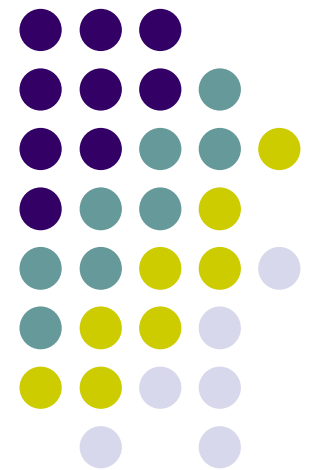
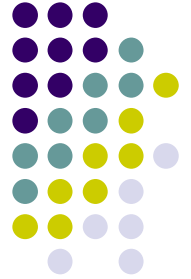


Procesamiento Digital de Imágenes

Pablo Roncagliolo B.

Nº 09





TRATAMIENTO DE IMÁGENES

EN EL DOMINIO DE LAS FRECUENCIAS

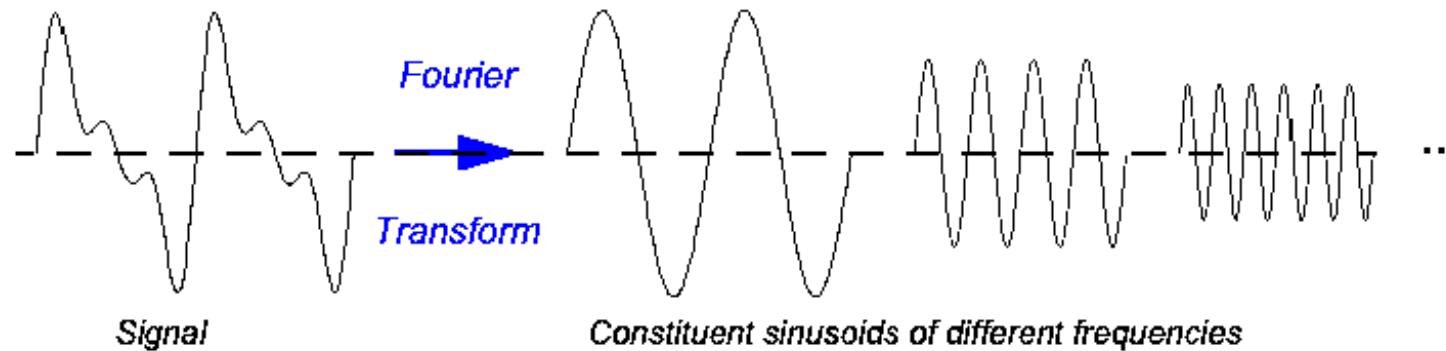
Transformada de Fourier



A principios del siglo XIX, Joseph Fourier indica que toda función periódica puede ser expresada como una suma infinita de senos y cosenos de distintas frecuencias.

El análisis de Fourier de una señal permite determinar sus frecuencias, pero a costa de perder la información de tipo temporal. (no dice cuando aparece cada frecuencia).

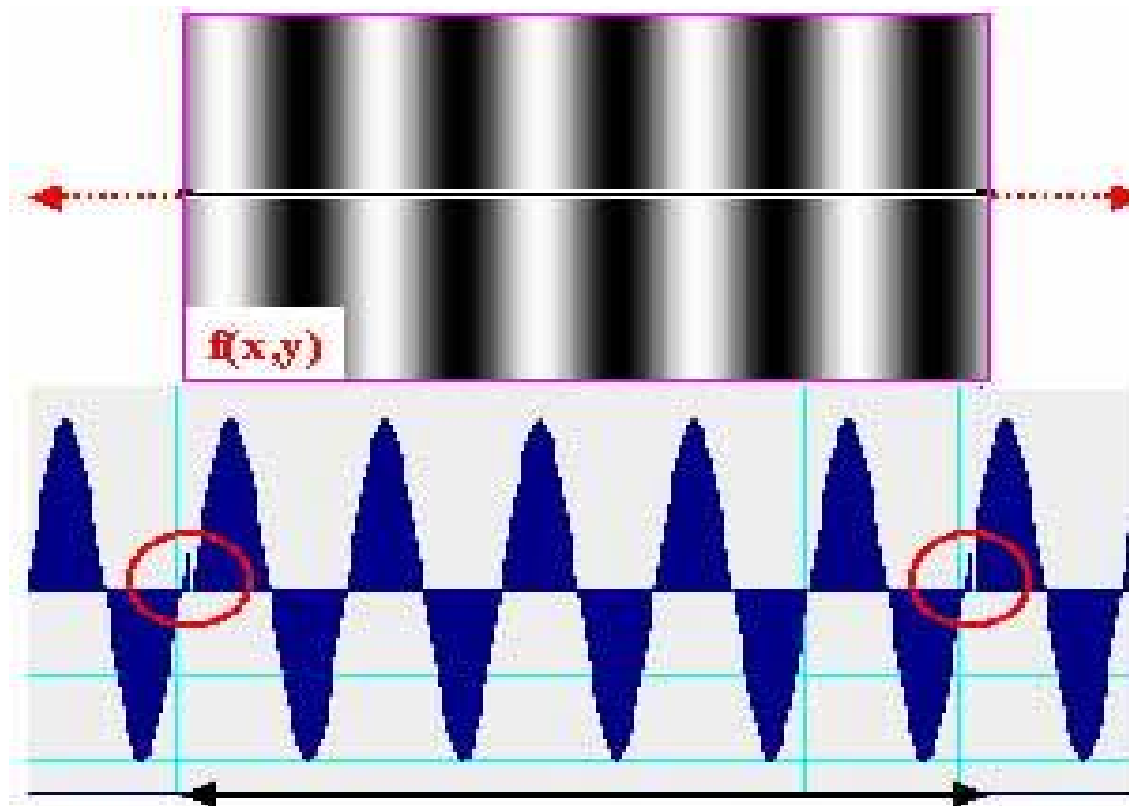
Transformada de Fourier



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.

El eje del tiempo se “reemplaza” por los ejes X-Y

Transformada de Fourier



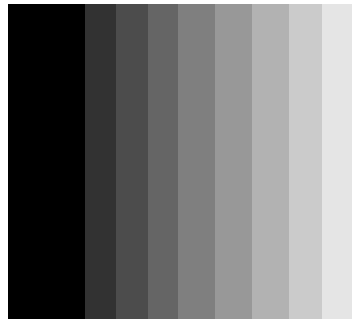
Transformada de Fourier



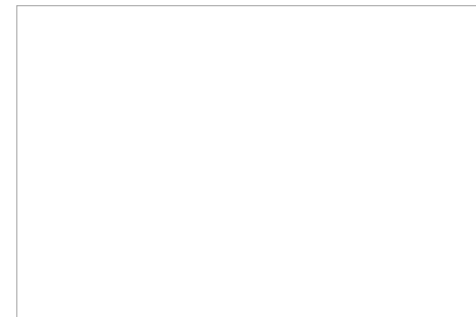
**IMAGEN CON ALTAS
FRECUENCIAS**



**IMAGEN CON BAJA
FRECUENCIA**



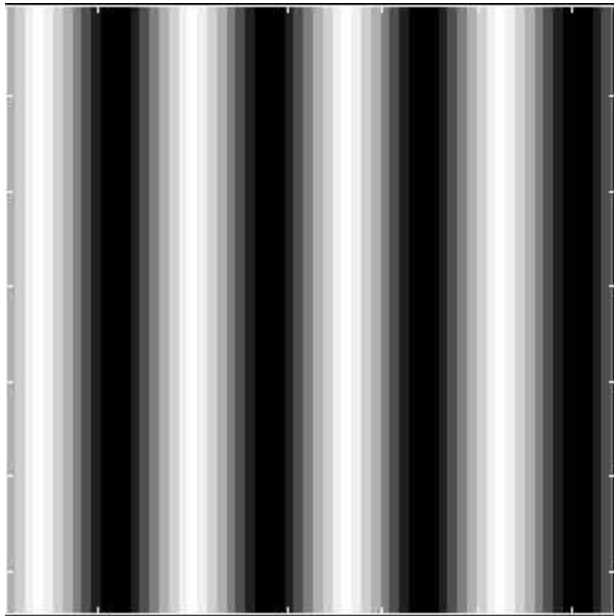
**IMAGEN CON FRECUENCIA
NULA**



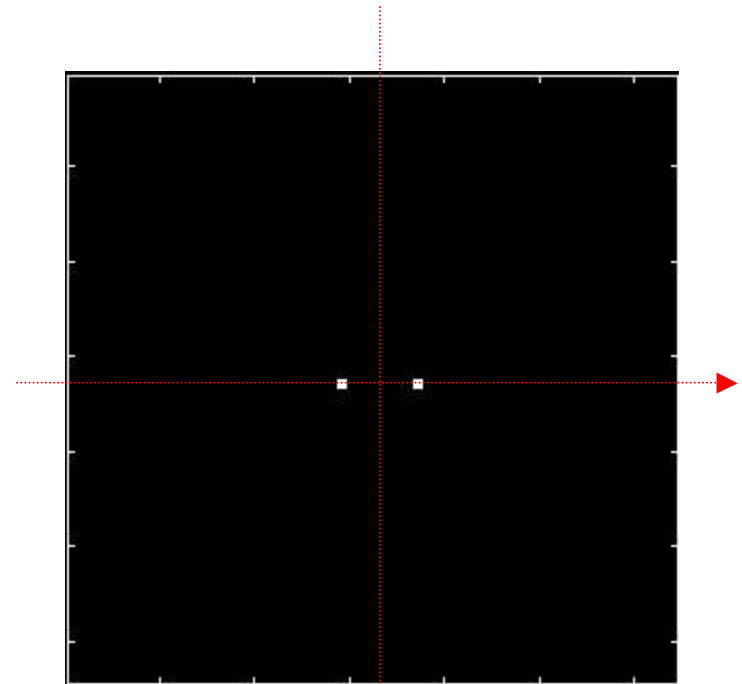
Transformada de Fourier



Imagen de grises con variación sinusoidal de intensidad a lo largo del eje x con período de 16 píxeles.



*Frecuencia: 1 ciclo / 16 píxel
... 0.0625 Hz**

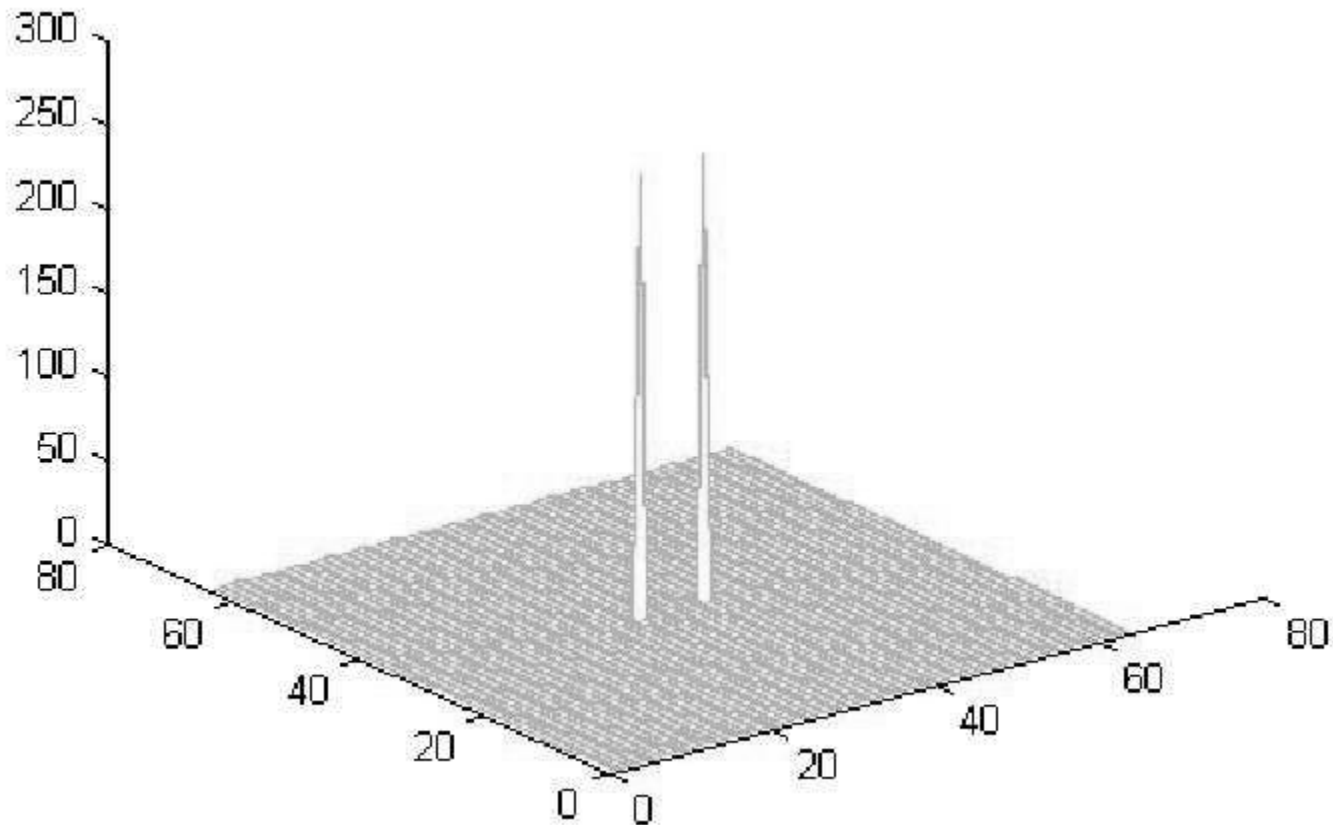


Transformada de Fourier, que muestra la frecuencia presente en el eje x (positiva y negativa)

Transformada de Fourier



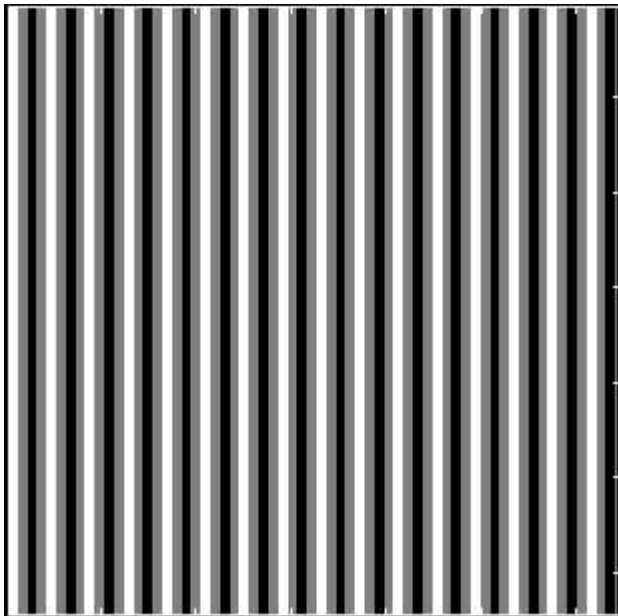
Transformada de Fourier en formato de “manto”



Transformada de Fourier

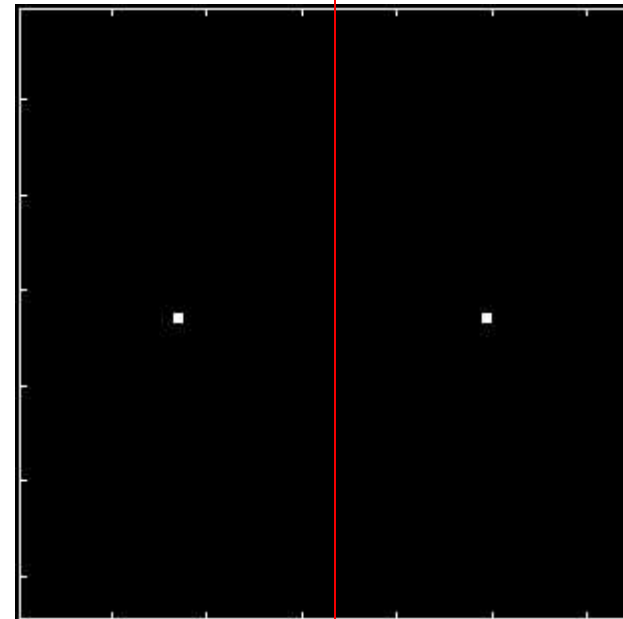


Imagen de grises con variación sinusoidal de intensidad a lo largo del eje x con período de 4 píxeles.



*Frecuencia: 1 ciclo / 4 píxel
... 0.25 Hz**

Hz = ciclos/píxel*

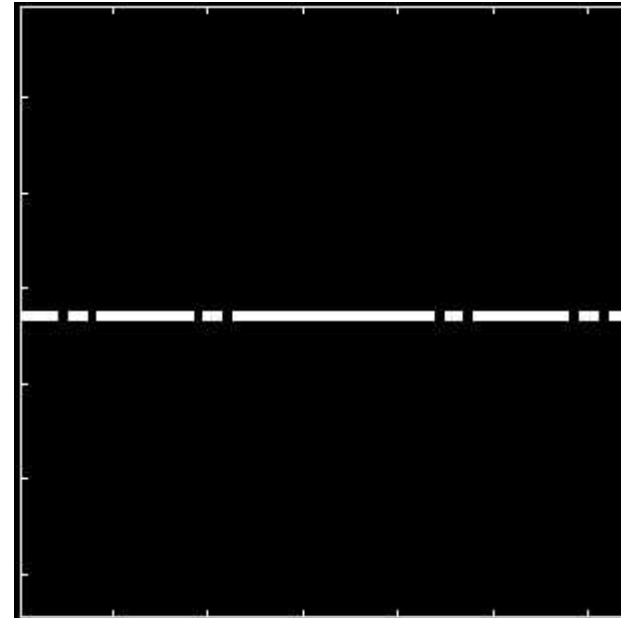
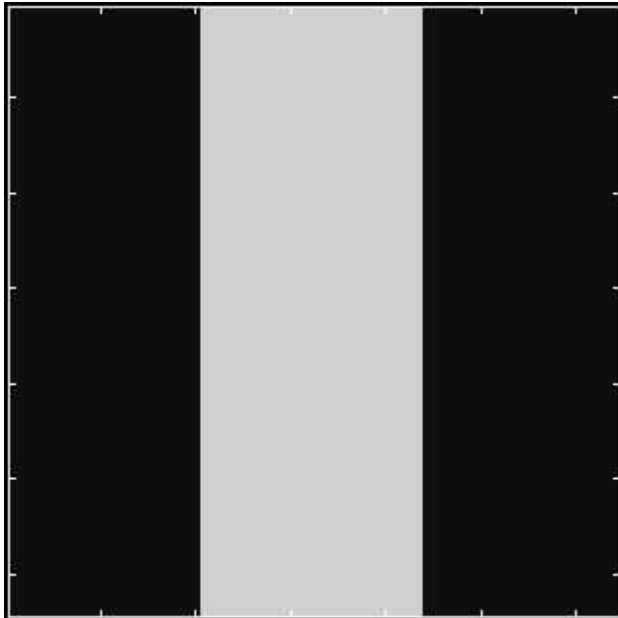


Transformada de Fourier, que muestra la frecuencia presente en el eje x (positiva y negativa)

Transformada de Fourier



*Imagen con una discontinuidad a lo largo
del eje X*

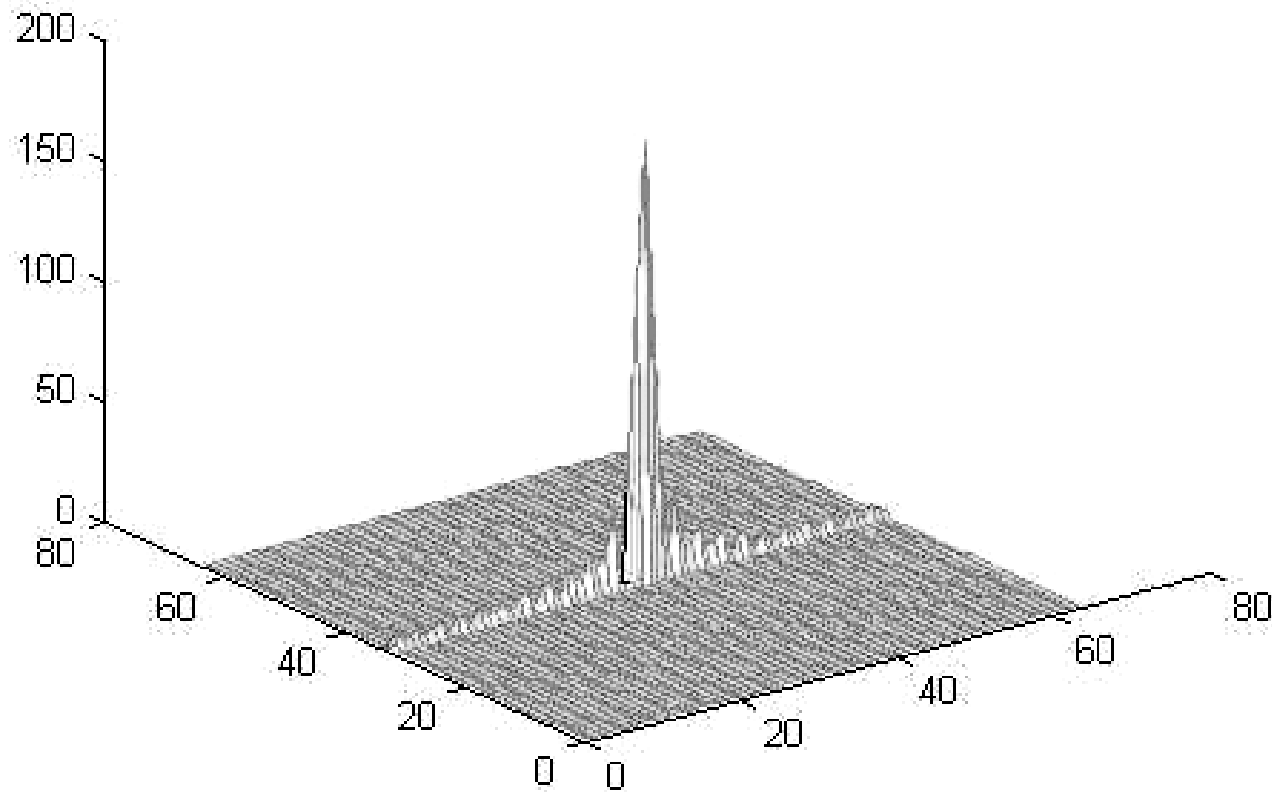


*Imagen binarizada de la transformada de
Fourier. Se observa como existen
frecuencias a lo largo de “u”*

Transformada de Fourier



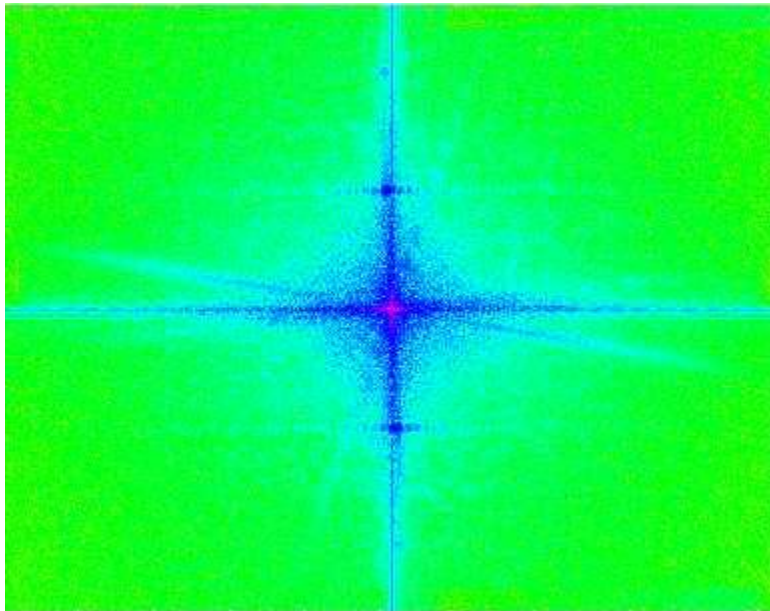
Vista del “manto” de la transformada de Fourier de una discontinuidad en el eje X



Transformada de Fourier



TF de una imagen



TF de un Filtro



Transformada de Fourier



Imagen original

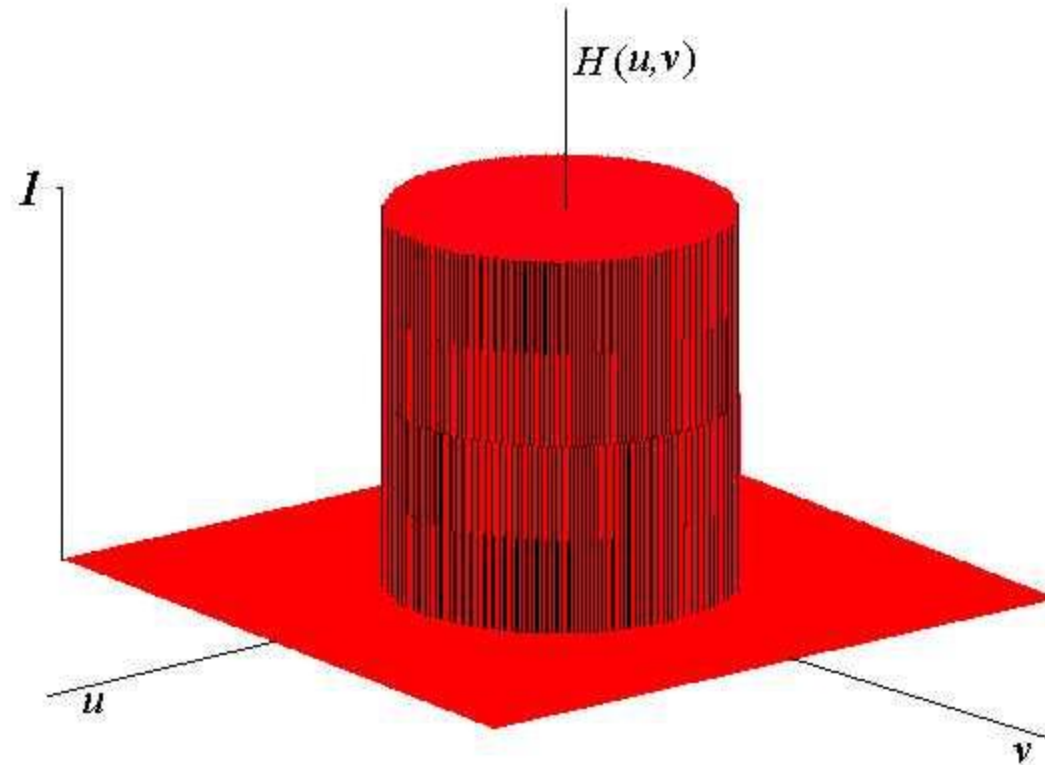


*TF inversa de la convolución de
TF original con el filtro*



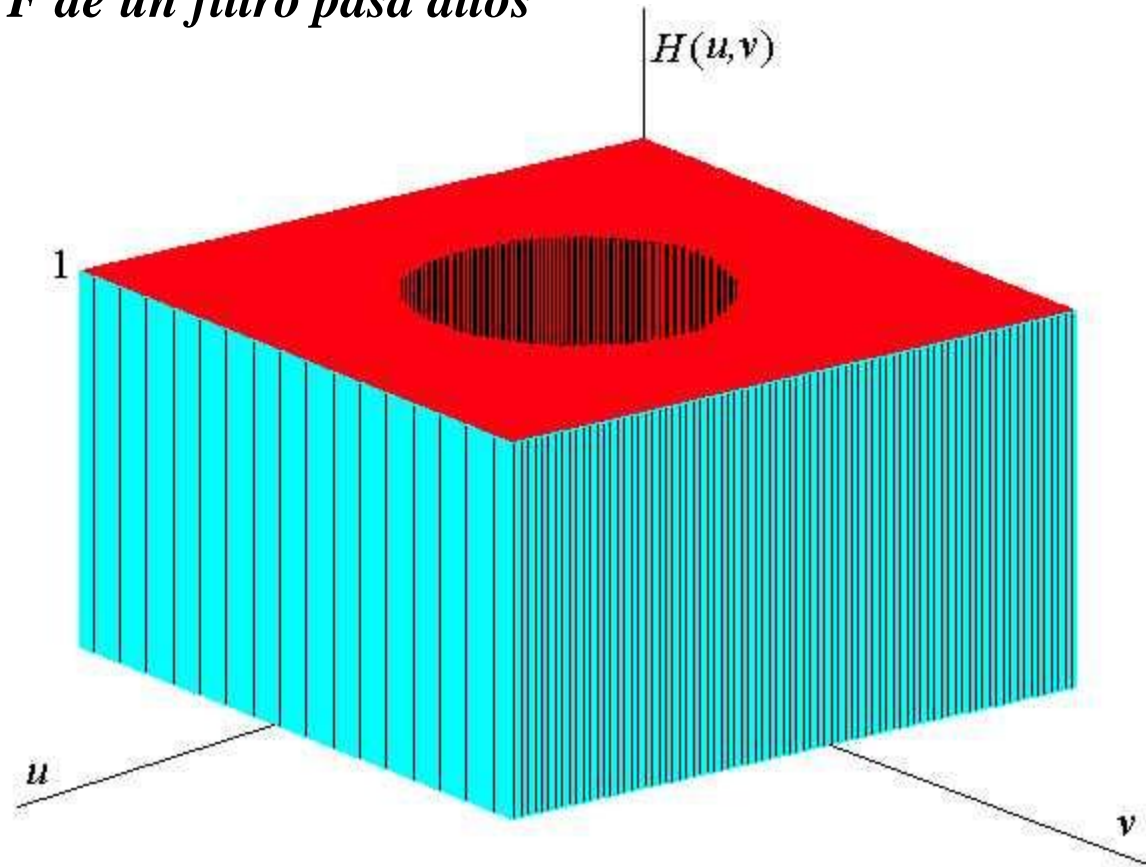
Transformada de Fourier

Imagen de TF de un filtro pasa bajos



Transformada de Fourier

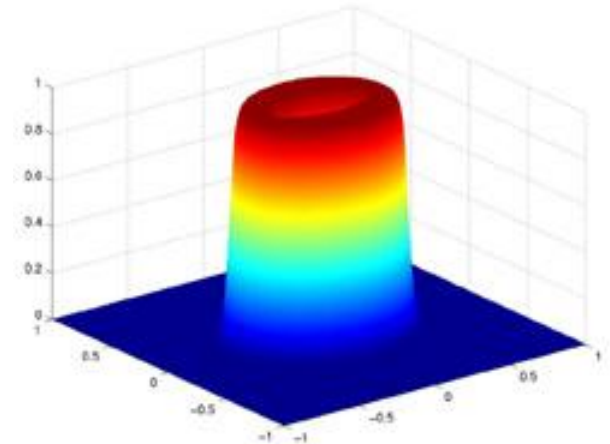
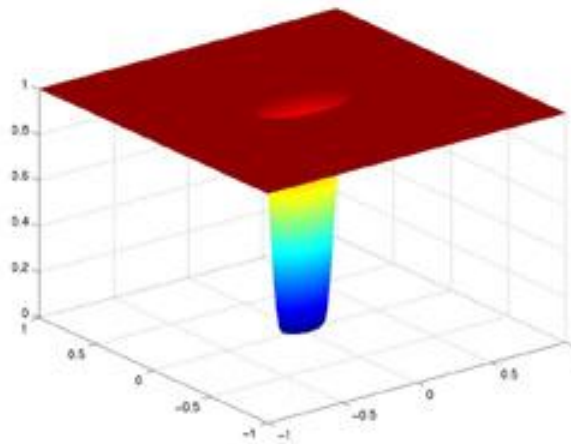
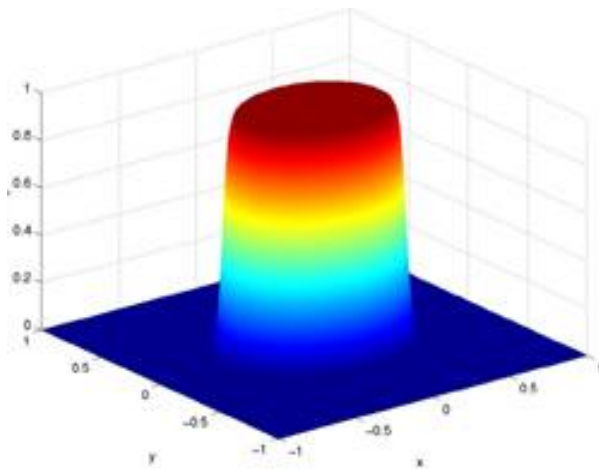
Imagen de TF de un filtro pasa altos



Transformada de Fourier



Filtros pasa banda:



Se pueden construir en base a un FPB y un FPA.

Tratamiento de Imágenes:

Dominio Frecuencias:

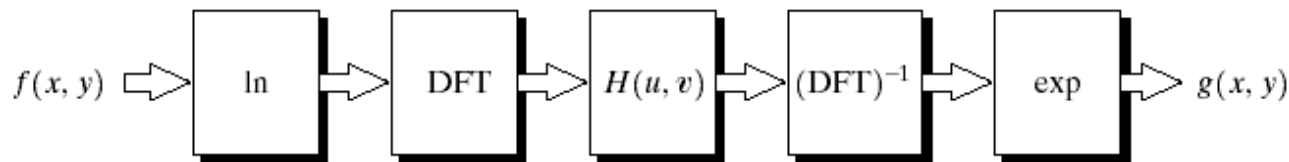
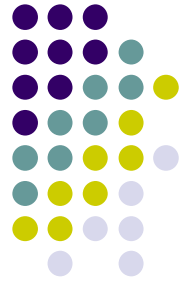


FIGURE 4.31
Homomorphic
filtering approach
for image
enhancement.

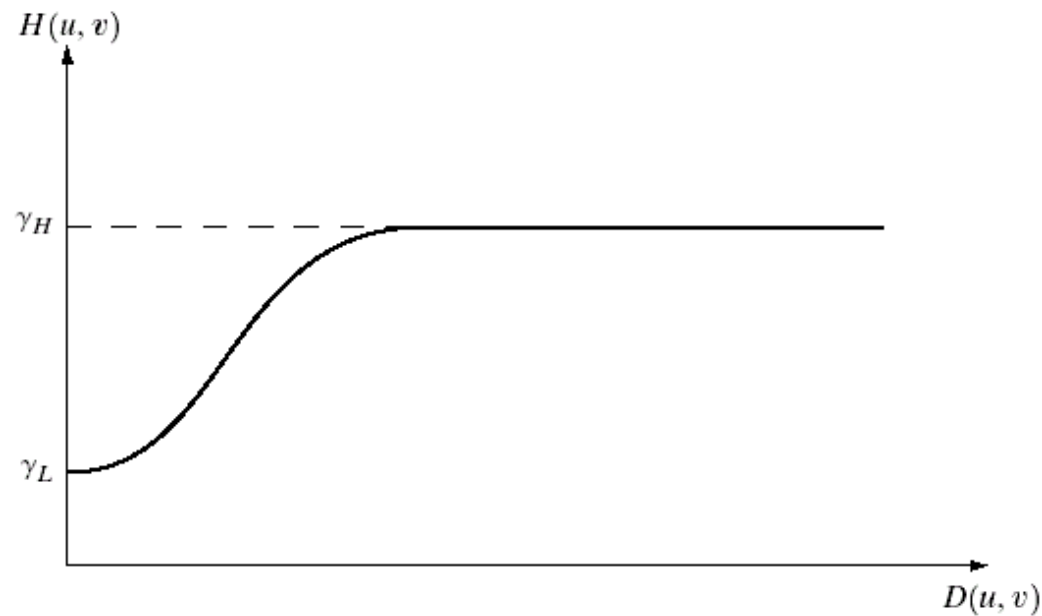


FIGURE 4.32
Cross section of a
circularly
symmetric filter
function. $D(u, v)$
is the distance
from the origin of
the centered
transform.

Tratamiento de Imágenes:

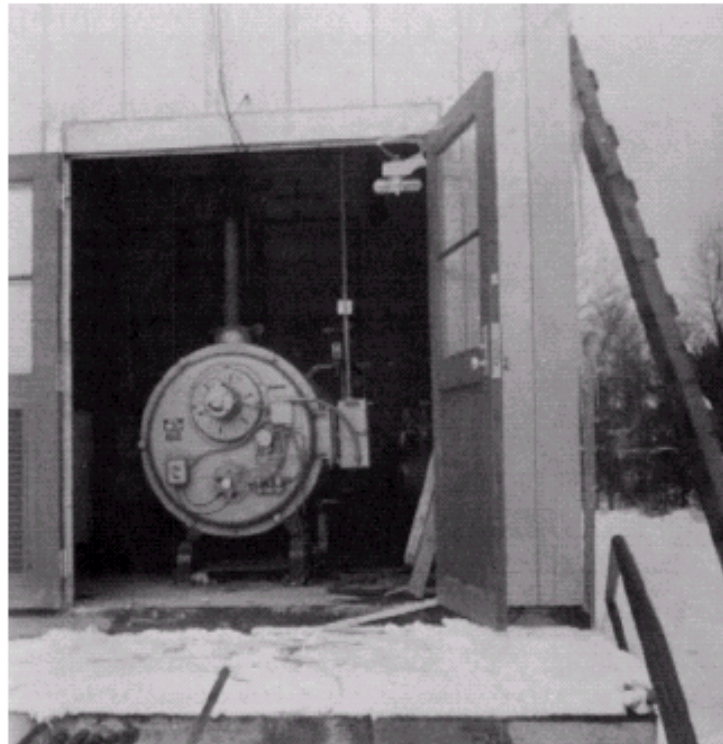
Dominio Frecuencias:



a b

FIGURE 4.33

(a) Original image. (b) Image processed by homomorphic filtering (note details inside shelter). (Stockham.)



Tratamiento de Imágenes:

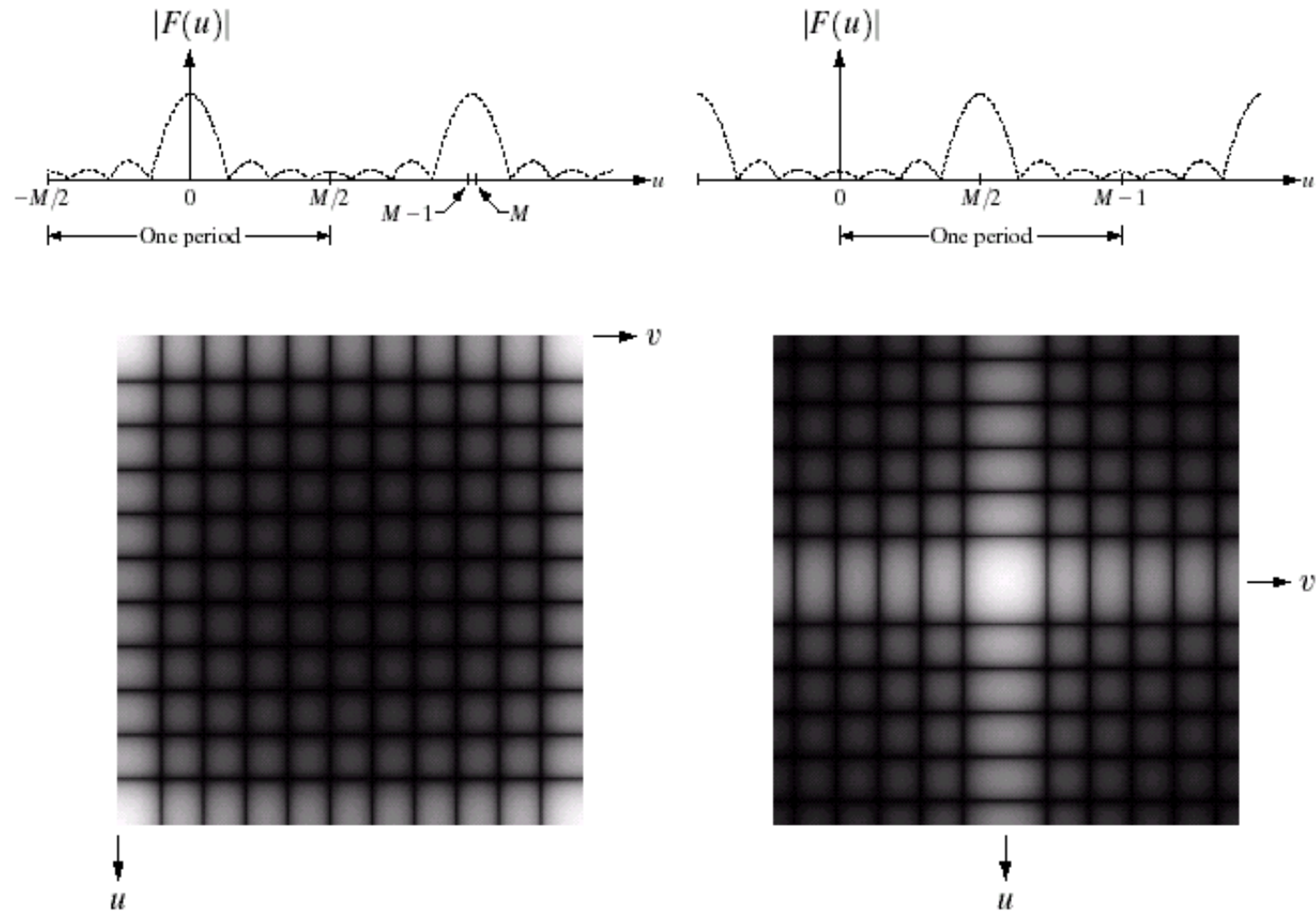
Dominio Frecuencias:



a b
c d

FIGURE 4.34

(a) Fourier spectrum showing back-to-back half periods in the interval $[0, M - 1]$.
 (b) Shifted spectrum showing a full period in the same interval.
 (c) Fourier spectrum of an image, showing the same back-to-back properties as (a), but in two dimensions.
 (d) Centered Fourier spectrum.



Tratamiento de Imágenes:

Dominio Frecuencias:

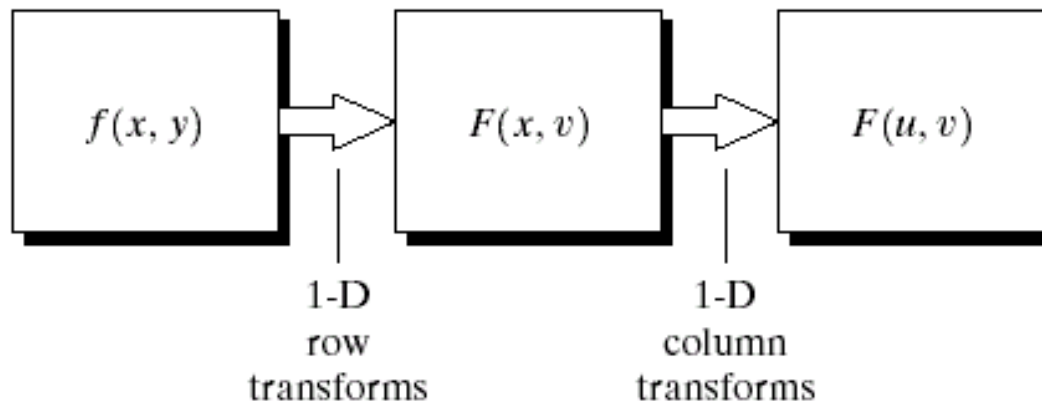
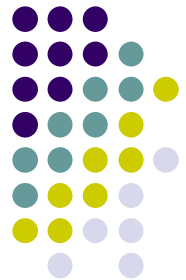


FIGURE 4.35
Computation of
the 2-D Fourier
transform as a
series of 1-D
transforms.

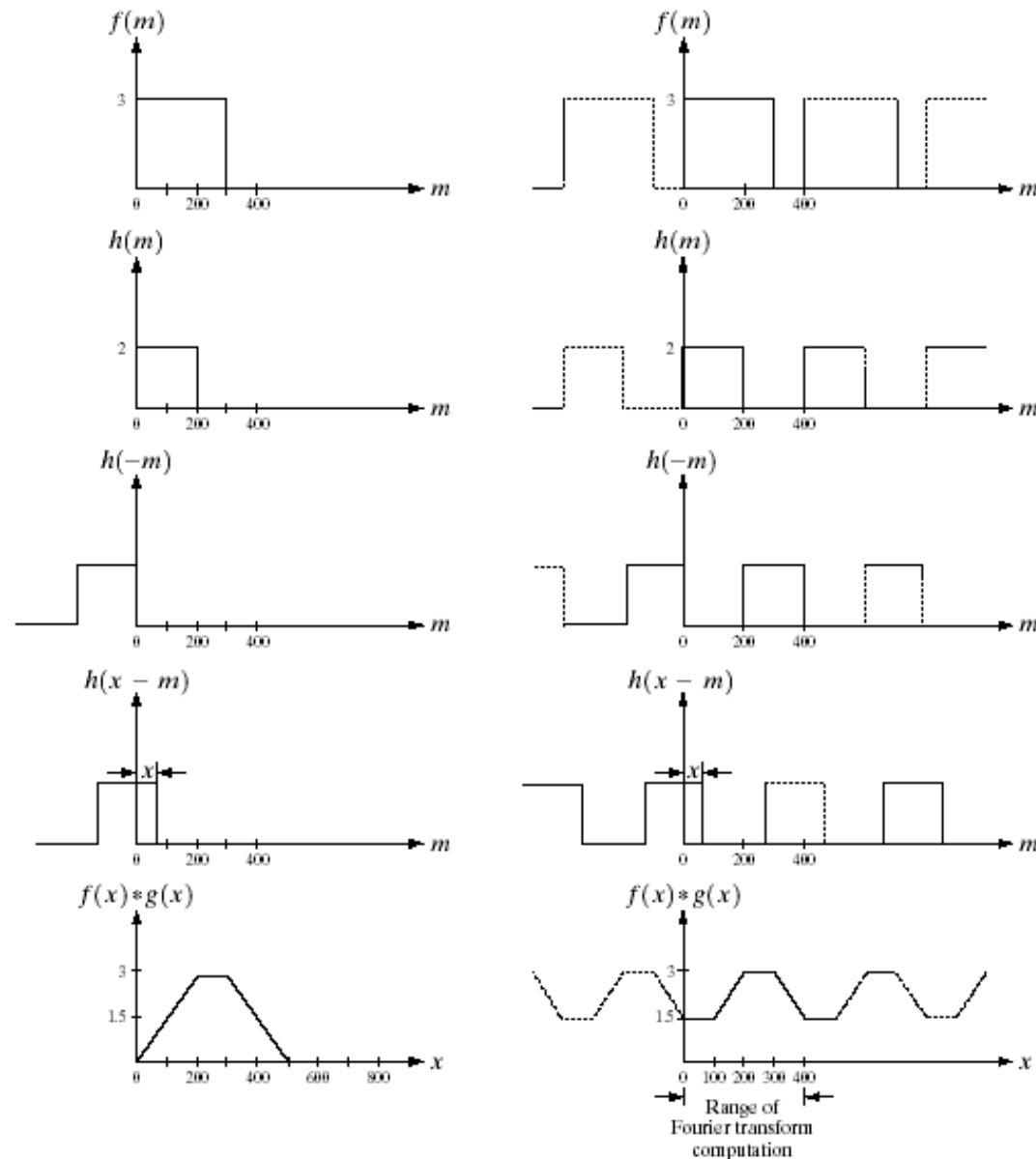
Tratamiento de Imágenes:

Dominio Frecuencias:



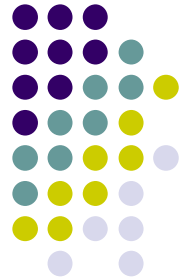
a f
b g
c h
d i
e j

FIGURE 4.36 Left: convolution of two discrete functions. Right: convolution of the same functions, taking into account the implied periodicity of the DFT. Note in (j) how data from adjacent periods corrupt the result of convolution.



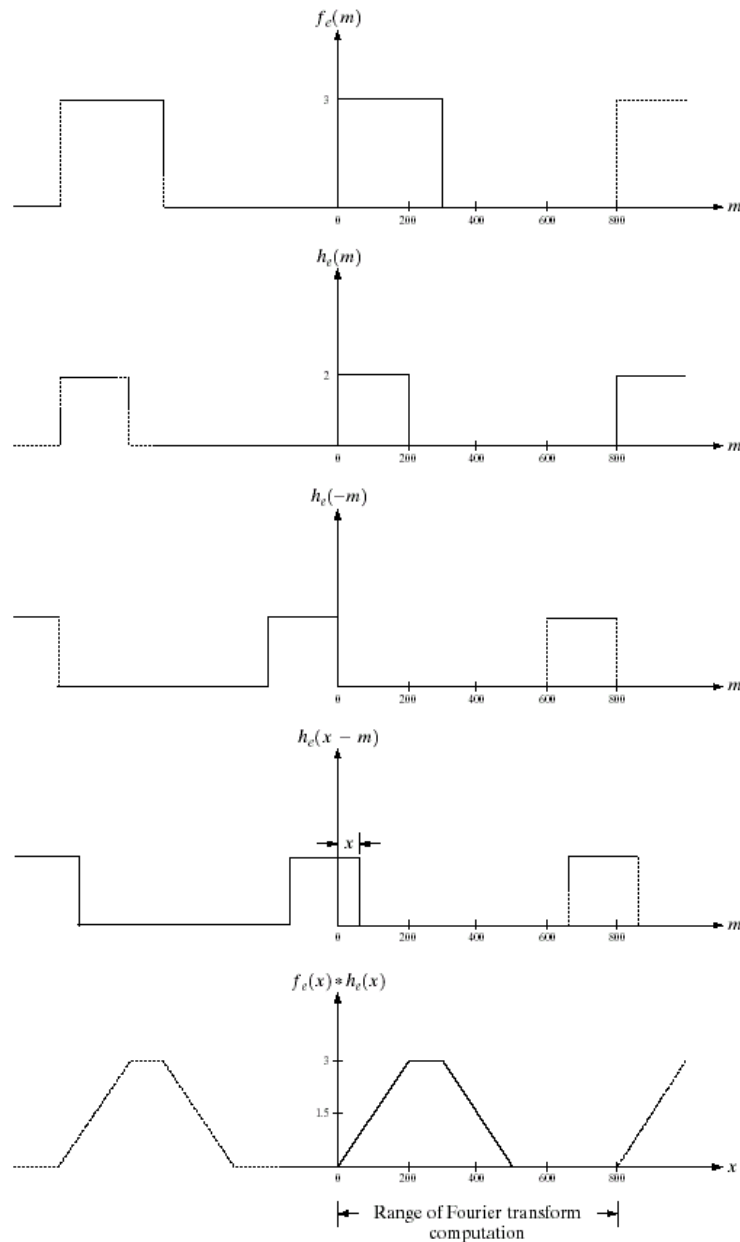
Tratamiento de Imágenes:

Dominio Frecuencias:



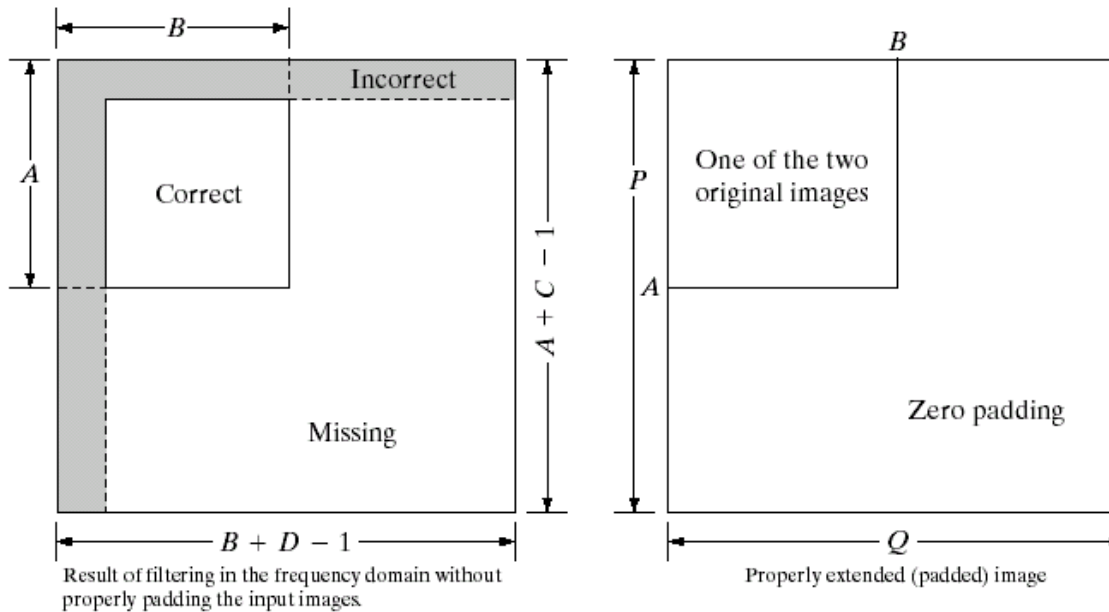
a
b
c
d
e

FIGURE 4.37
Result of performing
convolution with
extended
functions.
Compare
Figs. 4.37(e) and
4.36(e).



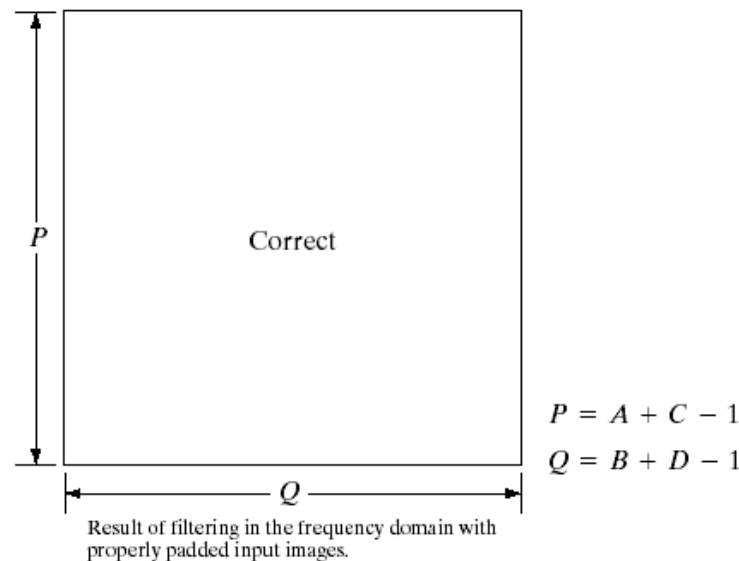
Tratamiento de Imágenes:

Dominio Frecuencias:



a b
c

FIGURE 4.38 Illustration of the need for function padding. (a) Result of performing 2-D convolution without padding. (b) Proper function padding. (c) Correct convolution result.



Tratamiento de Imágenes:

Dominio Frecuencias:

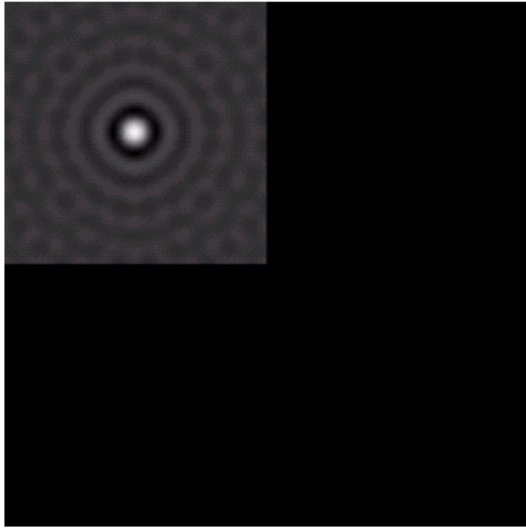


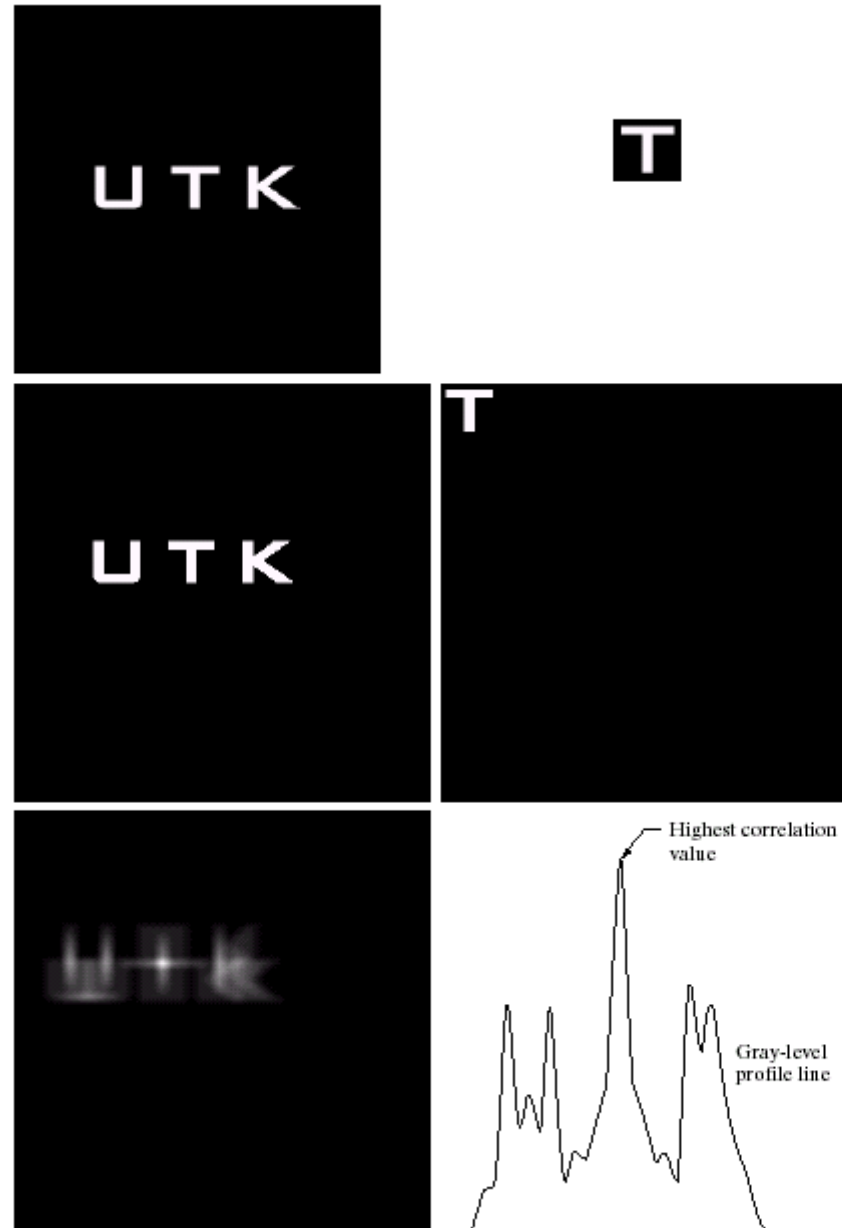
FIGURE 4.39 Padded lowpass filter is the spatial domain (only the real part is shown).



FIGURE 4.40 Result of filtering with padding. The image is usually cropped to its original size since there is little valuable information past the image boundaries.

Tratamiento de Imágenes:

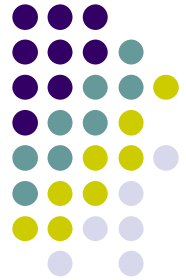
Dominio Frecuencias:



a	b
c	d
e	f

FIGURE 4.41

(a) Image.
(b) Template.
(c) and
(d) Padded
images.
(e) Correlation
function displayed
as an image.
(f) Horizontal
profile line
through the
highest value in
(e), showing the
point at which the
best match took
place.



Transformada de Fourier



$$F(u, v) = \frac{1}{nc \cdot nf} \sum_{c=1}^{nc} \sum_{f=1}^{nf} I(c, f) \cdot \exp(-2\pi i(u \cdot c / nc + v \cdot f / nf))$$

generalmente se visualiza el módulo :

$$|F(u, v)| = \sqrt{[\text{Re}(u, v)]^2 + [\text{Im}(u, v)]^2}$$

Transformada de Fourier

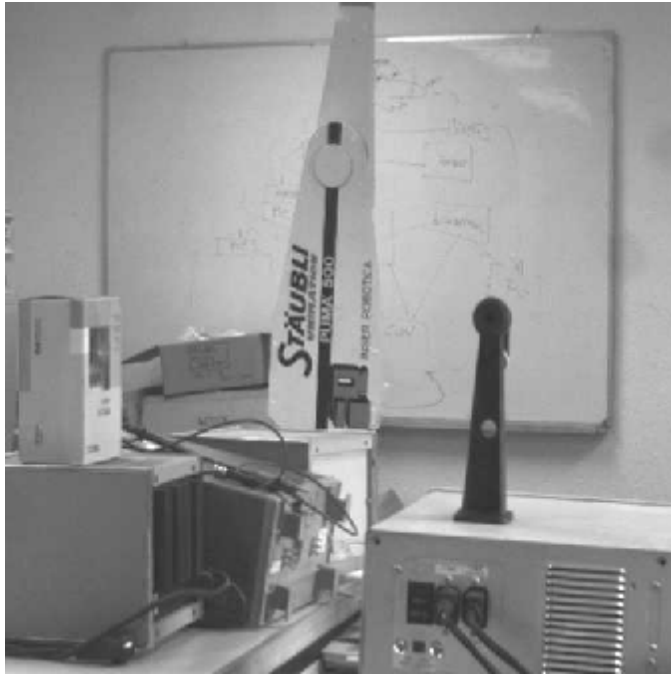


$$F(u, v) = \frac{1}{nc \cdot nf} \sum_{c=1}^{nc} \sum_{f=1}^{nf} I(c, f) \cdot \exp(-2\pi i(u \cdot c / nc + v \cdot f / nf))$$

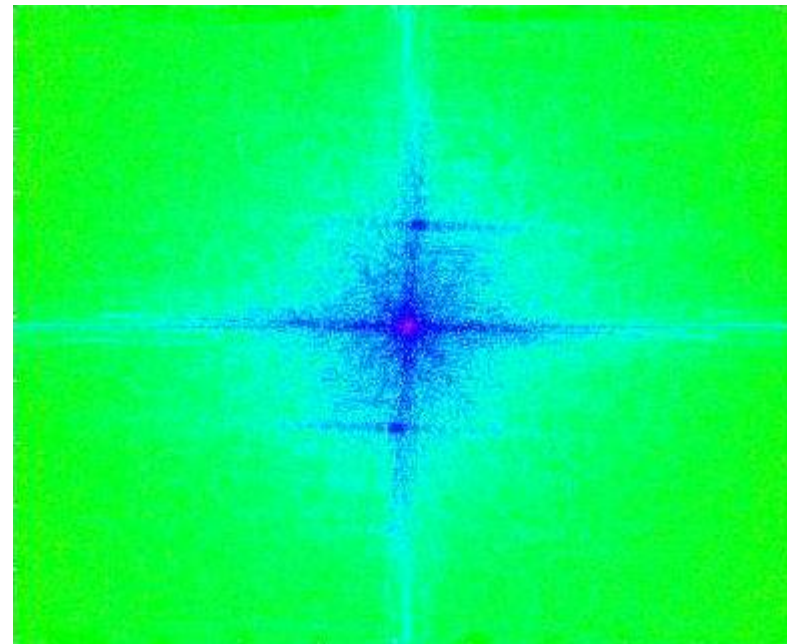
Expresión periódica de la Transformada de Fourier:

$$F(u, v) = \frac{1}{nc \cdot nf} I(c, f) \cdot [\cos(2\pi(u \cdot c / nc + v \cdot f / nf)) + i \cdot \sin(2\pi(u \cdot c / nc + v \cdot f / nf))]$$

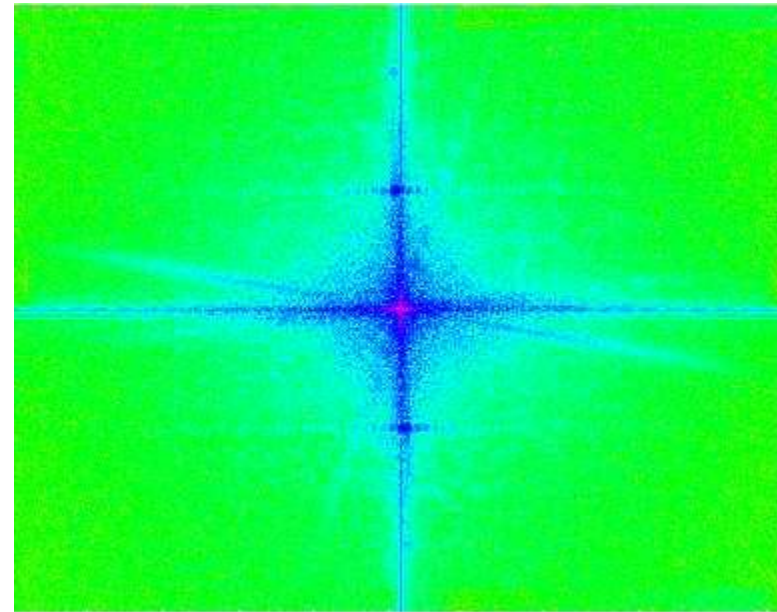
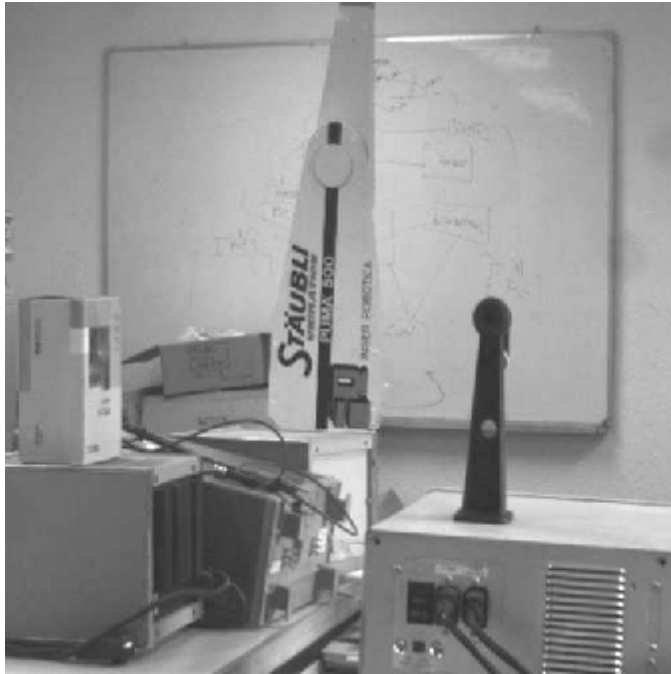
Transformada de Fourier



El rango dinámico del espectro de Fourier es mayor al rango típico de las imágenes (256). Para una mejor visualización se puede utilizar el logaritmo de la transformada.



Transformada de Fourier



$$FF(u,v)=255*\log(1+ |F(u,v)|)/\max(\log(1+|F(u,v)|))$$

Normalización de la matriz en el rango 0-255

Transformada de Fourier



Valor Medio:

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

evaluando para $u, v = 0, 0$:

$$F(0, 0) = \frac{1}{nc \cdot nf} \sum_{x=0}^{nc-1} \sum_{y=0}^{nf-1} f(x, y) \cdot \exp(0)$$

$$F(0, 0) = \frac{1}{nc \cdot nf} \sum_{x=0}^{nc-1} \sum_{y=0}^{nf-1} f(x, y) = \text{promedio}$$

Transformada de Fourier



Algoritmo:

Sin embargo, existe una implementación más eficiente de la transformada denominada FFT (Transformada rápida de Fourier)

```
double A[][xydim];
double freal[][xydim];
double fimag[][xydim];
double phi, sum1, sum2;

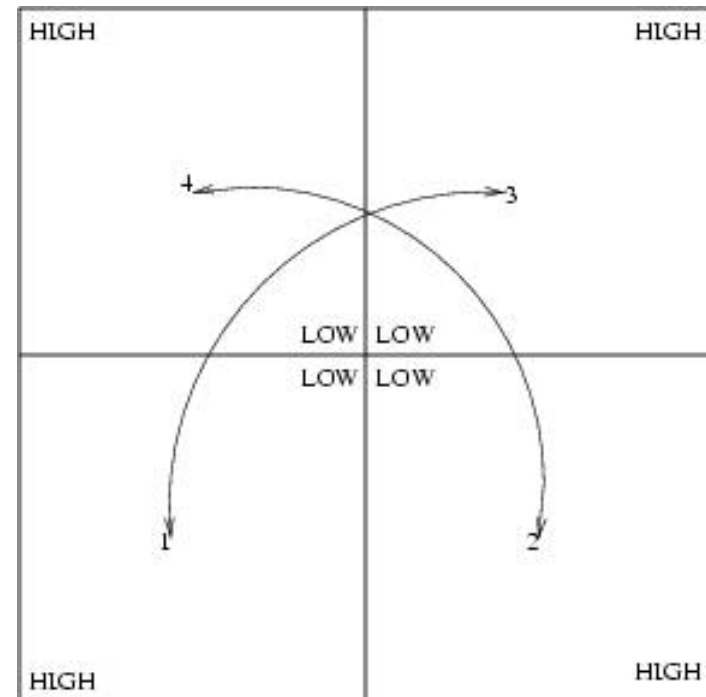
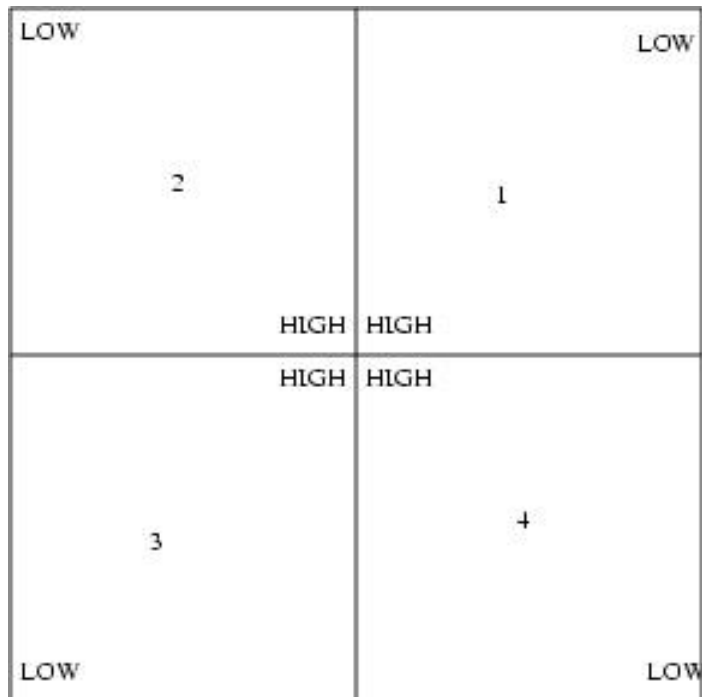
for (int i = 0; i <N; i++)
  for (int j = 0; j <N, j++) {
    sum1 = 0;
    sum2 = 0;
    for (int x = 0; x <N; x++)
      for (int y = 0; y <N, y++) {
        phi = 2 * PI *(i * x + j * y) / N;
        sum1 = sum1 + A[x][y] * cos(phi);
        sum2 = sum2 + A[x][y] * sin(phi);
      }

    freal[i][j] = sum1 / N;
    fimag[i][j] = -sum2 / N;
  }
```

Transformada de Fourier



Algoritmo:



Tratamiento de Imágenes:

Dominio Frecuencias:

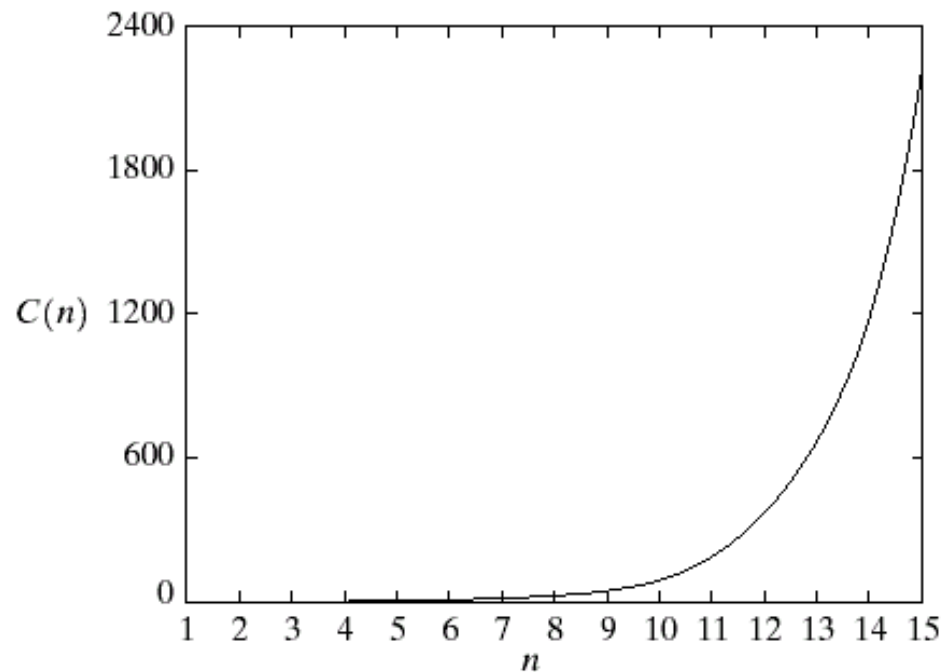
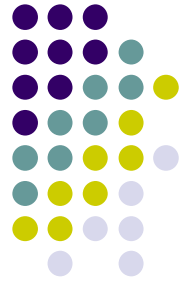


FIGURE 4.42
Computational advantage of the FFT over a direct implementation of the 1-D DFT. Note that the advantage increases rapidly as a function of n .

Tratamiento de Imágenes:

Dominio Frecuencias:



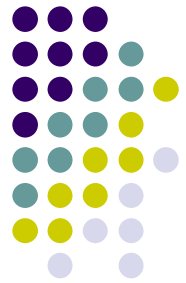
TABLE 4.1

Summary of some important properties of the 2-D Fourier transform.

Property	Expression(s)
Fourier transform	$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)}$
Inverse Fourier transform	$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)}$
Polar representation	$F(u, v) = F(u, v) e^{-j\phi(u, v)}$
Spectrum	$ F(u, v) = [R^2(u, v) + I^2(u, v)]^{1/2}, \quad R = \text{Real}(F) \text{ and } I = \text{Imag}(F)$
Phase angle	$\phi(u, v) = \tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right]$
Power spectrum	$P(u, v) = F(u, v) ^2$
Average value	$\bar{f}(x, y) = F(0, 0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)$
Translation	$f(x, y) e^{j2\pi(u_0 x/M + v_0 y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v) e^{-j2\pi(ux_0/M + vy_0/N)}$ <p>When $x_0 = u_0 = M/2$ and $y_0 = v_0 = N/2$, then</p> $f(x, y) (-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v) (-1)^{u+v}$

Tratamiento de Imágenes:

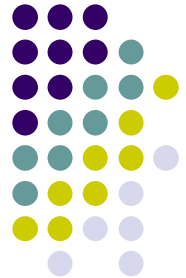
Dominio Frecuencias:



Conjugate symmetry	$F(u, v) = F^*(-u, -v)$ $ F(u, v) = F(-u, -v) $
Differentiation	$\frac{\partial^n f(x, y)}{\partial x^n} \Leftrightarrow (ju)^n F(u, v)$ $(-jx)^n f(x, y) \Leftrightarrow \frac{\partial^n F(u, v)}{\partial u^n}$
Laplacian	$\nabla^2 f(x, y) \Leftrightarrow -(u^2 + v^2)F(u, v)$
Distributivity	$\Im[f_1(x, y) + f_2(x, y)] = \Im[f_1(x, y)] + \Im[f_2(x, y)]$ $\Im[f_1(x, y) \cdot f_2(x, y)] \neq \Im[f_1(x, y)] \cdot \Im[f_2(x, y)]$
Scaling	$af(x, y) \Leftrightarrow aF(u, v), f(ax, by) \Leftrightarrow \frac{1}{ ab } F(u/a, v/b)$
Rotation	$x = r \cos \theta \quad y = r \sin \theta \quad u = \omega \cos \varphi \quad v = \omega \sin \varphi$ $f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$
Periodicity	$F(u, v) = F(u + M, v) = F(u, v + N) = F(u + M, v + N)$ $f(x, y) = f(x + M, y) = f(x, y + N) = f(x + M, y + N)$
Separability	<p>See Eqs. (4.6-14) and (4.6-15). Separability implies that we can compute the 2-D transform of an image by first computing 1-D transforms along each row of the image, and then computing a 1-D transform along each column of this intermediate result. The reverse, columns and then rows, yields the same result.</p>

Tratamiento de Imágenes:

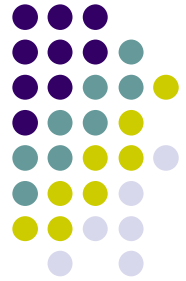
Dominio Frecuencias:



Property	Expression(s)
Computation of the inverse Fourier transform using a forward transform algorithm	$\frac{1}{MN} f^*(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v) e^{-j2\pi(ux/M + vy/N)}$ <p>This equation indicates that inputting the function $F^*(u, v)$ into an algorithm designed to compute the forward transform (right side of the preceding equation) yields $f^*(x, y)/MN$. Taking the complex conjugate and multiplying this result by MN gives the desired inverse.</p>
Convolution [†]	$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n)$
Correlation [†]	$f(x, y) \circ h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n) h(x + m, y + n)$
Convolution theorem [†]	$f(x, y) * h(x, y) \Leftrightarrow F(u, v) H(u, v);$ $f(x, y) h(x, y) \Leftrightarrow F(u, v) * H(u, v)$
Correlation theorem [†]	$f(x, y) \circ h(x, y) \Leftrightarrow F^*(u, v) H(u, v);$ $f^*(x, y) h(x, y) \Leftrightarrow F(u, v) \circ H(u, v)$

Tratamiento de Imágenes:

Dominio Frecuencias:



Some useful FT pairs:

Impulse $\delta(x, y) \Leftrightarrow 1$

Gaussian $A\sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2(x^2+y^2)} \Leftrightarrow Ae^{-(u^2+v^2)/2\sigma^2}$

Rectangle $\text{rect}[a, b] \Leftrightarrow ab \frac{\sin(\pi ua)}{(\pi ua)} \frac{\sin(\pi vb)}{(\pi vb)} e^{-j\pi(ua+vb)}$

Cosine $\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$
 $\frac{1}{2} [\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)]$

Sine $\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$
 $j \frac{1}{2} [\delta(u + u_0, v + v_0) - \delta(u - u_0, v - v_0)]$

[†] Assumes that functions have been extended by zero padding.

Transformada de Fourier



En Matlab:

```
F = fft2(IM);

% Se debe realizar un "swap" entre los cuadrantes
% para visualizar de forma estándar

F2 = fftshift(F);

% generalmente se visualiza el valor absoluto

S = abs(F2);
imshow(S,[]);

% para mejorar rango dinámico de visualización

S2 = log(1+S);
imshow(S2,[]);
```

Transformada de Fourier



En Matlab:

```
F = fft2(IM);

% Se debe realizar un "swap" entre los cuadrantes
% para visualizar de forma estándar

F2 = fftshift(F);

% generalmente se visualiza el valor absoluto

S = abs(F2);
imshow(S,[]);

% para mejorar rango dinámico de visualización

S2 = log(1+S);
imshow(S2,[]);
```