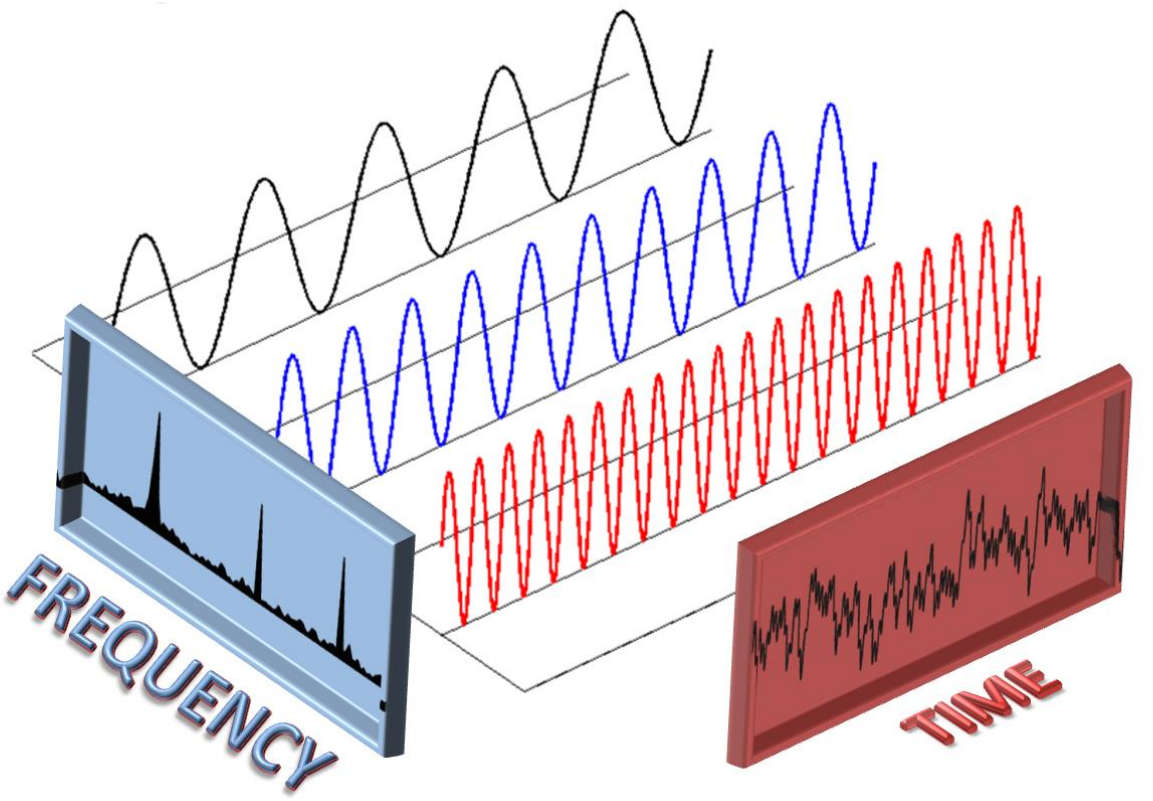
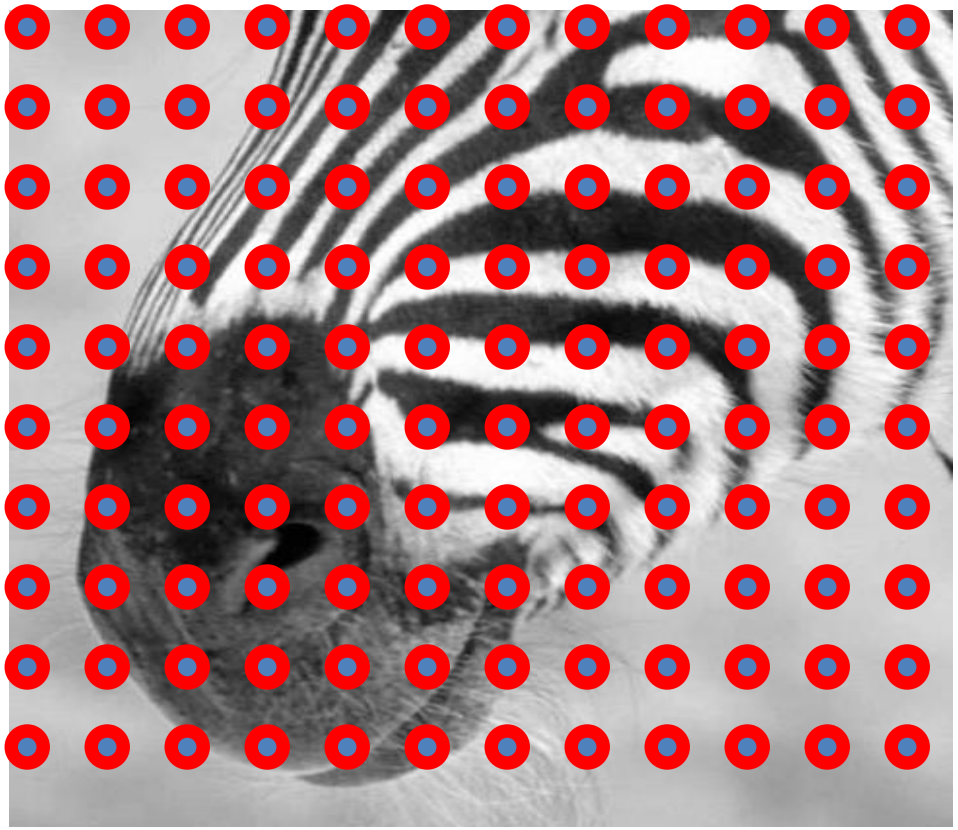
Thinking in frequencies

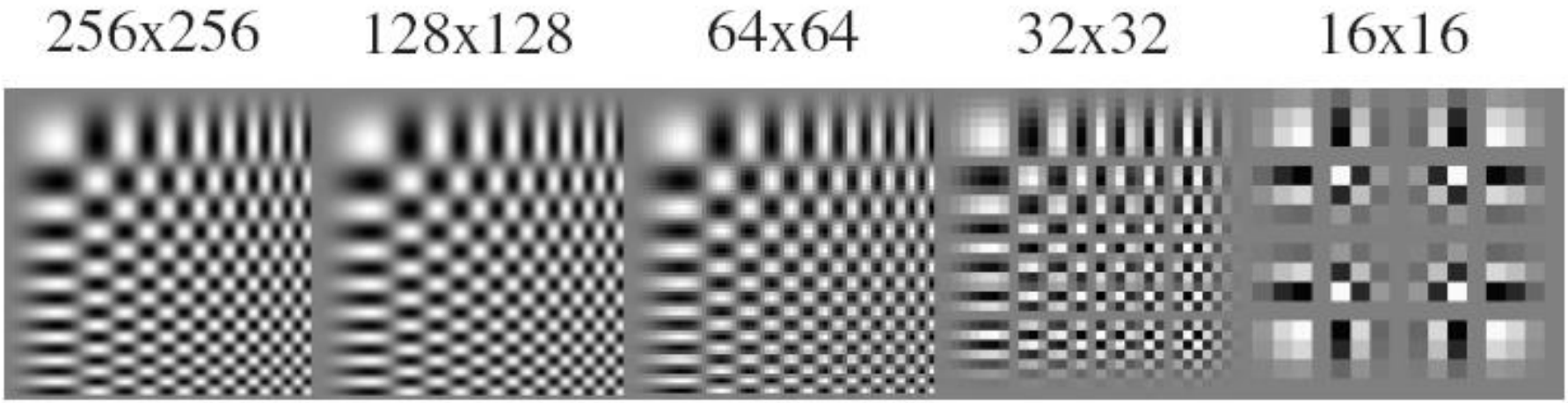


Subsampling by a factor of 2



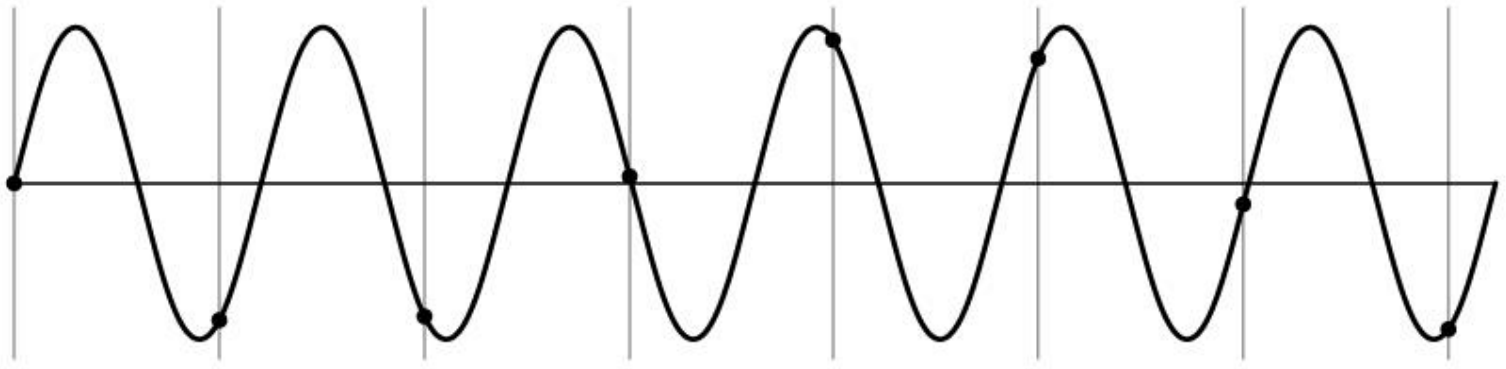
Throw away every other row and column to create a 1/2 size image

Sampling and aliasing



Aliasing problem

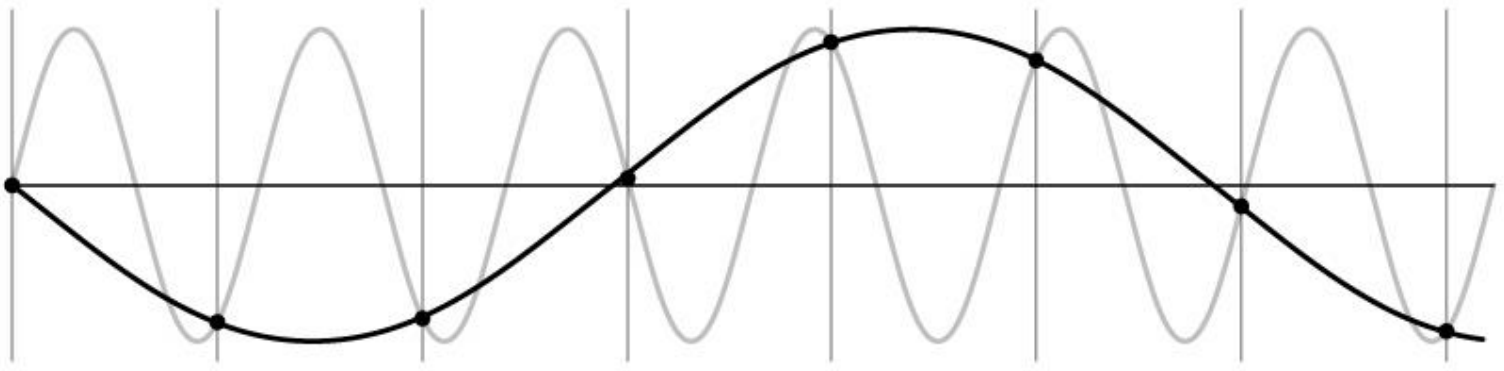
- 1D example (sinewave):



Source: S. Marschner

Aliasing problem

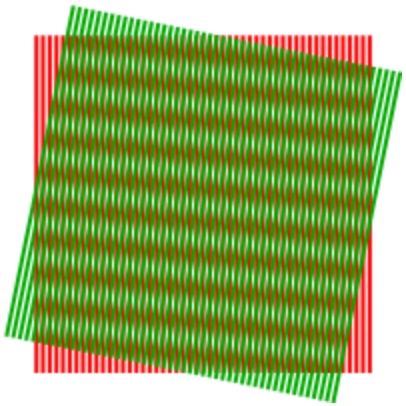
- 1D example (sinewave):



Source: S. Marschner

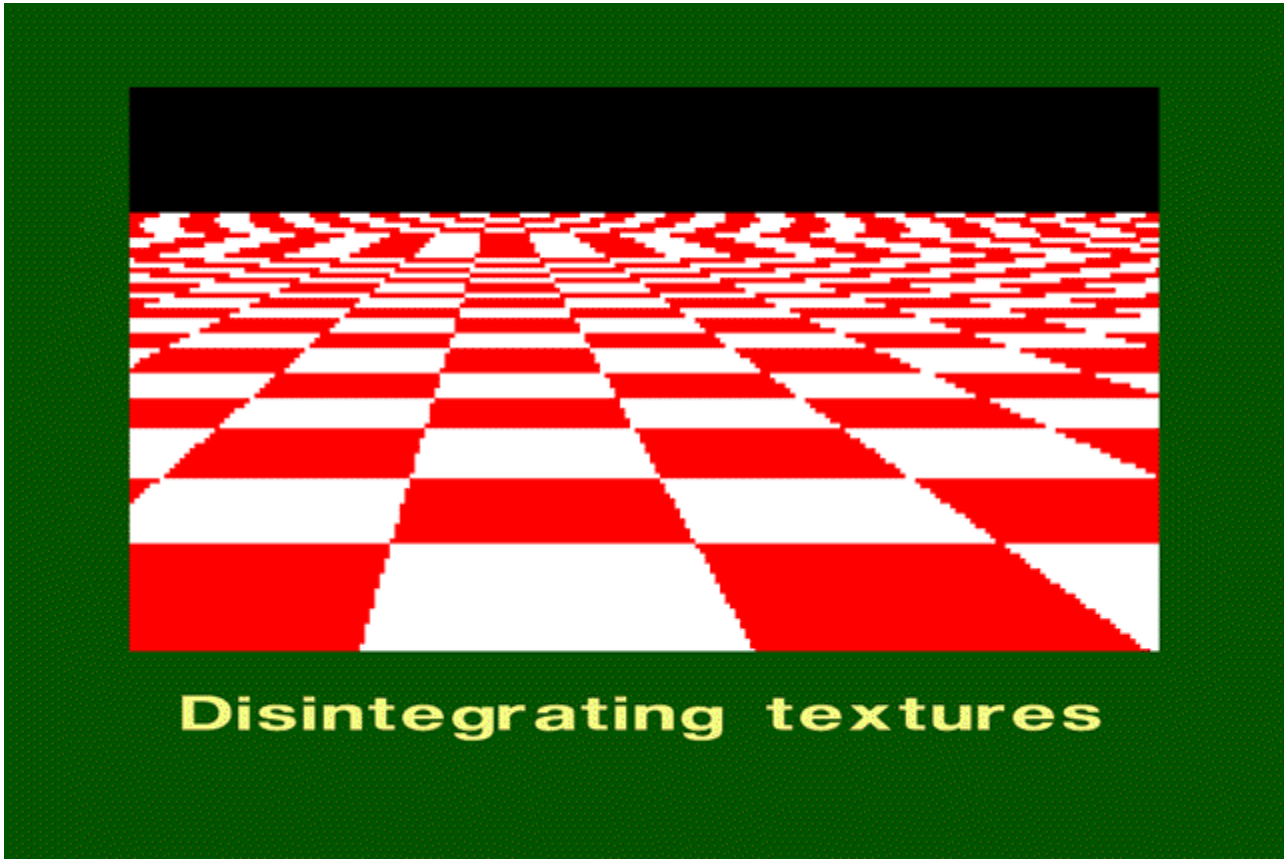
Aliasing problem

- Sub-sampling may be dangerous….
- Characteristic errors may appear:

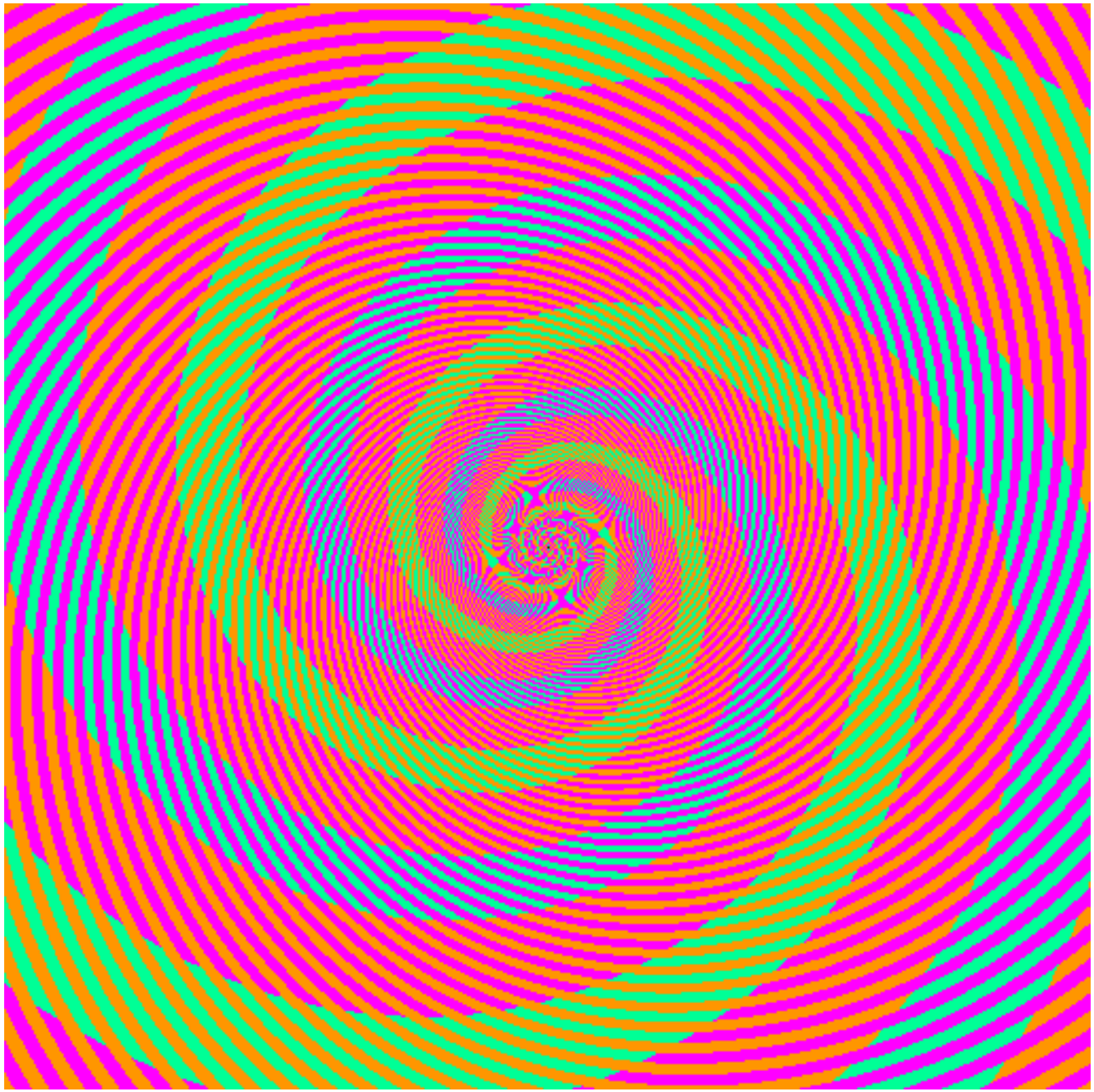


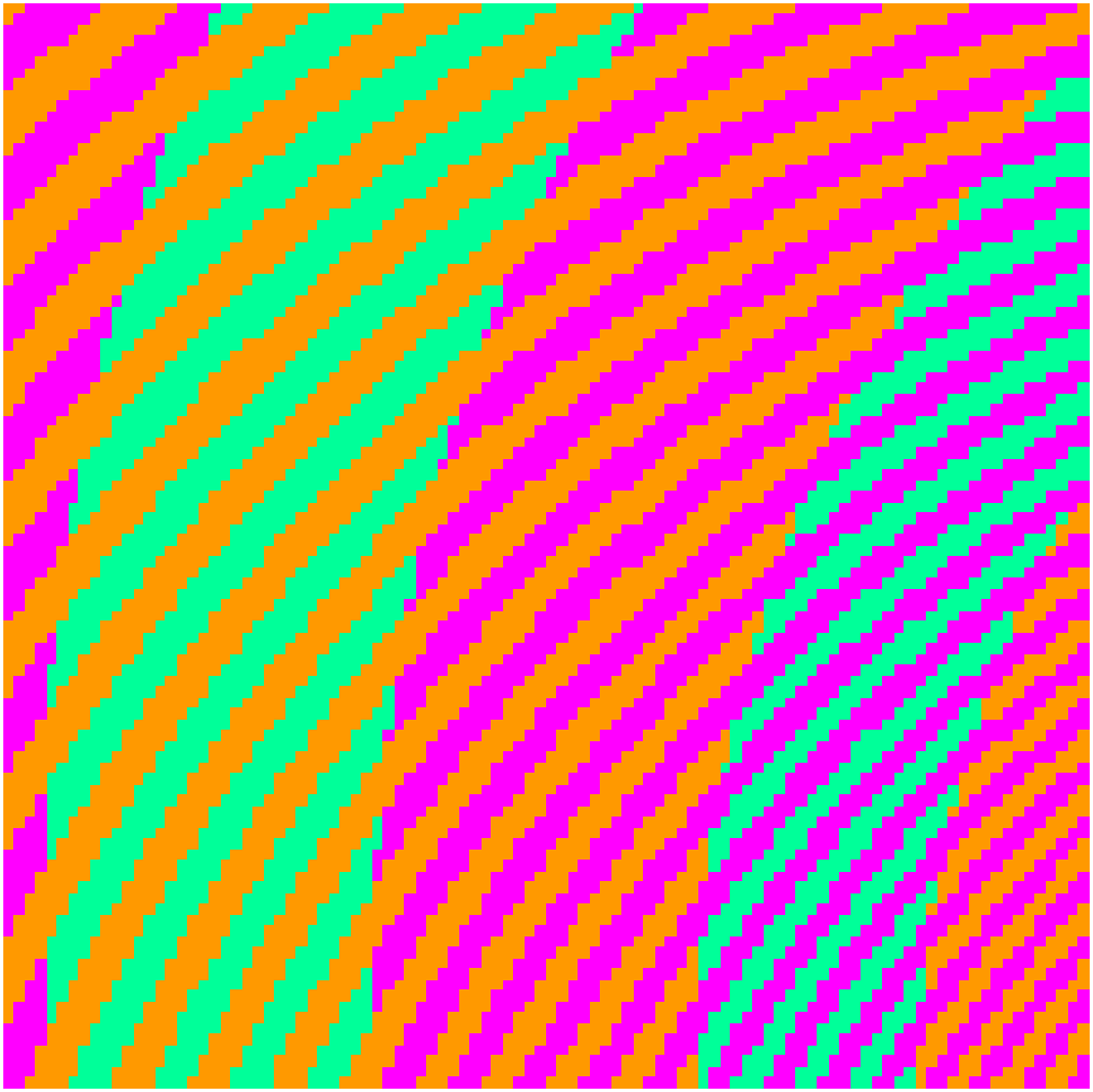
Source: D. Forsyth; Wikipedia

Aliasing in graphics



Source: A. Efros





The blue and green colors are actually the same

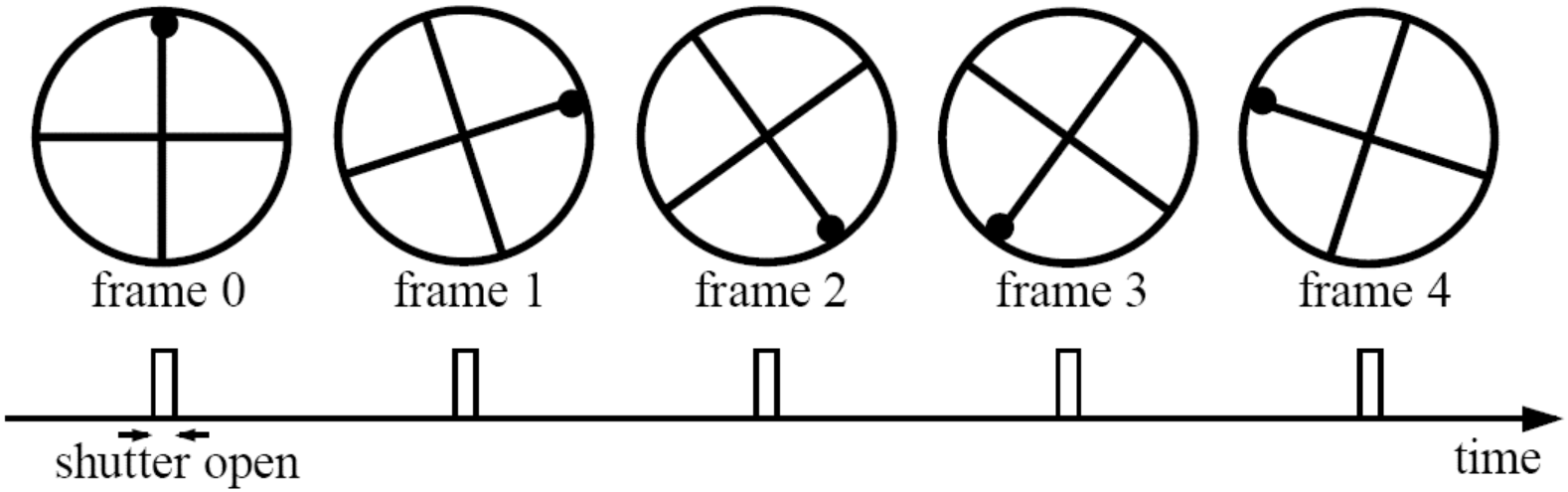
http://blogs.discovermagazine.com/badastronomy/2009/06/24/the-blue-and-the-green/

Videos

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):

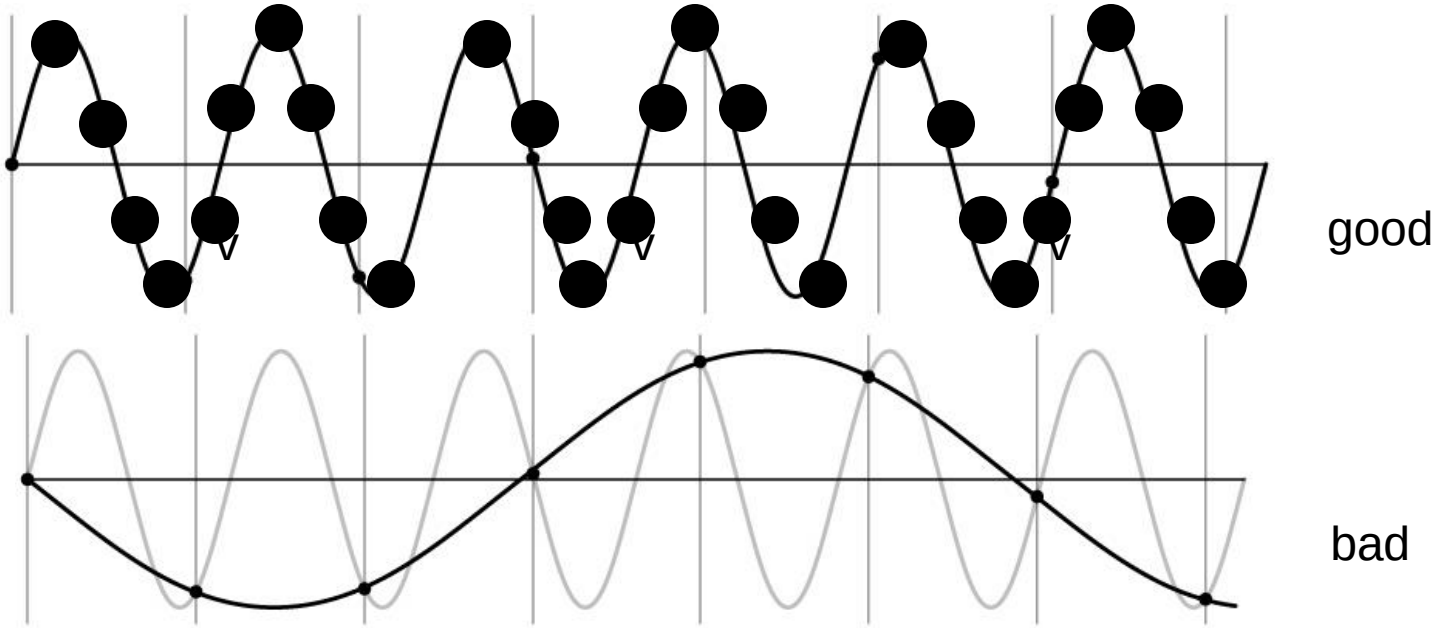


Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)

Slide by Steve Seitz

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



How to fix aliasing?

Solutions?

Better sensors

Solutions:

- Sample more often

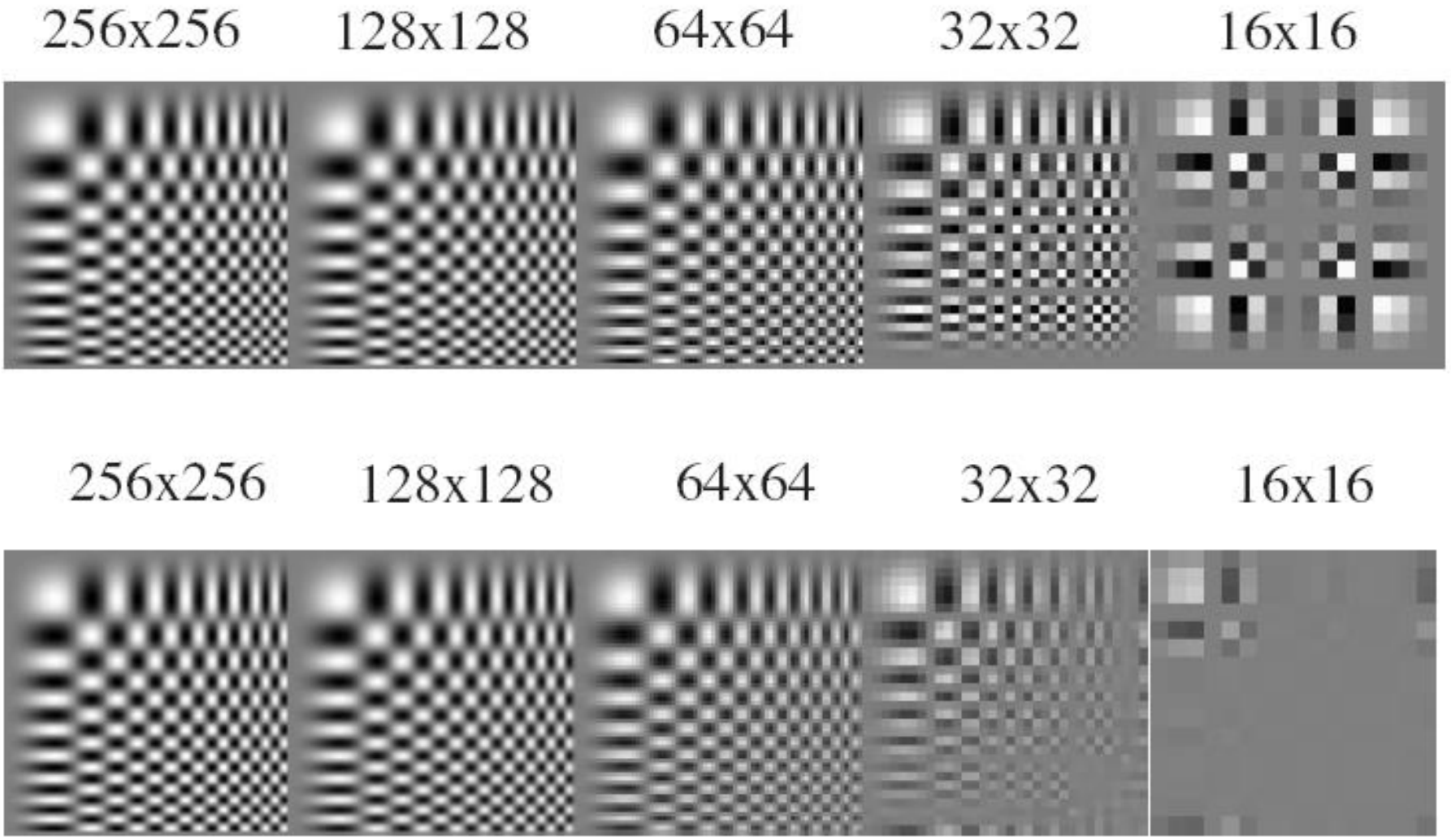
Anti-aliasing

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 (*low pass*) filter

Anti-aliasing



Forsyth and Ponce 2002

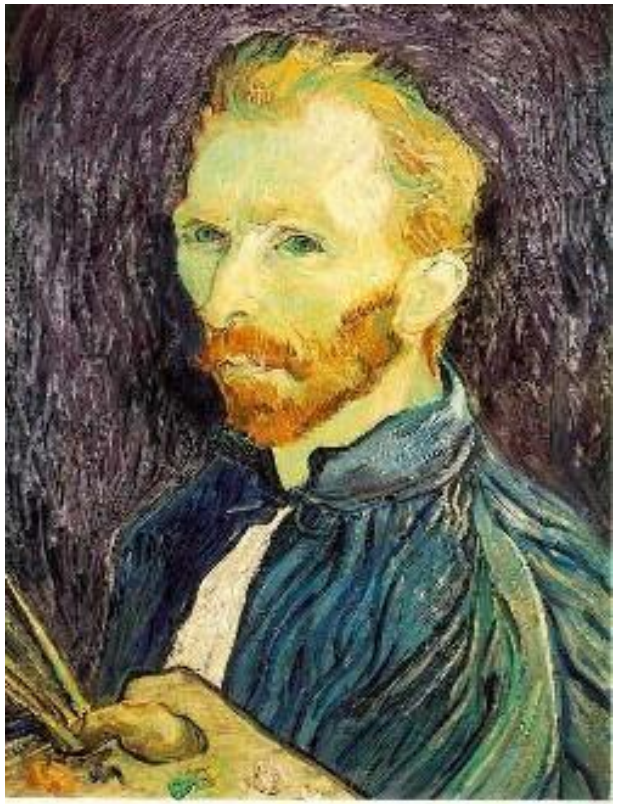
Algorithm for downsampling by factor of 2

1. Start with image(h, w)
2. Apply low-pass filter

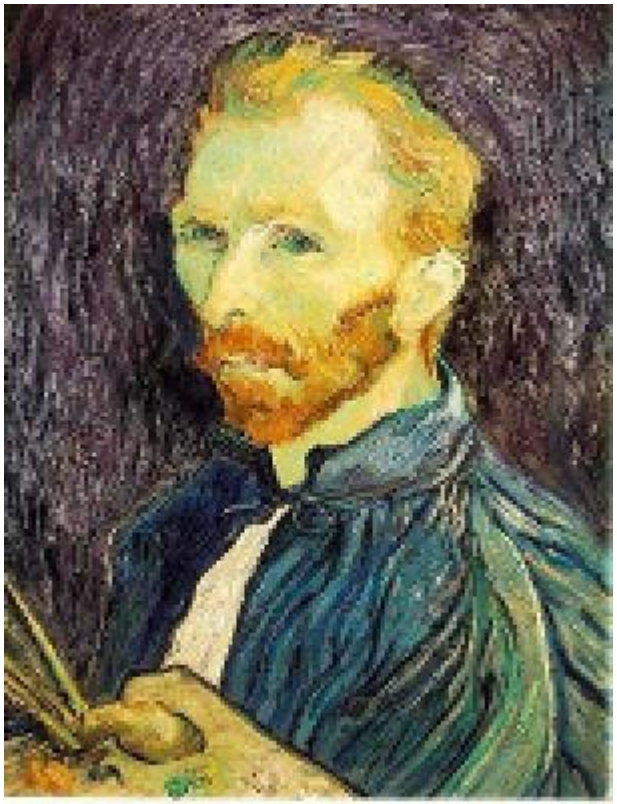
   im_blur = imfilter(image, fspecial('gaussian', 7, 1))
3. Sample every other pixel

   im_small = im_blur(1:2:end, 1:2:end);

Subsampling without pre-filtering



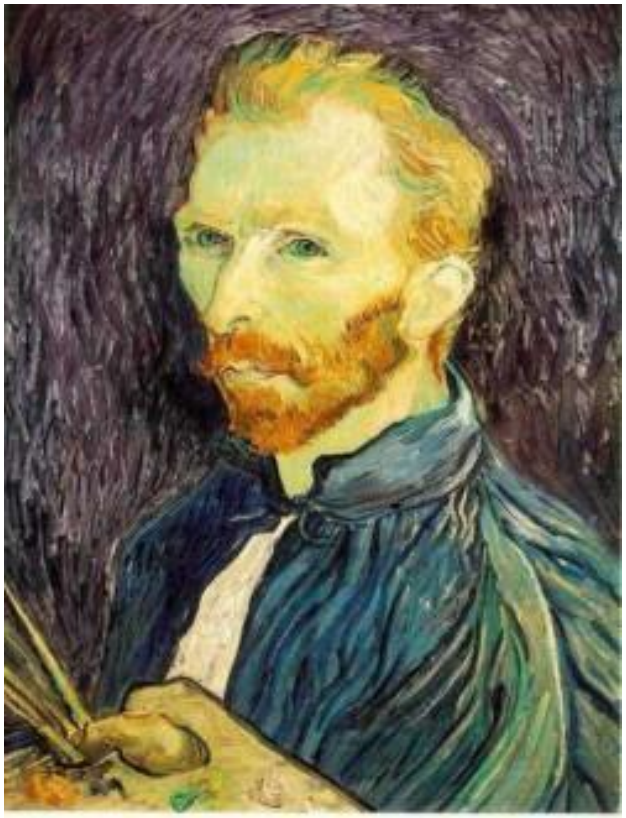
1/2



1/4 (2x zoom)

1/8 (4x zoom)

Subsampling with Gaussian pre-filtering



Gaussian 1/2

G 1/4

G 1/8

Steve Seitz

Another way of thinking about frequency

FOURIER SERIES & FOURIER TRANSFORMS

Fourier series

A bold idea (1807):

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

Jean Baptiste Joseph Fourier (1768-1830)

Fourier series

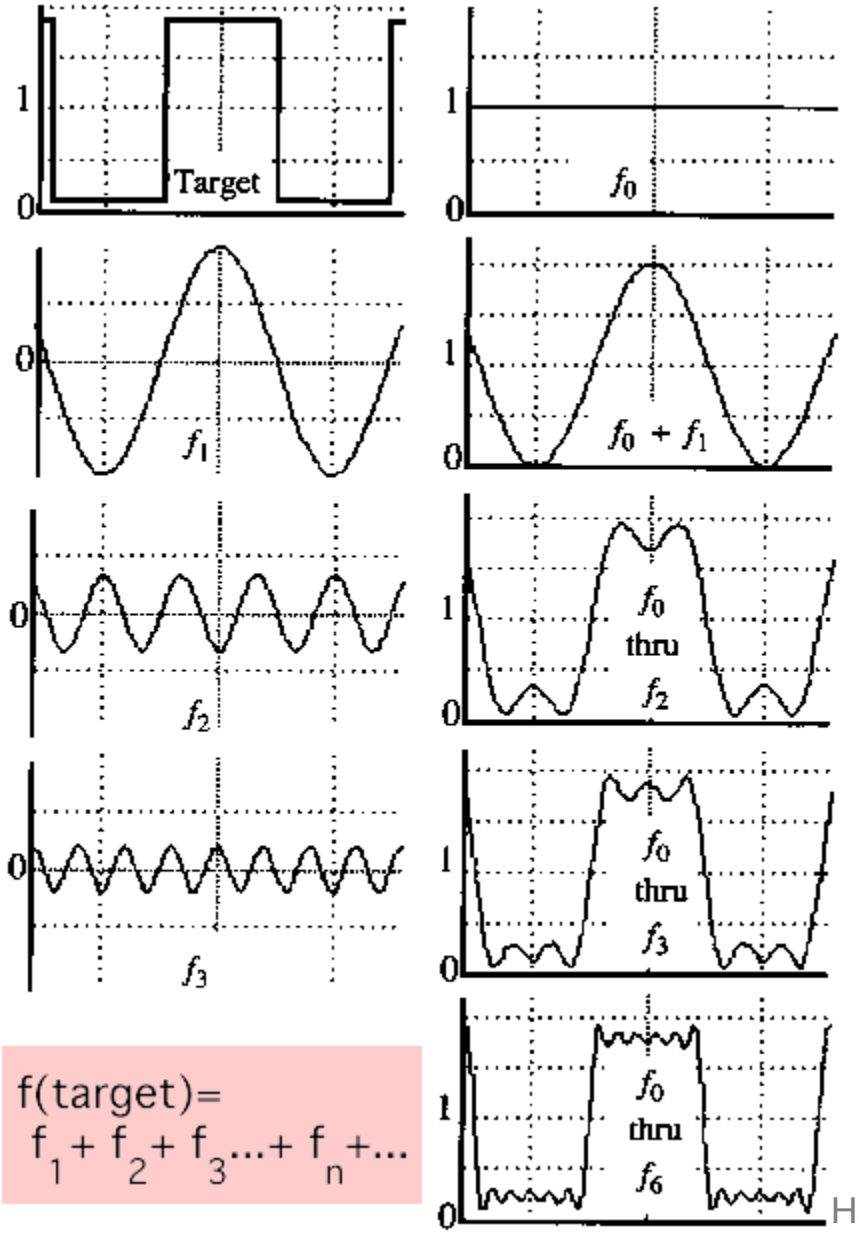
A bold idea (1807):

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

Our building block:

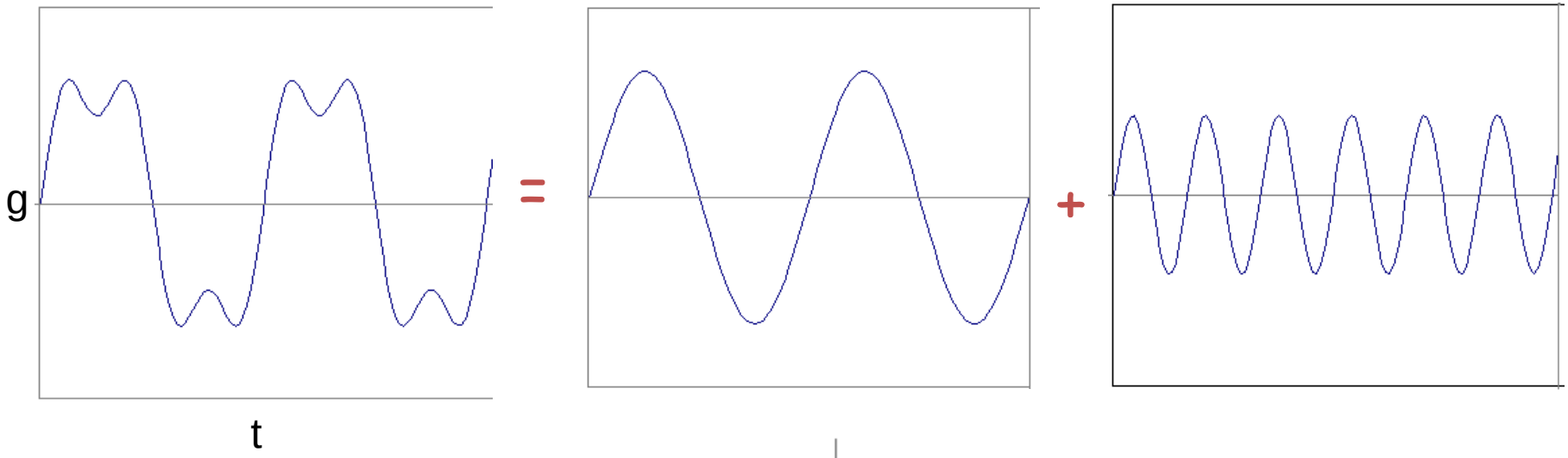
$$A\sin(\omega t) + B\cos(\omega t)$$

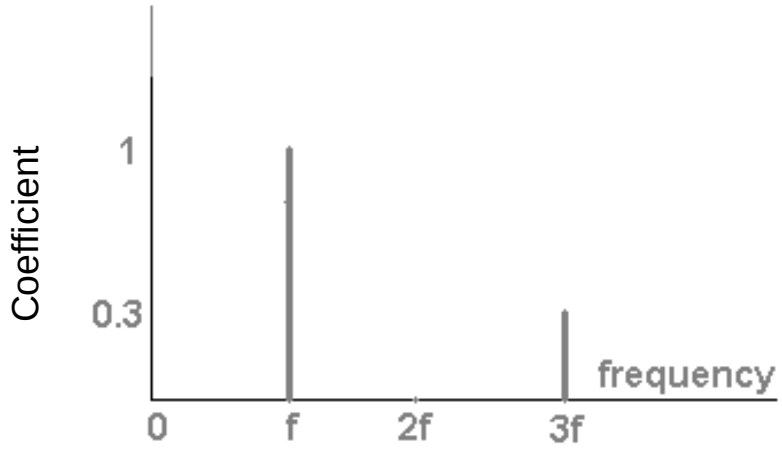
Add enough of them to get any signal *g(t)* you want!



$t = [0,2]$, $f = 1$

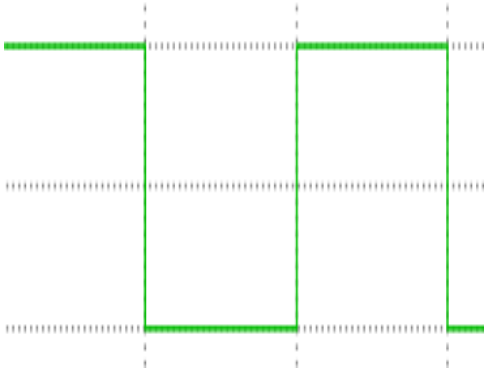
$$g(t) = \sin(2\pi f\, t) + (1/3)\sin(2\pi(3f)\, t)$$



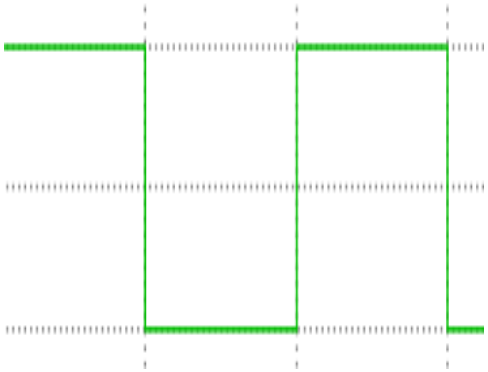


Slides: Efros

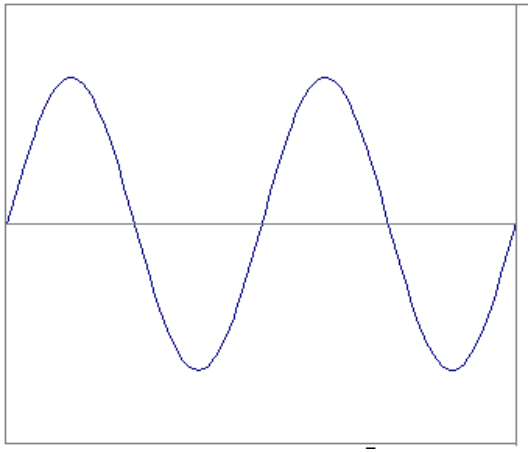
Square wave spectra



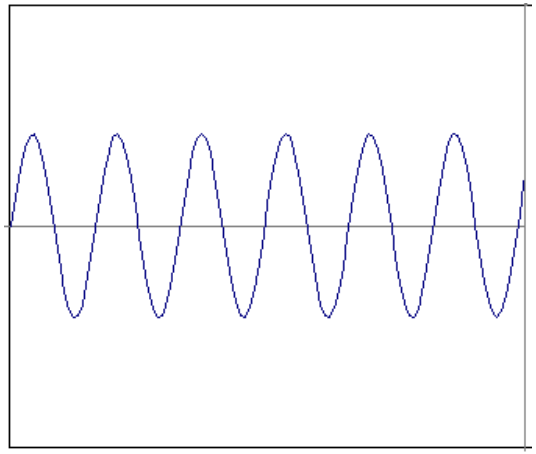
Square wave spectra



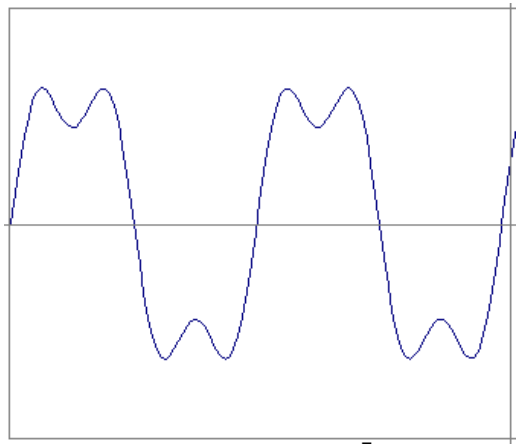
=



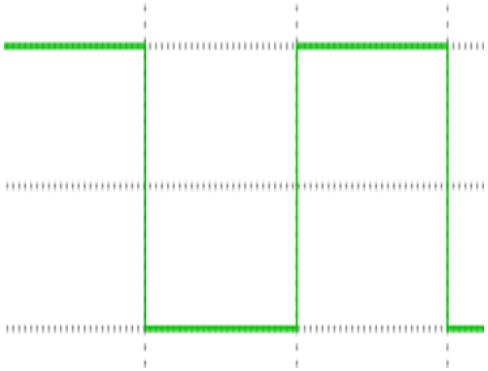
+



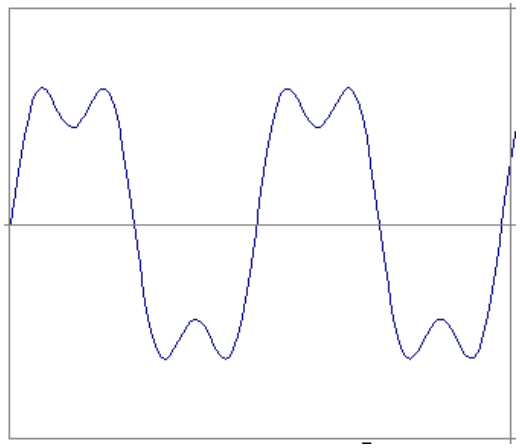
=



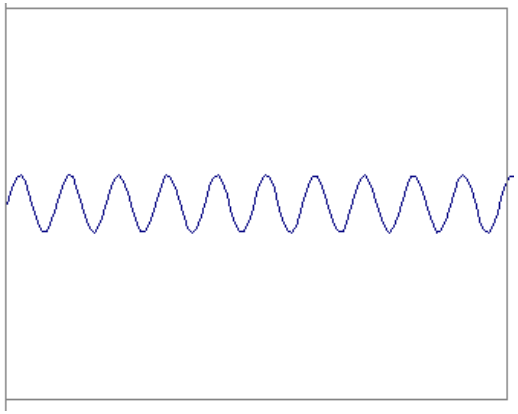
Square wave spectra



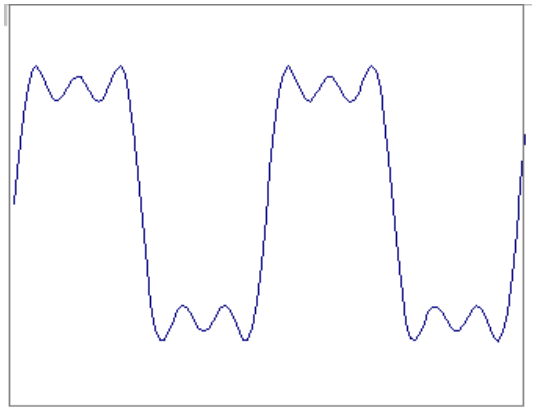
=



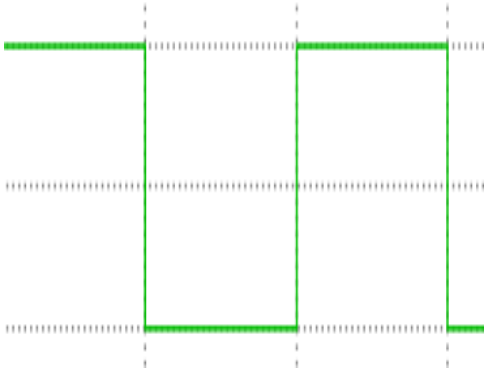
+

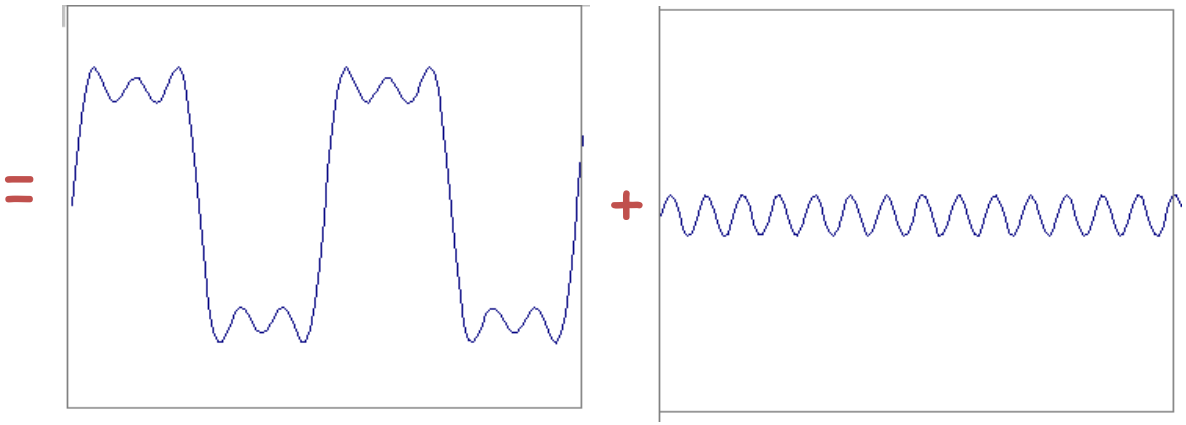


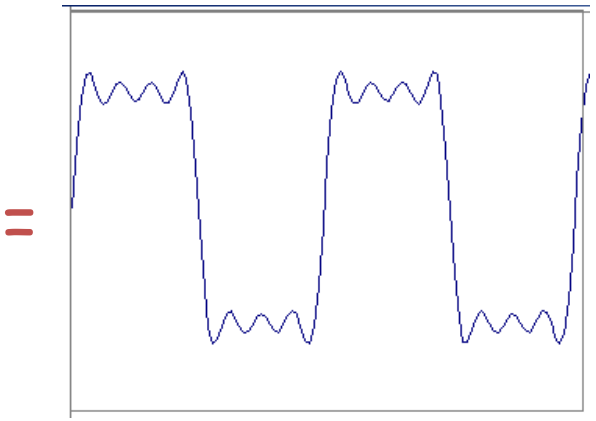
=



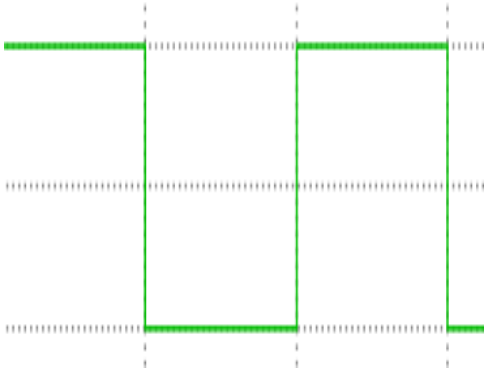
Square wave spectra



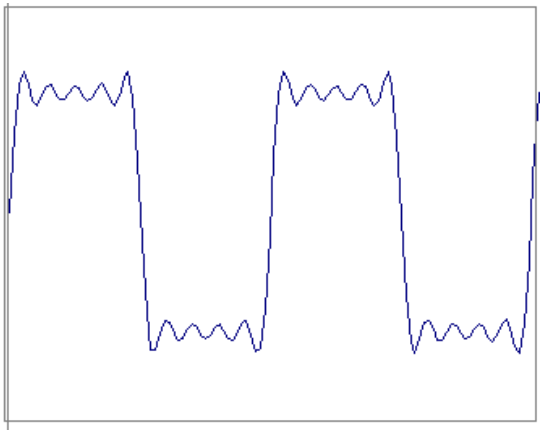




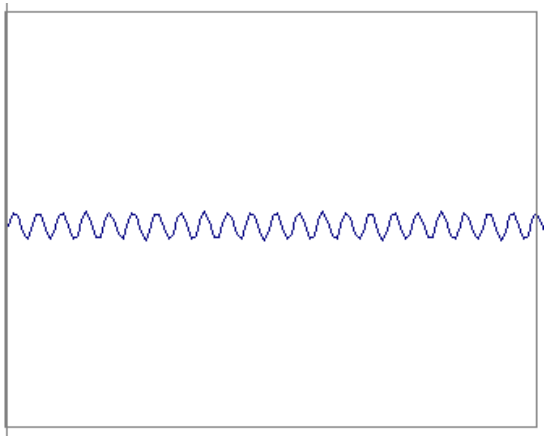
Square wave spectra



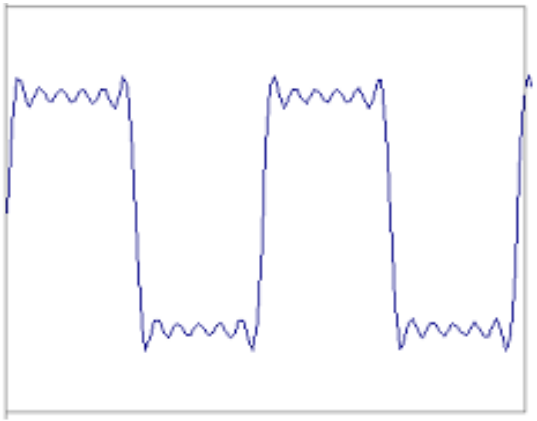
=



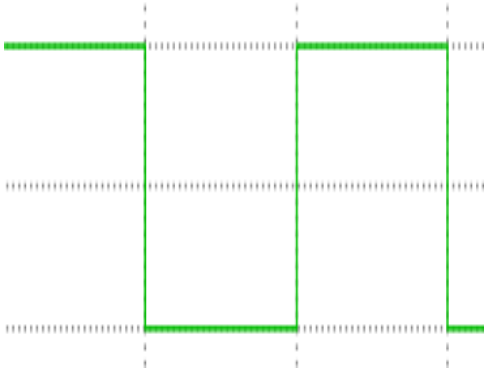
+

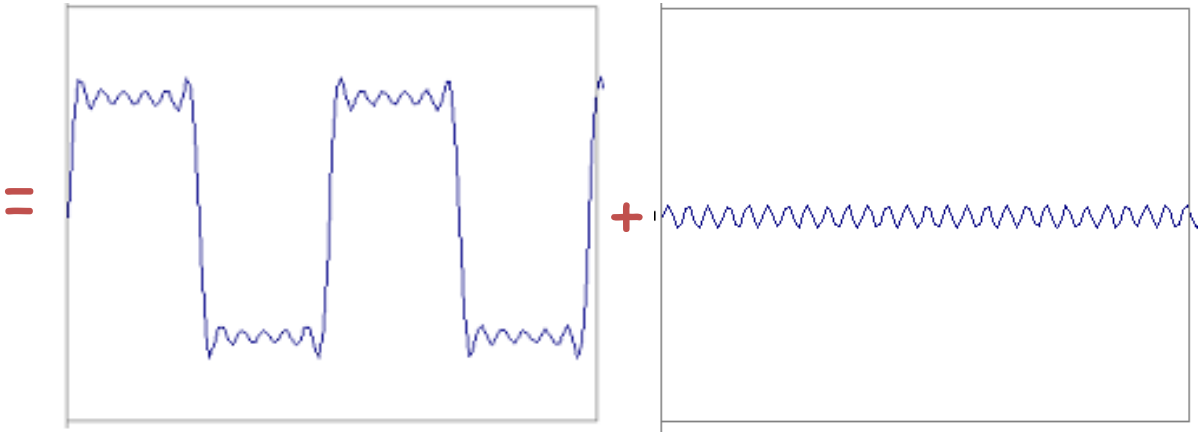


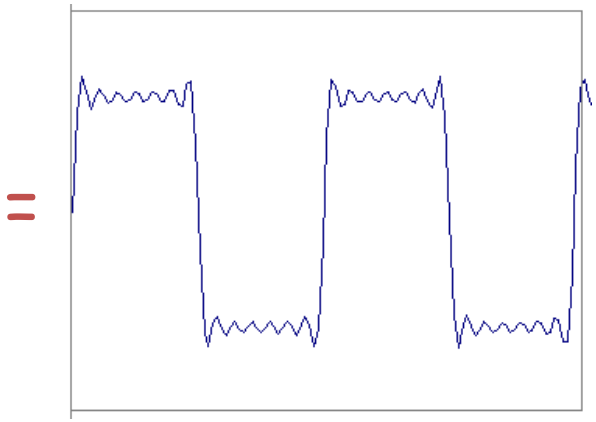
=



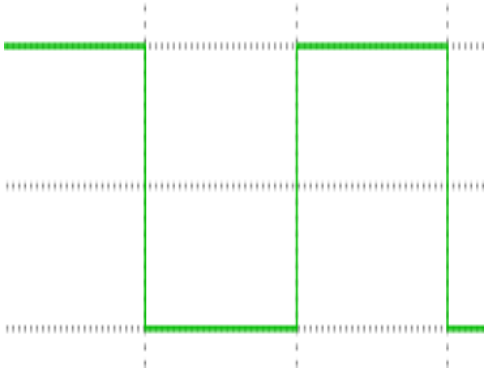
Square wave spectra



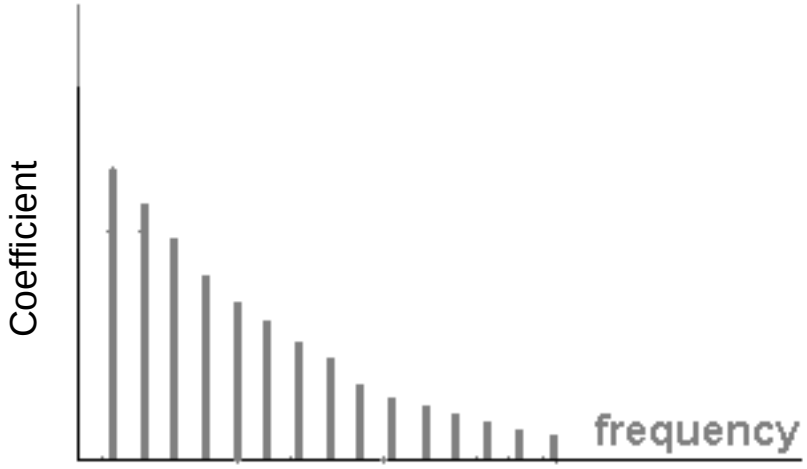


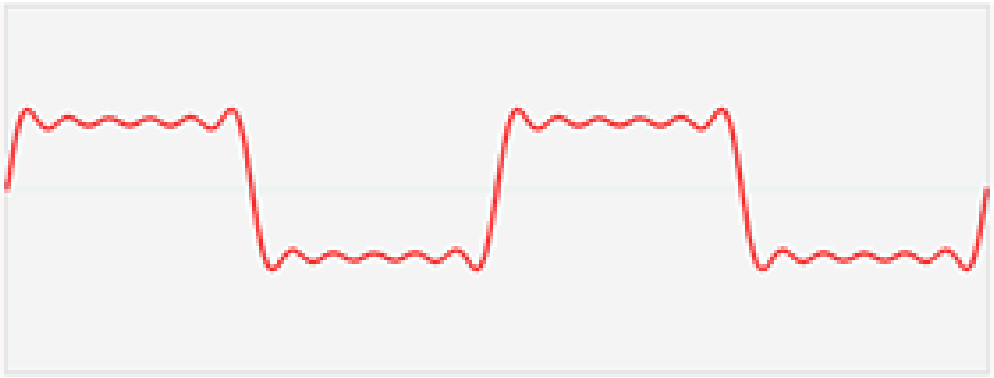


Square wave spectra



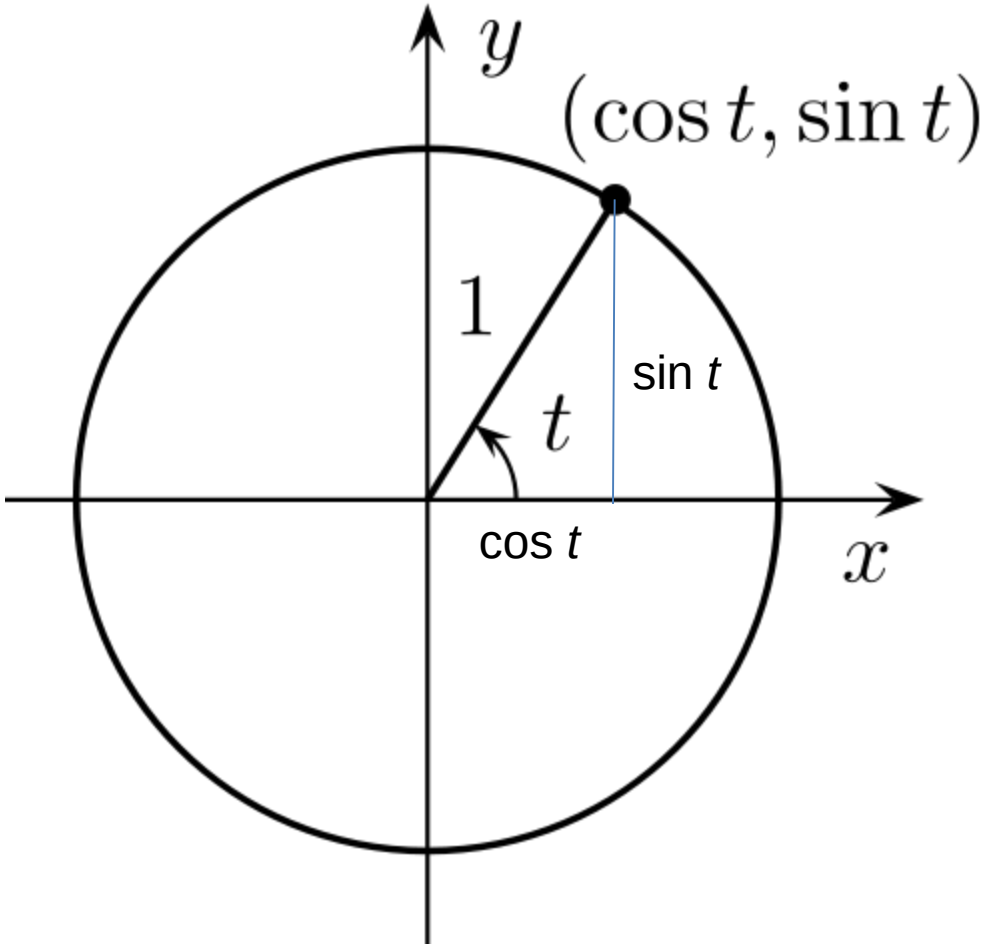
$$= A \sum_{f=1}^{\infty} \frac{1}{f} \sin(2\pi f t)$$





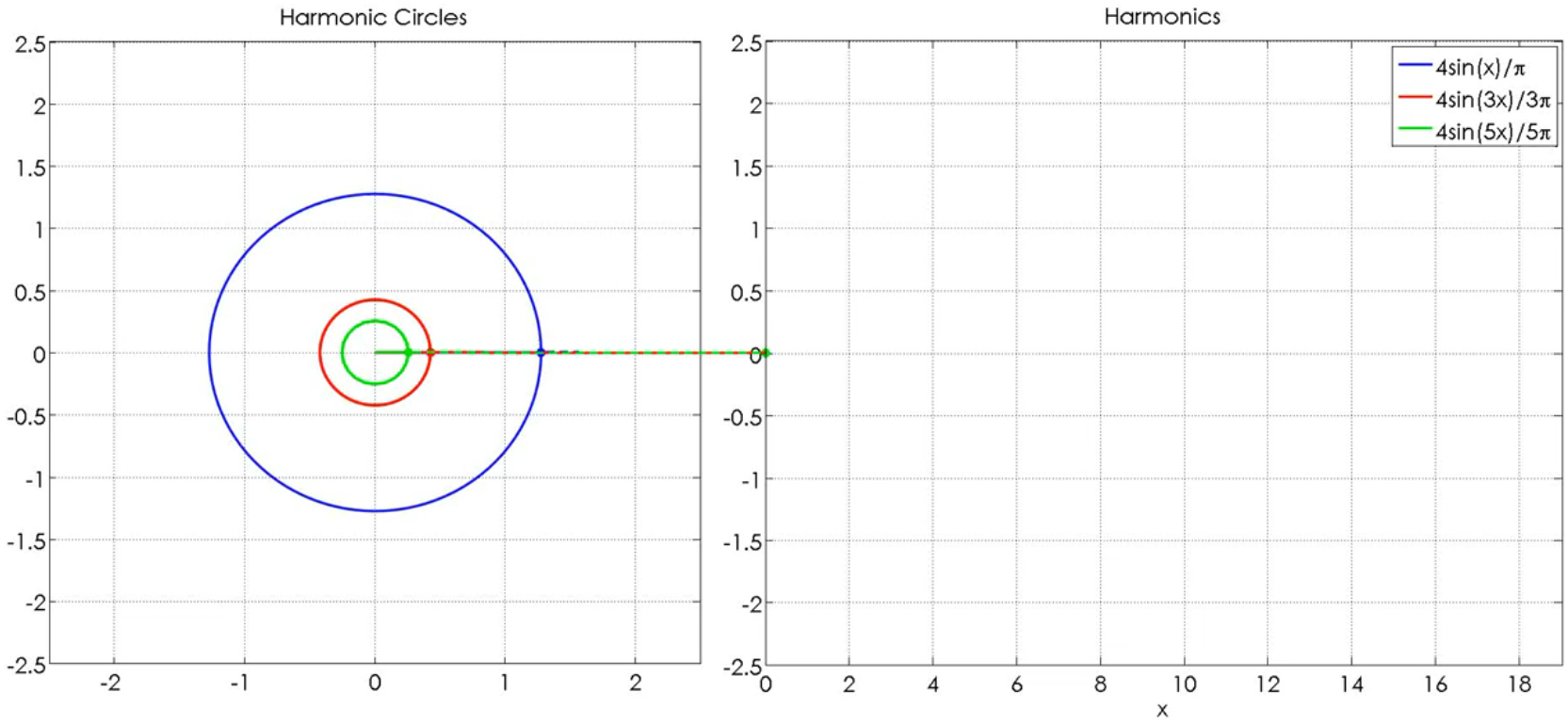
Wikipedia – Fourier transform

Sine/cosine and circle

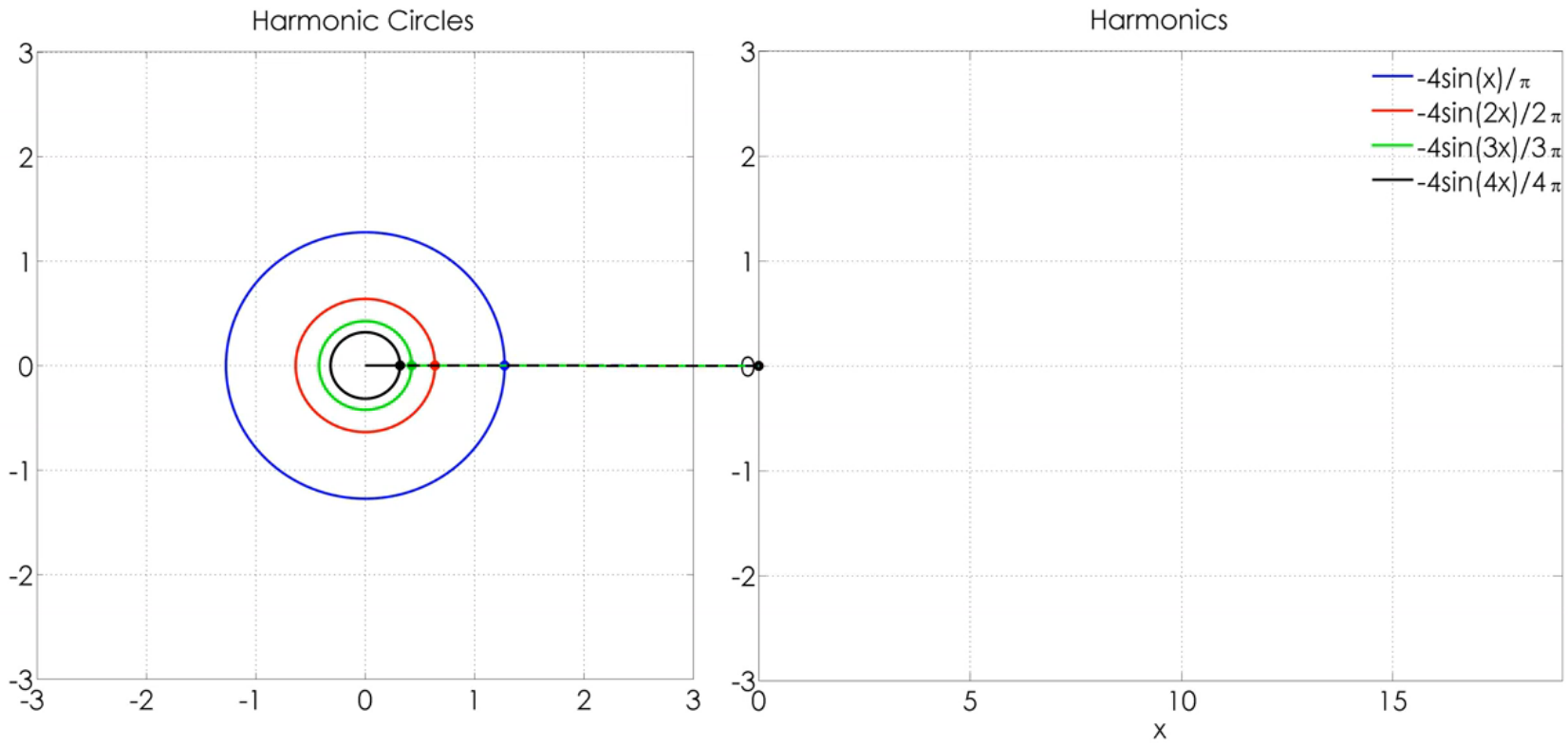


Wikipedia – Unit Circle

Square wave (approx.)

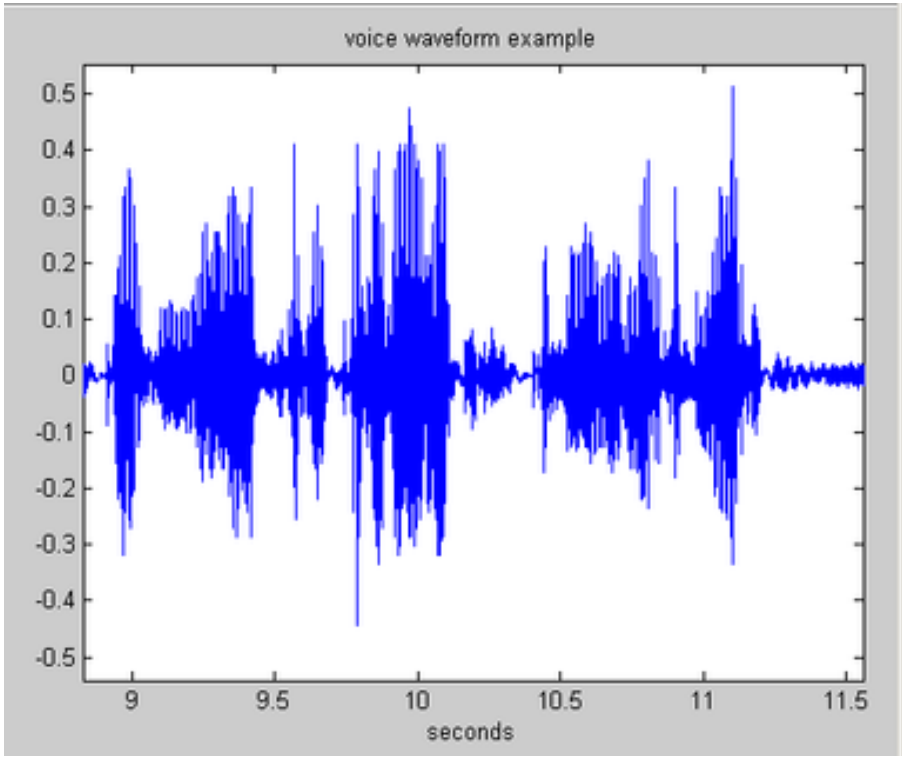


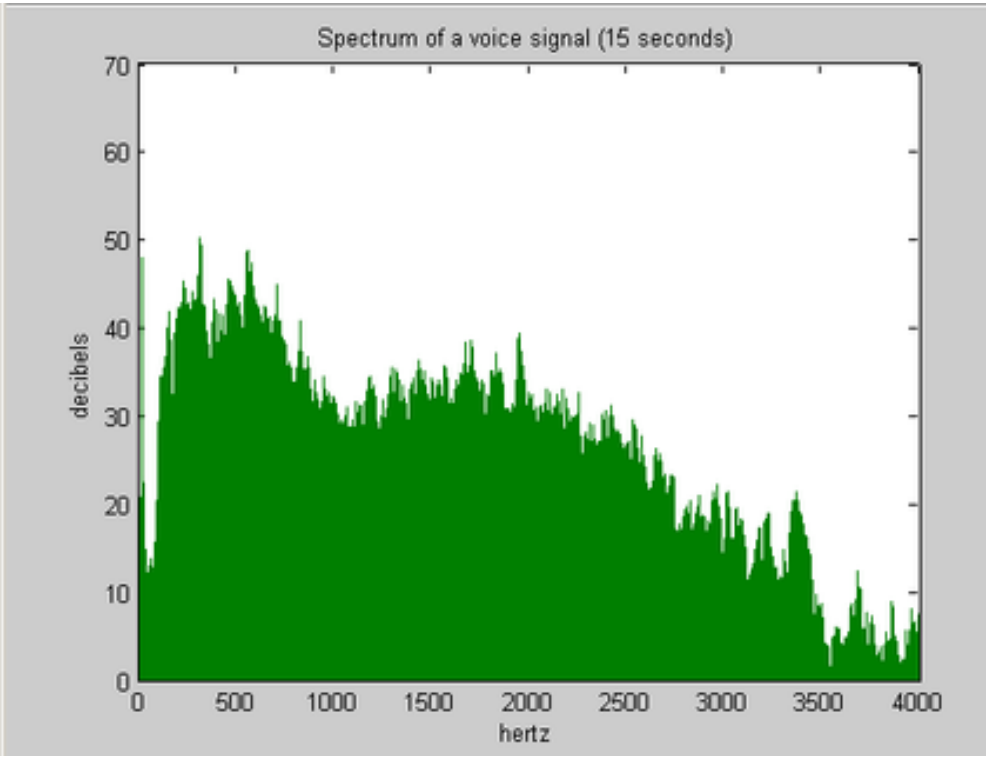
Sawtooth wave (approx.)



Example: Music

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





Slide: Hoiem

Computing Fourier transform

The Fourier transform of a discrete function of one variable, $f(x), x = 0, 1, 2, \ldots, M - 1$, is given by the equation

$$F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{-j2\pi ux/M} \qquad \text{for } u = 0, 1, 2, \ldots, M - 1. \quad (4.2\text{-}5)$$

This *discrete Fourier transform* (DFT) is the foundation for most of the work in this chapter. Similarly, given $F(u)$, we can obtain the original function back using the inverse DFT:

$$f(x) = \sum_{u=0}^{M-1} F(u) e^{j2\pi ux/M} \qquad \text{for } x = 0, 1, 2, \ldots, M - 1. \quad (4.2\text{-}6)$$

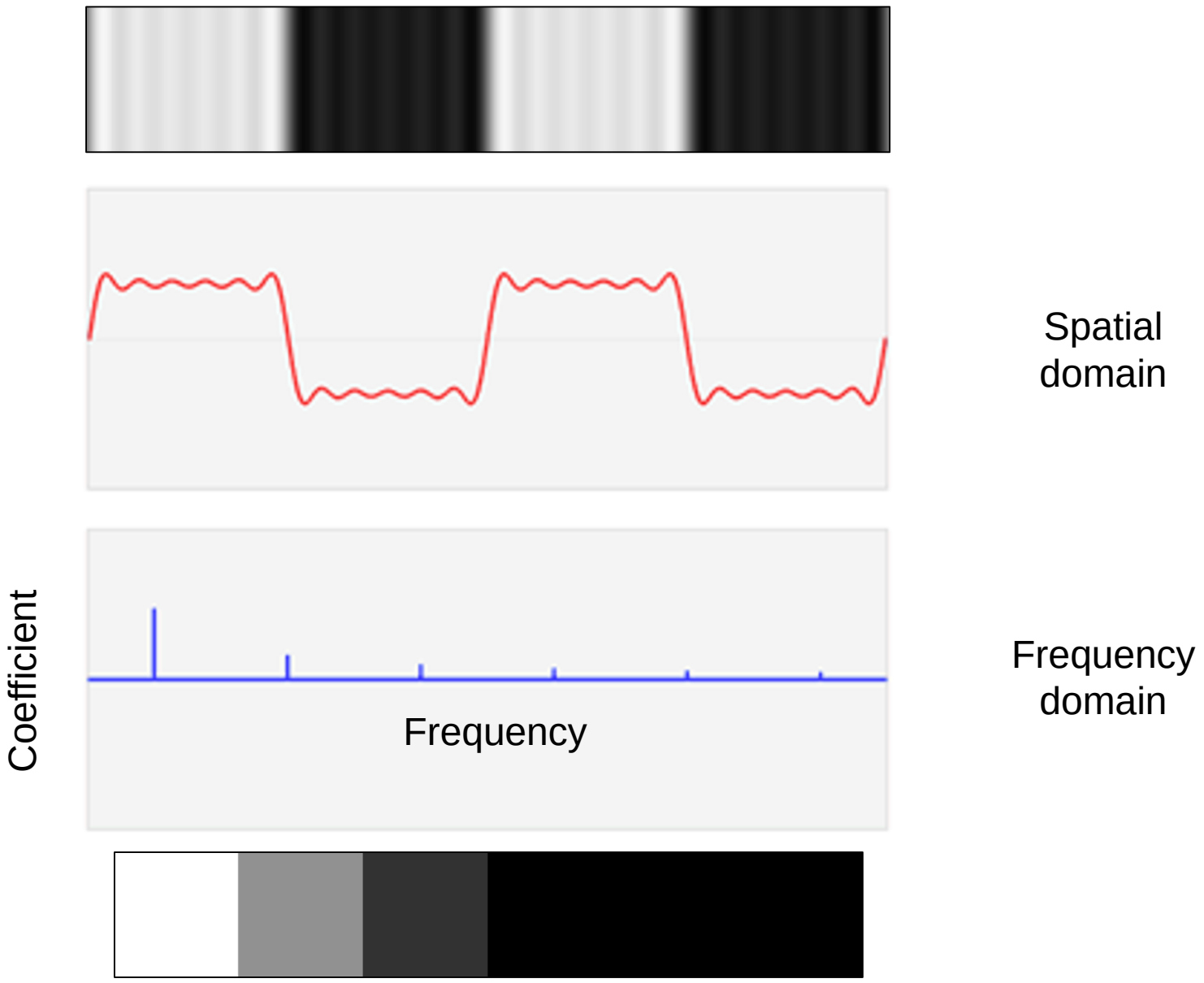
Computing Fourier transform

The concept of the frequency domain, mentioned numerous times in this chapter and in Chapter 3, follows directly from Euler's formula:

$$e^{j\theta} = \cos\theta + j\sin\theta.$$

Substituting this expression into Eq. (4.2-5), and using the fact that $\cos(-\theta) = \cos\theta$, gives us

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)[\cos 2\pi ux/M - j\sin 2\pi ux/M]$$



Wikipedia – Fourier transform

2D FFT

Direct transform

$$F(u,v) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) \exp\left[-2\pi i \left(\frac{mu}{M} + \frac{nv}{N}\right)\right] ,$$

$$u = 0, 1, \ldots, M-1 , \qquad v = 0, 1, \ldots, N-1 ,$$

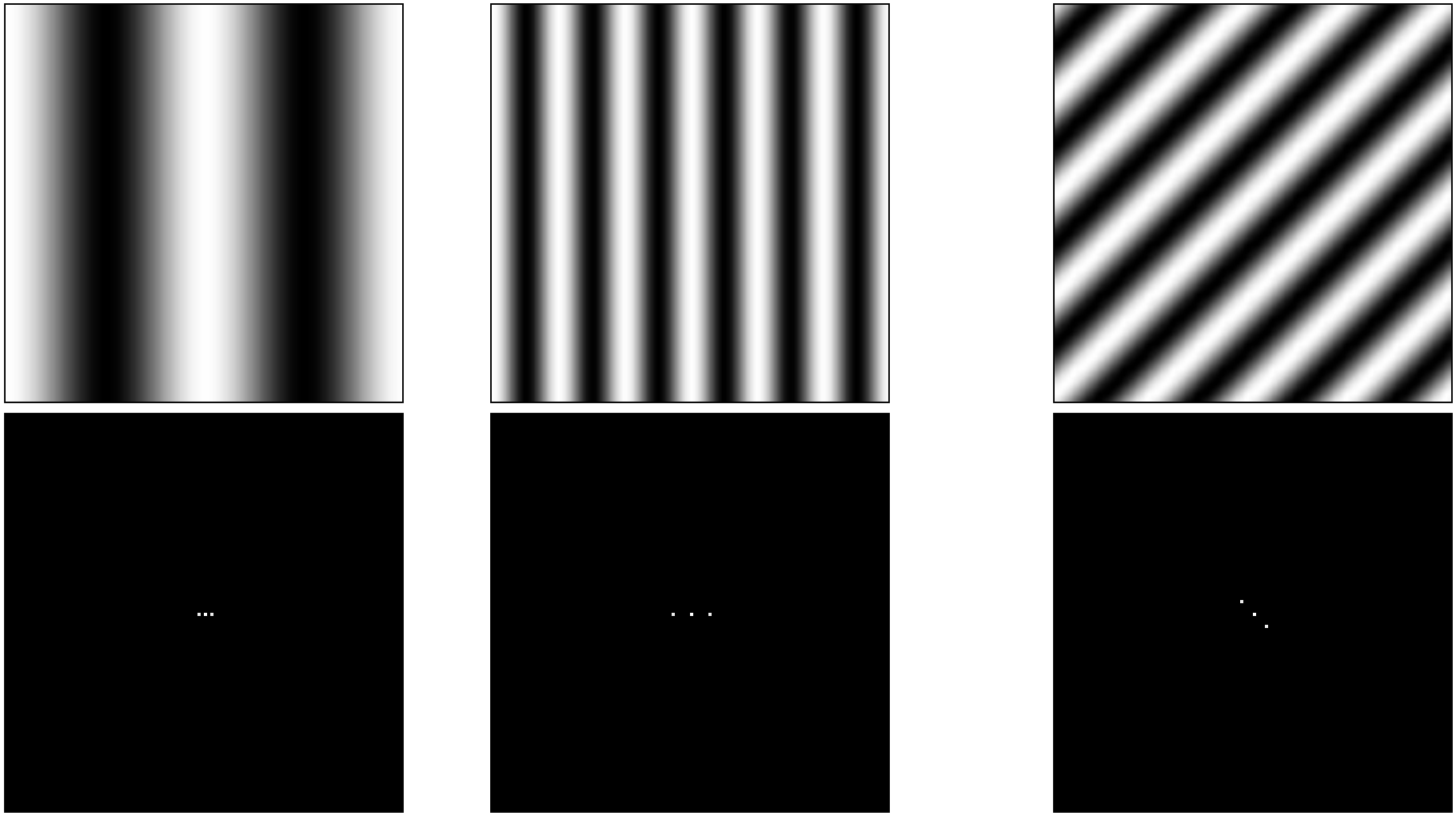
Inverse transform

$$f(m,n) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \exp\left[2\pi i \left(\frac{mu}{M} + \frac{nv}{N}\right)\right] ,$$

$$m = 0, 1, \ldots, M-1 , \qquad n = 0, 1, \ldots, N-1 .$$

Fourier analysis in images

Spatial domain images

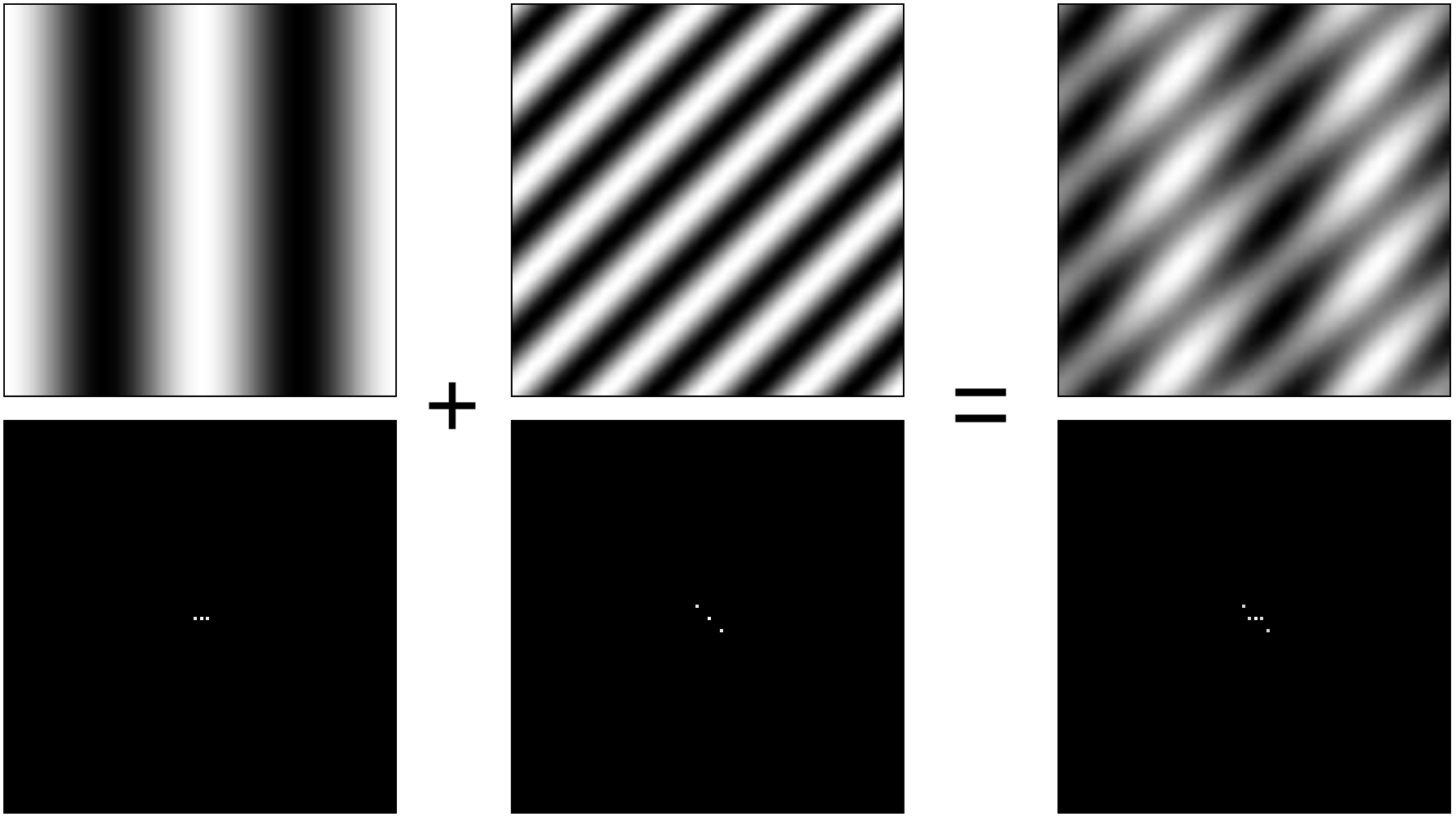


Fourier decomposition frequency amplitude images

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

Signals can be composed

Spatial domain images



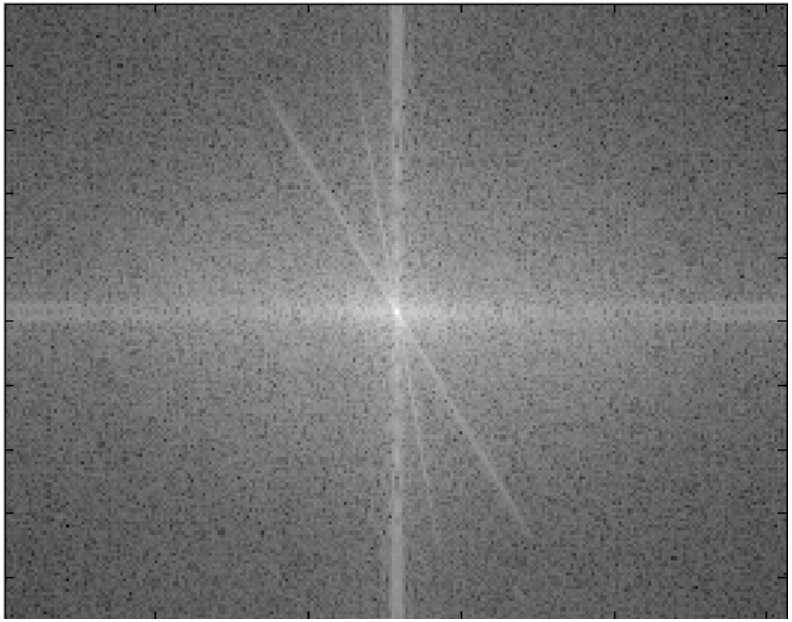
Fourier decomposition frequency amplitude images

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

Natural image

Natural image

Fourier decomposition
Frequency coefficients (amplitude)

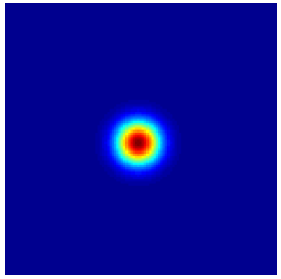


What does it mean to be at pixel x,y?
What does it mean to be more or less bright in the Fourier decomposition image?

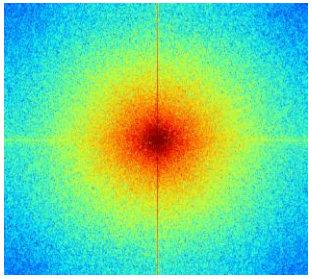
Think-Pair-Share

Match the spatial domain image to the Fourier magnitude image

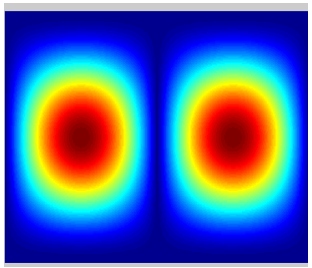
1



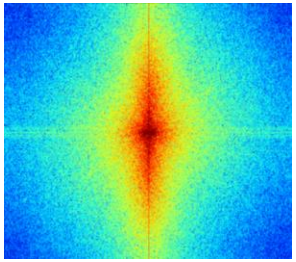
2



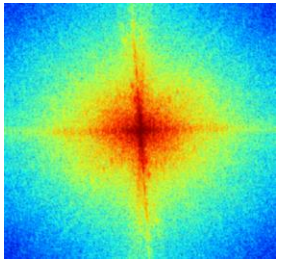
3



4



5

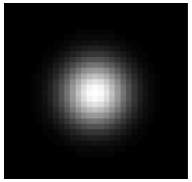


A

B

C

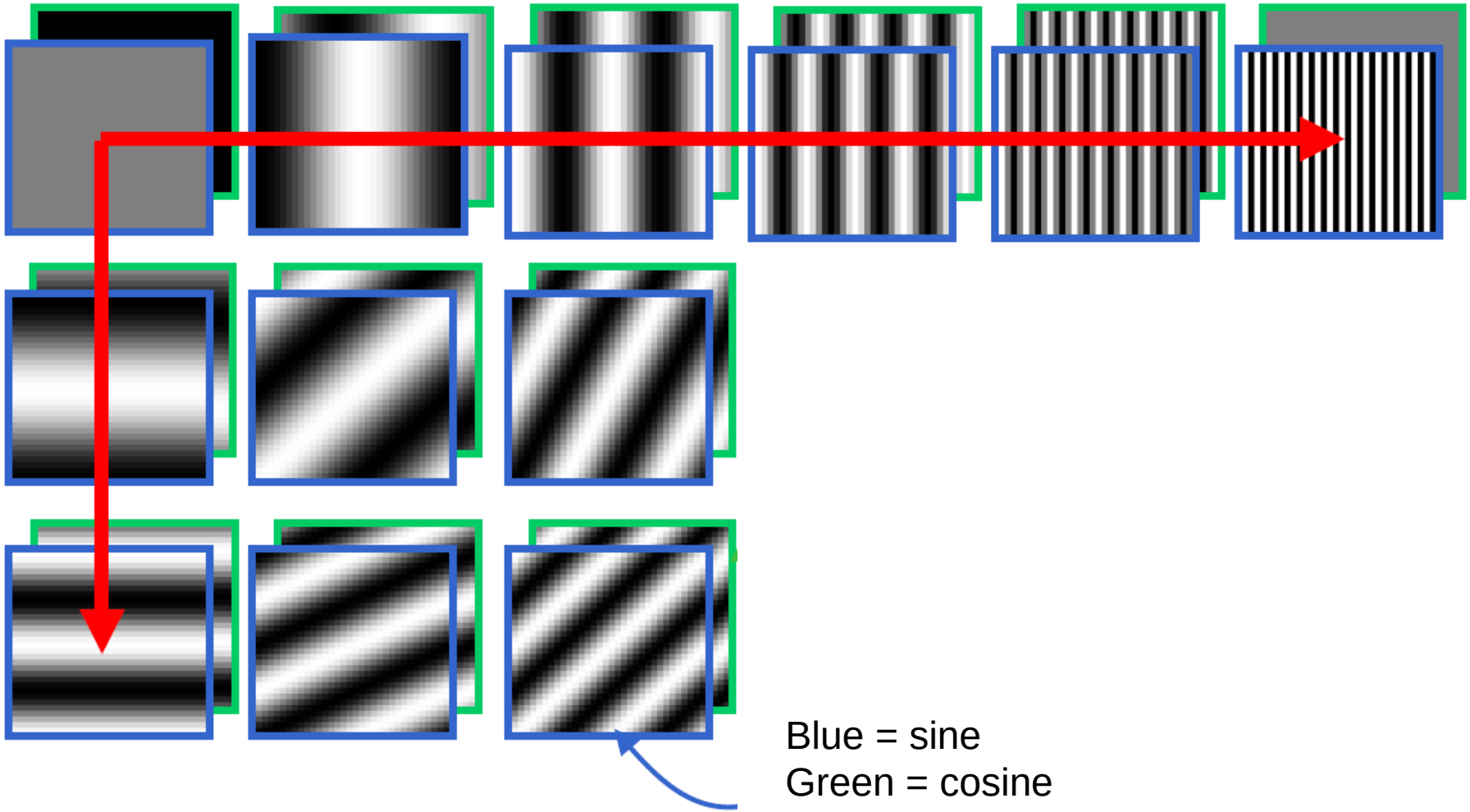
D



E

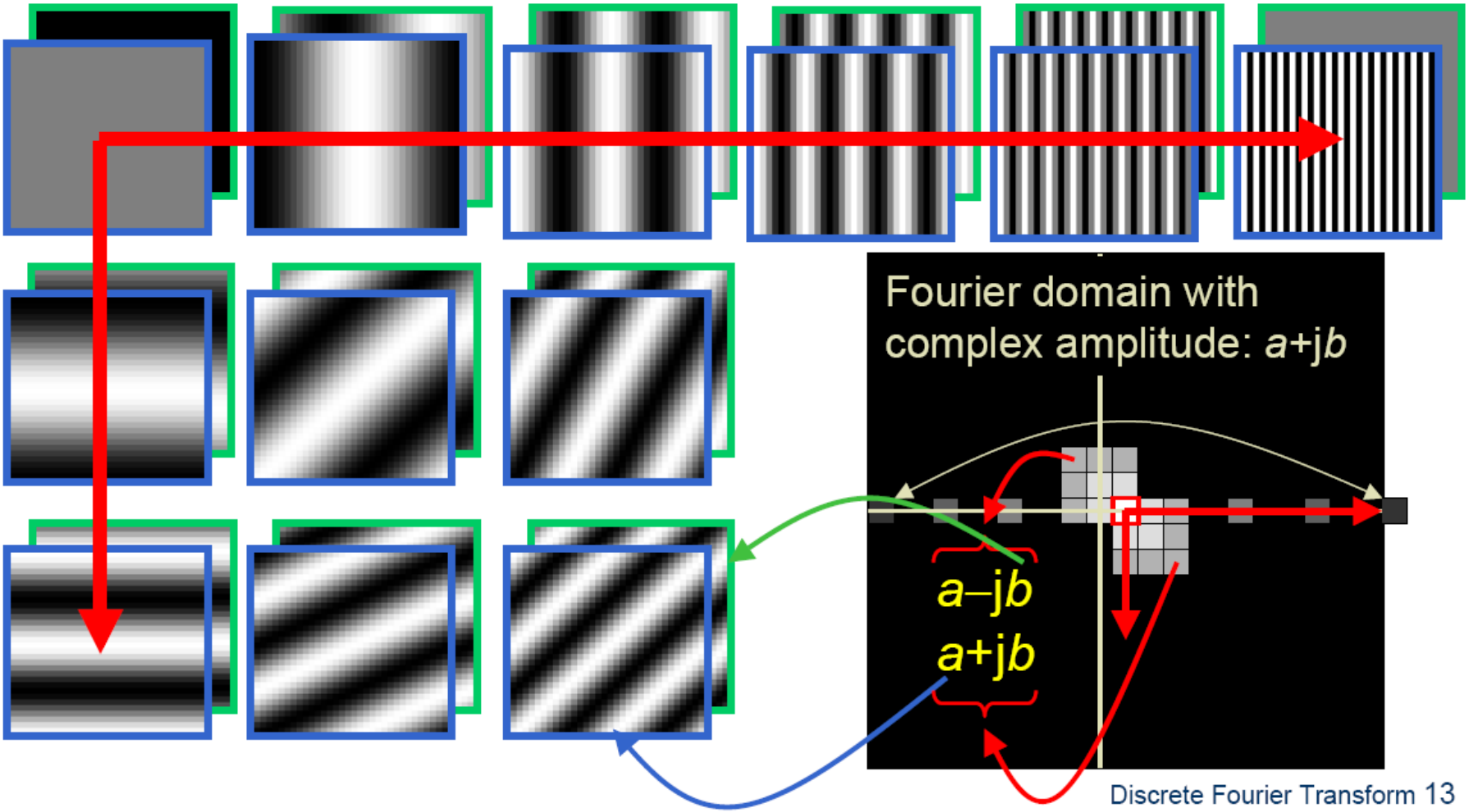
Fourier Bases

Teases away ‘fast vs. slow’ changes in the image.



This change of basis is the Fourier Transform

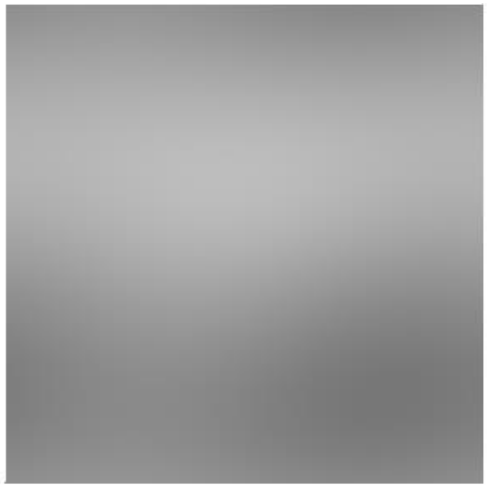
Fourier Bases



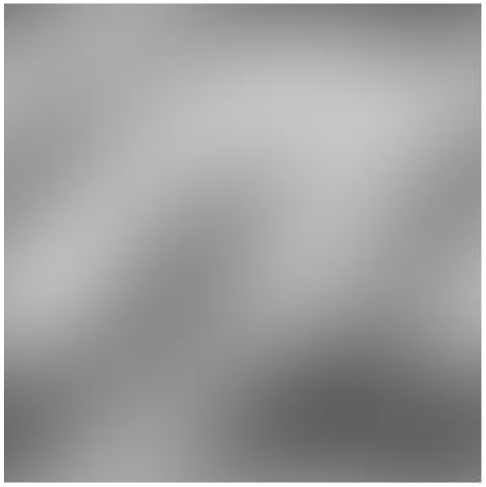
Basis reconstruction

Full image

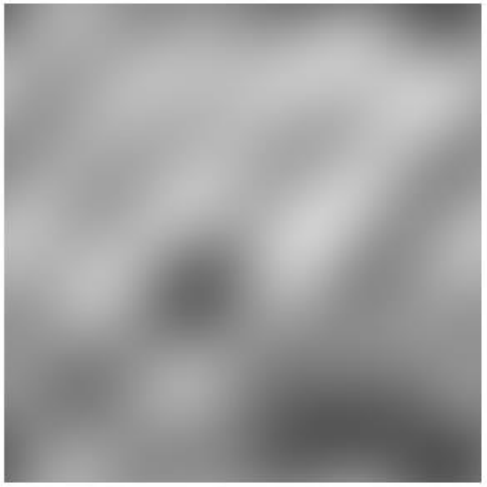
First 1 basis fn



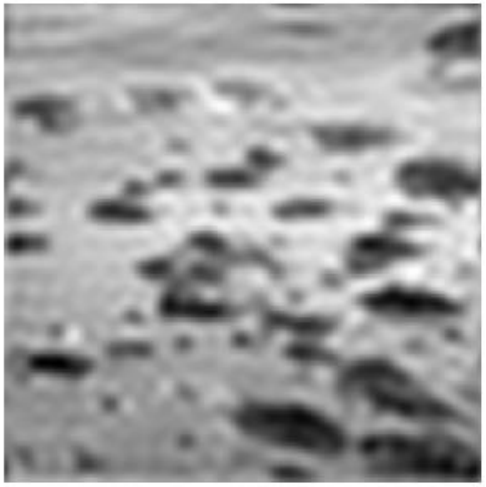
First 4 basis fns



First 9 basis fns



First 16 basis fns



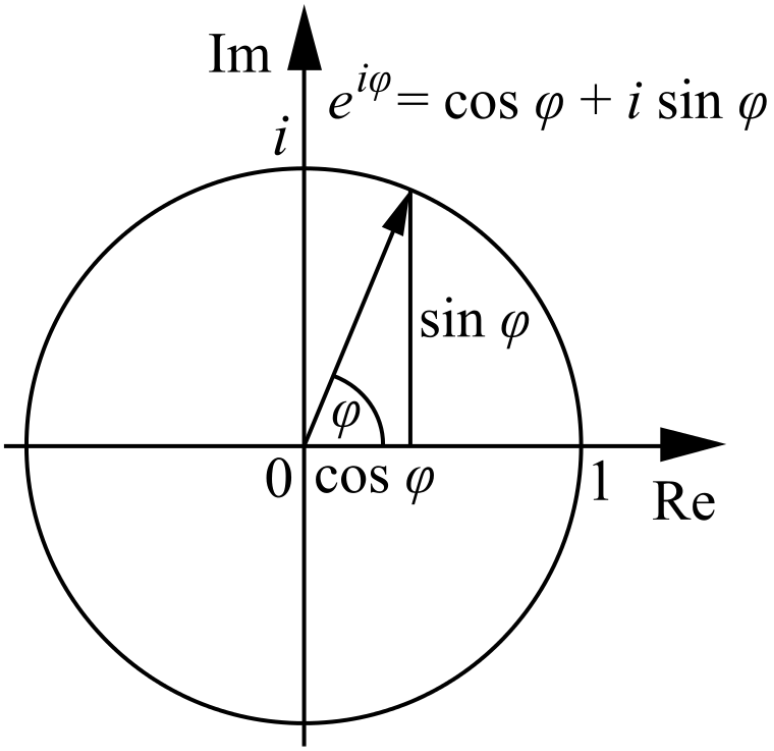
First 400 basis fns

Fourier Transform

- Stores the amplitude and phase at each frequency:
  - For mathematical convenience, this is often notated in terms of real and complex numbers
  - Related by Euler's formula

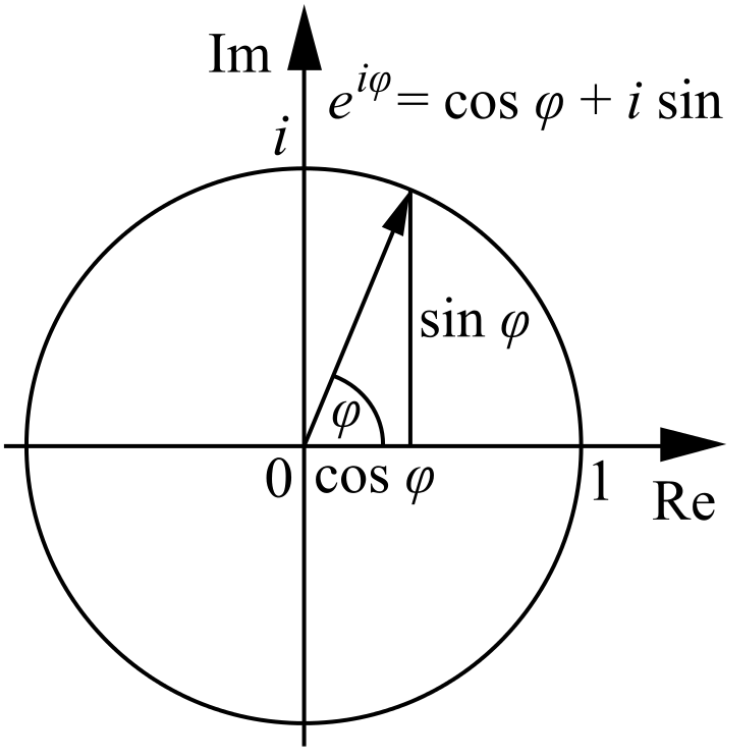
Fourier Transform

- Stores the amplitude and phase at each frequency:
  - For mathematical convenience, this is often notated in terms of real and complex numbers
  - Related by Euler's formula



Fourier Transform

- Stores the amplitude and phase at each frequency:
  - For mathematical convenience, this is often notated in terms of real and complex numbers
  - Related by Euler's formula



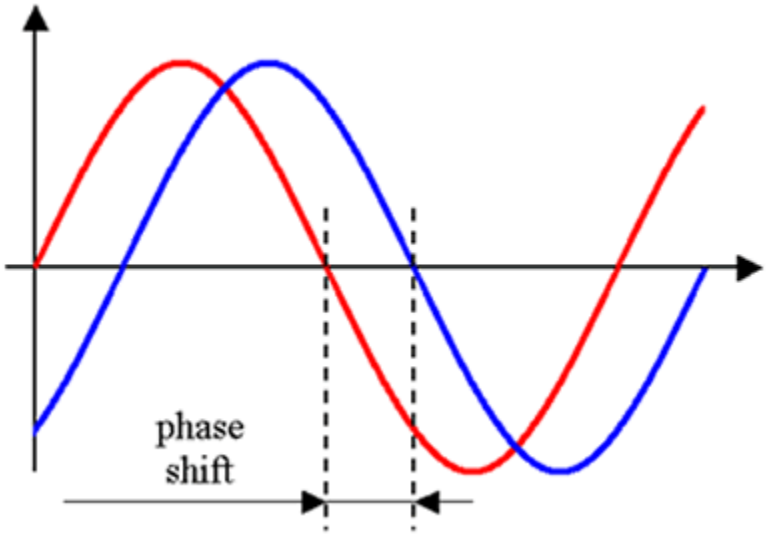
Amplitude encodes how much signal there is at a particular frequency:

$$A = \pm\sqrt{\mathrm{Re}(\varphi)^2 + \mathrm{Im}(\varphi)^2}$$

Phase encodes spatial information (indirectly):

$$\phi = \tan^{-1}\frac{\mathrm{Im}(\varphi)}{\mathrm{Re}(\varphi)}$$

Amplitude / Phase



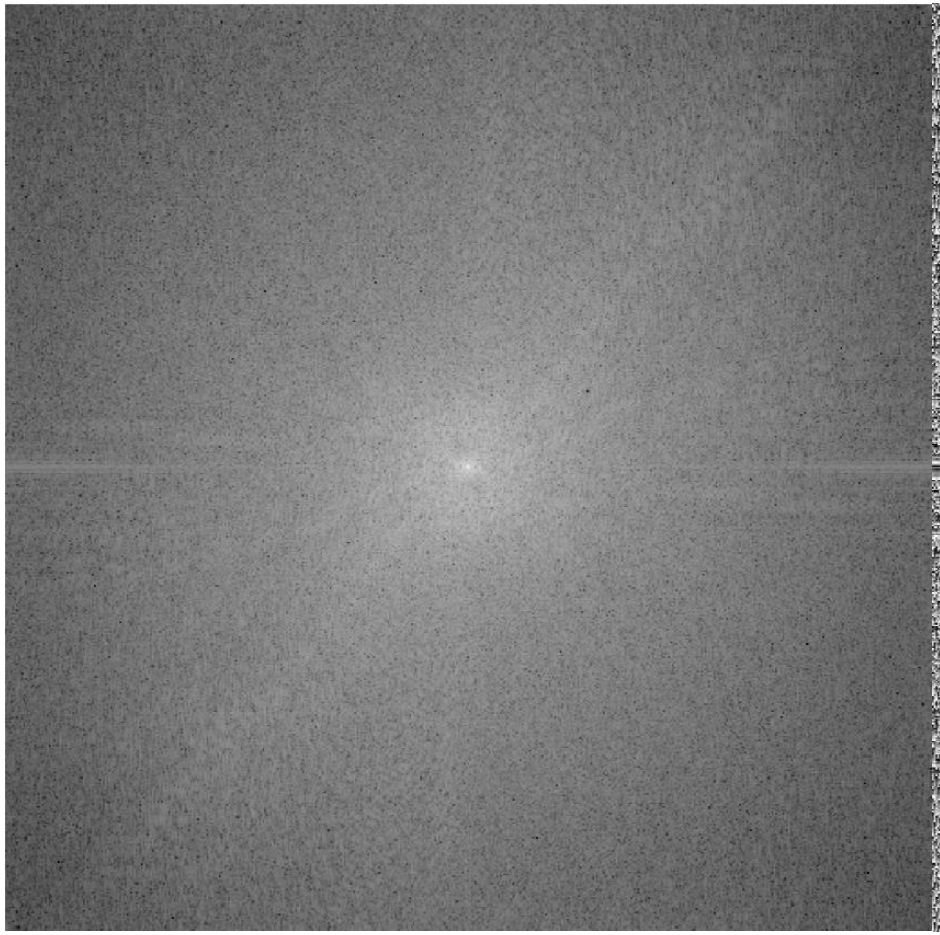
- Amplitude tells you “how much”
- Phase tells you “where”

- Translate the image?
  - Amplitude unchanged
  - Adds a constant to the phase.

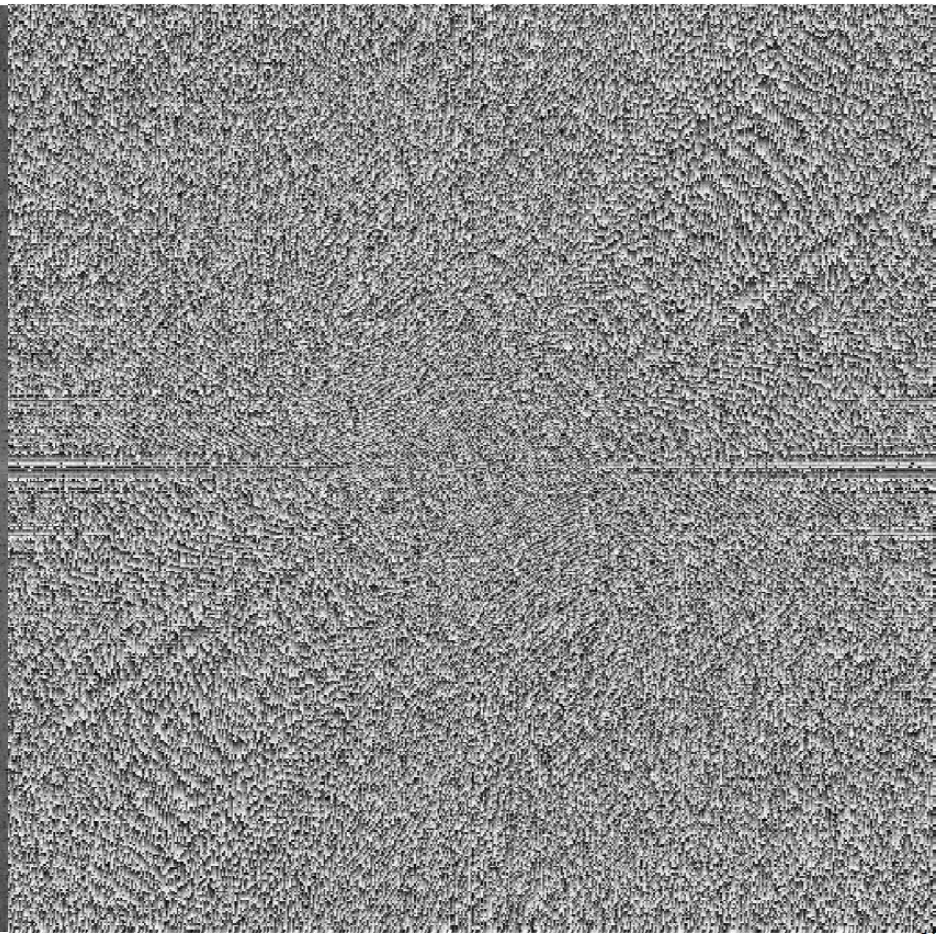
What about phase?

What about phase?

Amplitude



Phase

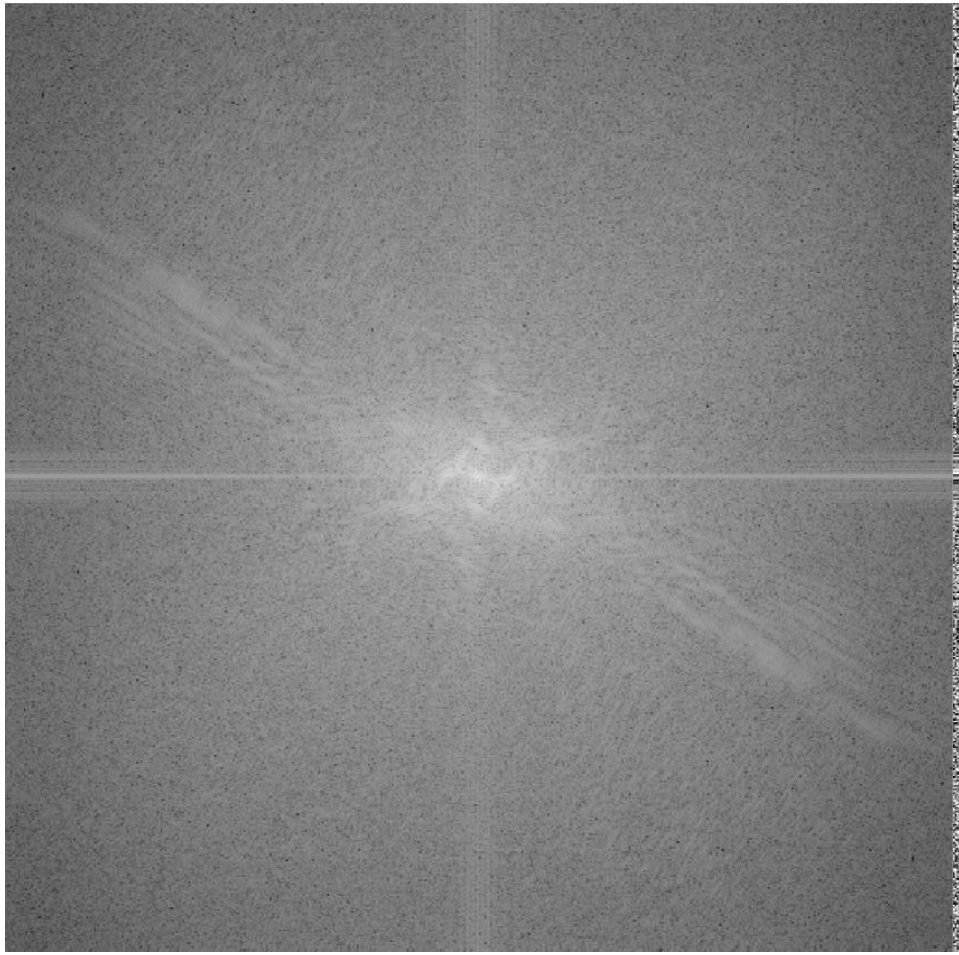


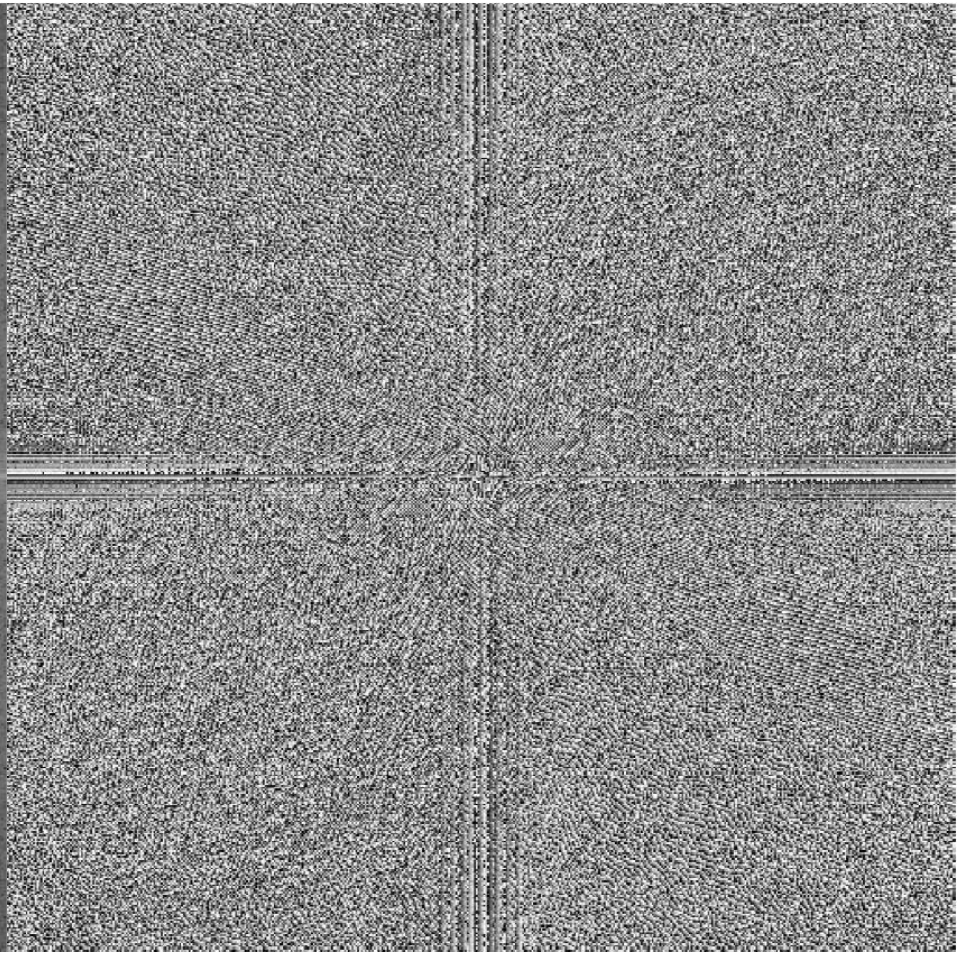
What about phase?

What about phase?

Amplitude

Phase



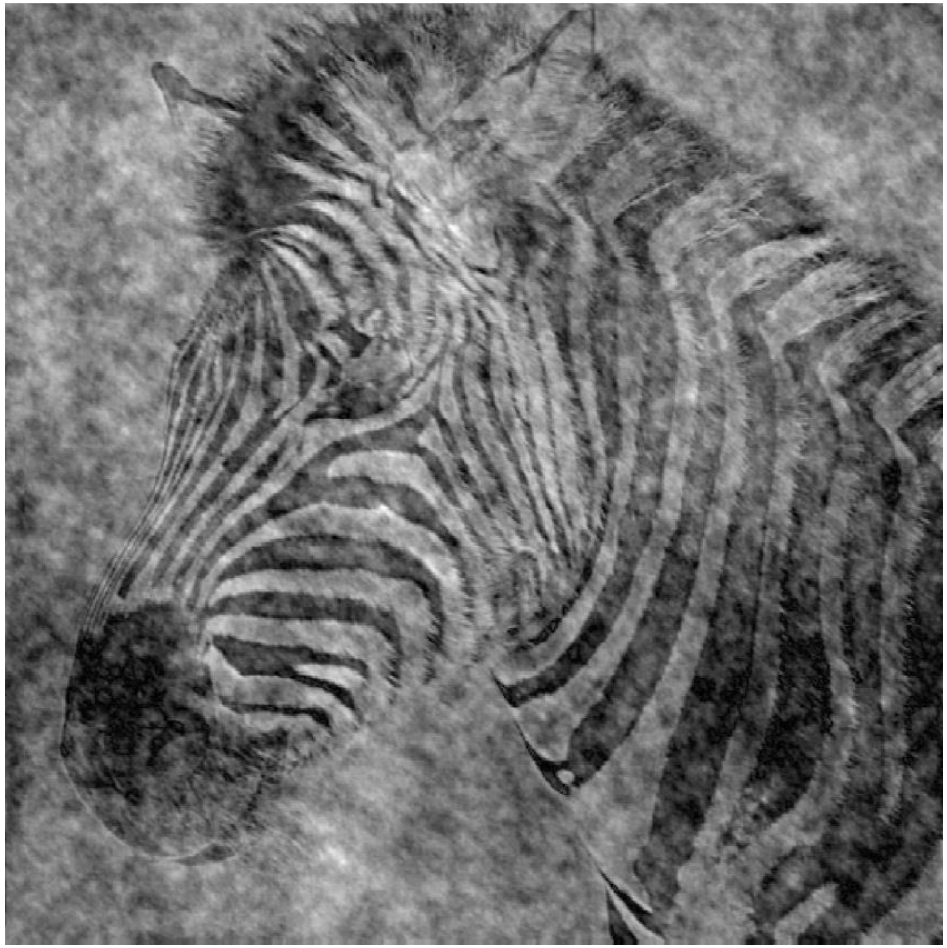


Think-Pair-Share

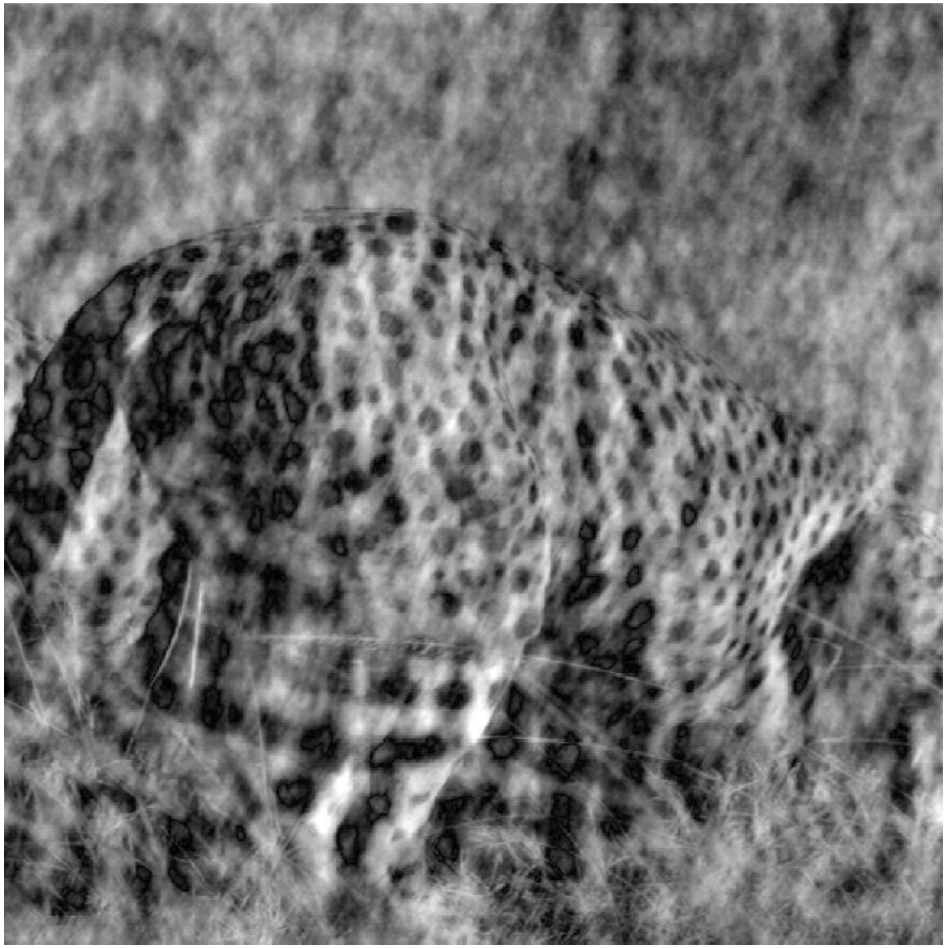
- In Fourier space, where is more of the information that we see in the visual world?
  - Amplitude
  - Phase

Cheebra

Zebra phase, cheetah amplitude



Cheetah phase, zebra amplitude



- The frequency amplitude of natural images are quite similar
  - Heavy in low frequencies, falling off in high frequencies
  - Will *any* image be like that, or is it a property of the world we live in?
- Most information in the image is carried in the phase, not the amplitude
  - Not quite clear why

Some properties of Fourier Transforms

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

- 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)

The Convolution Theorem

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

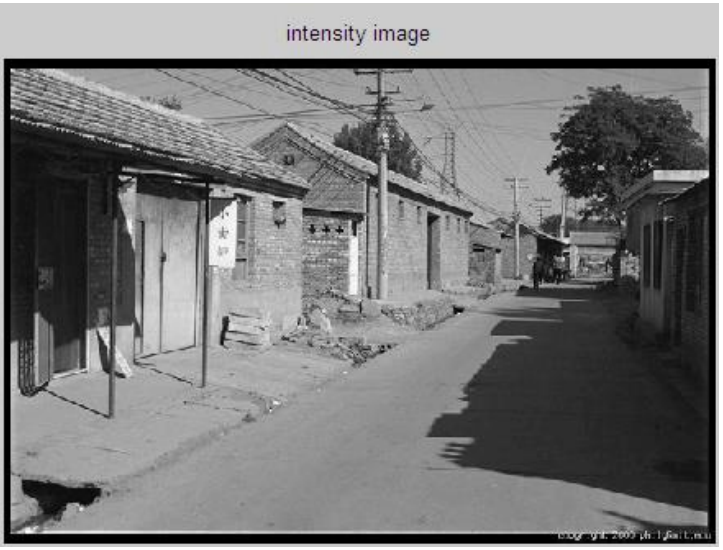
$$\mathrm{F}[g * h] = \mathrm{F}[g]\,\mathrm{F}[h]$$

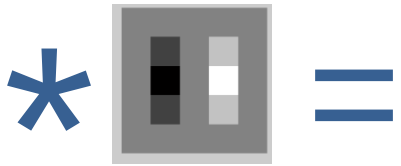
- **Convolution** in spatial domain is equivalent to **multiplication** in frequency domain!

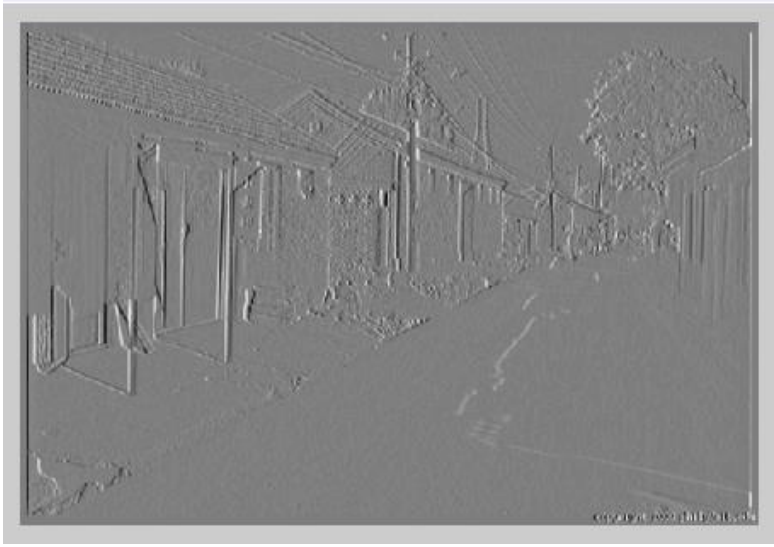
$$g * h = \mathrm{F}^{-1}[\mathrm{F}[g]\,\mathrm{F}[h]]$$

Filtering in spatial domain

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

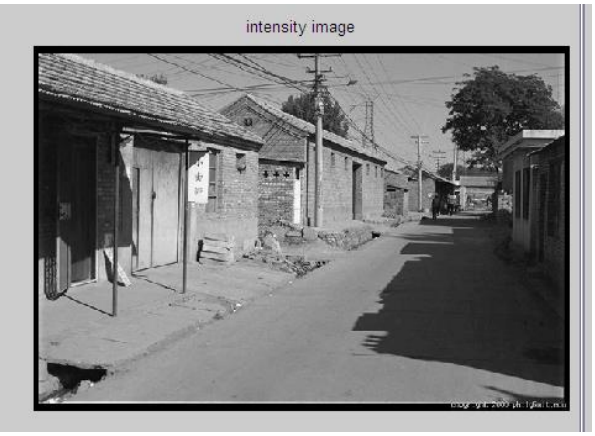




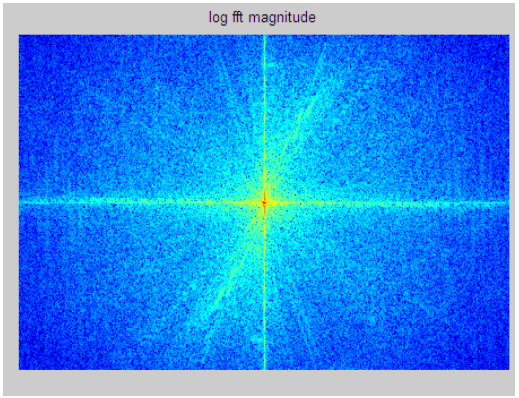


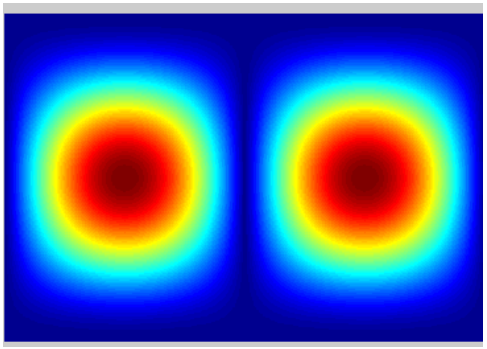
Filtering in frequency domain

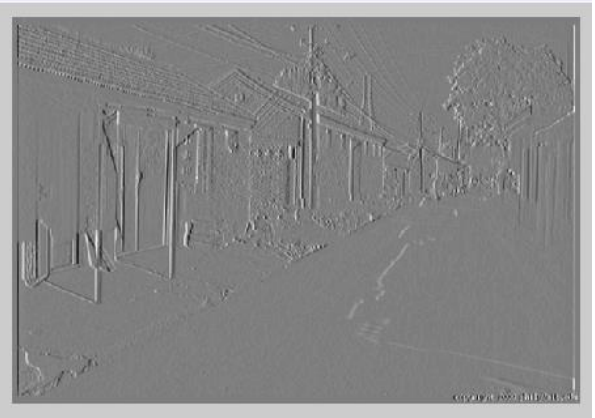
FFT



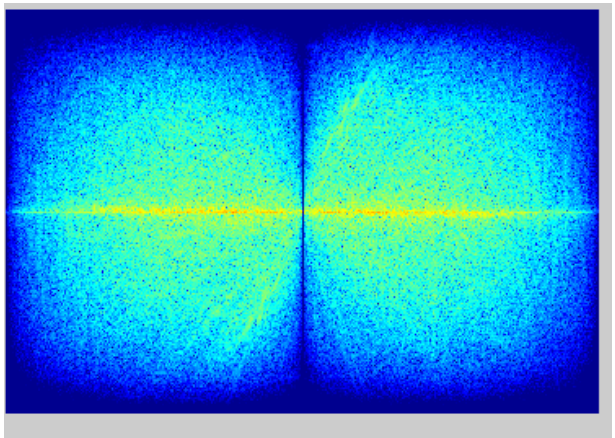
FFT







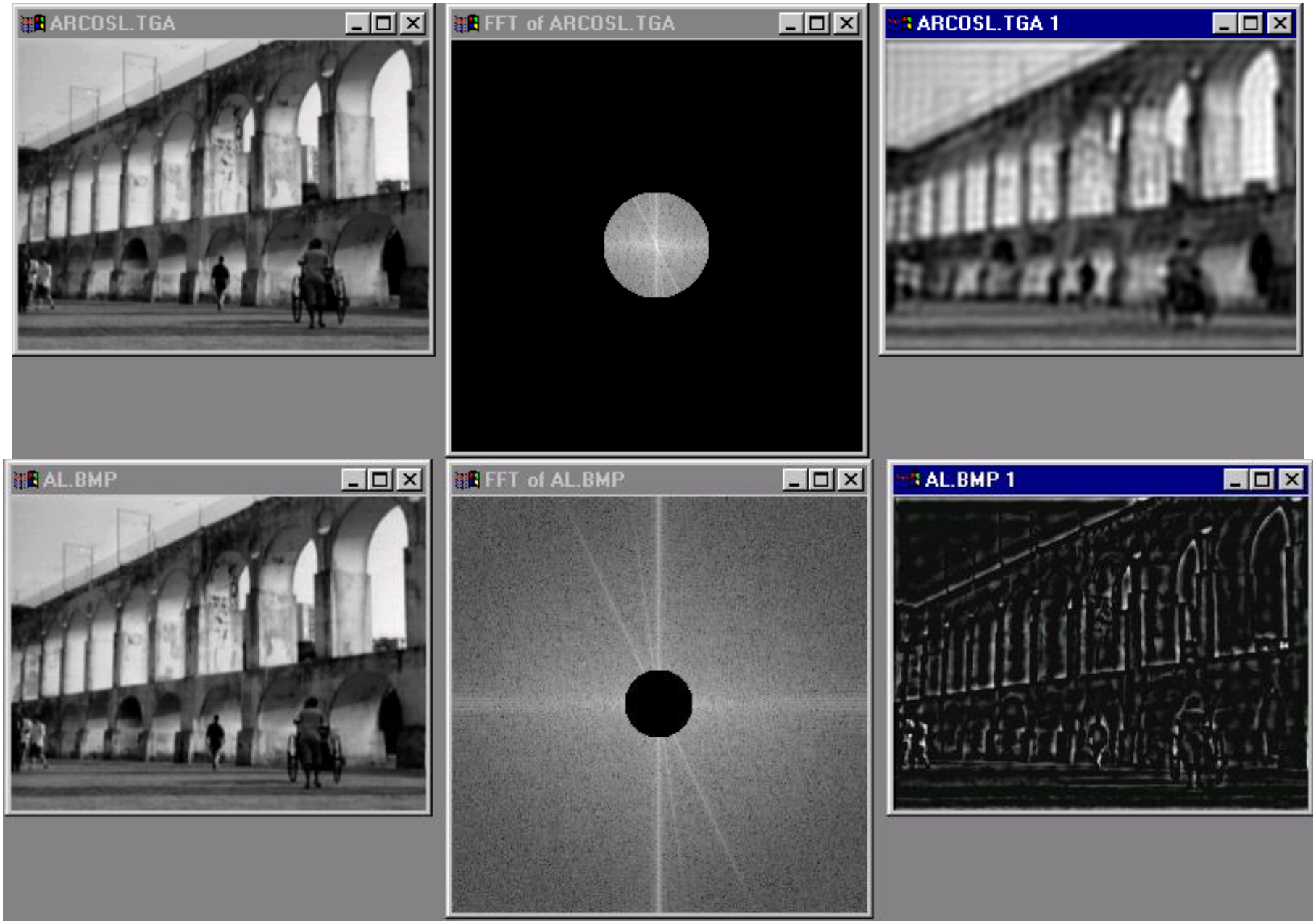
Inverse FFT



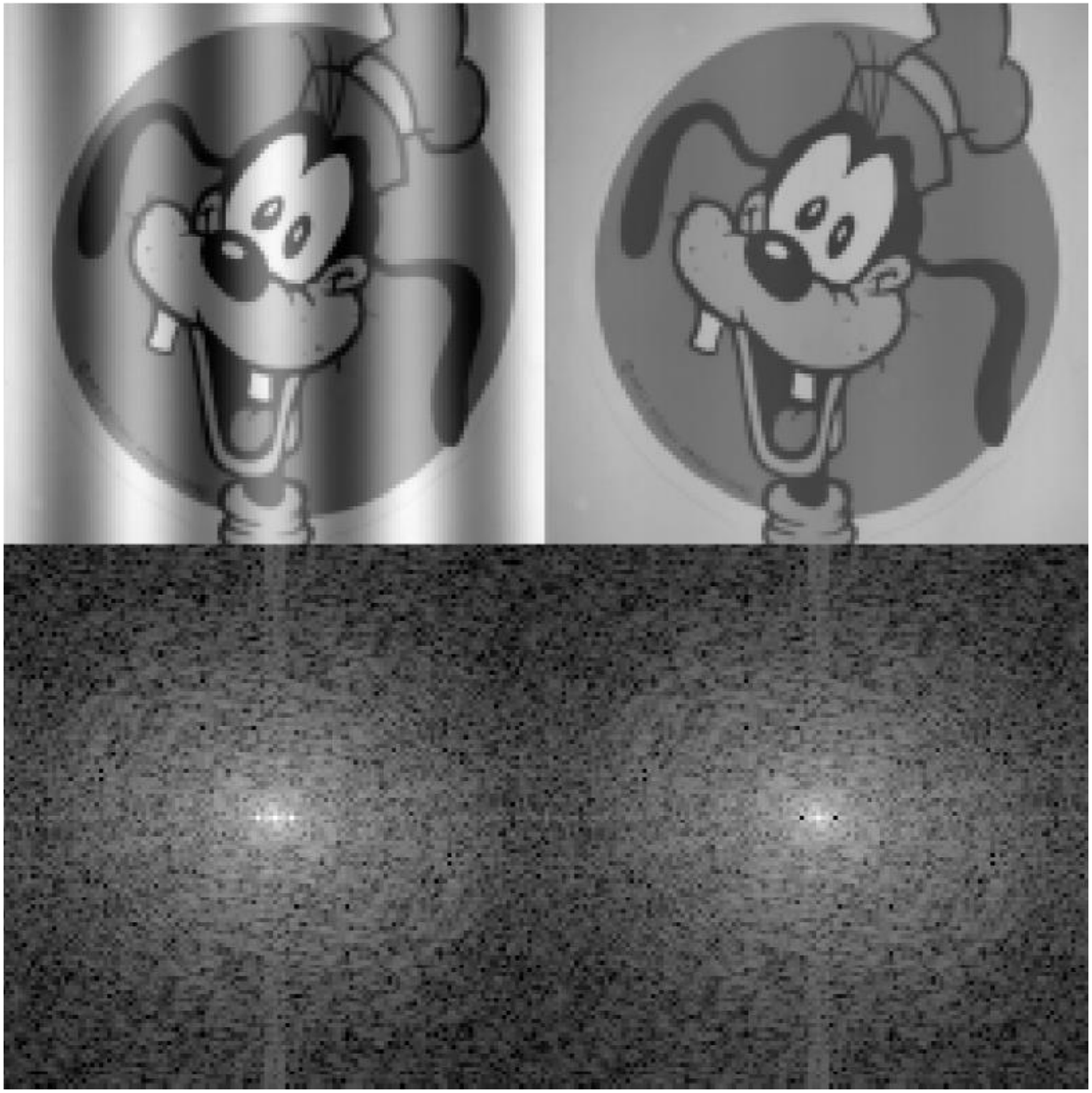
Slide: Hoiem

Now we can edit frequencies!

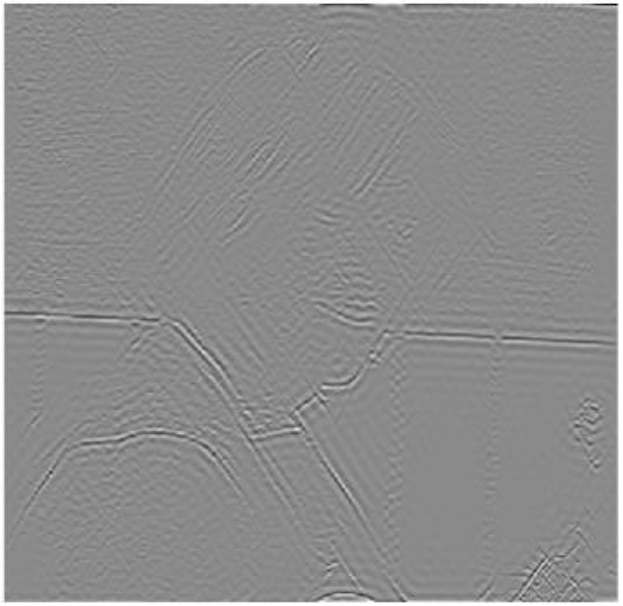
Low and High Pass filtering

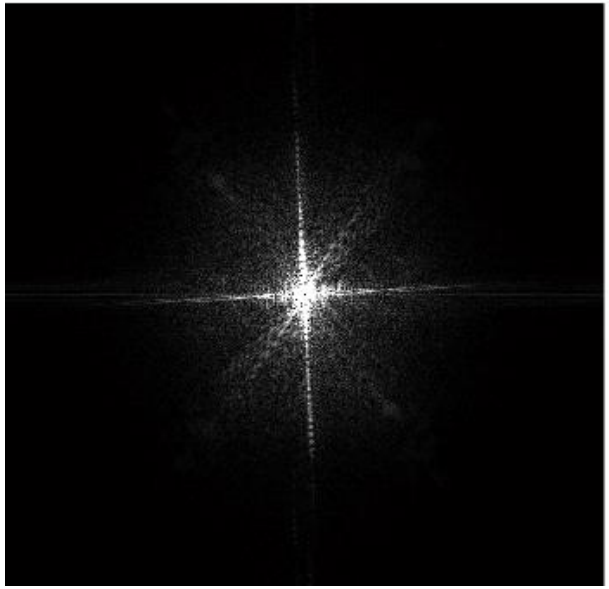


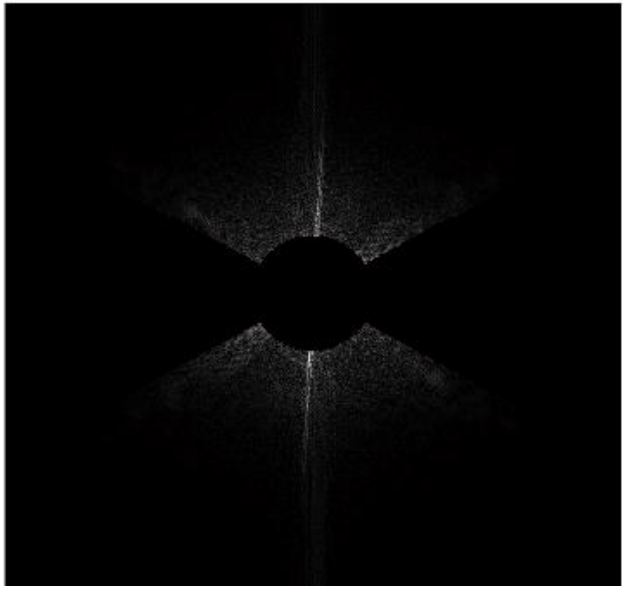
Removing frequency bands



High pass filtering + orientation

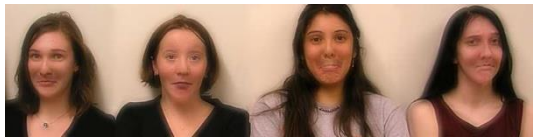






Application: Hybrid Images

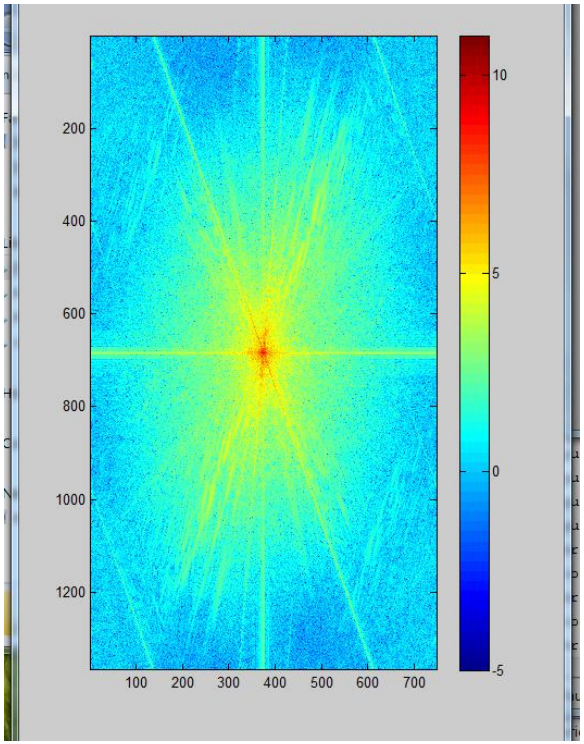
When we see an image from far away, we are effectively subsampling it!



A. Oliva, A. Torralba, P.G. Schyns, SIGGRAPH 2006

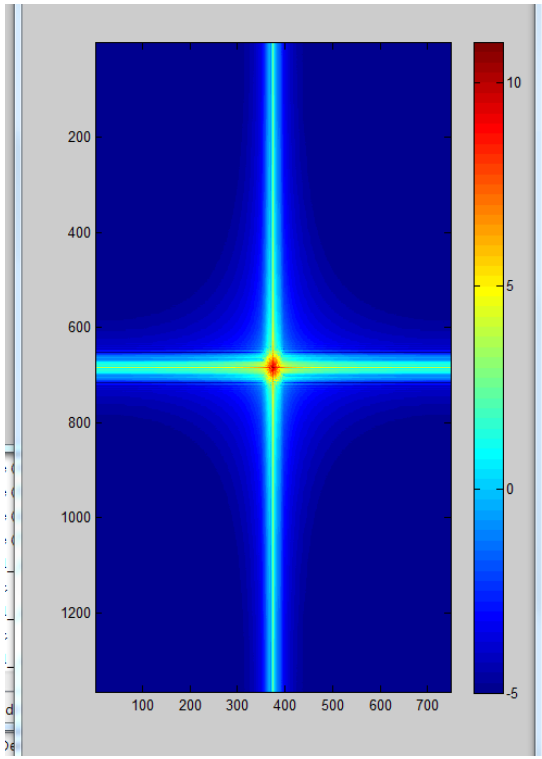
Hybrid Image in FFT

Hybrid Image



=

Low-passed Image



+

High-passed Image