Signal and Image processing
A short intro

Saad Jbabdi, FMRIB Centre, Oxford

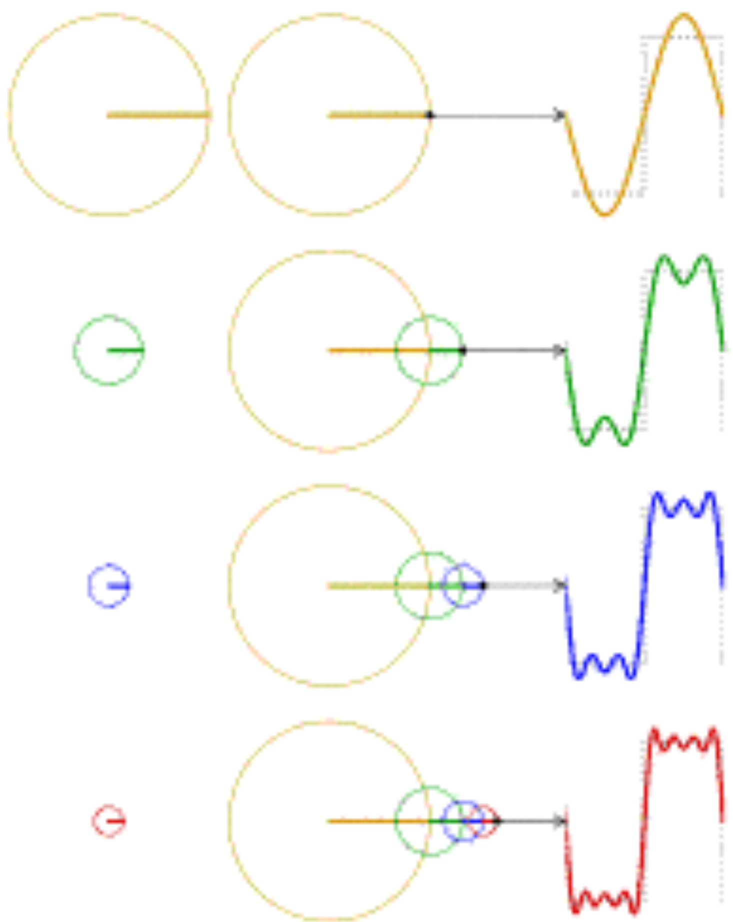
saad@fmrib.ox.ac.uk

When I say signal processing…

- Fourier series, Fourier transform, FFT
- A bit of image processing (but related to Fourier)
- A bit on image reconstruction (also related to Fourier)

Fourier Analysis
what's it all about?

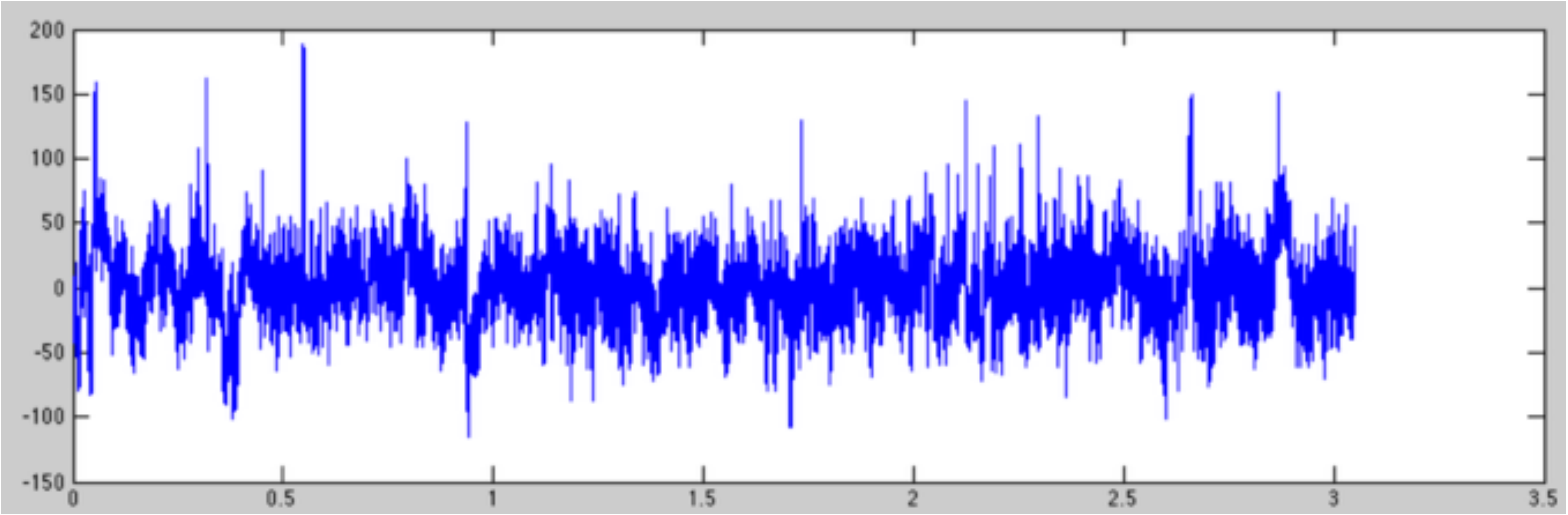
- **Sines and cosines are building blocks**



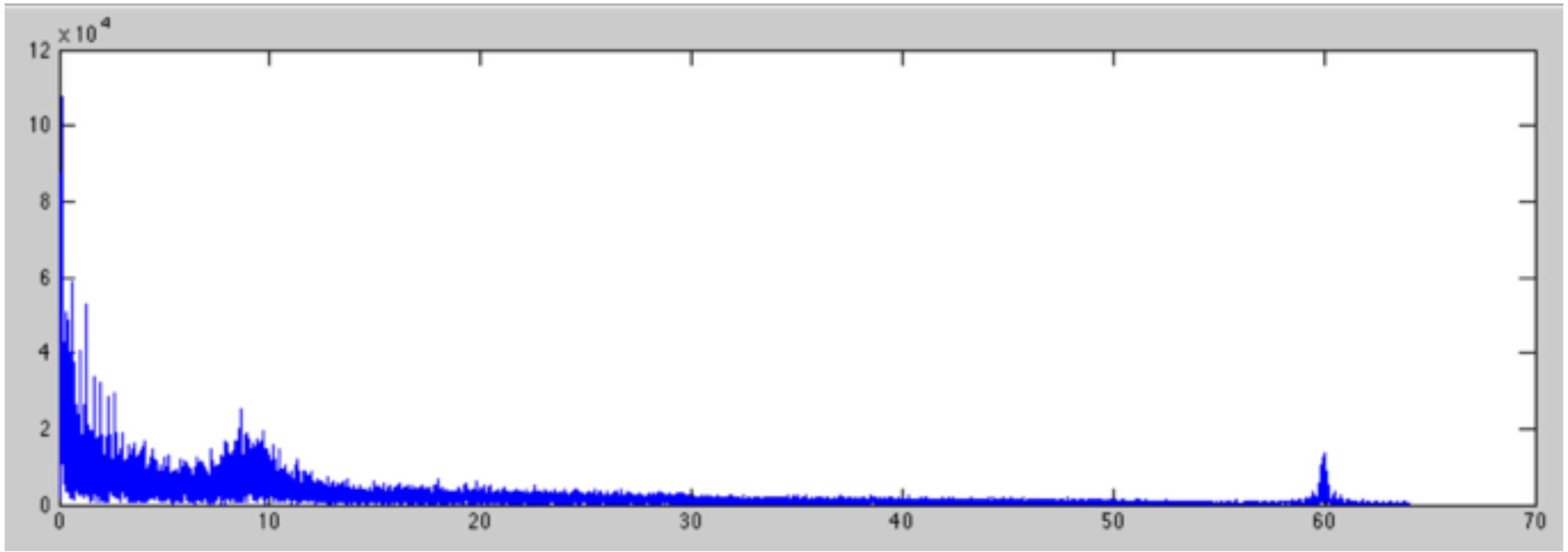
source: wikipedia

Why?

- **Frequency content**



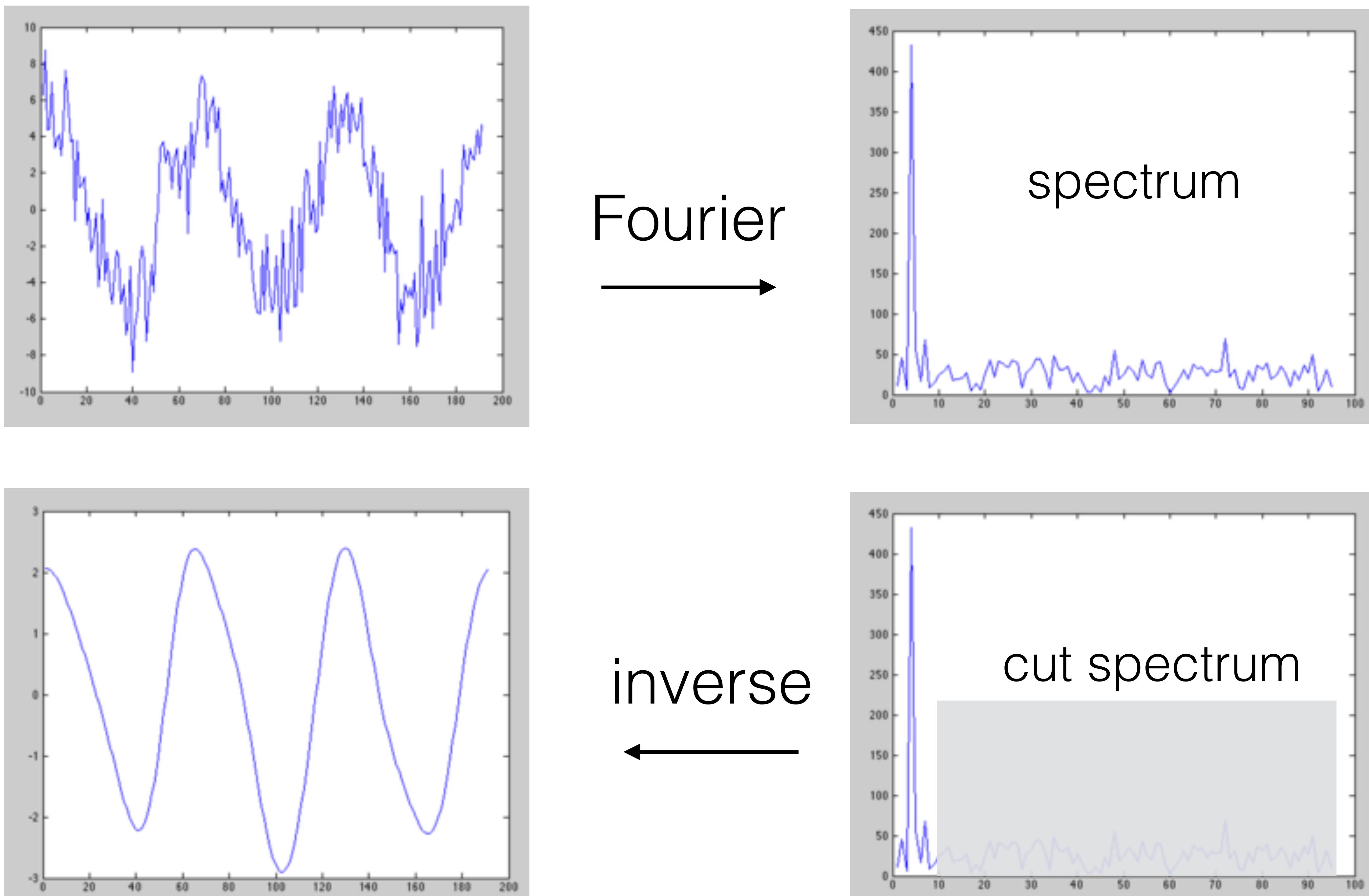
time



freq

Why?

- **Cleaning**

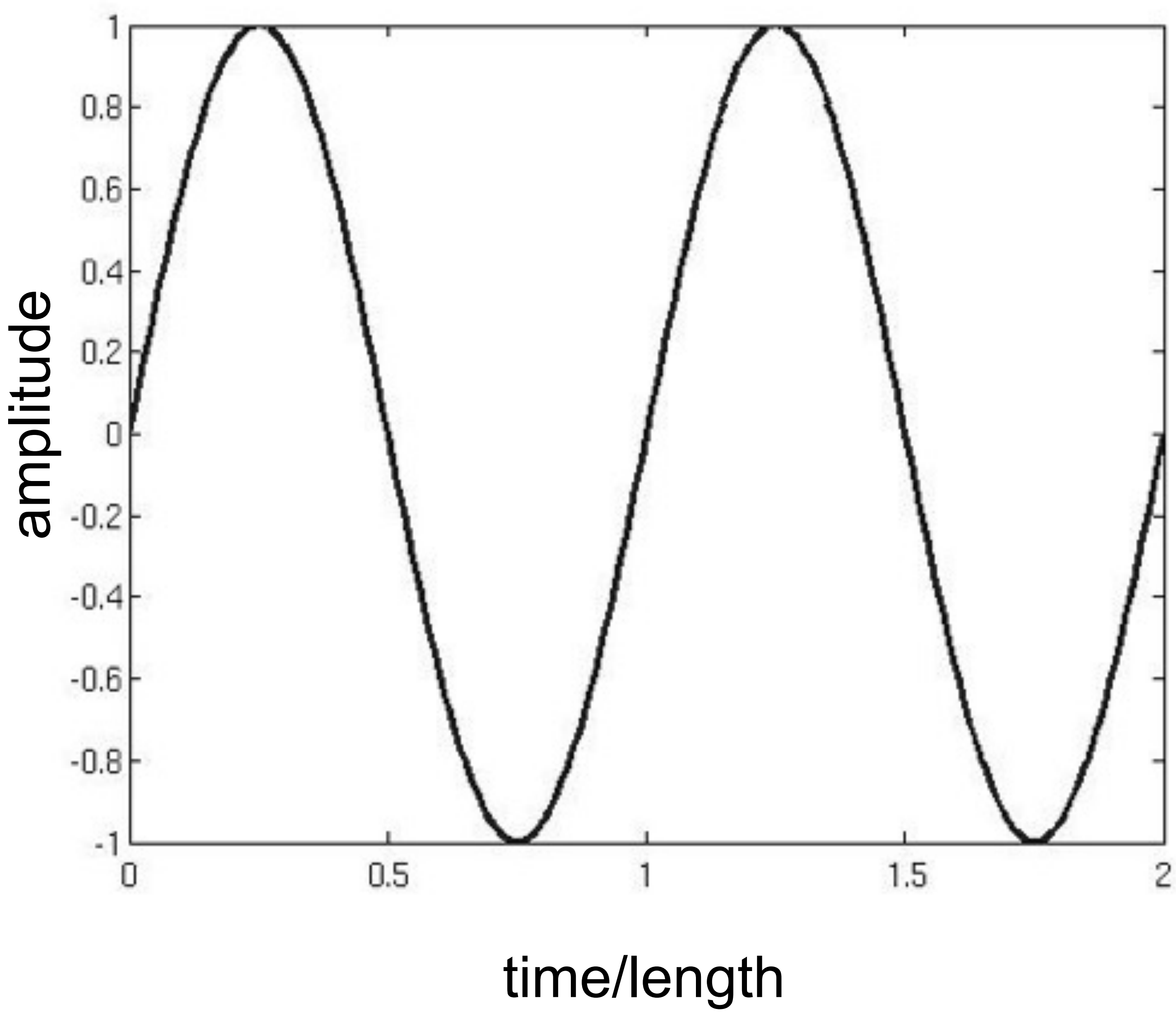


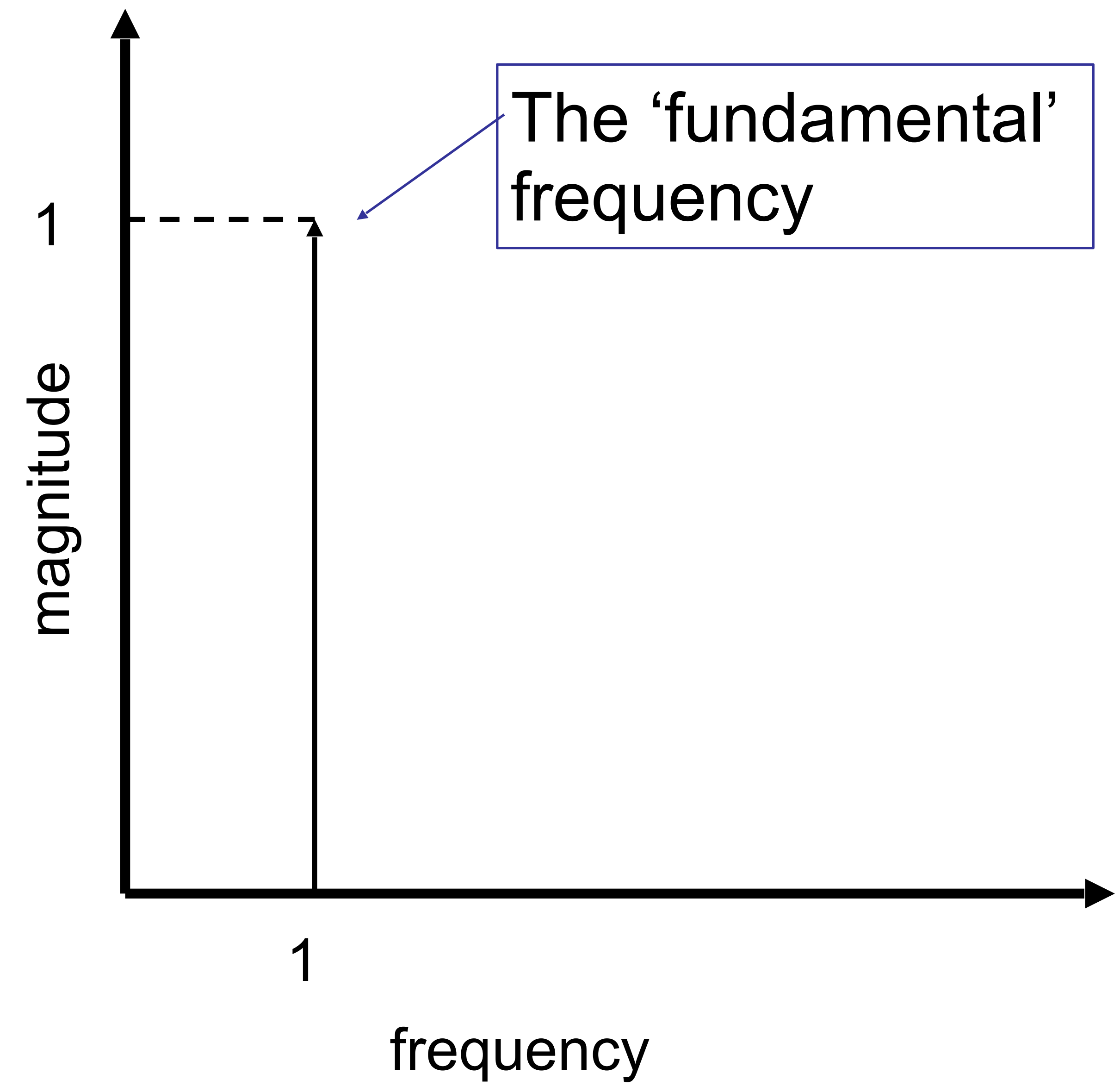
Why?

- Frequency content
- Cleaning (filtering)
- Computational efficiency
- Also very useful in abstract maths (beyond signal processing)

Fourier Series

- Spectrum of the sinusoid: $\boxed{y = \sin(2\pi t)}$

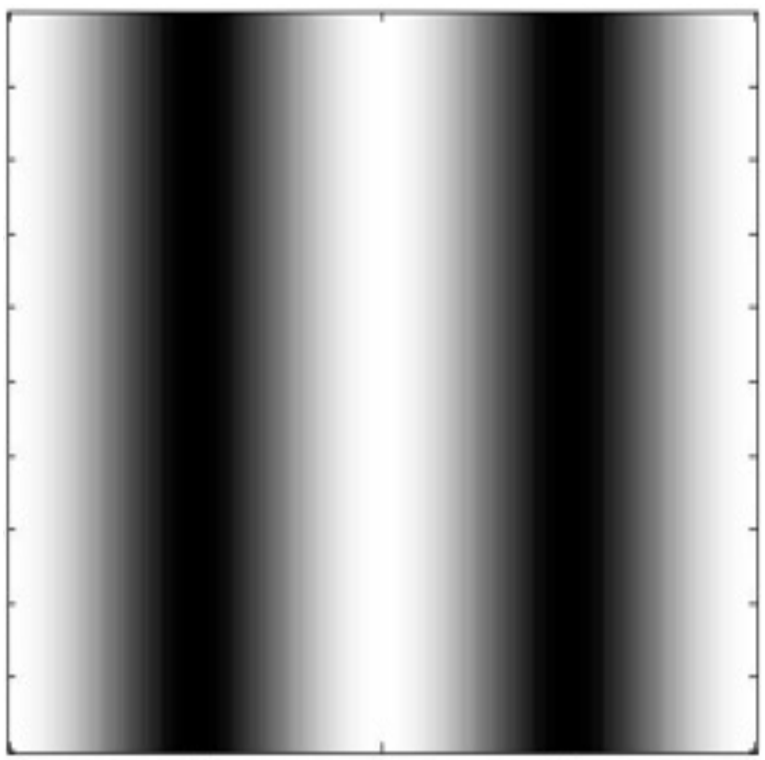




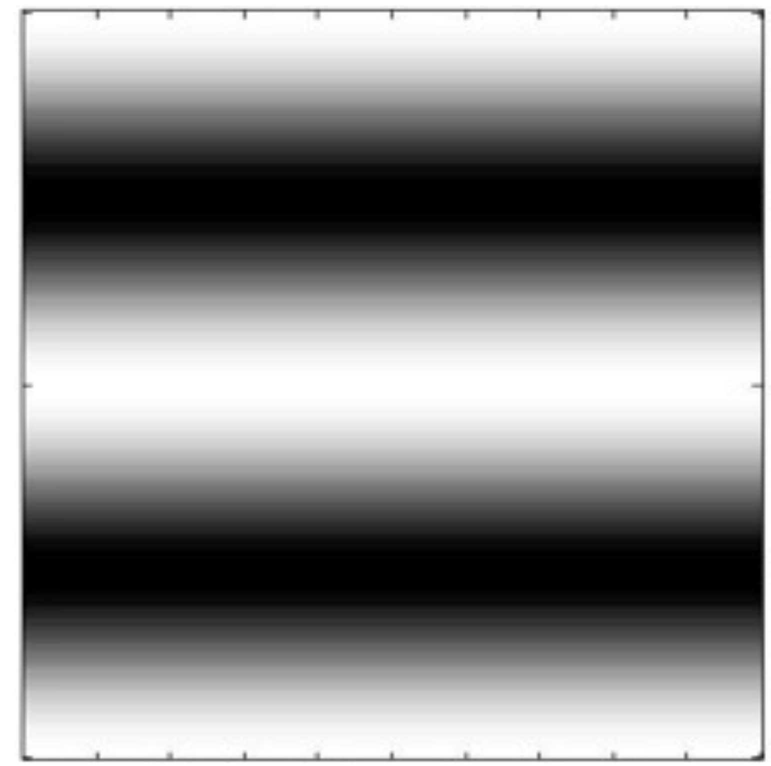
Signal domain ⟷ **Fourier domain**

Frequency - 2D

- Define sinusoid going in *x* and *y* directions.



x frequency = 2
y frequency = 0



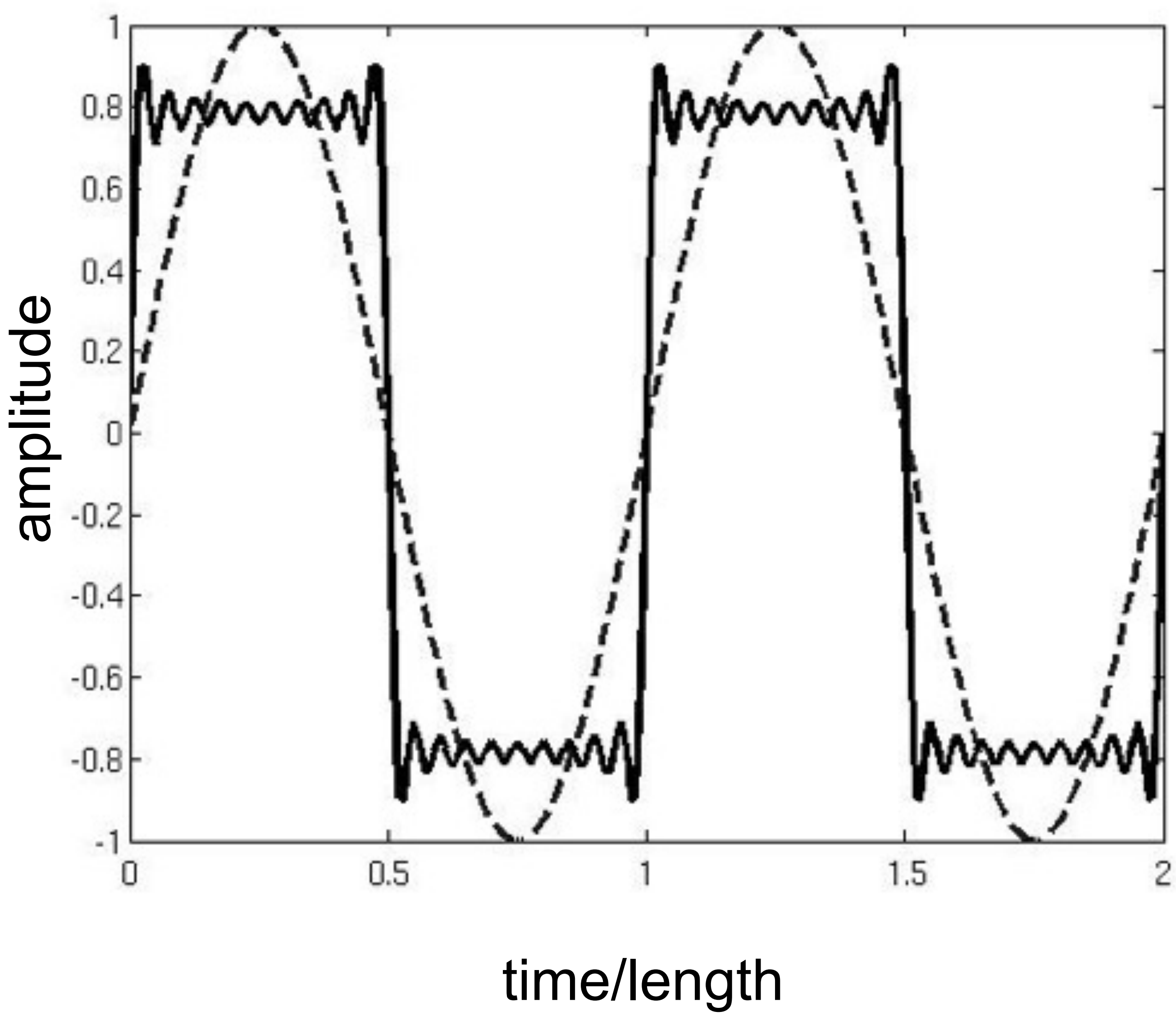
x frequency = 0
y frequency = 2

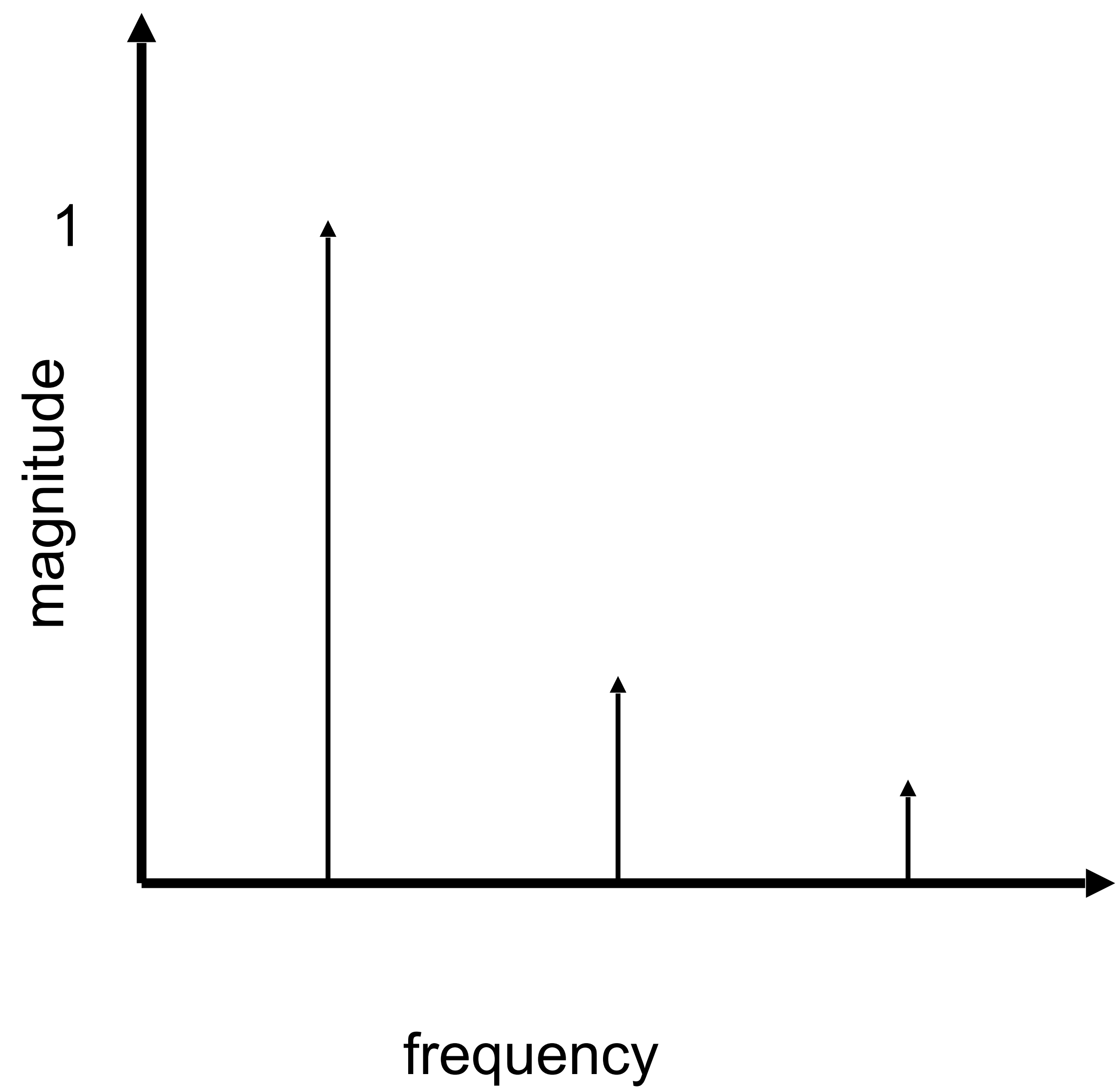
Fourier Series

Approximate **periodic signals** with sines and cosines

Fourier Series

- Keep going: $$y(t) = \sum_n a_n \sin(nf \times 2\pi t) + \sum_n b_n \cos(nf \times 2\pi t)$$





Signal ⟷ **Frequency Spectrum**

Fourier Analysis
what's it all about?

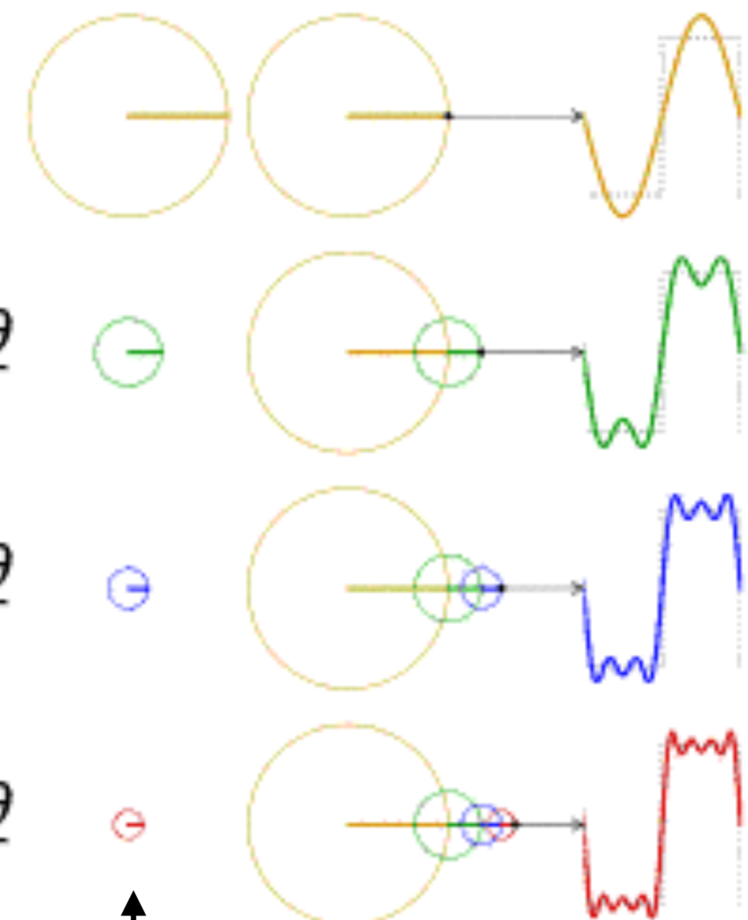
- **Sines and cosines are building blocks**

fundamental frequency $\frac{4\sin\theta}{\pi}$

harmonic $\frac{4\sin 3\theta}{3\pi}$

harmonic $\frac{4\sin 5\theta}{5\pi}$

harmonic $\frac{4\sin 7\theta}{7\pi}$



magnitude

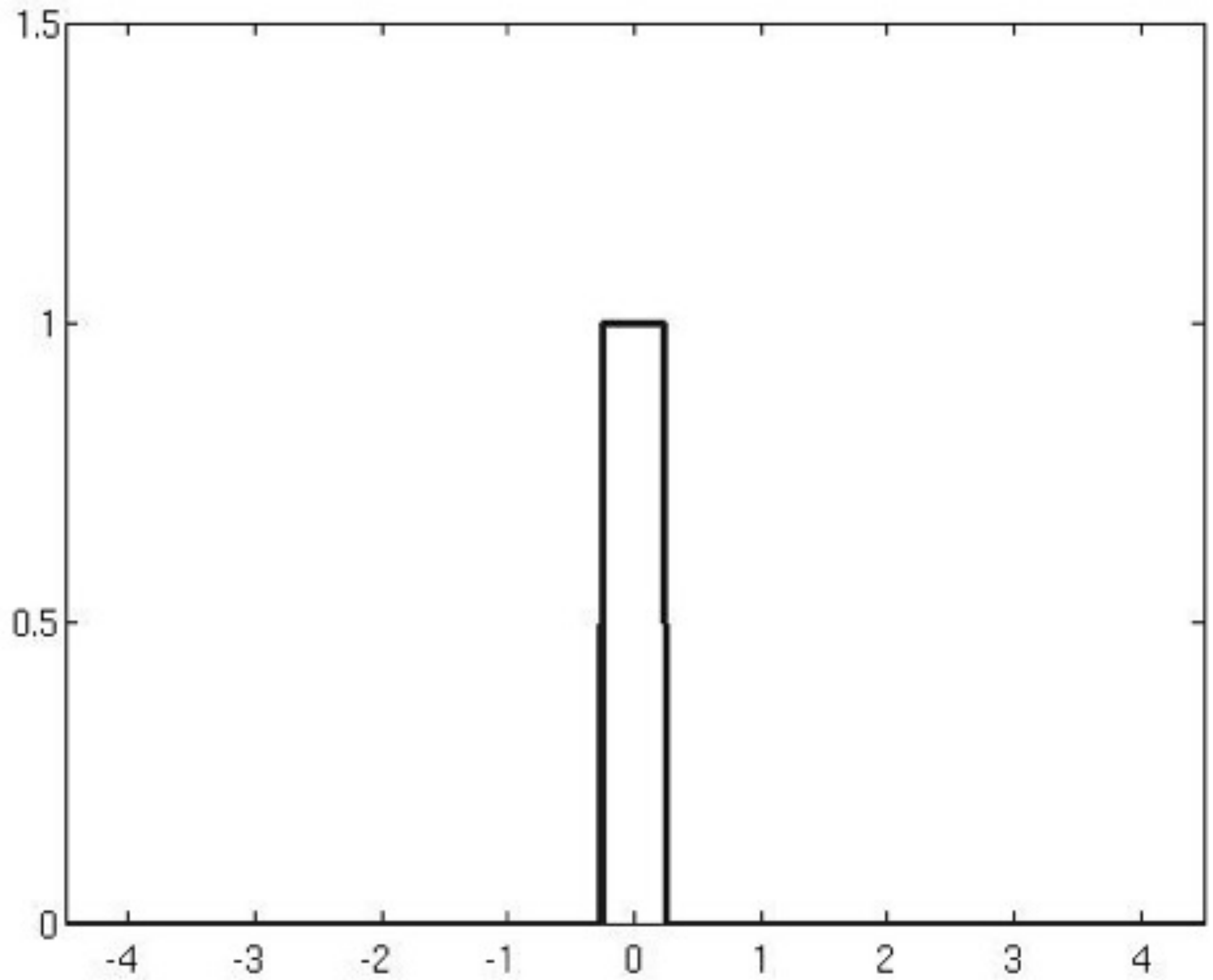
details increasing
with frequency

Fourier Transform

Approximate **non-periodic signals** with sines and cosines

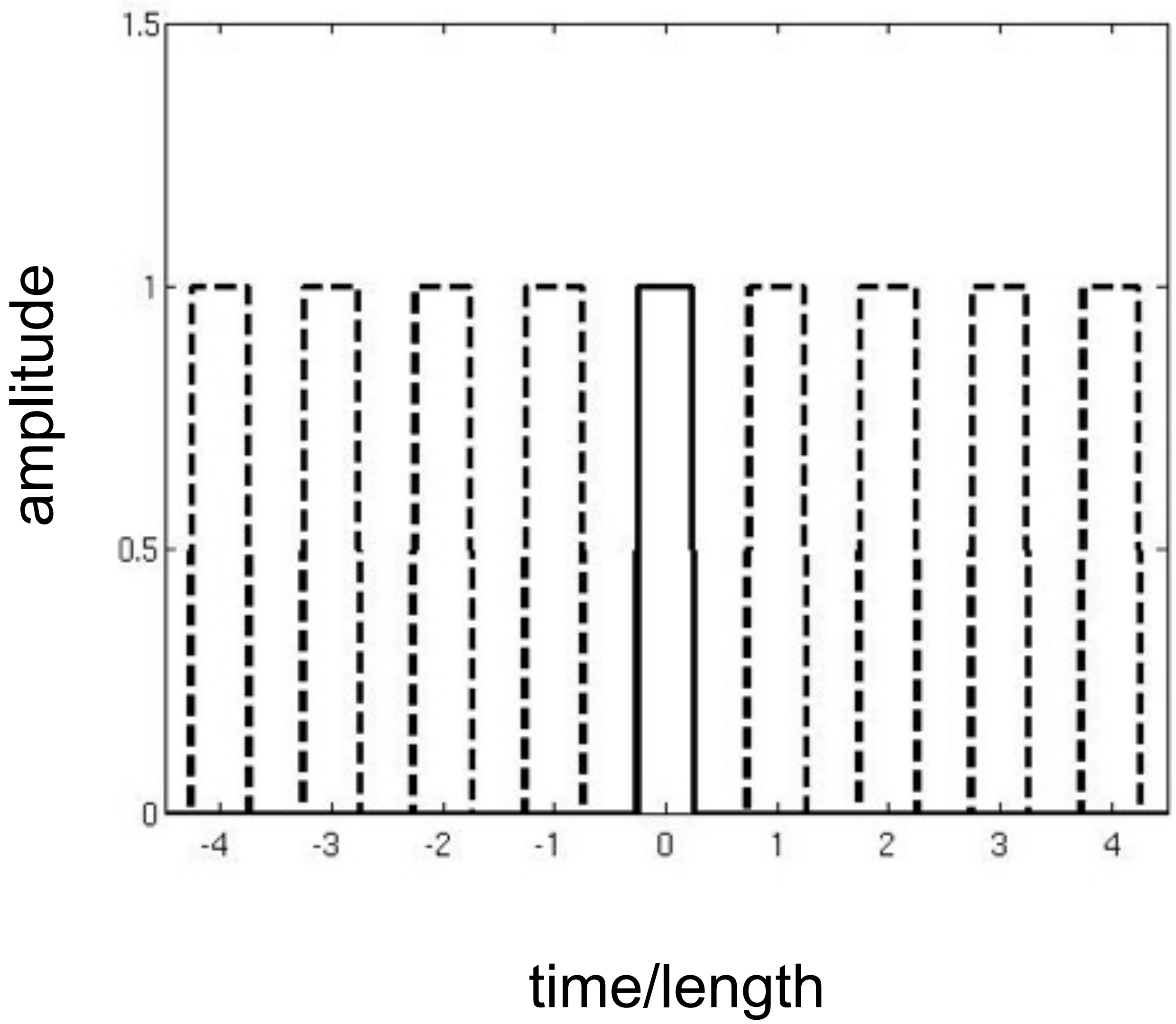
Fourier Series --> Transforms

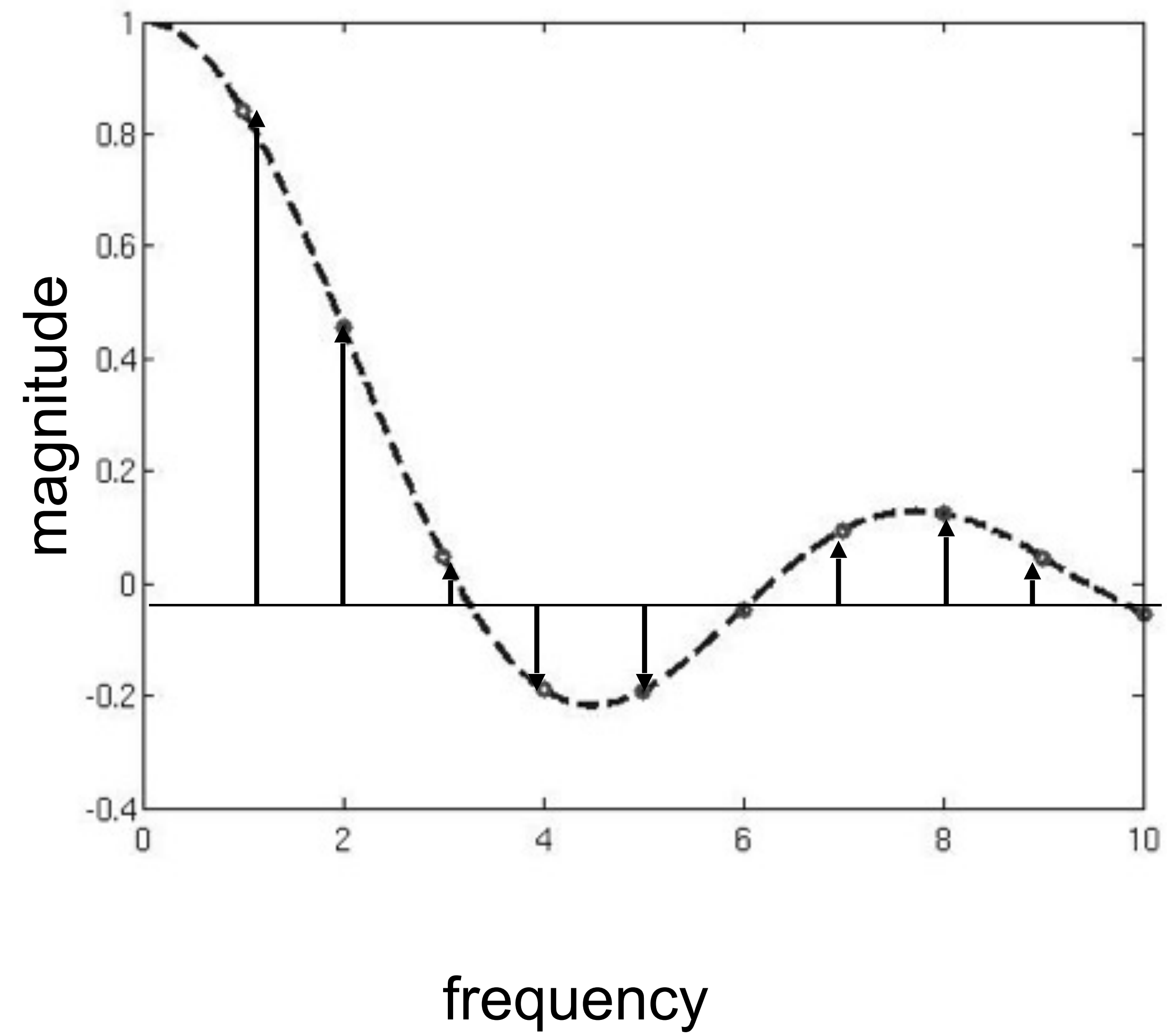
- Fourier Series good for periodic signals, what do we do if they are not?
  - Classic example: the top hat function



Fourier Series --> Transforms

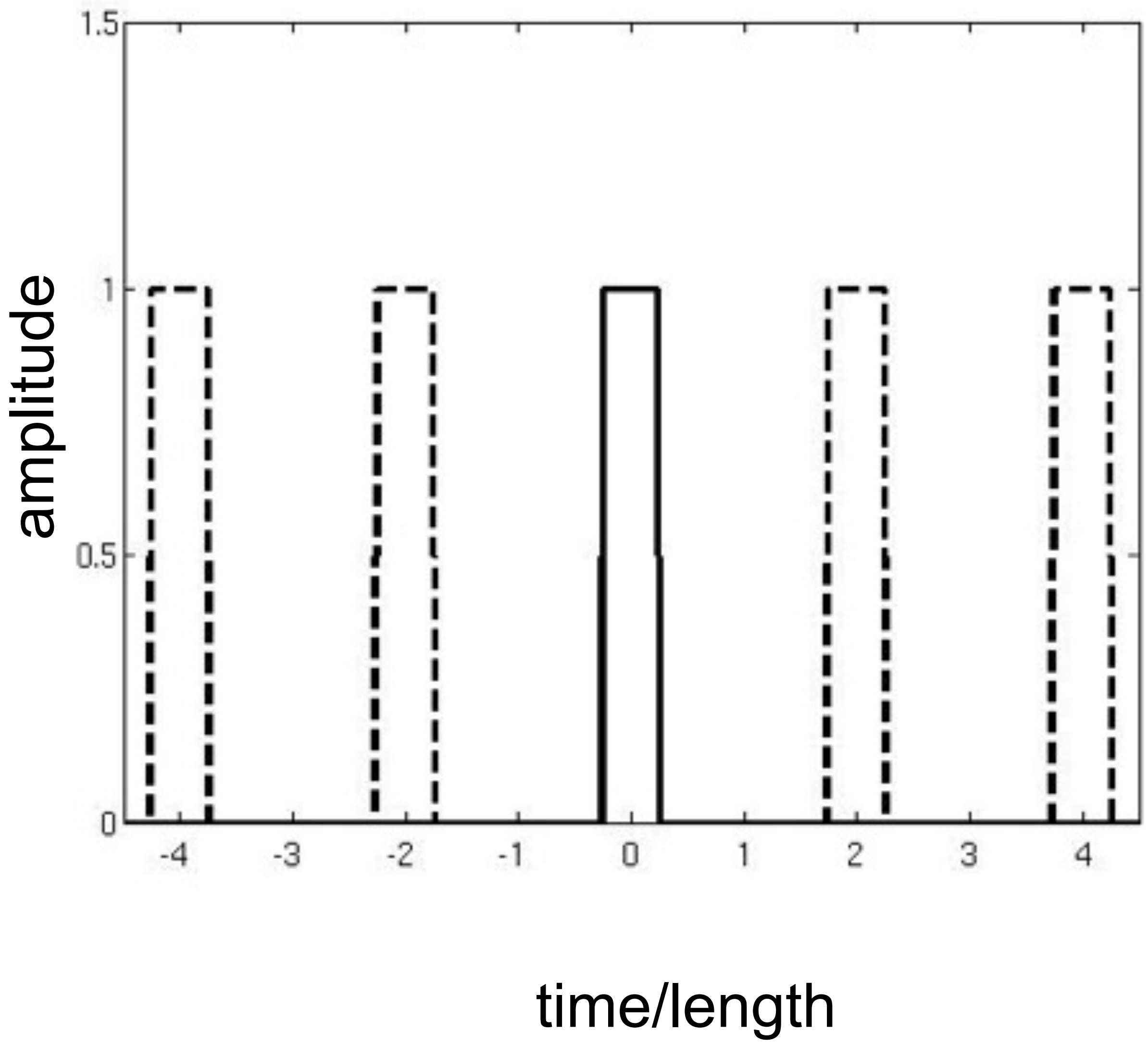
- Try making it periodic over a certain period:

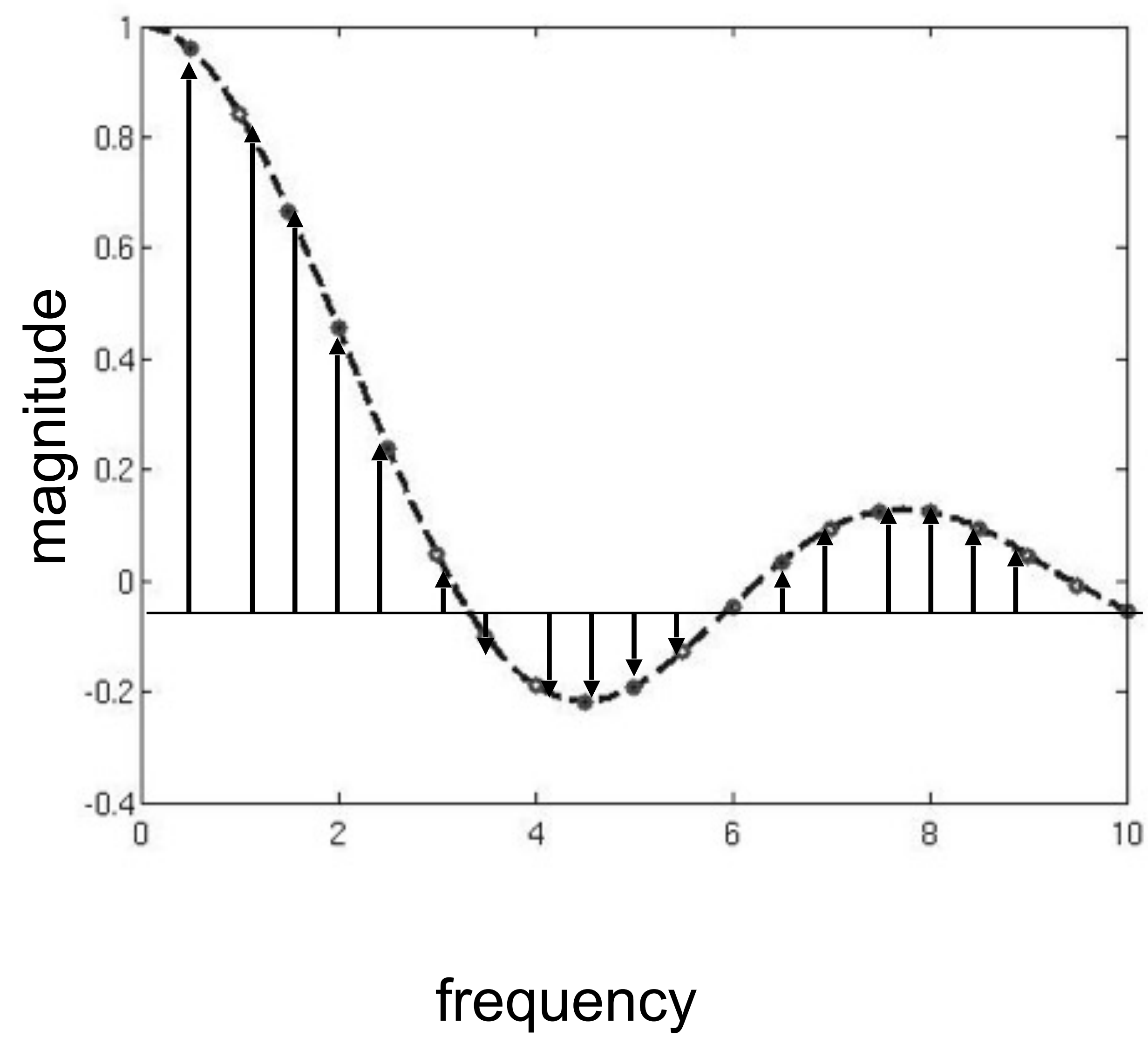




Fourier Series --> Transforms

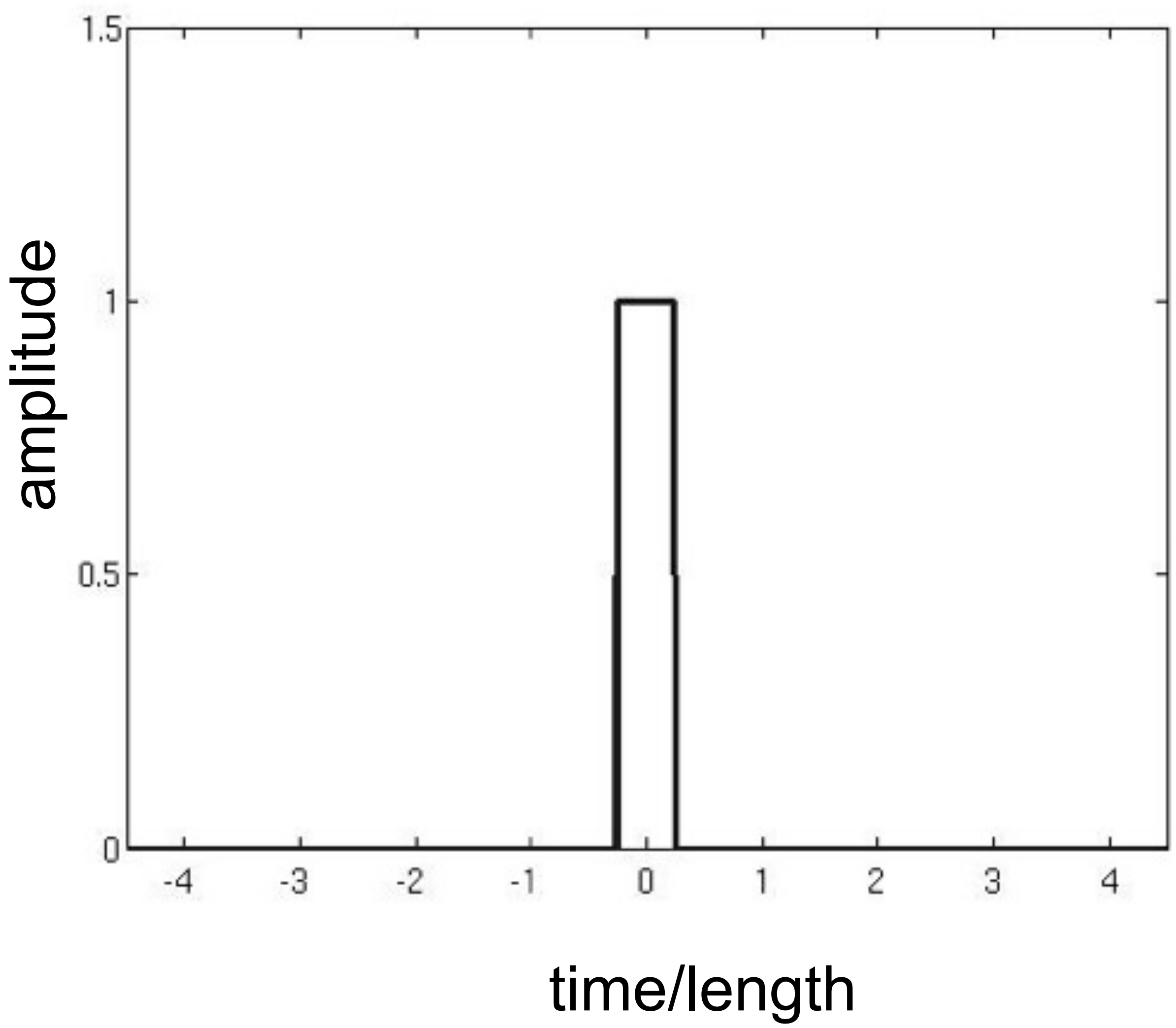
- Make the spacing larger, so that there is a larger space in which only the top hat appears:

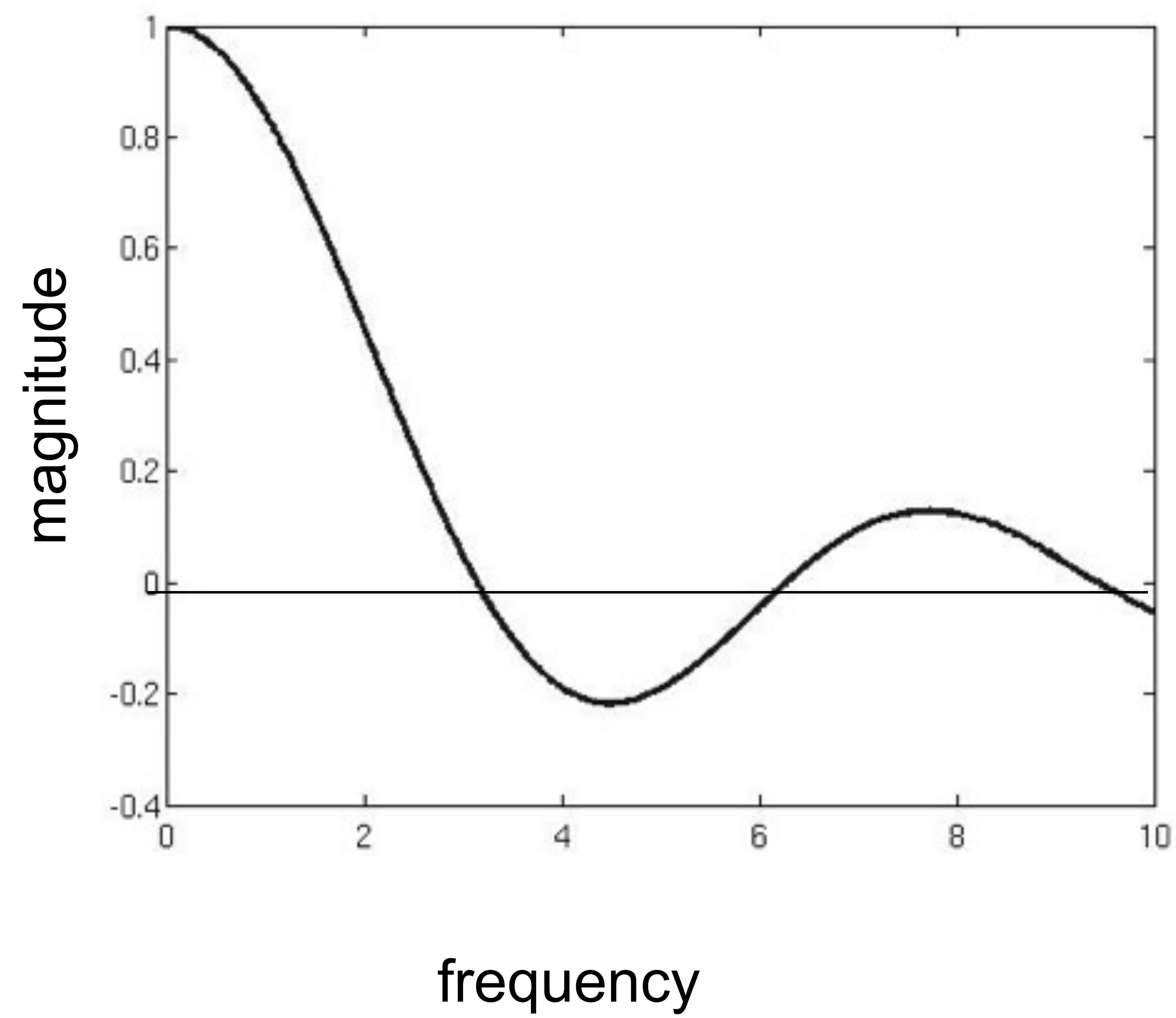




Fourier Series --> Transforms

- In the limit (T = infinity) we get a smooth spectrum:





Fourier Transform

- Fourier series was a sum at specific frequencies:

$$y(t) = \sum_n a_n \sin(nf \times 2\pi t) + \sum_n b_n \cos(nf \times 2\pi t)$$

- Fourier transform is a sum over all frequencies:

$$y(t) = \int_{\boxed{-\infty}}^{\infty} A(f) \boxed{e^{\boxed{j2\pi f}\, t}} df$$

Frequency

Negative frequencies

Sine/Cosine (compact notation)

  - Note: this formula is usually called the inverse FT.

Fourier Transform

- To find the frequency spectrum we need to do an integral:

$$A(f) = \int_{-\infty}^{\infty} y(t) e^{-j2\pi f\ t} dt$$

Basically: at frequency *f*, how much of sin(*f* x 2π*t*) do I need?

- In practice we get computers do deal with this using the Fast Fourier Transform (FFT).

Fourier transform

Mathematical operation

Continuous

Fast Fourier Transform

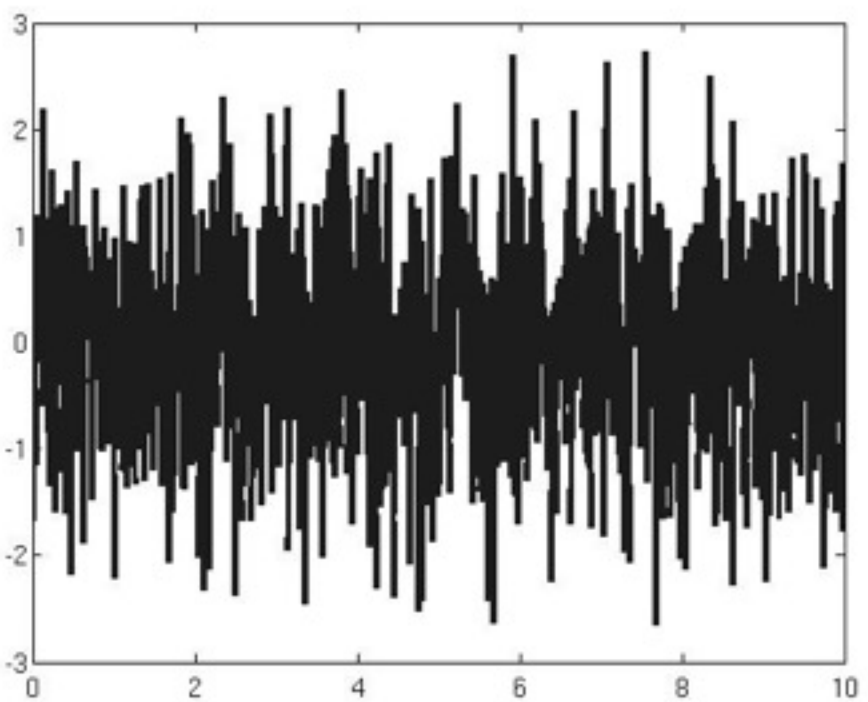
Algorithm

Discrete

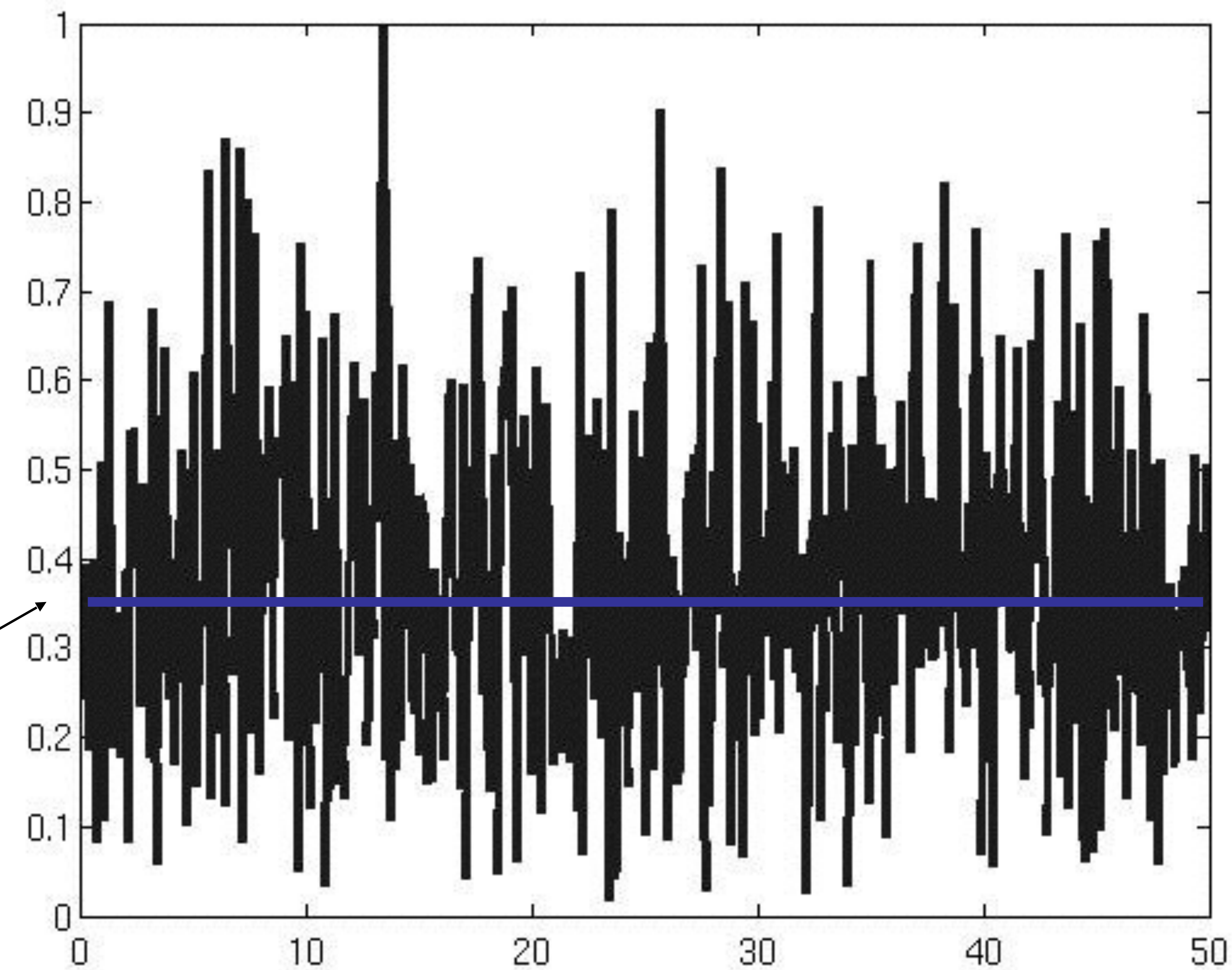
Fourier Transform

- Examples:

'White' noise



FFT Result

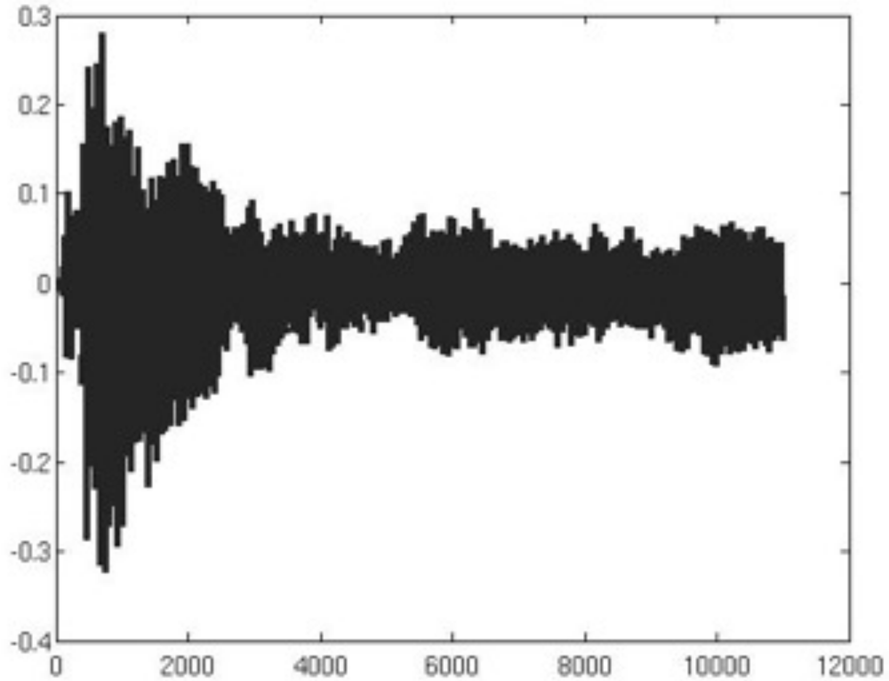


Ideally flat -
all frequencies

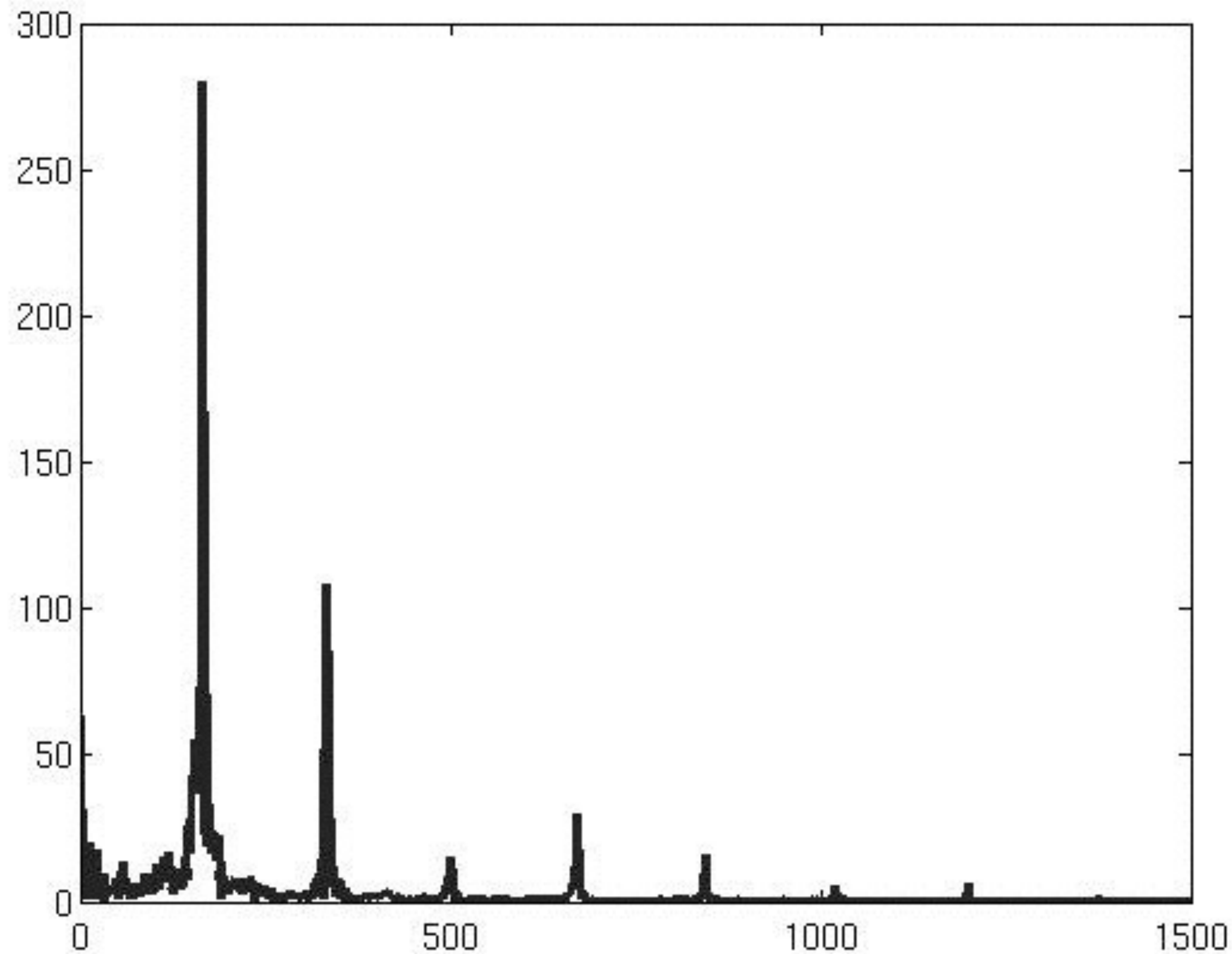
Fourier Transform - 1D

- Examples:

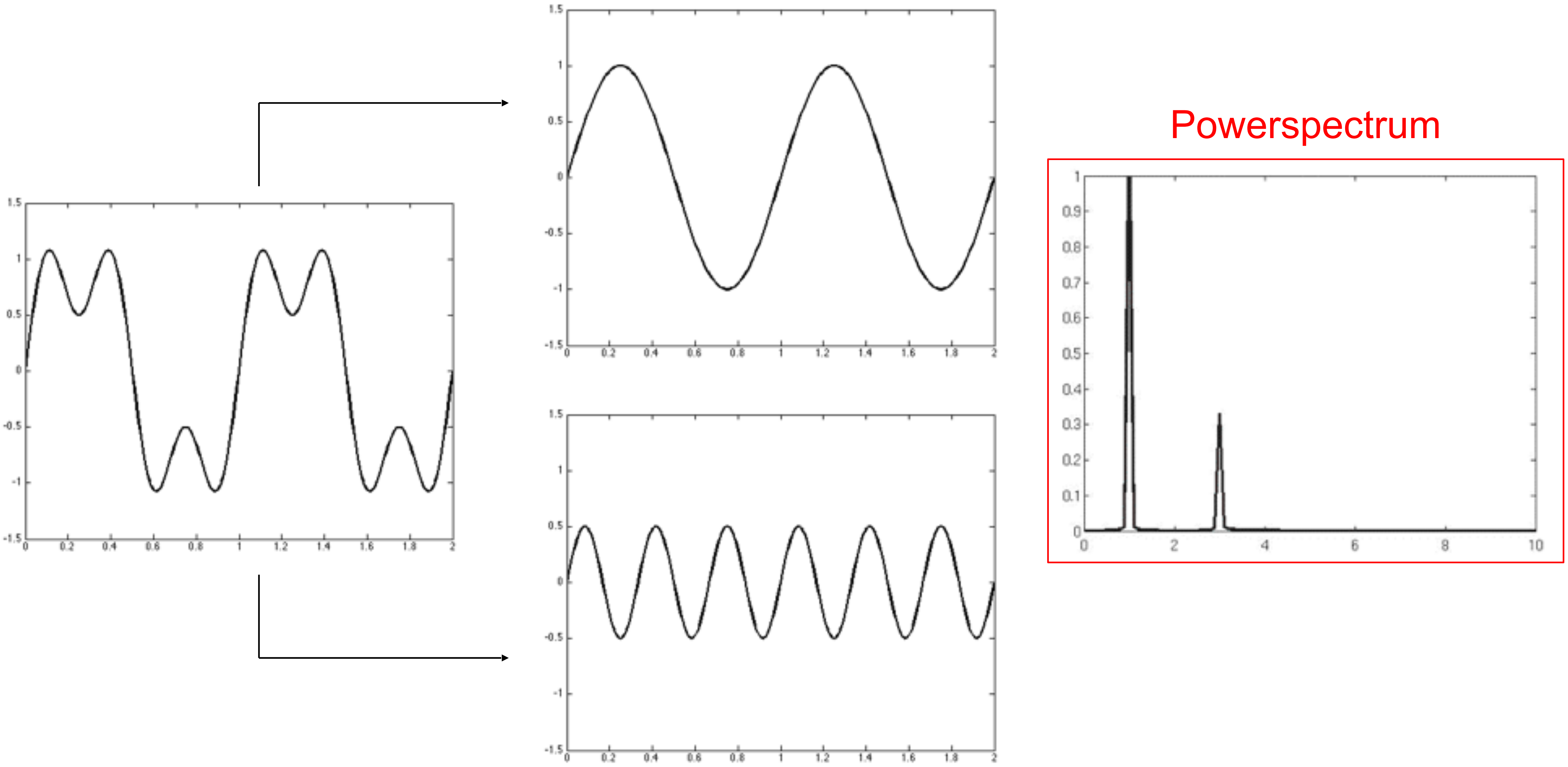
Piano note



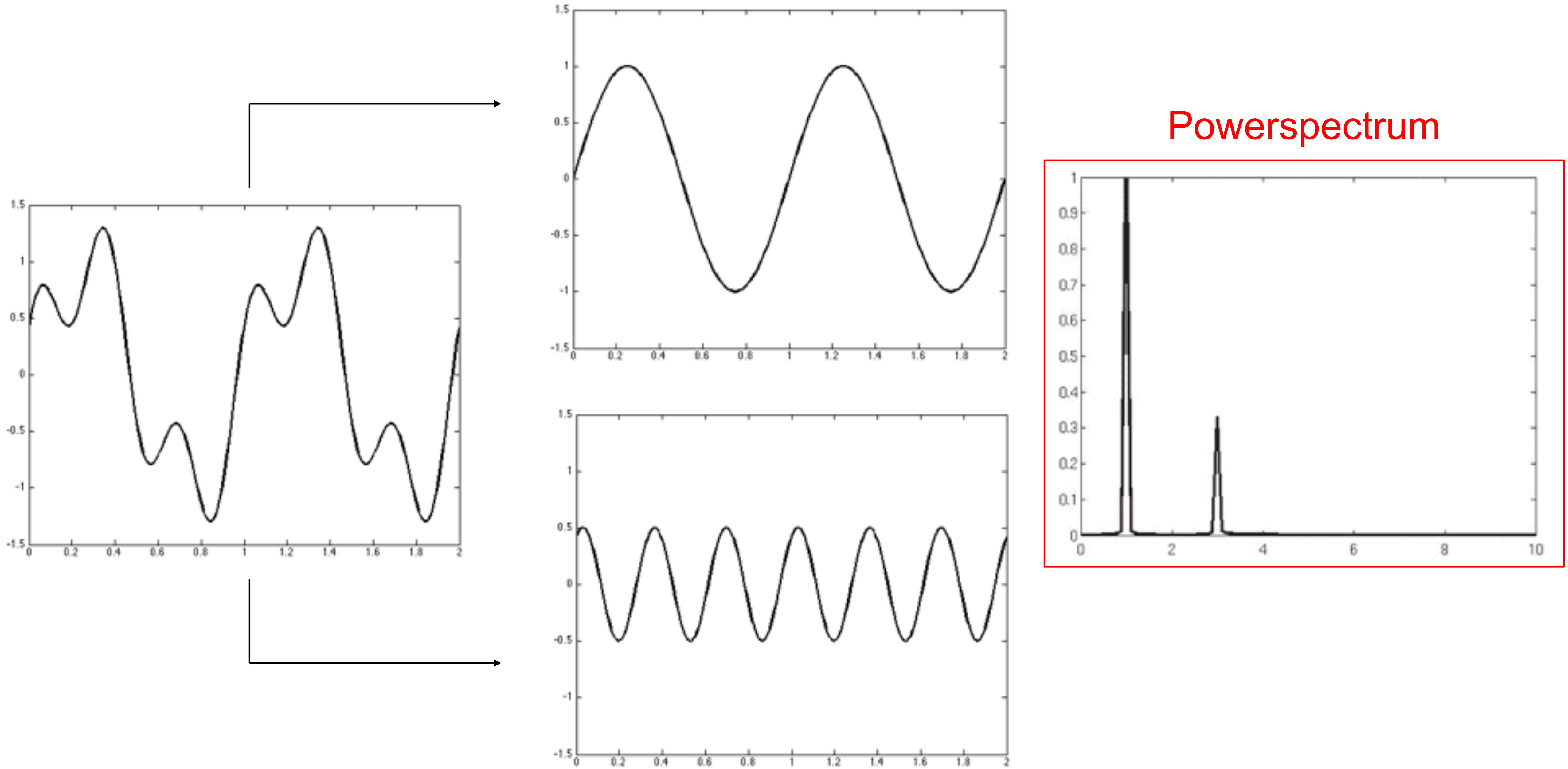
FFT Result



Fourier transform



Power spectrum lacks phase information



Fourier transform

signal

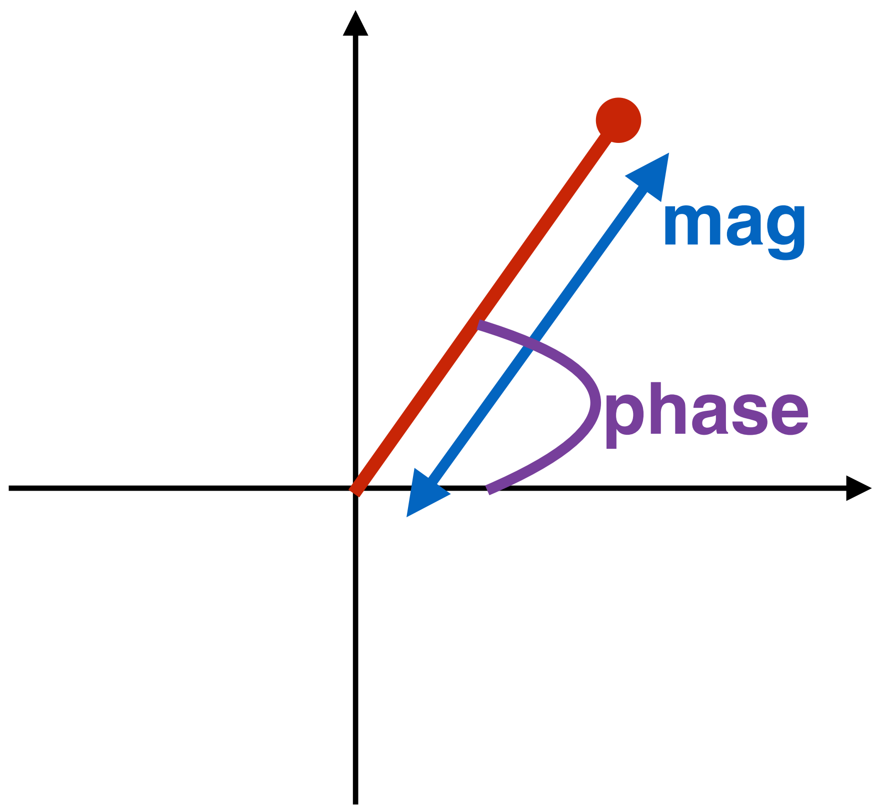
$$A(f) = \int_{-\infty}^{\infty} y(t)\, e^{-j2\pi f t}\, dt$$

complex number

complex number

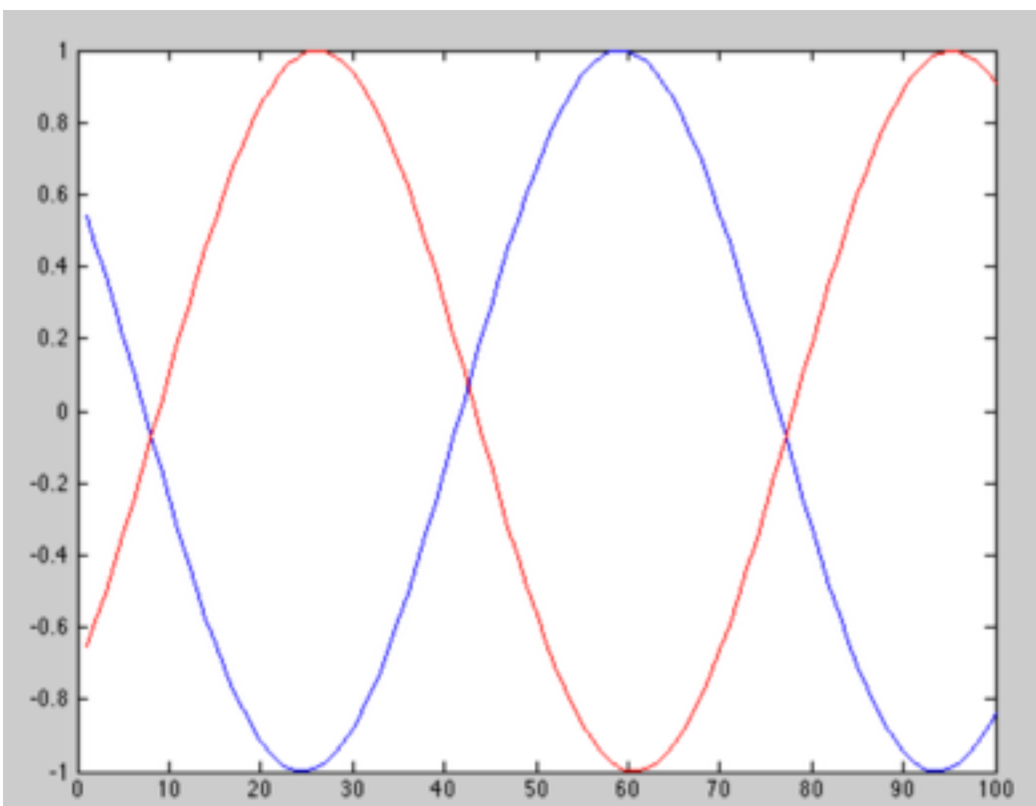
Complex numbers easier to manipulate than sine and cosine functions

Information on phase and magnitude



Magnitude, Phase

What? Where?

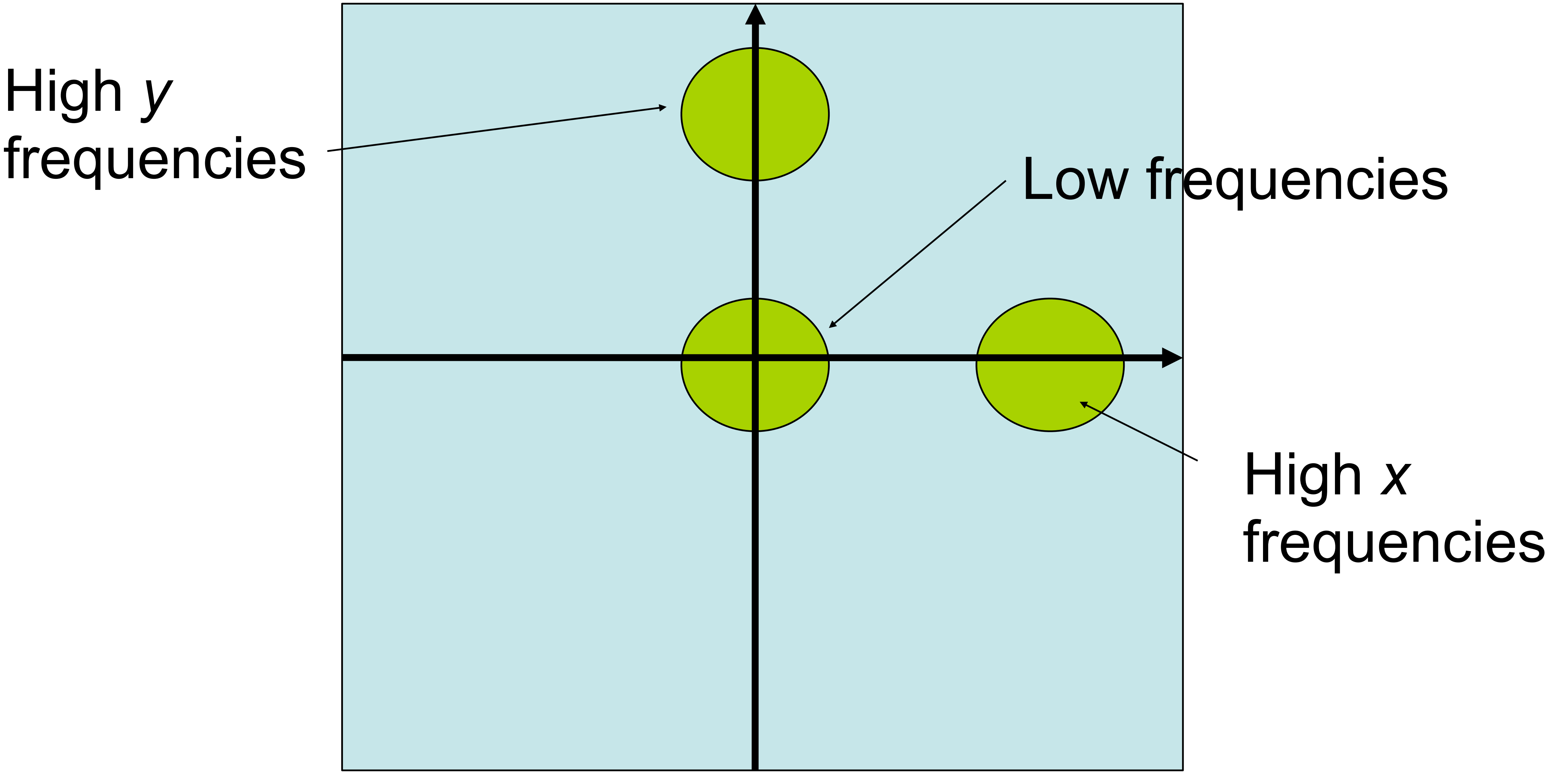


FFT → Same magnitude

Different phases

Fourier Transform - 2D

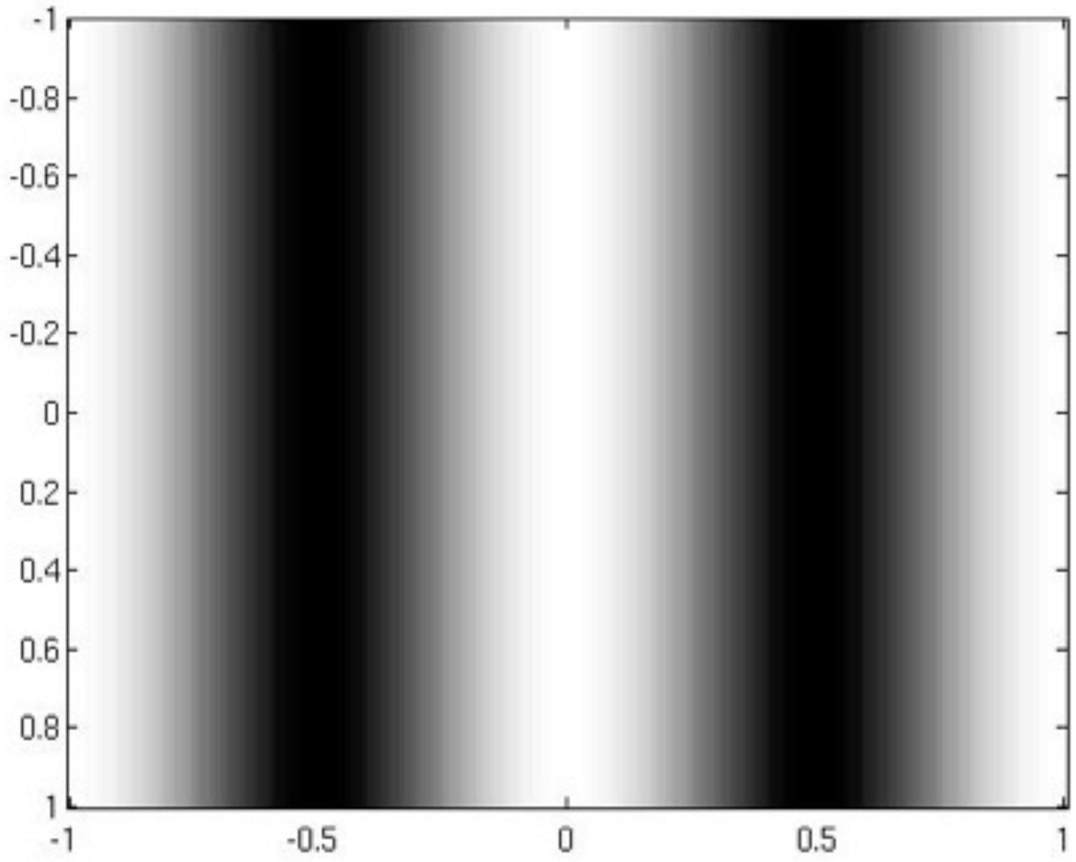
- 2D signals, i.e. images have 2D Fourier transforms
- We now have *x* and *y* frequencies:



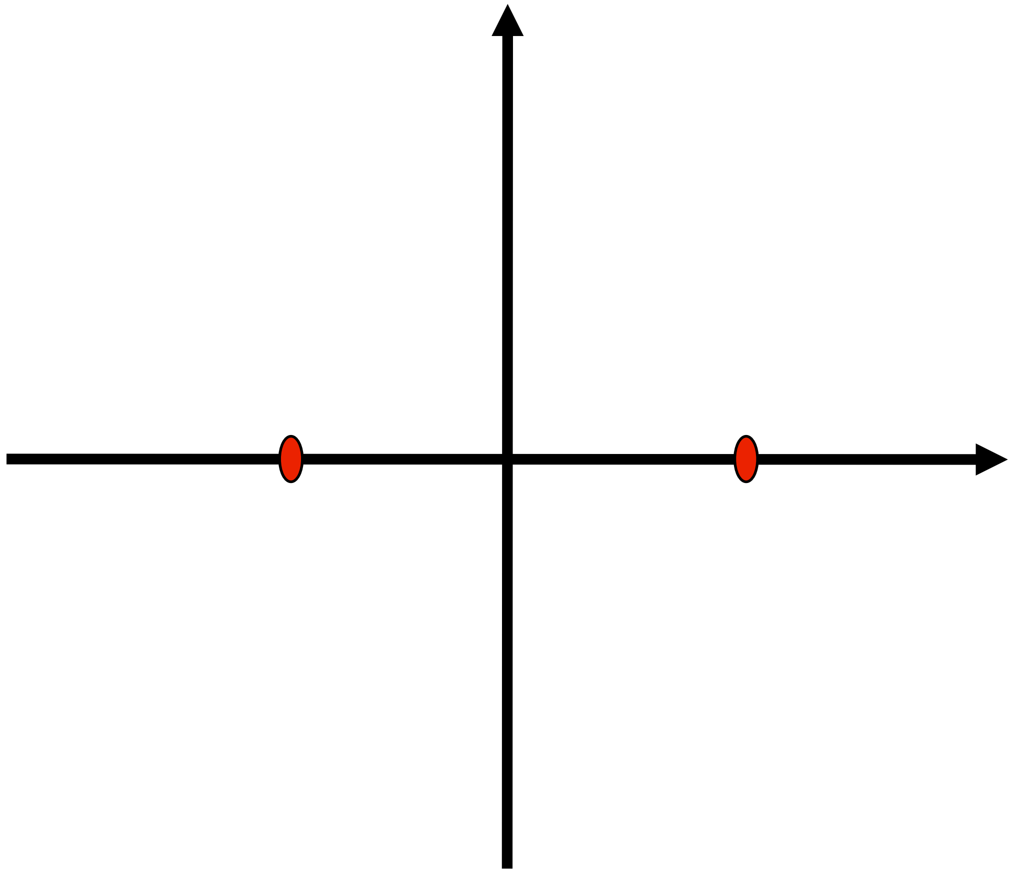
Fourier Transform - 2D

- Sinusoid in x direction

Image



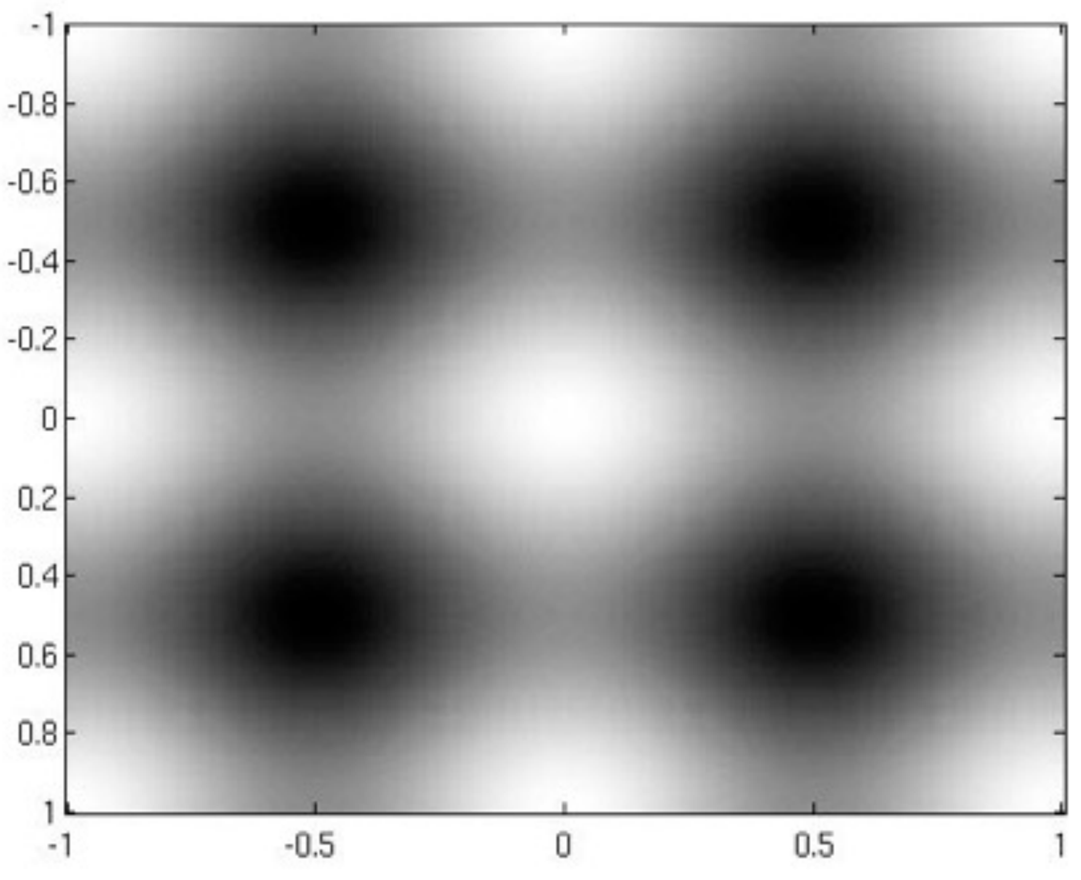
2D spectrum



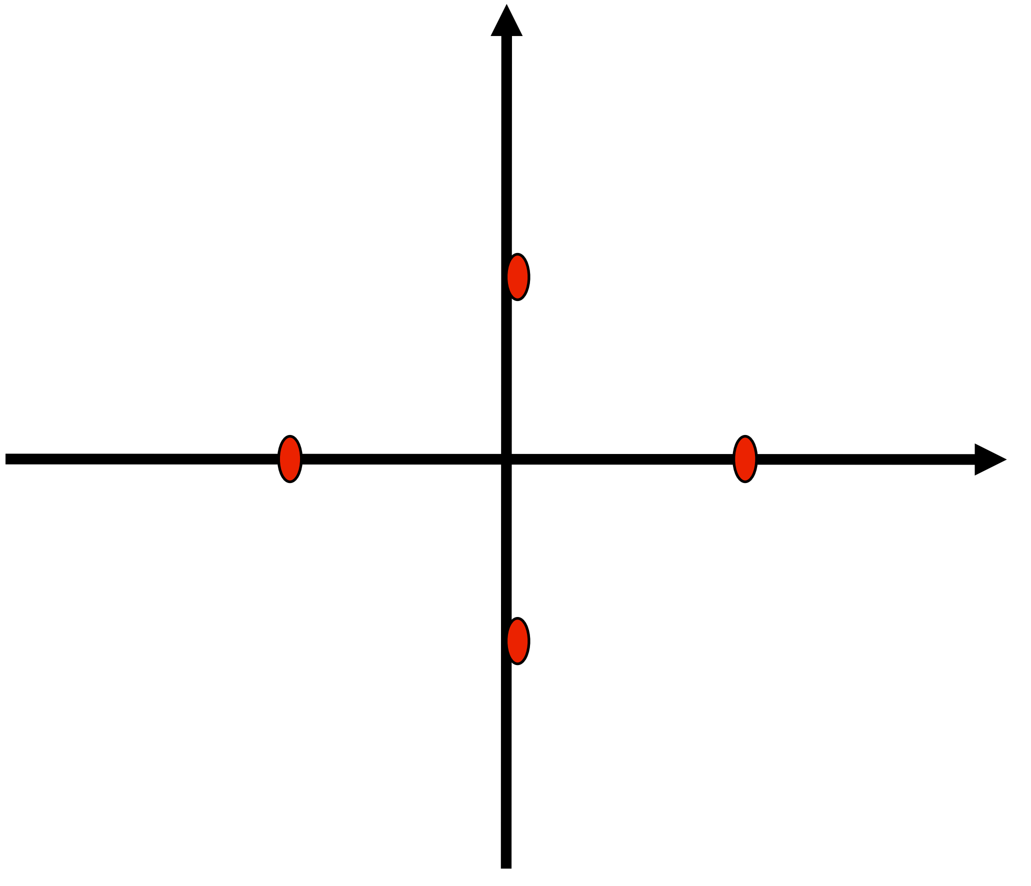
Fourier Transform - 2D

- Sinusoid in x plus sinusoid in y direction

Image



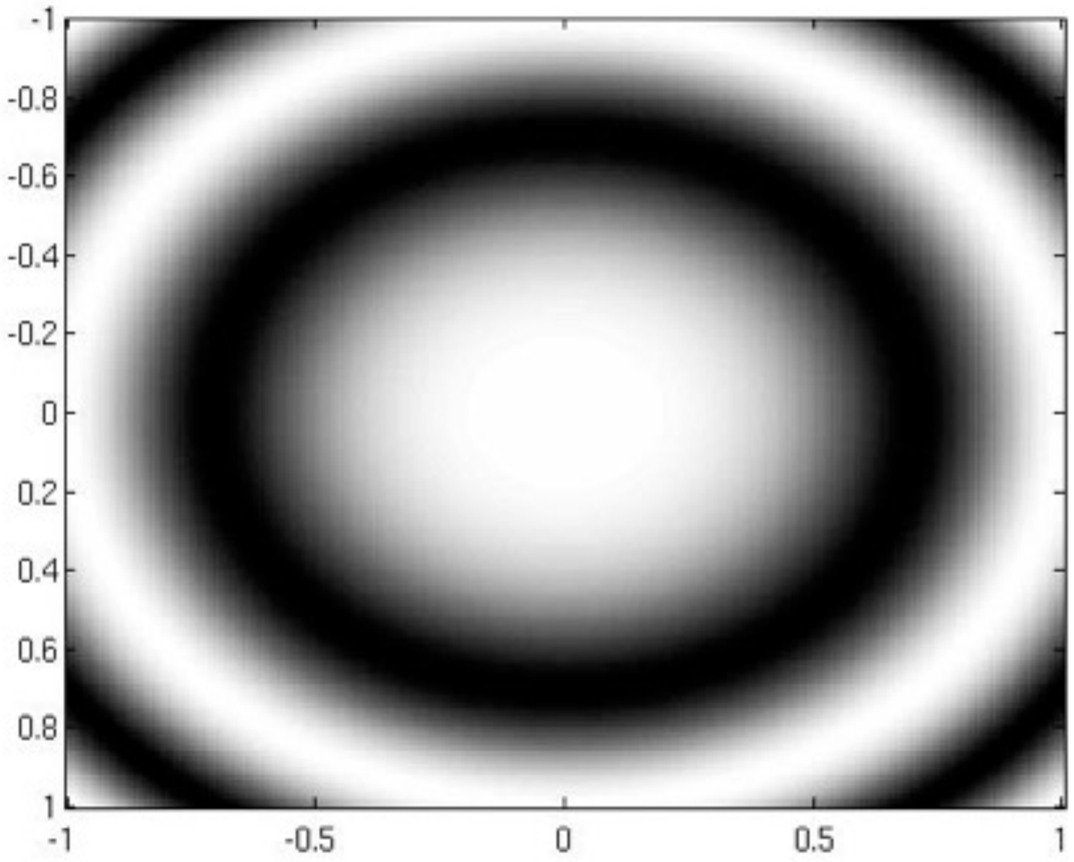
2D spectrum



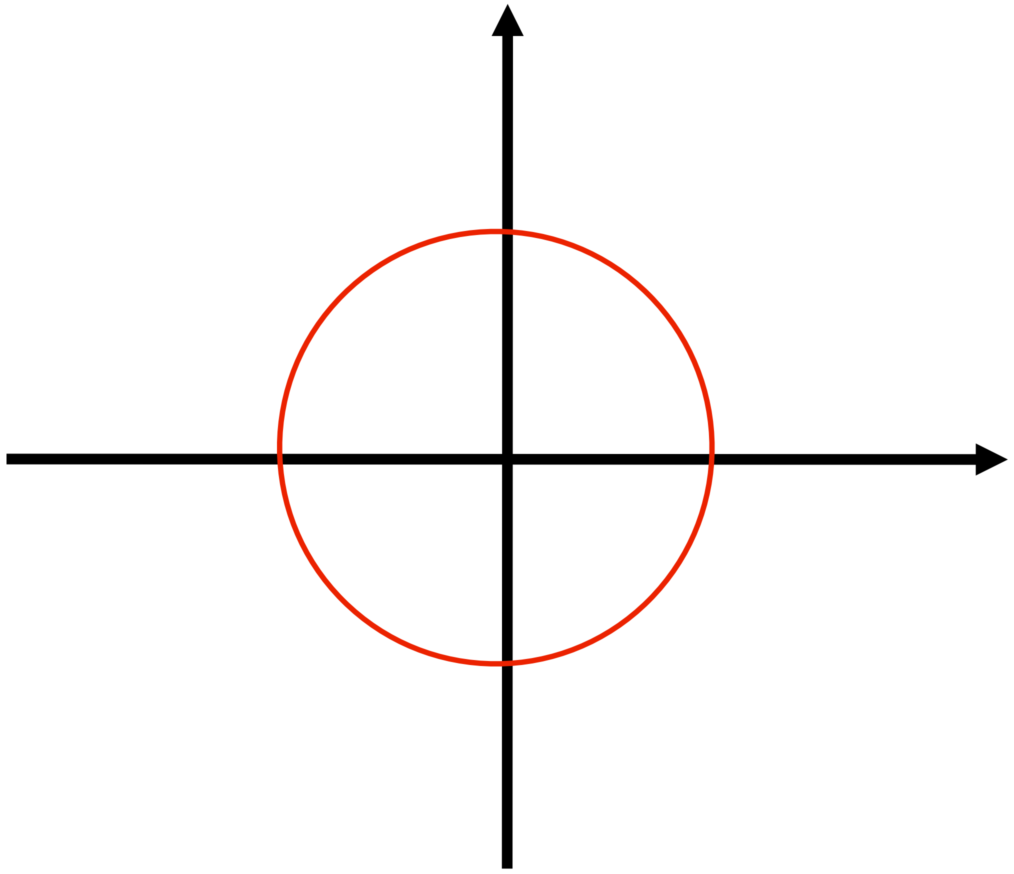
Fourier Transform - 2D

- Single frequency in all directions

Image



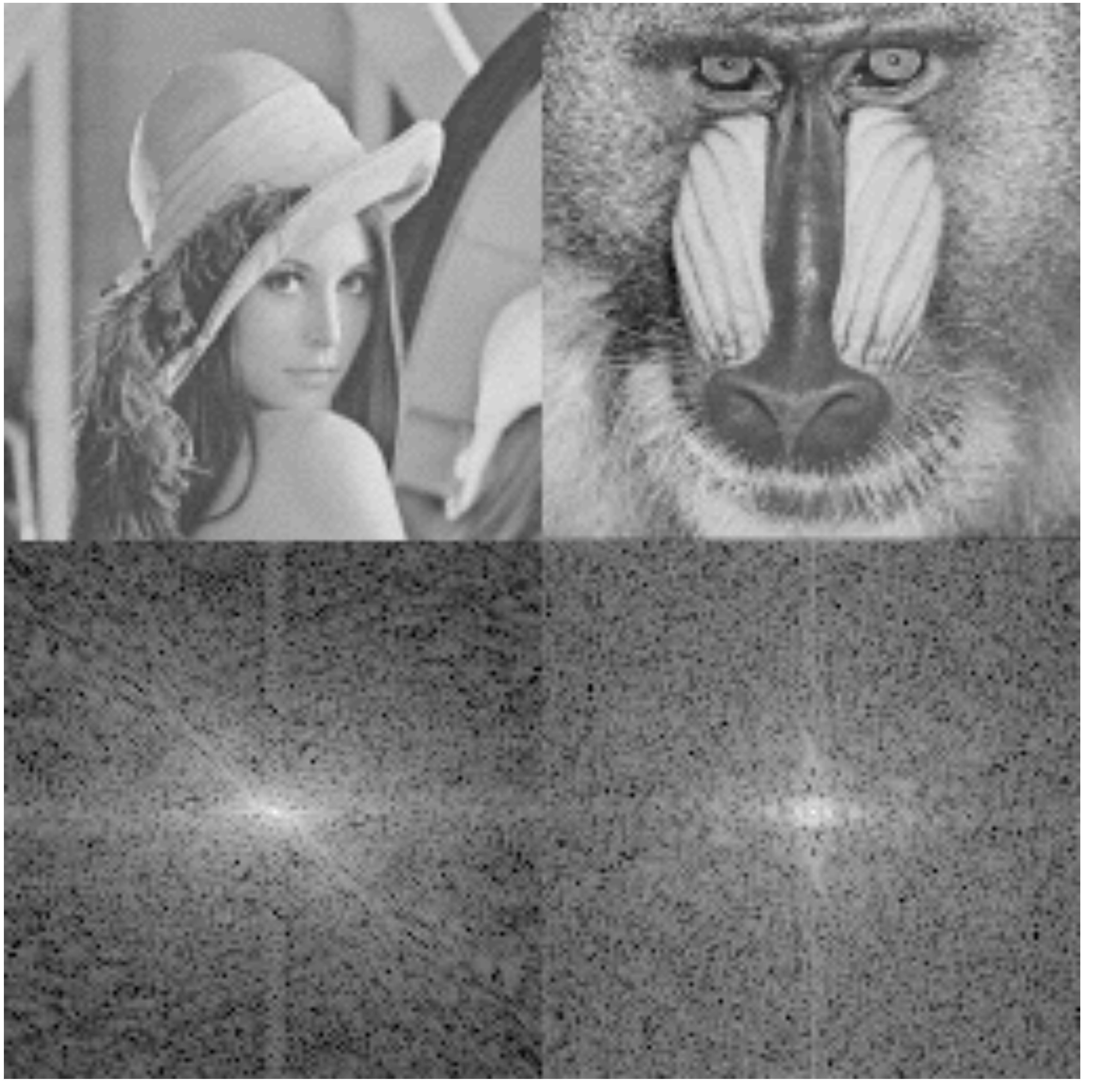
2D spectrum



Fourier Transform - 2D

- Examples:

- Spectrum is ‘bright’ in the centre.
- Detail involves high frequency.
- Spectrum is symmetric.



http://www.cs.unm.edu/~brayer/vision/fourier.html

Image 1

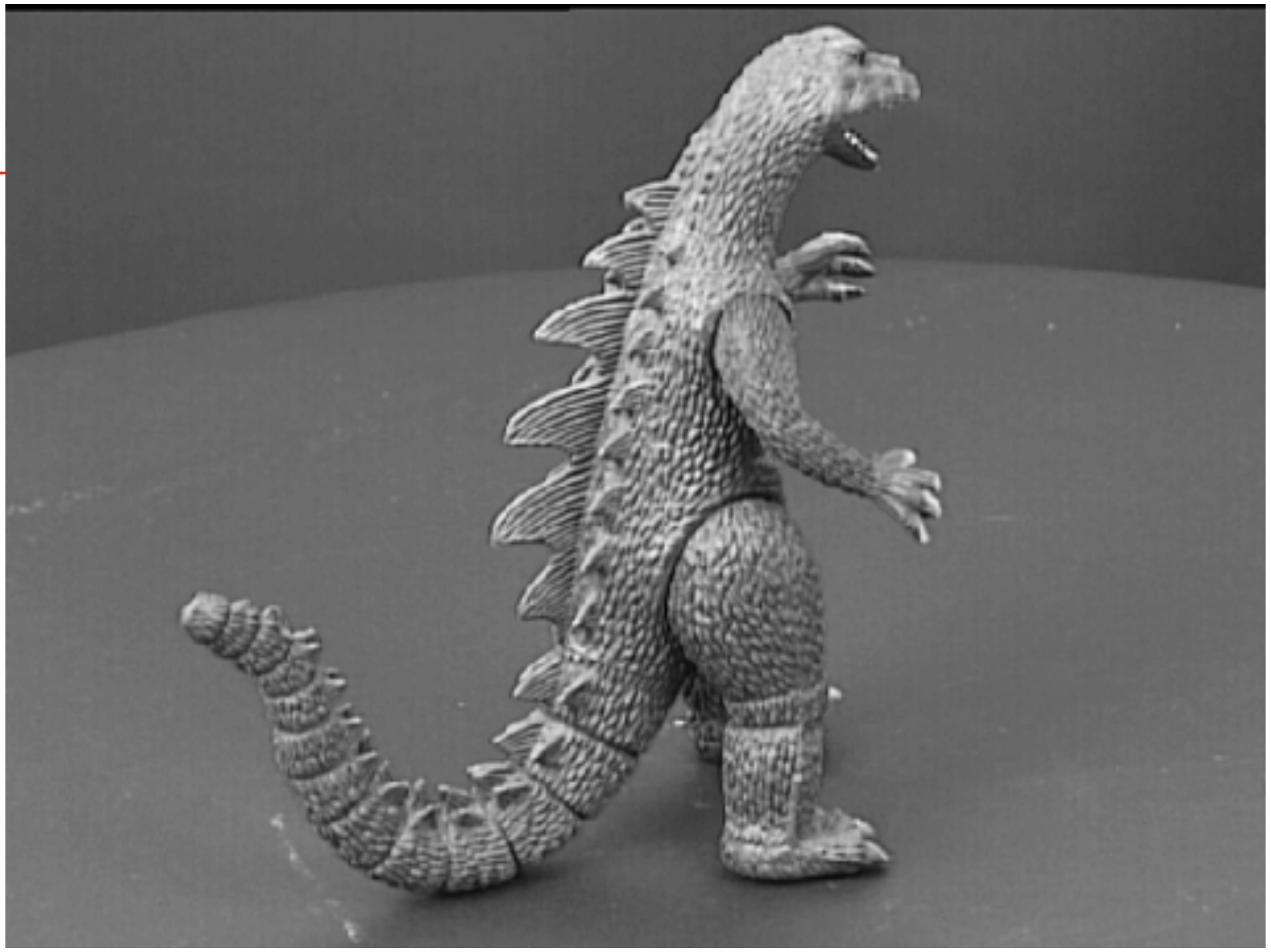
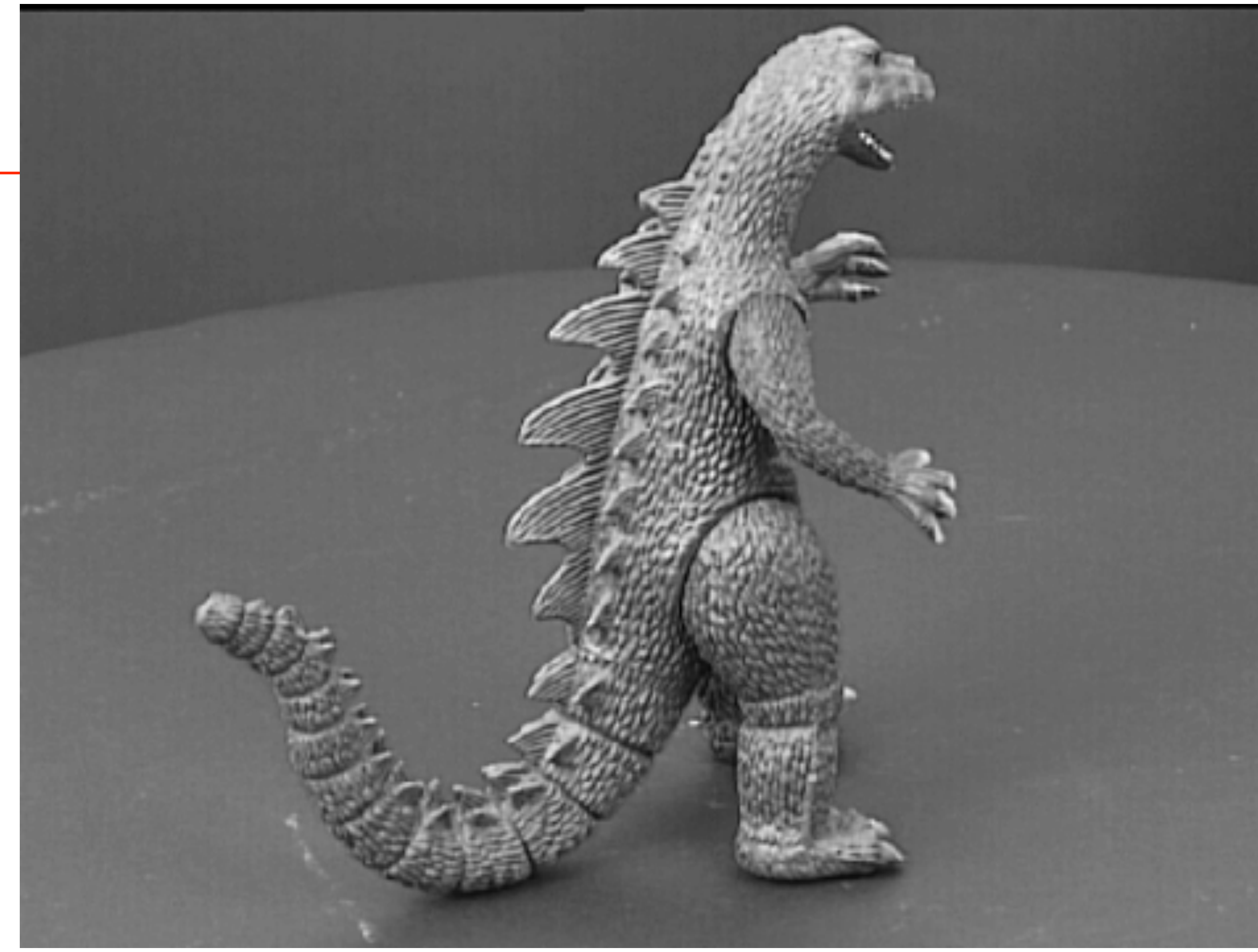


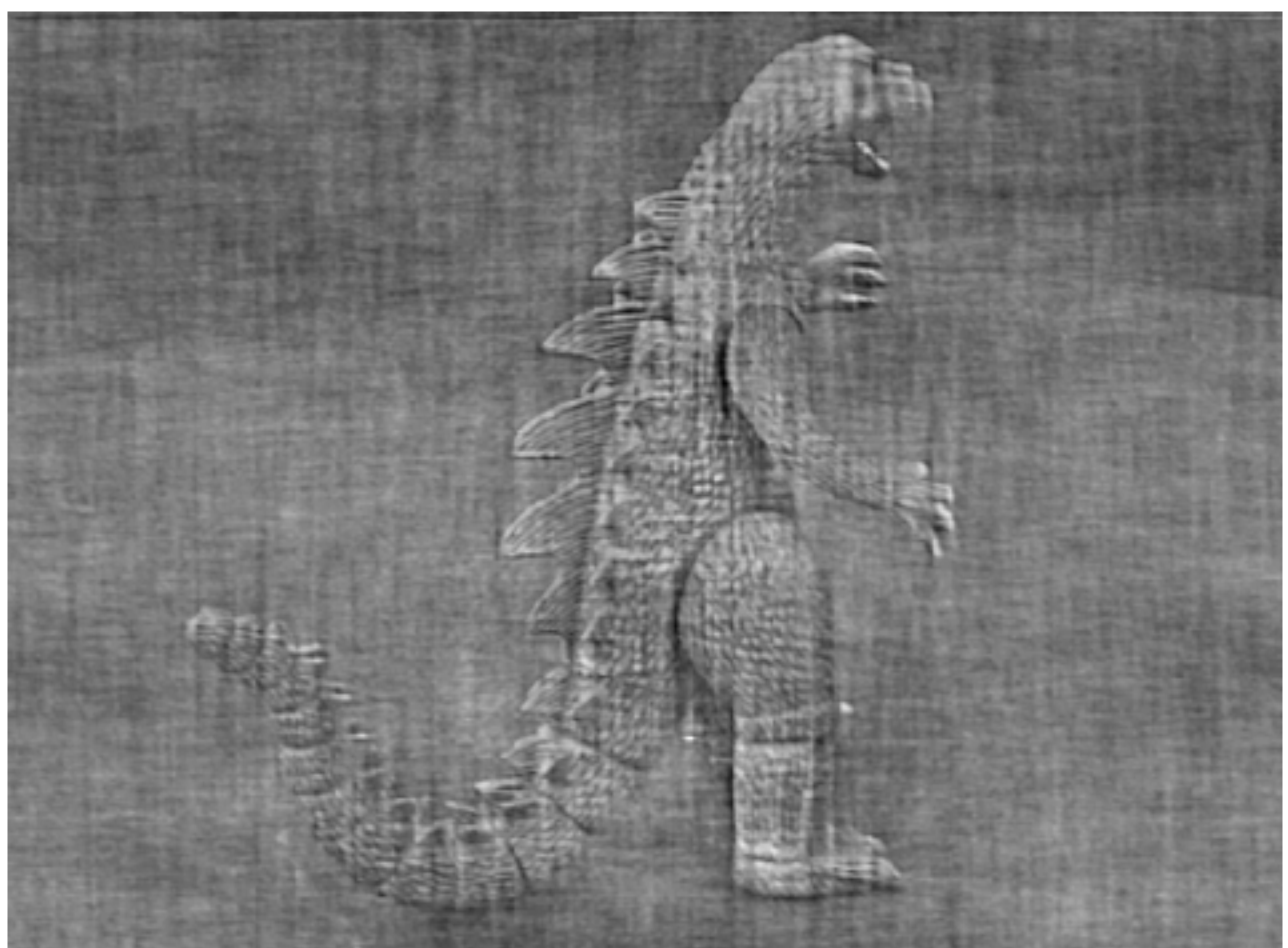
Image 2

Courtesy of JM Brady

What kind of image would we get if we mix the phase and magnitude from these?

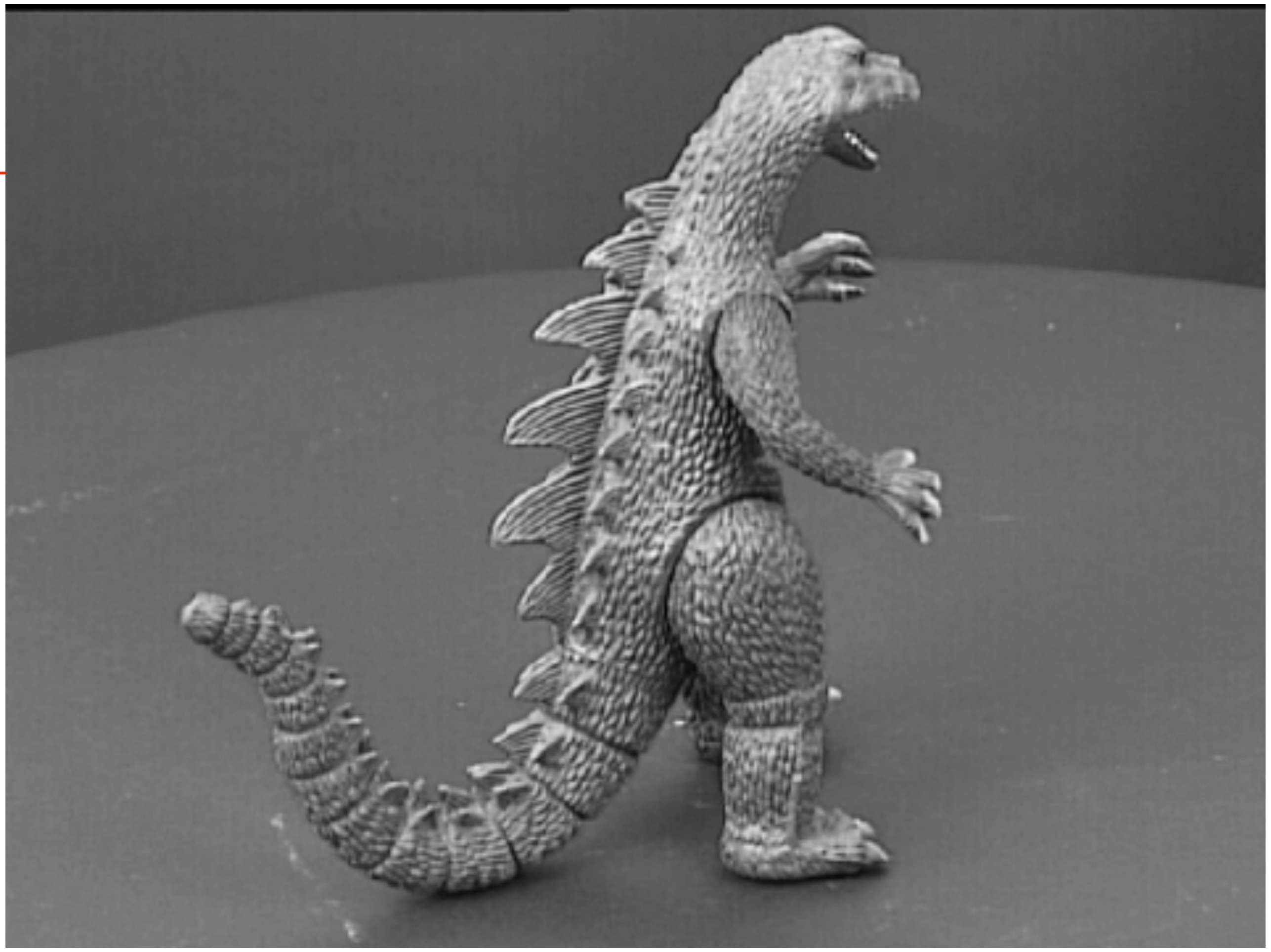


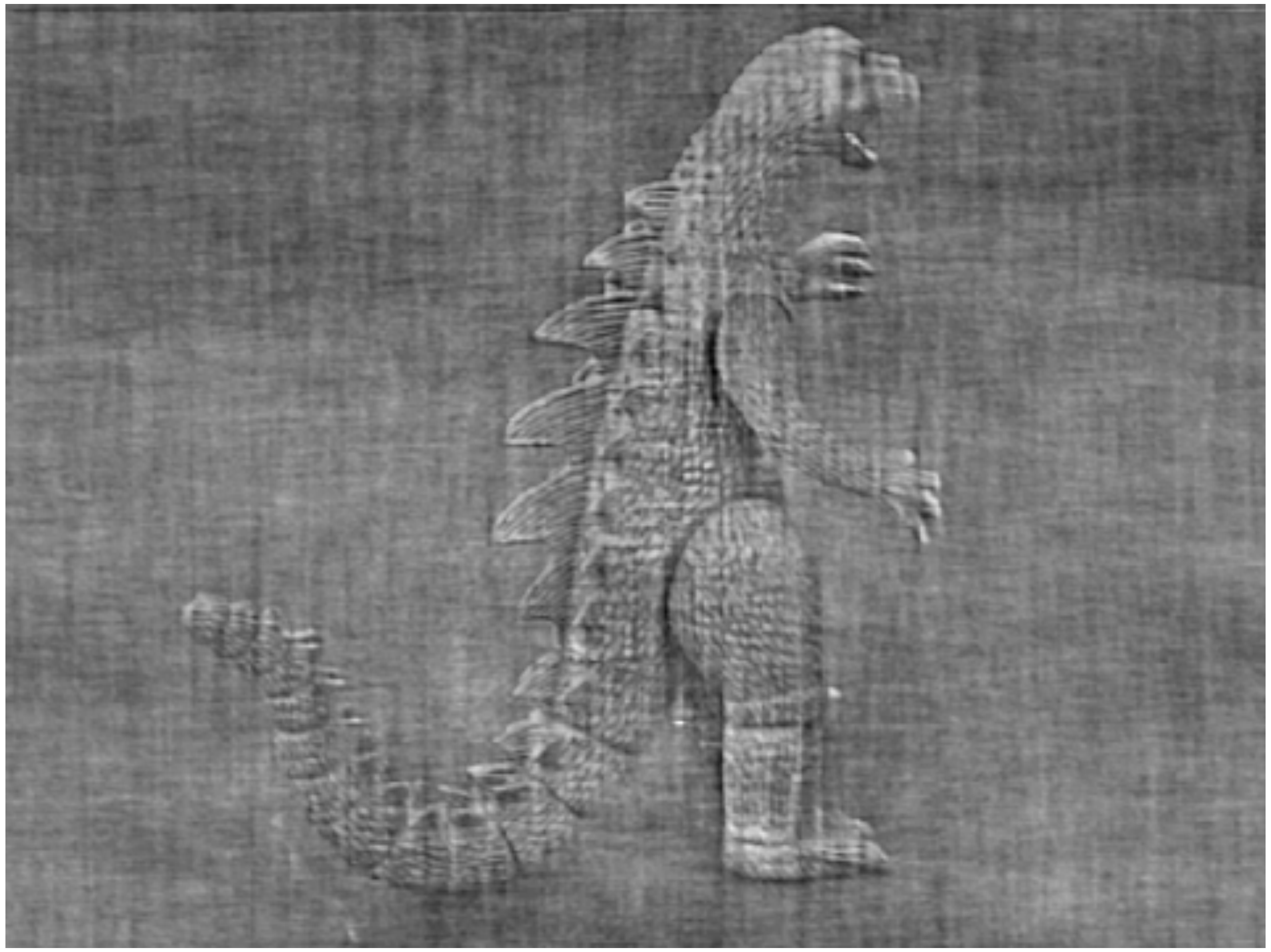
A: Phase
or
B: Magnitude



A: Magnitude
or
B: Phase

Courtesy of JM Brady

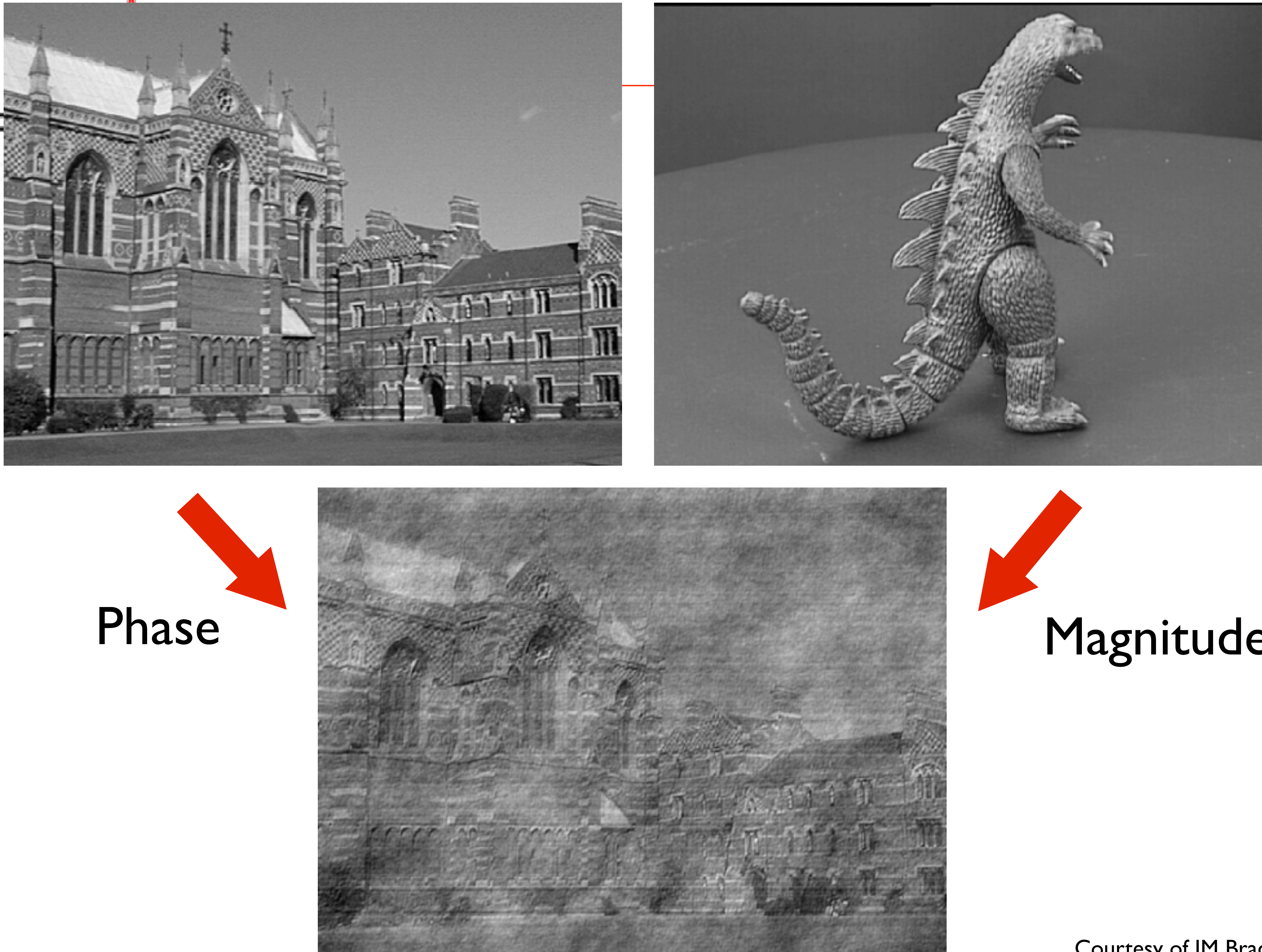




B: Magnitude

B: Phase

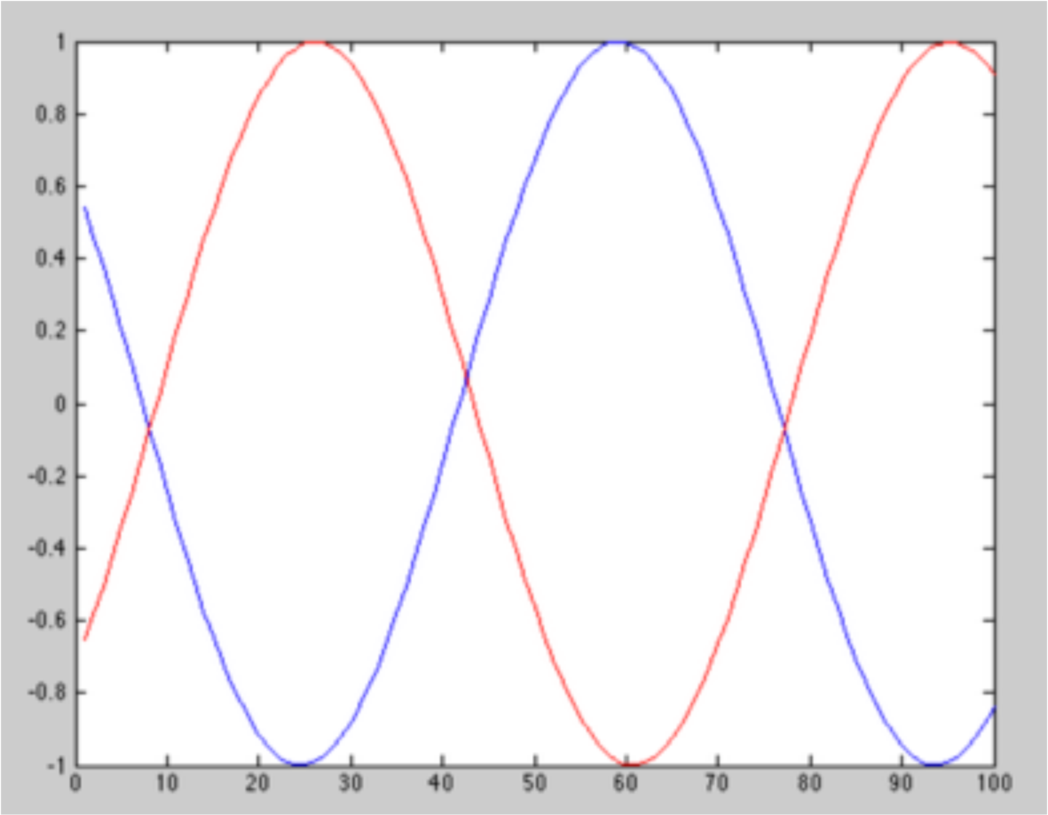
Courtesy of JM Brady



Courtesy of JM Brady

Magnitude [What?]()

Phase [Where?]()



$\xrightarrow{\text{FFT}}$ Same magnitude

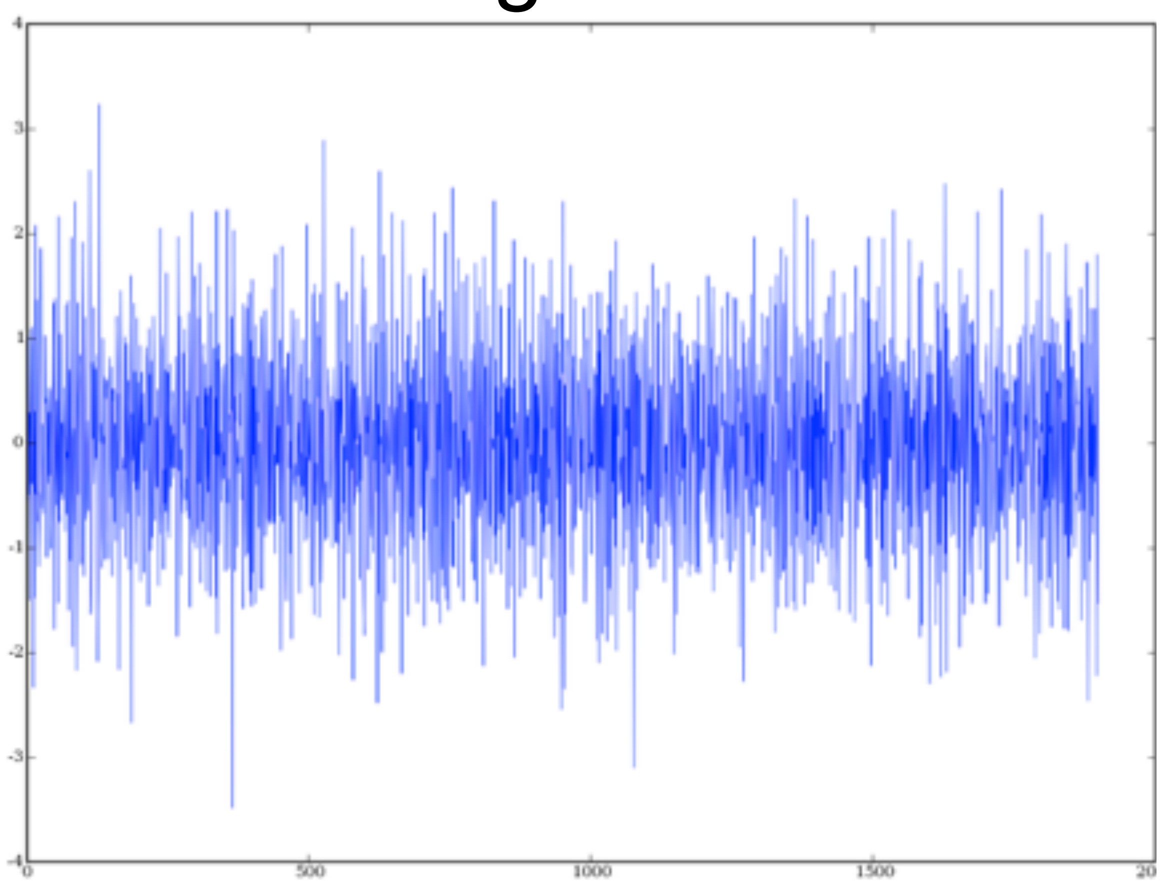
Different phases

Filtering

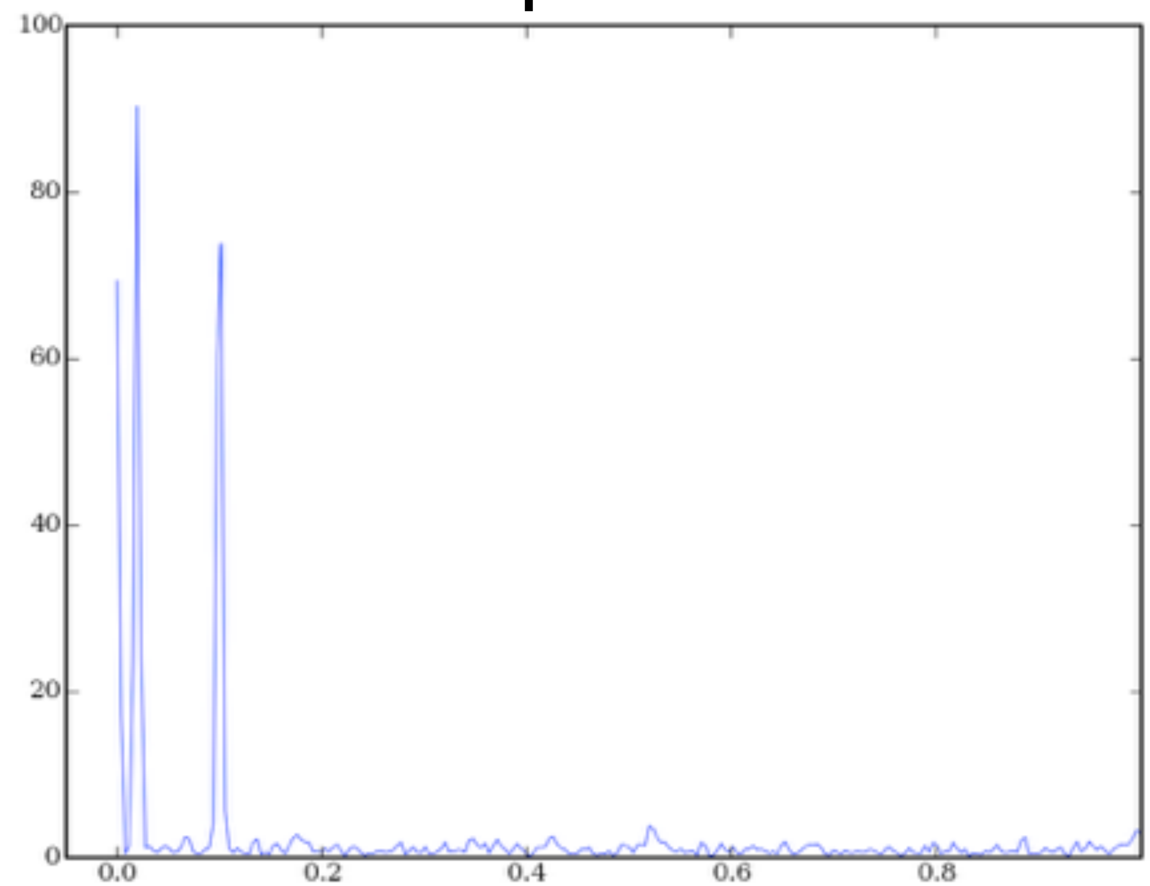
Filtering - 1D

- In Fourier signals are a mixture of different frequency components.
- Often we want some components and not others.

Signal



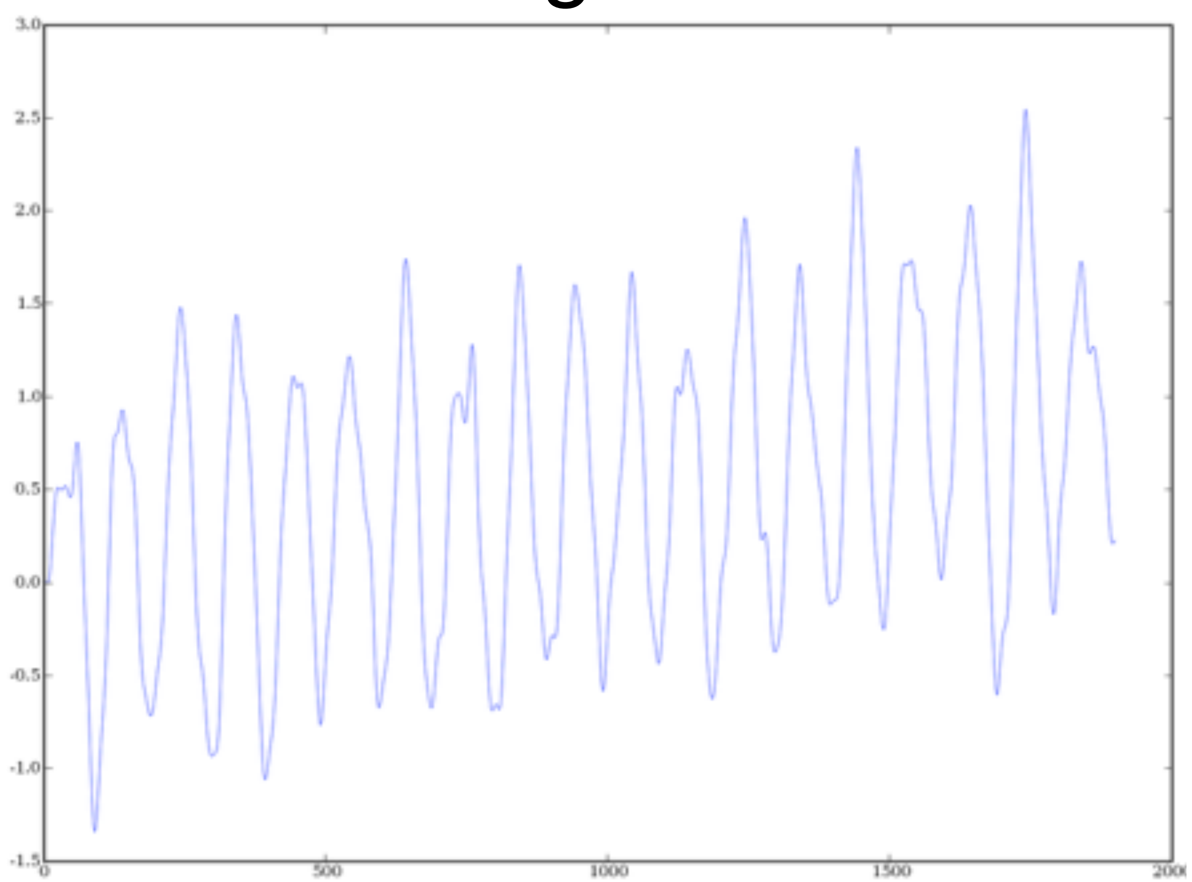
Spectrum



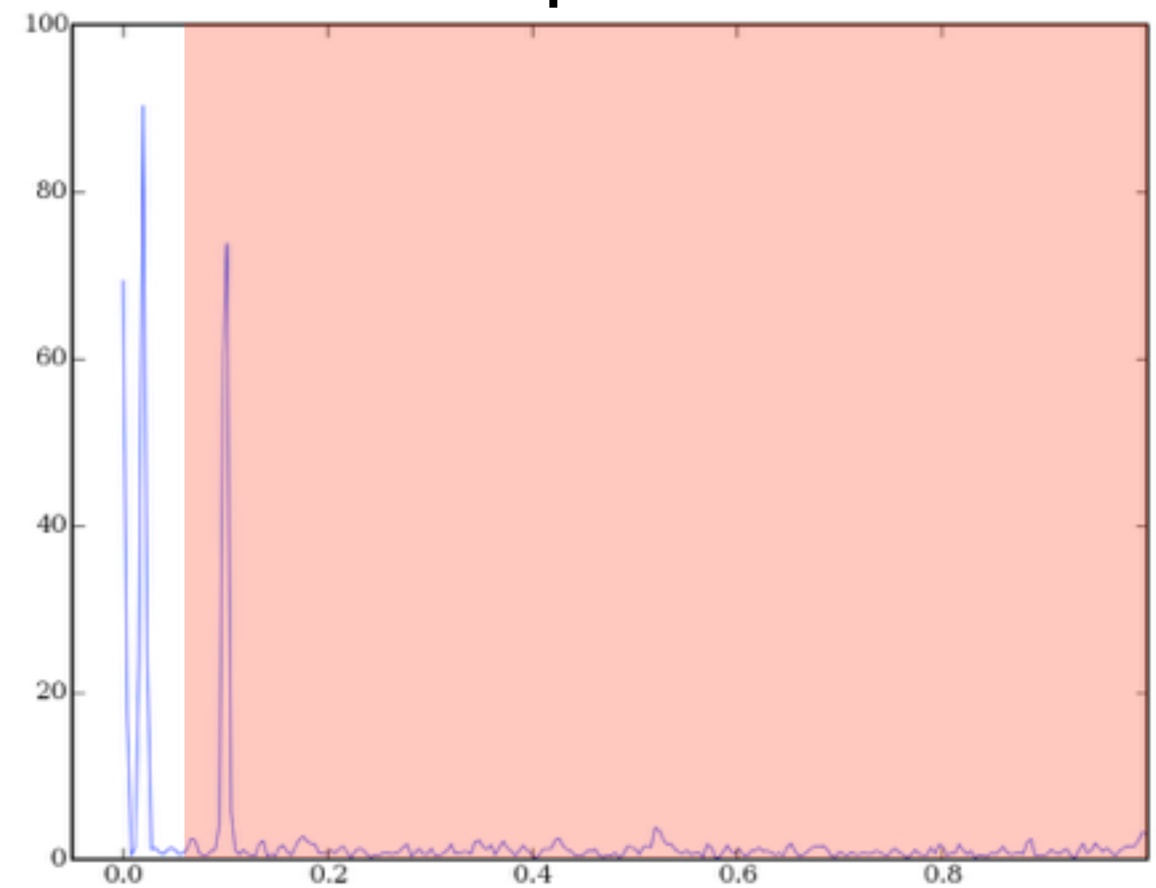
Filtering - 1D

- I want to get rid of *high frequency noise* component.
- **Low Pass** filter - throw away high frequencies

Signal



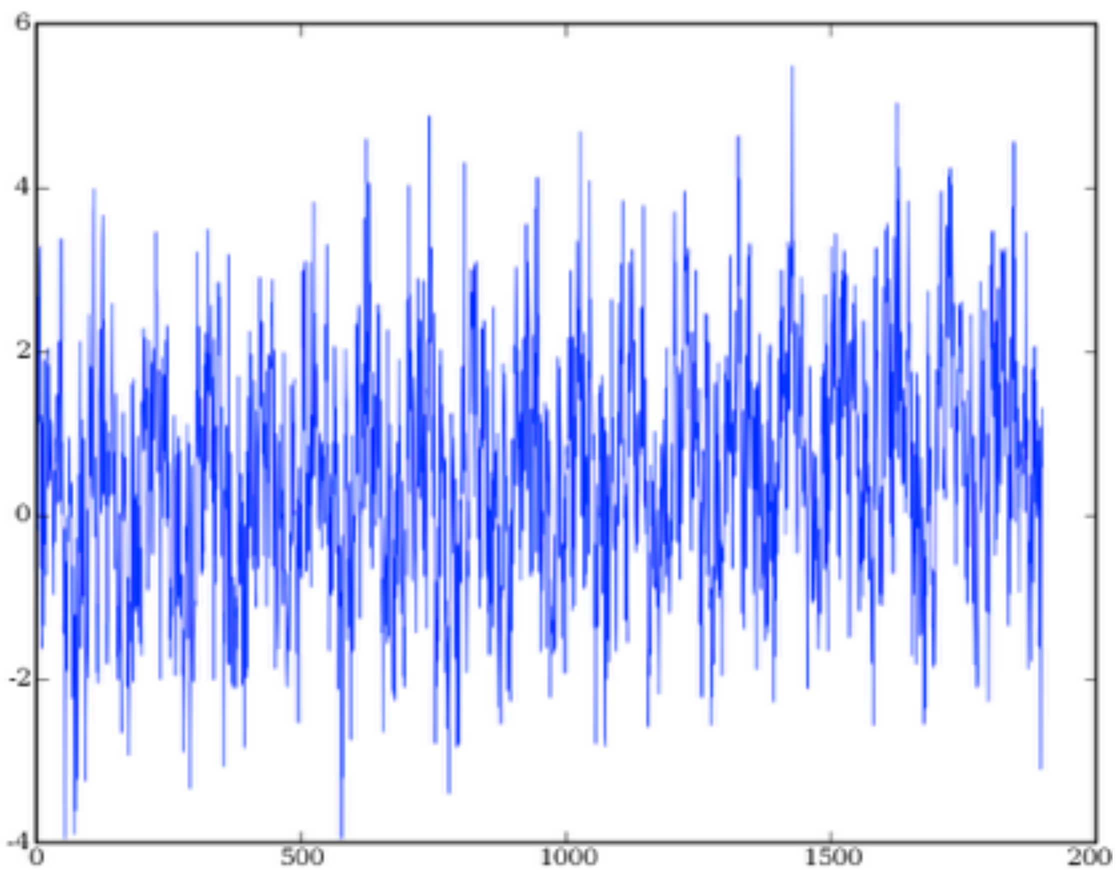
Spectrum



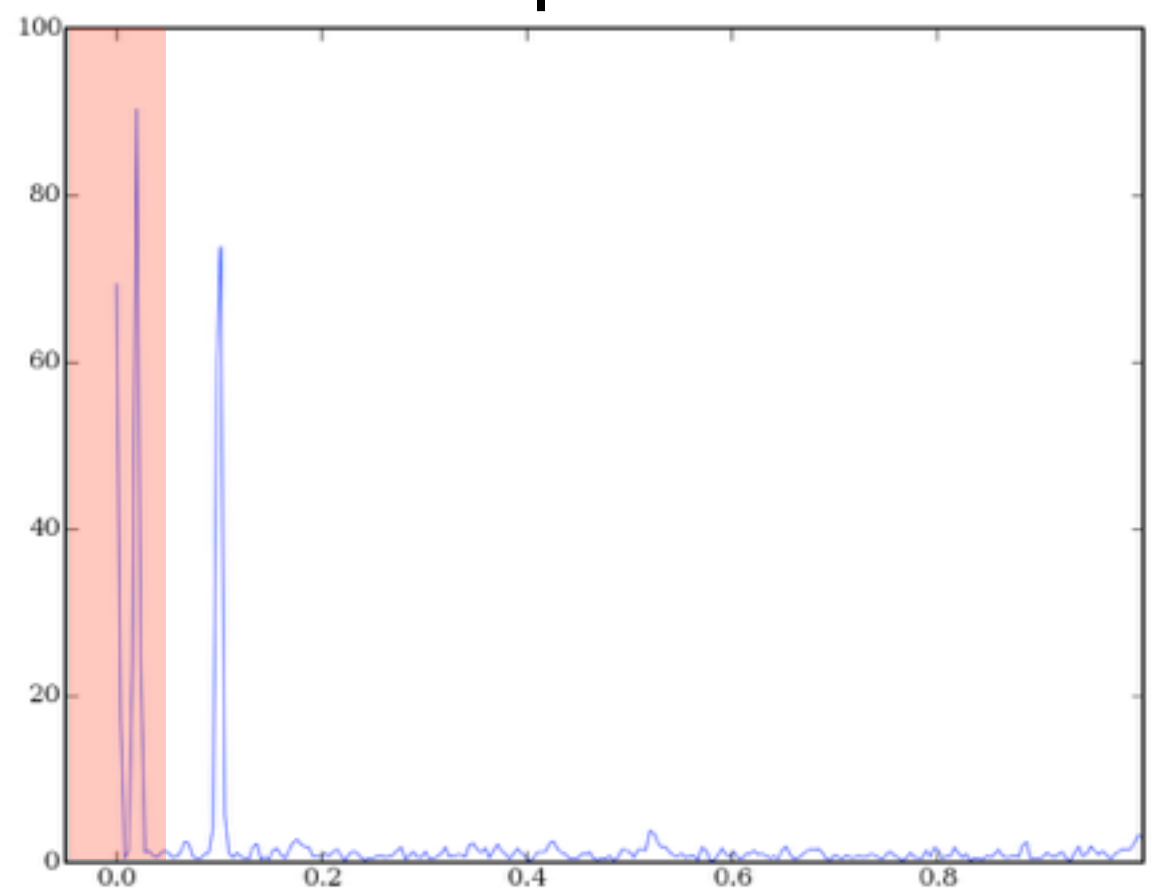
Filtering - 1D

- I want to get rid of low frequency nosie/drift.
- **High Pass** filter - throw away low frequencies.

Signal



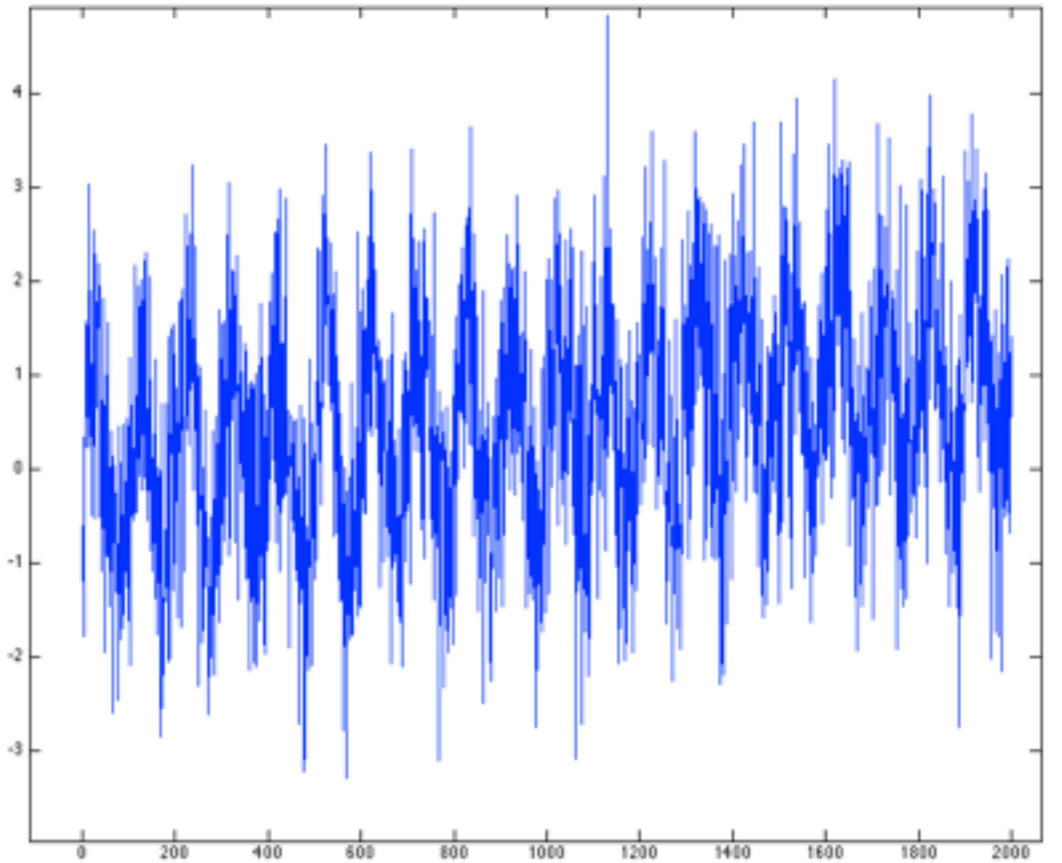
Spectrum



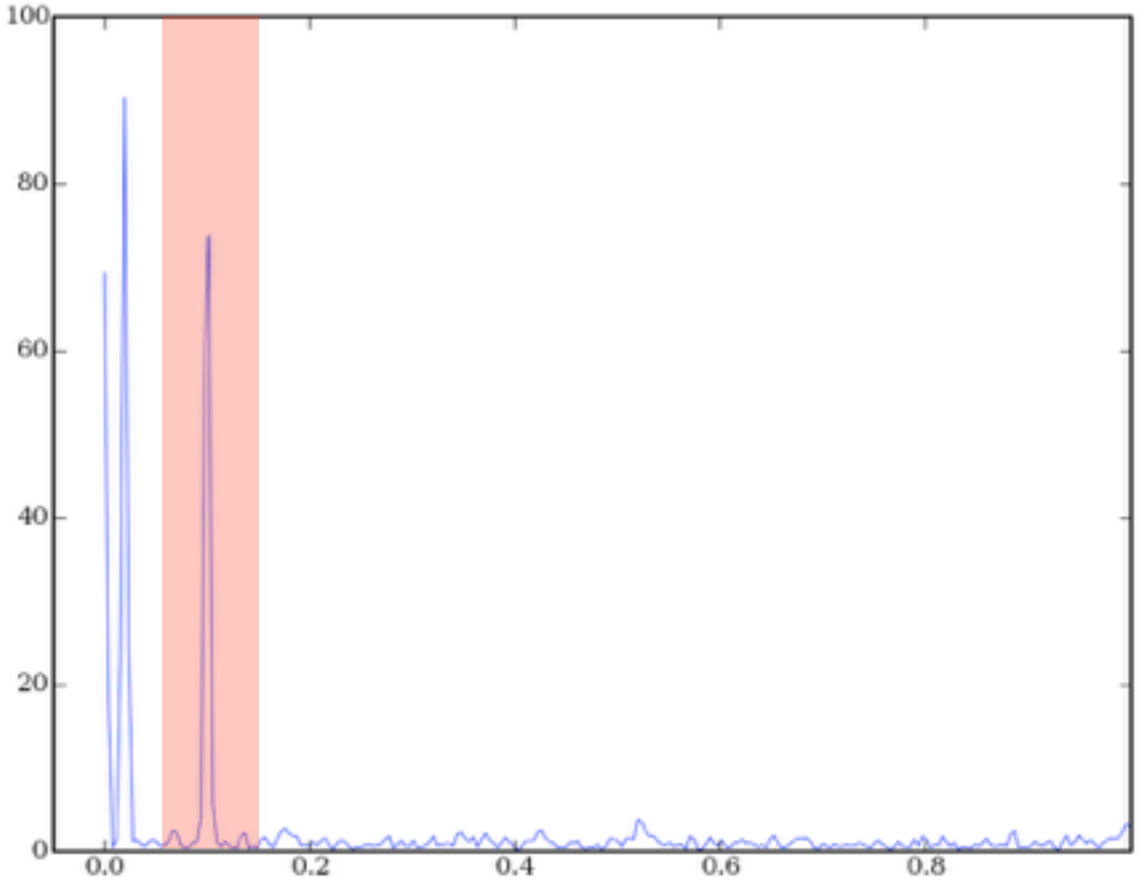
Filtering - 1D

- I want to get rid of that annoying component at *[pick a frequency]* (e.g. mains noise).
- **Band Stop** filter - let everything through except...

Signal

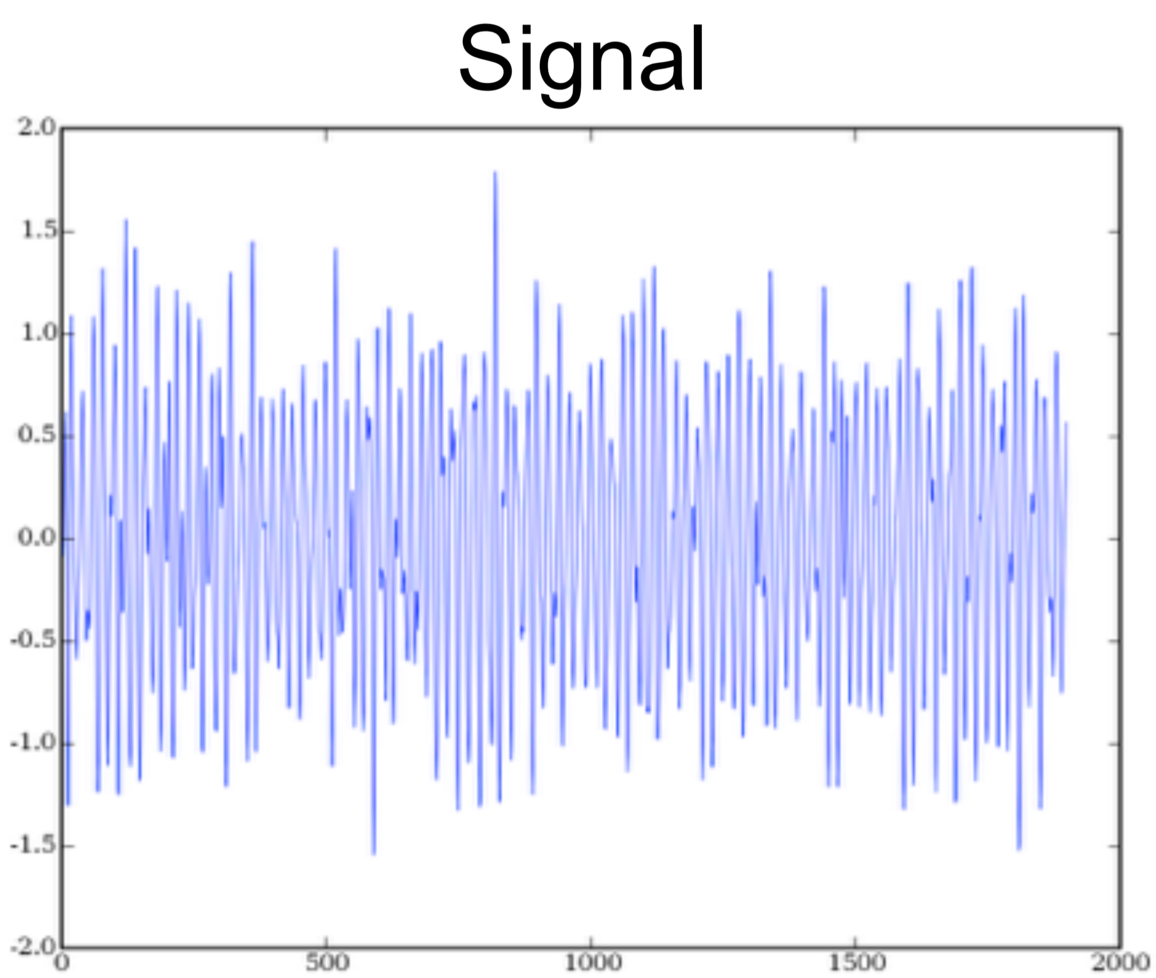


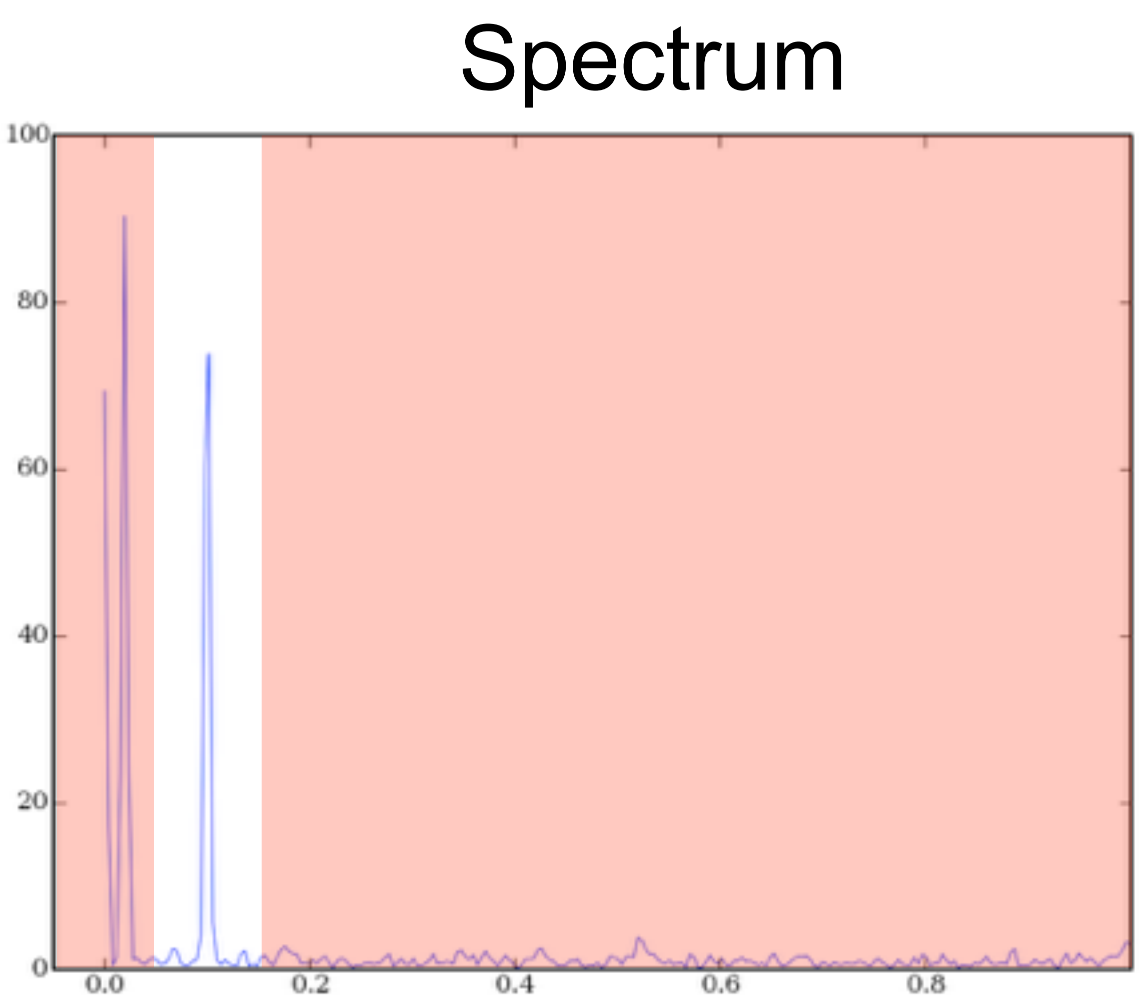
Spectrum



Filtering - 1D

- I want to get rid of all this other stuff that is not my signal (not always this simple!)
- **Band Pass** filter - get rid of everything but...

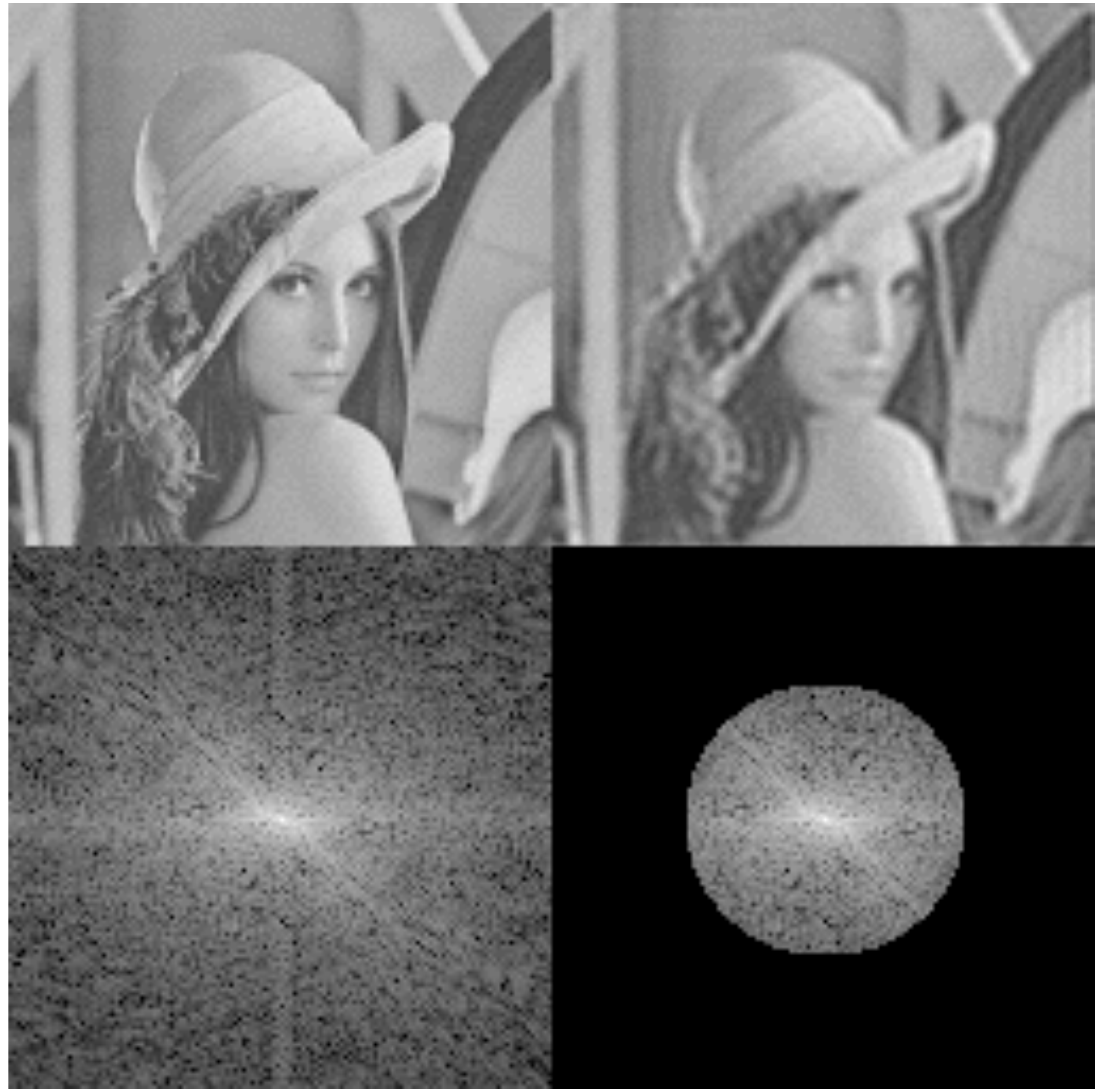




Filtering - 2D

- Same principles as in 1D.

- **Low Pass**
- Remove high frequencies.
- Loose detail.

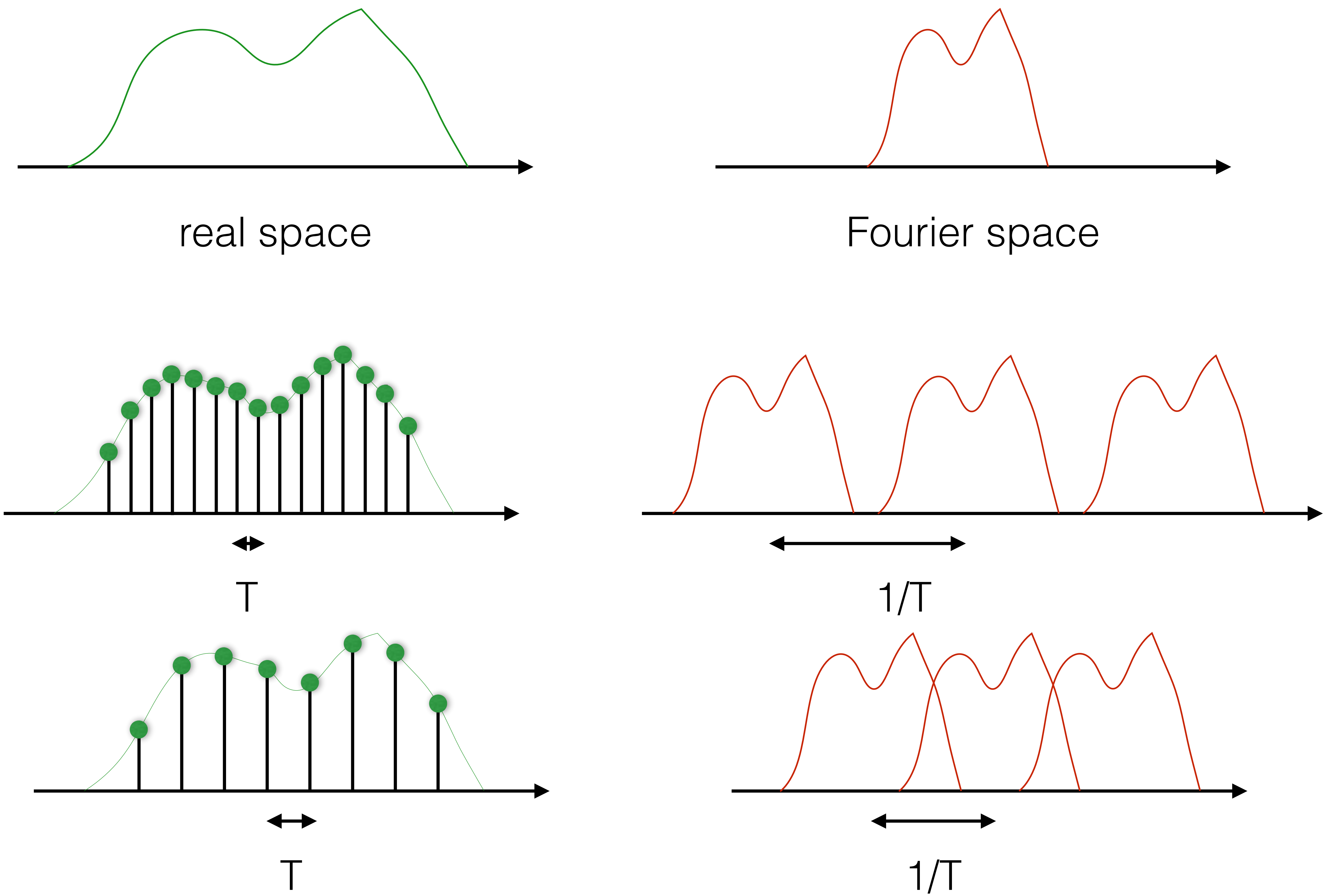


http://www.cs.unm.edu/~brayer/vision/fourier.html

Sampling, aliasing

—> details in the practical

Sampling in real space is like repeating in Fourier space and vice versa

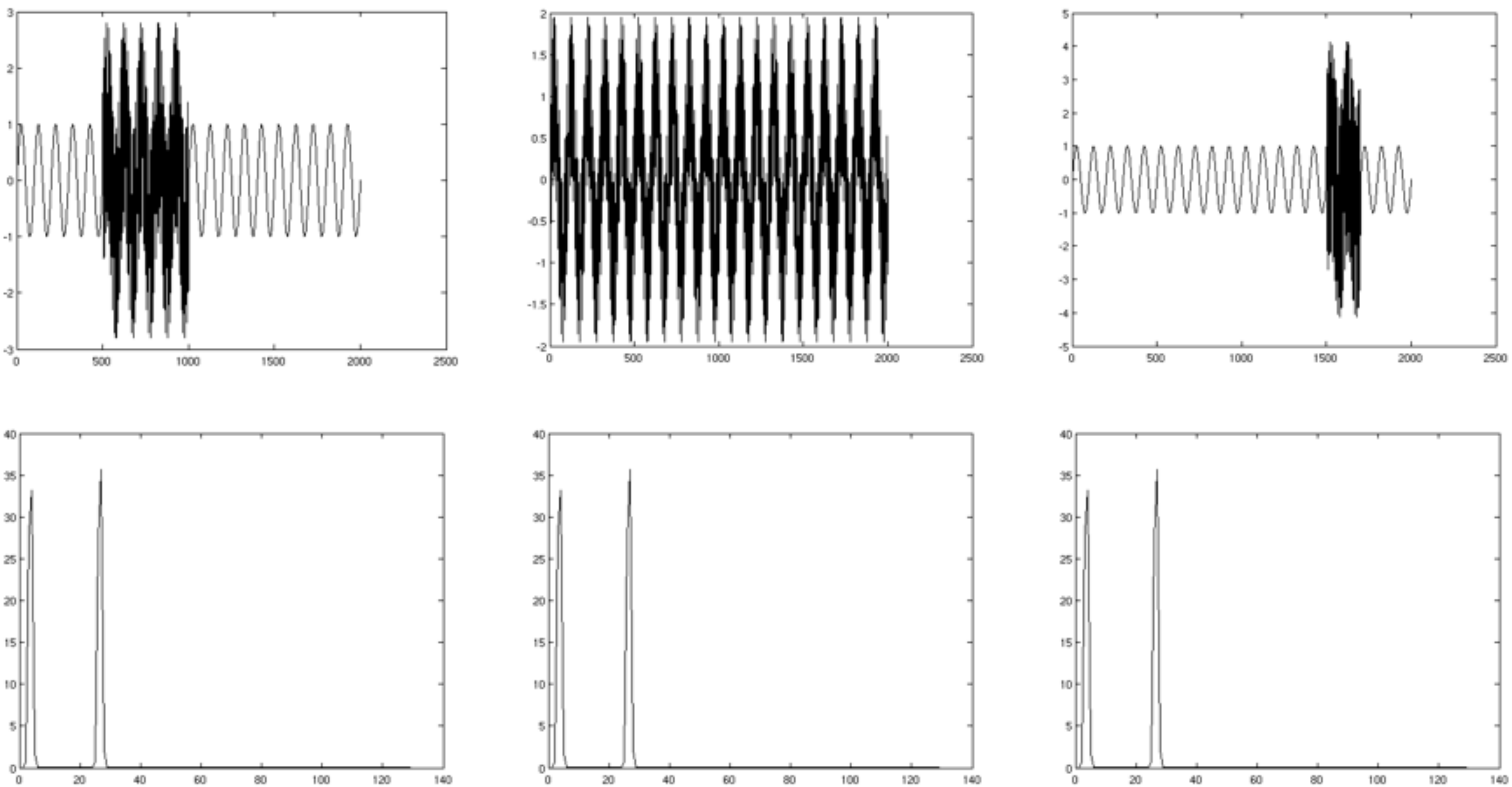


Time-frequency analysis

—> details in the practical

Time-frequency

- Three different signals - same frequency spectrum.



Time-frequency

- Need a ‘dynamic’ time-frequency representation.

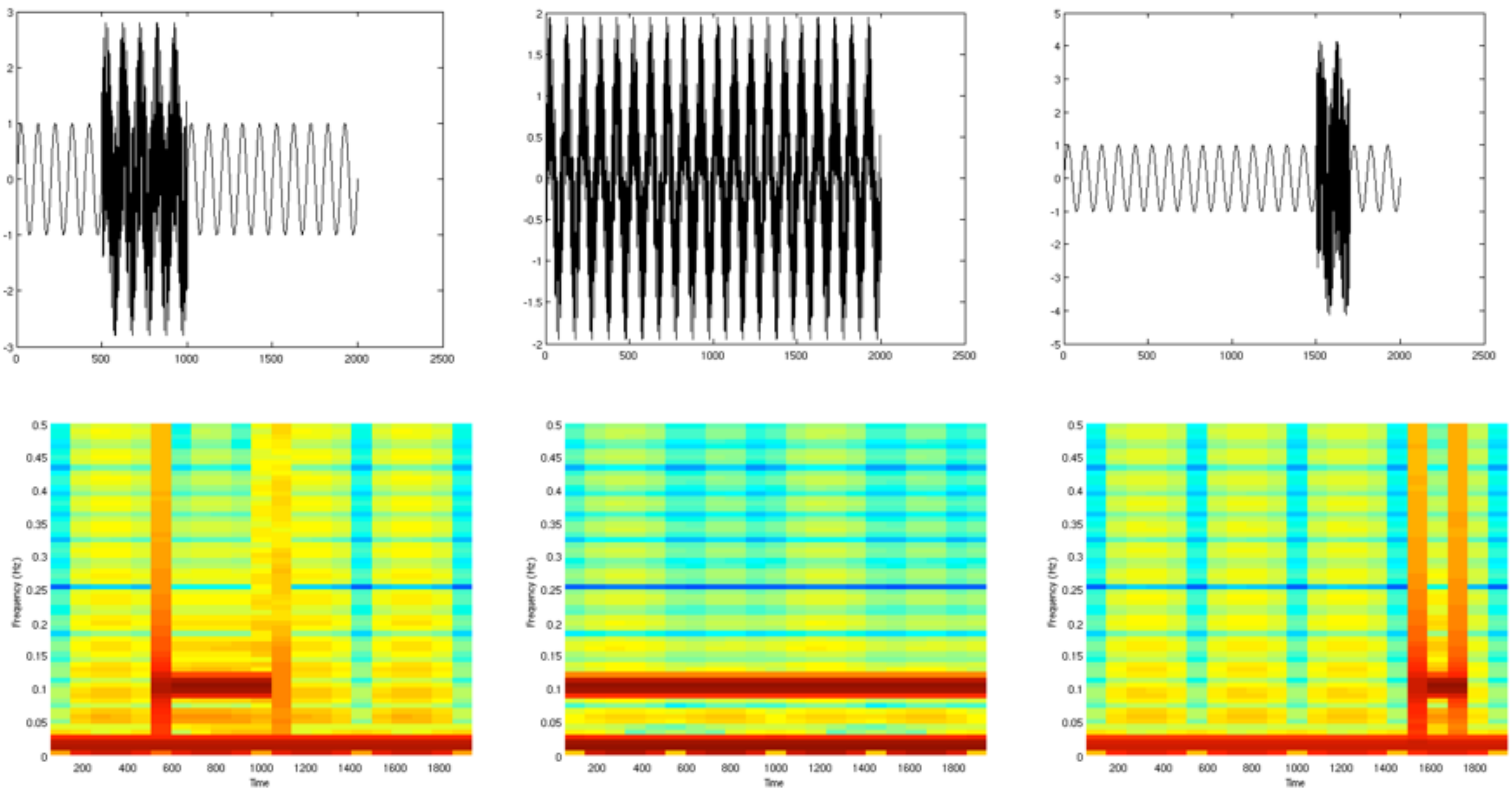
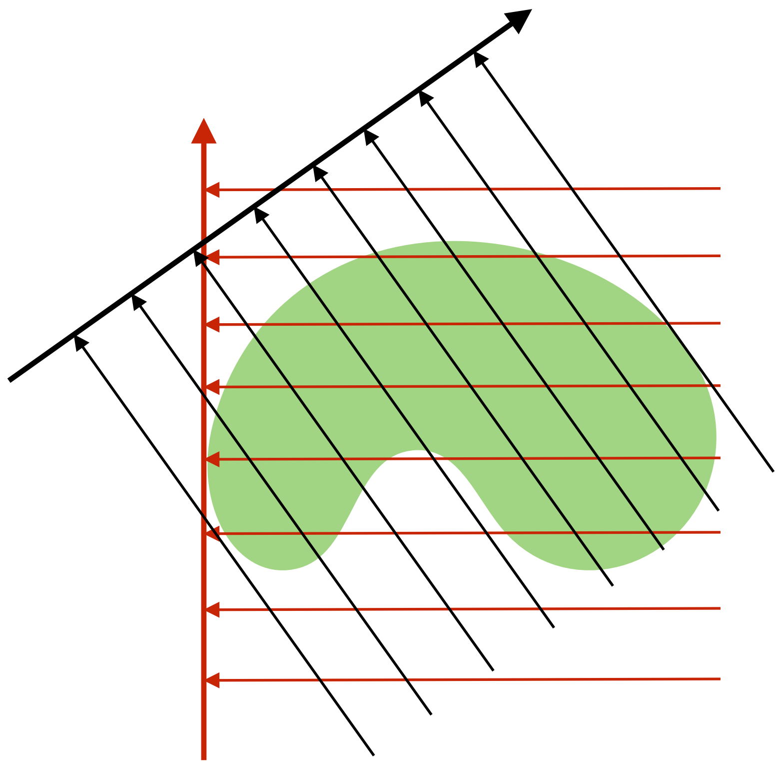


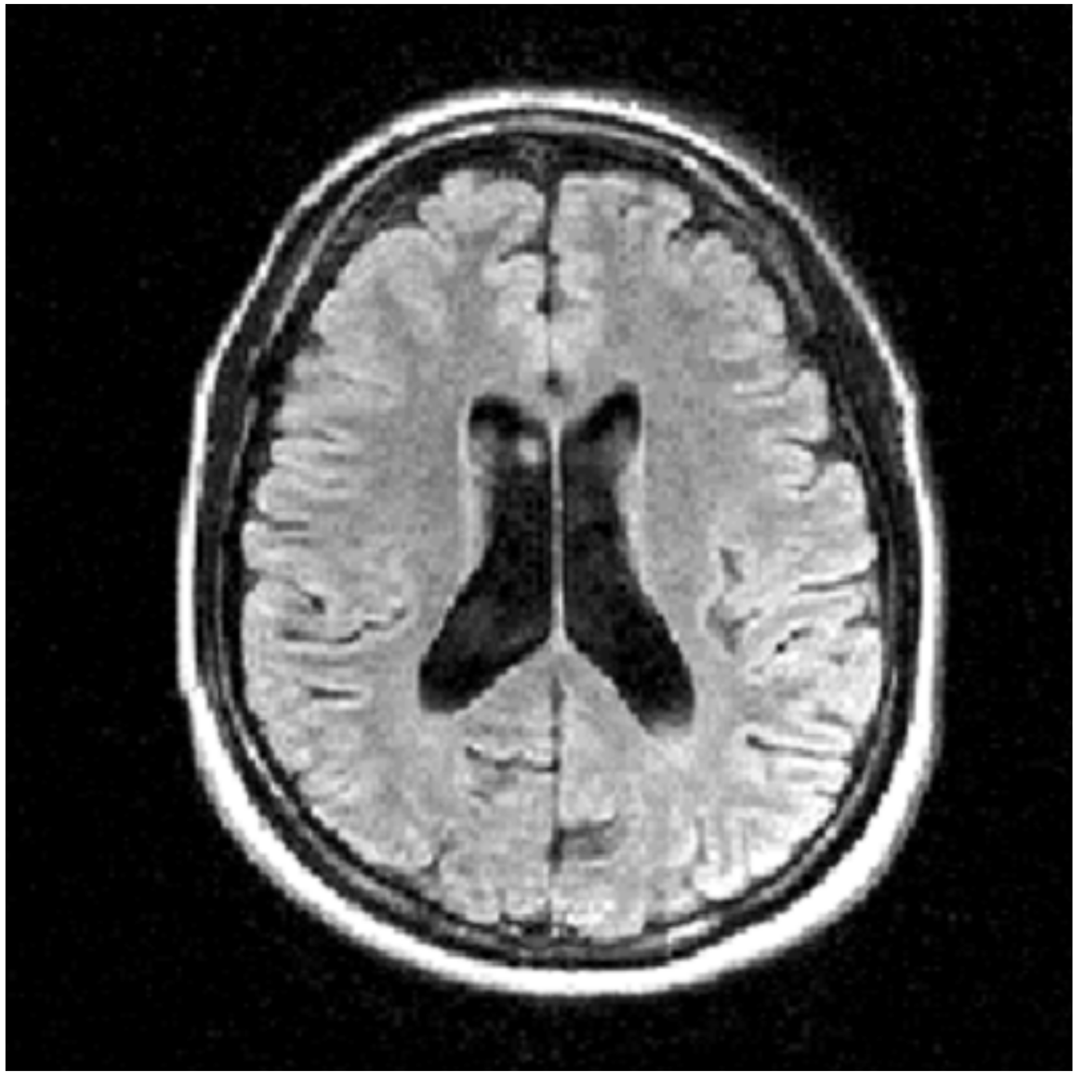
Image reconstruction

—> details in the practical

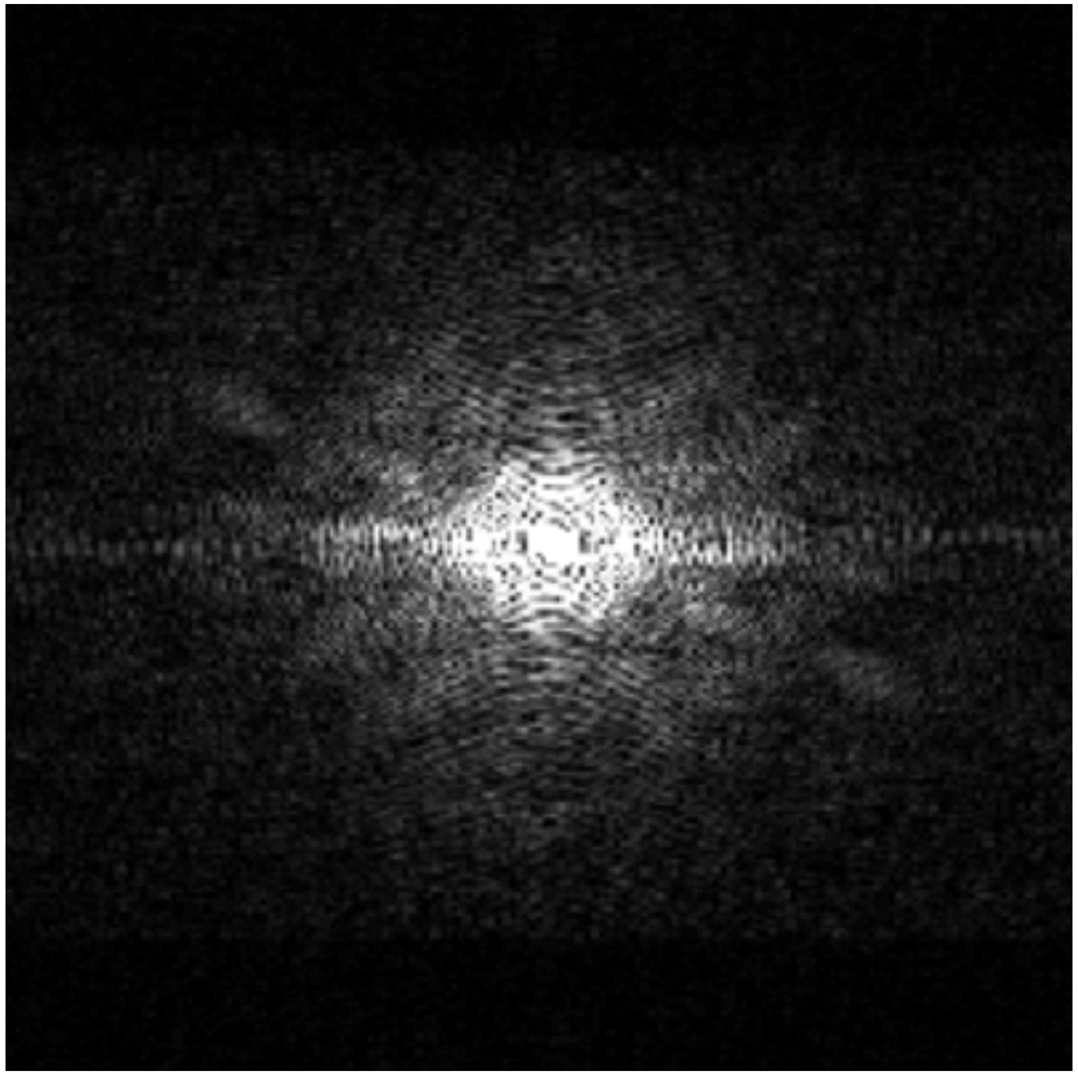
Radon transform (X-ray CT)



Magnetic Resonance Imaging



Object of interest



Measurement
(Fourier space)

www.fmrib.ox.ac.uk/~saad/GP/fourier.html

Prac Maths - Signal and Image Processing Practical

Overview

This practical requires Matlab. Go through the instructions and execute the listed commands in a Matlab command window (you can copy-paste). Don't click on the "answers" links until you have thought hard about the questions. Raise your hand if you need help, but perhaps try first the "help" command in Matlab if you are unsure about Matlab syntax issues.

Contents:

Fourier analysis

To start with, let us create some simple 1D signals and examine their Fourier transforms. In this first example, we will check the frequency content of a simple periodic signal.

Generate a cosine signal with given magnitude, frequency, sampling rate, and duration:

```
mag  = 2;     % magnitude (arbitrary units)
freq = 5;     % frequency in Hz
samp = 100;   % sampling rate in Hz

t        = 0:1/samp:1;  % time (1s of data)
t        = t(1:end-1);  % remove last time point
N        = length(t);   % store the number of time points

x = mag*cos(2*pi*freq*t);    % the signal equation
figure
plot(t,x,'.-');
```