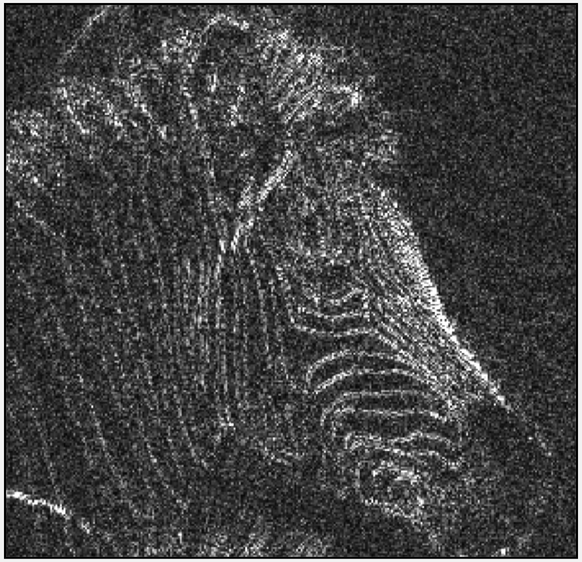
Department of Computer Science
University of Bristol

COMS30121 - Image Processing and Computer Vision

www.ole.bris.ac.uk/bbcswebdav/courses/COMS30121_2017/content



Lecture 03

Frequency Domain & Image Transforms

Andrew Calway | andrew@cs.bris.ac.uk
Tilo Burghardt | tilo@cs.bris.ac.uk
Majid Mirmehdi | majid@cs.bris.ac.uk

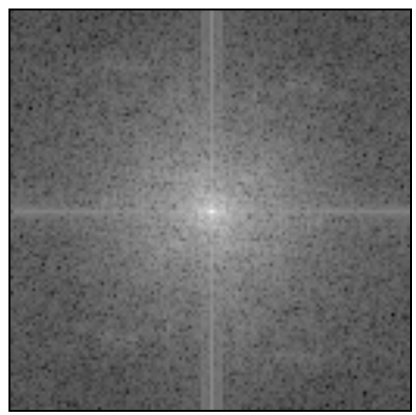
25 Slides

Image Transforms

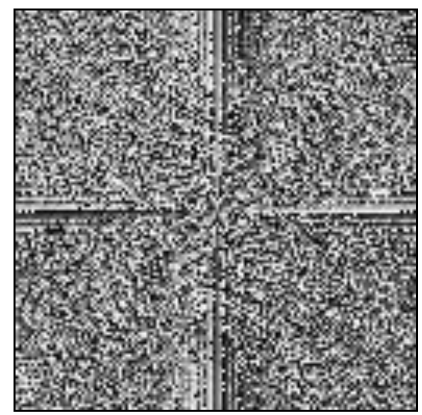
- **Image Transform** = derivation of a ***new representation*** of the input data by re-encoding the image in another parameter space (Fourier, DCT, Wavelet, Haar, etc.)
- Image Transforms are classic processing techniques, used in image filtering, compression, feature extraction, etc.

image f (x, y)

Fourier Transform



power spectrum

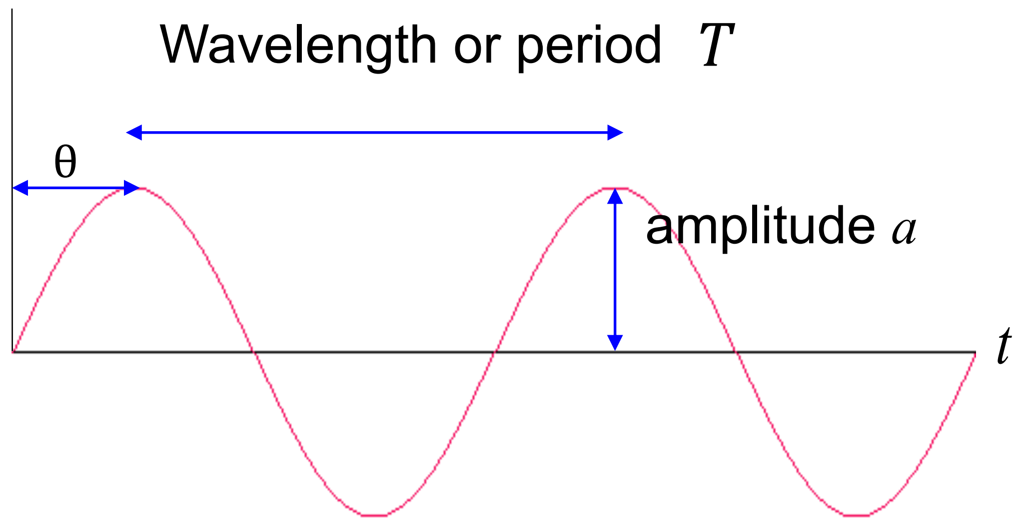


phase spectrum

Signals as Functions

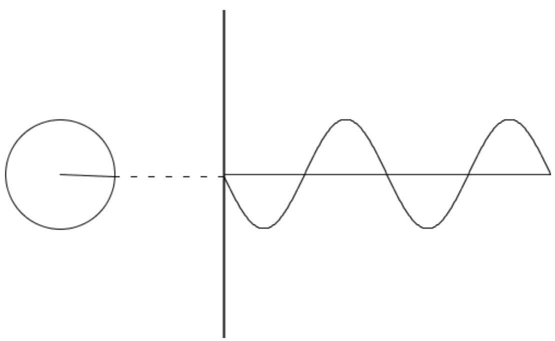
Frequency - allows us to characterise signals:

- 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

$$f(t) = a\ sin2\pi ut$$



Fourier's Theorem

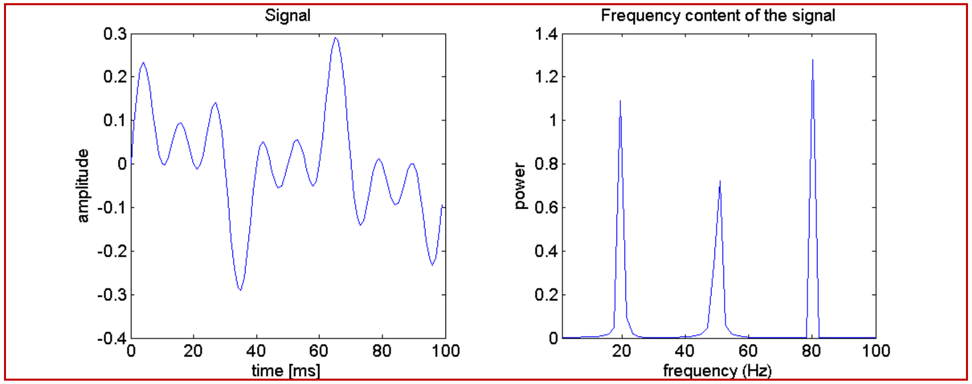
$$f(x) = \int a_n \cos(nx) + b_n \sin(nx) \;\; \delta n$$

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.

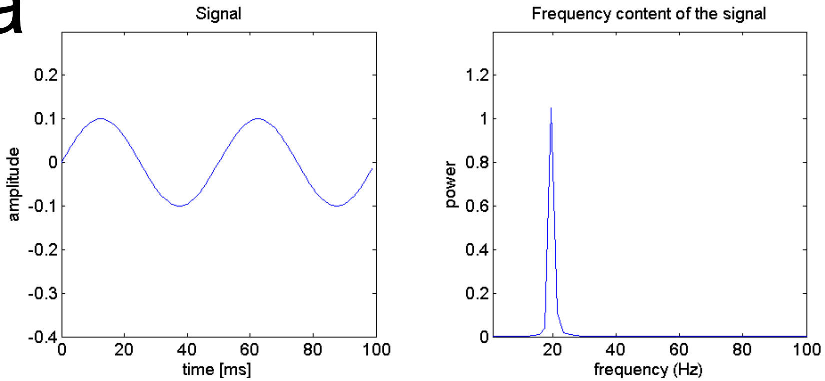
Intuition I: Simple 1D example

d = a + b + c



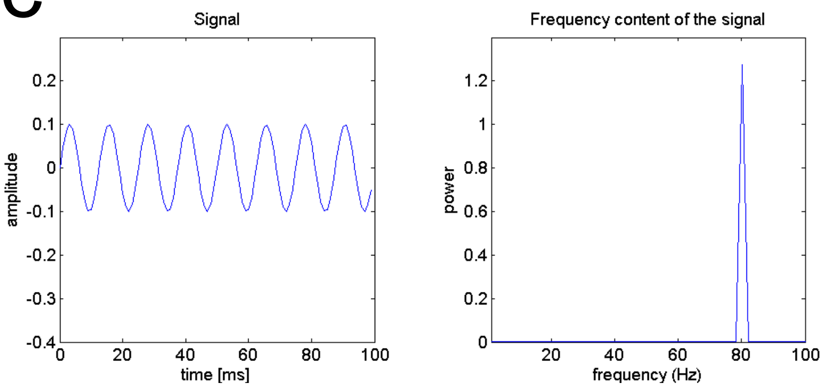
time domain frequency domain

a



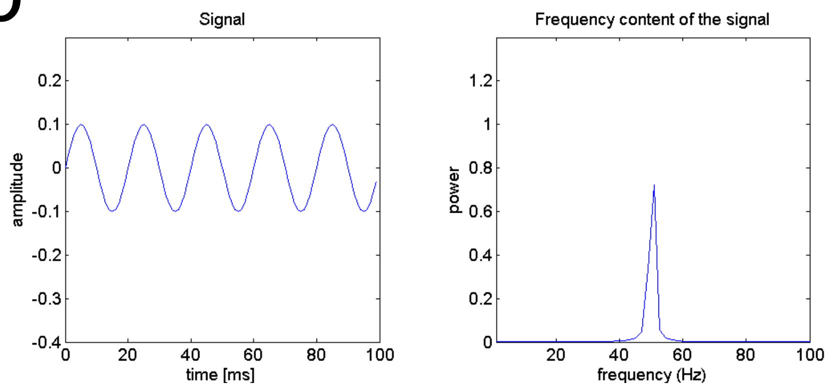
time domain frequency domain

c



time domain frequency domain

b

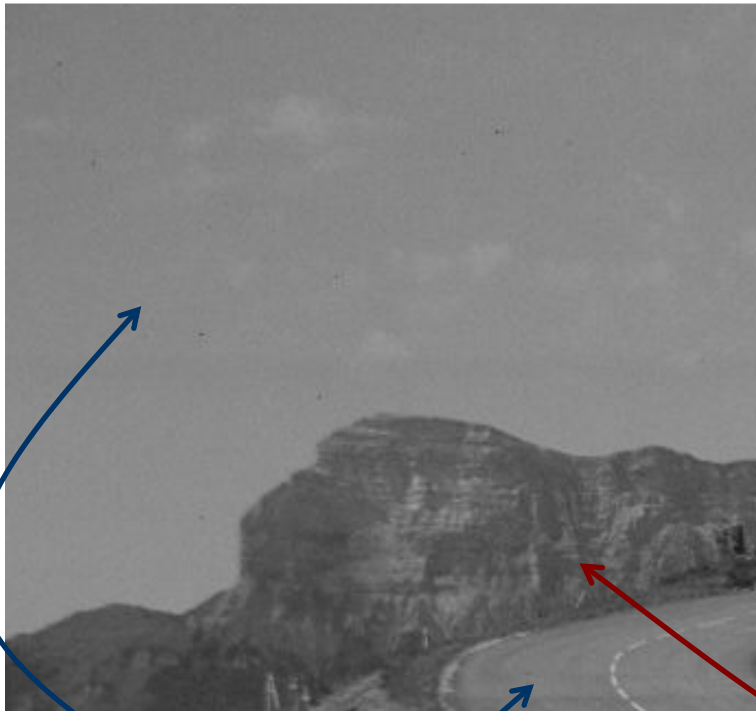


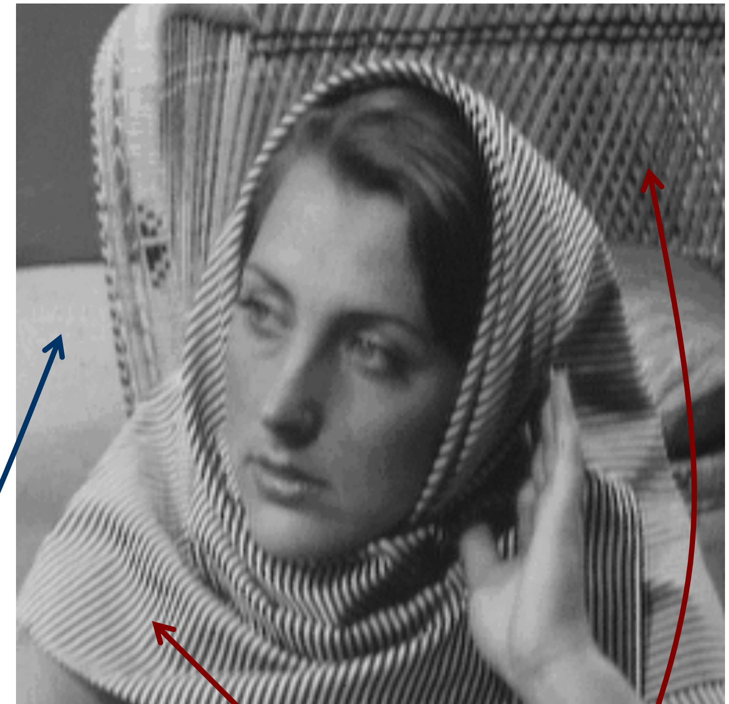
time domain frequency domain

Intuition II: Simple 1D example

Animation by Lucas V Barbosa

Intuition III: Concept of Frequency in Images



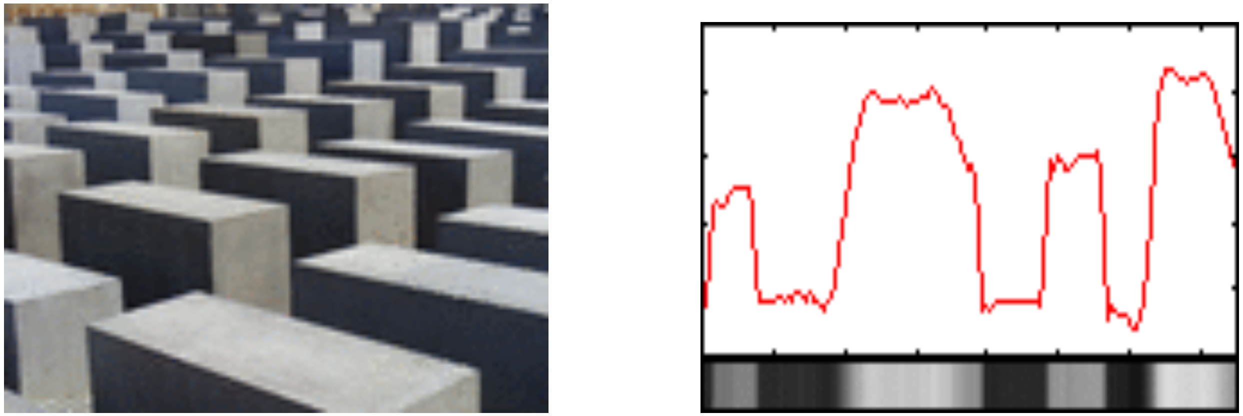


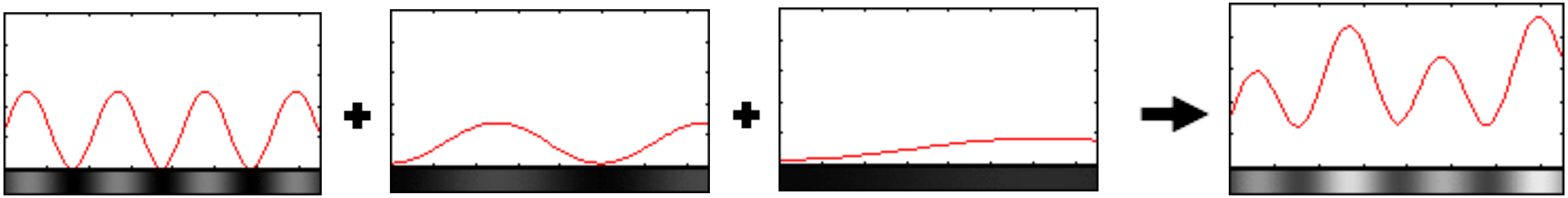
Slowly changing → low frequency

Rapidly changing →high frequency

Intuition IV: Images are waves!?

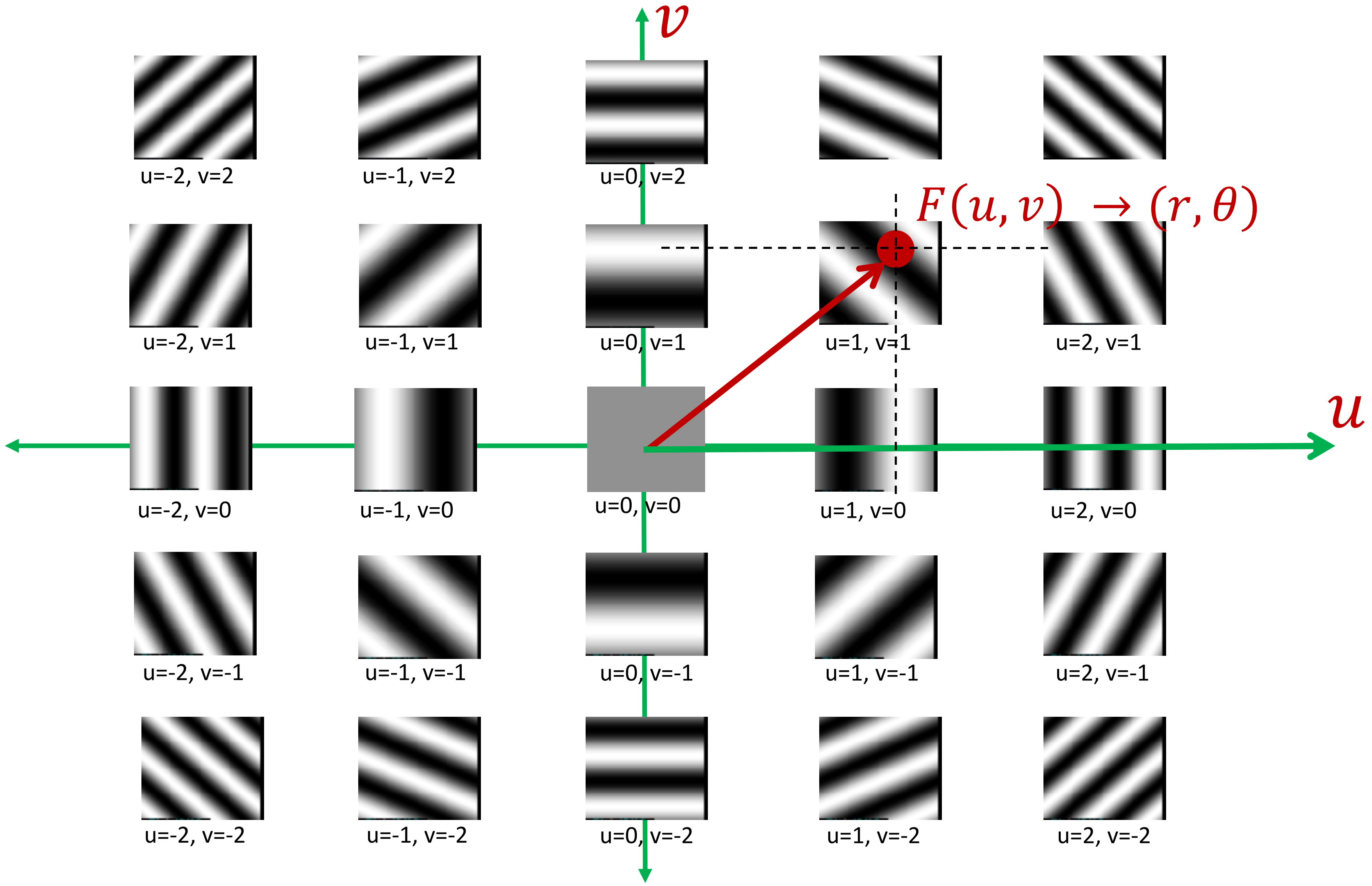
Take a single row or column of pixels from an image → a 1D signal





From ImageNagik

‘Fabric’ of the 2D Fourier Space (as kernels)

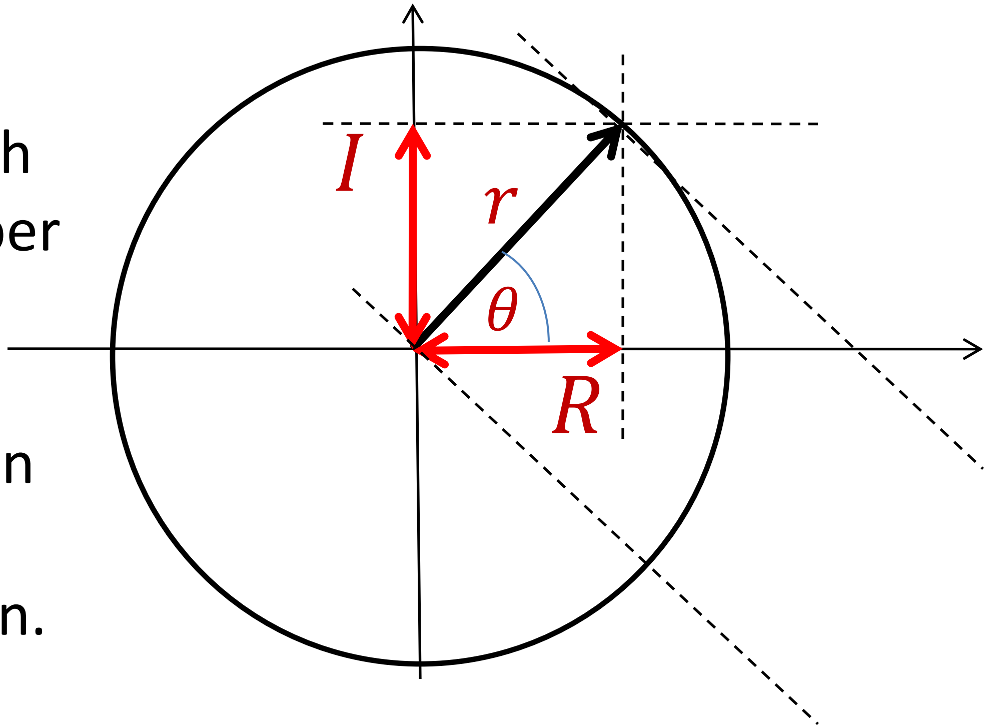


Euler's Equation

Instead of representing kernel functions as explicit sine or cosine functions, we can represent them compactly by exponentials:

$$e^{2\pi i(ux+vy)/N} = \cos(2\pi(ux+vy)/N) + i\sin(2\pi(ux+vy)/N)$$

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



Change of Base: The Fourier Transform

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

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

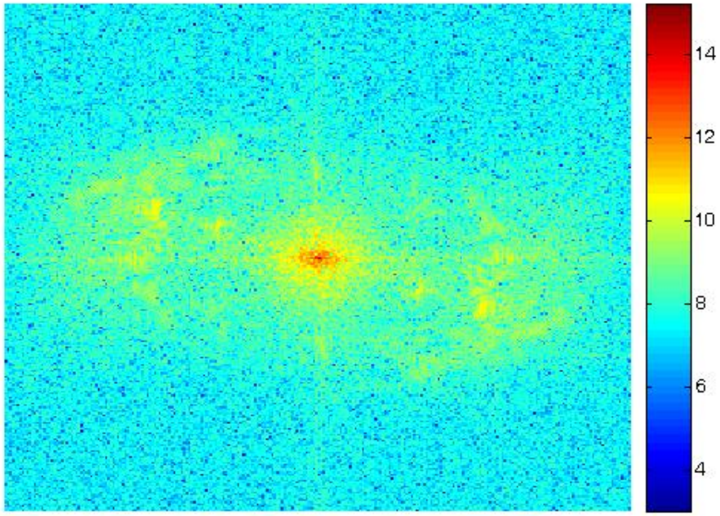
image kernels (probing functions)

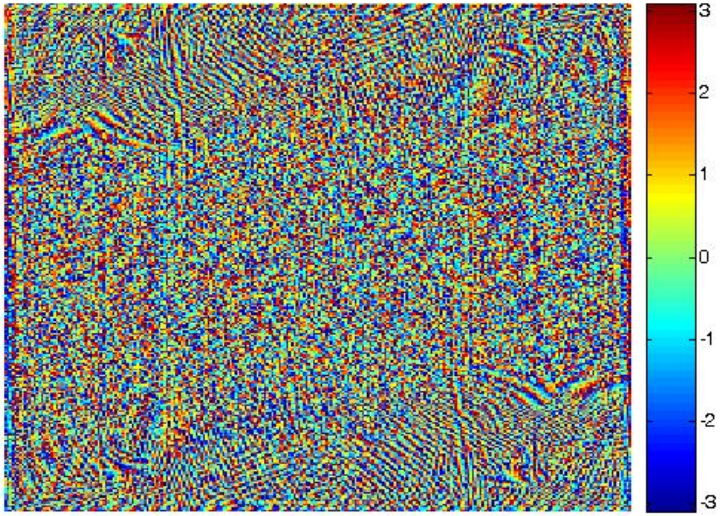
All kernels together form a new orthogonal basis for our image.

Thus, we have transformed the image f from a spatial domain indexed in (x,y) to a frequency domain representation in (u,v).

Example: Power Spectrum and Phase Spectrum

$f(x, y)$





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

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

Components of 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 Power Spectrum):

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

- Phase Angles (forming the Phase Spectrum):

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

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

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

2D Fourier Transform Pair

Given a Fourier transform of a discrete function of two variables:

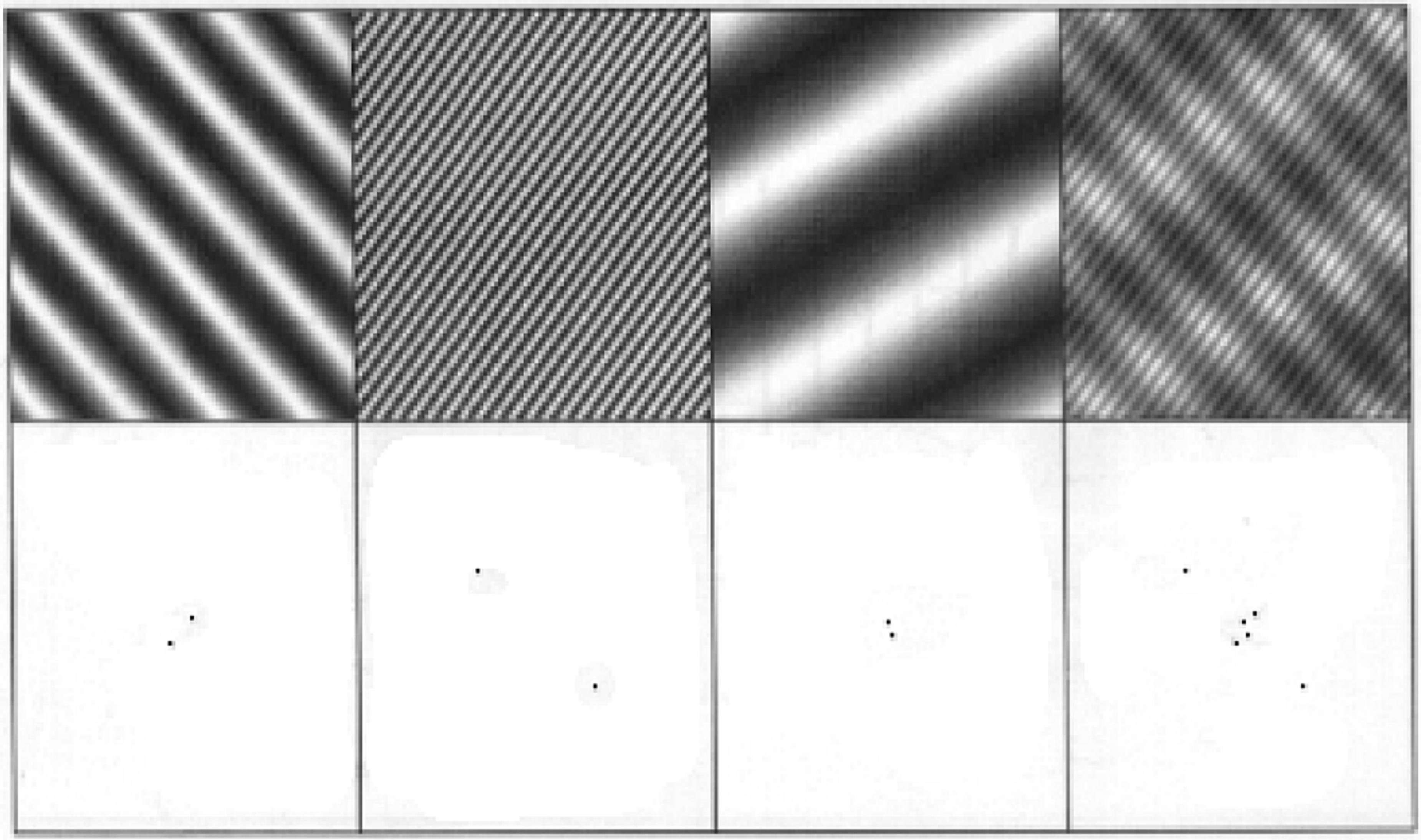
$$F(u,v) = \sum_{y=0}^{N-1} \sum_{x=0}^{N-1} f(x,y)\ e^{(-2\pi i(ux+vy)/N)}$$

There exists an inverse transform that can reconstruct the original image in spatial coordinates from its representation in the frequency domain. This is known as the Inverse Fourier Tansform:

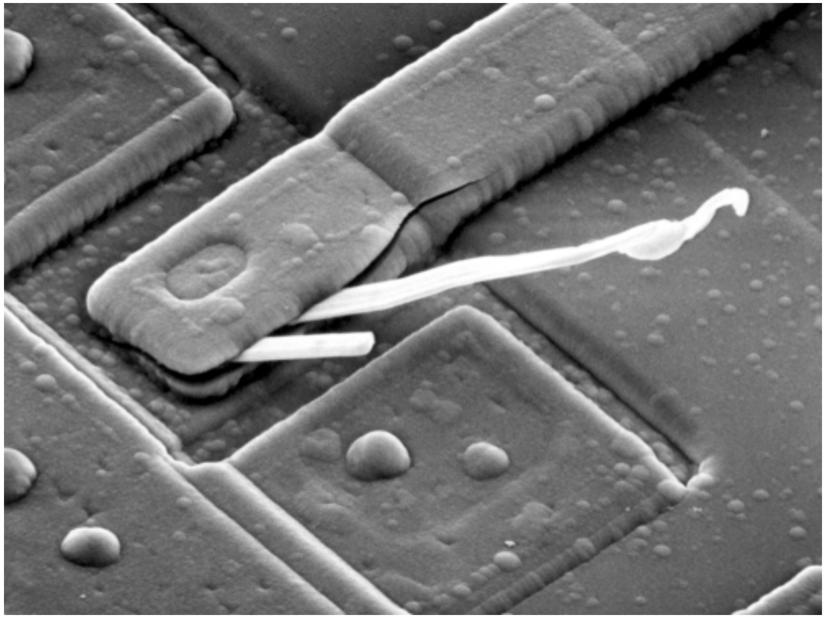
$$f(x,y) = \frac{1}{N^2} \sum_{v=0}^{N-1} \sum_{u=0}^{N-1} F(u,v)\ e^{(2\pi i(ux+vy)/N)}$$

Together the two equations form the Fourier Transform Pair.

Image Pairs: Spatial Domain vs Frequency Domain

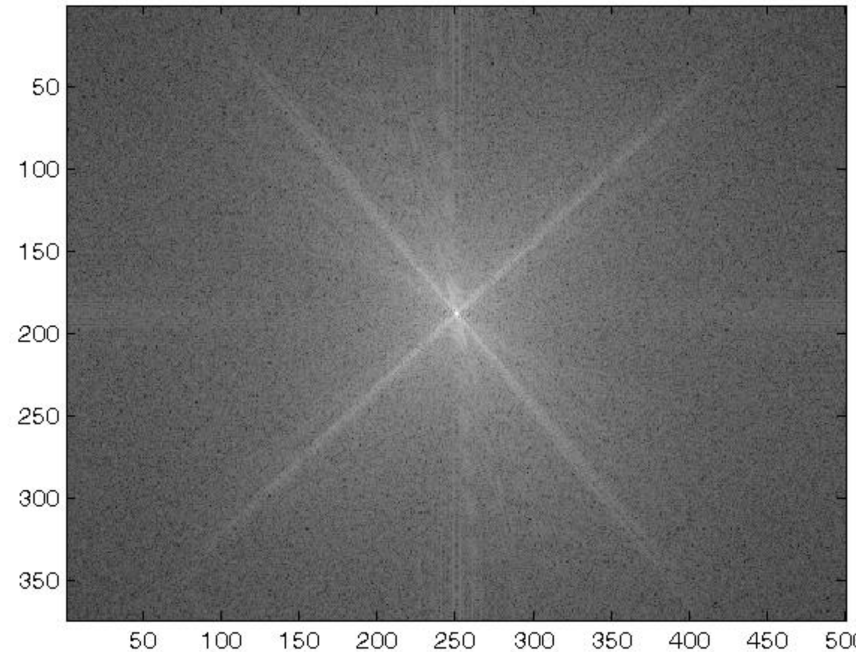


Example: Interpreting the FS

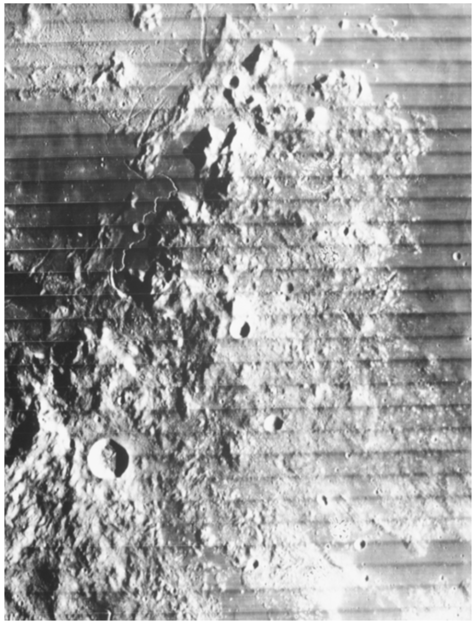


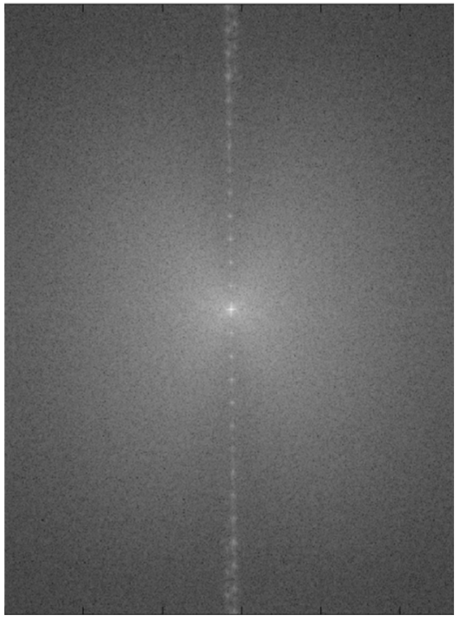
Scanning electron microscope image of an integrated circuit

Can we interpret what the bright components mean?

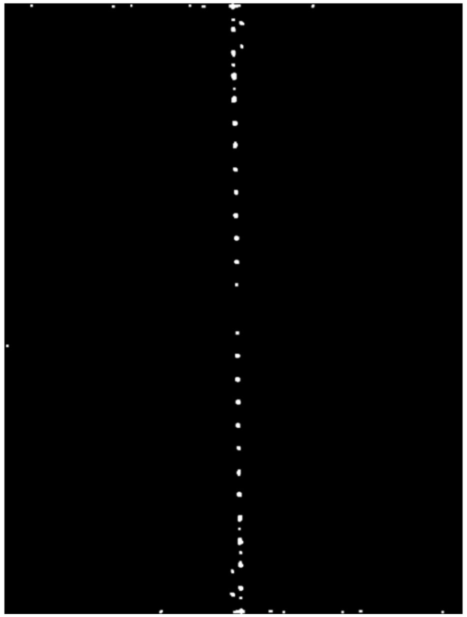


Lunar orbital image (1966)





$|F(u, v)|$



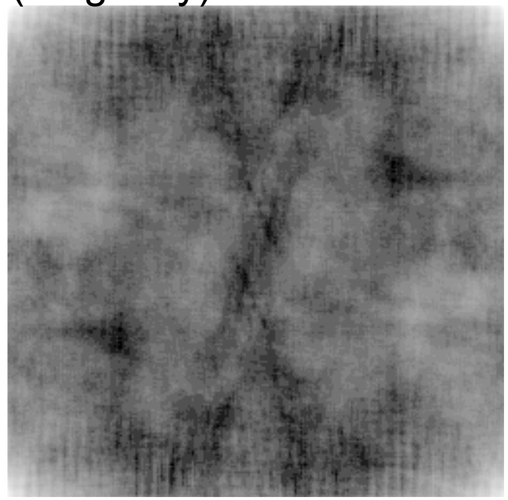
Remove peaks

$\mathrm{i}FFT(F(u, v))$

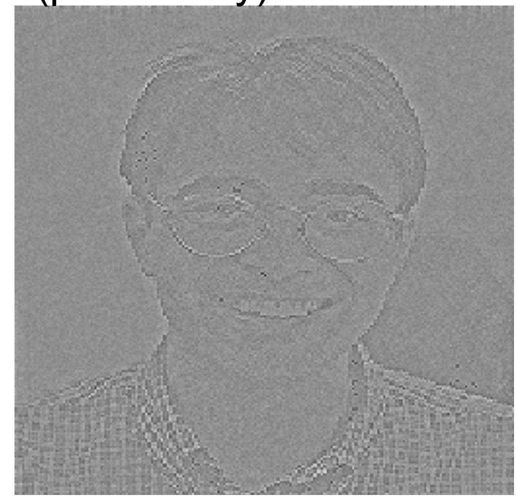
Slide by A. Zisserman

Importance of Phase

ifft(mag only)



ifft(phase only)



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

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

Recap: 2D Discrete Convolution

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

$$g(x,y)=\sum_{m}\sum_{n} f(x-m,y-n)h(m,n)$$

$$\underbrace{\begin{array}{c|c|c|c|c|c|}
 & & y-1 & y & y+1 & \\
\hline
 & & & & & \\
\hline
x-1 & & 43 & 12 & 61 & \\
\hline
x & & 44 & \boxed{45} & 60 & \\
\hline
x+1 & & 43 & 50 & 61 & \\
\hline
 & & & & & \\
\hline
\end{array}}_{f} * \underbrace{\begin{array}{c|c|c|c|}
 & -1 & 0 & 1 \\
\hline
-1 & -1 & 0 & 1 \\
\hline
0 & -2 & 0 & 2 \\
\hline
1 & -1 & 0 & 1 \\
\hline
\end{array}}_{h} = -68 = \begin{array}{l}
f(x+1,y+1)h(-1,-1) \\
+ f(x+1,y)h(-1,0) \\
+ f(x+1,y-1)h(-1,1) \\
+ f(x,y+1)h(0,-1) \\
+ f(x,y)h(0,0) \\
+ f(x,y-1)h(0,1) \\
+ f(x-1,y+1)h(1,-1) \\
+ f(x-1,y)h(1,0) \\
+ f(x-1,y-1)h(1,1)
\end{array}$$

Convolution in the Spatial/Frequency Domain

Convolution Theorem:

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

$$h = f * g \quad \textbf{\textit{implies}} \quad H = FG$$

$$h = fg \quad \textbf{\textit{implies}} \quad H = F * G$$

Deriving the Convolution Theorem

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

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

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

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

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

Fast Filtering using the Convolution Theorem

1D: $H(u) = F(u)G(u)$

2D: $H(u,v) = F(u,v)G(u,v)$

Input image $f(x,y)$ → Fourier Transform → $F(u,v)$ → *Filter Function* $G(u,v)$ → $H(u,v)$ → Inverse Fourier Transform → Output image $h(x,y)$

Reminder: Spatial Low/High Pass Filtering

f

Low Pass

***h*=**

1	**1**	**1**
1	**1**	**1**
1	**1**	**1**

f * h

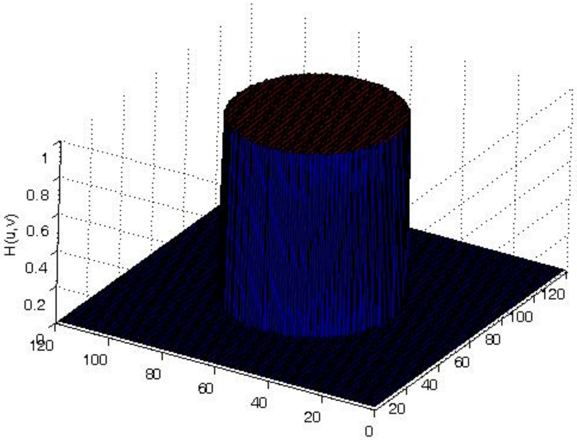
High Pass

***h*=**

-1	**-1**	**-1**
-1	**8**	**-1**
-1	**-1**	**-1**

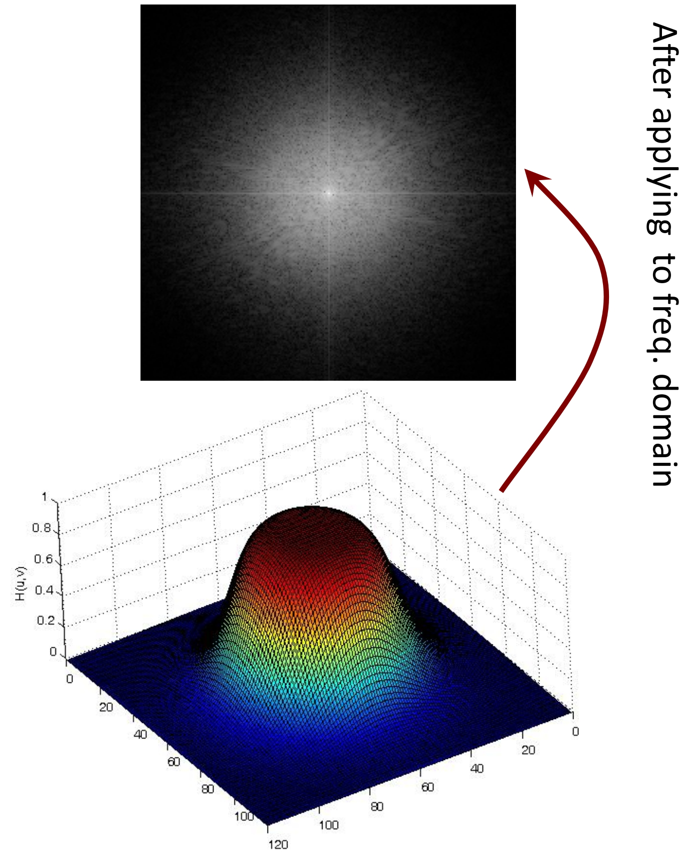
Low Pass Filtering in Frequency Domain

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



$$H(u,v) = \begin{cases} 1 & r(u,v) \le r_0 \\ 0 & r(u,v) > r_0 \end{cases} \qquad r(u,v) = \sqrt{u^2 + v^2}\,,\ r_0 \text{ is the filter radius}$$

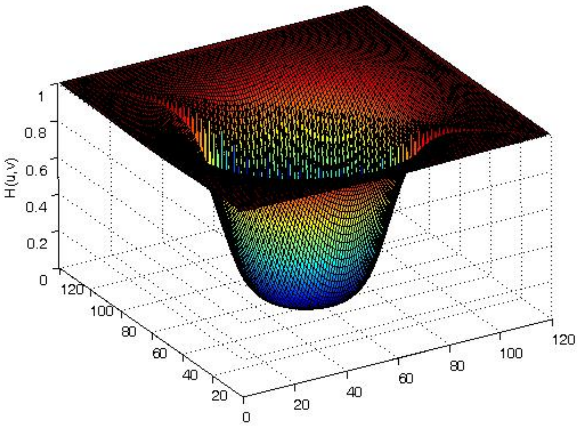
Butterworth’s Low Pass Filter



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

Butterworth's High Pass Filter

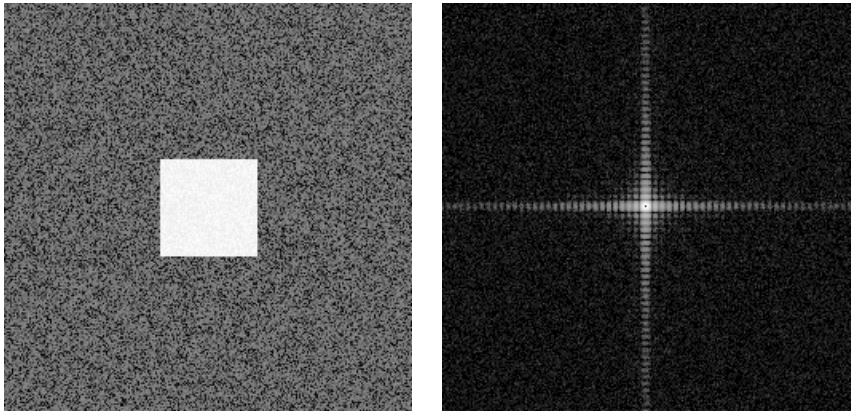
- 1D: turning the bass down on audio equipment!
- 2D: sharpen image



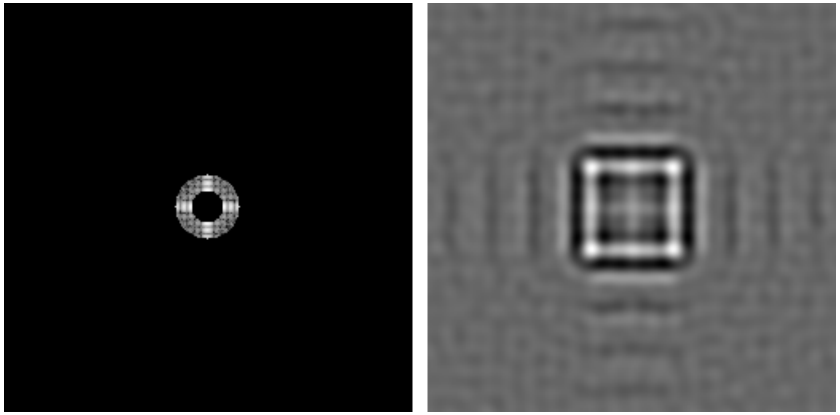
Order of $n=3$

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

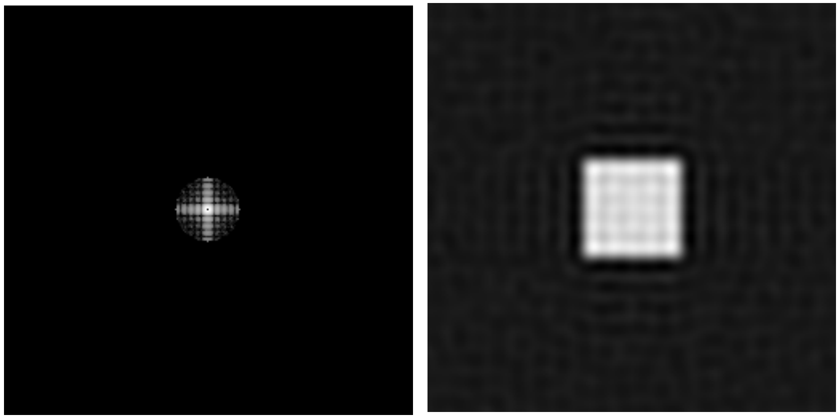
Example filters



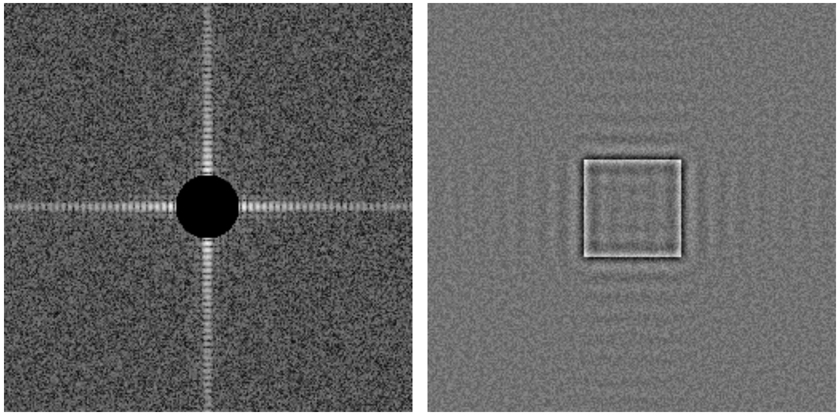
Filtered 1



Filtered 2



Filtered 3



Optional Excurse: Wiener De-Convolution

Norbert Wiener

- **<u>Idea:</u>** Restore an image by convolution with an adjusted inverse kernel that estimates the loss of information per frequency.

inverse of original kernel

$$\boxed{\frac{1}{H(f)}}\left[\frac{|H(f)|^2}{|H(f)|^2+\boxed{\frac{1}{\mathrm{SNR}(f)}}}\right]$$

estimated loss at frequency f