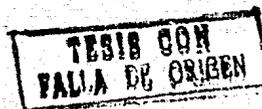


ÍNDICE

TEMA	PÁGINA
INTRODUCCIÓN	3
CAPÍTULO UNO. Procesamiento digital de imágenes.	5
I.1) Conceptos fundamentales.	5
I.2) Digitalización y compresión de imágenes.	7
I.2.1) Digitalización.	7
I.2.2) Compresión.	7
I.3) Realce y restauración de imágenes.	8
I.3.1) Realce de imágenes.	8
I.3.1.a) Suavizamiento espacial de regiones.	9
I.3.1.b) Reescalamiento de los niveles de gris.	11
I.3.1.c) Realce de bordes.	14
I.3.1.d) Filtros en el dominio de la frecuencia.	17
I.3.2) Restauración de imágenes.	19
I.4) Segmentación y descripción de imágenes.	20
I.4.1) Segmentación.	20
I.4.2) Descripción.	23
I.5) Recomendaciones sobre la literatura especializada.	24
CAPÍTULO DOS. Elementos de un sistema para PDI.	26
II.1) Digitalización.	26
II.2) Procesamiento.	29
II.3) Despliegue.	30
II.4) Sugerencias para la integración de un sistema de PDI.	33
CAPÍTULO TRES. Diseño y construcción de un paquete para PDI.	35
III.1) Selección de algoritmos.	36
III.1.a) Obtención de contornos.	39
III.1.a.1) El gradiente.	40
III.1.a.2) El laplaciano.	43
III.1.b) Realce general y de bordes.	44





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 está 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.

III.1.c) Restauración.	45
III.1.d) Suavizamiento general.	46
III.1.e) Sumas, restas y otras operaciones simples.	46
III.2) Rutinas de despliegue.	46
III.2.1) Despliegue en monitores de alta resolución.	47
III.2.2) Impresión de imágenes.	48
III.3) Programación de algoritmos.	50
III.3.1) Lenguajes de programación.	51
III.3.2) Memoria de proceso y de almacenamiento.	51
III.3.3) Estructuras de datos.	52
III.3.4) Construcción de una biblioteca de funciones.	53
III.3.5) Rutinas para despliegue e impresión.	54
III.3.6) Una alternativa en el dominio de la frecuencia.	55
CAPÍTULO CUATRO. Aplicaciones en Geofísica.	59
III.1) Procesos en imágenes de percepción remota.	62
CONCLUSIONES.	69
BIBLIOGRAFÍA.	71
APÉNDICE A.	73

INTRODUCCIÓN

El interés por las técnicas de procesamiento digital de imágenes (PDI) data de la década de los 20's. Sin embargo, su empleo no encontró amplia difusión sino hasta mediados de la década de los 60's, cuando la tercera generación de computadoras proporcionó las velocidades de proceso y la capacidad de almacenamiento requeridas para la implantación práctica de los algoritmos de procesamiento digital.

Desde entonces, esta área ha experimentado un crecimiento espectacular, convirtiéndose en una materia de interés para muchos campos del conocimiento.

En nuestro país, las ideas, métodos y resultados del PDI han sido empleados por diversos grupos de investigación en áreas como Física, Medicina, Neurociencias, Arqueología, Astronomía, Química, Biología, Geofísica, etc. Aunque, en contraste con lo que sucede en otros países, pocas veces ha sido aprovechado este conocimiento fuera de los centros de investigación.

En particular, en el campo de estudio de la Geofísica, las técnicas de PDI han sido decisivas en el avance alcanzado por una de sus ramas: la teledetección, mejor conocida como "percepción remota".

La percepción remota basada en el procesamiento digital de imágenes es hoy en día una poderosa herramienta para el estudio de los recursos terrestres. Su propósito esencial es la obtención de información acerca de la superficie terrestre y sus recursos midiendo las variaciones espectrales, espaciales y temporales de la energía reflejada y emitida que proviene de una cierta zona.

La intención principal de este trabajo fue construir un paquete para procesamiento digital que puede ser aplicado para el análisis de imágenes de percepción remota.

En las páginas de este texto se describen las etapas por las cuales hubo que pasar durante la construcción del paquete, proporcionando además algunas especificaciones para el diseño de un sistema completo para procesamiento digital de imágenes. Los conceptos se exponen en suficiente detalle para que puedan ser apreciadas las limitaciones y las tendencias actuales de este

campo.

La exposición se ha organizado en cuatro capítulos. Los primeros dos capítulos proporcionan los fundamentos requeridos para la construcción de un paquete de procesamiento digital de imágenes. El tercer capítulo se ocupa propiamente del diseño del paquete. En el capítulo cuatro, se muestran algunas aplicaciones de las técnicas de procesamiento digital en imágenes de percepción remota.

A continuación se describe brevemente el contenido de cada capítulo.

En el primer capítulo, llamado "procesamiento digital de imágenes", se presenta un compendio de las técnicas de PDI que, a nuestro juicio, resultan fundamentales y cuyo campo de aplicación no se limita a problemas especializados.

En el segundo capítulo, "elementos de un sistema para procesamiento digital de imágenes", se lleva a cabo un recuento del equipo (hardware) y los programas (software) necesarios para procesar imágenes. Para la descripción general del sistema de procesamiento se toma como modelo al que actualmente funciona en el laboratorio de digitalización de imágenes del Instituto de Física de la UNAM.

En el capítulo tres, "diseño y construcción de un paquete para procesamiento digital de imágenes (PDI)", se discute el problema de seleccionar los algoritmos para conformar un sistema de PDI. La parte restante del capítulo se dedica a la implantación práctica de los algoritmos elegidos en los sistemas de cómputo más comunes hoy en día, esto es, las computadoras personales.

En el último capítulo, "aplicaciones en Geofísica", se presentan imágenes que ilustran algunos de los procesos que se aplican en percepción remota. Las imágenes se acompañan de una explicación en la que se hace referencia a la técnica empleada al procesarlas, sus características y los resultados que produce su aplicación.

CAPÍTULO I

PROCESAMIENTO DIGITAL DE IMÁGENES (PDI).

El procesamiento digital de imágenes se refiere a aquella parte del procesamiento digital de señales en la cual la señal es una imagen. Sus áreas de interés incluyen [1]:

- (a) Digitalización y compresión: Convertir imágenes a la forma digital; codificación eficiente de la imagen para ahorrar espacio de almacenamiento.
- (b) Realce, y restauración : Recobrar o reconstruir imágenes que han sido degradadas.
- (c) Segmentación y descripción : Segmentar imágenes en partes, midiendo sus propiedades y las relaciones entre las partes, comparando las descripciones resultantes con modelos que definen clases de imágenes.

Las secciones posteriores de este capítulo se dedican a describir las áreas del procesamiento digital de imágenes y a discutir las técnicas más representativas de cada una de ellas. Las descripciones presentan un detalle que varía de acuerdo con la importancia que representan para el análisis de imágenes de percepción remota.

Antes de empezar a tratar cada una de las áreas del procesamiento digital de imágenes, se presentan algunos conceptos básicos.

I.1) Conceptos fundamentales.

Aunque nuestra asociación inherente a las sensaciones visuales nos predispone a concebir una imagen como un estímulo en la retina del ojo, la tecnología actual nos proporciona muchos medios diferentes para capturar imágenes. Los diversos mecanismos con que es posible crear y registrar imágenes quedan abarcados en un cuerpo matemático compacto.

Con el término imagen se denota a una función bidimensional de intensidades luminosas $f(x,y)$, donde x y y son las coordenadas espaciales. El valor de f en el punto (x,y) es proporcional a la

intensidad luminosa en ese punto de la imagen.

Una imagen digital es una imagen $f(x,y)$ que ha sido discretizada tanto en las coordenadas espaciales como en la intensidad luminosa. Podemos considerar a una imagen digital como una matriz cuyos índices de renglones y columnas identifican un punto en la imagen y el correspondiente valor del elemento de la matriz representa la intensidad luminosa en ese punto [2].

La intensidad de una imagen digital monocromática en un punto es llamada "nivel de gris" de dicho punto.

Una imagen a color puede ser representada como tres matrices, una para cada color primario (rojo, verde y azul). Debido a ello, en adelante trataremos exclusivamente el caso en que la imagen es monocromática.

Una imagen continua $f(x,y)$ se aproxima mediante un conjunto de muestras igualmente espaciadas en la forma de un arreglo de dimensión $M \times N$, es decir:

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

El lado derecho de la ecuación representa lo que llamamos una imagen digital y cada elemento del arreglo se denomina "pixel" o "pel" (abreviatura de "picture element").

Para realizar el proceso de digitalización debe elegirse un valor para M y N , así como para el número de niveles discretos de gris permitidos para cada pixel. Es muy conveniente que los valores asignados sean potencias de dos, pues esto facilita el procesamiento posterior de la imagen.

I.2) Digitalización y compresión de imágenes.

En esta sección se explican los procesos involucrados en la digitalización y compresión de imágenes. Una exposición detallada sobre la teoría en que se basan estos procesos sale del alcance de este trabajo. Para un estudio más profundo de estos temas se proponen algunas fuentes de información al final de cada sección.

I.2.1) Digitalización.

Para que una imagen pueda ser procesada con una computadora debe ser previamente discretizada.

La digitalización consiste en muestrear los niveles de gris de la imagen en un arreglo de $M \times N$ puntos. Dado que el nivel de gris de estos puntos puede tomar cualquier valor en un rango continuo, para procesarlo por computadora requiere ser cuantizado. Esto significa que se debe dividir el rango de niveles de gris en k intervalos, y el nivel de gris de cualquier punto sólo puede tomar uno de estos valores. La imagen resulta una reproducción más fiel conforme M , N y k son más grandes.

En otras palabras, el objetivo del muestreo es representar una imagen continua mediante un arreglo de números, llamados muestras. La única restricción sobre el número de muestras es que debe permitir reconstruir la imagen a partir de él.

El muestreo de una imagen debe cumplir con los mismos requisitos que impone la teoría del muestreo de señales en una dimensión.

Una exposición completa sobre la teoría del muestreo de señales, en particular de imágenes, puede encontrarse en la referencia [1].

I.2.2) Compresión.

Una imagen digital requiere de un gran número de bits para su almacenamiento. En muchas aplicaciones, es necesario considerar técnicas que permitan representar una imagen, o la información

contenida en ella, con un número sustancialmente menor de bits. A este proceso se le llama "codificación" de la imagen.

Las técnicas de compresión generalmente están orientadas a un tipo específico de imagen. En otras palabras, la elección de la técnica de compresión para un tipo determinado de imagen depende de las características de la imagen.

En la compresión de imágenes es preciso emplear técnicas de codificación que permitan reconstruir la imagen (por medio de un decodificador) con el menor error posible.

Aunque muy pocos grupos de investigación se dedican al estudio de las técnicas de compresión, existe una teoría abundante sobre ellas. Un tratamiento detallado de esta teoría puede encontrarse en las referencias [1] y [2].

I.3) *Realce y restauración de imágenes.*

Por su importancia en el análisis de imágenes de percepción remota, comenzaremos tratando las técnicas de realce con cierto detalle. Posteriormente hablaremos de los procesos de restauración en forma genérica.

En la parte relativa al realce de imágenes se recomiendan un gran número de técnicas diferentes, dada su sencillez computacional. Por otro lado, en la parte de restauración se presenta una revisión general y se propone al final una técnica bastante simple y flexible.

I.3.1) *Realce de imágenes.*

El realce de imágenes está diseñado para manipular la imagen sobre la base de las características psicofísicas del sistema visual humano. Puede inclusive llegar a distorsionar la imagen al atenuar los objetos irrelevantes y realzar los rasgos de interés [2].

La mayoría de las técnicas de realce existentes son heurísticas y orientadas a un problema específico. Cada técnica está diseñada para tratar un tipo especial de imagen. No se

requiere, en general, modelar el proceso de degradación; sin embargo, un conocimiento del fenómeno de degradación puede ayudar en la selección de las técnicas de realce.

La degradación puede subdividirse en tres clases: degradación puntual, degradación espacial y su combinación. La degradación puntual no afecta el cuadro general de la imagen, pues distorsiona los niveles de gris de cada pixel. La degradación espacial, al operar sobre regiones completas, hace más difusa la imagen, de modo que ésta pierde resolución y la transición de los objetos hacia el fondo se vuelve poco clara [3].

Dado que la mayoría de las técnicas de realce están orientadas a un problema específico, el éxito en la aplicación de una técnica a una imagen particular depende del juicio subjetivo del observador. No existe ningún criterio de evaluación cuantitativo con carácter universal.

De acuerdo a los métodos técnicos que involucran, las técnicas de realce pueden organizarse en cuatro grupos [3]. Estos son:

- (a) Suavizamiento espacial de regiones, que emplea filtros pasa bajas lineales o no lineales en el dominio espacial;
- (b) Reescalamiento de los niveles de gris, que manipula o récuantiza los niveles de gris para el realce por contraste;
- (c) Realce de bordes, que involucra filtros pasa altas lineales o no lineales en el dominio espacial;
- (d) Filtros en el dominio de la frecuencia, que utiliza filtros pasa altas o pasa bajas en el dominio de la frecuencia.

A continuación se presentan las técnicas de realce más conocidas, agrupadas siguiendo esta clasificación.

1.3.1.a) Suavizamiento espacial de regiones.

Este tipo de técnica se utiliza cuando la imagen ha sido degradada con ruido aleatorio. De esta forma, el suavizamiento de una región tiende a remover el ruido en cada pixel.

Con su empleo, el almacenamiento de datos requerido es

relativamente pequeño.

La técnica de suavizamiento más sencilla es el promedio aritmético de una vecindad. Se describe en la forma siguiente:

$$g'(x,y) = \sum_{i=-m}^m \sum_{j=-n}^n w(i,j) g(x-i,y-j) \quad (1)$$

donde los pesos $w(i,j)$ son iguales y

$$w(i,j) = \frac{1}{(2m+1)(2n+1)} \quad (2)$$

En la ecuación (1), el nivel de gris de (x,y) es reemplazado por el promedio de los niveles de gris en una vecindad rectangular de $(2m+1)$ por $(2n+1)$ alrededor de (x,y) . Este simple esquema puede remover eficientemente el ruido; sin embargo, hace más difusa la imagen especialmente en los bordes de los objetos que la componen. La difusión se torna más severa conforme m y n se incrementan.

Para reducir el efecto de difusión, han sido introducidas muchas técnicas de suavizamiento con pesos diferentes. Estas técnicas dan un mayor peso al nivel de gris en (x,y) que a sus vecinos. Graham [4] propuso una vecindad de 3×3 y la matriz de pesos

$$W = \begin{bmatrix} 0.25 & 0.5 & 0.25 \\ 0.5 & 1 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{bmatrix} \quad (3)$$

Brown [5] propuso, a su vez, la siguiente matriz

$$W = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4)$$

La matriz (2) tiende a disminuir severamente la definición de los bordes. Los procedimientos (3) y (4) no remueven tan eficientemente el ruido como (2), pero, en cambio, alteran los bordes en menor extensión.

Para mejorar los resultados obtenidos con los esquemas anteriores, se han propuesto otras técnicas sensibles al contexto de la imagen. Estas técnicas emplean operadores que varían con la

posición y los parámetros cambian en respuesta a las características locales de la imagen. Sin embargo, estas técnicas resultan mucho más complicadas computacionalmente y, debido a ello, su empleo es poco frecuente.

Otra técnica bastante popular es el llamado filtro de la mediana, propuesto inicialmente por Tukey [6], y adoptado posteriormente por otros. La operación consiste en reemplazar $g(x,y)$ por la mediana de los niveles de gris de una vecindad de $(2m + 1)$ por $(2n + 1)$ alrededor de (x,y) . Esta técnica de suavizamiento actúa satisfactoriamente en funciones escalón y funciones rampa. Sin embargo, las funciones impulso, cuyos periodos son menores que n (es decir, la mitad del ancho de la ventana), son suprimidas. De este modo, en algunos casos el filtro de la mediana puede causar la supresión de una parte de la señal. En un intento por remediar este problema, Pratt [7] sugiere el empleo de anchuras de ventana variables. Inicia la operación fijando $n=1$, y repite la operación incrementando n . El proceso termina cuando el filtro empieza a causar daño a la imagen. Este filtro de la mediana mejorado es muy consumidor de tiempo y en ocasiones requiere diferentes valores de n para diferentes regiones de una imagen.

1.3.1.b) Reescalamiento de los niveles de gris.

Las técnicas que pertenecen a esta categoría frecuentemente se apoyan en la información proveniente del histograma de la imagen. El histograma es una gráfica en la que se presentan los distintos niveles que conforman la escala de grises, con su respectiva frecuencia en la imagen (número de píxeles que poseen ese nivel).

El reescalamiento de los niveles de gris consiste en recuantizar o mapear cada píxel a un nuevo nivel de gris para mejorar el contraste de una imagen. En general, el nuevo valor que asume cada píxel depende solamente de su valor original y es independiente del de sus vecinos. El objetivo es el realce de objetos y la mayoría de las operaciones son globales.

En muchas imágenes, los niveles de gris de los objetos están

tan próximos a los del fondo que es difícil distinguirlos. En estos casos es preciso un realce del contraste para incrementar las diferencias en los niveles de gris entre los objetos y el fondo. En otras situaciones, los niveles de gris de un gran porcentaje de los píxeles se concentran en una estrecha porción del histograma. El reescalamiento resulta entonces una buena técnica para incrementar el rango dinámico y para destacar las estructuras finas.

La técnica más simple de esta categoría es un mapeo funcional de los niveles de gris, es decir,

$$g'(x,y) = f(g(x,y)) \quad (5)$$

donde f es una función preespecificada. Algunas de las funciones de mapeo más populares se ilustran en la figura 1. En la figura 1(a), el área oscura es extendida y el área clara es comprimida. De este modo, el contraste en la parte oscura se incrementa. En la figura 1(b), el operador realiza lo opuesto a 1(a), mientras que en 1(c) expande los niveles de gris de un rango intermedio.

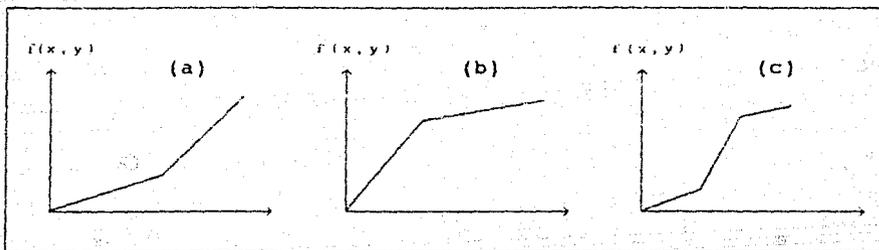


Figura 1. Funciones de Reescalamiento: (a) área oscura extendida; (b) área clara extendida; (c) rango intermedio extendido.

Estos mapeos son muy fáciles de implementar y usualmente reportan resultados satisfactorios. Sin embargo, para obtener resultados mejores es necesario depurarlos mediante ensayo y error. En muchos casos, uno debe diseñar sus propias funciones de mapeo.

El histograma de la mayoría de las imágenes frecuentemente

muestra picos claros u oscuros. Para realizar la información contenida en los picos existe una técnica llamada "linealización del histograma". Esta técnica consiste en mapear los niveles de gris observados en nuevos niveles de gris, en forma tal que la nueva imagen tenga un histograma uniforme. Consideremos la curva 1 de la figura 2(b), que representa la función de distribución acumulativa (FDA) de una imagen con niveles de gris continuos. La curva 2 es la FDA lineal deseada. El procedimiento para linealizar la FDA observada (curva 1), consiste en los siguientes pasos: (a) selección de intervalos iguales en las ordenadas, como se muestra en la figura 2(b); (b) a partir de la curva 1, mapear los intervalos en la ordenada a los niveles de gris en la abscisa; (c) recuantizar la imagen observada. Este procedimiento puede formularse en la forma siguiente:

$$g'(x,y) = [g_{\max} - g_{\min}] P(g(x,y)) + g_{\min} \quad (6)$$

donde $P(g(x,y))$ es la FDA, y g_{\max} , g_{\min} son los niveles de gris máximo y mínimo en la imagen observada, respectivamente.

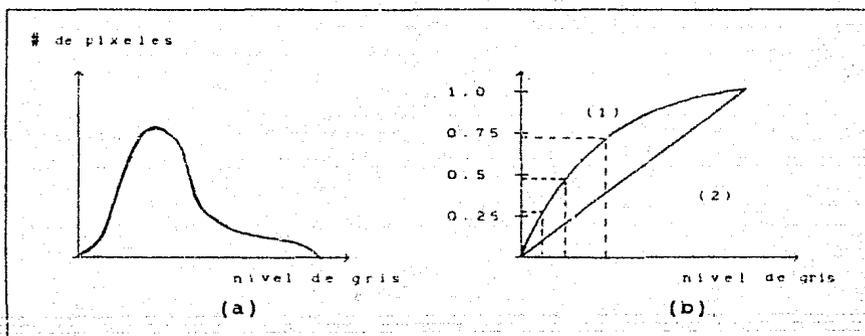


Figura 2. (a) Histograma típico de una imagen.
 (b) La curva 1 es la FDA de (a).
 La curva 2 es la FDA lineal deseada.

Se sabe que la percepción humana efectúa una transformación no lineal de la intensidad luminosa. Es por ello que la hiperbolización es preferida sobre la linealización del histograma de la imagen. Las transformaciones empleadas para obtener un

histograma de forma hiperbólica son las siguientes:

$$g'(x,y) = g_{\min} + \frac{1}{\alpha} \ln [1 - P(g(x,y))] \quad (7)$$

$$g'(x,y) = [(g_{\max}^{1/3} - g_{\min}^{1/3}) P(g(x,y)) + g_{\min}^{1/3}]^3 \quad (8)$$

$$g'(x,y) = g_{\min} \left[\frac{g_{\max}}{g_{\min}} \right]^{P(g(x,y))} \quad (9)$$

donde α es una constante.

Las transformaciones (7), (8) y (9) producen histogramas de tipo exponencial, de raíz cúbica y logarítmico respectivamente. La imagen resultante tendrá un histograma constante, lo cual facilita la percepción visual.

Para adelgazar los picos en el histograma, existe una técnica denominada "adelgazamiento iterativo del histograma". Esta técnica produce resultados opuestos a los de la linealización del histograma, a continuación la describimos brevemente.

Sean B_i el número de píxeles en la barra i del histograma. Este esquema examina cada barra i y sus 2γ barras vecinas a ambos lados. Si B_i es mayor que el promedio A de $B_{i-1}, \dots, B_{i+\gamma}$, se calcula un radio $\beta = (B_i - A) / B_i$. Entonces, los niveles de gris de $B_{i+j}\beta$ píxeles son cambiados de la barra $i+j$ a la barra $i+j-1$, $j=1, 2, \dots, \gamma$. De este modo, B_i se incrementa y sus barras vecinas decrecen. En esta forma, la técnica produce imágenes con fondo y objetos homogéneos con contraste realzado; sin embargo, puede producir error de cuantización [3].

Otra técnica comúnmente empleada es la resta de imágenes. Se utiliza para atenuar el material irrelevante y contrastar el material de interés.

I.3.1.c) Realce de bordes.

Esta técnica se emplea para incrementar la diferencia entre los niveles de gris de los píxeles del contorno del objeto y del fondo adyacente.

Las causas de la degradación en imágenes son principalmente

la difusión, el estar fuera de foco y el movimiento del objeto. El perfil de los niveles de gris en un borde ideal es una función escalón. La difusión y la salida de foco convierten el borde en una función rampa. El nivel de gris en el borde de un objeto es disminuido y el del fondo adyacente se ve incrementado. De esta forma, el contraste se reduce o inclusive desaparece. Para disminuir este tipo de degradación, se ha propuesto un operador antidifusión. Para eliminar el efecto del movimiento del objeto, puede aplicarse la restauración de imágenes.

Un operador antidifusión ataca la reducción del nivel de gris del contorno del objeto y el incremento del nivel de gris del fondo. El incremento en el nivel de gris del contorno realizará la imagen.

La operación antidifusión puede formularse en la siguiente forma:

$$g'(x,y) = g(x,y) - \gamma \nabla^2 g(x,y) \tag{10}$$

donde γ es una constante. Esta técnica puede proporcionar una visión más clara de los contornos, pero también enfatiza el ruido.

La enfatización de fluctuaciones accidentales es uno de los mayores problemas que presentan las técnicas de realce de bordes. Una solución posible a este problema puede ser ofrecida por el realce de bordes a lo largo de la dirección del gradiente local. Esta técnica realiza el borde según

$$g'(x,y) = g(x,y) - \gamma \frac{\partial^2 g(x,y)}{\partial n^2} \tag{11}$$

donde

$$\frac{\partial^2 g(x,y)}{\partial n^2} = \frac{\frac{\partial^2 g}{\partial x} \left[\frac{\partial g}{\partial x} \right]^2 - 2 \frac{\partial^2 g}{\partial x \partial y} \frac{\partial g}{\partial x} \frac{\partial g}{\partial y} + \frac{\partial^2 g}{\partial y^2} \left[\frac{\partial g}{\partial y} \right]^2}{\left[\frac{\partial g}{\partial x} \right]^2 + \left[\frac{\partial g}{\partial y} \right]^2}$$

Se ha propuesto una mejora a esta técnica, que puede ser formulada en la forma siguiente:

$$g'(x,y) = g(x,y) - \gamma \left[\frac{\partial^2 g(x,y)}{\partial n} - \frac{1}{3} \frac{\partial^2 g(x,y)}{\partial t^2} \right] \quad (12)$$

donde

$$\frac{\partial^2 g(x,y)}{\partial t^2} = \frac{\frac{\partial^2 g}{\partial x} \left[\frac{\partial g}{\partial y} \right]^2 - 2 \frac{\partial^2 g}{\partial x \partial y} \frac{\partial g}{\partial x} \frac{\partial g}{\partial y} + \frac{\partial^2 g}{\partial y^2} \left[\frac{\partial g}{\partial x} \right]^2}{\left[\frac{\partial g}{\partial x} \right]^2 + \left[\frac{\partial g}{\partial y} \right]^2}$$

que es la derivada de segundo orden a lo largo de la tangente del contorno. El cálculo resulta complicado, sin embargo, realiza las cuestas y valles y suprime los cambios accidentales en forma eficiente.

Un filtro que emplea estadística local fue propuesta por Wallis [8]. Esta técnica tiende a realizar los detalles finos en la imagen. El algoritmo es

$$g'(x,y) = \bar{g}(x,y) + k (g(x,y) - \bar{g}(x,y)) \quad (13)$$

donde $\bar{g}(x,y)$ es la media local de los niveles de gris que circundan al pixel (x,y) . Puede notarse que si en la ecuación anterior $k > 1$, la diferencia entre la media local y el nivel de gris en (x,y) es magnificada, y el borde será agudizado como si pasara a través de un filtro pasa altas. Cuando $k = 2$, la ecuación (13) es equivalente a (10). Si $0 < k < 1$, la diferencia se reduce, y la imagen será suavizada como si pasara por un filtro pasa bajas. Para $k = 0$, la operación es un simple suavizado. Los experimentos demuestran que cuando k es grande, todos los detalles finos, incluyendo el ruido, son realizados.

Otra técnica que realiza los bordes y limpia el ruido consiste en efectuar la operación antidifusión de la ecuación (10) si $|\nabla^2 g(x,y)|$ es menor que un parámetro de corte preespecificado T . O bien, suavizar el pixel siguiendo la ecuación (1) si $|\nabla^2 g(x,y)|$ es mayor que T .

Otra operación antidifusión consiste en hacer un barrido electrónico simultáneo con dos sensores, uno fino y otro ordinario, y restar las imágenes restantes electrónicamente. El algoritmo de esta técnica queda descrito por la ecuación (13).

I.3.1.d) Filtros en el dominio de la frecuencia.

Una imagen puede ser representada en forma única en el dominio de la frecuencia. La transformación del dominio espacial al dominio frecuencial puede ser realizada a través del algoritmo de la transformada rápida de Fourier.

Se ha observado que las componentes de Fourier de baja frecuencia corresponden a los objetos y el fondo homogéneos, y las altas frecuencias son introducidas debido a la ocurrencia de un borde agudo o de ruido. En esta forma, los filtros pasa altas o pasa bajas pueden ser empleados para realizar los bordes o suavizar la imagen. El diseño de los filtros ha sido un problema ampliamente tratado en el procesamiento digital de señales.

A continuación se describen los filtros de mayor aplicación en el realce de imágenes.

Duda y Hart [9] proponen el siguiente filtro pasa bajas

$$H(f_x, f_y) = [(\cos \pi f_x)(\cos \pi f_y)]^\alpha, \quad \alpha > 1 \quad (14)$$

y como un filtro pasa altas

$$H(f_x, f_y) = 1 - [(\cos \pi f_x)(\cos \pi f_y)]^\alpha, \quad \alpha \geq 1 \quad (15)$$

Un filtro pasa bajas remueve el ruido, sin embargo, degrada la imagen. Por otro lado, un filtro pasa altas agudiza los bordes, pero realza el ruido y hace ásperos los objetos. Se ha propuesto un filtro de compromiso para enfatizar las altas frecuencias, se le denomina "HFE". Duda y Hart proponen el siguiente filtro HFE:

$$H(f_x, f_y) = 2 - [(\cos \pi f_x)(\cos \pi f_y)]^\alpha, \quad \alpha \geq 1 \quad (16)$$

González y Wintz [2] sugieren un filtro de Butterworth. Un filtro de Butterworth pasa bajas, de orden n , es el siguiente

$$H(f_x, f_y) = \frac{1}{1 + [(f_x^2 + f_y^2)^{1/2} / f_0]^2} \quad (17)$$

un filtro pasa altas

$$H(f_x, f_y) = \frac{1}{1 + [f_0 / (f_x^2 + f_y^2)^{1/2}]^2} \quad (18)$$

y un filtro HFE

$$H(f_x, f_y) = 1 + \frac{1}{1 + \{f_0 / (f_x^2 + f_y^2)^{1/2}\}^2} \quad (19)$$

Hall [10] propone un filtro exponencial para realizar imágenes radiográficas. Su filtro exponencial pasa bajas está dado por

$$H(f_x, f_y) = \exp [-(f_x^2 + f_y^2)^{1/2} / f_0]^n \quad (20)$$

su filtro pasa altas está dado por

$$H(f_x, f_y) = \exp [-f_0 / (f_x^2 + f_y^2)^{1/2}]^n \quad (21)$$

y su filtro HFE es

$$H(f_x, f_y) = 1 + \exp [-f_0 / (f_x^2 + f_y^2)^{1/2}]^n \quad (22)$$

La mayor parte de la energía de una imagen se encuentra concentrada en la porción de las bajas frecuencias. Por ello, un filtro logarítmico ha sido propuesto. Está definido por

$$G'(f_x, f_y) = (\log |G(f_x, f_y)|) \exp (j \phi(f_x, f_y)) \quad (23)$$

De esta forma, la porción de las altas frecuencias es enfatizada en mayor proporción que la porción de las bajas frecuencias.

Un filtro similar, llamado proceso α , fue sugerido por Andrews [11]. La operación es

$$G'(f_x, f_y) = |G(f_x, f_y)|^\alpha \exp (j \phi(f_x, f_y)) \quad 0 \leq \alpha \leq 2 \quad (24)$$

Para α pequeña, especialmente $\alpha = 0$, resulta un filtro pasa altas.

La mayoría de estas técnicas producen excelentes resultados. Sin embargo, a menos que se tenga disponible un procesador especial para la FFT, el cómputo es laborioso y requiere de un enorme espacio de almacenamiento. Debe notarse, además, que estas técnicas requieren la especificación de algunas constantes; las corridas de ensayo las hacen aún más consumidoras de tiempo.

1.3.2) Restauración de imágenes.

La restauración es un proceso que tiene como objetivo reconstruir o recobrar una imagen que ha sido degradada, partiendo de un cierto conocimiento del fenómeno de degradación con que ésta fue afectada [2].

Las técnicas de restauración se ocupan de modelar los procesos de degradación y de aplicar el proceso inverso con objeto de recobrar la imagen original. Esto generalmente implica la formulación de un criterio de aceptabilidad que debe conducir a la estimación óptima del resultado deseado.

Las primeras técnicas de restauración se derivaron en su mayoría de conceptos en el dominio de la frecuencia. Sin embargo, en la actualidad existen métodos algebraicos más potentes que poseen la ventaja de permitir la derivación de numerosas técnicas de restauración, partiendo de los mismos principios básicos. Aunque una solución directa por estos métodos algebraicos involucra generalmente la manipulación de grandes sistemas de ecuaciones simultáneas, bajo ciertas condiciones, es posible reducir la complejidad computacional al mismo nivel que requieren las técnicas tradicionales en el dominio de la frecuencia.

El desarrollo matemático de estas técnicas resulta complicado y muy extenso. Por ello, nos limitaremos a recomendar las referencias [1], [2] y [11] como excelentes fuentes de consulta sobre estos métodos.

Un procedimiento simple de restauración, que resulta conveniente y rápido en una variedad de contextos, es el denominado método de sustituciones sucesivas [12].

Este método modela el proceso de degradación mediante un operador a . El diseño de este operador requiere de un cierto conocimiento previo del tipo de degradación que afecta a la imagen. Dos operadores especialmente útiles para revertir procesos de difusión en imágenes son aquéllos modelados a través de ventanas de 2×2 y 3×3 pixeles, con pesos de $1/4$ y $1/9$ para cada pixel, respectivamente.

La imagen degradada g , se relaciona con la imagen original f , que debe encontrarse, mediante una convolución bidimensional (representada por $**$) con el operador a , de tal modo que

$$g = a ** f \quad (25)$$

y entonces la primera etapa de restauración estará dada por

$$f_1 = 2g - a ** g \quad (26)$$

La segunda etapa se obtiene al sustituir f_1 por g en el término del lado derecho, es decir

$$f_n = 2g - a ** f_{n-1} \quad (27)$$

De esta forma se consigue un procedimiento iterativo que, bajo ciertas condiciones, converge rápidamente a la solución deseada.

I.4) Segmentación y descripción de imágenes.

La segmentación y descripción de imágenes son etapas esenciales en la mayoría de los problemas de reconocimiento de patrones y análisis automático de escenas. La selección de una técnica adecuada depende de las características peculiares del problema que se esté considerando. El objetivo de las técnicas de segmentación es dividir una imagen dada en regiones o componentes. Posteriormente se emplean los procedimientos de descripción regional para caracterizar individualmente las regiones segmentadas. Existen además procesos de descripción relacional que permiten organizar estas regiones en una estructura significativa. La última etapa consiste en establecer medidas de similitud entre las regiones de una imagen o entre conjuntos de imágenes diferentes [2].

I.4.1) Segmentación.

Los procedimientos para segmentar pueden agruparse en dos categorías. En la primera se ubican los métodos que se basan en un examen de la imagen pixel por pixel. La segunda categoría emplea la información contenida en una vecindad de la imagen. En

ambos casos podemos ver el problema como un proceso de reconocimiento de patrones cuyo objetivo es establecer fronteras entre regiones.

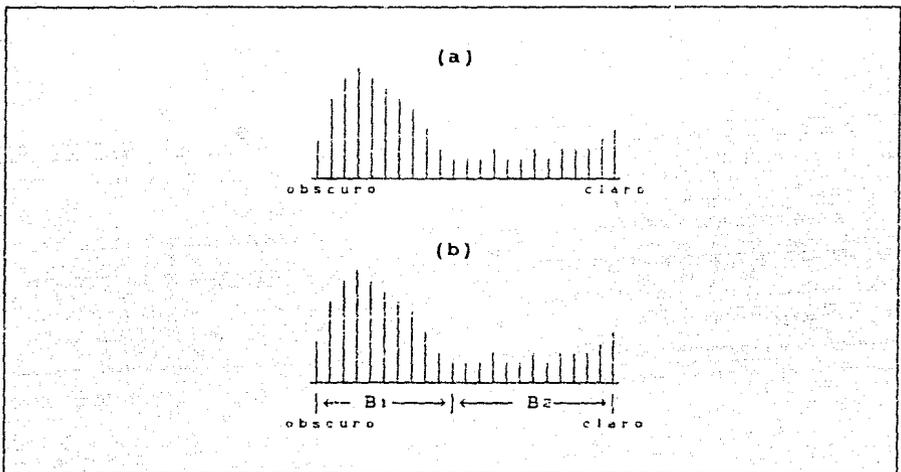


Figura 3. Corte de histograma

Un procedimiento simple, perteneciente a la primera categoría, consiste en dividir la escala de gris en bandas y emplear los umbrales para determinar regiones o para obtener puntos frontera.

Supongamos que los niveles de gris en una imagen $f(x,y)$ tienen el histograma que se muestra en la figura 3(a). Para resaltar la frontera entre los objetos y el fondo, se divide el histograma en dos bandas separadas mediante un umbral T , como se muestra en la figura 3(b). El objetivo es seleccionar T en tal forma que B_1 contenga niveles asociados con el fondo, mientras que B_2 contenga los niveles de los objetos. Cualquier cambio de nivel de una banda a la otra denota la presencia de una frontera. Una vez que se selecciona B_1 y B_2 , el procedimiento es el siguiente:

Para cada renglón de la imagen crear un renglón

correspondiente en una imagen intermedia g_1 , empleando la siguiente relación:

$$g_1(x,y) = \begin{cases} L_B & \text{Si los niveles de } f(x,y) \text{ y } f(x,y-1) \\ & \text{se encuentran en bandas diferentes.} \\ L_F & \text{En cualquier otro caso.} \end{cases}$$

donde L_B y L_F son los niveles del borde y del fondo especificados, respectivamente.

Para cada columna se crea una columna correspondiente en una imagen intermedia g_2 siguiendo la misma relación.

La imagen deseada se forma empleando la siguiente relación:

$$g(x,y) = \begin{cases} L_B & \text{si } g_1(x,y) \text{ o } g_2(x,y) \text{ son iguales a} \\ & L_E. \\ L_F & \text{En cualquier otro caso.} \end{cases}$$

El procedimiento anterior se puede generalizar fácilmente para considerar un mayor número de bandas en el histograma. Con ello se incrementa el poder de esta técnica de segmentación.

Una técnica perteneciente a la segunda categoría emplea una ventana para segmentar la imagen en regiones. El procedimiento es el siguiente:

Consideremos la ventana de la figura 4. El centro de la ventana se mueve de pixel en pixel por toda la imagen. En cada posición, se multiplica cada punto de la imagen que está dentro de la ventana por el número indicado en la entrada correspondiente, y posteriormente se suman los resultados. Si todos los puntos de la imagen que se encuentran dentro de la ventana tienen el mismo valor (en el caso del fondo), la suma será cero. Si, por otro lado, el centro de la ventana se localiza en un punto aislado, la suma será diferente de cero. Si el punto se encuentra en la ventana, pero no en el centro, la suma será también diferente de cero, pero la magnitud de la respuesta será menor que si el punto estuviera en el centro. Empleando un umbral, estas respuestas menores pueden ser eliminadas.

-1	-1	-1
-1	8	-1
-1	-1	-1

Figura 4. Ventana puntual.

1.4.2) Descripción.

Una vez que ha sido identificada una región de una imagen, resulta de interés caracterizar la imagen mediante un conjunto de descriptores que no son sensibles a cambios en tamaño, rotaciones, traslaciones, etc. Generalmente se emplean descriptores regionales pues facilitan la diferenciación de regiones con diferentes atributos.

Una técnica muy empleada es la de los denominados "descriptores de Fourier" [2]. Esta técnica emplea la transformada discreta de Fourier.

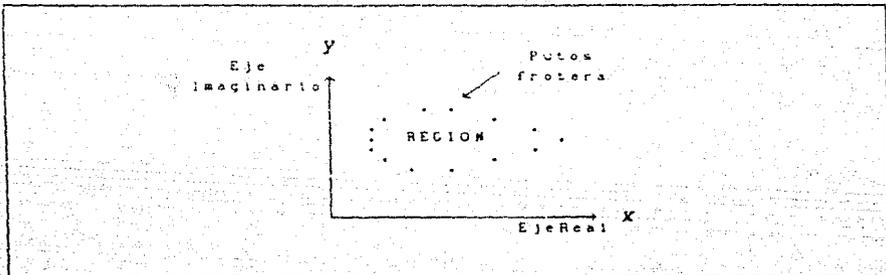


Figura 5. Representación de un contorno en el plano complejo.

Supongamos que tenemos identificados M puntos pertenecientes a la frontera de una región. Podemos visualizar la región como si estuviera en el plano complejo, con la ordenada como el eje imaginario y la abscisa como el eje real, como se observa en la figura 3. Las coordenadas x - y de cada punto en el contorno se convierten en números complejos, $x + iy$. Comenzando en un punto arbitrario del contorno y trazando a partir de éste los demás puntos, obtenemos una secuencia de números complejos. La transformada discreta de Fourier de esta secuencia se le conoce como el descriptor de Fourier del contorno.

Es posible encontrar una revisión mucho más completa de estas técnicas en las referencias [1] y [2].

1.5) Recomendaciones sobre la literatura especializada.

Entre los principales libros sobre procesamiento digital de imágenes, considerados inclusive como clásicos en la materia, encontramos los siguientes:

- 1.- H. C. Andrews, "Computer Techniques in Image Processing." Academic Press, New York, 1970.
- 2.- K. R. Castleman, "Digital Image Processing." Prentice Hall, Englewood Cliffs, New Jersey, 1979.
- 3.- R. O. Duda and P. E. Hart, "Pattern Classification and Scene Analysis." Wiley, New York, 1973.
- 4.- R. C. Gonzalez and P. Wintz, "Digital Image Processing." Addison-Wesley, Massachusetts, 1977.
- 5.- E. L. Hall, "Computer Image Processing and Recognition." Academic Press, New York, 1979.
- 6.- T. Pavlidis, "Structural Pattern Recognition." Springer Publ., New York, 1977.
- 7.- A. Rosenfeld, "Picture Processing by Computer." Academic Press, New York, 1969.
- 8.- W. K. Pratt, "Digital Image Processing." Wiley, New York, 1978.
- 9.- A. Rosenfeld y A. C. Kak, "Digital Picture Processing." Academic Press, New York, 1976.

En cualquiera de estos textos es posible encontrar una revisión completa de las técnicas de procesamiento digital de imágenes y la teoría en la cual se fundamentan.

Además, cada año se publican más de mil artículos sobre procesamiento digital de imágenes y sus aplicaciones en revistas especializadas. Algunas de las revistas que frecuentemente publican trabajos sobre este tema son:

- 1.- IEEE Transactions on Acoustics, Speech and Signal Processing.
- 2.- IEEE Transactions on Communications.
- 3.- IEEE Transactions on Computers.
- 4.- IEEE Transactions on Information Theory.
- 5.- IEEE Transactions on Pattern Analysis and Machine Intelligence.
- 6.- IEEE Transactions on Systems, Man, and Cybernetics.
- 7.- Computer Vision, Graphics and Image Processing.
- 8.- Pattern Recognition.

CAPÍTULO II

ELEMENTOS DE UN SISTEMA PARA PROCESAMIENTO DIGITAL DE IMÁGENES.

Un sistema de procesamiento de imágenes, dependiendo de la aplicación particular a la cual esté orientado, puede contar con un cierto tipo especial de *hardware* y *software*.

Sin embargo, la operación de cualquier sistema para PDI se puede dividir en tres categorías principales: digitalización, procesamiento y despliegue [2].

Los elementos de un sistema para procesar imágenes los podemos apreciar en la figura 6.

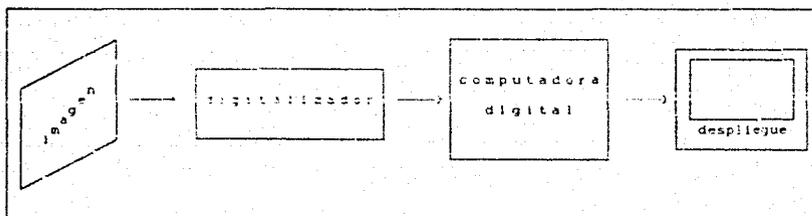


Figura 6. Elementos de un sistema para procesamiento digital de imágenes.

II.1) Digitalización.

Como se indica en el capítulo I, el proceso de digitalización consiste en convertir una imagen en una representación numérica adecuada para proporcionarla como entrada a una computadora digital. Para ello es necesario un digitalizador de imágenes.

Entre los digitalizadores más comúnmente utilizados están los microdensitómetros, los barreadores (scanners), los disectores de imágenes y los digitalizadores con cámara de TV. Los primeros dos dispositivos requieren que la imagen de entrada se encuentre en forma de transparencia o fotografía. Los disectores y los

digitalizadores con cámara de TV pueden aceptar imágenes en esa forma, pero poseen la ventaja adicional de permitir digitalizar imágenes en vivo.

En los microdensitómetros el barrido de la imagen se acompaña de su iluminación con un haz de luz y su traslación o rotación con respecto al haz. En el caso de transparencias el haz pasa a través de la película; en el caso de fotografías se refleja en la superficie de la imagen. En ambos casos el haz se enfoca en un fotodetector y el nivel de gris en cualquier punto de la imagen es registrado por el detector con base en la intensidad del haz. La imagen digital se obtiene al permitir solamente valores discretos en la intensidad y la posición de cada punto. Aunque los microdensitómetros son dispositivos lentos, proporcionan altos grados de exactitud en la posición debido a la naturaleza esencialmente continua de la traslación mecánica que emplean en el proceso de digitalización.

Un barredor también opera bajo el principio de enfocar un haz (transmitido o reflejado) sobre un fotodetector. En este caso, sin embargo, la imagen es estacionaria y la fuente luminosa es un tubo de rayos catódicos. El haz que se consigue produce un punto luminoso que se mueve, siguiendo un patrón de barrido, sobre la superficie del tubo. El hecho de que el haz sea movido electrónicamente permite alcanzar grandes velocidades de barrido. Los barredores son dispositivos ideales en aplicaciones en las que sea deseable controlar el patrón de barrido del haz externamente.

En los disectores y digitalizadores con cámara de TV la imagen es enfocada directamente sobre la superficie de un tubo fotosensible cuya respuesta es proporcional al patrón del rayo incidente. La operación del disector se basa en el principio de la emisión electrónica, donde la imagen incidente en la superficie fotosensible produce un haz de electrones cuya sección transversal tiene aproximadamente la misma geometría de la superficie del tubo. La obtención de la imagen se realiza con el empleo de electromagnetos que dirigen el haz hacia un orificio localizado en la parte trasera del tubo disector. El orificio permite únicamente el paso de una pequeña porción del haz y, de esta forma, apunta hacia un punto de la imagen en cada instante. Dado que los materiales fotoemisivos son muy ineficientes, el tiempo que el

orificio tiene que apuntar al punto fuente con objeto de coleccionar suficientes electrones tiende a convertir a los disectores en digitalizadores bastante lentos. La mayoría de los dispositivos integran la emisión de cada punto de entrada sobre un intervalo de tiempo especificado antes de producir una señal que sea proporcional a la intensidad del punto. Esta capacidad de integración es benéfica en términos de la reducción de ruido, haciendo así atractivos a los disectores de imágenes en aplicaciones donde son requeridas altas relaciones señal-ruido.

Muchos digitalizadores con cámara de TV emplean un tubo de vidicon, cuya operación se basa en el principio de la fotoconductividad. Una imagen enfocada en la superficie del tubo produce un patrón de conductividades variantes que registra la distribución de intensidades en la imagen óptica. Un haz independiente de electrones, finamente enfocado, barre la superficie del blanco fotoconductor, y por neutralización de cargas, dicho haz genera una diferencia de potencial y produce en un colector una señal proporcional al patrón de intensidades de la imagen. Al cuantizar esta señal y la posición correspondiente del haz, se obtiene la imagen digital [2].

Aunque los digitalizadores estándar de vidicon son, en general, mucho menos precisos que los otros sistemas, poseen numerosas ventajas que en muchas aplicaciones compensan esta relativa falta de precisión. Por ejemplo, los sistemas de vidicon se encuentran entre los sistemas más baratos en el mercado. Cuentan además con la ventaja de que las imágenes digitalizadas con ellos pueden ser observadas en un monitor de TV. Dado que los sistemas de vidicon emplean un barrido electrónico y los tubos fotoconductivos son razonablemente eficientes, estos dispositivos digitalizadores son mucho más rápidos que los microdensitómetros y los disectores de imágenes [2].

Los dispositivos digitalizadores de imágenes, en general, son controlados vía software. El software que utilizan depende de las características particulares del equipo. Es por ello que cuando se adquiere un cierto tipo de digitalizador, también se obtiene el software para controlar los procesos de digitalización y almacenamiento de las imágenes bajo un cierto formato.

De esta forma, una persona interesada en conformar un sistema

de procesamiento de imágenes no se ocupa de la elaboración del software para digitalización, salvo el caso en que tenga en mente construir el dispositivo digitalizador.

II.2) Procesamiento.

Los sistemas que se emplean para procesar imágenes varían desde microprocesadores para aplicaciones especiales hasta grandes computadoras capaces de ejecutar una gran variedad de operaciones sobre arreglos de imágenes de alta resolución.

La estructura del sistema computacional depende básicamente del tipo de procesos que deseen realizarse sobre las imágenes. Por ejemplo, una minicomputadora resulta adecuada en el caso de un laboratorio para aplicaciones de propósito general.

Dado que las imágenes discretas resultan demasiado grandes para ser soportadas por la memoria de una computadora pequeña, la clave al estructurar un sistema de este tipo es contar con recursos adecuados y eficientes para almacenamiento masivo. Los medios de almacenamiento más populares son las cintas y discos magnéticos. Generalmente, las cintas se emplean para archivar las imágenes y los discos se emplean durante el procesamiento con objeto de agilizar la velocidad de transferencia de los datos [2].

En términos de los requerimientos anteriores, un sistema de cómputo para procesar imágenes puede ser formado empleando una minicomputadora de 32,000 a 64,000 palabras de memoria central, dos unidades para discos, una unidad para cinta magnética y algunos periféricos tales como terminales e impresoras [2].

Los programas para un sistema de procesamiento de imágenes generalmente se codifican en lenguaje ensamblador, con objeto de ganar en velocidad de ejecución. Sin embargo, para hacer más flexible el sistema, el desarrollo de nuevos programas debe hacerse empleando al menos un lenguaje de alto nivel.

Un sistema con las componentes descritas anteriormente puede emplearse sin dificultad para desarrollar la mayoría de las técnicas de procesamiento discutidas en el primer capítulo. La eficiencia en su operación depende del tamaño de las imágenes y del tipo de procesos que se pretenda aplicarles.

Con el desarrollo alcanzado por las computadoras personales, muchos de las técnicas de procesamiento de imágenes pueden ser implantadas en ellas. Esto resulta de particular interés en nuestro país, pues los costos de las minicomputadoras representan un serio obstáculo para muchas personas interesadas en aplicar, en su campo de estudio, las técnicas de procesamiento digital de imágenes.

Existen algunos paquetes para procesamiento de imágenes que operan en computadoras personales. El costo de este tipo de paquetes es relativamente elevado. Además, su empleo para una aplicación particular, como en el caso del análisis de imágenes de percepción remota, presenta algunos inconvenientes.

Como se vió en el primer capítulo, la selección de la técnica para procesar un cierto tipo de imagen depende, sobre todo, de las características propias de dicho tipo de imagen. Es por ello que, cuando se utiliza un conjunto de programas diseñados para propósitos generales, es necesario incorporar nuevos programas o realizar modificaciones a los ya existentes para poder aplicarlos adecuadamente a una problemática particular.

En muchas ocasiones, el diseño propio del software se convierte, más que en una alternativa posible, en una necesidad.

En el capítulo tres se trata el problema del diseño de software para procesar imágenes, tomando en cuenta recursos de cómputo limitados a una computadora personal de la generación actual.

II.3) *Despliegue.*

La función de la unidad de despliegue en un sistema de procesamiento de imágenes es convertir los arreglos numéricos almacenados en la memoria de la computadora en una forma adecuada para la interpretación humana [2].

Los principales instrumentos de despliegue son los tubos de rayos catódicos (CRT), los sistemas de TV y los dispositivos de impresión.

En los sistemas con CRT, la posición horizontal y vertical de cada elemento de la imagen se convierte en un voltaje que se

emplea para orientar el haz de electrones del tubo, obteniendo así la dirección bidimensional necesaria para producir la imagen de salida. En cada punto, la intensidad del haz es modulada empleando un voltaje que es proporcional al valor del punto correspondiente en el arreglo numérico. El patrón de intensidades luminosas resultante se registra mediante una cámara fotográfica enfocada en la superficie del tubo de rayos catódicos. Algunos sistemas emplean un tubo de fósforo de gran persistencia que permite observar la imagen una vez que se ha completado el proceso de barrido. Aunque las imágenes registradas a través del proceso fotográfico pueden ser de excelente calidad, las mismas imágenes generalmente poseen una tonalidad pobre cuando son mostradas en un CRT de gran persistencia, debido a las limitaciones del sistema visual humano para responder a este tipo de despliegue [2].

Los sistemas de despliegue con TV convierten la imagen almacenada en la memoria de la computadora en una señal de video que puede ser desplegada en un monitor de TV. La ventaja de estos sistemas radica en que los despliegues realizados en un monitor de video poseen una tonalidad que se acerca mucho a una fotografía, produciendo así una salida que es fácilmente asimilada por el sistema visual humano. La desventaja de los sistemas de despliegue en TV es que deben refrescar el monitor con aproximadamente 30 cuadros por segundo. Debido a que la mayoría de los sistemas de cómputo para propósitos generales no son capaces de transmitir datos a esta velocidad, el principal problema en el diseño de estos sistemas de despliegue es proporcionar alguna memoria de almacenamiento intermedio para transferir los datos al monitor de video. La mayoría de los sistemas comerciales resuelven este problema en dos formas distintas. Una forma es emplear una memoria rápida de estado sólido para almacenar la imagen completa; la pantalla se refresca con 30 cuadros por segundo reciclando la memoria y combinando la información binaria almacenada en una señal analógica por medio de circuitos acondicionadores y convertidores digital-analógicos de alta velocidad. El otro método es almacenar la imagen en un disco de alta densidad. El almacenamiento se realiza nuevamente en forma binaria y cada track del disco contiene un bit para todos los pixeles de la imagen. Una imagen de $N \times N$ con 2^n niveles posibles, por ejemplo, requiere n

tracks de almacenamiento, cada uno de $N \times N$ bits. La velocidad de transferencia necesaria se logra al hacer girar el disco a través de n cabezas sensoras. Para cualquier posición del disco, la información binaria en las n cabezas se combina para producir un voltaje proporcional al nivel de gris de un pixel en la imagen. Conforme el disco gira, se crea una señal analógica al acondicionar estos niveles de voltaje e introducirlos en un convertidor digital-analógico de alta velocidad [2].

Los dispositivos de impresión de imágenes son especialmente útiles cuando se trabaja con imágenes de baja resolución. Una forma simple de generar imágenes en tonos de gris directamente sobre el papel es emplear la capacidad de encimar caracteres de las impresoras de línea convencionales. El nivel de gris de cualquier punto de la impresión se puede controlar mediante el número y la densidad de los caracteres sobreimpresos en ese punto. Con una selección adecuada del conjunto de caracteres, es posible conseguir distribuciones razonablemente adecuadas para los niveles de gris con un programa de computadora bastante sencillo. Otra forma de imprimir imágenes en papel es utilizar la capacidad de las impresoras de matriz de permitir definir caracteres para impresión empleando secuencias de escape. De esta forma, es posible definir tantos niveles de gris como lo permita las dimensiones de la matriz de puntos con que se diseñan los caracteres.

Cuando se emplean dispositivos como los tubos de rayos catódicos y los sistemas de TV, es necesario diseñar programas que permitan transferir los datos desde la computadora y posteriormente desplegarlos. En general, este tipo de programas se realizan en el lenguaje ensamblador propio del procesador que controla al sistema de despliegue.

Es muy común que los equipos de digitalización se acompañen de un sistema de despliegue. Al adquirir un sistema comercial de este tipo, también se obtiene el software con que éste funciona. Sin embargo, por razones obvias, no incluyen los programas fuente de dicho software y esto dificulta hacer cualquier modificación.

En el tercer capítulo se presentan algunos programas para desplegar imágenes en el monitor de una computadora personal. Utilizar como sistema de despliegue el monitor de la computadora

con la cual se procesan las imágenes representa un enorme ahorro de recursos. También se trata el diseño de programas para imprimir imágenes con una impresora de matriz estándar.

II.4) *Sugerencias para la integración de un sistema de PDI.*

Un sistema para procesar imágenes debe integrarse tomando en cuenta dos factores esenciales: el tipo de procesos que se pretenden realizar y los recursos económicos con que se cuenta.

Como hemos visto, un sistema de procesamiento debe contar con dispositivos para digitalizar, procesar y desplegar las imágenes. Esto significa que, dependiendo de los factores citados, se debe seleccionar un cierto tipo de dispositivo para realizar cada una de estas funciones.

Para ilustrar el proceso de integración de un sistema para procesamiento de imágenes, tomaremos como modelo el sistema que actualmente funciona en el laboratorio de digitalización de imágenes del Instituto de Física de la UNAM.

Los procesos de digitalización y despliegue se realizan por medio de un equipo comercial para diseño gráfico. Este equipo permite digitalizar imágenes en 128 tonos de gris ó colores. Las imágenes se introducen mediante una cámara de TV y son desplegadas en un monitor de TV que soporta arreglos de hasta 512 X 512 pixeles. El equipo está conectado con una computadora personal tipo AT, en la cual se almacenan las imágenes en forma digital.

Para el procesamiento de los datos se emplea la PC tipo AT ó bien, cuando el proceso lo requiere, se utiliza una minicomputadora VAX 11-780, que está conectada con la PC.

Otro elemento del sistema es una impresora láser, con la cual se consigue imprimir imágenes con 64 tonos de gris. Cuando se requiere una impresión de mayor calidad se obtienen fotografías de las imágenes desplegadas en el monitor de TV.

En lo que respecta al software, se cuenta con un gran número de programas de diseño propio. Estos programas permiten realizar prácticamente todos los procesos relatados en el primer capítulo.

Todo este sistema puede ser reducido sin perder funcionalidad para muchas aplicaciones. La configuración más simple, con la cual

se pueden realizar la mayoría de los procesos que se trataron en el primer capítulo, es la siguiente:

- a) Digitalización y despliegue: equipo comercial para diseño gráfico. Su costo aproximado es de cinco mil dólares.
- b) Procesamiento: computadora personal. Su costo depende del modelo que se trate, pero se recomienda una máquina tipo AT dotada con coprocesador numérico (entre 3 y 4 mil dólares).
- c) Adicionalmente puede realizarse la impresión en papel con una impresora de matriz, cuyo costo no rebasa los 1500 dólares.

En particular, este sistema puede funcionar para el análisis de imágenes de percepción remota. Si además tomamos en cuenta que muchas de las imágenes que se manejan en este campo (por ejemplo las imágenes de satélite) pueden obtenerse ya en forma discreta cuando se compran, el sistema podría reducirse inclusive a la computadora personal (con un monitor de alta resolución para el despliegue) y a la impresora de matriz.

En el tercer capítulo, se presentan algunas recomendaciones para el desarrollo de software para procesamiento, despliegue e impresión de imágenes. Especial atención se dedica al diseño de programas de despliegue e impresión para computadoras personales e impresoras de matriz, respectivamente. En lo que se refiere a la etapa de procesamiento, las especificaciones que se brindan son válidas para cualquier tipo de computadora, a menos que se indique lo contrario.

CAPÍTULO III

DISEÑO Y CONSTRUCCIÓN DE UN PAQUETE PARA PDI

En el capítulo anterior se indican algunos elementos que deben ser tomados en cuenta cuando se desea adquirir un equipo para procesar imágenes. Como se menciona entonces, dos factores fundamentales son el tipo de procesos que se pretenda realizar y los recursos económicos con que se cuenta.

Una vez que se ha resuelto lo relativo a la selección del equipo, se presenta nuevamente una disyuntiva: desarrollar o comprar el software requerido.

En el mercado del software, es posible encontrar algunos paquetes para procesamiento de imágenes. Estos paquetes han sido desarrollados por diversas universidades, agencias gubernamentales y empresas comerciales. Entre los más conocidos se encuentran los siguientes:

VICAR. - Desarrollado por el "Jet Propulsion Laboratory" de la Nasa.

LARSYS. - Sistema desarrollado por la Universidad de Purdue, USA. Permite analizar imágenes multiespectrales.

IMAGE 100. - Desarrollado en el laboratorio de análisis digital de imágenes de la General Electric.

ERMAN II. - Desarrollado en el laboratorio de recursos terrestres de la división de sistemas de IBM.

Muchas veces, invertir en la compra de un paquete comercial no resulta la mejor elección. Uno de los inconvenientes, sin duda, es su elevado costo. Sin embargo, la principal desventaja que poseen es que generalmente incluyen programas de aplicación general, que no pueden ser fácilmente modificados para emplearlos en algún problema específico.

Debido a ello, la compra de un paquete comercial solamente se recomienda cuando el desarrollo del software no sea posible, o bien, cuando el tipo de procesamiento que se pretenda realizar se apegue a esa generalidad que proporcionan los paquetes comerciales.

Las secciones restantes de este capítulo se ocupan del

problema que representa diseñar un paquete para procesamiento de imágenes. El objetivo, más que presentar listados de programas, es brindar una metodología que pueda ser adoptada tanto al emprender el diseño de los programas básicos con que debe contar el paquete inicialmente, como para la adaptación de algoritmos novedosos que puedan ser paulatinamente incorporados.

Los programas de un paquete, como hemos podido ver en el capítulo segundo, deben servir fundamentalmente para dos propósitos: el procesamiento y el despliegue de las imágenes.

La primera sección del capítulo se dedica a tratar lo relativo a la selección de los procesos básicos, suponiendo que nos interesan aplicaciones muy generales.

En la segunda sección se brindan especificaciones para el diseño de programas para despliegue de imágenes digitales.

Una sección posterior se ocupa de la programación de los algoritmos seleccionados en la primera sección y de programas para despliegue basados en la teoría expuesta en la segunda sección.

III.1) Selección de algoritmos.

Un sistema de propósito general debe contar al menos con un mínimo de algoritmos que abarquen todas las áreas de interés del procesamiento de imágenes.

Como se mencionó en su momento, el proceso de digitalización se realiza generalmente empleando el software que acompaña al dispositivo digitalizador. Cualquier modificación a dicho software debe hacerse tomando en cuenta las especificaciones que el fabricante proporciona. Por todo ello, cuando se diseña un paquete de procesamiento, no se contempla el desarrollo de programas para digitalizar.

Las técnicas más importantes, cuando se trata de aplicaciones generales, son las de realce de imágenes. Los procesos de restauración, segmentación y descripción pueden ser cubiertos con sólo implementar algunas de las técnicas más simples y de amplia difusión en la literatura.

El conjunto de procesos más elemental que debe tenerse en

mente es el siguiente:

- a) Obtención de contornos.
- b) Realce general y realce de bordes.
- c) Restauración.
- d) Suavizamiento general (para eliminar ruido).
- e) Sumas, restas y otras operaciones simples.

Todos estos procedimientos, con excepción de las sumas y restas de imágenes, pueden ser realizados en el dominio de la frecuencia (empleando la transformada de Fourier) o en el dominio espacial (a través de ventanas). Cuál opción elegir depende de muchos factores tales como la facilidad de empleo, la familiaridad que se tenga con cada método, la disponibilidad de equipo, etc. Una consideración más profunda se relaciona con los tiempos de cómputo requeridos. En nuestro caso, emplearemos este último factor para determinar cuándo un método puede ser elegido en lugar del otro.

Para ilustrar la manera de comparar el costo computacional de los procesos en los dominios espacial y frecuencial se considera el caso de la técnica de realce geométrico global de la imagen.

Ambos procesos, en el dominio de la frecuencia y en el dominio espacial, se componen simplemente de un conjunto de multiplicaciones y sumas. Ningún otro tipo de operación matemática se emplea. Por consiguiente, es suficiente hacer una comparación basada en el número de multiplicaciones y sumas necesario para conseguir un cierto resultado. Si además se ignoran las sumas, dado que generalmente son más rápidas que las multiplicaciones en la mayoría de las computadoras y debido también a que son comparables en número a las multiplicaciones involucradas.

Considerando, por simplicidad, una imagen cuadrada de $K \times K$ píxeles y una ventana de dimensión $M \times N$, el número total de multiplicaciones necesario para efectuar un realce geométrico sobre todos los píxeles de la imagen (ignorando cualquier dificultad con los bordes de la imagen) es

$$N_c = M N K^2 \quad (28)$$

El número de multiplicaciones requerido en el dominio de la frecuencia es,

$$N_F = 2 K^2 \log_2 K + K^2 \quad (29)$$

De esta forma, la comparación de costos es

$$\frac{N_C}{N_F} = M N / (2 \log_2 K + 1) \quad (30)$$

Cuando esta razón es menor de la unidad, es más económico emplear el proceso en el dominio del espacio. Aunque esta comparación no toma en cuenta muchos factores, es un punto de arranque razonable para elegir entre ambos métodos [13].

En la tabla 1 se presenta un conjunto de valores de N_C/N_F para varios tamaños de imágenes y ventanas. En esta tabla se puede observar que cuando se utiliza una ventana de 3 X 3 para efectuar una operación de realce, siempre se emplea menos tiempo que en un realce basado en la transformada de Fourier. También se observa que una ventana de 3 X 5 resulta más consumidora de recursos para la mayoría de las dimensiones prácticas de las imágenes. Sin embargo, se aprecia que para dimensiones de ventana por debajo de 3 X 5, la técnica en el dominio del espacio siempre resultará más económica.

Tabla 1. Relación entre el número de operaciones requeridas para un realce geométrico en los dominios espacial y frecuencial [13].

Dimensiones de la imagen	Dimensiones de la ventana				
	3 X 3	3 X 5	5 X 5	5 X 7	7 X 7
16 X 16	1.00	1.67	2.78	3.89	5.44
64 X 64	0.82	1.15	1.92	2.69	3.77
128 X 128	0.60	1.00	1.67	2.33	3.27
256 X 256	0.53	0.88	1.47	2.06	2.88
512 X 512	0.47	0.79	1.32	1.84	2.58
1024 X 1024	0.43	0.71	1.19	1.67	2.33
2048 X 2048	0.39	0.65	1.09	1.52	2.13
4096 X 4096	0.36	0.60	1.00	1.40	1.96

Sin embargo, debe remarcarse que existen varios procesos que pueden ser realizados en el dominio frecuencial y que no resultan

viables en el dominio espacial. Un ejemplo es la remoción de la información periódica espuria en una imagen.

Este tipo de comparaciones son especialmente útiles cuando los recursos de cómputo disponibles son limitados. En nuestro medio, los sistemas de procesamiento más universales son las microcomputadoras. En este tipo de equipos todavía no es posible implantar la mayoría de los algoritmos más efectivos, pues involucran cálculos extremadamente consumidores de tiempo.

A continuación se procede a seleccionar los algoritmos adecuados para realizar cada uno de los procesos básicos que han sido enumerados. En dicha selección, se ven favorecidas las técnicas en el dominio espacial con operadores de hasta 3×3 píxeles.

Sin embargo, con objeto de no cerrar el acceso a las técnicas en el dominio de la frecuencia, en la segunda sección, donde se aborda la programación de las técnicas seleccionadas, también se incluye el tratamiento de la transformada rápida de Hartley (FHT). La FHT es un algoritmo alternativo a la FFT, que aunque no es tan popular como este, posee características que modifican el balance presentado entre los métodos en los dominios espacial y frecuencial.

III.1.a) *Obtención de contornos.*

Es posible encontrar en la literatura especializada un buen número de técnicas para la detección de los contornos de una imagen. Dos técnicas que, a pesar de ser relativamente antiguas, son razonablemente eficientes y bastante económicas, son las basadas en los operadores gradiente y laplaciano.

Estas técnicas no han sido desechadas con el paso de los años y continúan siendo utilizadas al lado de las modernas técnicas presentes en los algoritmos de visión por computadora.

Debido a que la teoría de estos algoritmos no se expone en el primer capítulo, a continuación se presenta una breve exposición acerca de ella.

III.1.a.1) El gradiente.

Si $\partial f/\partial x$ y $\partial f/\partial y$ son las razones de cambio de una función f en dos direcciones perpendiculares, entonces la razón de cambio en cualquier dirección θ (medida a partir del eje x) es una combinación lineal de éstas:

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial x} \cos \theta + \frac{\partial f}{\partial y} \sin \theta \quad (31)$$

La dirección en la cual esta razón de cambio tiene la mayor magnitud es $\tan^{-1}((\partial f/\partial y) / (\partial f/\partial x))$, y su magnitud es $[(\partial f/\partial x)^2 + (\partial f/\partial y)^2]^{1/2}$. Al vector que posee esta magnitud y dirección se le denomina el gradiente de f .

Si empleamos una derivada direccional como una medida de la extensión del borde, su respuesta variará con la orientación del borde. Para evitar esto, podemos emplear simplemente la magnitud del gradiente, puesto que automáticamente proporciona la razón de cambio en la dirección de máxima variación [1].

En forma análoga, para una imagen digital podemos emplear primeras diferencias en vez de primeras derivadas, es decir,

$$\begin{aligned} (\Delta_x f)(x, y) &= f(x, y) - f(x-1, y) \\ (\Delta_y f)(x, y) &= f(x, y) - f(x, y-1) \end{aligned} \quad (32)$$

Nótese que estos son operadores de convolución digital que convolucionan f con los patrones

$$\begin{bmatrix} -1 & 1 \end{bmatrix} \quad y \quad \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad (33)$$

respectivamente. Podemos combinar entonces $\Delta_x f$ y $\Delta_y f$ tomando, por ejemplo, la raíz cuadrada de la suma de los cuadrados. Sin embargo, no parece correcto combinar $\Delta_x f$ y $\Delta_y f$ en la misma posición (x, y) , dado que las diferencias que estos operadores miden no están localizadas simétricamente con respecto a (x, y) ; Δ_x

emplea un par de píxeles centrado en $(x-1/2, y)$ mientras que Δ_y emplea un par centrado en $(x, y-1/2)$.

Podemos evitar esta objeción empleando otros operadores diferenciales. Las siguientes son dos posibilidades:

$$\begin{aligned}(\Delta_{2x}f)(x, y) &= f(x+1, y) - f(x-1, y) \\(\Delta_{2y}f)(x, y) &= f(x, y+1) - f(x, y-1)\end{aligned}\tag{34}$$

Estos operadores miden los cambios horizontales y verticales en f a través de (x, y) . Sus valores no dependen de $f(x, y)$ mismo. Son las convoluciones de f con

$$\begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \quad \text{y} \quad \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}\tag{35}$$

respectivamente, miden "diferencias centrales".

$$\begin{aligned}(\Delta \cdot f)(x, y) &= f(x+1, y+1) - f(x, y) \\(\Delta - f)(x, y) &= f(x, y+1) - f(x+1, y)\end{aligned}\tag{36}$$

Estos operadores miden los cambios diagonales a 45° y 135° en f , usando pares de puntos que circundan simétricamente $(x+1/2, y+1/2)$. (Recuérdese que pueden emplearse las derivadas en cualquier par de direcciones perpendiculares para calcular el gradiente). Estos operadores convolucionan f con

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad \text{y} \quad \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\tag{37}$$

El operador gradiente digital basado en $\Delta \cdot$ y $\Delta -$ es conocido como el operador de Roberts.

En (34) las diferencias son tomadas a intervalos de dos distancias unitarias, mientras que en (36) se involucran distancias de $\sqrt{2}$. Con objeto de obtener razones de cambio por

unidad de distancia, Δ_x y Δ_y deben ser divididos entre 2, mientras que Δ^+ y Δ^- deben dividirse entre $\sqrt{2}$. Además, (34) responde a un borde perfecto en dos posiciones, es decir, cuando (x,y) es adyacente al borde en ambos lados; de este modo, produce respuestas de dos puntos de espesor.

Cómo deben combinarse los dos operadores (Δ_x y Δ_y , o Δ^+ y Δ^-) para formar un gradiente digital? En analogía con el caso continuo, podemos tomar la raíz cuadrada de la suma de los cuadrados. Sin embargo, como veremos, obtenemos una mejor invariancia a la orientación si los combinamos en una forma más simple (1). Sean E_H , E_V , E_{D^+} y E_{D^-} bordes digitales perfectos de valor h en las direcciones 0° , 90° , 45° y 135° , respectivamente, esto es,

	n	n	n	n			h	h	0	0		
	h	h	h	h			h	h	0	0		
	0	0	0	0			h	h	0	0		
	0	0	0	0			h	h	0	0		
			h	h	0	0		0	h	h		
		h	h	0	0	y		0	0	h	h	
	h	h	0	0				0	0	h	h	
h	h	0	0						0	0	h	h

Las magnitudes de las respuestas de Δ_x , Δ_y , Δ^+ y Δ^- a estos bordes son las siguientes:

	E_H	E_V	E_{D^+}	E_{D^-}
$ \Delta_x $	0	h	h	h
$ \Delta_y $	h	0	h	h
$ \Delta^+ $	h	h	0	h
$ \Delta^- $	h	h	h	0

Las respuestas de $\sqrt{\Delta_{2x}^2 + \Delta_{2y}^2}$ a estos bordes son h , h , $h\sqrt{2}$ y $h\sqrt{2}$, existiendo un factor de $\sqrt{2}$ en favor de los bordes diagonales; y las respuestas de $\sqrt{\Delta_x^2 + \Delta_y^2}$ son $h\sqrt{2}$, $h\sqrt{2}$, h , h , dando un factor en contra de los bordes diagonales. Estos factores pueden ser eliminados si empleamos el máximo de los valores absolutos, en vez del RMS, como la magnitud del gradiente; esto hace todas las respuestas iguales a h . Esta forma posee además otras dos ventajas:

- (1) Evita el costo computacional de las operaciones de los cuadrados y la raíz cuadrada.
- (2) Produce valores en el mismo rango de la escala original de grises, lo cual es conveniente para propósitos de despliegue.

III.1.a.2) El laplaciano.

El laplaciano es un operador diferencial que no varía con la orientación. Para imágenes digitales está dado por:

$$(\nabla^2 f)(x, y) = [f(x-1, y) + f(x+1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y) \quad (38)$$

que es la convolución digital de f con

$$\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix} \quad (39)$$

$\nabla^2 f$ es proporcional a $f - \bar{f}$, donde

$$\bar{f}(x, y) = 1/5 [f(x, y) + f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] \quad (40)$$

de modo que \bar{f} es el resultado del promedio local de f .

El laplaciano digital, dado que es un operador de segundas

diferencias, tiene respuesta cero a las rampas lineales, pero responde a las cuestas y valles, donde existe una variación en la razón de cambio de los niveles de gris. Así, responde a cada lado de un borde una vez con signo positivo y otra con signo negativo. Si queremos solamente respuestas positivas, podemos emplear el valor absoluto $|\nabla^2 f|$, o el valor positivo $(\nabla^2 f)^+$ ($=\nabla^2 f$ cuando $\nabla^2 f \geq 0$; $=0$ cuando $\nabla^2 f < 0$). Estas respuestas pueden tener valores tan altos como cuatro veces el máximo nivel de gris; si deseamos asegurar que el rango de grises sea preservado, debemos dividirlos entre cuatro.

El laplaciano digital responde a bordes, pero responde aún más a esquinas, líneas, extremos de líneas y puntos aislados. De este modo, en una imagen con ruido, el ruido producirá mayores valores que los bordes, a menos de que tenga un contraste mucho menor. (Nótese además que la respuesta a los bordes diagonales es dos veces más grande que las respuestas a los bordes horizontales y verticales, de modo que el laplaciano digital no es realmente invariante a la orientación). El gradiente digital responde a los bordes en la misma forma que el ruido. De este modo, el gradiente generalmente será un mejor detector de bordes que el laplaciano.

Si empleamos el máximo del valor absoluto, en lugar de la suma, de las segundas diferencias en las direcciones x y y , obtenemos un "pseudo-laplaciano" que tampoco responde a rampas, pero es un tanto menos sensible al ruido que el laplaciano. Otro operador es el valor absoluto de la diferencia entre la media y la mediana, calculada sobre la misma vecindad del pixel dado [1].

III.1.b) *Realce general y realce de bordes.*

En el capítulo primero se describen varias técnicas de realce en el dominio espacial que, debido a su sencillez, pueden ser soportadas inclusive en microcomputadoras. La explicación que ahí se presenta es suficiente para conseguir programarlas.

Un algoritmo, que no se incluye en el capítulo I, pero que resulta especialmente adecuado para realzar imágenes, es el que a continuación se expone.

Una forma de modelar algunos procesos de difusión que afectan a la imágenes es mediante la conocida ecuación

$$\partial g / \partial t = k \nabla^2 g \quad (41)$$

donde g es una función de x, y y t (tiempo), y $k > 0$ es una constante. En $t = 0$, $g(x, y, 0)$ es la imagen no degradada $f(x, y)$; en algún $t = \tau > 0$, tenemos a la imagen degradada $g(x, y, \tau)$. Si expandemos $g(x, y, t)$ en una serie de Taylor alrededor de $t = \tau$, tenemos

$$g(x, y, 0) = g(x, y, \tau) - \tau \frac{\partial g}{\partial t}(x, y, \tau) + \frac{\tau^2}{2} \frac{\partial^2 g}{\partial t^2}(x, y, \tau) - \dots \quad (42)$$

Si despreciamos los términos cuadráticos y de orden superior, y sustituimos f por $g(x, y, 0)$ y $k \nabla^2 g$ por $\partial g / \partial t$, obtenemos

$$f = g - k \tau \nabla^2 g \quad (43)$$

De este modo, en una primera aproximación, podemos recuperar la imagen no degradada f restando de g un múltiplo positivo de su laplaciano. Si se desea, pueden emplearse aproximaciones de mayor orden basadas en la expansión en serie de Taylor [1].

III.1.c) Restauración.

La restauración de imágenes es uno de los procesos más consumidores de tiempo en el PDI. Las técnicas más efectivas, tanto en el dominio frecuencial como en el dominio espacial, difícilmente pueden ser implantadas en microcomputadoras.

En el primer capítulo se muestra un algoritmo de restauración que posee como virtud principal su relativa sencillez computacional. Como ahí se indica, este algoritmo es muy efectivo para muchas aplicaciones.

Por todo ello, el método de sustituciones sucesivas, cuyo

algoritmo se muestra en el primer capítulo, es la técnica de restauración elegida para conformar el paquete.

III.1.d) *Suavizamiento general.*

Entre las técnicas que se presentan en el primer capítulo, se encuentran dos bastante populares. Son las técnicas conocidas como la media y la mediana. Deben su popularidad a que son esquemas muy simples, que proporcionan resultados suficientemente buenos.

Estas dos técnicas se encuentran presentes en la mayoría de los paquetes de procesamiento de imágenes. En nuestro caso, basta con el desarrollo de estos dos algoritmos para suavizamiento, cuyo planteamiento general se expone en el primer capítulo.

III.1.e) *Sumas, restas y otras operaciones simples.*

Muchas veces es de gran utilidad el contar con los programas necesarios para realizar operaciones simples como sumas, restas o cocientes de imágenes.

Estas operaciones operan exclusivamente sobre un pixel de la imagen, es decir, no requieren de la aplicación de ningún tipo de ventana. La suma, resta y cociente de imágenes consiste en realizar dichas operaciones entre los pixeles correspondientes de cada imagen.

Estos algoritmos son demasiado simples y programarlos no representa ningún problema. Por ello, no son tratados en la tercera sección.

III.2) *Rutinas de despliegue.*

Un sistema de procesamiento debe contar con rutinas para el despliegue de las imágenes digitales. Esta función puede ser realizada utilizando un monitor de alta resolución o alguno de los dispositivos adicionales que se mencionan en el capítulo II.

En la primera parte de esta sección se expone la teoría general que debe tomarse en cuenta al diseñar rutinas de despliegue para monitores de alta resolución. En la segunda parte se explica brevemente la forma en que debe diseñarse un programa de impresión de imágenes en tonos de gris para una impresora de matriz.

III.2.1) Despliegue en monitores de alta resolución.

Para una escala de grises de N bits y un sistema de color (R, G, B) con M bits disponibles para cada color primario, únicamente M bits de la escala de grises verdadera pueden ser simulados en el monitor a color [14].

El valor de los píxeles de una imagen en una escala de grises de N bits se ubica en el rango $0 - (2^N - 1)$, donde 0 corresponde al negro y $(2^N - 1)$ representa al blanco. Cualquier valor entre estos límites proporciona un valor en una escala ordenada. Una imagen en color (de M bits) posee píxeles cuyo color se determina mediante la tripleta (R, G, B), donde $0 \leq R, G, B \leq 2^M - 1$.

Estas tripletas pueden ser vistas como coordenadas en un espacio tridimensional. En particular (0, 0, 0) representa al negro y $(2^M - 1, 2^M - 1, 2^M - 1)$ al blanco.

En la figura 7 observamos lo siguiente:

- (i) Los niveles de gris reales caen en la diagonal blanco-negro y son los puntos (x, x, x) , para $0 \leq x \leq 2^M - 1$.
- (ii) Entre más lejana este la tripleta (R, G, B) de la diagonal blanco/negro, serán más distintas las tres componentes y el valor resultante será desplegado con mayor colorido.
- (iii) La diagonal blanco/negro es normal al triángulo de Maxwell.
- (iv) Existen $3(2^M - 1)$ planos paralelos al triángulo de Maxwell que intersectan los ejes coordenados R, G, B en puntos enteros. En el espacio tridimensional (R, G, B) estos planos quedan definidos por la ecuación

$$R + G + B = K \quad \text{para } K = 0, 1, \dots, 3(2^M - 1) \quad (44)$$

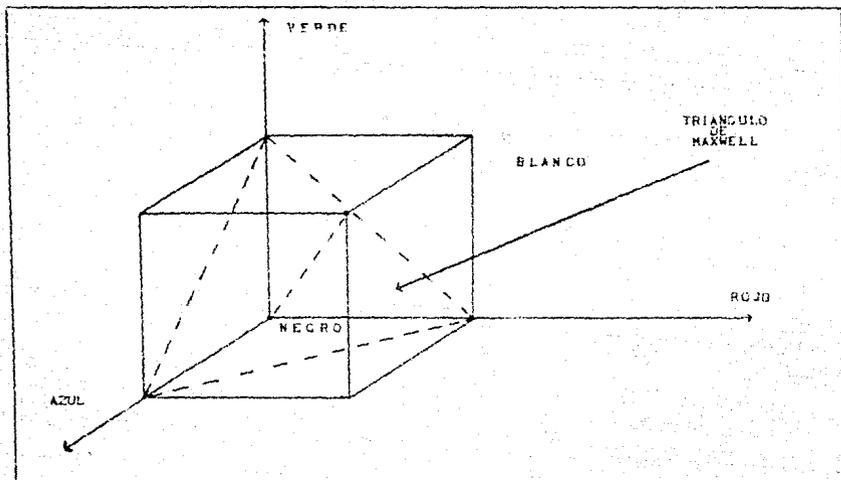


Figura 7. Espacio R, G, B.

Una pseudo-escala de colores debe ser diseñada considerando aquellos puntos con coordenadas enteras que caen en dichos planos, que están próximos a la diagonal blanco/negro y que están contenidos en el cubo

$$0 \leq R, G, B \leq 2^M - 1 \quad (45)$$

Si lo que se desea es realizar un despliegue en colores falsos, entonces las componentes (R, G, B) deben ser seleccionadas en tal forma que los colores que compongan la escala tengan un colorido especial. Esto se consigue eligiendo algunos colores, distantes entre sí, pero contenidos en un plano cercano al plano del blanco.

III.2.2) Impresión de imágenes.

Los problemas técnicos que representa el diseño de un programa de impresión son básicamente los siguientes [15]:

- i) Resolución espacial: la máxima resolución se obtiene al

maximizar el número de tonos de gris y minimizar el tamaño del pixel considerado.

ii) Contraste entre tonos: El contraste máximo entre tonos está condicionado por el procedimiento de estructuración del pixel.

iii) Número de tonos: es función de la resolución espacial.

Para representar un pixel de la imagen se utiliza una matriz regular de puntos. En dicha matriz, tanto el número de puntos como su distribución controla el nivel de gris de cada pixel.

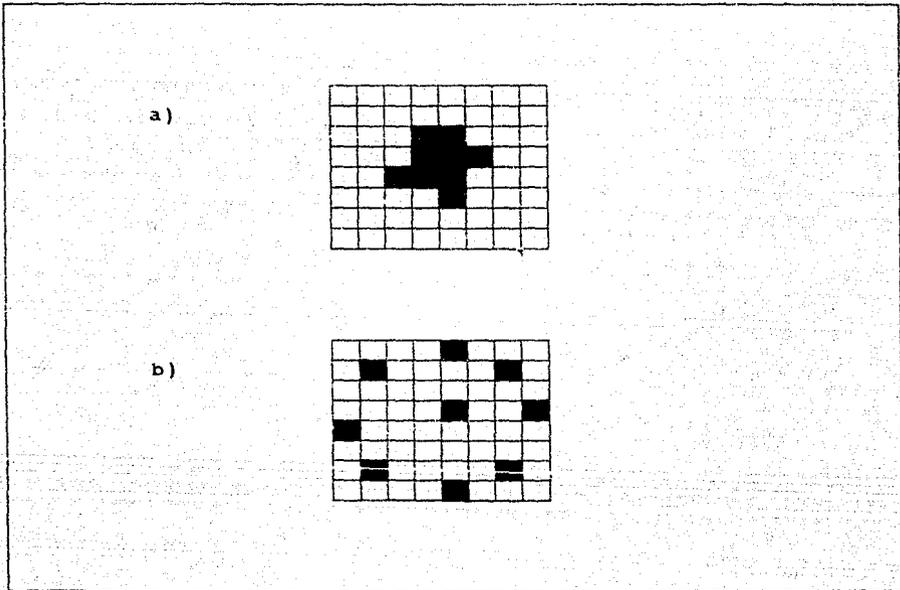


Figura 8. Distribuciones:
a) centrifuga; b) sistemática.

Puesto que cada matriz de $M \times M$ representa un pixel, entonces un punto de la imagen queda representado por M bytes y cada punto de la matriz representa un punto de la impresora. Esto implica que la imagen es barrida sobre la superficie del papel como una sucesión de líneas equidistantes; cada línea compuesta por un mismo número de pixeles.

Para la generación de tonos se puede utilizar una técnica denominada "tonos fijos", con dos variantes: 1) distribución centrífuga y 2) distribución sistemática.

En el caso de la distribución centrífuga la distribución que de los puntos se haga dentro de la matriz debe ser en forma de espiral centrífuga. En el caso de la distribución sistemática, los puntos se distribuyen en tal forma que queden lo menos cercano posible para evitar el efecto de parches. Ambas distribuciones se presentan en la figura 8.

III.3) Programación de algoritmos.

La última etapa en el proceso de construcción del paquete para procesamiento de imágenes, es la codificación de los algoritmos seleccionados en la forma de programas de computadora.

A continuación se brindan una serie de especificaciones generales cuyo objetivo es facilitar la programación de las técnicas de procesamiento que se presentan en los primeros capítulos.

Para ilustrar esta etapa, se hacen algunos comentarios sobre la integración de una biblioteca de funciones. Las funciones con las que se ha integrado esta biblioteca corresponden con algunos de los algoritmos tratados previamente. Un listado completo de tales funciones se encuentra en el apéndice A.

También en el mencionado apéndice, se incluyen dos rutinas para despliegue e impresión de imágenes, así como un procedimiento para calcular la transformada de Hartley. Estas rutinas se comentan en detalle en las secciones posteriores.

III.3.1) *Lenguajes de programación.*

Como se indica en el segundo capítulo, con objeto de ganar velocidad, el lenguaje de programación que se seleccione debe ser del nivel más bajo posible. Una alternativa es programar en lenguaje ensamblador, pero esto disminuye la transportabilidad del software.

Como sucede en muchas áreas del procesamiento digital de señales, uno de los lenguajes que se utilizan frecuentemente para procesar imágenes es FORTRAN. Sin embargo, cuando los recursos se limitan a una microcomputadora, los tiempos de proceso que se obtienen con este lenguaje son demasiado altos.

Un lenguaje que cobra cada vez más adeptos, debido a su gran flexibilidad, es el lenguaje C. Algunos autores lo incluyen entre los lenguajes de alto nivel, sin embargo, es el lenguaje con nivel más cercano al ensamblador y probablemente el más versátil de todos.

Una opción intermedia es PASCAL, que es un lenguaje fuertemente estructurado. Existen compiladores muy eficientes para este lenguaje en microcomputadoras.

III.3.2) *Memoria de proceso y de almacenamiento.*

Otro factor muy importante es la memoria disponible para procesar los datos. En el caso de computadoras del tipo mini o mayores, la memoria es suficiente para procesar una imagen de alta resolución teniendo todos los datos en la memoria principal de la máquina. Cuando se trata de microcomputadoras, la imagen debe ser procesada por partes.

Los sistemas de cómputo más comunes en nuestro país son las computadoras personales. Este tipo de máquinas cuentan con una capacidad de memoria RAM relativamente limitada.

Todos los algoritmos seleccionados en la primera sección, por emplear operadores de dimensiones no mayores de 3 X 3 píxeles, pueden ser implementados teniendo en memoria solamente una porción de la imagen.

Por otra parte, como hemos visto, el procesamiento de imágenes genera una gran cantidad de datos que deben ser almacenados en alguna parte. Una imagen digital en 256 tonos de gris se almacena empleando un byte por pixel. Cuando la imagen se digitaliza en color, se deben emplear dos o más bytes por pixel. Una imagen digital de alta resolución puede medir 256 o más pixeles por lado.

Las computadoras personales actuales pueden ser equipadas con discos duros con capacidad de algunas decenas de mega-bytes. Por tal motivo, el almacenamiento de datos en este tipo de computadoras representa un serio problema.

III.3.3) Estructuras de datos.

Cuando fue definida una imagen digital, se dijo que ésta era un arreglo matricial de ciertas dimensiones. Sin embargo, almacenar los datos de una imagen en forma de matriz requiere un espacio de almacenamiento muy grande. En realidad, una imagen digital se almacena en un archivo secuencial de datos, generalmente codificados en ASCII.

Los procesos que emplean ventanas requieren los valores de cada pixel de la imagen y de los vecinos que caen dentro de la ventana en cada momento. En esta clase de procesos, una estructura matricial, con las dimensiones que permita la memoria principal de la máquina, resulta adecuada.

En los procesos donde el valor que asume cada pixel no depende del valor actual de ninguno de sus vecinos, la lectura y procesamiento de los datos se realiza en forma secuencial. Para este tipo de operaciones, una estructura muy adecuada es un apuntador (que apunte a un vector de datos tan grande como la memoria lo permita).

En general, los procesos seleccionados en la primera sección, no emplean más que dos tipos de datos. Para la lectura inicial y la escritura final de los datos, éstos deben ser convertidos de caracteres ascii a enteros y de enteros a caracteres ascii, respectivamente.

III.3.4) Construcción de una biblioteca de funciones.

Dado que los datos de una imagen se almacenan en un archivo empaquetado codificado en *ascii*, resulta muy conveniente colocar un encabezado al principio del archivo, en donde se indiquen las dimensiones de la imagen y algún otro tipo de información que sea de interés.

La lectura de los datos de entrada depende del encabezado que se emplee y de la forma en que estén organizados los archivos del sistema.

Como se menciona anteriormente, la programación de operaciones puntuales y de operaciones con ventanas sigue una lógica diferente.

Las operaciones puntuales son muy simples de programar, pues nunca agotan la memoria de proceso, por limitada que sea.

Un ejemplo de operación puntual es la empleada por la técnica de reescalamiento de los niveles de gris, descrita en el primer capítulo. Este proceso se realiza empleando la función *normal* del apéndice A.

Las operaciones con ventanas, por necesitar a la vez los valores de varios píxeles, deben ser programadas empleando arreglos matriciales.

Si la memoria de la máquina no es suficiente, la imagen debe ser leída en varios fragmentos. Después de leer cada fragmento, se efectúa su procesamiento y los datos resultantes son escritos en un archivo similar al archivo de entrada.

Las funciones *gradiente* y *laplaciano* (ver apéndice A) utilizan los algoritmos del mismo nombre para detectar contornos.

Las funciones *prom4*, *prom9* y *mediana* sirven para suavizar una imagen empleando las técnicas de promedios con ventanas de 4 y 9 píxeles y de la mediana, respectivamente.

La rutina de *realce* general y de bordes, expuesta en la primera sección de este capítulo, se realiza a través de la función *realce*.

Las funciones *restaurar1* y *restaurar2* emplean el algoritmo de restauración conocido como "método de sustituciones sucesivas", descrito en el capítulo I. La función *restaurar1* supone un patrón

de difusión modelado con una ventana de 4 píxeles, todos con pesos de 1/4. La función *restaura2* emplea un patrón de 9 píxeles, con pesos de 1/9.

III.3.5) Rutinas para despliegue e impresión.

Los equipos especiales para despliegue de imágenes digitales generalmente están acompañados de un conjunto de programas para realizar despliegues en pseudo-colores o colores falsos, recuadros, expansiones, etc. En la mayoría de los casos, estas funciones son más que suficientes.

Una tendencia actual es la proliferación de microcomputadoras en muchos campos. Las microcomputadoras modernas generalmente están equipadas con monitores de alta resolución. En el futuro, será posible configurar un sistema eficiente de procesamiento de imágenes en torno a una microcomputadora.

Las rutinas de despliegue que se discuten a continuación y se incluyen en el apéndice A, fueron diseñadas para una computadora personal con tarjeta de gráficos tipo VGA. Los compiladores del lenguaje C para PC's ofrecen una serie de funciones primitivas para pintar un píxel, modificar la paleta de colores, etc.

La rutina *grises* cambia la paleta original de 16 colores por una escala uniforme con igual número de niveles de gris. Emplea algunas funciones definidas en el compilador de C para PC's llamado "Turbo C", versión 2.0.

La rutina *pseudo_colores* cambia la escala de colores original por una escala aleatoria de colores que conservan la misma relación que los colores de la escala original (cambian gradualmente desde el color más oscuro hasta el más claro).

La rutina *falsos_colores* reemplaza la escala de colores original por otra escala aleatoria diseñada de tal forma que los colores que la conforman resalten a la vista. Esto permite apreciar detalles que el ojo humano no alcanza a percibir en un despliegue convencional.

La evolución de las microcomputadoras se ha visto acompañada del vertiginoso desarrollo de sus dispositivos periféricos. Las

impresoras, en particular, han alcanzado una gran calidad de impresión con altas velocidades.

Todos los tipos de impresoras comerciales se acompañan de un manual de operación, en el que se explica la manera de diseñar nuevos caracteres de impresión. Es posible, en particular, crear un conjunto de caracteres que asemeje una escala de grises. La forma de crear estos caracteres se trata en la sección anterior de este capítulo.

El cuerpo de un programa de impresión de imágenes sigue, en general, la secuencia mostrada en la rutina imagen (apéndice A).

III.3.6) Una alternativa en el dominio de la frecuencia.

Como hemos podido constatar, los tiempos de proceso que se emplean efectuando una operación en el dominio de la frecuencia generalmente son mayores que los empleados en el dominio del espacio. Los procesos en el dominio de la frecuencia generalmente se reservan para el caso en que se cuenta con un procesador especial para realizar transformadas de Fourier.

Si los recursos se limitan a una computadora pequeña, las técnicas en el dominio de la frecuencia pierden parte de su atractivo.

En el procesamiento de señales, particularmente de imágenes, ha empezado a ganar popularidad una transformada alternativa a la de Fourier, conocida como transformada de Hartley.

La transformada discreta de Hartley (TDH) se asemeja a la de Fourier, pero está libre de algunas características indeseables computacionalmente [16].

La transformada discreta de Hartley toma valores reales y no complejos, evitando el manejo de la aritmética compleja o el tratamiento por separado de las partes real e imaginaria.

En la mayoría de las aplicaciones del procesamiento de imágenes, la convolución de dos secuencias de datos f_1 y f_2 está dada por TDH de $[(TDH \text{ de } f_1) \times (TDH \text{ de } f_2)]$. Esto asegura un cómputo más rápido que con la TDF, pues la velocidad de la transformada de Fourier depende del número de multiplicaciones y

una multiplicación compleja equivale a cuatro multiplicaciones reales.

Por otra parte, los arreglos reales requieren únicamente la mitad del espacio de almacenamiento de los arreglos complejos. Esto significa que la transformada de Hartley requerirá considerablemente menos memoria que la transformada de Fourier.

La forma analítica de la transformada de Hartley queda expresada por la ecuación (46). La ecuación (47) muestra la transformada inversa.

$$H(f) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} x(t) \text{cas}(2\pi ft) dt \quad (46)$$

$$X(t) = \int_{-\infty}^{+\infty} H(f) \text{cas}(2\pi ft) df \quad (47)$$

donde $\text{cas}(2\pi ft) = \cos(2\pi ft) + \text{sen}(2\pi ft)$.

La función $\text{cas}(2\pi ft)$ fue introducida por R. V. L. Hartley, quien propuso originalmente la transformada de Hartley en 1942.

La forma discreta de la transformada de Hartley es la siguiente:

$$H(f) = \frac{1}{N} \sum_{t=0}^{N-1} F(t) \text{cas}(2\pi ft/N) \quad (48)$$

y

$$X(t) = \sum_{f=0}^{N-1} H(f) \text{cas}(2\pi ft/N) \quad (49)$$

Como se aprecia en las ecuaciones (48) y (49), el cálculo de la transformada discreta de Hartley es un problema similar al de la transformada discreta de Fourier.

En esencia, un algoritmo de transformada rápida de Fourier (FFT) emplea un proceso de permutación para bisectar los datos

hasta conseguir parejas de datos. Calcular la transformada de Fourier de dichos pares es un proceso sencillo y rápido.

La idea que encierra el proceso de permutación es que es más rápido descomponer los datos en parejas, calcular la transformada de las parejas y recombinar éstas para formar la transformada completa que calcular la transformada para el conjunto total de datos.

La permutación es particularmente rápida cuando el número de datos es muy grande. Si se superponen todas las parejas de datos empleando un proceso conocido como "mariposa", se puede calcular la transformada de Fourier del conjunto de datos de entrada.

Bracewell [17] ha demostrado que es posible emplear una metodología similar en el caso de la transformada de Hartley. Una vez más, se emplea un proceso de permutación para bisectar los datos hasta conseguir parejas.

La transformada de Hartley de una pareja de datos (a,b) es $1/2(a+b, a-b)$, y su cómputo es trivial. También es posible superponer estas parejas para calcular la transformada de Hartley del conjunto de datos de entrada.

De esta forma, la transformada rápida de Hartley puede emplearse para muchas de las aplicaciones para las cuales comúnmente se utiliza la transformada de Fourier. Esto incluye procesos de convolución y deconvolución, así como la generación de espectros de potencia para aplicar filtros digitales. Además se puede obtener la transformada de Fourier a partir de la transformada de Hartley. De hecho, en ocasiones es más rápido generar la transformada de Fourier y el espectro de potencia con la transformada rápida de Hartley que con la FFT, debido a que el proceso de mariposa empleando números reales en lugar de números complejos requiere una menor cantidad de operaciones con punto flotante. Las partes real e imaginaria de la FFT se pueden abotener al final del cómputo, empleando las ecuaciones

$$F_r = H(f) + H(N-f)$$

y

(50)

$$F_{im} = H(f) - H(N-f),$$

donde F_r y F_{im} son las partes real e imaginaria de la transformada de Fourier compleja. N es el número de elementos del conjunto de datos.

El espectro de potencia se puede obtener directamente a partir de la transformada de Hartley empleando la ecuación

$$P_s(f) = [H(f)^2 + H(N-f)^2] / 2 \quad (51)$$

donde P_s es el espectro de potencia.

En el apéndice A se incluye el programa `hartley.pas`, tomado de la referencia [16], que sirve para calcular la transformada discreta de Hartley.

Esta rutina funciona sin modificación alguna en micros tipo PC (Macintosh e IBM). Como un indicador de su desempeño, el tiempo que emplea en una computadora IBM modelo PS-60 dotada con coprocesador numérico, para un arreglo de 256 X 256, es de 6 minutos y 10 segundos aproximadamente. Si se compara esto con los 12 minutos que aproximadamente emplea la transformada de Fourier optimizada que contiene el paquete VICAR en su versión para PC, se puede considerar como un buen comienzo.

CAPIULO IV APLICACIONES EN GEOFISICA

Desde finales de la década de los 70's la exploración geofísica ha venido incorporando a sus métodos de aplicación el procesamiento digital de imágenes.

Los avances que en esta materia utiliza la exploración geofísica moderna fueron desarrollados inicialmente en el ámbito de la percepción remota. Sin embargo, en la actualidad los datos gravimétricos, magnetométricos y sísmicos también son analizados a través de imágenes digitales que muestran su comportamiento y facilitan en gran medida su interpretación.

La percepción remota encuentra aplicaciones en muchas áreas distintas. Entre otras, podemos citar las siguientes: geotermia, geohidrología, agricultura, silvicultura, meteorología, geografía, oceanografía, etc.

En particular, la percepción remota ha causado un gran impacto en la exploración geológica. El trabajo de investigación geológica puede dividirse en dos grandes categorías [18]: la primera, que concierne esencialmente a fenómenos estáticos, como lo es la distribución, carácter y estructura de cuerpos rocosos, geoformas, litologías y fallas estructurales, y la segunda que tiene que ver con fenómenos dinámicos como el vulcanismo, las modificaciones en la plataforma continental y la sismología.

En las observaciones de fenómenos estáticos, uno de los atributos de mayor valor de los datos de sensores remotos se deriva simplemente de la gran distancia desde la cual es observada la Tierra, con la consecuente visión sinóptica del área cubierta por una sola observación. En geología es particularmente importante la visión de conjunto, pues las geoformas, tales como las cadenas montañosas, los valles, las cuencas o las mesetas, están estrechamente interrelacionadas y su extensión puede abarcar una gran área.

Diversos elementos estructurales dentro de los confines de un área pequeña, pueden revelarse como lineamientos de extensión

regional, además de que las unidades rocosas prominentes pueden ser trazadas en forma más precisa que su reconocimiento original.

En zonas donde la cubierta vegetal es escasa y permite apreciar directamente las estructuras geológicas, los rasgos estructurales pueden ser observados a simple vista en una impresión fotográfica de una imagen de satélite. En regiones cubiertas por vegetación densa, son más convenientes las imágenes de radar, puesto que, además de proporcionar una penetración adecuada de la vegetación, permiten establecer rasgos tales como los patrones de textura del terreno (18).

En resumen, las observaciones desde el espacio de fenómenos estáticos tienen dos aplicaciones prácticas de consecuencias inmediatas: una mejoría en la cartografía geológica y la planeación más eficiente de los recursos en la exploración de campo.

Aunque los sensores remotos rara vez conducen de por sí a una localización directa de fuentes de recursos, como minerales o petróleo, permiten establecer zonas de máxima probabilidad de existencia de los mencionados recursos, reduciendo con esto el universo de búsqueda y abaratando en consecuencia los gastos de exploración.

La segunda gran área de aplicaciones geológicas está relacionada con fenómenos dinámicos, en donde el valor particular de las imágenes de satélite radica en la capacidad de hacer observaciones repetitivas y de gran precisión que hacen posible la detección de cambios en el terreno, permitiendo su identificación y cuantificación.

Los datos obtenidos por satélites meteorológicos, más los datos geohidrológicos obtenidos por satélites de recursos naturales, conjuntamente con los de contaminación ambiental y de gases atmosféricos y los de corrientes y vientos marítimos proporcionados por satélites oceanográficos, permiten en conjunto elaborar modelos climatológicos muy complejos y a gran escala. De esta forma es posible estudiar fenómenos globales del terreno como son la desertificación y la erosión.

Dos aspectos que han recibido atención en los últimos años son la geodesia y la sismología. En lo que se refiere a la

geodesia, con la utilización de las señales de varios satélites y de un conjunto de estaciones receptoras es posible obtener las coordenadas geográficas de un punto del terreno, así como su elevación sobre el nivel del mar, con una precisión muy alta. Con estos datos se construye una imagen digital de la topografía de la superficie terrestre. Esta imagen topográfica se combina con imágenes multispectrales o con imágenes de radar, con lo que se obtienen imágenes con dimensión estereoscópica útiles en la elaboración y manipulación de información cartográfica y geográfica y también en la obtención de mapas temáticos relativos a los recursos naturales de una región determinada.

Se sabe que los desplazamientos que sufren los puntos que se encuentran a lo largo de fallas y zonas de subducción constituyen indicadores de actividad sísmica. Contar con la variación temporal de la posición de puntos selectos en las zonas de riesgo sísmico puede permitir correlacionar dicha variación con la actividad sísmica del lugar.

El carácter digital de las mayoría de las imágenes que se obtienen en la actualidad con los sensores remotos, ha permitido alcanzar resultados sorprendentes en las aplicaciones descritas.

Es común encontrar en las mejores publicaciones geofísicas y geológicas un buen número de trabajos que aplican la información obtenida a través de sensores remotos. Estos trabajos se acompañan de imágenes digitales en las que se observan rasgos de interés con extraordinaria claridad, gracias al procesamiento previo al que son sometidas.

Las técnicas básicas que hemos tratado en los tres capítulos anteriores son las herramientas que utilizan los expertos geólogos y geofísicos para analizar las imágenes de percepción remota. Las modernas técnicas para el análisis semi-automático de imágenes no son más que una combinación de estas técnicas básicas con procesos de reconocimiento de patrones.

En la siguiente sección de este capítulo se muestra la forma sencilla en que se aplican los procesos estudiados para mejorar en distintos grados una imagen, facilitando con ello su análisis e interpretación.

IV.1) Procesos en imágenes de percepción remota.

En cierta medida, las aplicaciones más sofisticadas en el campo de la percepción remota involucran la utilización del procesamiento digital de imágenes.

La técnica más frecuentemente utilizada en percepción remota es el despliegue de la imagen empleando pseudocolores o colores falsos. Este procedimiento se ilustra en la imagen 1, que es una imagen captada por un satélite meteorológico en la que se observa al huracán Gilberto en los momentos en que arremetía contra el territorio nacional. Este tipo de imágenes, además de servir para realizar los pronósticos sobre el estado del tiempo que cotidianamente aparecen en los periódicos, también se utilizan para formular complicados modelos climatológicos que en el futuro podrían aplicarse en la prevención de catástrofes como la provocada por Gilberto.



Imagen 1. Despliegue en colores falsos.

Existen otro tipo de aplicaciones, como la exploración geológica, en las que se requiere sacar el mayor provecho posible a la imagen captada por el sensor remoto. Para ello es necesario combinar algunos de los procesos que han sido discutidos a lo largo de este trabajo.

La utilidad principal de las imágenes en relación con la geología se da en la detección de rasgos lineales en la superficie de la tierra. Estos se reconocen directamente como líneas bien definidas entre superficies contrastadas totalmente, o bien por el alineamiento de rasgos topográficos o hidrográficos [19].

Algunos de estos lineamientos pueden ser apreciados directamente en la imagen, sin necesidad de aplicarle ningún tipo de operación. Sin embargo, en ocasiones existe información que no es tan evidente y debe, en consecuencia, ser realizada.

La secuencia de imágenes 2-8 ejemplifica el problema de la utilización de imágenes de satélite como herramienta auxiliar en los trabajos de geología regional.



Imagen 2. Aplicaciones en trabajos de Geología regional.

La imagen 2 es una imagen LANDSAT en proceso, que cubre una región de 30,000 km² aproximadamente y en ella quedan comprendidas, entre otras, las ciudades de Monterrey, Saltillo y Monclova. En ella es posible reconocer con cierta claridad una serie de lineamientos y algunas estructuras como pliegues, fallas y fracturas [19].

La imagen 3 es el resultado de aplicar a la imagen 2 la técnica de realce denominada reescalamiento de los niveles de gris, que se discute en el primer capítulo.

Si se observa detenidamente la imagen 2, se puede apreciar que la mayoría de los niveles de gris de los píxeles que la componen se encuentran en la porción superior de la escala de grises, es decir, tienden hacia el blanco. Esto se hace más evidente al analizar su histograma. Cuando se distribuyen uniformemente las intensidades de los píxeles en la escala de grises completa, se consigue elevar el contraste global. Esto facilita en gran medida la identificación de detalles en la imagen.



Imagen 3. Reescalamiento de los niveles de gris.

En la imagen 4 se han aplicado, además, los procedimientos de realce general y realce de bordes. Estos procedimientos se introducen en el primer capítulo. El algoritmo empleado es el mismo que se presenta en la función realce del capítulo tres. Con estos procesos se ha conseguido enfatizar un conjunto de rasgos apenas perceptibles en la imagen 2.



Imagen 4. Realce general y de bordes.

En la imagen 5 se ha sustituido la escala de grises original por una pseudo escala de colores. Este artificio puede ser de gran utilidad, pues permite distinguir muchos otros detalles que en el despliegue original no es posible.

Si nos interesa analizar las características de una porción relativamente pequeña dentro del área total que comprende una imagen de satélite, resulta adecuado contar con programas que permitan aumentar en un cierto factor dicha porción de la imagen, tanto en la escala horizontal como en la vertical.

Generalmente, los equipos de despliegue comerciales cuentan con este tipo de programas. Sin embargo, la expansión de la imagen

se realiza desplegando varias veces el valor de cada pixel de la imagen original. Si el factor de expansión es muy grande, entonces la imagen pierde detalle y aparece el llamado efecto de "parches".



Imagen 5. Despliegue en pseudo-colores.

Un procedimiento frecuentemente empleado es el de suavizar la imagen, con alguna de las técnicas tratadas en este trabajo, como etapa posterior a la expansión.

Un método más efectivo consiste en interpolar el valor de los pixeles que deberán componer, junto con los pixeles de la imagen original, la imagen expandida. Existen varios métodos de interpolación y casi cualquiera de ellos arroja buenos resultados para factores de expansión pequeños.

Cuando el factor de expansión es grande, la imagen interpolada comienza a mostrar los defectos de una imagen expandida por duplicación de pixeles: bordes poco agudos, aparición de "parches" en zonas de la imagen, etc.

Es posible evitar estos defectos aplicando procesos de realce y restauración antes de cada expansión. De esta forma, se consigue un procedimiento iterativo que es comparable, en cuanto a

efectividad, con los procedimientos más poderosos basados exclusivamente en el proceso de interpolación.

La imagen 6 se obtuvo como producto de expandir en un factor de 4 (tanto en la escala vertical como en la horizontal), por duplicación de píxeles, la porción central de la imagen 4.

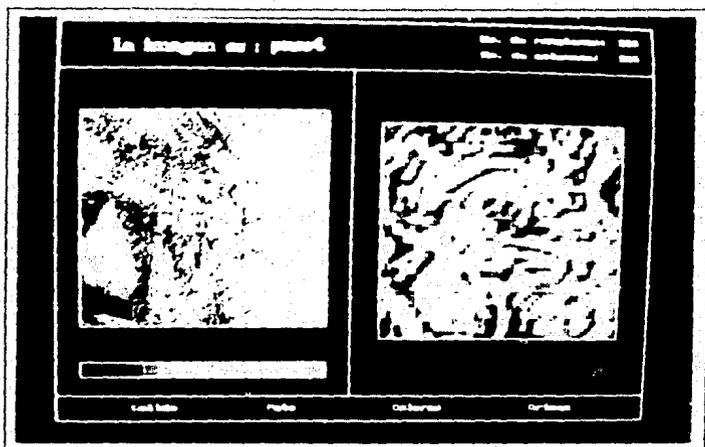


Imagen 6. Porción central de la imagen 5 ampliada cuatro veces duplicando píxeles.

La última de las imágenes que presentamos en este trabajo es el resultado de ampliar la porción central de la imagen 5, combinando un algoritmo de interpolación y otro de restauración.

El primer paso consiste en interpolar la porción original de la imagen para duplicar sus dimensiones. En este caso, la técnica de interpolación utilizada fue la denominada interpolación *bilineal*. Esta técnica consiste simplemente en tomar los valores de los píxeles más próximos al pixel que deseamos interpolar y efectuar un promedio ponderado de ellos (los pesos dependen de la cercanía relativa de cada pixel) para obtener el valor de este último.

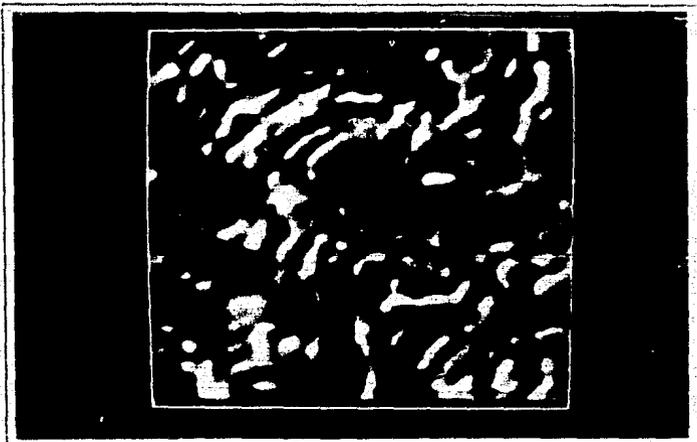


Imagen 7. Ampliación de la porción central de la imagen 5. El algoritmo empleado es una combinación de dos técnicas de interpolación y restauración.

La segunda etapa es restaurar la imagen suponiendo que ha sido afectada por un proceso de degradación en el cual cada uno de los píxeles (de la imagen ideal) se ha diluido para dar origen a cuatro píxeles de la imagen actual. Utilizando este modelo aplicamos el algoritmo de aproximaciones sucesivas.

En seguida se procede a duplicar la imagen restaurada, para posteriormente aplicarle el algoritmo de aproximaciones sucesivas.

Al comparar las imágenes 6 y 7, se aprecia que existen muchos detalles en la imagen 7 que se ven distorsionados en la imagen 6.

Cuando el factor de expansión aumenta, las diferencias entre los dos métodos de ampliación descritos se vuelven más evidentes.

CONCLUSIONES

El análisis digital de imágenes ha demostrado su utilidad en muchos campos del conocimiento. En nuestro país, diversos grupos de investigación lo han aplicado con éxito en la solución de problemas de física, química, astronomía, ingeniería, etc. Sin embargo, fuera de los centros de investigación, esta experiencia pocas veces ha sido aprovechada.

En algunos países, una experiencia similar ha servido para conseguir aplicaciones tecnológicas que alcanzan muchas esferas de la vida.

De particular importancia son los cambios que actualmente se gestan debido al empleo de sensores remotos de alta eficiencia.

Existen una serie de barreras en nuestro país que dificultan el acceso a las ventajas que ofrece la moderna ciencia de la percepción remota. Algunas de ellas, dadas las condiciones imperantes, parecen imposibles de superar. Sin embargo, un esfuerzo considerable debe ser invertido para la creación de los recursos técnicos y humanos que permitan absorber la alta tecnología que implica el trabajo en este campo.

Hemos podido observar últimamente una serie de espectaculares aplicaciones de la percepción remota cuando las sondas espaciales nos han hecho llegar imágenes de las partes más distantes del sistema solar. Las imágenes enviadas han aportado una cantidad mucho mayor de datos de los que se habían recolectado con los telescopios y estaciones terrenas. Todo esto ha ayudado enormemente a la comprensión de la creación y evolución del sistema solar y en particular nos ha permitido entender con más detalle la estructura de nuestro planeta.

No existen elementos que permitan afirmar que nuestro país, en un futuro cercano, se incorporará al círculo selecto de las naciones que generan y aprovechan los avances tecnológicos en el campo de la percepción remota. Pero esto no significa que no exista la posibilidad de incorporar algunos conocimientos básicos, generados en este campo, en la solución de problemas que hasta hoy

son tratados con los métodos tradicionales.

En las ciencias de la tierra aún se tiene un largo camino por recorrer. Antes que pensar en emprender el diseño de modernas tecnologías para la prospección minera o petrolera, se debe realizar una labor de convencimiento que muestre a los profesionales de este campo las ventajas que ofrece el empleo de los sensores remotos.

Paralelamente, deben crearse las condiciones para que los avances recientes en percepción remota sean incorporados a la brevedad posible en los programas de estudio que actualmente rigen la formación de los futuros profesionales de las ciencias de la tierra.

La percepción remota, como casi cualquier otra actividad científica, requiere del concurso de especialistas de muy diversas áreas. Es una moderna ciencia en rápida expansión que no admite improvisaciones, pues requiere de una extensiva investigación y desarrollo de nuevos métodos y técnicas.

En nuestro país se empieza a observar una cierta tendencia a enfrentar, con cierta autonomía, algunos problemas que involucran tecnologías de punta.

Creemos que este trabajo, al exhibir el estado actual de las técnicas de análisis digital de imágenes que se aplican en percepción remota, contribuye en cierta forma a acentuar dicha tendencia.

BIBLIOGRAFIA

1. Rosenfeld A. y Kak, A. C., *Digital Picture Processing*, Academic Press, New York, 1976.
2. Gonzalez, R. C. y Wintz, P., *Digital Image Processing*, Addison-Wesley, Reading, Mass., 1988.
3. Wang, D. C. y Li, C. C., *Digital Image Enhancement: A Survey*, *Computer Graphics and Image Processing*, 4, 1984, 363-381.
4. Graham, R. E. Snow removal- A noise stripping process for picture signals, *IRE Trans. Inf. Theor.*, IT-8, 1962, 129.
5. Brown, D. W., Digital computer analysis and display of the radionuclide scan, *J. Nucl. Med.* 7, 1966, 740.
6. Tukey, J. W., *Exploratory Data Analysis*, Addison-Wesley, Reading, Mass., 1977.
7. Pratt, W. K., *Digital Image Processing*, Wiley, New York, 1978.
8. Wallis, R. H., An approach for the space variant restoration and enhancement of images, *Proceedings, Symposium on Current Math. Prob. in Image Science*, Monterrey, Calif., 1976.
9. Duda, R. O. y Hart, P. E., *Pattern Clasification and Scene Analysis*, Wiley, New York, 1973.
10. Hall, E., Almost uniform distribution for computer image enhancement, *IEEE Trans. Comput.* C-23, 1974, 207.
11. Andrews, H. C., Monochrome digital image enhancement, *Appl. Opt.* 15, 1976, 495.
12. Andrews, H. C. and Hunt, B. R., *Digital Image Restoration*, Prentice-Hall, New Jersey, 1977.
13. Richards, J. A., *Remote Sensing Digital Image Analysis*, Springer-Verlag, New York, 1986.
14. Gourlay, A. R., Grayscale simulation on a color display. *Computer Graphics and Image Processing*, 9, 1984, 92-96.
15. Chávez, M., *Generación de imágenes mediante impresión raster*, *Comunicaciones técnicas*, IGFUNAM, México, 1988.
16. O'Neill, M. A., Faster than fast Fourier, *BYTE*, 4, 1988, 293-300.

17. Bracewell, R. N., *The fast Hartley Transform*, Oxford University Press, New York, 1986.
18. Lira, J., *Percepción Remota*, Fondo de Cultura Económica, México, 1988.
19. Mitre, L. M., *Las imágenes Landsat - una herramienta útil en la interpretación Geológico-Estructural; un ejemplo en el noreste de México*, *IGUNAM, Revista*, vol. 5 núm. 1, 1981, 37-46.

APÉNDICE A

BIBLIOTECA DE FUNCIONES.

Las funciones que se presentan en este apéndice se comentan a lo largo del capítulo III. Emplean un conjunto de variables, que a continuación se definen:

```
integer ren,col; /* número de renglones y columnas */
integer mat[ren][col], mat1[ren][col]; /* matrices de datos a la
                                     entrada y a la salida */
```

```
normal()
{
  int *pix,dim,n,t,s,nton;

  dim=ren*col;
  for (i=1;i<=dim;i++)
  {
    n=nton/(t-s);
    pix=pix*n;
    *pix++;
  }
}
```

```
gradiente();
{
  for (i=2;i<=ren-1;i++){
    for (j=2;j<=col-1;j++){
      d2x=abs(mat[i+1][j]-mat[i-1][j]);
      d2y=abs(mat[i][j+1]-mat[i][j-1]);
      dpos=abs(mat[i+1][j+1]-mat[i,j]);
      dneg=abs(mat[i][j+1]-mat[i+1][j]);
      grad1=d2y;
      if(d2x >= d2y) {
        grad1=d2x;}
      grad2=dneg;
      if(dpos >= dneg) {
        grad2=dpos}
      grad=grad2;
      if(grad1 >= grad2) {
        grad=grad1;}
      mat1[i][j]=grad;
    }
  }
}
```

```
laplaciano()
```

```
{  
  for (i=2;i<=ren-1;i++){  
    for (j=2;j<=col-1;j++){  
      lap[i][j]=(-4)*mat[i][j]+mat[i][j-1]+mat[i][j+1]  
                +mat[i-1][j]+mat[i+1][j];  
    }  
  }  
}
```

```
prom4()
```

```
{  
  for (i=2;i<=ren-1;i++){  
    for (j=2;j<=col-1;j++){  
      prom[i][j]=(mat[i][j]+mat[i][j-1]+mat[i+1][j-1]  
                  +mat[i+1][j])/4;  
    }  
  }  
}
```

```
prom9()
```

```
{  
  for (i=2;i<=ren-1;i++){  
    for (j=2;j<=col-1;j++){  
      prom[i][j]=(mat[i][j]+mat[i][j-1]+mat[i+1][j-1]  
                  +mat[i+1][j]+mat[i-1][j-1]+mat[i-1][j]  
                  -mat[i-1][j+1]+mat[i+1][j-1]+mat[i][j-1])/9;  
    }  
  }  
}
```

```
ordena()
```

```
{  
  /* Esta función es empleada por el procedimiento  
  mediana */  
  
  n=0;  
  for (i=1;i<=9;i++){  
    for (o=1;o<=8;o++){  
      if (p[o] > p[o+1])  
      {  
        x=p[o+1];  
        p[o+1]=p[o];  
        p[o]=x;  
      }  
    }  
    n=n+1;  
  }  
}
```

```

mediana()
{
  for (i=2;i<=ren-1;i++){
    for (j=2;j<=col-1;j++){
      p[1]=m[i-1][j-1];
      p[2]=m[i-1][j];
      p[3]=m[i-1][j+1];
      p[4]=m[i][j-1];
      p[5]=m[i][j];
      p[6]=m[i][j+1];
      p[7]=m[i+1][j-1];
      p[8]=m[i+1][j];
      p[9]=m[i+1][j+1];
      ordena();
      mat[i][j]=p[5];
    }
  }
}

realce()
{
  for (i=2;i<=ren-1;i++){
    for (j=1;j<=col-1;j++){
      rest[i][j]=mat[2][j]-(-4*mat[2][j]+mat[2][j-1]
      +mat[2][j+1]+mat[1][j]+mat[3][j])*factor;
    }
  }
}

restaural()
{
  for (i=2;i<=ren-1;i++){
    for (j=2;j<=col-1;j++){
      van[i][j]=2*mat[i][j]-(mat[i][j]+mat[i][j+1]
      +mat[i+1,j+1]+mat[i+1,j]) / 4;
    }
  }
}

restaura2()
{
  for (i=2;i<=ren-1;i++){
    for (j=2;j<=col-1;j++){
      van2[i][j]=2*mat[i][j]-(mat[i][j]+mat[i][j+1]
      +mat[i+1][j+1]+mat[i+1][j]+mat[i-1][j-1]
      +mat[i-1][j]+mat[i-1][j+1]+mat[i+1][j-1]
      +mat[i][j-1]) / 9;
    }
  }
}

```

RUTINAS PARA DESPLIEGUE E IMPRESIÓN.

Las rutinas para despliegue e impresión de imágenes que se presentan en este apéndice, son comentadas ampliamente en el capítulo III.

La rutina de despliegue emplea las funciones de gráficos con que cuenta el compilador Turbo C, versión 2.0. En particular, esta rutina fue diseñada para manejar 16 colores simultáneos en el despliegue, por lo cual, para poder utilizarla sin modificación alguna, se debe contar con una tarjeta de gráficos que proporcione dicho número de colores (EGA, VGA, MCGA, etc.).

despliega()

```
{
    /* Realiza el despliegue de la imagen */
    unsigned int a,b,x1,y1,color;

    for (j = 1; j <= renglon; j++)
    {
        buffer[j] = (char*) malloc (columna);
        fread(buffer[j],columna,1,archivo);
        for (k = 1; k <= columna ; k++)
        {
            color = (*buffer[j] - 128) / 8;
            tg[color] = tg[color] + 1;
            putpixel(k,j,color);
            buffer[j]++;
        }
    }
}
```

grises()

```
{
    /* Carga la paleta con la escala de grises */

    for ( i = 1 ; i <= 5 ; i++)
    {
        setrgbpalette(i,i*4,i*4,i*4);
    }
    for (i = 56; i <= 62; i++)
    {
        setrgbpalette(i,(i-48)*4,(i-48)*4,(i-48)*4);
    }
    setrgbpalette(20,24,24,24);
    setrgbpalette(7,28,28,28);
}
```

```

falsos_colores()
{
    /* genera colores aleatorios y reemplaza. con ellos,
    la paleta de colores original */
    randomize();
    for (i=1;i<=5;i++)
    {
        setrgbpalette(i,random(64),random(64),random(64));
    }
    for (i = 56; i <= 62; i++)
    {
        setrgbpalette(i,random(64),random(64),random(64));
    }
    setrgbpalette(20,random(64),random(64),random(64));
    setrgbpalette(7,random(64),random(64),random(64));
}

pseudo_colores()
{
    /* genera una paleta de colores aleatorios que guardan
    una relación de escala. Esto es, los primeros colores
    de la escala son los más oscuros y los últimos son
    los más claros */
    randomize();
    for (i=1;i<=5;i++)
    {
        setrgbpalette(i,random(i*4+4),random(i*4+4)
        ,random(i*4+4));
    }
    for (i = 56; i <= 62; i++)
    {
        setrgbpalette(i,random((i-47)*4),random((i-47)*4)
        ,random((i-47)*4));
    }
    setrgbpalette(20,random(7*4),random(7*4),random(7*4));
    setrgbpalette(7,random(8*4),random(8*4),random(8*4));
}

```

La siguiente rutina muestra el algoritmo general para diseñar un programa de impresión de imágenes para una impresora de matriz convencional. Los parámetros que ocupa dependen de las características particulares de la impresora. Generalmente, estos parámetros pueden encontrarse en los manuales de empleo que acompañan a dicha impresora.

```

Procedure imagen();
begin
  inicia;
  { Se manda a la impresora la secuencia de escape que indica
    el inicio de la impresión y la resolución que se desea.
  }
  for i:=1 to ren do
    begin
      for k:=1 to dim
        { De acuerdo a las dimensiones de la matriz de puntos,
          se mandan una cantidad dim de números a la impresora. }
        begin
          num_bytes;
          { Se indica a la impresora el número de bytes que se
            le van a transmitir. }
          for j:=1 to col do
            transmite;
            { Se transmiten el número de bytes indicados, que
              representan el patrón de gris correspondiente. }
          end;
        end;
      final;
      { Se indica el final de la impresión. La secuencia que
        debe ser enviada se puede encontrar en el manual de
        la impresora. }
    end.
end.

```

TRANSFORMADA DE HARTLEY (FHT).

El siguiente procedimiento permite calcular la transformada de Hartley de un vector de datos. Puede encontrarse una explicación detallada sobre su funcionamiento en el capítulo III.

```

procedure fht (var data_array:data_array_type;
              power_index, syze:integer; transform:direction_type);

var
  i, j, k, trg_ind, trg_inc, power, t_a, f_a, i_temp,
  section, s_start, s_end:integer;
  sne, csn:array[1..datasize] of single;
  accu:array[1..2, 1..datasize] of single;

function permute(index:integer):integer;

var
  i, j, s:integer;

```

```

begin
  j:=0;
  index:=index-1;
  for i:=1 to power_index do
    begin
      s:=-index div 2;
      j:=j + 1 + index - s - s ;
      index:=s;
    end;
  permute:=j+1;
end;

procedure trig_table(npts:integer);

const
  pi=3.14159265;

var
  i:integer;
  angle,omega:single;
begin
  angle:=0;
  omega:=2 * pi/npts;
  for i:=1 to npts do
    begin
      sne[i]:=sin (angle);
      csn[i]:=cos (angle);
      angle:=angle+omega;
    end;
  end;

function modify(power,s_start,s_end,index:integer):integer;

begin
  if(s_start=index) or (power < 3) then
    modify:= index
  else
    modify:= s_start + s_end - index + 1;
  end;

procedure butterfly(trig_ind,i_1,i_2,i_3:integer);

begin
  accu[t_a, i_1]:=accu[f_a,i_1]+accu[f_a,i_2]*csn[trig_ind]+
    accu[f_a, i_3] * sne[trig_ind];
  trig_ind:=trig_ind + syze div 2;
  accu[t_a, i_2]:=accu[f_a,i_1]+accu[f_a,i_2]*csn[trig_ind]+
    accu[f_a,i_3]*sne[trig_ind];
end;

```

```

begin
power:=1;
f_a:=1;
t_a:=2;
trig_table(syze);
for i:=1 to syze do
accu[f_a, permute(i)]:=data_array[i];
for i:=1 to power_index do
begin
j:=1;
section:=1;
trg_inc:=syze div (power + power);
repeat
trg_ind:=1;
s_start:=section * power + 1;
s_end:=(section - 1) * power;
for k:=1 to power do
begin
butterfly(trg_ind, j, j+power,
modify(power, s_start, s_end, j+power));
trg_ind:=trg_ind + trg_inc;
j:=j+1;
end;
j:=j+power;
section:=section-2;
until j>syze;
power:=power + power;
i_temp:=t_a;
t_a:=f_a;
f_a:=i_temp;
end;
case transform_direction of
directa:
for i:=1 to syze do
data_array[i]:=accu[f_a, i]/syze;
inversa:
for i:=1 to syze do
data_array[i]:=accu[f_a, i];
end;
end;
end;

```