INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO
DEPARTAMENTO DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
ACADEMIA DE INGENIERÍA DE SOFTWARE

Profesora: M. en C. Ma. Elena Cruz Meza,
e-mail: mcruzm@ipn.mx,

ANÁLISIS DE IMÁGENES

Análisis de Imágenes

Unidad III

Análisis en el dominio de la frecuencia

Unidad III

Contenido

LA TRANSFORMADA DE FOURIER

Jean Baptiste Joseph Fourier
(1768 - 1830)

La Théorie Analitique de la Chaleur (1822)
(La teoría analítica del calor)

3.1 La Transformada discreta de Fourier

- Fourier desarrollo una representación de funciones basada en la frecuencia, que ha tenido una gran importancia en numerosos campos de matemáticas y ciencia:
- *"El teorema de Fourier no solamente es uno de los resultados más hermosos del análisis moderno, sino que además se puede decir que proporciona una herramienta indispensable en el tratamiento de casi todos los enigmas de la física moderna"* [Lord Kelvin]

3.1 La Transformada discreta de Fourier

- *Toda señal periódica, sin importar cuan complicada parezca, puede ser reconstruida a partir de sinusoides cuyas frecuencias son múltiplos enteros de una frecuencia fundamental, eligiendo las amplitudes y fases adecuadas.*

3.1 La Transformada discreta de Fourier

- Fourier propone que mediante la suma de señales co/sinusoidales de diferentes amplitudes, frecuencias y fases, es posible construir casi cualquier función arbitraria. Dentro de este conjunto de señales puede existir una con frecuencia cero, que es un término constante, a menudo referido como la componente continua (DC), debido al hecho de que cierta terminología en este área está derivada del procesado de señal y electrónica.

3.1 La Transformada discreta de Fourier

- ***La variación de la brillantez de una imagen, medida a lo largo de una dirección cualquiera es entonces una función que se puede representar mediante el teorema de Fourier, con una suma de distribuciones senoidales de varias frecuencias.***
- Sin entrar en detalles técnicos innecesarios, atenuar o reforzar individualmente algunas de estas componentes senoidales puede tener un efecto dramático en la calidad de una imagen, mejorándola o empeorándola, según el caso.

3.1.1 La TF de funciones contínuas

- Recordando los métodos ya estudiados en el dominio de la frecuencia:
  - Métodos que se basan en el teorema de convolución, el cual cumple con lo siguiente:

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

donde *G, H* y *F* son las transformadas de Fourier de *g, h, y f.*

f(x,y) ⇒ [*h(x,y)*] ⇒ *g(x,y)* — a) La salida del sistema es la convolución de h(x,y) con la señal de entrada.

F(u,v) ⇒ [*H(x,y)*] ⇒ *G(u,v)* — b) La salida es el producto de H(u.v) por la señal de entrada.

3.1.1 La TF de funciones contínuas

- Podemos decir que todos los dominios transformados, que se utilizan dentro del tratamiento digital de imagen, tienen la misma forma básica que puede expresarse como:

$$T(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y)\, b(x,y,u,v)$$

- donde *T es la imagen transformada, I la imagende entrada de tamañoMxN, y b es la función base de* la transformación.

3.1.1 La TF de funciones contínuas

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft}\, dt$$

t: Tiempo
f: Frecuencia en Hz
x(t): Señal de prueba
$e^{-j2\pi t}$: Fasor de Sondeo (Kernel Function)
X(f): Espectro en función de la frecuencia f

- x(t) ↔ X(f), es decir para una función x(t) existe un equivalente X(f).
- X(f), el espectro, revela la fuerza (energía) de varias componentes de frecuencia, ordenadas por frecuencia.
- La transformada de Fourier actúa como un detector de energía en frecuencia-dependiente

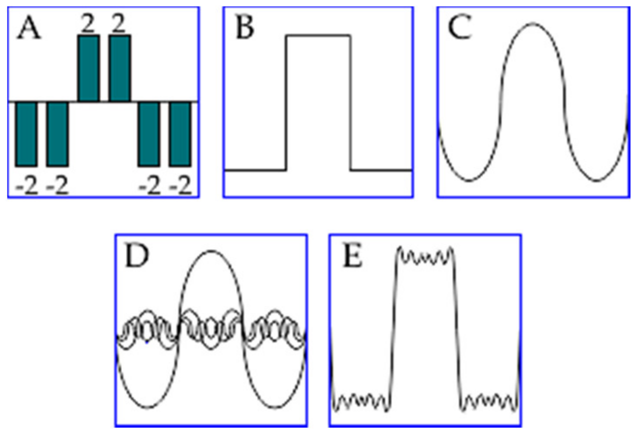
3.1.1 La TF de funciones contínuas

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-2\pi iux} dx$$

Observemos que la transformada de una función real es una función compleja. Es decir, *F(u)=R(u)+I(u)i,* donde *R(u)* e *I(u)* son la parte real e imaginaria de *F(u),* respectivamente.

- La variable u recibe el nombre de ***variable de frecuencia.***
- El módulo de *F(u),* $|F(u)|= (R(u)^2+ I(u)^2)^{1/2}$ recibe el nombre del ***espectro de Fourier.***
- El cuadrado del espectro se denomina ***espectro de potencias*** ó ***densidad espectral*** de *f(x).*
- Su ángulo *P(u)=arctg(I(u)/R(u))* recibe el nombre de ***fase.***

3.1.1 La TF de funciones contínuas



La oscilación sobre un valor medio **(A)** puede representarse por una forma lineal **(B)** y ésta puede reproducirse como una suma de ondas. La onda **C** describe la forma **B** mucho peor que las cinco ondas del gráfico **D** que vemos sumadas en **E**.

3.1.2 La TF de funciones discretas

El equivalente en tiempo y frecuencia discreta es la Transformada Discreta de Fourier

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n] e^{-\frac{2\pi j}{N}kn}}$$

N: Número de Samplers en x[n]
x[n]: Señal de prueba discreta (con índice n)
X[k]: Espectro en función de la frecuencia discreta (con índice k)
$e^{-jkln/N}$: Fasor de Sondeo discreto (Kernel Function)

3.1.2 La TF de funciones discretas

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{-j2\pi ux/M}$$

$$F_k = \frac{1}{N}\sum_{n=0}^{N-1} f_n W_N^{-nk}$$
$$\text{para} \quad k = 0,1,\ldots,N-1$$

$$f_n = \sum_{k=0}^{N-1} F_k W_N^{nk}$$
$$\text{para} \quad n = 0,1,\ldots,N-1$$

Donde: $W_N = e^{j2\pi/N}$

3.1.2 La TF de funciones discretas

Complejidad de la TDF

- El cálculo de la transformada discreta de Fourier involucra dos pasos:

$$F_k = \frac{1}{N}\sum_{n=0}^{N-1} f_n W_N^{-nk}$$
$$\text{para} \quad k = 0,1,\ldots,N-1$$

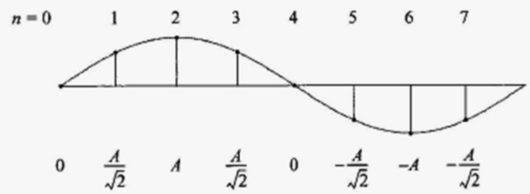
1. Cálculo de WN–nk , para n, k = 0, 1, ... , N-1
   Complejidad = O(N2)

2. Cálculo de Fk = suma de N números, k = 0, 1, ..., N-1
   Complejidad = O(N2)

3.1.2 La TF de funciones discretas

Ejemplos:

$$x(n) = A\sin\left(f2\pi\frac{n}{N}\right)$$



Para simplificar este análisis, se descompone la Transformada en:

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n]\cdot\cos\left(\frac{2\pi}{N}kn\right)} \quad DCT$$

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n]\cdot -j\sin\left(\frac{2\pi}{N}kn\right)} \quad DST$$

3.1.2 La TF de funciones discretas

Ejemplos:

Para simplificar más operemos en una DST real positiva:

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n]\sin\left(\frac{2\pi}{N}kn\right)}$$

Para N = 8, f = 1 y k = 1, es decir ambas señal de sondeo y de prueba son iguales:

Sample	0	1	2	3	4	5	6	7
Input	0	$\frac{A}{\sqrt{2}}$	A	$\frac{A}{\sqrt{2}}$	0	$-\frac{A}{\sqrt{2}}$	$-A$	$-\frac{A}{\sqrt{2}}$
Sondeo	0	$\frac{1}{\sqrt{2}}$	1	$\frac{1}{\sqrt{2}}$	0	$-\frac{1}{\sqrt{2}}$	-1	$-\frac{1}{\sqrt{2}}$
Producto	0	$\frac{A}{\sqrt{2}}$	A	$\frac{A}{\sqrt{2}}$	0	$\frac{A}{\sqrt{2}}$	0	$\frac{A}{\sqrt{2}}$

3.1.2 La TF de funciones discretas

Ejemplos:

Para $x(n) = A \sin f\, 2\pi (n/N)$ $con\ k = f,$

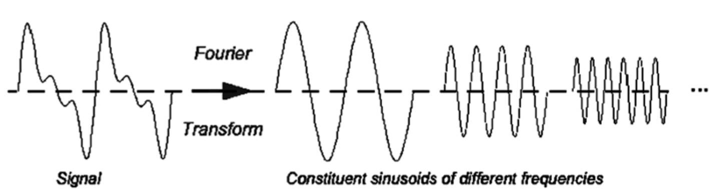
$$\sum_{n=0}^{N-1} x[n] \sin\left(\frac{2\pi}{N}kn\right) = 0 + \frac{A}{2} + A + \frac{A}{2} + 0 + \frac{A}{2} + A + \frac{A}{2}$$
$$= 4A$$

Para un N general, se obtiene:

$$\frac{1}{N}\sum_{n=0}^{N-1} x[n] \sin\left(\frac{2\pi j}{N}kn\right) = \frac{A}{2}$$

3.1.2 La TF de funciones discretas

- En el caso de las imágenes, las "señales" corresponden a los niveles de gris o intensidad de las diferentes filas o columnas de la matriz de la imagen, donde el eje del tiempo se reemplaza por los ejes x,y.



3.1.2 La TF de funciones discretas

Sea f(x,y) una imagen en niveles de grises, tal que *x=0,1,...,N-1 e y=0,1,…,N-1;* y *f(x,y)* toma valores discretos representando el nivel de gris del píxel *(x,y)* entonces, la transformada discreta de Fourier de la imagen consiste en una función *F(u,v)* tal que *u=0,1,...,N-1* y *v=0,1,...,N-1:*

$$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\, e^{-2\pi i(ux+vy)/N}$$

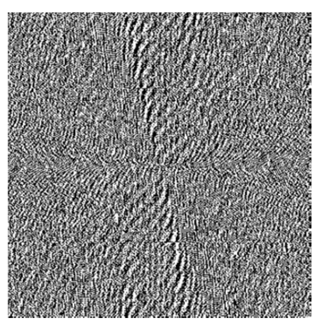
y su inversa como

$$f(x,y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} F(u,v)\, e^{2\pi i(ux+vy)/N}$$

3.1.2 La TF de funciones discretas

Ejemplo:

Consideremos una imagen de tamaño 100x100 en la que en una esquina tiene impresa tres objetos en blanco (las letras abc), la **TDF** en 2D de tal imagen nos da una matriz de números complejos. Considerando el argumento (ángulo) de cada una de las entradas de esa matriz, que es un número entre -180º y 180º. Si ese número es negativo, ponemos un cero (o el color negro), y si el número es positivo, ponemos un uno (o el color blanco).



3.1.2 La TF de funciones discretas

Diferentes frecuencias de pixeles en una imagen:

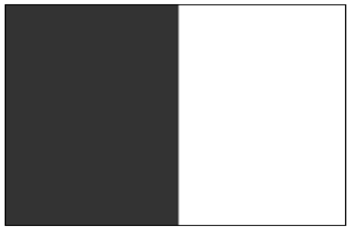


Imagen con altas frecuencias

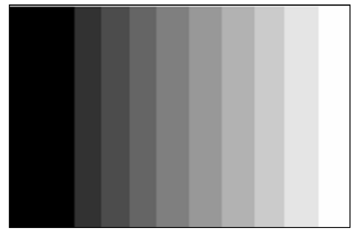


Imagen con baja frecuencia

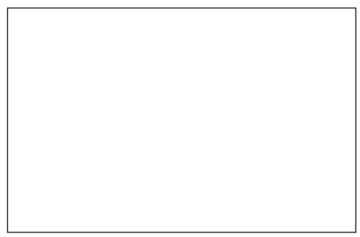


Imagen con frecuencia nula

3.1.2 La TF de funciones discretas

Las propiedades de la transformada de Fourier discreta bidimensional (TFD):

Núcleo separable y simétrico

La ventaja que aporta esta propiedad es el hecho de poder obtener la transformada F(x,y) (o la inversa f(x,y)) en dos pasos, mediante la aplicación de la Transformada de Fourier 1-D (o su inversa):

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} F(x,v) e^{-2\pi i u x/N}$$

Donde:

$$F(x,v) = \sum_{y=0}^{N-1} f(x,y) e^{-2\pi i v y/N}$$

En particular, esto significa que la matriz de la transformada se puede obtener mediante un producto de matrices $T = A^T FA$

3.1.2 La TF de funciones discretas

La linealidad
La transformada de Fourier y su inversa son transformaciones lineales, es decir, poseen la propiedad distributiva respecto de la suma.

La traslación
• $TF[f(x,y)\ e^{i2Pi(Ux+Vy)/N}]=F(u-U,\ v-V)$ *(se traslada el origen de la transformada a (U, V))*

• $TF[f(x-X,\ y-Y)]=F(u,\ v)\ e^{-i2Pi(uX+vY)/N}$

Un caso particular de esta propiedad consiste en mover el origen de la transformada de Fourier de *f(x,y)* al centro de la matriz *N X N* que le corresponda, es decir al punto *(N/2,N/2)*. Para ello, podemos hacer uso de que: $TF[f(x,y)(-1)^{x+y}]$ se hace corresponder con $F(u-N/2,v-N/2)$.

También cabe resaltar, que un desplazamiento en la función f(x,y), no provocará un cambio en la magnitud de su transformada de Fourier. Véase esto matemáticamente en la siguiente expresión:

$$|F(u,v)e^{-i2Pi(uX+vY)/N}|=|F(u,v)|$$

3.1.2 La TF de funciones discretas

La simetría y periodicidad
Si f(x,y) es real, la transformada de Fourier satisface:

$$|F(u,v)|=|F(-u,\ -v)|$$

La transformada discreta de Fourier y su inversa son funciones periódicas de periodo N; es decir,

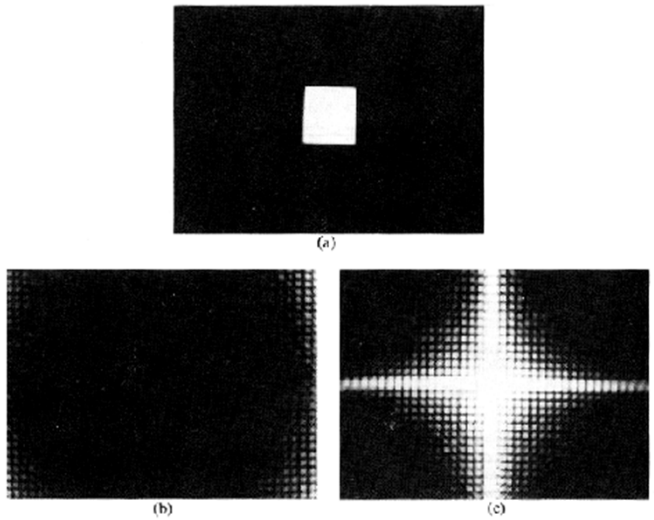
$$F(u,v)=F(u+N,\ v)=\ F(u,\ v+N)=F(u+N,\ v+N).$$

Consecuencia:

Si se desplaza el origen de la transformada al punto (N/2, N/2), para calcular la transformada de Fourier , F(u-N/2, v-N/2), en un periodo completo sólo necesitamos calcularla en los N/2 + 1 puntos primeros.

3.1.2 La TF de funciones discretas

Ej. La simetría y periodicidad



3.1.2 La TF de funciones discretas

Valor promedio
Una definición ampliamente utilizada del valor promedio de una función discreta de dos dimensiones es:

$$\bar{f}(x,y) = \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

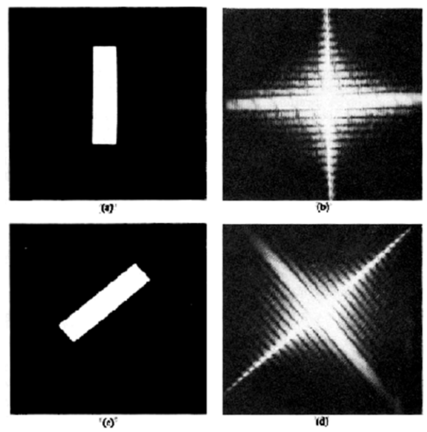
Propiedad:

$$\bar{f}(x,y) = \frac{1}{N}F(0,0)$$

3.1.2 La TF de funciones discretas

La rotación

Si rotamos la función f(x,y) un ángulo determinado, la transformada de Fourier también será afectada por una rotación del mismo ángulo. Esta propiedad también se da a la inversa, es decir, si la transformada se rota en un determinado ángulo, la transformada inversa también se verá rotada ese mismo ángulo.



3.1.3 La Transformada Rápida de Fourier

- *La transformada rápida de Fourier* sigue la estrategia de: *divide y vencerás*!!
- La idea:

Por: J.W. Cooley y J.W. Tokey, 1965

$$F_k = \frac{1}{N}\sum_{n=0}^{N-1} f_n W_N^{-nk}$$

$$\frac{1}{N}\sum_{n=0}^{N/2-1} g_n W_{N/2}^{-nk} \qquad \frac{1}{N}\sum_{n=0}^{N/2-1} h_n W_{N/2}^{-nk}$$

$$\updownarrow \qquad\qquad \updownarrow$$

$$TDF(g, N/2) \qquad TDF(h, N/2)$$

http://www.youtube.com/watch?v=a66i5TFr3M8&feature=related

3.1.3 La Transformada Rápida de Fourier

- De manera que se divide la TDF en coeficientes en posiciones *pares* e *impares* como sigue:

$$F_{2k} = \frac{1}{N}\sum_{n=0}^{N/2-1}(f_n + f_{n+N/2})W_{N/2}^{-nk} = TDF(g, N/2)$$

$$F_{2k+1} = \frac{1}{N}\sum_{n=0}^{N/2-1}[(f_n - f_{n+N/2})W^n]W_{N/2}^{-nk} = TDF(h, N/2)$$

3.1.3 La Transformada Rápida de Fourier

- Por lo que calcular la TDF de N coeficientes es igual a calcular 2 TDF de N/2 coeficientes. Se aplica esta idea de manera recursiva y obtenemos la FFT.

- La complejidad de la FFT

O(N log2 N)

3.1.3 La Transformada Rápida de Fourier

- El algoritmo de la FFT es complicado y sus detalles son generalmente dejados para aquellos que se especializan en ella. En esta sección sólo esbozaremos las ideas principales del método.

- El material utilizado para tal fin fué tomado del libro: *The Scientist and Engineer´s Guide to Digital Signal Processing*, cuyo autor es *Steven W. Smith* y pueden encontrarlo en la siguiente página web:

http://www.dspguide.com/pdfbook.htm

3.1.3 La Transformada Rápida de Fourier

Representación de la TDF real y compleja

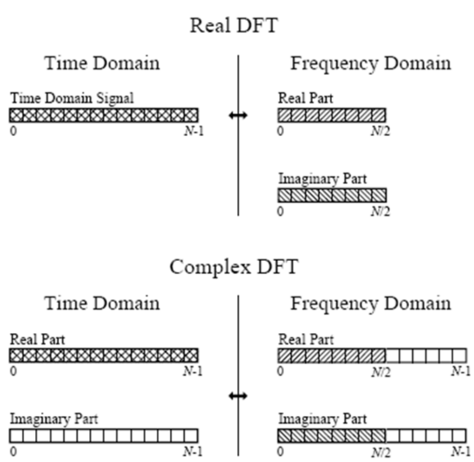


FIGURE 12-1
Comparing the real and complex DFTs. The real DFT takes an N point time domain signal and creates two $N/2+1$ point frequency domain signals. The complex DFT takes two N point time domain signals and creates two N point frequency domain signals. The crosshatched regions shows the values common to the two transforms.

La *TDF real* toma *N* puntos en el dominio del tiempo y crea 2 conjuntos de *N/2+1* puntos en el dominio de la frecuencia.

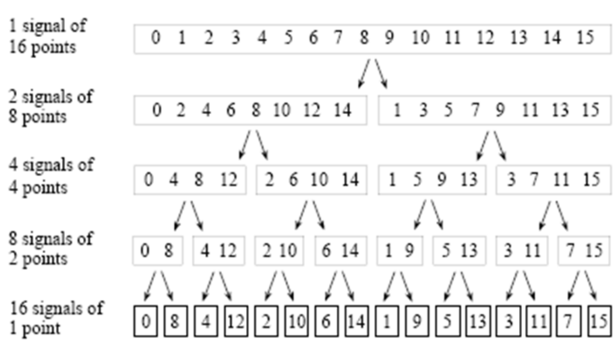
La *TDF compleja* toma N puntos en el dominio del tiempo y crea 2 conjuntos de *N* puntos en el dominio de la frecuencia.

Los cuadros sombreados muestran los valores comúnes entre las dos transformadas.

3.1.3 La Transformada Rápida de Fourier

- El algoritmo de la FFT opera:
  - (1) descomponiendo *una señal* del dominio del tiempo de *tamaño N* puntos en *N señales* del dominio del tiempo cada una compuesta por *un sólo punto*.
  - (2) El segundo paso es calcular los *N espectros de frecuencia* correspondientes a estas *N señales* en el dominio del tiempo.
  - (3) Finalmente, los *N espectros* se sintetizan en *un arreglo de espectros de frecuencia.*

3.1.3 La Transformada Rápida de Fourier



Descomposición de FFT: Una señal de *N* puntos se descompone en N señales de un sólo punto cada una.

Cada estado utiliza una *descomposición entrelazada*, separando las muestras enumeradas como *pares* e *impares*.

FIGURE 12-2
The FFT decomposition. An *N* point signal is decomposed into *N* signals each containing a single point. Each stage uses an *interlace decomposition*, separating the even and odd numbered samples.

Esta es una señal que tiene inicialmente 16 puntos y es descompuesta en 16 señales de un sólo punto cada una.

3.1.3 La Transformada Rápida de Fourier

Descomposición entrelazada

- La *descomposición entrelazada* se utiliza cada vez que la señal se divide en dos, esto es, la señal se separa en sus muestras numeradas como *pares* e *impares*.

- Se requieren $log_2\ N$ estados para esta descomposición, por ejemplo: una señal de 16 puntos (2^4) requiere de 4 estados, una señal de 512 puntos (2^7) requiere de 7 estados, una señal de 4096 (2^{12}) requiere de 12 estados, etc

Nota: Recuerda el valor de $log_2\ N$, será mencionado más adelante

3.1.3 La Transformada Rápida de Fourier

Reordenamiento de las muestras

- La descomposición no es más que un *reordenmiento* de las muestras.

Sample numbers in normal order		Sample numbers after bit reversal	
Decimal	*Binary*	*Decimal*	*Binary*
0	0000	0	0000
1	0001	8	1000
2	0010	4	0100
3	0011	12	1100
4	0100	2	0010
5	0101	10	1010
6	0110	6	0100
7	0111	14	1110
8	1000	1	0001
9	1001	9	1001
10	1010	5	0101
11	1011	13	1101
12	1100	3	0011
13	1101	11	1011
14	1110	7	0111
15	1111	15	1111

FIGURE 12-3
The FFT bit reversal sorting. The FFT time domain decomposition can be implemented by sorting the samples according to bit reversed order.

A la izquierda se ve una lista de valores decimales con sus equivalentes valores binarios.

A la derecha las muestras se encuentran reordenadas también con sus equivalentes binarios.

La idea importante aquí es que el número binario son los reversos de cada uno. La muestra 3 (0011) se cambia por 12 (1100), la muestra 14 (1110) se cambia por 7 (0111).

A esto se le llama *ordenamiento reverso de bit* (*bit reversal sorting*). Reordena las *N* muestras del dominio del tiempo, *invirtiendo los bits* de izquierda a derecha.

3.1.3 La Transformada Rápida de Fourier

- El siguiente paso en el algoritmo de la TRF es *encontrar el espectro de frecuencia* de las señales de tiempo de un punto. El espectro de frecuencia de una señal de un punto es *igual a si misma!* Esto significa que no se requiere hacer nada en este paso.

- Aunque no hay ningún trabajo en este paso, recordemos que ahora *cada punto* es *un espectro de frecuencia* y no una señal del tiempo.

3.1.3 La Transformada Rápida de Fourier

- El último paso en la TRF es *combinar* los *N* espectros de frecuencia en el *orden inverso* en que se llevó la descomposición en el dominio del tiempo. (Aquí es donde el *algoritmo se vuelve complicado*!)

- Desafortunadamente no se puede regresar con la misma rapidéz y hay que pasar por un estado cada vez. En el primer estado 16 espectros de frecuencia (1 punto c/u) se sintetizan en 8 espectros de frecuencia (2 puntos c/u). En el segundo estado 8 espectros de frecuencia (2 puntos c/u) se sintetizan en 4 espectros de frecuencia (4 puntos c/u), etc. El último estado resulta el espectro de frecuencia de 16 puntos esperado como salida de la TRF.

3.1.3 La Transformada Rápida de Fourier

Síntesis de la TRF

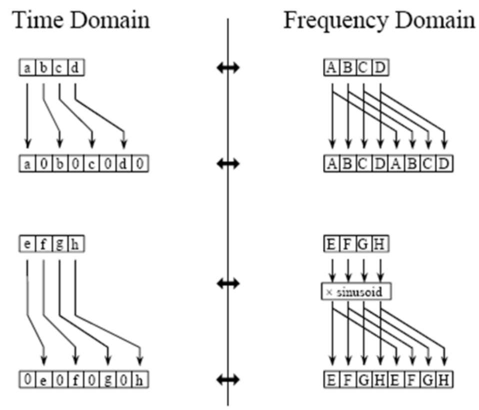


FIGURE 12-4
The FFT synthesis. When a time domain signal is diluted with zeros, the frequency domain is duplicated. If the time domain signal is also shifted by one sample during the dilution, the spectrum will additionally be multiplied by a sinusoid.

Dos espectros de frecuencia de 4 puntos c/u se *combinan* en un sólo espectro de frecuencia de 8 puntos.

Diluir (mezclar) los puntos en el dominio del tiempo *con ceros* corresponde a una *duplicación* en el dominio de la frecuencia.

El espectro de frecuencia se combina en la TRF duplicándolos, y luego sumando los espectros duplicados.

3.1.3 La Transformada Rápida de Fourier

Síntesis de la TRF

- De manera que correspondan a la hora de unirse, las dos señales de tiempo se mezclan de manera algo diferente. A una señal se le ponen en *cero las posiciones pares*, mientras que a la otra se le ponen en *cero las posiciones impares*. En otras palabras, una de las señales en el dominio del tiempo se *recorre a la derecha una muestra*.

- El *corrimiento* en el dominio del tiempo corresponde a la *multiplicación* del espectro de frecuencia por una *senoidal.*

3.1.3 La Transformada Rápida de Fourier

Síntesis de la TRF

FIGURE 12-5
FFT synthesis flow diagram. This shows the method of combining two 4 point frequency spectra into a single 8 point frequency spectrum. The ×S operation means that the signal is multiplied by a sinusoid with an appropriately selected frequency.

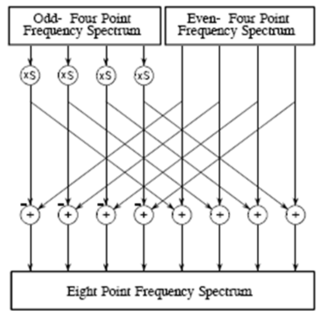
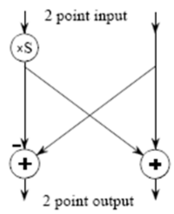


Diagrama de la unión de dos espectros de 4 puntos c/u en un espectro de 8 puntos.

FIGURE 12-6
The FFT butterfly. This is the basic calculation element in the FFT, taking two complex points and converting them into two other complex points.



Elemento de cálculo básico para unir 2 números complejos en otros 2 números complejos que se repiten una y otra vez durante esta parte del algoritmo. Llamado “mariposa”.

3.1.3 La Transformada Rápida de Fourier

$$X(n) = \sum_{k=0}^{N-1} x_0(k)\cdot \exp\left(\frac{-j\cdot 2\cdot\pi\cdot n\cdot k}{N}\right) \quad \text{con} \quad n = 0 .. N-1$$

Sea: $W = \exp\left(\frac{-j\cdot 2\cdot \pi}{N}\right)$

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} W^0 & W^0 & W^0 & W^0 \\ W^0 & W^1 & W^2 & W^3 \\ W^0 & W^2 & W^4 & W^6 \\ W^0 & W^3 & W^6 & W^9 \end{bmatrix} \cdot \begin{bmatrix} x_0(0) \\ x_0(1) \\ x_0(2) \\ x_0(3) \end{bmatrix}$$

Transformada rápida:

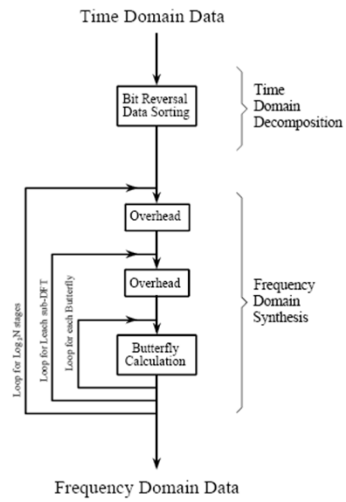
$$\begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & W^0 & 0 \\ 0 & 1 & 0 & W^0 \\ 1 & 0 & W^2 & 0 \\ 0 & 1 & 0 & W^2 \end{bmatrix} \cdot \begin{bmatrix} x_0(0) \\ x_0(1) \\ x_0(2) \\ x_0(3) \end{bmatrix}$$

$$\begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix} = \begin{bmatrix} x_2(0) \\ x_2(1) \\ x_2(2) \\ x_2(3) \end{bmatrix} = \begin{bmatrix} 1 & W^0 & 0 & 0 \\ 1 & W^2 & 0 & 0 \\ 0 & 0 & 1 & W^1 \\ 0 & 0 & 1 & W^3 \end{bmatrix} \cdot \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix}$$

3.1.3 La Transformada Rápida de Fourier

Diagrama de flujo de la TRF

FIGURE 12-7
Flow diagram of the FFT. This is based on three steps: (1) decompose an N point time domain signal into N signals each containing a single point, (2) find the spectrum of each of the N point signals (nothing required), and (3) synthesize the N frequency spectra into a single frequency spectrum.

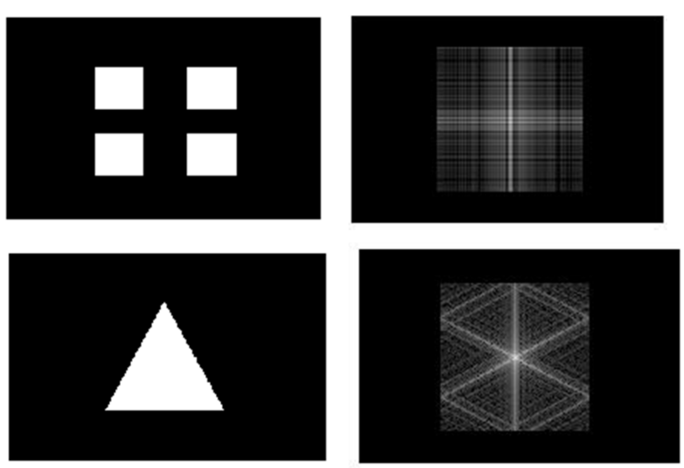


La descomposición del dominio del tiempo se realiza por ordenamiento reverso de bits. Transformar los datos descompuestos a frecuencia no involucra ninguna operación, asi que no aparece en el digrama.

La síntesis requiere de tres ciclos: (1) externo: $log_2\ N$ estados. (2) medio: se mueve en cada espectro de frecuencia individual, (3) interno: utiliza la mariposa para calcular cada punto del espectro de frecuencia.

3.1.3 La Transformada Rápida de Fourier

Algoritmo de la TRF



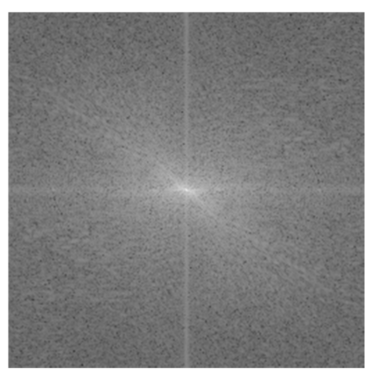
Imágenes con su correspondiente espectro mediante la TRF

3.1.3 La Transformada Rápida de Fourier

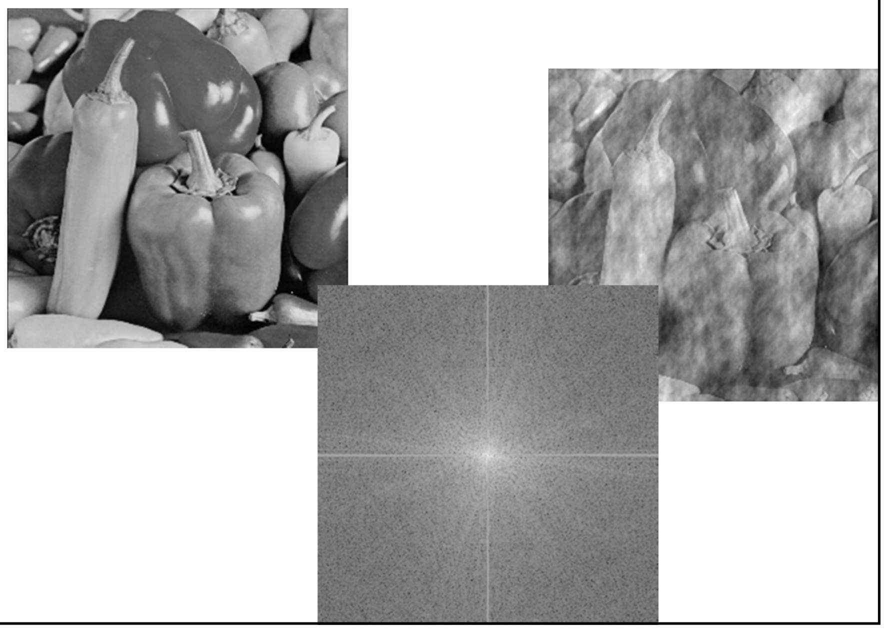
- La descripción anterior de la TRF es para señales en 1D. Para implementarla en 2D, es necesario hacer el cálculo:
  1- Para los renglones de la imagen
  2- Para las columnas,
  (en cada caso se aplica el algoritmo de la TRF en 1D)

- Existen numerosos lugares donde se puede encontrar el código de la TRF (casi en cualquier lenguaje de programación)
  - Ej. En lenguaje C aplicado a imágenes: http://local.wasp.uwa.edu.au/~pbourke/miscellaneous/dft/index.html

3.1.3 La Transformada Rápida de Fourier



3.1.3 La Transformada Rápida de Fourier



3.1.3 La Transformada Rápida de Fourier

Aplicaciones más importantes de la FFT en el tratamiento de imagen:

- ✓Filtros y convolución
- ✓ Análisis espectral en tiempo real - Análisis del Cepstrum
- ✓ Estimación de funciones de transferencia
- ✓ Análisis de sistemas
- ✓ Demoduladores
- ✓ Filtros digitales de alta velocidad
- ✓ Compresión del ancho de banda de vídeo
- ✓ Restauración de imágenes

3.1.4 La Transformada Inversa de Fourier

La **inversa** de la transformada de Fourier se define como:

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{2\pi i u x} dx$$

Análogamente, se define la transformada de Fourier de una función continua e integrable de 2 variables:

$$F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) e^{-2\pi i(ux+vy)} dx\, dy$$

y su inversa como

$$f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) e^{2\pi i(ux+vy} dx$$

3.1.4 La Transformada Inversa de Fourier

La **inversa** de la transformada de Fourier puede implementarse a partir de la TF con mínimas modificaciones de entrada, ya que puede reescribirse como:

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp[-j2\pi ux / N]$$

y:

$$f(u) = \frac{1}{N} \sum_{u=0}^{N-1} F(u) \exp[-j2\pi ux / N]$$

Tomando el complejo conjugado de la primera ec. se observa que la parte inferior de la de derecha de la segunda ec. corresponde a la forma de la TF. Por lo que tomando como entrada $F^*(u)$ en un algoritmo para calcular la TF directa se obtiene la función $f^*(x)/N$. Tomando el complejo conjugado y dividiendo por N se obtiene la función $f(x)$ buscada.

3.1.4 La Transformada Inversa de Fourier

Para matrices cuadradas bidimensionales se toma el complejo conjugado de la ecuación:

$$f(x) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{V}^{N-1} F(u,v) \exp[-j2\pi(ux+vy)/N]$$

Cuando f(x) o f(x,y) son reales la operación complejo conjugado es inn:

$$f^*(x) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{V}^{N-1} F(u,v) \exp[-2j2\pi(ux+vy)/N]$$

3.1.5 Equivalencias entre la convolución y la TF

La Convolución y la Correlación

- Son dos relaciones de la transformada de Fourier que constituyen la unión fundamental entre los dominios espacial y de frecuencias
- Ambas son de importancia fundamental para la comprensión de las técnicas de procesado de imágenes basadas en la transformada de Fourier.

3.1.5 Equivalencias entre la convolución y la TF

La convolución y las operaciones relacionadas se encuentran en muchas aplicaciones de ingeniería y matemáticas

- Estadística: un promedio móvil ponderado es una convolución
- Teoría de la probabilidad: la distribución de probabilidad de la suma de dos variables aleatorias independientes es la convolución de cada una de sus distribuciones de probabilidad
- Óptica: una fotografía desenfocada es la convolución de la imagen correcta con el círculo borroso formado por el diafragma del iris

3.1.5 Equivalencias entre la convolución y la TF

Definición:

✓La convolución de f y g se denota por $f*g$
✓Se define como la integral del producto de ambas funciones después de que sea invertida y desplazada una distancia T
✓La convolucion requiere que una de las dos funciones sea refejada respecto al origen, antes de que la integral sea calculada sobre la variable de desplazamiento

$$\left.(f*g)(t) = \int f(\tau)g(t-\tau)d\tau \right\}$$ Espacio continuo 2D

3.1.5 Equivalencias entre la convolución y la TF

Supongamos que en lugar de ser contínuas, *f(x)* y *g(x)* están discretizadas en matrices de tamaño *A* y B, respectivamente: {*f(0), f(1), f(2),…, f(A-1)},…,{g(B-1)}* .

$$\left. f[m] * g[m] = \sum_{n} f[n]g[m-n] \right\} \text{Espacio discreto 2D}$$

3.2 El uso de la TF en imágenes digitales

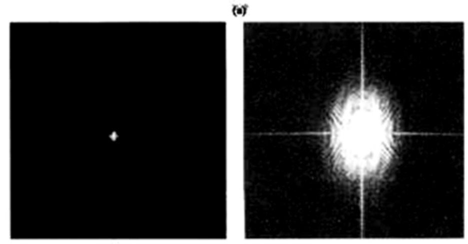
Representación del logaritmo del espectro

El espectro de Fourier suele tener un rango mucho mayor que los usuales para mostrar una imagen. Una técnica usual para evitar esto es considerar el logaritmo del espectro usando la fórmula

D(u,v)=C(log(1+|F(u,v)|))

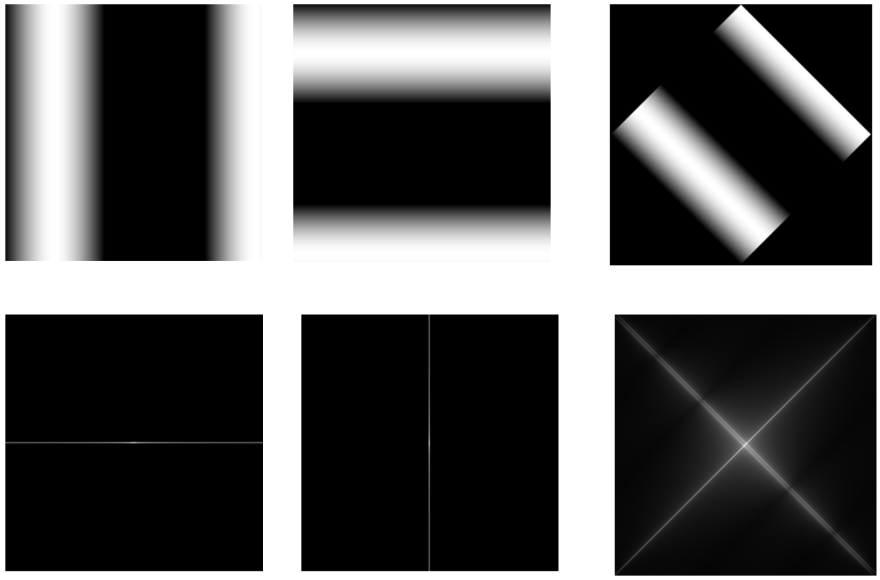
donde C es una constante adecuada de reescalado de la imagen, que se aplica para obtener valores dentro de la paleta de colores disponible.

(a)



(b) (c)

3.2 El uso de la TF en imágenes digitales



3.2 El uso de la TF en imágenes digitales

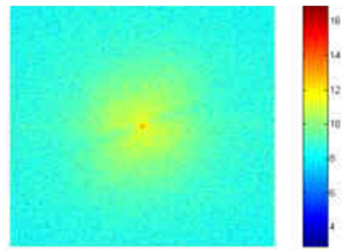
La convolución y las operaciones relacionadas se encuentran en muchas aplicaciones de ingeniería y matemáticas

- Acústica: un eco es la convolución del sonido original con una función que represente los objetos que se reflejen
- Ingeniería eléctrica y otras disciplinas: la salida de un sistema lineal es la convolución de la entrada con la respuesta del sistema a un impulso
- Física: en un sistema lineal con un principio de superposición aparece una operación de convolución

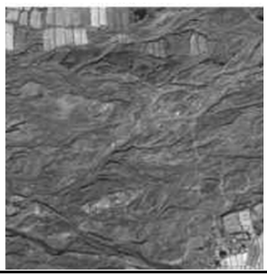
3.2 El uso de la TF en imágenes digitales

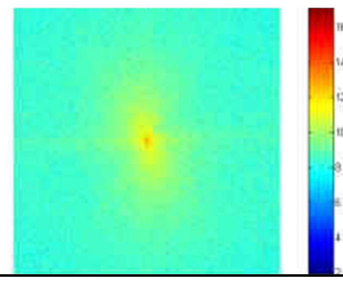
Aplicación del logaritmo del espectro de Fourier: Analizador de texturas:

Textura de bosque



Textura de barro

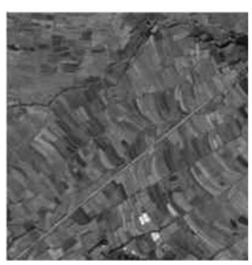


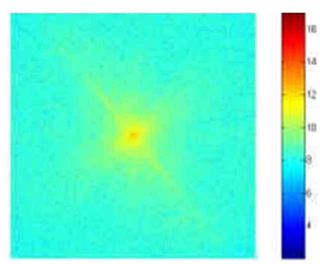


3.2 El uso de la TF en imágenes digitales

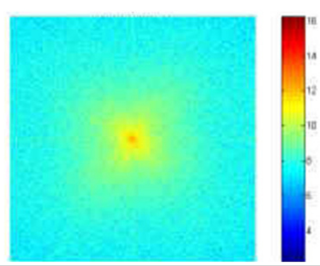
Aplicación del logaritmo del espectro de Fourier: Analizador de texturas:

Texturas de campos





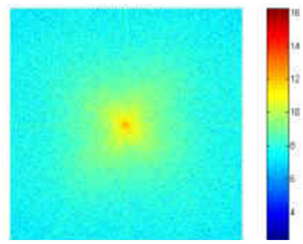
Texturas de charcas



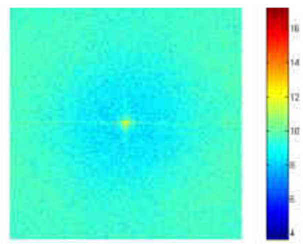
3.2 El uso de la TF en imágenes digitales

Aplicación del logaritmo del espectro de Fourier: Analizador de texturas:

Texturas de ciudad



Texturas de agua



3.2 .1 Ajuste de brillo

3.2.2 Filtros pasa bajas

Filtrado

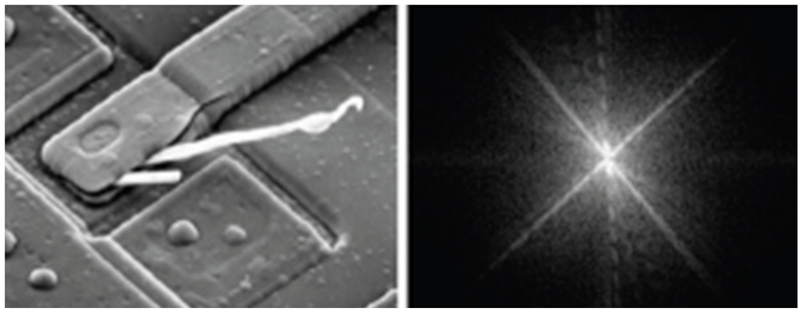


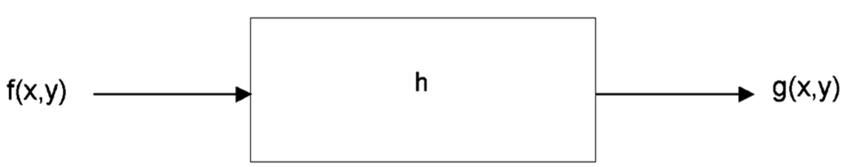
Imagen SEM de un microchip dañado y su espectro de Fourier

3.2.2 Filtros pasa bajas

Filtrado

- Filtrar una imagen consiste en aplicar una transformación de forma que se acentúen o disminuyan ciertos aspectos

$$g(x,y) = T[f(x,y)]$$



3.2.2 Filtros pasa bajas

Tipos de Filtros

- Dominio espacial- convolución.

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

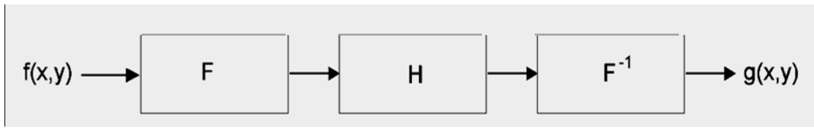
- Dominio de la frecuencia - multiplicación + transformadas de Fourier

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

3.2.2 Filtros pasa bajas

Filtros en frecuencia

- Se realiza una transformación de la imagen al dominio de la frecuencia mediante la transformada de Fourier
- Esto permite que el filtrado sea más sencillo (multiplicación) y pueda ser más *preciso* en frecuencia



3.2.2 Filtros pasa bajas

Transformadas

- Transformado de Fourier

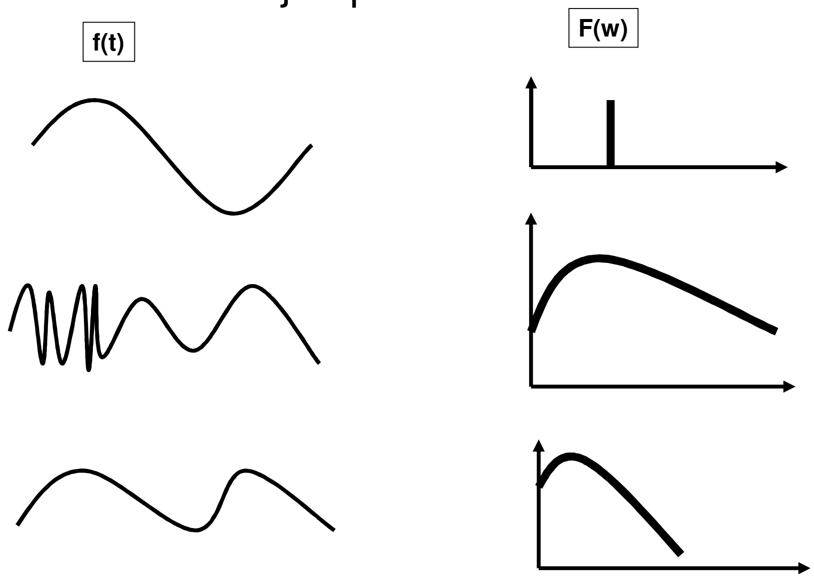
$$F(u) = \int f(x)e^{[-j2\pi ux]}dx$$

- Transformada inversa

$$f(x) = \int F(u)e^{[j2\pi ux]}du$$

3.2.2 Filtros pasa bajas

Ejemplos:



3.2.2 Filtros pasa bajas

Transformadas de 2 variables

- Para el caso de una imagen se requiere aplicar la transformación en 2-D
- Transformado de Fourier

$$F(u) = \iint f(x,y)e^{[-j2\pi(ux+vy)]}dxdy$$

- Transformada inversa

$$f(x) = \iint F(u,v)e^{[j2\pi(ux+vy)]}dudv$$

3.2.2 Filtros pasa bajas

Transformadas discreta

- Para el caso de una imagen digital se aplica la *transformada discreta de Fourier (DFT)*
- Transformado de Fourier

$$F(u) = (1/MN)\Sigma\ \Sigma\ f(x,y)e^{[-j2\pi(ux/M+vy/N)]}$$

- Transformada inversa

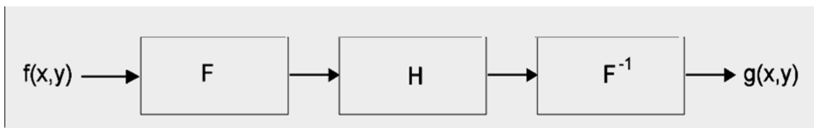
$$f(x) = \Sigma\ \Sigma\ F(u,v)e^{[j2\pi(ux/M+vy/N)]}$$

- Existe una forma eficiente de implementar la DFT llamada *transformada rápida de Fourier* (FFT)

3.2.2 Filtros pasa bajas

Pasos para un Filtrado

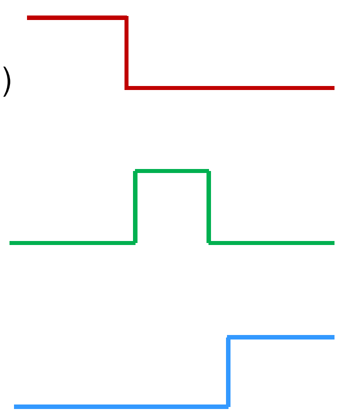
- Se aplica la Transformada de Fourier
- Se aplica el filtro
- Se aplica la transformada inversa



3.2.2 Filtros pasa bajas

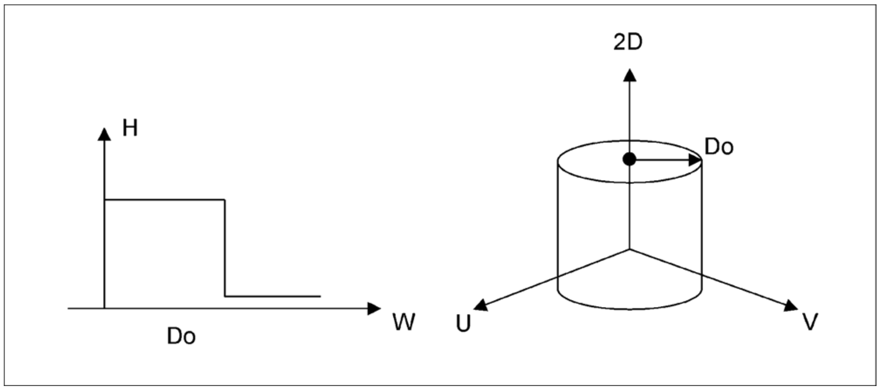
Tipos de Filtrado

- **Pasa bajos**
  - (Ej. Filtros butterworth)
- **Pasa banda**
- **Pasa altos**



3.2.2 Filtros pasa bajas

Filtro ideal pasa bajos



3.2.2 Filtros pasa bajas

Filtro ideal pasa bajos

- Un filtro de paso bajo bidimensional es aquel cuya función de transferencia verifica la relación:

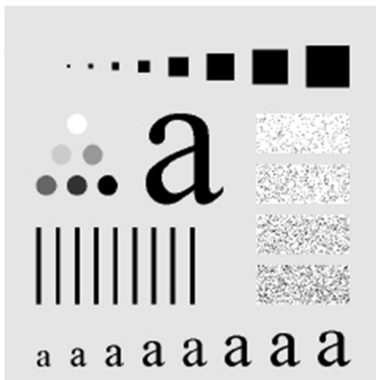
$$H(u,y)=\begin{cases}1 si D(u,v)\leq D_0\\0 si D(u,v)\succ D_0\end{cases}$$

Donde D_0 es una cantidad especificada no negativa , y $D(u.v)$ es la distancia desde el punto (u,v) al origen de coordenadas ene l plano de frecuencias, es decir:

$$D(u.v)=(u^2, v^2)^{1/2}$$

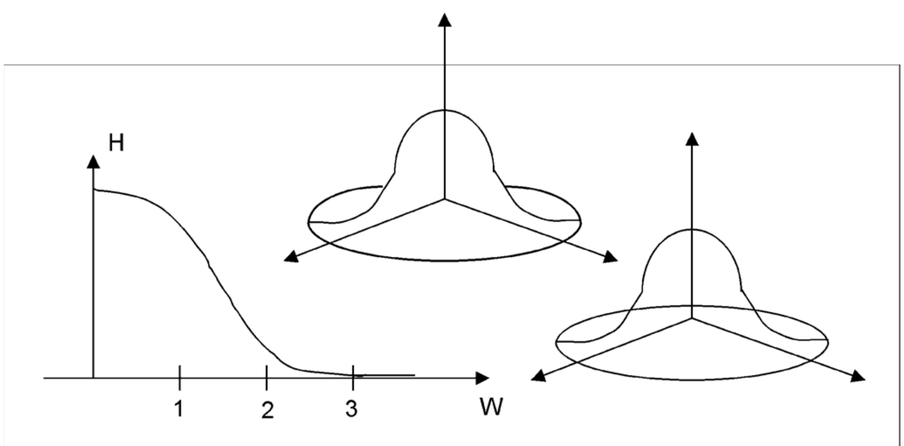
3.2.2 Filtros pasa bajas

- Filtro paso-bajo: “suavizan” las transiciones de grises



3.2.2 Filtros pasa bajas

Filtro Butterworth pasa-bajos



3.2.3 Filtros pasa altas

Filtro ideal pasa altas

- Un filtro de paso bajo bidimensional viene caracterizado por una función de transferencia que verifica la relación:

$$H(u,y) = \begin{cases} 0\, si D(u,v) \leq D_0 \\ 1\, si D(u,v) \succ D_0 \end{cases}$$

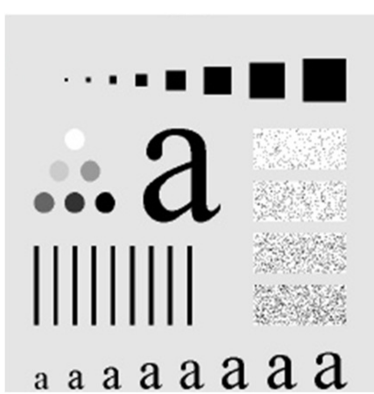
Donde D_0 es la distancia de corte medida desde el origen del plano de frecuencias, y $D(u.v)$ viene dado por la ecuación:

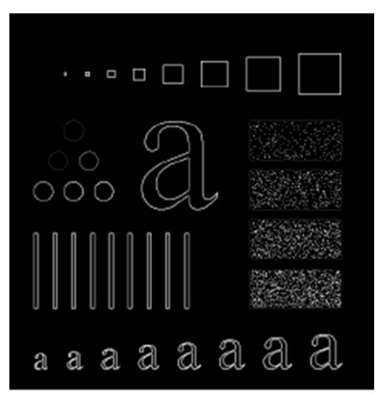
$$D(u.v)=(u^2,\ v^2)^{1/2}$$

NOTA: Al igual que el filtro ideal paso bajo, el filtro ideal paso alto no es físicamente realizable.

3.2.3 Filtros pasa altas

- Filtro paso-alto: enfatizan las transiciones de grises





Transformada de Fourier

Ejemplos:

- **Aplicación del Filtro Gaussiano y Laplaciano:**
http://www.youtube.com/watch?v=HXE4QC1kDDI

Para practicar:

✓ Applet de Java

✓ http://www.dai.ed.ac.uk/HIPR2/fourier.htm

✓ http://www.ee.siue.edu/~cvip/

✓ http://rsbweb.nih.gov/ij/applet/