmp con el nombre que se le dio a la imagen degradada.
2. Archivo creado en formato bmp con el nombre que se le dio a la imagen restaurada.

Flujos alternativos

1. En 3, si la verificación de la entrada del nombre del archivo no es válida despliega un mensaje de error y vuelve al diálogo Guardar.

Excepciones

1. Ninguna

Recursos especiales utilizados

1. Memoria en disco.

Interfaces de Usuario relacionadas

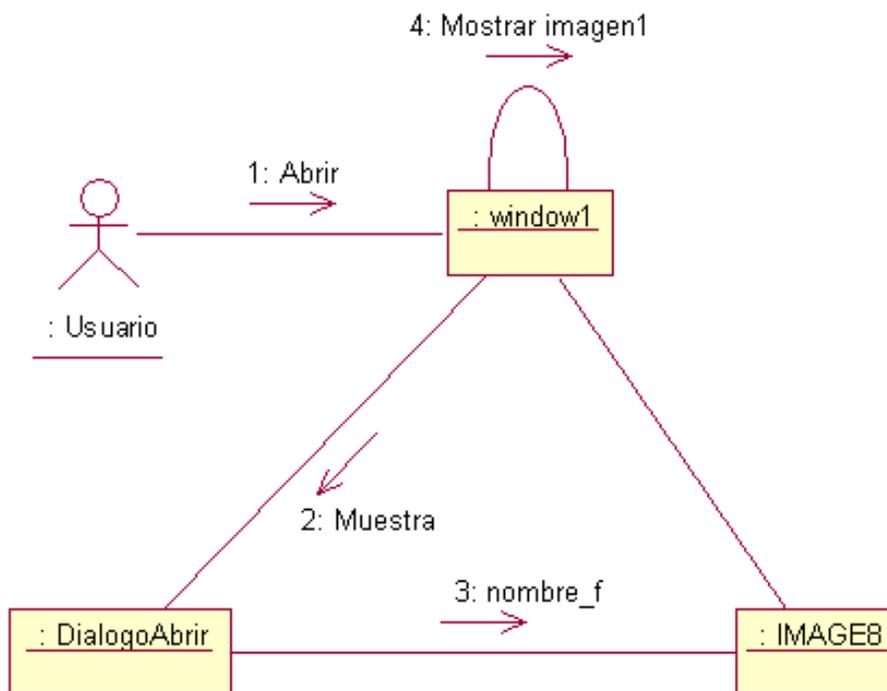
- GUI_Inicial. (Contiene menús, barra de herramientas, áreas de dibujo y barra de progreso)
- Dialogo Seleccione Guardar (Contiene los botones OK y Cancelar y la forma de acceder a los directorios y archivos.)

5.3 DIAGRAMAS DE COLABORACIÓN

5.3.1 Diagrama de colaboración para el caso de uso Abrir imagen

Una vez el usuario de la orden de abrir la aplicación desplegara un dialogo para seleccionar el archivo a cargar, este dialogo retorna el path del archivo para que un objeto de clase IMAGE8 lo abra. Una vez cargada la imagen desde su archivo la interfaz grafica la despliega.

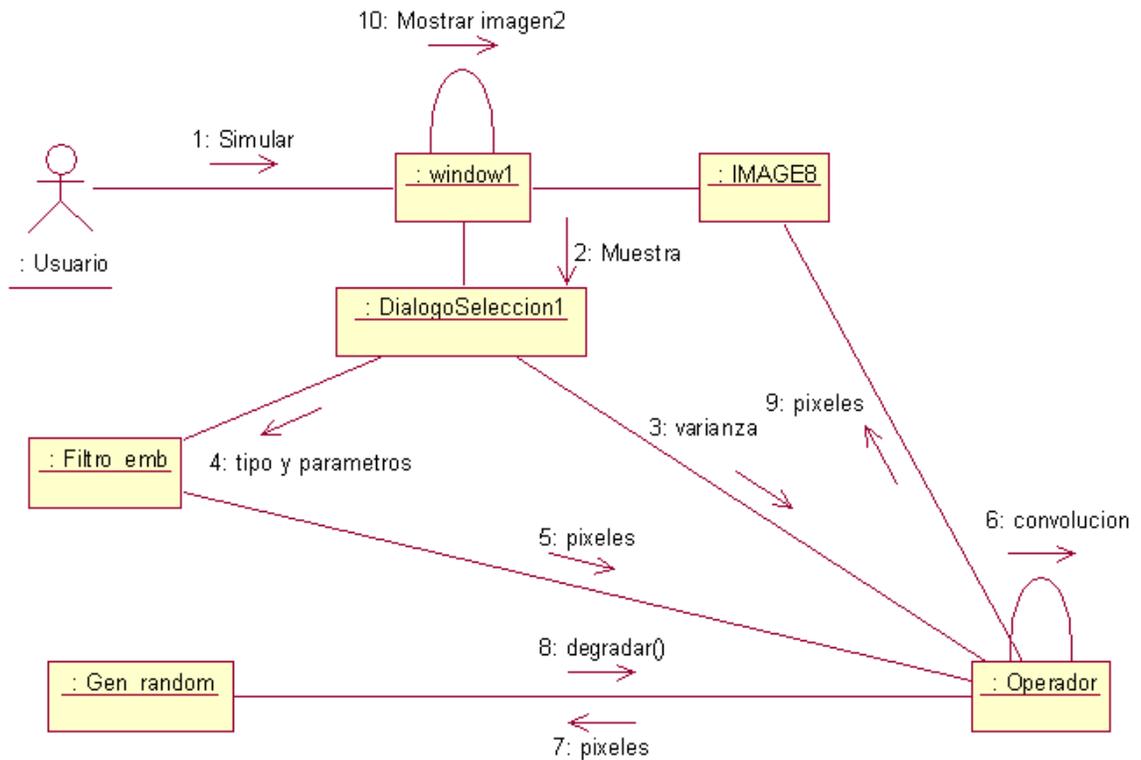
Figura 5.2 Diagrama de colaboración para el caso de uso Abrir imagen



5.3.2 Diagrama de colaboración para el caso de uso simular degradación

Una vez el usuario da la orden de simular la aplicación despliega un dialogo por el cual el usuario introducirá la varianza del ruido el tipo de emborramiento que se desea simular y su parámetro. Con estos parámetros se instanciará una clase de tipo Filtro_emb, que pasará sus datos para que un objeto de clase tipo Operador efectúe la operación de convolución con esta y el objeto tipo Gen_random agregará el ruido a los píxeles de Operador. Terminado este procedimiento se desplegará la imagen obtenida.

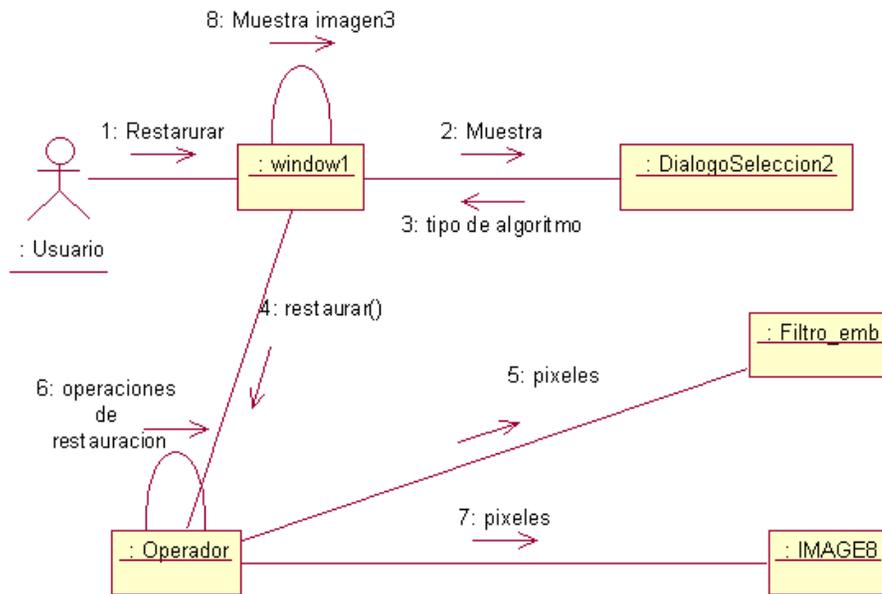
Figura 5.3 Diagrama de colaboración para el caso de uso Simular degradación



5.3.3 Diagrama de colaboración para el caso de uso Restaurar imagen

Una vez el usuario da la orden de simular la aplicación despliega el dialogo para escoger el algoritmo usado para la restauración, este dialogo aparece si la degradación solo se efectuó agregando ruido. El objeto de clase Filtro_emb y el objeto de tipo Operador proporcionan los métodos con los cuales se puede efectuar el tipo de restauración escogida. Una vez la restauración ha sido completada la imagen restaurada se despliega en pantalla.

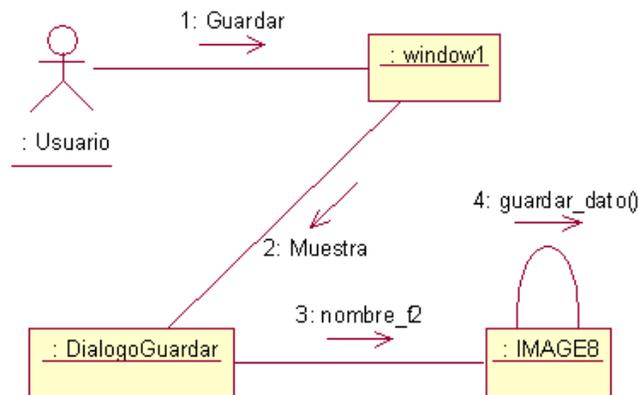
Figura 5.4 Diagrama de colaboración para el caso de uso Restaurar imagen



5.3.4 Diagrama de colaboración para el caso de uso Guardar imagen en disco

Una vez el usuario haya dado la orden de guardar la aplicación desplegará un dialogo para introducir el nombre con el cual se desea guardar los archivos para la imagen degradada y restaurada. Este dialogo entrega el path en el cual se escribirá el archivo a un objeto de clase IMAGE8 el cual invocará el respectivo método que gravará la imagen en disco con formato bmp.

Figura 5.5 Diagrama de colaboración para el caso de uso Guardar imagen en disco



5.4.1 Descripción del diagrama de clases

La interfaz gráfica esta representada por las clases window_glade, window1 y las clase con el nombre compuesto por dialogo. La clase IMAGE, IMAGE8, IMAGER, Operador, Filtro_emb implementan los métodos que permiten simular la degradación y efectuar las técnicas de restauración. A continuación se presenta una descripción más detallada de cada clase.

window1_glade: clase base contruida por la herramienta de diseño utilizada.

window1: clase que contiene como atributos las clases que realizan las operaciones de simulación de degradación y restauración, además implementa los métodos que manejan los eventos de la aplicación.

Atributos:

IMAGE8 *original: puntero a un objeto que maneja los datos de la imagen leída desde archivo.

IMAGE8 *degradada: puntero a un objeto que maneja los datos de la imagen degradada.

IMAGE8 *restaurada: puntero a un objeto que maneja los datos de la imagen restaurada.

Operador *observed: puntero a un objeto que manipula los datos de la imagen original.

Operador *aux: puntero a objeto que ayuda a manipular los datos de la imagen original.

Filtro_emb *blur: puntero a objeto que almacena y manipula la forma de la función de emborramiento.

DialogoNombre: clase para implementar un dialogo que permite obtener el path de un archivo específico.

DialogoAbrir: Hereda de DialogoNombre y obtiene el path de un archivo para abrirlo.

DialogoGuardar: Hereda de DialogoNombre y retorna el path de un archivo para guardarlo en disco.

DialogoSeleccion1: Hereda de DialogoNombre y es la interfaz para que el usuario introduzca los datos de la simulación de degradación de imagen.

DialogoSeleccion2: Hereda de DialogoNombre y es la interfaz para que el usuario seleccione el tipo de técnica para restaurar la imagen.

IMAGE: Clase base para construir clases más especializadas para manipular los datos de una imagen.

Atributos:

int row: el número de filas que tiene el arreglo de píxeles de la imagen.

int colum: el número de columnas que tiene el arreglo de píxeles de la imagen.

IMAGE8: Hereda de IMAGE, implementa los métodos para abrir y guardar archivos de imagen BMP.

Atributos:

unsigned char *pixels: puntero a un arreglo de píxeles en formato de 256 niveles de gris.

Métodos:

void liberar_m(): libera la memoria utilizada para pixels.

void reset(): fija el valor de cada pixel a cero.

void imrtoim8(int a,int b,float *c): iguala pixels a los valores a los que apunta el puntero c; pixels y c apuntan a arreglos del mismo tamaño.

void imctoim8(int a,int b,float *c): iguala pixels a los valores a los que apunta el puntero c en sus índices pares; c es el doble de tamaño de c.

unsigned long int write (char *filename, unsigned long int width, unsigned long int height, unsigned char* image, unsigned long int numPaletteEntries): escribe un archivo BMP con nombre filename, con valores de píxeles iguales a los que apunta image.

void const_paleta(RGBQUAD *paleta): Crea la paleta de colores para un archivo de 256 niveles de gris.

IMAGER: Hereda de IMAGE, implementa los métodos para transformar los valores de pixels a un formato que permita con ellos realizar operaciones matemáticas con los valores de píxeles leídos por ejemplo por un objeto de IMAGE8.

Atributos:

flota *pixels: puntero a un arreglo de valores tipo float para tratamiento matemático.

Métodos:

void liberar_m(): libera la memoria reservada para pixels.

void im8toimr(int a,int b, unsigned char *c): iguala los valores de c a los de pixels, el tamaño de pixels y c es igual.

void im8toimc(IMAGE8 &dato): iguala pixels, en sus índices pares, a c; los índices impares los fija a cero. El tamaño de pixels es el doble del de c.

void writedata(char *nombre): grafa en un archivo binario los datos apuntados por pixels.

void imctoimr(float *c): iguala pixels, en sus índices pares, a c; los índices impares los fija a cero. El tamaño de pixels es el doble del de c.

Operador: Hereda de IMAGER, implementa las transformaciones y las operaciones a realizar sobre pixels.

Métodos:

int suma(Operador &dato2): suma elemento a elemento de pixels de dato2 con los de pixels propio.

int resta(Operador &dato2): resta elemento a elemento de pixels de dato2 con los de pixels propio.
void fourn(int ndim,int isign): calcula la FFT o IFFT a los datos almacenados en pixels.
int convolucion(Operador &dato2): multiplica en forma compleja los elementos de pixels de dato2 con los de pixels propio.
void conj(void): multiplica por menos uno los elementos de pixels de índice impar.
int correlacion(Operador &dato2): multiplica en forma compleja conjugada los elementos de pixels de dato2 con los de pixels propio.
int div(Operador &dato2): divide elemento a elemento los valores de pixels propio por los de pixels de dato2.
void pot2(): calcula la suma de el cuadrado de un valor de índice par con uno de índice impar de pixels.
void magnitud():calcula la suma de el cuadrado de un valor de índice par con uno de índice impar de pixels, y posteriormente su raíz cuadrada.
void fase():calcula la arcontangente de la división de un valor de índice par con uno de índice impar de pixels
void preal(): fija los valores de indice impar a cero;
void suma_esc(float x):suma a cada elemento de pixels es valor de x.

Filtro_emb: Hereda de Operador, implementa los métodos para hacer que pixels tenga los valores de las funciones de emborramiento a simular.

Atributos:

float sigma: parámetro para simular emborramiento por turbulencia atmosférica.
int mov: parámetro para simular emborramiento por movimiento.
int radio: parámetro para simular emborramiento por fura de foco.

Métodos:

void movimiento(int mov1): hace que pixels se encuentren los valores que simulen una función de emborrmiento por movimiento con parámetro mov1.
void f_foco(int radio1): hace que pixels se encuentren los valores que simulen una función de emborrmiento por fuera de foco con parámetro radio1.
void turbulencia(float sigma1): hace que pixels se encuentren los valores que simulen una función de emborrmiento por turbulencia atmosférica con parámetro sigma1..

Gen_random: implementa generadores de números aleatorios para simula el ruido.

Atributos:

float varianza: almacena la varianza del ruido blanco gaussiano.

Métodos:

float ran1(long *idum): generador de números aleatorios con distribución uniforme.
float gaussdev(long *idum): generador de números aleatorios con distribución normal de media cero y varianza 1;
float degradar(float *image, int nc, int nr, float var:n): agrega ruido blanco gaussiano de varianza var_n a los valores contenidos en imagen.

5.5 INTERFAZ GRÁFICA

Figura 5.7 GUI Inicial.

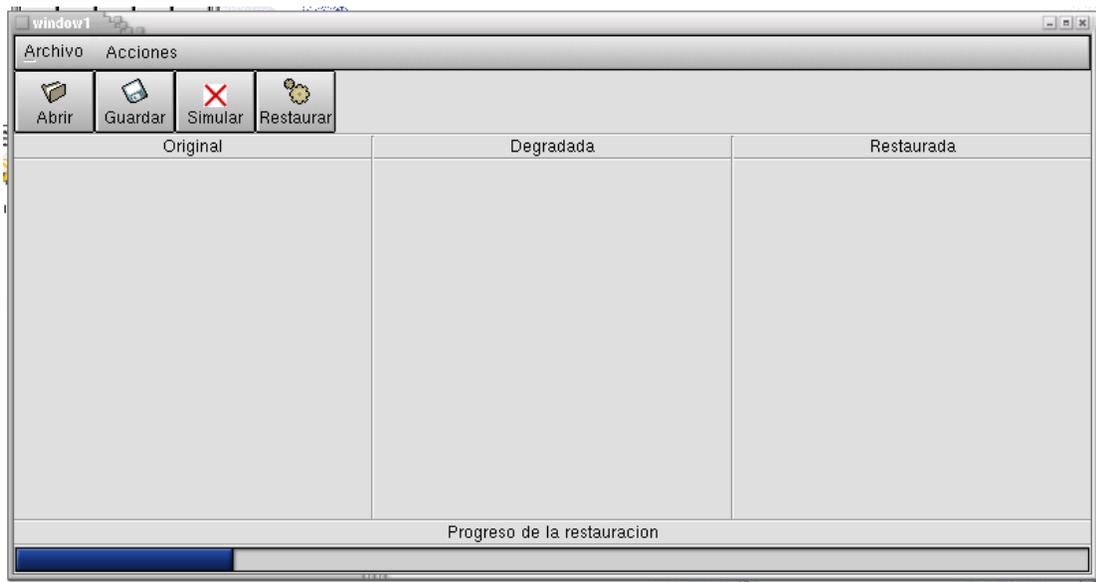


Figura 5.8 Dialogo_Abrir

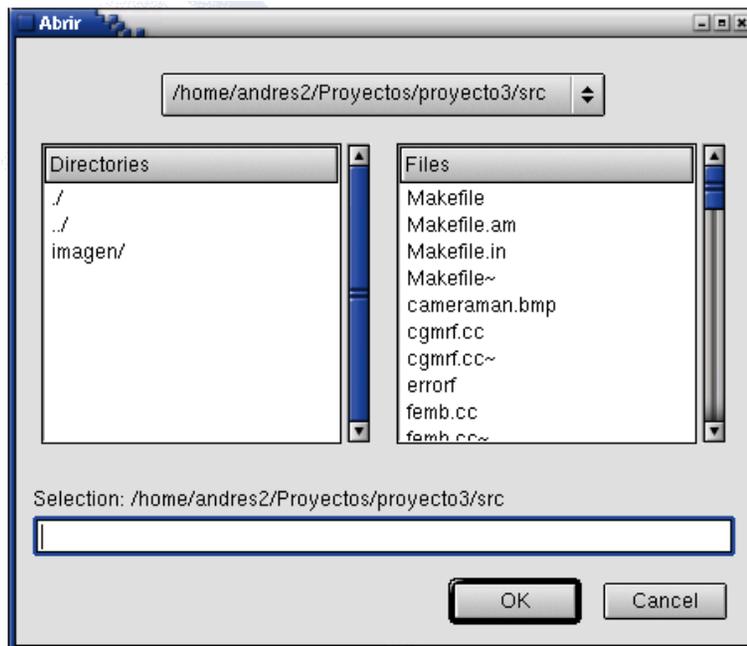


Figura 5.9 Dialogo_Opciones_Degradacion

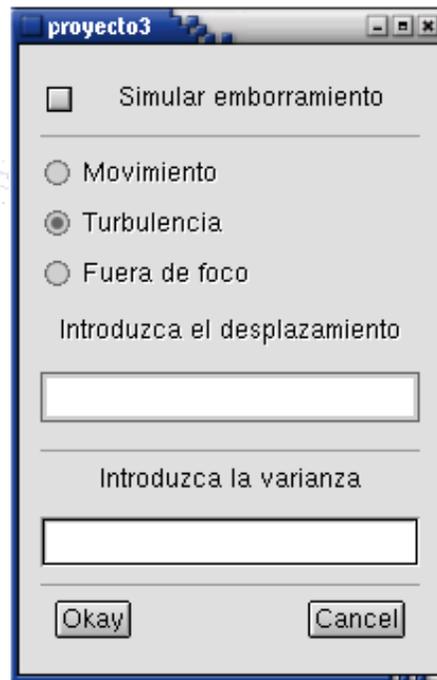


Figura 5.10 Dialogo Seleccione Técnica de Reducción de Ruido

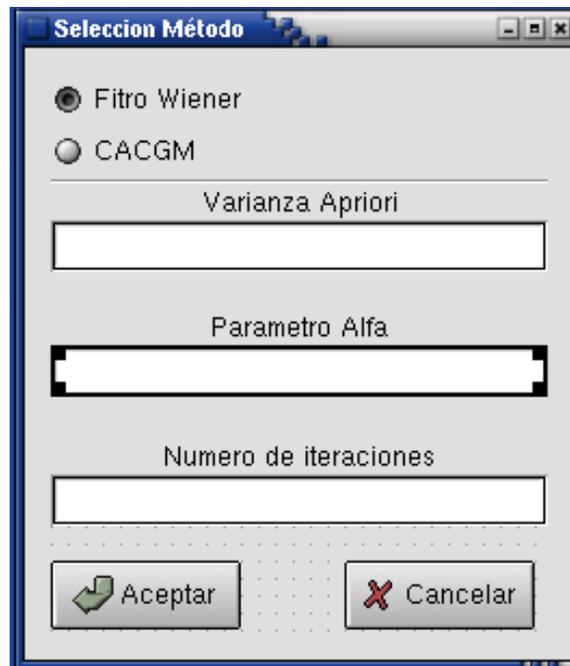


Figura 5.11 Dialogo_Guardar

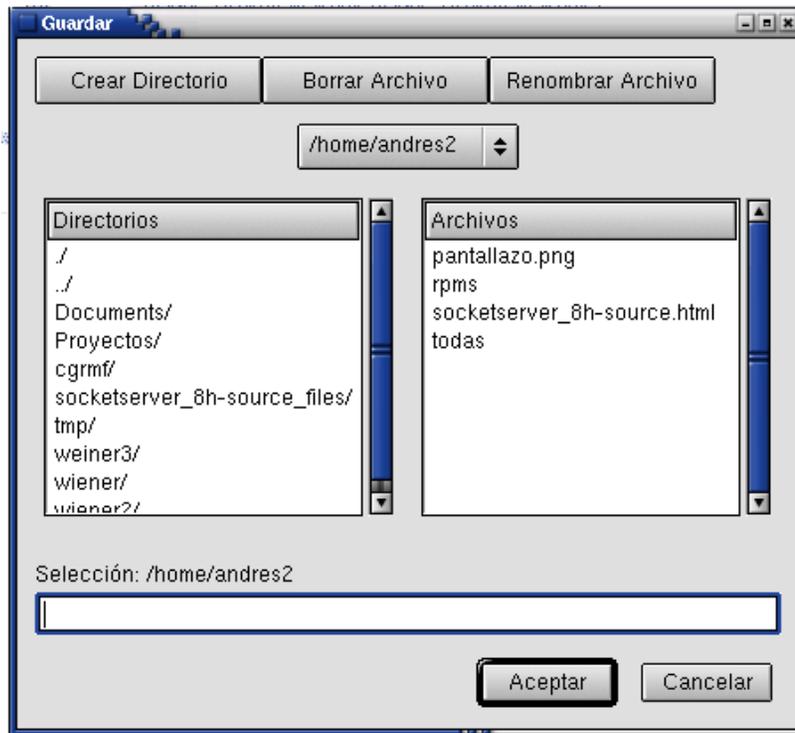


Figura 5.12 GUI con todas las imágenes desplegadas



BIBLIOGRAFIA

- [1] KATSAGGELOS, Aggelos ; BANHAM, Mark. Digital Image Restoration. <http://ivpl.ece.nwu.edu/Publications/Journals/1997/IEEESPMag97a.pdf>

- [2] <http://cxc.harvard.edu/ciao2.2.1/dictionary/psf.html>

- [3] Image Restoration Newsletter / Sumer 1993. http://www.stsci.edu/ftp/documents/image-restoration/IRP_newsletter.ps

- [4] VOGEL, Curt. Regularization Methods An Applied Mathematician's Perspective. Inverse-Vogel.dpf

- [5] KATSAGGELOS, Aggelos. Iterative Image Restoration. http://ivpl.ece.nwu.edu/Publications/Journals/1989/SPIE_OE98a.pdf

- [6] MOLINA, R; NUÑEZ, J; MATEOS, J. Image Restoration in Astronomy. A Bayesian Perspective. <http://decsai.ugr.es/~jmd/proyecto/papers/00spmazine.pdf>

- [7] MOLINA, Rafael ; KATSAGGELOS, Aggelos. On the Hierarchical Bayesian Approach to Image Restoration and the Iterative Evolution of the Regularization Parameter. <http://decsai.ugr.es/~jmd/proyecto/papers/94vcip.pdf>

- [8] KUNDUR, Deepa; HATZINAKOS, Dimitrios. Blind Image Deconvolution. <http://www.comm.toronto.edu/~deepa/pdf/00489268.pdf>

- [9] GRAY, Robert. Toeplitz And Circulant Matrices: A Review. <http://ee.stanford.edu/~gray/toeplitz.pdf>

- [10] MOLINA, R. Introducción al Procesamiento y Análisis de Imágenes Digitales. http://www-etsi2.ugr.es/depar/ccia/mia/complementario/Procesamiento_Imagenes.pdf

- [11] <http://www.cis.rit.edu/class/simg713/notes/randomvariables.pdf>

- [12] Power Spectrum Estimation.
<http://www.cis.rit.edu/class/simg713/Lectures/Lecture713-15-4.pdf>
- [13] CSA Procesos Estocásticos.
http://www.tsc.uvigo.es/BIO/Docencia/CSA/Teoria/csa_teoría06.pdf
- [14] CASTLEMA, Kenneth. Digital Image Processing. Prentice-Hall.
- [15] Constrained Least Squares Restoration.
http://www.eecs.wsu.edu/~cs445/Lecture_13.pdf
- [16] etsi2.ugr.es/depar/ccia/mia/complementario/T1/book-4.pdf book-4.pdf
- [17] <http://www.owl.net.rice.edu/~elec539/Projects99/BACH/proj2/wiener.html>
- [18] WANSCHURA, Thorsten. Statistical Mechanical Models for Image Processing.
<http://docserver.bis.uni-oldenburg.de/publikationen/dissertation/2001/wansta01/pdf/wansta01.pdf>
- [19] LI, S. Markov Random Field Modeling in Computer Vision.
<http://www.vision.ee.ethz.ch/~rpaget/Markov/book.html>
- [20] PUJOL, M; RIZO, R; ARQUES, P; COMPAÑ; P, ESCOLANO, F; MOLINA, R. Aplicación de los modelos de campos aleatorios de Markov en visión artificial.
<http://www.cvc.uab.es/revc/revista/04/0401-tut.pdf>
- [21] MOLINA, Rafael; KATSAGGELOS, Aggelos; MATEOS, Javier; HERMOSO, Aurora; SEGAL, Andrew. Restoration of severely blurred high range image using stochastic and deterministic relaxation algorithms in compound Gauss-Markov random fields.
<http://ivpl.ece.northwestern.edu/Publications/Journals/PatternRecognition00.pdf>
- [22] NEAL, Radford. Probabilistic Inference Using Markov Chain Monte Carlo Methods.
<http://ftp.cs.toronto.edu/pub/radford/review.ps.Z>

- [23] GEMAN, Stuart; GEMAN, Donald. Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images. IEEE transactions on Pattern Analysis and Machine Intelligence. Vol. PAM1-6. No. 6 páginas 721 - 740.
- [24] JENG, Fure-Ching; WOODS, Jhon. Simulated Annealing in Compound Random Fields. IEEE Transaction on Information Theory, Vol. 36 No. 1. páginas 94 – 107.

ABREVIATURAS

AM: Proceso de movimiento medio.

AR: Proceso autorregresivo.

ARMA: Proceso de movimiento medio autorregresivo.

BSNR: Razón señal a ruido emborrada.

CACGM: Campos aleatorios compuestos de Gauss-Markov.

CAM: Campos aleatorios de Markov.

CAR: Proceso condicional autorregresivo.

CCD: Sensor de dispositivos acoplados por carga.

FFT: Transformada rápida de fourier.

ISNR: Mejoramiento en relación señal ruido.

MAP: Maximun a posteriori.

PSF: Función de esparcimiento de punto

OTF: Función de transferencia óptica.

UML: Lenguaje de modelamiento unificado.