The Frequency Domain, without tears

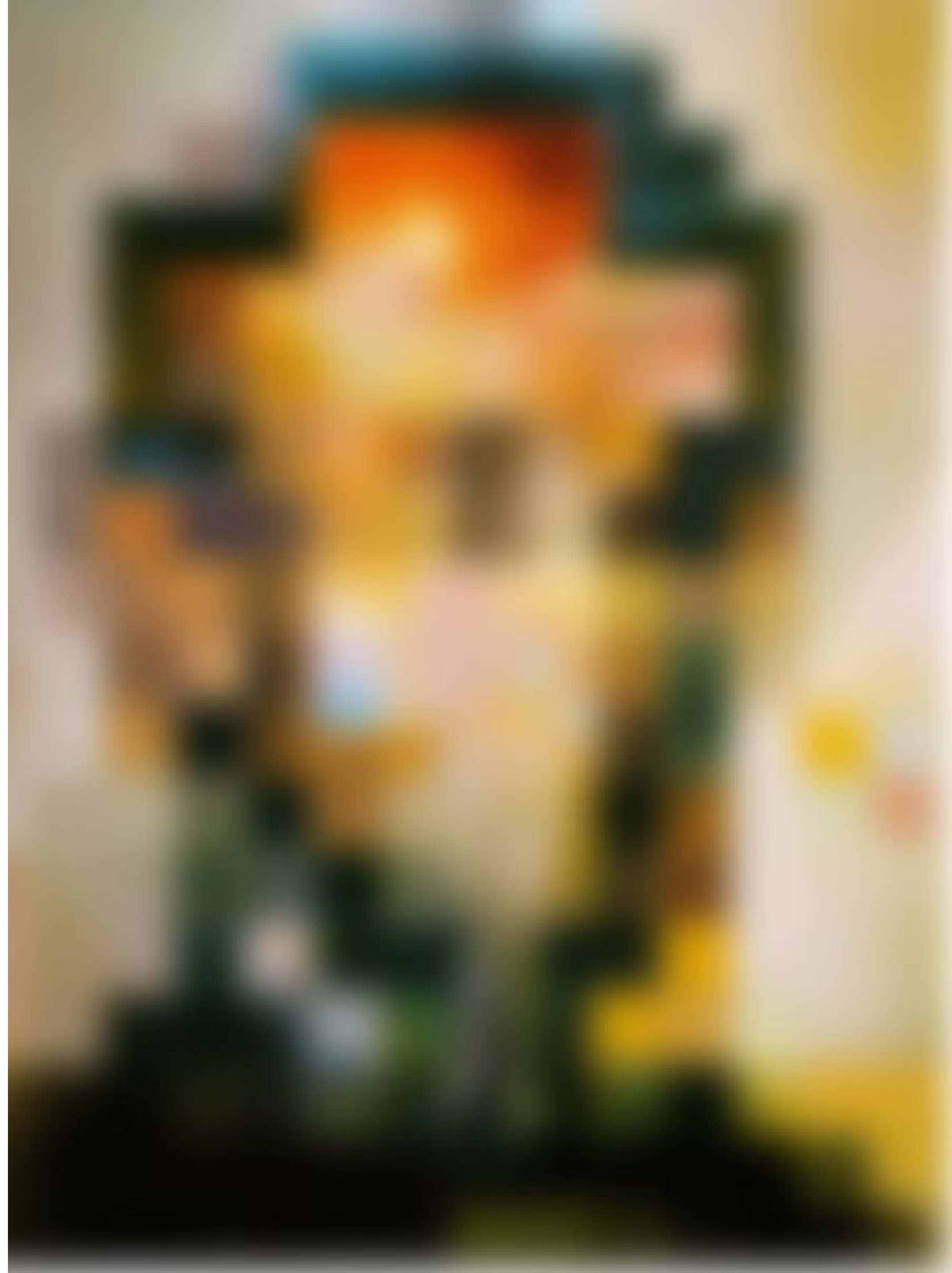
Somewhere in Cinque Terre, May 2005

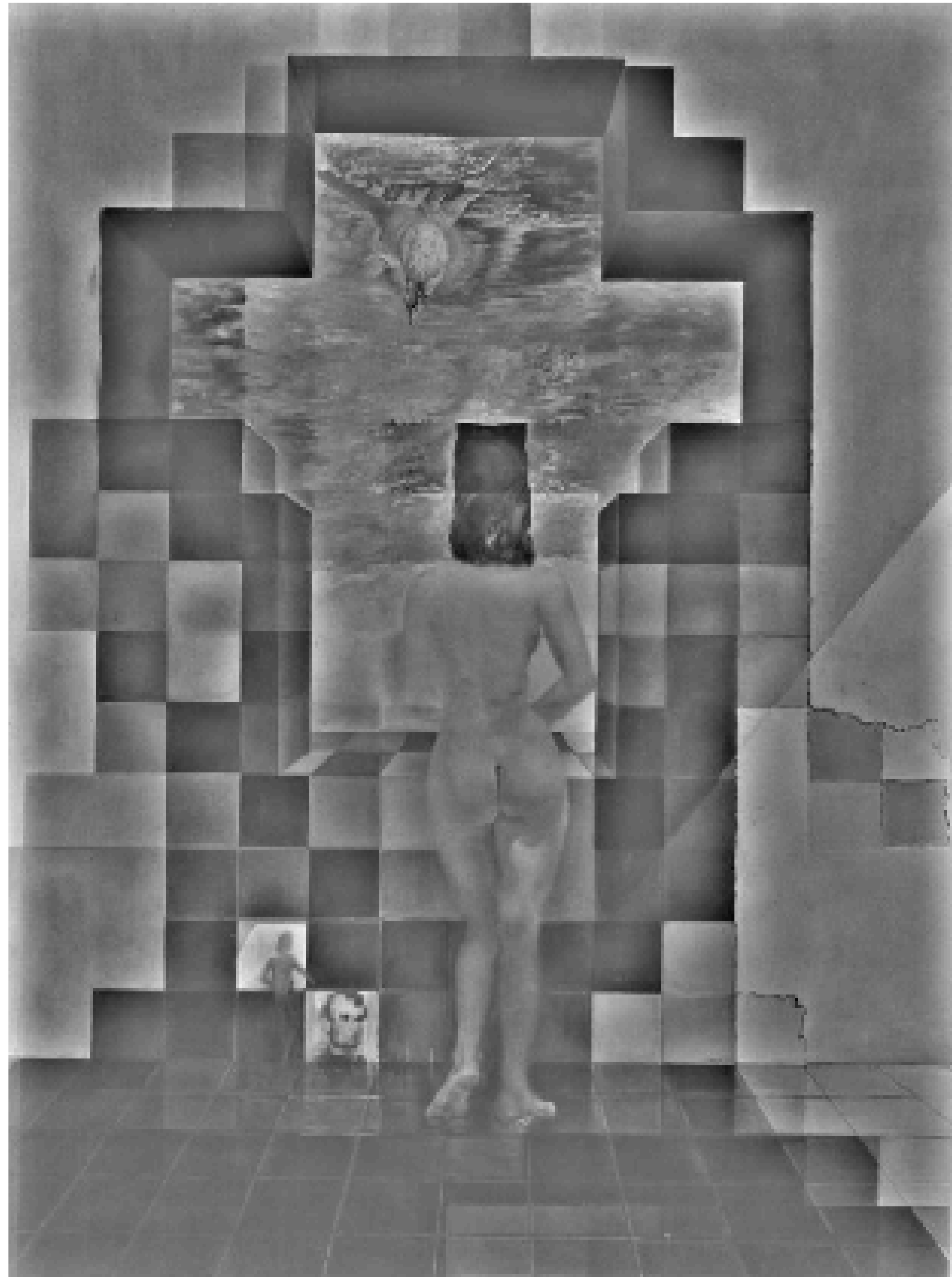
Many slides borrowed from Steve Seitz

CS180: Intro to Computer Vision and Comp. Photo

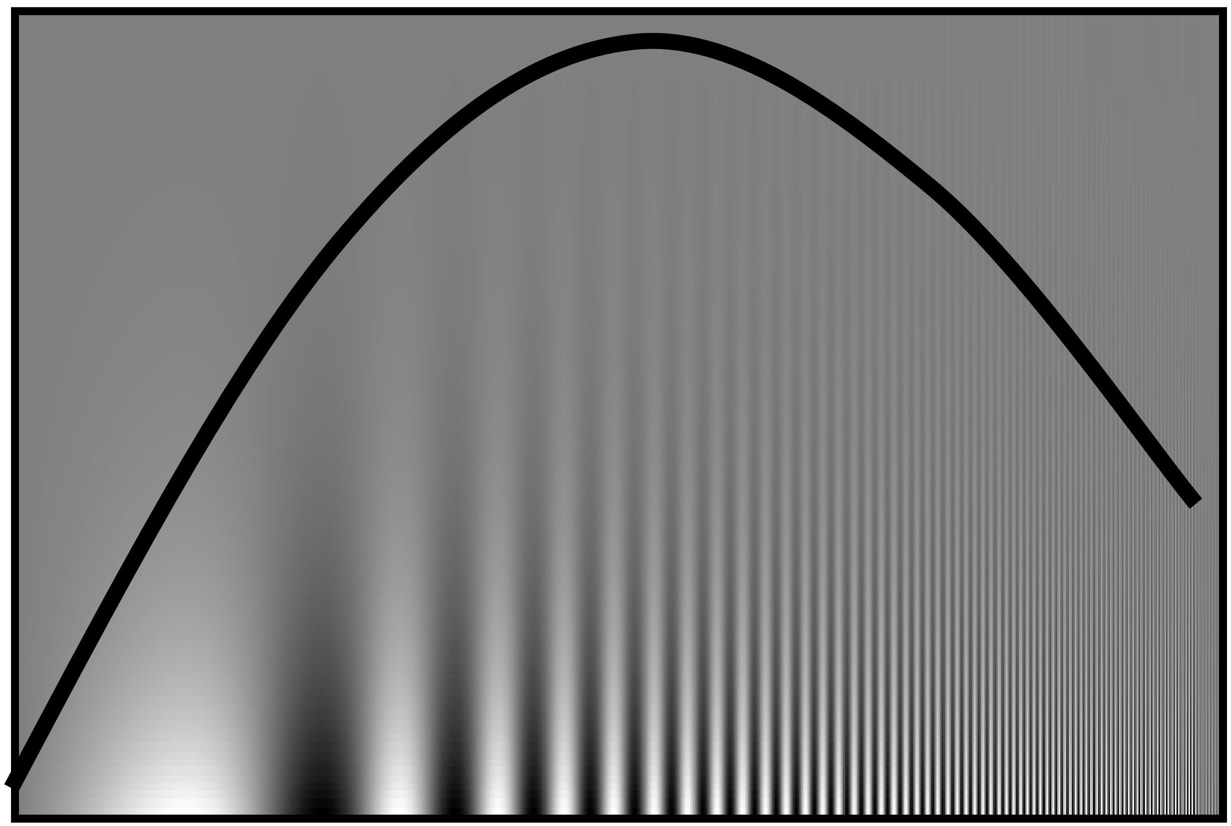
Alexei Efros, UC Berkeley, Fall 2024

Salvador Dali
“Gala Contemplating the Mediterranean Sea, which at 30 meters becomes the portrait of Abraham Lincoln”, 1976





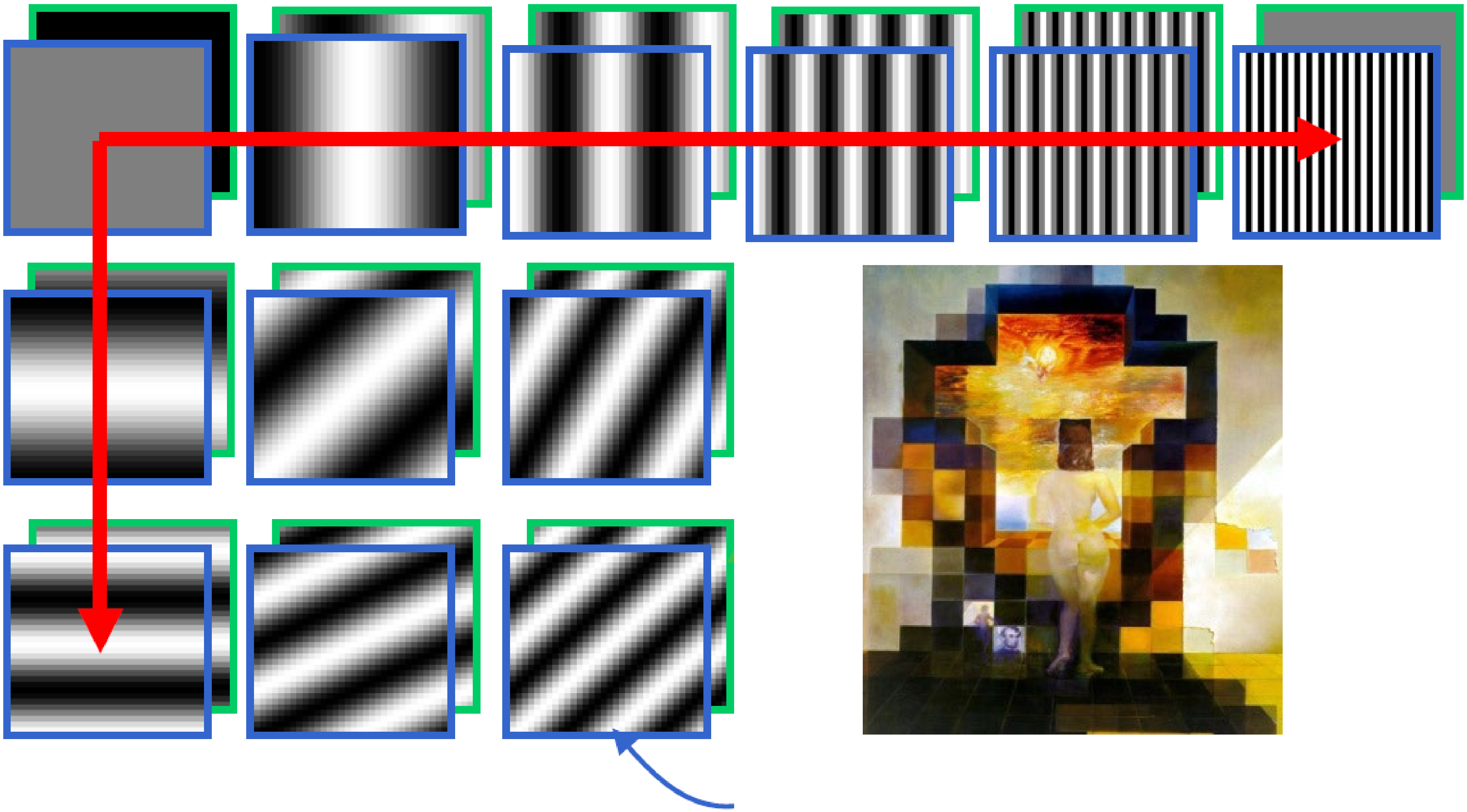
Spatial Frequencies and Perception



Campbell-Robson contrast sensitivity curve

A nice set of basis

Teases away fast vs. slow changes in the image.



This change of basis has a special name…

Jean Baptiste Joseph Fourier (1768-1830)

had crazy idea (1807

Any univariate function ca
be rewritten as a weighted
sum of sines and cosines
different frequencies.

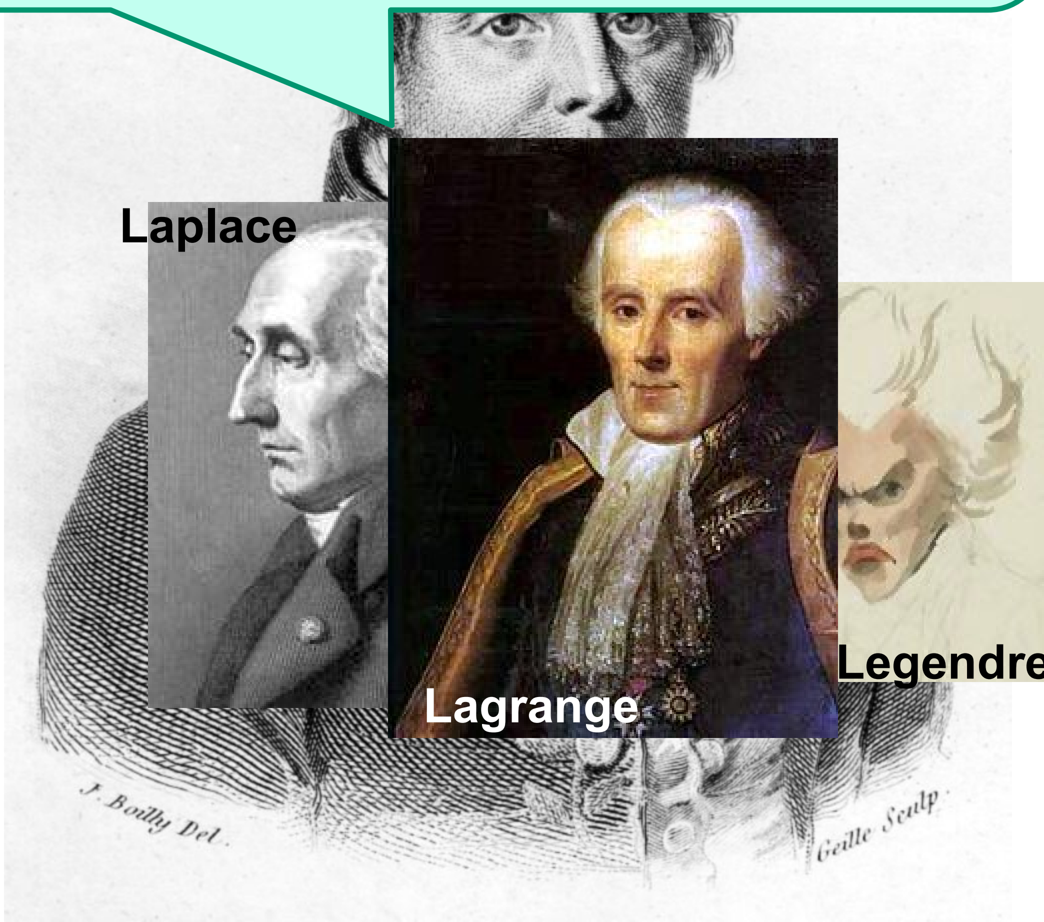
...the manner in which the author arrives at these equations is not exempt of difficulties and... his analysis to integrate them still leaves something to be desired on the score of generality and even rigour.

Don't believe it?

- Neither did Lagrange, Laplace, Poisson and other big wigs
- Not translated into English until 1878!

But it's (mostly) true!

- called Fourier Series



A sum of sines

Our building block:

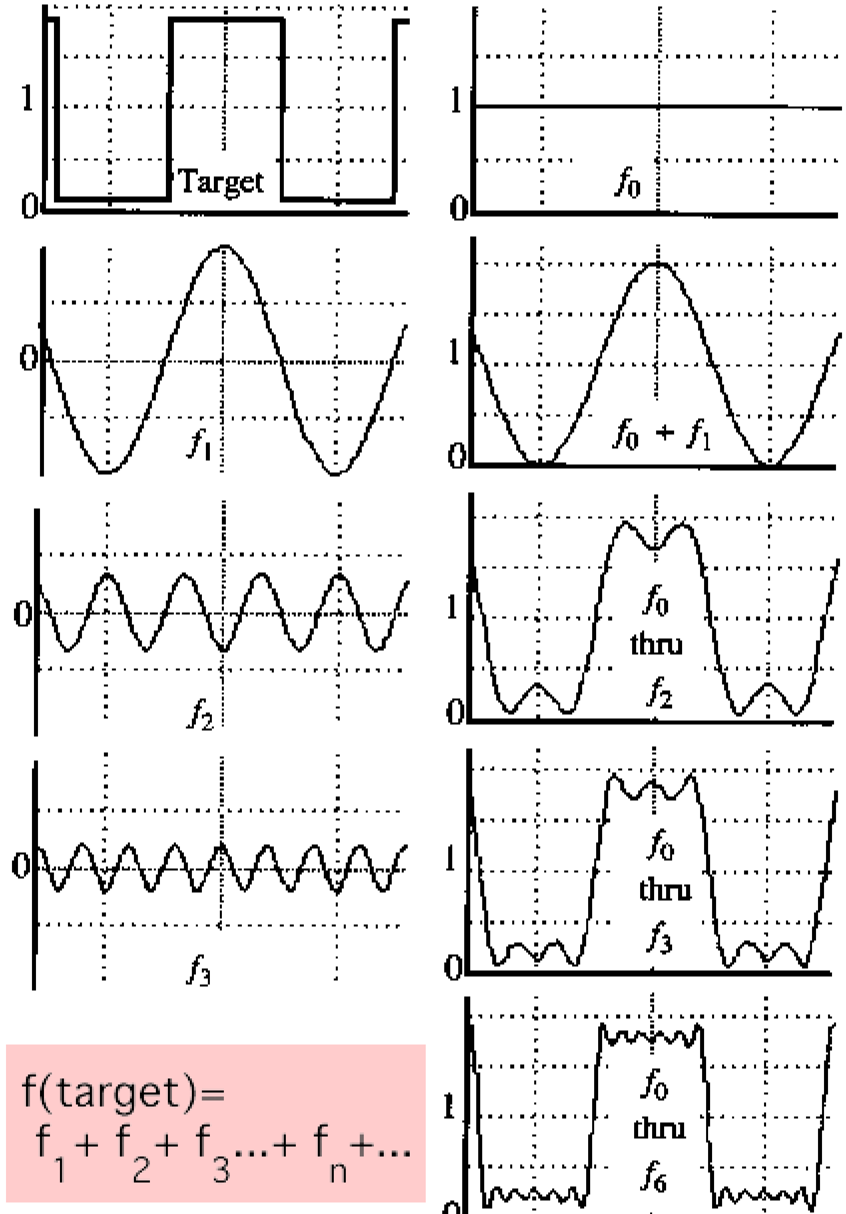
$$A \sin(\omega x + \phi)$$

Add enough of them to get any signal *f(x)* you want!

How many degrees of freedom?

What does each control?

Which one encodes the coarse vs. fine structure of the signal?



f(target)= $f_1 + f_2 + f_3 ... + f_n + ...$

Fourier Transform

We want to understand the frequency ω of our signal. So, let's reparametrize the signal by ω instead of *x*:

f(x) → Fourier Transform → ***F(ω)***

For every ω from 0 to inf, ***F(ω)*** holds the amplitude *A* and phase ϕ of the corresponding sine $A\sin(\omega x + \phi)$

- How does *F* hold both?

$$F(\omega) = R(\omega) + iI(\omega)$$

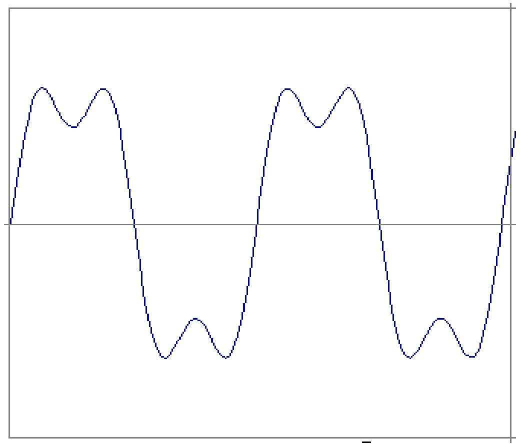
$$A = \pm\sqrt{R(\omega)^2 + I(\omega)^2} \qquad \phi = \tan^{-1}\frac{I(\omega)}{R(\omega)}$$

We can always go back:

F(ω) → Inverse Fourier Transform → ***f(x)***

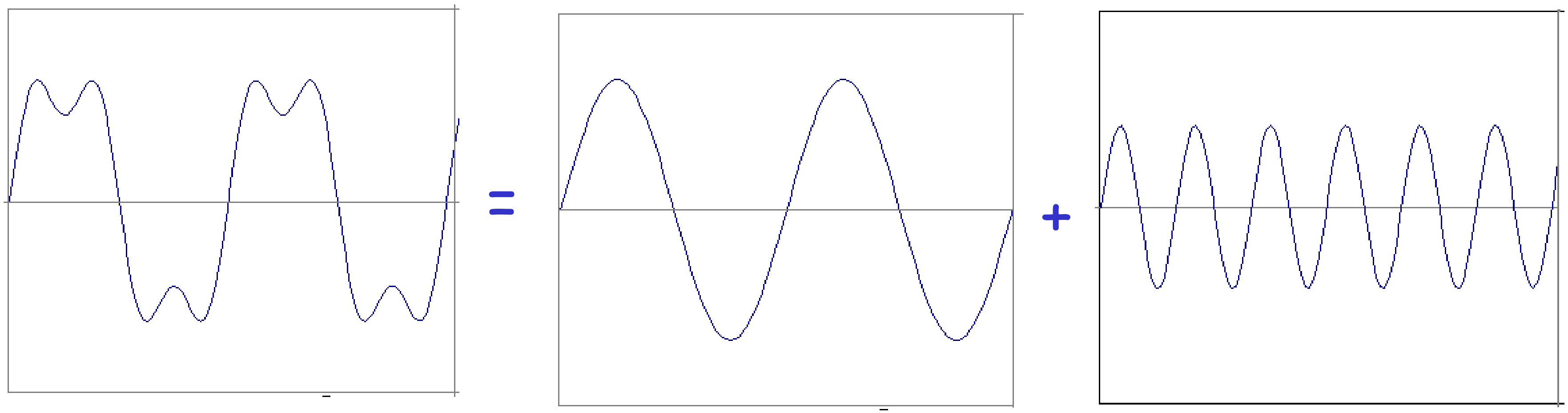
Time and Frequency

example : $g(t) = \sin(2pf\, t) + (1/3)\sin(2p(3f)\, t)$



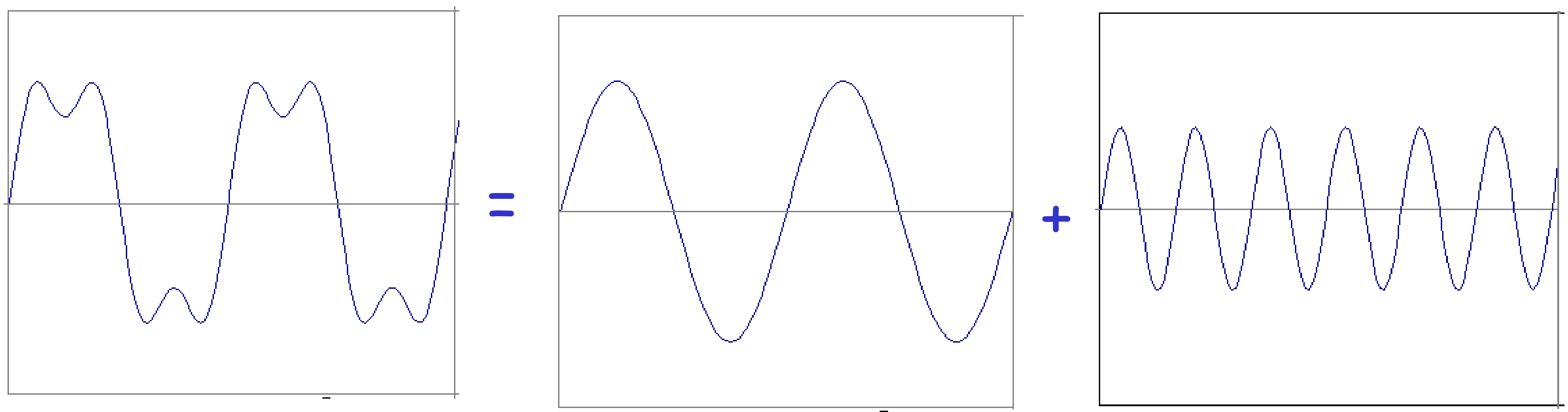
Time and Frequency

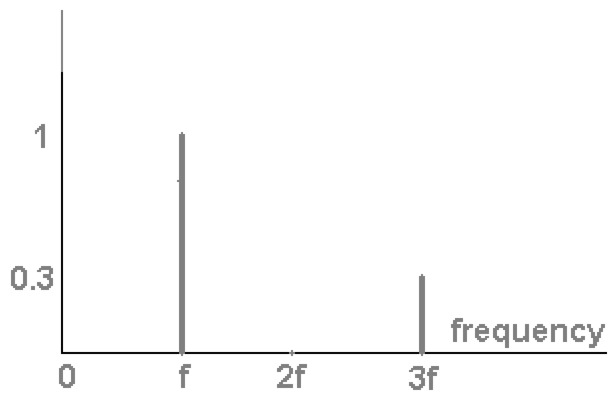
example : $g(t) = \sin(2pf\ t) + (1/3)\sin(2p(3f)\ t)$



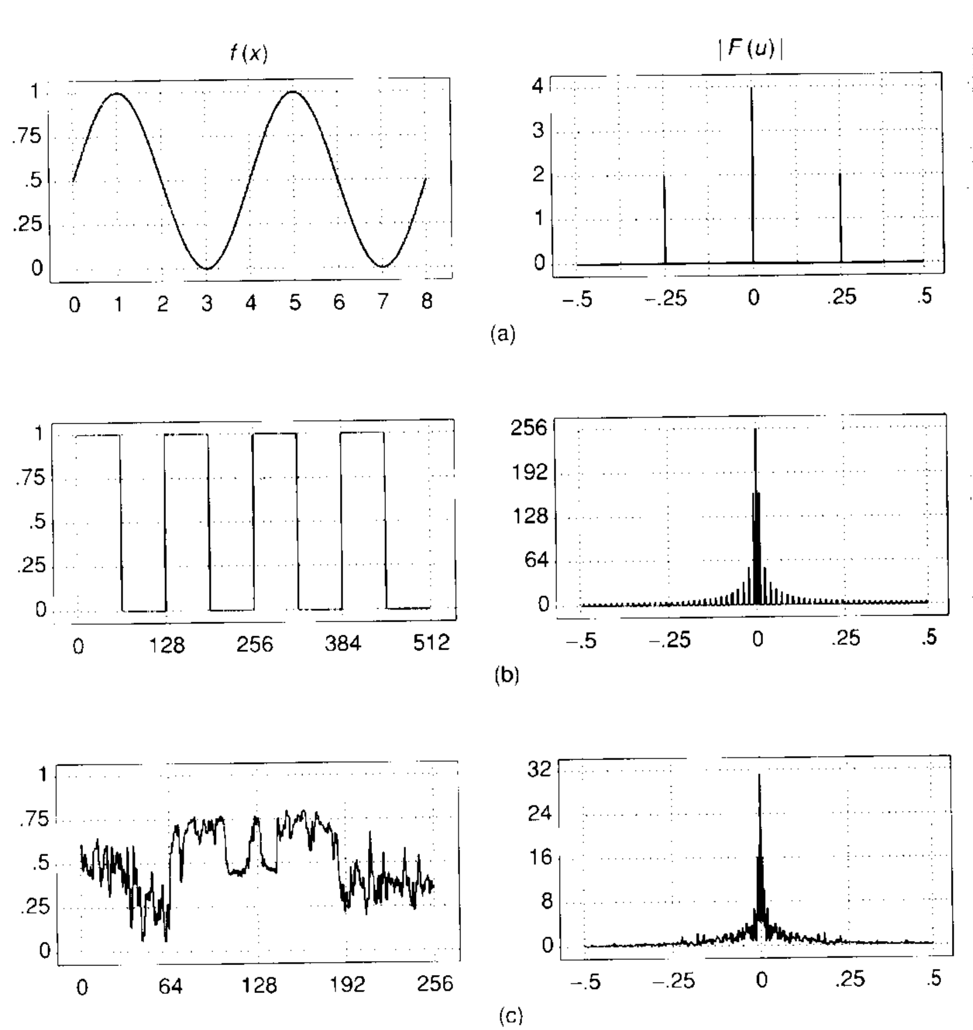
Frequency Spectra

example : $g(t) = \sin(2pf\ t) + (1/3)\sin(2p(3f)\ t)$



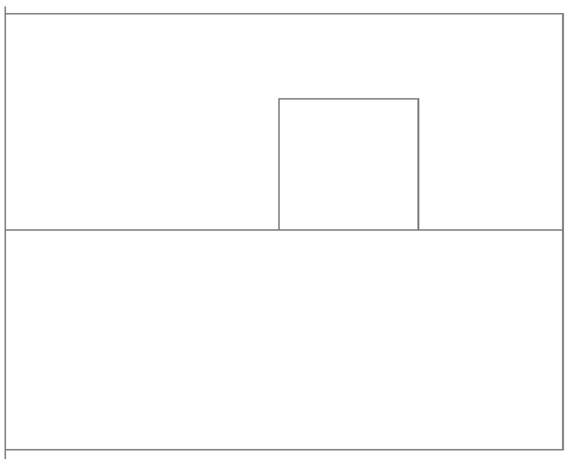


Various Frequency Spectra

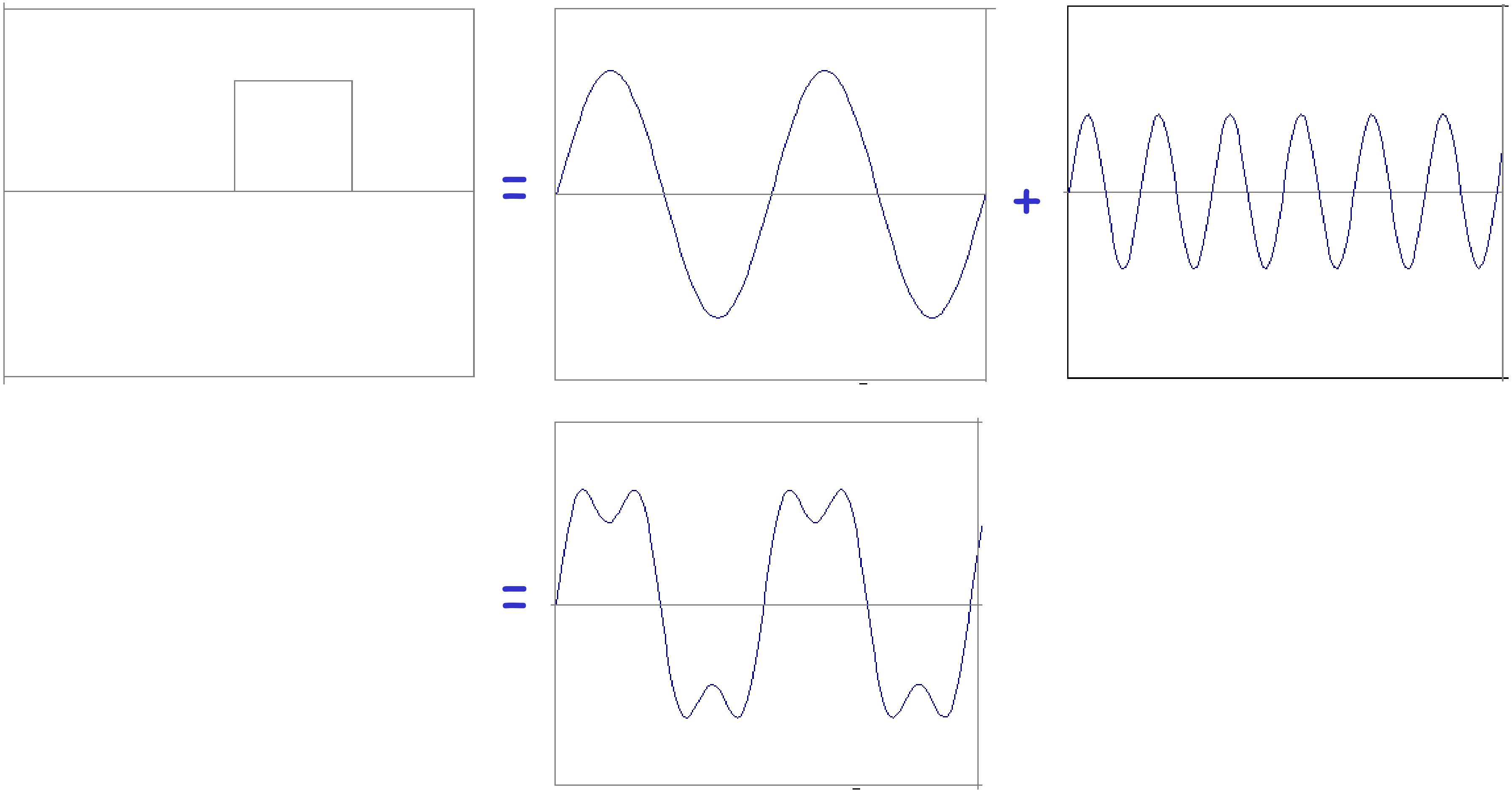


Frequency Spectra

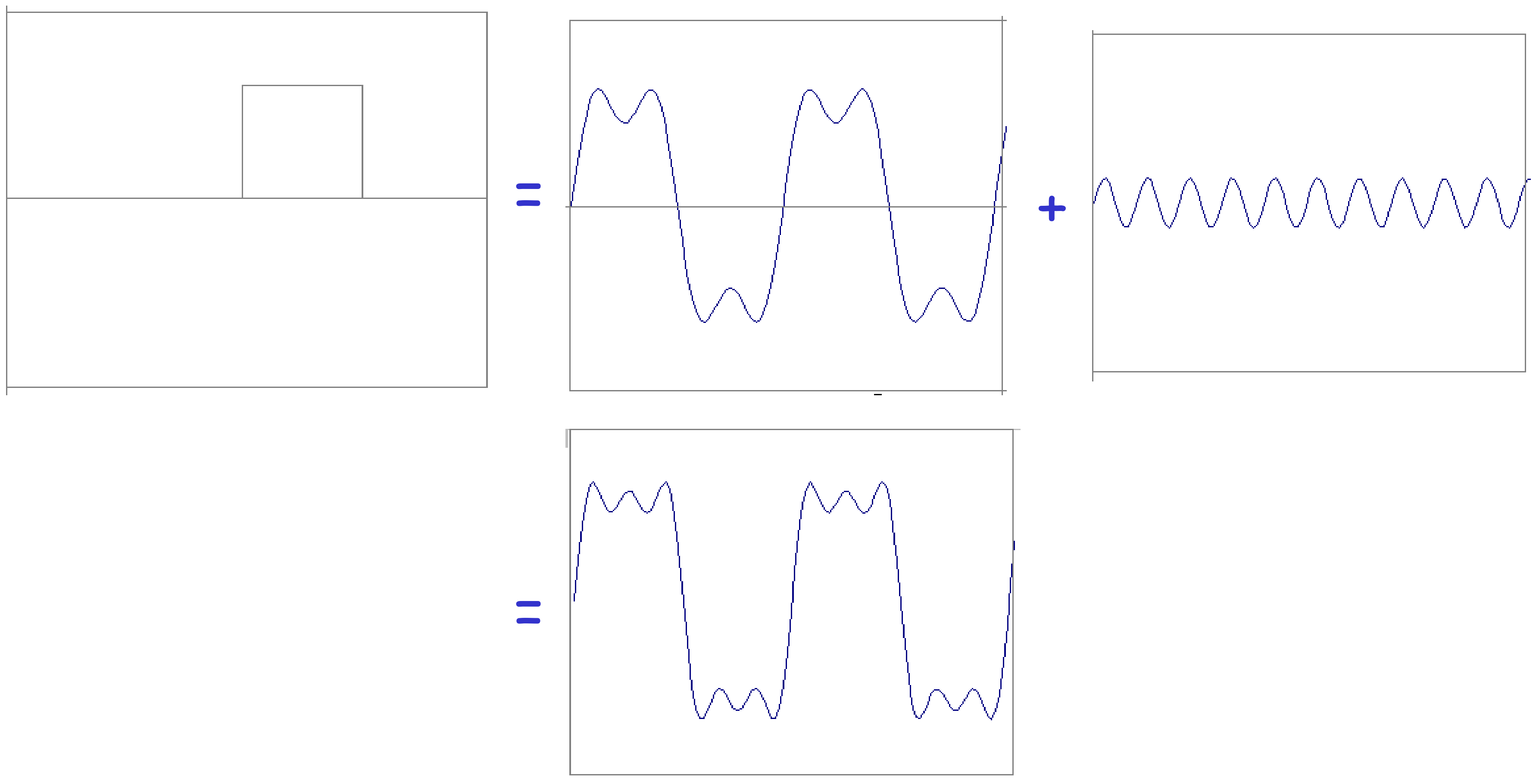
Usually, frequency is more interesting than the phase



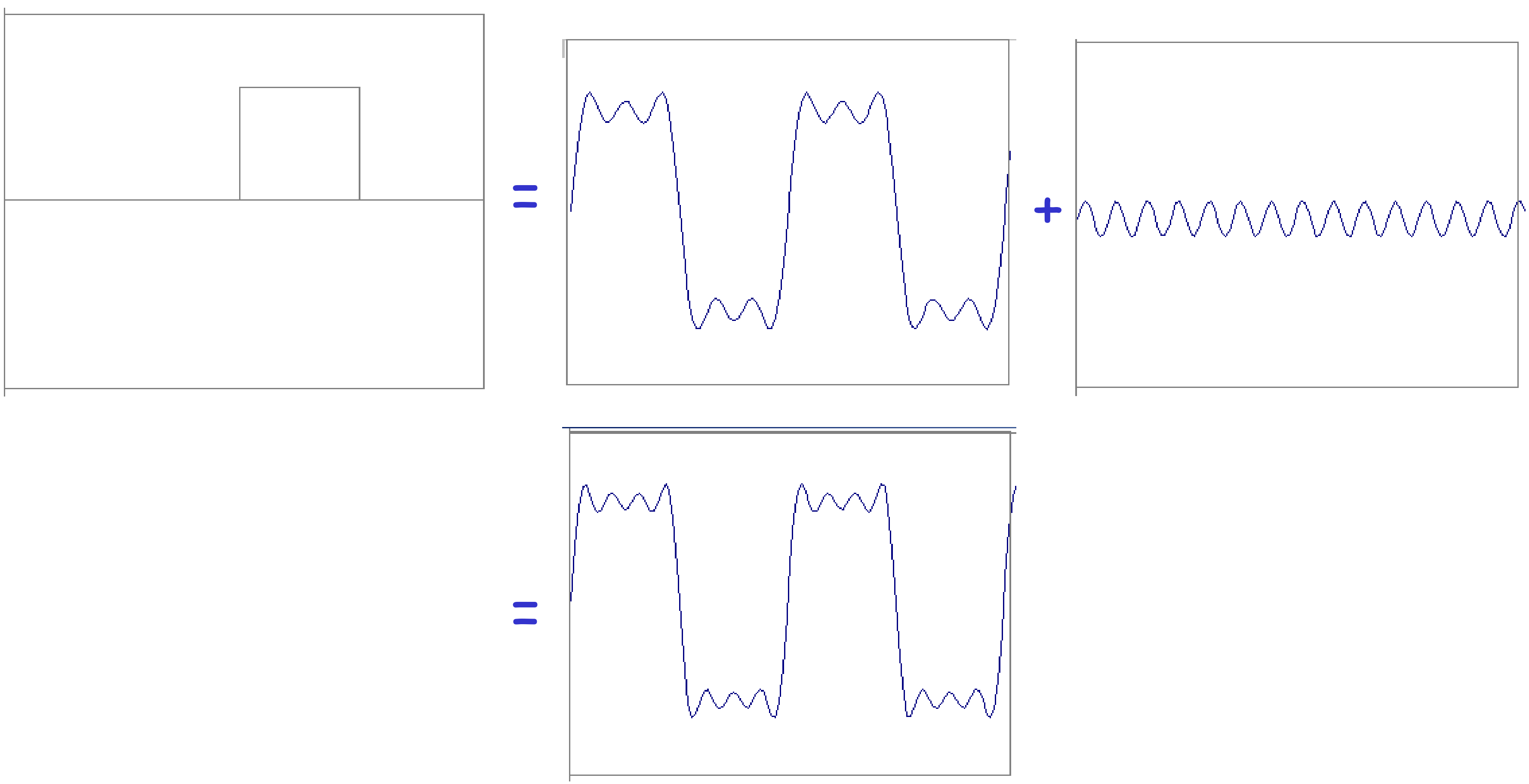
Frequency Spectra



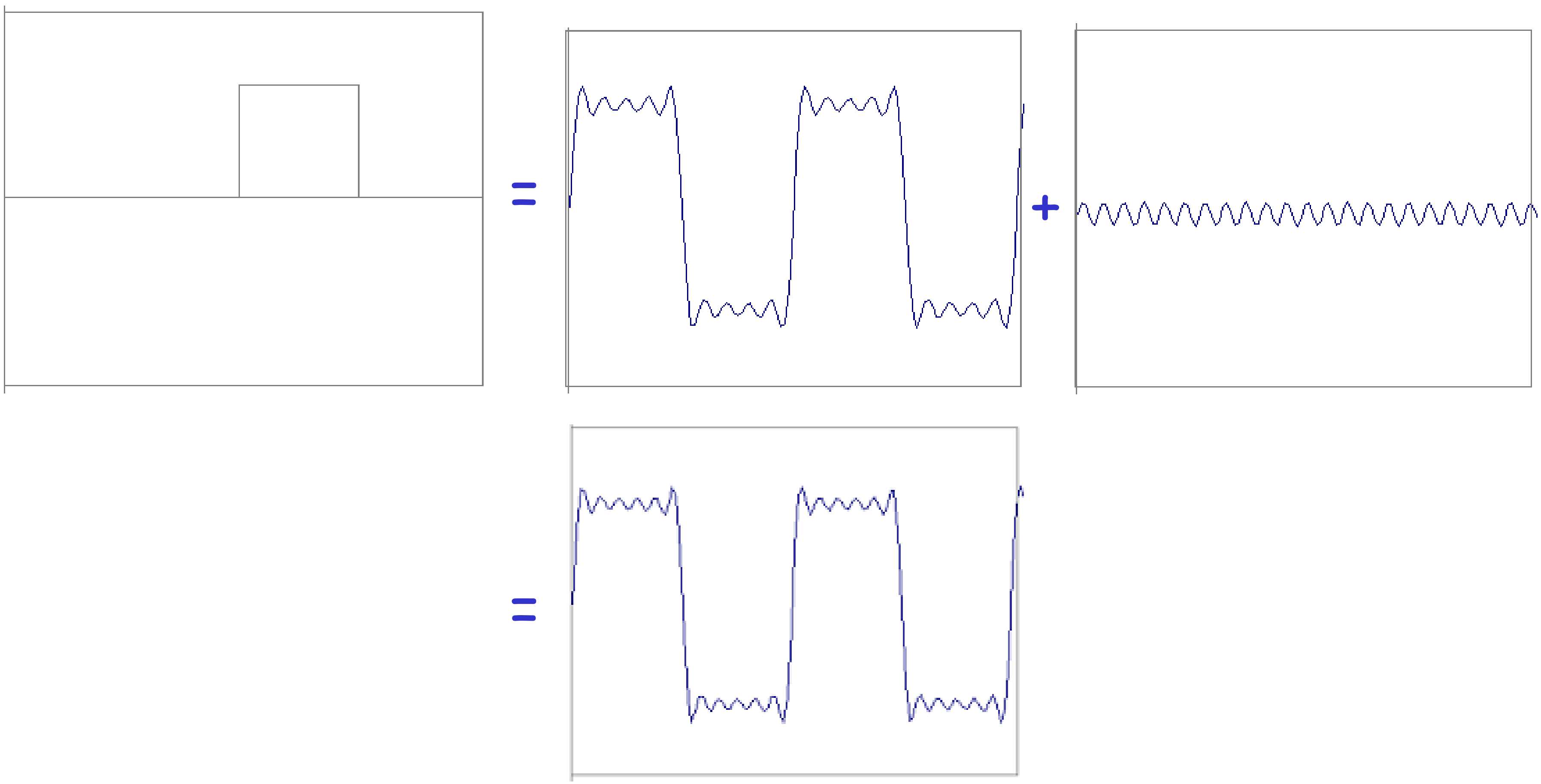
Frequency Spectra



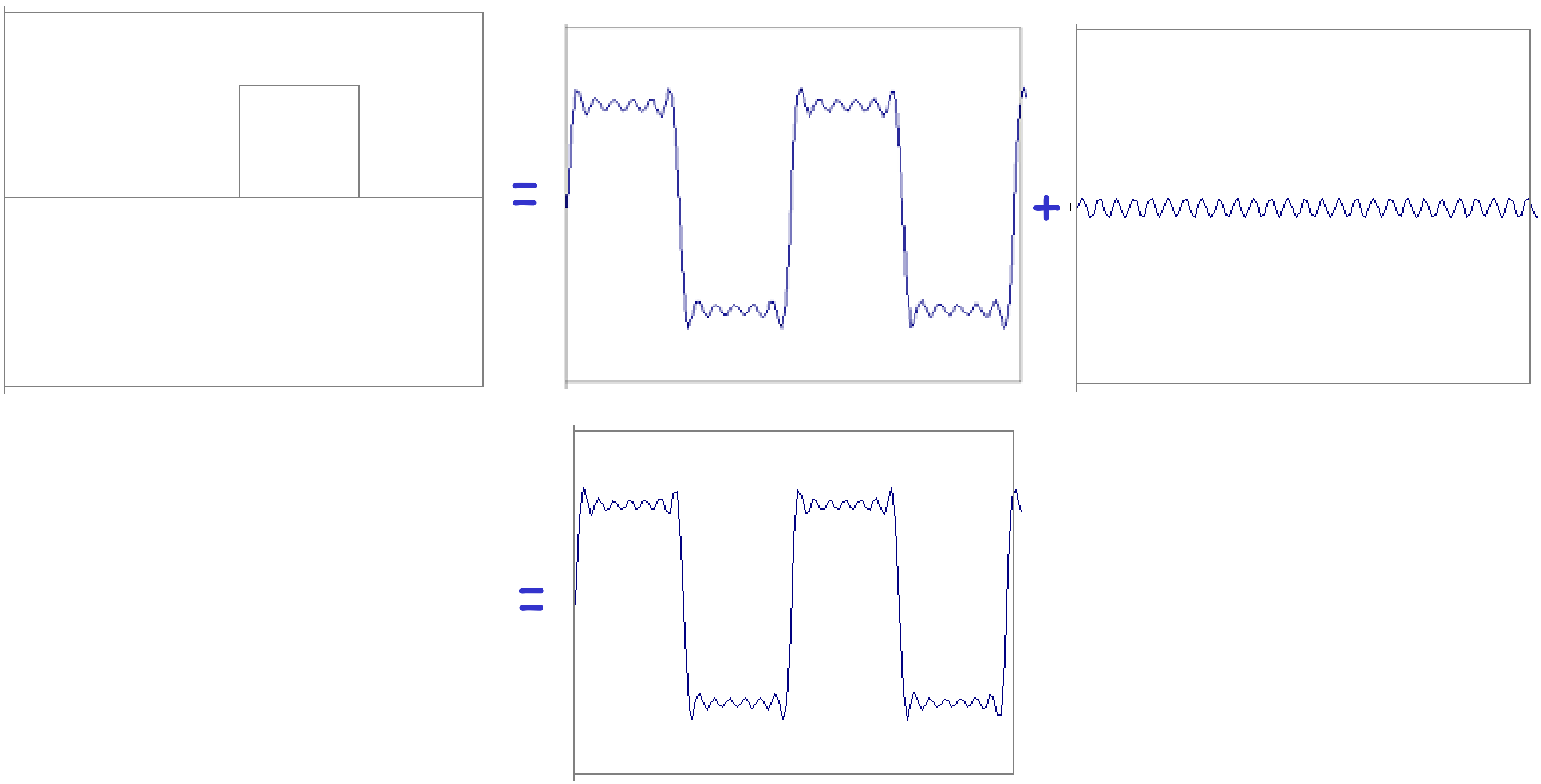
Frequency Spectra



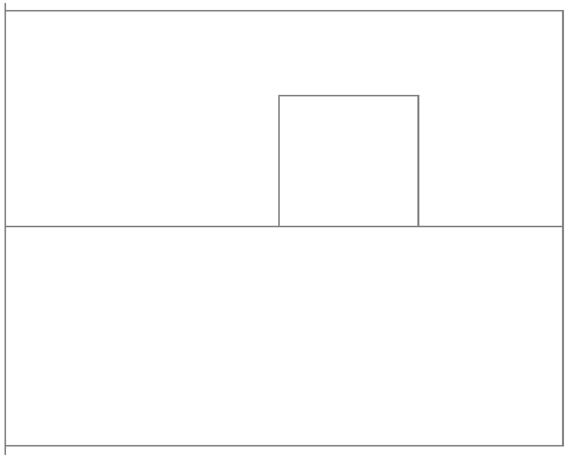
Frequency Spectra



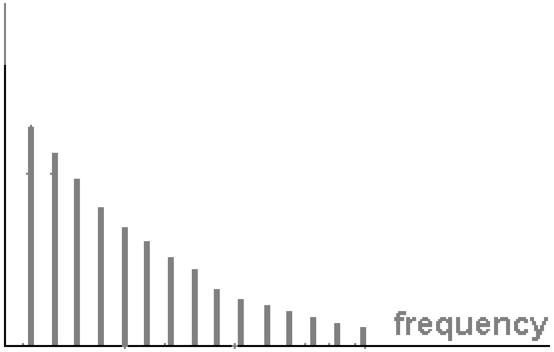
Frequency Spectra



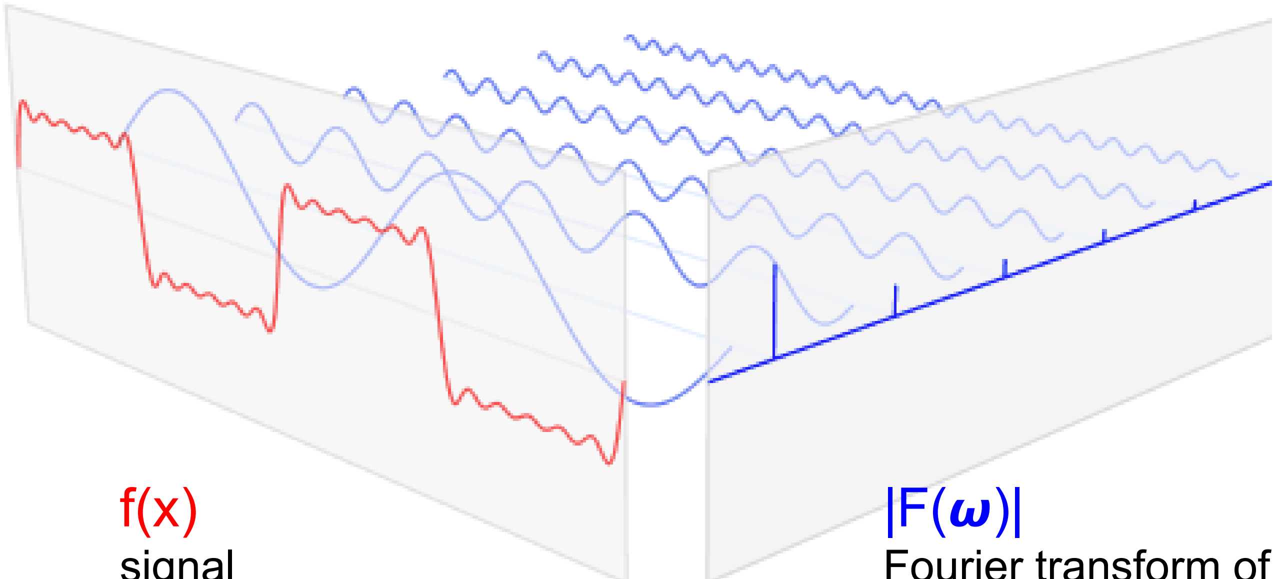
Frequency Spectra



$$= A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)$$



Signal and its Furrier Transform

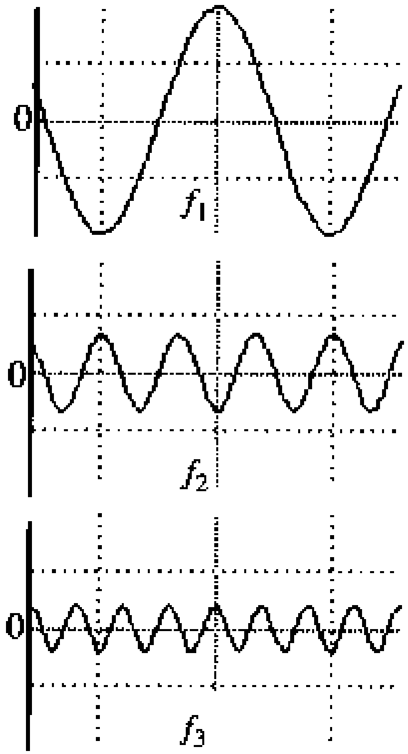


f(x)
signal

$|F(\omega)|$
Fourier transform of signal

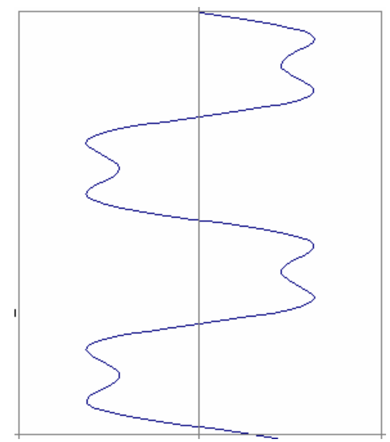
FT: Just a change of basis

$$M * f(x) = F(\omega)$$



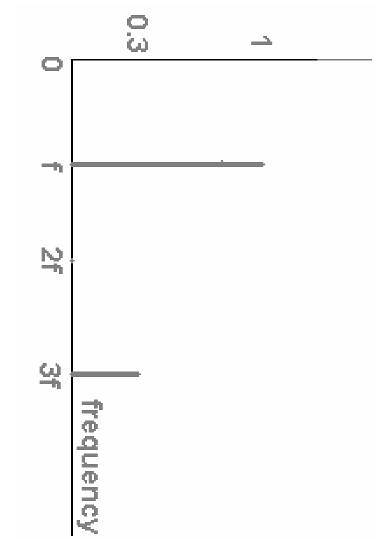
.
.
. F x N

*



N x 1

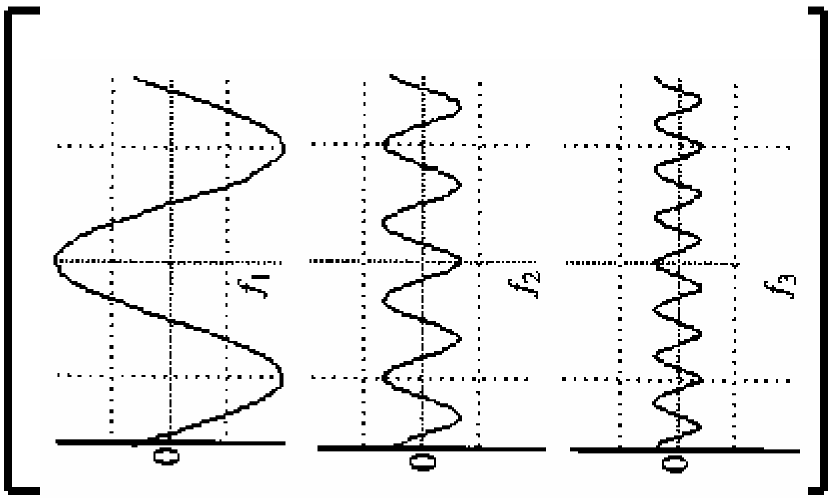
=



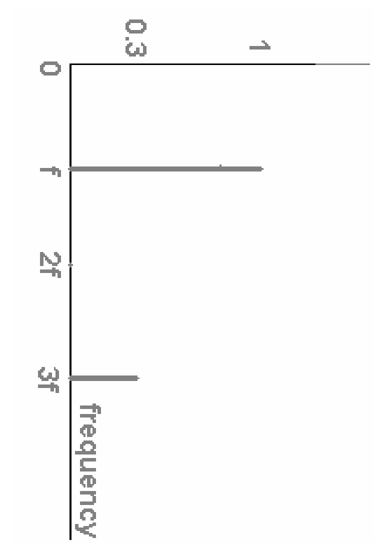
F x 1

IFT: Just a change of basis

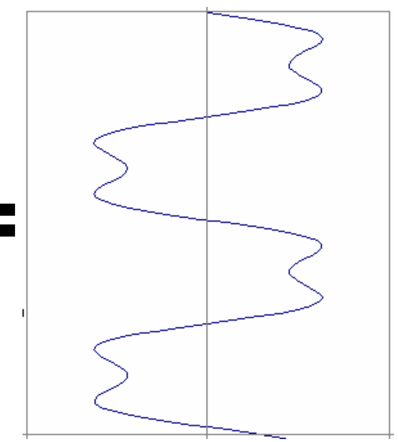
$$M^{-1} * F(\omega) = f(x)$$



*



=



-
-
- N x F   F x 1   N x 1

Finally: Scary Math

$$\text{Fourier Transform}: \quad F(\omega) = \int_{-\infty}^{+\infty} f(x) e^{-i\omega x} dx$$

$$\text{Inverse Fourier Transform}: \quad f(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega) e^{i\omega x} d\omega$$

Finally: Scary Math

$$\text{Fourier Transform}: \quad F(\omega) = \int_{-\infty}^{+\infty} f(x) e^{-i\omega x} dx$$

$$\text{Inverse Fourier Transform}: \quad f(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(\omega) e^{i\omega x} d\omega$$

…not really scary: $e^{i\omega x} = \cos(\omega x) + i \sin(\omega x)$

is hiding our old friend: $\sin(\omega x + \phi)$

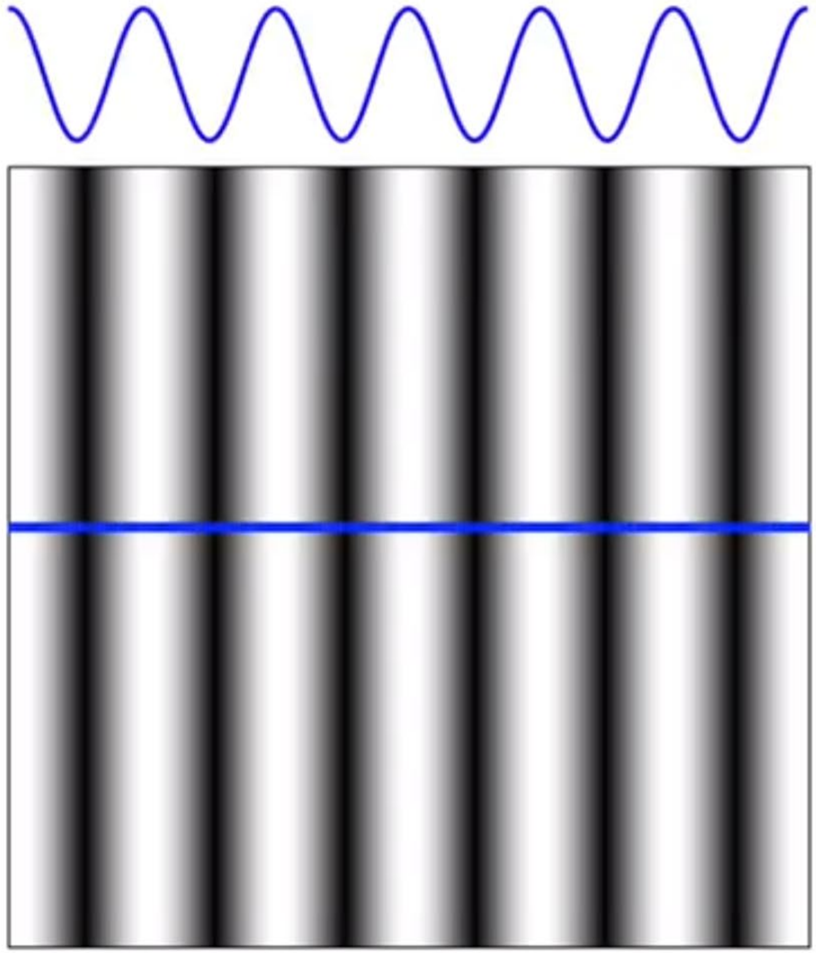
phase can be encoded by sin/cos pair →

$$P\cos(x) + Q\sin(x) = A\sin(x + \phi)$$

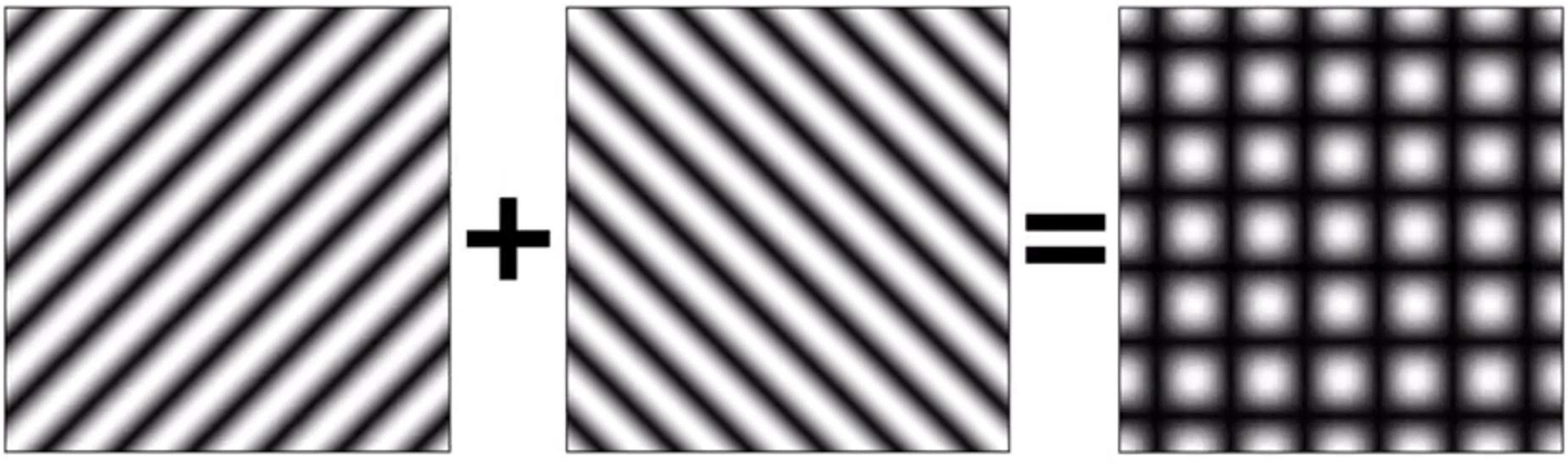
$$A = \pm\sqrt{P^2 + Q^2} \qquad \phi = \tan^{-1}\left(\frac{P}{Q}\right)$$

So it's just our signal *f(x)* times sine at frequency ω

Extending to 2D



Addition still works in 2D



Extension to 2D

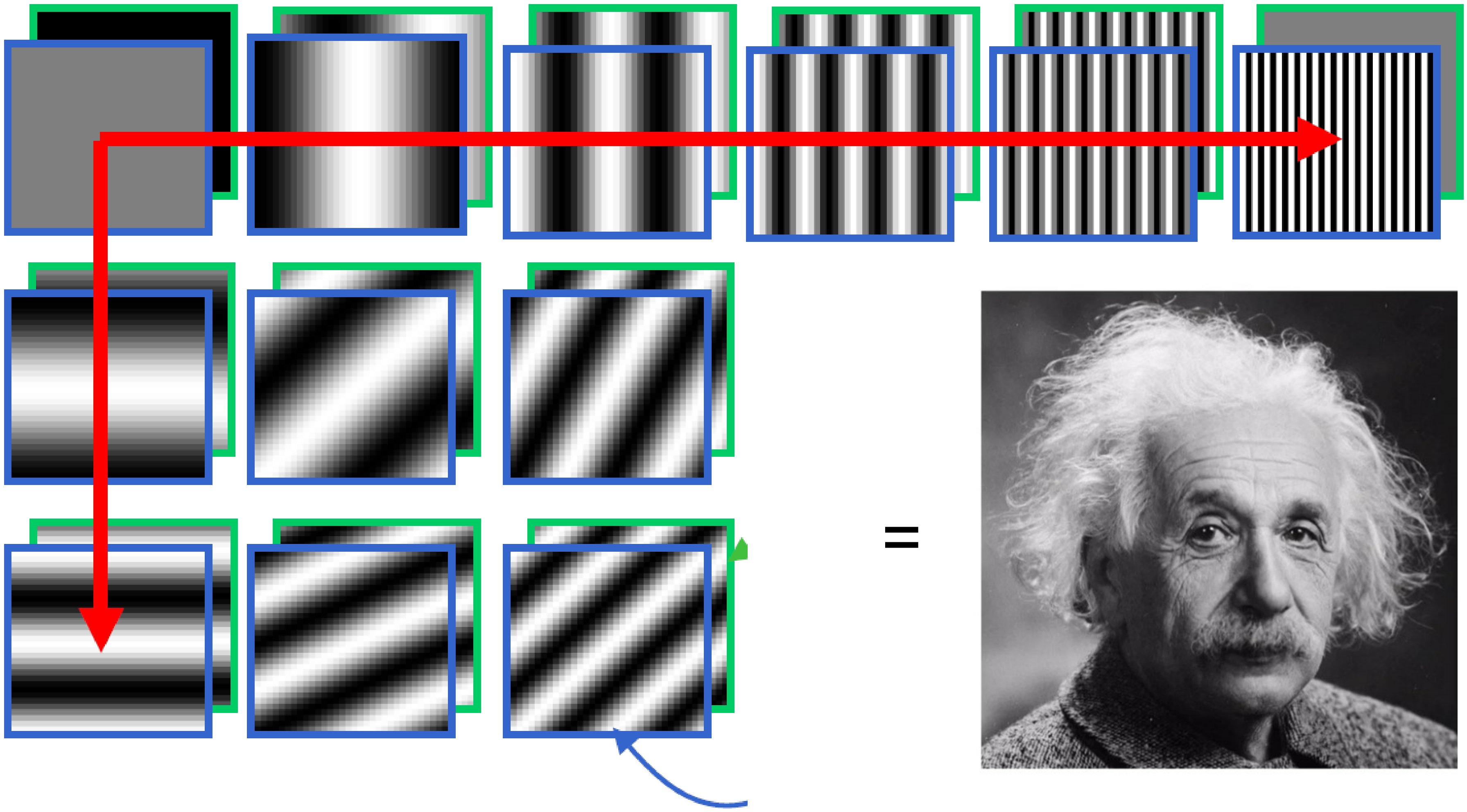
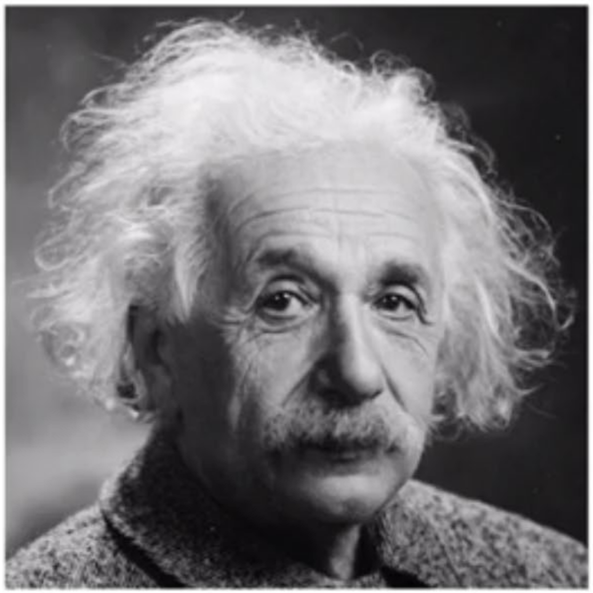
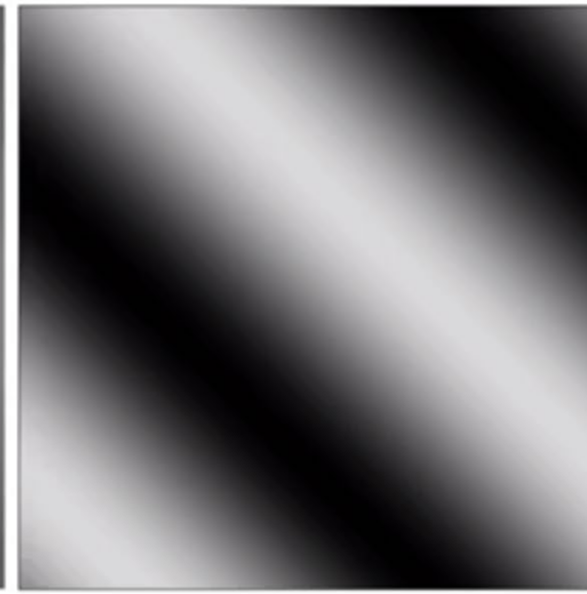
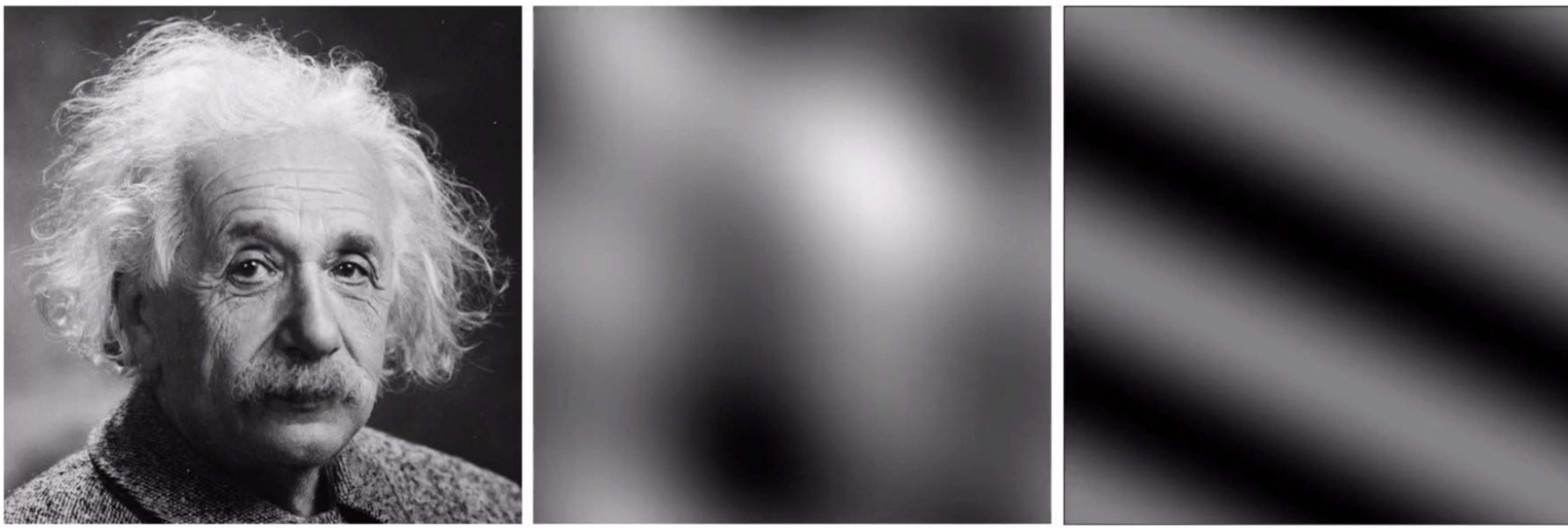


Image as a sum of basis images

Contrast x3

2



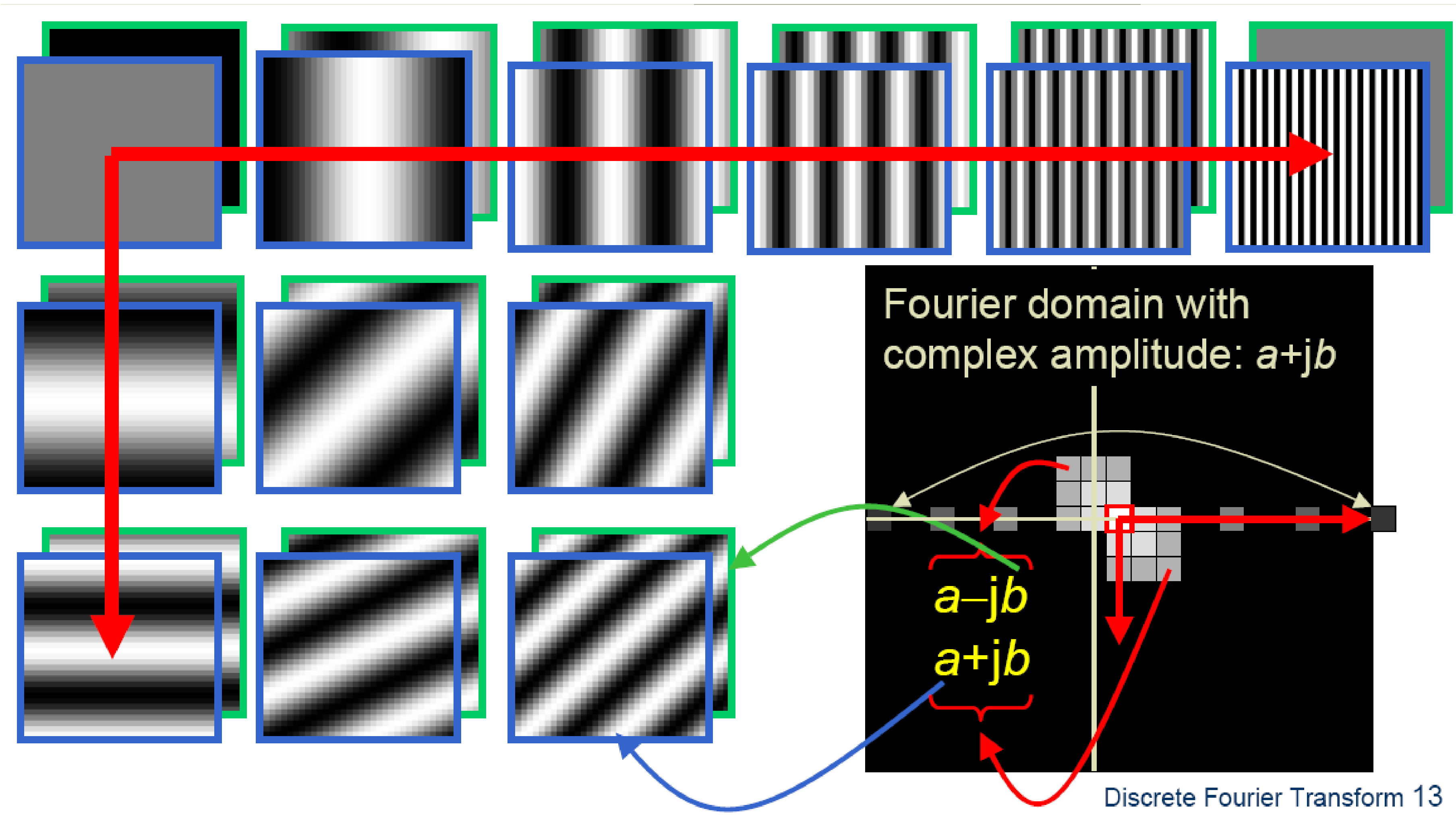
4

13

26

100

Extension to 2D

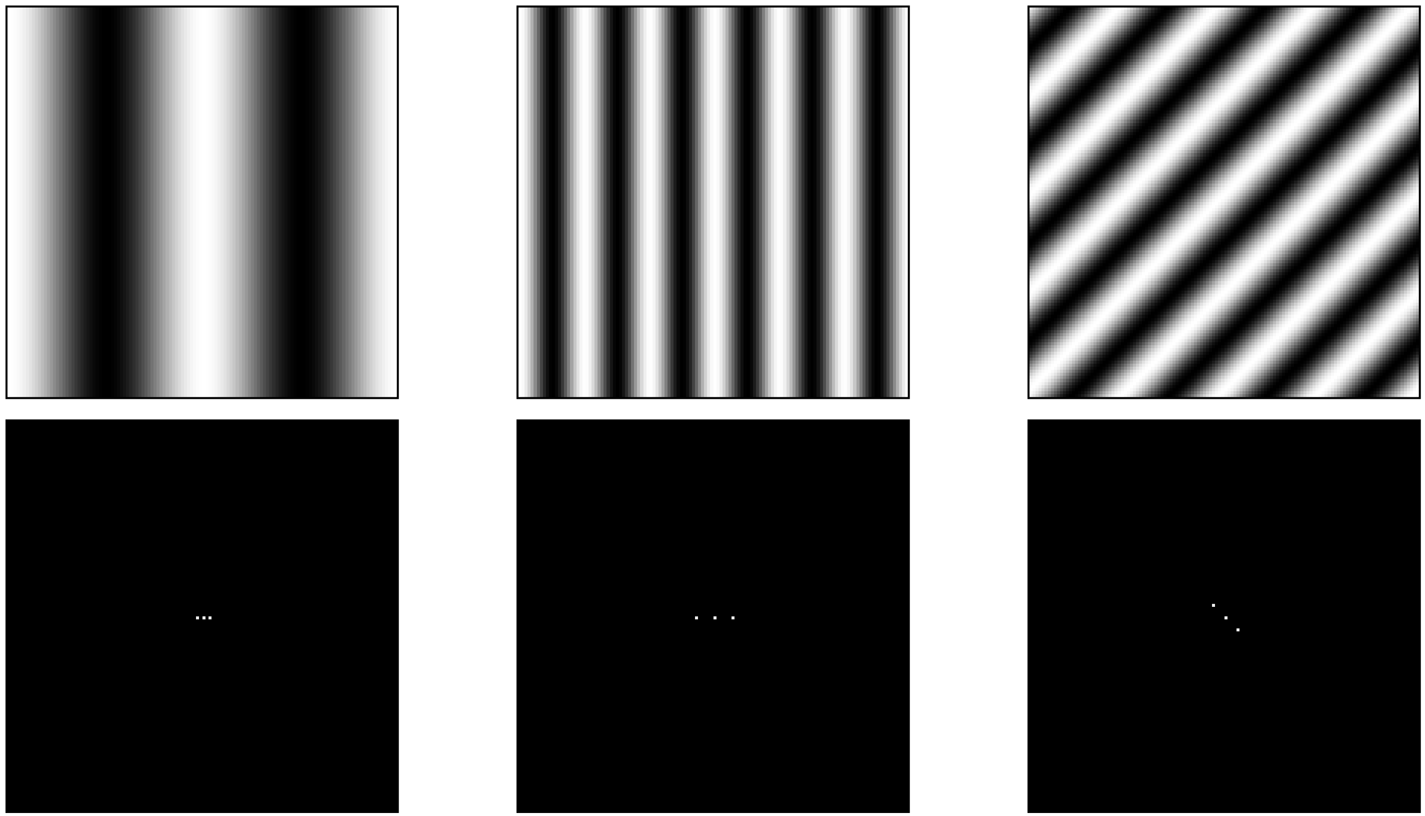


in Matlab, check out: imagesc(log(abs(fftshift(fft2(im)))));

Fourier analysis in images

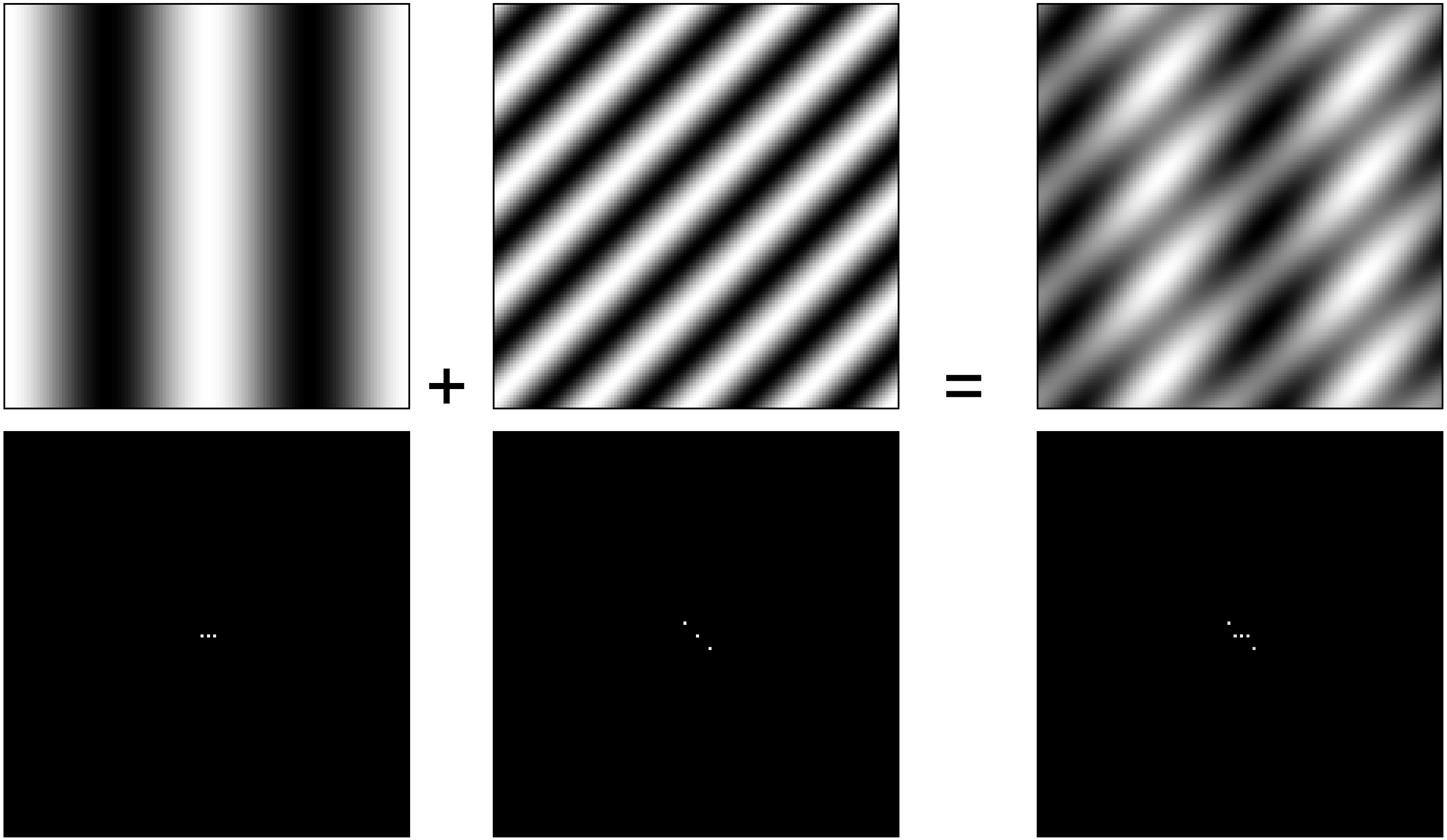
Intensity Image

Fourier Image



http://sharp.bu.edu/~slehar/fourier/fourier.html#filtering

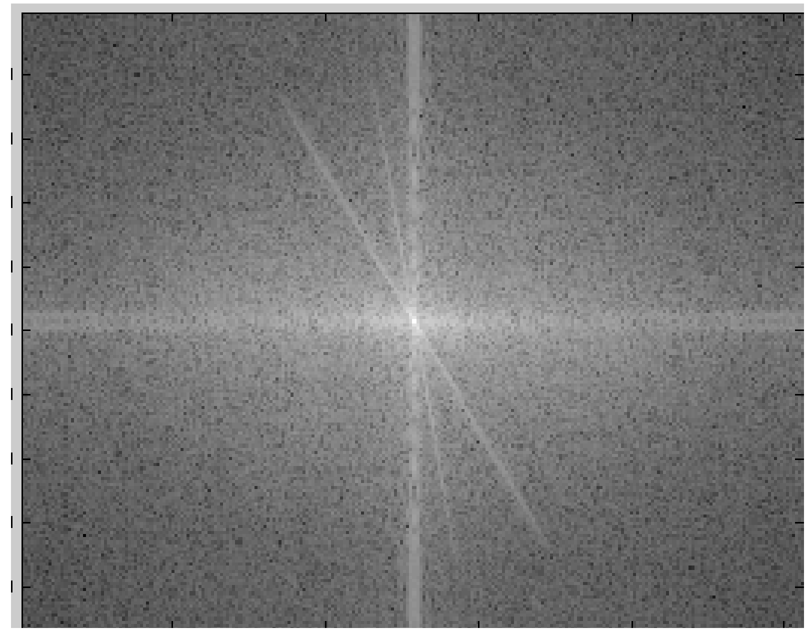
Signals can be composed



http://sharp.bu.edu/~slehar/fourier/fourier.html#filtering
More: http://www.cs.unm.edu/~brayer/vision/fourier.html

Man-made Scene

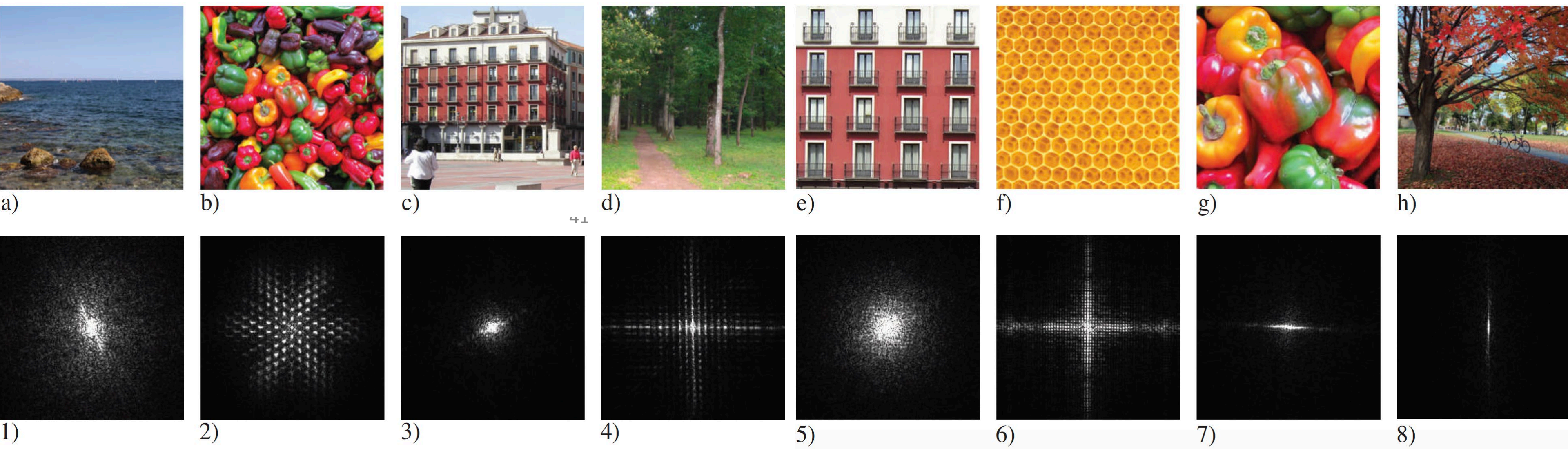
Amplitude Spectrum



Can change spectrum, then reconstruct

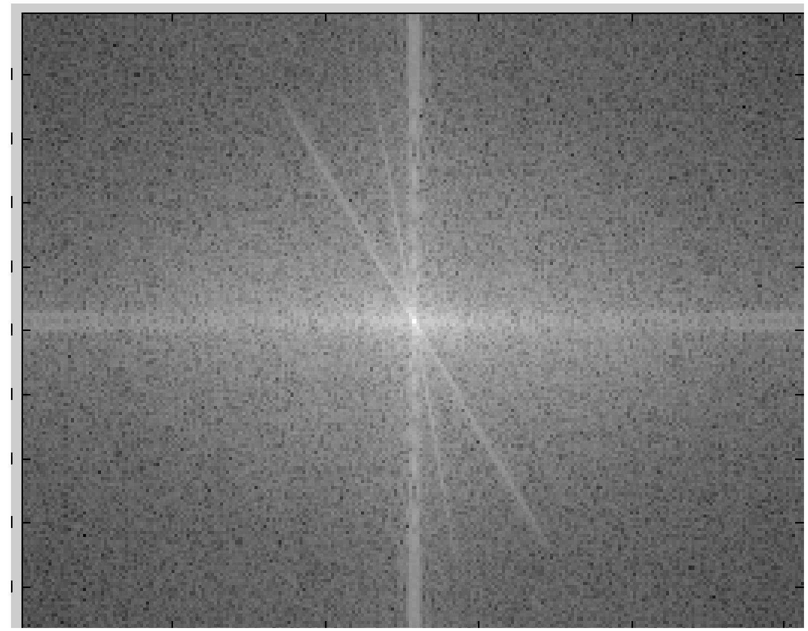
Local change in one domain, courses global change in the other

The Furrier Game: find the right pairs

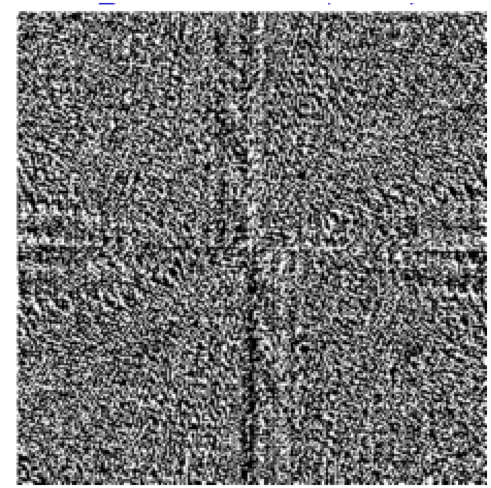


What about phase?

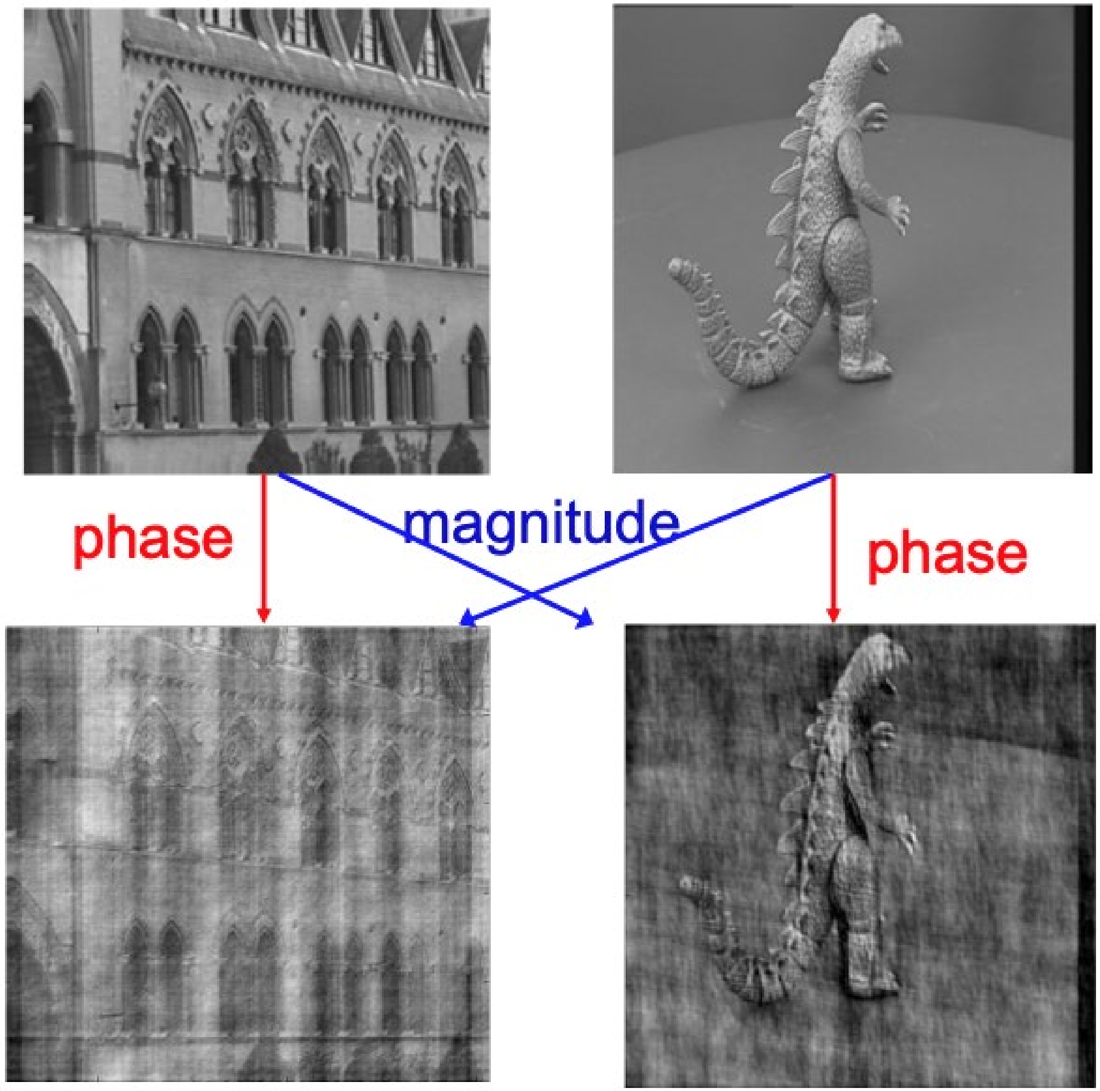
Amplitude Spectrum



what does phase look like, you ask?
(less visually informative)

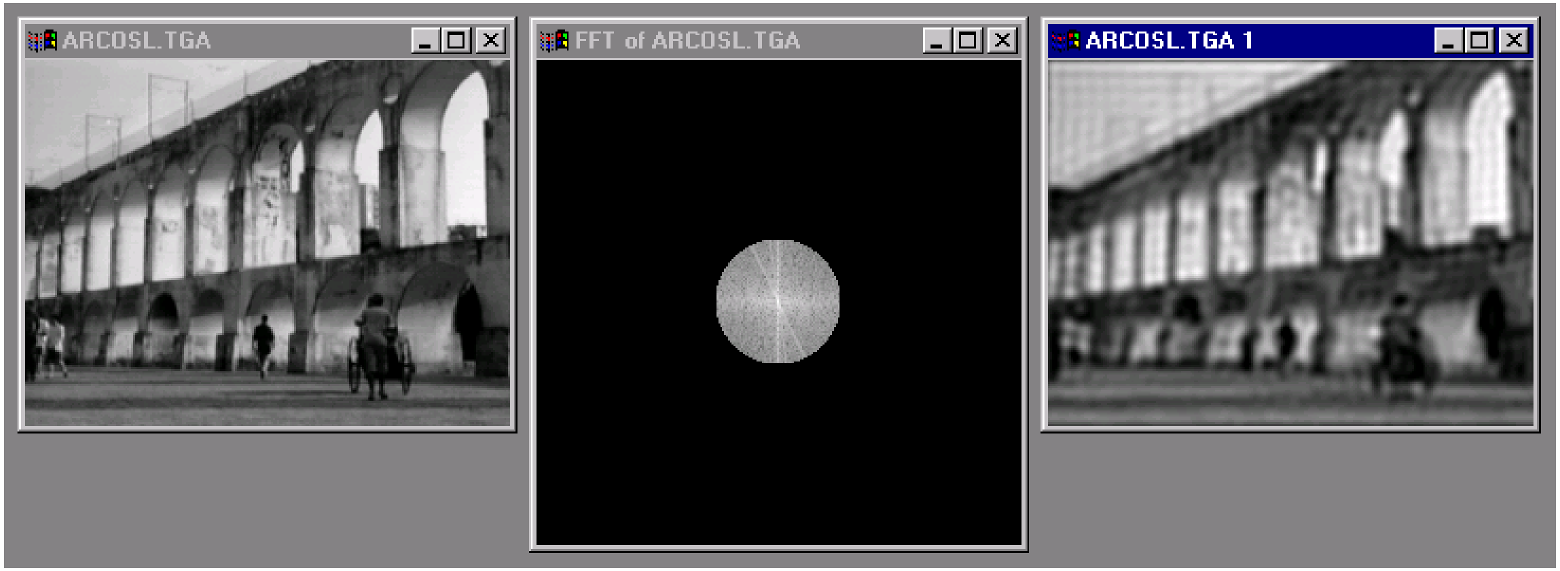


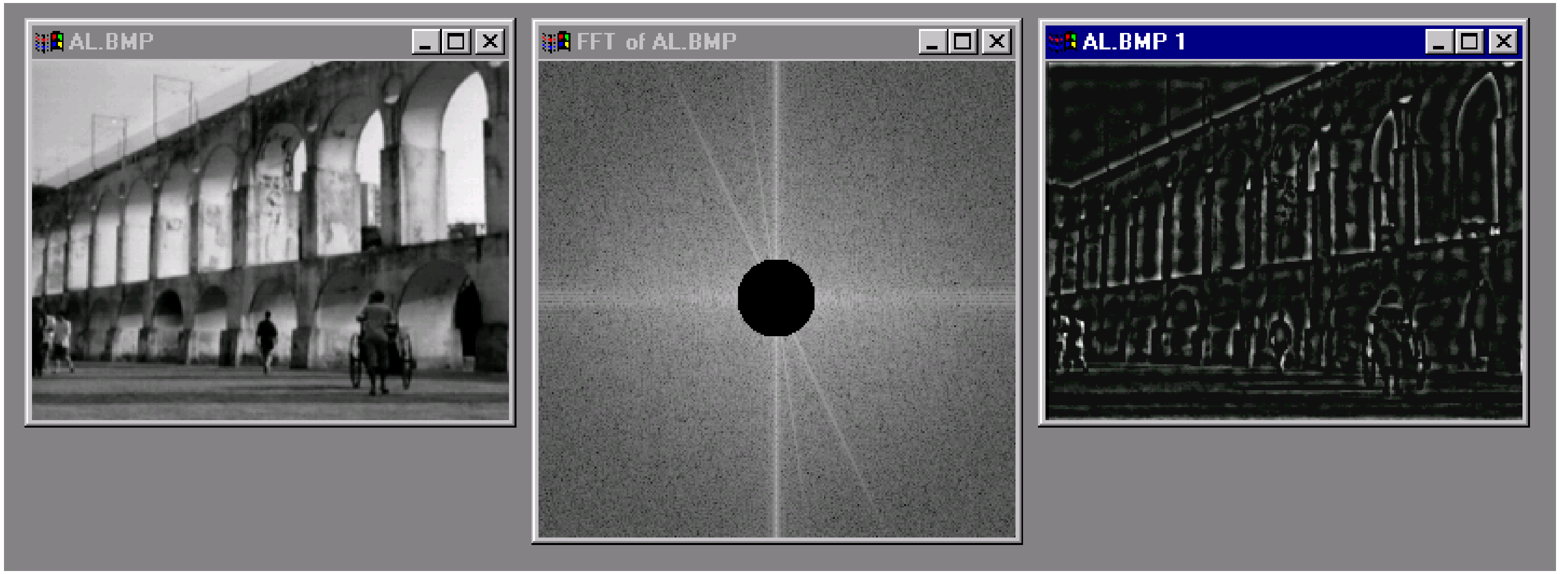
The importance of Phase



Slide: Andrew Zisserman

Low and High Pass filtering





The Convolution Theorem

The greatest thing since sliced (banana) bread!

- The Fourier transform of the convolution of two functions is the product of their Fourier transforms

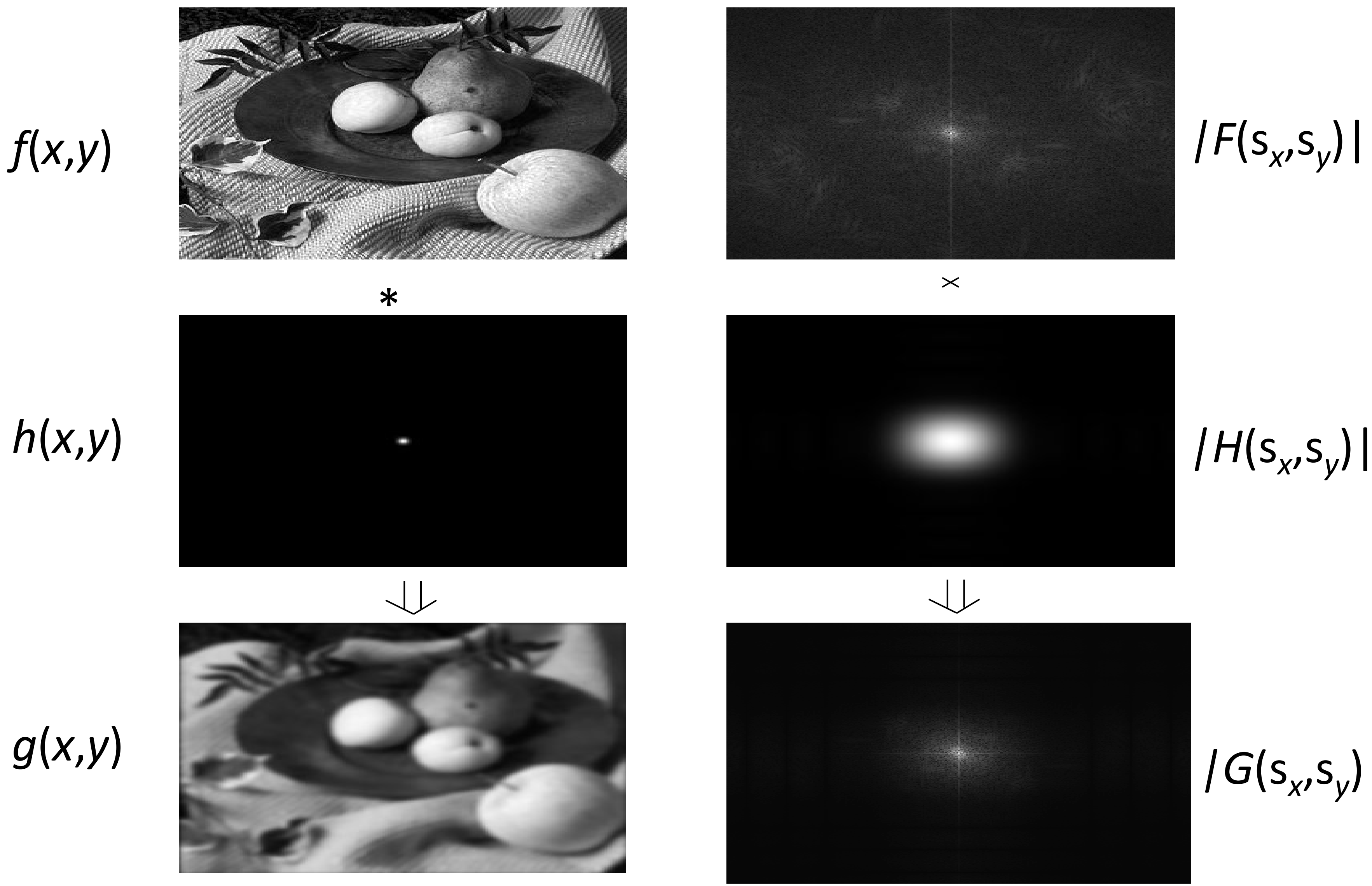
$$\mathrm{F}[g * h] = \mathrm{F}[g]\,\mathrm{F}[h]$$

- The inverse Fourier transform of the product of two Fourier transforms is the convolution of the two inverse Fourier transforms

$$\mathrm{F}^{-1}[gh] = \mathrm{F}^{-1}[g] * \mathrm{F}^{-1}[h]$$

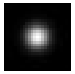
- **Convolution** in spatial domain is equivalent to **multiplication** in frequency domain!

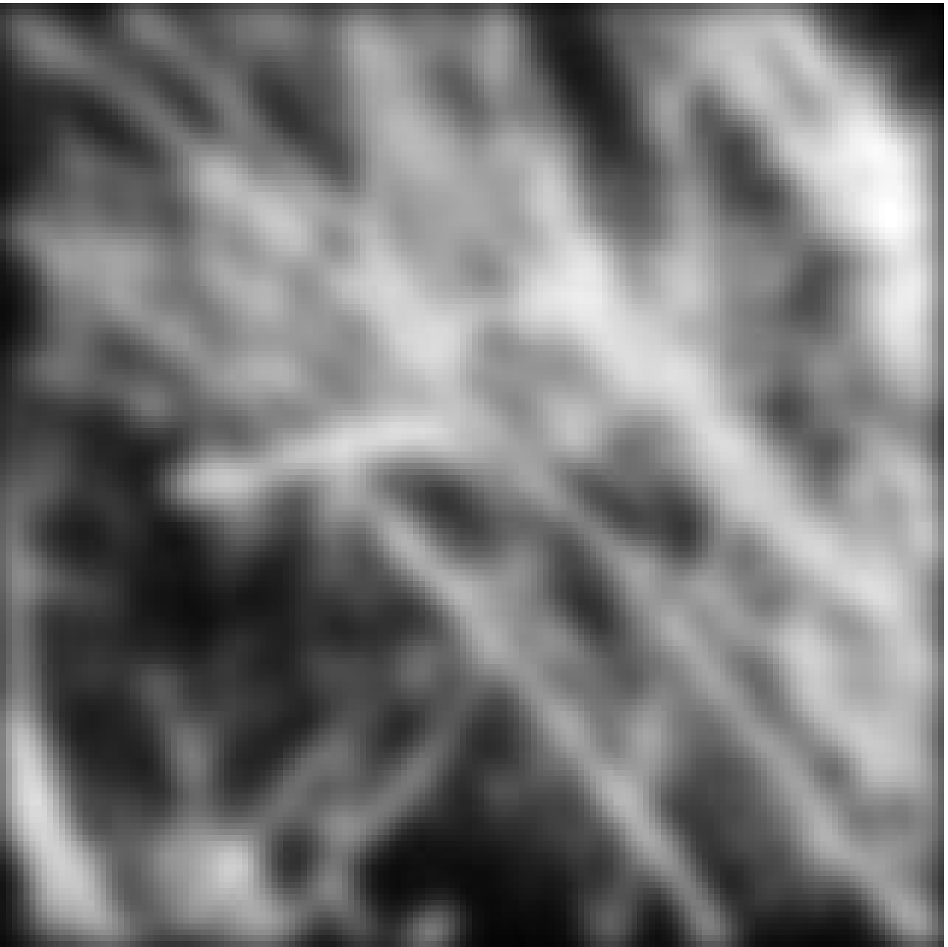
2D convolution theorem example

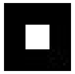


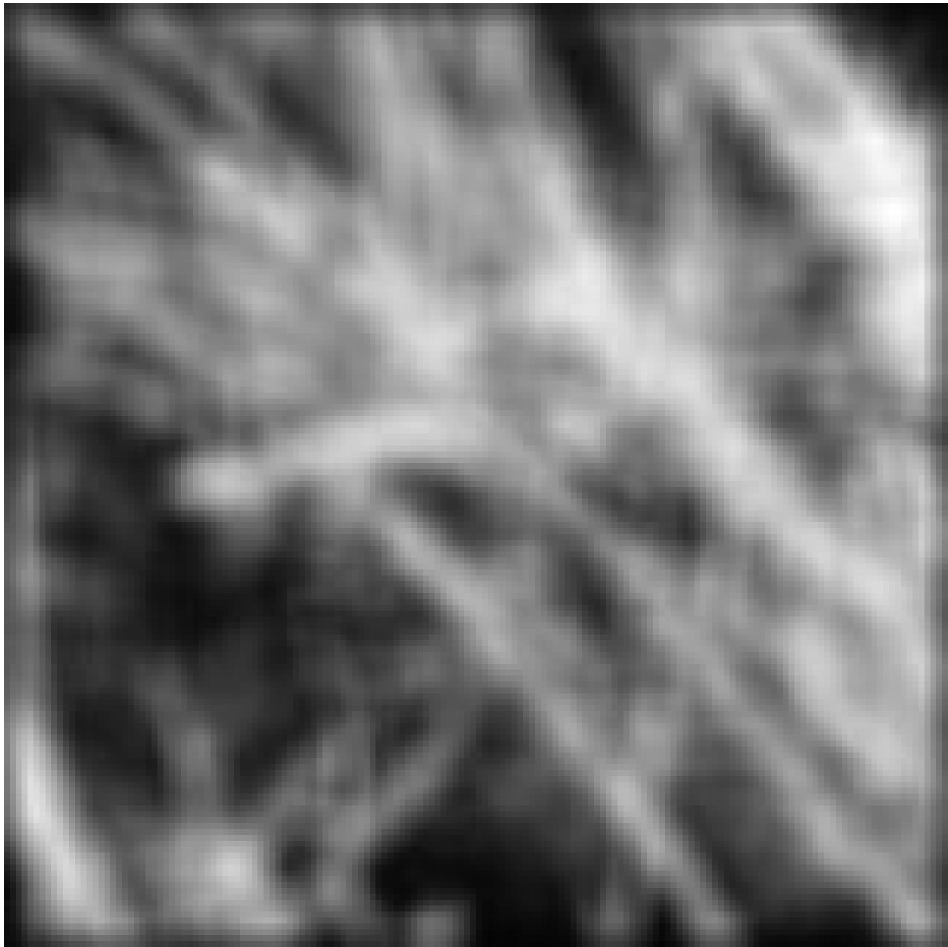
Filtering

Why does the Gaussian give a nice smooth image, but the square filter give edgy artifacts?

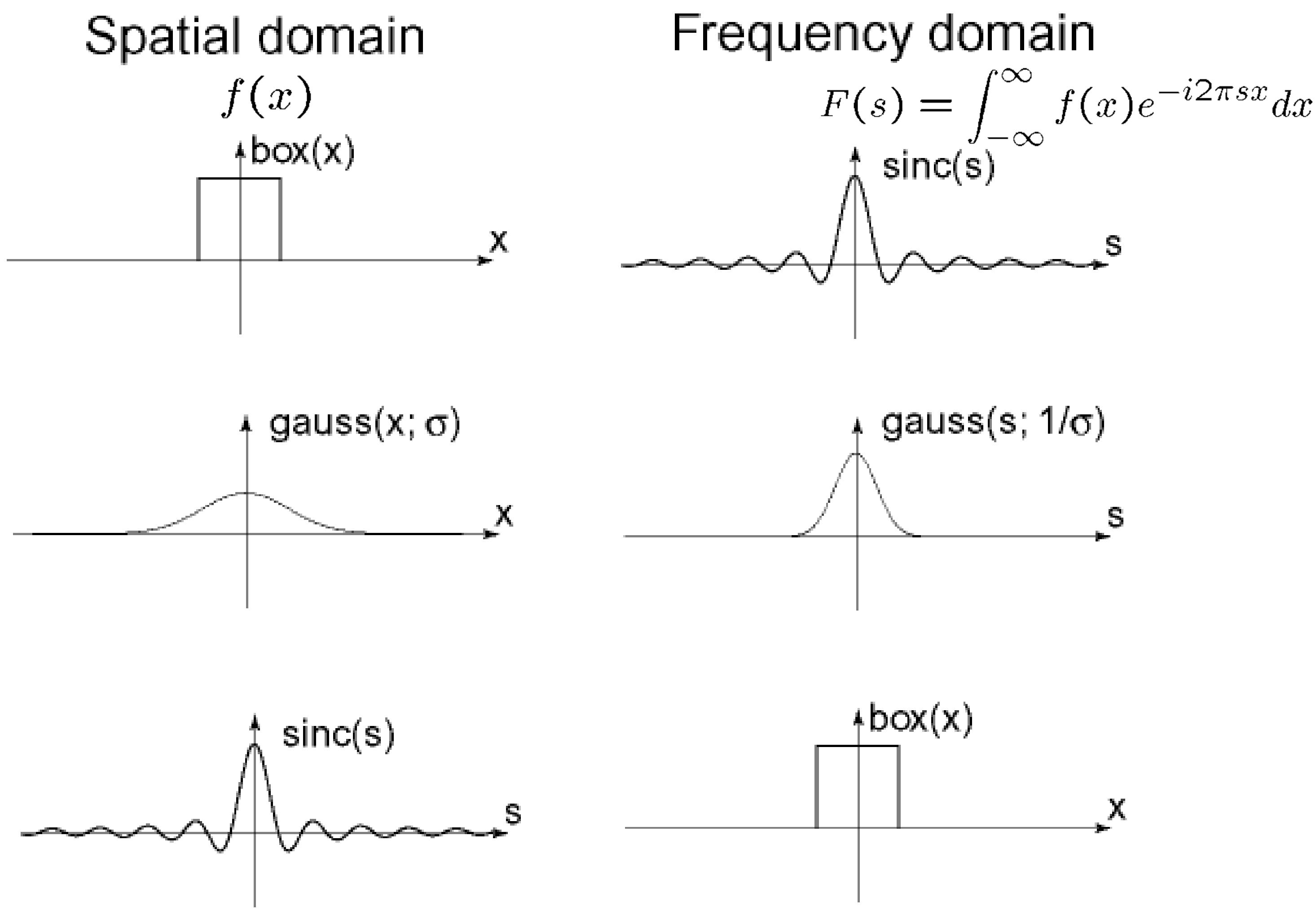
Gaussian 



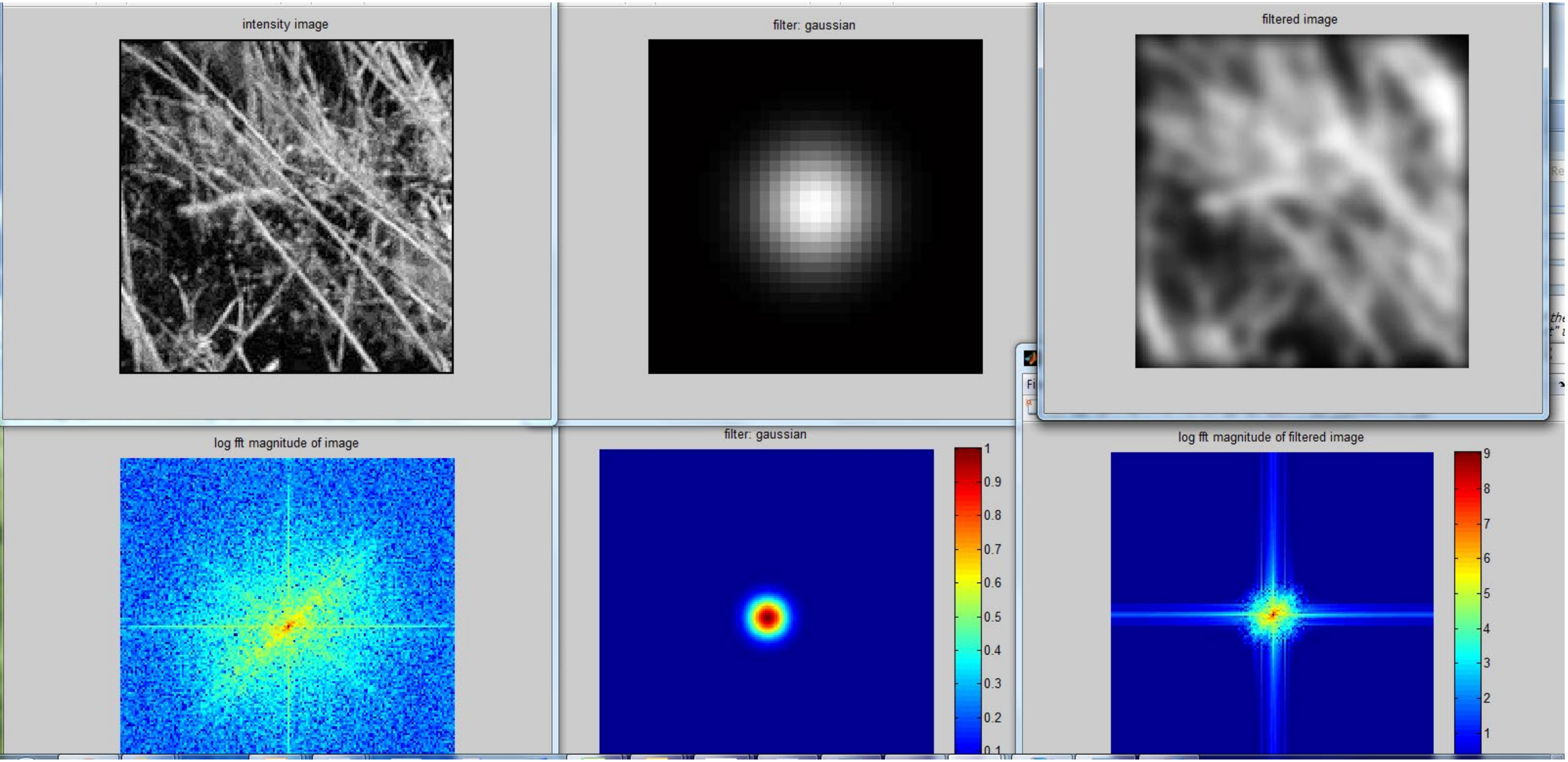
Box filter 



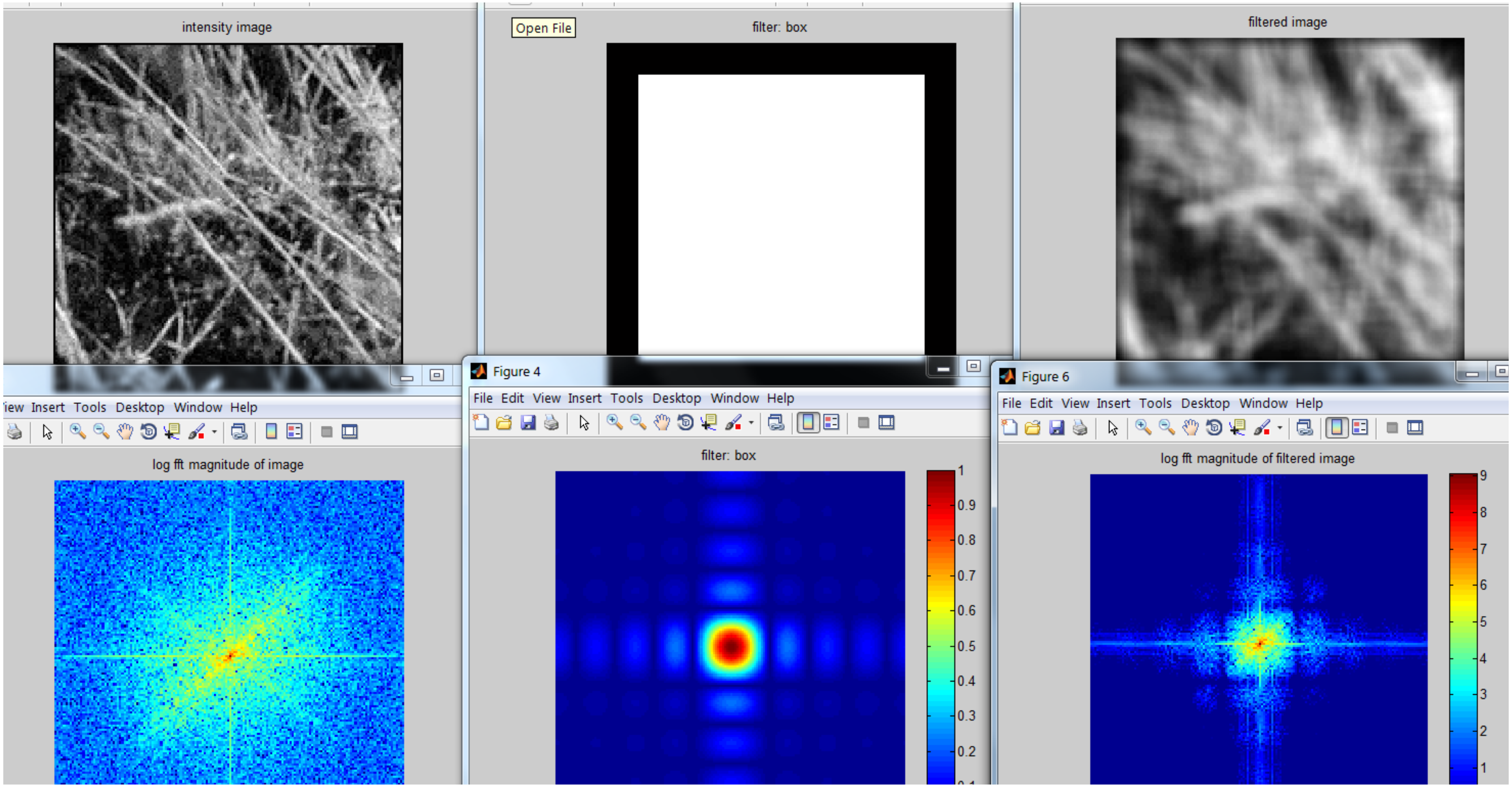
Fourier Transform pairs



Gaussian


intensity image
filter: gaussian
filtered image
log fft magnitude of image
filter: gaussian
log fft magnitude of filtered image
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
9
8
7
6
5
4
3
2
1

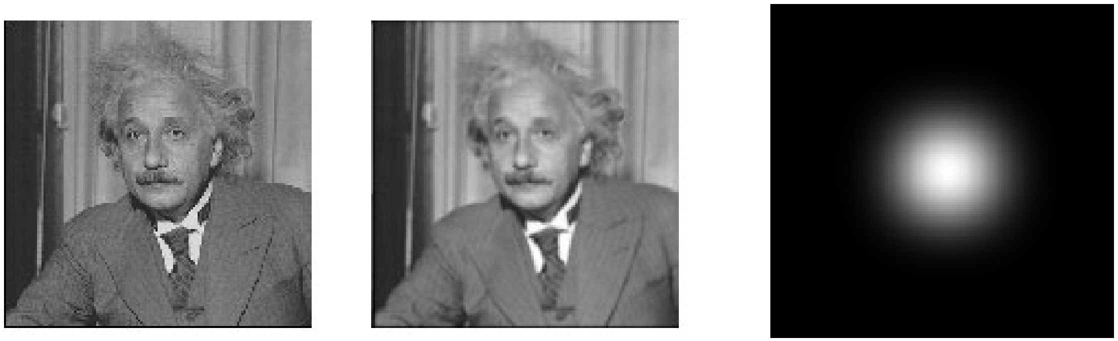
Box Filter



intensity image
Open File
filter: box
filtered image
Figure 4
Figure 6
File Edit View Insert Tools Desktop Window Help
log fft magnitude of image
filter: box
log fft magnitude of filtered image
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
9
8
7
6
5
4
3
2
1

Low-pass, Band-pass, High-pass filters

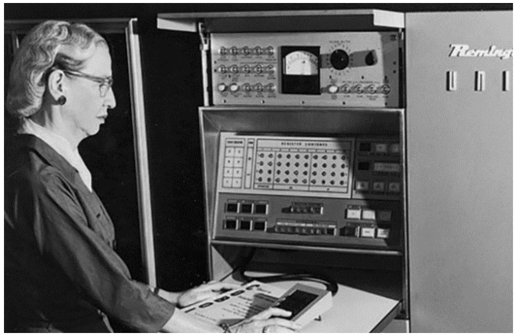
low-pass:



High-pass / band-pass:

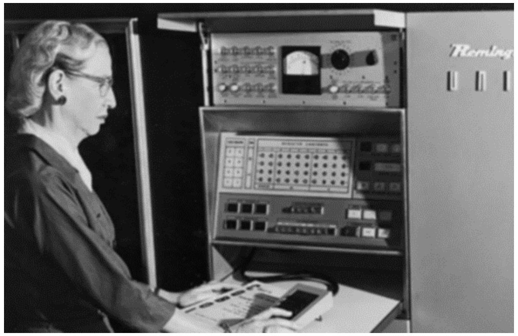
Low Pass vs. High Pass filtering

Image



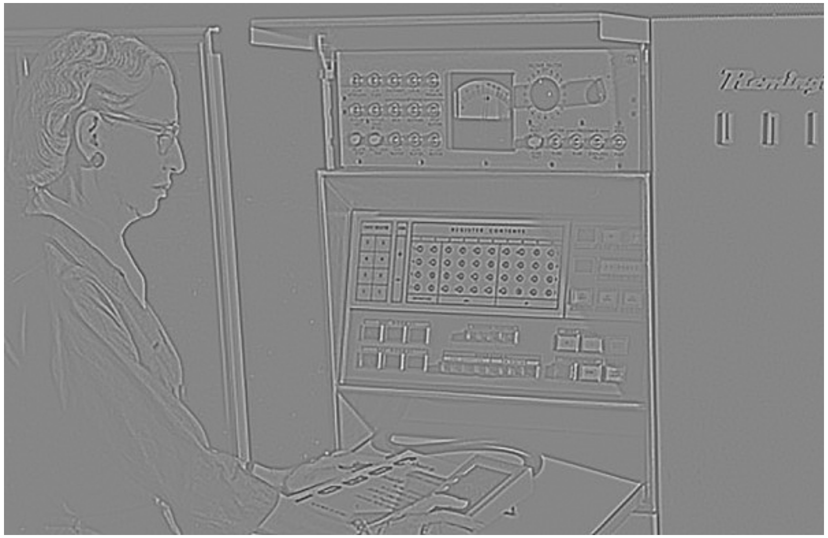
-

Smoothed



Details

=

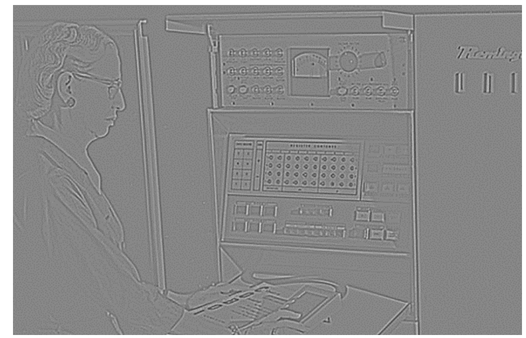


Filtering – Sharpening

Image

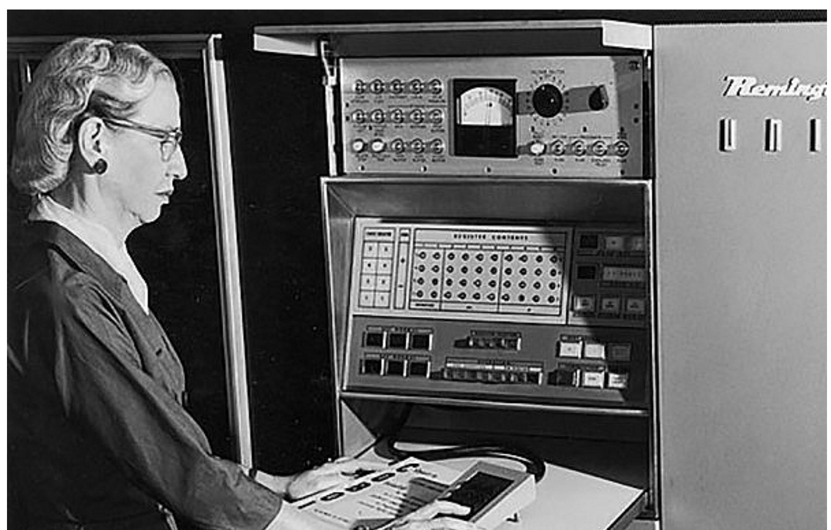
+α

Details



“Sharpened” α=2

=



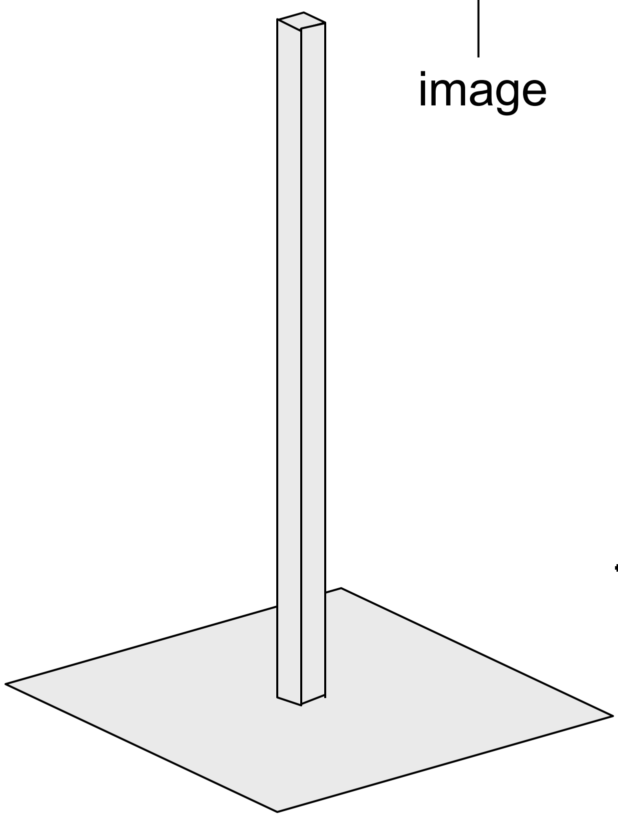
Unsharp mask filter

$$f + \alpha(f - f * g) = (1+\alpha)f - \alpha f * g = f * ((1+\alpha)e - \alpha g)$$

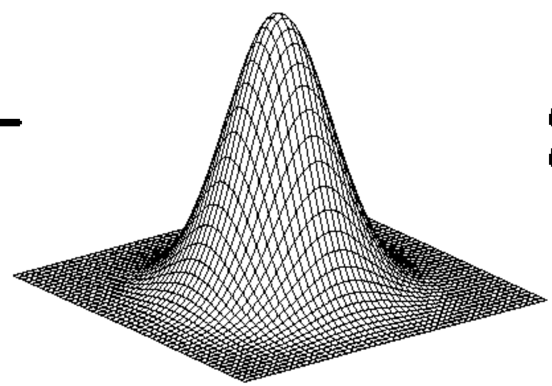
image

blurred image

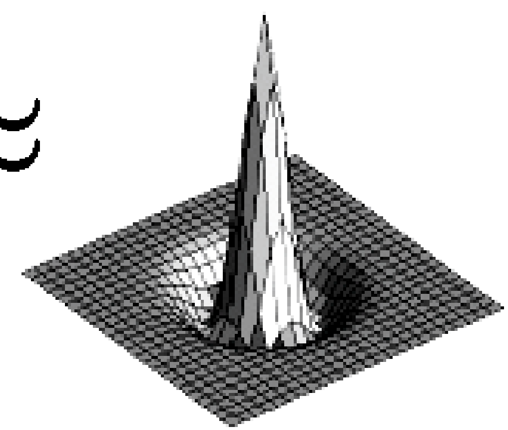
unit impulse (identity)



unit impulse

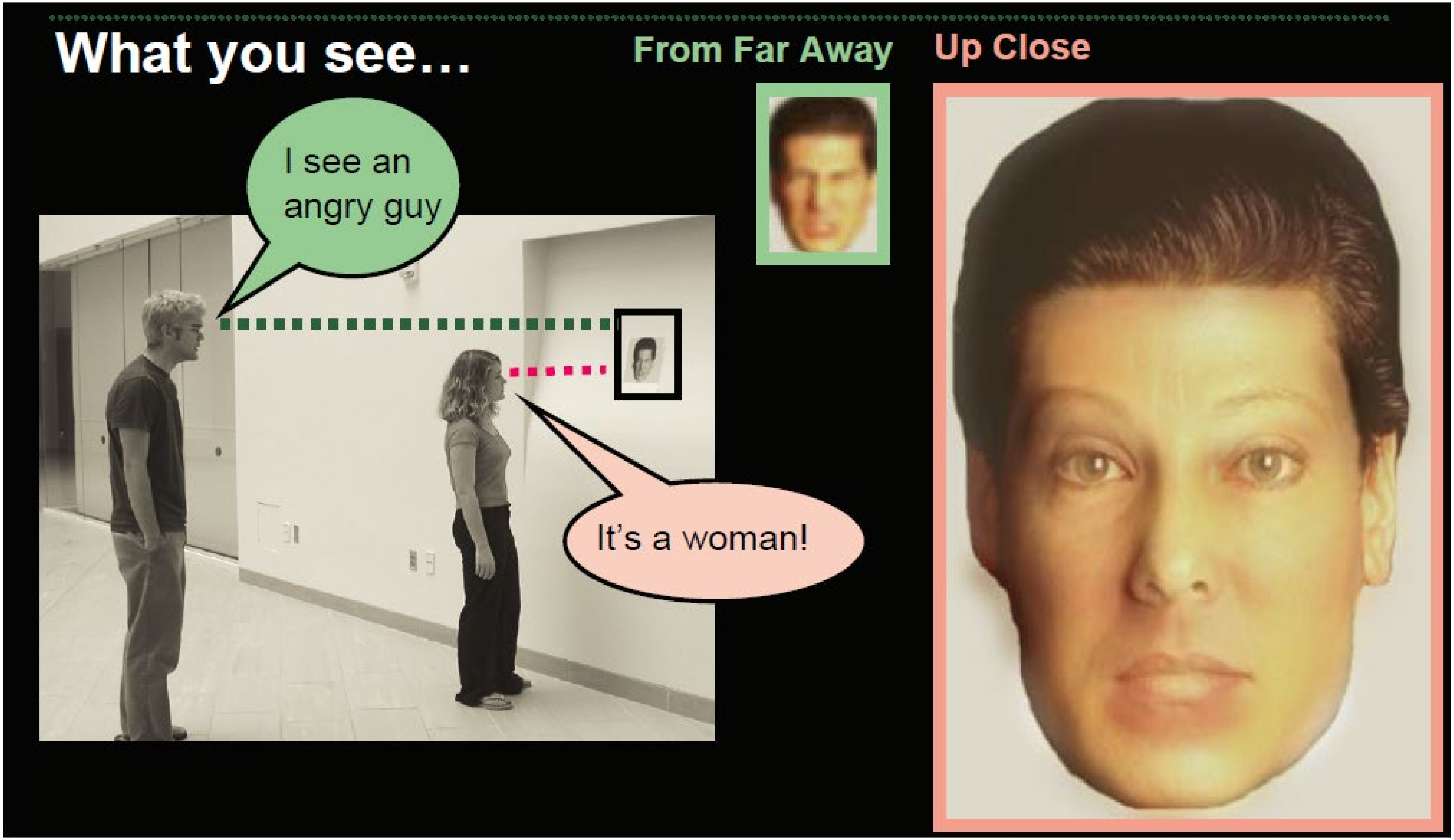


Gaussian



Laplacian of Gaussian

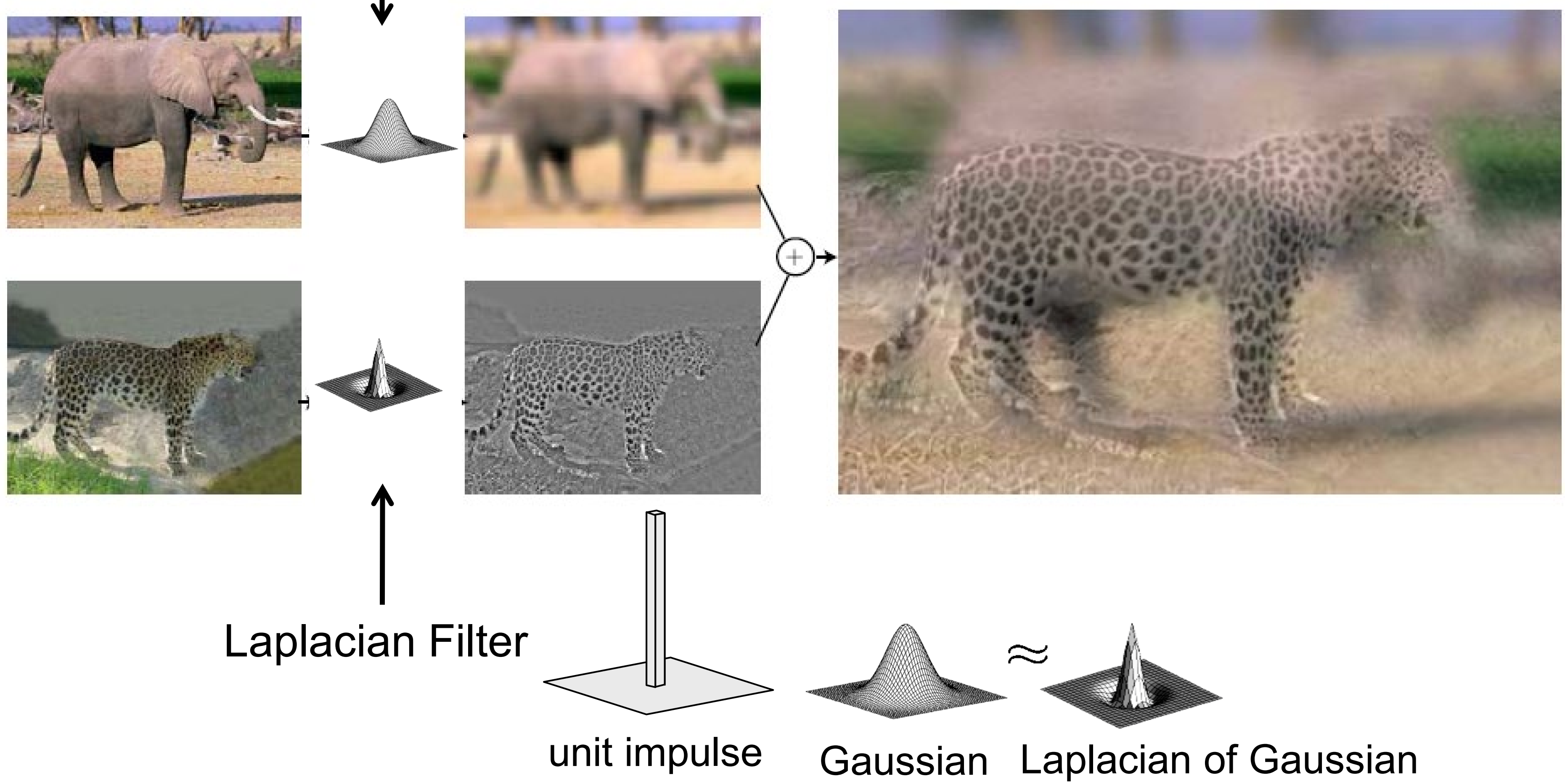
application: Hybrid Images



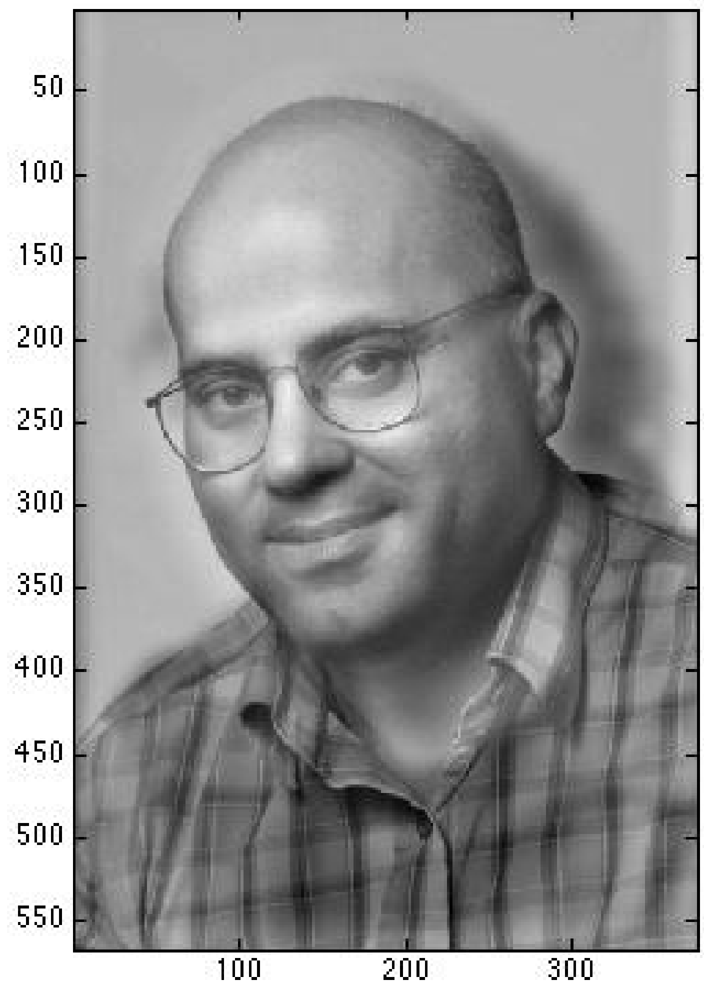
Aude Oliva & Antonio Torralba & Philippe G Schyns, SIGGRAPH 2006

Application: Hybrid Images

A. Oliva, A. Torralba, P.G. Schyns, "Hybrid Images," SIGGRAPH 2006



Yestaryear's homework



Prof. Jitendros Papadimalik

CS180:
Riyaz Faizullabhoy

5 min recap

Fourier Transform in 5 minutes: The Case of the Splotched Van Gogh, Part 3

https://www.youtube.com/watch?v=JciZYrh36LY