Photograph by Frans Lanting

Slow mo guys – Saccades

- https://youtu.be/Fmg9ZOHESgQ?t=4s

Thinking in Frequency

Computer Vision

James Hays

Slides: Hoiem, Efros, and others

Recap of Wednesday

- Linear filtering is dot product at each position
  - Not a matrix multiplication
  - Can smooth, sharpen, translate (among many other uses)

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline\end{array}$$

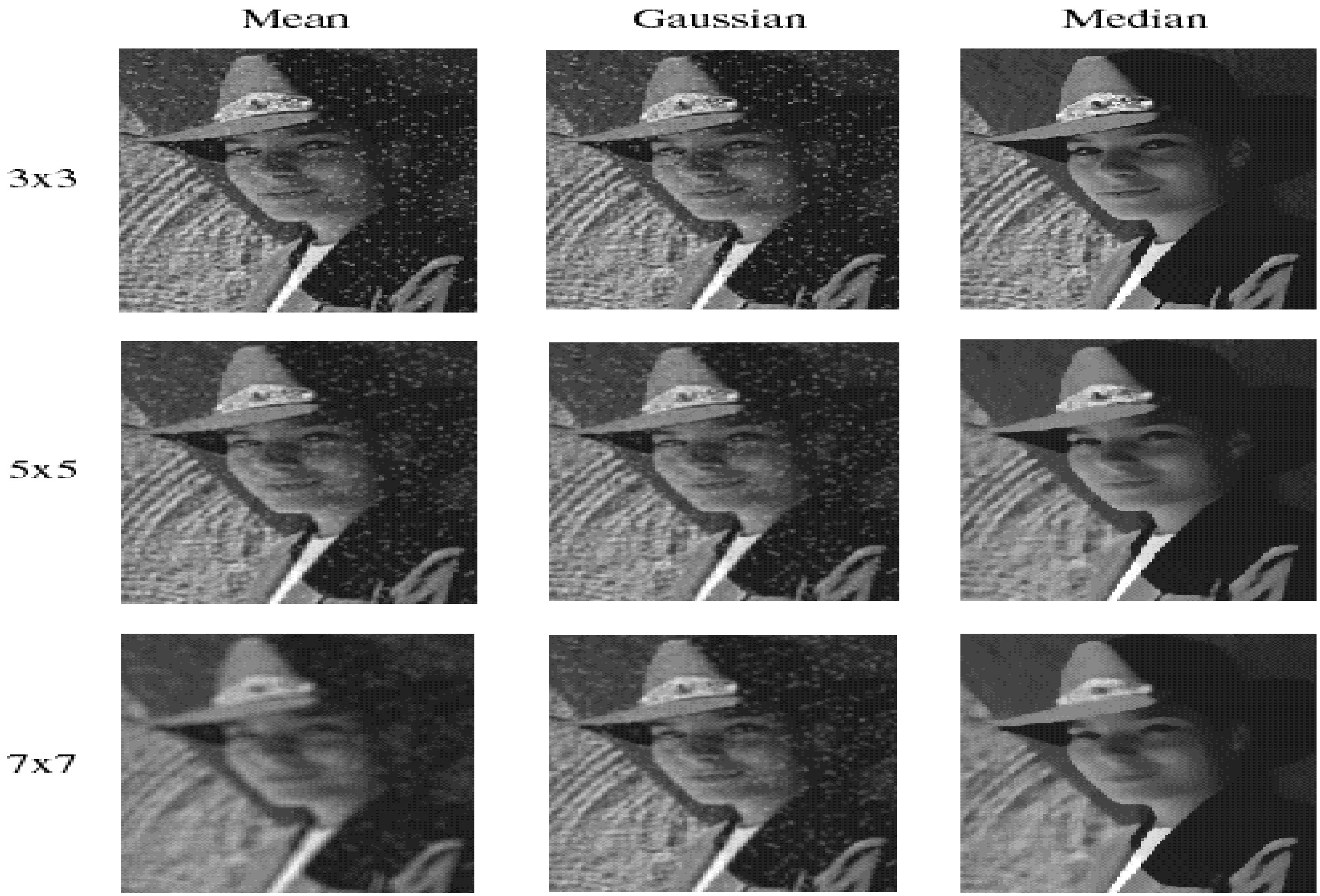
- Be aware of details for filter size, extrapolation, cropping

Median filters

- A **Median Filter** operates over a window by selecting the median intensity in the window.
- What advantage does a median filter have over a mean filter?
- Is a median filter a kind of convolution?

Slide by Steve Seitz

Comparison: salt and pepper noise



Slide by Steve Seitz

Review: questions

1. Write down a 3x3 filter that returns a positive value if the average value of the 4-adjacent neighbors is less than the center and a negative value otherwise

2. Write down a filter that will compute the gradient in the x-direction:

   ```
   gradx(y,x) = im(y,x+1)-im(y,x) for each x, y
   ```

Slide: Hoiem

Review: questions

Filtering Operator

3. Fill in the blanks:

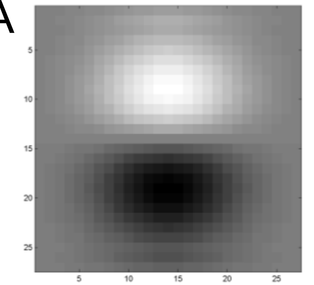
a) _ = D * B

b) A = _ * _

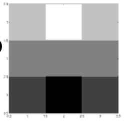
c) F = D * _

d) _ = D * D

A

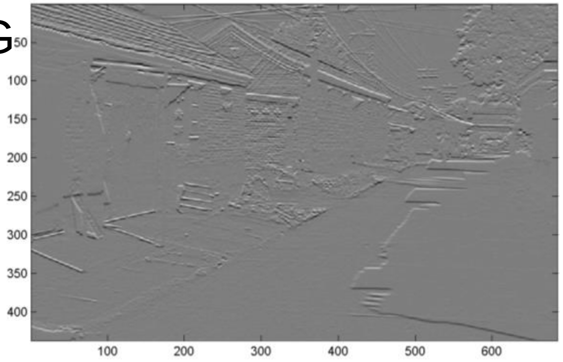


B

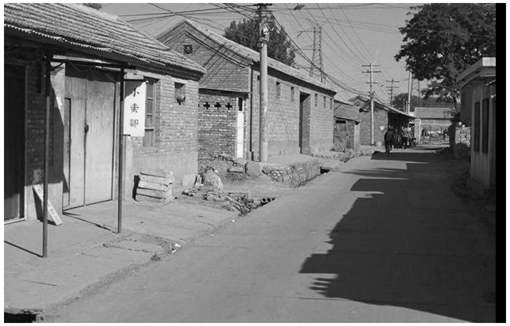


E

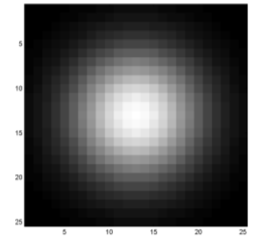
G



F

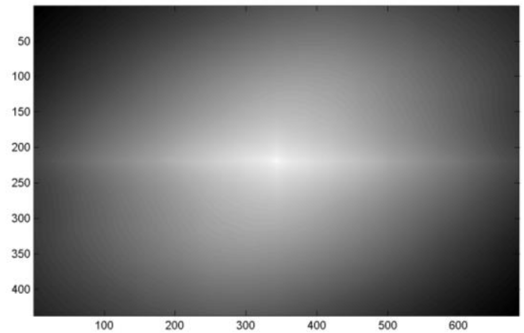


C

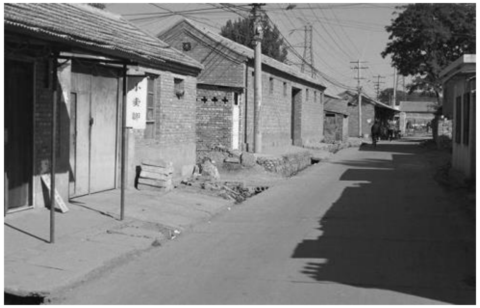


H

I



D



Slide: Hoiem

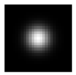
Today's Class

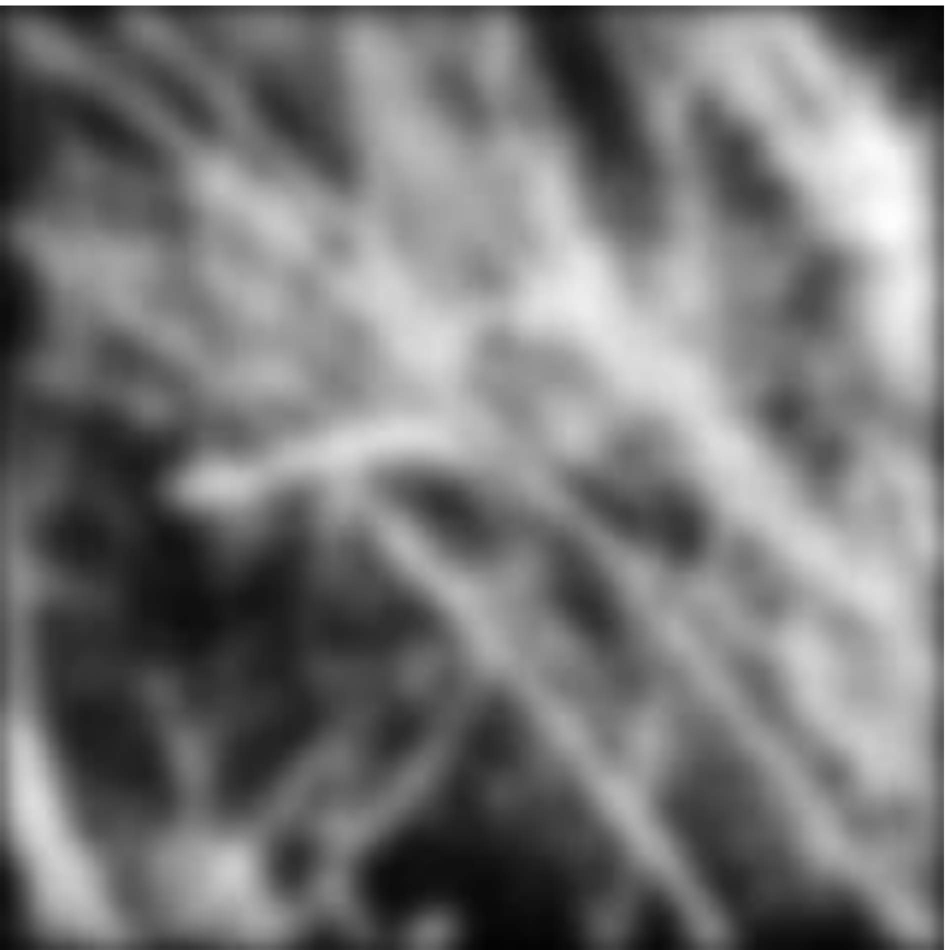
- Fourier transform and frequency domain
  - Frequency view of filtering
  - Hybrid images
  - Sampling
- Reminder: Read your textbook
  - Today's lecture covers material in 3.4

Slide: Hoiem

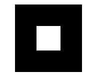
Why does the Gaussian give a nice smooth image, but the square filter give edgy artifacts?

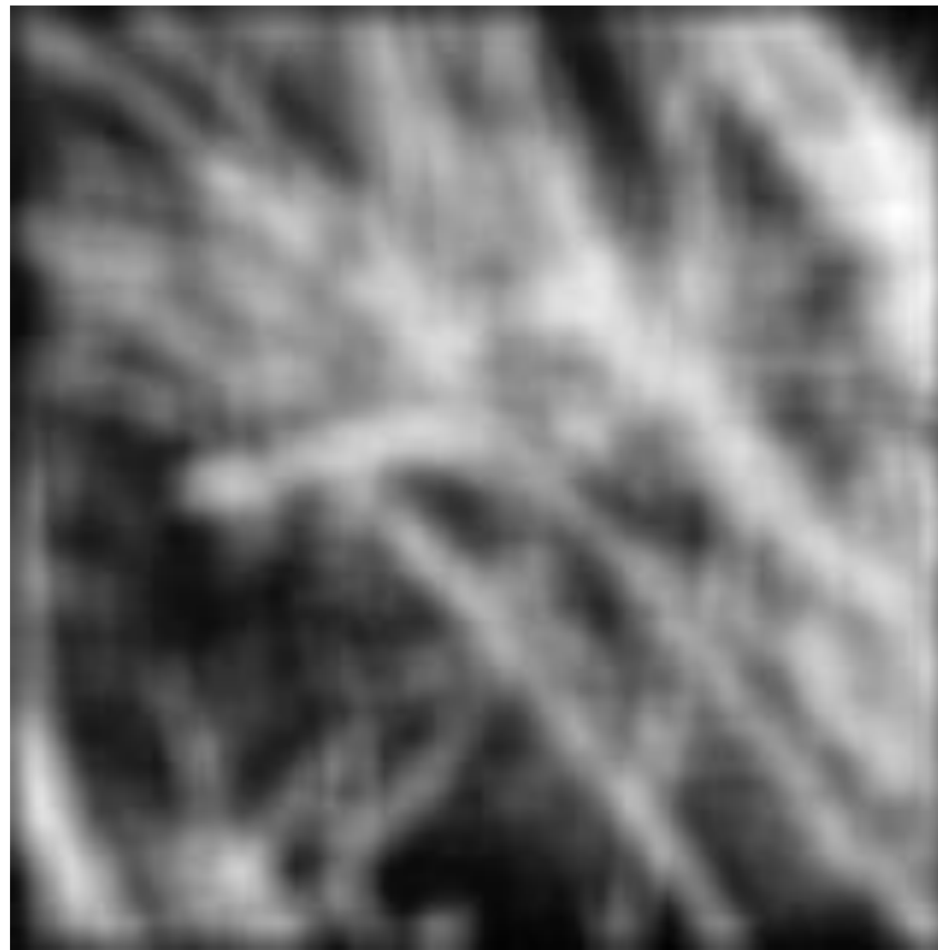
Gaussian



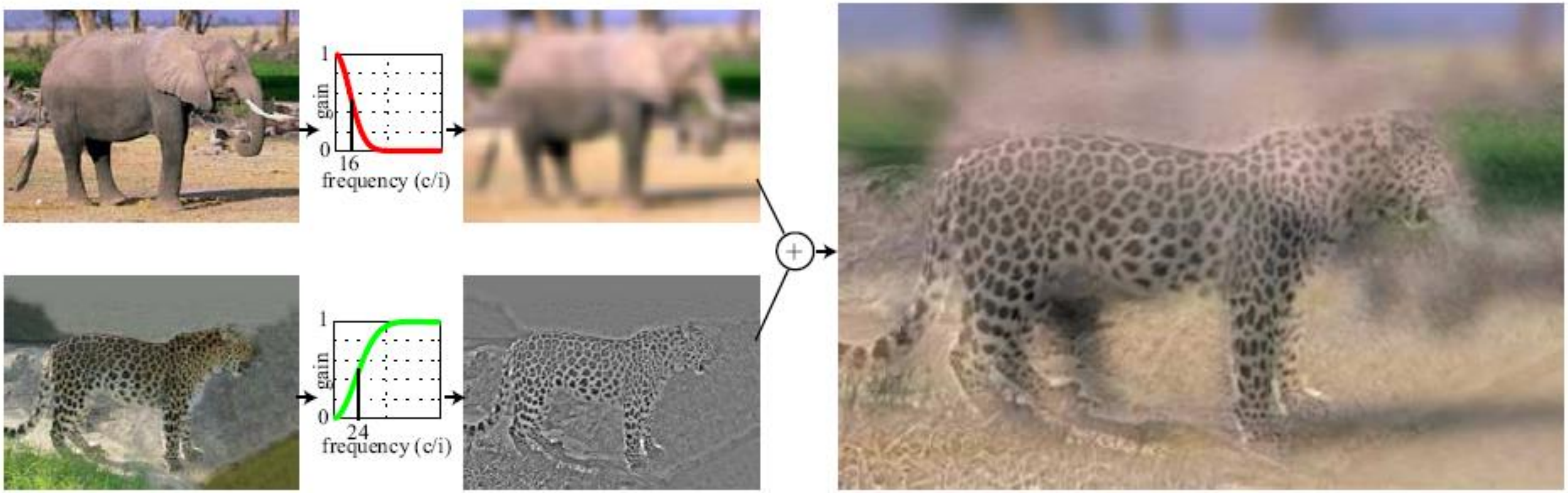


Box filter





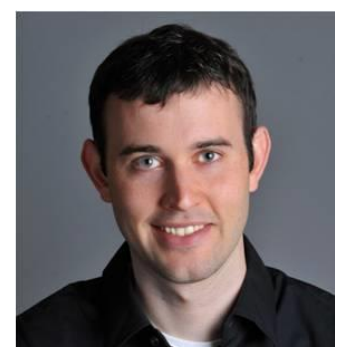
Hybrid Images



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

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

?



Slide: Hoiem

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

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

Thinking in terms of frequency

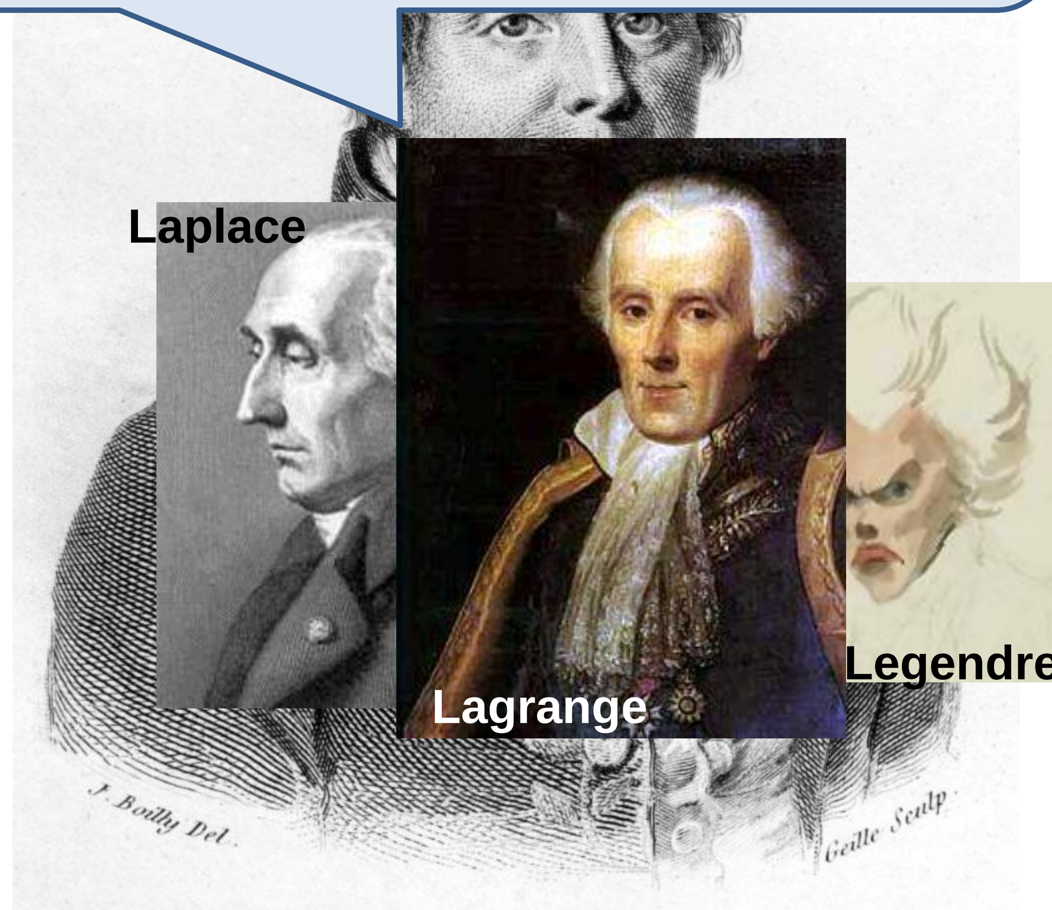
Jean Baptiste Joseph Fourier (1768-1830)

had crazy idea (1807):

> ***Any** univariate function can*
> *rewritten as a weighted sum*
> *sines and cosines of 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
  - there are some subtle restrictions



Fourier, Joseph (1768-1830)

French mathematician who discovered that any periodic motion can be written as a superposition of sinusoidal and cosinusoidal vibrations. He developed a mathematical theory of heat in *Théorie Analytique de la Chaleur (Analytic Theory of Heat)*, (1822), discussing it in terms of differential equations.

Fourier was a friend and advisor of Napoleon. Fourier believed that his health would be improved by wrapping himself up in blankets, and in this state he tripped down the stairs in his house and killed himself. The paper of Galois which he had taken home to read shortly before his death was never recovered.

SEE ALSO: Galois

Additional biographies: MacTutor (St. Andrews), Bonn

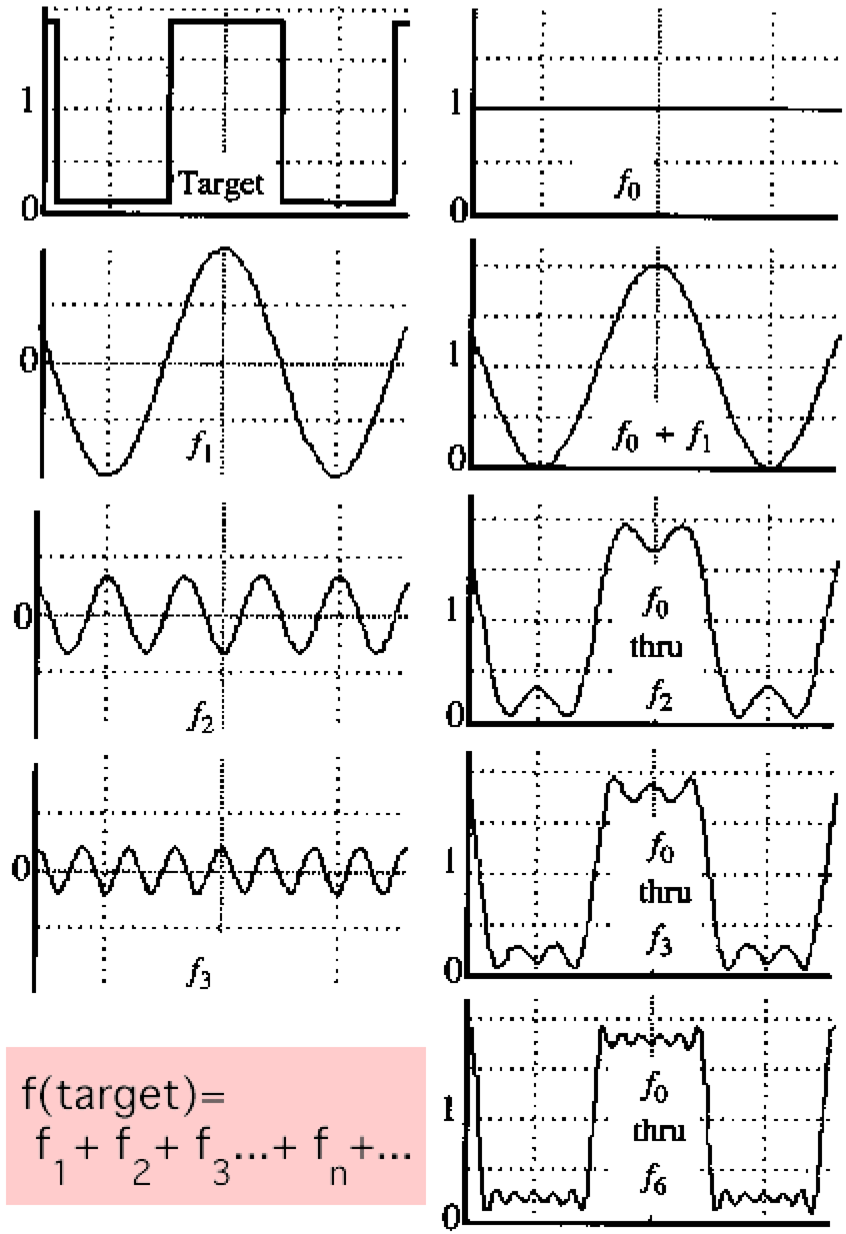
How would math have changed if the Slanket or Snuggie had been invented?

A sum of sines

Our building block:

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

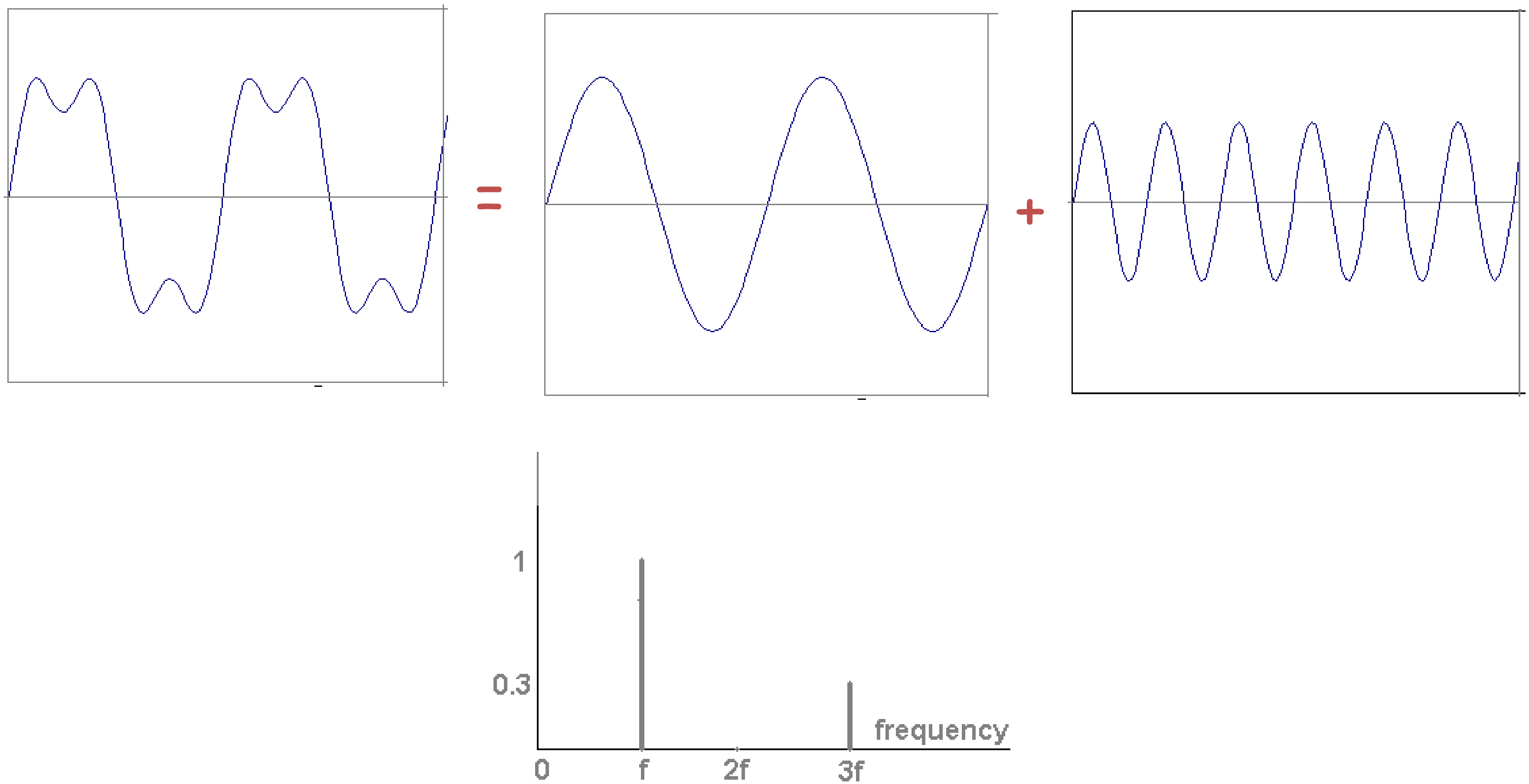
Add enough of them to get any signal *g(x)* you want!



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

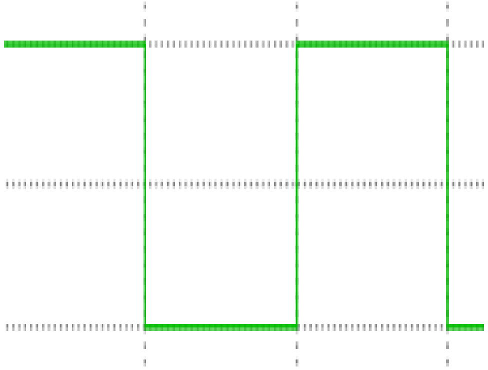
Frequency Spectra

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

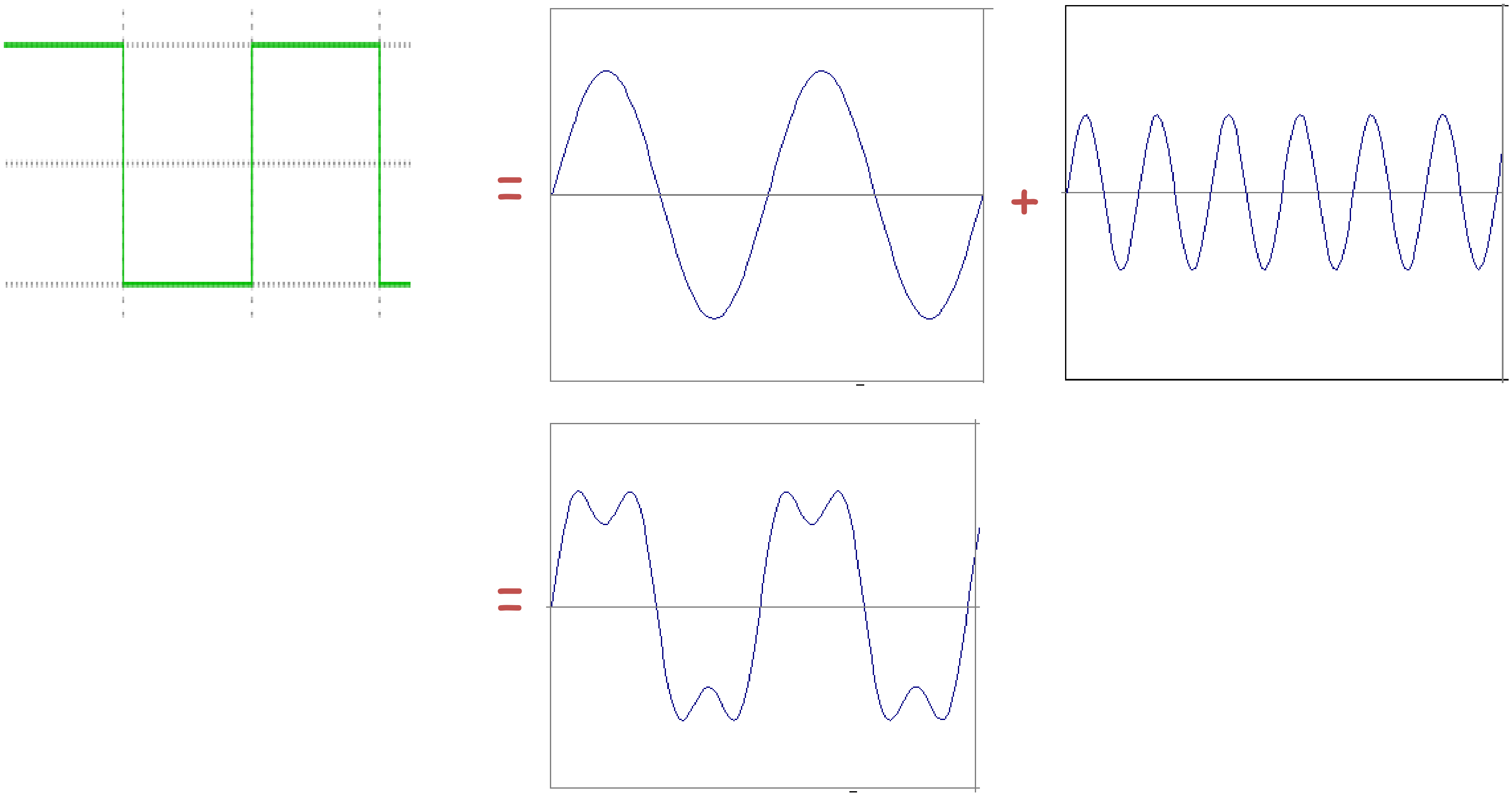


Slides: Efros

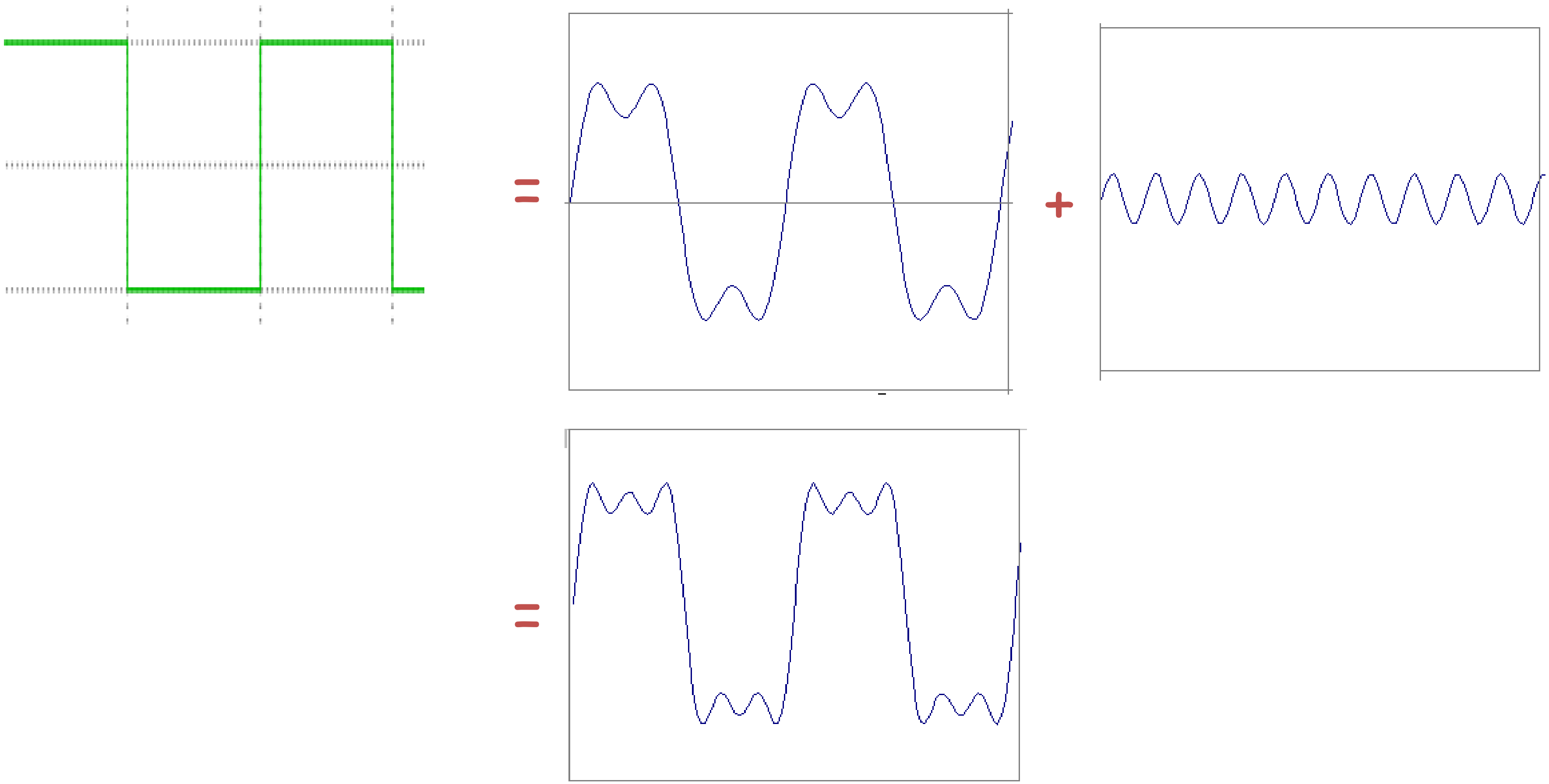
Frequency Spectra



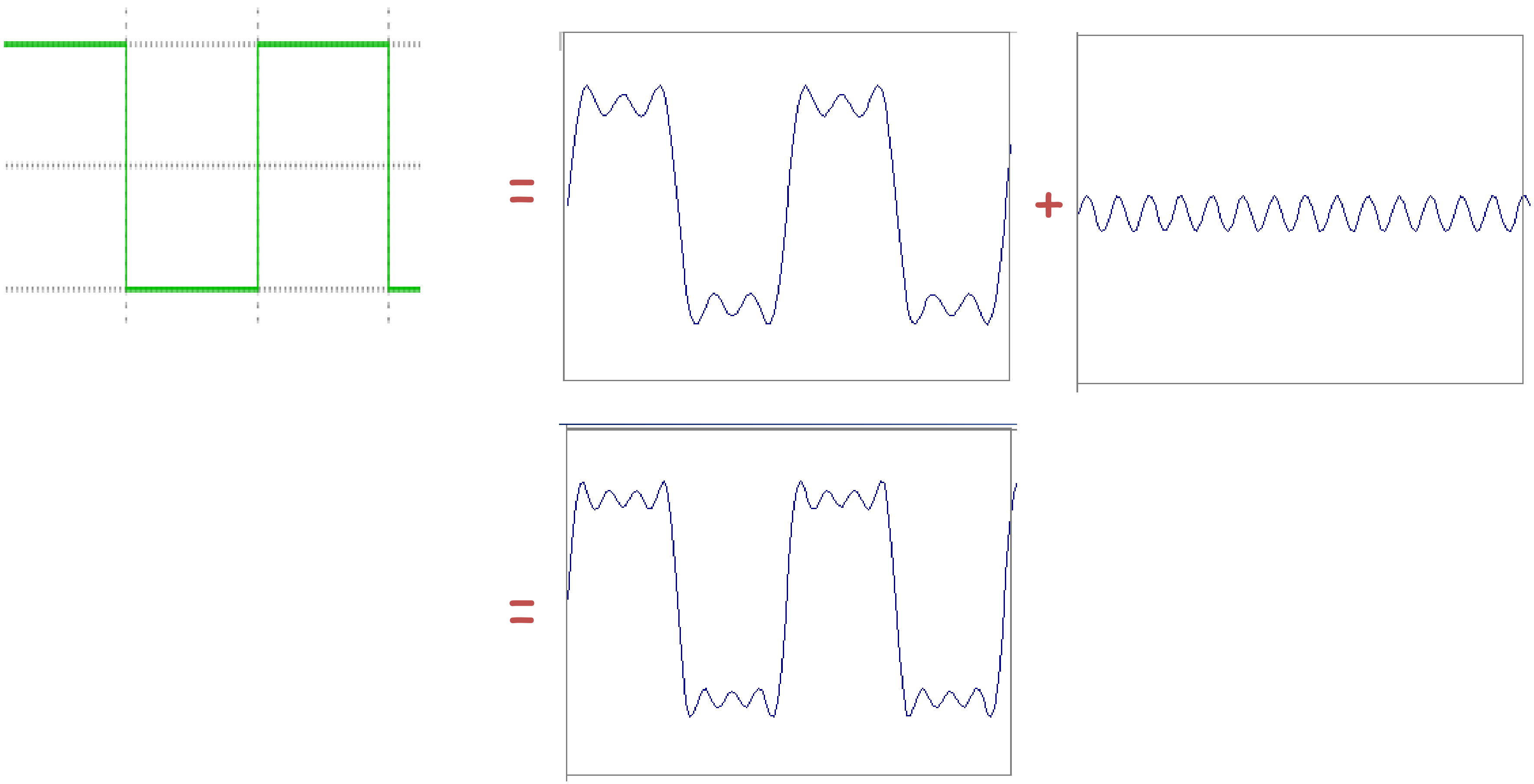
Frequency Spectra



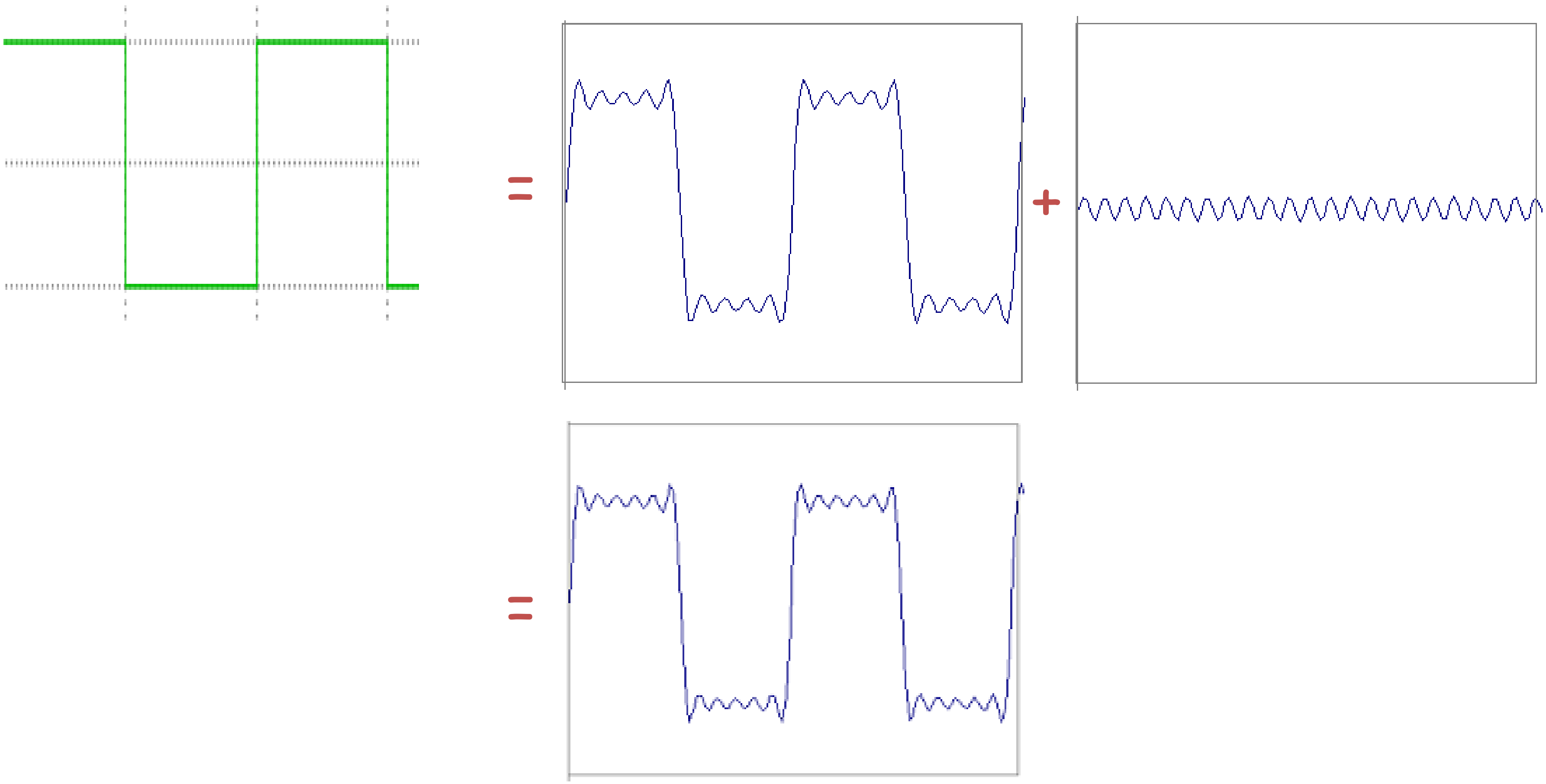
Frequency Spectra



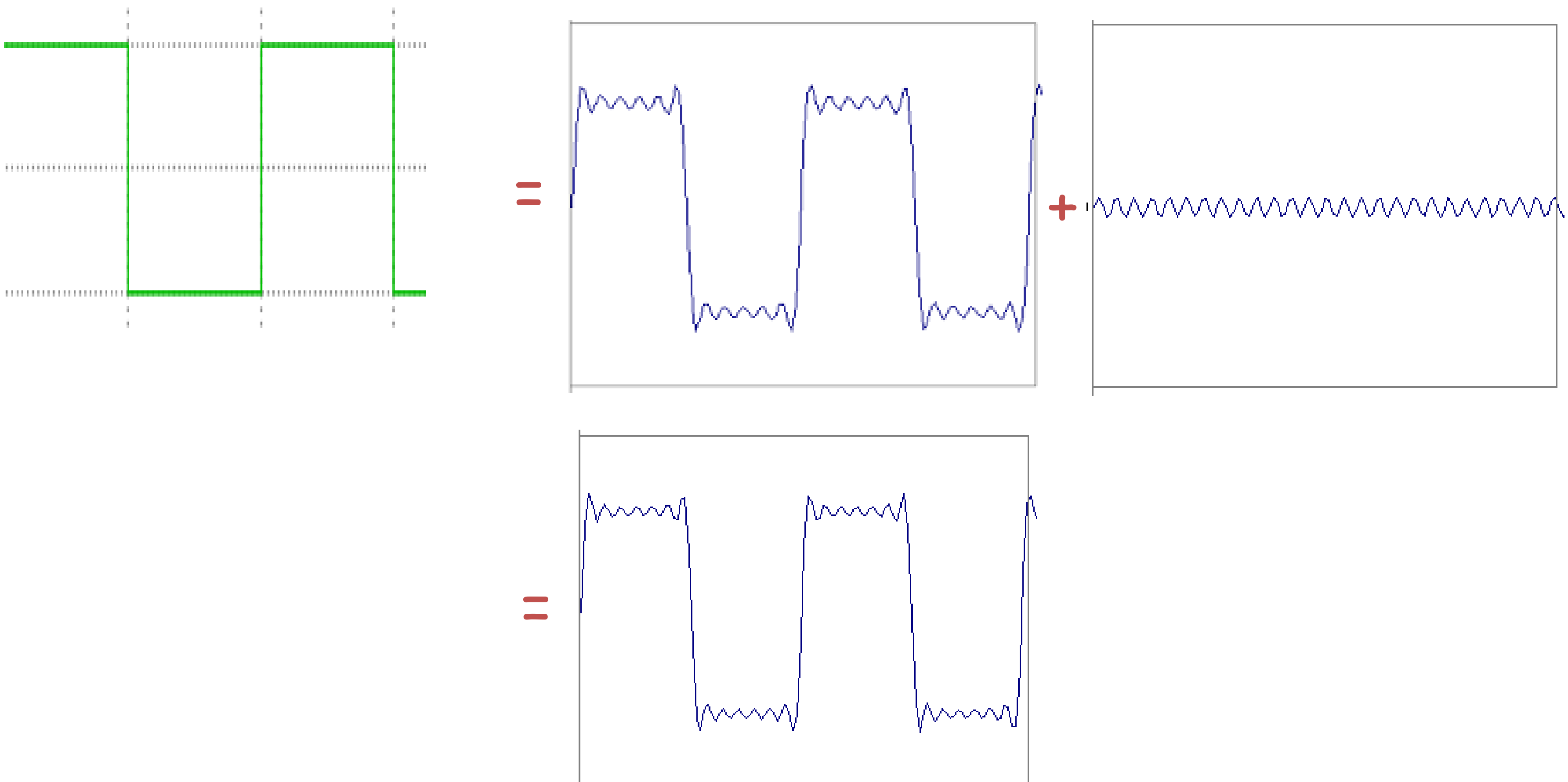
Frequency Spectra



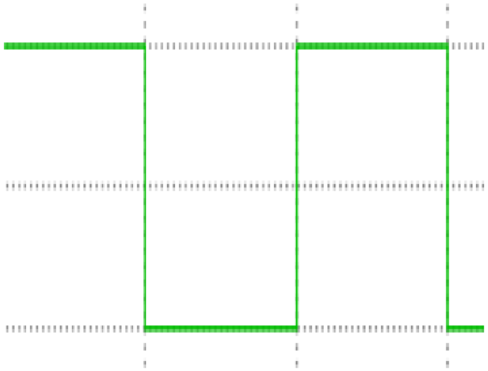
Frequency Spectra



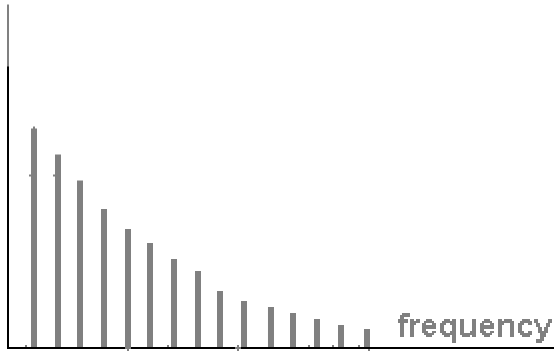
Frequency Spectra



Frequency Spectra

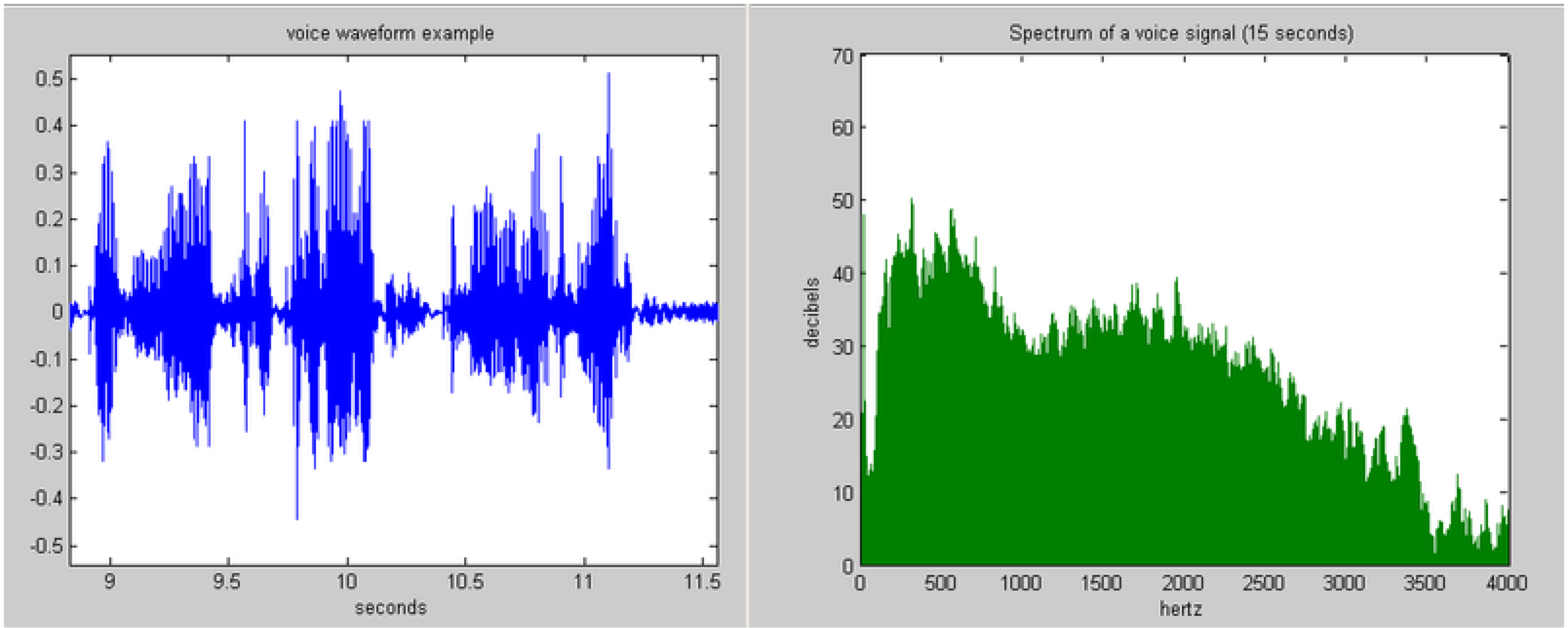


$$= \boxed{A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)}$$



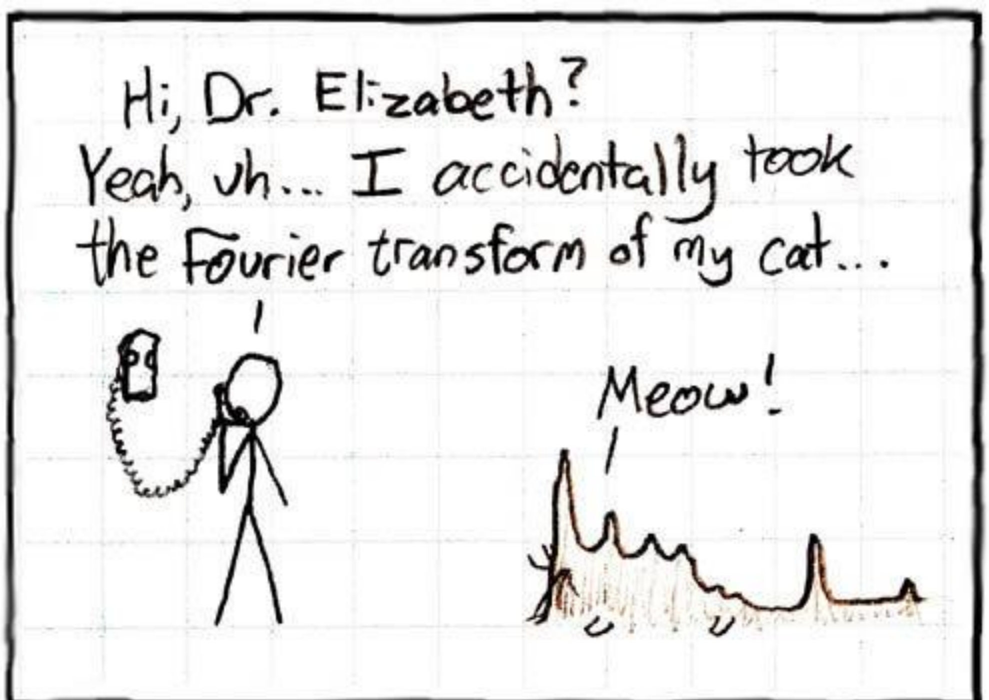
Example: Music

- 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

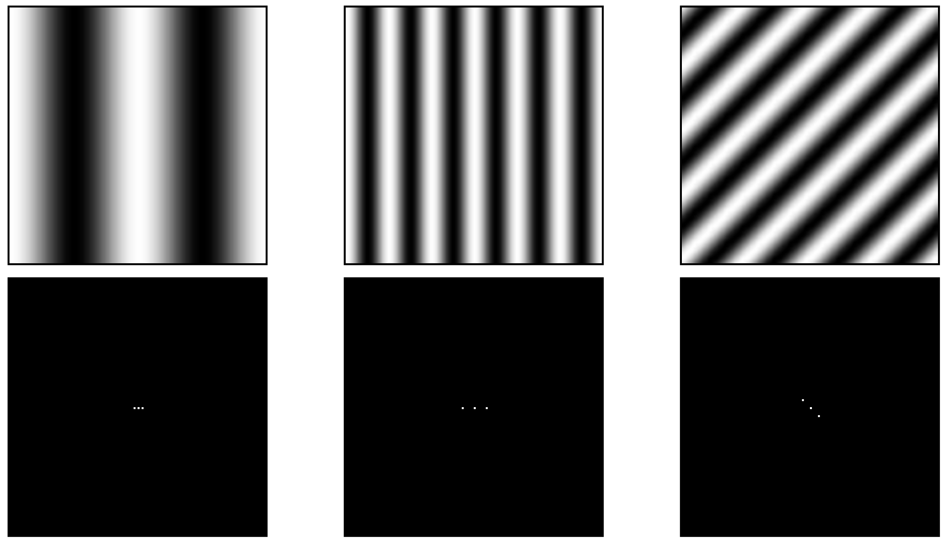


xkcd.com

Fourier analysis in images

Intensity Image

Fourier Image



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

Fourier Transform

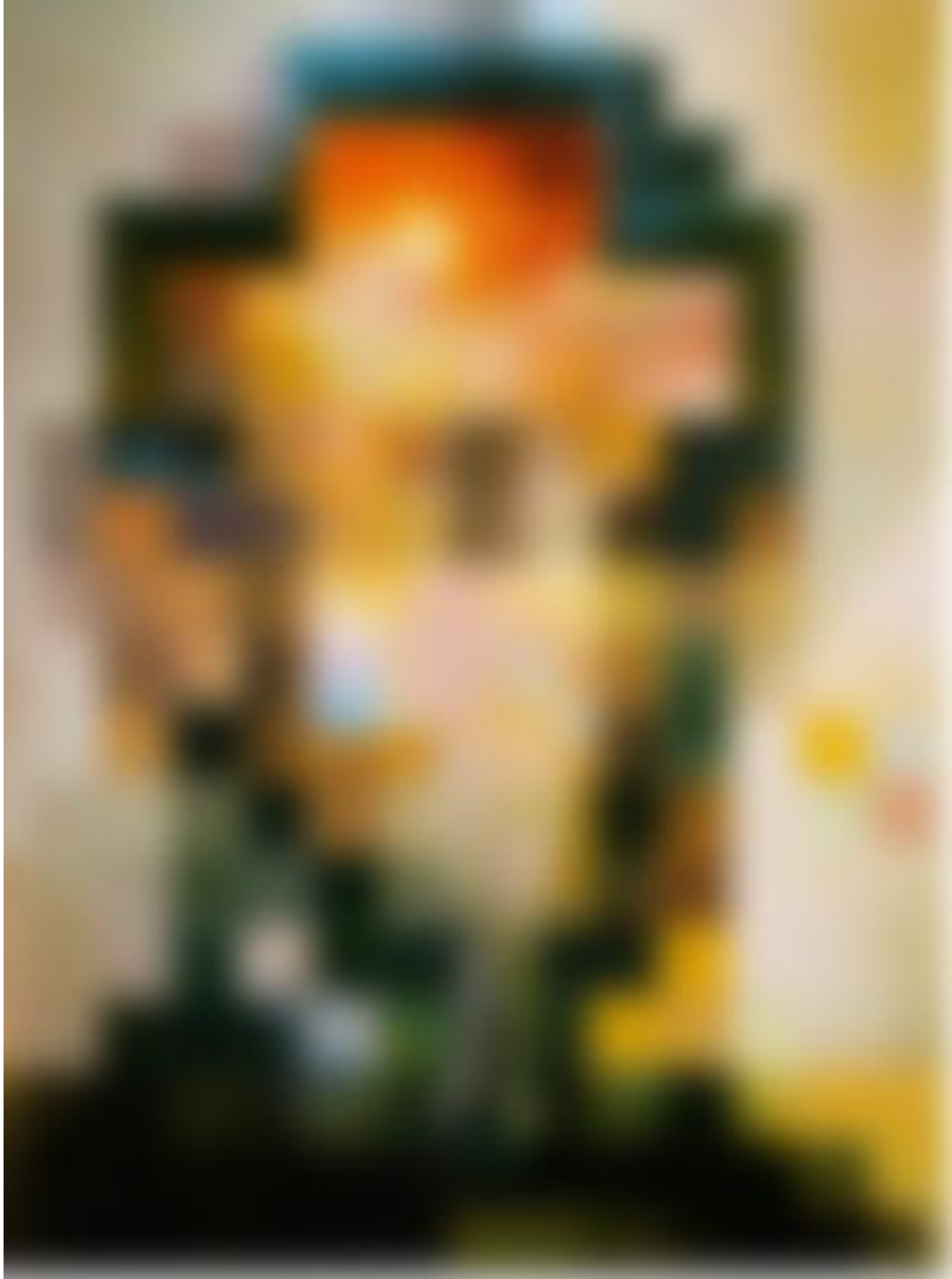
- Fourier transform stores the magnitude and phase at each frequency
  - Magnitude encodes how much signal there is at a particular frequency
  - Phase encodes spatial information (indirectly)
  - For mathematical convenience, this is often notated in terms of real and complex numbers

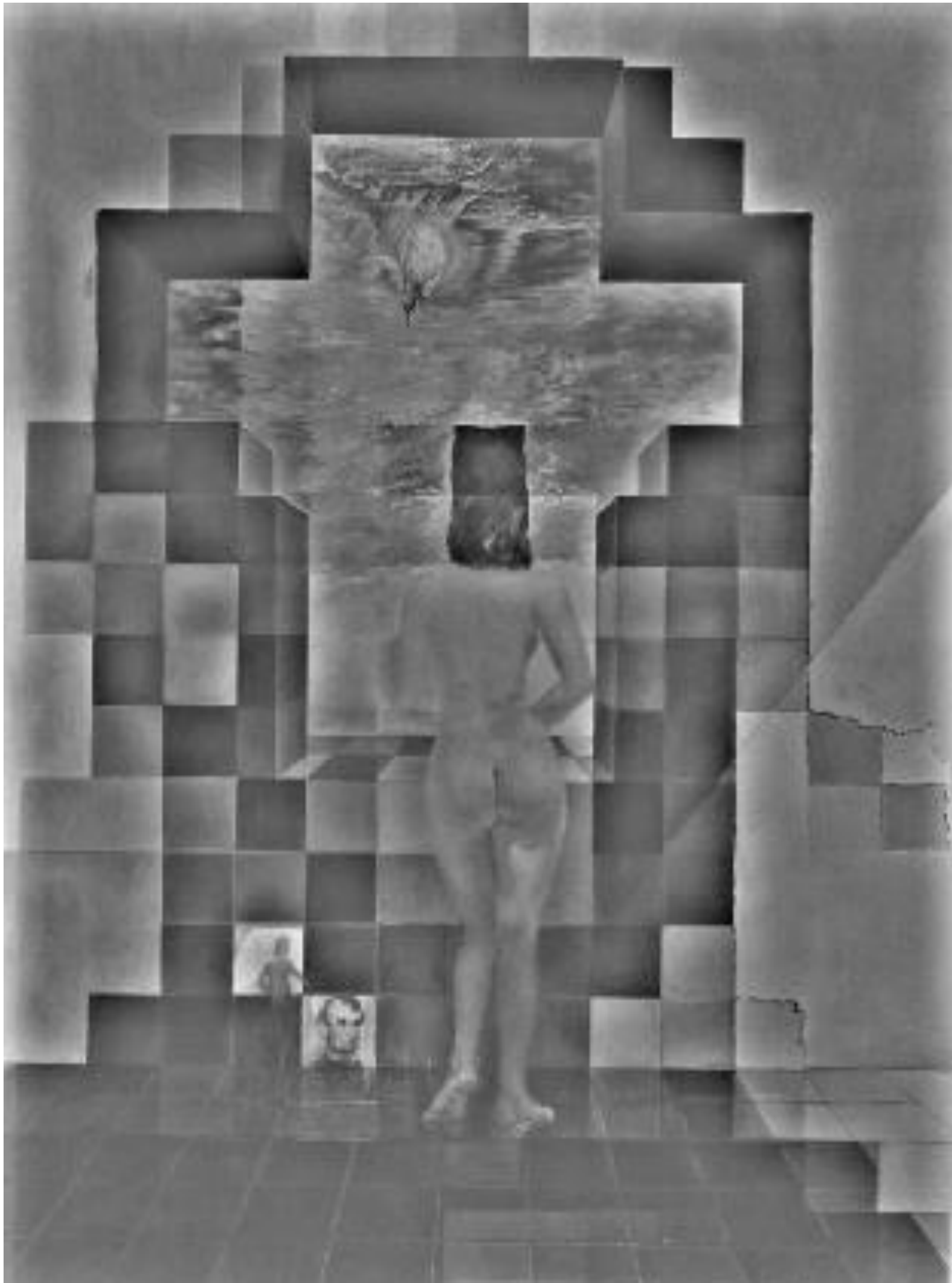
Amplitude: $A = \pm\sqrt{R(\omega)^2 + I(\omega)^2}$

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

Salvador Dali invented Hybrid Images?

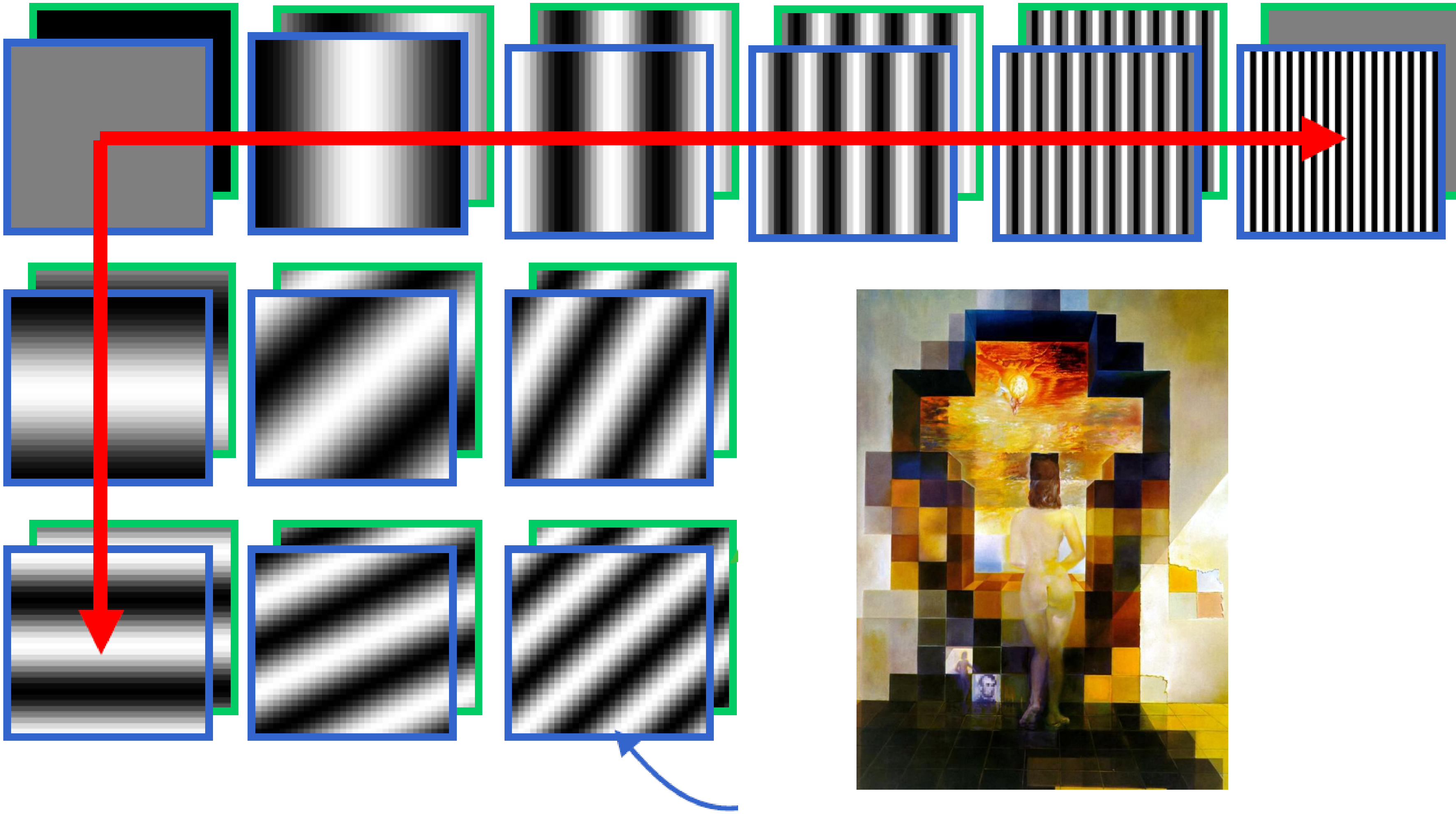
Salvador Dali
"Gala Contemplating the Mediterranean Sea, which at 30 meters becomes the portrait of Abraham Lincoln", 1976





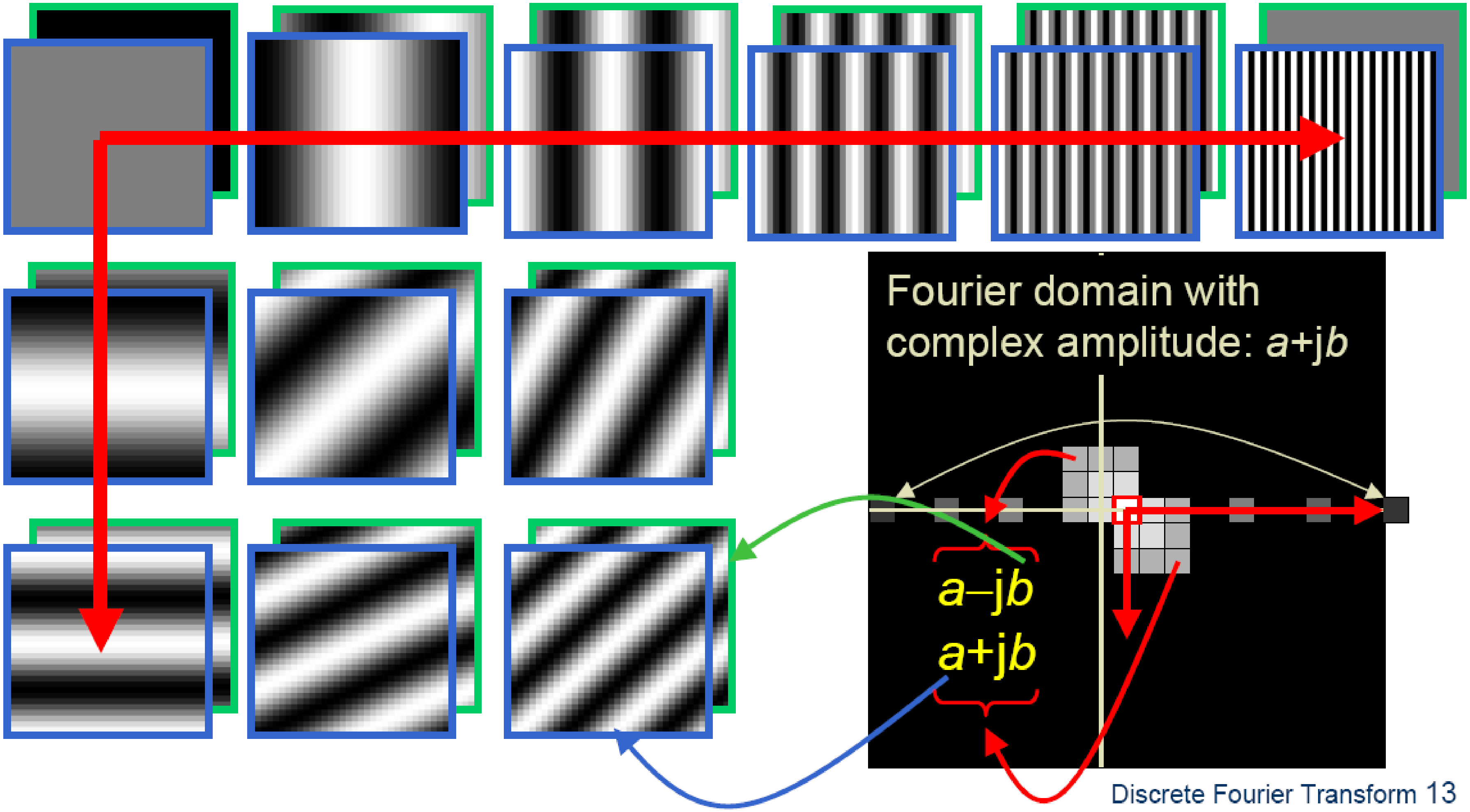
Fourier Bases

Teases away fast vs. slow changes in the image.



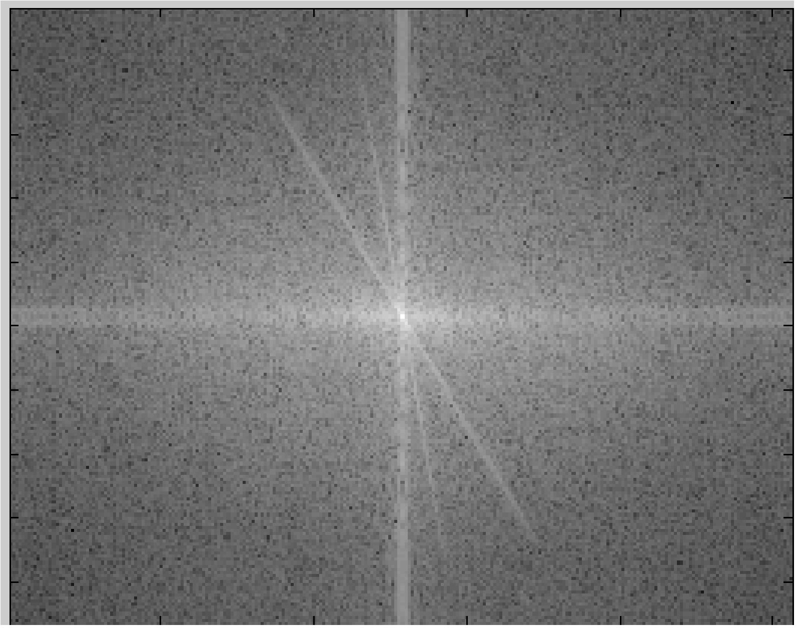
This change of basis is the Fourier Transform

Fourier Bases



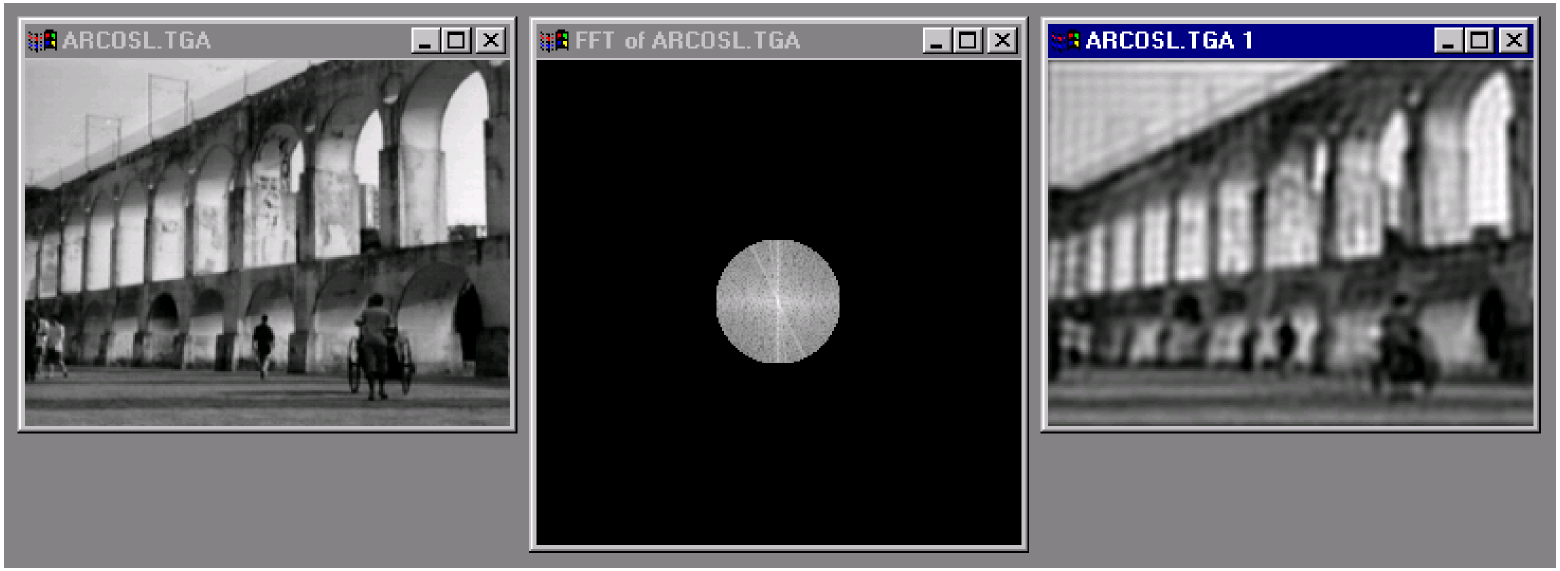
in Matlab, check out: imagesc(log(abs(fftshift(fft2(im)))));

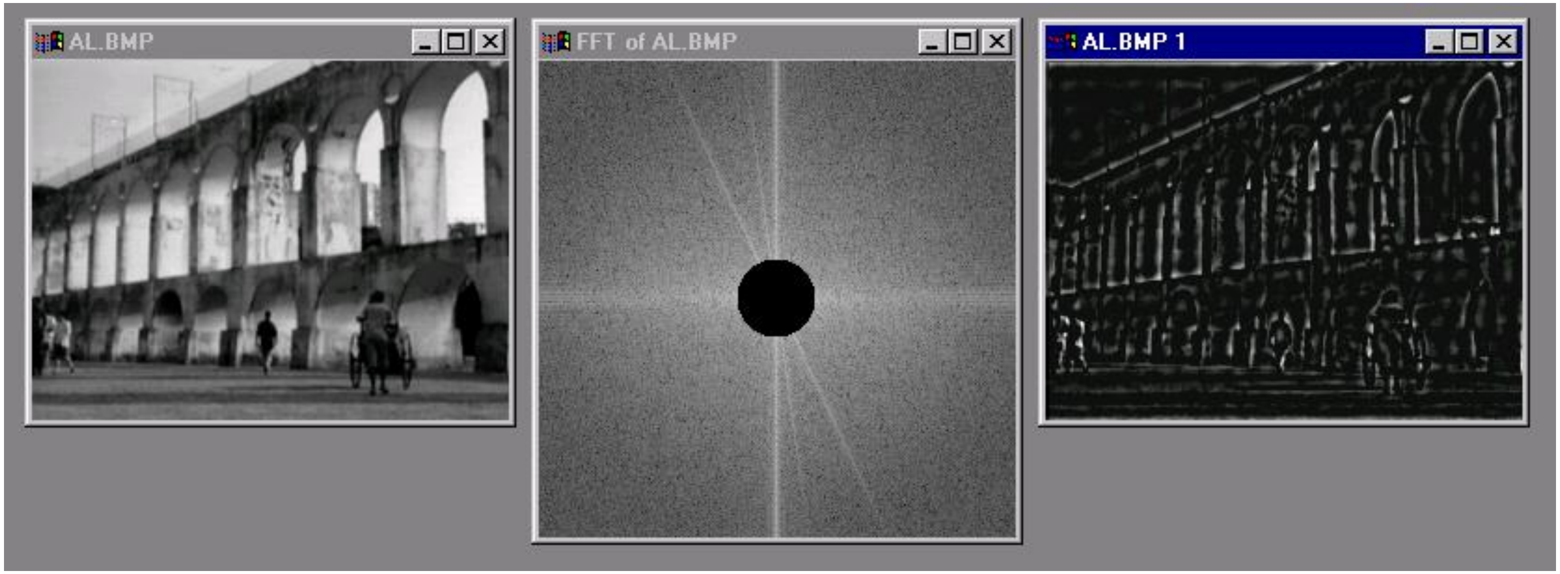
Man-made Scene



Can change spectrum, then reconstruct

Low and High Pass filtering





The Convolution Theorem

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

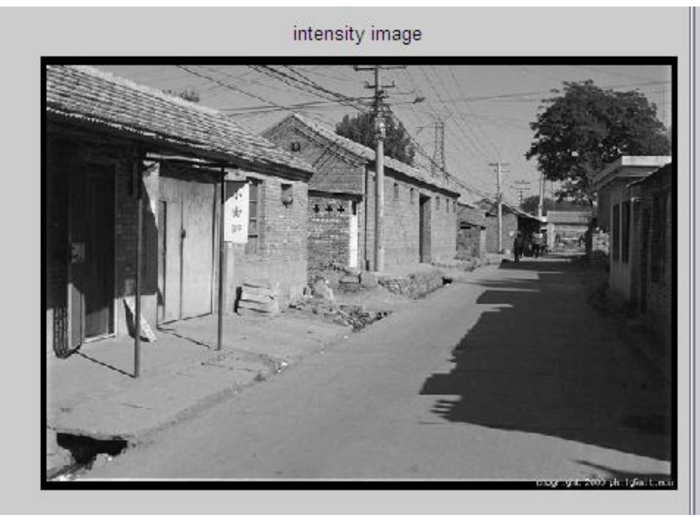
$$\mathrm{F}[g * h] = \mathrm{F}[g]\,\mathrm{F}[h]$$

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

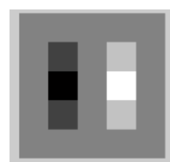
$$g * h = \mathrm{F}^{-1}[\mathrm{F}[g]\,\mathrm{F}[h]]$$

Filtering in spatial domain

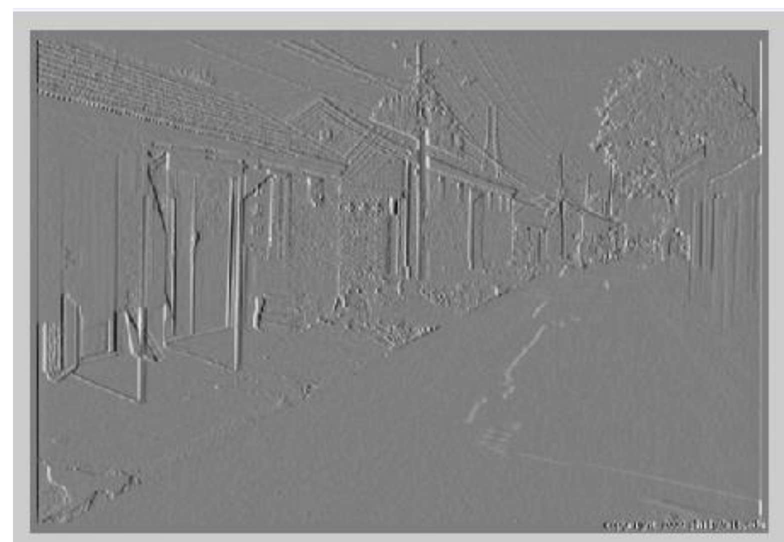
1	0	-1
2	0	-2
1	0	-1



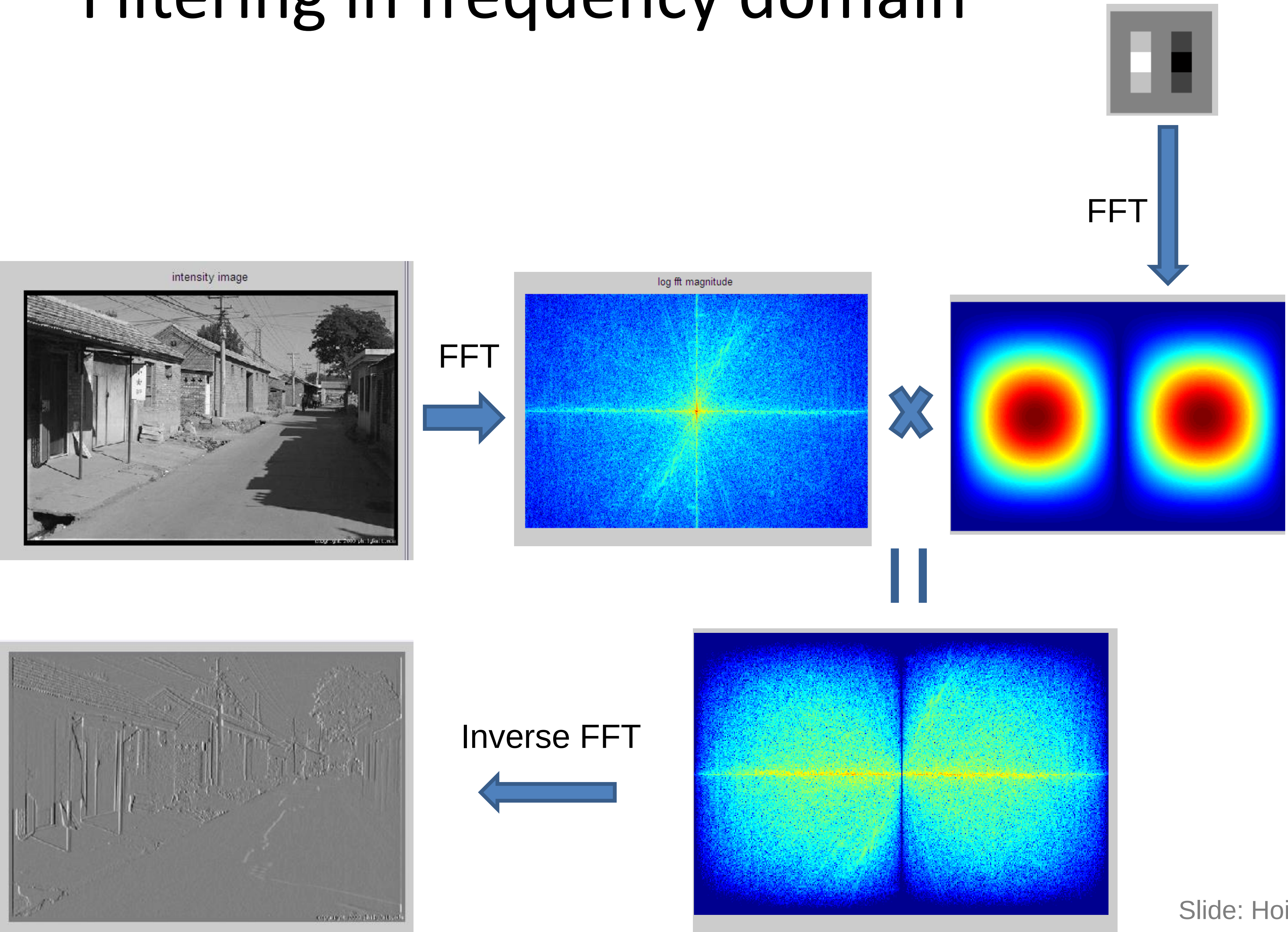
*



=



Filtering in frequency domain



Slide: Hoiem

Fourier Matlab demo

FFT in Matlab

- Filtering with fft

```
im = double(imread('…'))/255;
im = rgb2gray(im); % "im" should be a gray-scale floating point image
[imh, imw] = size(im);

hs = 50; % filter half-size
fil = fspecial('gaussian', hs*2+1, 10);

fftsize = 1024; % should be order of 2 (for speed) and include padding
im_fft  = fft2(im,  fftsize, fftsize);                          % 1) fft im with padding
fil_fft = fft2(fil, fftsize, fftsize);                          % 2) fft fil, pad to same size as
image
im_fil_fft = im_fft .* fil_fft;                                 % 3) multiply fft images
im_fil = ifft2(im_fil_fft);                                     % 4) inverse fft2
im_fil = im_fil(1+hs:size(im,1)+hs, 1+hs:size(im, 2)+hs); % 5) remove padding
```

- Displaying with fft

```
figure(1), imagesc(log(abs(fftshift(im_fft)))), axis image, colormap jet
```

Slide: Hoiem