

# Thinking in Frequency

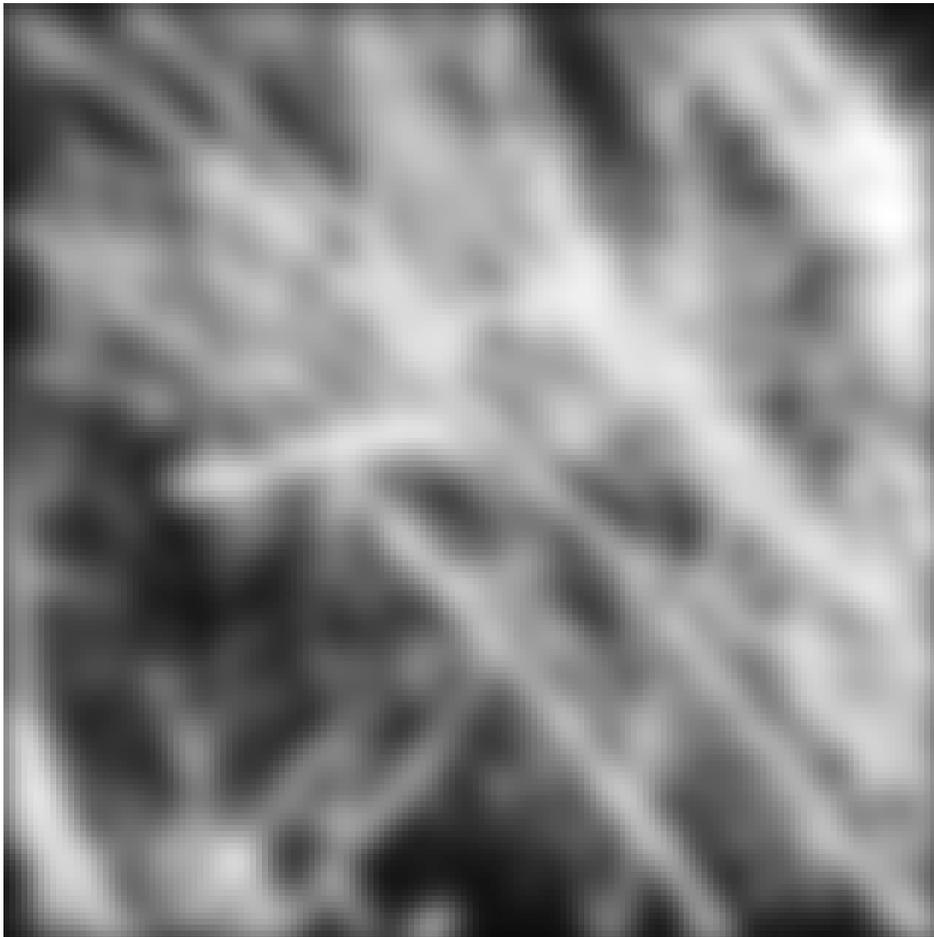
Computer Vision

Brown

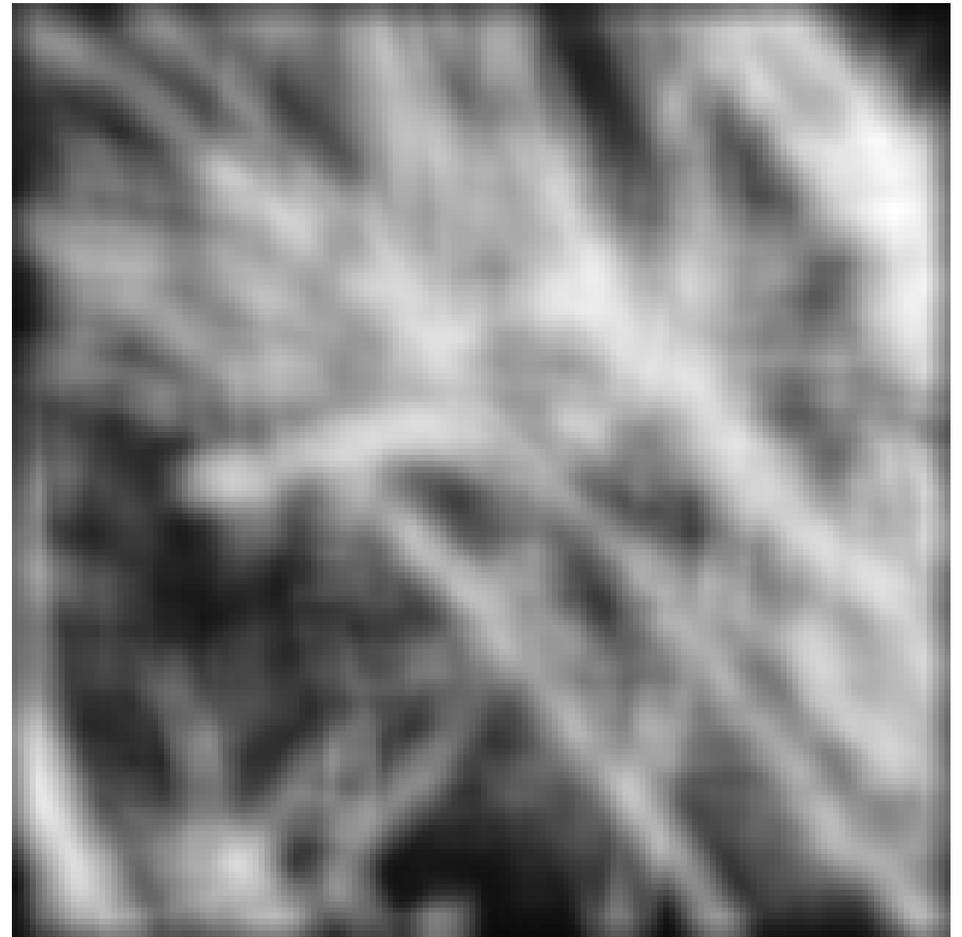
James Hays

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

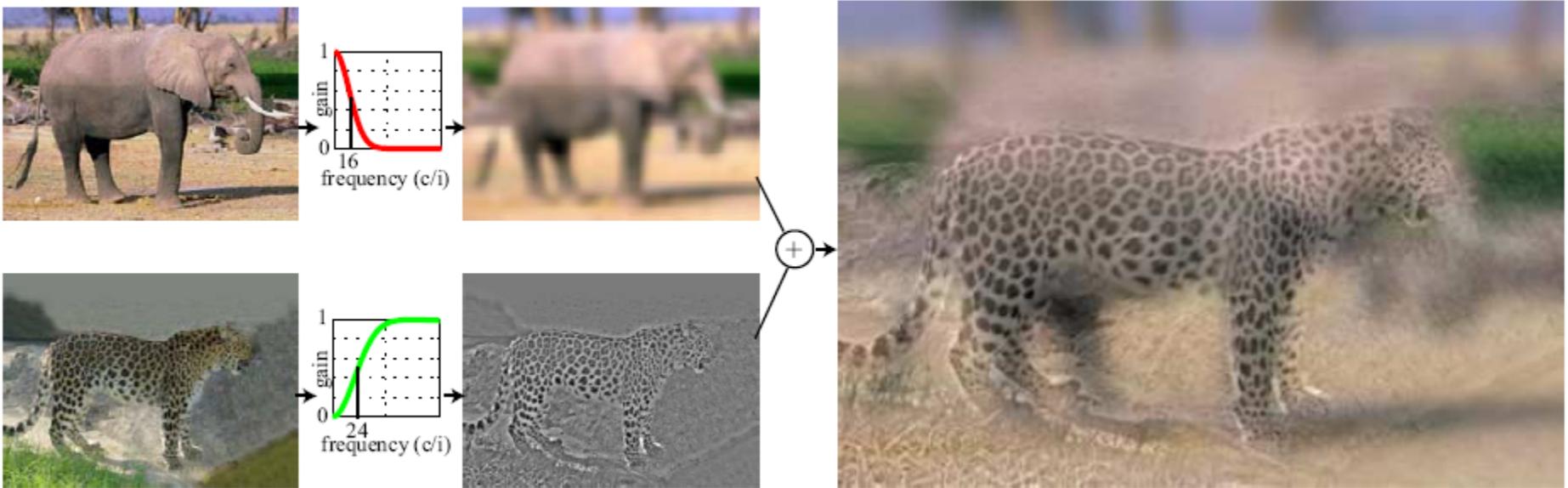
Gaussian



Box filter

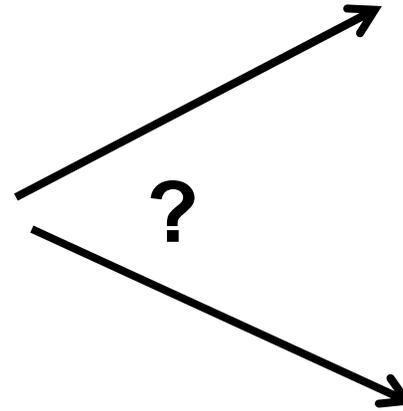


# Hybrid Images

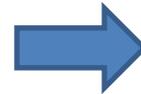


- A. Oliva, A. Torralba, P.G. Schyns, [“Hybrid Images,”](#) SIGGRAPH 2006

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



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



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

Thinking in terms of frequency

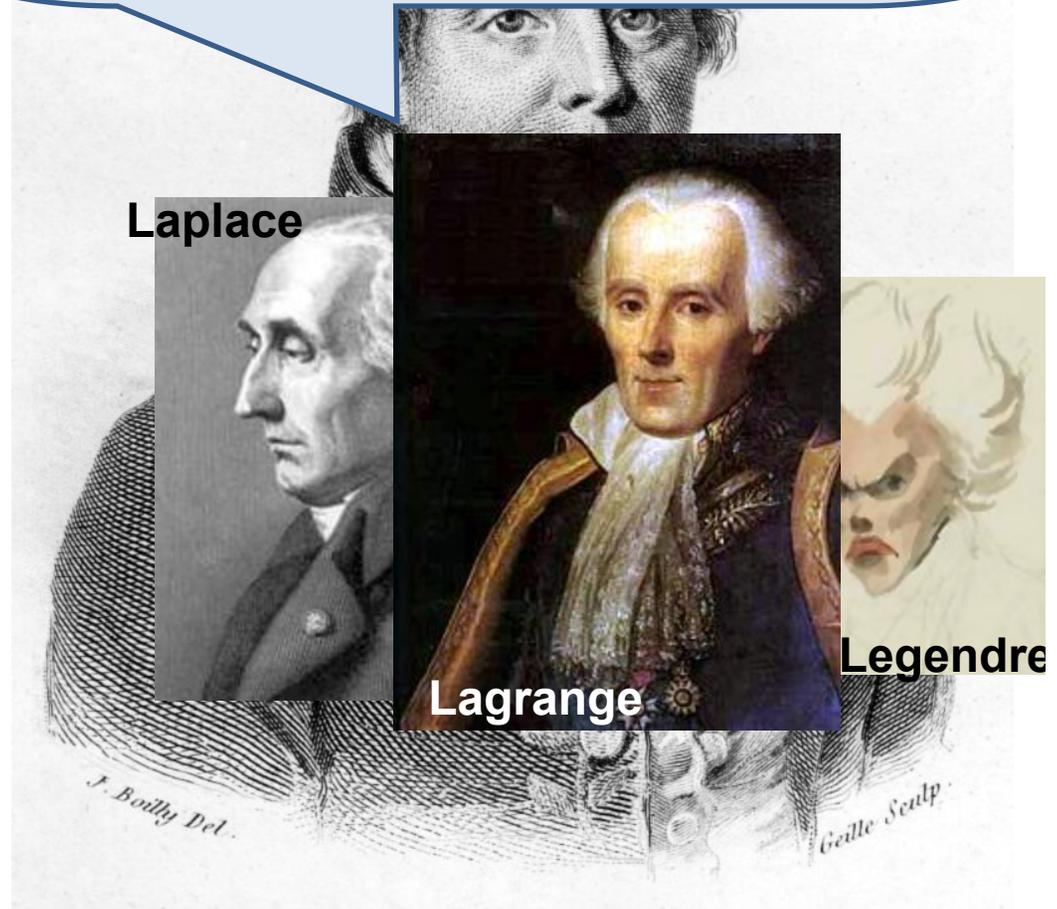
# Jean Baptiste Joseph Fourier (1768-1830)

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

*...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.*

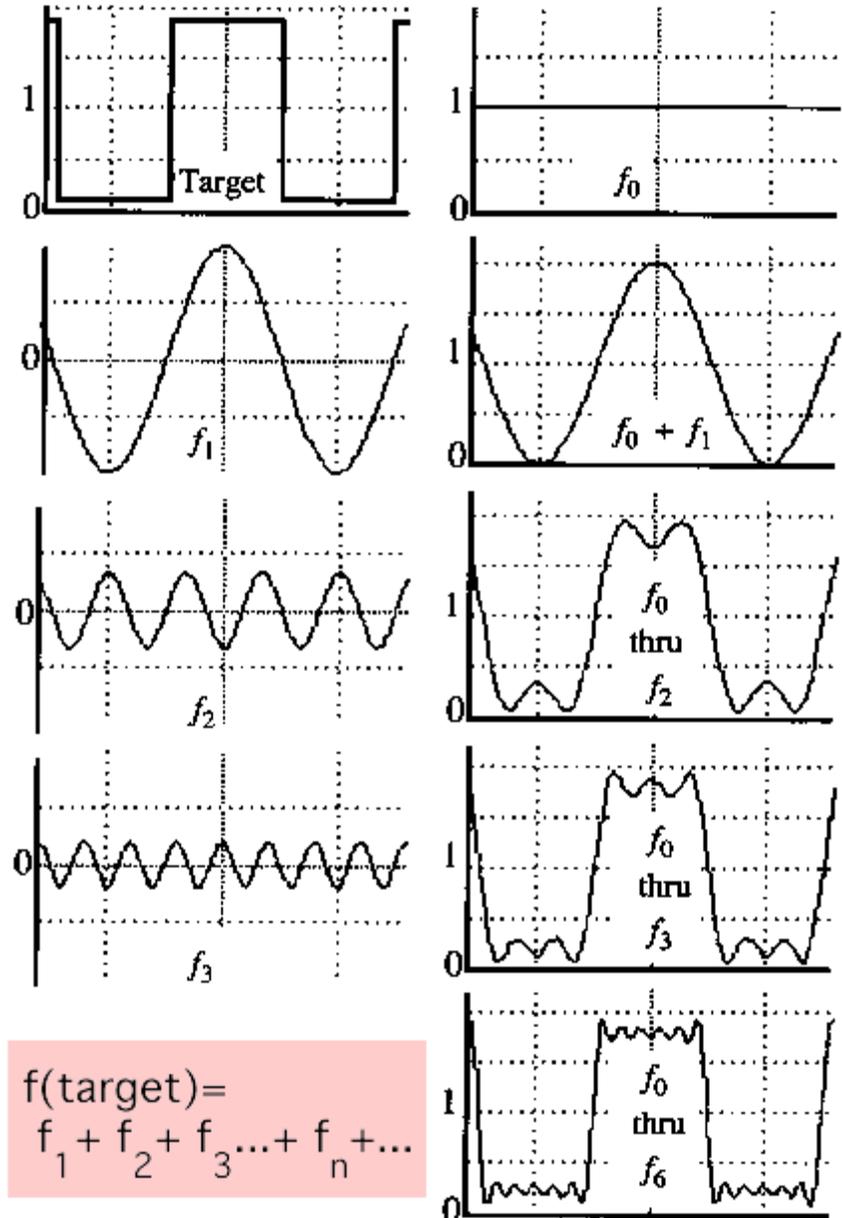


# A sum of sines

Our building block:

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

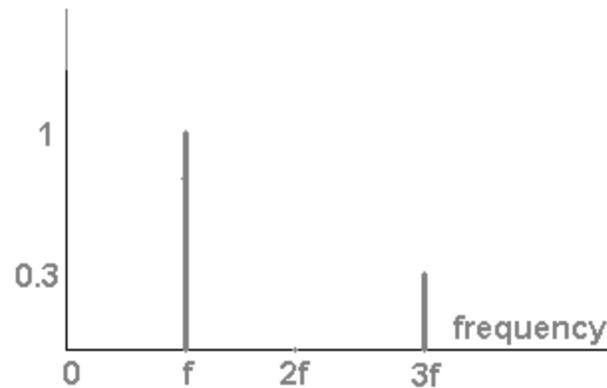
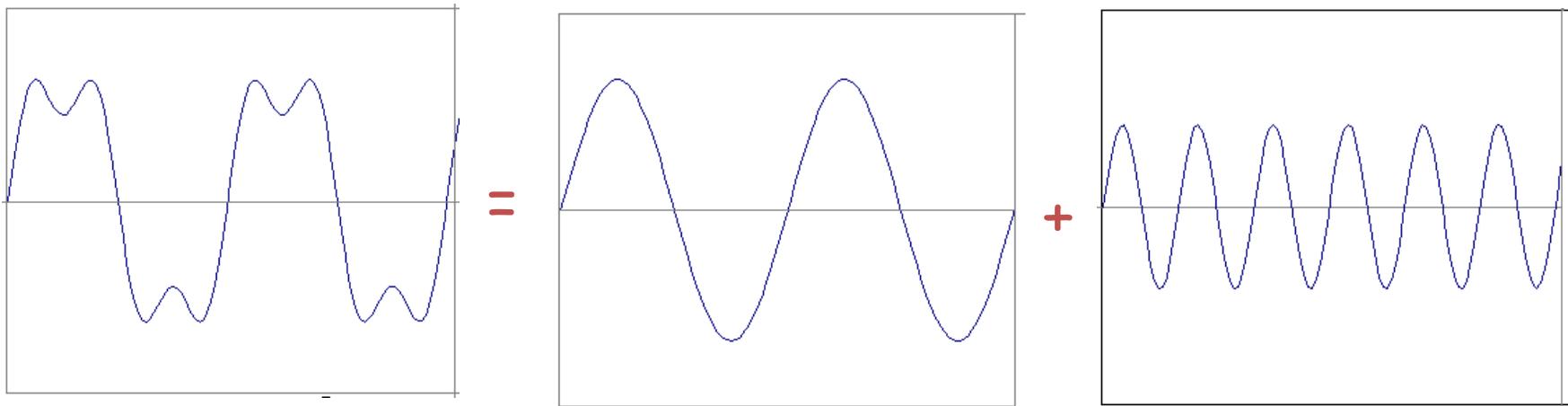
Add enough of them to get any signal  $f(x)$  you want!



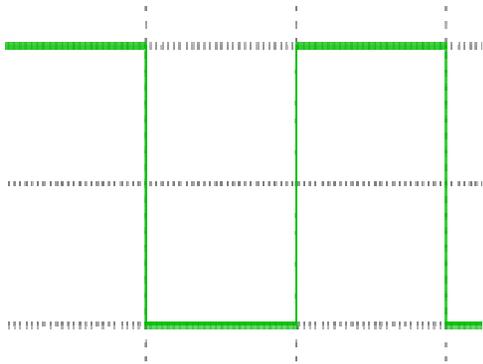
$$f(\text{target}) = f_1 + f_2 + f_3 + \dots + f_n + \dots$$

# Frequency Spectra

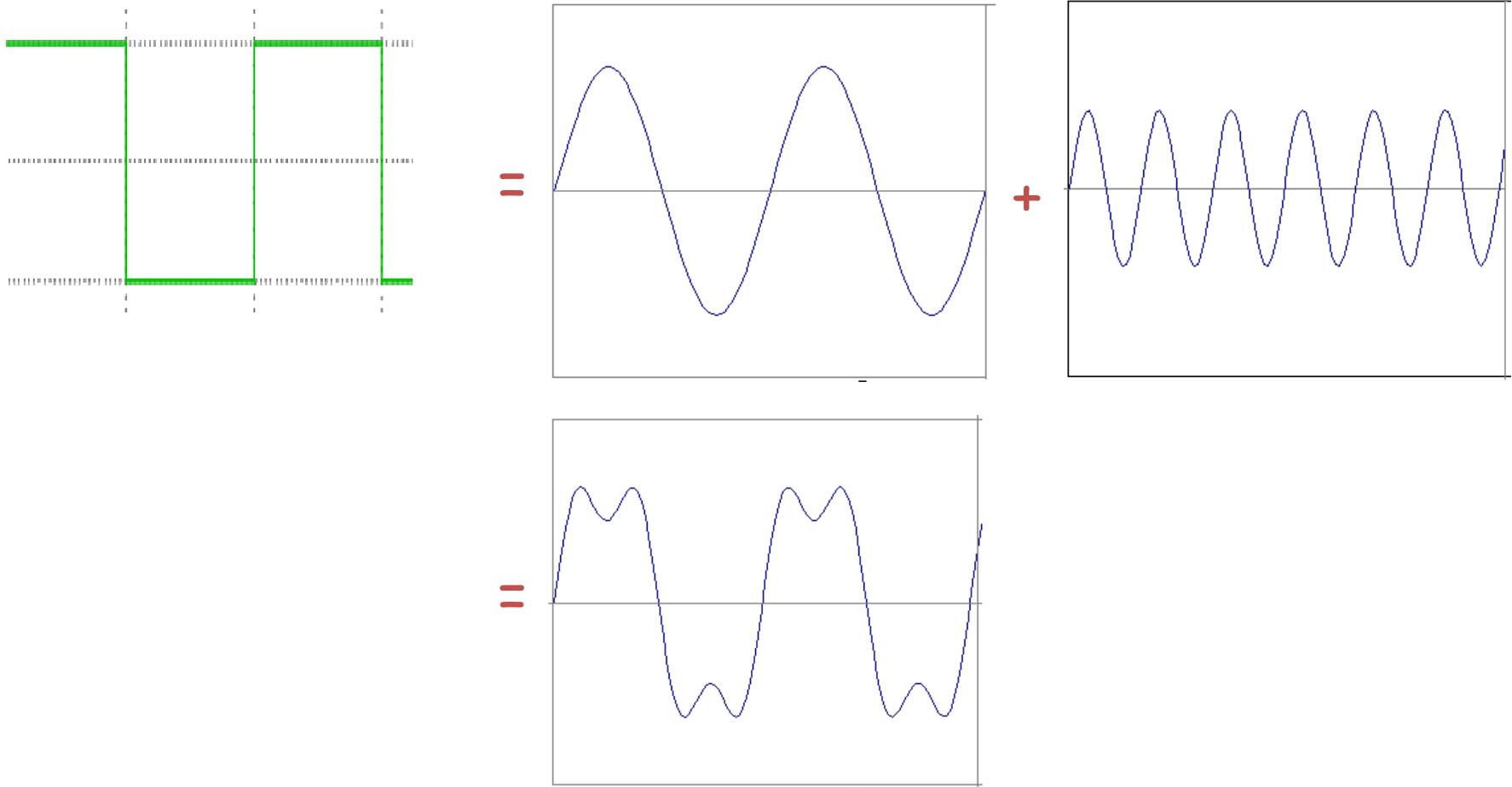
- example :  $g(t) = \sin(2\pi f t) + (1/3)\sin(2\pi(3f)t)$



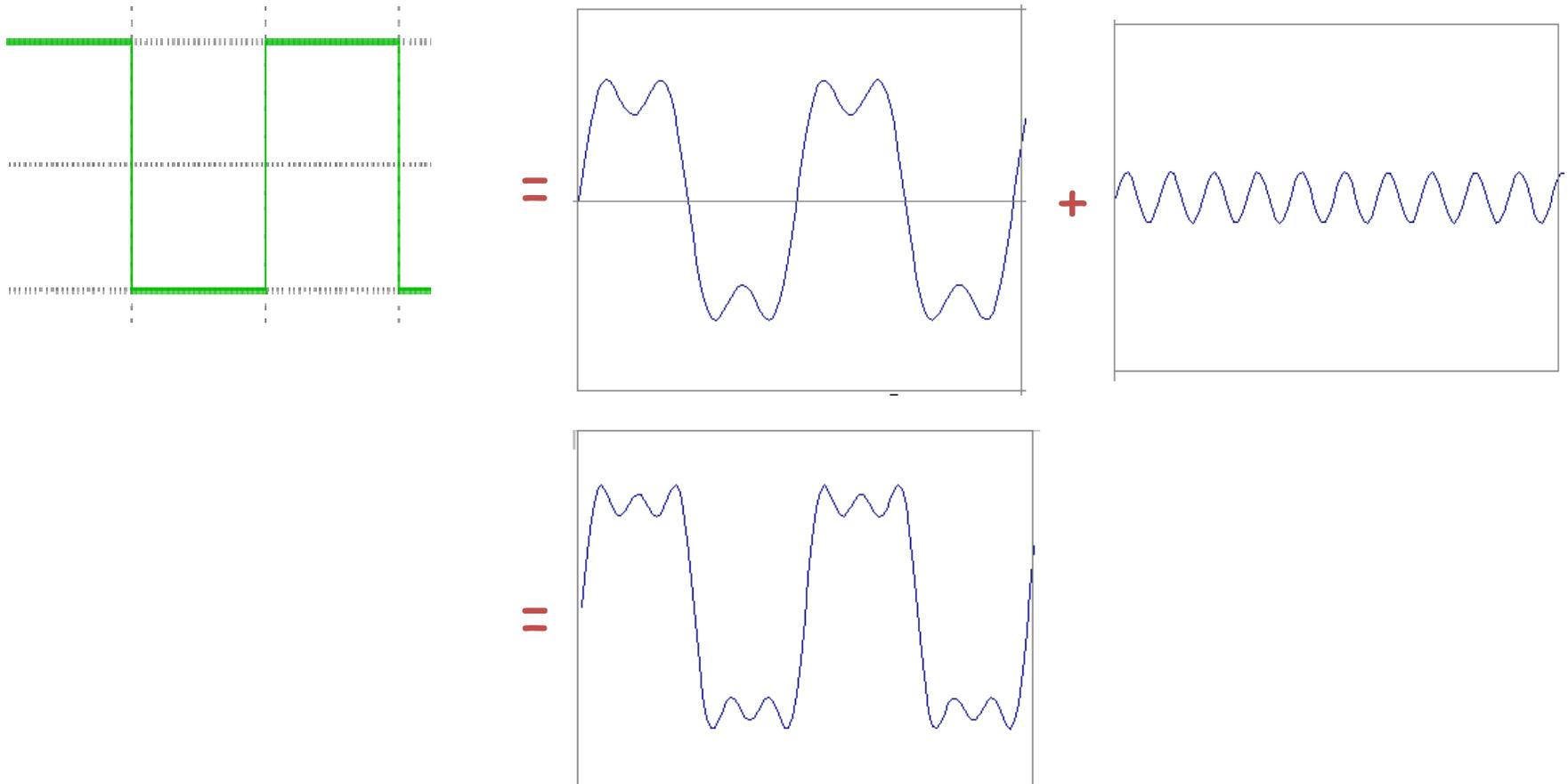
# Frequency Spectra



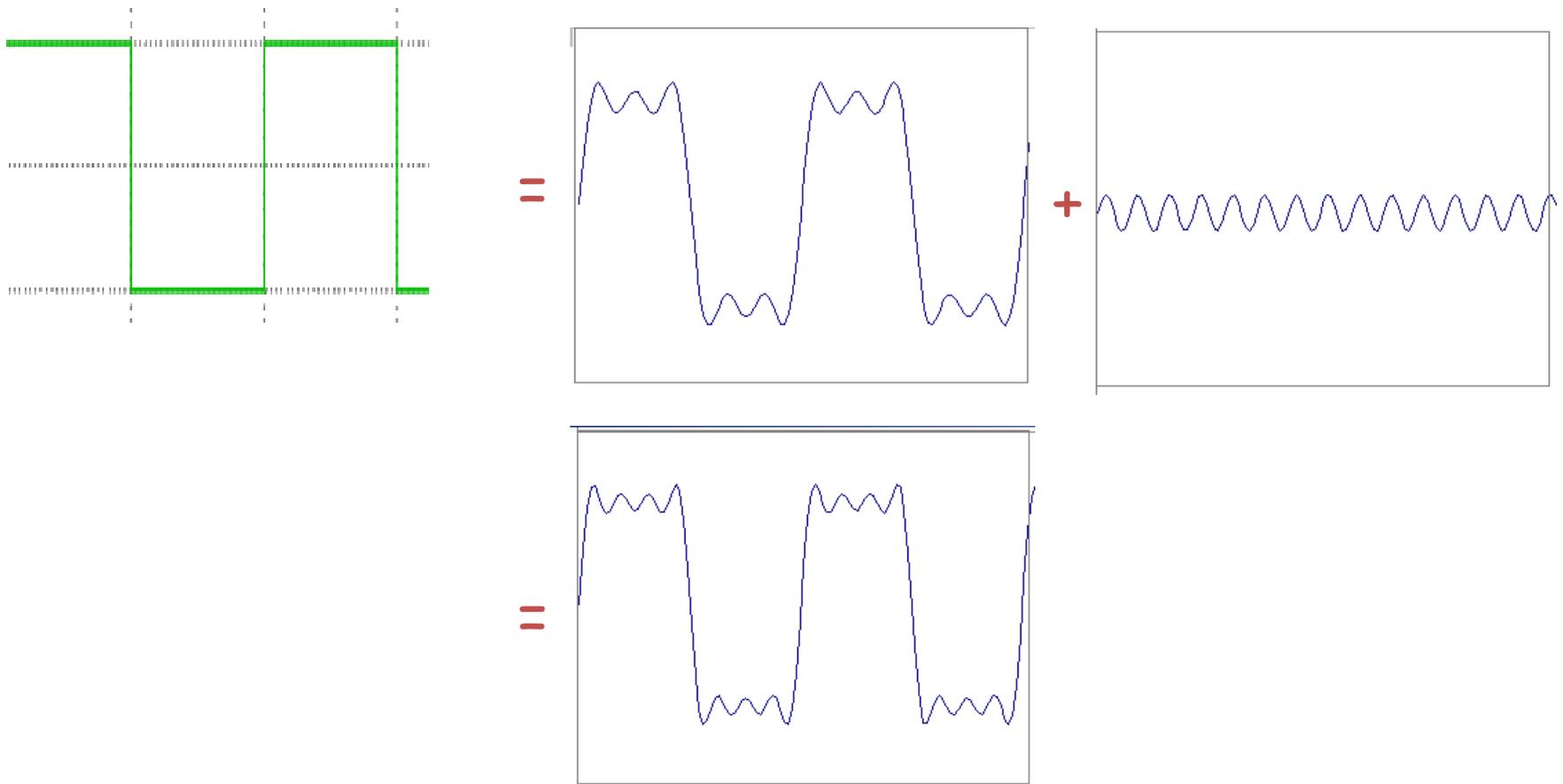
# Frequency Spectra



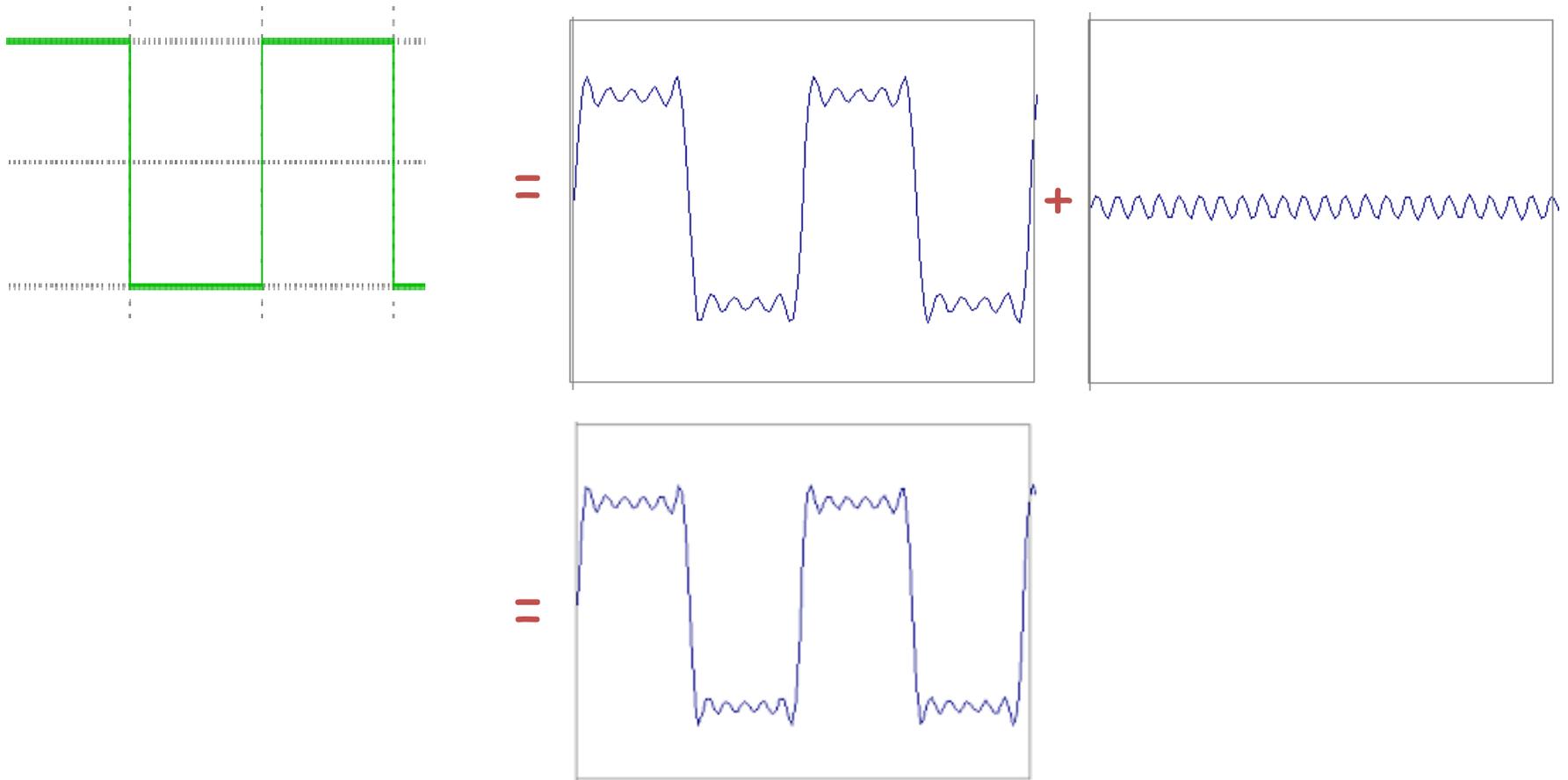
# Frequency Spectra



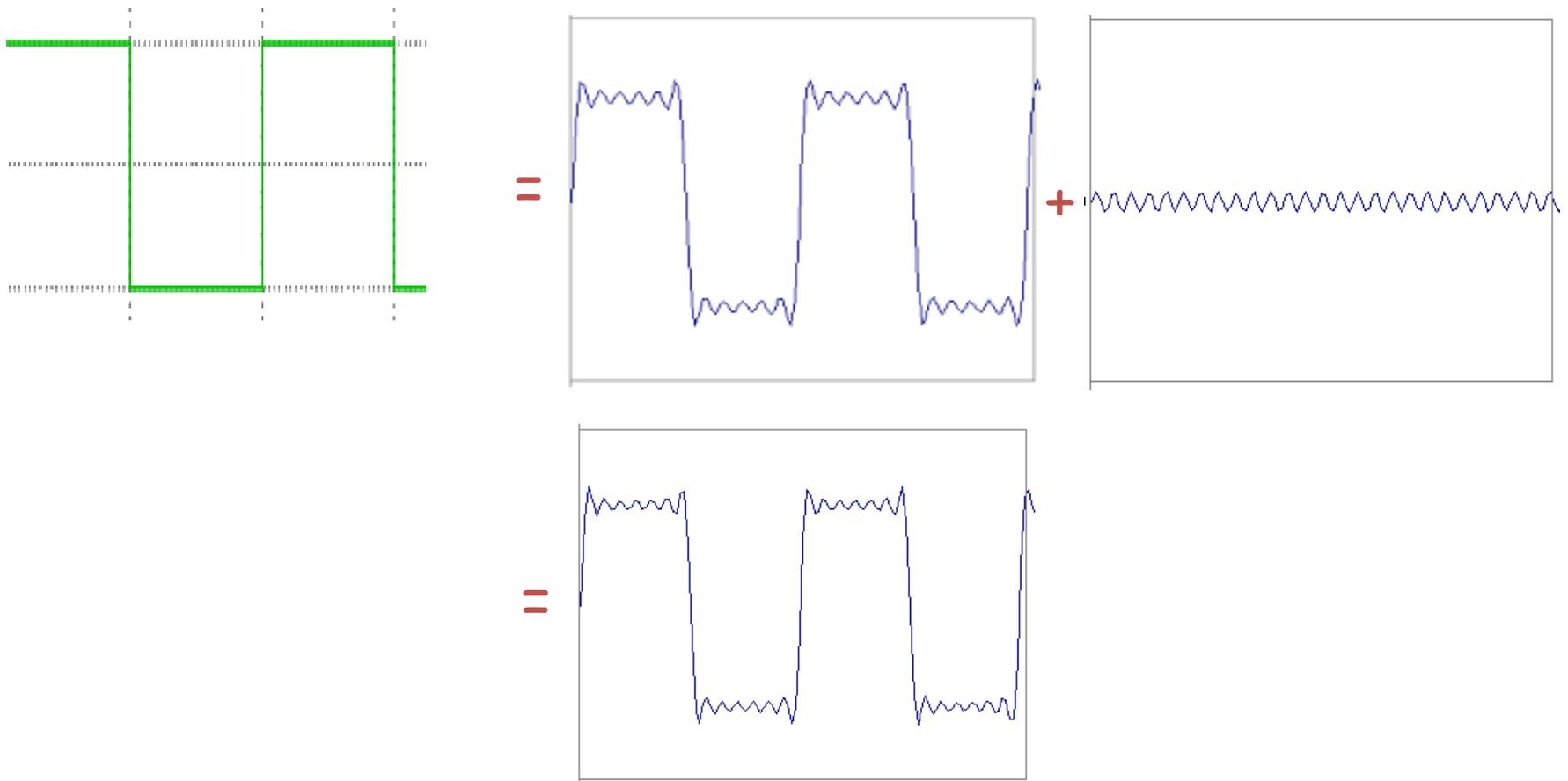
# Frequency Spectra



# Frequency Spectra

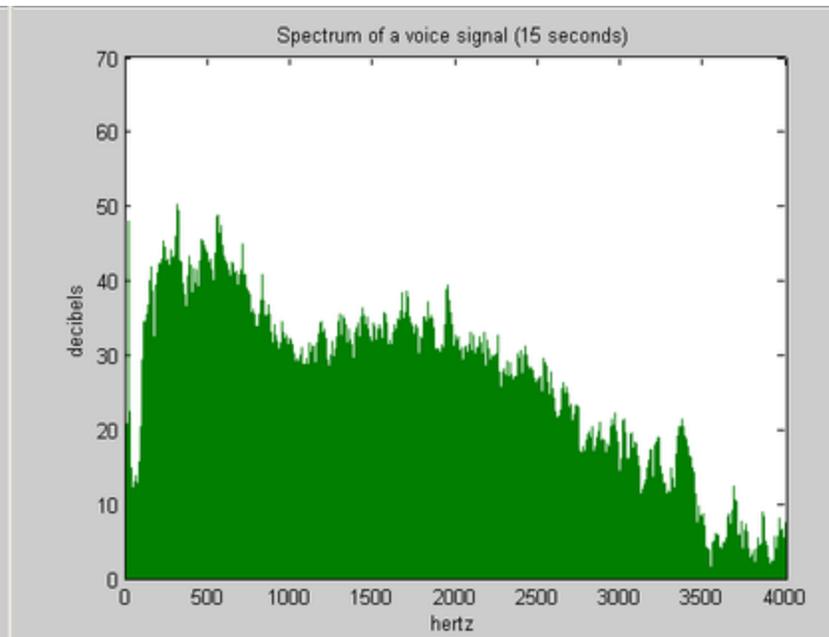
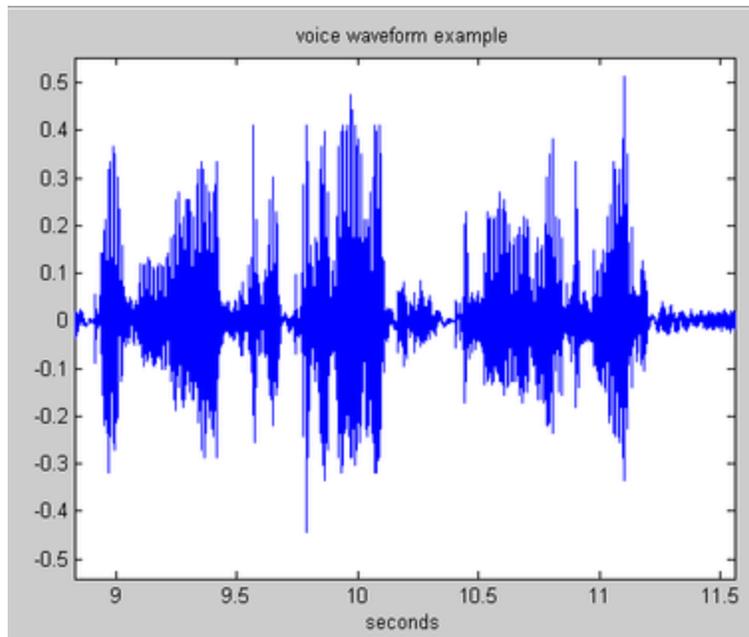


# Frequency Spectra



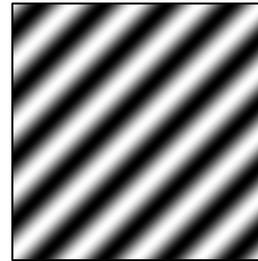
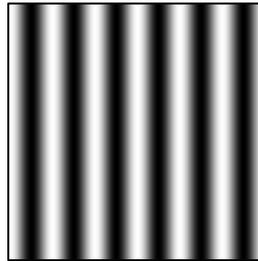
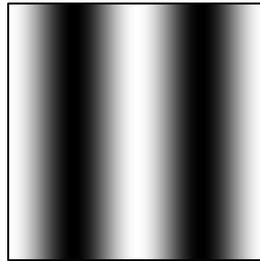
# Example: Music

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

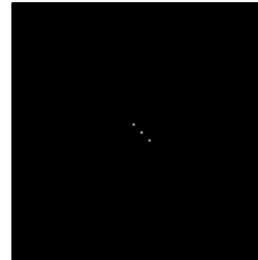
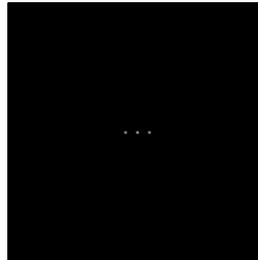
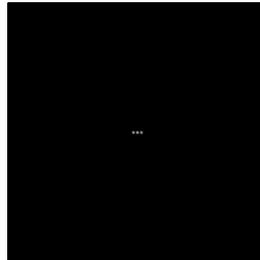


# Fourier analysis in images

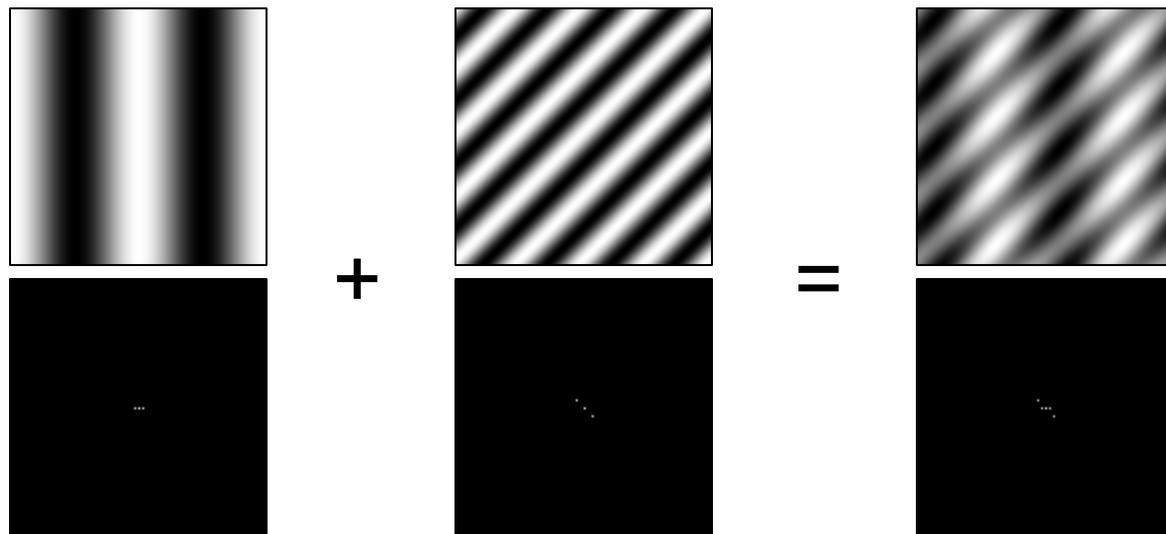
Intensity Image



Fourier Image



# Signals can be composed



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

# 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:  $\phi = \tan^{-1} \frac{I(\omega)}{R(\omega)}$

# Computing the Fourier Transform

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

Continuous

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

Discrete

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

Fast Fourier Transform (FFT):  $N \log N$

# The Convolution Theorem

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

$$F[g * h] = F[g]F[h]$$

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

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

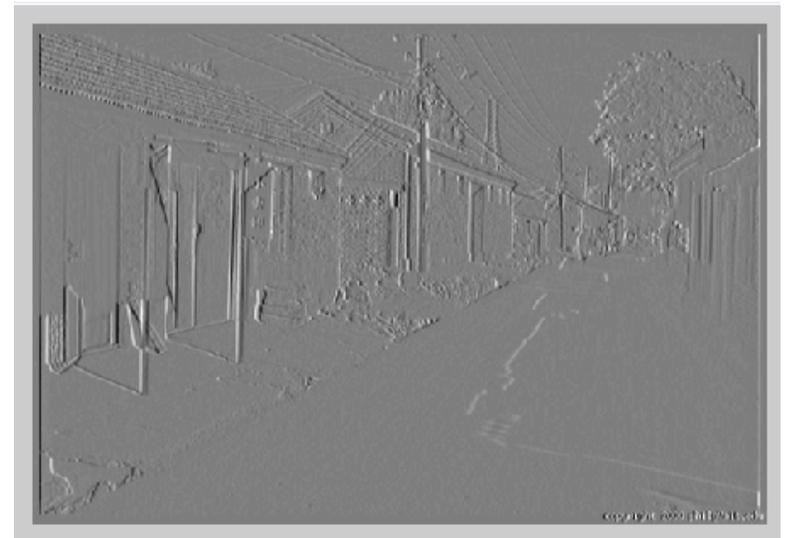
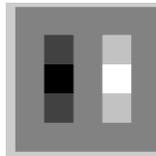
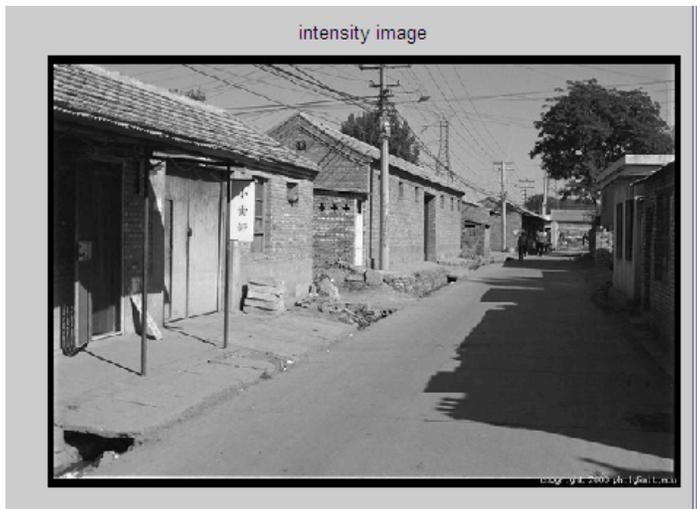
- Convolution** in spatial domain is equivalent to **multiplication** in frequency domain!

# Properties of Fourier Transforms

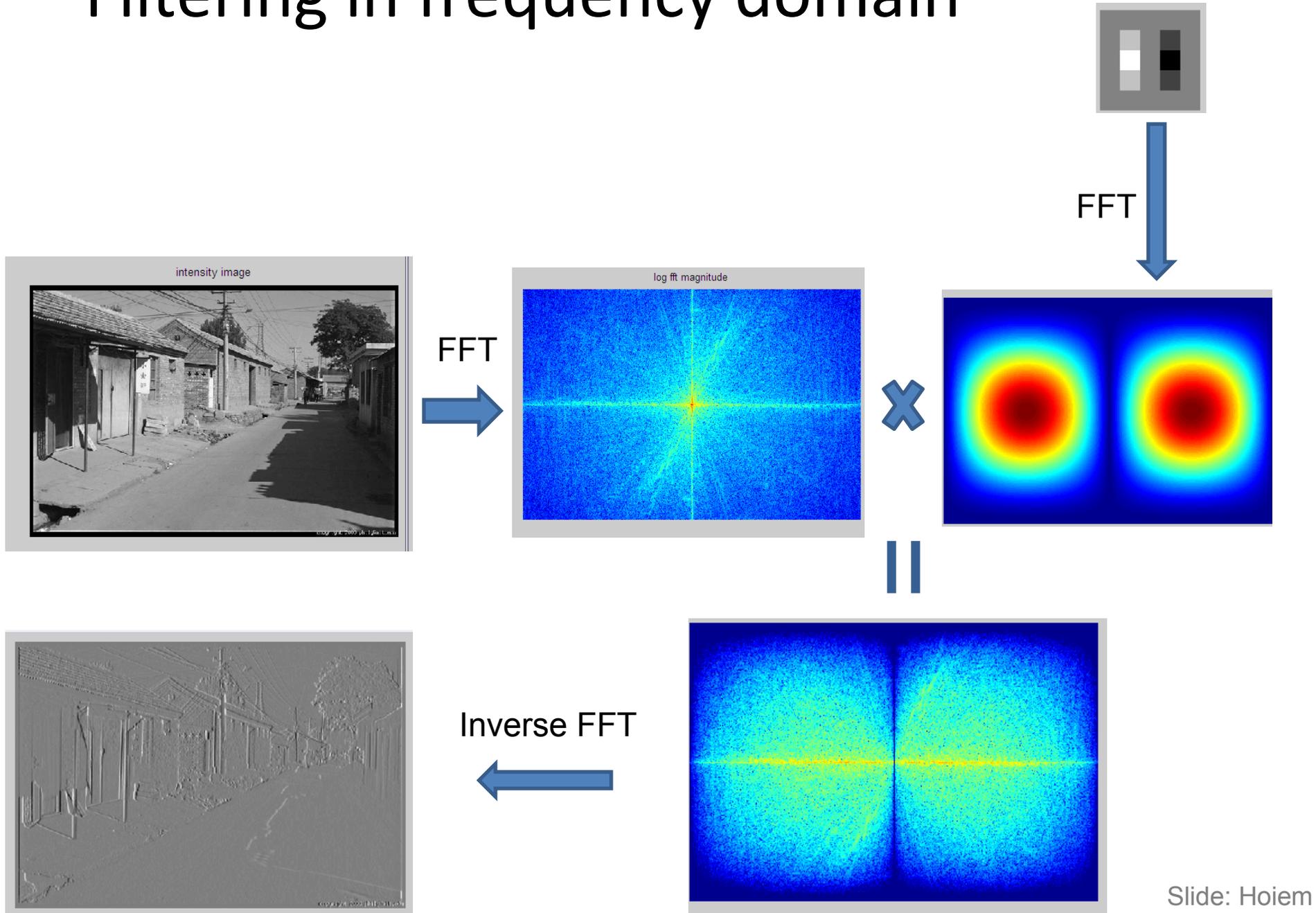
- Linearity  $\mathcal{F}[ax(t) + by(t)] = a\mathcal{F}[x(t)] + b\mathcal{F}[y(t)]$
- Fourier transform of a real signal is symmetric about the origin
- The energy of the signal is the same as the energy of its Fourier transform

# Filtering in spatial domain

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



# Filtering in frequency domain



# Questions

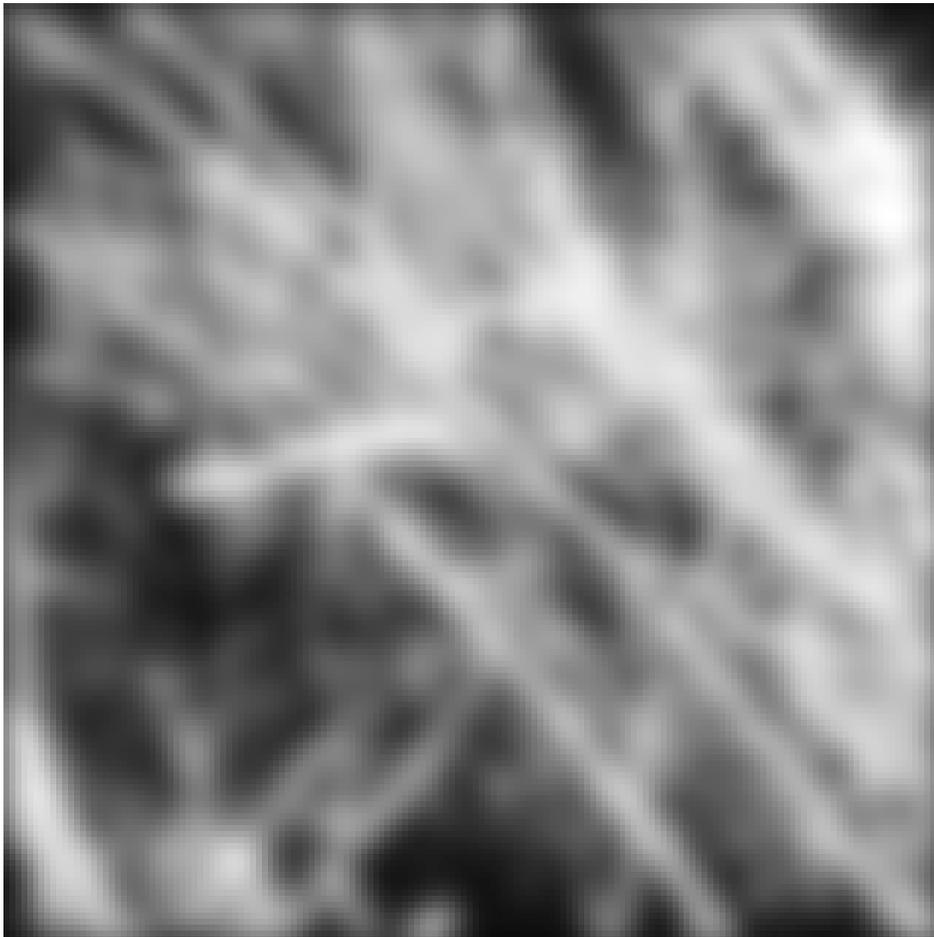
Which has more information, the phase or the magnitude?

What happens if you take the phase from one image and combine it with the magnitude from another image?

# Filtering

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

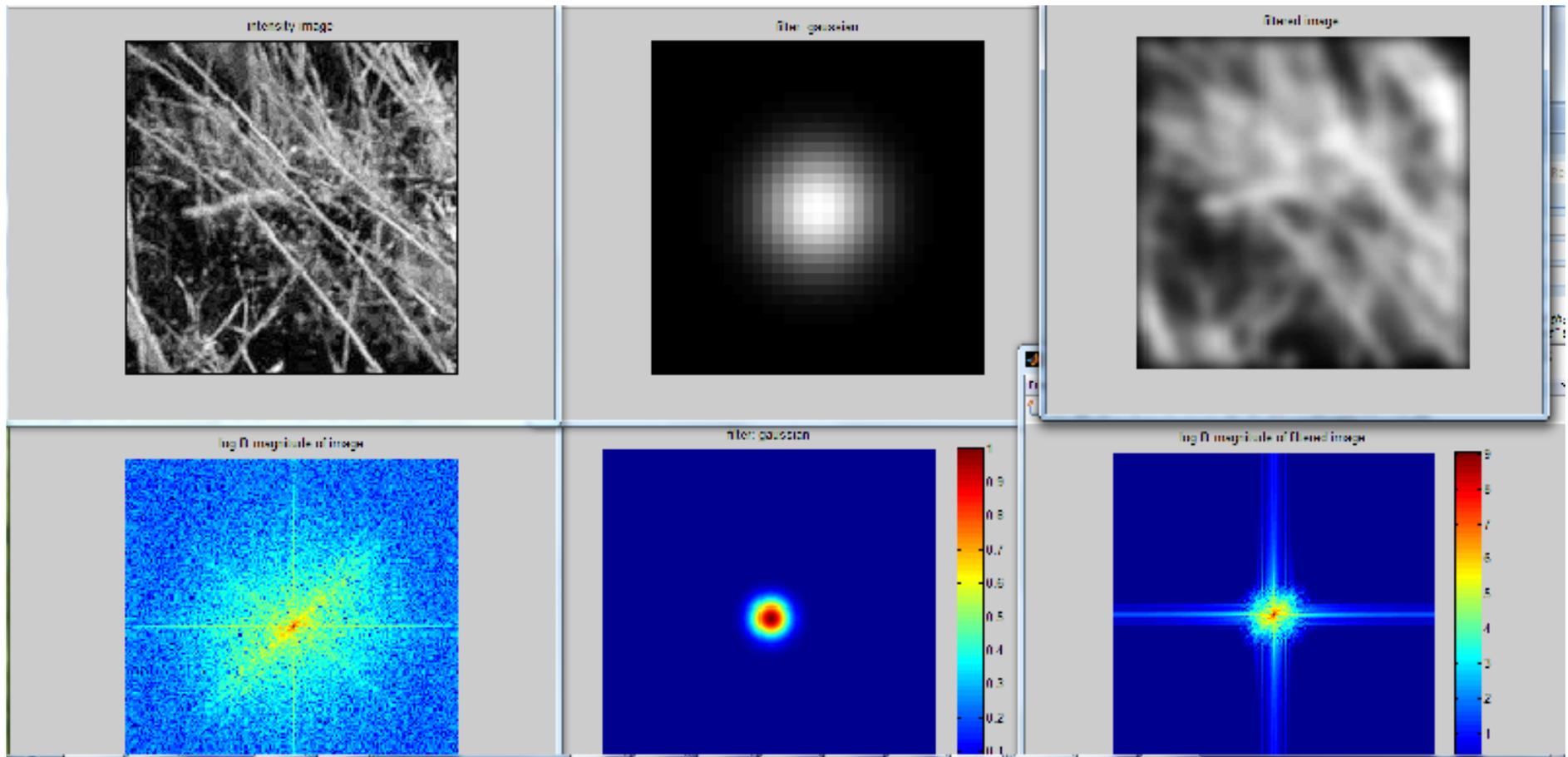
Gaussian



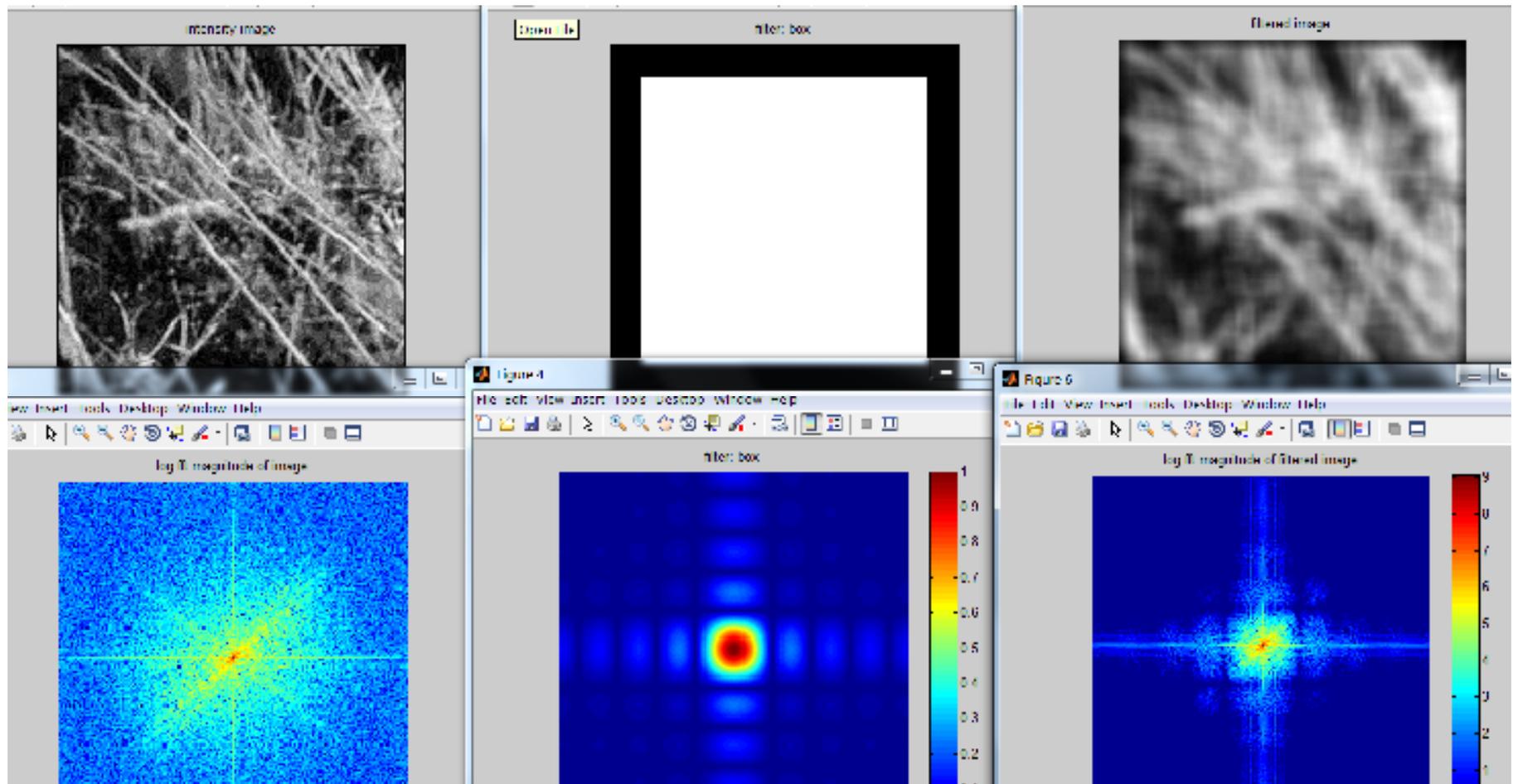
Box filter



# Gaussian



# Box Filter



# Sampling

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

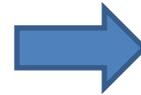
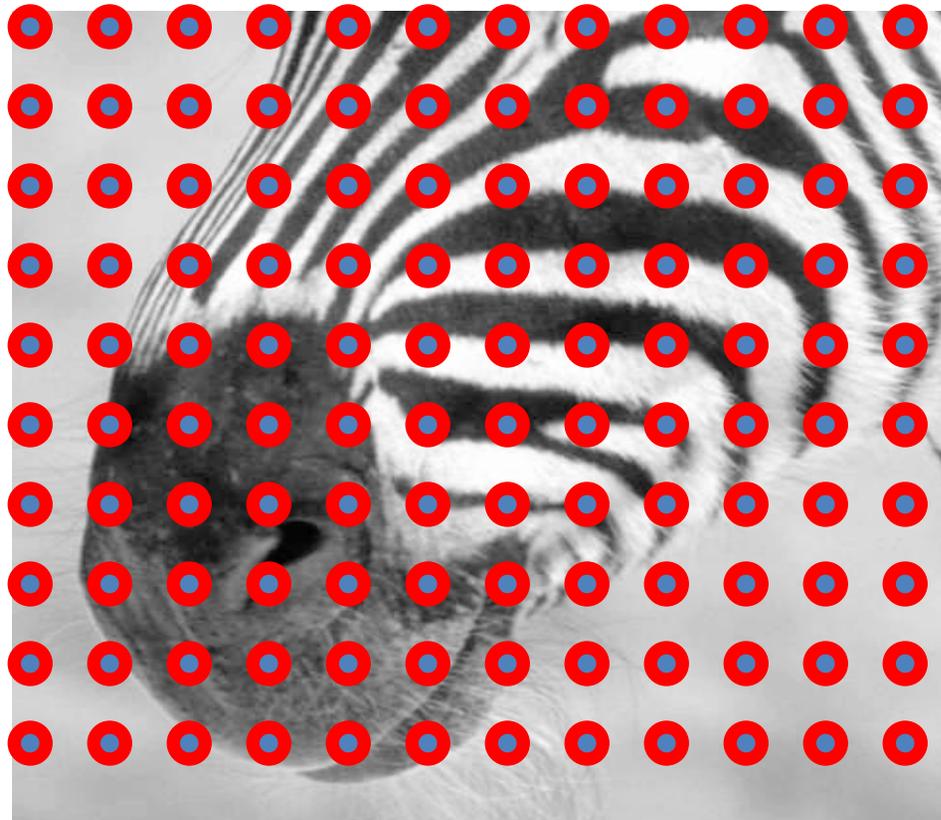


Image: <http://www.flickr.com/photos/igorms/136916757/>

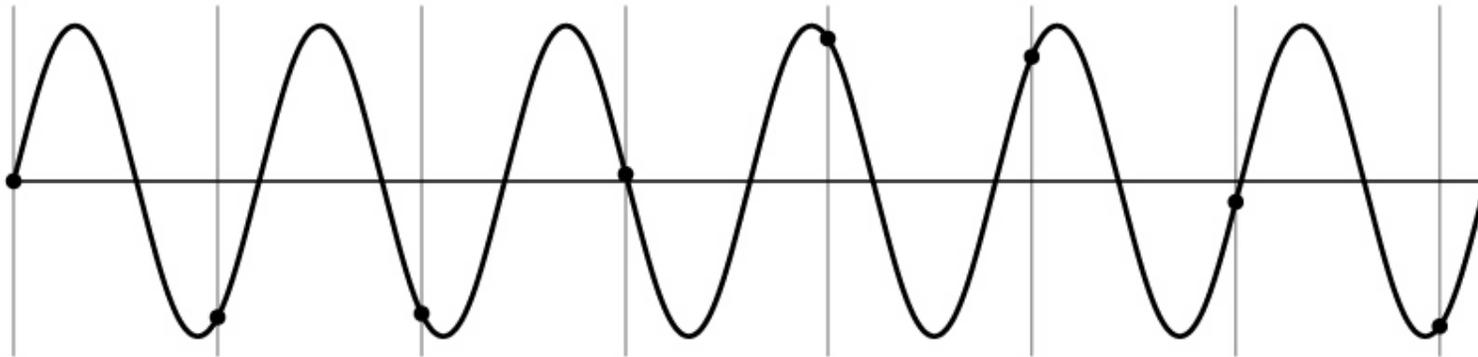
# Subsampling by a factor of 2



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

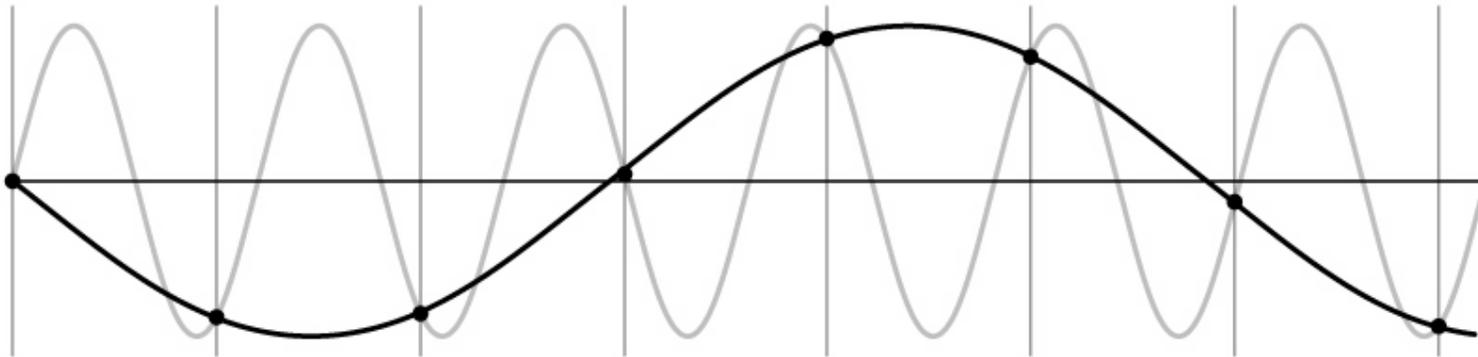
# Aliasing problem

- 1D example (sinewave):



# Aliasing problem

- 1D example (sinewave):



# Aliasing problem

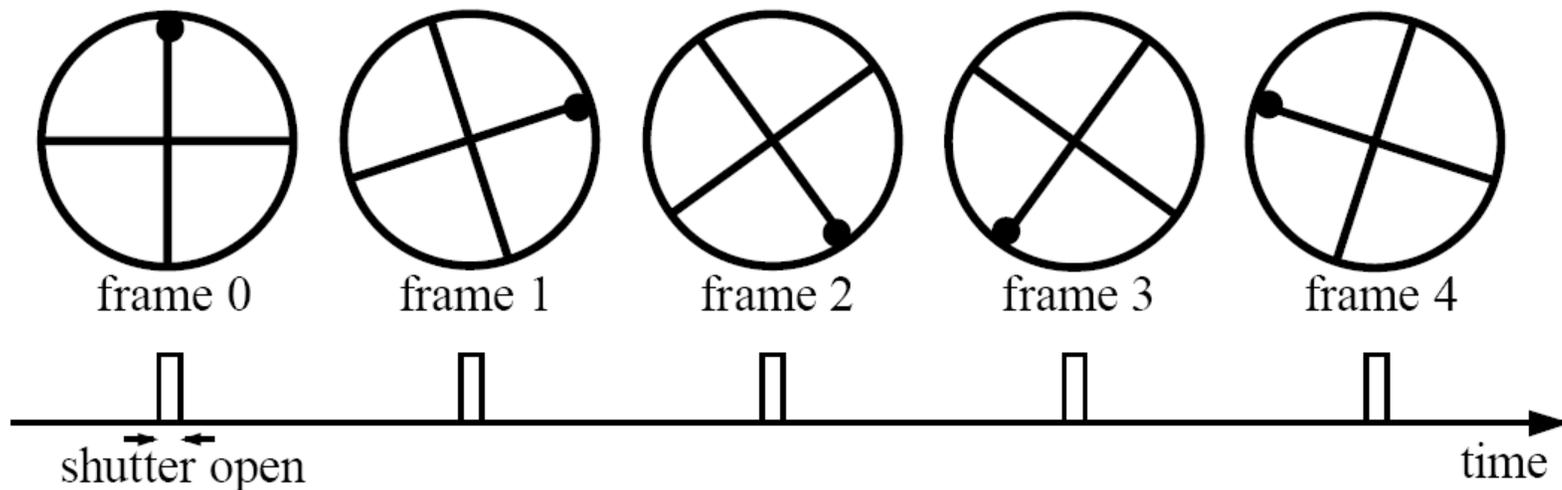
- Sub-sampling may be dangerous....
- Characteristic errors may appear:
  - “Wagon wheels rolling the wrong way in movies”
  - “Checkerboards disintegrate in ray tracing”
  - “Striped shirts look funny on color television”

# 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



# Sampling and aliasing

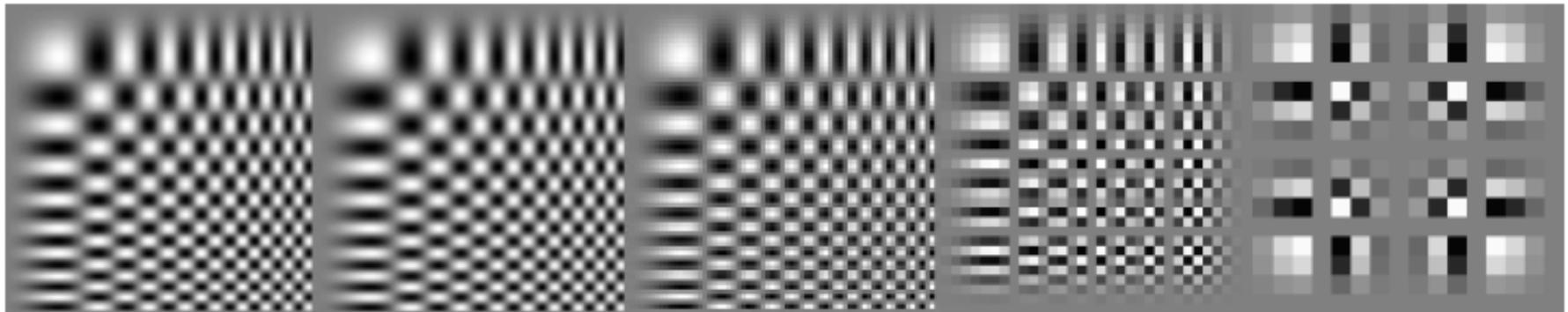
256x256

128x128

64x64

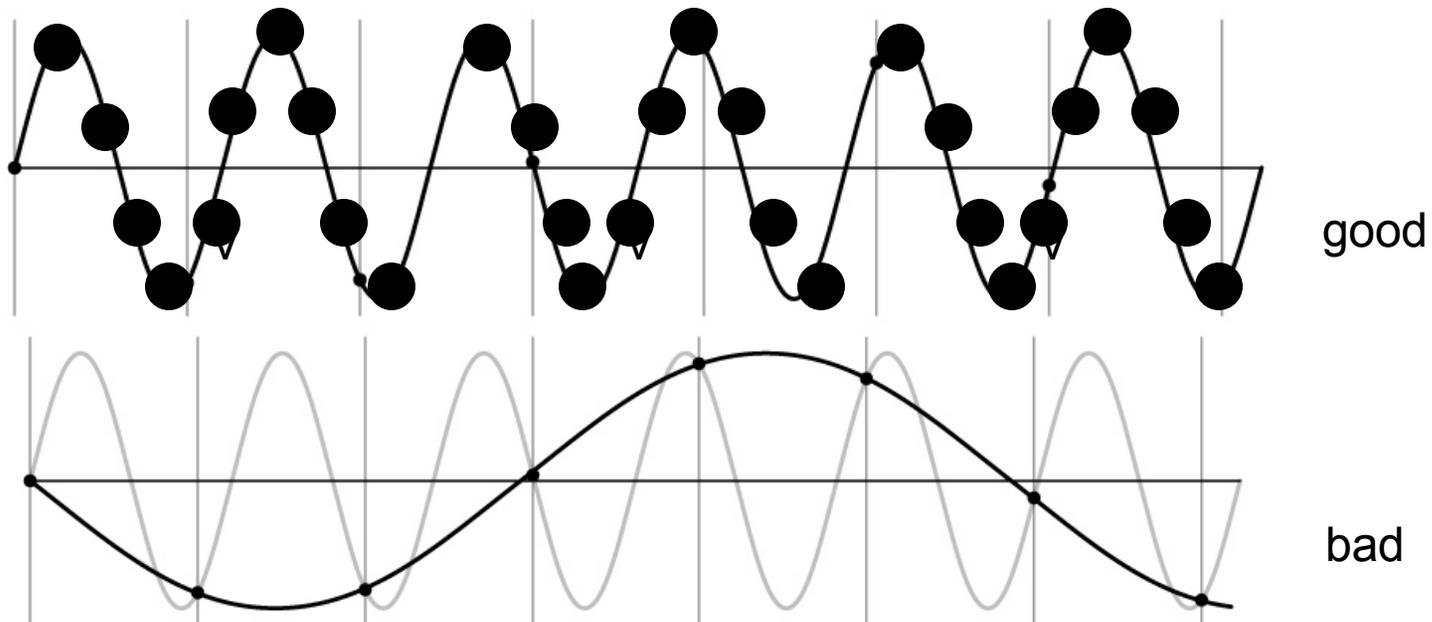
32x32

16x16



# 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 allow to reconstruct the original perfectly from the sampled version



# 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

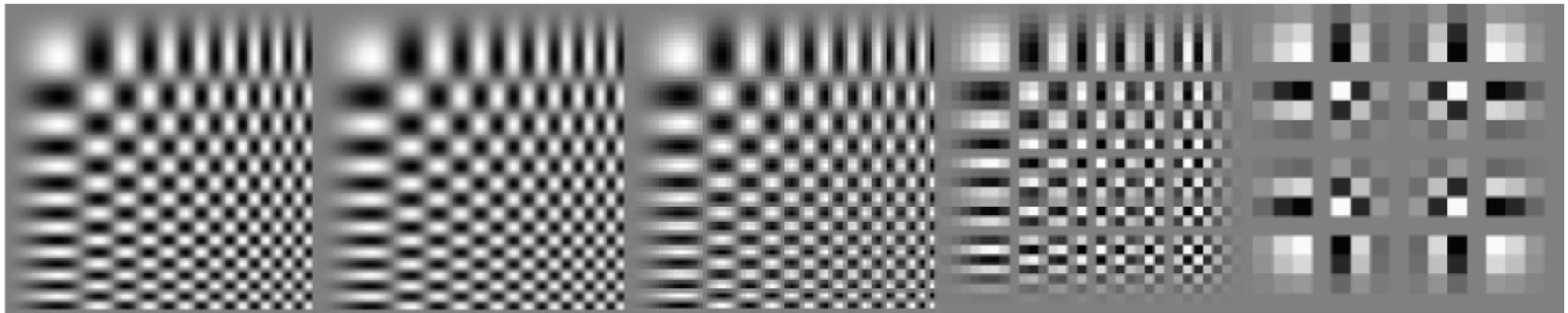
256x256

128x128

64x64

32x32

16x16



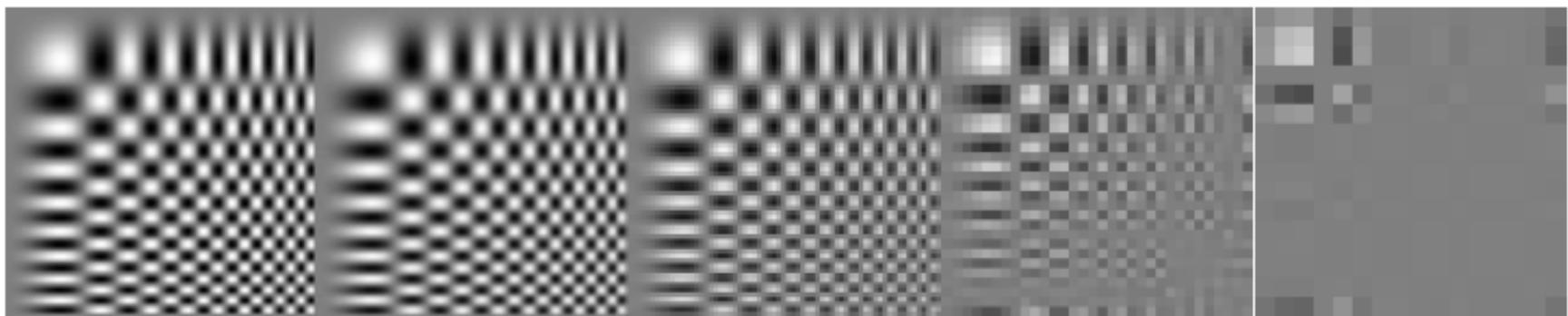
256x256

128x128

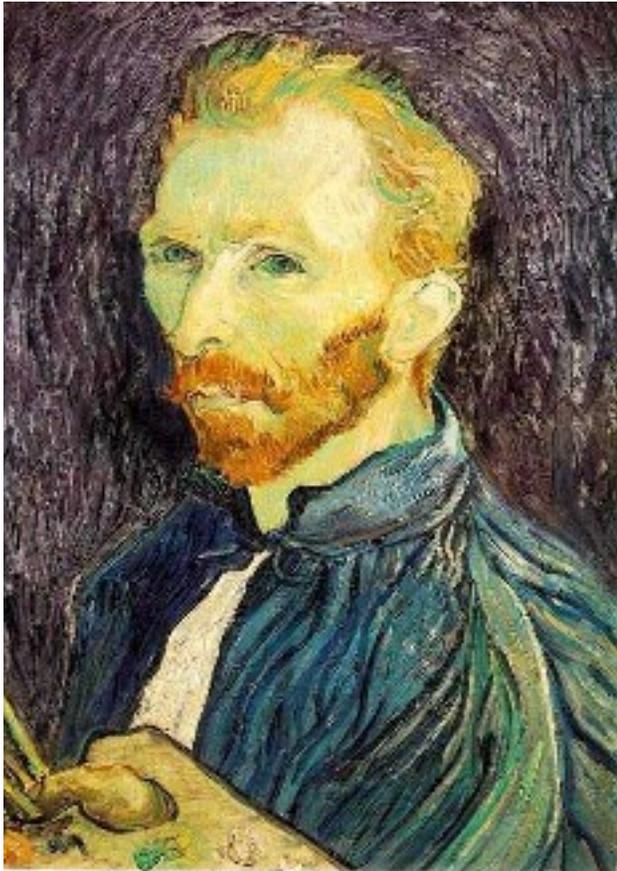
64x64

32x32

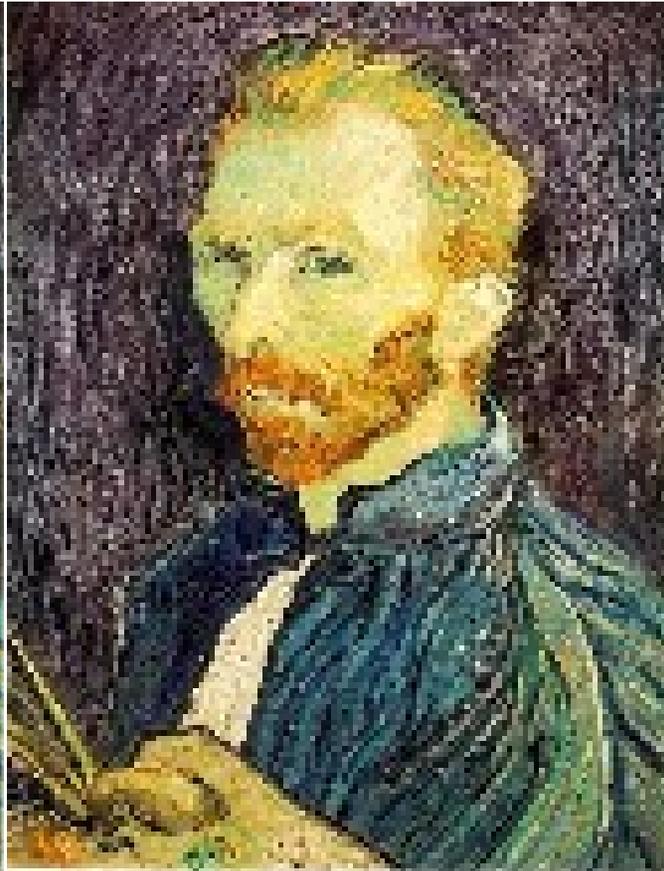
16x16



# Subsampling without pre-filtering



1/2

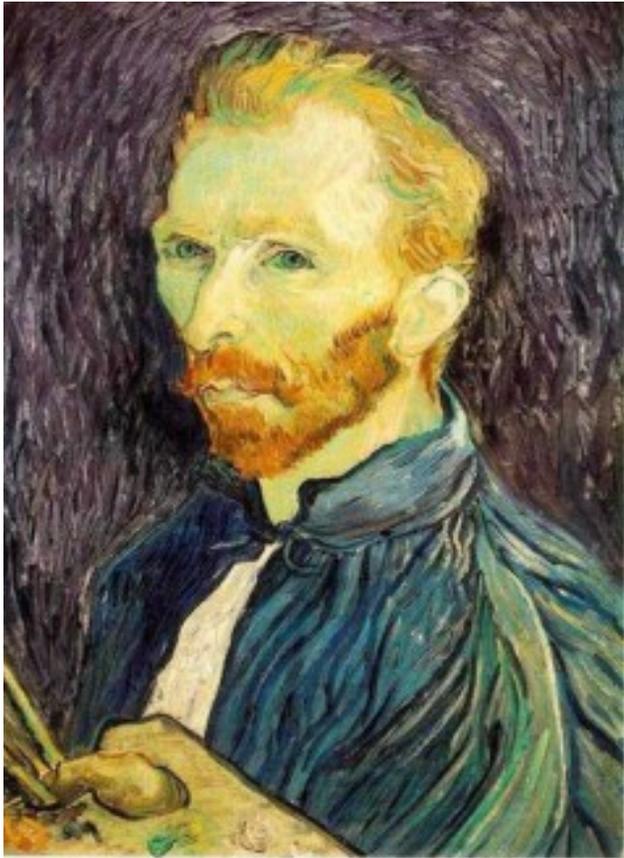


1/4 (2x zoom)

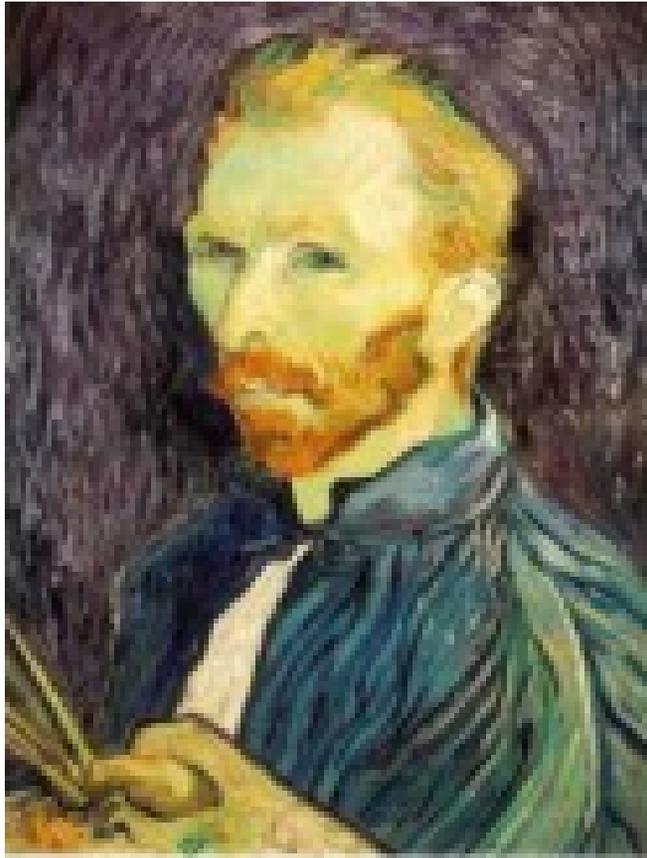


1/8 (4x zoom)

# Subsampling with Gaussian pre-filtering



Gaussian 1/2



G 1/4



G 1/8

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

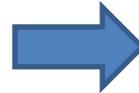
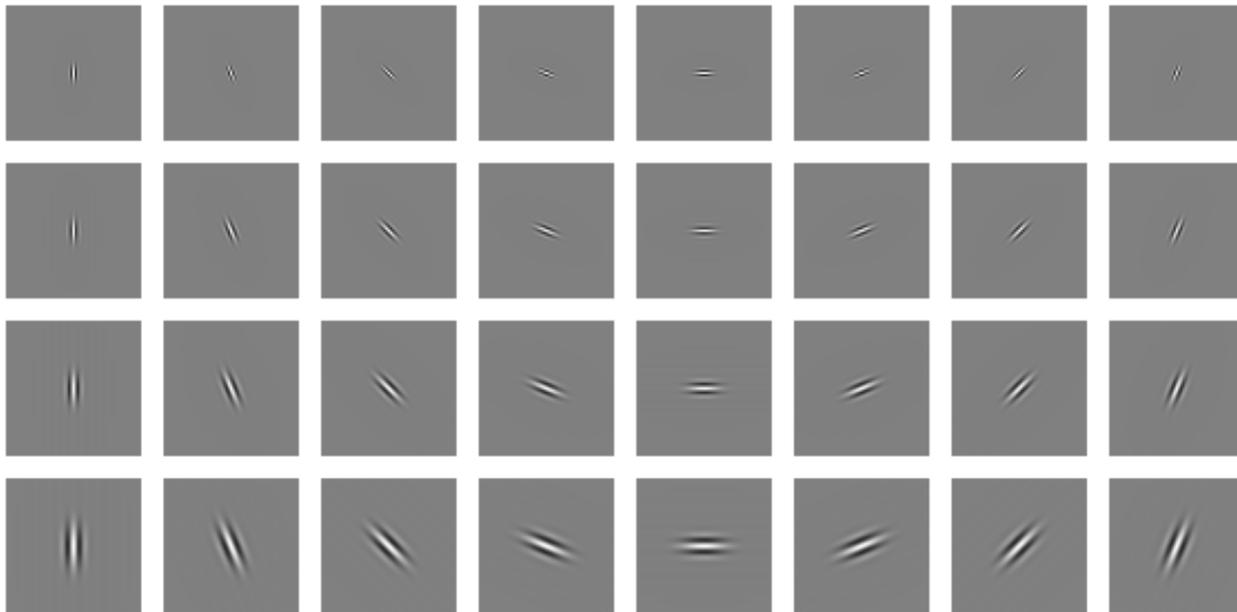


Image: <http://www.flickr.com/photos/igorms/136916757/>

# Clues from Human Perception

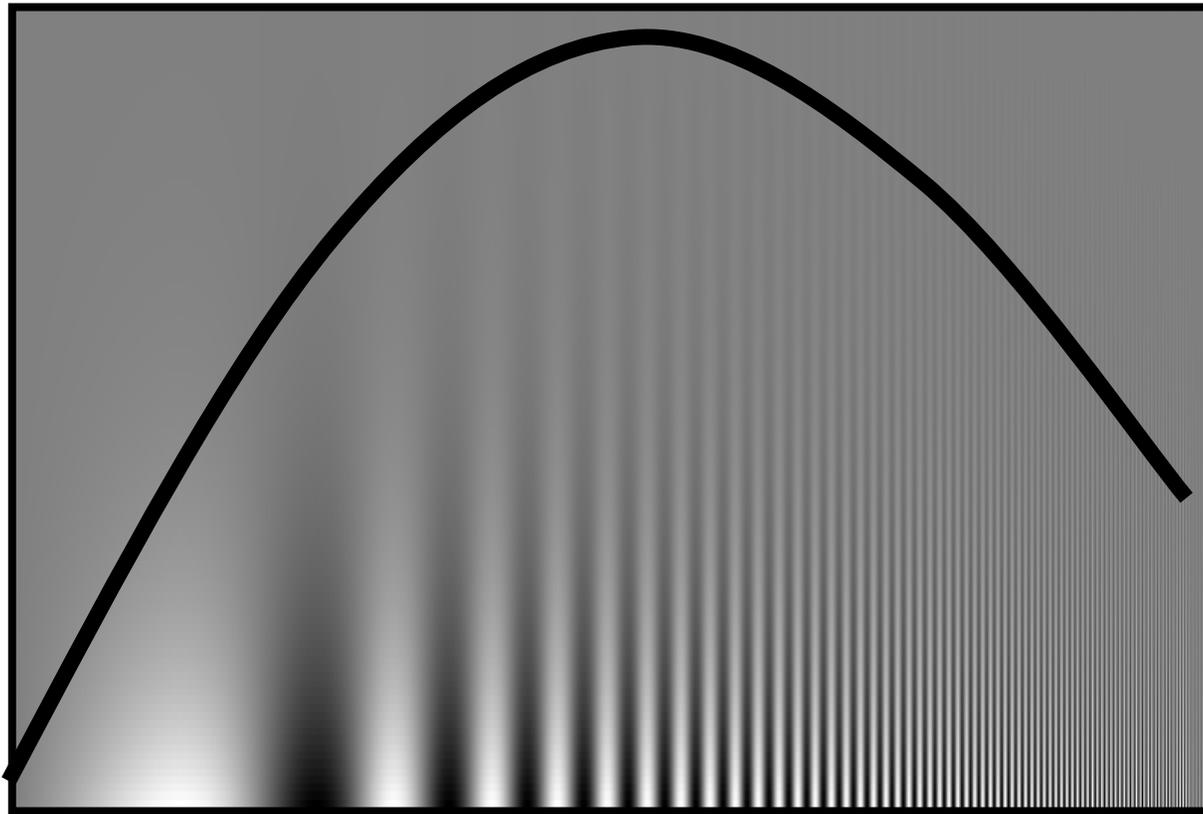
- Early processing in humans filters for various orientations and scales of frequency
- Perceptual cues in the mid-high frequencies dominate perception
- When we see an image from far away, we are effectively subsampling it



Early Visual Processing: Multi-scale edge and blob filters

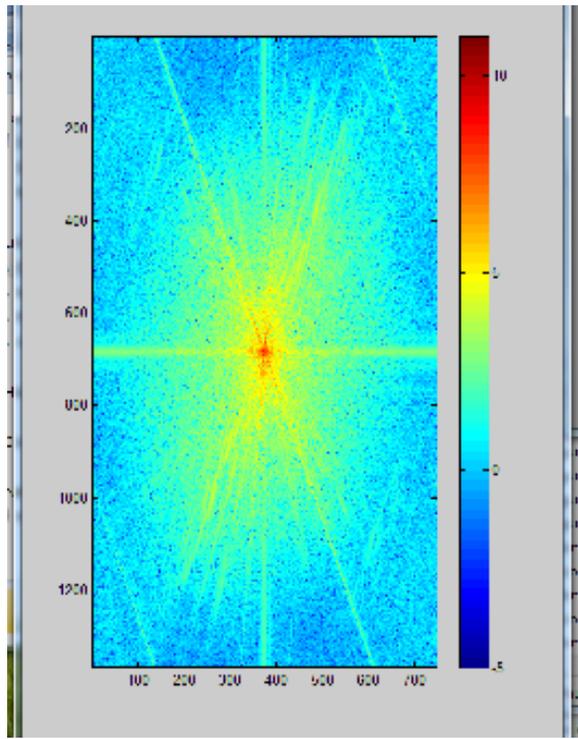
# Campbell-Robson contrast sensitivity curve

---

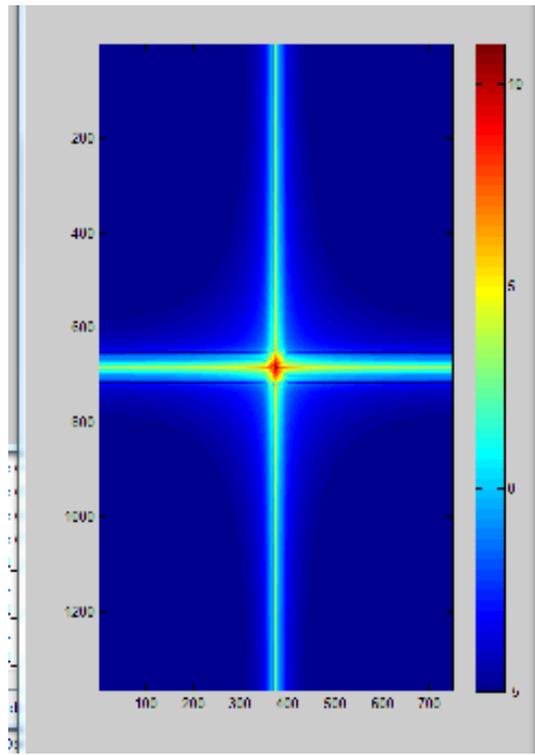


# Hybrid Image in FFT

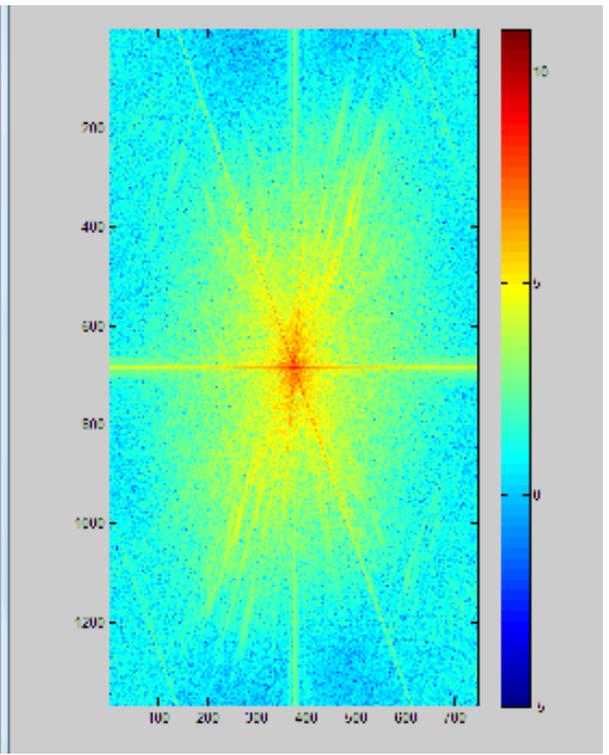
Hybrid Image



Low-passed Image



High-passed Image



# Perception

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

