ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

"Restablecimiento de Imágenes Renderizadas afectadas por Ruido Blanco Gaussiano Aditivo, Ruido Sal & Pimienta y Distorsión Geométrica Espacial con problemas de rotación y movimiento vertical, mediante la herramienta Matlab"

INFORME DE MATERIA DE GRADUACIÓN

Previo a la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Presentado por:

Julio César Almachi González

GUAYAQUIL – ECUADOR

AÑO: 2009

AGRADECIMIENTO

A Dios por la terminación de este proyecto, así como también a todas las personas que colaboraron con la realización del mismo, en especial a la MSc. Patricia Chávez Directora de Tesis por su invaluable ayuda y consejos.

DEDICATORIA

A Dios, mis Padres, mis hermanas y mi tía,

por su gran apoyo para dar este paso

crucial en mi carrera profesional.

TRIBUNAL DE SUSTENTACIÓN

M.Sc. Patricia Chávez
PROFESORA DE LA MATERIA

M.Sc. Juan Avilés
PROFESOR DELEGADO DEL DECANO

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de esta Materia de Graduación, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL".

(REGLAMENTO DE GRADUACIÓN DE LA ESPOL)

____

Julio César Almachi González

RESUMEN

El presente informe hace un estudio de los factores que afectan a las imágenes renderizadas en programas de diseño arquitectónicos y gráficos como Autocad, Adobe Photoshop y Arcón 6.5, el procesamiento de imágenes se basará en añadir ruido y distorsión geométrica espacial a las mismas para más tarde mediante el uso de algoritmos directos y filtros ser eliminados, se usa al error cuadrático medio normalizado como estimador de calidad en comparación con la imagen original que se procesa.

En el primer capítulo se da una referencia generalizada de los términos y definiciones usadas a lo largo del proyecto con la finalidad de tener claros los conceptos y poder desarrollar la idea de este trabajo.

En el segundo capítulo se da una explicación de cómo trabajan y la aplicación que tienen los algoritmos para eliminar el ruido de las imágenes destacando sus ventajas, además se explica en qué consiste la distorsión geométrica espacial con problemas de rotación y movimiento vertical, así como también como eliminarla. En el capítulo final se explica el desarrollo de la herramienta, los inconvenientes que se presentaron en su elaboración y las soluciones que se fueron hallando. Se añade un manual para el usuario y el código del programa terminado.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

Una vez terminado el diseño de una casa, edificio, departamento u otra estructura decorativa de una vivienda, se suele renderizar el mismo valiéndonos de alguna clase de software de diseño arquitectónico o gráfico con el objetivo de mostrarle a la persona interesada una perspectiva de cómo quedará visualmente hablando su bien inmueble, sobre el lugar que se vaya a construir; esta muestra es mediante una fotografía o imagen de dicho lugar. La idea es que mediante la renderización la persona interesada indique la clase de materiales, forma, color y ubicación que desea tenga la vivienda.

El ruido así como la distorsión geométrica espacial son problemas propios de la naturaleza y están asociados a cualquier toma de datos de la realidad. El problema de su añadidura involuntaria es muy común en la adquisición de voz, video y datos, son provocados por diversas causas, las cuales muchas veces no se pueden evitar. De tal forma que en el tratamiento de imágenes, se presentan como una fluctuación aleatoria y desmedida de los valores en los pixeles que componen dicha imagen distorsionándolos de la realidad visual.

En el procesamiento digital de señales se utilizan algunas técnicas sustentadas en el conocimiento de los operadores lineales y no lineales. El

avance de las tecnologías de comunicación e información permiten hoy en día la aplicación de imágenes (matrices) y transformaciones del álgebra lineal a diversas áreas de las ciencias puras y aplicadas así como de la ingeniería.
Los conceptos de espacio vectorial y algebra lineal resultan ser muy naturales en el procesamiento de señales. Y más aun en el procesamiento de imágenes digitales, ya que una imagen digital es una matriz. Las operaciones matriciales juegan un papel importante en el procesamiento de imágenes digitales. Una noción fundamental en el desarrollo de este estudio tiene que ver con que una imagen puede visualizarse de diferentes formas, y la manipulación de su representación es una de las más poderosas herramientas disponibles. Las manipulaciones útiles pueden ser lineales (ejemplo, las transformaciones) o no lineales (ejemplo, la cuantificación).

Para este proyecto se procesarán imágenes, a las cuales se les añadirá Ruido Blanco Gaussiano Aditivo ó Sal y Pimienta, Distorsión Geométrica Espacial con problemas de Rotación ó Movimiento Vertical, o la combinación de ambas; como técnicas para degradar las imágenes. Para restaurar las imágenes se utilizará cuatro algoritmos que servirán para eliminar el Ruido y dos para eliminar la Distorsión Geométrica Espacial de la imagen que esté siendo objeto de nuestro análisis.

Las pruebas que se llevaran a cabo son el error cuadrático medio normalizado, como variable principal analítica del proyecto, teniendo en cuenta que mientras menor sea el valor de dicho error, mejor será la calidad de la imagen restablecida.

Se incluye además un manual de ayuda de la herramienta de tal forma que pueda ser utilizada por cualquier persona con conocimiento básico de procesamiento digital de señales, seleccionando los algoritmos recomendados para la restauración de la imagen.

CAPÍTULO 1

1. Conceptos Generales

1.1 Definición de Renderización.-

Es el proceso de generar una imagen (imagen en 3D o una animación en 3D) a partir de un modelo, usando una aplicación de computadora. El modelo es una descripción en tres dimensiones de objetos en un lenguaje o estructura de datos estrictamente definidos. Este proceso se desarrolla con el fin de imitar un espacio 3D formado por estructuras poligonales, comportamiento de luces, texturas, materiales (agua, madera, metal, plástico, tela, etcétera) y animación, simulando ambientes y estructuras físicas verosímiles.

Figura 1.1 Imagen de una casa renderizada

Una de las partes más importantes de los programas dedicados a la infografía son los motores de renderizado, los cuales son capaces de realizar técnicas complejas como radiosidad, raytrace (trazador de rayos), canal alfa, reflexión, refracción o iluminación global. La renderización se utiliza en la producción de imágenes en 3D para juegos, diseño computacional, efectos especiales del cine y la TV, etc.

1.2 Definición de Ruido Blanco Gaussiano Aditivo.-

Ruido blanco quiere decir que no hay correlación en el tiempo y gaussiano que indica que la densidad de probabilidad responde a una distribución normal (o distribución de Gauss).

Figura 1.2 Comparación entre imagen sin ruido y con ruido

1.3 Definición de Ruido Sal y Pimienta.-

Es aquel en el que hay ocurrencias aleatorias de pixeles completamente blancos y completamente negros.

1.4 Definición de Distorsión Geométrica Espacial.-

Las principales causas de distorsión geométrica espacial en las imágenes están originadas por la falta de linealidad de los gradientes magnéticos y por la falta de homogeneidad del campo magnético estático al que están sometidos los núcleos de hidrógeno del cuerpo. Cualquier separación del carácter ideal conducirá a distorsiones geométricas en la imagen.

Figura 1.3 Imagen con problemas de Rotación y Movimiento vertical

1.5 Definición de Algoritmos de Eliminación del Ruido.-

Dado que este objetivo se puede cumplir de diferentes formas, por ello es que se han desarrollado diferentes formas que permiten analizar una señal y posteriormente sintetizarla reduciendo el ruido que presentaban originalmente.

1.6 Definición de Transformada Wavelet.-

Un tipo especial de la transformada de Fourier, la transformada wavelet o transformada óndula representa una señal en términos de versiones trasladadas y dilatadas de una onda finita (denominada óndula madre). La teoría de óndulas está relacionada con campos muy variados. Todas las transformaciones de óndulas pueden ser consideradas formas de representación en tiempo-frecuencia y, por tanto, están relacionadas con el análisis armónico.

1.7 Definición de Filtros Wavelet.-

Estos filtros son eficientes para el análisis local de señales no estacionarias y de rápida transitoriedad y, al igual que ciertos filtros de Fourier con Ventana, mapean la señal en una representación de tiempo-escala. El aspecto temporal de las señales es preservado. La diferencia está en que la Transformada Wavelet provee análisis de multiresolución con ventanas dilatadas. El análisis de las frecuencias de mayor rango se realiza usando ventanas angostas y el análisis de las frecuencias de menor rango se hace utilizando ventanas anchas.

1.8 Definición de Error Cuadrático Medio Normalizado.-

Es un estimador de las desviaciones en general entre los valores previstos y medidos. Se define como:

$$MSE_{normalizado_j} = \frac{1}{Var(x_t)} * \frac{\sum_{t=m+1}^{M} [x_t - \widehat{x_t}]}{M}$$

$$MSE_{normalizado_j} = Error\ Cuadrático\ Medio\ Normalizado$$

$$Var(x_t) = Varianza\ de\ la\ serie\ temporal$$

$$x_t = Valor\ predicho$$

$$M = El\ número\ total\ de\ observaciones$$

CAPÍTULO 2

2. Algoritmos usados para restablecer las imágenes del Ruido y la Distorsión Geométrica Espacial

2.1 Algoritmos para restaurar las señales del Ruido

Las señales, durante su transmisión, siempre se encuentran bajo la influencia de otras señales no deseadas. Incluso, cualquier procesamiento que se realice a una señal tiende a introducir perturbaciones desagradables en ella misma. A estas perturbaciones que contaminan la señal transmitida o procesada se le llama ruido, y constituye una señal molesta que no guarda relación alguna con la útil.

2.1.1 Algoritmo Mediana

El filtro de mediana es una técnica de filtrado lineal digital, a menudo utilizado para eliminar el ruido de las imágenes o de otras señales. La mediana de filtrado es un paso común en el procesamiento de imágenes. Es particularmente útil para reducir el ruido speckle y sal y pimienta. Su borde de conservación de la naturaleza lo hace útil en los casos en que el borde confusión no es deseable.

La idea es calcular la mediana de los valores de píxeles vecinos. Esto se puede hacer mediante la repetición de estos pasos para cada píxel en la imagen.

- Almacene los píxeles vecinos en una matriz. Los píxeles vecinos pueden ser elegidos por cualquier tipo de forma, por ejemplo, una caja o una cruz. La matriz se llama la ventana, y debe ser tamaño impar.
- Ordenar la ventana en orden numérico
- Elegir la mediana de la ventana como el valor de píxeles.

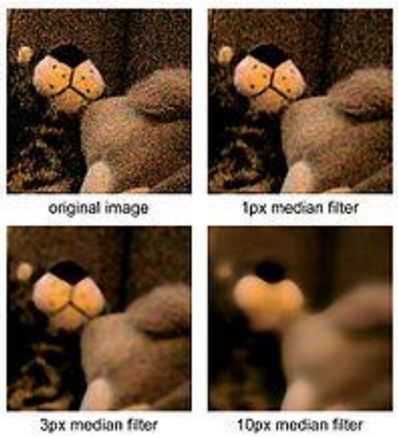

Figura 2.1 Ejemplo filtro de mediana, para 3 diferentes radios aplicados a la misma fotografía con ruido.

2.1.2 Algoritmo Hard Wavelet Tresholding

El valor del umbral fuerte se refiere a la operación de la aplicación de la no linealidad a todos los coeficientes wavelet excepto los promedios de escala gruesa, es una transformada wavelet basada en Symmlets con 8 momentos de fuga. La experiencia previa con umbral duro tradicional no invariante de-noising sugiere que produce un poco más las oscilaciones en las proximidades de las discontinuidades que su umbral suave. Sin embargo, es de esperar que la utilización del umbral fuerte obstaculizará algunas de esas oscilaciones.

2.1.3 Algoritmo Hyperbolic Wavelet Tresholding

Se han estudiado diversos aspectos relativos a la utilización de las ondas y los regímenes de hiperbólica aproximación adaptativa para expansiones wavelet de funciones de onda correlacionadas. Con el fin de analizar las consecuencias de la reducción de la regularidad de la función de onda de electrones en la cúspide de la electrónica, lo primero que considera realista resoluble exactamente muchos modelos de partículas en una dimensión. Las tasas de convergencia de las expansiones de wavelets, con respecto a la L2 y las normas H1 y la energía, se establecieron para este modelo. Se compara el rendimiento de las ondas hiperbólicas y sus extensiones a través de refinamiento de adaptación en la región de la cúspide, a un tratamiento totalmente adaptable basado en la contribución de la energía de las ondas individuales. A pesar que la onda hiperbólica muestra un comportamiento de convergencia inferior, que pueden ser fácilmente refinados en la región de la cúspide de la obtención de una tasa de convergencia óptima de la energía. Los resultados preliminares para el átomo de helio se presentan, lo que demuestra la transferencia de nuestras observaciones a los sistemas más realista. Proponemos un esquema de la contracción de ondas en la región de la cúspide, lo que reduce el número de grados de libertad y de los rendimientos de un costo-beneficio favorable para la evaluación de elementos de la matriz.

2.1.4 Algoritmo Mediana Inteligente

Posee las mismas características que el Algoritmo mediana, con la diferencia de que con este algoritmo se elige la mediana de un conjunto de pixeles vecinos y no únicamente de un solo grupo de pixeles, allí es donde radica la diferencia con el anterior algoritmo.

2.2 Distorsión Geométrica Espacial

Las transformaciones Geométricas modifican la relación espacial entre píxeles. En términos del procesamiento de imágenes digitales una transformación geométrica consiste de dos operaciones básicas:
Una transformación espacial que define la reubicación de los píxeles en el plano imagen.

Interpolación de los niveles de grises, los cuales tienen que ver con la asignación de los valores de intensidad de los píxeles en la imagen transformada.

En términos Matemáticos las transformaciones afines son las más usadas en imágenes digitales 2D por su representación y manejo matricial. Se aplica la transformada T sobre el espacio (n1,n2) para convertirlo en el plano (m1,m2):

$$T\left(n_1, n_2\right) = \left(m_1, m_2\right)$$

2.2.1 Matriz de Transformación de Rotación

$$T = \begin{bmatrix} \cos(\theta) & sen(\theta) & 0 \\ -sen(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \begin{aligned} m_1 &= n_1 \times \cos(\theta) - n_2 \times sen(\theta) \\ m_2 &= n_1 \times sen(\theta) + n_2 \times \cos(\theta) \end{aligned}$$

2.2.2 Matriz de Transformación para Movimiento Vertical

$$T = \begin{bmatrix} 1 & \beta & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \begin{aligned} m_1 &= n_1 \\ m_2 &= \beta \times n_1 + n_2 \end{aligned}$$

CAPÍTULO 3

3. Metodología del proyecto

En el desarrollo de este proyecto se presentaron una serie de inconvenientes, los cuales fueron resueltos mientras se calibraba el programa diseñado usando varias imágenes para detectar las fallas que iban apareciendo.

3.1 Toma de la foto que servirá de fondo al diseño

Se procedió a la toma de la fotografía que servirá de background del diseño sea este una casa, edificio, departamento u otra estructura decorativa del mismo.

Se utilizó una cámara digital de 800 mega pixeles de resolución incluyendo el flash, con el objeto que la imagen obtenida sea lo más clara posible y minimizar en algo el ruido; para minimizar la distorsión geométrica con problema de rotación se uso un trípode que incluía la cámara, y finalmente se optó por capturar ambientes a plena luz del día y también en la noche.

Como segunda alternativa también se han procesado imágenes renderizadas cuyo fondo fue directamente hecho en el programa gráfico usado.

3.2 Diseño, Renderizado y Montaje

Pare hacer el diseño, renderización y el montaje se utilizaron varios programas de diseño arquitectónico y gráfico como Autocad, Adobe Photoshop y Arcón 6.5. Cabe destacar que el montaje sobre el ambiente que servirá de background de la casa, departamento, edificio o estructura decorativa se realizó de dos maneras:

- Sobre la fotografía tomada en campo del sector donde se quiere construir o ubicar el bien.
- Sobre un ambiente creado ya con los mismos softwares de diseño antes mencionados, dado que en ciertos casos el terreno donde se va a construir estaba muy alejado de la ciudad.

3.3 Funcionalidad de la herramienta

En la construcción de esta herramienta opté por el uso de MATLAB 7.4.0 (R2007a), herramienta de lenguaje sencillo matemático que entre sus cualidades tiene la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos de hardware.

La herramienta que he planteado, está hecha mediante una interface gráfica de usuario comúnmente conocida como GUI, la misma que posee (Ver ANEXO A):

- Una fila de menús amigable para el usuario.
- Botones para abrir y guardar la imagen que se encuentre como actual.
- Panel de Ruido con sus respectivos controles para añadir ruido aditivo blanco gaussiano, o ruido sal y pimienta.
- Panel de Distorsión Geométrica Espacial con sus respectivos controles que me permitirán añadir rotación o movimiento vertical.
- Panel de Restauración de la imagen donde se seleccionará el algoritmo y filtro wavelet que se quieren usar para restablecer la imagen del ruido o la distorsión geométrica espacial.
- Panel del Error cuadrático medio normalizado que será un indicador de la calidad con la que se ha restaurado la imagen procesada.

- Y finalmente el panel del Procesamiento Digital de la Imagen que es donde se podrán observar los cambios que en la imagen se vayan dando luego de ir usando los controles de los paneles antes mencionados.

3.4 Resultados Obtenidos

Luego de analizar 6 imágenes renderizadas y a color para con cada uno de los 4 algoritmos utilizados, (también se podrían procesar imágenes en escala de grises pero no tendría sentido porque no se distinguiría la renderización aplicada motivo de este proyecto) se han alcanzado resultados aceptables obteniendo en el peor de los casos errores cuadráticos medios normalizados relativamente bajos.

Dicho error cuadrático medio normalizado difiere dependiendo del algoritmo y filtro wavelet seleccionados, por tal razón hay que buscar una combinación de dichos 2 parámetros que me arrojen un error bajo de tal forma que mi imagen restablecida sea de muy buena calidad (Ver ANEXO B).

Los resultados obtenidos del análisis cuantitativo dejan entrever que se produjo una mejor calidad de imágenes restablecidas para con niveles bajos tanto de ruido como de distorsión geométrica espacial.

Mientras que del análisis cualitativo, después de realizada una encuesta a 100 personas sobre que algoritmo que reconstituye de mejor manera y visualmente hablando una imagen previamente procesada con esta herramienta se obtuvo que para dichas 100 personas; el algoritmo de la mediana inteligente era quien mejor restablecía las imágenes.

LAS CONCLUSIONES SON:

1) La eliminación de ruido utilizando la trasformada wavelet tiene muchas ventajas en comparación de los filtros de dominio espacial. Siendo una transformación del dominio frecuencial, se comporta mejor (en muchos casos) que los filtros tradicionales basados en la transformada de Fourier. Sin embargo, entender la transformada de Fourier es esencial para entender la transformada wavelet y la comprensión de esta herramienta nos hace más expertos al momento de aplicarla.

2) Los algoritmos de Mediana y Mediana Inteligente son excelentes para eliminación de los ruidos tanto el blanco gaussiano aditivo, como el ruido sal y pimienta, puesto que al compararla con la imagen original procesada el error cuadrático medio normalizado fueron aceptablemente bajos.

3) Los Algoritmos de Hard Wavelet Threshold y Hyperbolic Wavelet Threshold son ideales para eliminar ruido blanco gaussiano aditivo, con la ventaja de que con estos dos algoritmos deberé escoger al filtro wavelet que menor error me arroje.

4) Del análisis cuantitativo concluyo que para cuando uso el algoritmo de la mediana es recomendable usar un valor de N=7; que para cuando use el algoritmo de hard wavelet threshold arroja bajos valores de error usar un L=80 y seleccionar filtros wavelet como son el sym17 y bior6.8; que para cuando use el algoritmo hyperbolic wavelet threshold es apropiado usar un valor de L=80 y un filtro wavelet db10; que para cuando use el algoritmo de la mediana inteligente use un valor de umbral=0.1.

5) Del análisis cualitativo deduzco que el mejor algoritmo para restablecer imágenes procesadas en la actual herramienta es el de la mediana inteligente.

LAS RECOMENDACIONES SON:

1) Se deben tomar fotografías para el background con el flash encendido, alta resolución de pixeles y desde una posición estratégica desde donde se pueda observar claramente a la construcción.

ANEXOS

ANEXO A

MANUAL DEL USUARIO

La herramienta está diseñada para añadir dos clases de ruido, uno a la vez bien sea el ruido blanco gaussiano aditivo ó el ruido sal y pimienta; también se puede añadir distorsión geométrica espacial con problemas de rotación ó con problemas de movimiento vertical; estos son a elección del usuario, es decir, si desea añadir ruido y distorsión o al menos uno de los dos.

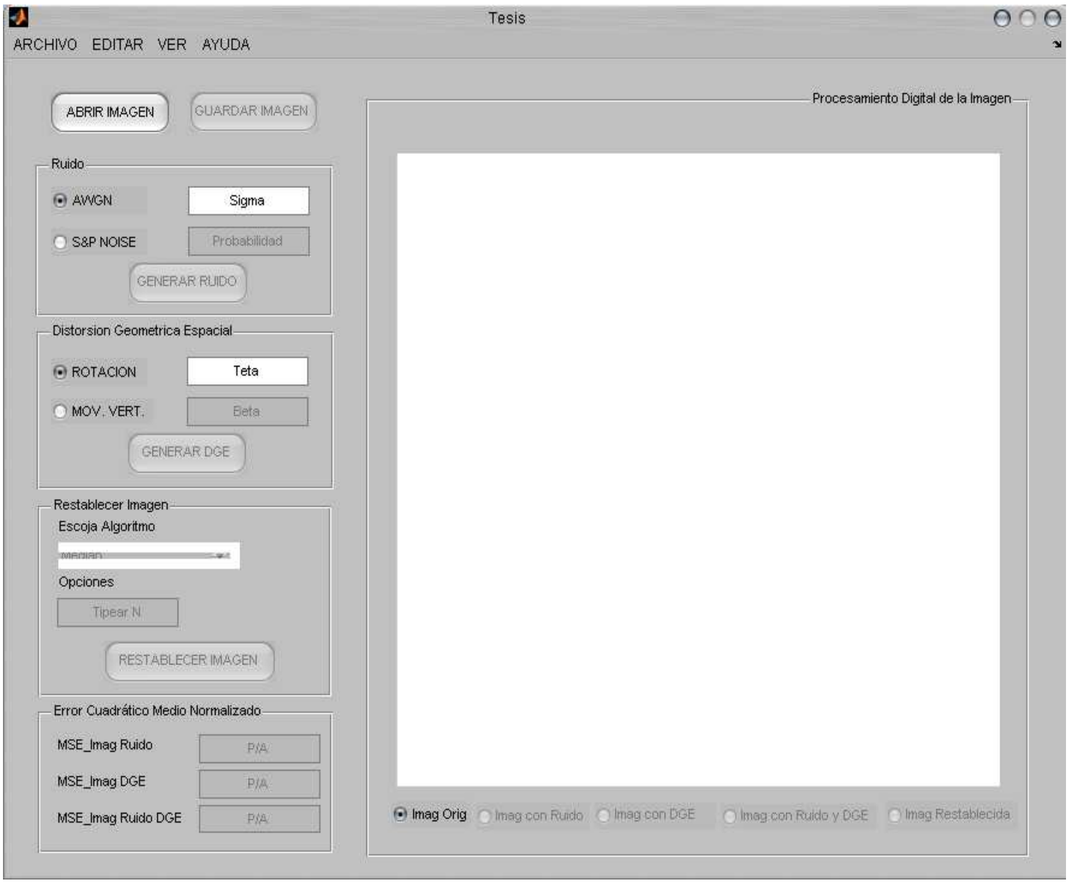

Figura A-1 Ventana principal de la herramienta

MENÚS

- **Menú Archivo**.- abre una imagen renderizada, guarda la imagen que se encuentre como actual y sale de la herramienta.

Figura A-2 Menú ARCHIVO

- **Menú Ayuda**.- Muestra una ayudad de la herramienta y un ejemplo para aprender a usarla.

Figura A-3 Menú AYUDA

PANEL DE RUIDO

- **Radiobutton AWGN**.- Cuando esta activo sirve para añadir ruido blanco gaussiano aditivo a la imagen y el valor de Sigma debe estar entre [0 y 1].

- **Radiobutton S & P Noise.**- Cuando esta activo sirve para añadir ruido sal y pimienta a la imagen y el valor de Probabilidad debe estar entre [0 y 1].

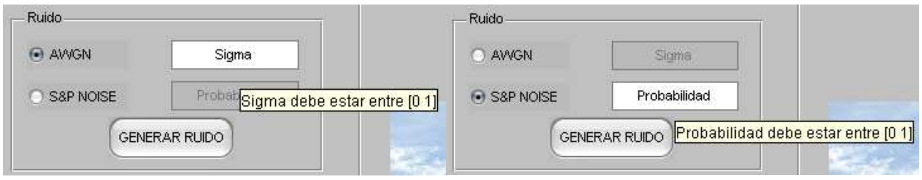

Figura A-4 Panel Ruido

PANEL DE DISTORSIÓN GEOMÉTRICA ESPACIAL

- **Radiobutton ROTACIÓN.**- Cuando esta activo sirve para añadir rotación a la imagen donde el valor de Teta debe estar entre [-pi y pi]

- **Radiobutton MOVIMIENTO VERTICAL.**- Cuando esta activo sirve para añadir rotación a la imagen donde el valor de Beta debe estar entre [-1 y 1]

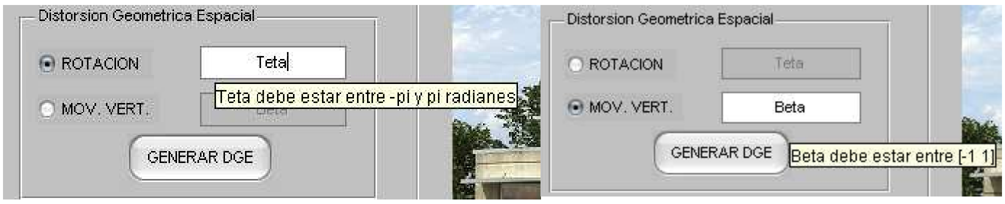

Figura A-5 Panel de Distorsión Geométrica Espacial

PANEL DE RESTABLECIMIENTO DE LA IMAGEN

- Se debe escoger al algoritmo a usar para eliminar el ruido de las cuatro siguientes opciones:
  - Median
  - Hard Wavelet Threshold
  - Hyperbolic Wavelet Threshold
  - Intelligent Median
- Si escoge los algoritmos de Mediana o Mediana Inteligente simplemente debe escribir el valor de N entre 2 y 20 o el valor de Umbral entre 0 y 1.

Figura A-6 Panel Restablecer Imagen usando mediana y mediana inteligente

- Mientras que si escoge los algoritmos de Hard Wavelet Threshold o Hyperbolic Wavelet Threshold deberá escribir el nivel L entre 2 y 100, y además seleccionar uno de los siguientes filtros wavelet:

  "db1, haar, db2, db10, db45, coif1, coif5, sym2, sym8, sym17, sym18, bior4.4, bior5.5 y bior6.8"

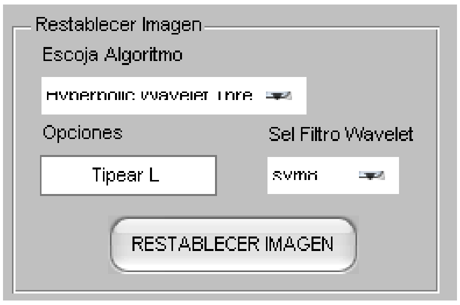

Figura A-7 Panel Restablecer Imagen usando hypebolic walet threshold

PANEL ERROR CUADRÁTICO MEDIO NORMALIZADO

Arrojará los errores cuadráticos medios normalizados de las imágenes:

- Sin ruido.
- Sin distorsión geométrica espacial.
- Sin ruido ni distorsión geométrica espacial, respectivamente.

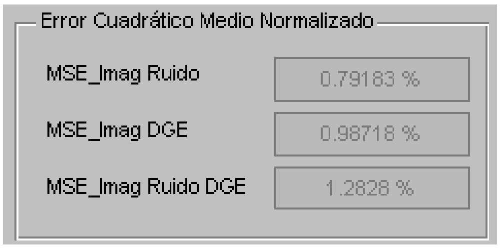

Figura A-8 Panel de cálculo del Error Cuadrático Medio normalizado

PANEL PROCESAMIENTO DIGITAL DE LA IMAGEN

Se van observando los cambios que va sufriendo la imagen a lo largo de un análisis y haciendo uso de los controles enumerados en los paneles antes nombrados.

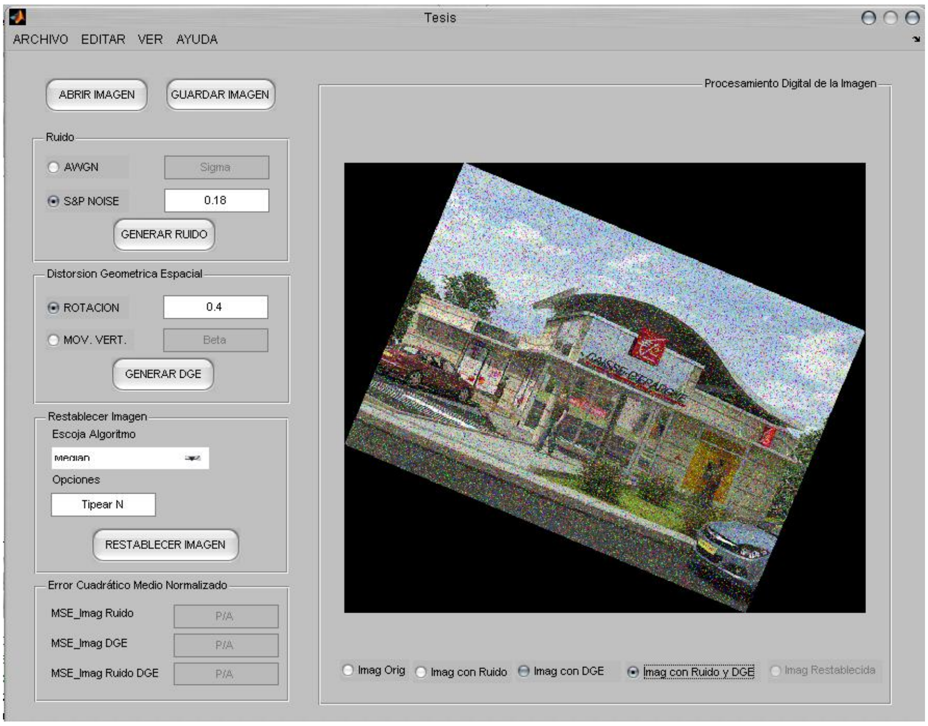

Figura A-9(a) Panel Procesamiento digital de la imagen

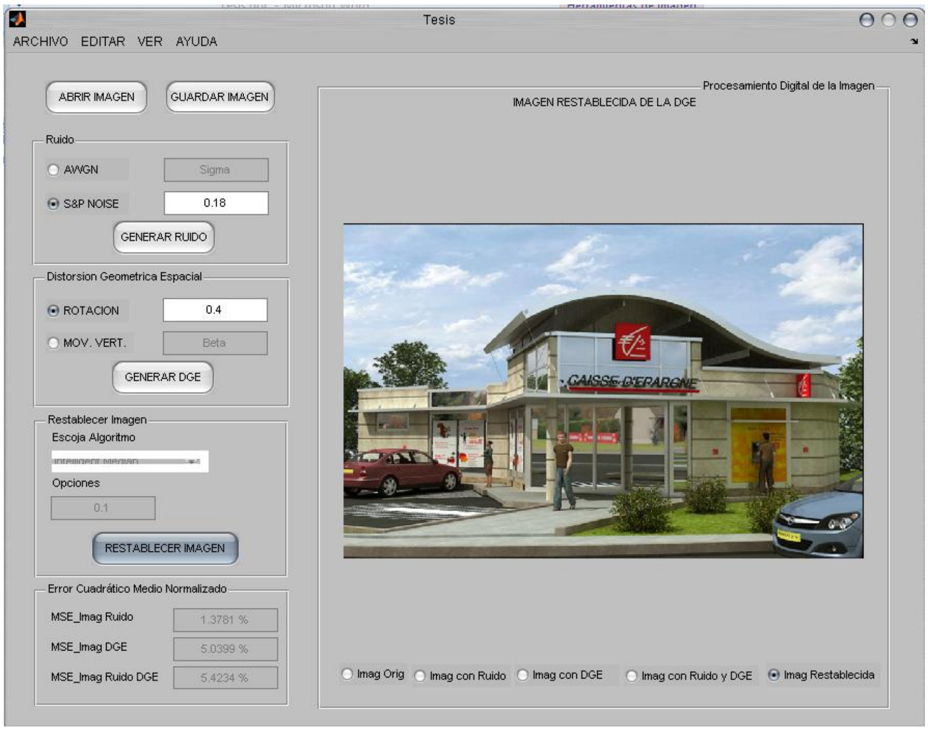

Figura A-9(b) Panel Procesamiento digital de la imagen

ANEXO B

Análisis Cuantitativo

RESTABLECIMIENTO DE IMAGENES USANDO ALGORITMO DE LA MEDIANA CON N=7

DGE\RUIDO — **MSE n**

FIGURA Nº	NOMBRE FIGURA	ROT.			MOV. VERT.			AWGN			S&P			
		bajo	medio	alto	bajo	medio	alto	bajo 0.1	medio 0.15	alto 0.2	bajo 0.1	medio 0.2	alto 0.3	
1	a.jpg	pi/10						1.12%	1.22%	1.38%	1.18%	1.47%	2.06%	0.29%
			pi/5					1.14%	1.24%	1.39%	1.19%	1.49%	2.08%	0.29%
				-pi/1,3				1.13%	1.23%	1.38%	1.19%	1.49%	2.18%	0.34%
					0.1			1.15%	1.26%	1.42%	1.16%	1.30%	1.56%	0.09%
						0.5		1.18%	1.28%	1.45%	1.24%	1.40%	1.72%	0.37%
							-0.9	1.16%	1.27%	1.42%	1.16%	1.30%	1.56%	0.10%
								1.19%	1.31%	1.51%	1.20%	1.38%	1.63%	
2	f.jpg	pi/10						0.81%	0.95%	1.16%	0.83%	1.23%	2.20%	0.50%
			pi/5					0.80%	0.94%	1.15%	0.82%	1.22%	2.15%	0.40%
				-pi/1,3				0.74%	0.88%	1.08%	0.76%	1.17%	2.06%	0.22%
					0.1			0.80%	0.94%	1.13%	0.75%	0.87%	1.13%	0.25%
						0.5		0.76%	0.89%	1.07%	0.69%	0.81%	0.98%	0.08%
							-0.9	0.85%	0.99%	1.18%	0.81%	0.93%	1.17%	0.48%
								0.78%	0.91%	1.12%	0.70%	0.84%	1.03%	
3	g.jpg	pi/10						1.54%	1.69%	1.87%	1.65%	2.12%	2.83%	0.73%
			pi/5					1.32%	1.50%	1.72%	1.44%	1.93%	2.63%	0.20%
				-pi/1,3				1.43%	1.59%	1.79%	1.57%	2.07%	2.83%	0.47%
					0.1			1.37%	1.58%	1.82%	1.35%	1.65%	2.04%	0.07%
						0.5		1.47%	1.67%	1.89%	1.54%	1.90%	2.42%	0.39%
							-0.9	1.38%	1.59%	1.83%	1.35%	1.67%	2.04%	0.09%
								1.42%	1.67%	1.95%	1.27%	1.62%	1.96%	
4	h.jpg	pi/10						1.36%	1.48%	1.67%	1.49%	2.04%	3.00%	0.58%
			pi/5					1.16%	1.31%	1.51%	1.26%	1.74%	2.47%	0.22%
				-pi/1,3				1.17%	1.32%	1.52%	1.27%	1.75%	2.47%	0.23%
					0.1			1.17%	1.34%	1.54%	1.16%	1.53%	2.08%	0.08%
						0.5		1.33%	1.48%	1.68%	1.47%	1.84%	2.47%	0.43%
							-0.9	1.17%	1.34%	1.54%	1.16%	1.53%	2.09%	0.08%
								1.17%	1.35%	1.58%	1.10%	1.70%	1.23%	
5	j.jpg	pi/10						1.16%	1.29%	1.47%	1.23%	1.61%	2.21%	0.57%
			pi/5					1.07%	1.21%	1.39%	1.14%	1.50%	2.07%	0.20%
				-pi/1,3				1.11%	1.24%	1.42%	1.18%	1.57%	2.22%	1.38%
					0.1			0.07%	1.27%	1.49%	1.03%	1.25%	1.60%	0.06%
						0.5		1.14%	1.31%	1.52%	1.14%	1.37%	1.77%	0.32%
							-0.9	1.11%	1.29%	1.50%	1.05%	1.26%	1.60%	0.07%
								1.14%	1.35%	1.61%	0.99%	1.30%	1.72%	
6	n.jpg	pi/10						0.70%	0.80%	0.93%	0.75%	1.03%	1.57%	0.19%
			pi/5					0.72%	0.82%	0.95%	0.77%	1.04%	1.59%	0.20%
				-pi/1,3				0.71%	0.80%	0.94%	0.75%	1.05%	1.70%	0.23%
					0.1			0.74%	0.86%	1.01%	0.71%	0.84%	1.08%	0.06%
						0.5		0.76%	0.87%	1.02%	0.78%	0.92%	1.20%	0.22%
							-0.9	0.75%	0.87%	1.02%	0.72%	0.85%	1.09%	0.07%
								0.77%	0.91%	1.11%	0.71%	0.94%	1.16%	

Tabla # B.1 Tabla del algoritmo de la mediana

RESTABLECIMIENTO DE IMAGENES USANDO ALGORITMO Hard Wavelet Threshold CON L=80

FILTRO WAVELET SELECCIONADO									bior 6.8			sym17			
		DGE\RUIDO						AWGN bajo	0.1						
								medio		0.15					
								alto			0.2				
								S&P bajo				0.1			
								medio					0.15		
								alto						0.2	
FIGURA Nº	NOMBRE FIGURA	ROT.			MOV. VERT.			MSE n							
		bajo	medio	alto	bajo	medio	alto								
1	a.jpg	-pi/10							0.69%	1.07%	1.57%	1.22%	1.57%	1.97%	0.26%
			-pi/2,1						0.83%	1.20%	1.72%	1.27%	1.61%	2.00%	0.45%
				-pi/1,4					0.67%	1.06%	1.56%	1.23%	1.57%	1.98%	0.24%
					-0.1				0.68%	1.06%	1.56%	1.65%	2.06%	2.51%	0.09%
						-0.5			0.81%	1.15%	1.61%	1.68%	2.08%	2.52%	0.37%
							-0.9		0.68%	1.07%	1.57%	1.67%	2.07%	2.52%	0.10%
									0.68%	1.20%	1.87%	2.34%	2.88%	3.43%	
2	f.jpg	-pi/10							0.87%	1.26%	1.79%	1.31%	1.77%	2.39%	0.53%
			-pi/2,1						0.98%	1.40%	1.99%	1.34%	1.82%	2.43%	0.62%
				-pi/1,4					0.73%	1.13%	1.67%	1.27%	1.76%	2.38%	0.32%
					-0.1				0.79%	1.19%	1.73%	1.78%	2.37%	3.16%	0.26%
						-0.5			0.72%	1.18%	1.79%	2.10%	2.63%	3.39%	0.08%
							-0.9		0.90%	1.12%	1.80%	1.81%	2.39%	3.18%	0.48%
									0.75%	1.35%	2.05%	3.04%	3.68%	4.94%	
3	r.jpg	-pi/10							1.28%	1.85%	2.62%	1.58%	1.76%	1.95%	0.77%
			-pi/2,1						0.98%	1.58%	2.37%	1.51%	1.68%	1.88%	0.42%
				-pi/1,4					0.88%	1.47%	2.23%	1.48%	1.68%	1.86%	0.30%
					-0.1				0.93%	1.70%	2.50%	1.52%	1.77%	1.94%	0.06%
						-0.5			1.09%	1.82%	2.58%	1.58%	1.82%	2.00%	0.32%
							-0.9		0.91%	1.67%	2.51%	1.49%	1.73%	1.93%	0.06%
									0.91%	1.62%	2.42%	1.48%	1.78%	2.02%	
4	h.jpg	-pi/10							1.11%	1.49%	1.98%	2.02%	2.39%	2.78%	0.72%
			-pi/2,1						0.81%	1.26%	1.82%	1.85%	2.20%	2.61%	0.35%
				-pi/1,4					0.70%	1.12%	1.63%	1.79%	2.17%	2.57%	0.23%
					-0.1				0.75%	1.28%	1.92%	1.88%	2.31%	2.73%	0.08%
						-0.5			1.02%	1.48%	2.05%	2.05%	2.44%	2.85%	0.43%
							-0.9		0.74%	1.28%	1.93%	1.92%	2.33%	2.76%	0.08%
									0.78%	1.37%	2.10%	1.87%	2.32%	2.76%	
5	z.jpg	-pi/10							0.61%	1.11%	1.76%	1.01%	1.46%	1.98%	0.14%
			-pi/2,1						0.67%	1.18%	1.85%	1.01%	1.45%	1.97%	0.22%
				-pi/1,4					0.60%	1.10%	1.76%	1.02%	1.47%	2.01%	0.13%
					-0.1				0.61%	1.08%	1.68%	1.71%	2.20%	2.83%	0.05%
						-0.5			0.68%	1.08%	1.61%	1.63%	2.15%	2.76%	0.21%
							-0.9		0.61%	1.08%	1.68%	1.70%	2.25%	2.82%	0.06%
									0.70%	1.30%	2.09%	2.79%	3.37%	4.44%	
6	n.jpg	-pi/10							0.56%	0.93%	1.39%	0.96%	1.27%	1.68%	0.17%
			-pi/2,1						0.64%	1.02%	1.49%	0.98%	1.29%	1.69%	0.28%
				-pi/1,4					0.56%	0.91%	1.36%	0.95%	1.27%	1.69%	0.16%
					-0.1				0.55%	0.88%	1.29%	1.40%	1.81%	2.29%	0.06%
						-0.5			0.60%	0.90%	1.26%	1.38%	1.78%	2.25%	0.22%
							-0.9		0.55%	0.89%	1.29%	1.40%	1.80%	2.29%	0.07%
									0.60%	1.03%	1.54%	2.31%	2.76%	3.35%	

Tabla # B.2 Tabla del algoritmo Hard Wavelet Threshold

RESTABLECIMIENTO DE IMAGENES USANDO ALGORITMO Hyperbolic Wavelet Threshold CON L=80

FILTRO WAVELET SELECCIONADO: db10

DGE\RUIDO — MSE n

FIGURA Nº	NOMBRE FIGURA	ROT. bajo	ROT. medio	ROT. alto	MOV. VERT. bajo	MOV. VERT. medio	MOV. VERT. alto	AWGN bajo 0.1	AWGN medio 0.15	AWGN alto 0.2	S&P bajo 0.1	S&P medio 0.15	S&P alto 0.2	
1	a.jpg	pi/30						0.80%	1.05%	1.38%	1.53%	1.95%	2.48%	0.52%
			pi/3					1.09%	1.29%	1.60%	1.75%	2.13%	2.62%	0.94%
				pi/1,1				1.06%	1.26%	1.56%	1.72%	2.10%	2.60%	0.89%
					0.1			0.64%	0.93%	1.24%	1.60%	1.95%	2.40%	0.09%
						0.4		0.89%	1.10%	1.38%	1.71%	0.03%	2.49%	0.62%
							0.9	0.63%	0.93%	1.25%	1.61%	1.95%	2.41%	0.10%
								0.62%	0.94%	1.32%	1.87%	2.18%	2.60%	
2	f.jpg	pi/30						0.86%	1.14%	1.56%	1.94%	2.59%	3.29%	0.63%
			pi/3					0.87%	1.11%	1.48%	1.83%	2.39%	3.00%	0.69%
				pi/1,1				0.67%	0.96%	1.38%	1.68%	2.31%	3.02%	0.35%
					0.1			0.68%	0.99%	1.41%	1.92%	2.46%	3.05%	0.25%
						0.4		0.77%	1.05%	1.46%	1.98%	2.49%	3.09%	0.48%
							0.9	0.77%	1.05%	1.46%	1.99%	2.50%	3.11%	0.48%
								0.62%	0.96%	1.40%	2.26%	2.67%	3.24%	
3	r.jpg	pi/30						1.20%	1.55%	2.00%	1.92%	2.36%	2.79%	0.90%
			pi/3					0.94%	1.30%	1.74%	1.64%	2.07%	2.54%	0.59%
				pi/1,1				0.71%	1.09%	1.49%	1.46%	1.87%	2.32%	0.30%
					0.1			0.71%	1.15%	1.59%	1.86%	2.28%	2.75%	0.06%
						0.4		0.83%	1.23%	1.66%	1.85%	2.25%	2.72%	0.39%
							0.9	0.70%	1.13%	1.58%	1.84%	2.24%	2.73%	0.06%
								0.71%	1.09%	1.43%	1.90%	2.17%	2.41%	
4	h.jpg	pi/30						0.73%	1.04%	1.42%	1.51%	2.04%	2.59%	0.34%
			pi/3					1.33%	1.55%	1.88%	1.96%	2.38%	2.91%	1.13%
				pi/1,1				1.23%	1.48%	1.81%	1.90%	2.32%	2.86%	1.01%
					0.1			0.69%	1.09%	1.53%	1.84%	2.38%	2.91%	0.08%
						0.4		0.80%	1.15%	1.58%	1.84%	2.36%	2.87%	0.30%
							0.9	0.71%	1.11%	1.56%	1.85%	2.40%	2.93%	0.08%
								0.71%	1.10%	1.56%	2.03%	2.52%	2.99%	
5	z.jpg	pi/30						0.59%	0.91%	1.33%	1.49%	2.08%	2.73%	0.27%
			pi/3					0.71%	1.01%	1.42%	1.57%	2.13%	2.78%	0.47%
				pi/1,1				0.68%	0.98%	1.39%	1.54%	2.11%	2.75%	0.42%
					0.1			0.50%	0.81%	1.16%	1.69%	2.11%	2.61%	0.05%
						0.4		0.67%	0.93%	1.27%	1.74%	2.10%	2.65%	0.39%
							0.9	0.50%	0.81%	1.17%	1.70%	2.11%	2.61%	0.06%
								0.50%	0.82%	1.20%	2.02%	2.32%	2.84%	
6	n.jpg	pi/30						0.58%	0.81%	1.10%	1.29%	1.69%	2.21%	0.32%
			pi/3					0.74%	0.95%	1.21%	1.41%	1.79%	2.27%	0.58%
				pi/1,1				0.73%	0.94%	1.20%	1.40%	1.77%	2.26%	0.55%
					0.1			0.50%	0.72%	0.99%	1.42%	1.70%	2.12%	0.06%
						0.4		0.65%	0.82%	1.07%	1.47%	1.73%	2.14%	0.42%
							0.9	0.50%	0.72%	0.99%	1.42%	1.70%	2.12%	0.07%
								0.51%	0.75%	1.02%	1.61%	1.91%	2.28%	

Tabla # B.3 Tabla del algoritmo Hyperbolic Wavelet Threshold

RESTABLECIMIENTO DE IMAGENES USANDO ALGORITMO DE LA MEDIANA INTELIGENTE CON UMBRAL=0.1

		DGE\RUIDO						AWGN bajo	AWGN medio	AWGN alto	S&P bajo	S&P medio	S&P alto	
								0.1	0.24	0.33	0.1	0.2	0.3	
FIGURA Nº	NOMBRE FIGURA	ROT. bajo	ROT. medio	ROT. alto	MOV. VERT. bajo	MOV. VERT. medio	MOV. VERT. alto	MSE n						
1	a.jpg	pi/90						1.26%	2.36%	3.13%	1.32%	2.14%	3.27%	0.72%
			pi/45					1.31%	2.40%	3.22%	1.39%	2.28%	3.59%	0.83%
				pi/10				1.02%	2.10%	2.87%	1.08%	1.79%	2.79%	0.29%
					0.1			1.26%	2.14%	2.81%	0.81%	1.14%	1.66%	0.09%
						0.5		1.35%	2.22%	2.96%	0.91%	1.17%	1.68%	0.37%
							0.8	1.29%	2.18%	2.83%	0.82%	1.14%	1.60%	0.15%
								1.35%	2.27%	2.93%	0.78%	0.92%	1.06%	
2	f.jpg	pi/90						1.18%	2.38%	3.31%	1.26%	2.08%	3.27%	0.70%
			pi/45					1.20%	2.46%	3.47%	1.27%	2.20%	3.65%	0.71%
				pi/10				1.05%	2.30%	3.30%	1.10%	1.98%	3.37%	0.50%
					0.1			1.18%	2.20%	3.01%	0.59%	0.81%	1.19%	0.25%
						0.5		1.12%	2.13%	2.87%	0.50%	0.68%	0.94%	0.08%
							0.8	1.12%	2.11%	2.86%	0.56%	0.86%	1.32%	0.12%
								1.20%	2.10%	2.76%	0.43%	0.52%	0.68%	
3	g.jpg	pi/90						1.20%	2.43%	3.28%	1.34%	2.17%	3.26%	0.51%
			pi/45					1.03%	2.26%	3.12%	1.16%	2.03%	3.10%	0.27%
				pi/10				1.38%	2.50%	3.36%	1.50%	2.29%	3.35%	0.73%
					0.1			1.22%	2.28%	3.13%	0.75%	1.23%	1.79%	0.07%
						0.5		1.42%	2.45%	3.27%	0.99%	1.43%	2.07%	0.39%
							0.8	1.26%	2.31%	3.13%	0.78%	1.27%	1.81%	0.13%
								1.29%	2.41%	3.20%	0.52%	0.75%	1.05%	
4	h.jpg	pi/90						1.48%	2.62%	3.43%	1.64%	2.62%	3.97%	0.93%
			pi/45					1.04%	2.22%	3.06%	1.19%	2.05%	3.19%	0.38%
				pi/10				1.19%	2.38%	3.21%	1.37%	2.37%	3.72%	0.58%
					0.1			1.12%	2.11%	2.85%	0.68%	1.11%	1.77%	0.08%
						0.5		1.36%	2.33%	3.11%	0.95%	1.39%	2.15%	0.43%
							0.8	1.15%	2.14%	2.88%	0.72%	1.12%	1.76%	0.12%
								1.15%	2.16%	2.90%	0.47%	0.70%	1.03%	
5	j.jpg	pi/90						1.12%	2.49%	3.28%	1.19%	2.03%	3.02%	0.44%
			pi/45					0.96%	2.35%	3.20%	1.02%	1.90%	2.88%	0.24%
				pi/10				1.23%	2.50%	3.29%	1.30%	2.12%	3.06%	0.57%
					0.1			1.20%	2.41%	3.26%	0.58%	1.03%	1.63%	0.65%
						0.5		1.33%	2.48%	3.29%	0.69%	1.04%	1.69%	0.32%
							0.8	1.23%	2.42%	3.25%	0.60%	1.04%	1.62%	0.12%
								1.28%	2.53%	3.50%	0.33%	0.50%	0.76%	
6	n.jpg	pi/90						0.99%	2.05%	2.75%	1.02%	1.81%	2.88%	0.44%
			pi/45					1.05%	2.11%	2.81%	0.06%	1.96%	3.20%	0.52%
				pi/10				0.84%	1.89%	2.58%	0.86%	1.54%	2.47%	0.19%
					0.1			1.06%	1.97%	2.58%	0.52%	0.84%	1.31%	0.06%
						0.5		1.12%	1.98%	2.64%	0.59%	0.80%	1.20%	0.22%
							0.8	1.09%	1.97%	2.59%	0.54%	0.82%	1.25%	0.10%
								1.12%	2.12%	2.82%	0.40%	0.50%	0.64%	

Tabla # B.4 Tabla del algoritmo de la mediana inteligente

ANEXO C

Encuesta

¿Cuál de los siguientes algoritmos restablece de mejor manera una imagen afectada por ruido y distorsión geométrica espacial?

a) Median(_)  Figura C-1 Imagen establecida con la mediana	b) Hard Wavelet Threshold(_)  Figura C-2 Imagen restablecida con Hard Wavelet Threshold
c) Hyperbolic Wavelet Threshold(_)  Figura C-3 Imagen restablecida con Hyperbolic Wavelet Threshold	d) Intelligent median(_)  Figura C-4 Imagen restablecida con la mediana inteligente

ANEXO D

Análisis Cualitativo

MEJOR ALGORITMO PARA RESTABLECER UNA IMAGEN			
Algoritmo	Nº personas	Frec. Reltiva	Frec. Reltiva Acumulada
Median	10	0.20	0.20
Hard Wavelet Threshold	8	0.18	0.38
Hyperbolic Wavelet Threshold	20	0.10	0.48
Intelligent Median	62	0.52	1.00

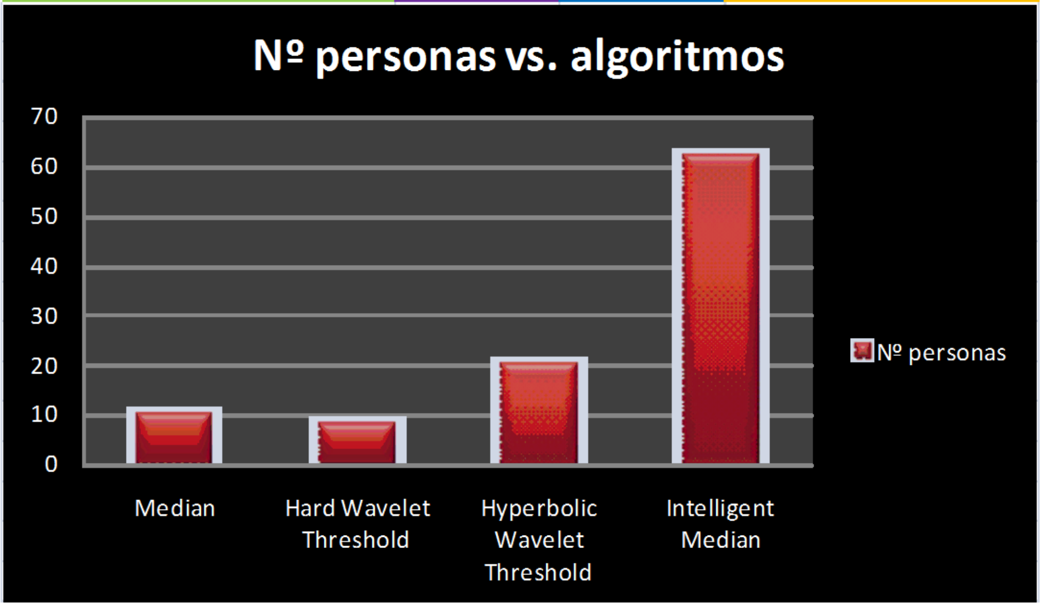

Tabla # D.1 Tabla cualitativa de los algoritmos

ANEXO E

Código del programa

```
function varargout = Tesis(varargin)
% TESIS M-file for Tesis.fig
%      TESIS, by itself, creates a new TESIS or raises the existing
%      singleton*.
%
%      H = TESIS returns the handle to a new TESIS or the handle to
%      the existing singleton*.
%
%      TESIS('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in TESIS.M with the given input arguments.
%
%      TESIS('Property','Value',...) creates a new TESIS or raises the
%      existing singleton*.  Starting from the left, property value pairs
%      are applied to the GUI before Tesis_OpeningFunction gets called.  An
%      unrecognized property name or invalid value makes property
%      stop.  All inputs are passed to Tesis_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
% edit6 the above text to modify the response to help Tesis
% Last Modified by GUIDE v2.5 13-Sep-2009 16:10:56
% Begin initialization code - DO NOT EDIT6
    gui_Singleton = 1;
    gui_State = struct('gui_Name',       mfilename, ...
                       'gui_Singleton',  gui_Singleton, ...
                       'gui_OpeningFcn', @Tesis_OpeningFcn, ...
                       'gui_OutputFcn',  @Tesis_OutputFcn, ...
                       'gui_LayoutFcn',  [] , ...
                       'gui_Callback',   []);
    if nargin && ischar(varargin{1})
        gui_State.gui_Callback = str2func(varargin{1});
    end

    if nargout
        [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
    else
        gui_mainfcn(gui_State, varargin{:});
    end
% End initialization code - DO NOT EDIT6
%% --- Executes just before Tesis is made visible.
function Tesis_OpeningFcn(hObject, eventdata, handles, varargin)
    % Choose default command line output for Tesis
    handles.output = hObject;

    % Update handles structure
    guidata(hObject, handles);

    % UIWAIT makes Tesis wait for user response (see UIRESUME)
    % uiwait(handles.figure1);
```

```
    axisoff;
    clc;
    format compact;

%% NO MOSTRAR LOS NUMEROS EN LOS EJES
function axisoff

%% BOTON PARA GENERAR RUIDO
function pushbutton3_Callback(hObject, eventdata, handles)
    global Imag_Orig;
    global Imag_Ruido Imag_DGE Imag_Ruido_DGE;
    global ruido dge;
    global tform_rotate tform_shear;

%% BOTON PARA GENERAR DISTORSION GEOMETRICA ESPACIAL(DGE)
function pushbutton4_Callback(hObject, eventdata, handles)
    global Imag_Orig;
    global Imag_DGE Imag_Ruido Imag_Ruido_DGE;
    global Teta Beta;
    global ruido dge;
    global tform_rotate tform_shear;

%% DESHACER DGE CON PROBLEMAS DE ROTACION O DE MOVIMIENTO VERTICAL
function Deshacer_DGE()
    global Imag_sin_DGE Teta Beta;

%% DESHACER DGE CON ROTACION
function [Imag_sin_Rot] = Deshacer_Rotacion()
    global Imag_Orig;
    global Teta;
    global Imag_actual;

%% DESHACER DGE CON MOVIMIENTO VERTICAL
function [Imag_sin_Mov_Vert] = Deshacer_Mov_Vert()
    global Imag_Orig;
    global Beta;
    global Imag_actual;

%% RESTABLECER LA IMAGEN DE LA DGE Y DEL RUIDO AWGN O S&P
function pushbutton5_Callback(hObject, eventdata, handles)
    global Imag_sin_Ruido Imag_sin_DGE Imag_restablecida  Imag_actual;
    global Teta Beta MSE_n str dr Imag_Ruido_DGE;

%% DESHACER RUIDO
function Deshacer_Ruido(handles)%validar esta funcion
    global Imag_Orig Imag_actual Imag_sin_Ruido dr

%% DETERMINACION DEL ERROR CUADRATICO MEDIO NORMALIZADO(MSE_n)
function Calcular_MSE_norm()
    global Imag_Orig Imag_restablecida MSE_n;

    [n1 n2 n3] = size(Imag_Orig);
    [l1 l2 l3] = size(Imag_restablecida);
    m1 = min(n1,l1); m2 = min(n2,l2);
    MSE_n = (mean(mean((((Imag_Orig(1:m1,1:m2)-
Imag_restablecida(1:m1,1:m2)).^2)/mean(mean(mean((Imag_Orig(1:m1,1:m2)).^2))
))))))*100;

%% ABRIR IMAGEN
```

```
function pushbutton1_Callback(hObject, eventdata, handles)
    [nomb dir]=uigetfile({'*.jpg';'*.jpeg';'*.bmp';'*.png';'*.tif'},'Cargar
Imagen a Procesar');
    if nomb == 0
        return
    end

    clear global Imag_Orig Imag_sin_Ruido Imag_sin_DGE Imag_restablecida;
    clear global Imag_Ruido Imag_DGE Imag_Ruido_DGE Imag_actual;
    clear global ruido dge var_rgb MSE_n str Beta Teta Sigma Probabilidad;
    clear global tform_rotate tform_shear dr;

    global Imag_Orig Imag_Ruido Imag_Ruido_DGE;
    global var_rgb;
    global ruido dge;

%% MUESTRA LA IMAGEN QUE SE ENCUENTRE COMO ACTUAL
function Mostrar_Imagen_Actual(img,a,b,c,d,e,handles)
    global Imag_actual;

    Imag_actual = img;
    set(handles.radiobutton5,'Value',a);
    set(handles.radiobutton6,'Value',b);
    set(handles.radiobutton7,'Value',c);
    set(handles.radiobutton8,'Value',d);
    set(handles.radiobutton9,'Value',e);
    imshow(Imag_actual);

%% PARA GRABAR LA IMAGEN QUE SE HALLE COMO ACTUAL
function pushbutton2_Callback(hObject, eventdata, handles)
    global Imag_actual;


[nomb_s,dir_s]=uiputfile({'*.jpg';'*.jpeg';'*.bmp';'*.png';'*.tif'},'Guardar
Imagen Actual');
    if nomb_s == 0
        return
    end
    imwrite(Imag_actual,[dir_s,nomb_s]);

%% SELECCIONADO EL RADIOBUTTON AWGN
function radiobutton1_Callback(hObject, eventdata, handles)
    set(handles.radiobutton1,'Value',1);
    set(handles.radiobutton2,'Value',0);
    set(handles.edit1,'Enable','on');
    set(handles.edit2,'Enable','off');
    set(handles.edit2,'String','Probabilidad');

%% SELECCIONADO EL RADIOBUTTON S&P NOISE
function radiobutton2_Callback(hObject, eventdata, handles)
    set(handles.radiobutton1,'Value',0);
    set(handles.radiobutton2,'Value',1);
    set(handles.edit1,'Enable','off');
    set(handles.edit2,'Enable','on');
    set(handles.edit1,'String','Sigma');

%% SELECCIONADO EL RADIOBUTTON ROTACION
function radiobutton3_Callback(hObject, eventdata, handles)
    set(handles.radiobutton3,'Value',1);
    set(handles.radiobutton4,'Value',0);
```

```
    set(handles.edit3,'Enable','on');
    set(handles.edit4,'Enable','off');
    set(handles.edit4,'String','Beta');

%% SELECCIONADO EL RADIOBUTTON DESP. VERT.
function radiobutton4_Callback(hObject, eventdata, handles)
    set(handles.radiobutton3,'Value',0);
    set(handles.radiobutton4,'Value',1);
    set(handles.edit3,'Enable','off');
    set(handles.edit4,'Enable','on');
    set(handles.edit3,'String','Teta');

%% SELECCIONANDO EL ALGORITMO PARA ELIMINAR EL RUIDO
function popupmenu1_Callback(hObject, eventdata, handles)

    pum1=get(handles.popupmenu1,'Value');%pum1=popupmenu

    set(handles.edit5,'Visible','on');
    set(handles.text3,'Visible','off');
    set(handles.popupmenu2,'Visible','off');

%% MENU ARCHIVO->ABRIR IMAGEN
function Untitled_5_Callback(hObject, eventdata, handles)
    pushbutton1_Callback(hObject, eventdata, handles)

%% MENU ARCHIVO->GUARDAR IMAGEN
function Untitled_6_Callback(hObject, eventdata, handles)
    pushbutton2_Callback(hObject, eventdata, handles)


%% MENU ARCHIVO->SALIR DE LA HERRAMIENTA
function Untitled_7_Callback(hObject, eventdata, handles)
    clear all;
    close all;

%% MUESTRA LA IMAGEN ORIGINAL
function radiobutton5_Callback(hObject, eventdata, handles)
    global Imag_Orig;
    set(handles.text7,'String','');
    set(handles.popupmenu1,'Enable','off');
    set(handles.popupmenu2,'Enable','off');
    set(handles.edit5,'Enable','off');

    Mostrar_Imagen_Actual(Imag_Orig,1,0,0,0,0,handles);

%% MUESTRA LA IMAGEN CON RUIDO
function radiobutton6_Callback(hObject, eventdata, handles)
    global Imag_Ruido;
    set(handles.text7,'String','');
    set(handles.popupmenu1,'Enable','on');
    set(handles.popupmenu2,'Enable','on');
    set(handles.edit5,'Enable','on');
    Mostrar_Imagen_Actual(Imag_Ruido,0,1,0,0,0,handles);

%% MUESTRA LA IMAGEN CON DISTORSION GEOMETRICA ESPACIAL(DGE)
function radiobutton7_Callback(hObject, eventdata, handles)
    global Imag_DGE;
    set(handles.text7,'String','');
    set(handles.popupmenu1,'Enable','off');
    set(handles.popupmenu2,'Enable','off');
```

```
    set(handles.edit5,'Enable','off');

    Mostrar_Imagen_Actual(Imag_DGE,0,0,1,0,0,handles);

%% MUESTRA LA IMAGEN CON RUIDO Y DISTORSION GEOMETRICA ESPACIAL(DGE)
function radiobutton8_Callback(hObject, eventdata, handles)
    global Imag_Ruido_DGE;
    set(handles.text7,'String','');
    set(handles.popupmenu1,'Enable','on');
    set(handles.popupmenu2,'Enable','on');
    set(handles.edit5,'Enable','on');

    Mostrar_Imagen_Actual(Imag_Ruido_DGE,0,0,0,1,0,handles);

%%  MUESTRA LA IMAGEN RESTABLECIDA
function radiobutton9_Callback(hObject, eventdata, handles)
    global Imag_restablecida str;

    set(handles.text7,'String',str);
    set(handles.popupmenu1,'Enable','off');
    set(handles.popupmenu2,'Enable','off');
    set(handles.edit5,'Enable','off');

    Mostrar_Imagen_Actual(Imag_restablecida,0,0,0,0,1,handles);

%% MENU AYUDA->AYUDA DE LA HERRAMIENTA
function Untitled_8_Callback(hObject, eventdata, handles)
    helpdlg(manual,'AYUDA PARA EL USUARIO');

%% MENU AYUDA->EJEMPLO
function Untitled_9_Callback(hObject, eventdata, handles)
    winopen('ejemplo.pdf');

 %%FUNCION QUE REALIZA EL ALGORITMO DE HARD WAVELET THRESHOLD
function x_est=hard_shrink(y,L,wfilt)

%%FUNCION QUE REALIZA EL ALGORITMO HYPERBOLIC WAVELET THRESHOLD
function  x_est=hyp_shrink(y,L,wfilt)

%%FUNCION QUE ELIMINA EL RUIDO SAL Y PIMIENTA
function  x_filt=salt_pepper_remove(x_nx,threshold,iteration)


%%FUNCION QUE VERIFICA EL UMBRAL
function A=thresholder(B,thr,k)

%%FUNCION QUE REALIZA EL ALGORITMO DE LA MEDIANA INTELIGENTE
function x_filt=wmedfilt2(x,window_size,weights)
```

BIBLIOGRAFÍA

DOCUMENTOS

[1] MADISETTI Vijay K., WILLIAMS Douglas B., “Digital Signal Processing Handbook”, Chapman & Hall/CRCnetBase, 1999.

[2] VASEGHI Sabed V. “Advanced Digital Signal Processing and Noise Reduction”, John Wiley & Sons Ltda., Second Edition, 2000.

[3] MATLAB, “Image Processing Toolbox User’s Guide”, Mathworks Inc., 2007.

BASE DE DATOS:

[4] Imágenes renderizadas encontradas en internet:

[4.1] http://proyectoblogspace.files.wordpress.c8casa-cipres_01.jpg

[4.2] http://api.ning.com/files/MjUxsRX/CASA5.jpg

[4.3] http://img201.imageshack.us/i/arqgaliciacasabalancan0eb4.jpg/

[4.4] http://img82.imageshack.us/i/c68.jpg/

[4.5] http://www.infografia-3d./Images/country_casas_renderizados.jpg

[4.6] http://api.ning.com/files/MjUxsRX/CASA6.jpg

[4.7] http://api.ning.com/files/MjUxsRX/CASA7.jpg

[4.8] http://api.ning.com/files/MjUxsRX/CASA8.jpg

PROGRAMA GUI EN MATLAB:

[5] AYAZOGLU Mustafa., “Image Denoising Tool Northeastern University”, 2007.