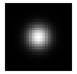
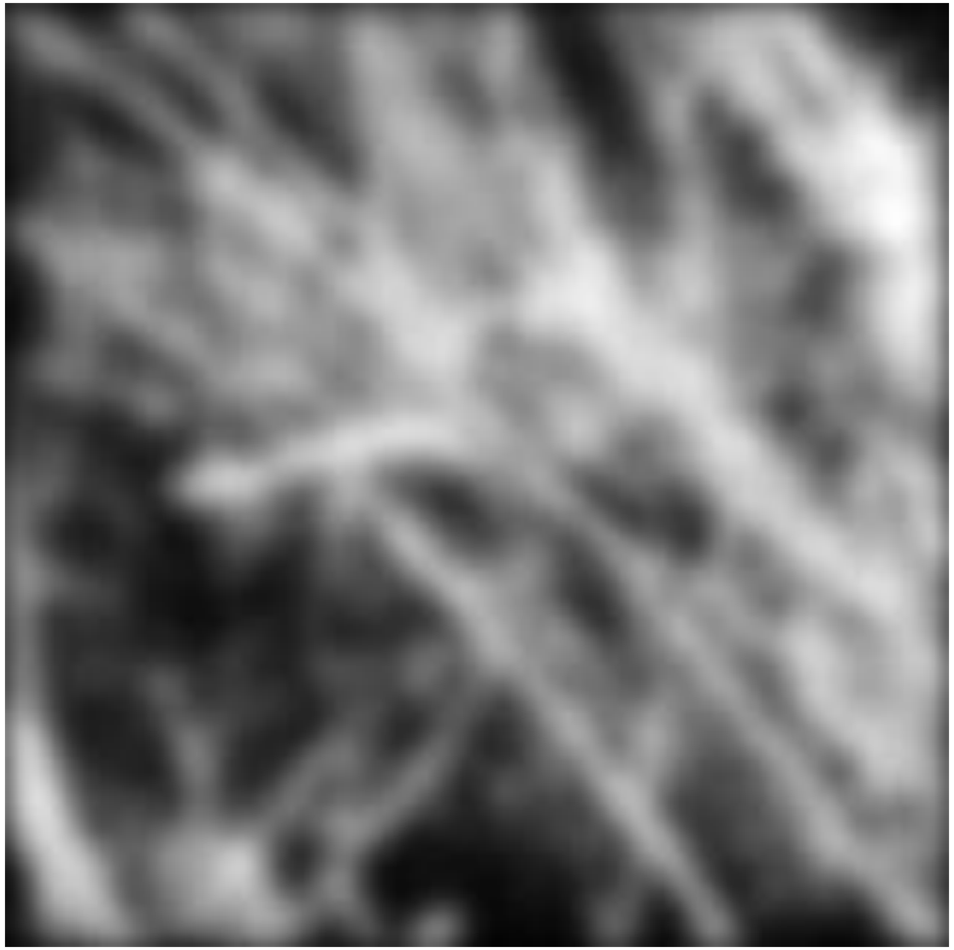
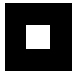
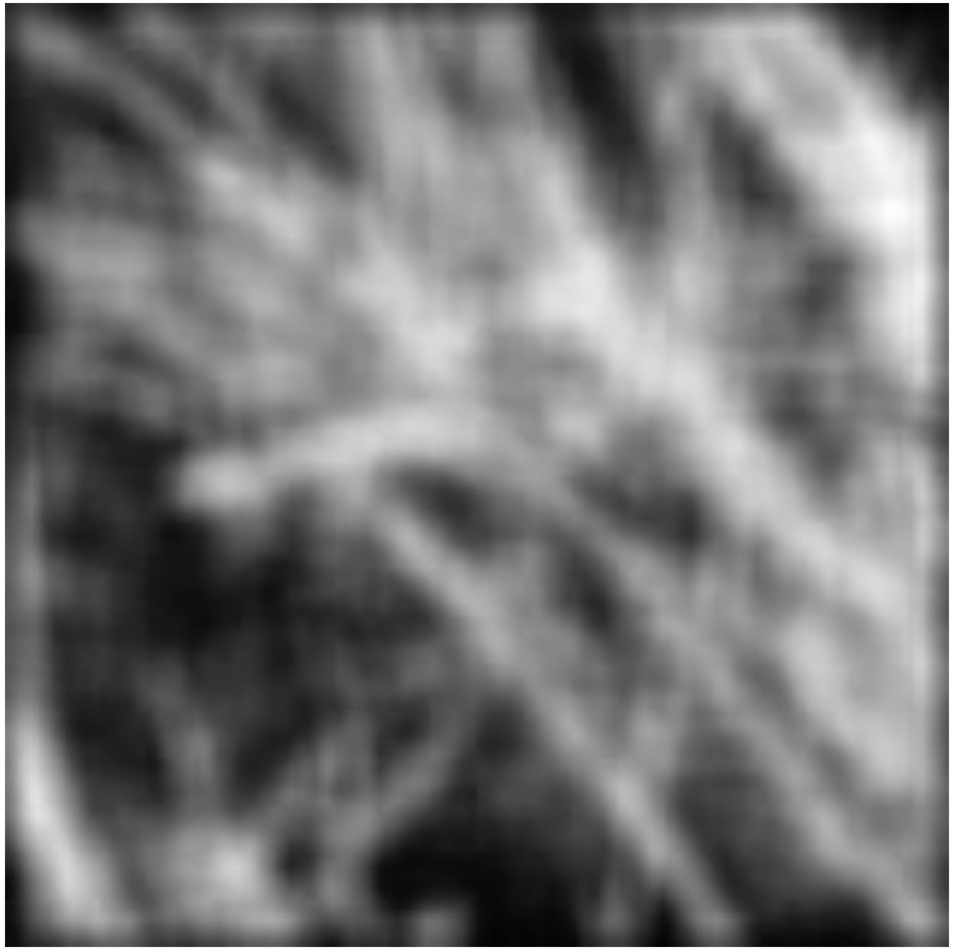
Fourier Transform and Frequency Domain

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

Gaussian 



Box filter 



Why does a lower resolution image still make sense to us? What do we lose?

Image: http://www.flickr.com/photos/igorms/136916757/

How is it that a 4MP image can be compressed to a few hundred KB without a noticeable change?

Image: http://www.flickr.com/photos/igorms/136916757/

Answer to these questions?

- Thinking images in terms of frequency

- Treat images as infinite-size, continuous periodic functions

- Fourier transform and frequency domain
  - Frequency view of filtering
  - Sampling

Jean Baptiste Joseph Fourier (1768-1830)

had crazy idea (1807):

__Any__ univariate function can be rewritten as a weighted sum of sines and cosines of different frequencies.

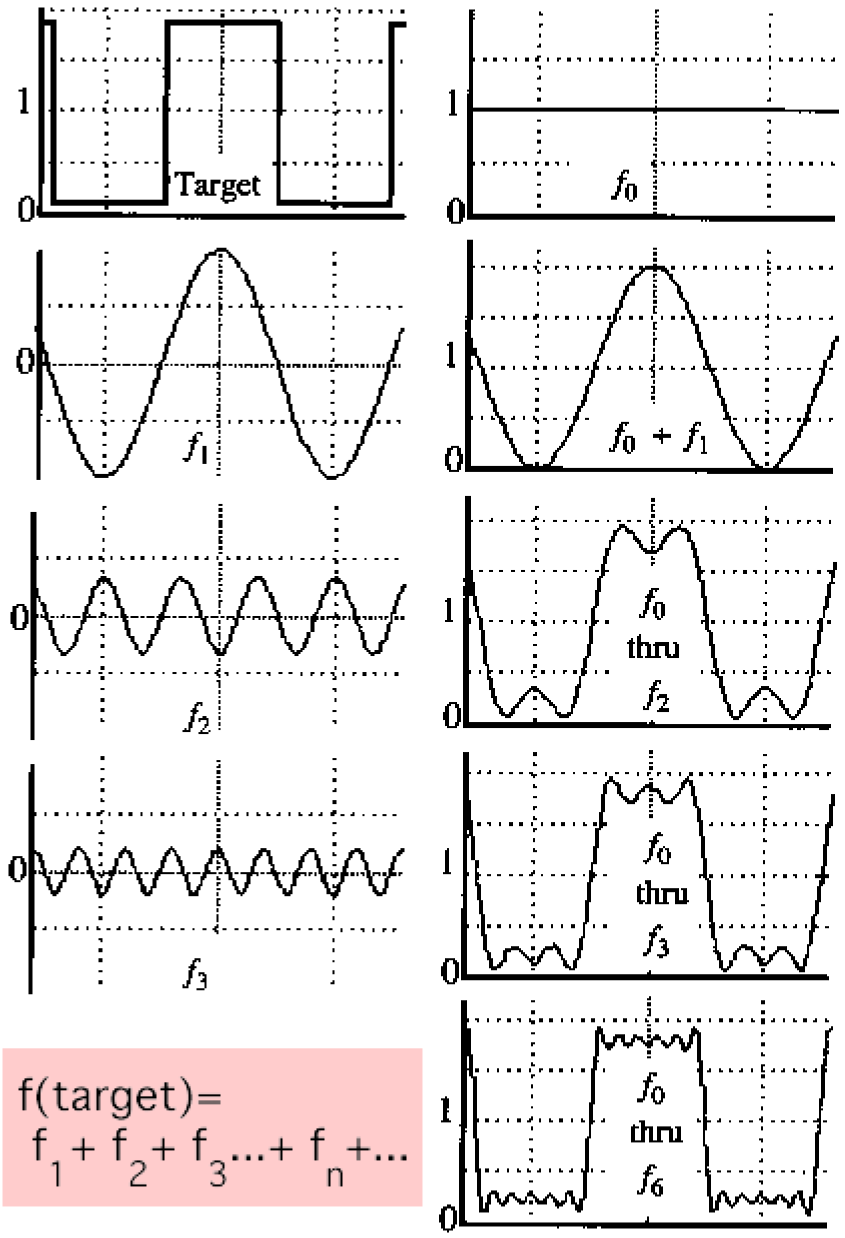
- Don't believe it?
  - Neither did Lagrange, Laplace, Poisson and other big wigs
  - Not translated into English until 1878!
- But it's (mostly) true!
  - called Fourier Series
  - there are some subtle restrictions

A sum of sines

- Our building block:

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

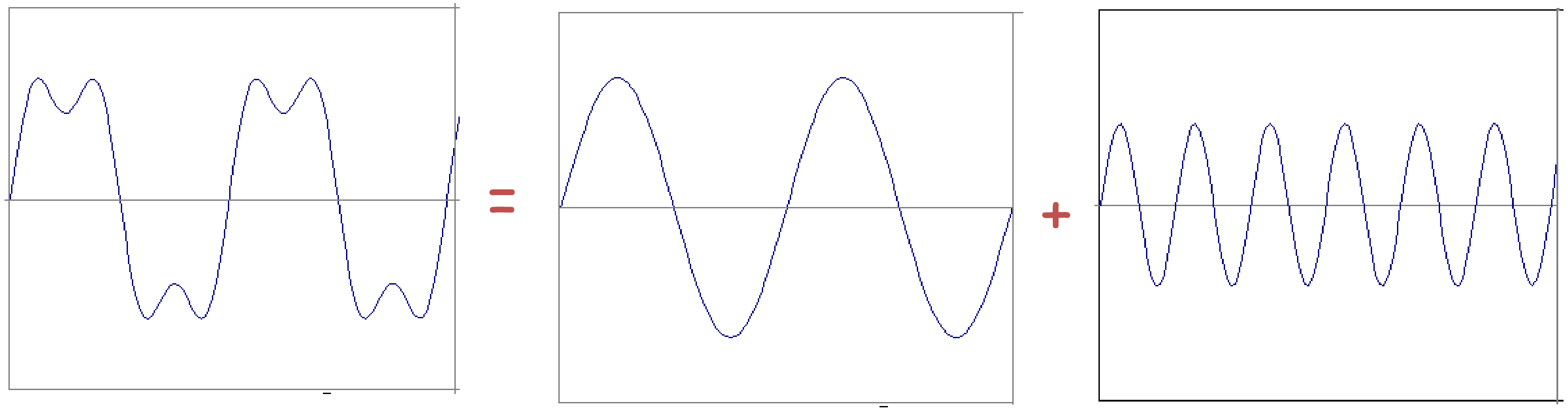
- Add enough of them to get any signal f(x) you want!

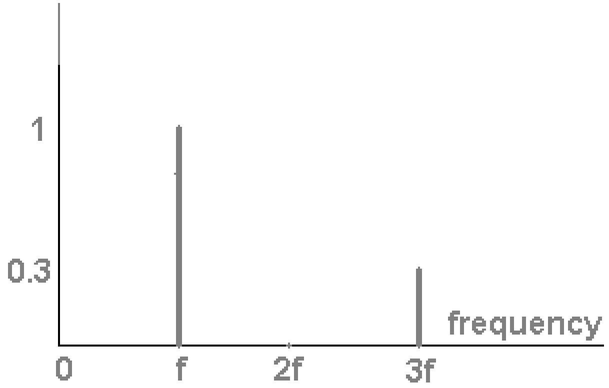


f(target)=
$f_1 + f_2 + f_3 ... + f_n + ...$

Frequency Spectra

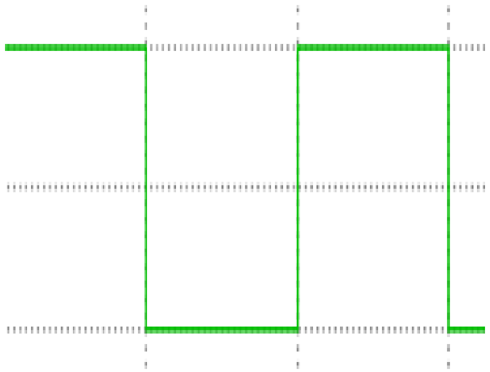
- example : $g(t) = \sin(2\pi f\, t) + (1/3)\sin(2\pi(3f)\, t)$



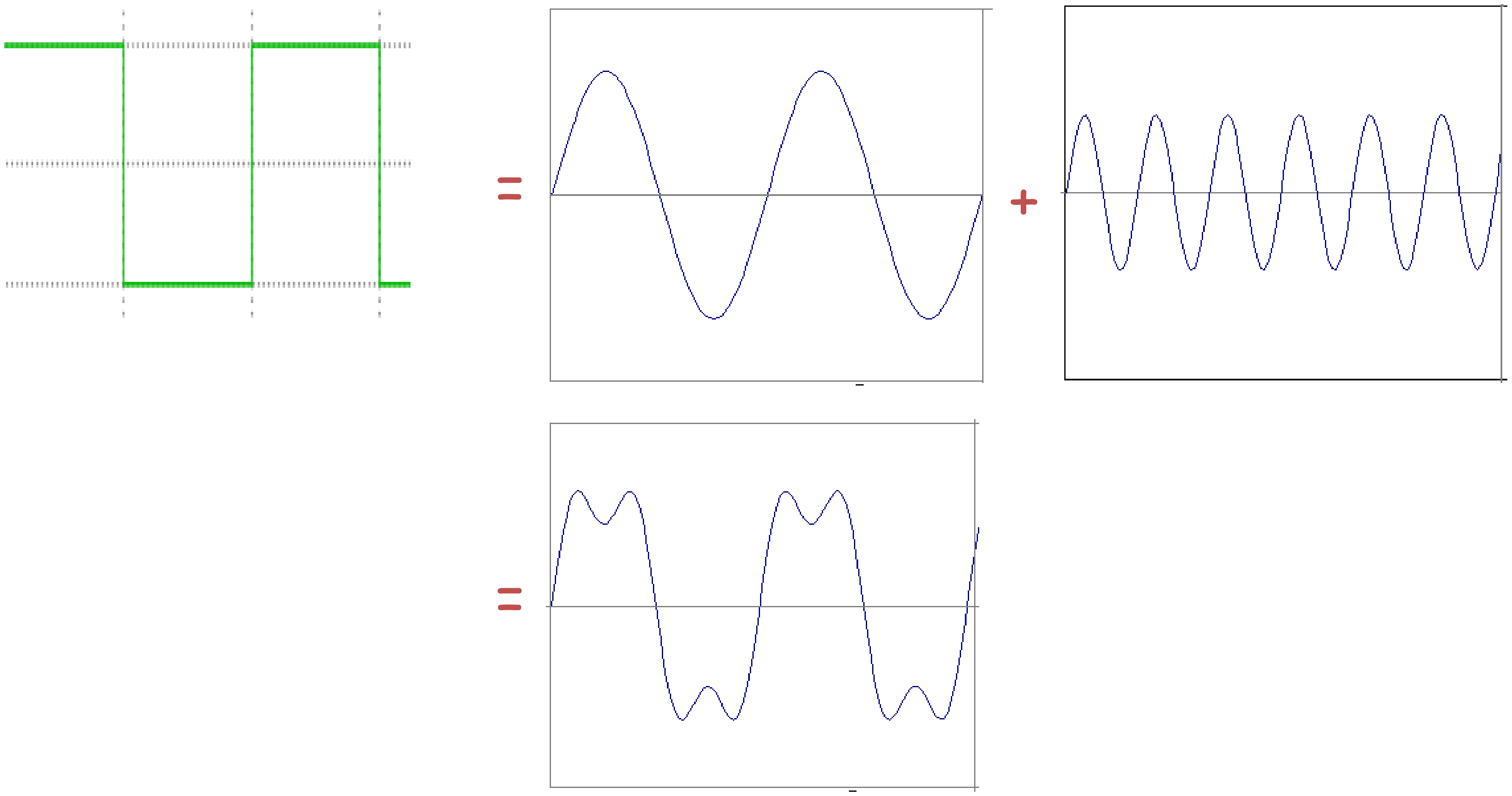


Slides: Efros

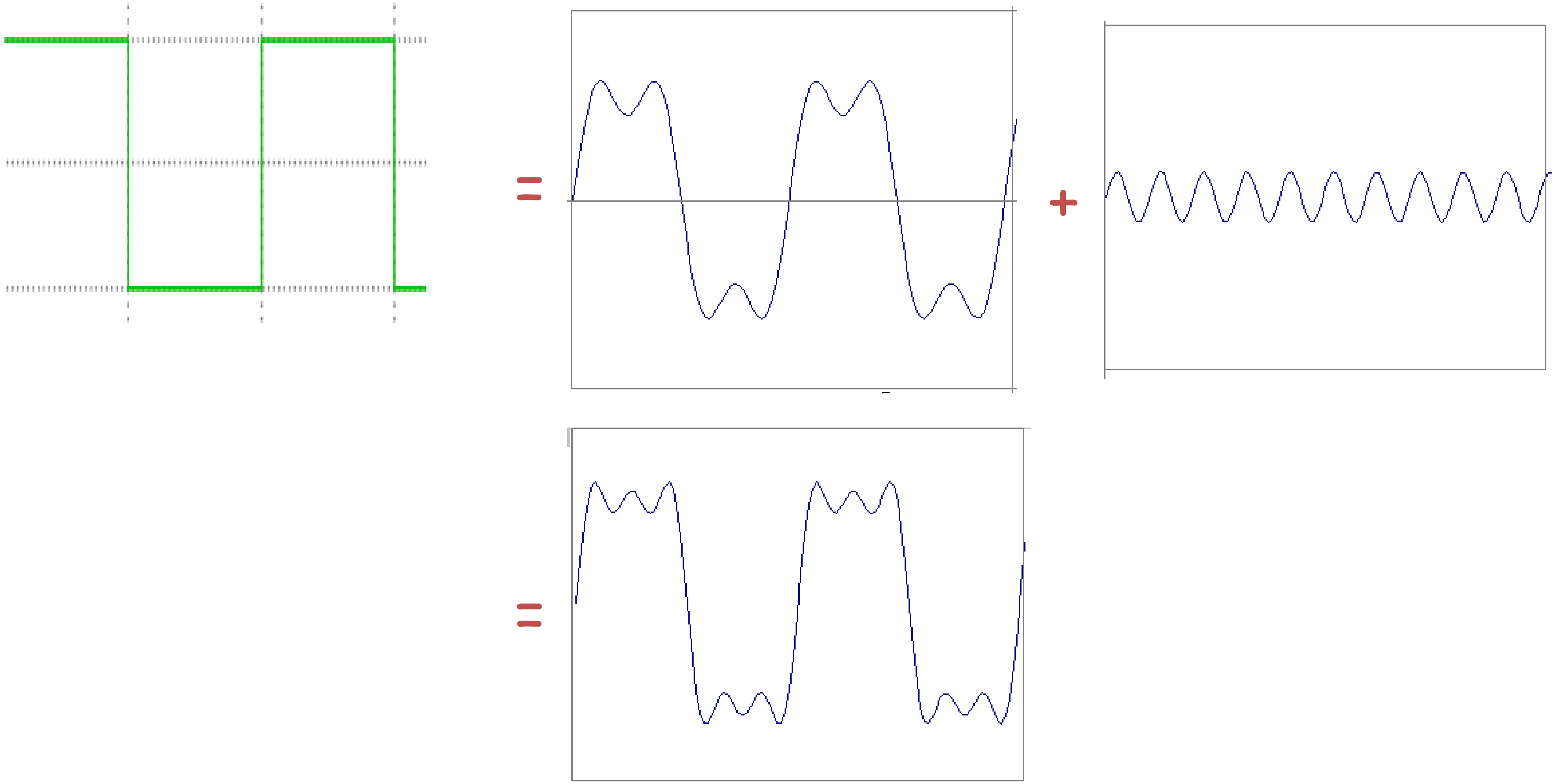
Frequency Spectra



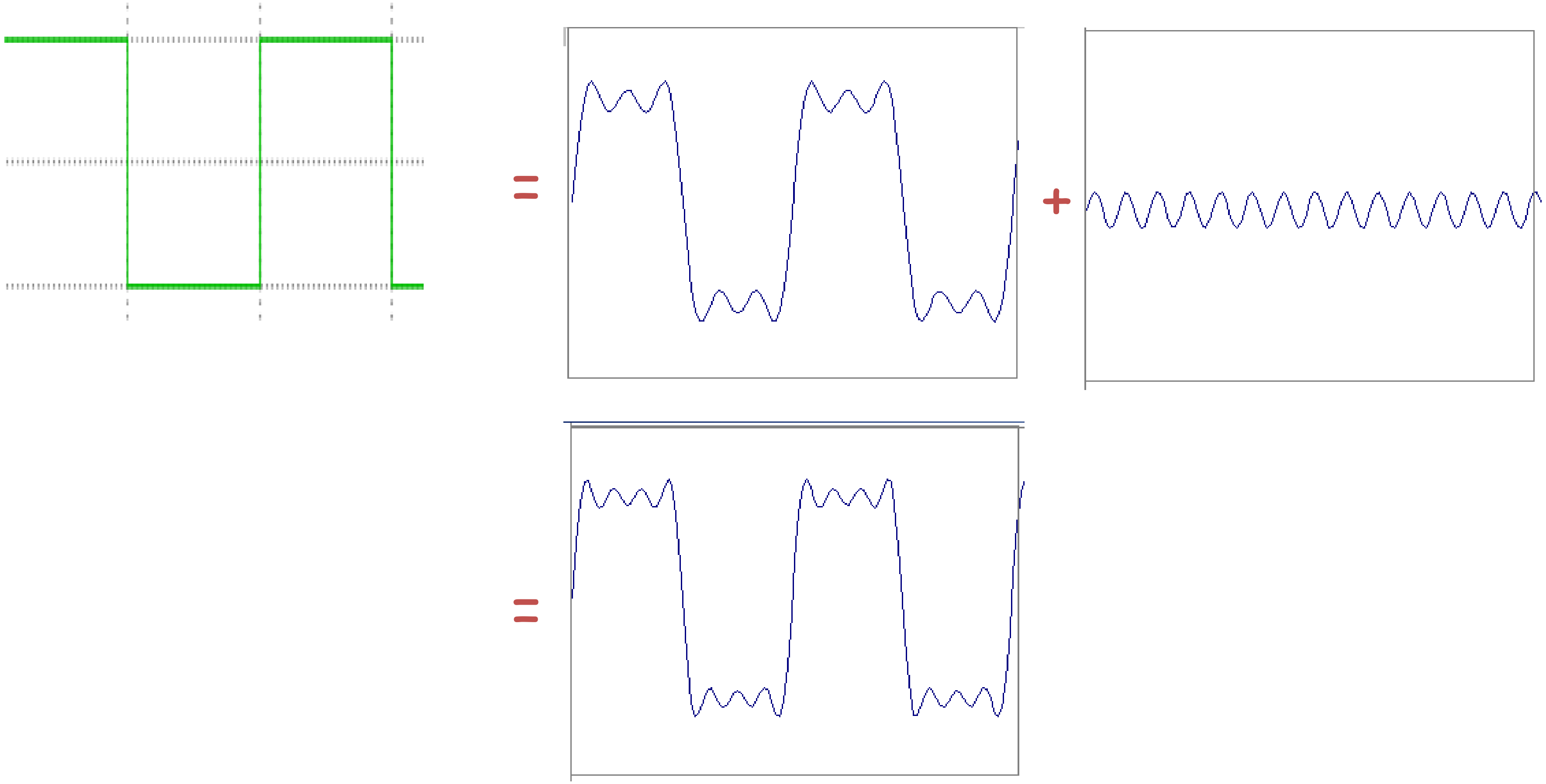
Frequency Spectra



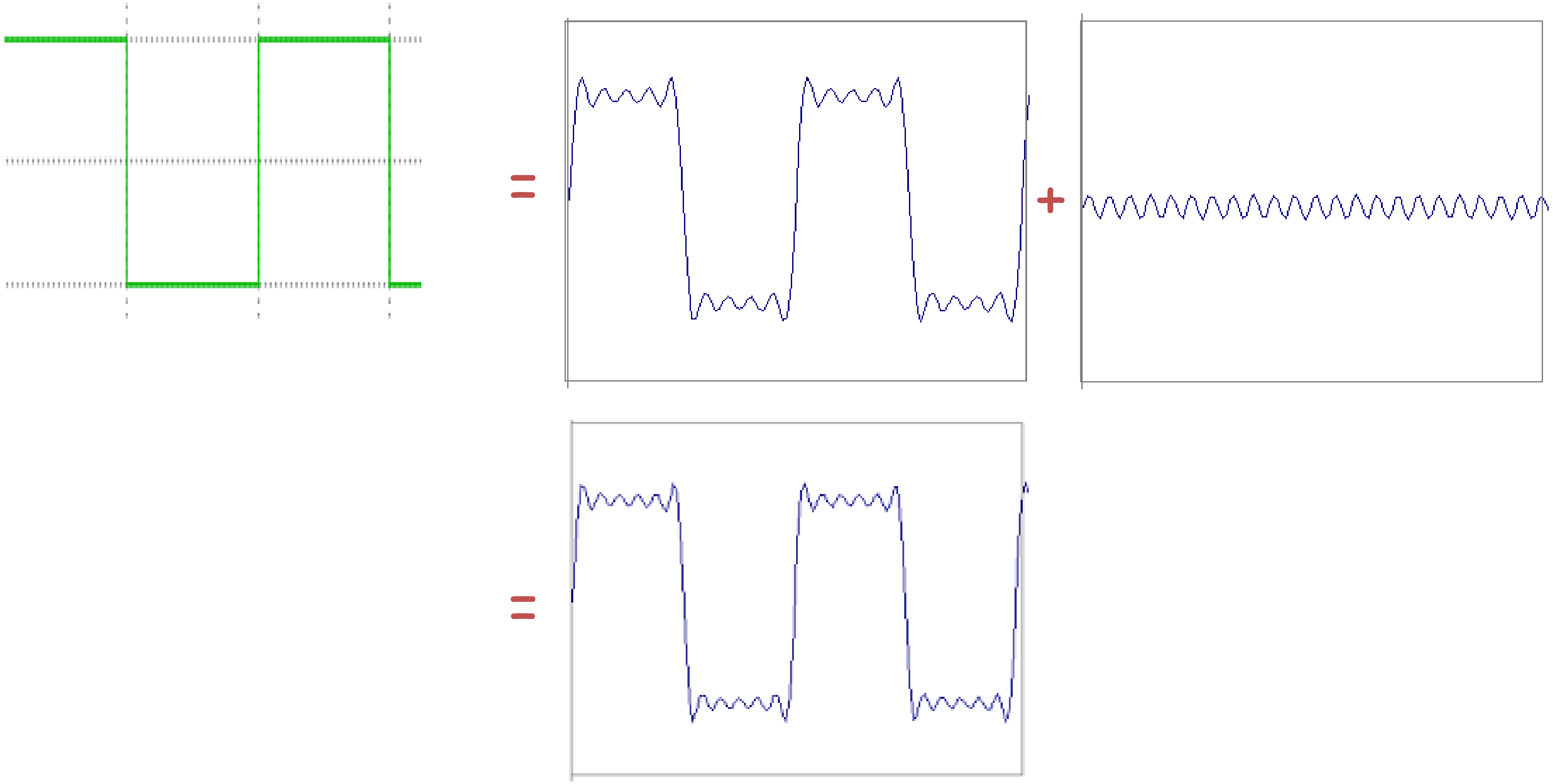
Frequency Spectra



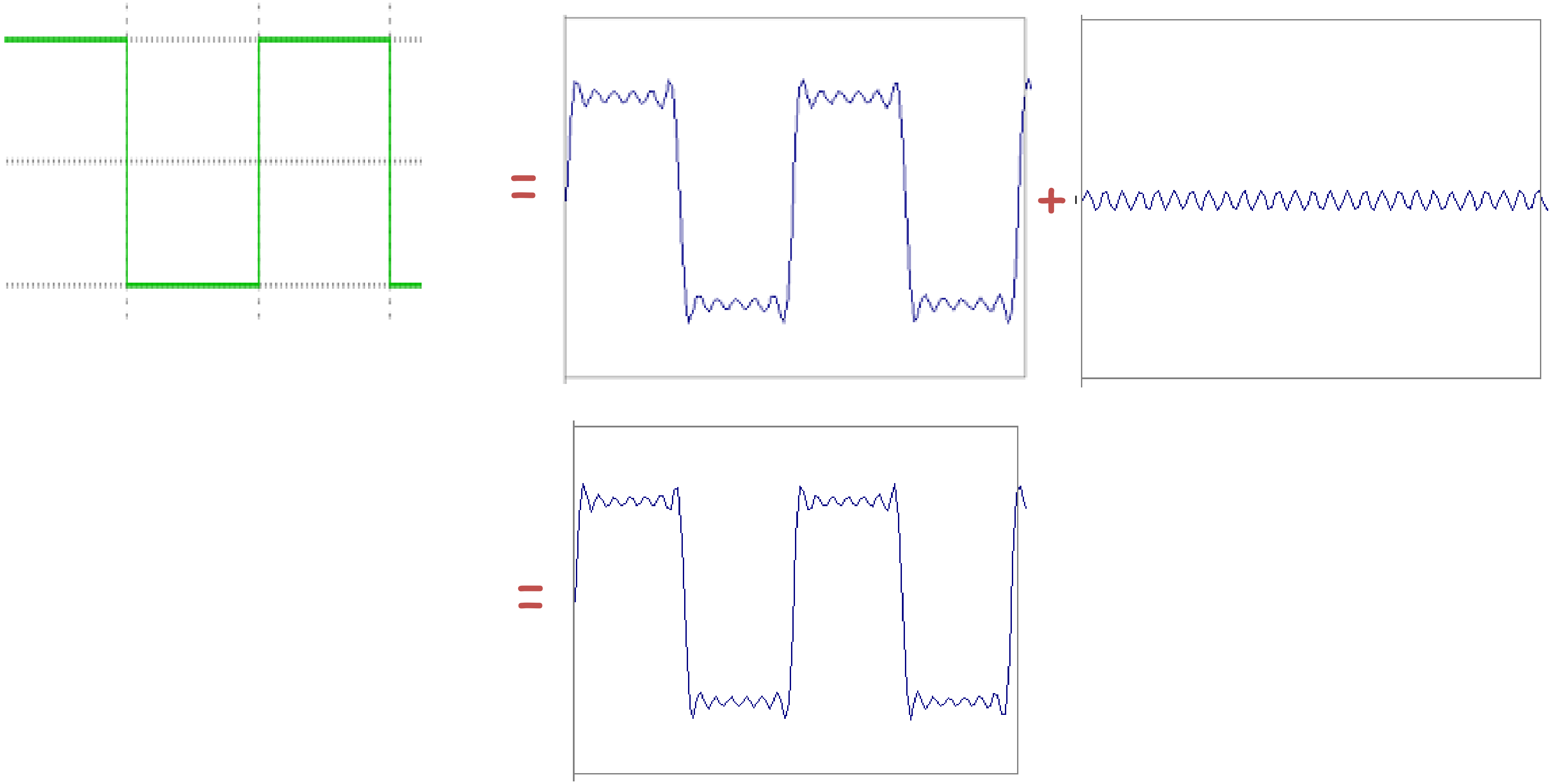
Frequency Spectra



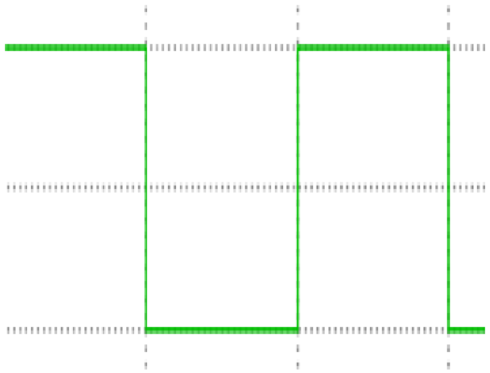
Frequency Spectra



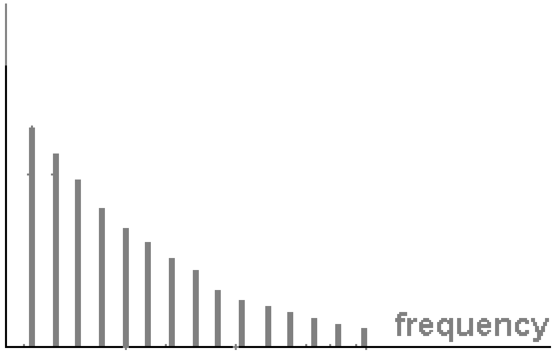
Frequency Spectra



Frequency Spectra

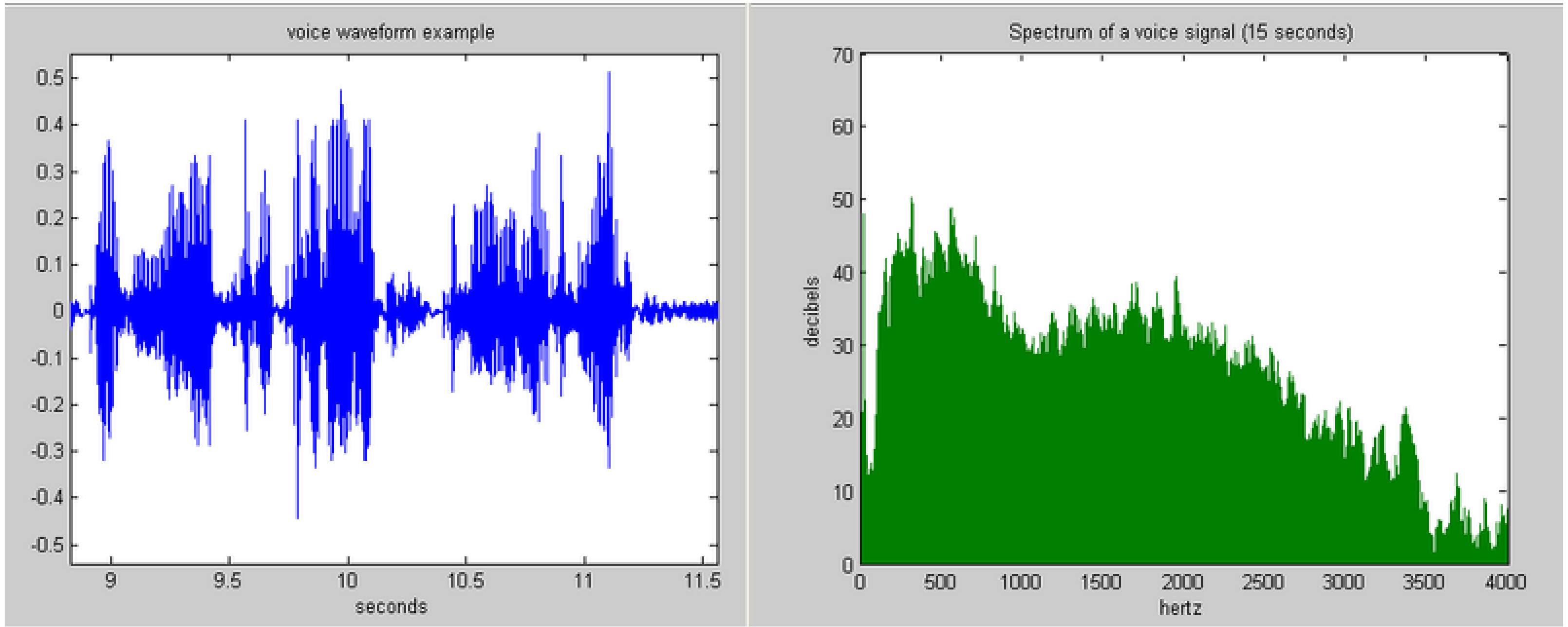


$$= \boxed{A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)}$$



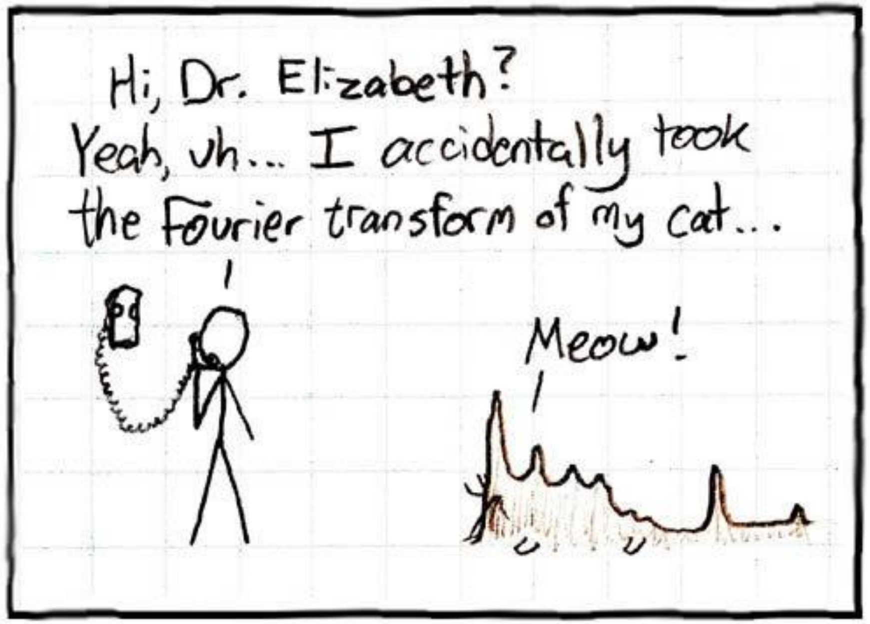
Example: Music

- We think of music in terms of frequencies at different magnitudes



Other signals

- We can also think of all kinds of other signals the same way



xkcd.com

Spectral Techniques

- Technique for representation and analysis of signals in frequency domain including audio, images, video
- How to decompose signal into summation of a series of sine and cosine functions (also called harmonic functions)
- Spectral techniques can improve efficiency of image processing
- Some image processing effects, concepts, techniques easier in frequency domain
- Includes **fourier transform**, **discrete fourier transform** and **discrete cosine transform**

Fourier Transform

- Fourier transform stores the magnitude and phase at each frequency
  - Magnitude encodes how much signal there is at a particular frequency
  - Phase encodes spatial information (indirectly)
  - For mathematical convenience, this is often notated in terms of real and complex numbers

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

Euler's formula: $e^{inx} = \cos(nx) + i\sin(nx)$

Computing the Fourier Transform

$$H(\omega) = \mathcal{F}\left\{h(x)\right\} = Ae^{j\phi}$$

Continuous

$$H(\omega) = \int_{-\infty}^{\infty} h(x)e^{-j\omega x}dx$$

Discrete

$$H(k) = \frac{1}{N}\sum_{x=0}^{N-1} h(x)e^{-j\frac{2\pi kx}{N}}$$

k = -N/2..N/2

Fast Fourier Transform (FFT): NlogN

Fourier analysis in images

- You can consider an image as a signal which is sampled in two directions
- Taking Fourier transform in both X and Y directions gives you the frequency representation of image
  - More intuitively, for the sinusoidal signal, if the amplitude varies so fast in short time, you can say it is a high frequency signal
  - If it varies slowly, it is a low frequency signal
- You can extend the same idea to images
  - Where does the amplitude varies drastically in images?
    - At the edge points, or noises.
    - So we can say, edges and noises are high frequency contents in an image
  - If there is no much changes in amplitude, it is a low frequency component

Discrete Fourier Transform (DFT) for image (1)

- The DFT is the sampled Fourier Transform and therefore does not contain all frequencies forming an image, but only a set of samples which is large enough to fully describe the spatial domain image
- The number of frequencies corresponds to the number of pixels in the spatial domain image, *i.e.* the image in the spatial and Fourier domain are of the same size

Discrete Fourier Transform (DFT) for image (2)

- For a square image of size N×N, the two-dimensional DFT is given by:

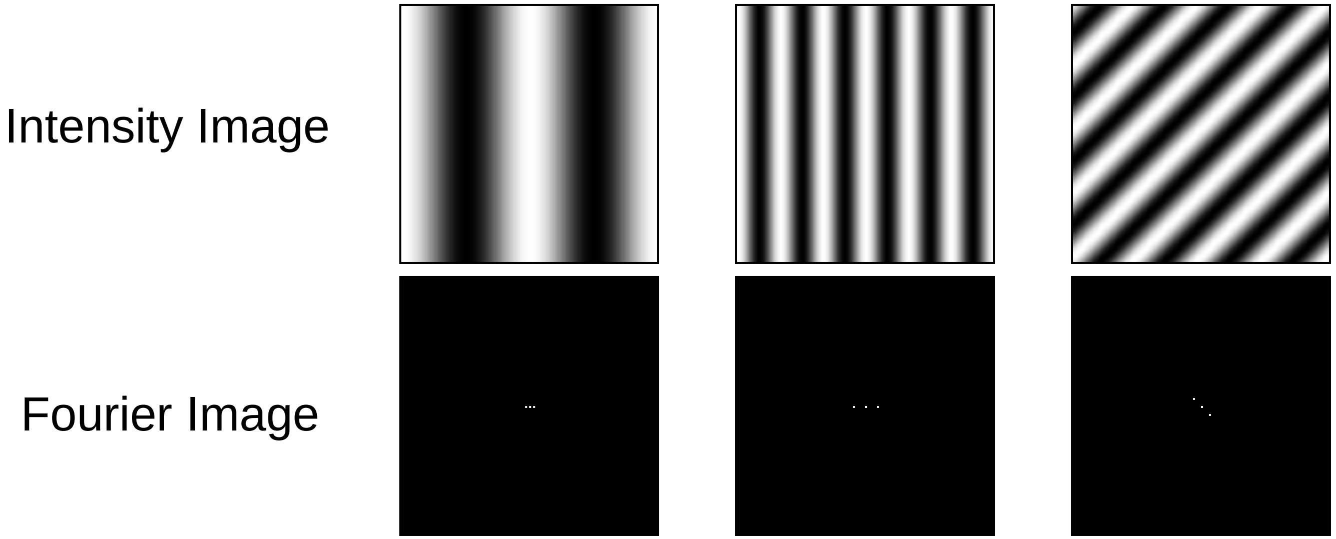
$$F(k,l) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i,j)\, e^{-\iota 2\pi \left(\frac{ki}{N} + \frac{lj}{N}\right)}$$

- where f(a,b) is the image in the spatial domain and the exponential term is the basis function corresponding to each point F(k,l) in the Fourier space
- The equation can be interpreted as: the value of each point F(k,l) is obtained by multiplying the spatial image with the corresponding base function and summing the result
- The basis functions are sine and cosine waves with increasing frequencies, i.e. F(0,0) represents the DC-component of the image which corresponds to the average brightness and F(N-1,N-1) represents the highest frequency

Discrete Fourier Transform (DFT) for image (3)

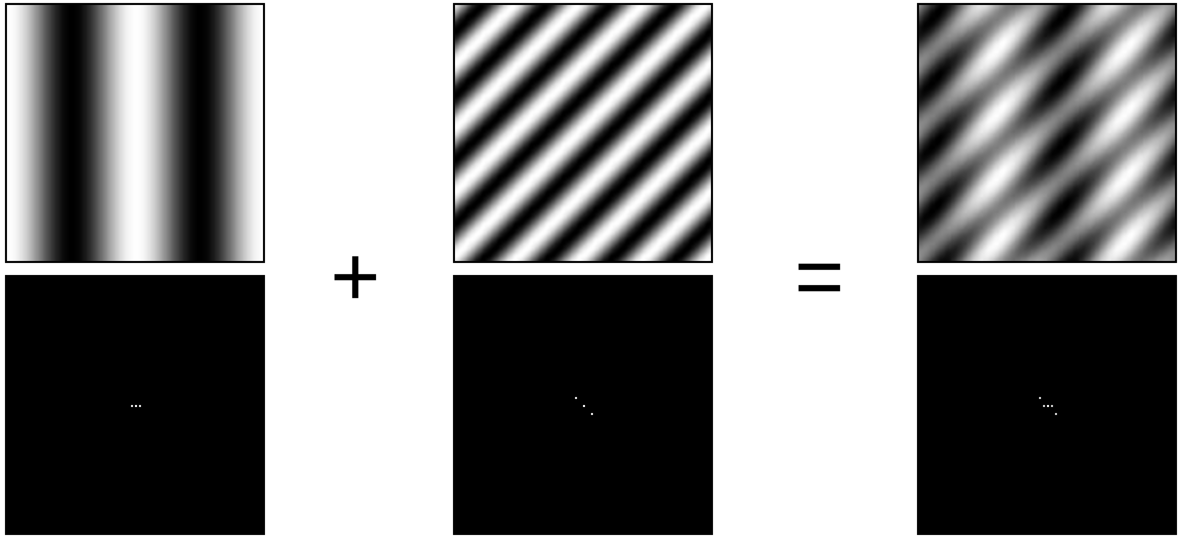
- The Fourier Transform is used if we want to access the geometric characteristics of a spatial domain image
  - Because the image in the Fourier domain is decomposed into its sinusoidal components, it is easy to examine or process certain frequencies of the image, thus influencing the geometric structure in the spatial domain
- In most implementations the Fourier image is shifted in such a way that the DC-value (*i.e.* the image mean) *F(0,0)* is displayed in the center of the image. The further away from the center an image point is, the higher is its corresponding frequency
- The DC-value is by far the largest component of the image. However, the dynamic range of the Fourier coefficients (i.e. the intensity values in the Fourier image) is too large to be displayed on the screen.
- We apply a logarithmic transformation to the image we obtain

Fourier analysis in images

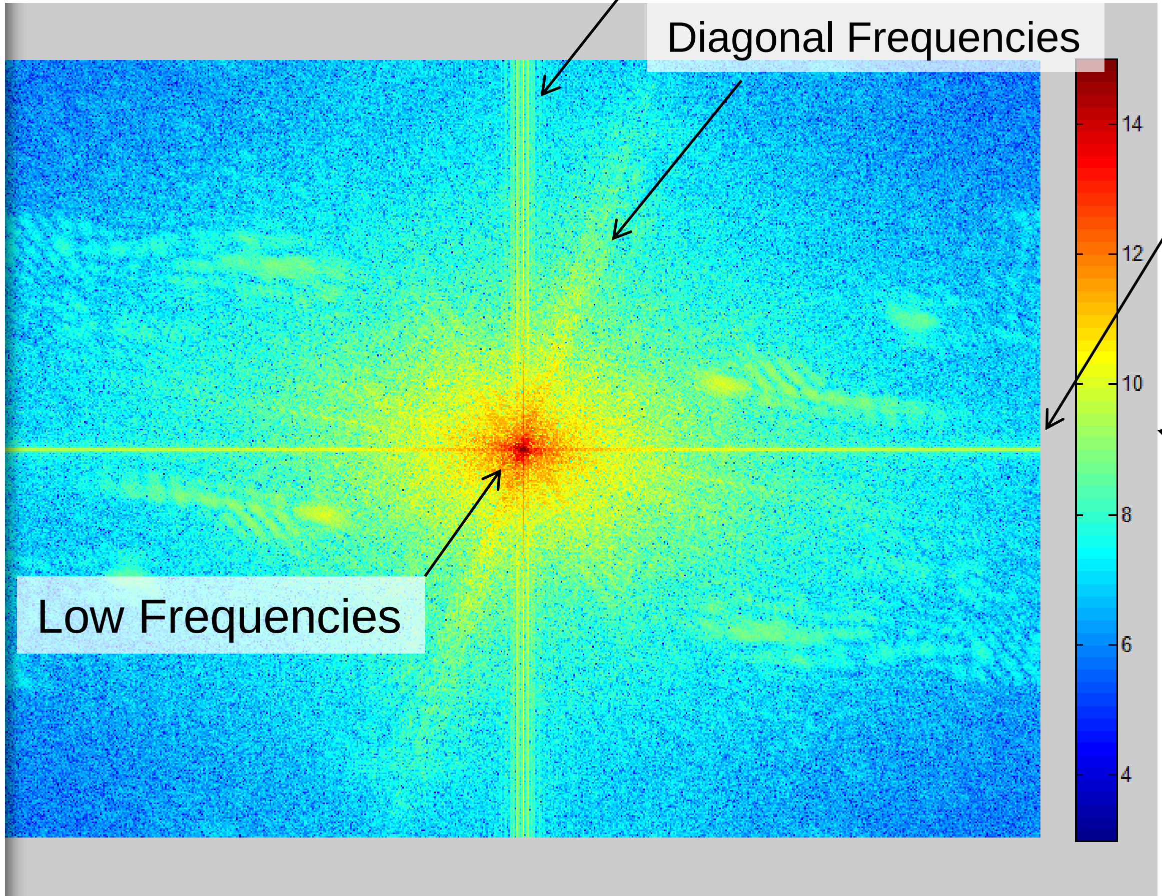


http://sharp.bu.edu/~slehar/fourier/fourier.html#filtering

Signals can be composed



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



Strong Vertical Frequency
(Sharp Horizontal Edge)
Diagonal Frequencies
Strong Horz. Frequency
(Sharp Vert. Edge)
Log Magnitude
Low Frequencies
14
12
10
8
6
4

The Convolution Theorem

- 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!

Properties of Fourier Transforms

- Linearity $\mathcal{F}[ax(t) + by(t)] = a\mathcal{F}[x(t)] + b\mathcal{F}[y(t)]$

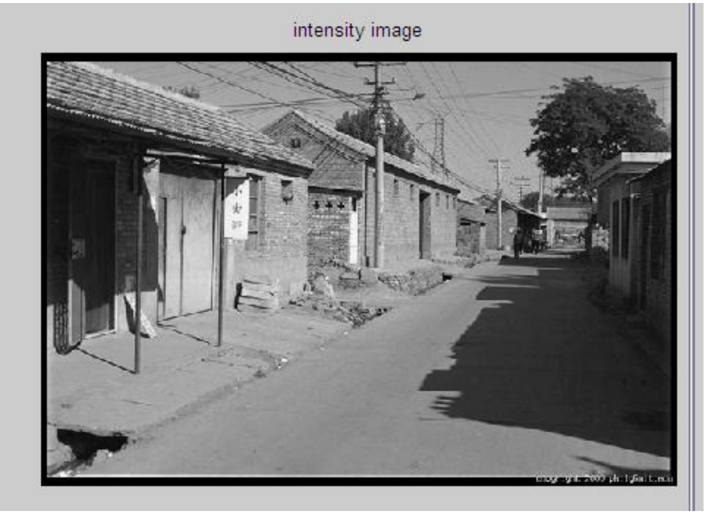
- Fourier transform of a real signal is symmetric about the origin

- The energy of the signal is the same as the energy of its Fourier transform

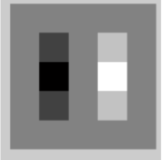
See Szeliski Book (3.4)

Filtering in spatial domain

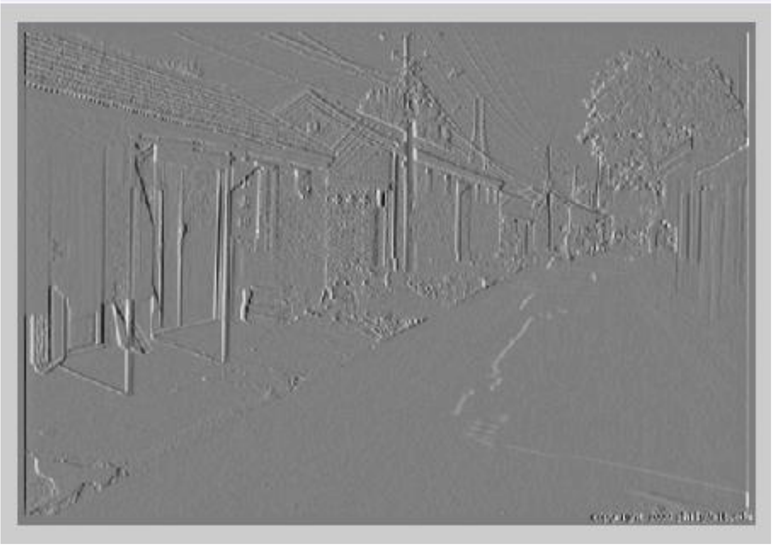
1	0	-1
2	0	-2
1	0	-1



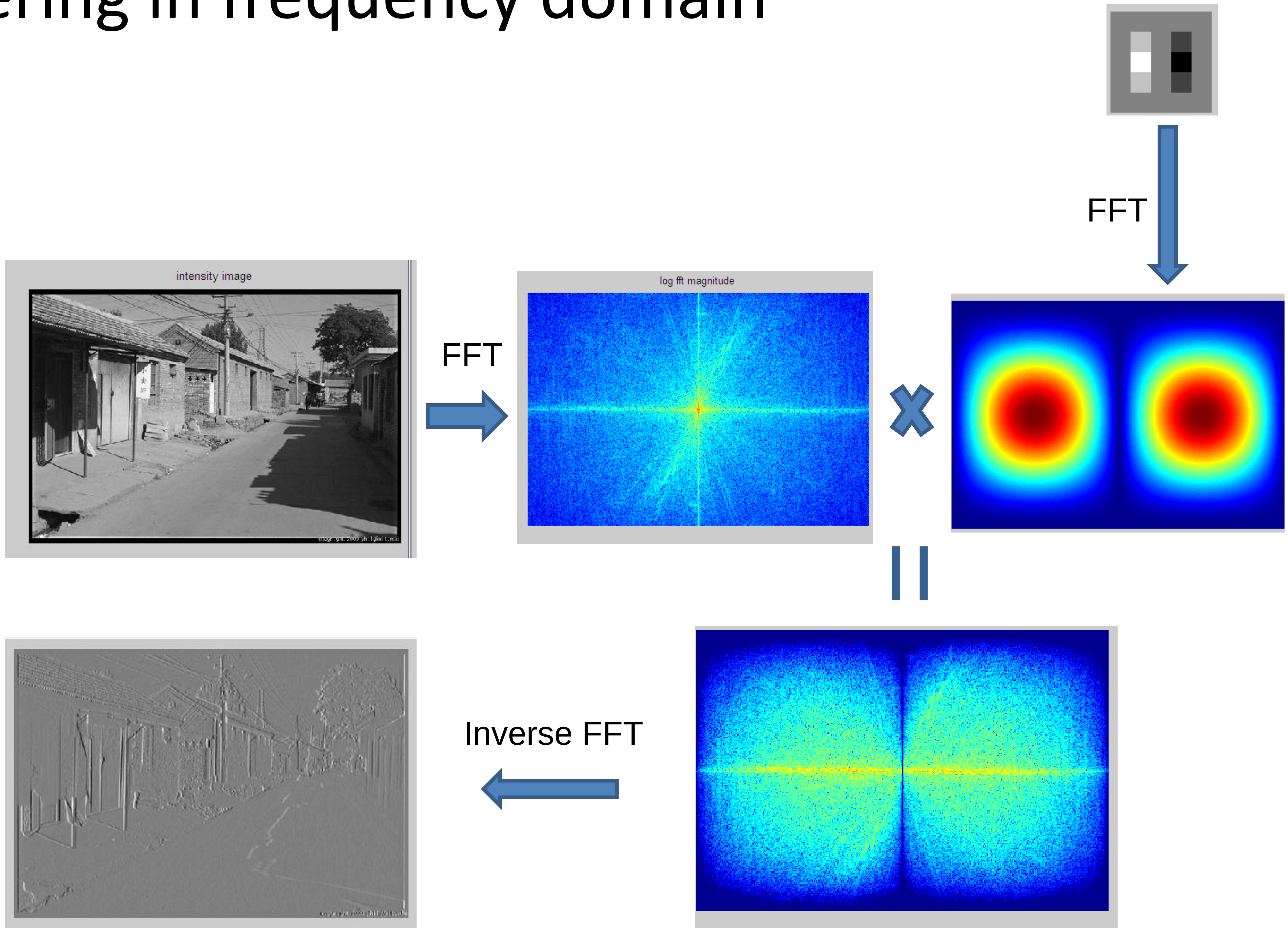
*



=



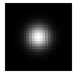
Filtering in frequency domain

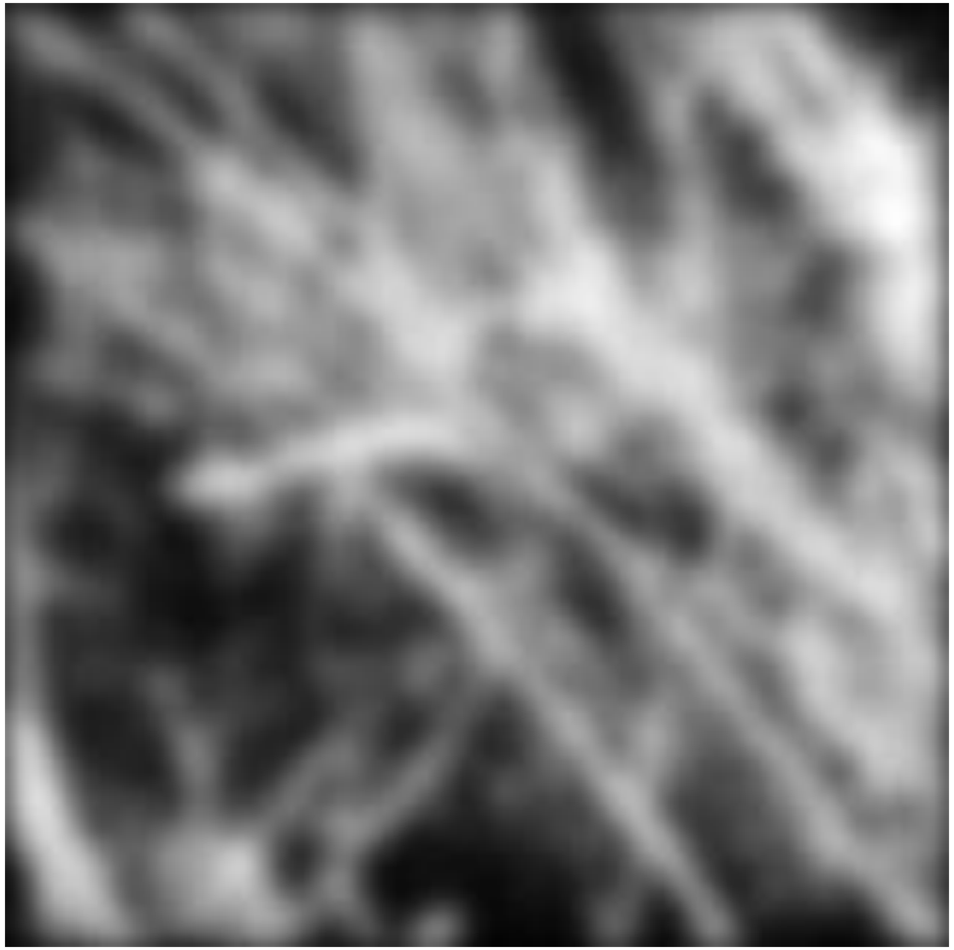


Filtering

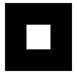
Why does the Gaussian give a nice smooth image, but the square filter give edgy artifacts?

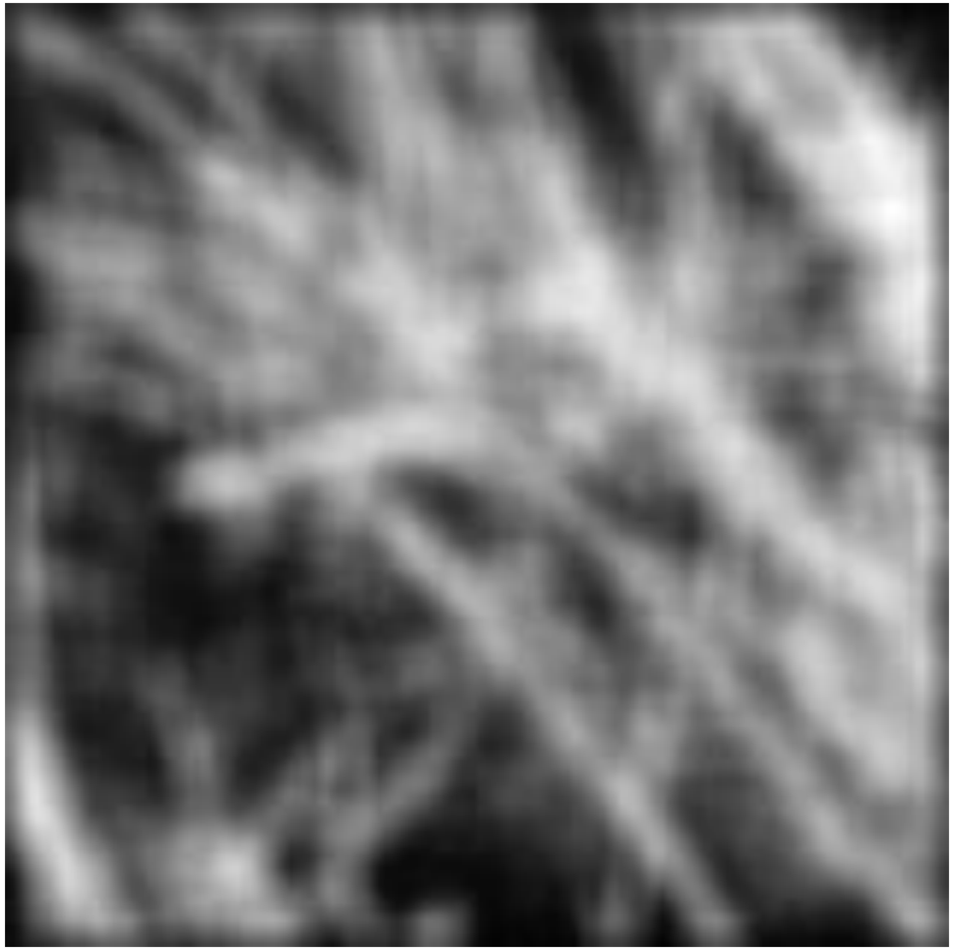
Gaussian



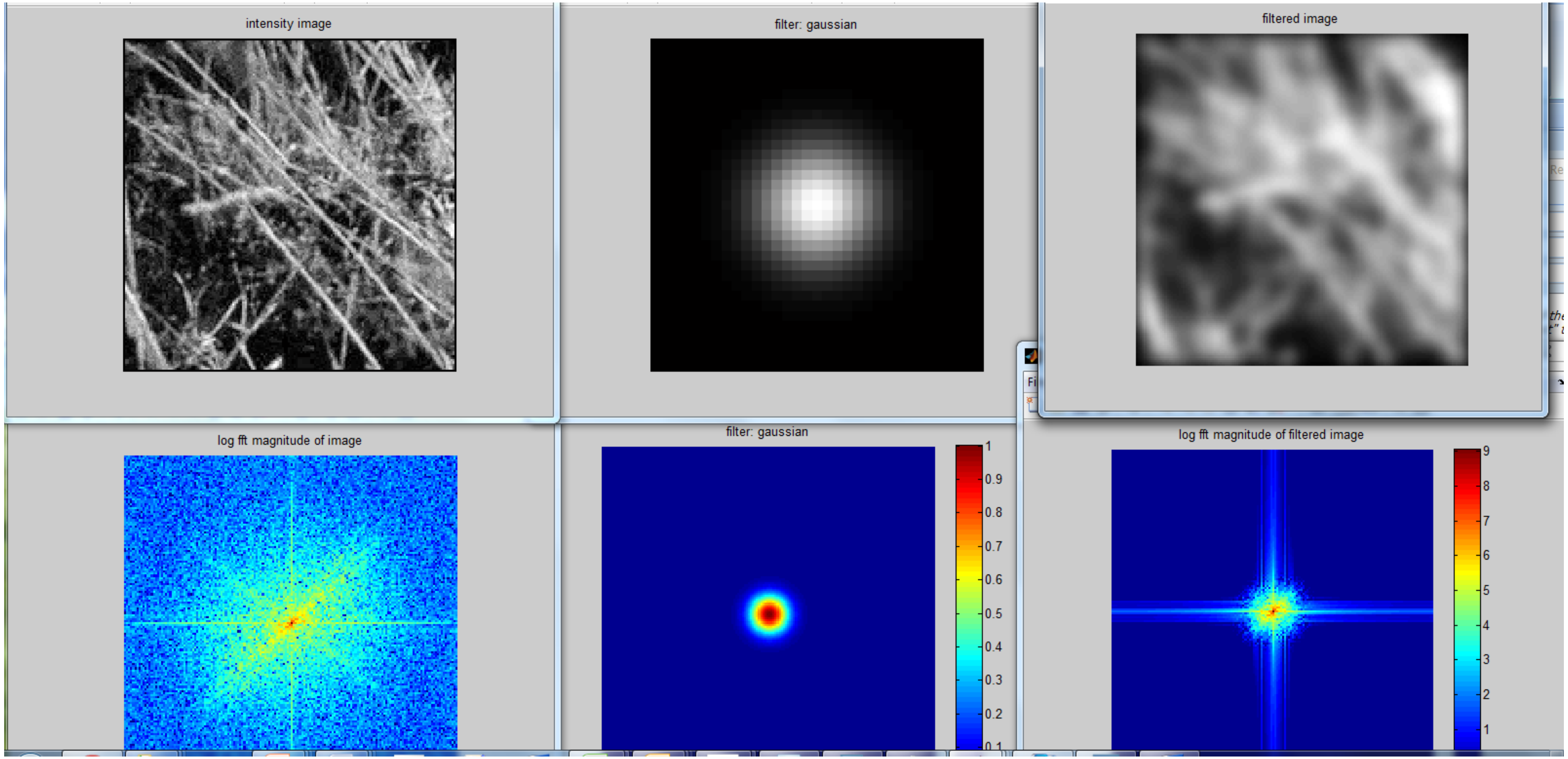


Box filter



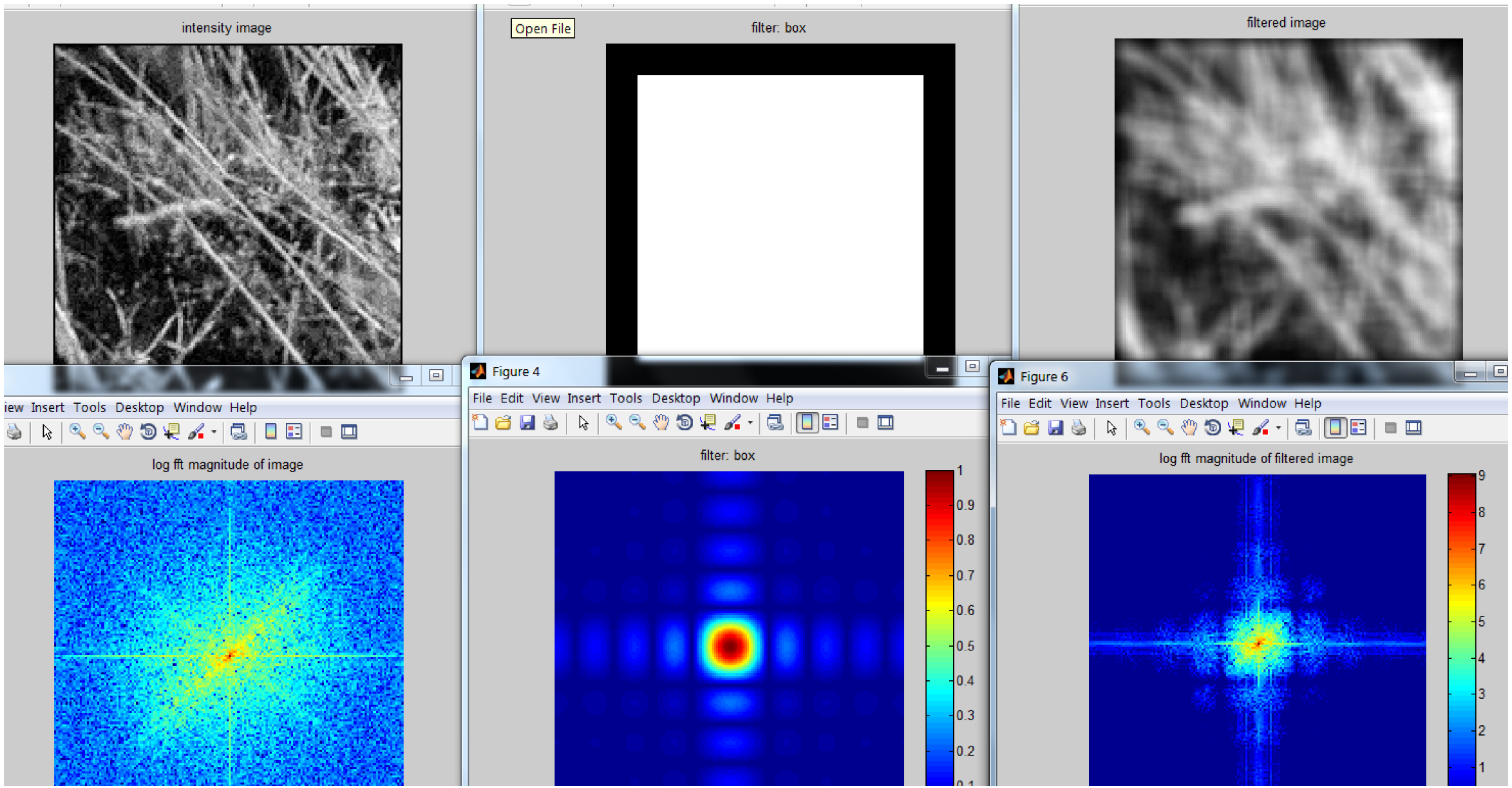


Gaussian



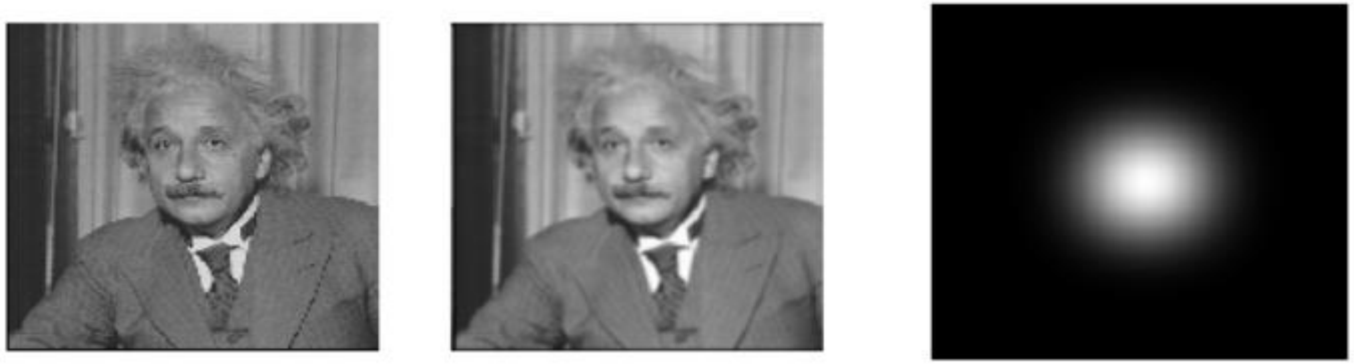
intensity image
filter: gaussian
filtered image
log fft magnitude of image
filter: gaussian
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
log fft magnitude of filtered image
9
8
7
6
5
4
3
2
1

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

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

low-pass:



High-pass / band-pass:

Edges in images

Input Image

Magnitude Spectrum

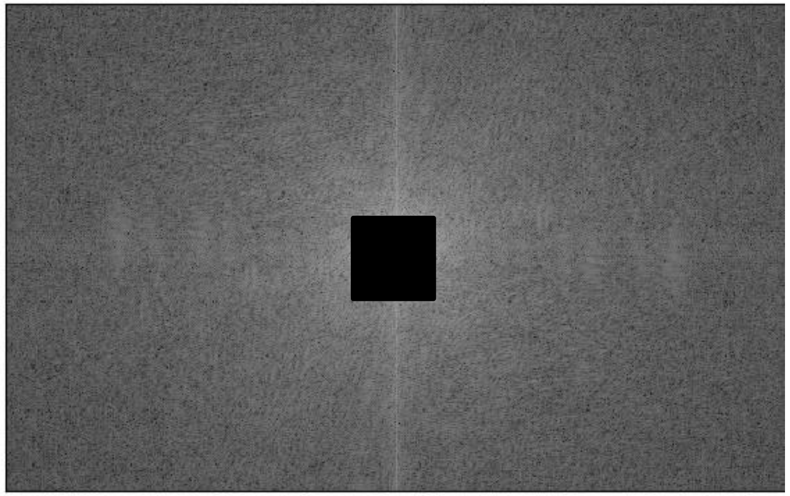
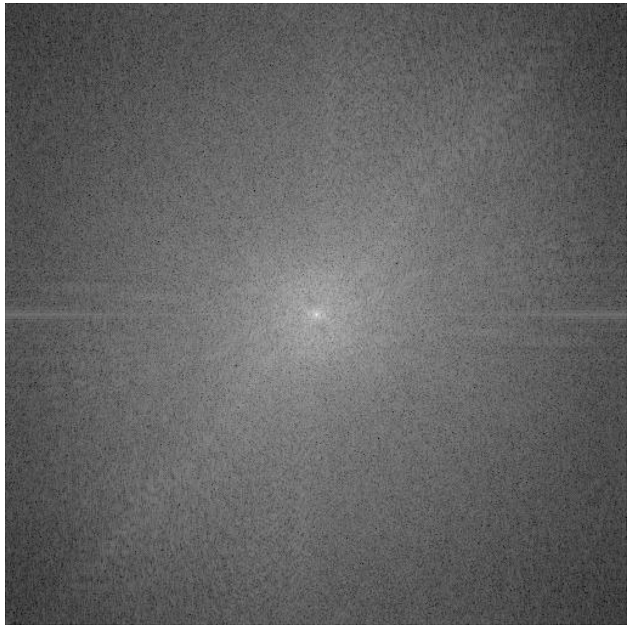


Image after HPF

Phase and magnitude

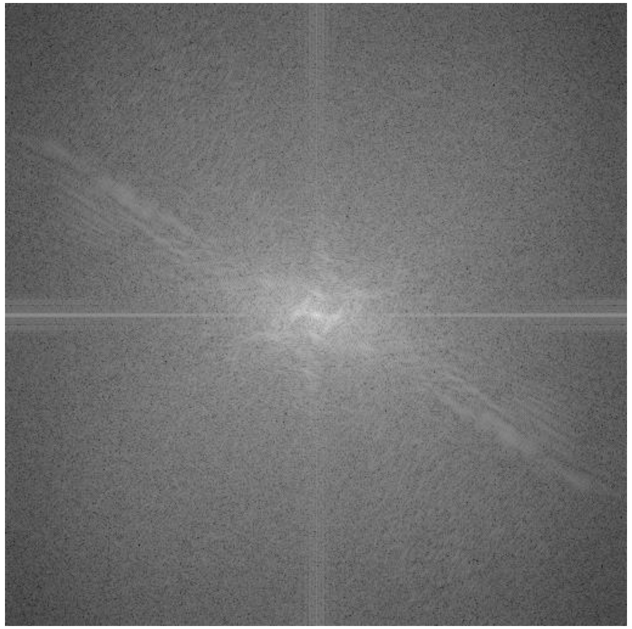
- Curious fact
  - all natural images have about the same magnitude transform
  - hence, phase seems to matter, but magnitude largely doesn't
- Demonstration
  - Take two pictures, swap the phase transforms, compute the inverse – what does the result look like?

This is the magnitude transform of the cheetah picture

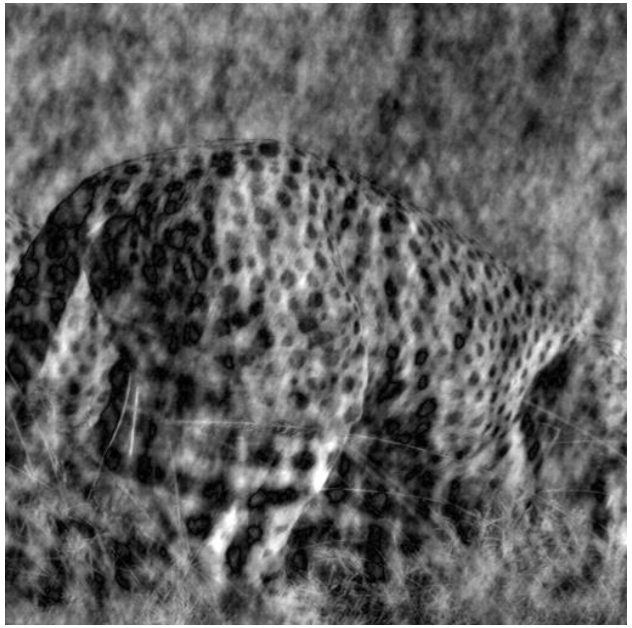


Reconstruction with zebra phase, cheetah magnitude

This is the magnitude transform of the zebra picture



Reconstruction with cheetah phase, zebra magnitude



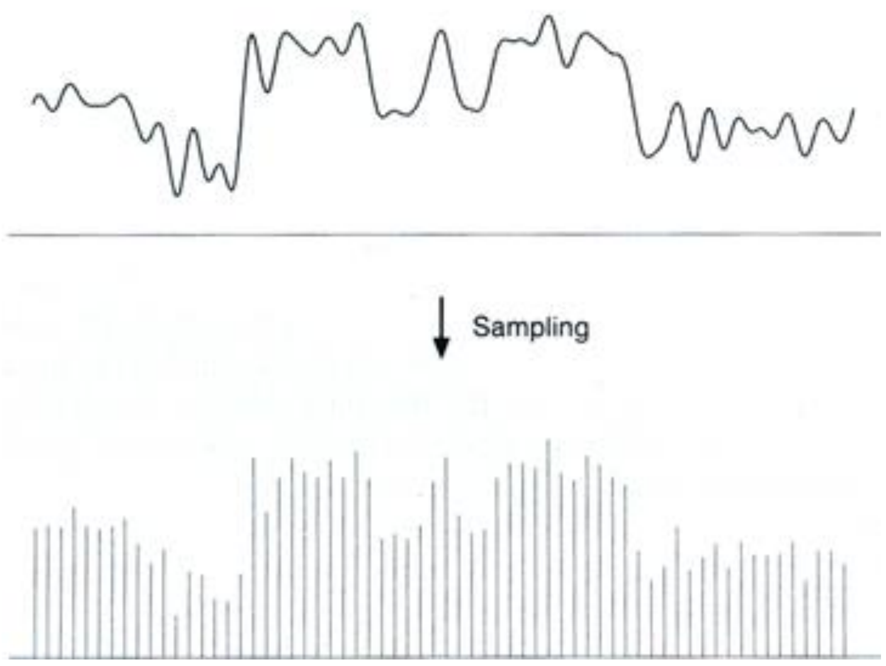
Sampling

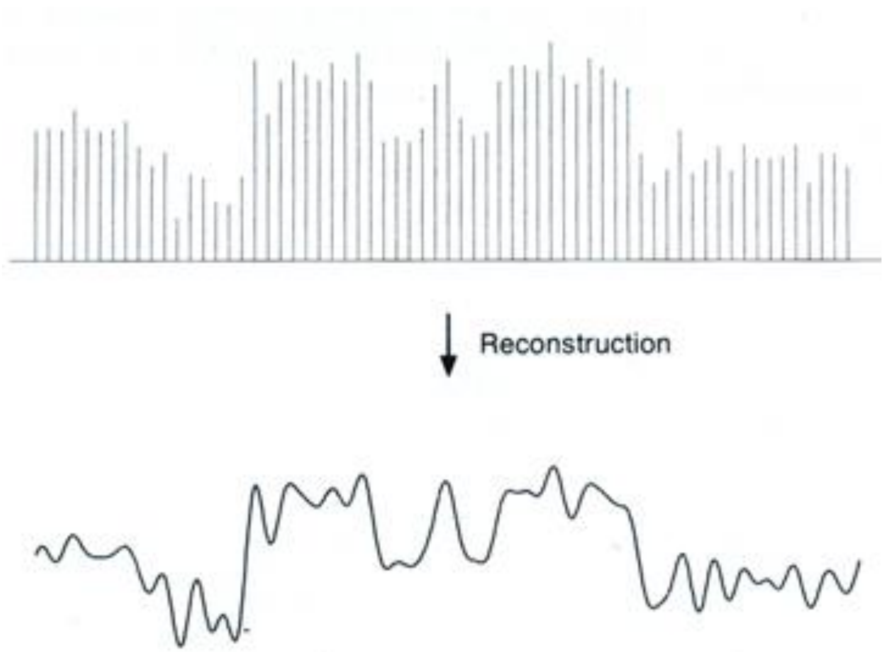
Why does a lower resolution image still make sense to us? What do we lose?

Image: http://www.flickr.com/photos/igorms/136916757/

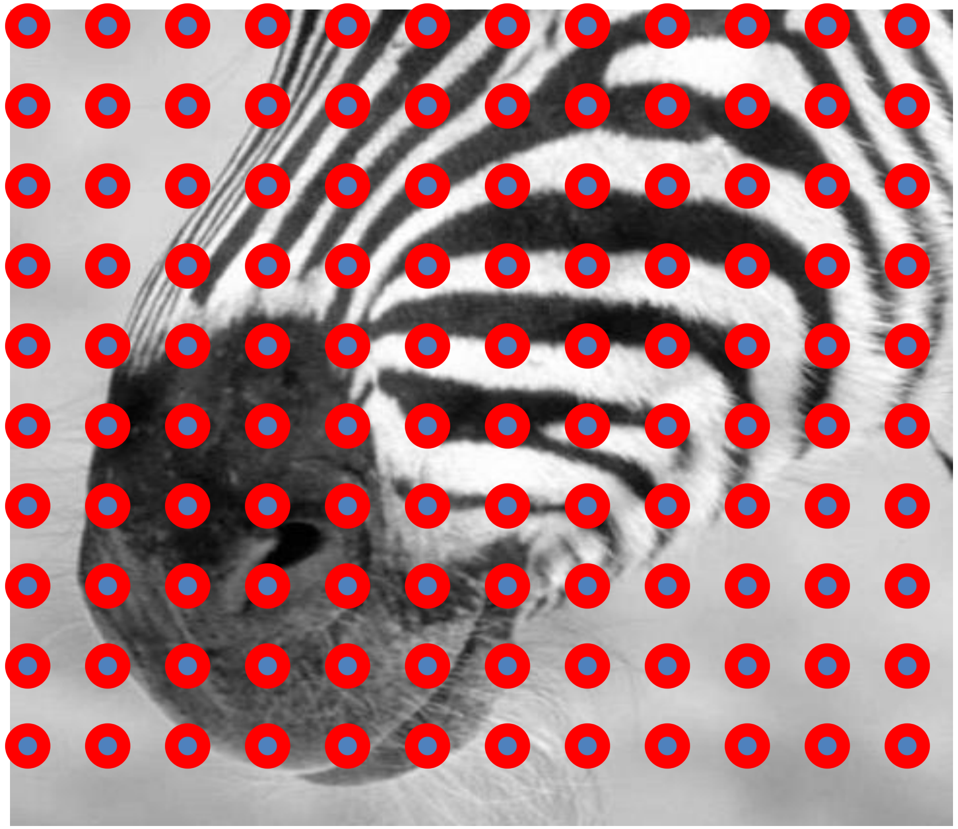
Sampled representations

- How to store and compute with continuous functions?
- Common scheme for representation: samples
  - write down the function's values at many points
- Reconstruction: making samples back into a continuous function





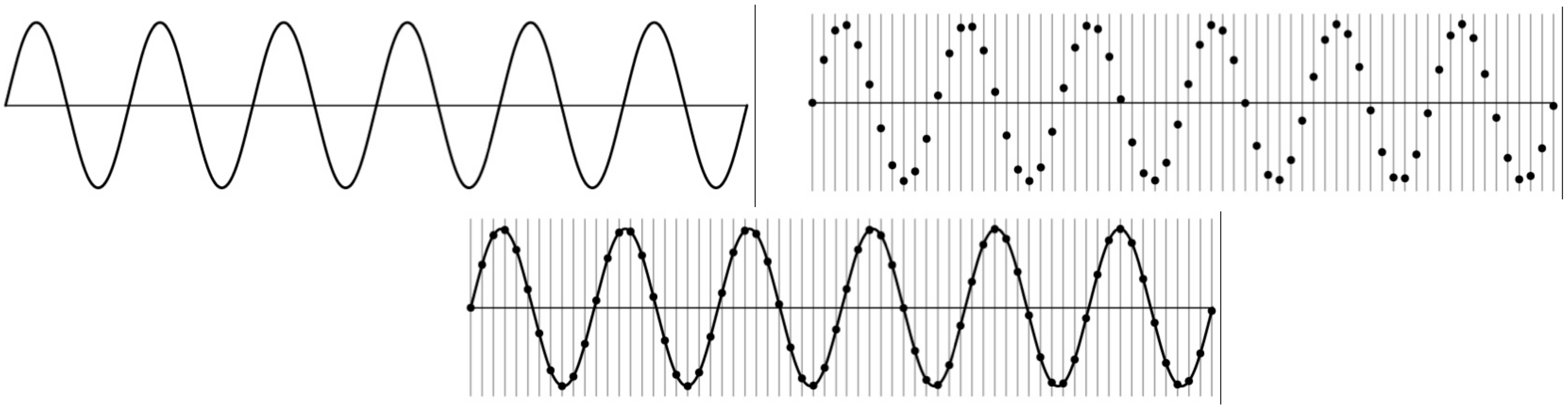
Subsampling by a factor of 2



Throw away every other row and
column to create a 1/2 size image

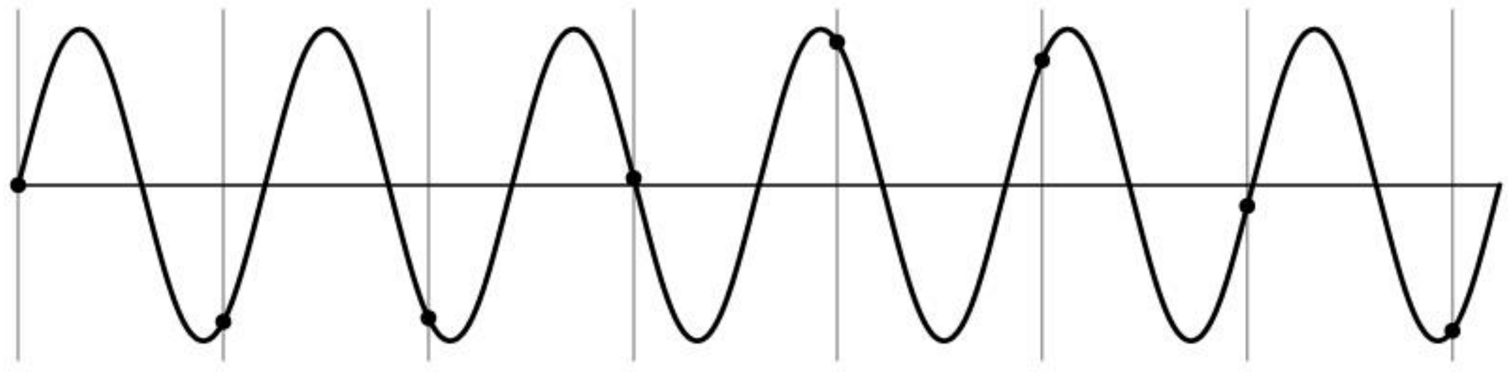
Undersampling

- What if we missed things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost
  - surprising result: indistinguishable from lower frequency
  - also was always indistinguishable from higher frequencies
  - aliasing: signals ‘traveling in disguise’ as other frequencies



Aliasing problem

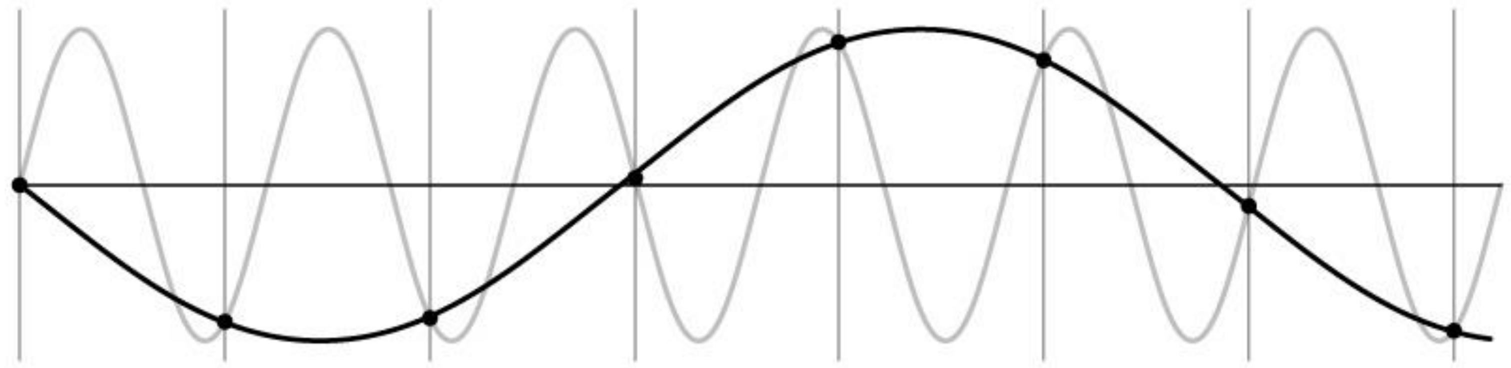
- 1D example (sinewave):



Source: S. Marschner

Aliasing problem

- 1D example (sinewave):



Source: S. Marschner

Aliasing problem

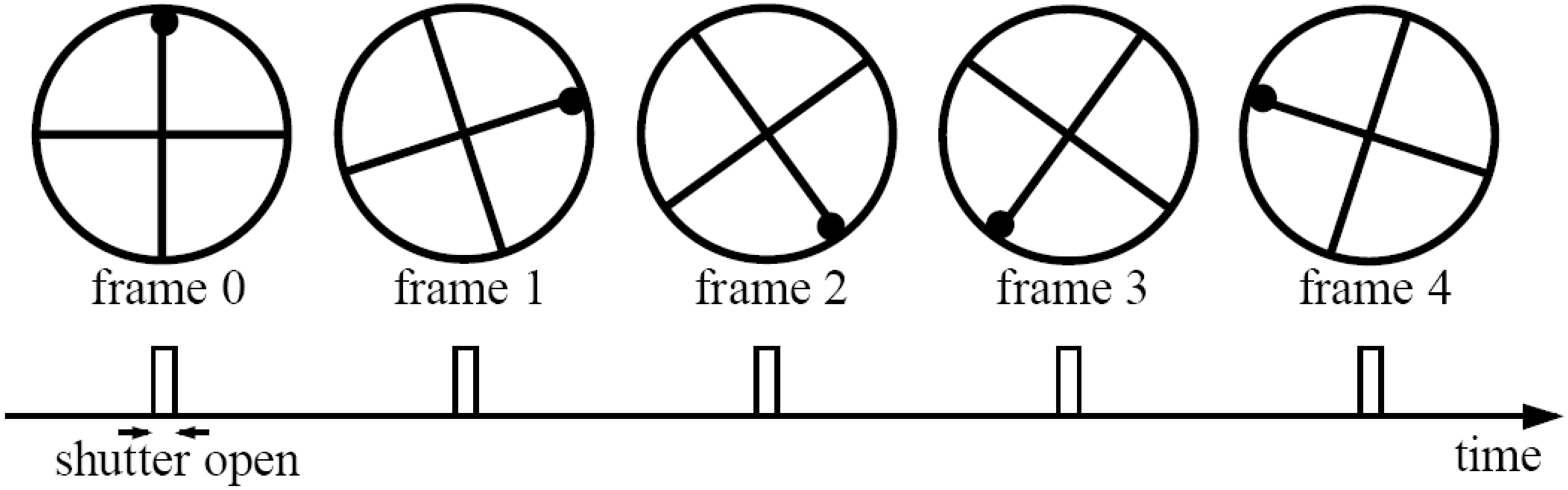
- Sub-sampling may be dangerous….
- Characteristic errors may appear:
  - “Wagon wheels rolling the wrong way in movies”
  - “Checkerboards disintegrate in ray tracing”
  - “Striped shirts look funny on color television”

Source: D. Forsyth

Aliasing in video

Imagine a spoked wheel moving to the right (rotating clockwise). Mark wheel with dot so we can see what's happening.

If camera shutter is only open for a fraction of a frame time (frame time = 1/30 sec. for video, 1/24 sec. for film):

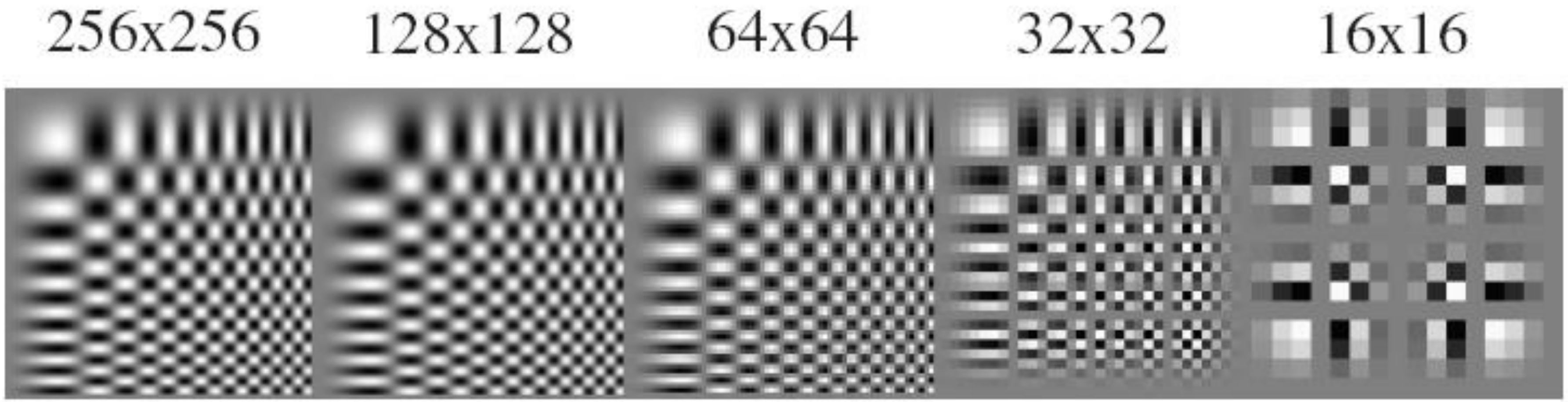


Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)

Aliasing in graphics

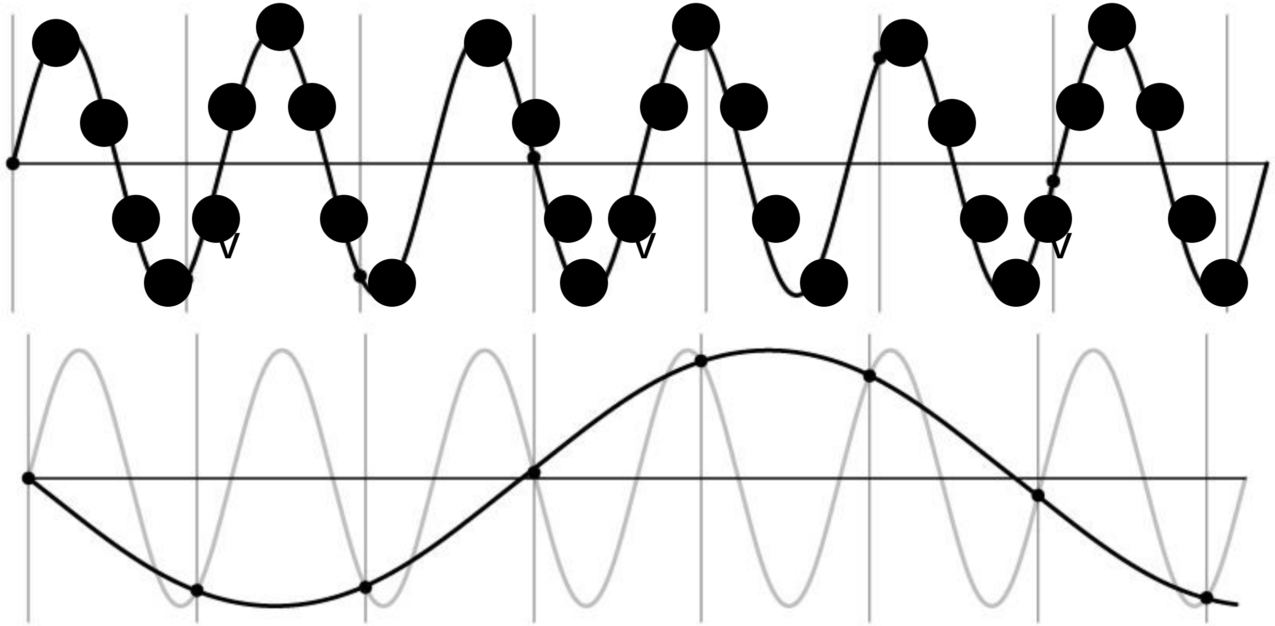
Source: A. Efros

Sampling and aliasing



Nyquist-Shannon Sampling Theorem

- When sampling a signal at discrete intervals, the sampling frequency must be $\geq 2 \times f_{max}$
- f_{max} = max frequency of the input signal
- This will allows to reconstruct the original perfectly from the sampled version



good

bad

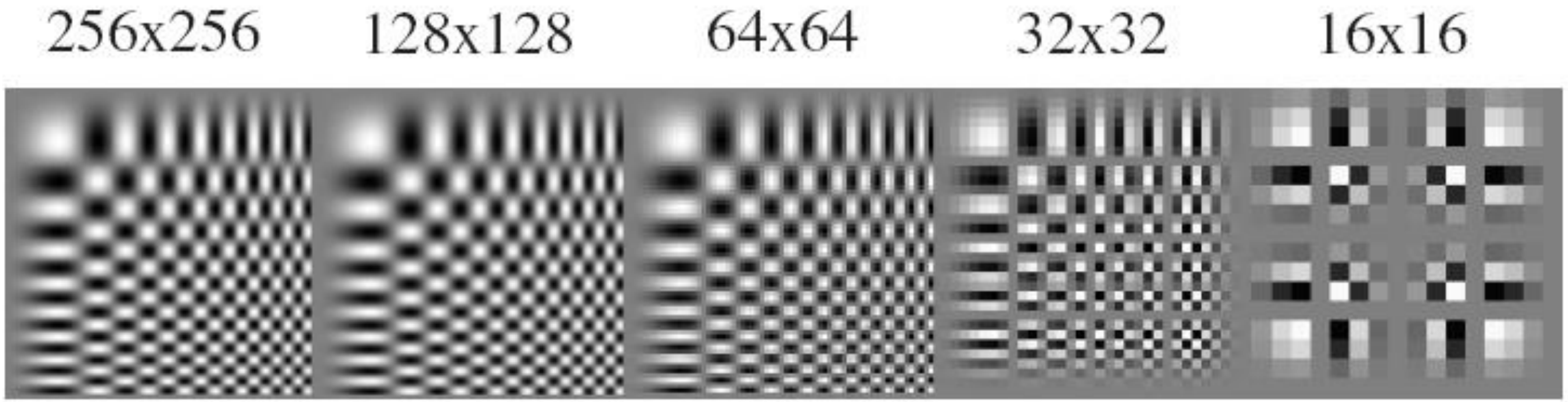
Anti-aliasing

Solutions:

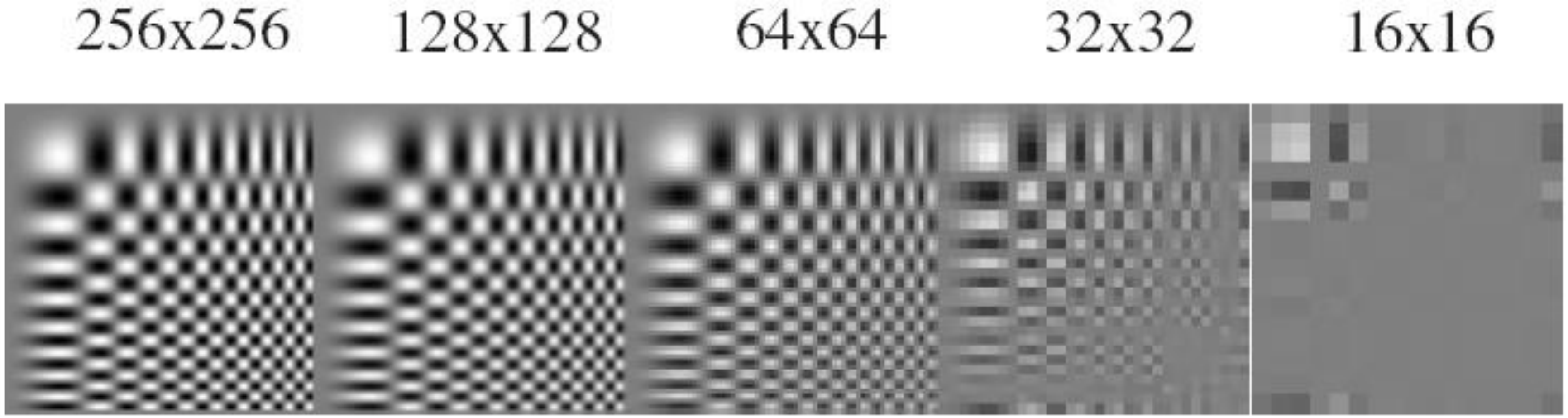
- Sample more often

- Get rid of all frequencies that are greater than half the new sampling frequency
  - Will lose information
  - But it’s better than aliasing
  - Apply a smoothing filter

Anti-aliasing

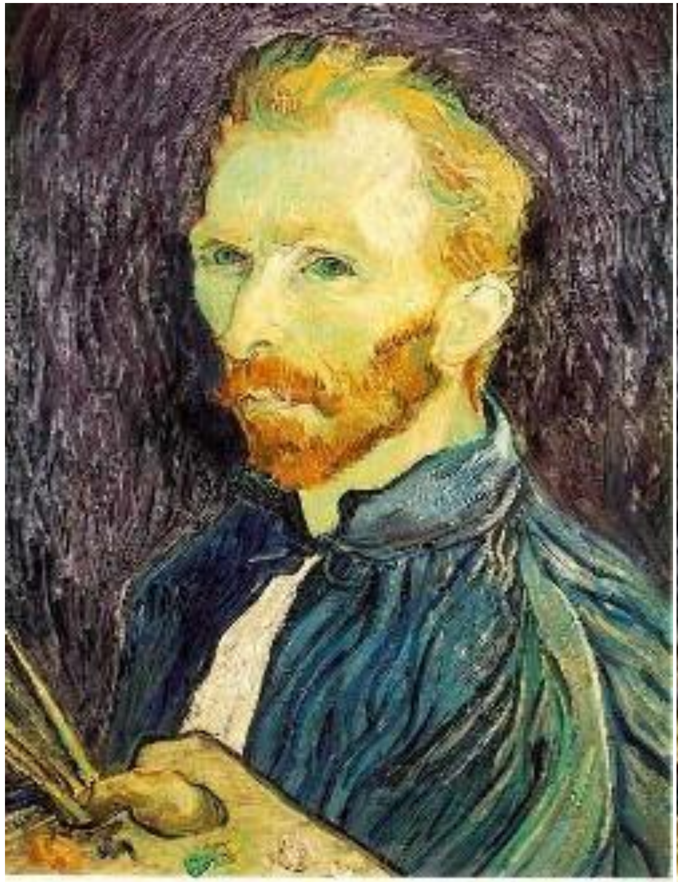


Apply low-pass filter and Sample every other pixel



Forsyth and Ponce 2002

Subsampling without pre-filtering



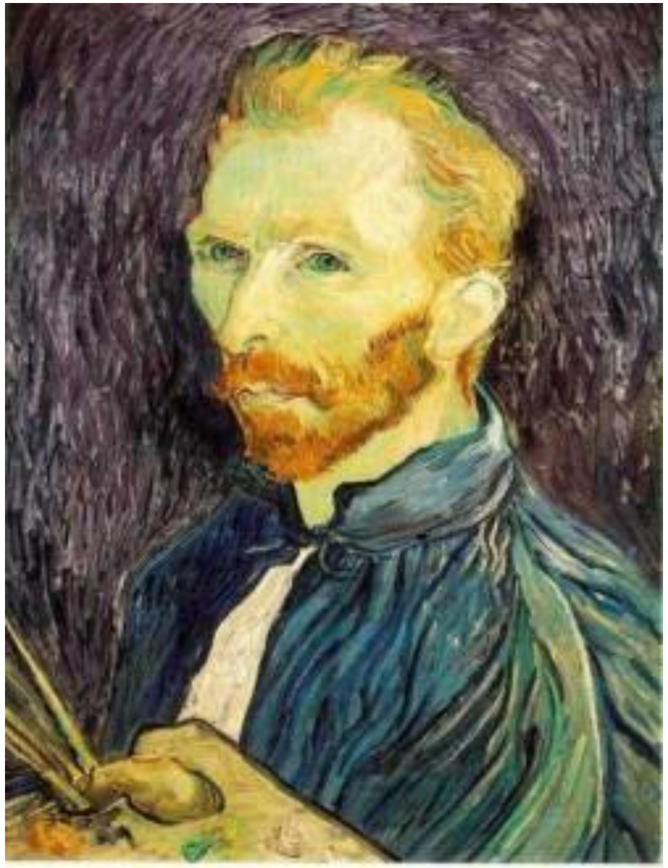
1/2

1/4 (2x zoom)

1/8 (4x zoom)

Slide by Steve Seitz

Subsampling with Gaussian pre-filtering



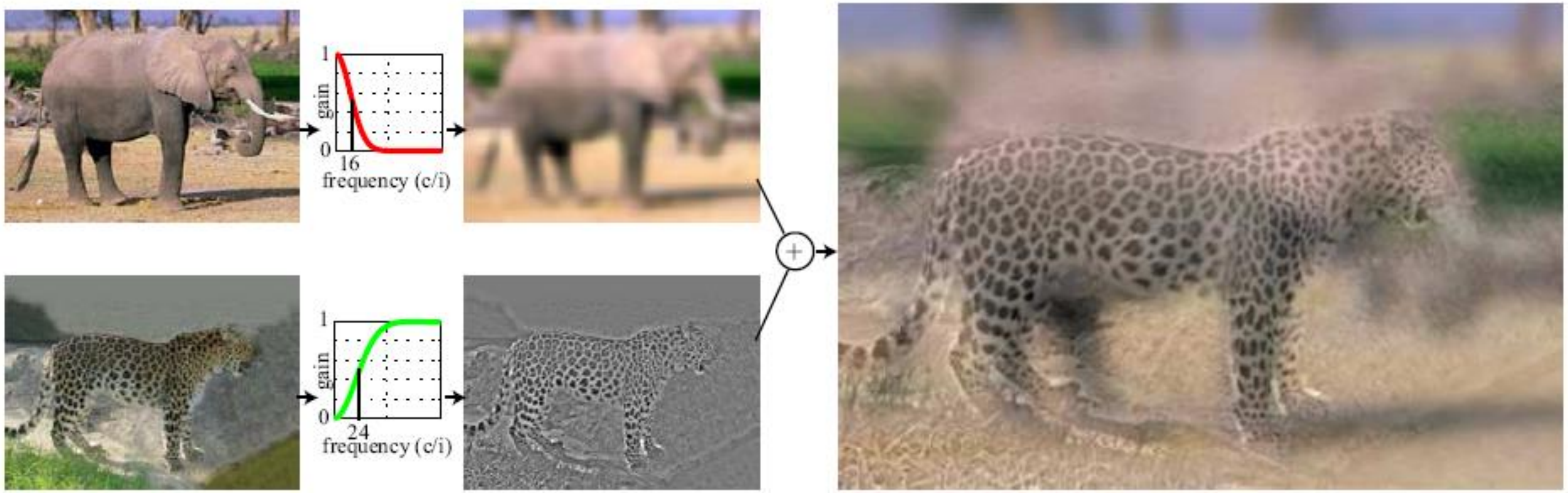
Gaussian 1/2

G 1/4

G 1/8

Slide by Steve Seitz

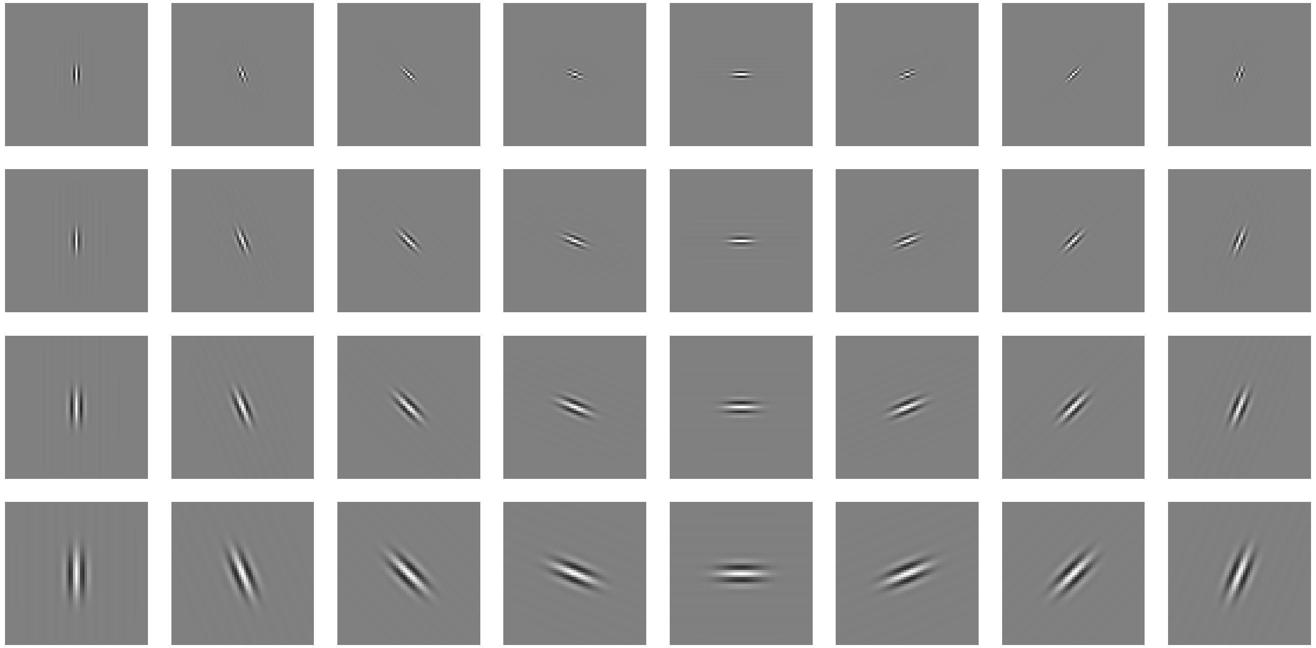
Hybrid Images



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

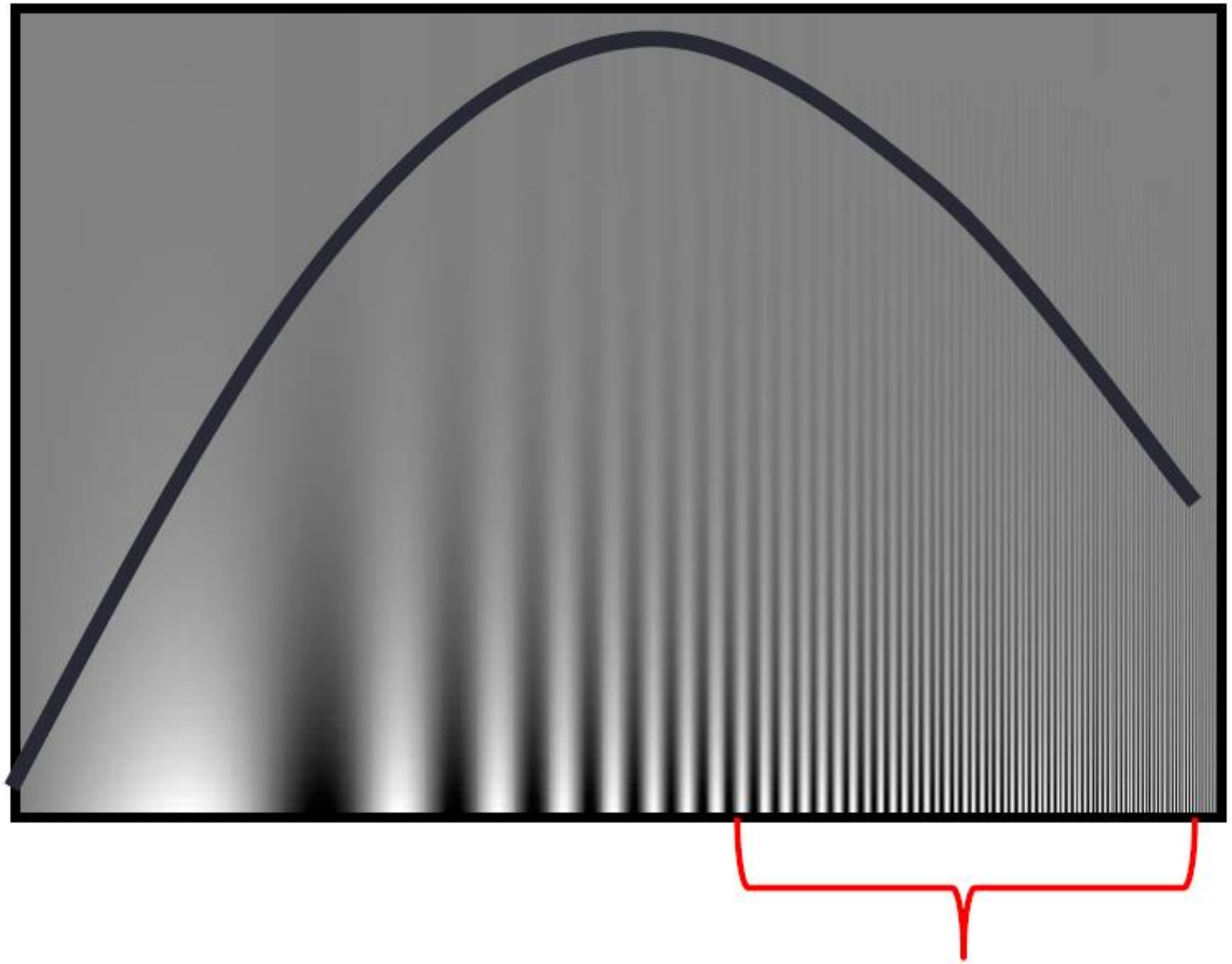
Clues from human perception

- Early processing in humans filters for various orientations and scales of frequency
- Perceptual cues in the mid-high frequencies dominate perception
- When we see an image from far away, we are effectively subsampling it



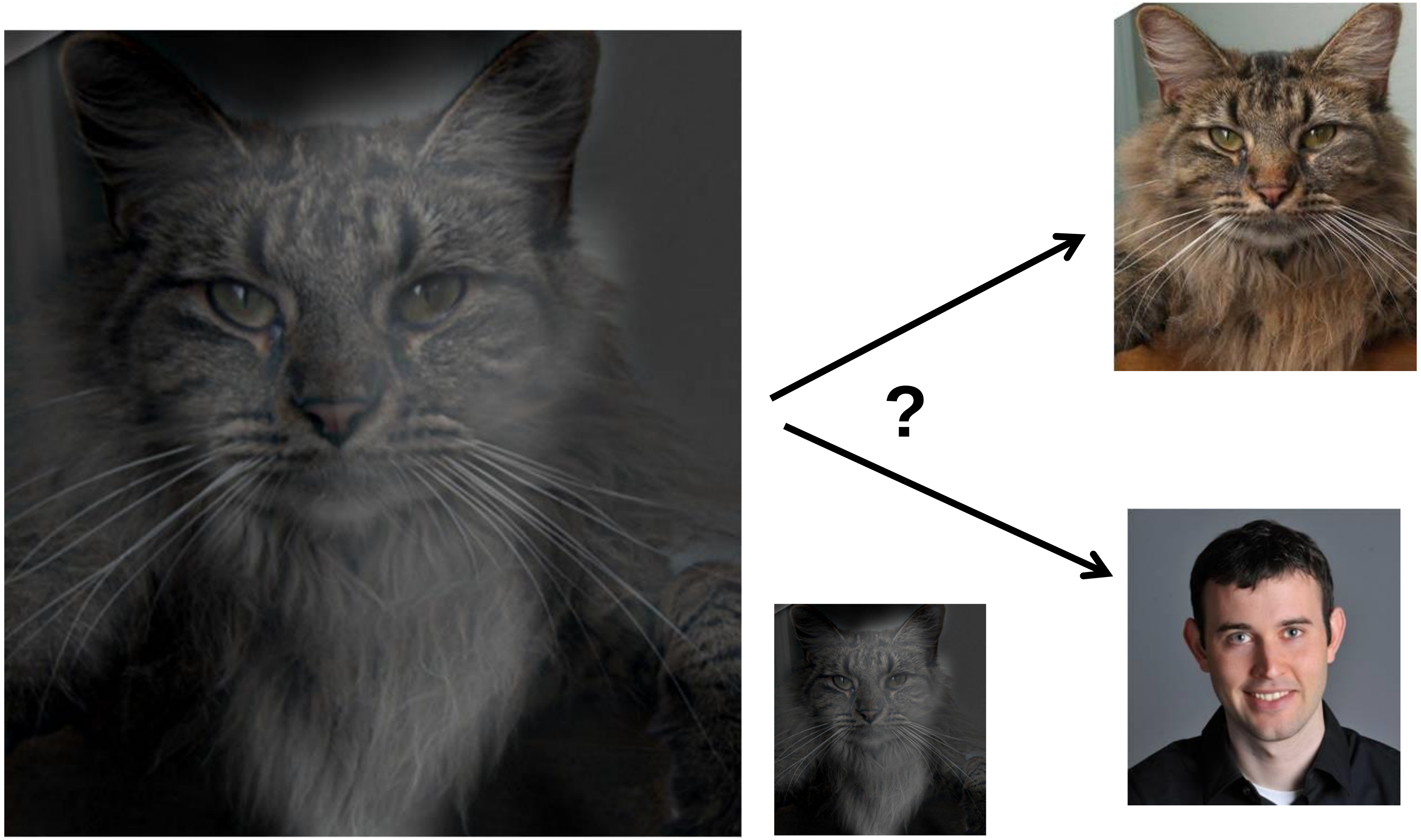
Early Visual Processing: Multi-scale edge and blob filters

Campbell-Robson contrast sensitivity curve



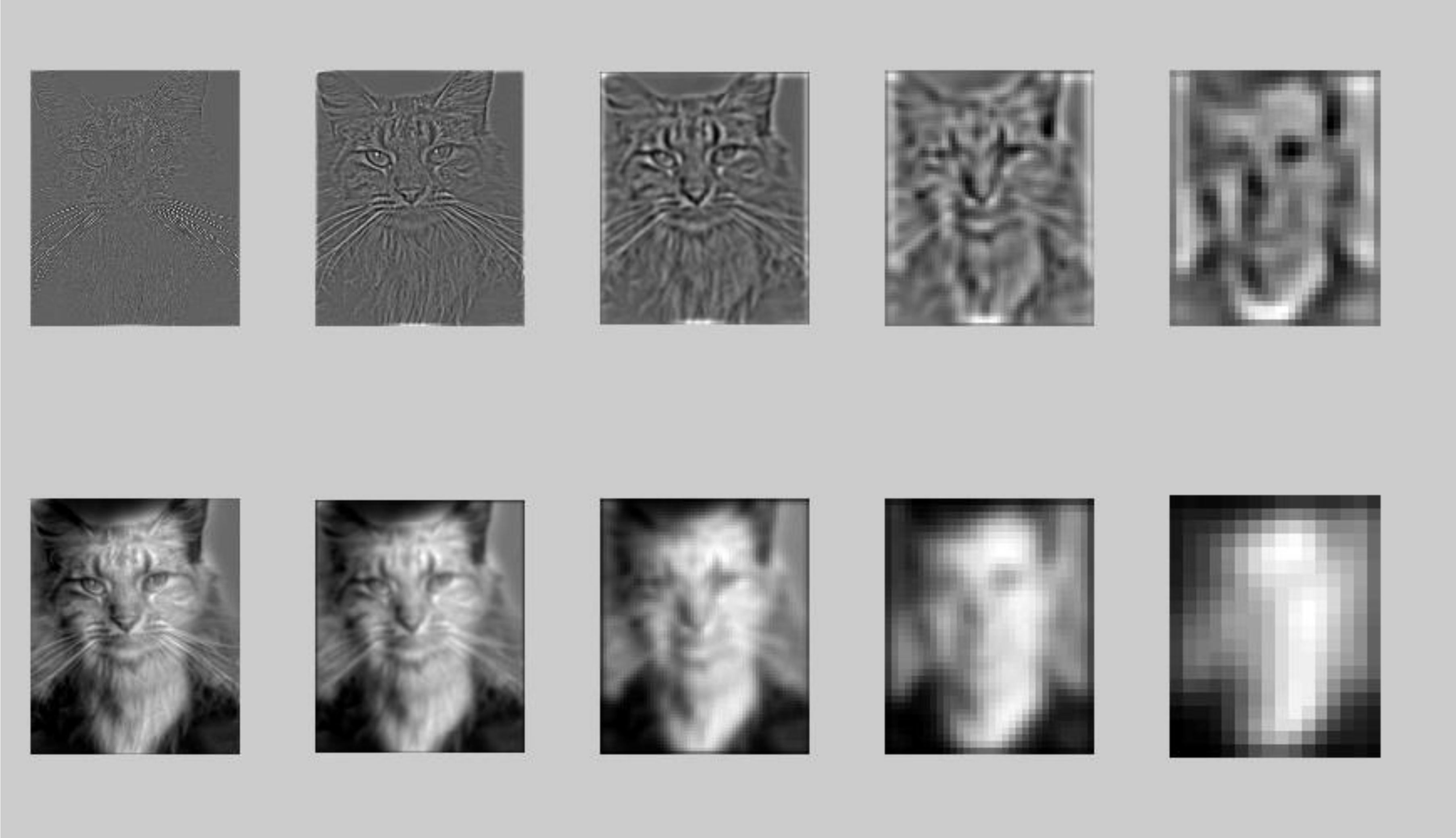
The higher the frequency the less sensitive human visual system is…

Why do we get different, distance-dependent interpretations of hybrid images?



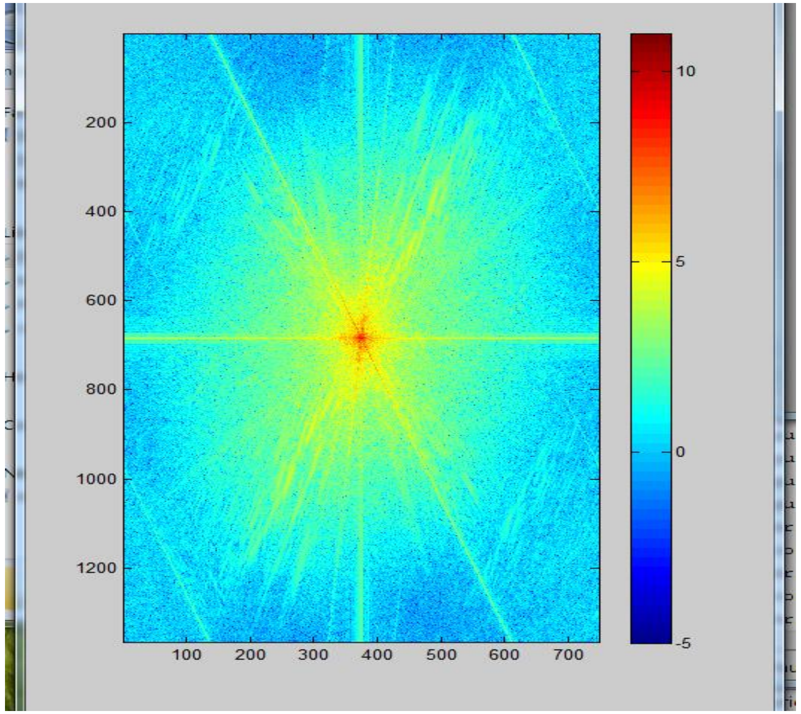
Hybrid Image in Laplacian Pyramid

High frequency → Low frequency

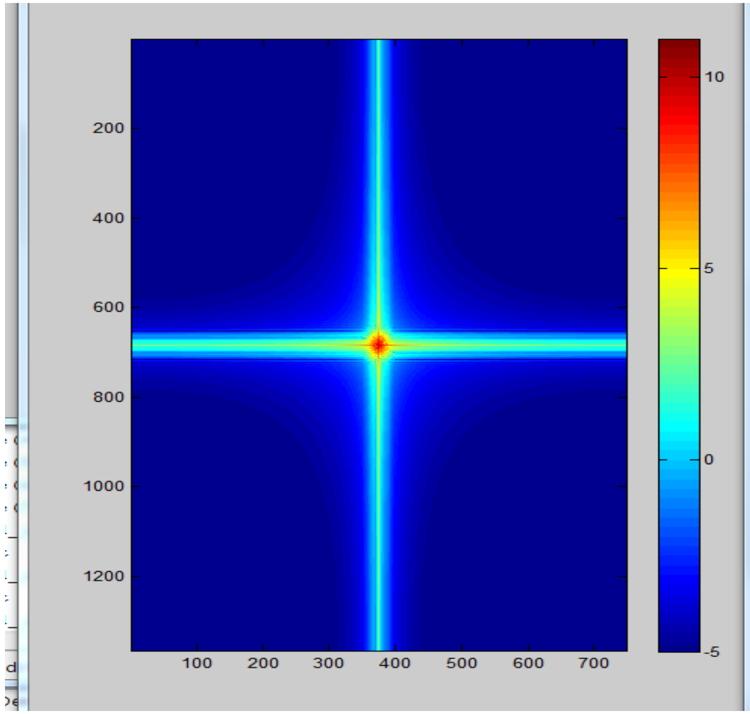


Hybrid Image in FFT

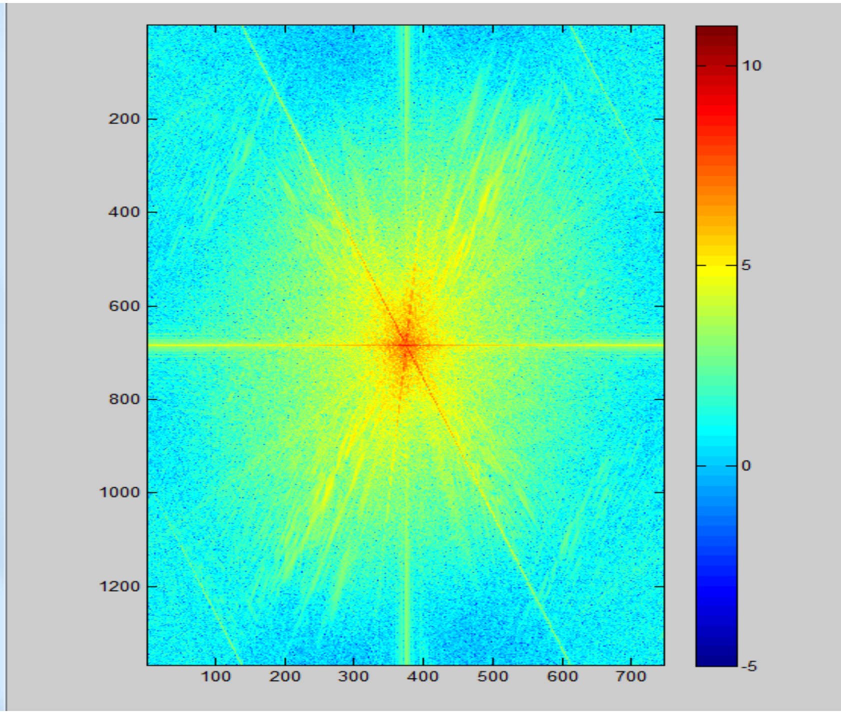
Hybrid Image



Low-passed Image

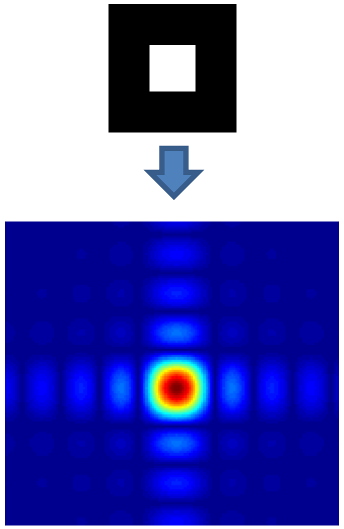


High-passed Image

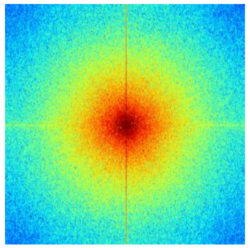


Things to Remember

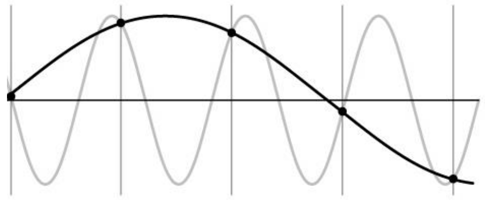
- Sometimes it makes sense to think of images and filtering in the frequency domain
  - Fourier analysis



- Can be faster to filter using FFT for large images (N logN vs. N^2 for convolution)



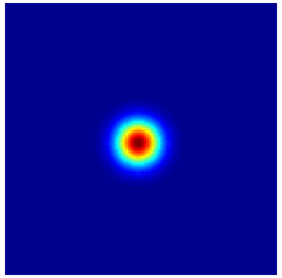
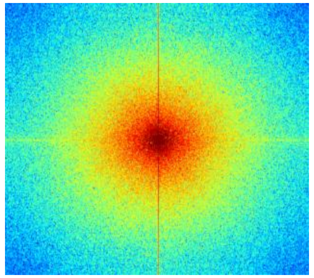
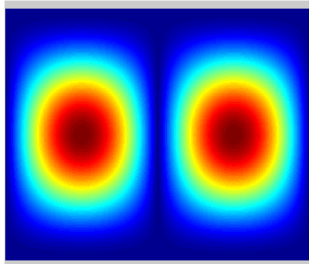
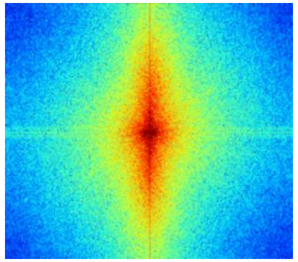
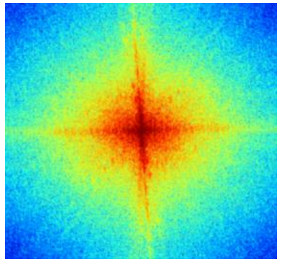
- Images are mostly smooth
  - Basis for compression



- Remember to low-pass before sampling

Practice question

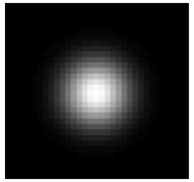
1. Match the spatial domain image to the Fourier magnitude image

1

2

3

4

5


A

B

C

D

E