Fourier Transform - Part II

- Discrete Fourier Transform - 1D
- Discrete Fourier Transform - 2D
- Fourier Properties
- Convolution Theorem
- FFT
- Examples

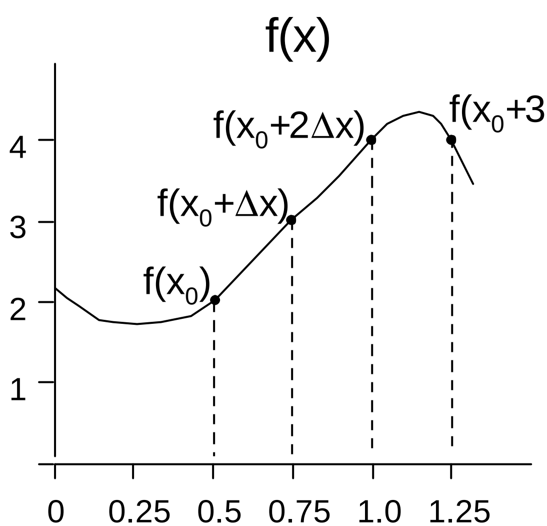
Discrete Fourier Transform

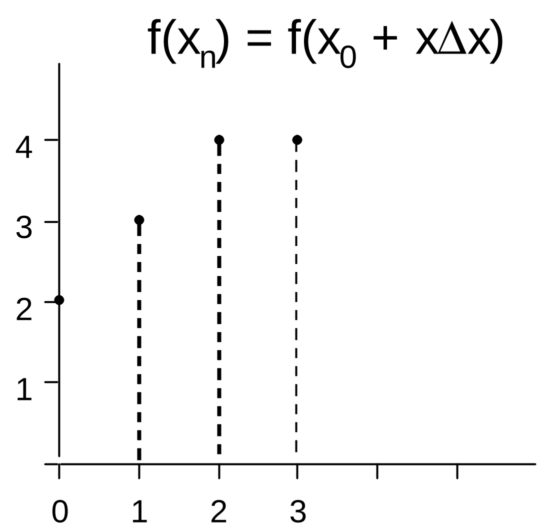
Move from f(x) $(x \in R)$ to f(x) $(x \in Z)$
by sampling at equal intervals.

$f(x_0), f(x_0+\Delta x), f(x_0+2\Delta x), \ldots , f(x_0+[n-1]\Delta x),$

Given N samples at equal intervals, we redefine f as:

$$f(x) = f(x_0+x\Delta x) \qquad x = 0, 1, 2, \ldots , N-1$$





Discrete Fourier Transform

The Discrete Fourier Transform (DFT) is defined as:

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{\frac{-2piux}{N}} \qquad u = 0, 1, 2, ..., N-1$$

Matlab: F=fft(f);

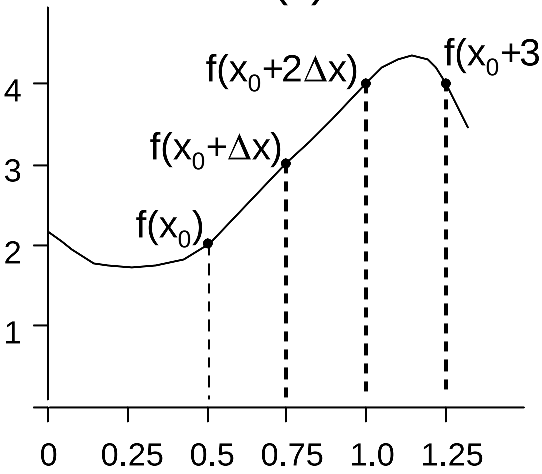
The Inverse Discrete Fourier Transform (IDFT) is defined as:

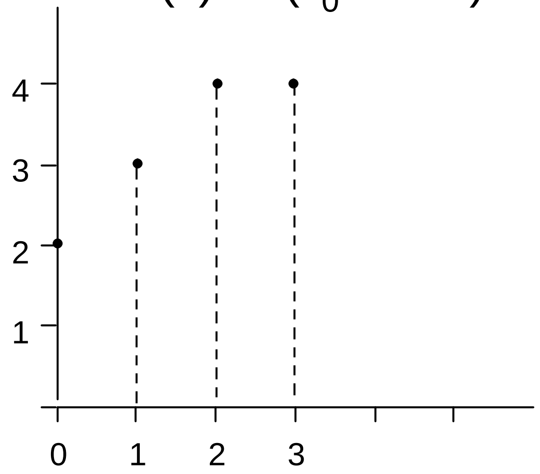
$$f(x) = \sum_{u=0}^{N-1} F(u)e^{\frac{2piux}{N}} \qquad x = 0, 1, 2, ..., N-1$$

Matlab: F=ifft(f);

Discrete Fourier Transform - Example

$f(x)$ $\qquad\qquad$ $f(x) = f(x_0 + x\Delta x)$





$$F(0) = 1/4 \sum_{x=0}^{3} f(x)\, e^{\frac{-2\pi i 0x}{4}} = 1/4 \sum_{x=0}^{3} f(x)\, 1$$

$$= 1/4(f(0) + f(1) + f(2) + f(3)) = 1/4(2+3+4+4) = 3.25$$

$$F(1) = 1/4 \sum_{x=0}^{3} f(x)\, e^{\frac{-2\pi i x}{4}} = 1/4\, [2e^{0}+3e^{-i\pi/2}+4e^{-\pi i}\; 4e^{-i3\pi/2}] = \frac{1}{4}\, [-2+i]$$

$$F(2) = 1/4 \sum_{x=0}^{3} f(x)\, e^{\frac{-4\pi i x}{4}} = 1/4\, [2e^{0}+3e^{-i\pi}+4e^{-2\pi i}\; 4e^{-3\pi i}] = \frac{-1}{4}\, [-1-0i] = \frac{-1}{4}$$

$$F(3) = 1/4 \sum_{x=0}^{3} f(x)\, e^{\frac{-6\pi i x}{4}} = 1/4\, [2e^{0}+3e^{-i3\pi/2}+4e^{-3\pi i}\; 4e^{-i9\pi/2}] = \frac{1}{4}\, [-2-i]$$

Fourier Spectrum:

$$|F(0)| = 3.25$$

$$|F(1)| = [(-1/2)^2 + (1/4)^2]^{0.5}$$

$$|F(2)| = [(-1/4)^2 + (0)^2]^{0.5}$$

$$|F(3)| = [(-1/2)^2 + (-1/4)^2]^{0.5}$$

Discrete Fourier Transform - 2D

Image f(x,y) x = 0,1,...,N-1 y= 0,1,...,M-1

The Discrete Fourier Transform (DFT) is defined as:

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

u = 0, 1, 2, ..., N-1

v = 0, 1, 2, ..., M-1

Matlab: F=fft2(f);

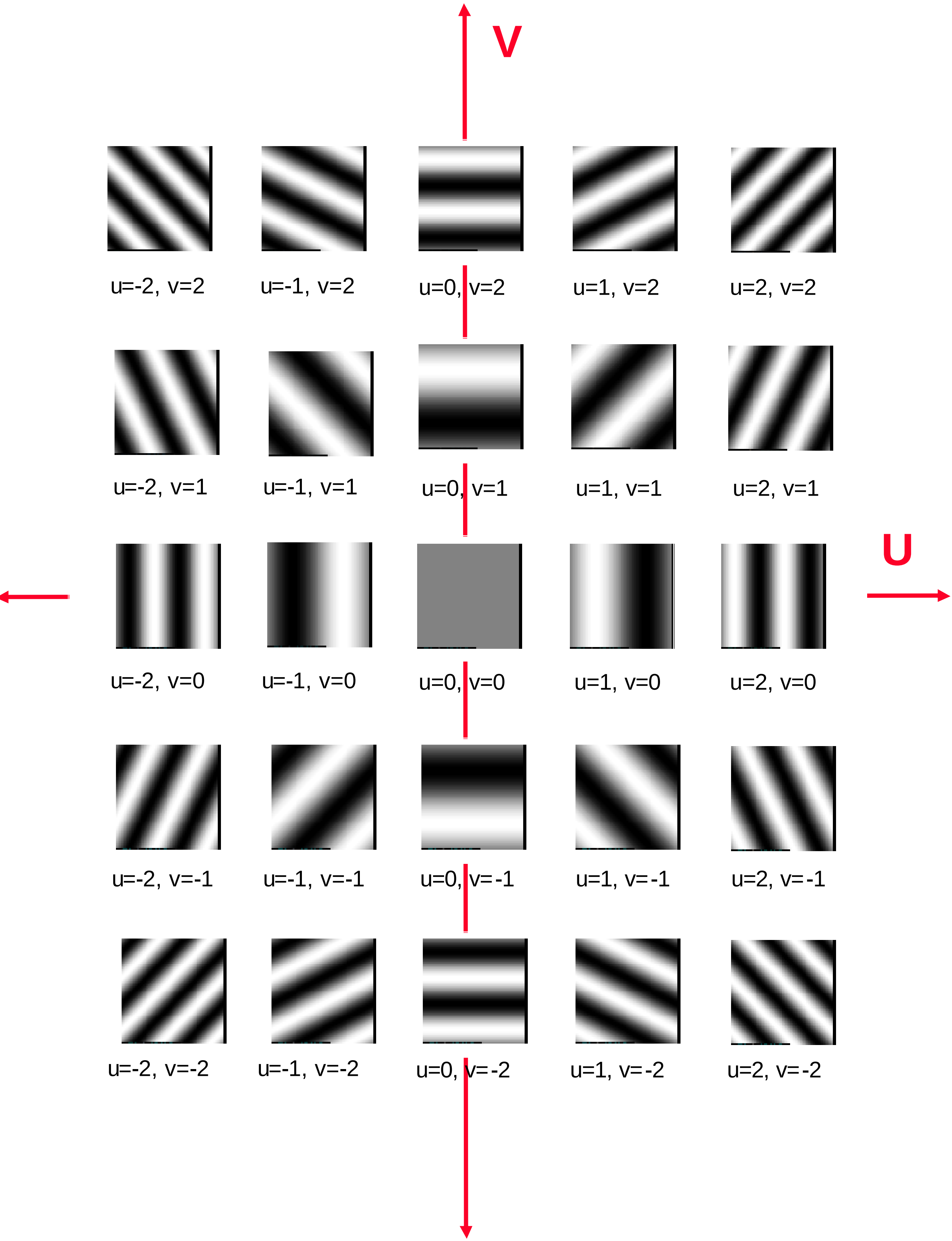
The Inverse Discrete Fourier Transform (IDFT) is defined as:

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

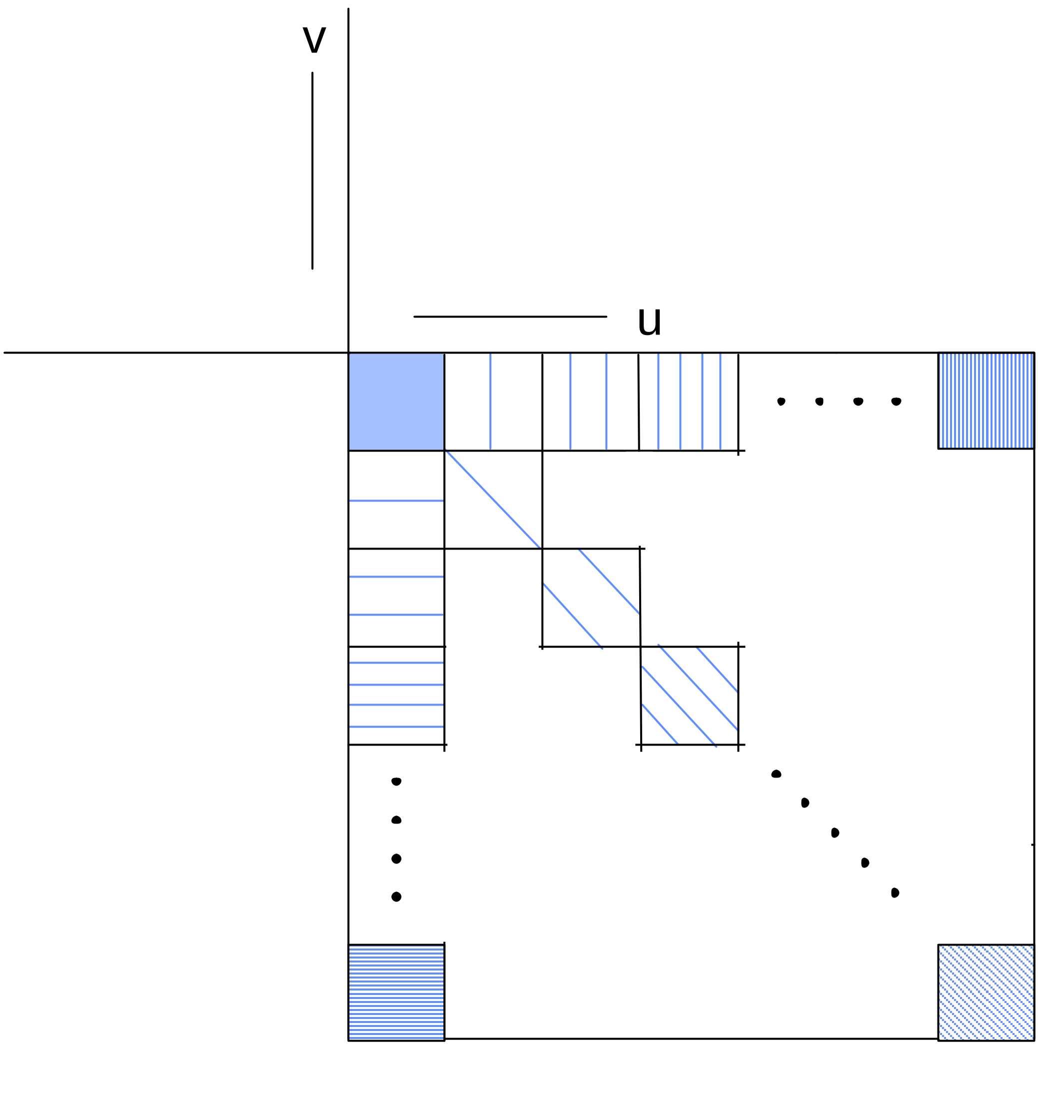
x = 0, 1, 2, ..., N-1

y = 0, 1, 2, ..., M-1

Matlab: F=ifft2(f);


V
U
u=-2, v=2
u=-1, v=2
u=0, v=2
u=1, v=2
u=2, v=2
u=-2, v=1
u=-1, v=1
u=0, v=1
u=1, v=1
u=2, v=1
u=-2, v=0
u=-1, v=0
u=0, v=0
u=1, v=0
u=2, v=0
u=-2, v=-1
u=-1, v=-1
u=0, v=-1
u=1, v=-1
u=2, v=-1
u=-2, v=-2
u=-1, v=-2
u=0, v=-2
u=1, v=-2
u=2, v=-2

Fourier Transform - Image



Visualizing the Fourier Transform Image using Matlab Routines

- F(u,v) is a Fourier transform of f(x,y) and it has complex entries.

$$F = fft2(f);$$

- In order to display the Fourier Spectrum |F(u,v)|
  - Cyclically rotate the image so that F(0,0) is in the center:

$$F = fftshift(F);$$

  - Reduce dynamic range of |F(u,v)| by displaying the log:

$$D = log(1+abs(F));$$

Example:

|F(u)| = 100 4 2 1 0 0 1 2 4

Cyclic |F(u)| = 0 1 2 4 100 4 2 1 0

Display in Range([0..10]):

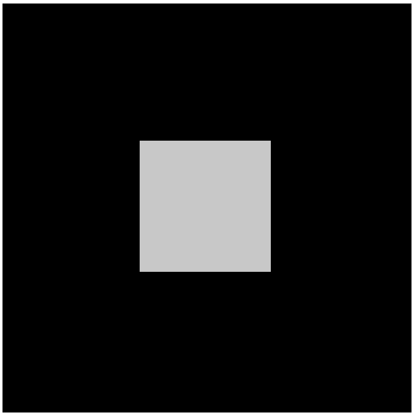
|F(u)|/10 = 0 0 0 0 10 0 0 0 0

log(1+|F(u)|) = 0 0.69 1.01 1.61 4.62 1.61 1.01 0.69 0

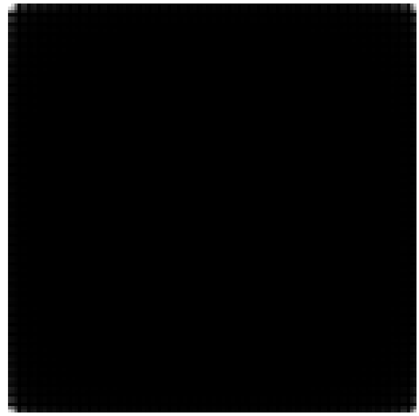
log(1+|F(u)|)/0.462 = 0 1 2 4 10 4 2 1 0

Visualizing the Fourier Image - Example

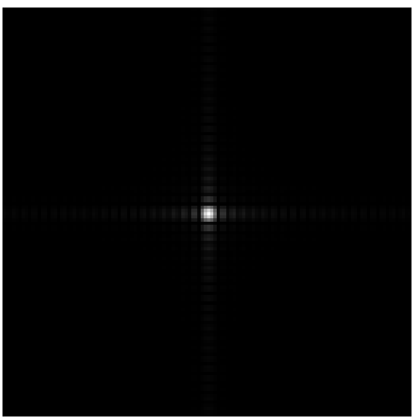
Original



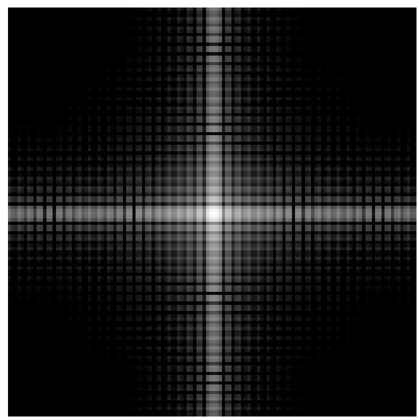
|F(u,v)|



|F(0,0)| at center

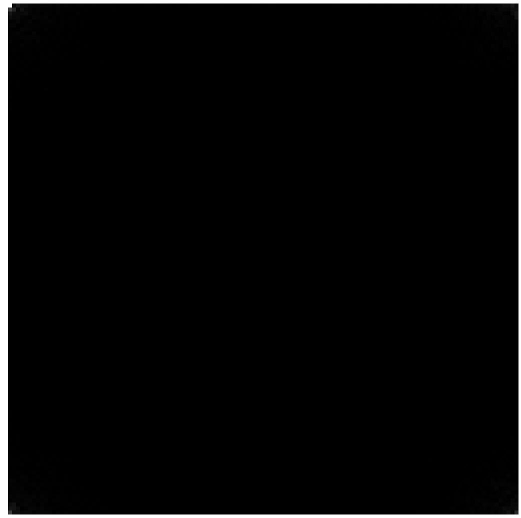


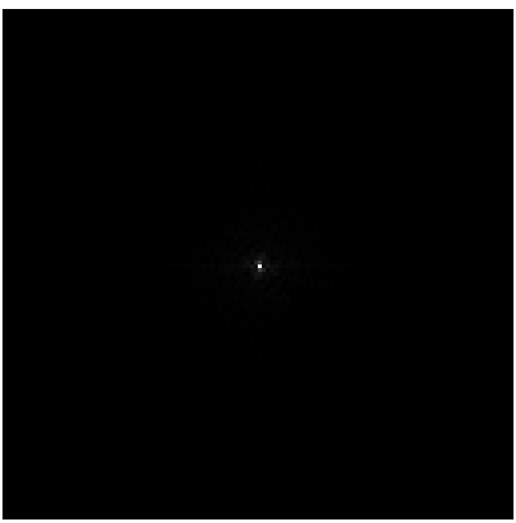
log(1 + |F(u,v)|)



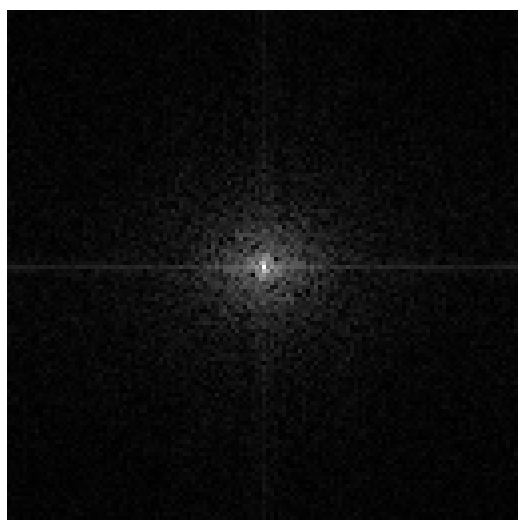
Original

Fourier Image = |F(u,v)|





Shifted Fourier Image



Shifted Log Fourier Image = log(1+ |F(u,v)|)

Properties of The Fourier Transform

Distributive (addition)

$$\tilde{F}[f_1(x,y) + f_2(x,y)] = \tilde{F}[f_1(x,y)] + \tilde{F}[f_2(x,y)]$$

Linearity

$$\tilde{F}[a\, f(x,y)] = a\tilde{F}[f(x,y)]$$

$$a\, f(x,y) \longleftrightarrow a\, F(u,v)$$

Cyclic

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

$$F(x,y) = F(x+N,y+N)$$

Symmetric

if f(x) is real:

$$F(u,v) = F^*(-u,-v)$$

thus:

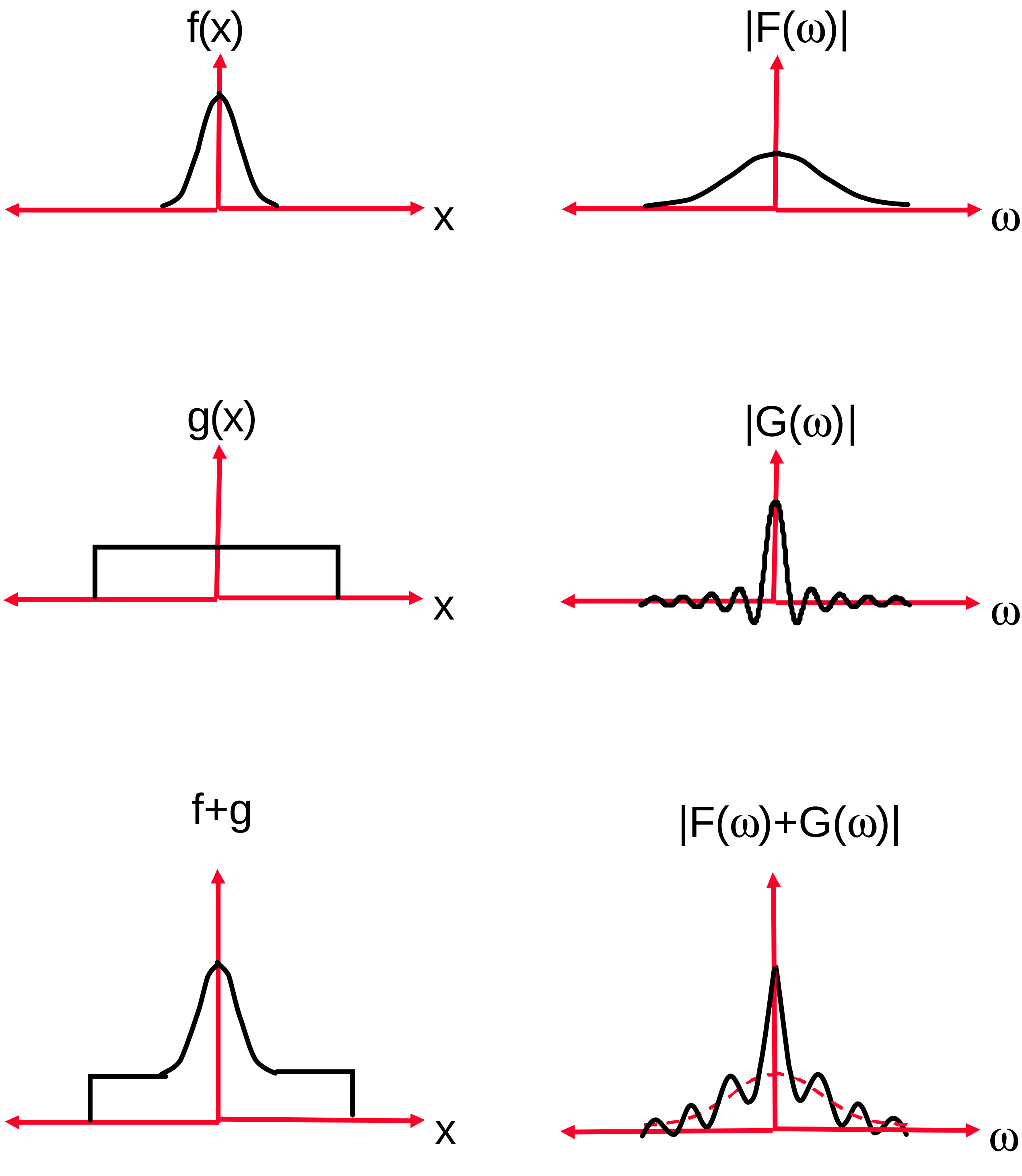
$$|F(u,v)| = |F(-u,-v)|$$

Fourier Spectrum is symmetric

DC (Average)

$$F(0,0) = \frac{1}{N}\frac{1}{M}\sum_{x=0}^{N-1}\sum_{y=0}^{M-1} f(x)\, e^0$$

Distributive: $\tilde{F}\{f+g\}=\tilde{F}\{f\}+\tilde{F}\{g\}$



Cyclic and Symmetry of the Fourier Transform - 1D Example

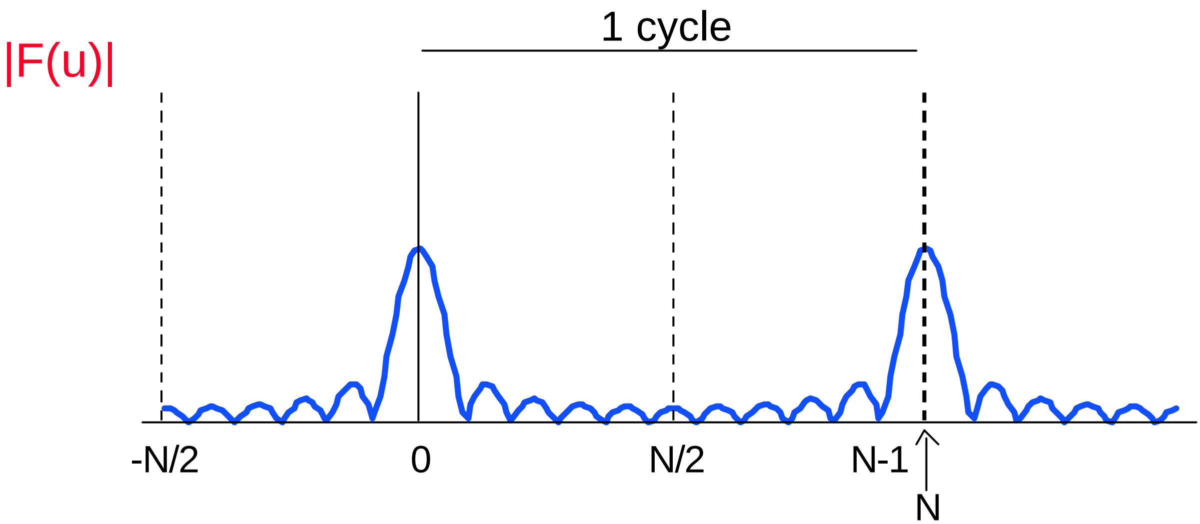


Image Transformations

Translation

$$f(x-x_0,y-y_0) \quad \text{———} \quad F(u,v)e^{\frac{-2\pi i(ux_0+vy_0)}{N}}$$

$$f(x,y)e^{\frac{2\pi i(u_0x+v_0y)}{N}} \quad \text{———} \quad F(u-u_0,v-v_0)$$

The Fourier Spectrum remains unchanged under translation:

$$|F(u,v)| = |F(u,v)e^{\frac{-2\pi i(ux_0+vy_0)}{N}}|$$

Rotation

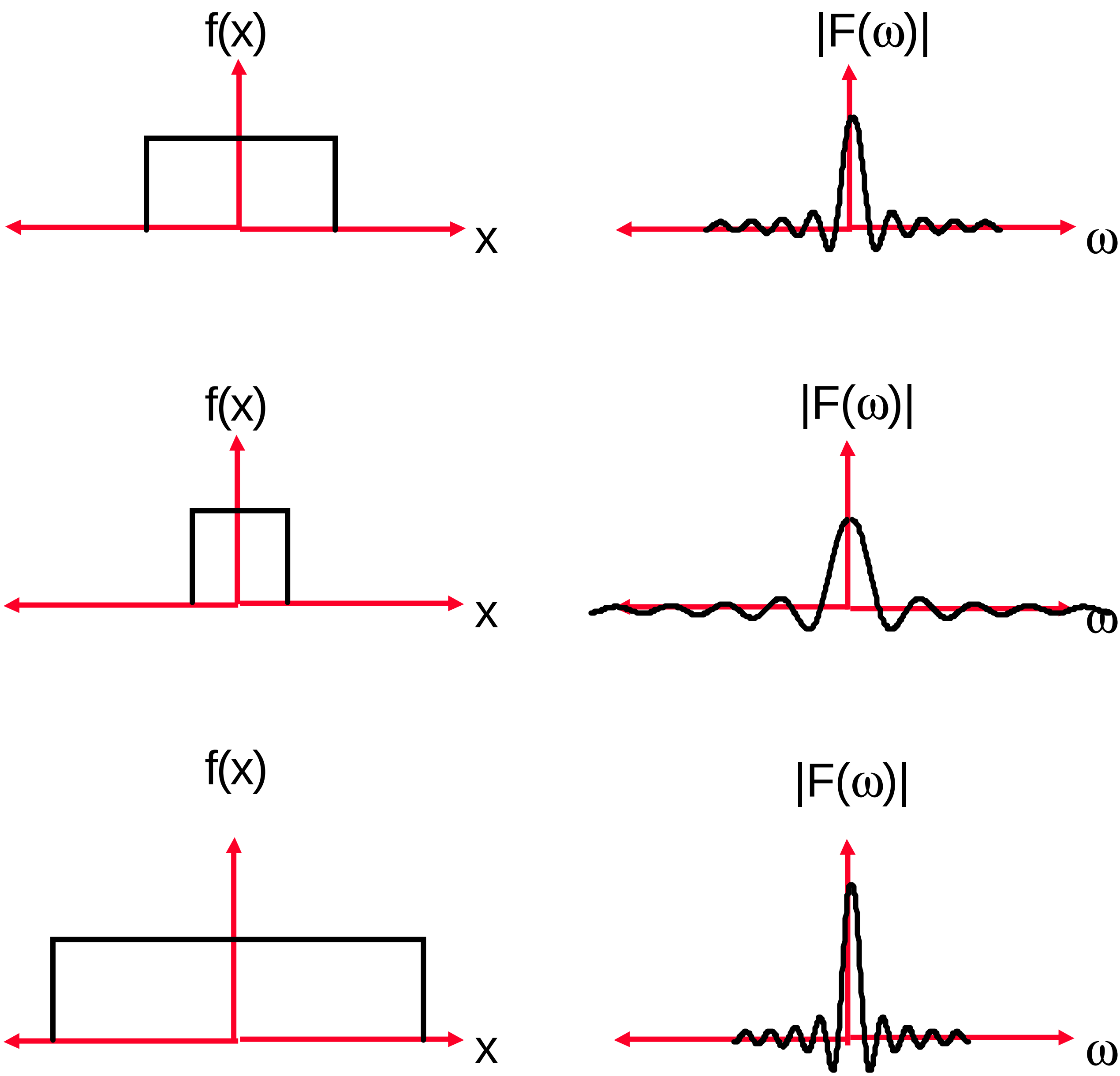
Rotation of f(x,y) by θ ——— Rotation of F(u,v) by θ

Scale

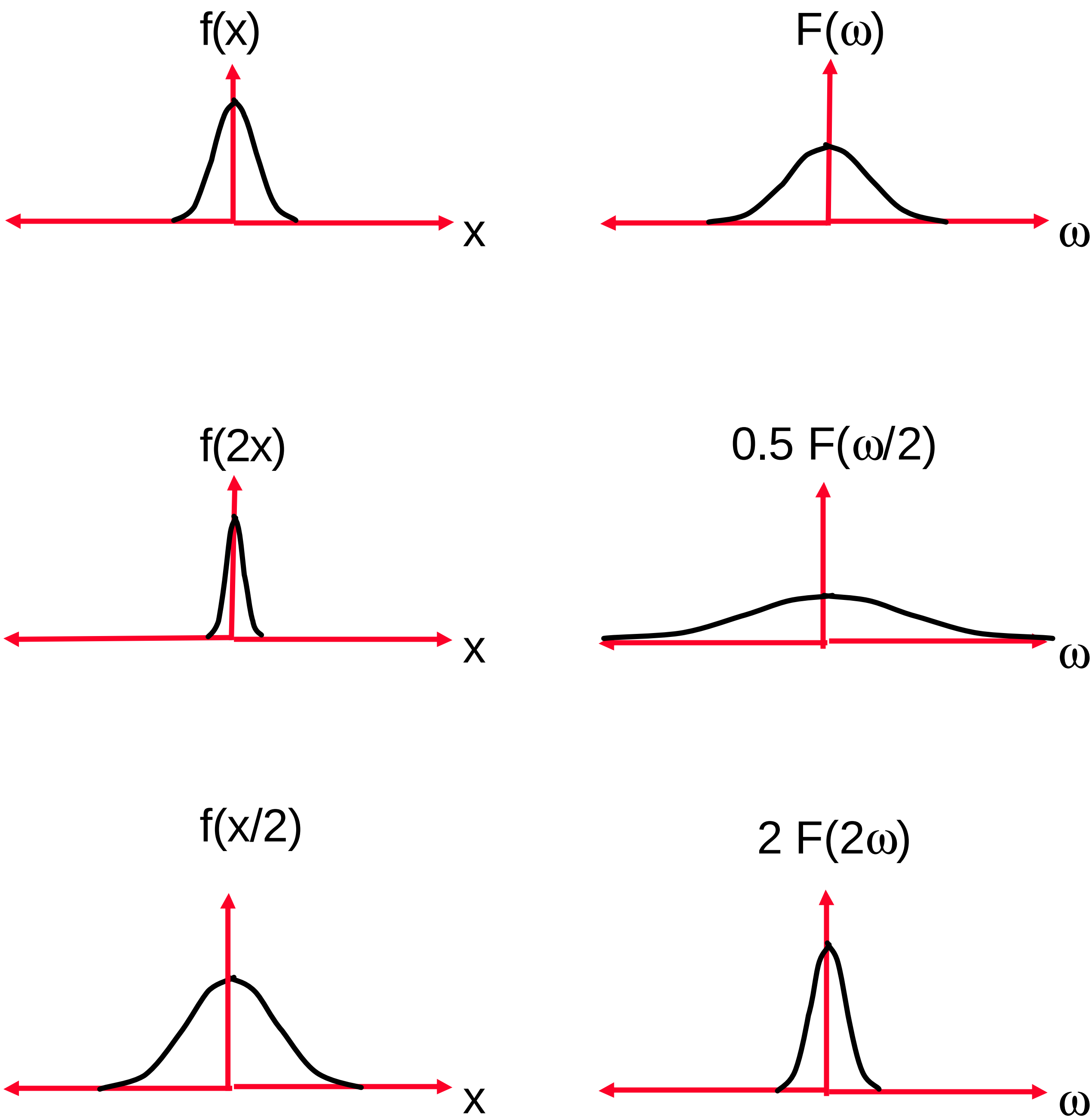
$$f(ax,by) \quad \text{———} \quad \frac{1}{|ab|}F(u/a,v/b)$$

Change of Scale- 1D:

$$\text{if } \tilde{F}\{f(x)\} = F(w) \quad \text{then} \quad \tilde{F}\{f(ax)\} = \frac{1}{|a|} F\left(\frac{w}{a}\right)$$

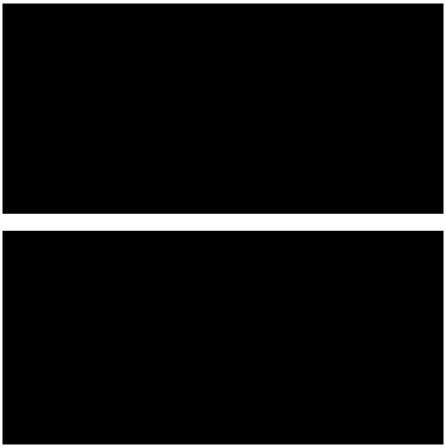


Change of Scale

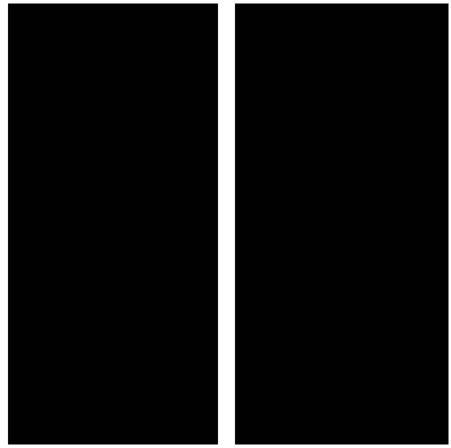


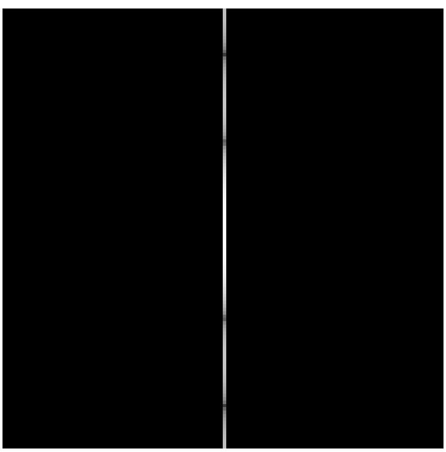
Example - Rotation

2D Image

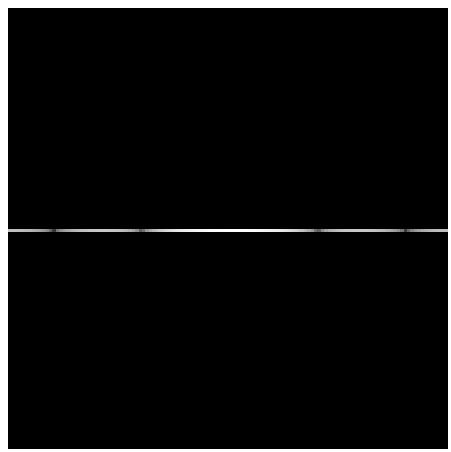


2D Image - Rotated





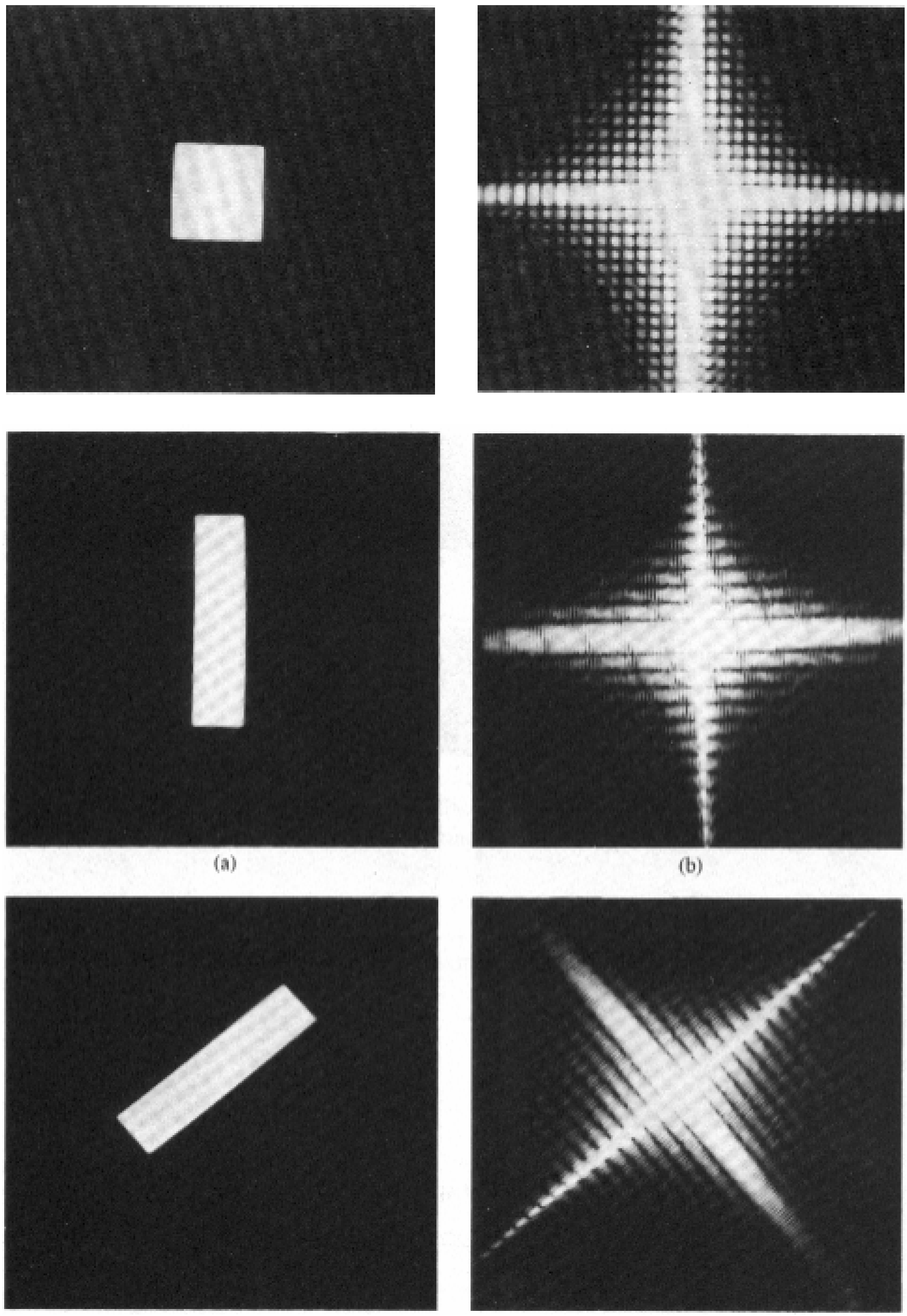
Fourier Spectrum



Fourier Spectrum

Fourier Transform Examples

Image Domain   Frequency Domain



Separabititity

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

$$= \frac{1}{N} \sum_{x=0}^{N-1} \left(\sum_{y=0}^{N-1} f(x,y)\, e^{-2\pi ivy/n} \right) e^{-2\pi iux/n}$$

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

Thus, to perform a 2D Fourier Transform is equivalent to performing 2 1D transforms:

1) Perform 1D transform on EACH column of image f(x,y). Obtain F(x,v).

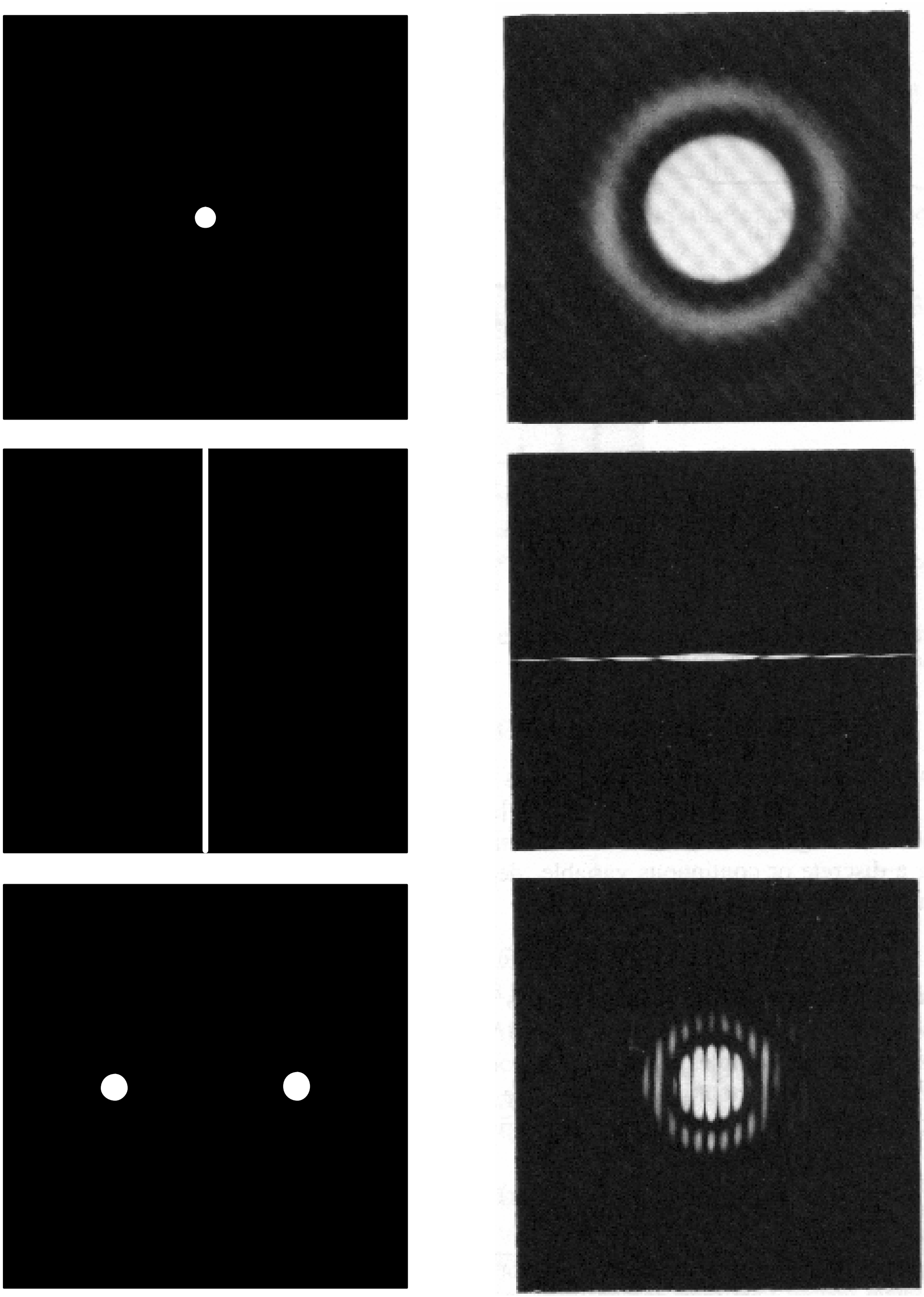
2) Perform 1D transform on EACH row of F(x,v).

Higher Dimensions:

Fourier in any dimension can be performed by applying 1D transform on each dimension.

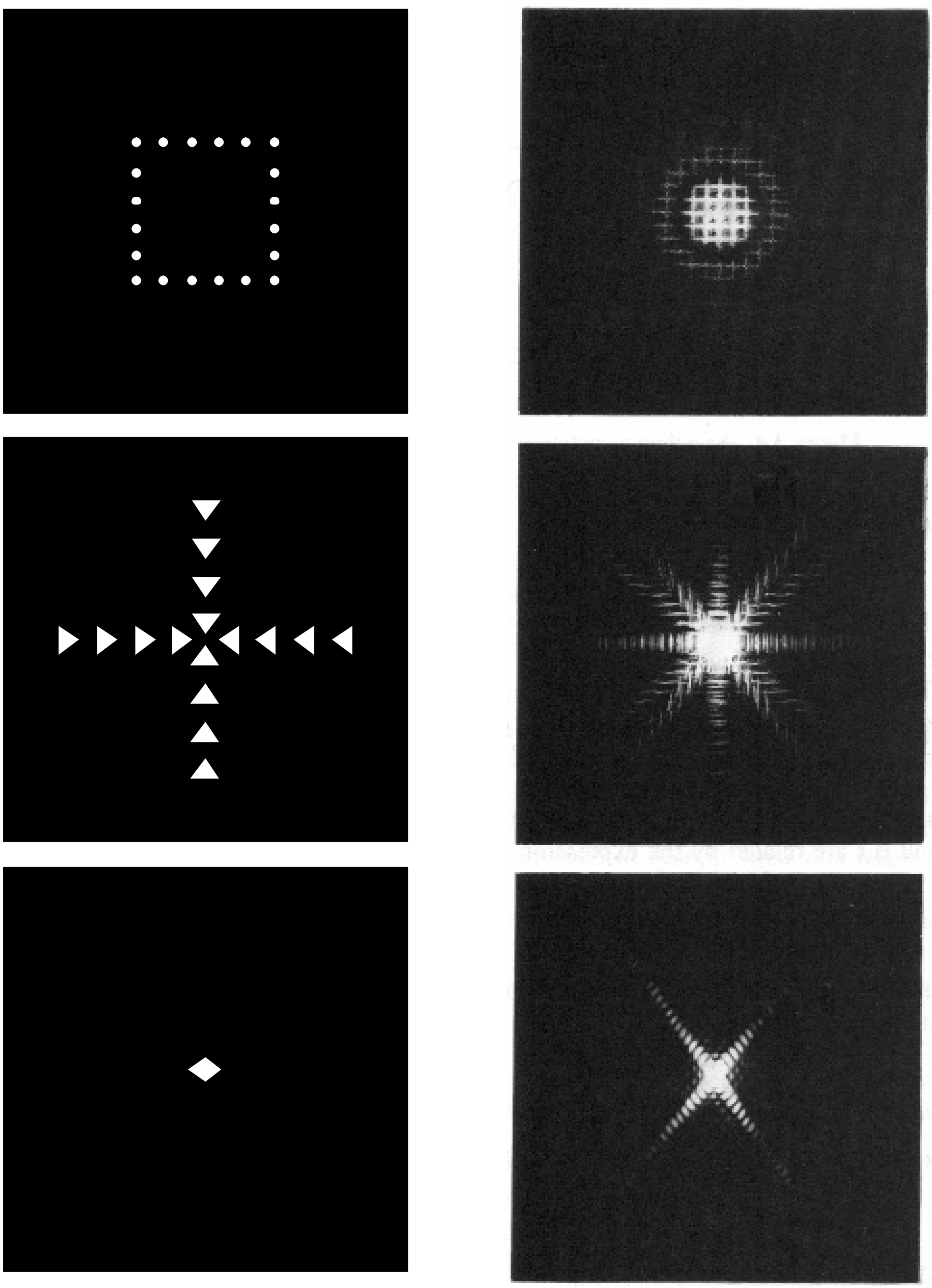
Fourier Transform Examples

Image Domain | Frequency Domain



Fourier Transform Examples

Image Domain   Frequency Domain



Linear Systems and Responses

	Spatial Domain	Frequency Domain
Input	f	F
Output	g	G
Impulse Response	h	
Freq. Response		H
Relationship	g=f*h	G=FH

The Convolution Theorem

$g = f * h$	$g = f\,h$
implies	implies
$G = F\,H$	$G = F * H$

Convolution in one domain is multiplication in the other and vice versa

The Convolution Theorem

$$\tilde{F}\{f(x) * g(x)\} = \tilde{F}\{f(x)\}\tilde{F}\{g(x)\}$$

and likewise

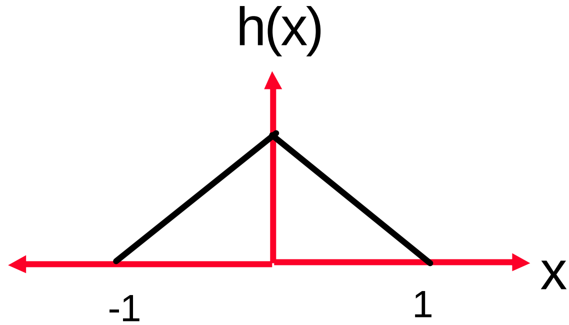
$$\tilde{F}\{f(x)g(x)\} = \tilde{F}\{f(x)\} * \tilde{F}\{g(x)\}$$

f(x,y) * g(x,y) ——— F(u,v) G(u,v)

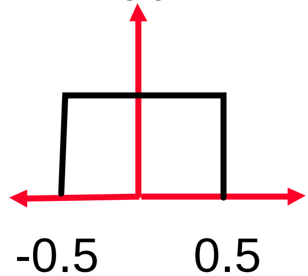
f(x,y) g(x,y) ——— F(u,v) * G(u,v)

Convolution in one domain is multiplication in the other and vice versa

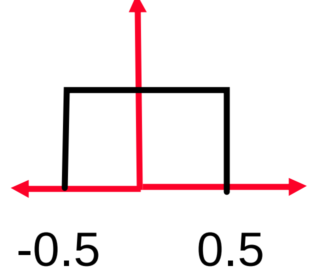
Example:



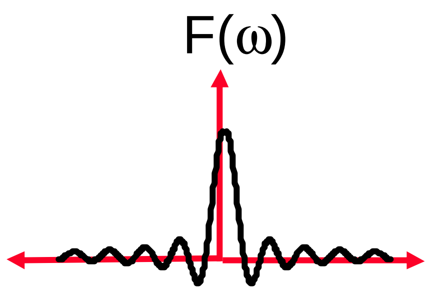
h(x) =



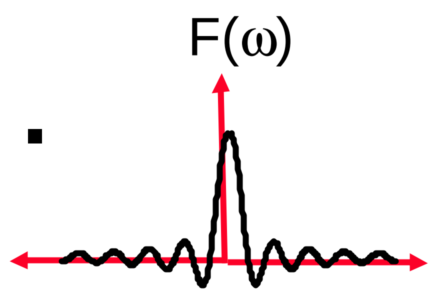
*



H(w) =

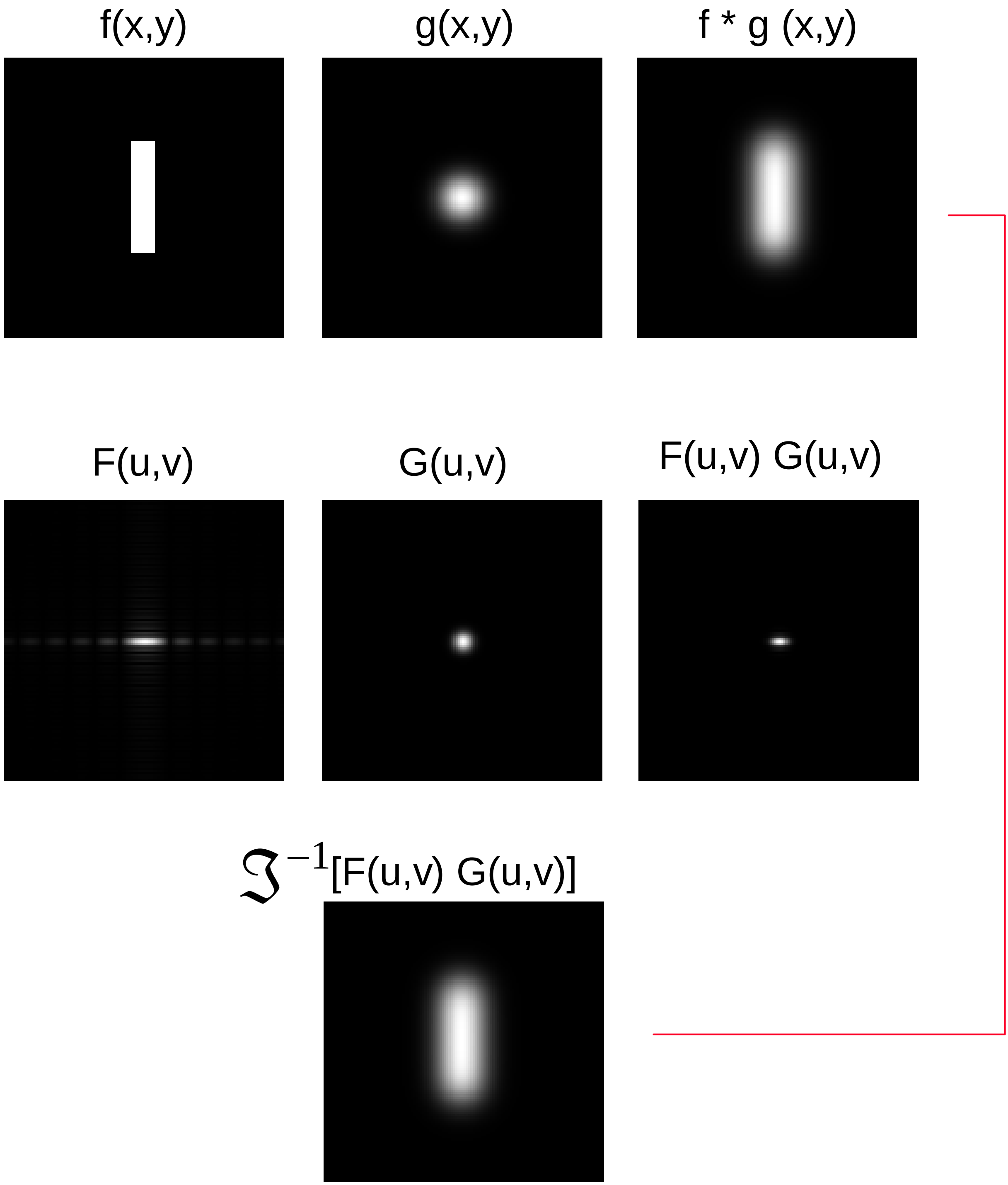


·

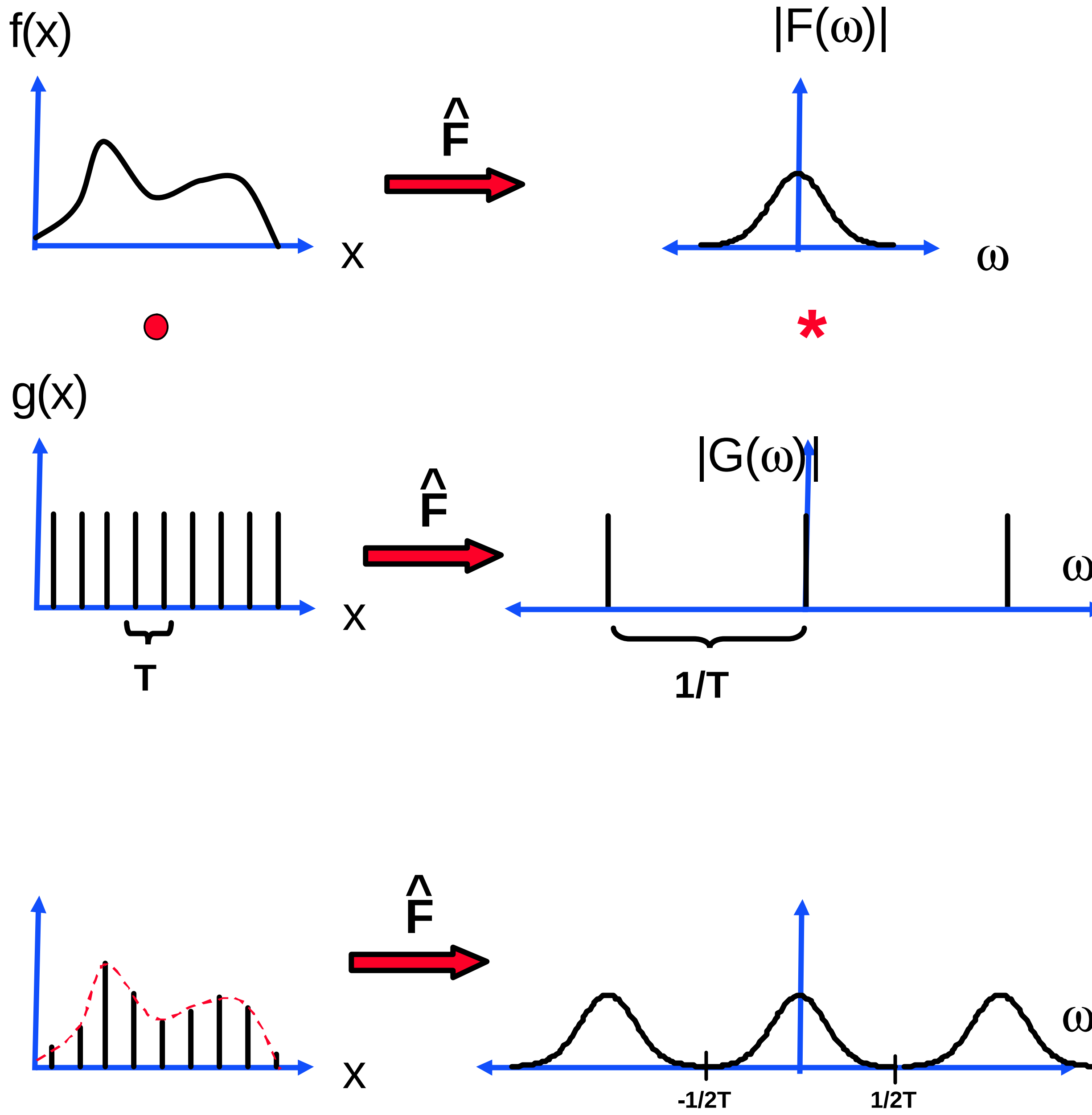


=

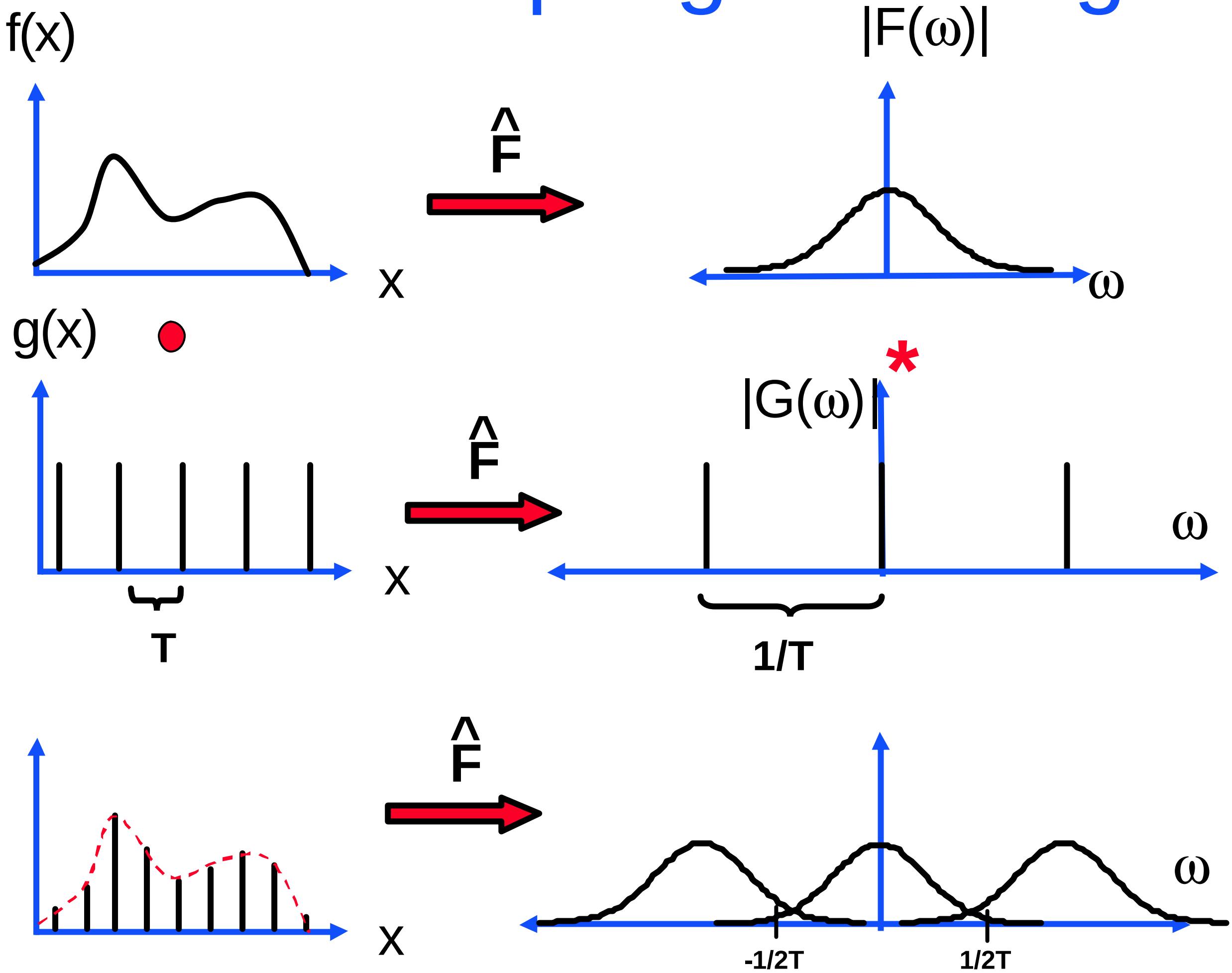
Convolution Theorem - 2D Example



Sampling the Image



Undersampling the Image



Critical Sampling

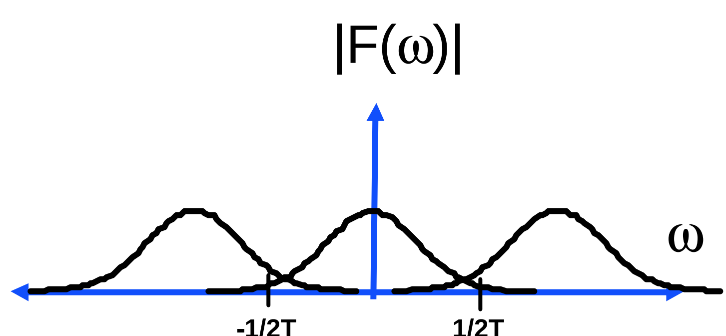
- If the maximal frequency of f(x) is ω_{max} , it is clear from the above replicas that ω_{max} should be smaller that 1/2T.

- Alternatively:

$$\frac{1}{T} > 2w_{\max}$$

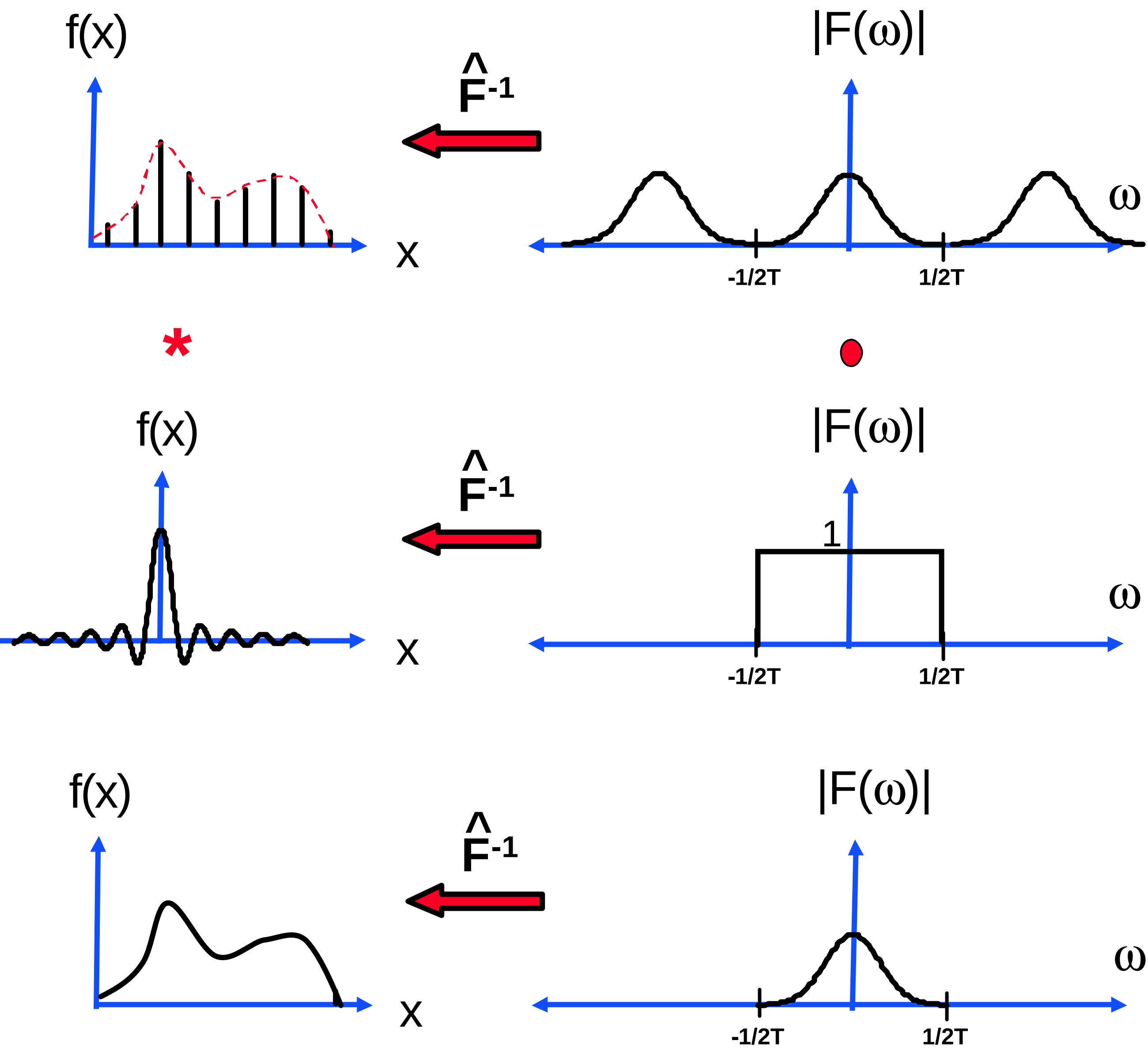
- **<u>Nyquist Theorem</u>**: If the maximal frequency of f(x) is ω_{max} the sampling rate should be larger than $2\omega_{max}$ in order to fully reconstruct f(x) from its samples.

- If the sampling rate is smaller than $2\omega_{max}$ overlapping replicas produce **aliasing**.



Optimal Interpolation

- It is possible to fully reconstruct f(x) from its samples:



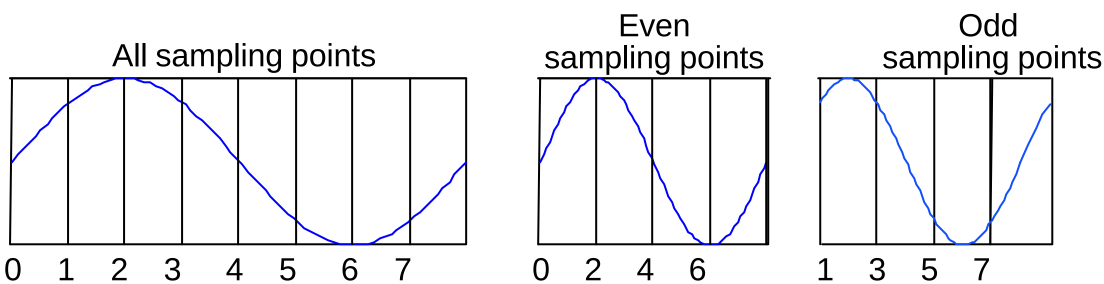
Fast Fourier Transform - FFT

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x) e^{\frac{-2piux}{N}} \qquad u = 0, 1, 2, ..., N-1$$

$O(n^2)$ operations

$$F(u) = \underbrace{\frac{1}{N}\sum_{x=0}^{N/2-1} f(2x) e^{\frac{-2piu2x}{N}}}_{\text{even x}} + \underbrace{\frac{1}{N}\sum_{x=0}^{N/2-1} f(2x+1) e^{\frac{-2piu(2x+1)}{N}}}_{\text{odd x}}$$

$$= \frac{1}{2}\left[\underbrace{\frac{1}{N/2}\sum_{x=0}^{N/2-1} f(2x) e^{\frac{-2piux}{N/2}}}_{\text{Fourier Transform of of N/2 even points}} + e^{\frac{-2piu}{N}} \underbrace{\frac{1}{N/2}\sum_{x=0}^{N/2-1} f(2x+1) e^{\frac{-2piux}{N/2}}}_{\text{Fourier Transform of of N/2 odd points}} \right]$$



The Fourier transform of N inputs, can be performed as 2 Fourier Transforms of N/2 inputs each + one complex multiplication and addition for each value i.e. O(N).

Note, that only N/2 different transform values are obtained for the N/2 point transforms.

$$F_N(u)=\frac{1}{2}\left[\frac{1}{N/2}\sum_{x=0}^{N/2-1} f(2x)e^{\frac{-2piux}{N/2}} + e^{\frac{-2piu}{N}}\frac{1}{N/2}\sum_{x=0}^{N/2-1} f(2x+1)e^{\frac{-2piux}{N/2}}\right.$$

$$F_N(u)=\frac{1}{2}\left[F^e_{N/2}(u)+e^{\frac{-2piu}{N}}F^o_{N/2}(u)\right]$$

For u' = u+N/2 : $e^{\frac{-2piu'}{N}} = e^{\frac{-2pi(u+N/2)}{N}} = e^{\frac{-2piu}{N}}e^{-pi} = -e^{\frac{-2piu}{N}}$

obtain :

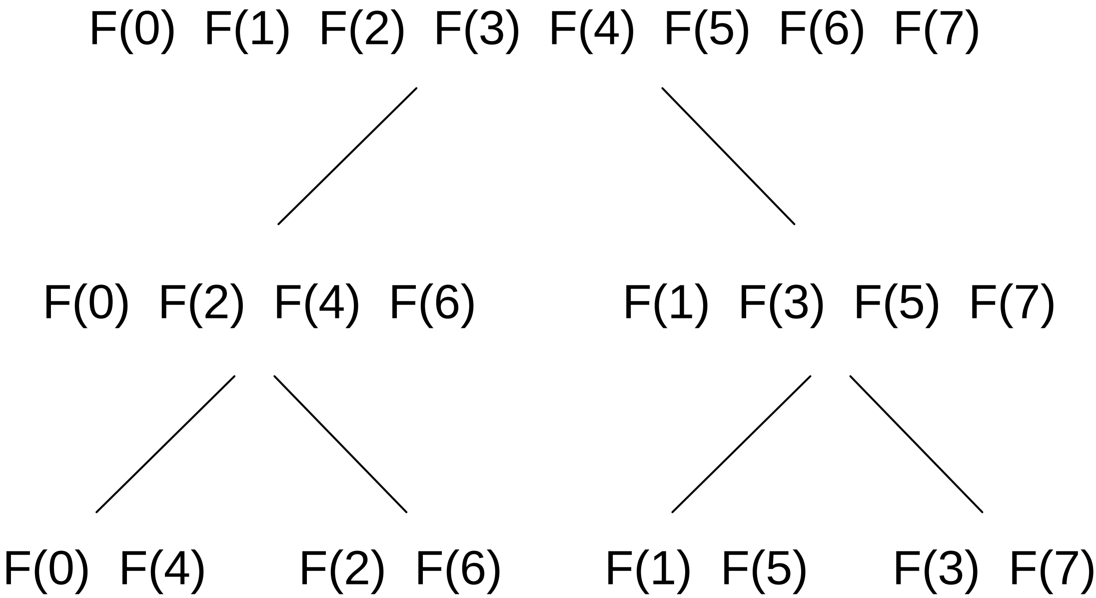
$$F_N(u)=\frac{1}{2}\left[F^e_{N/2}(u)+e^{\frac{-2piu}{N}}F^o_{N/2}(u)\right]$$

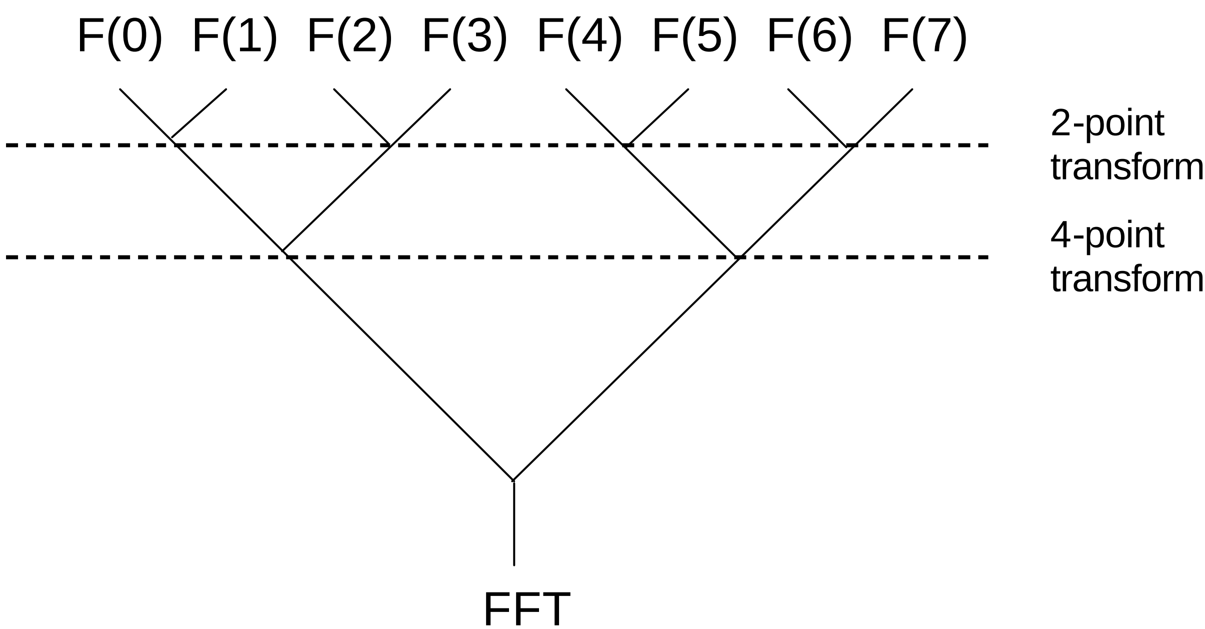
For
u = 0, 1, 2, ..., N/2-1

$$F_N\left(u+\frac{N}{2}\right)=\frac{1}{2}\left[F^e_{N/2}(u)-e^{\frac{-2piu}{N}}F^o_{N/2}(u)\right]$$

Thus: only one complex multiplication is needed for two terms.

Calculating $F^e_{N/2}(u)$ and $F^o_{N/2}(u)$ is done recursively by calculating $F^e_{N/4}(u)$ and $F^o_{N/4}(u)$.





FFT : O(nlog(n)) operations

FFT of NxN Image: $O(n^2 \log(n))$ operations

Frequency Enhancement