

COMS30030 - Image Processing and Computer Vision



Lecture 03

Frequency Domain & Transforms

Majid Mirmehdi | majid@cs.bris.ac.uk

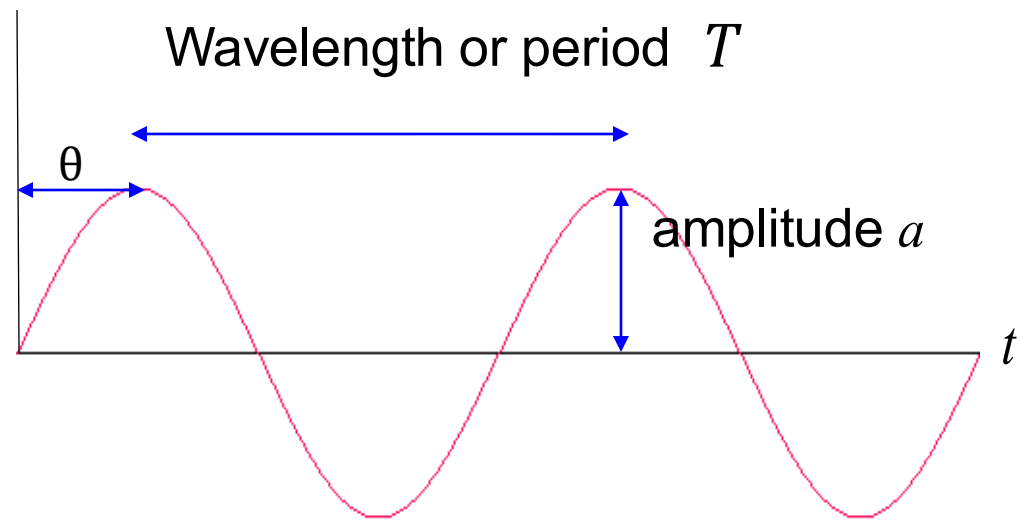
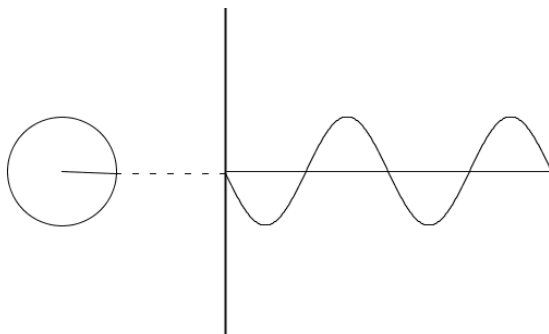
Signals as Functions

Frequency - allows us to characterise signals:

- Repeats over regular intervals with Frequency $u = \frac{1}{T}$ cycles/sec (Hz)
- Amplitude a (peak value)
- the Phase θ (shift in degrees)

Example: sine function

$$f(t) = a \sin 2\pi ut$$



Fourier's Theorem

$$f(x) = \int a_n \cos(nx) + b_n \sin(nx) \delta n$$

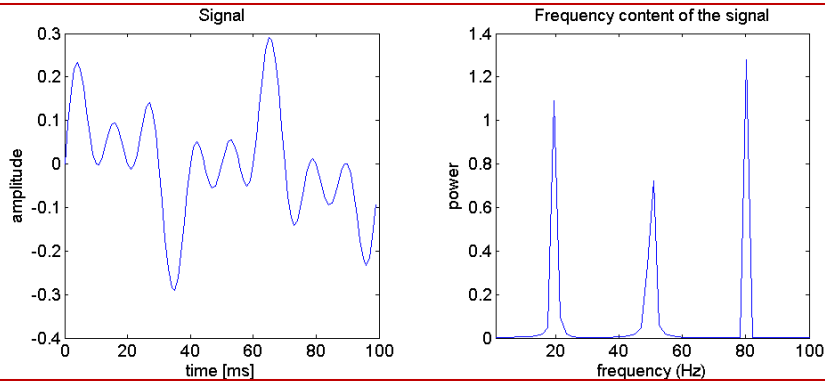


Jean-Baptiste Joseph Fourier

- The sines and cosines are the **Basis Functions** of this representation. a_n and b_n are the **Fourier Coefficients**.
- The sinusoids are harmonically related: each one's frequency is an integer multiple of the fundamental frequency of the input signal.

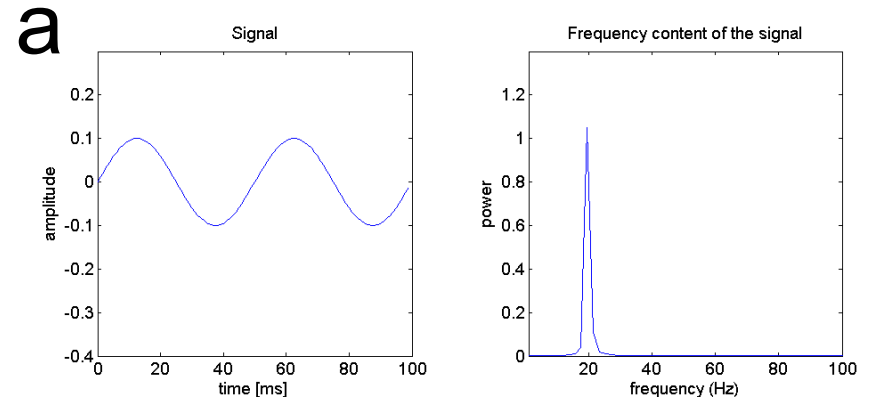
Intuition I: Simple 1D example

$$d = a + b + c$$



time domain

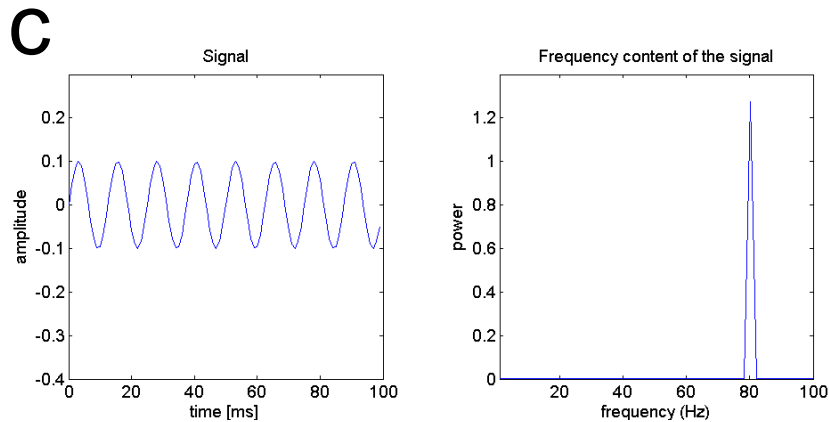
frequency domain



a

time domain

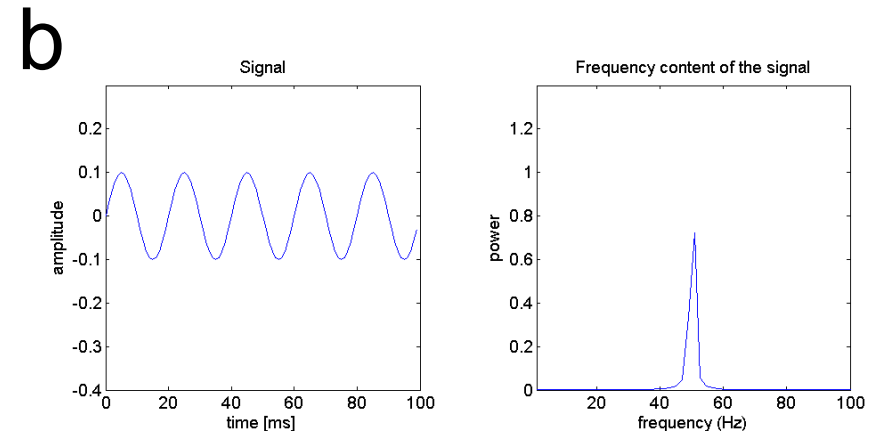
frequency domain



c

time domain

frequency domain



b

time domain

frequency domain

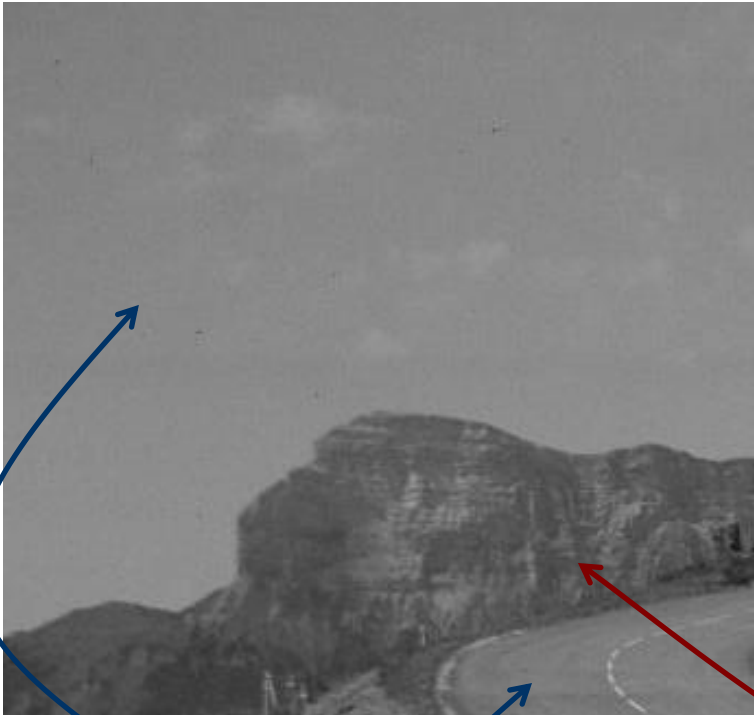
Intuition II: Simple 1D example



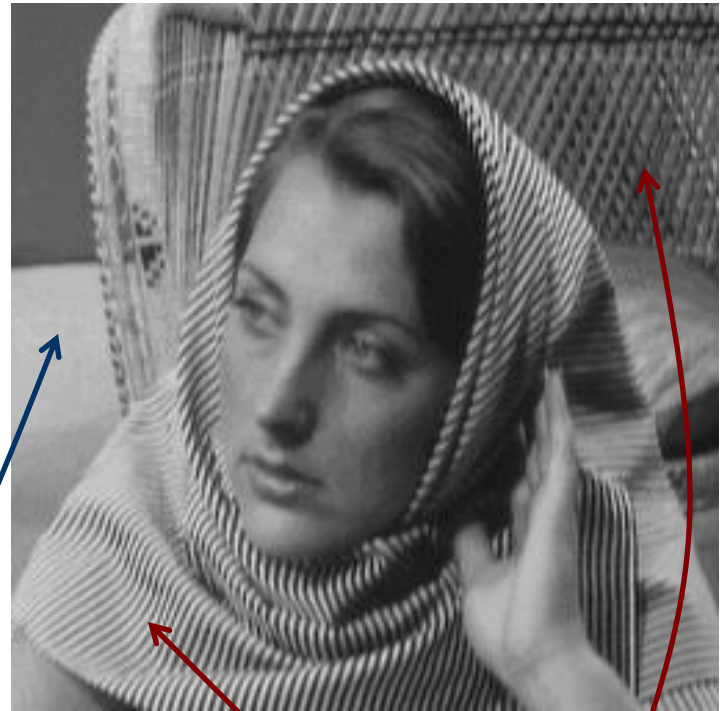
Animation by Lucas V Barbosa

Intuition III: Concept of Frequency in Images

Rate of change of intensity along the two dimensions



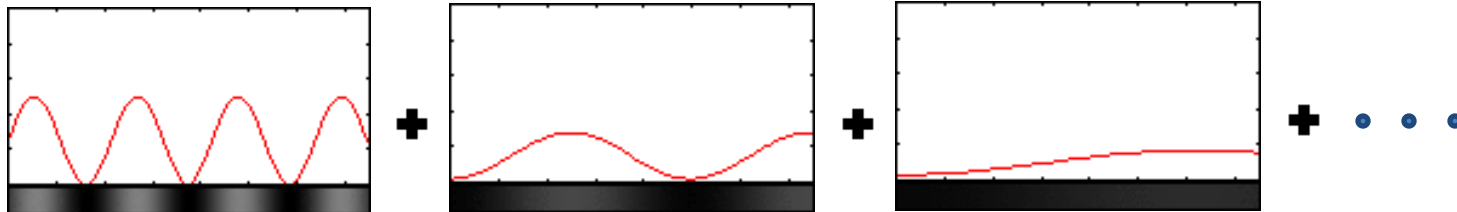
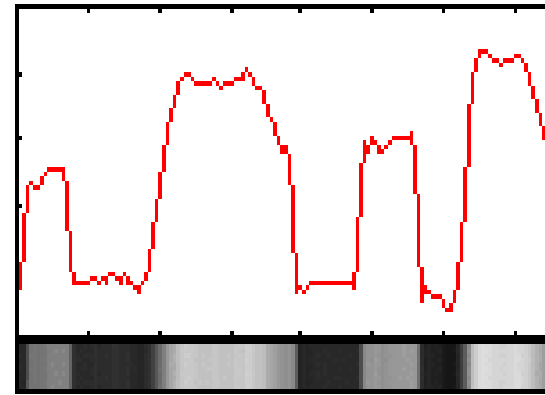
Slowly changing → low frequency



Rapidly changing → high frequency

Intuition IV: Images as waves!?

Take a single row or column of pixels from an image \rightarrow a 1D signal



From ImageNagik

2D Fourier Transform: Continuous Form

- The Fourier Transform of a continuous function of two variables $f(x,y)$ is:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i2\pi(ux+vy)} dx dy$$

- Conversely, given $F(u,v)$, we can obtain $f(x,y)$ by means of the *inverse* Fourier Transform:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{i2\pi(ux+vy)} du dv$$

These two equations are also known as the Fourier Transform Pair.

Note, they constitute a lossless representation of data.

2D Fourier Transform: Discrete Form

- The FT of a discrete function of two variables, $f(x,y)$, is:

$$F(u, v) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \overset{\text{image}}{f(x, y)} \overset{\text{kernels (probing functions)}}{e^{-i2\pi(\frac{ux+vy}{N})}} \quad \text{for } u, v = 0, 1, 2, \dots, N-1.$$

- Conversely, given $F(u,v)$, we can obtain $f(x,y)$ by means of the *inverse FT*:

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) e^{i2\pi(\frac{ux+vy}{N})} \quad \text{for } x, y = 0, 1, 2, \dots, N-1.$$

These two equations are also known as the Fourier Transform Pair.

Note, they constitute a lossless representation of data.

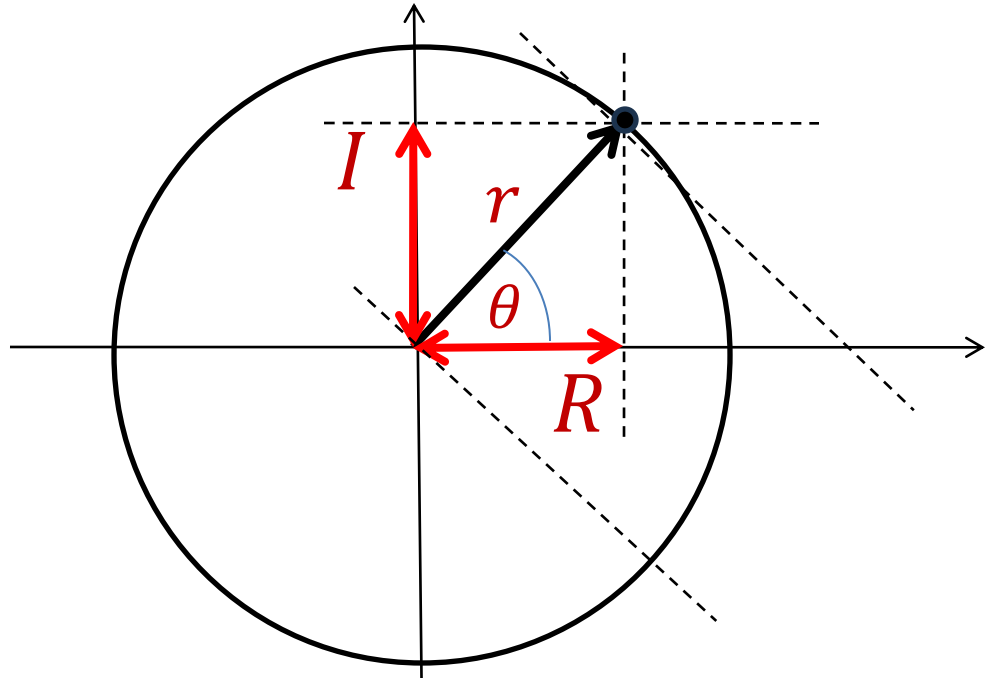
Euler's Formula

$$e^{i2\pi(\frac{ux+vy}{N})}$$



$$e^{i\theta} = \cos \theta + i \sin \theta$$

The kernel is associated with a complex number (r, θ) in polar coordinates or $R(u, v), I(u, v)$ in standard complex notation.



2D Fourier Transforms

- Euler's Formula:

$$e^{i\theta} = \cos \theta + i \sin \theta$$

- Thus, each term of the Fourier Transform is composed of the sum of all values of the image function $f(x,y)$ multiplied by a particular kernel at a particular frequency and orientation specified by (u,v) :

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

for $u, v = 0, 1, 2, \dots, N - 1$.

All kernels together form a new orthogonal basis for our image.

2D Fourier Transforms

- Euler's Formula:

$$e^{i\theta} = \cos \theta + i \sin \theta$$

- Thus, each term of the Fourier Transform is composed of the sum of all values of the image function $f(x,y)$ multiplied by a particular kernel at a particular frequency and orientation specified by (u,v) :

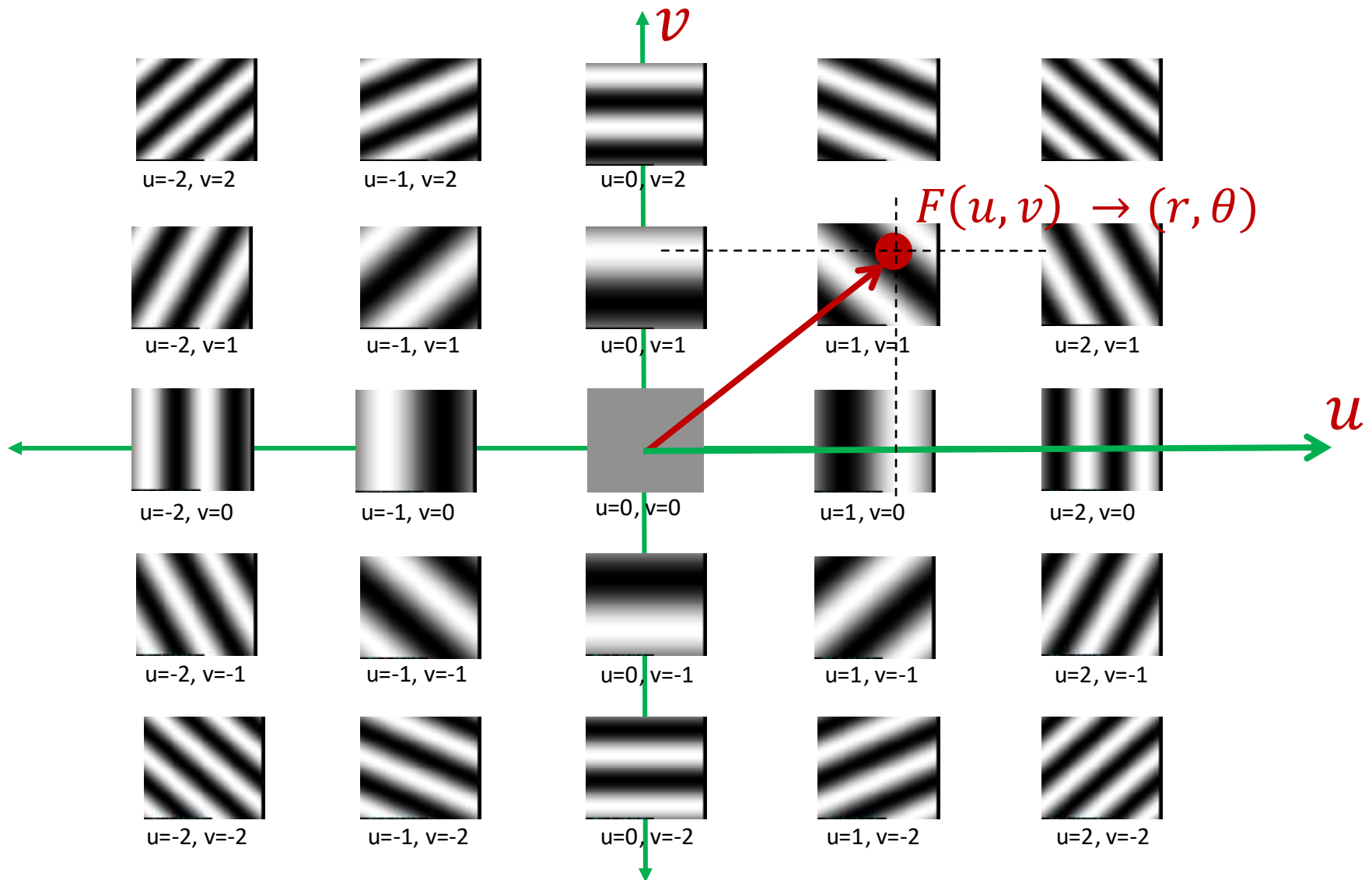
$$F(u, v) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \left[\overset{\mathbf{1}}{\cos\left(\frac{2\pi(ux + vy)}{N}\right)} - i \overset{\mathbf{0}}{\sin\left(\frac{2\pi(ux + vy)}{N}\right)} \right]$$

for $u, v = 0, 1, 2, \dots, N - 1$.

The slowest varying frequency component, i.e.
when $u=0, v=0 \rightarrow$ average image graylevel

All kernels together form a new orthogonal basis for our image.

'Fabric' of the 2D Fourier Space (as kernels)



Power Spectrum and Phase Spectrum

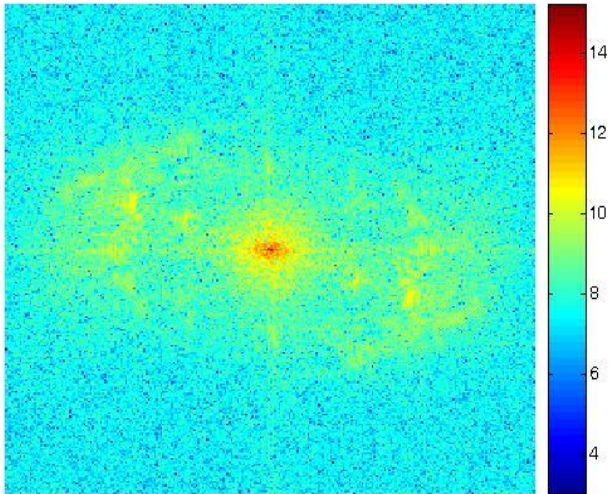
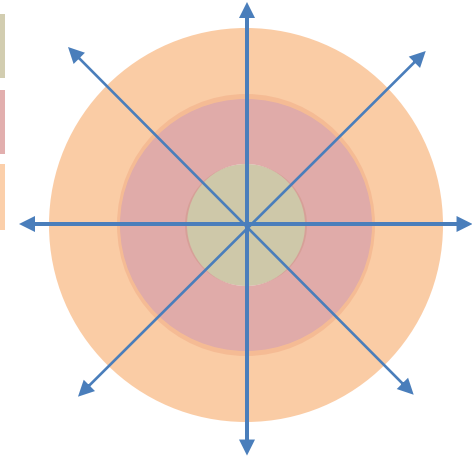
$f(x, y)$



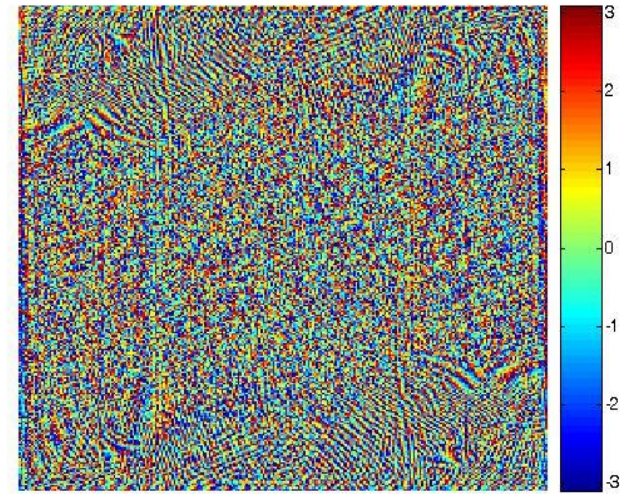
Low to Low-ish frequencies

Mid-range frequencies

High frequencies



$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)}$$

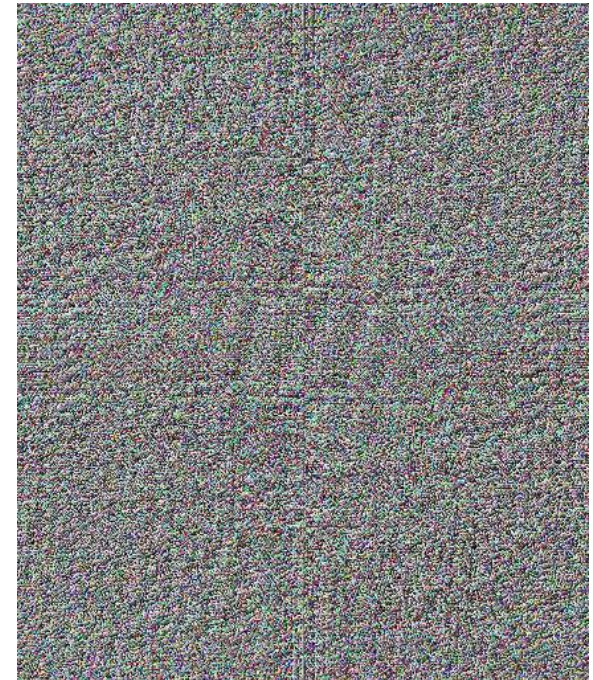
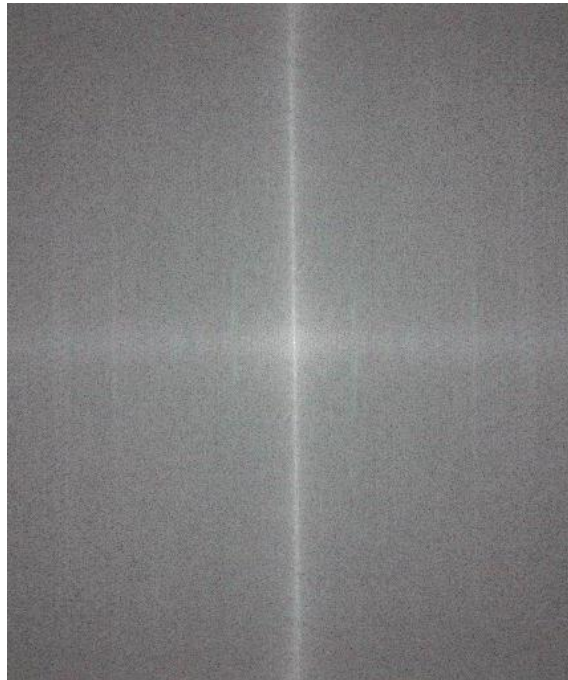


$$\theta(u, v) = \tan^{-1} [I(u, v)/R(u, v)]$$

Example: Relating Frequencies to Images

Magnitude

Phase



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

$$\angle F(I)$$

The Frequency Domain

- $F(u, v)$ is a complex number and has real and imaginary parts:

$$F(u, v) = R(u, v) + iI(u, v)$$

- Magnitudes
(forming the Magnitude Spectrum):

$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)}$$

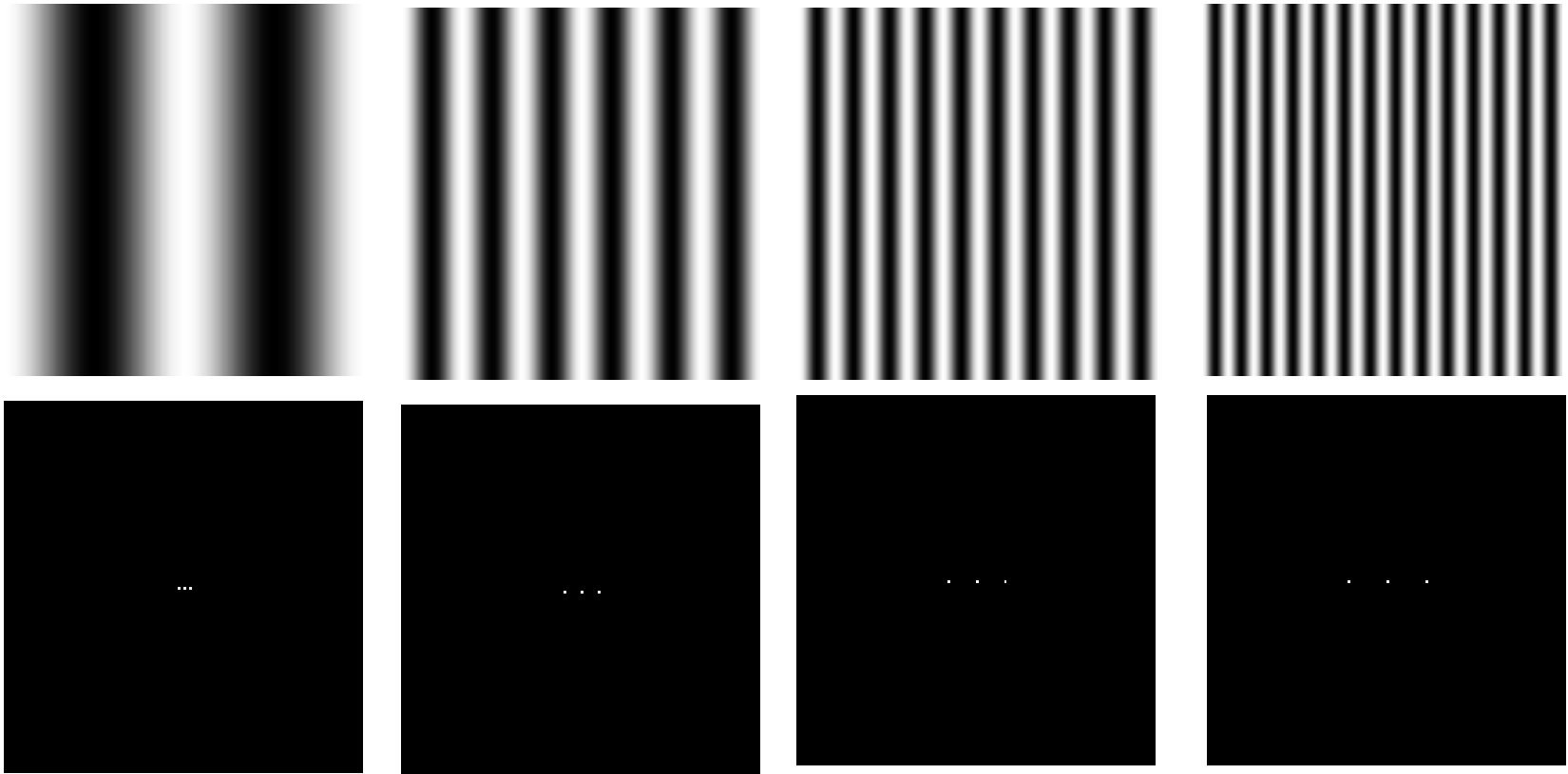
- Phase Angles
(forming the Phase Spectrum):

$$\theta(u, v) = \tan^{-1} [I(u, v)/R(u, v)]$$

- Expressing $F(u, v)$ in polar coordinates (r, θ) :

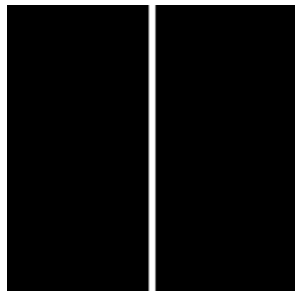
$$F(u, v) = |F(u, v)|e^{i\theta(u, v)} = re^{i\theta}$$

Spatial Domain \longleftrightarrow Frequency Domain



Perpendicular relationship

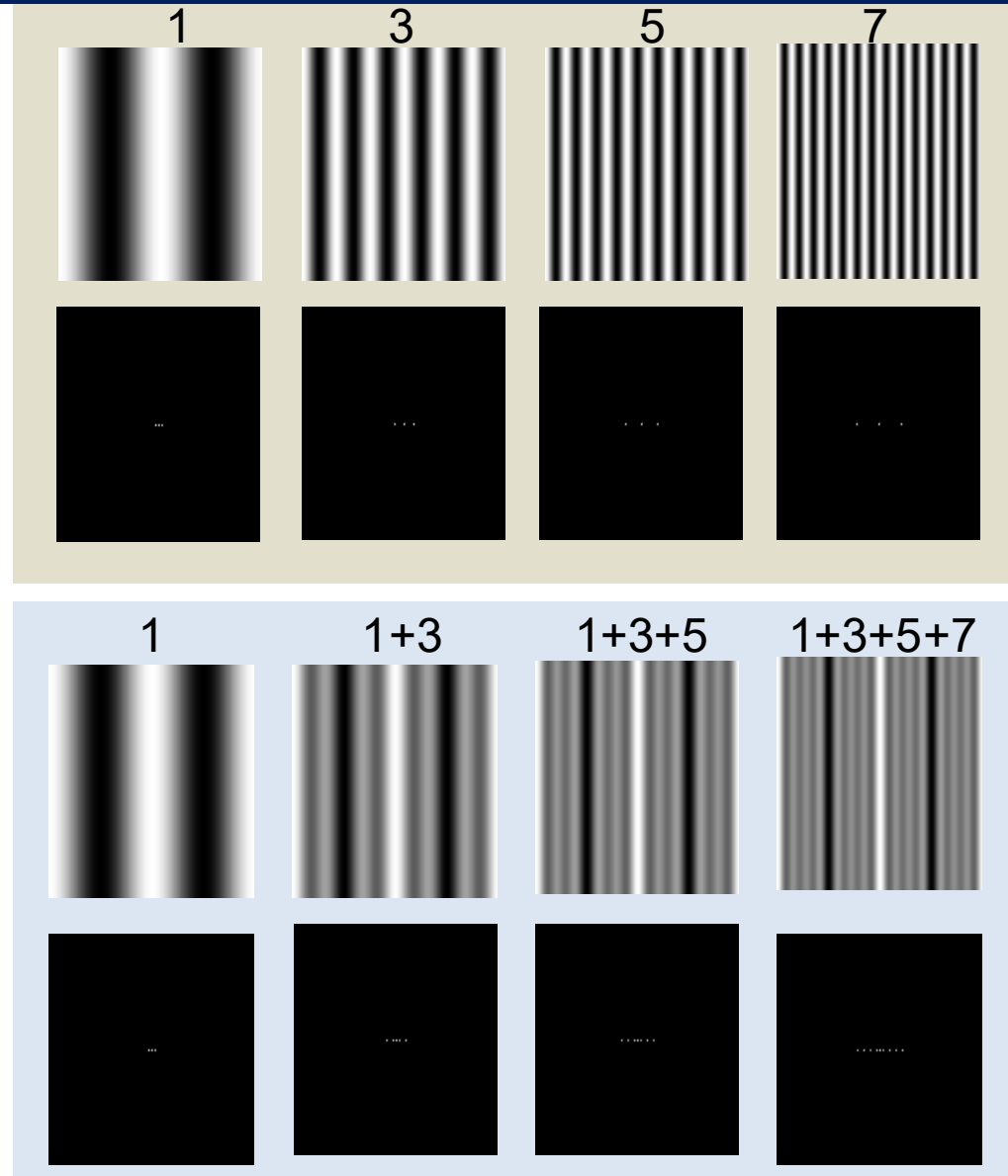
Ideal edge and line structures have a concentration along a line passing through the origin in the frequency domain and in a **direction perpendicular to their orientation** - they are 'constructed' by adding together all 2D sinusoidal waves that 'travel' perpendicular to the edge or line.



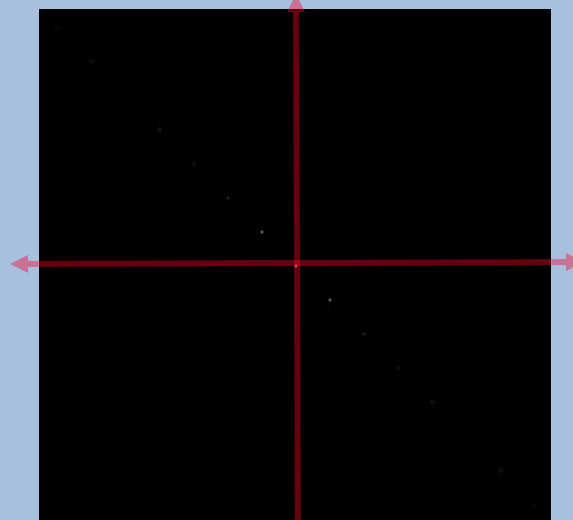
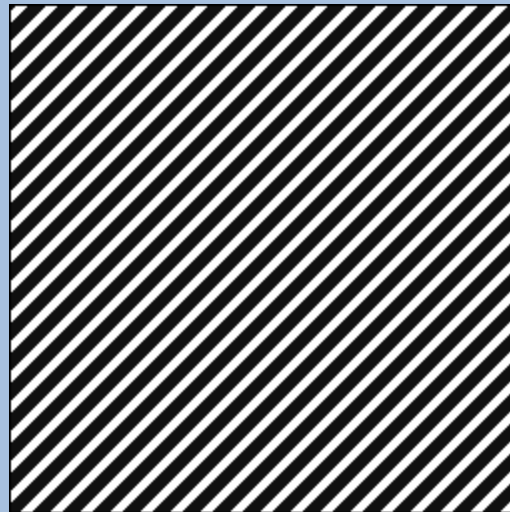
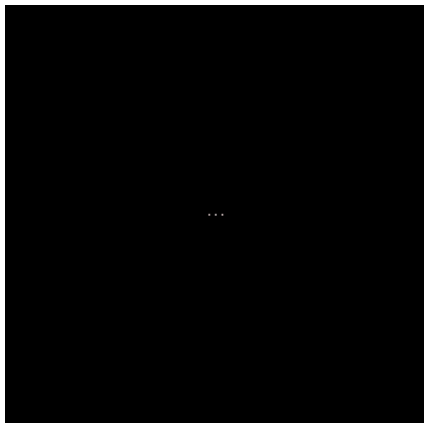
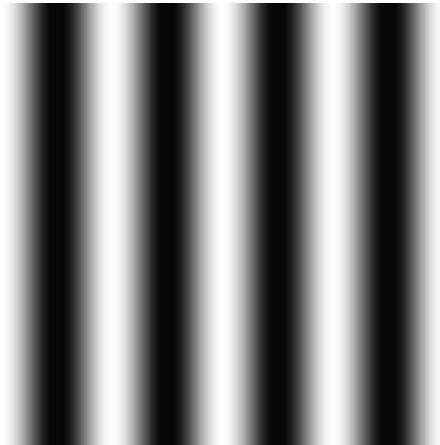
Image



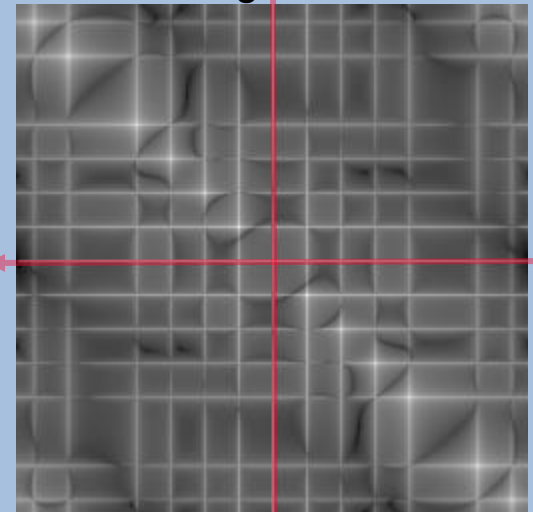
FT



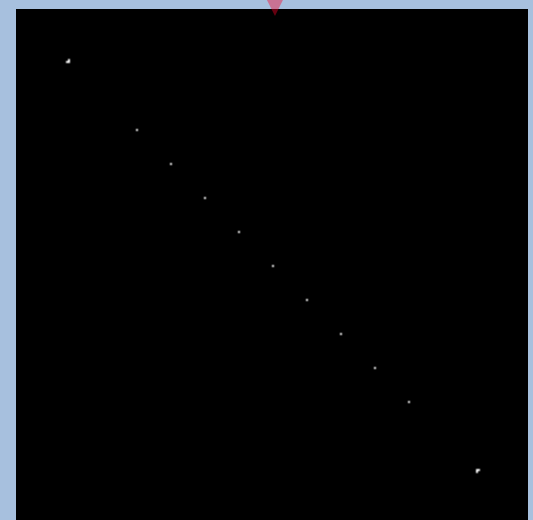
Spatial Domain \longleftrightarrow Frequency Domain



FT

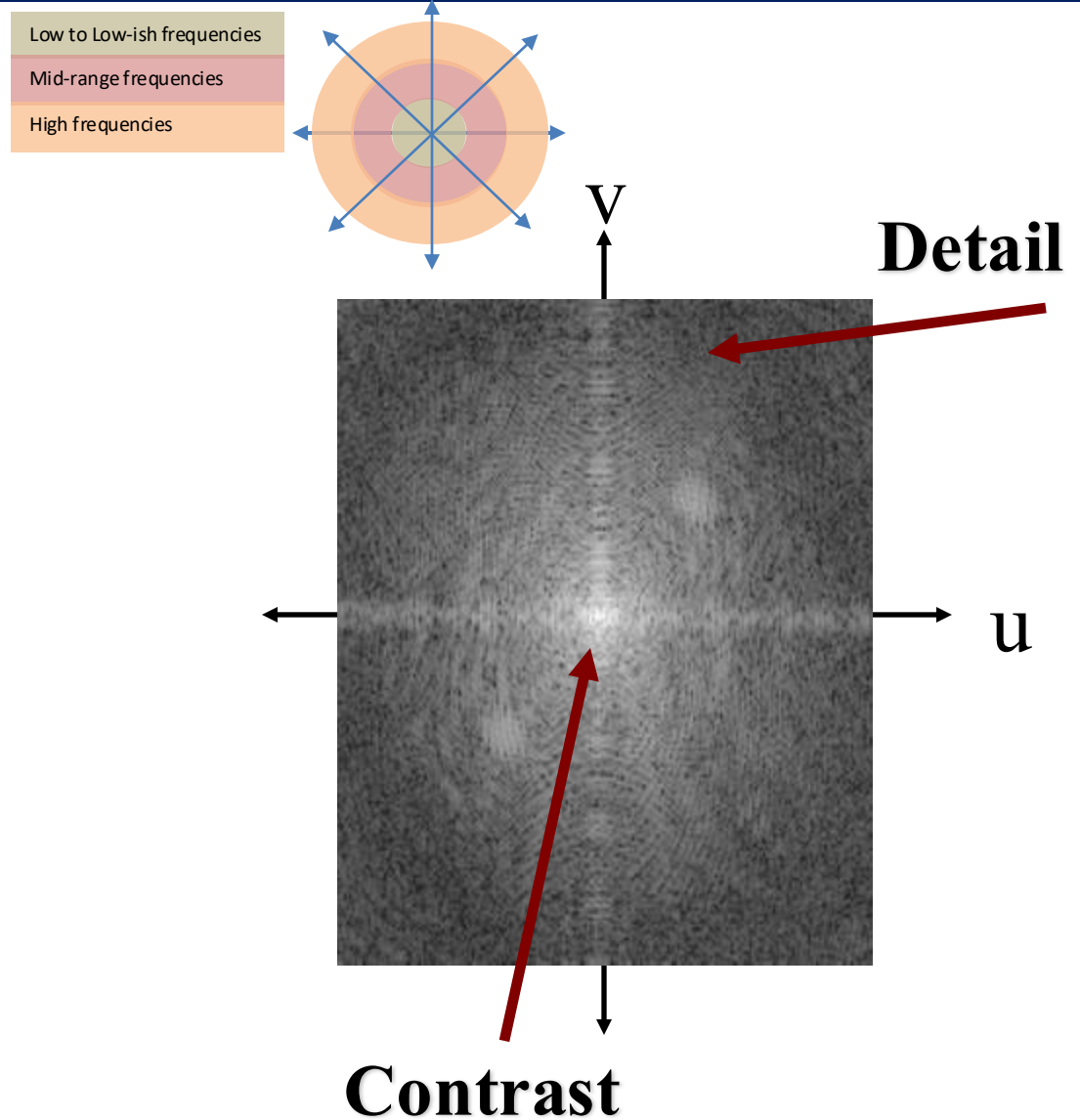


log of FT + 1



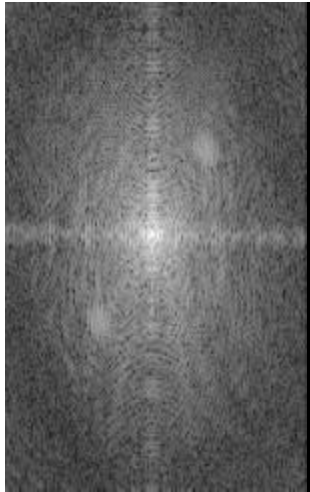
Thresholded log of FT+1

Relating Frequencies to Images

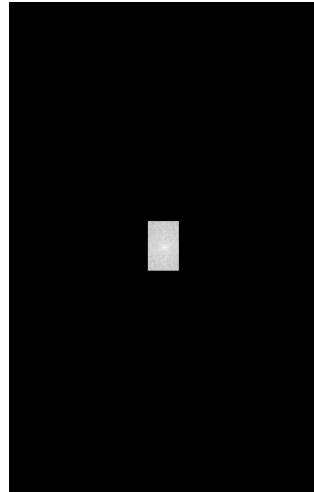


Example: Relating Frequencies to Images

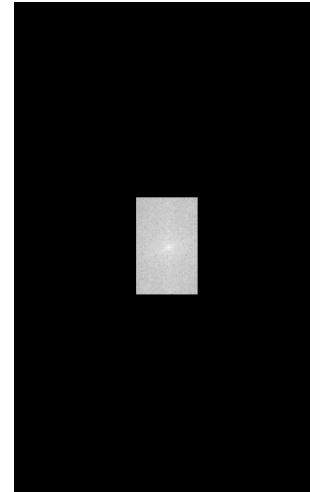
Fourier Space



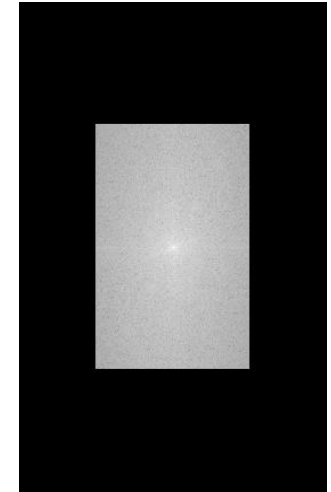
5%



10%

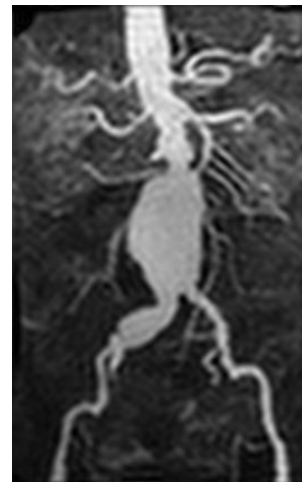
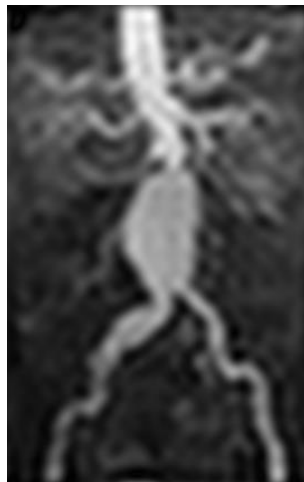
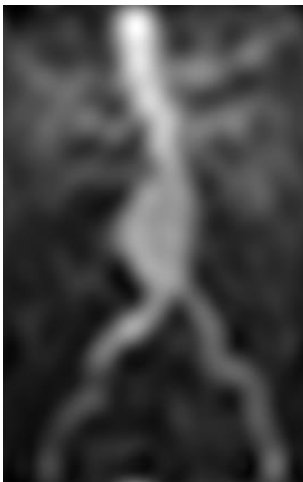


20%



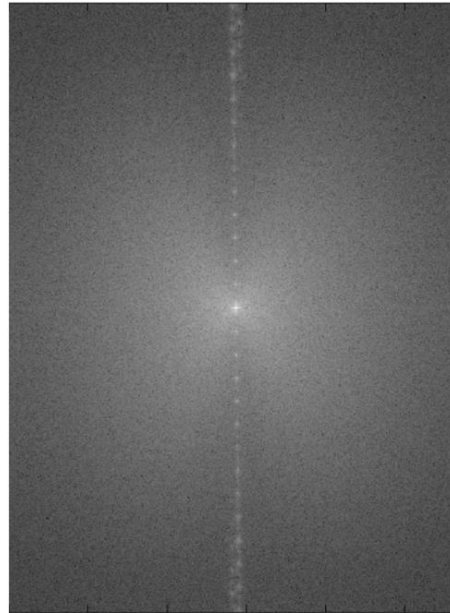
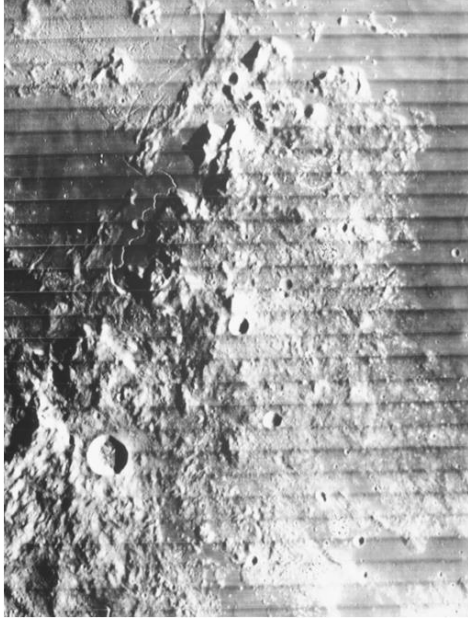
50%

Inverse Transform
back to image Space

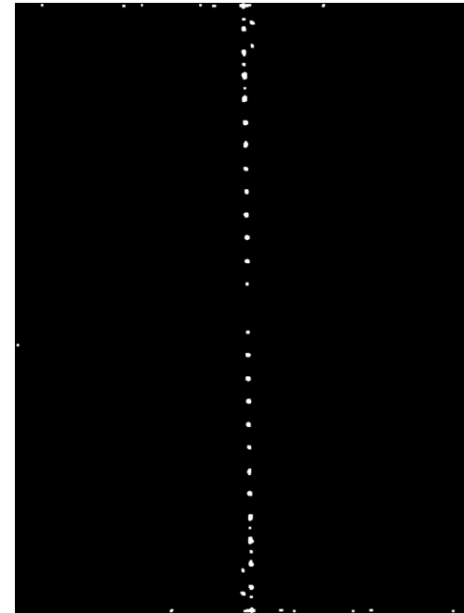


Example: Manipulating the FS

Lunar orbital image (1966)



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



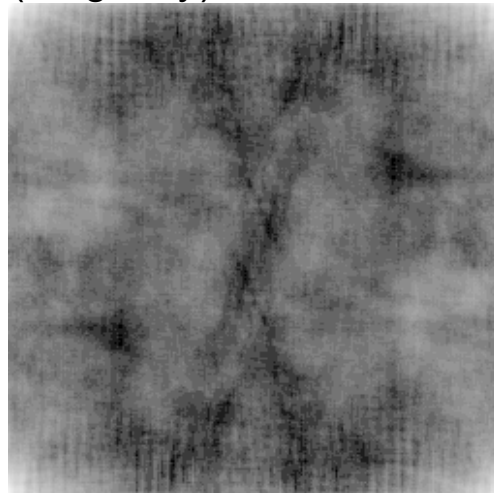
Mask used to
remove peaks



$$\text{iFFT}(F(u, v))$$

Importance of Phase

`ifft(mag only)`



`ifft(phase only)`



`ifft(mag(Peter) and Phase(Andrew))`

`ifft(mag(Andrew) and Phase(Peter))`

Separability

- Important property of the FT: *Separability*
- If a 2D transform is separable, the result can be found by successive application of two 1D transforms. This is a principle aspect of the Fast Fourier Transform (FFT).

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} F(x, v) e^{-j2\pi ux/N} \quad \text{where} \quad F(x, v) = \frac{1}{N} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi vy/N}$$

(,)

|

*1D row
transforms*

(,)

|

*1D column
transforms*

(,)

Recall: 2D Discrete Convolution

- The discrete version of 2D convolution is defined as:

$$g(x, y) = \sum_m \sum_n f(x - m, y - n)h(m, n)$$

		$y-1$	y	$y+1$	
$x-1$		43	12	61	
x		44	45	60	
$x+1$		43	50	61	

f *

	-1	0	1
-1	-1	0	1
0	-2	0	2
1	-1	0	1

h =

-68 =

$$\begin{aligned}
 & f(x+1, y+1)h(-1,-1) \\
 & + f(x+1, y)h(-1,0) \\
 & + f(x+1, y-1)h(-1,1) \\
 & + f(x, y+1)h(0,-1) \\
 & + f(x, y)h(0,0) \\
 & + f(x, y-1)h(0,1) \\
 & + f(x-1, y+1)h(1,-1) \\
 & + f(x-1, y)h(1,0) \\
 & + f(x-1, y-1)h(1,1)
 \end{aligned}$$

Convolution in the Spatial/Frequency Domain

Convolution Theorem:

Convolution in spatial domain
is equivalent to
multiplication in frequency domain
(and vice versa)

$$h = f * g$$

implies

$$H = FG$$

$$h = fg$$

implies

$$H = F * G$$

Deriving the Convolution Theorem

NOT EXAMINABLE

$$h(x) = f(x) * g(x) = \sum_y f(x - y)g(y)$$

$$H(u) = \sum_x \left(\sum_y f(x - y)g(y) \right) e^{(-iux2\pi/N)}$$

$$H(u) = \sum_y g(y) \left(\sum_x f(x - y) e^{(-iux2\pi/N)} \right)$$

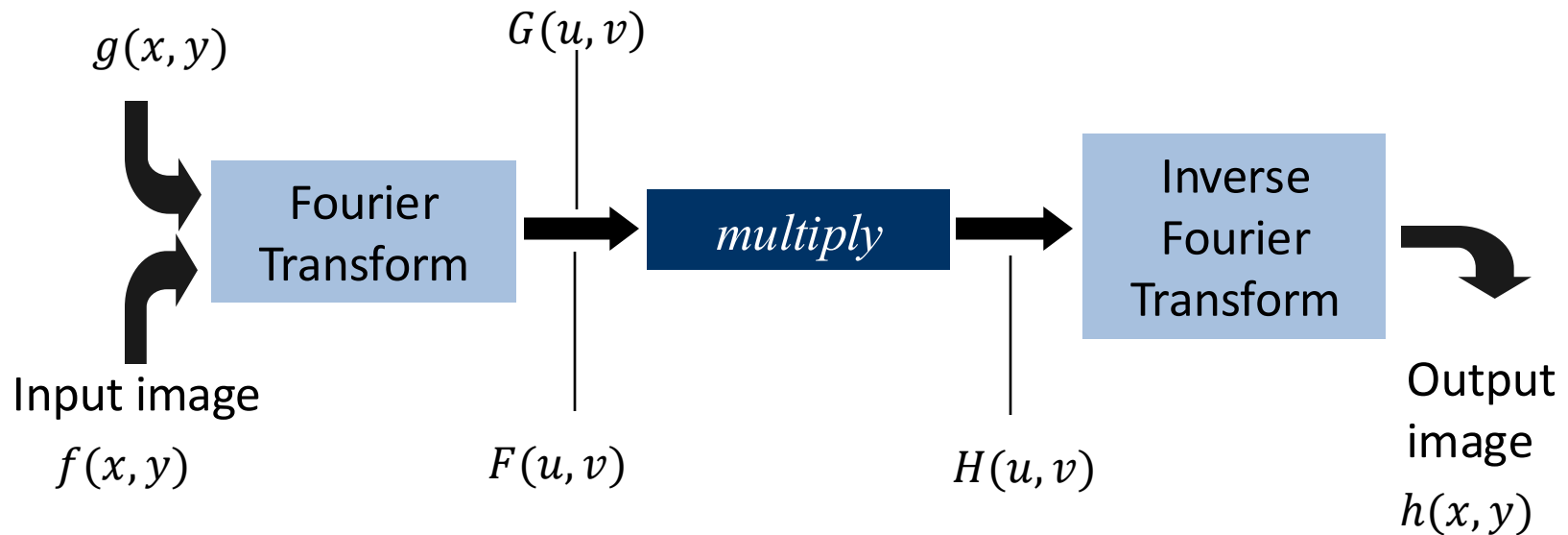
$$H(u) = \sum_y g(y) \left(F(u) e^{(-iuy2\pi/N)} \right)$$

$$H(u) = \sum_y g(y) e^{(-iuy2\pi/N)} F(u) = G(u) \cdot F(u) = F(u) \cdot G(u)$$

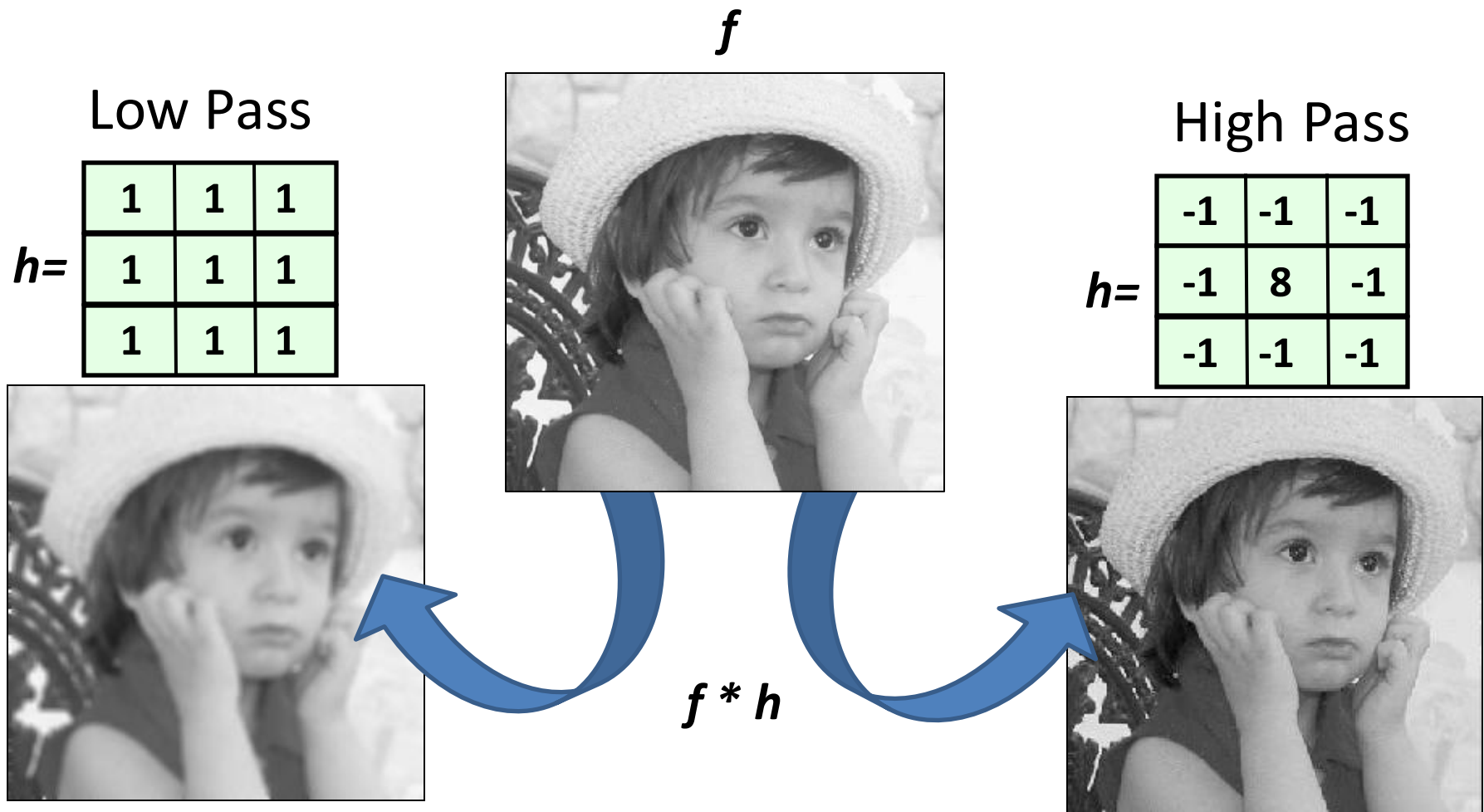
Fast Filtering using the Convolution Theorem

$$1\text{D: } H(u) = F(u)G(u)$$

$$2\text{D: } H(u, v) = F(u, v)G(u, v)$$

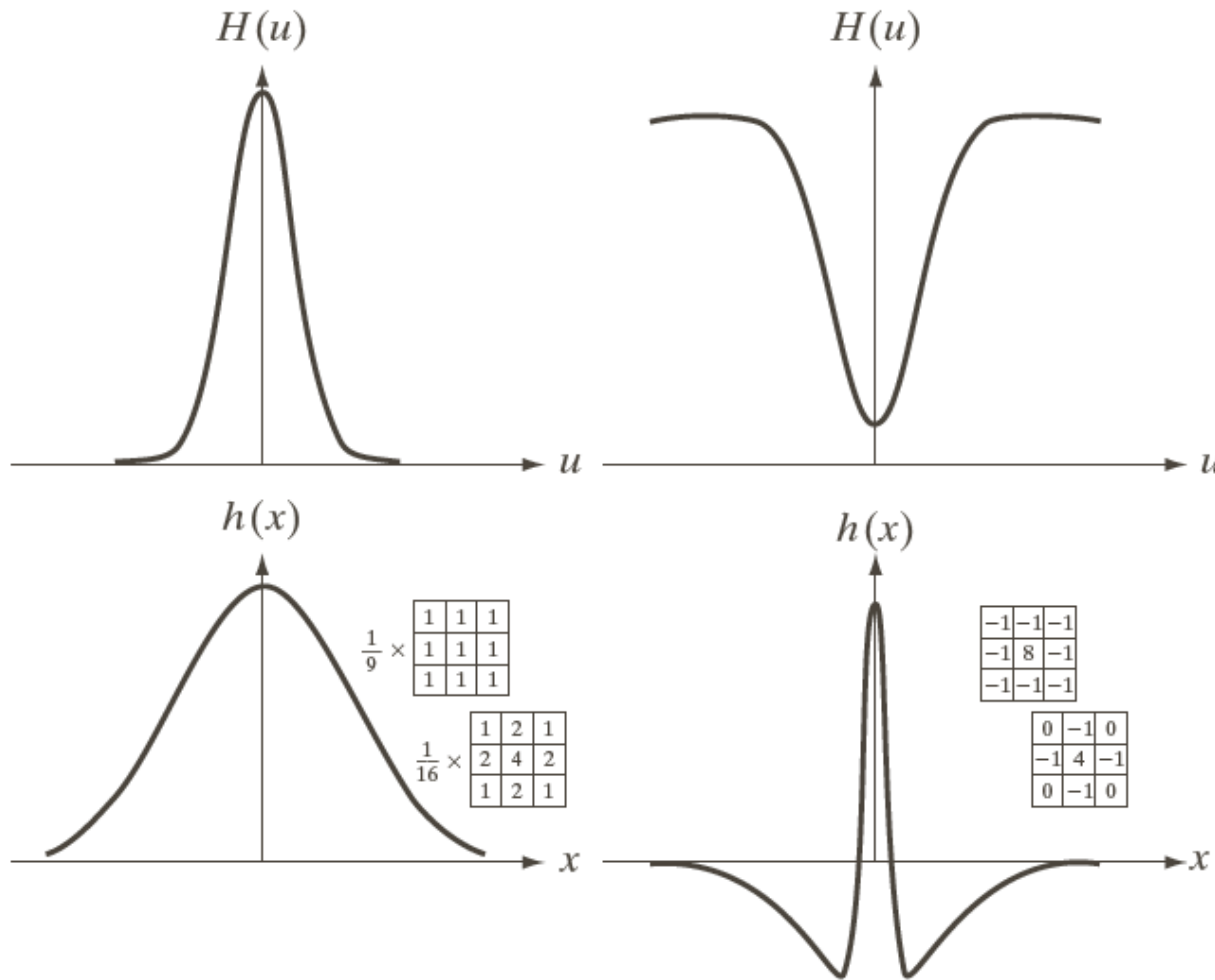


Recall: Spatial Low/High Pass Filtering



Frequency Domain Low/High Pass Filtering

a	c
b	d



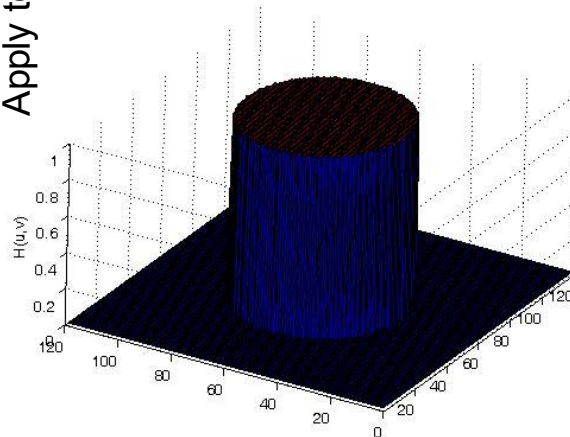
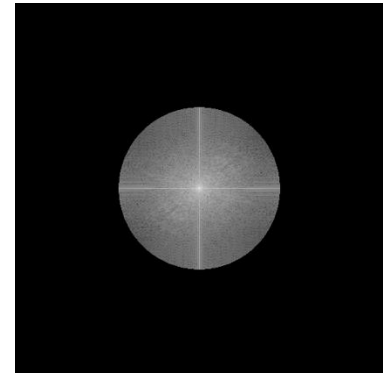
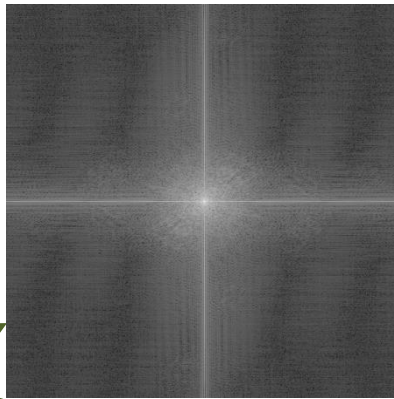
(a) A 1-D Gaussian lowpass filter in the frequency domain. (b) Spatial lowpass filter corresponding to (a). (c) Gaussian highpass filter in the frequency domain. (d) Spatial highpass filter

Ideal Low Pass Filter

- 1D: turning the “treble” down on audio equipment!
- 2D: smooth image



Apply to freq. domain

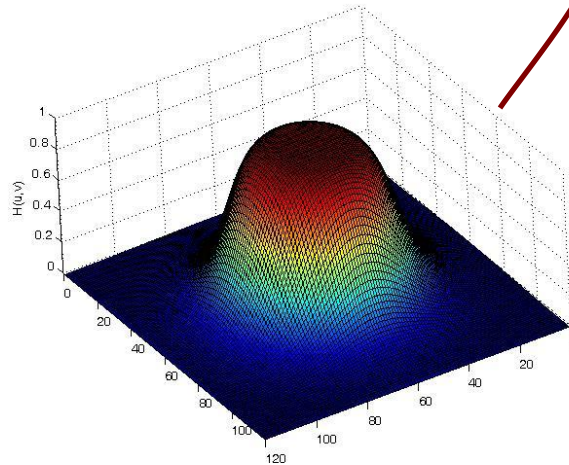


$$H(u, v) = \begin{cases} 1 & r(u, v) \leq r_0 \\ 0 & r(u, v) > r_0 \end{cases}$$

$$r(u, v) = \sqrt{u^2 + v^2}, \quad r_0 \text{ is the filter radius}$$

Butterworth's Low Pass Filter

Input image



After applying to freq. domain

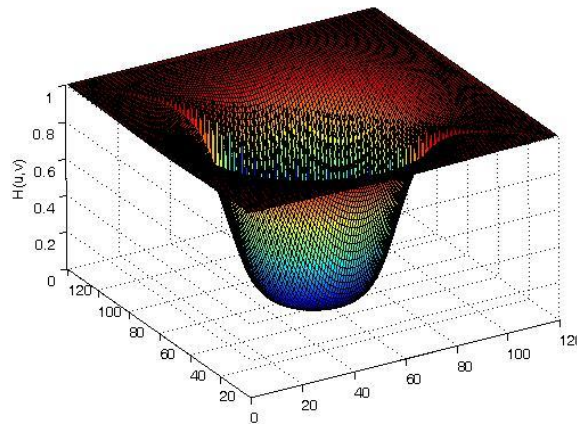
After filtering



$$H(u,v) = \frac{1}{1 + [r(u,v) / r_0]^{2n}} \quad \text{of order } n$$

Butterworth's High Pass Filter

Input image



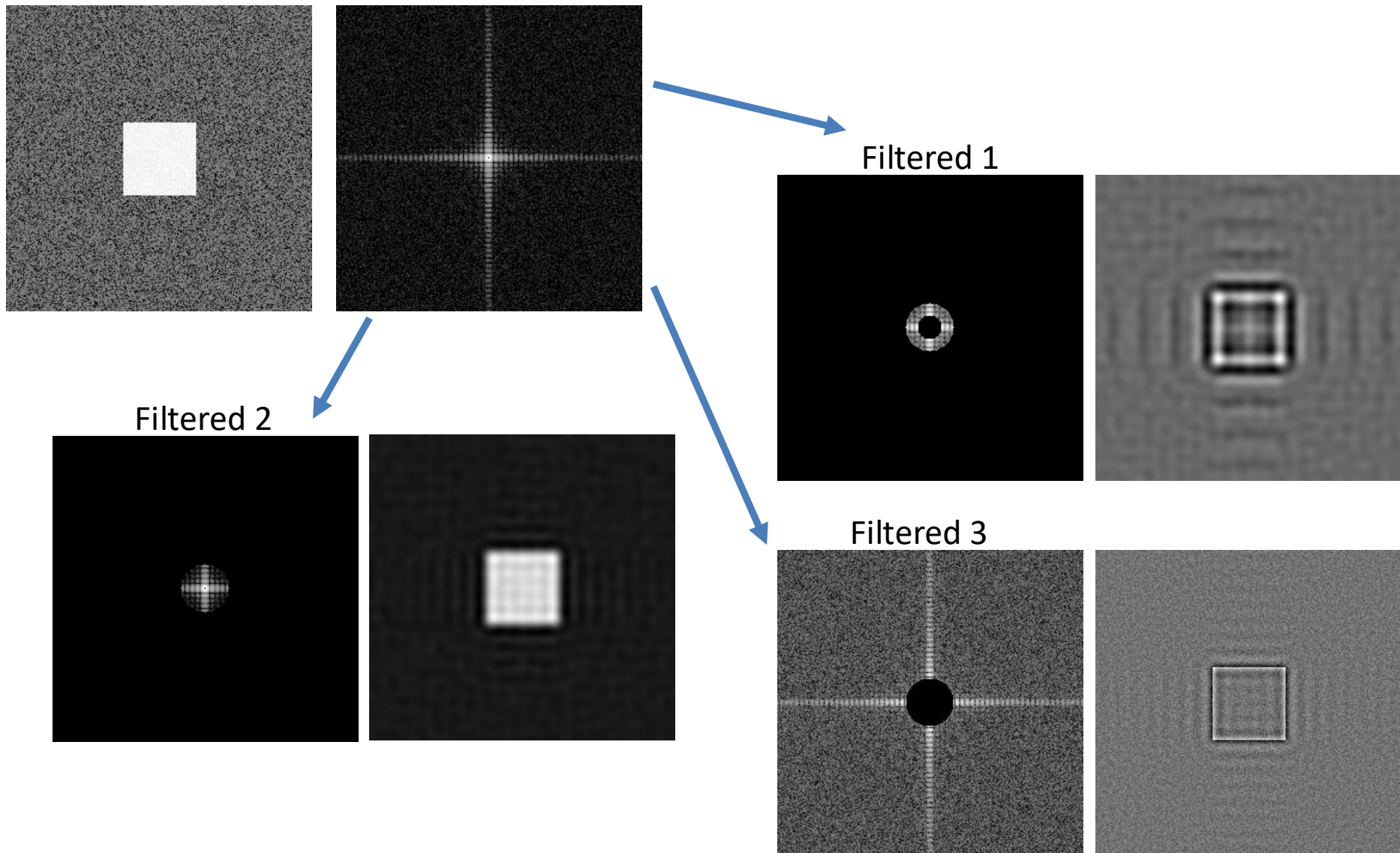
After filtering



Order of $n=3$

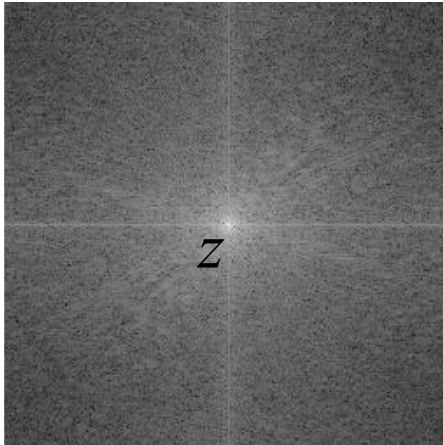
$$H(u, v) = \frac{1}{1 + [r_0 / r(u, v)]^{2n}} \quad \text{of order } n$$

Other Custom/Example filters



Other Custom/Example filters

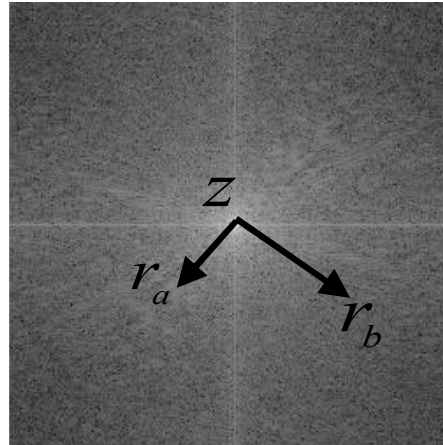
- Fourier space, with origin at $z=(u=0,v=0)$.



$$a \leq u \leq b$$

$$c \leq v \leq d$$

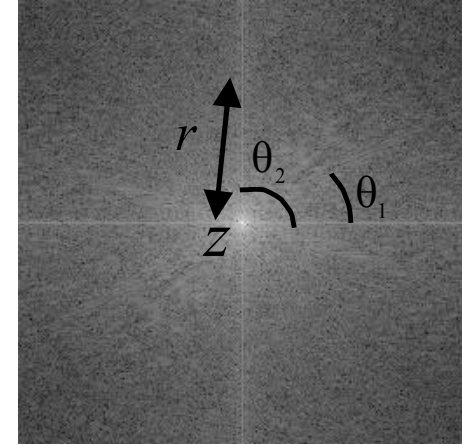
box



$$-r_b \leq u \leq r_b$$

$$\pm \sqrt{r_a^2 - u^2} \leq v \leq \pm \sqrt{r_b^2 - u^2}$$

ring

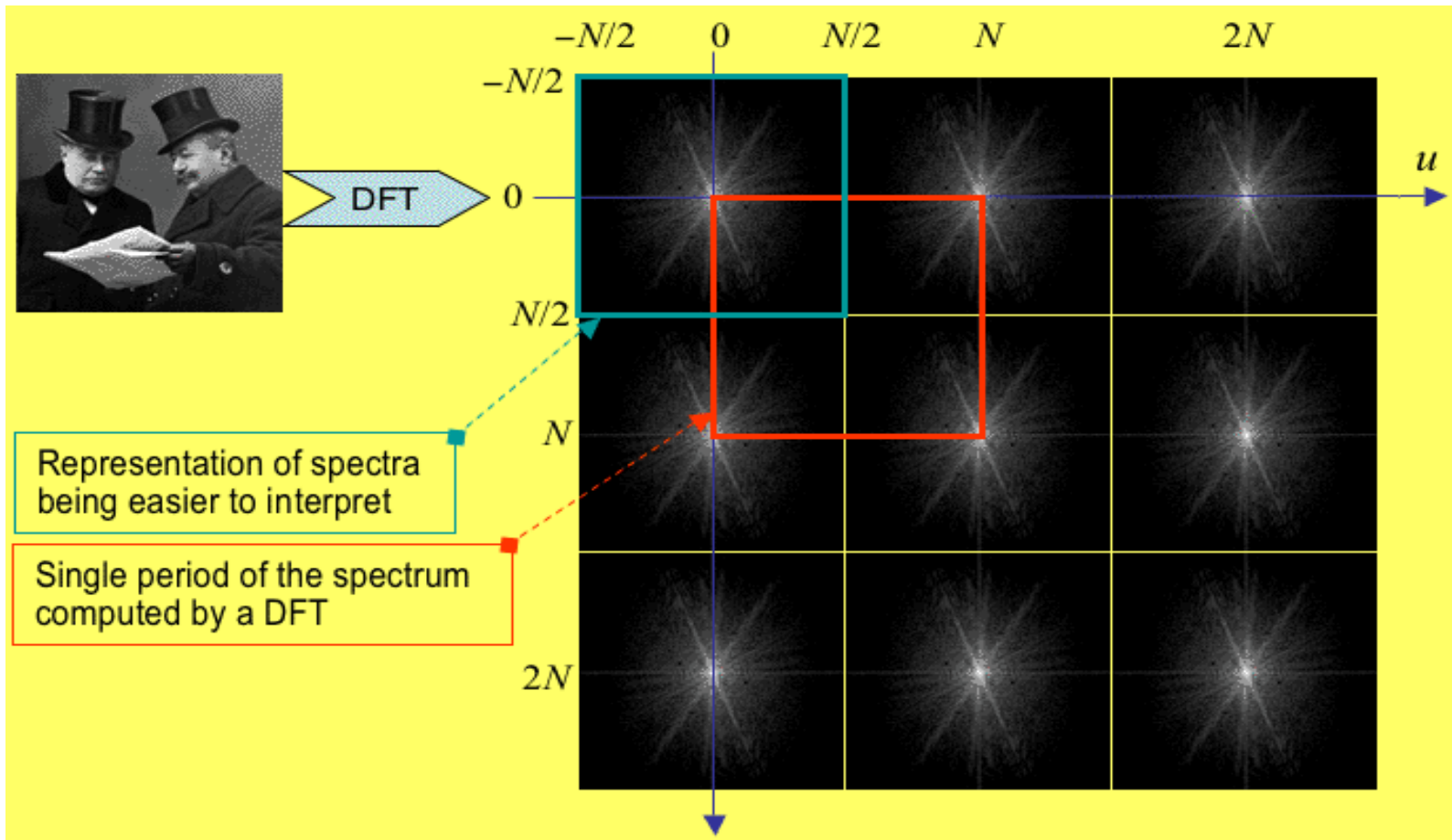


$$u^2 + v^2 = r^2$$

$$\theta_1 \leq \tan^{-1} \frac{v}{u} \leq \theta_2$$

sector

Periodic Spectrum

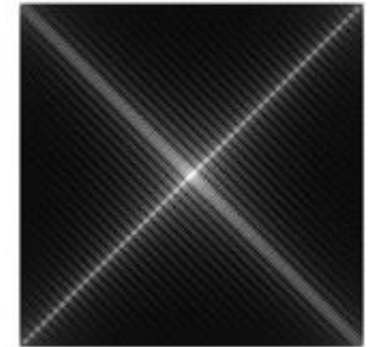
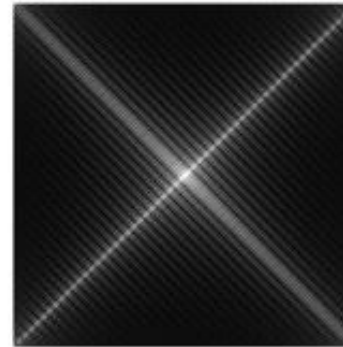
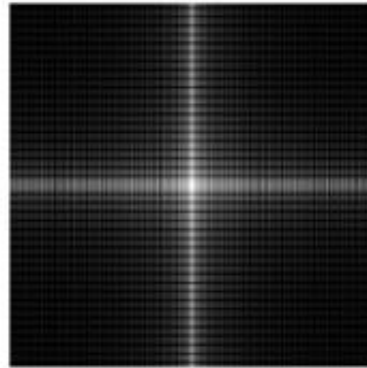


Self-Study: Effects of Rotation/Translation Illustrated

Translation or shift in Spatial Domain

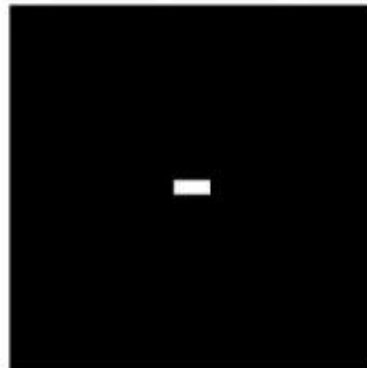
Spatial shifts result in a linear phase change in the frequency domain, but no change in the magnitude spectrum. Hence, the magnitude spectrum of a line or dot, for example, looks the same wherever it is in the image.

FT

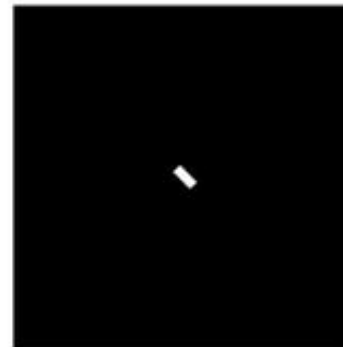


Rotation in Spatial Domain

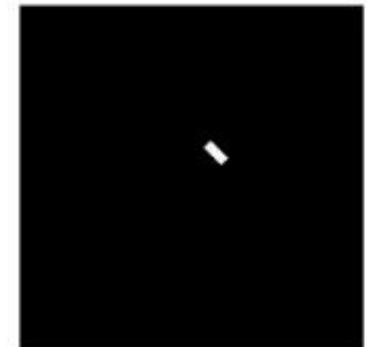
Rotation of an image in the spatial domain results in a corresponding rotation in the Fourier domain.



Image



Rotation



Rotation & translation

Self-Study: Summary of Filter Definitions

Notation notice! Using D instead of r

Lowpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

Ideal	Butterworth	Gaussian
$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$	$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$	$H(u, v) = e^{-D^2(u,v)/2D_0^2}$

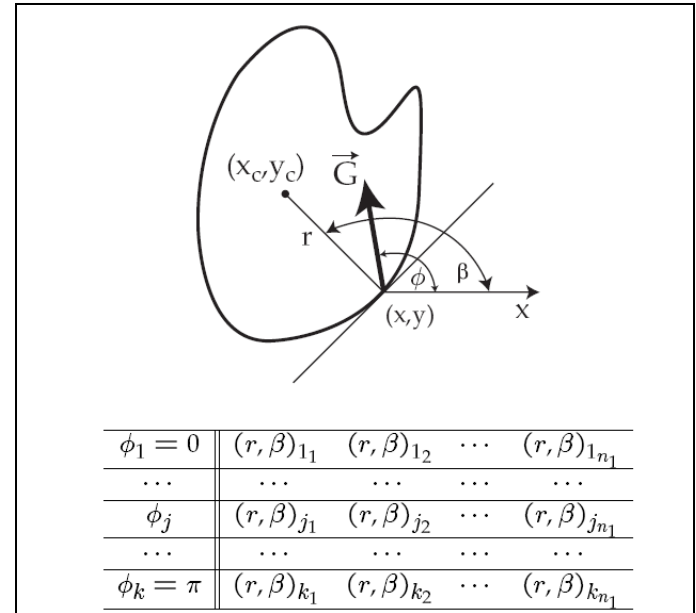
Highpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

Ideal	Butterworth	Gaussian
$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$	$H(u, v) = \frac{1}{1 + [D_0/D(u, v)]^{2n}}$	$H(u, v) = 1 - e^{-D^2(u,v)/2D_0^2}$

Next Lecture



Edge Detection



Hough Transform