

162
2ij



**UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO**

FACULTAD DE INGENIERIA

IMPLANTACION DE TECNICAS HALFTONING PARA
EL MEJORAMIENTO DE LA CALIDAD DE
VISUALIZACION Y DE LA IMPRESION DE
IMAGENES BINIVEL

T E S I S

QUE PARA OBTENER EL TITULO DE
INGENIERO MECANICO ELECTRICISTA
(AREA SISTEMAS DIGITALES)

P R E S E N T A :

OSCAR EDUARDO SORIANO RAMIREZ



DIRECTOR: DR. ROGELIO ALCANTARA SILVA

MEXICO, D. F.

1996

**TESIS CON
FALLA DE ORIGEN**

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Dedicatoria

*Este trabajo esta dedicado a dos personas a las cuales quiero y
admiro mucho y porque les debo todo lo que soy.*

A mis padres.

Emma Ramirez

*Por ese gran corazón que siempre esta conmigo en todas partes
y por esa gran fuerza de voluntad que me apoya siempre.*

Ignacio Larios

*Porque siempre me ha enseñado el camino a seguir durante
toda mi vida y por ese gran respeto que siento hacia usted.*

gracias.

Agradecimientos

A la Universidad Nacional Autónoma de México

Al Doctor Rogelio Alcántara Silva por su colaboración para este
proyecto

A la Fundación UNAM

A mis hermanos:

Rosa, Angel, Abel, Susi y Tere por todo su apoyo.

Indice

Capítulo 1

Introducción

1.1 Digitalización de Documentos	2
1.2 Conceptos Básicos	4
1.2.1 Imagen Digital	4
1.2.2 Relaciones entre Pixeles	4
1.2.3 Operaciones Básicas	5
1.2.3.1 Operaciones de Puntos	5
1.2.3.2 Operaciones Locales	5
1.3 Organización de la Tesis	6

Capítulo 2

Planeación y Análisis del Proyecto.

2.1 Análisis del Problema	7
2.1.1 Desplegado	7
2.1.2 Impresión	8
2.2 Estado del Arte	8
2.3 Que es Halftoning ?	8
2.3.1 Halftoning Puntos Dispersos	9
2.3.2 Halftoning Puntos Agrupados	9
2.4 Marco Teórico a Utilizar	9
2.5 Metodología de Investigación	10
2.6 Resultados Esperados	11

Capítulo 3

Análisis y Procesamiento de Imágenes Digitales Halftone

3.1 Modelo Visual	12
3.2 Sistema de Producción de Imágenes	13
3.3 Filtrado	15
3.3.1 Métodos en el Dominio Espacial	16
3.3.1.1 Filtros Derivativos	17
3.3.2 Métodos en el Dominio de Frecuencia	18
3.4 Ajuste de la Escala de Tono	18
3.4.1 Mejoramiento de Contraste	18
3.4.2 Mapeo de Brillantez	20
3.5 Halftoning	21

Capítulo 4

Métodos de Procesamiento para la Generación de Imágenes Halftone

4.1 Métodos de Procesamiento	22
4.1.1 Thresholding	22
4.1.2 Método de Bayer	23
4.1.3 Dither Ordenado	24
4.2 Métodos de Difusión de Error	28
4.2.1 Filtro de Floyd-Steinberg	29
4.2.2 Filtro Stucki	30
4.2.3 Filtro Burkes	30
4.3 Difusión de Error con Perturbación	31

Capítulo 5

Desplegado de Imágenes Halftone

5.1 El Adaptador Gráfico	32
5.2 Mapas de Memoria	35
5.3 Operación del Adaptador Gráfico	36
5.3.1 Memoria de Video Multiplano por Pixel	36
5.3.2 Memoria de Video Multibit por Pixel	36
5.4 Monitores	37
5.5 El lenguaje de Programación	38
5.5.1 Versatilidad	38
5.5.2 Potencia	38
5.5.3 Velocidad	39
5.5.4 Portabilidad de Programas en C	39
5.6 Desarrollo del Sistema de Desplegado	39

Capítulo 6

Impresión de Imágenes Halftone en Impresoras Blanco y Negro

6.1 Comunicación Computadora - Impresora	42
6.2 El Verdadero Halftone	43
6.3 Patterning	46
6.4 Dithering	48
6.5 Implementación de Algoritmos Halftoning para Impresión	49
6.5.1 Impresión en Matriz de Puntos	49
6.5.2 Impresión Laser	51

Capítulo 7

Codificación de Imágenes Halftone

7.1 Métodos de Compresión sin Pérdidas Lossless	54
7.2 Métodos de Compresión con Pérdidas Lossy	54
7.3 Codificación por Transformación	55
7.3.1 Transformada Coseno Discreta	57
7.4 Algoritmo JPEG	58
7.4.1 Descripción del Sistema JPEG	58
7.5 Implementación del Esquema de Codificación	60
7.6 Mapeo del Dominio Binario al Dominio de Tonos de Gris	61
7.6.1 Codificador no Lineal	61
7.7 Compresión de Imágenes en el Dominio binario	63
7.8 Compresión de Imágenes en el Dominio de Tonos de Gris	63

Capítulo 8

Resultados, Perspectivas y Conclusiones

8.1 Método de Bayer y Dither Ordenado	67
8.2 Métodos de Difusión de Error	69
8.3 Modificación de la Escala de Tono y Afinamiento	72
8.4 Desplegado	78
8.5 Impresión	80
8.6 Compresión	80
8.7 Conclusiones	87
8.8 Perspectivas	88

Apéndices

Apéndice A Códigos de Control FX80

Apéndice B Tipos de Impresoras

Glosario

Bibliografía

Capítulo 1

Introducción

Debido al gran avance tecnológico que se a venido dando en los últimos años en el campo de la computación, nuevas interfaces gráficas han permitido al hombre interactuar de forma mas directa con la computadora. Adaptando su ambiente fisico a su medio de trabajo. Ahora es posible utilizar además de texto y gráficas, imágenes, video y audio. Fomentando el desarrollo de dispositivos a fin de incrementar su capacidad en el manejo de la información. Se visualizan nuevas áreas en computación y electrónica enfocadas al manejo y mejor aprovechamiento de recursos, tanto fisicos (hardware) como lógicos (software).

Nuevos métodos de adquisición de datos han permitido capturar información de una forma mas directa, a través de dispositivos tales como: cámaras de video, scanners, digitalizadores etc. los cuales transforman la información visual en una representación digital. Tomografías Computarizadas e Imágenes de Resonancia Magnética son algunos ejemplos de información adquirida por este medio y han tenido gran aplicación en el campo de la medicina, siendo esta la única forma para su obtención.

Contando con la imagen en un formato digital es posible analizar, manipular y procesarla de tal forma que se resalten las características principales de interés, creando así una nueva área con amplias perspectivas de evolución como es el procesamiento de imágenes digitales.

Lo conveniente de un formato digital, es que una vez que se ha llevado a cabo un estudio detallado de la imagen en cuestión, es posible presentar los resultados por algún medio, despliegue y/o impresión, además de almacenar y transmitir de manera eficiente.

La representación de una imagen visual en una computadora es una secuencia de números sobre los cuales se aplican los algoritmos de procesamiento que los transforman, obteniendo datos que de nuevo son trasladados a una representación visual la que es interpretada por el usuario. A estos algoritmos de procesamiento se les conoce como *algoritmos visuales*.

1.1 Digitalización de Documentos.

Existen varios dispositivos para digitalizar documentos e imágenes los más comunes son los scanners, los hay de varios tamaños, precios y modelos, sin embargo todos ellos tienen los mismos componentes básicos.

Su parte central está compuesta por un dispositivo de carga acoplada (CCD). Los CCD usualmente consisten de un circuito semiconductor con un arreglo de elementos sensores de luz. Una lámpara ilumina la página durante el proceso y un conjunto de lentes reflejan y concentran la luz dentro del circuito. Como el sistema de alimentación principal varía, ocurren cambios en el patrón de luz reflejada. Los CCD envían un alto voltaje correspondiente a áreas blancas, un voltaje bajo correspondiente a áreas negras y niveles de voltaje intermedios para niveles de gris hacia un controlador.

Los valores producidos por los sensores son continuos en un rango finito y tienen una relación lineal con la intensidad de energía radiante de cada localidad muestreada. El proceso de conversión es llamado *digitalización*.

El esquema más común para realizar la digitalización es una malla rectangular mostrada en la figura 1.1. Idealmente cada muestra debiera corresponder a una región infinitesimalmente pequeña de la fuente, pero a consecuencia de la naturaleza física de los sensores y de su óptica asociada, cada muestra constituye el valor integrado de un área finita de la fuente. La imagen es dividida en líneas horizontales y hecha de píxeles adyacentes. A cada posición del píxel, la brillantez de la imagen es muestreada y cuantizada. Este paso genera un entero para cada píxel representando la brillantez de la imagen en ese punto. Cuando este procedimiento es realizado para todos los píxeles, la imagen es representada mediante un arreglo rectangular de enteros. Cada píxel tiene una posición o dirección y un valor entero llamado nivel de gris, [CAS89].

La digitalización de la imagen en coordenadas espaciales (x,y) es llamado *muestreo* de la imagen y la digitalización de amplitud es llamada *cuantización* de niveles de gris.

La calidad, o resolución de una imagen escaneada es medida en píxeles por pulgada (dpi). Utilizándose 100, 200, 300, 400 o más píxeles o puntos por pulgada; mientras más resolución, mejor calidad de la imagen.

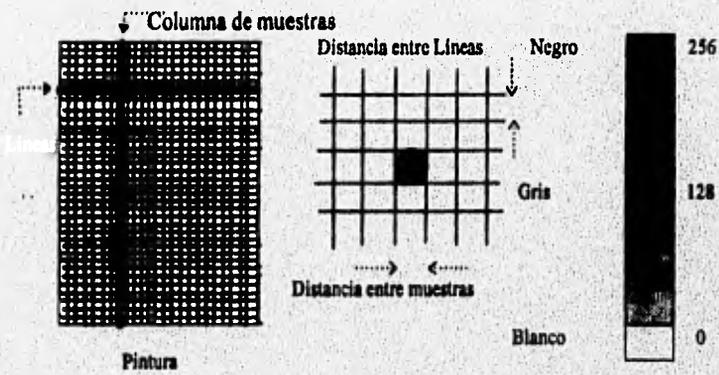


Figura 1.1 Muestreo y Cuantización de niveles de gris.

1.2 Conceptos Básicos

1.2.1 Imagen Digital

Una vez que se a discretizado una imagen en un formato digital, tanto en amplitud como espacialmente. Podemos definir un modelo matemático el cual describa el concepto de Imagen.

Una imagen es una función de intensidad de luz bidimensional, denotado como $f(x,y)$, donde el valor o amplitud de f en coordenadas espaciales (x,y) indica la intensidad (brillantez) de la imagen en ese punto, [GON92].

Así, consideraremos una imagen como un arreglo de elementos de tamaño $M \times N$ donde M es el número de renglones y N es el número de columnas y el valor de los elementos del arreglo corresponden a los niveles de gris.

1.2.2 Relaciones entre Píxeles

Muchos de los métodos de estudio están basados en términos de la posición de los elementos que constituyen la imagen, llamado análisis espacial.

Un pixel puede tener de 3 a 8 vecinos dependiendo de su ubicación en la imagen. Los vecinos principales de un pixel son los que lo rodean, generalmente se tienen dos vecinos horizontales, dos vecinos verticales y cuatro vecinos diagonales; la figura 1.2 muestra los ocho vecinos principales de un pixel, junto con sus coordenadas.

Es claro que en los bordes de la imagen el número de vecinos disminuye.

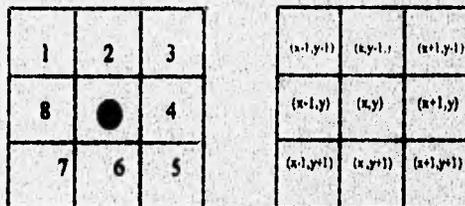


Figura 1.2 Vecinos de un píxel.

1.2.3 Operaciones Básicas

1.2.3.1 Operaciones de Puntos.

Operaciones de punto constituye una simple pero importante clase de procesamiento de imágenes. Estas admiten que el usuario modifique la forma en la cual sus datos cubren el rango disponible de niveles de gris. Una operación de punto es por definición, una operación que toma una imagen de entrada simple y produce una imagen de salida simple; de tal forma que cada pixel de salida depende únicamente del pixel de entrada correspondiente. Así, una operación de punto no puede modificar las relaciones espaciales dentro de una imagen. Una operación de punto puede ser vista como operaciones de copia de pixel a pixel, excepto que el pixel es modificado en una forma predeterminada. Una operación de punto la cual toma como entrada una imagen $A(x,y)$ y produce una imagen de salida $B(x,y)$ puede ser expresada matemáticamente como: $B(x,y) = f[A(x,y)]$. La operación de punto esta determinada por la función f la cual especifica el mapeo de niveles de entrada al de salida, [CAS89].

Para aplicaciones donde la minimización de tiempo de cálculo y/o hardware es una necesidad, entonces la operación de punto es preferida.

1.2.3.2 Operaciones Locales.

Cuando un proceso genera una imagen de salida a partir de una imagen de entrada, debe existir una correspondencia entre los pixeles de las dos imágenes. Cada pixel de la imagen de salida debe corresponder a un pixel de la imagen de entrada.

Una operación local, utiliza un vecindario de pixeles de entrada para determinar el valor de los pixeles de salida. Operaciones locales son también conocidas como *operaciones de vecindad*.

Operaciones de vecindad generalmente producen resultados de más alta calidad, ya que se hace referencia a la relación que existe entre un grupo de pixeles agrupados, sin embargo necesitan más tiempo de cálculo para sus operaciones.

1.3 Organización de la Tesis

El capítulo 2 describe el análisis y planteamiento del problema, la metodología a seguir para la elaboración del proyecto. El capítulo 3 se enfoca al Sistema de Producción de Imágenes, modelo con el cual se puede obtener imágenes digitales de alta calidad con únicamente elementos binarios. En el capítulo 4 se describe la base teórica para la generación e implementación de algoritmos Halftoning, haciendo una comparación para determinar principales ventajas y desventajas. Los capítulos 5 y 6 describen el procedimiento de despliegado de imágenes a través de diferentes medios, enfocándose en el capítulo 5 a monitores, y el capítulo 6 sobre impresoras, adaptando los métodos halftoning para los diferentes dispositivos. El capítulo 7 describe modelos de codificación de imágenes halftoning e imágenes binarias en general para lograr almacenamiento y transmisión óptimos. El capítulo 8 hace un análisis de los resultados presentados, perspectivas y conclusiones.

Capítulo 2

Planeación y Análisis del Proyecto.

2.1 Análisis del Problema.

2.1.1 Desplegado.

La calidad de una imagen vista a través de una computadora depende de la capacidad que el monitor pueda ofrecer; es decir, mientras nuestro equipo tenga mejor resolución y pueda reproducir color, nuestra imagen será de buena calidad; sin embargo, si nuestro equipo tiene un número limitado de color o bien un número limitado de niveles de gris, en el peor de los casos pensemos en un monitor monocromático, la calidad de nuestra imagen será deficiente.

El problema ha sido planteado, que hacer cuando se tiene un monitor binivel?

Para los dispositivos con un número limitado de niveles de gris, donde no se obtiene una buena aproximación de la imagen original, es necesario realizar una serie de operaciones para optimizar en gran medida la calidad de la imagen y corregir algunas deficiencias.

El enfoque que sigue nuestro proyecto es evaluar las capacidades que nuestro equipo presente y asociar un pre-procesador que transforme los datos de la imagen digital a una forma tal, que se adapte a las características peculiares del dispositivo.

Enfocaremos entonces el problema al desarrollo de algoritmos mas óptimos adecuados a los parámetros específicos del dispositivo de desplegado.

2.1.2 Impresión.

En el caso de la impresión se persiguen los mismos objetivos, cuando se tiene una impresora a color o bien una impresora laser con capacidad de producir niveles de gris la calidad de impresión de imágenes es buena, sin embargo el costo es muy elevado. Y generalmente a las impresoras a las cuales tenemos acceso es a las impresoras blanco y negro, su calidad de impresión en imágenes es mala, sin embargo aplicando nuestros algoritmos podemos aumentar la calidad de impresión de manera tal que la diferencia entre nuestra impresora blanco y negro y una impresora laser con capacidad de reproducir tonalidades de grises sea muy pequeña.

Entonces podremos imprimir imágenes de buena calidad a muy bajo costo.

2.2 Estado del Arte.

El origen de Halftoning data del siglo XVII donde los artesanos dibujaban sobre una placa de cobre una pintura que contenía una elaboración de patrones aleatorios que daban la ilusión de producir delicadas tonalidades de gris en la pintura, esta técnica fue llamada Mezzotint "media tinta". La etimología de esta palabra de origen italiano es la de "Halfone", una palabra usada para diferenciar la mecánica moderna de impresión de la del arte antiguo, [ULI87].

Esta técnica a sido empleada para imprimir imágenes en periódicos, libros y revistas. Tiene bases matemáticas que la sustentan y a pesar de ser una de las más antiguas hoy en día tiene muchas aplicaciones.

2.3 Que es Halftoning ?

Digital Halftoning conocido también como *Spatial Dithering* se refiere a algoritmos de procesamiento de imágenes los cuales crean la ilusión de tonos de gris mediante el juicioso arreglo de los pixeles que constituyen la imagen.

Algoritmos cuya base teórica describe el proceso de reconstrucción de la imagen a través de una secuencia de muestras tomadas de la imagen original.

Existen varias de estas técnicas las cuales se pueden enfocar a un problema distinto, a continuación describiremos algunas de ellas.

2.3.1 Halftoning Puntos Dispersos.

Se puede producir un nivel de gris cubriendo una pequeña área con cualquier agrupación o dispersión de puntos. Si un dispositivo de despliegue puede acomodar satisfactoriamente un pixel aislado blanco o negro, entonces, halftoning puntos dispersos es la opción indicada, el cual maximiza el uso de la resolución.

2.3.2 Halftoning Puntos Agrupados.

Una agrupación de puntos halftoning imita el proceso de fotograbado usado en impresiones donde los pixeles se comprimen colectivamente en puntos de varios tamaños.

2.4 Marco Teórico a Utilizar.

El desarrollo del proyecto se basa en el estudio de los Algoritmos Halftoning existentes, su fundamento matemático, las diferencias entre cada uno de ellos, su implementación ventajas y desventajas.

La simulación de los algoritmos se programan en lenguaje 'C', debido a sus características de portabilidad en diferentes sistemas, el manejo de las comunicaciones entre computadora e impresora, y la facilidad de ensamblarlo con algunas de las principales interfaces gráficas.

2.5 Metodología de Investigación.

La metodología de investigación permite sistematizar el diseño y desarrollo de actividades de manera tal que se vayan cumpliendo gradualmente los objetivos, proponiendo etapas y tareas cuyo desarrollo y evaluación permitan avanzar hacia el objetivo planteado. Las etapas son las siguientes:

- Definición del problema real a resolver.
- Conceptualización del problema: Solución matemática.

En base al análisis del problema se proponen soluciones, en este caso los algoritmos Halftoning existentes, sus posibilidades de implantación, sus rendimientos y posibles mejoras.

- Proposición de una solución basada en las técnicas de procesamiento digital de imágenes: Solución algorítmica.

Los algoritmos para el procesamiento de imágenes son independientes de las características físicas del dispositivo en el cual se quieran desplegar o bien imprimir, esto es, el procesamiento de la imagen es un módulo aparte del sistema. Los módulos de las interfaces gráficas o bien la comunicación con la impresora se tratarán por separado.

- Simulación en lenguaje de alto nivel de la propuesta algorítmica: Solución software.

Las imágenes de entrada, sobre las cuales trabajamos son imágenes con tonalidades de gris, aplicamos la transformación de nuestra imagen a través de los algoritmos de procesamiento y obtenemos una imagen procesada que es ahora binivel, esto es, solo contiene blancos y negros. Hacemos una comparación entre la imagen de tonos de gris y la imagen halftoning resultante, evaluamos la calidad, los posibles errores y la rapidez de procesamiento.

- Simulación en el lenguaje de un dispositivo dedicado: Solución micro-software.

La interface entre la computadora y la impresora debe ser un módulo que se debe trabajar por separado, una vez que tenemos nuestra imagen procesada, es necesario realizar una serie de transformaciones para la comunicación con la impresora.

Para esto manejamos un modulo de impresión mediante software y las interrupciones directas al BIOS para comunicarnos con la impresora. Esto incluye los códigos y secuencias de escape de la impresora, para trabajar en los diferentes modos gráficos que soporta.

- Evaluación, pruebas y validación de resultados.

Se debe hacer la evaluación , partiendo de la solución matemática, la solución algorítmica, la solución de programación en lenguaje de alto nivel y lenguaje ensamblador y la velocidad de respuesta de nuestro sistema.

2.6 Resultados Esperados.

- Incrementar la calidad de desplgado de imágenes en monitores con niveles de gris limitados, con el fin de que se exploten sus recursos al máximo.

- Mejorar la calidad de impresión de imágenes en impresoras blanco y negro, con el propósito de abatir costos y de explotar nuestros recursos al máximo.

- Plantear un esquema de compresión de imágenes para almacenamiento óptimo.

- Proponer un esquema de codificación y transmisión de imágenes.

- Hacer una comparación de nuestras simulaciones con los resultados presentados por la literatura.

Capítulo 3

Análisis y Procesamiento de Imágenes Digitales Halftoning

3.1 Modelo Visual.

El ojo es un sensor de luz y tiene un conjunto de características de respuesta conocidas. Esas características ayudan a explicar muchos de los principios de halftoning y también determinan las características operacionales de un algoritmo halftoning dado.

A causa de que las imágenes son desplegadas como un conjunto discreto de puntos brillantes; la habilidad del ojo para discriminar entre diferentes niveles de brillantez es una consideración importante en la presentación de los resultados del procesamiento de imágenes.

Cuando un objeto es enfocado, la luz que este refleja pasa a través de unos lentes y se proyecta directamente en las paredes de la retina, la cual cuenta con dos clases de receptores de luz discreta principales : bastones y conos. El número de bastones oscila entre 75 y 150 millones, su función principal es proporcionar un panorama general de la escena, reduciendo la cantidad de detalles discernibles dado que algunos de ellos se encuentran conectados a un solo nervio y distribuidos a lo largo de la pared de la retina. El número de conos esta entre 6 a 7 millones y se encuentran localizados en la porción central de la retina, llamada fovea, y son altamente sensibles al color. Cada uno de ellos esta conectado a su propio nervio, [GON92].

El aspecto más característico de los lentes, es que varían su radio de curvatura dependiendo de la distancia de observación, esto es, cuando se enfoca un objeto lejano, los músculos que controlan el lente lo obligan a contraerse, y expandirse cuando un objeto esta cercano al ojo, modificando de esta manera su potencia refractiva, que es, la cantidad de luz de incidencia que permite pasar hacia la retina.

Entonces, la percepción de la imagen se realiza mediante la excitación relativa de los receptores de luz, los cuales transforman la energía radiante en impulsos eléctricos que finalmente son decodificados por el cerebro.

Muchas evidencias experimentales indican que la brillantez subjetiva es una función logarítmica de la intensidad de luz incidente en el ojo. Este comportamiento describe las limitantes del sistema visual humano (SVH) para apreciar una variación de niveles de intensidades de una forma lineal.

3.2 Sistema de Producción de Imágenes.

El sistema de producción de imágenes que propongo, es un modelo el cual describe el proceso a seguir para producir imágenes halftoning de alta calidad. Los elementos del sistema son identificados en la figura 3.1.

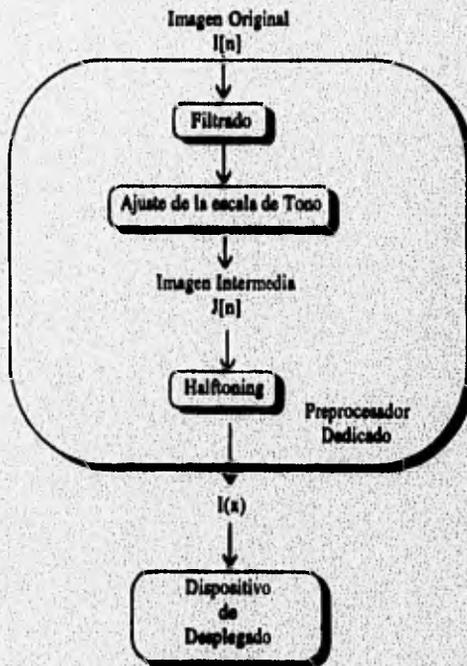


Figura 3.1 Sistema de Producción de Imágenes

La entrada del Sistema de Producción de Imágenes es una imagen digital con tonalidades de gris, la cual a sido digitalizada previamente por algún dispositivo.

En un sistema de red distribuido, una imagen digital dada puede ser desplegada en cualquiera de los diferentes dispositivos de copia o de video. La función principal del preprocesador dedicado es realizar todas las manipulaciones de procesamiento de la imagen necesarias para transformar una imagen digital de tono continuo dada, $I[n]$, en una conveniente imagen binaria, $I(x)$, que será continua en el espacio visual cuando sea presentada al dispositivo de desplegado.

Al realizar una transformación de los niveles de gris de una imagen en otra distribución (binarización) se puede tener tremendo impacto en la calidad de la imagen. Una de las características de binarizar una imagen, es que la imagen resultante es más opaca que la imagen original y los bordes se suavizan; de esta manera, es necesario modificar la escala de tonos para resaltar zonas de alto contraste y afinar o suavizar la imagen previo a la binarización, para contrarrestar tales efectos.

Una de las funciones de la etapa de filtrado es corregir posibles deficiencias que pueda presentar la imagen, con el fin de eliminar asperezas, de tal forma que se garantice una imagen con un número mínimo de errores. Sin embargo la función principal de la etapa de filtrado es realzar los detalles de la imagen para compensar el suavizamiento producido por la binarización.

Una distinción debe ser hecha entre el mejoramiento de la imagen y la integridad del sistema de producción de imágenes. Generalmente, las imágenes que tienen mejor apariencia, es porque alguna modificación se les hizo. Pueden ser clasificadas como imágenes mejoradas, un proceso que crea un cambio de la imagen, la cual por algún criterio es mejor que la original. La técnica de crear la ilusión de niveles de gris mediante una juiciosa distribución de pixeles binarios, la esencia de Halfoning, tiende a desalinearse una imagen, es decir, los detalles finos de la imagen pueden ser perdidos en los patrones halfone. Preaminando una imagen compensamos estos efectos, manteniendo tan secretamente como sea posible la inalteración en la integridad de la imagen digital.

El ajuste de la escala de tonos es utilizado como modificador de intensidades entre los elementos mas claros y mas oscuros de una imagen. Ya que forzamos un cambio brusco de un valor de intensidad a únicamente dos posibles valores. Por ejemplo, un valor de nivel de gris igual a 156 es obligado repentinamente a cambiar a un valor de 0 o 1 dependiendo del criterio que apliquemos. De esta forma, un cambio repentino ocurre en la escala de grises, modificando el contraste de la pintura.

3.3 Filtrado

Muchas de las imperfecciones que tiene una imagen se debe al medio de digitalización, el cual por algún factor externo modificó la pintura generando elementos visuales no deseados. Para eliminar algunas formas de asperezas de nuestra imagen es posible realizar una serie de modificaciones, con la finalidad de seleccionar los rasgos de mayor importancia de nuestra pintura. Una de las principales herramientas en procesamiento de imágenes se refiere a filtrado, el cual es un proceso de selección de las características de interés principal.

Existen dos formas de realizar el filtrado de imágenes digitales, el método en el dominio espacial y en el dominio de la frecuencia. El dominio espacial se refiere al plano de la imagen en sí mismo; y los enfoques en esta categoría están basados en la manipulación directa de los píxeles en una imagen. Por otra parte, las técnicas de procesamiento en el dominio de la frecuencia se basan en la modificación de la Transformada de Fourier de la imagen.

Las componentes de frecuencia determinan rasgos característicos de las imágenes :

* *Componentes de alta frecuencia* : caracterizan ejes y otros detalles de forma de una imagen.

* *Componentes de Baja frecuencia* : son responsables de las características de una lenta variación de la imagen, tales como contraste total e intensidad promedio.

Con lo cual, podemos seleccionar rangos de frecuencias a través de un filtro específico; principalmente se cuenta con tres tipos : Paso Bajas, Paso Altas, Paso Banda.

Filtro Paso Bajas : También conocido como Filtros de suavizamiento; atenúan o eliminan las componentes de alta frecuencia. El efecto de un filtrado paso bajas es crear una imagen borrosa. Este tipo de filtrado es utilizado para remover detalles de una imagen previo a la extracción de un objeto.

Filtro Paso Altas : atenúan o eliminan componentes de baja frecuencia. El resultado de un filtrado paso altas resulta en un afinamiento aparente de ejes y otros detalles de forma.

Filtros Paso Banda : remueven una región de frecuencia seleccionada entre frecuencias altas y bajas . Estos filtros son usados para restauración de imágenes y raramente son de interés en mejoramiento de imágenes, [PRA91].

3.3.1 Métodos de Dominio Espacial

El termino dominio espacial se refiere a el conjunto de pixeles que componen una imagen y los métodos de dominio espacial son procedimientos que operan directamente sobre esos pixeles. Esta es una operación de punto descrita en la sección 1.2.3.1.

Funciones de procesamiento de imágenes en el dominio espacial puede ser expresado como : $g(x,y) = T[f(x,y)]$ donde $f(x,y)$ es la imagen de entrada, $g(x,y)$ es la imagen procesada y T es un operador en f definida sobre algún vecindario de (x,y) .

El enfoque principal para definir un vecindario alrededor de (x,y) es utilizar un cuadrado o área de subimagen rectangular centrada en (x,y) .

El uso de máscaras espaciales para procesamiento de imágenes usualmente es llamado "filtrado espacial" y las máscaras por si mismas son llamadas filtros espaciales. El enfoque básico de procesamiento en el dominio espacial, es sumar los productos entre los coeficientes de la máscara y las intensidades de los pixeles bajo la máscara en una posición específica en la imagen. La figura 3.2 muestra una máscara espacial general de 3x3. Denotando los niveles de gris bajo la máscara en cualquier posición como $z_1, z_2, z_3, \dots, z_9$ la respuesta de una máscara lineal es :

$$R = w_1z_1 + w_2z_2 + w_3z_3 + \dots + w_9z_9$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Figura 3.2 Máscara general de 3x3.

Si el centro de la máscara esta en la posición (x,y) en la imagen, el nivel de gris localizado en (x,y) es reemplazado por R . La máscara es entonces movida a la siguiente posición del siguiente pixel en la imagen y el proceso es repetido. Esto se realiza a través de toda la imagen. Para los pixeles localizados en los bordes, el valor de R es calculado en vecindarios parciales, [GON92].

Como se mencionó anteriormente, la función principal de la etapa de filtrado es realzar los bordes de la imagen, de tal forma que cuando se aplique la binarización a la imagen, los efectos que atenúan los bordes se compensen. Por lo tanto utilizaremos un filtrado paso altas a través de filtros derivativos que se analizan en la siguiente sección.

3.3.1.1 Filtros Derivativos.

El objetivo principal de afinamiento es realzar los detalles finos de una imagen o realzar detalles que han sido opacados por un error o como un efecto natural de un método particular de adquisición de la imagen.

El promedio de pixeles sobre una región tiende a opacar detalles en una imagen. Como promediar es análogo a una integración, se puede esperar que la diferenciación tenga el efecto opuesto, de esta forma se afina la imagen. Un método común de diferenciación en aplicaciones de procesamiento es el Laplaciano.

El laplaciano de una función bidimensional $f(x,y)$ es una derivada de segundo orden definida como :

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Puede ser implementado en forma digital de varias formas. Para una región de 3 x 3 la forma mas conocida es :

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8)$$

Donde las z 's fueron definidas en la sección 3.1.1. Los requerimientos básicos para definir el laplaciano digital es que el coeficiente asociado con el pixel central sea positivo y que los coeficientes asociados verticales y horizontales sean negativos. A causa de que el Laplaciano es una derivada, la suma de los coeficientes tiene que ser cero, de ahí que la respuesta es cero si el punto en cuestión tiene valores semejantes a sus vecinos. La máscara que se utiliza para implementar el laplaciano se muestra a continuación, [ROS92].

0	-1	0
-1	4	-1
0	-1	0

Figura 3.3 Máscara usada para calcular el Laplaciano

3.3.1.2 Filtrado en el Dominio de la Frecuencia

En el dominio de la frecuencia hay que calcular la Transformada de Fourier de la imagen que va a ser modificada y tomar la Transformada de Fourier inversa para producir la imagen mejorada.

En la práctica pequeñas máscaras espaciales son usadas considerablemente más que la Transformada de Fourier debido a su simplicidad de implementación y velocidad de operación.

El enfoque básico en el dominio de frecuencia se realiza mediante la atenuación de un rango específico de componentes de frecuencia en la transformada de Fourier de una imagen dada.

$$G(u,v) = H(u,v)F(u,v)$$

donde $F(u,v)$ es la Transformada de Fourier de la imagen. El problema es seleccionar una función de transferencia de un filtro $H(u,v)$ que produzca $G(u,v)$ para atenuar las componentes de frecuencia de $F(u,v)$, [GON92].

La transformada de Fourier inversa de $G(u,v)$ producirá la imagen modificada.

3.4 Ajuste de la Escala de Tono

Los métodos Halftoning de umbral son un ejemplo de una función de mejoramiento de contraste; ya que representan el límite máximo de incremento de contraste que puede ser impuesto sobre una imagen: bajas intensidades son forzadas a cero y altas intensidades son forzadas a uno.

3.4.1 Mejoramiento de Contraste

Es un ejemplo de mapeo de intensidades y puede ser definido como un proceso que amplifica, o incrementa las diferencias de intensidad entre los elementos más claros y más oscuros de una imagen.

Este proceso no altera el rango de intensidades de la imagen; esto es, si el posible rango de intensidades es de 0 a 255 previo a la alteración, será también de 0 a 255 después de la alteración. El rango de un mapeo de intensidades debe ser el mismo que su dominio.

Si un valor de intensidad se mapea a sí mismo, esto es si $f(x)=x$, no ocurren cambios de intensidad, y la gráfica de la función será una línea recta de pendiente 1.

El proceso de binarización se grafica como una función escalón con únicamente dos posibles valores 0 o 1 (blanco o negro). Esas dos funciones representan los extremos del posible incremento de contraste, generalmente se utiliza una función que mapea entre estos dos extremos.

El objetivo de un mapeo de tonos es tratar de evitar diferencias apreciables entre pixeles adyacentes; mejorando el contraste, pero a consecuencia de esto hay una atenuación de bordes.

Una función utilizada para el mapeo de tonos es la función seno, la cual se muestra en la figura 3.4.

En la práctica actual, la función de contraste de mapeo exacta no es importante, sino como se puede realizar el procedimiento mostrado en la figura 3.4.

Por ejemplo el rango de operación de la función seno es de $-\pi/2$ a $\pi/2$ y puede ser utilizada para realizar el mapeo de contraste con un escalamiento y transacción conveniente de valores de intensidad para el dominio de la función y del rango de la función, [LUS93]. Asumiendo que i representa un valor de intensidad con un rango de intensidad posible de 0 a m , la siguiente función puede ser usada para calcular una delta de intensidad :

$$\theta = \pi \left(\frac{i}{m} - \frac{1}{2} \right) m$$

$$f(\theta) = \frac{m}{2} [\sin(\theta) + 1] - i$$

esta delta puede ser sumada a los valores de intensidad para realizar el incremento de contraste o restada de los valores de intensidad para realizar una reducción de contraste.

3.4.2 Mapeo de Brillantez

Es otro tipo de mapeo de intensidad, el proceso es similar al incremento de contraste; donde la alteración de contraste mapea valores de intensidad altos y bajos de forma diferente, sin embargo el mapeo de brillantez los mapea en la misma forma, [LUS93].

Niveles altos y bajos son incrementados cuando un rango de intensidades es mapeado para realizar un incremento de brillantez. Esto es mostrado en la fig. 3.5

El mapeo de brillantez utiliza la siguiente función :

$$f(i) = \left(\sqrt{\frac{i}{m}} - \frac{i}{m} \right) m$$

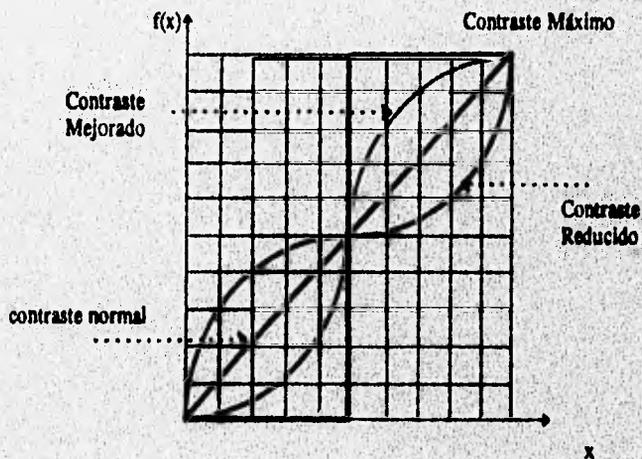


Figura 3.4 Función de Mejoramiento de Contraste

3.5 Halftoning

Hay un gran número de opciones para implementar Halftoning, por lo que el enfoque de estos algoritmos es tratado en el siguiente capítulo.

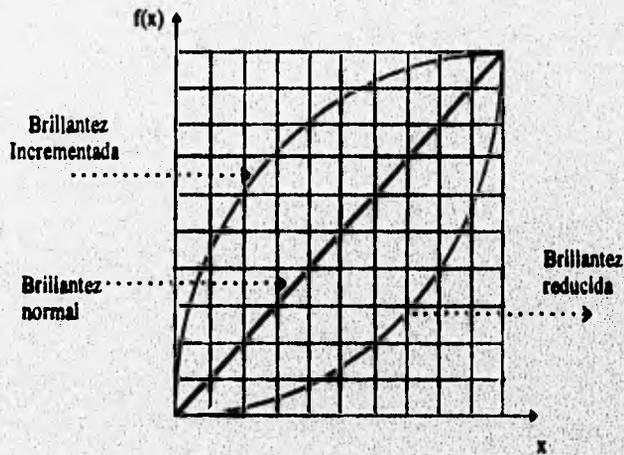


Figura 3.5 Función de Mapeo de Brillantez.

Capítulo 4

Métodos de Procesamiento para la Generación de Imágenes Halftoning.

Los diferentes métodos para generar Imágenes Halftoning son vistos en este capítulo, los cuales serán comparados y evaluados para determinar cual de ellos ofrece mayores ventajas como son la calidad de visualización de la imagen y el tiempo de procesamiento. Haciendo uso de operaciones de punto (respecto de su ubicación) y operaciones locales. Este capítulo es la base teórica para la implementación de los algoritmos Halftoning.

4.1 Métodos de Procesamiento

4.1.1 Thresholding

La forma mas fácil de crear una imagen binivel a partir de una imágenes con tonos de gris es mediante un procedimiento llamado *thresholding*. El cual define un nivel de comparación, contra quien se comparan los elementos de la imagen, a dicho valor se le conoce como *umbral*. Cuando el valor del pixel es menor que el umbral se considerará que el pixel es negro, de otra forma será considerado blanco, [SCH89].

Generalmente este umbral se encuentra en la mitad del rango de valores de niveles de gris. Por ejemplo si tenemos una escala de 256 niveles de gris (0-255), el valor del umbral sería 127, y de este modo aquellos valores que esten por debajo de este nivel los consideraremos negros, y los que sean mayores a 127 serán considerados blancos.

Esta es la forma más simple con la cual nosotros podemos binarizar una imagen. thresholding no es una simulación de Halftoning, sino que es un método crudo para producir la única opción viable que pueden ofrecer los dispositivos binivel. La figura 4.1 muestra la imagen mediante este procedimiento.

El inconveniente de este enfoque es claro, los detalles de la imagen se han desvanecido. Se pueden obtener diferentes resultados modificando el valor del umbral, pero la imagen resultante nunca será lo suficientemente buena.

4.1.2 Método de Bayer

En el pasado, las técnicas más comúnmente utilizadas era mediante la creación de patrones. Uno de los más conocidos y de resultados más aceptables es el Método de Bayer, [LUS93].

Su principio se basa en comparar los niveles de gris de la imagen fuente con un patrón fijado previamente. Este patrón consiste de una matriz de 8x8, distribuyendo homogéneamente 64 niveles de gris. Esta es una limitante, ya que el rango de valores de la imagen debe ser de 0 a 63 tonalidades.

Entonces cada bloque de 8x8 de la imagen es comparada con el patrón. Aquellos pixeles de la imagen fuente que son mas grandes que los correspondientes elementos en la matriz serán pixeles blancos en la imagen resultante y aquéllos que no lo son serán pixeles negros.

El patrón dither de Bayer es el siguiente :

0	32	8	40	2	34	10	42
48	16	56	24	50	18	58	26
12	44	4	36	14	46	6	38
60	28	52	20	62	30	54	22
3	35	11	43	1	33	9	41
51	19	59	27	49	17	57	25
15	47	7	39	13	45	5	37
63	31	55	23	61	29	53	21

Figura 4.1 Patrón de Bayer de 8x8

El problema con el dither de Bayer es que impone un patrón rígido en la imagen que es procesada, sin estimar el contenido de la misma. Aplicado a una imagen grande, frecuentemente resulta en un patrón reconocible si la imagen es impresa y vista completamente.

El patrón que el dither de Bayer imprime en una imagen es llamado *artefacto*. Los artefactos no son efectos visuales deseables derivados del procesamiento de la imagen, [LIN91].

En dithering uno intenta crear un proceso que produzca una ilusión de tonos de gris continuos con artefactos no visibles. En el sentido real esto es imposible, pero se puede reducir la ocurrencia de artefactos hasta cierto punto donde no sean tan obvios.

4.1.3 Dither Ordenado

Dither ordenado es una técnica para reducir datos bidimensionales de alta precisión a una precisión menor utilizando información posicional para retener la mayor cantidad de información que sea posible.

Al igual que todas las técnicas Halftoning, Dither ordenado puede ser utilizado para reducir imágenes multinivel de tonos de gris a un número menor de niveles de gris y en el caso más simple a blanco y negro.

Esto puede ser usado para reducir el espacio de almacenamiento de una imagen mientras se retiene una gran cantidad del contenido de información de la imagen.

De esta forma, si nosotros tenemos una imagen de una dimensión de 512 x 512 pixeles y 256 niveles de gris, estaríamos hablando de un espacio de almacenamiento de 262,144 bytes a un byte por pixel, convirtiendola en una imagen Halftoning, se reduciría ocho veces el espacio de almacenamiento (32,768 bytes), teniendo un bit por pixel, conservando la imagen aún buena calidad.

Esta técnica es muy rápida y eficiente, sin embargo, como veremos mas adelante, existen otras formas de implementar halftoning que producen una mejor calidad.

El procedimiento es muy simple, Halftoning típicamente requiere una comparación, un arreglo de consulta y un módulo de operaciones por pixel. La idea de dither ordenado es distribuir uniformemente el rango esperado de niveles de gris a través de una matriz la cual actuará como una pantalla.

La matriz será desplazada a través de la imagen y sus elementos son comparados con los correspondientes niveles de gris de la imagen. Para aquellos niveles de gris que sean mayores que el elemento de la matriz, se considerará que son píxeles negros, de otro modo serán considerados como píxeles blancos.

El problema es como diseñar la matriz que nos sirve como base de comparación, donde cada nivel de gris produce un patrón de salida.

Consideremos un caso de una matriz Halftoning para 256 niveles de gris.

$$M_0 = [0]$$

Esta es la matriz dithering cero, sus dimensiones son 2^0 por 2^0 . Contiene el rango de 0 a 255 pero no es usada.

M_1 es la primer matriz dithering, sus dimensiones son 2^1 por 2^1 . Contiene el rango de 0 a 255 pero únicamente para cuatro números. Esto significa que M_1 únicamente puede generar cuatro patrones halftone distintos.

$$M_1 = \begin{bmatrix} 0 & 192 \\ 128 & 64 \end{bmatrix}$$

Si usamos unas M_i mas grandes obtendremos más patrones y mejor calidad de la imagen.

Podemos imaginar que M_i son 4 copias de M_1 con algunas constantes sumadas :

$$M_i = \begin{bmatrix} M_1 + 0 & M_1 + 192 \\ M_1 + 128 & M_1 + 64 \end{bmatrix}$$

Similarmente M_2 :

$$M_2 = \begin{bmatrix} 0 & 192 & 48 & 240 \\ 128 & 64 & 176 & 112 \\ 32 & 224 & 16 & 208 \\ 160 & 96 & 144 & 80 \end{bmatrix}$$

En este caso M_2 es justamente cuatro copias de M_1 con constantes sumadas :

$$M_2 = \begin{bmatrix} M_1 + 0 & M_1 + 48 \\ M_1 + 32 & M_1 + 16 \end{bmatrix}$$

En general M_k puede ser definida como sigue :

$$M_k = \begin{bmatrix} M_{k-1} + 0 \times 2^{n-2k} & M_{k-1} + 3 \times 2^{n-2k} \\ M_{k-1} + 2 \times 2^{n-2k} & M_{k-1} + 1 \times 2^{n-2k} \end{bmatrix}$$

donde 2^n es el limite superior del rango sobre el cual se esta ditherando (en el caso del ejemplo n es 8, ya que $2^8 = 256$) y k es tal que $n \geq 2k$.

De esta forma podemos generar las matrices halftoning, [GLA90].

A continuación se muestra la matriz generada de 8×8 y la imagen ditherada con la misma. Vale la pena comparar la calidad de la imagen generada con este procedimiento y la imagen generada con el método de Bayer, ya que las dos tienen secuencias semejantes.

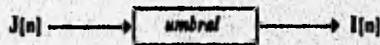
Matriz Dither Ordenado de 8×8 :

0	192	48	240	12	204	60	252
128	64	176	112	140	76	188	124
32	224	16	208	44	236	28	220
160	96	144	80	172	108	156	92
8	200	56	248	4	196	52	244
136	72	184	120	132	68	180	116
40	232	24	216	36	228	20	212
168	104	152	85	164	100	148	84

Figura 4.3 Dither Ordenado Patrón de 8×8

La generación de imágenes Halftone por este método es buena, sin embargo, se pueden apreciar los patrones impresos (artefactos) que se han generado, ya que este método impone al igual que el de Bayer patrones rígidos.

Hasta ahora, los modelos Halftone considerados hacen uso únicamente de la información posicional de la imagen, sin tomar en cuenta los píxeles que se encuentran alrededor de los que están siendo procesados. Este esquema de procesamiento se conoce como *procesamiento de punto* (descrito en la sección 1.2.3.1), donde la salida depende únicamente del píxel actual de entrada. Estos pueden ser modelados como se muestra en la figura.



Procesamiento de Punto

Los píxeles de la imagen digital de tono continuo, $J[n]$, son comparados simplemente con un umbral para determinar el estado de los píxeles de salida, $I[n]$.

4.2 Métodos de Difusión de Error.

La forma mas eficiente para llegar a un procedimiento que no impone artefactos tan obvios en una imagen es emplear el método de difusión de error.

Los métodos de difusión de error requieren operaciones de vecindad y son de este modo mas intensivos en cálculo, sin embargo son de mas alta calidad y son los preferidos para generar imágenes Halftoning.

El modelo es ilustrado gráficamente en la figura 4.4

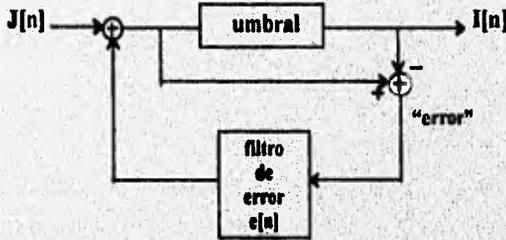


Figura 4.4 Difusión de Error.

El umbral es fijado a la mitad de la entrada, $J[n]$, varía como es usual de 0 a $N-1$ niveles de gris. El valor de nivel de gris del pixel es comparado con el umbral, la diferencia es llamada el "error" para la posición n . La señal consistente de valores de error pasados es llevada a través de un filtro de error $e[n]$, para producir un factor de corrección que será sumado a valores de entrada futuros. Los errores son de esta forma "difundidos" sobre una vecindad de pixeles determinada por $e[n]$, [UL187].

El resultado será una serie de pixeles blancos y negros alternados que se asemejará a la escala de gris original.

Este enfoque tiene algunos problemas, el mayor de los cuales es encontrar la forma de difundir el error de los pixeles. Uno de los más molestos derivados del proceso es la tendencia de suavizar bordes entre áreas de niveles de gris. Por ejemplo, imágenes que involucran algún texto usualmente no se reproducen bien.

4.2.1 Filtro de Floyd-Steinberg

El enfoque de difusión del error más popular es el filtro de Floyd-Steinberg. Este filtro actualmente es uno de los más viejos, ya que se dio a conocer en 1975, [UL187].

$$\left(\frac{1}{16}\right)^{\times} \quad \begin{array}{c} \bullet \\ 7 \\ 3 \ 5 \ 1 \end{array}$$

Diagrama de difusión del error
Floyd y Steinberg 1975

Podemos notar que la suma de los elementos es igual a 1, así que los errores no se amplifican ni se reducen.

Dado que hay un error entre el pixel \bullet y el umbral entre blanco y negro. El esquema se interpreta de la siguiente manera:

Una vez que se ha calculado el error de ese pixel (es decir la diferencia que hay entre el valor de gris y el umbral) tomamos en consideración 4 pixeles.

Sumamos :

7/16 de el error a el pixel que se encuentra a la derecha de \bullet .

3/16 del error al pixel que se encuentra a la izquierda y una posición abajo de \bullet .

5/16 del error directamente al pixel debajo de \bullet .

1/16 del error a el pixel que se encuentra una posición abajo y a la derecha de \bullet .

Habiendo hecho esto, debemos movernos al siguiente pixel de la imagen fuente y repetir el proceso. Los resultados del filtro de Floyd y Steinberg son realmente buenos. Este resulta usualmente en una imagen final que tiene mejores detalles en los bordes entre áreas de diferentes niveles de gris en la imagen fuente.

4.2.2 Filtro Stucki

El filtro de Stucki mejora la ejecución del filtro de Floyd y Steinberg. Este produce diferentes resultados que podrían trabajar mejor sobre algunas imágenes.

$$\left(\frac{1}{42}\right) \times \begin{array}{cccccc} & & & & \bullet & 8 & 4 \\ & & & & 2 & 4 & 8 & 4 & 2 \\ & & & & 1 & 2 & 4 & 2 & 1 \end{array}$$

Filtro Stucki

Estos números suman 42, así que el pixel a la derecha de \bullet tendría $8/42$ del error y así sucesivamente, [UL187].

El inconveniente del filtro Stucki es que es lento. Ya que hace la comunicación con demasiados píxeles en tres líneas. Sin embargo más que esto, requiere demasiadas multiplicaciones enteras y divisiones.

Las imágenes que han sido procesadas con el filtro Stucki generalmente se ven más aceptables que las que se manejan con el filtro de Floyd y Steinberg, pero el incremento de tiempo en usar el filtro Stucki es mayor, especialmente para imágenes grandes.

4.2.3 Filtro Burkes

El tercer filtro dither encontrado más comúnmente es el filtro de Burkes. Este fue diseñado como un intermedio entre los dos filtros anteriores. Se comunica con menos píxeles que el anterior y lo hace usando una matemática más simple que puede ser manejada con corrimientos de bits y restas. [LUS93]

$$\left(\frac{1}{32}\right) \times \begin{array}{cccccc} & & & & \bullet & 8 & 4 \\ & & & & 2 & 4 & 8 & 4 & 2 \end{array}$$

Filtro de Burkes

Hay numerosos filtros que se han ido desarrollando a través de los años entre ellos están el filtro de Sierra, Jarvis, Judice, Ninke, Stevenson y Arce filter, [UL187]. Este último produce resultados mas atractivos que los vistos anteriormente, solo que hace uso de 1/200 de el error, pero su cálculo es tan intenso que se hace impráctico para computadoras personales.

$$\left(\frac{1}{32}\right) \times \begin{array}{ccccccc} & & & & \bullet & 5 & 3 \\ & & & & & 2 & 4 & 5 & 4 & 2 \\ & & & & & & 2 & 3 & 2 & \end{array}$$

Filtro Sierra

$$\left(\frac{1}{48}\right) \times \begin{array}{ccccccc} & & & & \bullet & 7 & 5 \\ & & & & & 3 & 5 & 7 & 5 \\ & & & & & & 1 & 3 & 5 & 3 & 1 \end{array}$$

Filtro de Jarvis, Judice y Ninke (1976)

$$\left(\frac{1}{200}\right) \times \begin{array}{ccccccc} & & & & \bullet & & 32 \\ & & & & & 2 & 26 & 30 & 16 \\ & & & & & & 12 & 26 & 12 \\ & & & & & & & 5 & 12 & 12 & 5 \end{array}$$

Filtro de Stevenson y Arce (1985)

4.3 Difusión de error con perturbación.

Los elementos de los filtros en el método de difusión del error son llamados pesos. La suma de los pesos es igual a 1, así que el error no se amplifica ni se reduce. Un enorme cantidad de opciones para la creación de filtros esta disponible. Un filtro de error puede consistir de pesos de algún :

- 1.- número
- 2.- posición
- 3.- valor

Para lograr la eficiencia de cálculo un filtro lo mas pequeño posible es preferido, [UL187]

Capítulo 5

Desplegado de Imágenes Halftoning

El sistema de producción de imágenes visto en la sección 3.2 ofrece la posibilidad de desplegar la imagen a través de diferentes medios, ya sea por pantalla o bien mediante impresión. Una vez que el preprocesador dedicado a creado una imagen halftoning de alta calidad, depende ahora del dispositivo de desplegado hacer la presentación final. Su función es muy importante y debe realizar su trabajo a toda su capacidad para ofrecer los mejores resultados. Una gran variedad de arquitecturas pueden ofrecer el servicio, sin embargo nos enfocaremos al peor de los casos: monitores monocromáticos.

Tres factores controlan la calidad y la cantidad de imágenes que pueden ser generados en cualquier sistema de computadora: el Adaptador Gráfico, el Monitor y el Lenguaje de Programación.

5.1 El Adaptador Gráfico.

El adaptador gráfico determina la máxima capacidad de gráficos del sistema. Sus funciones principales son:

- Determinar el modo gráfico apropiado para ser utilizado por un hardware en particular, definiendo la resolución y el número de colores posibles.
- Desplegar la información de la memoria de video a través de la pantalla de la computadora.

El medio ambiente de las computadoras personales cuenta con cuatro tipos de adaptadores principales que pueden soportar gráficos monocromáticos: el adaptador gráfico a color CGA, el adaptador gráfico mejorado EGA, el adaptador Hercules HGC y el arreglo gráfico de video VGA; los cuales son capaces de generar una diversa selección de modos gráficos que van desde dos hasta 256 colores y variando la resolución del video de 320x200 a 640x480 pixeles, [KLI91].

La figura 6.1 ilustra las diferentes resoluciones que pueden ofrecer estos adaptadores.

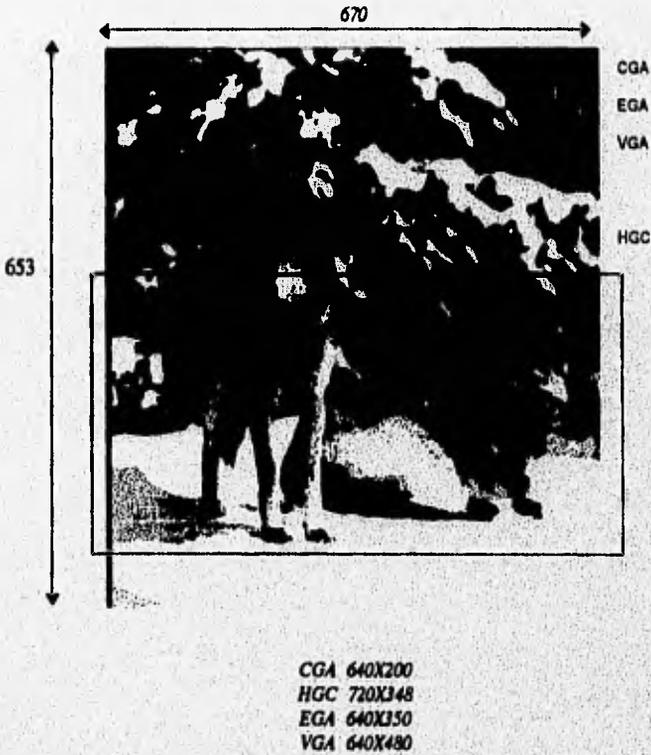


Figura 6.1 Dimensiones Físicas de Adaptadores Gráficos Monocromáticos.

El primer adaptador gráfico, el CGA, cuenta con dos modos gráficos únicamente y una resolución muy pequeña debido a que fue diseñado para lograr la compatibilidad con los conjuntos de televisión. Una sus principales limitaciones fue el alto costo de los circuitos de memoria, lo cual limitó el uso de la misma. Posteriormente surgieron nuevas alternativas como el adaptador Hercules, el cual dio además algunas pequeñas variaciones y nuevas alternativas. El EGA ofreció las características del CGA y modos gráficos nuevos, es aquí donde se empezó a tener un panorama mas claro de lo que son ahora los adaptadores gráficos actuales. Finalmente apareció el arreglo gráfico de video VGA el cual reunió las características de los tres anteriores, incrementó el número de colores reproducibles y además mejoró la resolución. Nuevos adaptadores han surgido sin embargo todos ellos son una variación del VGA.

El problema más grande entre las tarjetas gráficas fue la falta de estándares. A pesar de que todos ellos pueden reproducir el CGA, su resolución limitada y la opción de colores restringida lo hacen menos que ideal para la programación de imágenes y gráficos avanzados. Adicionalmente, los métodos de programación usados para muchas de las tarjetas son inconsistentes.

El adaptador gráfico Hercules fue diseñado originalmente debido a que los gráficos ofrecidos por el CGA fueron demasiado malos. Una tarjeta Hercules ofrece una resolución de 720x348 pixeles en dos colores, además de contar con dos páginas de información gráfica en su buffer de video. Cualquier página puede ser hecha visible. Lo que le da una gran ventaja para desplegar dos pantallas distintas instantáneamente y lo hace el adaptador gráfico monocromático mas veloz de todos. [LUS93]

El EGA intento permanecer compatible con el CGA, extendiendo la memoria disponible y agregando soporte al BIOS para lograr una resolución más alta.

El gran número de registros y modos operacionales hacen difícil su programación. Muchos de los registros son únicamente de escritura y determinar el estado actual del adaptador es imposible, esto impide el desarrollo de sistemas multitarea y programas de memoria residente, los cuales necesitan guardar el estado del video antes de cambiar de tarea.

El VGA es muy similar al EGA, extiende algunas de sus funciones y agrega nuevos modos de alta resolución. El cambio mas significativo fue el uso de registros de lectura y escritura. A pesar de que las llamadas al BIOS permanecen compatibles a través de varios adaptadores, existen pocas diferencias entre los puertos de interface del hardware común. [KLI91].

La arquitectura monocromática CGA, EGA y VGA tienen pocas variaciones. Los adaptadores originales están basados en el controlador de gráficos de Motorola 6485, mientras que el EGA y VGA utilizan circuitos propios de IBM, [KLI91].

Los sistemas de cómputo más recientes ofrecen muchas ventajas que sus antecesores no pueden brindar, son capaces de realizar tareas a altas velocidades y con mejor calidad en la presentación de la información, sin embargo, su costo es un factor muy importante que los hace inaccesibles. Esta es la razón principal del porqué no podemos estar renovando nuestro equipo por periodos de tiempo cortos. De esta forma debemos tratar de utilizar medios alternativos que nos puedan ofrecer algunas ventajas con el equipo que contamos para estar al margen de los avances que surgen día con día.

La forma en la que los adaptadores gráficos funcionan es básicamente la misma. Todos son dispositivos mapeados de memoria, esto es, aparecen en la PC como memoria direccionable tal y como un programa normal lo hace. En teoría se puede cargar un programa dentro del buffer de video y ejecutarlo ahí. Pero en la práctica es diferente. La diferencia entre la memoria de un programa y la memoria de video es que la memoria de video tiene algún hardware extra que continuamente lo está chequeando y hace que el contenido de la memoria sea desplegado en el monitor.

5.2 Mapas de Memoria

Cada uno de los controladores gráficos guardan su memoria de video en un segmento de memoria particular y por supuesto cada uno utiliza un segmento diferente.

Una tarjeta CGA tiene su buffer de video en el segmento 0B800H, cada línea es de 640 píxeles o bien 80 bytes de longitud. A causa del hardware limitado, no se tiene un arreglo de memoria lineal para el buffer gráfico. Las líneas de memoria en el buffer no tienen una relación uno a uno con las líneas que aparecen en la pantalla. Generalmente se separan líneas pares en un segmento y las líneas impares en otro.

Las tarjetas Hercules tienen un arreglo de memoria similar, sus líneas son de 90 bytes de longitud en vez de 80, la memoria de video inicia en el segmento 0B000H para su primer página y en el segmento 0B800H para la segunda.

Las tarjetas EGA y VGA se caracterizan porque su memoria de video es lineal, esto es, el orden de las líneas en la pantalla corresponden a el orden de las líneas en la memoria de video, lo cual los hace mucho más simples. El área del bus de direcciones de la memoria de video lineal inicia en el segmento 0A000H y todas las tarjetas más recientes han empezado sus buffers en ese punto. Se puede calcular la cantidad de memoria de cada línea dividiendo el número de píxeles de la línea entre 8, [KLI91].

5.3 Operación del Adaptador Gráfico.

La forma en la cual la memoria de video esta organizada y la manera en la que el hardware la despliega en pantalla es diferente para los modos CGA, EGA y VGA. Los adaptadores gráficos CGA utilizan una memoria de video *multibit por pixel*, mientras que los adaptadores EGA y VGA usan memoria de video *multiplano por pixel*.

5.3.1 Memoria de Video Multiplano por Pixel.

Los adaptadores gráficos EGA y VGA no son adaptadores monocromáticos, pueden reproducir mas de dos colores; utilizan una estrategia multiplano por pixel.

Cuatro imágenes gráficas separadas son almacenadas en la memoria de video. Cada imagen es llamada un *bitplane* o un plano de bits. Cada bit plan almacena la imagen completa con uno de los cuatro colores primarios usados por el VGA y EGA: rojo, verde, azul y un nivel de intensidad. El controlador de video lee simultáneamente los bits correspondientes de los 4 planos y usa la suma para determinar cual de los 16 colores disponibles deben ser desplegados para esa posición en la pantalla. A causa de que cada plano puede guardar un valor de 0 o 1 en un punto en particular, sumar los 4 bit planes se tendrá un rango de 0000 a 1111 (16 opciones). El color que es asociado a cada una de las 16 opciones es determinado por el hardware del controlador del video, [LUS93].

5.3.2 Memoria de Video Multibit por Pixel.

El CGA utiliza una estrategia llamada multibit por pixel. Unicamente un solo mapa de bits es usado. Para gráficas 320X200 de 4 colores se requieren dos bits por pixel. Esto es porque dos bits pueden expresar cuatro diferentes atributos de color 00 01 10 11. Un byte en el mapa de bits controla 4 pixeles en la pantalla. Para el modo de dos colores 640X200 se requiere 1 bit por pixel porque un solo bit puede expresar 2 atributos de color diferentes. 0 y 1, [LUS93].

5.4 Monitores.

Los adaptadores gráficos, el monitor y el lenguaje de programación, cada uno juega un papel determinante en la cantidad y calidad de los gráficos que pueden ser producidos. De estos tres factores, los monitores son segundos en importancia después del adaptador gráfico. El tipo de monitor de display que esta conectado a un adaptador gráfico particular impondrá una limitación física en la salida del adaptador gráfico.

Hay 4 tipos principales de monitores de video que son usados por las computadoras personales compatibles. Estos son el display de color estándar (SCD), el display de color mejorado (ECD), el display analógico y el display de frecuencia variable (multiscanning).

Cada uno de estos cuatro tipos de monitores utilizan una tecnología raster; las pistolas eléctricas del tubo de rayos catódico están continuamente rastreando los fósforos sensitivos de luz en el tubo de izquierda a derecha y de arriba hacia abajo. Cada línea examinada es llamada una línea raster. Mediante alternar las pistolas encendidas y apagadas a tiempos apropiados, los fósforos son encendidos. Antes de que los fósforos excitados puedan desvanecerse las pistolas de nuevo exploran la pantalla, el número de veces que la pantalla completa es barrida por las pistolas es llamado *razón de refresco*.

Los dos primeros SCD y ECD utilizan una tecnología digital, lo que significa que las imágenes en la pantalla son creadas por una serie de pulsos electrónicos los cuales son encendidos o apagados.

El display de color estandar SCD provee una resolución de pantalla máxima de 640X200 pixeles y puede desplegar simultáneamente 16 colores diferentes. Lo cual lo hace apto únicamente para trabajar con el adaptador CGA y con algunos modos del EGA.

El display de color mejorado ECD provee una resolución de pantalla máxima de 640X350 pixeles y puede desplegar un máximo de 16 colores simultáneos. El ECD puede ser utilizado por un adaptador CGA y EGA sin ningún problema.

Los monitores de display analógico usan una señal que varía continuamente capaz de generar más colores. Cuando son usados con un VGA, un monitor analógico puede desplegar todos los modos gráficos CGA y EGA además de la resolución del VGA 640X480 de 2 colores, (KLI91).

Capítulo 5

Monitores de frecuencia variable multiscanning pueden desplegar señales de video digital y señales de video analógicas. El adjetivo "multi" usado por algunos fabricantes se refiere a la habilidad del monitor para rastrear la señal de salida variable del adaptador gráfico para determinar cual de los modos gráficos esta siendo usado.

5.5 El Lenguaje de Programación.

Diferentes lenguajes de programación pueden ser escogidos, C ofrece 3 ventajas para programación : versatilidad , potencia y velocidad.

5.5.1 Versatilidad.

Debido a la versatilidad en el manejo de memoria y control de procesos, es una buena opción para programación.

El manejo de memoria es importante porque las imágenes gráficas son almacenadas como bloques de datos en memoria . La versatilidad de C en el manejo de memoria incluye la habilidad para organizar la memoria de la computadora en diferentes formas, para cumplir las necesidades de diferentes tipos de programas gráficos. Algunos programas usan gran cantidad de datos y un modulo relativamente pequeño de código ejecutable y usan muy pocos datos numéricos. C puede satisfacer ambas necesidades de medio ambiente. La versatilidad en el manejo de memoria también incluye la habilidad para mover rápidamente el contenido de memoria a otro bloque esta habilidad es vital para algunas formas de animación. El control de procesos es importante porque los programas gráficos frecuentemente llaman a subrutinas, que pueden ser utilizadas como modulos independientes, [GLA90].

5.5.2 Potencia.

La potencia de C es resultado directo de su bajo, medio y alto nivel. Un lenguaje de bajo nivel es un lenguaje de programación de quien las sintaxis refleja la operación de los registros de hardware internos de la computadora. Se dice que un lenguaje de bajo nivel funciona a nivel de bits.

Un lenguaje de bajo nivel ofrece potencia para un programador pero algunas veces el código es difícil de leer. Si utilizamos un lenguaje de bajo nivel tal como el lenguaje ensamblador los programas pueden hacer todo lo que el hardware pueda realizar.

5.5.3 Velocidad.

El lenguaje de programación C produce código que corre más rápido a causa de su cercana relación con el lenguaje ensamblador. Ofrece además una variedad de subrutinas especializadas llamadas funciones; las cuales en combinación con subrutinas de lenguaje ensamblador de alta velocidad pueden lograr una óptima ejecución.

5.5.4 Portabilidad de Programas en C

Una cualidad esencial del lenguaje de prog. C es su portabilidad la cual se refiere a la habilidad del programa para correr en diferente hardware de computadora usando diferentes sistemas operativos y diferentes compiladores de C. Se pueden escribir nuestros propios controladores gráficos en lenguaje ensamblador pero no es una tarea trivial el escribir esas rutinas.

Y una de las principales ventajas que el lenguaje de programación C ofrece es la habilidad para adaptarse a la principales interfaces gráficas como son Xwindows, Windows, Macintosh, OpenLook etc, [GLA90].

5.6 Desarrollo del Sistema de Desplegado.

Es evidente que el dispositivo de desplegado no puede mejorar la calidad de una imagen de entrada si esta no se encuentra en una forma tal que pueda desplegarla a su máxima capacidad.

Refiriéndonos al Sistema de Producción de Imágenes (sección 3.2), una vez que la imagen a pasado por todas las etapas del preprocesador dedicado, se encuentra en condiciones de ser desplegada por cualquier dispositivo binivel, sin embargo nuevas modificaciones pueden ser hechas para adaptarse a un dispositivo en particular, únicamente se debe observar la resolución del monitor, y adaptarla la imagen a ese sistema para ser mostrada. (de ser el caso se debe modificar la escala de tonos y realizar pruebas para determinar las mejores opciones).

Se aprovechan las características del lenguaje C para la programación de las rutinas de desplegado de imágenes halftoning, entre las cuales se mencionan algunas funciones para detección de modos gráficos y el controlador de video, [SCH87].

Capítulo 5

Las funciones son :

```
int gdriver = DETECT, gmode, errorcode;
```

detecta el controlador Gráfico, y los posibles modos gráficos que puede manejar.

```
initgraph(&gdriver, &gmode, "");
```

Inicializa gráficos y variables locales. Como resultado de la inicialización esta función regresa un código de error, que indica si el proceso se llevo a cabo sin problemas.

Algunas otras funciones importantes son las que indican la resolución vertical y horizontal del video, además de la función que coloca un pixel en la pantalla para una posición dada.

Todas estas funciones pueden ser implementadas en lenguaje ensamblador, la única ventaja de esto sería la velocidad de respuesta del sistema, sin embargo las rutinas que C ofrece son de rápida ejecución y muy fáciles de usar.

Capítulo 6

Impresión de Imágenes Halftoning en Impresoras Blanco y Negro

La constante evolución que sufren los sistemas de cómputo nos mantienen en una etapa de transición entre los diferentes medios y herramientas de que disponemos para trabajar, por lo que debemos situar nuestros conocimientos lo más próximo a estos avances que día con día surgen incontenibles.

La impresora se encarga de la obtención de copias impresas de los programas y de los datos que trata la computadora, de hecho, cualquier función que visualice la pantalla o almacene la computadora puede ser obtenida en forma de papel impreso. Generalmente la calidad de impresión está íntimamente relacionada con el precio de la impresora y de los materiales que se utilizan para la impresión. Sin embargo, hablar de buenos equipos no se refiere a equipos de costo elevado, los hay muy buenos, excelentes, que no son muy caros y a veces brindan más provecho que otros más costosos y más complicados.

Ante la gran oferta de un gran número de impresoras diferentes, no existe ningún modelo que resulte perfecto para todas las aplicaciones, corresponde al usuario seleccionar la que se ajuste mejor a sus necesidades y a sus posibilidades económicas.

La mejor reproducción de una imagen digital es hecha por impresoras sofisticadas a color, las cuales cuentan con una alta resolución pero con un costo demasiado elevado.

El precio por copia es muy alto, debido en gran parte a que utilizan tinta y papel especiales. Es claro que para adquirir una impresora de este tipo se debe tener los recursos económicos suficientes; y aún así, solo se utilizaría para trabajos muy especiales. Sin embargo las impresoras más comunes y más accesibles como la de matriz de puntos y las impresoras laser son blanco y negro y apesar de que la calidad que ofrecen para imágenes no es buena, existen métodos alternativos que pueden incrementar. Con halftoning se puede lograr muy buenas copias de salida a precios muy accesibles.

6.1 Comunicación Computadora-Impresora

La impresora es un equipo periférico que se encuentra bajo las ordenes de la computadora por lo que la comunicación entre ambos debe ser lo más clara posible. A esto se le conoce como *protocolo*. El protocolo debe ser muy claro:

Cuando conectamos la impresora, esta envía a la computadora una señal que le indica que va a empezar la comunicación. Esta operación se conoce con el nombre de *On Line*, entonces, la computadora y en respuesta a esta señal, le envía su configuración, es decir, como ha de ponerse la impresora para recibir la información que la computadora le enviará posteriormente y como debe de imprimirla, a este programa se le conoce como controlador (*driver*).

Una vez que la impresora ya se encuentra configurada, envía a la computadora la señal de que ya esta preparada para recibir información. Tras lo cual la computadora empieza a enviarla.

Una primer duda que puede surgir es como se pueden imprimir bloques muy grandes si la velocidad de transmisión de la información es mucho mayor que la de su impresión. La respuesta es que la impresora tiene un pequeño almacén (*buffer*) de memoria, donde se guarda la información antes de ser impresa. Una vez que el *buffer* esta lleno, la impresora envía una señal a la computadora con el objeto de que no se le envíe más información hasta que el *buffer* este vacío de nuevo.

Todo este protocolo se establece hasta que se acaba la cantidad de información que hay que imprimir, enviando la computadora una señal de fin de archivo.

Esto se establece en un espacio muy corto de tiempo, proceso en el que la perfecta configuración de la impresora juega un papel muy importante, [ORO90].

6.2 El Verdadero Halftone

Llamado el verdadero Halftone porque es la parte teórica para implementar Halftoning en impresión. Su principio es sencillo; si observamos microscópicamente un pixel, notaremos que está compuesto de una serie de minúsculos puntos, los cuales se agrupan para darle forma. Si pudiéramos variar la cantidad de los elementos que constituyen el pixel, obtendríamos pixeles de varios tamaños. Un pixel que contenga todos sus elementos lo consideraríamos el elemento más oscuro de la imagen, los pixeles que gradualmente vayan perdiendo sus elementos serán percibidos como niveles de densidad intermedios; y un pixel sin elementos, sería el elemento más claro en la pintura.

Gracias a las limitaciones del sistema visual humano es difícil percibir la diferencia entre el número de elementos que constituyen un pixel y otro, de esta forma si unimos dos pixeles de diferentes densidades, el ojo difícilmente será capaz de notar la diferencia.

Ahora bien, para aquellas zonas de la imagen más oscuras, podemos utilizar una agrupación de pixeles los cuales contengan todos o casi todos sus elementos. Y para zonas de la imagen más claras podemos utilizar pixeles con muy pocos o sin elementos, [KIR93].

Recordando que el complejo imaginario del cerebro humano realiza una integración espacial promediando intensidades en pequeñas áreas para reducir la complejidad aparente de la imagen, la variación de intensidades de los puntos negros produce una simulación de imágenes de tono continuo.



Figura 6.1 El Verdadero Halftone

Capítulo 6

Si los niveles de densidad discreta creados por la impresora no son suficientes para generar variaciones de densidad lisa en imágenes de alta calidad, se puede tener aún otra alternativa. Puesto que un pixel aislado de impresión es lo suficientemente pequeño para no ser detectado a simple vista, dos niveles discretos pueden ser mezclados sobre pixeles vecinos utilizando una variación del algoritmo de difusión de error descrito en el capítulo 4.

La figura 6.2 ilustra como un nivel deseado para un pixel de 4.7 elementos puede ser creado mediante el mezclado de 4 y 5 gotas (Samuelson 1987 para impresoras de inyección de tinta).

El entero del número de elementos deseado es depositado en el papel impreso. El error decimal es sumado al siguiente valor del pixel. La difusión del error resultará en un valor promedio corregido a lo largo de la línea de impresión. La resolución espacial, es algo que será reducido en la dirección horizontal y no afectará la posición vertical.

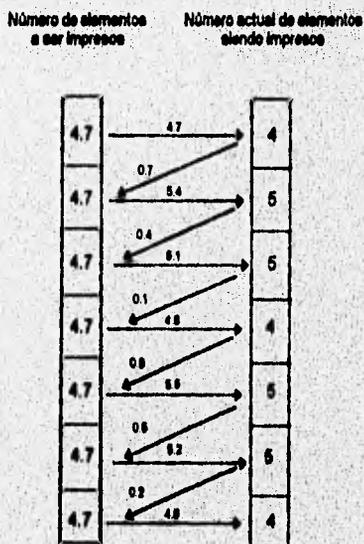


Figura 6.2 Algoritmo de Difusión de Error

Un algoritmo de difusión de error bidimensional crearía un nivel promedio sobre un área cuadrada, esto es, una reducción uniforme en resolución espacial en ambas direcciones, pero demandaría un incremento notable en esfuerzo de cálculo, [KIR93].

El sistema que genera puntos de varios tamaños tiene algunas limitaciones :

Primero, el número total de intensidades esta determinado por el número de los diferentes tamaños de puntos negros disponibles. Por ejemplo, si un pixel esta constituido por 33 elementos, generará únicamente 33 niveles diferentes de intensidad además del blanco.

La segunda y la más importante es que para controlar el número de elementos para cada pixel se debe de tener una lógica de control que le indique a la impresora el número exacto de elementos a depositarse para cada pixel acorde con los datos de la imagen. Pero esta lógica de control debe ser incluida mediante hardware, y las impresoras comerciales no cuentan con ella, solo pueden producir pixeles de tamaño fijo, esta es la principal desventaja ya que no se tiene la posibilidad de variar el tamaño de los pixeles. Sin esta capacidad de nuestras impresoras, otros métodos deben ser empleados.

Tres son los métodos más comunes en la práctica: *Thresholding*, *Patterning* y *Dithering*.

El primero *Thresholding* fué descrito en la sección 4.1.1, no es una simulación de halftoning sino más bien es un método crudo para producir salidas impresas de tono continuo. Los resultados de *thresholding* generalmente son demasiado pobres. *Thresholding* puede tener aplicación en algunas formas de trabajo artístico pero para usarse como una técnica de copia generalizada es inaceptable.

6.3 Patterning.

Patterning el cual simula halftoning, permite que la salida impresa retenga alguna cantidad de información de niveles de gris aún cuando sea impresa en un dispositivo binivel.

Patterning es muy usado si el dispositivo de salida tiene más resolución que aquel que esta produciendo la imagen. Si este es el caso, no habrá degradación en la copia de salida, y patterning producirá una muy buena reproducción de la imagen fuente. Si el dispositivo no tiene suficiente resolución, la salida impresa tendrá una resolución más pobre. En todos los casos se requiere una resolución más alta del dispositivo de impresión.

Patterning utiliza un grupo de elementos o *celdas* en el dispositivo de salida para representar un pixel simple de la imagen fuente. El número de elementos de la pintura dentro del grupo determina la brillantez relativa u oscuridad del grupo total.

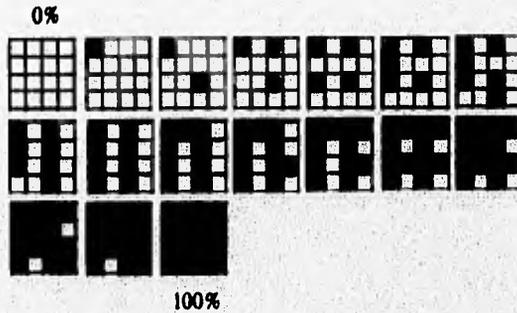
Las celdas usualmente tienen la forma de una matriz cuadrada. Un grupo de $n \times n$ elementos puede producir $n^2 + 1$ niveles de intensidad únicos. Un grupo de 4×4 provee 17 niveles de intensidad en la salida impresa. Es importante que el tamaño de las celdas no sea tan grande, de otro modo podremos percibir detalles dentro de una celda por sí misma y su función como un punto gris simple es perdido, [LIN91].

Las celdas individuales usadas en halftoning son conocidas como patrones halftone. La figura 6.3 muestra los diferentes patrones halftone para una celda de 4×4 elementos.

El esquema usado para generar los patrones halftone tiene importantes implicaciones visuales, el más común *Dither Ordenado de Puntos Agrupados* genera $n + 1$ patrones de n celdas mediante encender o apagar puntos adyacentes a aquellos puntos que ya están encendidos o apagados en la celda previa. El efecto de esto es una serie de puntos más grandes.

El orden de los puntos en la celda debe ser seleccionado para ajustarse al tipo de dispositivo de desplegado que va a ser usado. Esto es debido a que muchos de las impresoras no pueden imprimir puntos adyacentes dadas sus limitaciones mecánicas en diferentes modos de resolución.

Otros métodos para generar patrones halftone son por supuesto posibles. Si nuestro dispositivo de impresión puede imprimir puntos aislados. Entonces el halftone resultante es definido como *Dither Ordenado de Puntos Dispersos*.



0	8	2	10
4	12	6	14
3	11	1	9
7	15	5	13

Figura. 6.3 Patrones Halftone de 4X4 y Notación Matricial

Puede haber otras posibilidades utilizando patrones geométricos, por ejemplo, si los puntos encendidos en el patrón halftone están en el mismo renglón, entonces resultará un patrón de líneas horizontales en áreas de la pintura con niveles constantes.

Estos métodos se pueden representar mediante una matriz halftone de orden $n \times n$, donde sus elementos son números enteros en el rango de 1 a $m \times n$, los cuales indican el orden en el cual los puntos son encendidos para generar la secuencia de celdas (figura 6.3).

Un aspecto que se debe notar es que en la generación de patrones para puntos agrupados, la secuencia de los puntos encendidos en una celda y en las celdas subsiguientes producen una progresión de intensidad lineal. Para simular la característica de respuesta logarítmica del ojo humano es necesario descartar muchas de las posibles celdas, reteniendo únicamente aquellas en las cuales la razón de puntos blancos en celdas consecutivas es más o menos constante. Lo cual reduce el número de posibles niveles de gris, [LIN91].

Al imprimir mediante Patterning se reduce la resolución del dispositivo de salida en 4 en cada dirección. Por ejemplo, si un dispositivo de salida, tiene una resolución de 640x480, la resolución efectiva con patterning será de 160x120 con la capacidad agregada de producir 17 niveles de gris. Si un número mas grande de niveles de intensidad es requerido en la salida impresa, una matriz más grande de patrones debe ser usada.

Los patrones de los elementos de la pintura que representan los diferentes niveles de intensidad deben ser escogidos cuidadosamente, de otra forma, se crearán artefactos visuales no deseables que serán vistos en regiones de la imagen de intensidad constante. El patron escogido usualmente representa una secuencia de crecimiento que es generada recursivamente. Para obtener buenos resultados es importante que cada elemento de la pintura considerado encendido para un nivel de intensidad específico permanezca encendido en los niveles de intensidad más grandes.

6.4 Dithering.

Esta técnica puede ser empleada en impresión de imágenes con tonos de gris cuando la resolución del dispositivo de salida es la misma que la de la imagen que esta siendo producida o cuando una reducción en la resolución del dispositivo de salida no puede ser tolerada.

Dithering produce mas textura que imágenes impresas mediante patterning, debido a que suelen tener mas componentes de alta frecuencia. Dentro del grupo de dithering se encuentran los métodos de difusión de error las cuales fueron vistos en el capítulo 4.

Una vez que se ha procesado la imagen con tonos de gris a través de un preprocesador dedicado (sección 3.1 Sistema de Producción de Imágenes), se tiene ya una imagen con calidad mejorada que al ser presentada ante la impresora solamente hay que decidir si se coloca un punto negro en la impresora o no de acuerdo a la imagen halftone.

Serán vistos dos ejemplos de impresión de imágenes halftone a través de impresoras de matriz de puntos e impresoras laser.

6.5 Implementación de Algoritmos Halftoning para Impresión

6.5.1 Impresión en Matriz de Puntos.

Una de las primeras impresoras en salir al mercado fué la Epson FX-80, una impresora de matriz de puntos, la cual tenía nueve agujas en su cabeza de impresión. Siendo una de las precursoras y dado que sus características físicas y de operación eran fáciles de utilizar, pronto se convirtió en una de las impresoras más usadas y sirvió como base para la creación de nuevas generaciones de impresoras. Nunca se adoptó como un estándar debido a que la evolución de estos dispositivos creció aceleradamente y surgieron nuevas tecnologías, sin embargo, muchas de sus características esenciales no se han modificado y han permanecido eficientes en casi todos los modelos modernos de impresión. Es por esto que aún cuando no se ha establecido un estándar para impresión de gráficos en impresoras de impacto, la Epson FX-80 puede ser emulada por casi todas las impresoras que surgieron posteriores a ella. Sabiendo como manejar la Epson FX-80 tenemos las bases para poder controlar cualquier tipo de impresora de matriz de puntos.

El sistema de impresión para impresoras de matriz de puntos es descrito en el apéndice A, las agujas de la cabeza de impresión están alineadas en una columna vertical, cada una de las agujas golpea la cinta y produce un punto negro en el papel. Dada la disposición física del grupo de agujas, solo es posible la impresión de gráficos en la dirección horizontal, produciendo columnas de puntos que se desplazan hacia la derecha.

La impresora interpreta la información mediante un código binario de 8 bits, el cual corresponde a ocho píxeles verticales de la imagen, esto se ilustra en la figura 6.4. El siguiente paso es mover la cabeza de impresión el ancho de un píxel para imprimir los siguientes ocho píxeles verticales y así sucesivamente. Una vez que se han impreso las primeras 8 líneas de la imagen, la cabeza de impresión debe colocarse ocho píxeles abajo de su posición y regresar hacia el margen izquierdo y repetir la operación.

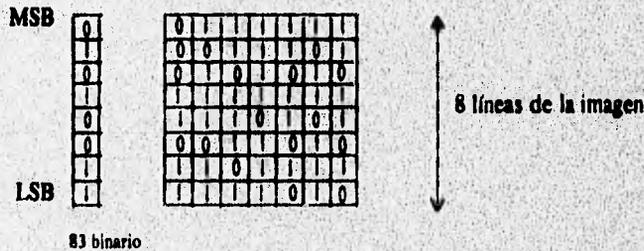


Figura 6.4 Código binario para impresoras matriciales.

El procedimiento es simple, sin embargo el control de la impresora no lo es del todo. Primero porque los datos de la imagen se encuentran almacenados en memoria como una secuencia consecutiva de bits, y para que la impresora los interprete es necesario realizar una transformación.

Los códigos de escape más importantes que se deben tener en mente son el grupo que selecciona el modo gráfico y le dice a la impresora que tan larga va a ser la siguiente línea.

Cada línea de información es precedida por una secuencia de escape que le indica a la impresora que resolución debe usar para esa línea y el número de bytes subsecuentes de información a procesar como gráficos.

Cuando la línea esta completa, debemos enviar a la impresora algún código de escape para mover la cabeza de impresión de regreso al lado izquierdo de la página y abajo exactamente 8 puntos, [EPS90].

Hay varios modos de resolución para imprimir gráficos con diversos grados de densidad. Se puede introducir un modo gráfico normal o bien un modo gráfico maestro :

ESC * Modo LOHI

Donde LO es el byte de orden bajo del entero que tiene la longitud de la línea en pixeles y HI es la parte de orden bajo de ese entero. Por ejemplo si se quiere imprimir una línea de 64 puntos, el valor hexadecimal es 0280h, el byte orden bajo es 80h y el byte de orden alto es 02h. Los diferentes modos a los cuales se puede acceder son descritos a detalle en el apéndice A. Uno de los modos más usados es el modo 06H de pixeles cuadrados, su densidad esta entre el modo de resolución normal y doble resolución, su razón de aspecto es la más aceptable. A este modo se le conoce como modo CRT indicando que es para uso de impresión de pantalla.

La misma imagen impresa en modo normal y cualquier otro modo ocupará diferencias de papel considerables.

La comunicación con el puerto de la impresora se realiza a través de llamadas al BIOS, por medio de la interrupción int86 de software general del 8086 la cual se encarga de enviar los datos por el puerto paralelo de la impresora, [MUR89].

Algunos otros códigos de control importantes son :

ESC @

Inicializa la impresora, causando que todos los datos del buffer sean limpiados.

ESC J VALUE

Regresar la cabeza de impresión a la izquierda y abajo exactamente 8 puntos, VALUE es la distancia que debe bajar la cabeza de impresión la cual fue determinada experimentalmente, (cabe notar aquí que no existe compatibilidad con otras impresoras), [EPS90].

6.5.2 Impresión Laser

Las impresoras que emulan la HP laserJet Plus son el equivalente de la Epson FX80.

Una laserJet almacena la información en memoria igual a como lo hace la computadora que la creó, por lo que la impresión es mucho mas simple. Hay algunas consideraciones que se deben tener en cuenta.

La primera es considerar la memoria con que cuentan las impresoras, por ejemplo, la laserJet (sin el Plus) no tienen memoria suficiente para reproducir páginas completas. La laserJet necesita al menos 1.5 Mb de memoria para manejar imágenes grandes. Sin embargo estas impresoras ya son muy escasas.

La laserJet soporta 4 modos de resolución para imágenes. La resolución de 75 puntos por pulgada, es análoga a la resolución básica de las impresoras de matriz de puntos. Hay densidades mas altas como 100, 150 y 300 puntos por pulgada, [LUS93].

El comando de resolución es de la forma :

ESC * I MODE R

El valor de MODE es una de las 4 resoluciones descritas (75, 100, 150 y 300 dpi).

La siguiente secuencia le dice a la impresora que algunas líneas están en camino :

ESC * r POSITION A

POSITION indica donde debe ir el margen izquierdo, si es cero iniciará en el margen izquierdo del papel, si es 1 la pintura iniciará en la posición actual del cursor.

Cada línea de datos de la imagen representa una línea horizontal de la pintura. Si se desea imprimir una línea de 720 píxeles, la línea gráfica consistirá de 90 bytes de información precedida por otra secuencia de escape. En el fin de los 90 bytes la laserJet automáticamente posiciona el cursor interno en el lado izquierdo de la página y abajo una línea por si otra línea pudiera aparecer.

Mientras tanto, se regresa a modo texto para esperar texto o alguna otra secuencia de escape.

La secuencia de escape que señala el inicio de una línea de información gráfica es:

ESC * b COUNT W

donde COUNT es el número de bytes de la línea.

La impresora interpreta los siguientes 90 bytes como gráficas sin problemas ignorando todas las siguientes secuencias de escape para los siguientes 90 bytes. Cuando se ha completado la impresión de la imagen se debe enviar la siguiente secuencia :

ESC * r B

que le indica a la impresora que la imagen ha terminado.

Frecuentemente es necesario imprimir imágenes que son demasiado grandes para ocupar un simple segmento de memoria, con lo cual se debe imprimir en múltiples segmentos.

Es claro que se puede obtener el mayor provecho de nuestra impresora si se conoce los principios fundamentales de operación y la forma en la cual procesa la información. La imagen de entrada a la impresora debe de contar con una buena calidad, ya que la impresora no es capaz por sí sola de incrementarla.

Capítulo 7

Codificación de Imágenes Halftoning

El volumen de datos en un sistema de cómputo es un factor determinante para el buen rendimiento del mismo, muchas veces, si la cantidad de información es muy grande, el tiempo de procesamiento, la capacidad del canal de transmisión y los medios de almacenamiento, se ven en dificultades de operación.

El procesamiento de imágenes sufre también de estos inconvenientes, dado que la cantidad de datos en una imagen casi siempre es muy grande. Pensemos en una imagen con tonos de gris de tamaño 512×512 pixeles, la cual utiliza un byte por pixel, el tamaño de esta imagen sería $512 \times 512 \times 8 = 2,097,152$ bits o bien 262,144 bytes, y tomaría aproximadamente 3.6 minutos transmitirla usando un módem a 9600 bauds (bits/seg) lo cual es inaceptable para muchas aplicaciones.

Una alternativa de operación para el procesamiento de imágenes, es codificar las imágenes de una forma mas eficiente. Existen dos formas de hacerlo: una es hacer uso de la correlación que existe en los datos de la imagen a causa de la naturaleza de las mismas, y la otra es encontrar métodos para describir imágenes numéricamente que sean más eficientes en términos visuales.

De esta manera, los métodos de compresión de datos son divididos en dos categorías: los que preservan la información (o sin pérdidas) y los que admiten alguna degradación, comúnmente conocidos como *lossless* y *lossy* respectivamente.

El primero tiene como objetivo principal la reproducción exacta de las imágenes, mientras que el segundo explota las propiedades del sistema visual humano para producir imágenes equivalentes (pero no idénticas) con menos datos, o subjetivamente, mejores imágenes con igual número de datos. Aún cuando los sistemas *lossless* reproducen la imagen igual a la original, la razón de compresión es menor comparada con la obtenida con los sistemas *lossy*.

Esta división de métodos de codificación es algo engañosa. Las señales analógicas no pueden ser reproducidas exactamente por ningún sistema real y cualquier representación de la imagen es meramente una aproximación. Los mejores métodos en efecto hacen uso de muchas propiedades visuales, ya que no está garantizado que un cambio en una representación digital debe necesariamente degradar una imagen y en muchas ocasiones esos cambios pueden producir mejor calidad.

7.1 Métodos de Compresión Sin Pérdidas Lossless.

Este método es utilizado cuando no es posible tolerar ningún cambio por insignificante que parezca en una imagen, dado que en algunas aplicaciones podría haber errores de apreciación, por ejemplo, si se cuenta con una radiografía, el diagnóstico de un médico podría variar si la calidad y la representación de la imagen original se ha modificado.

Existen varios métodos de compresión lossless. Entre los principales se encuentran: la codificación Huffman, la codificación aritmética y runlength.

7.2 Métodos de Compresión Con Pérdidas Lossy.

La estructura general para un esquema lossy es mostrada en la figura 7.1.



Figura 7.1 Estructura General de un Esquema de Compresión con Pérdidas (Lossy).

Incluye tres componentes : transformación o descomposición de la imagen, cuantización y codificación de símbolos. La importancia relativa de cada componente varía de una técnica lossy a otra, y no todas las componentes deben ser incluidas necesariamente en una técnica particular.

Como una regla general, los esquemas más sofisticados producen la mejor calidad que pueda ser obtenida para una razón de bits dada.

La descomposición de la imagen o *transformación* es hecha para reducir el rango dinámico de la señal, con el fin de eliminar la información redundante, o en general proveer una representación que pueda ser codificada mas eficientemente. Esta etapa es también hecha en técnicas lossless y generalmente es una operación reversible.

La diferencia primaria entre los esquemas lossy y lossless es la inclusión de la etapa siguiente en técnicas lossy, llamada *cuantización*. Mediante la cuantización de los datos, el número de posibles símbolos de salida es reducido. El tipo y grado de cuantización tiene un gran impacto en la razón de bits y en la calidad de un esquema lossy. También es deseable hacer la cuantización de tal forma que la secuencia de salida resultante pueda ser codificada posteriormente de manera eficiente.

El proceso de *codificación* de símbolos podría incluir técnicas tales como codificación huffman o codificación aritmética como un medio para obtener razones cercanas a la entropía de los símbolos fuente cuantizados, [RAB91].

En general cualquiera de las componentes de un esquema lossy puede ser implementado en un modo *adaptivo* o *no adaptivo*. Un esquema de compresión es adaptivo si la estructura de un componente o sus parámetros cambian localmente dentro de una imagen para tomar ventaja de las variaciones de las estadísticas locales. La adaptabilidad ofrece el potencial para producir un alto rendimiento a cambio de un incremento en complejidad.

7.3 Codificación por Transformación

Un esquema de codificación por transformación general consiste en subdividir una imagen de tamaño $N \times N$ en bloques mas pequeños de $n \times n$ y producir una transformación unitaria en cada subimagen. Una transformación unitaria es una transformación lineal reversible de quien el kernel describe un conjunto de funciones completas bases discretas ortonormales, [RAB91].

El fin de la transformación es correlacionar la señal original donde la correlación generalmente resulta en la distribución de la energía de la señal a través de únicamente un conjunto pequeño de coeficientes de la transformación. En esta forma muchos coeficientes pueden ser descartados después de la cuantización y previo a la codificación.

Una transformación puede ser unidimensional 1D o bien bidimensional 2D; una transformada es unidimensional, si es realizada a través de una sola dimensión de la imagen, por ejemplo a través de una línea o columna de píxeles. Una transformación 1D hecha sobre n píxeles es definida como una transformación de n puntos. Una transformación es bidimensional si es hecha en un bloque de dos dimensiones de píxeles.

Las transformadas 2D consideradas en este capítulo son separables, esto es, el kernel de la transformada puede ser descompuesto en dos kernels 1D especificando operaciones separadas vertical y horizontal. De esta forma una transformación separable en un bloque de $n \times n$ píxeles puede ser hecho en dos pasos. Primero una transformación 1D de n puntos es realizada a través de cada renglón del bloque y entonces otra transformación de n puntos 1D es hecha a través de cada columna de la salida resultante.

Algunas características que son deseables en una transformación cuando es usada para propósitos de compresión de imágenes son las siguientes:

Correlación de la imagen. La transformación idealmente correlaciona completamente los datos en un bloque, esto es, empaqueta la mayor cantidad de energía en el número mas bajo de coeficientes.

Rápida implementación. El número de operaciones requeridas para una transformación de n puntos es generalmente del orden de n^2 . Algunas transformaciones tienen implementaciones rápidas, lo cual reduce el número de operaciones a $(n \log n)$. Para una transformación 2D $n \times n$ separable hacer la transformación 1D de renglón y columna en sucesión reduce el número de operaciones a $(2n^2 \log n)$ en vez de (n^4) . [RAB91]

7.3.1 Transformada Coseno Discreta.

Para aplicaciones de procesamiento de imágenes, la Transformada Coseno Discreta (DCT, Discrete Cosine Transform) 2D de un bloque de $n \times n$ píxeles es definida como :

$$F[u, v] = \frac{4C(u)C(v)}{n^2} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} f[j, k] \cos \left[\frac{(2j+1)u\pi}{2n} \right] \cos \left[\frac{(2k+1)v\pi}{2n} \right]$$

Donde:

u, v son variables de frecuencia discreta (0, 1, 2, ..., n-1)
 $f[j, k]$ = píxeles de la imagen de tamaño $n \times n$.
 $F[u, v]$ = DCT resultante.

y la transformada inversa :

$$f[j, k] = \frac{1}{N^2} \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} C[u] C[v] F[u, v] \cos \left[\frac{(2j+1)u\pi}{2n} \right] \cos \left[\frac{(2k+1)v\pi}{2n} \right]$$

Donde :

j, k son los índices del pixel resultante (0, 1, 2, 3, ..., N-1)
 $F[u, v]$ = DCT de $n \times n$ que va a ser antitransformada.
 $c[\lambda] = \begin{cases} \frac{1}{\sqrt{2}}, & \text{para } \lambda = 0 \\ 1, & \text{para } \lambda = 1, 2, 3, \dots, n-1 \end{cases}$
 $f[j, k]$ = DCT inversa resultante de tamaño $N \times N$.

7.4 Algoritmo JPEG

La estructura del algoritmo propuesto JPEG consta de tres componentes principales. El primero es un sistema que provee un simple y eficiente algoritmo que es adecuado para muchas aplicaciones de codificación de imágenes. El segundo es un conjunto de características del sistema extendido para satisfacer un amplio rango de aplicaciones. Finalmente un método independiente lossless es incluido para aplicaciones que requieren ese tipo de compresión.

7.4.1 Descripción del Sistema JPEG.

La imagen original es particionada en bloques de 8x8 pixeles y cada bloque es transformado independientemente usando la DCT.

Todos los coeficientes de la transformada son normalizados (ponderados) mediante la aplicación de un arreglo de normalización definido por el usuario que es fijo para todos los bloques. Cada componente del arreglo de normalización es un entero de 8 bits y es pasado al receptor como parte de la información del encabezado que es requerido para cada imagen.

Los coeficientes normalizados son entonces cuantizados uniformemente redondeando al entero más cercano. El arreglo de normalización (figura 7.2a) puede ser visto como un escalar del cuantizador que controla la cantidad de error de cuantización introducido en cada coeficiente. La función de sensibilidad al contraste (SVH) puede ser usada como una guía para desarrollar un arreglo de normalización que pondera a cada coeficiente acorde a su importancia perceptual.

La cuantización de los coeficientes produce muchos ceros, especialmente en las frecuencias más altas. Para tomar ventaja de esos ceros, el arreglo 2D de los coeficientes de la DCT es formateado en un vector 1D usando un ordenamiento de zigzag (figura 7.2b). Este ordena los coeficientes en orden decreciente de su energía promedio (o bien en orden del incremento de su frecuencia espacial) con el fin de crear grandes corridas de valores ceros.

Es ahora posible codificar el vector unidimensional de coeficientes mediante runlength y Huffman admitiendo un número de coeficientes determinado con el propósito de alto rendimiento del sistema, y mayor razón de compresión, [RAB91].

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figura 7.2 a Arreglo de Normalización de 8 x 8

0	1	3	6	10	15	21	28
2	4	7	11	16	22	29	36
5	8	12	17	23	30	37	43
9	13	18	24	31	38	44	49
14	19	25	32	39	45	50	54
20	26	33	40	46	51	55	58
27	34	41	47	52	56	59	61
35	42	48	53	57	60	62	63

Figura 7.2b Ordenamiento de Coeficientes zigzag

El proceso de decodificación es simple, la cadena de bits codificada es decodificada mediante Huffman y las tablas asociadas y con esto, el arreglo de coeficientes cuantizados DCT es recuperado, cada coeficiente es desnormalizado al multiplicarlo por su correspondiente componente de la matriz de normalización. El arreglo resultante es antitransformado por la DCT inversa para obtener una aproximación del bloque de la imagen original. El error de reconstrucción resultante depende de la cantidad de cuantización la cual es controlada por la matriz de normalización.

7.5 Implementación del Esquema de Codificación.

El modelo de compresión propuesto es un sistema basado en un esquema de compresión con pérdidas lossy, debido a que la razón de compresión en un sistema sin pérdidas es muy baja dado que existe una gran dependencia entre los píxeles de las imágenes halftone.

El modelo ha sido diseñado para cubrir el estándar de compresión JPEG para imágenes de tono continuo. Dicho modelo consta de 4 etapas; la primera es el mapeo del dominio binario al dominio de escala de grises, y las tres etapas típicas de un sistema lossy, descritas anteriormente. El modelo produce una buena razón de compresión comparado con métodos convencionales de compresión sin pérdidas, además de ser más eficiente que varios diseños propuestos en la literatura, produciendo resultados de alta calidad de imágenes a una razón de compresión excelente. La figura 7.3 ilustra el esquema propuesto.

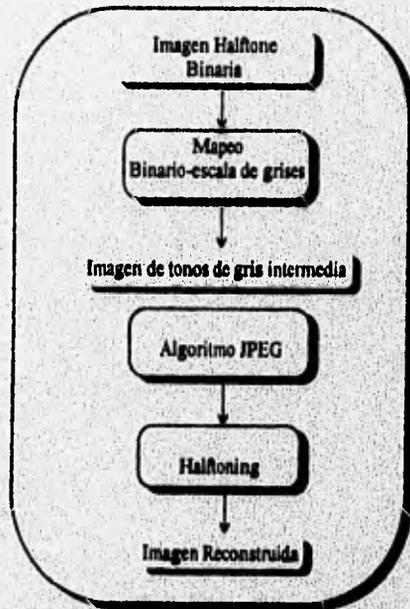


Figura 7.3 Esquema de Codificación Propuesto para imágenes Halftone de alta calidad.

7.6 Mapeo del Dominio Binario al Dominio de Tonos de Grises.

La contribución original es utilizar filtros no lineales para codificar imágenes halftoning para comprimir las en el dominio de escala de grises, esto produce mejor calidad de la imagen que comprimir directamente las imágenes binarias, el método es de baja complejidad computacional y puede trabajar con cualquier algoritmo halftoning.

A pesar del hecho que la razón de datos de imágenes halftone ya ha sido reducida de la razón original (1 bit contra 8 o mas bits por pixel), grandes imágenes halftone aun requieren de una gran cantidad de datos.

Técnicas de compresión lossy puede ser utilizada para imágenes halftone con un buen rendimiento a causa de que algún error de compresión puede ser enmascarado u ocultado mediante el proceso halftone.

Si se tiene la imagen de tonos de gris de la cual la imagen halftone fue generada , entonces es posible obtener buenos resultados de compresión mediante comprimir la imagen de escala de grises antes de que sea halftoneada [NEU91]. Mejores resultados pueden ser obtenidos mediante comprimir conjuntamente y halftonear la imagen [VAN92].

Supongamos que deseamos comprimir una imagen halftone, pero no tenemos acceso a las imagen de tonos de gris de la cual fue derivada. El aplicar cuantización vectorial directa VQ a imágenes halftone binarias, permite obtener una calidad de imagen muy pobre a una razón de bits de 0.25 bits por pixel. De esta manera, como fue sugerido por Limb [MIN93], es recomendable diseñar un codificador para mapear las imágenes halftone de regreso al dominio de tonos de grises.

7.6.1 Codificador No Lineal

Típicamente una calidad mucho mejor de la imagen es obtenida cuando se comprimen imágenes de tonos de gris en comparación a la obtenida mediante comprimir imágenes binarias.

Consideramos "codificar" imágenes halftone para que puedan ser comprimidas en el dominio de tonos de gris.

Capítulo 7

Convertir una imagen halftone a escala de grises es similar a diseñar un codificador en modulación sigma-delta $\Sigma\Delta$, una técnica para conversión sobremuestreada analógica a digital, [MIN93].

Halftoning puede ser visto como una forma de modulación bidimensional típicamente para codificar la salida sigma-delta $\Sigma\Delta$. La aproximación a la entrada continua es tomada como el promedio de la salida binaria sobre un intervalo de la longitud de la razón sobremuestreada usada para muestrear los datos de entrada. De tal forma que los humanos perciben la ilusión de tonos de gris continuos mediante un promedio espacial local sobre un área de patrones binarios en la imagen halftone.

Lo que implica que la imagen de tonos de gris puede ser reconstruida aplicando un filtro paso bajas a la imagen halftone. Sin embargo un filtrado paso bajas lineal convencional usualmente no produce imágenes de tonos de gris reconstruidas aceptables.

La idea original de implementar un codificador no lineal para las imágenes halftone basadas en una tabla de búsqueda fue sugerido por Gersho y Hein, [MIN93].

El proceso se basa en la construcción de tablas de búsqueda para lograr una correspondencia binaria y escala de gris entre la imagen halftone de entrada y la imagen de salida de tono continuo.

Utilizando una máscara binaria de 3×3 para formar un índice dentro de una tabla de tamaño $2^9 = 512$ patrones binarios distintos.

El índice satisface :

$$\text{índice} = \sum_{i=0}^8 (b_i 2^i)$$

donde b_i es el valor binario del pixel i . Para cada índice (bloque de 3×3), se calcula un valor de salida de tono de gris para el pixel central de el bloque.

Para todas las ocurrencias de un bloque dado en la secuencia de instrucciones binaria, la salida es simplemente el valor esperado del pixel central g en los bloques correspondientes en la secuencia de instrucciones de tonos de grises. Un esquema similar basado en un bloque de 3×3 para reconstruir imágenes halftone fue usado primero por Netravali y Bowen en imágenes de error de difusión, [MIN93]. Una vez que la tabla ha sido construida, reconstruimos la imagen de tonos de gris de su versión halftone basados en la correspondencia binaria a escala de grises.

7.7 Compresión de Imágenes Halftone en el Dominio Binario

Primero investigamos compresión Lossles, puesto que tiene la ventaja de que la imagen original puede ser recuperada perfectamente de los datos comprimidos. Claramente la compresión lossless puede producir únicamente razones de compresión muy bajas en esta imagen, como una alternativa investigamos compresión lossy.

Un enfoque simple para técnicas de compresión lossy de imágenes halftone es aplicar VQ directamente a los datos binarios, la calidad de la imagen es muy pobre. En el siguiente capítulo se muestra la razón de compresión para varias implementaciones de codificación mediante métodos de compresión convencionales. Es claro que la razón de compresión es muy baja debido a la gran dependencia entre los píxeles y a que estos métodos hacen uso de longitud de las corridas.

7.8 Compresión de Imágenes Halftone en el Dominio de Tonos de Gris

Muchos esquemas de compresión pueden ser utilizados para comprimir la nueva imagen de tonos de gris reconstruida, entre ellos podemos aplicar cuantización vectorial, transformada coseno, wavelets etc. Sin embargo, buscando satisfacer un estándar de compresión utilizamos codificación JPEG, basándonos en un codificador para transformar imágenes binarias a niveles de gris.

Producimos una imagen de tonos de gris a 0.27 bpp. Aplicamos halftone a esta imagen para obtener una imagen halftone comprimida la cual requiere únicamente de 1/4 del espacio de almacenamiento de la imagen halftone original. La calidad total es aún muy cercana a la imagen original.

Capítulo 8

Resultados, Perspectivas y Conclusiones

La calidad de visualización de una imagen vista a través de un dispositivo binivel (ya sea mediante pantalla o impresión), es una cualidad subjetiva y depende del criterio del observador determinar si es buena, regular o mala.

A pesar del hecho de que la binarización de imágenes tiene ya muchos años de ser usada, cabe destacar que algunos de los métodos empleados anteriormente siguen siendo de utilidad y son la base para nuevos esquemas en la presentación de resultados.

Dos muestras representativas de imágenes vistas a través de un dispositivo binivel sirven como referencia en la comparación de los resultados; los cuales son mostrados a través del Sistema de Producción de Imágenes (sección 3.2).

La figura 8.1 muestra la popular "lena", las dimensiones de esta imagen son de 512 x 512 píxeles, la binarización de esta pintura fue producida mediante Thresholding (sección 4.1), la forma más simple de producir una representación binaria de la imagen con tonalidades de gris. La figura 8.2 es otro ejemplo ilustrativo de imágenes con tonos de gris, visto mediante un dispositivo binivel. Las dimensiones de esta figura son 650 x 670 píxeles.

Como se puede observar, no se aprecian detalles finos en ninguna de las pinturas, los contornos y formas se pierden entre los cambios bruscos de blanco a negro.



Figura 8.1 Imagen Original "Leona"



Figura 8.2 Imagen Original "Lobos"

8.1 Método de Bayer y Dither Ordenado

Las figuras 8.3, 8.4 muestran los resultados obtenidos a través de patrones fijos, como el de Bayer y Dither ordenado (sección 4.12 y 4.13).

Aún cuando se aprecian artefactos en las imágenes, esto es, patrones reconocibles a simple vista, algunos detalles de forma y mas claridad en la escena son posibles. 65 niveles de gris pueden ser apreciados en estas imágenes. Con el método de Bayer, se pueden apreciar la ilusión de algunos tonos de gris, a cambio, la brillantez es escasa y los bordes no son suaves.



Figura 8.3 Método de Bayer

Capítulo 8

En dither ordenado parece que la imagen descansa sobre una malla de puntos, y la claridad es notable, sin embargo, a diferencia del patrón de bayer, los patrones que generan los tonos de gris están más aislados, lo que provoca que la imagen sea más clara y algunos detalles principalmente en los contornos, sean perdidos.

Aún cuando la forma de operación de los dos métodos es la misma, la distribución de los elementos de la matriz de comparación usada, es fundamental para la correlación entre patrones, esto es, se busca tener agrupados siempre, patrones semejantes.



Figura 8.4 Dither Ordenado

8.2 Métodos de Difusión de Error

Los métodos de difusión de error (sección 4.2) son ilustrados en las figuras 8.5, 8.6, 8.7, 8.8 y 8.9 como puede observarse, la apariencia de las imágenes generadas por los diferentes filtros es muy semejante, la única variación que se aprecia es la textura de las pinturas. La rugosidad de algunas de estas imágenes es debida a que utiliza un vecindario más extenso para propagar el error, lo cual produce un promedio entre varios pixeles, alternatively, se puede seguir un proceso de filtrado para lograr suavizar las imágenes, a cambio de una atenuación en los bordes.

Los métodos de difusión de error producen la mejor calidad que puede ofrecer un algoritmo Halftoning, es cierto que hace muchas mas operaciones que los algoritmos que utilizan operaciones de puntos, sin embargo, vale la pena pagar ese precio a cambio de la calidad que ofrecen, además de que su tiempo de procesamiento no es muy alto.



Figura 8.5 filtro de Floyd y Steinberg



Figura 8.6 filtro Burkes



Figura 8.7 filtro Sierra



Figura 8.8 filtro Jarvis, Judice y Ninke



Figura 8.9 filtro Stevenson y Arce

8.3 Modificación de la Escala de Tono y Afinamiento

Las figuras 8.10, 8.11a y 8.11b muestran la modificación de la escala de tonos utilizando la función de mejoramiento de contraste y mapeo de brillantez (sección 3.4). Esta función puede ser manipulada multiplicando la función por un escalar, ajustando el contraste y brillantez a la imagen.



Figura 8.10 Modificación de Contraste



Figura 8.11a Modificación de Brillantez



Figura 8.11b Modificación de Brillantez

La etapa de afinamiento puede ser mezclada con el ajuste de la escala de tonos, además del método halftoning usado, para compensar los efectos visuales que se pudieran presentar. Las figuras 8.12a, 8.12b, 8.13, 8.14 y 8.15 ilustran el resultado final de aplicar el preprocesamiento. Hasta este punto, las imágenes han salido del preprocesador dedicado (sección 3.2) del Sistema de Producción de Imágenes propuesto, ahora las imágenes ya están en condiciones de ser presentadas ante cualquier dispositivo.

Haciendo una comparación del rendimiento del Sistema de Producción de Imágenes que se propuso, los resultados obtenidos son de mayor calidad comparados con las imágenes originales, la eficiencia de los métodos de difusión de error, está por encima de los métodos que utilizan patrones fijos, independientemente de el tiempo de procesamiento, que es en sí, casi el mismo. Los diferentes filtros que propagan el error usados en nuestros algoritmos ofrecen resultados muy similares, es aquí donde el observador puede a su juicio determinar cual ofrece para él, el mejor resultado y se adapta mejor a sus necesidades, sin embargo, señalaré en este punto que el filtro de Floyd y Steinberg ofrece la respuesta más rápida.

Para un dispositivo en particular, debemos elegir la mejor alternativa que ofrecen los diferentes algoritmos halftoning, de ser posible, modificar la escala de tono y el afinamiento.



Figura 8.12a filtro de Floyd y Steinberg Modificado



Figura 8.12b filtro Floyd y Steinberg Modificado



Figura 8.13 filtro Sierra Modificado



Figura 8.14 filtro Jarvis, Judice y Ninke Modificado



Figura 0.15 filtro Floyd y Steinberg Modificado

8.4 Desplegado

La implementación de los algoritmos halftoning para computadoras personales se basa en la detección del adaptador gráfico que soporta ese sistema, se puede implementar de varias formas, una de ellas es mediante lenguaje ensamblador con la finalidad de buscar una respuesta inmediata del sistema, sin embargo se tienen muchas complicaciones para realizar esto, ya que existe una gran variedad de adaptadores para las diferentes arquitecturas. Una forma alternativa que ofrece una buena posibilidad de implementación es utilizar las herramientas que nos ofrece el lenguaje C, como son: la posibilidad de detectar el adaptador gráfico con una sola instrucción además de obtener las dimensiones físicas del video y la posibilidad para desplegar los píxeles en forma independiente.

La única consideración que se debe tener en cuenta es la posibilidad de implementarlo para diferentes arquitecturas y diferentes sistemas operativos.

Para computadoras personales de la familia 8086 la implementación fue realizada con Turbo C, el cual ofrece las ventajas antes descritas, además de poder combinarse con el lenguaje ensamblador para lograr una mayor velocidad.

Una de las características distintivas de las diferentes arquitecturas, es que la razón de aspecto es diferente. La resolución del monitor determina la calidad de la imagen a través de este. Los sistemas 8086, 80286 cuentan con la resolución más limitada de todos los sistemas. Las imágenes apreciadas en estas máquinas es pobre a una distancia de observación normal, es decir 75 cm aproximadamente, sin embargo, verlas alrededor de 1 metro de distancia, el ojo humano hace una integración del contenido de la pantalla y la calidad mejora notablemente.

La simulación vista por medio de otro controlador de gráficos monocromático para computadoras personales como el EGA, VGA es mucho mejor y las imágenes se aprecian con una calidad más aceptable debido a cuentan con una resolución mayor.

En estaciones de trabajo, donde las resoluciones son muy altas, la imagen tiene excelente calidad y se pueden apreciar detalles finos de la pintura. Se creó una pequeña aplicación mediante la interface gráfica Open look además del lenguaje C ANSI. (figuras 8.16)

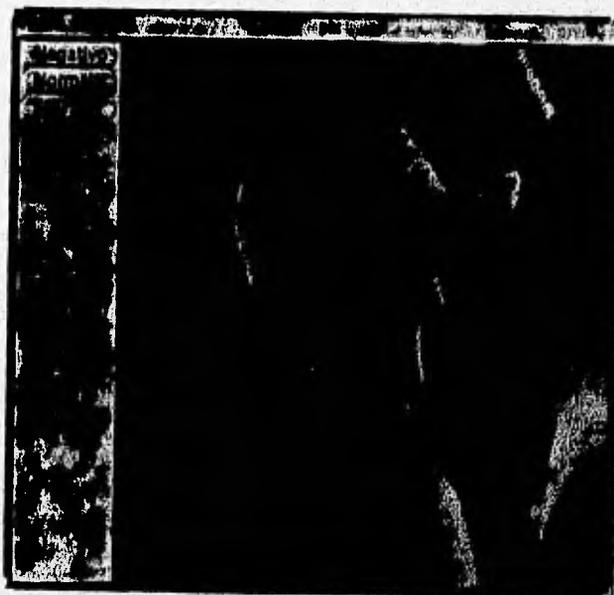


Figura 8.16 Implementación del Sistema de Desplegado

8.5 Impresión

Las ilustraciones mostradas durante el texto, es una implementación de los algoritmos halftoning en impresoras láser. Una muestra mas representativa de una impresión LaserJet es mostrada en la figura 8.17a , junto con la misma imagen impresa por una impresora de matriz de puntos (Epson FX80) Figura 8.17b.

Las diferencias entre ambas imágenes pueden ser apreciadas ya que las dos están impresas con la misma resolución (720 puntos por pulgada)

La razón de aspecto es importante, las diferentes densidades de impresión para la impresora de matriz de puntos determina la cantidad de papel que utiliza. El mejor resultado es logrado para el modo 06H para pixeles cuadrados, el cual ofrece la razón de aspecto más uniforme , es decir, el ancho y largo del pixel es casi el mismo.

Esta implementación se puede llevar a diferentes arquitecturas de impresoras, adaptándose a cualquier sistema. La única consideración que se debe tomar en cuenta es la resolución de la maquina, una mayor resolución es preferible. A cambio, se obtienen impresiones de imágenes de tono continuo a precios accesibles y de calidad aceptable.

8.6 Compresión

Debido a que existe una gran dependencia entre un grupo de pixeles los cuales generan un patrón halftone, y dado que el número de puntos negros y blancos en el esta altamente relacionados, los métodos de compresión para imágenes binarias como son Runlengt, Huffman o codificación aritmética, los cuales hacen uso de la longitud de la variable de codificación no son buenos para representar imágenes Halftone. Algunos resultados son mostrados para ilustrar en la tabla 8.1 . Es por esto que se han planteado formas alternativas para hacer la codificación [MIN93].

Múltiples esquemas de compresión pueden ser elaborados para un tipo específico de imágenes, para el caso particular de imágenes halftone, se han diseñado diferentes enfoques, buscando la mejor forma de codificar eficientemente la información con el menor número de datos posibles, [VAN92].



Figura 8.17a Imagen Impresa en Impresión Laser
720 dpi



**Figura 8.17a Imagen Impresa en Matriz de Puntos
720 dpi**

Dichos esquemas están basados en la compresión de imágenes binarias, las cuales utilizan como herramientas principales los métodos convencionales de compresión, casi todos ellos han buscado la manera de identificar los patrones halftone generados mediante el procesamiento a través de operaciones de bloque o bien operaciones de ventanas, [TAK93].

Sin embargo y aunque los resultados son aceptables, el número de etapas por las cuales pasa el codificador se extiende complicando las operaciones, [COS94].

Un diseño original que puede haber sido implementado de manera eficiente el cual produce muy buenos resultados es realizado mediante la transformación de la imagen binaria halftone a una imagen con niveles de gris, la cual puede ser codificada mediante técnicas de compresión para imágenes de tono continuo, las cuales ofrecen una buena razón de compresión (sección 7.5).

La imagen halftone original es vista en la figura 8.18a, la cual ha sido procesada mediante el sistema de producción de imágenes propuesto del capítulo 3. La imagen de tonos continuo obtenida a partir de la imagen halftone mediante el codificador no lineal es mostrada en la figura 8.18b, esta impresión fue realizada mediante una impresora postscript que genera algunos tonos de gris.

Aplicando el sistema de compresión basados en un esquema JPEG el cual hace uso de la transformada coseno y de codificación Huffman, obtenemos una imagen codificada la cual utiliza únicamente $\frac{1}{4}$ del espacio de almacenamiento que requiere la imagen halftone original.

La imagen de tono continuo descomprimida es mostrada en la figura 8.19a, en primera instancia se puede apreciar que la imagen se ha subdividido en pequeños cuadros de tamaño 8x8, dentro de cada uno de ellos se puede ver que el mayor contenido de información por cuadro se encuentra en la parte superior izquierda, esto es debido a la selección de los coeficientes de la transformada coseno.

La figura 8.19b muestra la imagen reconstruida. Se puede apreciar que el contraste se ha modificado, la agrupación de puntos también ha variado y los puntos se ven más dispersos que en la imagen original, puesto que se admitieron algunas pérdidas de coeficientes en el sistema, sin embargo, la calidad de la imagen es muy buena. En comparación con los resultados presentados en el diseño original del codificador no lineal [MIN93], el cual utiliza un cuantizador vectorial mediante un árbol binario podado, los resultados son muy similares y las razones de compresión muy cercanas.



figura 8.18a Imagen Halftone Original



Figura 8.18b Imagen Binaria a Tono Continuo



figura 8.19a Imagen de tono continuo descomprimida



figura 8.19b imagen halftone reconstruida

La tabla 8.1 muestra los resultados de los diferentes esquemas de compresión, [OVI96], los resultados de los dos primeros esquemas demuestran que, por un lado Huffman comprime alrededor de 0.1%, lo cual indica que el tamaño de la imagen es casi el mismo, en el segundo caso, el signo negativo indica que la codificación mediante Runlength en vez de reducir el tamaño lo triplica, lo cual es el resultado de la gran dependencia que existe entre los elementos de una imagen halftone, y puesto que estos métodos hacen uso de la longitud de la variable de codificación.

La codificación aritmética por otra parte, solo reduce el tamaño de la imagen un 10 % lo cual lo hace más eficiente que los dos métodos anteriores.

Los resultados de la cuantización vectorial [MIN93], logran la mejor tasa de compresión de los métodos mostrados, comprimen cerca de un 75 % la imagen, además de que la calidad de la imagen es bastante buena. En comparación, el método JPEG desarrollado en este proyecto obtiene la segunda mejor marca, comprime alrededor de 70 % la información. En este método es posible obtener mayor nivel de compresión, a cambio se produce mas degradación en la imagen, puesto que el número de coeficientes posibles a seleccionar del arreglo DCT limita las pérdidas en el contenido de la información.

	Nivel de Compresión	Entropía	Porcentaje
Huffman	1.01:1	0.999 bpp	0.1 %
Runlength	0.26:1	3.846 bpp	-284.6 %
Codificación Aritmética	1.12:1	0.892 bpp	10.8 %
VQ	3.78:1	0.264 bpp	73.6 %
JPEG	3.4 :1	0.294 bpp	70.6 %

Tabla 8.1 Niveles de Compresión para Imágenes Halftoning

8.7 Conclusiones

El Sistema de Producción de Imágenes propuesto, el cual es la base de este proyecto, ofrece una excelente opción para crear imágenes de tono continuo con únicamente elementos binarios. Los resultados obtenidos son sin lugar a dudas de mejor calidad que las imágenes originales, podemos afirmar que el peor de los casos es la presentación de las imágenes mediante un dispositivo binivel sin procesar. Las implementaciones de estos algoritmos a través de diferentes medios de desplegado puede ser posible una vez procesada la información y de no ser suficiente, podemos hacer una serie de ajustes para adaptarnos a un dispositivo en particular, sin embargo, el proyecto tiene como finalidad lograr la generalidad.

Considerando que existen imágenes de diferente naturaleza, es conveniente puntualizar que para que el Sistema de Producción de Imágenes nos de sus mejores resultados, es necesario hacer una serie de pruebas y selección del algoritmo que sea mas conveniente para una aplicación específica y que satisfaga nuestras necesidades.

Una referencia de consulta que esta disponible para obtener el programa fuente de los algoritmos aquí desarrollados puede ser consultada en la Sección de Eléctrica de la División de Estudios de Posgrado de la Facultad de Ingeniería o bien vía e-mail a la dirección oscar@verona.dephi.unam.mx la cual puede servir como guía para conocer la forma de implementación.

Se cumplieron los objetivos propuestos, esperando este sea una pequeña contribución para el desarrollo de nuevos proyectos de investigación.

8.8 Perspectivas

La constante evolución de los sistemas nos mantiene en una etapa de transición entre el conocimiento y el tiempo. Es necesario estar al margen de los acontecimientos. Las perspectivas al cambio han creado nuevos proyectos de desarrollo de tal manera que los algoritmos serán utilizados para crear la ilusión no solo de tonos de gris, sino también de color, a partir de los tres colores primarios (rojo, verde, azul), con esto, una impresora, o bien un monitor requerirán producir únicamente tres colores y se podrán obtener patrones halftone que contengan una gran gama de color.

La limitación física de los equipos, el costo y las necesidades de algunas de las aplicaciones, nos están haciendo volver a las viejas herramientas, aquellas que inicialmente eran muy limitadas pero que sirvieron de soporte para crear nuevas perspectivas.

Apéndice A

Códigos de Control FX-80

La FX-80 es capaz de almacenar hasta 2k bytes de datos en su buffer de entrada para procesamiento rápido de datos.

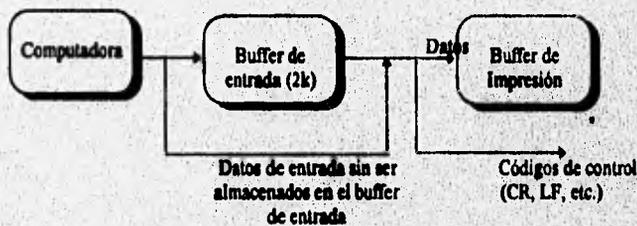


Figura A.1 Envío de Datos Computadora-Impresora

Normalmente los datos desde la computadora es enviado a la impresora sin ser almacenado en el buffer de entrada, como se muestra en la figura A.1. Cuando la función del buffer de entrada esta habilitada, todos los datos de salida de la computadora serán primero almacenados en el buffer de entrada antes de ser enviados a la impresora.

Dos tipos de datos son almacenados en el buffer de entrada : códigos de control y datos. Los datos o la información en sí, son enviados al buffer de la impresora. Estos datos son impresos cuando se ejecuta algún código de control, tales como CR, o bien, cuando el buffer llega a llenarse, [EPS90].

Inicialización de la impresora

ESC @

Cuando este código es de entrada, la impresora es inicializada, causando que todos los datos del buffer de impresión sean limpiados.

Apéndice A

Impresión de Mapas de Caracteres

Modo Maestro

Este modo permite diferentes modos de operación para la impresión de imágenes de diferentes densidades. La sintaxis es la siguiente:

ESC * MODO LO HI

m	Modo	Puntos/ 8"	Velocidad de Cabera
0	Densidad Normal	480	16
1	Doble Densidad	960	8
2	Doble Densidad Doble velocidad	960	16
3	Densidad Cuádruple	1920	8
4	Gráficos CRT	640	8
5	Gráficos de Plotter	576	12
6	Gráficos II CRT	720	8

El número de datos de la imagen serán enviados mediante los números LO y HI en forma hexadecimal o decimal. Donde HI es la parte alta y LO es la parte baja del entero que contiene la longitud de la línea en pixeles.

ESC K es lo mismo que modo 0

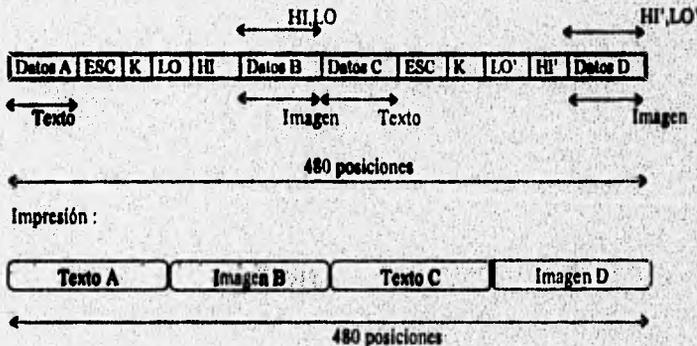
ESC L es lo mismo que modo 1

ESC Y es lo mismo que modo 2

ESC Z es lo mismo que modo 3

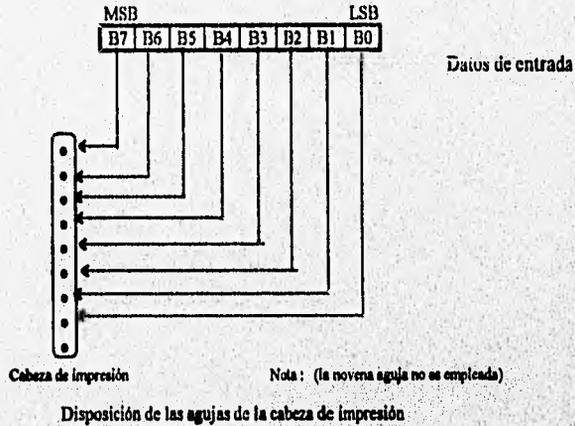
El número de puntos imprimible por línea depende de la densidad de los puntos que produzca el modo en el cual se está trabajando. Por ejemplo, en el modo de densidad normal, el máximo número de posiciones de puntos imprimible por línea es de 480. Entonces si especificamos un número mayor de puntos, solo se garantiza que se imprimirán 480 puntos máximo por línea.

Después de que se ha completado la impresión, la impresora regresará automáticamente a modo texto. En una línea se puede mezclar texto e imágenes, siempre y cuando se limite la longitud de la línea.

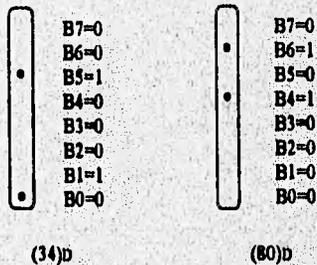


Códigos de Control Epson FX80

Disposición física de la cabeza de impresión, relación posición e información de entrada.



Si un bit es 1, la aguja de la cabeza de impresión golpea, si el bit es 0 no golpea. Por ejemplo, pensemos que los datos de entrada son los siguientes:



donde • denota bit 1 y un blanco denota bit 0

Es importante hacer notar que para imprimir una imagen, es necesario hacer una transformación de la información de tal forma que pueda ser interpretada por la impresora, (sección 6.5), [EPS90].

Apéndice B

Tipos de impresoras.

Impresión por impacto.

Existen principalmente dos técnicas en impresión por impacto; la técnica de impresión de caracteres completos y el sistema de formación de caracteres.

Impresión de Margarita.

Es muy similar a la de una maquina de escribir. El caracter ya esta formado en un modelo de metal o de plastico endurecido y da lugar a la impresión de un caracter completo y de contomo regular.

El resultado de este tipo de impresión es igual al de una maquina de escribir de alta calidad. La mayor parte de las impresoras de calidad son del tipo de margarita. Se le aplica el nombre de margarita por la forma del dispositivo sobre el que estan colocados los modelos de los caracteres. En un disco de metal o de plastico al final de los cuales se encuentra el molde de cada caracter. La impresora hace girar la margarita hasta colocar en posición el caracter elegido y entonces un martillo golpea la lámina protectora del modelo de caracter contra la cinta tintada y el papel, produciendose una impresión de gran nitidez, (ORO90).

Impresión de Matriz de Puntos.

En la técnica de formación de caracteres o también llamada técnica de matriz de puntos, los caracteres son el resultado de la impresión de una serie de puntos dispuestos de una forma determinada. A partir de la matriz de puntos se puede generar todo el conjunto de letras mayúsculas, minúsculas, números, signos de puntuación y símbolos especiales.

Aunque a simple vista la calidad de impresión de la de margarita es mejor, las aplicaciones de una matricial son mucho mayores que la de margarita, e incluso la calidad que se puede lograr con una matricial es muy similar a la de margarita.

Apéndice B

La impresión por impacto o matricial se realiza a base de un cabezal de agujas. Esta tecnología tiene algunos inconvenientes, como es el ruido causado durante el trabajo, debido al movimiento de las agujas (que golpean unas 1500 veces por segundo sobre la cinta), lo que resulta un poco molesto, [ORO90].

Impresión de Chorro de Tinta.

Esta clase de impresoras suelta un fino chorro de tinta sobre el papel, desde un depósito. Existen dos métodos para lanzar la tinta desde el depósito : el continuo y el de descarga a demanda, [KIR93].

En el primero se realizan constantes demandas de tinta , que son desviadas por un campo eléctrico hacia los puntos que lo requieran. Si no existe ese campo eléctrico, la tinta cae en un recipiente donde es reciclada.

En la descarga a demanda sólo se lanza la tinta cuando se solicita. Los primeros problemas con este tipo de impresoras se debían a la propia tinta, que tenía tendencia a solidificarse en el surtidor. Actualmente existen tintas que no se coagulan fácilmente y cabezales de nuevo diseño con un mecanismo de limpieza incorporado, que funciona después de la impresión o cuando no se esta imprimiendo. El lavado del cabezal elimina cualquier residuo y mantiene la perfecta fluidez del chorro de tinta.

Impresoras Electrostáticas.

El corazón de las impresoras láser esta en el tambor cargado de electricidad, que atrae el toner para luego pasar la copia al papel. Este método de carga se conoce como indirecto, ya que la carga no va directamente al papel, sino a través del tambor.

La impresión electrostatica se conoce como método directo. En lugar de aplicarse sobre el tambor, un cabezal especial pasa la carga directamente al papel.

El principal inconveniente de este sistema reside en la necesidad de emplear papeles especiales, impregnados de un material que retiene la carga eléctrica. A cambio, la ausencia de tambor reduce considerablemente el costo.

Este tipo de tecnología opera por la aplicación de carga eléctrica que es recogida por un electrodo colocado detrás del papel. Cuando esto ocurre, se activa un pixel concreto del papel que acepta el tóner. La hoja de papel cargado pasa sobre el tóner líquido compuesto de diminutas partículas de grafito en suspensión, que se adhieren a las zonas cargadas y reaccionan mediante una sustancia química presente en el papel, que las fija definitivamente impidiendo que se borren. No se requiere de un proceso térmico y el líquido se elimina por aspiración, [KIR93].

Características Principales de una Impresora.

Compatibilidad y Emulaciones.

Es importante que la impresora que se seleccione esté soportada por el paquete de software que se vaya a utilizar, o bien emule a algunas de las impresoras soportadas.

Facilidad de Uso y Programación.

Principalmente en la colocación del papel, la programación de diferentes tipos de letra, el cambio de los parámetros de funcionamiento, etc. La facilidad y rapidez de todas estas tareas sobre la máquina son fundamentales para lograr una alta eficacia en su uso cotidiano. Una documentación fácil, clara y concisa.

Flexibilidad.

Por flexibilidad se debe entender la posibilidad que puede tener la impresora de mejorar sus prestaciones para adaptarse a las necesidades específicas del usuario. Prestaciones tales como la posibilidad de ampliar la memoria o el buffer, capacidad de insertar cartuchos con nuevas fuentes o diferentes emulaciones, posibilidad de utilizar varias interfaces para conectarse a la computadora, etc. Todo esto puede servir por si las necesidades cambian, el usuario pueda seguir utilizando la impresora.

Además de la capacidad de una impresora para adaptarse a más de un tipo de aplicación. Manuales de usuario, instalación, y soporte.

Apéndice B

Costo de utilización.

Este puede ser uno de los factores determinantes para la decisión por un tipo de impresora u otro. En general se debe de tomar en cuenta el costo de los consumibles, de adquisición y de mantenimiento, el mayor costo por copia corresponde a las impresoras láser, después las de inyección de tinta y por último las de matriz de punto. Puede haber ocasiones en que el costo por página de una impresora haga pensar en inclinar nuestra decisión hacia otra de características inferiores; pero hay que tener en consideración, si es significativo un sobreprecio en los gastos totales y también si realmente nos interesa esa calidad inferior.

Precio.

Esta debe ser la última consideración después de haber analizado todos los factores anteriores. Es claro que es importante, pero es mas importante seleccionar el producto mas adecuado a las necesidades reales.

GLOSARIO

Adaptador Gráfico Controlador que determina la máxima capacidad de gráficos del sistema, además de desplegar la memoria de video en pantalla.

Address Número de ubicación particular de memoria o de almacenamiento periférico.

Algoritmo Secuencia de pasos ordenados para la solución de un problema.

Aliasing Superposición de espectros. Apariencia escalonada en líneas o bordes (rugosidad).

Anti-aliasing (suavizamiento) técnicas que se utilizan para suavizar la apariencia dentada de las líneas diagonales, por ejemplo los pixeles que rodean los bordes de una línea son llenados con varios tonos grises o de color con el fin de amortizar el filo con el fondo.

Arreglo Conjunto ordenado de elementos de datos. Un arreglo de una dimensión es llamado vector, y un arreglo de dos dimensiones es llamado matriz.

Artefacto Efectos visuales no deseables derivados del procesamiento de la imagen.

Aspect ratio Razón de aspecto. Relación entre ancho y alto de un marco, pantalla o imagen.

Background fondo, segundo plano. El color de la base del fondo de la pantalla.

Baud Bits por segundo. Una medida usada para en transferencia de datos via líneas telefónicas.

Bidimensional Dos dimensiones.

Binivel : Dos niveles, blanco y negro, o bien, unos y ceros.

Binarizar Transformación de imágenes multinivel a únicamente dos tonos.

BIOS (Basic Input Output System) Sistema Básico de Entrada y Salida Conjunto de rutinas de software que contiene las instrucciones detalladas para activar los periféricos conectados a la computadora.

Bit dígito binario. Unidad mas pequeña de información en una computadora, un 1 o un 0. Puede definir dos condiciones On o Off.

Bit rate régimen de velocidad en bits, tasa de bits, la velocidad de transmisión de un dato codificado en binario.

Bordes Contorno de un objeto dentro de una pintura.

Brillantez brillo, nivel de luz en una pantalla de visualización.

Buffer Porción reservada de memoria que se utiliza para almacenar datos mientras son procesados. Un buffer puede ser también un pequeño banco de memoria usado para fines especiales.

Bus de direcciones Canal interno desde el procesador a la memoria a través del cual se transmiten las direcciones de los datos (no los datos).

Byte conjunto de 8 bits.

CCD charge couple device dispositivo acopiado por carga.

Codificar representación óptima de la información con fines específicos, almacenamiento o transmisión.

Codificación Aritmética Método de compresión de datos estadístico que convierten cadenas de datos en simples números.

CGA Color Graphics Adapter, estandar de presentación de video de IBM que provee texto y graficos de baja resolución.

Compresión reducción del número de datos para representar la misma información.

Compresión adaptiva Técnica de compresión de datos que se ajusta dinámicamente al algoritmo utilizado, basandose en el contenido de los datos que debe comprimir.

Compression Ratio razón de compresión. Medida de los datos comprimidos.

Contraste Diferencia entre áreas mas brillantes y mas oscuras en una pantalla de visualización.

Controlador rutina de programa que contiene las instrucciones necesarias para controlar la operación de los dispositivos periféricos.

cpi caracteres por pulgada, mide la densidad de caracteres por pulgada en cinta o en papel.

Cuantización digitalización de la imagen en amplitud.

Digitalizador Dispositivo que transforma una señal analógica a una señal digital.

dpi dots por inch puntos por pulgada

Dither Ordenado Técnica Halftone que produce imágenes binivel mediante un arreglo ordenado de valores fijos.

Dithering : Halftoning en una computadora es llamado Digital Halftoning. Dither es una abreviación o acrónimo de Halftoning , por lo que el termino dithering ha sido mas ampliamente conocido. Ambos terminos se refieren a una colección de algoritmos de computadora que generan tonos de gris o color.

Difusión de error tecnica que propaga las diferencias entre valores reales y el valor medio (umbral) dentro de un vecindario de pixeles.

Display periférico de salida, dispositivo a través del cual se presenta la información visualmente.

Driver controlador.

EGA Enhanced Graphics Adapter adaptador de gráficos mejorado. Estandar de presentación de video que provee textos y graficos de media resolución, requiere de un monitor digital.

Ensamblador Software que traduce el lenguaje ensamblador a lenguaje maquina.

Error Diferencia entre el valor real y el estimado.

error código de error para la detección de graficos usados en turbo c.

Escala de grises : es una arreglo de diferentes tonos de gris , los cuales varían entre blanco y negro.

Estándares Referencias establecidas que cumplen varias normas de eficiencia y calidad.

Filtro Pasa Banda Dispositivo electrónico que prohíbe que pasen a través de él todas las frecuencias que no se encuentren en el rango especificado.

Filtros Selectores de frecuencia, únicamente permiten un rango específico de frecuencias.

Halftoning : es el proceso de combinar dos o mas colores para producir un tono particular de gris o color. Para asida a color , sistemas mas modernos utilizan los colores substractivos : cyan, magenta y amarillo así como el negro.

Hardware componentes físicos de una arquitectura de computadoras.

HGC Hercules Graphics Controller ver Adaptador Gráfico

Imagen Digital función de intensidad de luz bidimensional denotado como $f(x,y)$ donde

la amplitud de f en coordenadas espaciales (x,y) indica la intensidad de la imagen en ese punto.

Impresora periférico de salida que se encarga de la presentación de la información a través de papel.

Interfases Gráficas Medio ambiente gráfico, con el cual el usuario tiene la facilidad de observar su información a través de ventanas.

Interrupcion Petición de servicio.

JPEG

Lenguaje ensamblador lenguaje de programación que esta a un paso del lenguaje maquina.

Lossless técnicas de compresión sin pérdidas, es decir, no admiten degradación de la imagen.

Lossy técnicas de compresión las cuales admiten alguna modificación en las imágenes.

Mapear transformar, cambiar de dominio.

Matriz vesse Arreglo.

Mascara Pantalla, arreglo ordenado donde sus elementos representan un peso fijo para una operación específica.

Mapas de Memoria tipo de almacenamiento de datos en la computadora. Multibit y multipiano por pixel. Memoria de video segmento de memoria reservado para la información que se despliega en pantalla.

Modos Gráficos Selector que determina la máxima resolución del video y el número de posibles colores reproducibles.

Modulación Delta Una técnica que se emplea para sacar muestras de ondas de voz y convertirlas en código digital. La modulación delta prueba la onda 32000 veces por segundo pero genera solamente un bit por muestra.

Monitor Dispositivo que despliega la información contenida en la memoria de video.

Monitor Digital Monitor de video que acepta una señal digital de la computadora y la convierte en analógica para iluminar la pantalla, utilizado por controladores CGA y EGA.

Monitor analógico monitor de video que acepta una señal analógica de la computadora . (la conversión digital-analógica se realiza en el controlador de video) . Los monitores analógicos se

Glosario

diseñan para que acepten una pequeña gama de resoluciones de visualización, por ejemplo : VGA, CGA o SVGA, o monitores Multisync que pueden aceptar una amplia gama de resoluciones incluyendo TV.

Monocromático dispositivo que únicamente puede reproducir dos tonos.

Muestreo digitalización de la imagen en coordenadas espaciales.

Multibit estrategia de almacenamiento de memoria para dispositivos monocromáticos.

Multinivel múltiples niveles, es decir, varios tonos de gris o color.

Multipiano varios planos, conjunto de planos que en conjunto, determinan al tono o color de un pixel.

Niveles de gris secuencia de tonos de gris que van de blanco a negro y cualquier tono intermedio.

Operación de Punto procesamiento donde la salida depende de la entrada actual.

Operaciones de vecindad procesamiento que se realiza sobre un conjunto de píxeles.

Paterning técnica que produce imágenes binarias mediante la comparación de los elementos de la pintura y un valor previamente fijado.

Patrón : Es una colección de puntos ordenados que representan un tono de gris.

Peso elemento de una máscara o pantalla, su valor y ubicación dependen del criterio de análisis.

Periférico dispositivo de salida que se encuentra conectado a la computadora.

Pixel elemento básico de una imagen digital.

Preprocesador dedicado procesador que se encuentra efectuando operaciones específicas para una determinada tarea.

Protocolo de comunicaciones estándar de software o hardware para transmitir datos.

Razón de Refresco número de veces que la pantalla completa es examinada por las pistolas de electrones del tubo de rayos catódico.

Registros elementos internos de la computadora los cuales almacenan la información que es procesada.

Resolución número de puntos por pulgada.

Scanners dispositivos que digitalizan documentos.

Software conjunto de programas.

TCD Transformada coseno discreta.

Algoritmo similar a la Transformada rápida

de Fourier que convierte datos (elementos digitales, formas de onda, etc) en grupos de frecuencia. Las primeras frecuencias son las más significativas, las últimas en cambio, las menos significativas.

Thresholding Método para binarizar imágenes comparando simplemente con un umbral fijo

Umbral : Se considera como un nivel de comparación, generalmente se fija a la mitad del número de niveles de gris.

Unidimensional una sola dimensión.

Video Periférico de salida para visualizar información (imágenes)

Voltaje Diferencia de potencial

VGA (video graphics array) arreglo gráfico de video.

Windows interface gráfica mediante ventanas utilizado en computadoras personales

Xwindows interface gráfica para sistemas Unix.

Bibliografía

- [CAS89] Kenneth R. Castleman "Digital Image Processing" Prentice Hall
Signals Processing Series 1989
- [COS94] Pamela C. Cosman, Robert M. Gray, Eva A Riskin "Using Vector
Quantization for Image Processing " Society of information Display
International Symposium Digest of Technical Papers 1994
- [EPS93] Epson Corporation "FX-80 OPERATION MANUAL EPSON"
Nagano Japan 1993
- [GLA90] Andrew S. Glassner "Graphics Gems" Xerox Palo Alto Research
Center Academic Press Inc. 1990
- [GON92] Gonzalez / Woods "Digital Image Processing" University of
Tennessee. Addison Wesley Publishing Company 1992
- [KIR93] Tomas Kirkhorn "Continuous Ink Jet Printing of Medical Images"
Department of Electrical Measurements. Sweden 1993
- [KLI91] Bradley dyck Kilanver "EGA/VGA a Programmer's Reference
Guide" Computer Professional. Mc Graw Hill 1991
- [LIN91] Craig A. Lindley "Practical Image Processing in C" John Wiley &
Sons Inc. 1991
- [LUS93] Marv Luse "Bitmapped Graphics Programming in C++" Addison
Wesley Publishing Company 1993

- [MIN93] Ming Yuan Ting and Eve A. Riskin "Error Diffused Image Compression Using a Binary to Grayscale Decoder and Predictive Pruned Tree Structured Vector Quantization" Department of Electrical Engineering, Society of Information Display International Symposium Digest of Technical Papers 1993
- [MIT91] William J Mitchell "The Reconfigured Eye" The MIT Press Cambridge Massachusetts, London England 1991
- [MUR89] William H Murray , Chris H. Pappas. "80386-80286 Programación en Lenguaje Ensamblador". Osborne McGraw Hill
- [ORO90] Juan Carlos Oros "Impresoras Matriciales, Chorro de Tinta y Laser" España 1990
- [OVI96] Iram Oviedo Chávez "Análisis e Implementación de un Sistema de Compresión para imágenes binarias " Universidad Nacional Autónoma de México Facultad de Ingeniería México Julio 1996
- [PRA91] William K. Pratt "Digital Image Processing" Sun Microsystems Inc. California John Wiley & Sons Inc 1991
- [RAB91] Majid Rabbani and Paul W. Jones "Digital Image Compression Techniques". Tutorials Text in Optical Engineering 1991
- [ROS92] Azriel Rosenfeld and Avinash C. Kak Digital Picture Processing Academic Press Inc 1992
- [SCH87] Herbert Schildt "C The Complete Reference" Osborne Mc. Graw Hill 1987
- [SCH89] Robert J. Schalkof "Digital Image Processing and Computer Vision". John Wiley & Sons Inc. 1989
- [SCH91] W.F. Schreiber "Fundamentals of Electronic Imaging Systems" Spring Verlag 1991
-

Bibliografía

- [SOR96] Oscar E. Soriano "Biblioteca de Programas en C para crear imágenes Halftone" Departamento de Eléctrica, División de Estudios de Posgrado 1996
- [TAK93] Takeshi Masui, Yasumico Nasu "Data Compression Methods for Document Image Including Screened Halftones" Department of Electrical Engineering. Society of Information Display International Symposium Digest of Technical Papers 1993
- [ULI87] Robert Ulichney. "Digital Halftoning" The MIT Press Cambridge, Massachusetts 1987
- [VAN92] R. Vander Kam, P. A. Chou, E. A. Riskin "An algorithm for joint Vector Quantizer and Halftoned Desing" in Procaedings of ICASSP 1992
- [WEB95] Jon A. Webb ACCURATE PARALLEL DIGITAL HALFTONING Department of Electrical Engineering 1995
-