



2  
41

UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO

---

FACULTAD DE INGENIERIA

RESTAURACION DE IMAGENES DIGITALES  
SAR UTILIZANDO DIFUSION ANISOTROPICA

T E S I S

QUE PARA OBTENER EL TITULO DE  
INGENIERO EN TELECOMUNICACIONES  
PRESENTA EN

LUIS ARTURO CALDERON VILLEGAS  
ALEJANDRO ECHEVERRIA GARCIDUENAS

TESIS CON  
FALLA DE ORIGEN

CONSTITUYE PARTE DEL CURSO DE TESIS EN LA FACULTAD DE INGENIERIA

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

1997

TESIS CON  
FALLA DE ORIGEN





Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## RECONOCIMIENTOS

Al Dr. Boris Escalante Ramírez, por su gran amistad y apoyo como amigo y como un gran dirigente en la elaboración de este trabajo.

## DEDICATORIAS

### *A mis padres:*

Manuel Echeverría Arellano y Olga E. Garcidueñas Vivas  
Por su eterno amor, sacrificios, comprensión y apoyo

### *A mis hermanos:*

Manuel Echeverría Garcidueñas  
Raul Echeverría Garcidueñas  
Olga Leticia Echeverría Garcidueñas  
Juan José Echeverría Garcidueñas  
Por su amor y apoyo en todos mis esfuerzos

### *A mis compañeros de generación:*

Rodolfo, Víctor, Gonzalo, Francisco, Eduardo, Javier, Osvier, Luis, Gabriel, Hugo, Rita, Julia,  
Kesia, Anela  
Por su gran amistad que me brindaron durante mi formación.

### *A mi compañero y amigo de tesis:*

Luis A. Calderón, gracias por su amistad y apoyo a lo largo del desarrollo de esta tesis.

### *A mis amigos de la vida:*

Gracias por compartir cada momento de tu vida. Llévame y partí todas las experiencias con gran agradecimiento.

### *A todos mis maestros:*

Por su valioso trabajo en la formación, tanto personal y como profesional.

En  
1994

### *Aparentemente esto termina:*

A Dios que me ha guiado en esta gran aventura que es la vida.

Alfonso Echeverría Garcidueñas

## **Agradecimientos**

- A Jehová Dios por permitirme llegar hasta donde estoy ahora.
- A mi familia por su apoyo incondicional.
- A la UHAM, por regalarme esta hermosa cámara.
- A mis compañeros de cámara, con quienes paso momentos inolvidables.
- Al Dr. Bani Escobar por su experiencia, apoyo y amistad.
- A toda la gente que me ofrece profesiones, trabajos, etc., a quien no nombro individualmente por temor a que me falte alguno.

A todos ustedes agradezco por su apoyo de una u otra forma, que me ha permitido llegar a obtener el título de ingeniero en telecomunicaciones.

Luis Arturo Calderón Vilegas

# Índice

	Pag
<b>Introducción</b>	1
<b>Capítulo I</b>	
<b>Elementos básicos para el procesamiento digital de imágenes</b>	<b>2</b>
1.1 Antecedentes del procesamiento digital de imágenes	2
1.2 Elementos de la percepción visual humana	3
1.2.1 Estructura del ojo	3
1.2.2 Formación de una imagen en el ojo	6
1.2.3 El ojo y la cámara fotográfica	7
1.3 Representación de las imágenes digitales	8
1.4 Fundamentos del procesamiento digital de imágenes	9
1.4.1 Adquisición de la imagen	9
1.4.2 Almacenamiento	9
1.4.3 Procesamiento	9
1.4.4 Transmisión	10
1.4.5 Despliegue	10
1.5 Algoritmos para el procesamiento de imágenes	10
1.5.1 Mejoramiento de imágenes digitales	10
1.5.2 Restauración de imágenes digitales	11
1.5.3 Análisis de la imagen	11
1.6 Rendido en imágenes digitales	11
1.6.1 Rendido 2D	12
1.6.2 Rendido 3D	12
<b>Capítulo II</b>	
<b>Técnicas de restauración de imágenes con ruido</b>	<b>13</b>
II.1 Técnicas de filtrado de ruido gaussiano en el dominio espacial	13
II.1.1 Filtro de promedio	14
II.1.2 Filtro estadístico de orden	14
II.1.3 Filtro estadístico de orden adaptativo	16
II.1.4 Filtro de gradiente de momento y promediado	16
II.1.5 Filtro de orden	17
II.1.6 Filtro de orden	17
II.1.7 Filtro gaussiano	18
II.1.8 Filtro estadístico	18
II.2 Filtro de Wiener	20
II.3 Filtro de Laplace de la imagen	21
II.4 Filtro de Laplace de la imagen	24
II.5 Filtro de Laplace de la imagen de la imagen de la imagen	24
II.6 Filtro de Laplace de la imagen de la imagen de la imagen	24
II.7 Filtro de Laplace de la imagen de la imagen de la imagen	24
<b>Capítulo III</b>	
<b>Restauración de imágenes con ruido gaussiano en el dominio frecuencial</b>	<b>25</b>
III.1 Filtro de promedio	25
III.2 Filtro de Laplace de la imagen	28

III.2.1 Propiedades de la Difusion Anisotrópica .....	30
III.2.1.1 El principio máximo .....	30
III.2.1.2 Realce de bordes .....	30
III.3 Discretización de la ecuación de difusión .....	32
III.4 Implementación de la ecuación de difusión .....	34
III.5 Espacio escalado y difusión anisotrópica .....	39
Capítulo IV	
Restauración de imágenes usando Saravizado Adaptativo .....	42
IV.1 Principio del algoritmo .....	42
IV.2 Una nueva representación del espacio escalado .....	44
IV.3 Saravizado y Difusión .....	46
IV.4 Análisis del algoritmo .....	47
IV.5 Implantación paralela del algoritmo de saravizado adaptativo .....	50
Capítulo V	
Resultados experimentales y pruebas .....	55
V.1 Resultados experimentales del algoritmo de difusión anisotrópica .....	55
V.1.1 Resultados variando los parámetros $k$ y el número de iteraciones .....	55
V.1.2 Resultados variando el parámetro $\lambda$ .....	59
V.1.3 Resultados experimentales del algoritmo de saravizado adaptativo .....	61
V.1.4 Usaración de los algoritmos .....	64
V.1.5 Evaluación objetiva .....	65
V.1.6 Evaluación subjetiva .....	66
V.2 Resultados experimentales de restauración .....	66
Capítulo VI	
Conclusiones .....	71
Bibliografía .....	73
Referencias .....	73

## Índice de Figuras y sus Fuentes.

- Fig 1.1 Hecht-Jajac, *Optica*. Addison-Wesley Iberoamericana, Wilmington, Delaware, E. U. A. Pag. 148.
- Fig 1.2 Hecht-Jajac, *Optica*. Addison-Wesley Iberoamericana, Wilmington, Delaware, E. U. A. Pag. 149.
- Fig 1.3 Gonzalez Rafael, *Digital Images Processing*. Pag 25
- Fig 1.4 Gonzalez Rafael, *Digital Images Processing*. Pag 6
- Fig 1.5 Histopatalogia
- Fig 1.6 William K. Pratt, *Digital Image Processing*
- Fig 2.1 y Fig 2.2 J. S. Lee and J. K. Schick, *Speckle filtering of Synthetic Aperture Radar, Remote Sensing Reviews* 1994, April, pag. 339
- Fig 2.3 William K. Pratt, *Digital Image Processing*. Pag. 316
- Fig 3.1 Andrew P. White, *Edge space filtering: a new approach to multistage description*, *Technical Report for Artificial Intelligence Research*, pag. 84
- Fig 3.2 y 3.3 P. Perona and J. Malik, *Scale-space and edge detection using Anisotropic Diffusion*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, N. 7 July, Pag. 632.
- Fig 3.4 Perona and J. Malik, *Scale-space and edge detection using Anisotropic Diffusion*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, N. 7 July, Pag. 632
- Fig 4.1 Philippe Jeant-Marc, *Jan Den Orten Adaptive Smoothing: A General Tool for Early Vision*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, N. 6 June, Pag. 518
- Fig 4.2 Philippe Jeant-Marc, *Jan Den Orten Adaptive Smoothing: A General Tool for Early Vision*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, N. 6 June, Pag. 520

---

## Introducción

El procesamiento digital de imágenes se ha vuelto de gran importancia en los últimos años, tanto por su relevancia para percepción humana, así como su uso en visión computacional. En esta tesis presentamos la Difusión Anisotrópica como una técnica para el procesamiento digital de imágenes. La Difusión Anisotrópica es una técnica alternativa al Espacio Escalado para la representación de imágenes a múltiples resoluciones. En este trabajo presentamos su aplicación a la reducción de ruido en imágenes digitales.

El presente trabajo de tesis se ha dividido en seis partes que serán descritos a continuación, así como una sección de bibliografía.

El primer capítulo de esta tesis presentamos los elementos básicos para el entendimiento del procesamiento de imágenes tales como ¿Cómo percibimos las imágenes?, ¿Qué es ruido en las imágenes?, representación o modelado de una imagen digital, etc.

El segundo capítulo hace una revisión de los distintos métodos que existen para el procesamiento digital de las imágenes, los métodos presentados son recientes o modernos y gran parte de ellos han sido implementados en la División de Estudios de Posgrado de la Facultad de Ingeniería y en el Centro de Computación (CCAI).

El tercer capítulo trata exclusivamente de la Difusión Anisotrópica. Este capítulo contesta preguntas tales como ¿cómo se ve la Difusión Anisotrópica?, ¿cuáles son los bases teóricos de la Difusión Anisotrópica?, ¿cómo se implementa?, etc.

En el cuarto capítulo tratamos los detalles de los algoritmos Anisotrópicos, que es una técnica alternativa para eliminar el ruido de las imágenes.

El quinto capítulo presenta los resultados experimentales, donde comparamos los resultados experimentales del algoritmo de Difusión Anisotrópica con los resultados experimentales de otros algoritmos.

En el sexto capítulo presentamos los resultados visuales de nuestra tesis.

Las implementaciones del algoritmo de Difusión Anisotrópica y los Resultados Aditivos, se hizo en lenguaje C, para esto utilizamos los módulos de imágenes que incluimos con este libro en particularmente imágenes de fondo de Península Simulada (IPAP) que ha sido en imagen digital de 256x256 píxeles. Los algoritmos Anisotrópicos en libro (LAP) de imágenes como: método de procesamiento de imágenes por espacio de transformadas.

## Capítulo I

### Elementos básicos para el procesamiento digital de imágenes

#### I.1 Antecedentes del procesamiento digital de imágenes

El primer antecedente del procesamiento digital de imágenes se remonta al año 1957, cuando se aplicó el primer algoritmo de procesamiento digital de imágenes en un computador digital. Este algoritmo se utilizó para el procesamiento de imágenes de satélite y se basó en el uso de un algoritmo de procesamiento de imágenes de satélite. Este algoritmo se utilizó para el procesamiento de imágenes de satélite y se basó en el uso de un algoritmo de procesamiento de imágenes de satélite. Este algoritmo se utilizó para el procesamiento de imágenes de satélite y se basó en el uso de un algoritmo de procesamiento de imágenes de satélite.

Actualmente, el procesamiento digital de imágenes se utiliza en una gran variedad de aplicaciones, desde el procesamiento de imágenes de satélite hasta el procesamiento de imágenes de video. Este procesamiento se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas.

El procesamiento digital de imágenes se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas. Este procesamiento se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas. Este procesamiento se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas.

El procesamiento digital de imágenes se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas. Este procesamiento se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas. Este procesamiento se realiza mediante el uso de algoritmos de procesamiento digital de imágenes, que permiten mejorar la calidad de las imágenes y extraer información de ellas.



a cabo por medio de cambios en su forma, es decir, tiene una distancia focal variable.

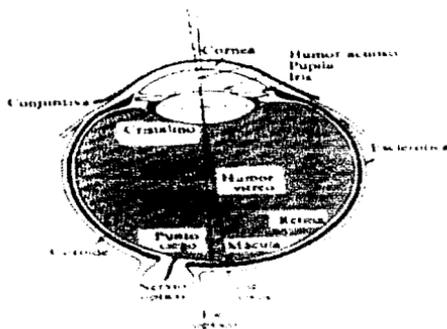


Fig. 1.1 Estructura del ojo humano

El ojo humano es un sistema óptico que permite captar imágenes del mundo exterior y convertirlas en señales eléctricas que el cerebro puede interpretar. Este sistema está formado por una serie de elementos que trabajan en conjunto para lograr este fin.

El primer elemento que entra en contacto con la luz es la cornea, que actúa como una lente convergente que comienza a enfocar la imagen. Después de pasar por la pupila y la iris, la luz atraviesa el cristalino, que es una lente biconvexa que puede cambiar su forma para ajustar el enfoque. El humor vítreo, un líquido gelatinoso que llena el interior del ojo, ayuda a mantener la forma del cristalino y también actúa como un medio de transmisión de la luz. Finalmente, la luz llega a la retina, que es una capa de tejido sensible a la luz que convierte la energía luminosa en señales eléctricas que son enviadas al cerebro a través del nervio óptico. El punto ciego es el lugar donde el nervio óptico sale del ojo y no tiene células fotorreceptoras. Las células cónicas son responsables de la visión de colores y de la visión detallada, mientras que las células bastonales son responsables de la visión en condiciones de poca luz y de la visión periférica. El músculo ciliar y el músculo esfínter ayudan a controlar la forma del cristalino y, por lo tanto, el enfoque de la imagen.

El sistema visual humano es un sistema complejo que involucra no solo la captación de la luz, sino también el procesamiento de la información que se recibe. El cerebro interpreta las señales que le llegan de la retina y las convierte en imágenes que podemos ver. Este proceso es muy rápido y eficiente, lo que nos permite interactuar con nuestro entorno de manera efectiva. Sin embargo, el sistema visual humano también tiene algunas limitaciones, como la capacidad limitada de captar imágenes en condiciones de poca luz o de ver colores en ciertas situaciones. Estas limitaciones son el resultado de la evolución y de las necesidades del ser humano en su entorno natural.

conos. (fig 1.2) Aproximadamente 125 millones de ellas están entremezcladas en forma no uniforme sobre la retina. El arreglo de bastones en algunos casos tiene las características de una película en blanco y negro de grano grueso y alta velocidad. Es extremadamente sensible funcionando en luz demasiado débil como para que respondan los conos, un emborgo es incapaz de distinguir el color y las imágenes que capta no están bien definidas. En contraste, el arreglo de 6 a 7 millones de conos se puede considerar como si fuera una película de color de grano fino y baja velocidad, superpuesta pero separada. Funciona con luz brillante dando vistas detalladas y en color, pero es muy insensible en niveles de luz bajos. El intervalo normal de longitudes de onda de visión humana se dice que es aproximadamente de 280 nm para el rojo a 390 nm para el violeta. La radiación de frecuencias altas es filtrada por la córnea y el cristalino. Esto es importante para evitar en la vida a moléculas por luz ultravioleta. La radiación de frecuencias bajas es transmitida por la óptica del ojo pero esto no afecta la fototransducción. A pesar de esto, los estudios han extendido estas lentes hacia abajo hasta 340 nm donde se absorben y se oxidan rápidamente y hacia arriba hasta aproximadamente 1050 nm donde se absorben por los pigmentos de la retina. Para información ver radiación X) Las células que se absorben por la retina se absorben por la retina de la retina ha mejorado la visión nocturna.



Fig. 1.2. Estructura de la retina del ojo de un conejo.

El ojo humano tiene una resolución de aproximadamente 1 ciclo por grado (CPD) en el campo central. Esto significa que puede distinguir aproximadamente 1 ciclo por grado en el campo central. Esto significa que puede distinguir aproximadamente 1 ciclo por grado en el campo central.

El ojo humano tiene una resolución de aproximadamente 1 ciclo por grado (CPD) en el campo central. Esto significa que puede distinguir aproximadamente 1 ciclo por grado en el campo central. Esto significa que puede distinguir aproximadamente 1 ciclo por grado en el campo central.



### 1.2.3 El ojo y la cámara fotográfica

Con la sola diferencia de su forma estanca, el ojo posee la mayor parte de todas las características estructurales de una cámara fotográfica común y corriente.

Los párpados constituyen el obturador (figura 1.4). El ojo tiene un diafragma que es el iris. Este tiene una ventanilla que no poseen los diafragmas fotográficos; se contrae y se dilata automáticamente según la intensidad de la luz. El ojo tiene una lente (figura 1.4); se halla compuesta de capas de células epiteliales modificadas, y es más elástica que el objetivo de una cámara. Esta elasticidad tiene ventajas, pues la lente del ojo está suspendida de tal manera que la acción muscular puede modificar su forma y, por lo tanto, su longitud focal. En consecuencia, cuando se desea enfocar objetos situados a distancias diversas, el ojo no tiene necesidad de modificarse ni alargarse como ocurre en una cámara fotográfica de objetivas fijas. Las partes laterales y posterior de una cámara, de plástico o de metal, que en el ojo forman su contrapartida en forma de una membrana de tejido conectivo denominada esclerótica (figura 1.4). En el ojo la parte sensible a la luz de los aparatos fotográficos está representada por una membrana de células nerviosas, la retina, que recubre no sólo la parte posterior sino también los lados del ojo (figura 1.4). Engranaje como una cámara fotográfica tiene pintada de negro la superficie interior de los párpados, la luz, entre la retina y la esclerótica, y en unos lugares donde puede haber una muy gran cantidad de pigmento negro.

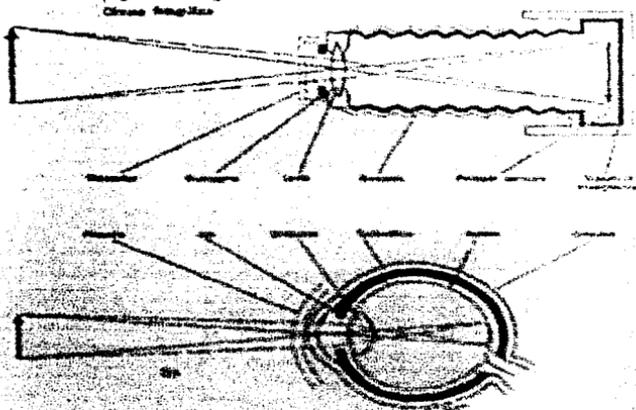


Fig. 1.4 Estructuras del ojo humano y una cámara fotográfica.



de la matriz identifica la luminancia de este punto. Los elementos de tal arreglo se llaman elementos de la imagen o píxeles.

#### 1.4 Fundamentos del procesamiento de imágenes

El procesamiento digital de imágenes incluye tanto hardware, software desarrollados alrededor de un fundamento teórico.

Las unidades que conforman un sistema de procesamiento digital de imágenes generalmente son: adquisición, almacenamiento, procesamiento, transmisión y despliegue.

##### 1.4.1 Adquisición de la imagen

Los elementos que requieren es para la adquisición de una imagen. El primero es un dispositivo físico que emite radiación de una banda de energía espectral electromagnética y que produce una señal eléctrica de acuerdo al nivel de energía recibido. Entre estos dispositivos podemos tener una cámara, scanner, sensores de luz visible e infrarrojo y arreglos fotosensibles de estado sólido. El segundo, llamado digitalizador, es un dispositivo físico que convierte la señal eléctrica de salida del sensor físico en una forma digital.

##### 1.4.2 Almacenamiento

Una vez que se ha terminado la etapa de adquisición es necesario pasar los datos a una estructura de almacenamiento que permita el trabajo en digitalización.

Una imagen en bits de 256 x 256 píxeles requiere más de un millón de bytes para su almacenamiento. El almacenamiento para el procesamiento de imágenes que requiere de más bytes puede utilizarse de diferentes maneras. Una opción es el acceso por direccionamiento directo. Aquí se usa directamente durante el procesamiento que se requiere tener la forma de un arreglo de memoria una o varias imágenes. Otra alternativa requiere de algún mecanismo de acceso a datos redundantes. Si se hace de un modo adecuado puede ser usado para el acceso a datos de un solo bit de una imagen. Como ejemplo de esto se pueden utilizar como una matriz de bits imágenes que como datos digitales. Los resultados de estas operaciones se en bytes (bytes) o como  $2^{24}$  = 16,777,216 bytes (16 MB) o como  $2^{24}$  = 16,777,216 bytes (16 MB).

##### 1.4.3 Procesamiento

El procesamiento de imágenes digitales requiere de procedimientos que permitan el acceso a los datos de imágenes. En un sistema de procesamiento digital de imágenes se usan algoritmos para el procesamiento de imágenes digitales. Estos algoritmos se usan para el procesamiento digital de imágenes y se ejecutan sobre el almacenamiento de imágenes.

Una vez que se ha terminado el sistema de procesamiento de imágenes, se puede pasar a la etapa de despliegue de la imagen. Esto puede ser en forma de una imagen digital o en forma de una imagen física.

#### 1.4.4 Transmisión

La transmisión involucra a las técnicas y sistemas necesarios para enviar la información de las imágenes digitales de un punto a otro. Actualmente se cuenta con protocolos, hardware y software estándar para las comunicaciones, cada uno de los diferentes medios o canales de transmisión presenta problemas diferentes para la transmisión de las imágenes. Entre estos posibles problemas encontramos la relación señal a ruido, la capacidad del canal, atenuación de la señal, dispersión y desvanecimiento, etc. Si a esto agregamos que los dispositivos transmisores y receptores también añaden ruido a las señales de transmisión o recepción, los problemas de transmisión de imágenes digitales se vuelven más complejos.

Una de las limitaciones más grandes con las que se enfrentan las técnicas de transmisión actualmente es la gran cantidad de datos contenida en una imagen.

Esto implica que una imagen transmitida por un canal de comunicación más ocupa un ancho de banda amplio.

Para transmitir señales de video por ejemplo, se requiere una capacidad de 10 Mbytes por segundo. Esto basta el tipo de canales por lo que es factible transmitir video.

Las áreas relacionadas por ejemplo, incluyen capacidad de 10 Mbytes por segundo.

Una alternativa a este problema es utilizar técnicas de compresión de imágenes que sean capaces de representar la misma imagen con menor cantidad de datos.

#### 1.4.5 Despliegue

Los monitores de despliegue más comunes son los que poseen los dispositivos más usados para el despliegue de imágenes digitales. Los monitores son manejados por unidades de salida o drivers de video de la computadora. Otro dispositivo de despliegue incluye dispositivos de impresión, impresoras de matriz de puntos, etc.

### 1.5 Algoritmos para el procesamiento de imágenes digitales

#### 1.5.1 Almacenamiento de las imágenes

El almacenamiento de las imágenes digitales en un sistema digital de despliegue es la primera tarea del procesamiento de imágenes.

El proceso de una imagen puede incluir desde los formatos binarios de punto, punto, coordenadas, almacenamiento de los datos, etc. El almacenamiento de la imagen es el primer paso de la representación de la imagen, el control de la imagen y el despliegue de la imagen.

El almacenamiento de una imagen que es el almacenamiento de la imagen en un sistema de almacenamiento de imágenes. Este sistema de almacenamiento de imágenes es el primer paso de la representación de la imagen, el control de la imagen y el despliegue de la imagen. Este sistema de almacenamiento de imágenes es el primer paso de la representación de la imagen, el control de la imagen y el despliegue de la imagen.



pixeles aislados que no están espacialmente correlacionados. Los pixeles erróneos frecuentemente aparecen en los despliegues y marcan una diferencia con sus pixeles vecinos. Estas observaciones son las bases de muchos algoritmos para limpiar el ruido.

### 1.6.1 Ruido aditivo

El ruido aditivo de una imagen generalmente tiene un espectro de frecuencia espacial más alto que las componentes de la imagen normal. Entonces un simple filtro paso-bajas puede utilizarse para limpiar el ruido aditivo, con la desventaja de difuminar la imagen y por tanto provocar pérdida de nitidez en la misma.

### 1.6.2 Ruido Multiplicativo

El filtrado no lineal, tal como es una técnica útil para el mejoramiento cuando una imagen está sujeta a ruido multiplicativo. En referencia a la figura 1.5 describe el proceso. La imagen de entrada  $F(x,y)$  puede ser modelada como el producto de una imagen libre de ruido  $I(x,y)$  y un arreglo de interferencias de iluminación  $T(x,y)$ . Obteniendo

$$F(x,y) = I(x,y)T(x,y) \quad (1.1)$$

Idealmente  $T(x,y)$  debería ser una constante para toda  $(x,y)$ . Tomando el logaritmo de la ecuación produce el resultado siguiente:

$$\log F(x,y) = \log I(x,y) + \log T(x,y) \quad (1.2)$$



Fig. 1.6. Filtrado no lineal.

Las técnicas convencionales de filtrado de ruido pueden ser usadas directamente por ciertos operadores. Como resultado es simplemente que todos los elementos de imagen después del filtrado multiplicado con el arreglo de interferencias para eliminar el ruido. El resultado es el logaritmo de la imagen original multiplicado por  $T(x,y)$ .

El filtro de interferencias logarítmico y el filtro de iluminación (interferencias de iluminación) pueden ser usados directamente en un arreglo de imagen. El filtro de iluminación es un arreglo de imagen que es el inverso de los elementos de interferencia de iluminación. Cuando se combinan estos dos filtros, el resultado es el arreglo de imagen original y el interferencia de iluminación. Este es el resultado de un filtro de ruido.

## Capítulo II

### Técnicas de restauración de imágenes con ruido

El ruido en las imágenes digitales siempre es un agente indeseable, algunas veces el ruido consiste en solamente unos cuantos píxeles en la imagen que obviamente la alteran perceptualmente pero la imagen permanece reconocible, hasta imágenes muy degradadas prácticamente irreconocibles. Como se dijo en el Capítulo I, el ruido en las imágenes puede ser ruido aditivo, ruido multiplicativo o una mezcla de ambos.

Existe una gran cantidad de métodos para la restauración de imágenes con ruido, que con mayor o menor éxito permiten obtener aproximaciones cercanas de la imagen original. En este capítulo se presentarán solamente algunos de los métodos más representativos, esto quiere decir que hay una gran cantidad de métodos la mayoría de los cuales no se mencionan a lo largo de esta tesis.

Los métodos considerados en la parte II I de este capítulo son filtros no lineales y muchos de ellos fueron creados para la restauración de imágenes con ruido tipo SAR (Imágenes de Radar de Apertura Sintética) mientras que los métodos de las siguientes partes como el Filtro de Wiener, Filtro de Wiener Adaptativo, Filtro en el Dominio de la Transformada Wavelet, Transformada Poissonson, Filtros Anisotrópicos y Suavizado Adaptativo I, los dos últimos considerados en los capítulos III y IV respectivamente, no solo son para imágenes tipo SAR. Respecto que una imagen SAR (Imagen de Radar de Apertura Sintética) se caracteriza por tener ruido multiplicativo, la Transformada Poissonson, Suavizado Adaptativo y por supuesto Filtros Anisotrópicos son algoritmos que fueron implementados con el mismo objetivo en la categoría [1] Filtros de Restoración de la Facilidad de Ingeniería de la USAR, mientras que el Filtro de Wiener y el Filtro de Line Adaptativo fue implementado por el Instituto de Investigación de la USAR. Por tal razón en donde presento resultados experimentales de cada uno de los algoritmos ya que cada algoritmo trata de un tipo de ruido (por lo menos Transformada Poissonson [1], Suavizado Adaptativo y Filtros Anisotrópicos).

#### II.1 Técnicas de filtrado de ruido granular en el dominio espacial.

Aquí se considerará el problema de eliminación de ruido granular de una imagen SAR mediante el método de filtro de Wiener, el filtro de Wiener, los varios métodos en el dominio espacial.

##### Consideraciones generales

Los problemas de restauración de imágenes con ruido granular se resuelven implementando algoritmos de filtrado lineal en el dominio de Fourier de la imagen degradada. Los algoritmos de los dos últimos métodos son basados en el uso de los filtros adaptativos, filtrados por un filtro adaptativo a su vez, permitiendo la eliminación de ruido granular de la imagen tipo SAR mediante el uso de los filtros adaptativos de imágenes tipo SAR. Los métodos de Wiener, Filtro de Wiener, Filtro de Wiener Adaptativo, Filtro en el Dominio de la Transformada Wavelet, Transformada Poissonson, Filtros Anisotrópicos y Suavizado Adaptativo I, los dos últimos considerados en los capítulos III y IV respectivamente, no solo son para imágenes tipo SAR. Respecto que una imagen SAR (Imagen de Radar de Apertura Sintética) se caracteriza por tener ruido multiplicativo, la Transformada Poissonson, Suavizado Adaptativo y por supuesto Filtros Anisotrópicos son algoritmos que fueron implementados con el mismo objetivo en la categoría [1] Filtros de Restoración de la Facilidad de Ingeniería de la USAR, mientras que el Filtro de Wiener y el Filtro de Line Adaptativo fue implementado por el Instituto de Investigación de la USAR. Por tal razón en donde presento resultados experimentales de cada uno de los algoritmos ya que cada algoritmo trata de un tipo de ruido (por lo menos Transformada Poissonson [1], Suavizado Adaptativo y Filtros Anisotrópicos).

adaptarse suavemente al ruido granular, conservando los bordes y las características de los límites afilados. También debe preservar sutilmente los detalles distinguibles.

Las técnicas digitales de filtrado de ruido como el filtro de Mediana y el filtro de Mediana fueron incapaces de limpiar el ruido granular en imágenes SAR. Los algoritmos presentados en esta sección, suponen que las imágenes y el ruido se pueden modelar en términos de ruido multiplicativo.

$$z(k, l) = x(k, l) \cdot v(k, l) \quad (2.1)$$

donde  $z(k, l)$  es la intensidad de píxel en  $(k, l)$  o la amplitud de una imagen SAR,  $x(k, l)$  es la reflectancia y  $v(k, l)$  es el ruido.

### II.1.1 Filtro de Mediana

El filtro de Mediana reemplaza el píxel central por la mediana de todos los píxeles en una ventana de  $m \times n$  píxeles. Esto es efectivo para remover ruido granular. Sin embargo, los bordes borrosos, el borrado final de las características lineales y la distorsión de los objetos y otros problemas comunes de este filtro. El filtro de mediana es una alternativa para sustituir el filtro de Mediana, es que en lugar de sustituir por la mediana sustituye los píxeles por el promedio en una ventana de  $m \times n$  píxeles.

### II.1.2 Filtro Estadístico Local [4-5] [6]

Para eliminar el ruido granular del promedio del filtro de Mediana, una técnica de filtro adaptativa que usa la mediana y la varianza de píxeles en una ventana de  $m \times n$  píxeles. Fue desarrollado por Lee [4] para imágenes de radar multibeam. El filtro adaptativo de una ventana local de  $m \times n$  píxeles,  $m$  y  $n$  son impares y  $m, n \geq 3$ . Sea  $z(k, l)$  la intensidad de píxel para  $(k, l)$  en la ventana de  $m \times n$  píxeles de  $z(k, l)$  en  $(k, l)$ .

El sistema de filtro estadístico promedio de Lee [4] es el filtro de mediana  $z(k, l)$  para eliminar el ruido granular y la varianza de píxeles de la mediana  $z(k, l)$  y la varianza local  $\sigma^2(k, l)$  en una ventana de  $m \times n$  píxeles. El filtro de Lee [4] es un algoritmo de filtro adaptativo de la ecuación (2.1) con  $z(k, l)$  y  $\sigma^2(k, l)$  en la ventana de  $m \times n$  píxeles de  $z(k, l)$  y la varianza de  $z(k, l)$  en la ventana de  $m \times n$  píxeles.

$$z(k, l) = \frac{z(k, l) \cdot \sigma^2(k, l)}{\sigma^2(k, l) + \sigma^2(k, l)}$$

o

$$\frac{z(k, l) \cdot \sigma^2(k, l)}{\sigma^2(k, l) + \sigma^2(k, l)} \quad (2.2)$$

donde  $z(k, l)$  es la intensidad de píxel en  $(k, l)$  en la ventana de  $m \times n$  píxeles de  $z(k, l)$  y la varianza local  $\sigma^2(k, l)$  en una ventana de  $m \times n$  píxeles de  $z(k, l)$  en la ventana de  $m \times n$  píxeles.

El sistema de filtro estadístico de Lee [4] es el filtro de mediana  $z(k, l)$  y la varianza local  $\sigma^2(k, l)$  en una ventana de  $m \times n$  píxeles de  $z(k, l)$  y la varianza de  $z(k, l)$  en la ventana de  $m \times n$  píxeles.

$$y = x + b(z + x) \quad (2.4)$$

donde

$$b = \text{var}(x) / \text{var}(z) \quad (2.5)$$

$x = z$  y  $\text{var}(y)$  son evaluadas de (2.3). El único parámetro de entrada es  $\alpha$ , el cual depende del procesamiento multi-local SAR. Se debe de tener cuidado de que la  $\text{var}(z)$  no sea negativa. Si esta es negativa se fija que  $\text{var}(z) = 0$ ; en otro caso podemos introducir artificialmente que  $\text{var}(z) = 0$ . Este problema podría ser evitado si se establece un límite inferior sobre las regiones homogéneas [9].

En la derivación original de Lee [4] se introduce un modelo linealizado, en el cual el valor óptimo de  $b$  tiene la forma

$$b = \text{var}(x) / (z^2 \alpha^2 + \text{var}(z)) \quad (2.6)$$

para comparar  $b$  de la ecuación (2.5) con  $b$  se sustituye  $\text{var}(z)$  de la ecuación (2.3) dentro de la (2.6). Así se tiene

$$b = \frac{\text{var}(x)}{z^2 \alpha^2 + \text{var}(z)} \quad (2.7)$$

suponiendo que  $(z + x) = 1$ . Esta linealización no afecta la efectividad del suavizamiento que puede producirse para todos los tipos de imágenes de amplitud procesada ( $\alpha = 0.25$ ). Para el caso de imágenes de gradiente SAR, la fórmula correcta de la ecuación (2.3) puede ser usada.

Una interpretación física de la ecuación de este filtro es que en las áreas de homogéneas (N),  $\text{var}(z)$  es el valor de la ecuación (2.4) a  $\alpha = 0.5$  en poder óptimo es  $x = 1$  (el promedio de píxeles en la vecindad). Para las regiones de discontinuidad (D) la border de áreas homogéneas produce  $x = 1$  del valor del píxel para una discontinuidad o punto de borde en cualquier dirección es que el filtro en el borde de los píxeles no está suavizado. Este problema puede ser eliminado usando un filtro adaptativo dirigido hacia los bordes [10].

Usando la generalización física, sustituyendo  $\text{var}(z)$  en la expresión (2.7) se obtiene una forma más simple de la ecuación (2.6). Usando el poder óptimo  $\alpha = 0.5$  en la ecuación (2.6) se obtiene una ecuación de  $b$  para la implementación de  $\text{var}(z)$ . Este procedimiento resulta en la ecuación de suavizado de Lee generalizada, en general, puede ser la forma de donde el filtro es implementado.

\* Este artículo está dedicado a las personas que han contribuido con sus comentarios y sugerencias para mejorar el contenido de este artículo.

Recientemente López[9], modificó el filtro LS para incorporar un límite superior y otro inferior sobre las regiones de homogeneidad. Cuando el radio,  $\text{var}(z)/z$ , es pequeño o igual al límite inferior,  $\sigma_1$ , el píxel central es reemplazado por la media  $\mu$ .

$\mu = z$ . El límite superior sobre el radio se fija en cada experimento, basado sobre un número de looks, resolución espacial, propiedades de sensor, etc. Entre los límites el filtro LS es aplicada, los resultados son similares a los producidos por el filtro LS. Sin embargo, el límite superior no es propiamente fijado, aparece ruido granulado de brillantez indeseable en las imágenes filtradas. Wei y Maitre[12] proponen un filtro de región máxima en el cual la ventana es una función de la media de la región homogeneizada. Este algoritmo tiene una excelente relación entre ruido granulado a lo largo de las regiones homogéneas. Otra variación ha sido proporcionada por Nathan y Curran[13] en el cual el filtro LS es aplicado con la restricción de ruido granulado para imágenes SAR de 14 bits. Arnaud[14] aplica una transformación logarítmica para convertir el modelo de ruido multiplicativo de la ecuación 2.1 al modelo de ruido aditivo. El filtro aditivo de Lee[5] se aplica a las imágenes transformadas. La imagen filtrada es entonces transformada de vuelta al dominio de la imagen por el algoritmo inverso.

### 11.1.3 Filtro estadístico local refinado[Lee 10]

Para tratar con los problemas de ruido en las imágenes de alta resolución producido por el filtro LS, Lee[10] diseñó un algoritmo basado en ventana adaptativa hacia los bordes. Una ventana no cuadrada de  $2n+1$  píxeles se utiliza en las áreas dentro de una ventana de  $2n+1$  píxeles. La media local es calculada usando los píxeles usando únicamente aquellos píxeles en la ventana orientada en la línea de los píxeles. El filtro estadístico local es aplicado. Los resultados son más suaves que el ruido en los bordes y en las áreas de contraste alto son efectivamente mejoradas. Los límites de los bordes de Durand[15] diseñó este algoritmo que fue aplicado a imágenes de radar en el grupo de algoritmos que fueron producidos para el procesamiento de imágenes de radar. La necesidad de procesamiento del ruido granulado de este filtro es menor que los algoritmos de procesamiento no cuadrado de Durand, especialmente en el caso de imágenes que poseen en la ventana de  $2n+1$ . Este filtro puede utilizarse para eliminar el ruido granulado de imágenes de radar grandes (1024 x 1024) obteniéndose imágenes de radar más suaves.

### 11.1.4 Filtro de probabilidad de Maitre & Pechansky

Este filtro adaptativo de ruido granulado es desarrollado en el contexto de una probabilidad estadística de Maitre & Pechansky[16] de un ruido por imagen,  $z(x,y)$ , que es un ruido  $(z(x,y))$  debido a:

$$z(x,y) = \frac{\exp(-\frac{z(x,y)}{\sigma^2})}{\sigma^2} \quad (11.1)$$

El ruido SAR es un ruido que es producido y posee una distribución exponencial. La probabilidad de Maitre & Pechansky es:

$$p(z) = \frac{1}{\sigma^2} \exp\left(-\frac{z}{\sigma^2}\right) \quad (11.2)$$

con la media  $\mu$  y la varianza,  $var(x)$ , las cuales son estimadas usando estadística local dentro de una ventana, (ecuación 2.3) siguiendo un procedimiento similar al filtro LS.

### II.1.5 Filtro Sigma ((Lee 16))

Este filtro está basado en la desviación estándar ( $\sigma$ ) de una distribución Gaussiana. Este filtra la imagen ruidosa solo por el promedio de aquellos píxeles dentro del intervalo dos  $\sigma$  medido desde el píxel central dentro de una ventana de barrido. Es bien conocido que la probabilidad de 2  $\sigma$  de una distribución Gaussiana es 0.955. Los píxeles fuera del lugar del intervalo de 2  $\sigma$  son ignorados. Consecuentemente las características de alto contraste y/o son preservadas. Sin embargo, el ruido granular oscuro con la remoción de la imagen SAP. Esto es para un intervalo promedio de sigma asociado con la información del píxel del modelo de ruido multiplicativo. Como un resultado de acción de no filtro, el ruido para los píxeles. Para el modelo de ruido multiplicativo del intervalo de dos sigma ( $2 \cdot 2\sigma_u$ ,  $2 \cdot 2\sigma_v$ ) estos límites son extendidos de arriba y abajo para todos los niveles de gris (por ejemplo 256 niveles de gris) y promediados en un arreglo. Para un píxel central en una ventana de barrido de intervalo de 2 sigma es recuperado el arreglo y los píxeles en las ventanas son promediados con los límites superiores e inferiores. Los píxeles dentro de los límites son preservados y terminados por promedio de los píxeles vecinos. Los otros píxeles son ignorados.

Aunque con una pérdida de contraste, esto es realizado por el algoritmo de Lee que tiene mucho ruido granular en los bordes. Este filtro debería ser aplicado a imágenes de color con valores descriptores de color. Este algoritmo es presentado en el capítulo 10 de este libro. El filtro Sigma puede ser usado para restaurar un píxel central que ha sido removido (16) el número de niveles de gris dentro de 2 sigma es mayor que el número de niveles de gris de este algoritmo fue reportado por Hubner y McWhorter (17) para la restauración de texturas.

### II.1.6 Filtro Frost (Frost) (18)

Frost (18) usó un filtro de ordenamiento estadístico de los píxeles dentro de una ventana de barrido de imagen que LS. La idea que es Frost es, cuando se aplica en las estadísticas de ruido multiplicativo, LS y Frost se aplican a los píxeles dentro de los bordes de las imágenes para el de preservar el ruido granular de los píxeles.

La ecuación de restauración de Frost granular ruido para este filtro tiene la forma

$$\hat{r}(x,y) = \frac{1}{N} \sum_{i=1}^N r_i(x,y) \quad (2.59)$$

donde  $\hat{r}(x,y)$  es la función de restauración de imagen,  $r_i(x,y)$  es el ordenamiento estadístico de los píxeles de una ventana de barrido de imagen del denominador y  $N$  es el número de píxeles que restaurados en el píxel central de  $r(x,y)$  con el ordenamiento estadístico.

La ecuación de restauración estadística de Frost,  $\hat{r}(x,y)$  es obtenida cuando el ruido granular de los píxeles de los bordes de las imágenes. El filtro de ordenamiento estadístico de Frost es escrito como siguiente

$$\hat{r}(x,y) = \frac{1}{N} \sum_{i=1}^N r_i(x,y) \quad (2.60)$$





Safa y Flouzat idean otro filtro morfológico llamado el "filtro comparativo". Este filtro opera en el plano de una imagen de dos dimensiones para comparar los niveles de gris del pixel central con los otros pixeles dentro de un template (template) seleccionada. Safa y Flouzat muestran que este algoritmo también corresponde a operaciones de apertura y cerradura. Las imágenes contienen regiones homogéneas muy grandes que pueden ser filtradas por un filtro comparativo completamente. El tiempo de uso máquina es excesivo para los dos filtros morfológicos.

## 11.2 Filtro de Wiener

El mejoramiento de la calidad de la restauración es posible con el filtro de Wiener, el cual incorpora un procedimiento a priori del campo de ruido.

Supongamos que la imagen ideal  $f(x,y)$  y la imagen observada  $F(x,y)$  de la figura 2.3 están en dos dimensiones con campos continuos espaciales y media espacial cero. La respuesta al ruido del filtro de restauración se elige para minimizar la media cuadrada del error de restauración.



Fig. 2.3 Método de Restauración de una Imagen.

$$F(x,y) + N(x,y) = \hat{F}(x,y) \quad (2.17)$$

Si el error cuadrado medio de restauración es una función de la siguiente función de ortogonalidad es esencialmente

$$E\{[\hat{F}(x,y) - F(x,y)]^2} = E\{N^2(x,y)} \quad (2.18)$$

para todos los valores de coordenadas de la imagen  $f(x,y)$  de  $x$  y  $y$ . Recuerde que la respuesta de un sistema lineal invariante en el tiempo ante una perturbación es

$$\hat{F}(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) H(u,v) e^{-j2\pi(ux+vy)} du dv \quad (2.19)$$

Por lo que el error cuadrado medio es un término de suma

$$E\{[\hat{F}(x,y) - F(x,y)]^2} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u,v) H(u,v) - F(u,v)|^2 e^{-j2\pi(ux+vy)} du dv \quad (2.20)$$

Apresuntando que la imagen ideal  $F(x,y)$  y el ruido  $N(x,y)$  son completamente independientes, el error cuadrado medio de restauración es esencialmente

$$E\{[\hat{F}(x,y) - F(x,y)]^2} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u,v) H(u,v) - F(u,v)|^2 e^{-j2\pi(ux+vy)} du dv \quad (2.21)$$

Entonces tomando la transformada de Fourier en dos dimensiones en ambos lados de la ecuación (2.21) y resolviendo para  $H_R(\omega_x, \omega_y)$ , la siguiente expresión general muestra la función de transferencia del filtro Wiener.

$$H_R(\omega_x, \omega_y) = W_{FIR}(\omega_x, \omega_y) W_{FIR}^{-1}(\omega_x, \omega_y) \quad (2.22)$$

En el caso especial del modelo de ruido aditivo (fig 2.3)

$$W_{FIR}(\omega_x, \omega_y) = H_{ID}^*(\omega_x, \omega_y) W_{FI}(\omega_x, \omega_y) \quad (2.23)$$

$$W_{FIR}(\omega_x, \omega_y) = |H_{ID}^*(\omega_x, \omega_y)|^2 W_{FI}(\omega_x, \omega_y) + W_N(\omega_x, \omega_y) \quad (2.24)$$

Esto permite modelar al filtro de Wiener para ruido aditivo como:

$$H_R(\omega_x, \omega_y) = \frac{H_{ID}^*(\omega_x, \omega_y) W_{FI}(\omega_x, \omega_y)}{|H_{ID}(\omega_x, \omega_y)|^2 W_{FI}(\omega_x, \omega_y) + W_N(\omega_x, \omega_y)} \quad (2.25)$$

o

$$H_R(\omega_x, \omega_y) = \frac{H_{ID}^*(\omega_x, \omega_y)}{|H_{ID}(\omega_x, \omega_y)|^2 + W_N(\omega_x, \omega_y) / W_{FI}(\omega_x, \omega_y)} \quad (2.26)$$

En la siguiente formula la función de transferencia del filtro restaurador puede ser expresada en términos de la potencia de la relación señal a ruido.

$$SNR(\omega_x, \omega_y) = \frac{W_{S1}(\omega_x, \omega_y)}{W_N(\omega_x, \omega_y)} \quad (2.27)$$

a cada frecuencia espacial. La figura 2.4 contiene un bosquejo de una idea típica del espectro de la imagen, espectro de ruido, función de transferencia inicial (o borrosa), y el resultado de la función de transferencia del Filtro de Wiener. Este filtro funciona como un filtro inverso a bajas frecuencias espaciales y como suavizador a la frecuencia de corte y filtro paso-bajas a altas frecuencias espaciales. La ecuación (2.26) es válida cuando la imagen ideal y el proceso estocástico de la imagen es de media cero.

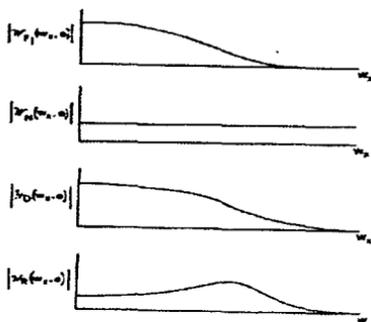


Fig. 2.4

En este caso la reconstrucción de la transformada de Fourier de la imagen es:

$$\hat{F}_f(\omega_x, \omega_y) = H_{Rf}(\omega_x, \omega_y) F_0(\omega_x, \omega_y) \quad (2.28)$$

Si la imagen ideal y la imagen observada son de media distinta de cero, la forma propia de la transformada de la imagen reconstruida es:

$$\hat{F}_f(\omega_x, \omega_y) = H_{Rf}(\omega_x, \omega_y) [F_0(\omega_x, \omega_y) + M_0(\omega_x, \omega_y)] \cdot M_f(\omega_x, \omega_y) \quad (2.29)$$

donde

$$M_0(\omega_x, \omega_y) = H_f(\omega_x, \omega_y) M_I(\omega_x, \omega_y) + M_N(\omega_x, \omega_y) \quad (2.30)$$

donde  $M_I(\omega_x, \omega_y)$  y  $M_N(\omega_x, \omega_y)$  son la Transformada de Fourier en dos dimensiones de la media de la imagen ideal y el ruido, respectivamente.

Se debe notar que la ecuación 2.29 se acomoda al modelo de la variación espacial de la media. En la práctica esto es común al estimar la media de la imagen observada por este promedio espacial  $M_0(x, y)$  y aplicando el filtro de Wiener de la ecuación 2.26 a la diferencia de la imagen observada  $F_0(x, y) - M_0(x, y)$ , agregando el antecedente de la media de la imagen ideal de  $M_I(x, y)$  al resultado del filtro de Wiener.

Esto es útil al investigar los casos especiales de la ecuación 2.26. Si se asume que la imagen ideal es no correlacionada con la unidad de energía entonces  $W_f(\omega_x, \omega_y) = 1$  y el filtro de Wiener llega a ser:

$$H_R(\omega_x, \omega_y) = \frac{H_D^*(\omega_x, \omega_y) W_{r,1}(\omega_x, \omega_y)}{|H_D(\omega_x, \omega_y)|^2 + W_N(\omega_x, \omega_y)} \quad (2.31)$$

Este versión del filtro de Wiener provee menor suavizado de ruido que la ecuación 2.26. Si la imagen original no es borrosa, entonces  $H_D(\omega_x, \omega_y) \approx 1$ , y el filtro de Wiener llega a ser un filtro de suavizado de ruido con función de transferencia

$$H_R(\omega_x, \omega_y) = (1 + W_N(\omega_x, \omega_y))^{-1} \quad (2.32)$$

En muchos sistemas de imágenes la respuesta impulso a el empañamiento no es invariante con el espacio; más bien, estos cambios ocurren en una manera aleatoria. Un ejemplo práctico es el empañamiento causada en una imagen transmitida a través de la turbulenta atmósfera. Obviamente, un filtro de Wiener aplicado a este problema funcionaría mejor si lo pudiéramos adaptar dinámicamente a los cambios del empañamiento de la respuesta al impulso, por lo que tal filtro sería un filtro de Wiener Adaptativo.

### II.3 Filtro adaptativo de Wiener

Una versión adaptativo del filtro de Wiener se obtiene considerando que la imagen original se puede modelar como un proceso estocástico cuya media y varianza son no-estacionarios, esto induce automáticamente al desarrollo de un Filtro Adaptativo de Wiener. Para ser específicos, sea  $R(i, j)$  un ruido aleatorio blanco Gaussiano con media cero y  $\sigma_r^2$  conocida o estimable, entonces su función de densidad de potencia espectral es  $P_R(\omega_x, \omega_y) = \sigma_r^2$ , además es posible modelar la imagen original  $I_0(i, j)$  en el ámbito local como un proceso estacionario, esto es  $I_0(i, j) = m_{i0}(i, j) + \sigma_{i0}(i, j)R(i, j)$ , donde  $m_{i0}(i, j)$  y  $\sigma_{i0}(i, j)$  representan la media y desviación estándar locales de la imagen original, localmente es válido que  $P_{i0}(\omega_x, \omega_y) = \sigma_{i0}^2$  y sustituyendo estos valores en la ecuación:

$$S_R(\omega_x, \omega_y) = \frac{P_{i0}(\omega_x, \omega_y)}{P_{i0}(\omega_x, \omega_y) + P_R(\omega_x, \omega_y)} \quad (2.33)$$

donde  $P_{i0}(\omega_x, \omega_y)$  y  $P_R(\omega_x, \omega_y)$  son las densidades espectrales de potencia de la imagen original y del ruido y se suponen conocidas, con lo cual es posible obtener un estimador óptimo de la imagen original  $I_0(i, j)$  usando el criterio del Mínimo Error Cuadrático Medio Lineal. Esto resulta en el siguiente sistema de restauración adaptativo de Wiener  $S_{AR}(\omega_x, \omega_y)$  cuyo modelo matemático es:

$$S_{AR}(\omega_x, \omega_y) = \frac{\sigma_{i0}^2}{\sigma_{i0}^2 + \sigma_r^2} \quad (2.34)$$

La principal característica del filtro descrito es que su comportamiento se adapta a las particularidades locales de la imagen, esto significa que trabaja como un filtro paso-bajas en las regiones homogéneas de ésta y como filtro paso-altas en sus

contornos, si bien la conmutación entre ambos modos de operación es paulatina según se deduce de la ecuación 2.18. La cualidad anterior está estrechamente relacionada con el enmascaramiento de contraste propio del sistema visual humano, el cual garantiza que las variaciones espurias provocadas por el ruido son menos perceptibles en las regiones de alto contraste (contornos) de una imagen y que en tales regiones sea válido dejar intacta la información degradada sin afectar en gran medida las preferencias del observador humano. Otra propiedad sobresaliente es que el desarrollo e implantación del algoritmo es en el dominio espacial con los ventajas consiguientes. Generalizaciones posteriores del esquema adaptativo de Wiener se reportan en [22, 23, 24] y algunos aspectos sobre la implantación se sugieren en [25].

#### II.4 Filtros en el dominio de la transformada Wavelet

Otra técnica novedosa para el modelado de imágenes es la transformada "Wavelet"<sup>3</sup>, la cual básicamente consiste en la descomposición de la imagen en un conjunto de coeficientes (imagen de coeficientes) que determinan las contribuciones locales y parciales de un grupo de funciones de soporte compacto\* a la descripción regional y multi-escala de la imagen sin transformar (imagen original). El método se asemeja a la transformada de Fourier cuyos coeficientes proporcionan una descripción global de la señal, sin embargo, en el caso de Wavelet la descripción es y está implícita en los coeficientes de la transformada Wavelet.

En el caso general, las funciones de soporte compacto pueden tomar cualquier forma y solo deben cubrir las restricciones impuestas por la teoría respectiva; sin embargo, el conocimiento actual sobre el sistema de visión humana indica su preferencia por los contornos de las imágenes y la gran cantidad de detectores de gradientes desarrollados en el campo de la visión por computadora justifican la utilización de funciones de soporte compacto cuya estructura se asemeje a tales detectores. Un detector óptimo de discontinuidades se relaciona con la primera derivada de la función gaussiana y una variante de la transformada Wavelet que lo utiliza para construir sus funciones de soporte compacto se describe en [26, 27, 28]. Aunque se conocen diversos esquemas de restauración de imágenes ruidosas en el dominio de la transformada Wavelet [26, 28, 29] a continuación describiremos el citado en [28] dada su relativa simplicidad matemático-conceptual y las prestaciones que reporta.

Para señales unidimensionales, las Wavelets son familias de funciones  $\Psi_{s,t}(x)$  generadas a partir de dilataciones y traslaciones de una Wavelet base  $\Psi(x)$  según la expresión:

$$\Psi_{s,t}(x) = \frac{1}{\sqrt{|s|}} \Psi\left(\frac{x-t}{s}\right), \quad (2.35)$$

en esta expresión  $s$  representa el parámetro de dilatación (escala o resolución) y  $t$  el parámetro de traslación. Las Wavelets deben tener media cero y un soporte compacto finito, pueden ser de tipo ortogonal o no-ortogonal, las primeras son útiles en

<sup>3</sup>Algunos autores la traducen como transformada por ondecitas, otros como transformada por ondículas  
\*El dominio de definición de estas funciones es finito en los ámbitos: espacial del plano de la imagen y en plano de la frecuencia espacial o dominio de Fourier

codificación y las segundas lo son en las otras etapas de un sistema de procesamiento digital de imágenes.

Sean las Wavelets  $\Psi_{m,n}(x)$  tales que abarcan el espacio de Hilbert  $L^2(\mathbb{R})$  cuando  $s=2^m$  (espacio de escalas discreta) y  $t=n$ , sustituyendo en la ecuación 2.35 se tiene:

$$\Psi_{m,n}(x) = 2^{-\frac{m}{2}} \psi(2^{-m}(x-n)), \quad (2.36)$$

aquí  $m$  es el índice de escala con  $m \in \{0, 1, 2, \dots\}$  y  $n \in \{\dots, -2, -1, 0, 1, 2, \dots\}$  se conoce como índice de traslación espacial. La transformada Wavelet en el espacio de escalas discreta de la señal unidimensional  $f(x)$  se define como su proyección en el conjunto formado por  $\Psi_{m,n}(x)$ , o sea:

$$W(m, n) = \int_{-\infty}^{\infty} \Psi_{m,n}^*(x) f(x) dx \quad (2.37)$$

y como el conjunto  $\Psi_{m,n}(x)$  abarca el espacio que contiene a la función  $f(x)$ , la reconstrucción de ésta a partir de su transformada Wavelet está sujeta a la ecuación:

$$\hat{f}(x) = \sum_m \sum_n \Psi_{m,n}^*(x) W(m, n) \quad (2.38)$$

donde  $\Psi_{m,n}^*(x)$  es la base dual normalizada de  $\Psi_{m,n}(x)$  y puede emplearse  $\Psi_{m,n}^*(x) \approx \Psi_{m,n}(x)$ .

Existen limitaciones prácticas para la implantación de las ecuaciones 2.37 y 2.38 cuando se trabaja con señales discretas, aquellas se relacionan con las resoluciones máxima y mínima que puede tener la descripción multi-escala [28, 27]; consecuentemente la transformada Wavelet discreta, no-ortogonal y dyadic<sup>3</sup> de la señal  $f(x)$  se define para el conjunto de coeficientes "Wavelet"  $\{W(m, n)\}$  que cumplen con  $m \in \{0, 1, 2, \dots\}$  y  $n \in \{\dots, -2, -1, 0, 1, 2, \dots\}$  y  $M = \log_2 N$ , el parámetro  $N$  representa la dimensión de la señal original  $f(x)$  y la información residual en la escala mayor  $M$  es igual al promedio:

$$S(M, N) = 2^{-M} \int_{-\infty}^{\infty} \psi(2^{-M}(x-n)) f(x) dx \quad (2.39)$$

en la ecuación previa  $\Phi(x)$  es la función de suavizamiento o escalamiento necesaria para discretizar la señal  $f(x)$  sin problemas de aliasing y con la consiguiente pérdida de resolución inherente a tal operación.

En la representación multi-resolución anterior existen  $M \times N$  coeficientes Wavelet (si  $N=256$  y  $M = \log_2(256) = 8 \Rightarrow M \times N = 2024$ ) y se clasifica como una descripción redundante o no-ortogonal, pues su dimensión ( $M \times N$ ) es mayor respecto a la dimensión de la señal original ( $N$ ). Las transformadas "Wavelets" ortogonales se caracterizan por

<sup>3</sup>El término dyadic se introdujo en [27] y se refiere a la representación de la señal a escalas separadas por 2<sup>m</sup>.

optimizar la cantidad de coeficientes de la representación en el espacio de escalas para efectos de codificación; pero dificultan la correlación en dicho espacio con las descripciones involucradas. Esta peculiaridad no existe en las descripciones redundantes. Las reflexiones previas son válidas en el caso general y en consecuencia también lo son cuando la Wavelet base se asemeje a la primera derivada de una función gaussiana

Por otro lado, se sabe que la potencia del ruido de una señal ruidosa se concentra en las resoluciones finas (altas) de su representación multi-escala y como la redundancia de esta descripción permite establecer una correlación multi-escala, entonces es posible implantar algoritmos competitivos de restauración de señales e imágenes ruidosas [26, 29, 30]. Dicha correlación multi-escala, en el dominio de la transformada Wavelet, ayuda a la detección de las singularidades "reales" de la señal original, despreciando aquellas estructuras espurias introducidas por la degradación puntual en la imagen observada. Un método para determinar la correlación espacial directa  $Corr(m, n)$  de la transformada Wavelet en las  $l$  resoluciones adyacentes usa:

$$Corr(m, n) = \prod_{l=0}^{m-1} W(m+l, n), \text{ para todo } n = 1, 2, 3, \dots, N \quad (2.40)$$

donde  $m = M-1$  y  $M$  es la cantidad total de escalas en la descripción. La correlación espacial directa definida por ecuación 2.40 resalta las singularidades abruptas de una señal, suprime el ruido, atenúa las posibles singularidades suaves (entiéndase contornos de contraste medio y bajo) de la señal restaurada! y mejora la exactitud en la ubicación de las discontinuidades relevantes de la señal original [26].

## II.5 Transformada Polinomial Hermiliana[1]

La transformada polinomial es una técnica que permite aproximar una imagen mediante patrones o componentes locales significativos, de diferente tamaño en su modalidad multiresolución. Los patrones utilizados en la descomposición local son funciones polinomiales de orden variable y el resultado de la misma es un conjunto de imágenes de coeficientes que determinan el peso o contribución de cada componente en la representación regional de la imagen. Es posible reconstruir la imagen original utilizando los coeficientes de la expansión y los filtros de síntesis apropiados; aunque, es válido efectuar cualquier procesamiento intermedio sobre aquellos con algún propósito particular. Una revisión más detallada de la Transformada Polinomial se presenta en [1].

## Capítulo III

### Restauración de imágenes con ruido usando Difusión Anisotrópica

La difusión Anisotrópica es un caso alternativo al espacio escalado. El espacio escalado está relacionado con el Sistema Visual Humano. El espacio de escalas representa una imagen por medio de múltiples versiones de la misma a diferentes resoluciones. Es una forma similar a la descomposición de la imagen que se lleva a cabo en las primeras etapas de sistema de visión humana por medio de un esquema multicanal que está formado por campos receptivos de distintos tamaños.

La palabra anisotrópica se aplica tal como se entiende en el comportamiento general de los fenómenos físicos: difusión de gases, conducción térmica y radiación de energía. Esto se refiere a que estos se manifiestan de manera heterogénea en las diferentes coordenadas del espacio.

#### III.1 Espacio escalado

El espacio escalado consiste en obtener una familia de señales derivadas obtenidas por la convolución de la señal original  $I_0(x)$  con un Kernel Gaussiano  $G(x, t)$  de varianza  $t$ :

$$I(x, t) = I_0(x) * G(x, t) = \int_{-\infty}^{\infty} I(u) \frac{1}{\sqrt{2\pi t}} e^{-\frac{(x-u)^2}{2t}} du \quad (3.1)$$

La convolución Gaussiana es atractiva por las siguientes características:

- La Gaussiana es diferenciable e integrable.
- La Gaussiana es simétrica alrededor de su eje vertical donde se tiene la media  $\mu$ .
- La moda, que es el punto sobre el eje horizontal donde la curva tiene su máximo ocurre, en  $x=\mu$ .
- La curva Gaussiana se acerca al eje horizontal en forma asintótica en cualquiera de las dos direcciones, alejándose de la media.
- Conforme  $t$  disminuyen pueden aparecer cruces por cero en la convolución, pero no podrán aparecer conforme  $t$  aumenta.

$I$  define la superficie sobre el plano  $(x, t)$ , la misma que se obtiene por la suave variación de la desviación estándar  $t$ . Llamaremos al plano  $(x, t)$  espacio escalado y la superficie  $I$  define el espacio escalado de la señal.

Para valores grandes de  $t$ , el parámetro del espacio escalado corresponde (en el caso de imágenes) a imágenes con resolución borrosas (o escala borrosa). ver fig 3.1, mientras que valores pequeños de  $t$  la resolución es fina.

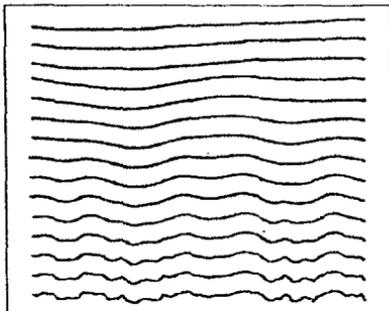


Figura 3.1 Una familia de señales de 1-D obtenida al convolucionar la imagen original con un Kernel Gaussiano cuya desviación estándar  $t$  crece de abajo hacia arriba

La familia de señales derivadas pueden ser equivalentes, a la solución de la ecuación de conducción de calor o difusión. Extendiendo la ecuación (3.1) a dos dimensiones  $(x, y, t)$  para imágenes, tenemos la siguiente ecuación de difusión:

$$I_t(x, y, t) = \Delta I = (I_{xx} + I_{yy}) \quad (3.2)$$

Si  $t=0$  tenemos que  $I(x, y, 0) = I_0(x, y)$  donde  $I_0$  es la imagen original. La formulación de la ecuación de difusión (3.2), debe de cumplir dos criterios.

1) **Causalidad**: No se deberían de generarse detalles espurios cuando la escala pasa de fina a gruesa o grande.

2) **Homogeneidad e isotropía**. La difuminación debería ser uniforme en toda la imagen. Esta propiedad permite la formulación natural de la ecuación. La Difusión Anisotrópica presentada en este documento, muestra que esta propiedad no es estrictamente necesaria y presenta un criterio más útil para el suavizado de imágenes.

### III.2 Difusión Anisotrópica

Perona y Malik [33] definen la difusión Anisotrópica como un método de supresión de detalles finos.

La difusión Anisotrópica consiste en una simple forma de modificar el espacio escalado lineal. La ecuación de difusión es vista como espacio escalado; el coeficiente de difusión  $c$  es una constante independiente de la localización espacial (ver ecuación 3.3). No hay una razón fundamental para que  $c$  sea constante. Si escogemos

adecuadamente a  $c(x,y,t)$  la ecuación va a satisfacer el criterio de localización inmediata y suavizado por regiones.

-Localización inmediata: A cada resolución, los límites de las regiones deberían ser bien definidos y coincidir con los límites significativos semánticamente a esa resolución.

-Suavizado por regiones: El suavizado de una región no debería afectar más allá de los límites o bordes de la misma región.

Consideremos la ecuación de difusión Anisotrópica:

$$I_t = \text{div}(c(x,y,t)\nabla I) = c(x,y,t)\Delta I + \nabla c \cdot \nabla I \quad (3.3)$$

donde  $\text{div}$  indica el operador divergencia,  $\nabla$  es el gradiente y  $\Delta$  el laplaciano. El espacio variable se reduce a la ecuación de calor isotrópico a  $I_t = c\Delta I$  si  $c(x,y,t)$  es una constante. Suponiendo que en el tiempo (escalado)  $t$ , conocemos la ubicación de la región dentro de los límites apropiados a cada escala, nosotros esperaríamos fomentar el suavizado dentro de una región en preferencia sobre suavizar límites entre regiones. Esto se puede lograr al asignar 1 al coeficiente de conducción en el interior de cada región y 0 en los límites. La difuminación se puede tomar separadamente en cada región sin interacciones entre regiones, por lo que las regiones límites permanecerían afiladas.

Por supuesto no conocemos las regiones límites por anticipado en cada escala, pero pueden ser calculadas éstas mediante una estimación de la ubicación de los límites apropiados para esta escala.

Sea  $E(x,y,t)$  un estimador de la posición de los bordes: Una función vectorial-escalada y definida sobre la imagen la cual idealmente debe tener las siguientes propiedades: 1)  $E(x,y,t) > 0$  en el interior de cada región. 2)  $E(x,y,t) = Ke$  ( $x,y,t$ ) a cada punto del borde donde  $e$  es un vector unitario normal al punto del borde, y  $K$  es el contraste local (diferencia en la intensidad de la imagen sobre la izquierda y derecha) de cada borde.

Si un estimador  $E(x,y,t)$  está disponible, el coeficiente de conducción  $c(x,y,t)$  puede ser escogido de tal forma que sea una función  $c = g(|E|/|J|)$  de la magnitud de  $E$ . Donde  $g(\cdot)$  tiene que ser una función monótonicamente decreciente no negativa con  $g(0) = 1$ . Esta forma de proceso de difusión podrá tomar principalmente lugares en los interiores de las regiones, y esto no afectará a las regiones límites donde la magnitud  $E$  es grande.

El éxito del proceso de la difusión es el de satisfacer las tres metas del espacio Escalado (causalidad, localización inmediata y suavizado por regiones), esto dependerá de la exactitud del estimador  $E$  que es como un "suponer" de la posición de los bordes. Sin embargo es caro el calcularlo y requiere de algoritmos complicados. Afortunadamente en [33] y los resultados experimentales dados en el siguiente capítulo muestran que, el gradiente de la función de contraste  $E(x,y,t) = \nabla I(x,y,t)$  da excelentes resultados.

Hay muchas posibilidades para escoger a  $g(\cdot)$ . La más conocida es la función valuada binariamente.

### III.2.1 Propiedades de la Difusión Anisotrópica

Se ha establecido que la difusión Anisotrópica satisface el criterio de causalidad. Recordamos también un resultado general de la teoría de las ecuaciones diferenciales parciales, que nos dice que todos los máximos y los mínimos de las soluciones de la ecuación de difusión en el espacio-tiempo, corresponden a la condición inicial (imagen original) y a las condiciones de frontera (contornos) en el dominio espacio-tiempo. También se vio que una difusión en la cual el coeficiente de conducción es escogido localmente como una función de la magnitud del gradiente de brillantez,

$$c(x,y,t) = g(|\nabla I(x,y,t)|) \quad (3.4)$$

no solo preserva los bordes si no que, también los perfiles y la brillantez de los bordes es función de  $g(\cdot)$ , si esta se escoge apropiadamente.

#### III.2.1.1 El principio Máximo

El criterio de causalidad requiere que no aparezcan nuevos rasgos en la imagen en proceso, cuando se va desde una escala fina a una burda en el espacio escalado. Si identificamos los rasgos en la imagen con "blobs" de la función de brillantez  $I(x,y,t)$  para diferentes valores de escala para el parámetro  $t$ , entonces el origen de un nuevo "blob" implicaría la creación de cada uno de los máximos o mínimos los cuales tendrían que pertenecer uno a uno al interior o los límites  $I(x,y,t)$  del espacio escalado. Así, el criterio de causalidad puede ser establecido para mostrar que todos los máximos y mínimos en el espacio escalado pertenecen a la imagen original.

La ecuación de difusión (3.3) es un caso especial de una clase general de la ecuación elíptica que satisface el principio máximo. El hecho de que los máximos de la solución de la ecuación en el espacio y en el tiempo pertenezcan a la condición inicial (la imagen original), y a los límites del dominio de interés hace que el coeficiente de conducción sea positivo. Si usamos condiciones de límite adiabáticas, el principio máximo es uniformemente fuerte: el máximo únicamente pertenece a la imagen original.

#### III.2.1.2 Realce de bordes

Con un convencional filtrado paso-bajas y difusión lineal, el precio a pagar para eliminar el ruido, y para el funcionamiento del espacio escalado, es la difuminación de los bordes. Por esta causa la detección y la localización puede ser difícil.

El realce de bordes y la reconstrucción de imágenes borrosas pueden ser proyectadas por un filtrado paso-altas o ejecutando la ecuación de difusión en el tiempo hacia atrás.

En [33] se muestra que si el coeficiente de conducción se escoge apropiadamente de tal forma que sea una función del gradiente de la imagen, se puede hacer que la Difusión Anisotrópica realce los bordes mientras corre hacia adelante en el tiempo. Entonces se tendrá la estabilidad de Difusión la cual es garantizada por el principio máximo.

Modelaremos un borde como una función escalón convolucionada con una Gaussiana. Sin perder la generalidad, podemos asumir que los bordes están alineados con el eje Y. La expresión para el operador de divergencia simplificado es:

$$\operatorname{div}(c(x, y, t)\nabla I) = \frac{\partial}{\partial x}(c(x, y, t)I_x)$$

Escogemos  $c$  de tal forma que sea una función del gradiente de  $I: c(x, y, t) = g(I_x(x, y, t))$  como en (3.4). Definiendo  $\phi(I_x) = g(I_x) \cdot I_x$ ,

la versión 1-D de la ecuación (3.3) se convierte en

$$I_t = \frac{\partial}{\partial x} \phi(I_x) = \phi'(I_x) \cdot I_{xx} \quad (3.5)$$

Nuestro interés se centra en observar la variación en tiempo de la pendiente del borde:  $(\partial/\partial t)I_x$ . Si  $c'(I_x) > 0$  la función  $I(x)$  es suavizada, y el orden del diferenciador puede ser invertido:

$$\begin{aligned} \frac{\partial}{\partial t} I_x &= \frac{\partial}{\partial x} I_t = \frac{\partial}{\partial x} \left( \frac{\partial}{\partial x} \phi(I_x) \right) \quad (3.6) \\ &= \phi''(I_x) I_x^2 + \phi'(I_x) I_{xxx} \end{aligned}$$

Supongamos que el borde está orientado de forma tal que  $I_x > 0$ . En el punto de inflexión  $I_{xx} = 0$  y  $I_{xxx} < 0$  puesto que este corresponde a el punto con máxima pendiente (fig 3.2). Entonces en una vecindad del punto de inflexión,  $(\partial/\partial t)I_x$  tiene signo contrario a  $\phi'(I_x)$ . Si  $\phi'(I_x) > 0$ , la pendiente del borde decrecerá con el tiempo; si sucede lo contrario es decir,  $\phi'(I_x) < 0$ , entonces la pendiente aumentará con el tiempo.

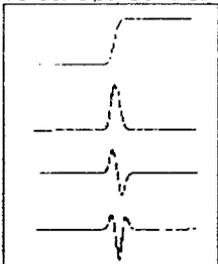


Fig 3.2 Una función escalón y 1ra, 2da y 3ra derivada.

Nótese que este incremento en la pendiente no podrá ser causado por un escalamiento en el borde, porque esto viola el principio máximo. Los bordes llegarán a ser afilados.

Es posible escoger  $\phi(\cdot)$ , por ejemplo,  $g(I_j) = C / (1 + (I_j/K)^{\alpha})$  con  $\alpha > 0$  (fig 3.3). Entonces existe un cierto valor de umbral que relaciona a  $K$  y  $\alpha$ . Antes del umbral  $\phi(\cdot)$  es monótonicamente creciente y después  $\phi(\cdot)$  es monótonicamente decreciente, dando el resultado deseable de pequeñas discontinuidades borrosas y afilamiento de bordes.

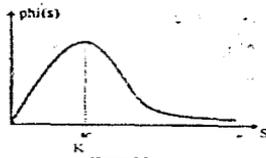


Figura 3.3

### III.3 Discretización de la ecuación de difusión

La ecuación (3.3) puede ser discretizada sobre una retícula cuadrada (o bien una matriz cuyos puntos tendrán coordenadas  $i, j$ ), con valores de brillantez asociados a los vértices, y coeficientes de conducción a los arcos (ver fig 3.4).

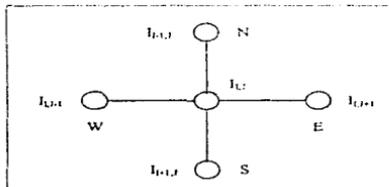


Fig 3.4

Se pueden usar los cuatro puntos vecinos a un punto para discretizar el operador Laplaciano y la ecuación (3.3) resultará en:

$$-I_{i,j}^{t+1} = I_{i,j}^t + \lambda \cdot [c_N \cdot \nabla_N + c_S \cdot \nabla_S + c_E \cdot \nabla_E + c_W \cdot \nabla_W]_{i,j} \quad (3.7)$$

donde  $0 \leq \lambda \leq 1/4$  para que el esquema numérico sea estable[33]. N, S, E y W son los mnemotécnicos para Norte, Sur Este y Oeste, los superíndices y los subíndices indican: número de iteración (i) y posición del punto en la matriz (i,j) respectivamente; el símbolo  $\nabla$  indica las diferencias entre puntos vecinos de tal forma que:

$$\begin{aligned}\nabla_N I_{i,j} &= I_{i-1,j} - I_{i,j} \\ \nabla_S I_{i,j} &= I_{i+1,j} - I_{i,j} \\ \nabla_E I_{i,j} &= I_{i,j+1} - I_{i,j} \\ \nabla_W I_{i,j} &= I_{i,j-1} - I_{i,j}\end{aligned}$$

El coeficiente de conducción es una función del gradiente de brillantez para cada iteración por lo que tenemos:

$$\begin{aligned}C'_{N,i,j} &= g(\|\nabla I\}'_{i-1,2,j}\|) \\ C'_{S,i,j} &= g(\|\nabla I\}'_{i+2,j}\|) \\ C'_{E,i,j} &= g(\|\nabla I\}'_{i,j+2}\|) \\ C'_{W,i,j} &= g(\|\nabla I\}'_{i,j-2}\|)\end{aligned}$$

El valor del gradiente puede ser evaluado sobre diferentes puntos vecinos proyectando diferentes compromisos entre exactitud y localización. Si se escoge simplemente a c como una proyección de la norma del gradiente podemos entonces calcular una aproximación de c mediante el valor absoluto de esta proyección a lo largo de la dirección del arco tenemos:

$$\begin{aligned}C'_{N,i,j} &= g(|(\nabla I)'_{i,j}|) \\ C'_{S,i,j} &= g(|(\nabla I)'_{i,j}|) \\ C'_{E,i,j} &= g(|(\nabla I)'_{i,j}|) \\ C'_{W,i,j} &= g(|(\nabla I)'_{i,j}|)\end{aligned}$$

Este esquema no es la discretización exacta de (3.3), pero conserva las propiedades de la ecuación continua por lo que la cantidad de brillantez de la imagen se conserva. El flujo de la brillantez en cada arco del enrejado solo depende de los valores de la brillantez de los dos nodos definidos por el arco.

Para la implementación de la ecuación discretizada la  $g(\cdot)$  que se utilizó es la siguiente:

$$g(\nabla I) = \frac{1}{1 + \left(\frac{|\nabla I|}{K}\right)^2} \quad (3.8)$$

### III.4 Implantación de la ecuación de Difusión Anisotrópica.

La implantación de la ecuación de Difusión Anisotrópica (ecuación 3.7), se realizó en Lenguaje C, en un principio en PC utilizando Borlandc 3.1 y finalmente en estaciones de trabajo SUN. La implementación en el Lenguaje C es debida a que el Estándar ANSI es prácticamente soportado por cualquier compilador, así que las diferencias entre un compilador y otro son mínimas, el programa en Lenguaje C es el siguiente:

```
#include<stdio.h>
#include<stdlib.h>
#include<malloc.h>

FILE *fe,*fs;
float *temp,*apun,*guarda;
unsigned char I;
void llenar();
void difusion();

int N,M,MAX;
float L,K;
char nombre[40],salida[40];

void main()
{
    int i,j;

    system("clear");
    printf("Numero de renglones:");
    scanf("%d",&N);
    printf("Numero de columnas:");
    scanf("%d",&M);
    printf("K=");
    scanf("%f",&K);
    printf("Lamda=");
    scanf("%f",&L);
    printf("Iteraciones=");
    scanf("%d",&MAX);
    printf("Archivo de entrada:");
    scanf("%s",nombre);
    printf("Archivo de salida:");
    scanf("%s",salida);
    if( !temp=(float *)malloc(N*M*sizeof(float)))==NULL)
    if( !apun=(float *)malloc(N*M*sizeof(float)))==NULL)
    {
        printf("No hay memoria suficiente para la imagen, presiona...");
        getch();
        exit(1);
    }
    else
```

```

{
difusion():
if(!!(f=fopen(salida,"wb"))!=NULL)
{
printf("Error al abrir archivo de salida");
getchar();
exit(1);
}
else
{
for(i=0;i<N;i++)
for(j=0;j<M;j++)
{
l=(unsigned char)temp[M*i+j];
fwrite(&l,sizeof(unsigned char),1,fs);
}
}
fclose(fs);
system("clear");
puts("Hecho");
}
}

void difusion()
{
register int i,j,l;
float dn,ds,dw,de;
int r1,r2,r3;

llena();
K*=K;
for(t=1;t<MAX;t++)
{
printf("\n%F",t);
for(i=0;i<N;i++)
{
r1=M*i;
r2=M*(i+1);
r3=M*(i-1);
for(j=0;j<M;j++)
{
/* Caso #1 */
if(i==0 && j==0)
{
ds=temp[r2+j]-temp[r1+j];
de=temp[r1+j+1]-temp[r1+j];
apun[r1+j]=temp[r1+j]*L*(1/(1+(ds*ds/K)))*ds+
1/(1+(de*de/K))*de;
}
/* Caso #2 */

```

```

if(i==0 && j>0 && j<M-1)
{
    ds=temp[r2+]-temp[r1+];
    de=temp[r1+1]-temp[r1+];
    dw=temp[r1+j-1]-temp[r1+j];
    apun[r1+]=temp[r1+]+L*((1/(1+(ds*ds/K)))^ds+
    1/(1+(de*de/K)))^de+{1/(1+(dw*dw/K))}^dw);
}
/*Caso #3*/
if(i==0 && j==M-1)
{
    ds=temp[r2+]-temp[r1+];
    dw=temp[r1+j-1]-temp[r1+j];
    apun[r1+]=temp[r1+]+L*((1/(1+(ds*ds/K)))^ds+
    1/(1+(dw*dw/K))}^dw);
}
/*Caso # 4*/
if((i>0 && i<N-1) && j==0)
{
    ds=temp[r2+]-temp[r1+];
    de=temp[r1+1]-temp[r1+];
    dn=temp[r3+]-temp[r1+];
    apun[r1+]=temp[r1+]+L*((1/(1+(ds*ds/K)))^ds+
    1/(1+(de*de/K)))^de+{1/(1+(dn*dn/K))}^dn);
}
/*Caso #5*/
if(i==N-1 && j==0)
{
    de=temp[r1+1]-temp[r1+];
    dn=temp[r3+]-temp[r1+];
    apun[r1+]=temp[r1+]+L*((1/(1+(dn*dn/K)))^dn+
    1/(1+(de*de/K))}^de);
}
/*Caso # 6 */
if(i==N-1 && j>0 && j<M-1)
{
    dn=temp[r3+]-temp[r1+];
    de=temp[r1+1]-temp[r1+];
    dw=temp[r1+j-1]-temp[r1+j];
    apun[r1+]=temp[r1+]+L*((1/(1+(dn*dn/K)))^dn+
    1/(1+(de*de/K)))^de+{1/(1+(dw*dw/K))}^dw);
}
/*Caso # 7*/
if(i==N-1 && j==M-1)
{
    dn=temp[r3+]-temp[r1+];
    dw=temp[r1+j-1]-temp[r1+j];
    apun[r1+]=temp[r1+]+L*((1/(1+(dn*dn/K)))^dn+
    1/(1+(dw*dw/K))}^dw);
}

```

```

/* Caso # 8 */
if((i>0 && i<N-1) && j==M-1)
{
    ds=temp[r2+j]-temp[r1+j];
    dw=temp[r1+j-1]-temp[r1+j];
    dn=temp[r3+j]-temp[r1+j];
    apun[r1+j]=temp[r1+j]+L*((1/(1+(ds*ds/K)))*ds+
        (1/(1+(dw*dw/K)))*dw+(1/(1+(dn*dn/K)))*dn);
}

/* Caso # 9 */
if((i>0 && i<N-1) && (j>0 && j<M-1))
{
    dn=temp[r3+j]-temp[r1+j];
    de=temp[r1+j+1]-temp[r1+j];
    dw=temp[r1+j-1]-temp[r1+j];
    ds=temp[r2+j]-temp[r1+j];
    apun[r1+j]=temp[r1+j]+L*((1/(1+(dn*dn/K)))*dn+
        (1/(1+(de*de/K)))*de+(1/(1+(dw*dw/K)))*dw
        +(1/(1+(ds*ds/K)))*ds);
}
}
}
guarda=apun;
temp=apun;
apun=temp;
}
}

```

```

void llena()
{
    int i;
    short int e;

    puts("Cargando imagen");
    if (fe=fopen(nombre,"rb")==NULL)
    {
        puts("Error al abrir archivo de entrada, presiona...");
        getchar();
        exit(1);
    }
    else
    {
        for(i=0;i<N;i++)
        {
            for(j=0;j<M;j++)
            {
                fread(&l,sizeof(unsigned char),1,fe);
                temp[M*i+j]=(float)l;
            }
        }
    }
}

```

```
}  
}  
fclose(fe);  
}  
puts("Imagen cargada");  
getchar();  
}
```

El programa requiere 7 parámetros de entrada que son:

1. 'N', variable entera que le dice al programa cuantos renglones (o pixeles de ancho) tiene la imagen a procesar.
2. 'M' variable entera que le dice al programa cuantas columnas (o pixeles de largo) tiene la imagen a procesar.
3. 'K' este parámetro determina la magnitud de los bordes a ser conservados durante el proceso, así, si se escoge una K grande, todos los bordes desaparecerán, mientras que si K es muy pequeña, todos los bordes serán conservados.
4. 'L' parámetro real que debe estar en el intervalo  $[0,1/4]$  para que el algoritmo sea estable.
5. 'MAX' es una variable entera que le indica al programa el numero de iteraciones por proceso.
6. El parámetro 'nombre' es una cadena de caracteres que le indica al programa el nombre del archivo de la imagen de entrada el cual deberá estar en formato RAW<sup>1</sup>.
7. El parámetro 'salida' le indica al programa el archivo de salida, o bien, la nueva imagen despues del proceso, en formato RAW.

---

<sup>1</sup>En los archivos con formato RAW, cada pixel de la imagen se almacena en un byte (se le asocia un numero a cada pixel) y el archivo no contiene ningun encabezado; por lo que el primer byte del archivo será el primer pixel de la esquina superior izquierda de la imagen, por tal razón se piden los parametros N y M.

### III.5 Espacio Escalado y Difusión Anisotrópica

Las siguientes imágenes muestran la representación de imágenes con el Espacio Escalado y con Difusión Anisotrópica, para distintos valores de  $t$  para el caso del Espacio Escalado y con distintos valores de  $K$  y el número de iteraciones para el caso de Difusión Anisotrópica.



Imagen Original



Espacio Escalado con una  $t=0.7071$



Difusión con  $K=5$ , 10 iteraciones y  $L=0.24$



Espacio Escalado con  $\sigma=1.22$



Difusión con  $K=5$ , 20 iteraciones y  $L=0.24$



Espacio Escalado con  $\sigma=1.58$



Difusión con  $K=5$ , 30 iteraciones y  $L=0.24$



Espacio Escalado con  $\sigma=1.94$



Difusión con  $K=5$ , 40 iteraciones y  $L=0.24$



Espacio Escalado con  $t = 3.02$



Difusión con  $K=5$ , 100 iteraciones y  $L=0.24$

## Capítulo IV

### Restauración de Imágenes usando Suavizado Adaptativo

El Suavizado Adaptativo es una técnica alterna para implementar la Difusión Anisotrópica presentada en el capítulo anterior, como consecuencia el Suavizado Adaptativo es un caso alterno del Espacio Escalado. En este capítulo se presenta el Suavizado Adaptativo así como su implementación en el Lenguaje C para estaciones de trabajo.

#### IV.1 Principio del algoritmo

Uno de los filtros más usado y comunes en el suavizamiento es el filtro gaussiano. Este filtro tiene propiedades importantes, en particular, no aparecen nuevos cruces por cero en el laplaciano de la señal suavizada conforme a la desviación estándar de la gaussiana, crece. Además, de acuerdo al Teorema del "Limite Central" la convolución gaussiana puede ser evaluada eficientemente por convoluciones en cascada con cualquier promedio finito de filtros usando una pequeña máscara de iguales pesos. Como ejemplo, en el caso de una señal 1-D, formularemos este proceso como sigue:

Suponiendo  $S^{(0)}(x)$  la señal antes del suavizamiento, la señal suavizada  $S^{(i+1)}(x)$  a la iteración  $(i+1)$  es simplemente

$$S^{(i+1)}(x) = \frac{1}{N} \sum_{i=-1}^{i=1} S^{(i)}(x+i) w^{(i)}(x+i) \quad (4.1)$$

con

$$N = \sum_{i=-1}^{i=1} w^{(i)}(x+i) \quad (4.2)$$

donde

$$w^{(i)}(x) = 1, \dots, \forall x \text{ y } \dots \forall t \quad (4.3)$$

Este filtro suaviza cualquier dato aún a lo largo de las discontinuidades. Si conocemos la localización de estos puntos discontinuos entonces podremos fijar los pesos correspondientes a la máscara de convolución  $w^{(i)}(x)$  a cero, así que el suavizamiento de la señal no tomaría en cuenta estos puntos que pertenecen a las discontinuidades; y dos puntos que pertenecen a las diferentes regiones separadas por una discontinuidad, no serán promediadas puesto que usamos una máscara muy pequeña. Para puntos pertenecientes a las discontinuidades el proceso de repeticiones promedio podrá forzar a la pertenencia de una de las regiones cercanas, de este modo se realizan los bordes.

Desafortunadamente no conocemos la localización de las discontinuidades. En este caso nuestros problemas estarían solucionados y no necesitarían de suavizado.

Podemos formular una suposición para evaluar a cada punto de la señal un valor continuo de  $w^{(1)}(x)$ , usando cualquier función decreciente  $f(d^{(1)}(x))$  tal que  $f(0) = 1$  y  $f(d^{(1)}(x)) \rightarrow 0$  conforme  $d^{(1)}(x)$  crece y donde  $d^{(1)}(x)$  representa la cantidad de discontinuidad al punto  $x$ . Una estimación de  $d^{(1)}(x)$  puede ser evaluada simplemente relacionando estos valores con la magnitud de la derivada a punto. Aquí escogemos la función  $f$  como:

$$w^{(1)}(x) = f(S^{(1)}(x)) = e^{-\frac{18|S^{(1)}(x)|^3}{2k^3}} \quad (4.4)$$

Donde  $S^{(1)}(x)$  es la derivada de la señal  $S^{(1)}(x)$  y  $k$  es un parámetro.

El parámetro  $k$  determina la magnitud de los bordes a ser conservados durante el proceso de suavizado. La influencia de este parámetro es completamente disculda en la siguiente subsección: resumiendo:

Si  $k$  se escoge grande todas las discontinuidades desaparecen y el resultado es el mismo, tal y como si el suavizado gaussiano fuera usado. Si  $k$  es pequeño, entonces todas las discontinuidades son conservadas y prácticamente no existe suavizado. Si nosotros esperamos a que el proceso de suavizado adaptativo converja, lo cual puede tomar un número extremadamente grande de iteraciones. La señal resultante será constante en una zona determinada. Es importante entender que hay 2 diferentes operaciones que afectan a la señal después del proceso de suavizado: uno es el afilamiento de los bordes subsistentes, el otro es el suavizado dentro de las regiones. El efecto del afilamiento es obtenido justo después de algunas iteraciones pero la parte de suavizado es extremadamente lenta y de efecto asintótico. Esto es razonable para el propósito de detección de bordes, el proceso iterativo finaliza después de unas cuantas iteraciones, puesto que los bordes no cambiarán. Se uso un simple esquema de umbral en el cual ésta es una simple función del parámetro  $k$ . Note que estos bordes se suelen usar para predecir el estado de convergencia final para una señal exacta y sin embargo la reconstrucción de ésta es necesaria.

Cuando la señal es una imagen 2-D  $I^{(1)}(x, y)$ , definimos  $d^{(1)}(x, y)$  como la magnitud del gradiente  $\left( \frac{\partial I^{(1)}(x, y)}{\partial x}, \frac{\partial I^{(1)}(x, y)}{\partial y} \right)^T - (G_x, G_y)$ , evaluada en una ventana de  $3 \times 3$ . Sin embargo el valor de continuidad  $w^{(1)}(x, y)$  es dado por las siguientes ecuaciones:

$$w^{(1)}(x, y) = f(d^{(1)}(x, y)) = e^{-\frac{18|G_x, G_y|^3}{2k^3}} \quad (4.5)$$

con

$$d^{(1)}(x, y) = \sqrt{G_x^2 + G_y^2} \quad (4.6)$$

la señal suavizada  $I^{(1)}(x, y)$  es entonces definida para cada punto  $(x, y)$  por

$$I^{(1)}(x, y) = \frac{1}{N^{(1)}} \sum_{i=-1}^{+1} \sum_{j=-1}^{+1} I^{(0)}(x+i, y+j) w^{(1)}(x+i, y+j) \quad (4.7)$$

con

$$N^{(1)} = \sum_{i=-1}^{+1} \sum_{j=-1}^{+1} w^{(1)}(x+i, y+j) \quad (4.8)$$

## IV.2 Una nueva representación del espacio escalado

El propósito de procesar una señal multiescala no es únicamente el poder identificar los rasgos importancia de las señales, sino también el de construir una representación de la señal con diferentes grados de detalle, la cual puede ser usada para tareas de visión de nivel alto. La causalidad es una propiedad útil para el procesamiento de señales multiescala, los rasgos detectados con una escala grande son causados por rasgos de las escalas más finas.

Una representación del espacio escalado de una señal en una dimensión puede ser visualmente desplegada por un diagrama el cual indique la posición de las discontinuidades de la señal, evaluando las diferentes escalas (ver fig. 4.1(c)). El eje Y indica la escala considerando el eje X que indica la posición de la discontinuidad a lo largo de la señal. Usando suavizado adaptativo pueden obtenerse dos diferentes representaciones del espacio escalado. La primera aproximación la cual es similar al espacio gaussiano, esto es fijando  $k$ , la sensibilidad o el umbral de la magnitud del gradiente. En este caso usamos el número de iteraciones como parámetro de escala (ver fig 4.1b).

El espacio escalado gaussiano es un caso especial de esta representación cuando  $k$  se fija en el infinito.

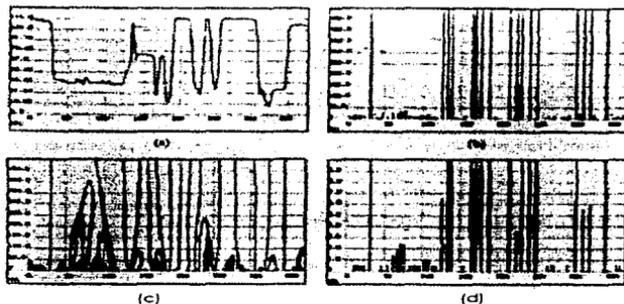


fig. 4.1

Este es un caso, en donde las localizaciones de los rasgos detectados no cambian mucho conforme las iteraciones aumentan (eje vertical).

En otro caso, cuando el suavizado gaussiano es usado los cruces por cero desaparecen y se unen conforme las iteraciones aumentan.

No es de sorprenderse que las discontinuidades desaparezcón completamente después de un cierto número de iteraciones.

La segunda aproximación, la cuál llamamos espacio de la escala adaptativo, se mantiene a  $k$  como parámetro de la escala, mientras que el número de iteraciones se mantiene fijo.

Preferimos esta aproximación porque la representación de las iteraciones en el espacio escalado no es útil en el caso del suavizado adaptativo, ya que la localización de los bordes no cambian a lo largo de la escala. Teniendo fijo el número de iteraciones en  $n=20$  el suavizado adaptativo puede ser aplicado a la señal de la fig 4.1a, con una  $k$  variante desde 0.5 hasta 10.5.

El diagrama de la fig 4.1d, nos muestra para cada valor de  $k$  (eje vertical) la localización de los cruces por cero a lo largo del suavizado de la señal de 1-D. Debido a que esta propiedad es deseable para el suavizado adaptativo, éste puede ser factible de representar una señal para una escala discreta, por ejemplo separado por  $1/8$ .

## IV.3 Suavizado y difusión

El promedio iterativo es comúnmente usado para suavizar las señales en tareas de visión temprana. El promedio de los pesos por iteración con los pesos de la constante  $k$  es exactamente implementado con un filtro lineal y convolucionado iterativamente con una máscara pequeña. Considerando el caso de una señal 1-D, podemos reformular el proceso promedio iterativo como sigue:

$$I^{n+1}(x) = c_1 \times I^1(x-1) + c_2 \times I^1(x) + c_3 \times I^1(x+1) \quad (4.9)$$

con

$$c_1 + c_2 + c_3 = 1 \text{ y } 0 \leq c_1, c_2, c_3 \leq 1 \quad (4.10)$$

para garantizar la estabilidad de procesos iterativos. Podemos reescribir el esquema de iteraciones como sigue:

$$I^{n+1}(x) - I^n(x) = c_1 \times (I^n(x-1) - I^n(x)) + c_3 \times (I^n(x+1) - I^n(x)) \quad (4.11)$$

En particular, cuando  $c_1 = c_3$  esto se reduce a:

$$I^{n+1}(x) - I^n(x) = c_1 \times ((I^n(x-1) - 2I^n(x)) + I^n(x+1)) \quad (4.12)$$

la cual es una aproximación discreta de la ecuación de difusión de calor:

$$\frac{\partial I}{\partial t} = c \nabla^2 I \quad (4.13)$$

Esta formulación es también equivalente al suavizamiento gaussiano y el número de iteraciones es relativo a  $\sigma$ , la desviación estándar del Kernel Gaussiano. Pero ya vimos que este filtro suaviza los datos uniformemente, incluso sobre las discontinuidades.

Cuando los pesos son diseñados para variar a lo largo de la localización espacial así como en tiempo, podemos reescribir el esquema del promedio del peso como sigue:

$$I^{n+1}(x) = c^1(x-1) \times I^n(x-1) + c^2(x) \times I^n(x) + c^3(x+1) \times I^n(x+1) \quad (4.14)$$

con

$$c^1(x-1) + c^2(x) + c^3(x+1) = 1 \quad (4.15)$$

y

$$0 \leq c^1(x-1), c^2(x), c^3(x+1) \leq 1 \quad (4.16)$$

La cual puede ser reorganizada como:

$$I^{n+1}(x) - I^n(x) = c^1(x-1) \times (I^n(x-1) - I^n(x)) + c^3(x+1) \times (I^n(x+1) - I^n(x)) \quad (4.17)$$

ó

$$I^{n+1}(x) - I^n(x) = c^1(x-1) \times (I^n(x+1) - I^n(x)) - c^3(x+1) \times (I^n(x) - I^n(x-1)) \quad (4.18)$$

Esta ecuación es implementada por medio del proceso de la difusión anisotrópica (de acuerdo a Perona y Malik)

$$\frac{\partial I}{\partial t} = \nabla(c\nabla I) \quad (4.19)$$

#### IV.4 Análisis del algoritmo

Ahora estudiaremos el suavizado adaptativo de la señal de 1-D en términos de la convergencia, y analizaremos la influencia del parámetro  $k$ .

1).-Comportamiento iterativo: Si esperamos a que el proceso del suavizamiento adaptativo converja lo cual pueda tomar un número extremadamente grande de iteraciones, el resultado de la señal en una zona determinada es constante.

El problema de estabilidad que nos concierne es el crecimiento limitado o la caída controlada los límites de la solución exacta de la ecuación de diferencia finita y también de todos los errores introducidos durante la evaluación porque los errores y la solución exacta son procesadas por las mismas operaciones aritméticas.

La estabilidad definida del criterio esencial es aquel proceso numérico que puede limitar la amplificación de todos los componentes de las condiciones iniciales.

Un proceso iterativo de coeficientes constantes tal como el suavizado Gaussiano (la solución de la ecuación diferencial parcial de difusión), puede ser escrita como:

$$u_{i,t+1} = Au_{i,t} + b_i \quad (4.20)$$

donde  $b_i$  es un vector columna que conocemos de los valores límites y ceros, y la matriz  $A$  es  $N \times N$  de elementos conocidos y  $N$  es el número de puntos. En orden para que el proceso pueda ser estable después para algunas normas  $A$  debe satisfacer:

$$\|A\| \leq 1 \quad (4.21)$$

La derivación básica de la condición de estabilidad viene de las siguientes ecuaciones de la norma

$$\|A^n\| = \|AA^{n-1}\| \leq \|A\| \|A^{n-1}\| \leq \dots \leq \|A\|^n \quad (4.22)$$

Si se tiene lo siguiente  $\|A\| \leq 1$  entonces  $\|A^n\| \leq 1$ , entonces la estabilidad es garantizada conforme la interacción progresiva.

El mismo principio puede ser aplicado a ecuaciones diferenciales parciales no lineales, más específicamente para el proceso iterativo la matriz  $A$  es realmente dependiente del tiempo. Permitiendo reescribir las iteraciones como sigue:

$$u_{i,t+1} = A_t u_{i,t} + b_i \quad (4.23)$$

Como una propiedad general de la definición de una norma, se tiene lo siguiente:

$$\|A_{n-1} A_{n-2} A_{n-3} \dots A_1 A_0\| \leq \|A_{n-1}\| \|A_{n-2}\| \dots \|A_1\| \|A_0\| \quad (4.24)$$

Por lo tanto, si podemos garantizar que  $\|A_k\| \leq 1$  para  $k=0,1,\dots,n-2,n-1$ , entonces podemos garantizar la estabilidad del proceso iterativo. En seguida mostraremos la matriz  $A_k$  del suavizado adaptativo que satisface los requerimientos de estabilidad.

Considerando la señal de 1-D, la ecuación de iteración del suavizado adaptativo:

$$u_k^{(i+1)} = \frac{1}{R^{(i)}} \sum_{s=-1}^{+1} u_{k+s}^{(i)} c_{k+s}^{(i)} \quad (4.25)$$

con

$$R^{(i)} = \sum_{s=-1}^{+1} c_{k+s}^{(i)} \quad (4.26)$$

Podemos reescribir la ecuación anterior en forma vectorial y matricial como:

$$u_{i+1} = A_i u_i \quad (4.27)$$

dónde  $A_i$  tiene la siguiente forma :

$$\begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & \frac{c_{x-2}}{c_{x-2} + c_{x-1} + c_x} & \frac{c_{x-1}}{c_{x-2} + c_{x-1} + c_x} & \frac{c_x}{c_{x-2} + c_{x-1} + c_x} & 0 & \dots \\ \dots & \dots & 0 & \frac{c_{x-1}}{c_{x-1} + c_x + c_{x+1}} & \frac{c_x}{c_{x-1} + c_x + c_{x+1}} & \frac{c_{x+1}}{c_{x-1} + c_x + c_{x+1}} & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (4.28)$$

La norma infinita de una matriz es simplemente la suma máxima del módulo de los elementos de la matriz. Los valores de continuidad son siempre positivos, la norma infinita de nuestra matriz de iteraciones es unitaria, por que:

$$\|A_i\|_{\infty} = \frac{c_{x-1}}{c_{x-1} + c_x + c_{x+1}} + \frac{c_x}{c_{x-1} + c_x + c_{x+1}} + \frac{c_{x+1}}{c_{x-1} + c_x + c_{x+1}} = 1 \quad (4.29)$$

En resumen el módulo de los valores característico de la matriz cuadrada no pueden exceder ésta norma infinita.

$$\rho(A) \leq \|A\|_{\infty} \quad (4.30)$$

Por eso la norma-2 de la matriz  $A_1$  es también limitada por la unidad y se deduce la estabilidad de la iteración del suavizado adaptativo como  $\|A_1\| \leq 1$ . Esto siempre es garantizado durante el proceso iterativo.

2) El parámetro  $k$ : En el suavizado adaptativo es uno de los parámetros que controla gradualmente el suavizado, es decir, tiene un papel similar a  $\alpha$  en el suavizado homogéneo. Escoger un valor para  $\alpha$ , es equivalente a escoger el número de convoluciones iterativas con una pequeña máscara promedio fijada.

El suavizado adaptativo es una implementación de la Difusión Anisotrópica y tiene un buen comportamiento en el espacio escalado del suavizado adaptativo cuando usa el número de iteraciones como el parámetro de escala en el espacio escalado. El suavizado adaptativo incluye interregiones del afilamiento de bordes tan buenas como las interregiones de los bordes suavizados. Las localizaciones de los interregiones de los bordes básicamente cambian conforme las iteraciones progresan, mientras la localización de la región de los bordes se desviará a lo largo de la escala de iteraciones hasta que ésta finalmente desaparezca. Se escoge el parámetro  $k$  (en el cálculo de la función de continuidad  $c$ ) para que sea el parámetro de escala del nuevo espacio escalado. Esta tiene una interpretación estable para cada escala cuando las iteraciones convergen y también la representación del espacio-escalado tiene la propiedad de la detección de los bordes exactos a diferentes escalas.

Consideremos una señal  $f(x)$  de 1-dimensión como se muestra (fig 4.2). Por la vecindad del borde,  $f(x)$  se incrementa conforme incrementa  $x$ ,  $f_x > 0$ , con  $f_{xxx} < 0$ . Para nuestro caso si recordamos la función de continuidad,

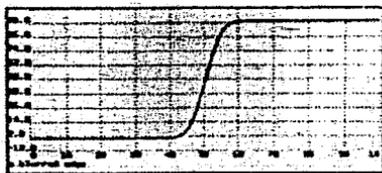


fig 4.2

$$C(x, t) = e^{-\frac{f_x(x,t)^2}{2k}} \quad (4.31)$$

y el suavizado adaptativo es la aproximación de la siguiente Difusión Anisotrópica.

$$\frac{df(x, t)}{dt} = \frac{d}{dx} (C(x, t) f_x(x, t)) \quad (4.32)$$

la cuál puede ser expandida a:

$$\frac{df(x, t)}{dt} = f_{\max}(x, t) e^{-\frac{f_x(x, t)^2}{2k^2}} \left( 1 - \frac{f_x(x, t)^2}{k^2} \right) \quad (4.33)$$

Se observa que para el efecto del afilamiento de bordes el término  $\frac{d}{dt} \left( \frac{df}{dx} \right)$  representa el cambio de el gradiente o contraste  $\left( \frac{df}{dx} \right)$  a lo largo de la dimensión de el tiempo a lo largo de iteraciones progresivas. Ya que  $f(x, t)$  es una función diferenciable el orden de diferenciación (con respecto a "x" y a "t") es intercambiable. Por lo tanto:

$$\frac{d}{dt} \left( \frac{df}{dx} \right) = \frac{d}{dx} \left( \frac{df}{dt} \right) = f_{\max} e^{-\frac{f_x^2}{2k^2}} \left( 1 - \frac{f_x^2}{k^2} \right) + f_{\max} \left( \frac{d}{dx} \left( e^{-\frac{f_x(x, t)^2}{2k^2}} \left( 1 - \frac{f_x(x, t)^2}{k^2} \right) \right) \right) \quad (4.34)$$

Puesto que  $f_{\max} < 0$ ,  $f_x = 0$  y el termino exponencial es siempre positivo la señal de  $\frac{d}{dt} \left( \frac{df}{dx} \right)$  esta determinada completamente por la señal de  $\left( 1 - \frac{f_x^2}{k^2} \right)$ .

$$\frac{d}{dt} \left( \frac{df}{dx} \right) > 0 \text{ cuando } f_x > k \quad (4.35)$$

y

$$\frac{d}{dt} \left( \frac{df}{dx} \right) < 0 \text{ cuando } f_x < k \quad (4.36)$$

Se observa que cuando el gradiente es suficientemente grande ( $f_x > k$ ) éste incrementará su magnitud conforme las iteraciones progresan, realizando el efecto de afilamiento de bordes, es decir si el gradiente es pequeño ( $f_x < k$ ) éste será suavizado eventualmente. El parámetro  $k$  es el que fija aproximadamente el umbral de la magnitud de el gradiente donde las discontinuidades son preservadas.

#### IV.5 Implantación paralela del algoritmo de suavizado adaptativo

El suavizado Adaptivo es apropiado para las nuevas redes paralelas de computadoras. Para una iteración paralela del Suavizado Adaptivo de una imagen  $I^{(n)}(x, y)$  tenemos que:

1) Calcular el gradiente  $G_x(x, y)$  y  $G_y(x, y)$ .

$$G_x(x, y) = \frac{1}{2} \left( I^{(n)}(x+1, y) - I^{(n)}(x-1, y) \right) \quad (4.37)$$

$$G_y(x, y) = \frac{1}{2} \{ I^{(1)}(x, y+1) - I^{(1)}(x, y-1) \} \quad (4.38)$$

2) Calcular los coeficientes de continuidad  $w(x, y)$

$$w(x, y) = e^{-\frac{G_x^2(x, y) + G_y^2(x, y)}{2k}} \quad (4.39)$$

3) Realizando la Promediación de los pesos para  $I^{(1)}(x, y)$ :

$$I^{(1)}(x, y) = \frac{\sum_{i=-1}^{+1} \sum_{j=-1}^{+1} I^{(1)}(x+i, y+j) w^{(1)}(x+i, y+j)}{\sum_{i=-1}^{+1} \sum_{j=-1}^{+1} w^{(1)}(x+i, y+j)} \quad (4.40)$$

Las operaciones elementales como la adición, multiplicación, exponenciales, son aplicadas simultáneamente para cada pixel de la imagen. Listando operaciones como el calculo de el gradiente, necesitamos solo la información vecina local (vecinos directos de los pixeles).

El algoritmo de Suavizado Adaptativo fue implantado en el Lenguaje C para estaciones de trabajo SUN, el programa es el siguiente.

```
#include<stdio.h>
#include<stdlib.h>
#include<malloc.h>
#include<math.h>

FILE *fe, *fs;
float *temp, *ante, *dir;
unsigned char l;
void llena();
void difusion();

int N,M,MAX;
float L,K;
char nombre[15],salida[15];

void main()
{
    int ij;

    system("clear");
    printf("Numero de renglones:");
    scanf("%d",&N);
    printf("Numero de columnas:");
    scanf("%d",&M);
    printf("K=");
    scanf("%d",&K);
```

```

printf("Iteraciones=");
scanf("%d",&MAX);
printf("Archivo de entrada.");
scanf("%s",nombre);
printf("Archivo de salida.");
scanf("%s",salida);
if ( (ante=(float *)malloc(N*M*sizeof(float)))==NULL ) )
if ( (temp=(float *)malloc(N*M*sizeof(float)))==NULL ) )
{
    printf("No hay memoria suficiente para la imagen, presiona ...");
    getch();
    exit(1);
}
else
{
    difusion();
    if((fs=fopen(salida,"wb"))==NULL)
    {
        printf("Error al abrir archivo de salida");
        getch();
        exit(1);
    }
    else
    {
        for(i=0;i<N;i++)
            for(j=0;j<M;j++)
            {
                *(unsigned char*)temp[M*i+j],
                fwrite(&1,sizeof(unsigned char),1,fs);
            }
        fclose(fs);
        system("clear");
        puts("¡Hecho!");
    }
}

void difusion()
{
    register int i,j,t,x,y;
    double gx,gy,w,num,deno;

    llenar();
    K**=K;

    for(t=1;t<=MAX;t++)
    {
        for(x=2;x<N-2;x++)
            for(y=2;y<M-2;y++)
            {
                num=0;
                deno=0;
                for(i=-1;i<=1;i++)

```

```

for(j=-1;j<=1;j++)
{
    gx=0.5*((double)temp[M*(x+i+1)+y+j]-(double)temp[M*(x+i-1)+y+j]);
    gy*=gx;
    gx=0.5*((double)temp[M*(x+i)+y+j+1]-(double)temp[M*(x+i)+y+j-1]);
    gy*=gy;
    num+=temp[M*(x+i)+y+j]*exp(-0.5*(gx+gy)/K);
    deno+=exp(-0.5*(gx+gy)/K);
}
ante[M*x+y]=(float)(num/deno);
}
dir=temp;
temp=ante;
ante=dir;
}
}

void llena()
{
    int i,j;

    puts("Cargando imagen");
    if( (fe=fopen(nombre,"rb"))==NULL)
    {
        puts("Error al abrir archivo de entrada, presiona...");
        getch();
        exit(1);
    }
    else
    {
        fgetc
        {
            for(i=0;i<N;i++)
            {
                for(j=0;j<M;j++)
                {
                    fread(&I,sizeof(unsigned char),1,fe);
                    temp[M*i+j]=1;
                }
            }
            fclose(fe);
        }
        puts("Imagen cargada");
        getch();
    }
}

```

El programa requiere 6 parámetros de entrada que son:

1. 'N', variable entera que le dice al programa cuantos renglones (o pixeles de ancho) tiene la imagen a procesar.
2. 'M' variable entera que le dice al programa cuantas columnas (o pixeles de largo) tiene la imagen a procesar.
3. 'K' este parámetro determina la magnitud de los bordes a ser conservados durante el proceso, así si se escoge una K grande, todos los bordes desaparecerán, mientras que si K es muy pequeña, todos los bordes serán conservados.

4. 'MAX' es una variable entera que le indica al programa el número de iteraciones por proceso.
5. El parámetro 'nombre' es una cadena de caracteres que le indica al programa el nombre del archivo de la imagen de entrada el cual deberá estar en formato RAW. El parámetro 'salida' le indica al programa el archivo de salida, o bien la nueva imagen después del proceso, en formato RAW.

## Capítulo V

### Resultados experimentales y pruebas

#### V.1 Resultados experimentales del algoritmo de Difusión Anisotrópica

##### V.1.1 Resultados variando los parámetros K y el número de iteraciones

En esta parte se presentan algunos resultados experimentales, en los cuales se muestran resultados distintos dependiendo de la variación de los parámetros K y número de las iteraciones, manteniendo al parámetro  $\lambda$  constante en 0.125. Una de las imágenes procesada tiene una dimensión de  $500 \times 500$  píxeles, es una imagen SAR. La otra es una imagen médica de  $256 \times 264$  píxeles.

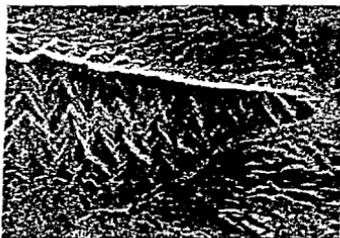


Imagen Original

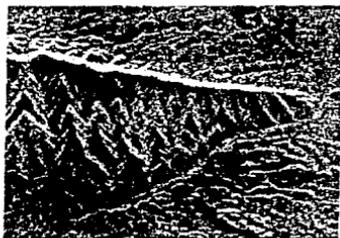


Imagen procesada K=1 y 9 iteraciones



Imagen procesada K=1 y 50 iteraciones



Imagen procesada K=5 y 10 iteraciones

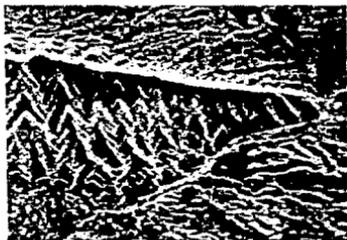


Imagen procesada  $K=5$  y 20 iteraciones



Imagen procesada  $K=5$  y 25 iteraciones



Imagen procesada  $K=5$  y 100 iteraciones

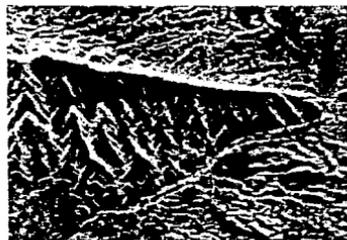


Imagen procesada  $K=10$  y 9 iteraciones



Imagen procesada  $K=10$  y 30 iteraciones

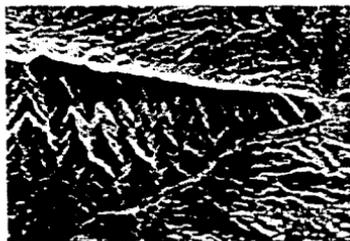


Imagen procesada  $K=20$  y 10 iteraciones



Imagen Procesada  $K=20$  y 29 iteraciones

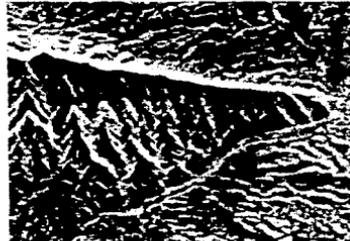


Imagen Procesadas  $K=40$  y 7 iteraciones



Imagen original



Imagen procesada con  $K=5$  y 20 Iteraciones

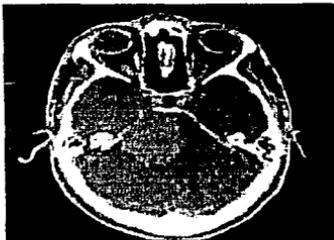


Imagen procesada  $K=5$  y 50 Iteraciones

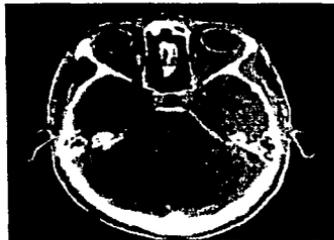


Imagen procesada  $K=2.5$  y 100 Iteraciones



Imagen procesada  $K=10$  y 50 Iteraciones



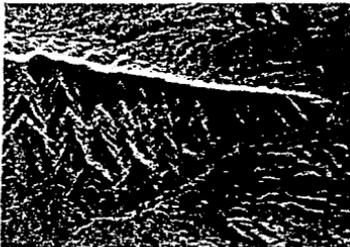
Imagen procesada  $K=50$  y 30 Iteraciones

Al observar las imágenes procesadas podemos decir lo siguiente: el parámetro  $K$  determina los bordes a preservar durante el proceso de suavizado, por ejemplo, cuando  $K=1$  y el número de iteraciones es 9 y 50, la imagen original y las procesadas son prácticamente iguales, todos los bordes han sido conservados al igual que el ruido; al observar las imágenes con  $K=5$  e iteraciones de 10, 20, 25 y 100 observamos como ciertos bordes han sido aplanados y el ruido se ha filtrado, el ruido se va suprimiendo (con este valor de  $K$ ) al ir aumentando el número de iteraciones mientras los bordes se conservan (aunque se aplanan lentamente si aumenta el número de iteraciones); se puede observar que cuando el valor de  $K$  aumenta el comportamiento del programa parece ser el mismo, por ejemplo cuando  $K=20$  y  $K=40$ , esto se debe a que la ecuación donde interviene  $K$  se comporta de manera asintótica (ver ecuación 3.8).

Podemos finalmente decir que  $K$  determina los bordes a conservar, mientras que el número de iteraciones ira suavizando la imagen mientras el número de iteraciones aumenta.

### V.1.2 Resultados variando el parámetro $L$

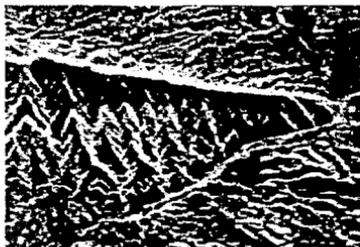
En esta parte se varia el parámetro  $L$ , el parámetro  $K=5$ , la imagen procesada es una imagen SAR de  $500 \times 500$  pixeles.



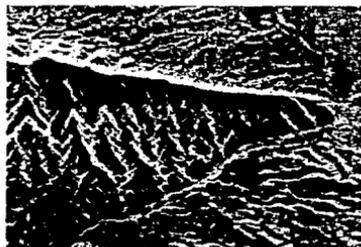
$L=0.010$  y 30 iteraciones



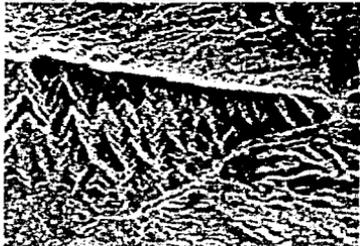
$L=0.040$  y 30 iteraciones



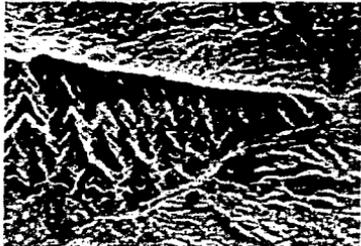
L=0.100 y 30 iteraciones



L=0.200 y 30 iteraciones



L=0.200 y 10 iteraciones



L=0.245 y 10 iteraciones

Al observar las imágenes podemos decir que el número de las iteraciones necesarias para obtener una "buena imagen" está en función del parámetro  $L$ , si  $L$  es pequeño el número de iteraciones será grande, por ejemplo cuando  $L=0,010$  y el número de iteraciones es 30, en esta imagen la cantidad del ruido es similar al de la imagen original por lo que el número de iteraciones tendrían que aumentarse para filtrar el ruido. Cuando  $L$  aumenta podemos llegar a imágenes similares con menor número de iteraciones, por ejemplo las imágenes cuando  $L=0,100$  con 30 iteraciones y  $L=0,200$  con 10 iteraciones son similares.

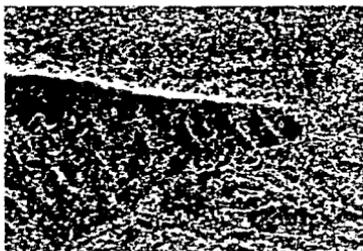
## V.2 Resultados experimentales del algoritmo de Suavizado Adaptativo

A continuación presentamos distintos resultados experimentales del algoritmo de Suavizado en función de la variación del parámetro  $K$  y el número de iteraciones.

La primera imagen presentada es la original llamada *seasat.raw* mientras las siguientes son las obtenidas después de utilizar el algoritmo de Suavizado Adaptativo.



Imagen Original Seasat.raw



k=1, 1 Iteración

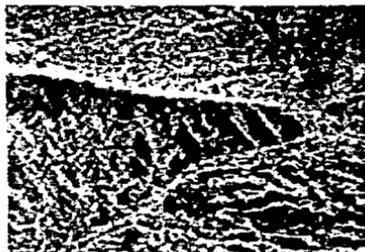


k=1, 2 Iteración

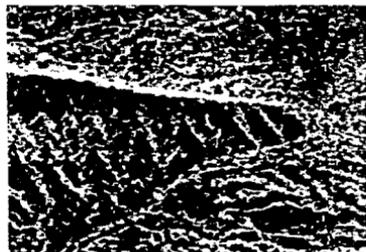


k=3, 1 Iteración

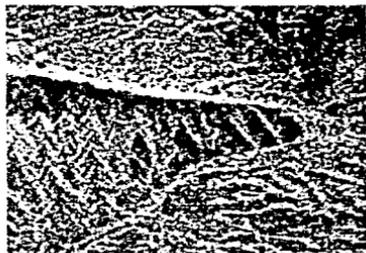
Como se puede observar, fijando el valor de  $K$  relativamente bajo las discontinuidades son conservadas y prácticamente no existe el suavizado.



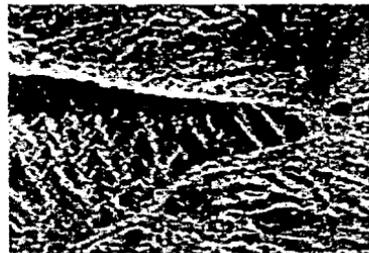
k=3, 4 Iteraciones



k=3, 6 Iteraciones



k=4, 2 Iteraciones

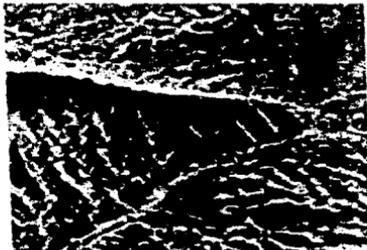


k=4, 4 Iteraciones

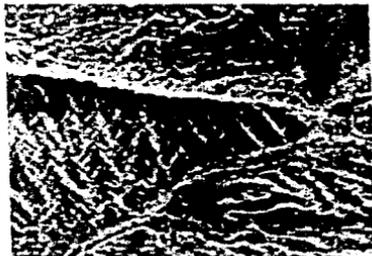
En las 4 siguientes imágenes observamos que teniendo una  $k$  relativamente mas grande la mayoria de las discontinuidades desaparecen como sucede en el suavizamiento gaussiano.



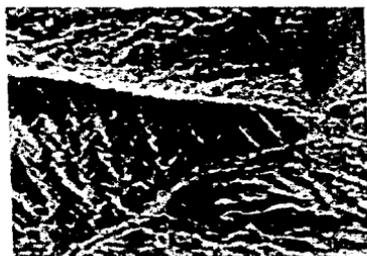
k=4, 5 Iteraciones



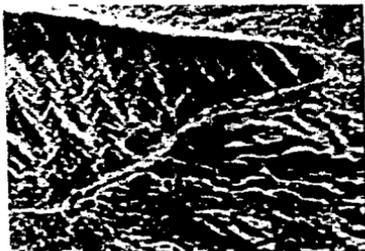
k=4, 6 Iteraciones



k=5, 4 Iteraciones



k=5, 5 Iteraciones



k=7, 3 iteraciones



k=7, 4 iteraciones



k=7, 7 iteraciones

En las tres últimas imágenes observamos que al tener una  $k$  mayor a las anteriores, ya hay un suavizamiento apreciable.

A partir de las imágenes que resultaron de aplicar el algoritmo de Suavizado Adaptativo, observamos que si la  $k$  es grande se suavizan los bordes, por lo que se van perdiendo los detalles finos de la imagen (ver imagen procesada con  $k=7$  y 3 iteraciones), pero si la  $k$  es pequeña tanto los bordes como el ruido se conservan (ver imagen procesada con  $k=1$  y 1 iteración).

### V.3 Evaluación de los algoritmos

Los resultados y pruebas que se realizaron y que presentamos en este capítulo se realizaron con los siguientes algoritmos para restauración de imágenes ruidosas SAR: Difusión Anisotrópica, Filtro de Mediana, Filtro de Lee Multiplicativo, Suavizado

Adaptativa, Filtro Geométrico y Transformada Polinomial, los algoritmos anteriores son métodos modernos para la reducción y restauración de imágenes ruidosas por lo que prácticamente todos estos métodos nos dan mejores resultados que los métodos estadísticos tradicionales.

Hicimos dos tipos de pruebas a los algoritmos: 1) Evaluaciones objetivas basadas en medidas locales y globales de contraste y relación de señal a ruido; estas evaluaciones fueron realizadas en el Instituto de Geofísica de la UNAM, y 2) Evaluaciones subjetivas en donde asumimos una escala numérica para valorar los resultados experimentales de los algoritmos en sus atributos básicos de percepción. Los atributos perceptuales usados fueron: cantidad de ruido, afilamiento de los bordes (nitidez de la imagen) y contraste [34].

Las evaluaciones subjetivas fueron realizadas en la División de Estudios de Posgrado de la Facultad de Ingeniería de la UNAM.

### V.3.1 Evaluación Objetiva

Una serie de pruebas fueron hechas sobre las imágenes que resultaron del procesamiento para evaluar el desempeño de cada algoritmo. Los parámetros considerados para estas evaluaciones fueron los siguientes:

- El factores de contraste  $cf$ , definido como:

$$cf = k \log \left( \frac{pc - p}{\sigma} + 1 \right) \quad (4.1), \text{ donde } k \text{ es una constante.}$$

- La media ( $\mu$ )
- La desviación estándar ( $\sigma$ )
- El factor de ruido granular ( $\sigma/\mu$ )

Estos parámetros fueron cuantificados sobre dos ventanas de la imagen, una contenía una textura suave y la otra con textura rugosa.

Factores Globales, imagen seosat.raw

Parámetros/Filtro	$cf$	$\mu$	$\sigma$	$\sigma/\mu$
Original	78	29.8993	16.2477	0.5434
Geométrico	31	30.7529	13.2870	0.4424
Mediana	42	29.2498	12.6186	0.4314
Lee Multiplicativo	41	29.8569	13.3490	0.4471
Difusión Anisotrópica	24	29.5045	13.6820	0.4637
Transformada Poli.	25	28.8887	13.0210	0.4507

Factores locales, zona homogénea, imagen seasat.raw

Parámetros/Filtro	cf	$\mu$	$\sigma$	$\sigma/\mu$	s/n
Original	71	23.1870	7.8838	0.3400	2.9411
Geométrico	19	24.2296	4.0607	0.1676	5.9668
Mediana	35	22.8626	4.8410	0.2117	4.7227
Lee Multiplicativo	34	23.1833	4.9988	0.2156	4.6378
Difusión Anisotrópica	12	22.7224	3.8724	0.1700	5.8807
Transformada Poli.	15	22.1888	4.0507	0.1826	5.4778

Factores locales, zona rugosa, imagen seasat.raw

Parámetros/Filtro	cf	$\mu$	$\sigma$
Original	83	34.8599	20.3255
Geométrico	41	35.9975	17.0496
Mediana	47	33.9090	15.8693
Lee Multiplicativo	46	34.7841	16.8119
Difusión Anisotrópica	36	34.4771	17.7123
Transformada Poli.	34	33.8403	16.5907

Tabla 5.1 Medidas objetivas.

El factor  $cf$  que es el logaritmo de la diferencia del pixel central y el promedio de los pixeles de la ventana de la imagen, nos dice que tan rugosa esta la imagen, por tal razón este factor es el más alto en la imagen original, mientras que el factor de las imágenes procesadas es menor, destacando la Difusión Anisotrópica que tiene el menor factor global y local, por lo que es "la imagen mas aplanada".

Otro factor importante de mencionar es el factor de ruido granular ( $\sigma/\mu$ ), que es el recíproco de la relación Señal a Ruido, donde el Filtro Geométrico, tiene el menor factor en una zona homogénea local, lo que significa que tiene una relación Señal a Ruido Mayor, en consecuencia es "el mejor" de acuerdo a este criterio.

### V.3.2 Evaluación Subjetiva

La evaluación de la calidad de la percepción de las imágenes ha sido un tema de estudios en la década pasada. La calidad de la percepción es usualmente estudiada como una simple impresión psicológica directamente determinada por varios parámetros físicos. Sin embargo, y especialmente cuando las imágenes son simultáneamente degradadas, por varios factores, es más ilustrativo el considerar a la calidad de la imagen como determinada por varios atributos básicos de percepción tales como percibir los bordes o el contraste, los cuales a la vez están determinados por los parámetros físicos.

Ocho sujetos tomaron parte en los experimentos. Dos de ellos tienen experiencia en imágenes SAR, así como en el procesamiento digital de imágenes. Los otros seis sujetos tienen poca experiencia en el análisis de imágenes SAR mientras considerable experiencia en el procesamiento de imágenes. Todos los sujetos tienen vista normal.

La imagen SAR de prueba usada tenía 990 x 990 pixeles. Este imagen fue procesada con el filtro de mediana (ventana de 3 x 3), Filtro de Lee Multiplicativo (ventana de 3 X 3 y varianza de 0.28), filtro geométrico (una iteración), Difusión

Anisotrópica, Suavizado Adaptativo y Transformada Polinomial, cada algoritmo tiene uno o más parámetros a ser ajustados. El número de combinaciones de parámetro de cada algoritmo es grande, por tanto se escogieron solo 15 imágenes las más representativas y generales por todos los algoritmos. Esto se enlistan en la tabla 5.1.

Cada uno de los sujetos observó las dieciséis imágenes y calificó la calidad global, la nitidez, el ruido y el contraste de cada imagen. Cada imagen fue desplegada en un monitor con una resolución de 1024 x 768 durante un periodo de 6 segundos. Las imágenes fueron presentadas de manera aleatoria y fueron repetidas cinco veces. Este proceso se repitió para calificar cada uno de los atributos por lo que cada sujeto observó 320 imágenes. El monitor fue puesto en cuarto donde las paredes estaban pintadas de color blanco. Los sujetos se sentaron a una distancia de unos 40cm de la pantalla del monitor, dando valores enteros, del 0 al 9, para darle un determinado valor a cada atributo de percepción. Una calificación de 9 significa una fuerte presencia del atributo, mientras que un 0 significa una ausencia del atributo.

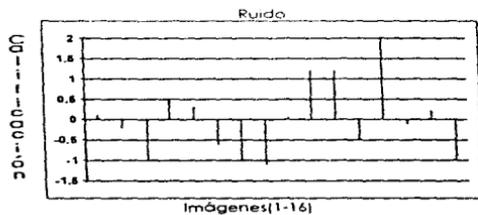
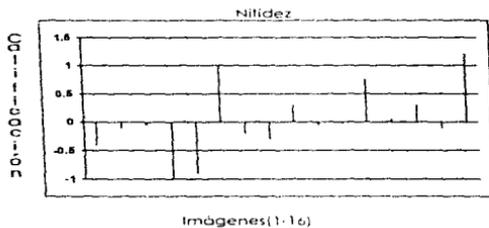
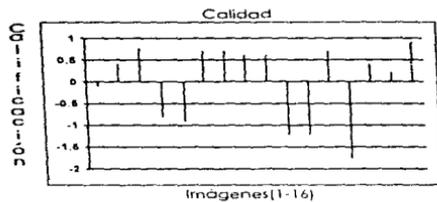
Para cada atributo fueron obtenidas 5 calificaciones por imagen. Los promedios fueron normalizados así que la media y la varianza fueron igual a cero y uno respectivamente. Se realizó el promedio para cada sujeto y al final el promedio para todos los sujetos, la figura 5.1 muestra los resultados obtenidos.

El número del eje de las 'x' representa los diferentes algoritmos como sigue:

Numero de la imagen	Algoritmo
1	Suavizado Adaptativo K=3, Iteraciones=2
2	Suavizado Adaptativo K=4, Iteraciones=2
3	Suavizado Adaptativo K=5, Iteraciones=4
4	Suavizado Adaptativo K=3, Iteraciones=5
5	Suavizado Adaptativo K=5, Iteraciones=5
6	Transformada Polinomial con bordes realzados
7	Difusión Anisotrópica K=5, Iteraciones=25
8	Difusión Anisotrópica K=5, Iteraciones=35
9	Difusión Anisotrópica K=5, Iteraciones=30
10	Filtro de Lee
11	Filtro de Mediana
12	Filtro Geométrico
13	Original
14	Transformada Polinomial resolución simple
15	Transformada Polinomial multiresolución
16	Transformada Polinomial filtrado direccional

Tabla 5.1

Los factores de correlación entre los atributos calificados muestran que la calidad depende del contraste (0.93) y el ruido (-0.92).



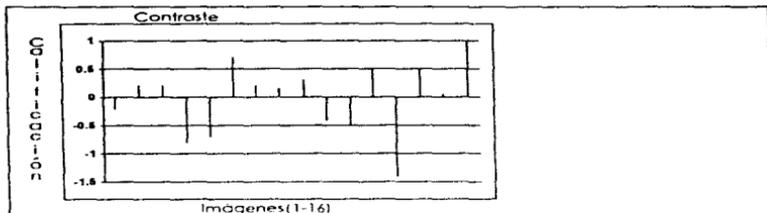


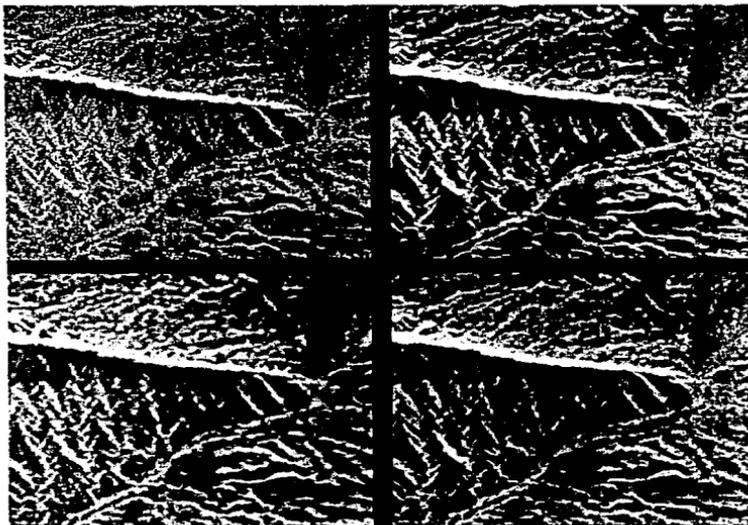
Fig 5.1 Calificaciones de la evaluación subjetiva. El eje de las "x" representa los diferentes algoritmos.

Al observar las graficas de la figura 5.1 y comparar el desempeño de los algoritmos concluimos que el mejor algoritmo es el de la Transformada polinomial, por ejemplo este algoritmo presenta la más alta puntuación en calidad, nitidez y contraste. Le sigue la Difusión Anisotrópica (Suavizado Adaptativo) y el filtro Geométrico que tienen la misma calidad, el Filtro Geométrico tiene mejor nitidez que la Difusión Anisotrópica, mientras que la Difusión Anisotrópica tiene la más baja puntuación de ruido de todos los algoritmos, esto significa que es el algoritmo que más redujo el ruido, a puntuación de contraste del Filtro Geométrico es más alta que la puntuación de contraste de Difusión Anisotrópica.

De acuerdo a las gráficas de la figura 5.1, la Transformada Polinomial presenta el mejor desempeño, seguido del Filtro Geométrico y la Difusión Anisotrópica.

ESTA TESIS NO DEBE  
SALIR DE LA BIBLIOTECA

El siguiente mosaico, muestra las mejores imágenes obtenidas con los algoritmos de Transformada Polinomial, Filtro Geométrico y Difusión Anisotrópica.



- La imagen de la esquina superior izquierda es la imagen original.
- La imagen de la esquina superior derecha fue procesada con Transformada Polinomial.
- La imagen de la esquina inferior izquierda fue procesada con Difusión Anisotrópica.
- La imagen de la esquina inferior derecha fue procesada con el Filtro Geométrico.

## Capítulo VI

### Conclusiones

- La Difusión Anisotrópica es un método para la representación de imágenes, presenta una solución al problema de la Teoría del Espacio Escalado.
- La Difusión Anisotrópica es un método útil para la restauración de imágenes SAR e imágenes medicas, tal afirmación queda sustentada en las pruebas del Capítulo IV, pero este método no se limita solo a los tipos de imágenes anteriores mencionados, ya que es un método general.
- Las pruebas presentadas en el Capítulo IV, muestran la utilidad del algoritmo de la Difusión Anisotrópica, al ser usado en imágenes SAR. En las evaluaciones de Capítulo IV, los mejores resultados fueron obtenidos con los algoritmos de la Transformada Polinomial, el Filtro Geométrico y la Difusión Anisotrópica. Sobresale el hecho que tanto la Transformada Polinomial como la Difusión Anisotrópica son métodos multiresolución generales, los cuales no fueron diseñados para imágenes SAR, a diferencia del Filtro Geométrico que se diseñó para imágenes SAR.
- De los resultados obtenidos en el Capítulo IV de la evaluación subjetiva, el ruido es identificado como el atributo más importante, la nitidez le sigue en importancia, mientras que la calidad y el contraste son dependientes del ruido y la nitidez [34] tradicionalmente se ha considerado a la nitidez como un factor con mayor importancia que el ruido. Esto no es una contradicción con los resultados presentados, pero esto es una evidencia que la nitidez varía poco entre los resultados de los algoritmos considerados en esta evaluación. Realmente esta la principal meta de las modernas técnicas de reducción de ruido, es decir, reducir el ruido sin perder la nitidez. En lo referente a ruido, las pruebas muestran buenos resultados con el Algoritmo de Difusión Anisotrópica, sin embargo introduce cierta apariencia artificial (lo que afecta la nitidez) esto es normal en este tipo de algoritmos, por otro lado esta característica es útil para la segmentación de imágenes.
- Mientras que en la evaluación objetiva el mejor algoritmo fue el filtro Geométrico, en la evaluación subjetiva la Transformada Polinomial presenta los mejores resultados, seguidos por el Filtro Geométrico y la Difusión Anisotrópica.
- La Difusión Anisotrópica puede ser utilizada para otras tareas tales como: detección de bordes, realce de bordes, segmentación de imágenes multiescala.
- En esta Tesis implantamos la Difusión Anisotrópica de dos formas, la primera se presenta en el capítulo III y la segunda en el capítulo IV, llamaremos para diferenciar a las implantaciones algoritmo 1 y algoritmo 2, ya que es necesario mencionar en cual de las implantaciones se obtuvieron mejores resultados. Construiremos una tabla a partir de las calificaciones obtenidas para los algoritmos presentadas en la figura 5.1, para presentar las diferencias de puntuación de cada uno de los algoritmos.

Párametro	Algoritmo 1	Algoritmo 2
Puntuación de ruido en las pruebas	-1	-1
Puntuación de calidad en las pruebas	0,6	0,8
Puntuación de nitidez en las pruebas	-0,3	-0,1
Puntuación de contraste en las pruebas	0,15	0,15

El algoritmo 2 tiene mejor puntuación que el algoritmo 1 lo que significa que se obtuvieron mejores resultados con el algoritmo 2, las calificaciones son muy similares aunque el algoritmo 2 presenta una ligera ventaja sobre el algoritmo 1. Podemos concluir que el algoritmo 2 de acuerdo a las pruebas que les hicimos a cada uno, resultó ser el mejor, si consideramos que el algoritmo 2 presenta mejores resultados encunto a calidad en la imagen y nitidez de la imagen.

Mientras el algoritmo 2 necesita de 2 parámetros K y el numero de iteraciones para su funcionamiento, el algoritmo 1 necesita de 3 parámetros K, L y el número de iteraciones para poder trabajar, lo que representa una ventaja para el algoritmo 2.

Una ventaja que presenta el algoritmo 1 sobre el algoritmo 2, es que el primero no usa la función exponencial para su implementación, además de ser mas sencillo ya que solo utiliza la primera diferencia reflejandote esto en el tiempo de ejecución, es decir el algoritmo 1 tarda menos que el algoritmo 2 por iteración.

- Hemos presentado la Difusión Anisotrópica, este método es un método útil y eficiente para la restauración de imágenes con ruido. Además de que el algoritmo de Difusión Anisotrópica es simple, se puede implementar en una arquitectura paralela y es posible su implementación usando redes híbridas analógico-digitales. Por otro lado si se observa la implementación en el Lenguaje C de ambas implantaciones de la Difusión Anisotrópica, observaremos que las operaciones utilizadas son las aritméticas elementales y ciclos, esto hace que al ejecutarse el programa, este sea muy "rápido por iteración".

---

## BIBLIOGRAFÍA

### 1.-Digital Images Processing

Autor: González, Rafael

### 2.-Scale space filtering: a new approach to multi-scale description

Autor: Andrew P. Witkin

Fairchild Laboratory for Artificial Intelligence Research

### 3.-Digital Image Processing

Autor: William K. Pratt

Segunda Edición.

### 4.-Optica

Autor: Zajac

Ed. Iberoamericana

### 5.-Speckle Filtering of Synthetic Aperture Radar Images: A Review

Autor: J.S. Lee and I. Jurkevich

Remote Sensing Reviews, 1994, Vol 8, pp 313-340

1994 Harwood Academic Publishers

### 6.-Adaptative Soomthing: A Generi Tool for Early Vision

Autor: Philippe Saint-Marc, Jet-Sen Chen

IEEE.

### 7.-Tratado de Histologia

Autor: Dr. Arthur W. Ham, University of Toronto

Sexta Edición.

### Referencias

- (1) Recibe y restauración de imágenes con degradación estadística en el dominio de la transformada polinomial hermitiana  
Tesis de Maestría de la Facultad de Ingeniería de la UNAM.  
Autor: Juan Román López Miranda
- (3) Lee, J. S. (1983) . Digital image noise smoothing and the sigma filter. *Computer and Image Processing* 24, 255-269.
- (4) Lee, J. S. (1980) Digital image enhancement and noise filtering by use local statistics. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 2 (2):165-168.
- (5) Lee, J. S. (1981) Speckle analysis and smoothing of synthetic aperture radar images. 1 *Computer Graphics and Image Processing* 17:24-32
- (6) Kuan, D. T, et al (1981) Adaptive noise filtering for images with signal-dependent noise. *IEEE Trans. Pattern Analysis and Machine and Intelligence PAMI-7* (2):165-177

- (7) Lin, Q. and Alleback, J. P. (1990) Combatting speckle in SAR images. Vector filtering and sequential classifications based on multiplicative noise model. IEEE Trans. Geoscience and Remote Sensing 28 (4): 647-653.
- (8) Nezry, E., Lopes, A. and Touzi, R. (1991) Detection of structural and textural features for Sar image filtering. Proceedings of IGARSS'91 IV: 2169-2160.
- (9) Lopes, A., Touzi, R. and Nezry, E. (1990) Adaptive speckle filters and scene heterogeneity. IEEE Trans. Geoscience and Remote Sensing 28 (6): 992-1000.
- (10) Lee, J. S. (1981) Refined filtering of image noise using local statistics. Computer Vision, Graphics, and Image Processing 15: 380-389.
- (11) Huang, T. S. et al. (1976) Speckle reduction in synthetic aperture radars. J. Opt. Soc. Am 66 (1): 1305-1311.
- (12) Wu, Y. and Maitre, H. (1990) A speckle suppression method for SAR image using maximum homogeneous region filter. Proceedings of IGARSS'90 III: 2413-2416.
- (13) Nathan, K. S. and Curlander, J. C. (1987) Speckle noise reductions of 1-look SAR imagery. Proceedings of IGARSS'87 II: 1457-1462.
- (14) Lee, J. S. (1986) Speckle suppression and analysis for SAR images. Optical Engineering 25 (5): 636-643.
- (15) Durand, J. M. et al. (1987) SAR data filtering for classification. IEEE Geoscience and Remote Sensing GE-25(5): 679-687.
- (16) Lee, J. S. (1983) A simple speckle smoothing algorithm for SAR. IEEE Trans. System, Man, Cybernetics SMC-13 (1): 636-643.
- (17) Frost, V. S. et al. (1982) A model for radar images and its application to adaptive digital filtering of multiplicative noise. IEEE Trans on Pattern Analysis and Machine Intelligence PAMI-4 (2): 157-165.
- (18) Frost, V. S. et al. (1981) An adaptive filter for smoothinh noisy radar images. IEEE Proceedings 69 (1): 133-135.
- (19) Crimmins, T. R. (1985) Geometric Filter for speckle reductions. Applied Optics 24(10): 1438-1443.
- (20) Crimmins, T. R. (1986) Geometric Filter for reducing speckle. Optical Engineering 1 (5): 651-654.
- (21) Safa, F. and Flouzat, G. (1989) Speckle removal on radar imagery based on mathematical morphology. Signal Processing 16: 319-333.
- (22) Kuan, Darwin, T., A. A. Sawchuck, T. C. Strand y P. Chavel. Adaptive noise smoothing filter for images with signal-dependent noise. IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-7,2, March 1985, 165-177.
- (23) F. C. Jeng and J. W. Woods. Inhomogeneous Gaussian Image Models for image estimation and restorations. IEEE Trans. Speech, Signal Processing Vol ASSp-36, 1305-1312, Agosto 1988.
- (24) Jeng, F. C. and Woods, J. W. . Compound Gauss-Markov random fields for image estimation. IEEE Trans. Signal Processing Vol 39 N.3, march 1991, 683-697.
- (25) D. Brzakavik. Automated iterative noise filtering. IEEE Trans. on Signal Processing Vol. 39 , N.1, Jan. 1991, 238-242.
- (26) Masiat, S. Hwang, W. L. Singularity detection and processing with wavelets. IEEE Trans. Inf Theory, Vol 38, N.2, March 1992, 617-643.

- 
- (27) Mallat, S. G. Characterization of signals from multiscales edges. IEEE Trans. Pattern Analysis Machine Intelligence Vol. 14, N. 7, Jul. 1992, 732-740.
- (28) Xu, Y., Weaver, J. B., Healy, D. M. Jr. Wavelet transform domain filters: a spatially selective noise filtration technique. IEEE Trans. Image Processing, V. 3 N. 6 Nov. 1994, 747-758.
- (29) Peters, R. A. A new algorithm for image noise reductions using mathematical morphology. IEEE Trans. Image Processing Vol. 4 N. 5 May 1995, 559-568.
- (30) Rochin, T., Couston, F. Speckle reductions in synthetic aperture radar imagery using Wavelets. SPIE Vol. 2034 Mathematical imaging: Wavelet transform applications to signal and image processing.
- (31) Donoho, D. L. Denoising by soft-thresholding. IEEE Trans. Inf. Theory Vol. 41, N. 3, May 1995, 613-627.
- (32) Barham, M. R., Wolfsones, N. P., Gonzalez. Multichannel restoration of signal channel images using wavelet-based subband decomposition. IEEE Image Processing Vol. 4, N. 6, Nov. 1994, 821-833.
- (33) P. Perona and J. Malik. Scale-space and edge detection using Anisotropic Diffusion. IEEE Trans. on Pattern Analysis and Machine Intelligence, V. 12, N. 7, Jul. 1990.
- (34) B. Escalante Ramirez, J. B. Martens and H. de Ridder. Multidimensional characterization of the perceptual quality of noise-reduced. Computed Tomography images. Journal of Visual Communication and Image Representation, 4, pp.317-334, 1993.
- (35) Jean Serra. Image Analysis And Mathematical Morphology. Centre de Morphologie Mathématique, Ecole Nationale Supérieure des Mines de Paris, Fontainebleau, France.  
Volume 2 Theoretical Advances  
4, pp.72-74, 1998.
-