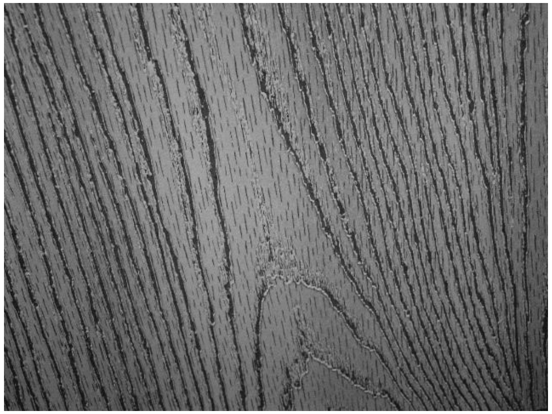
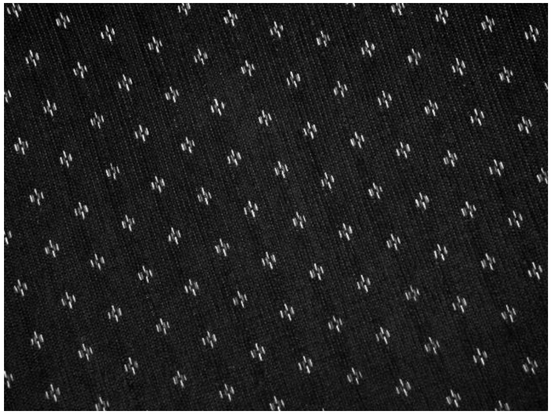
Application: Representing Texture

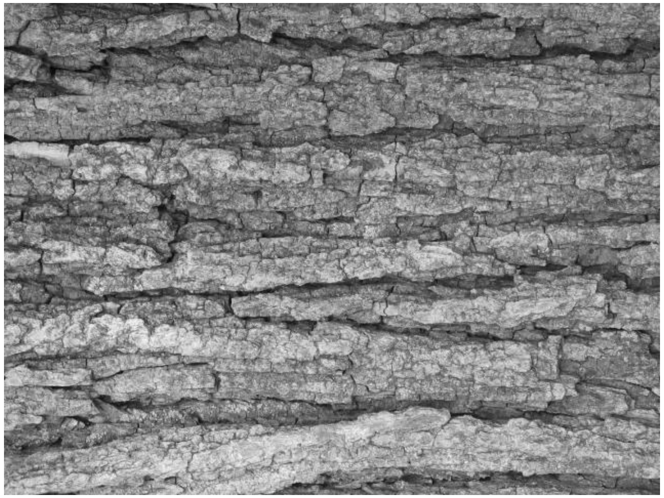
Source: Forsyth

Texture and Material

http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database/samples/

Texture and Orientation



http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database/samples/

Texture and Scale

http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database/samples/

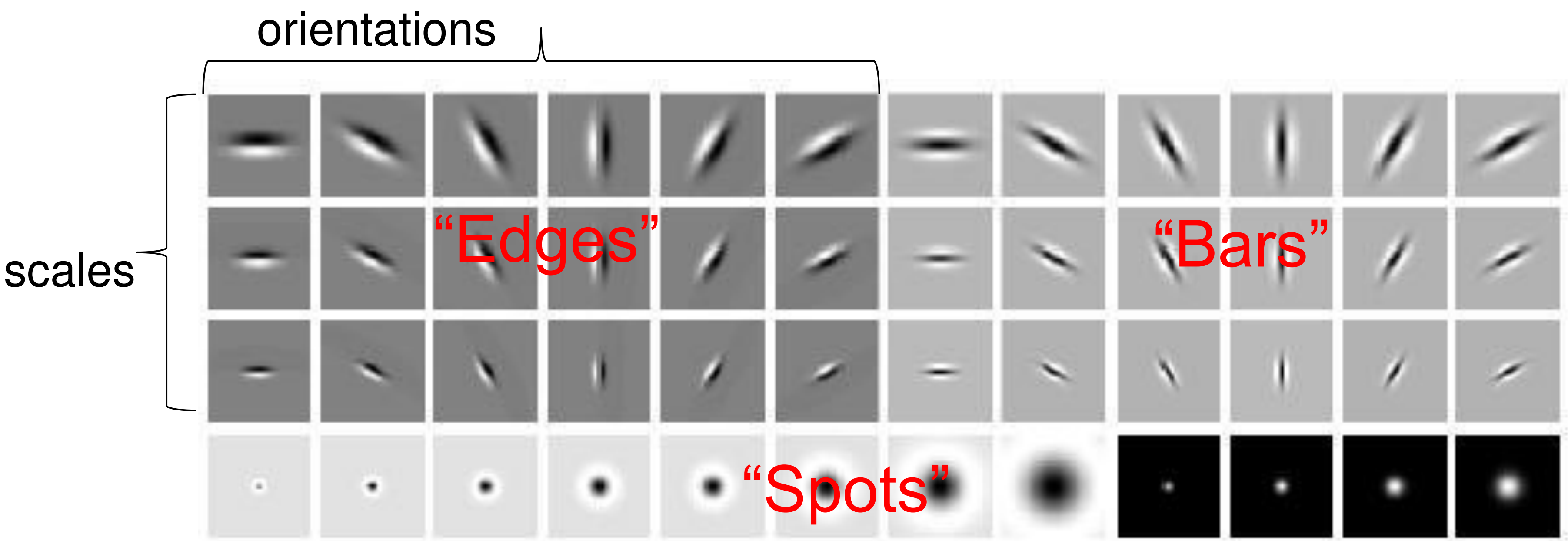
What is texture?

Regular or stochastic patterns caused by bumps, grooves, and/or markings

How can we represent texture?

- Compute responses of blobs and edges at various orientations and scales

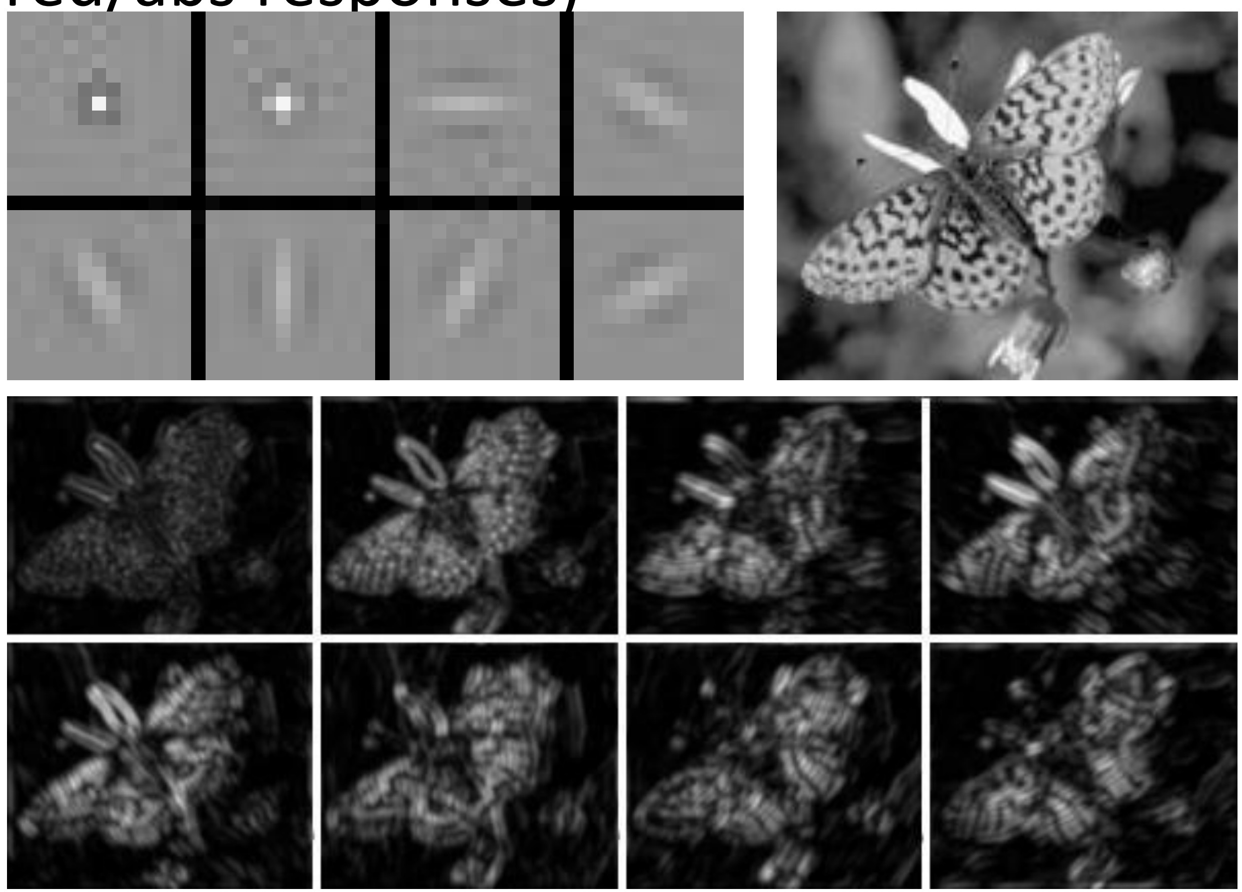
Overcomplete representation: filter banks



Code for filter banks: *www.robots.ox.ac.uk/~vgg/research/texclass/filters.html*

Filter banks

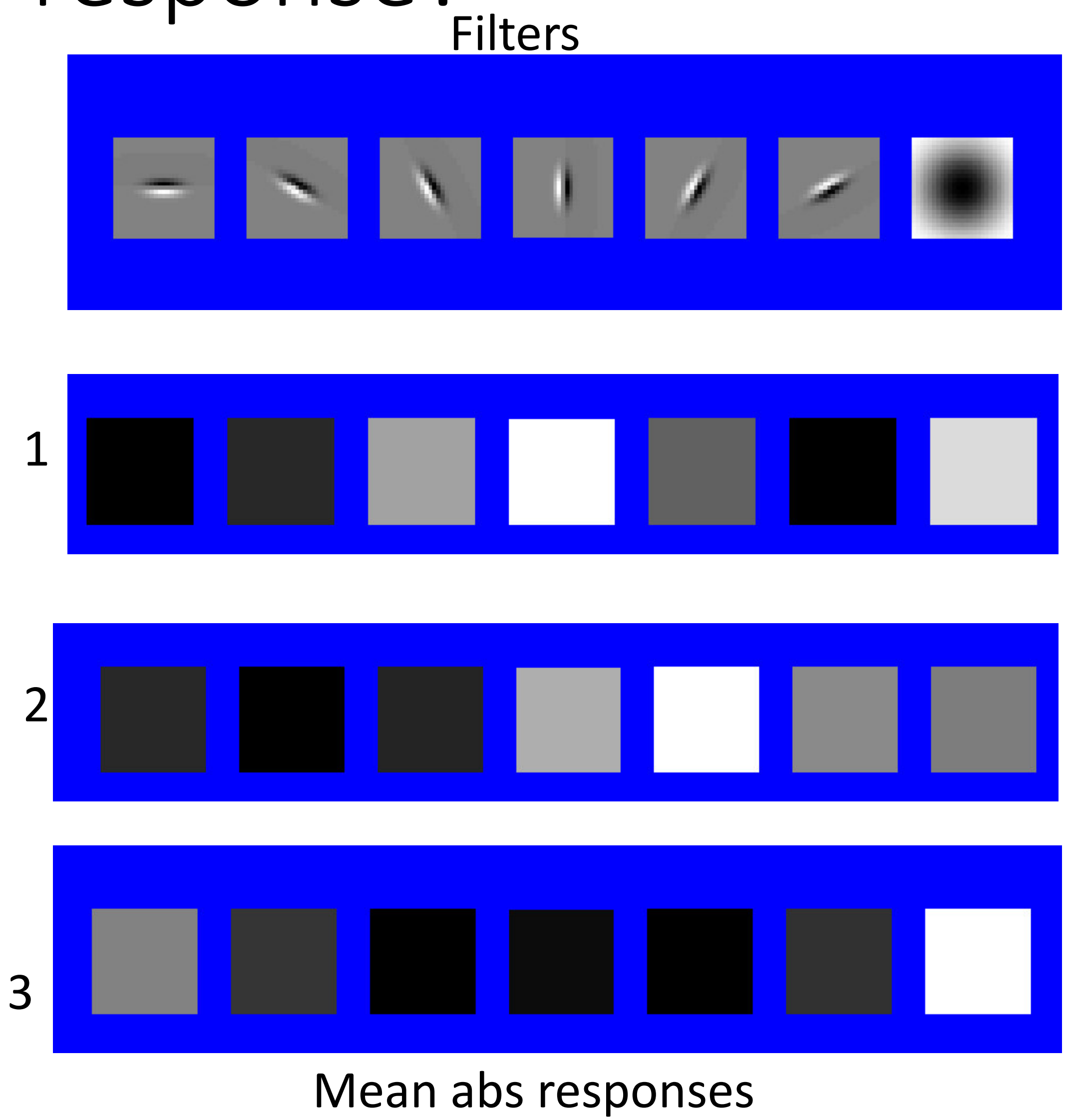
- Process image with each filter and keep responses (or squared/abs responses)



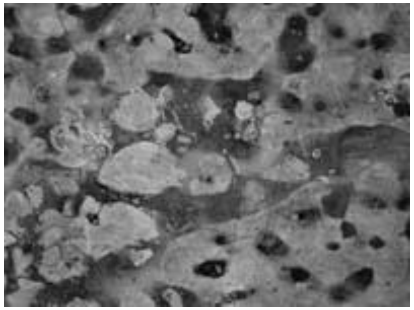
How can we represent texture?

- Measure responses of blobs and edges at various orientations and scales
- Idea 1: Record simple statistics (e.g., mean, std.) of absolute filter responses

Can you match the texture to the response?

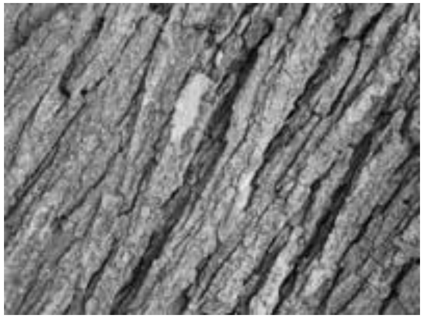


A

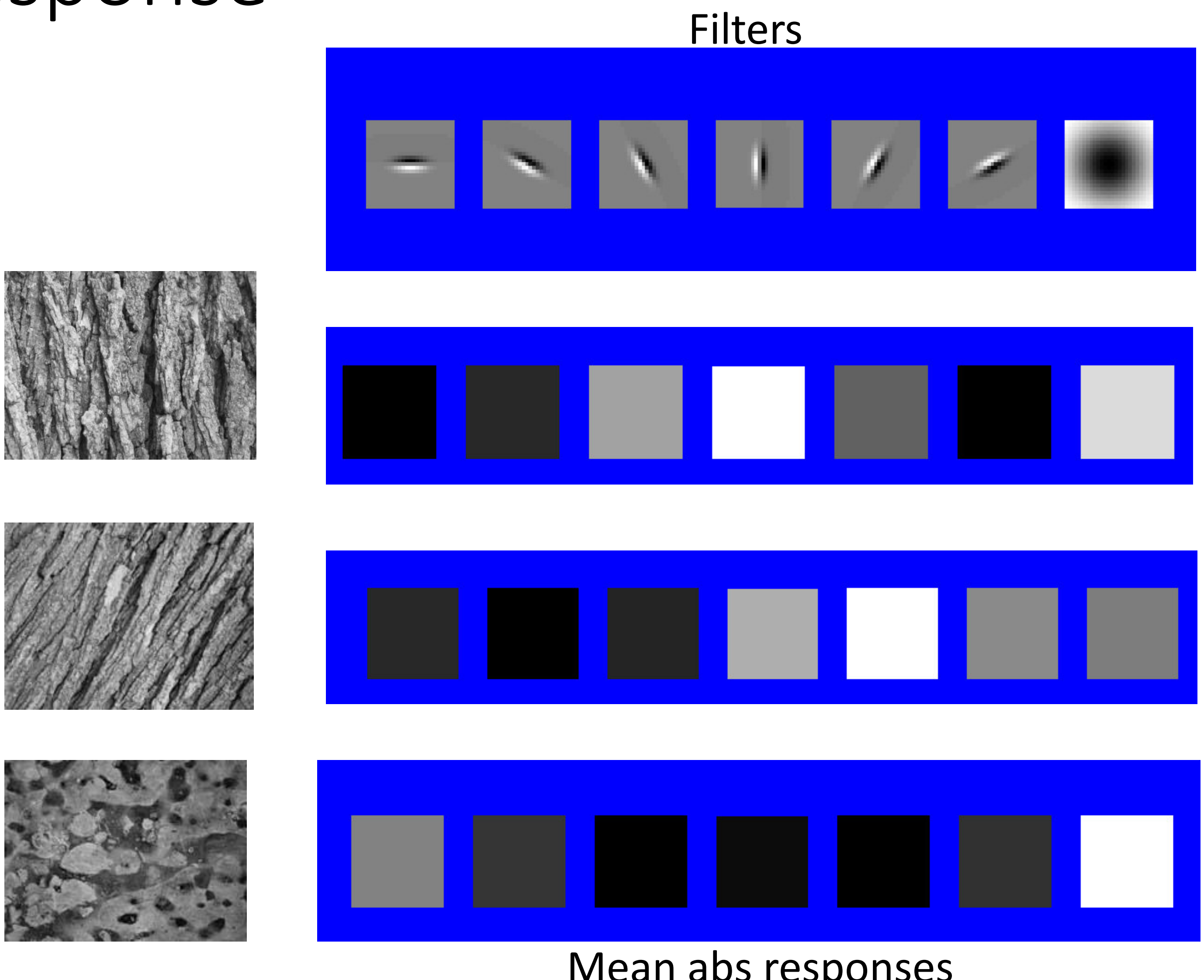


B

C

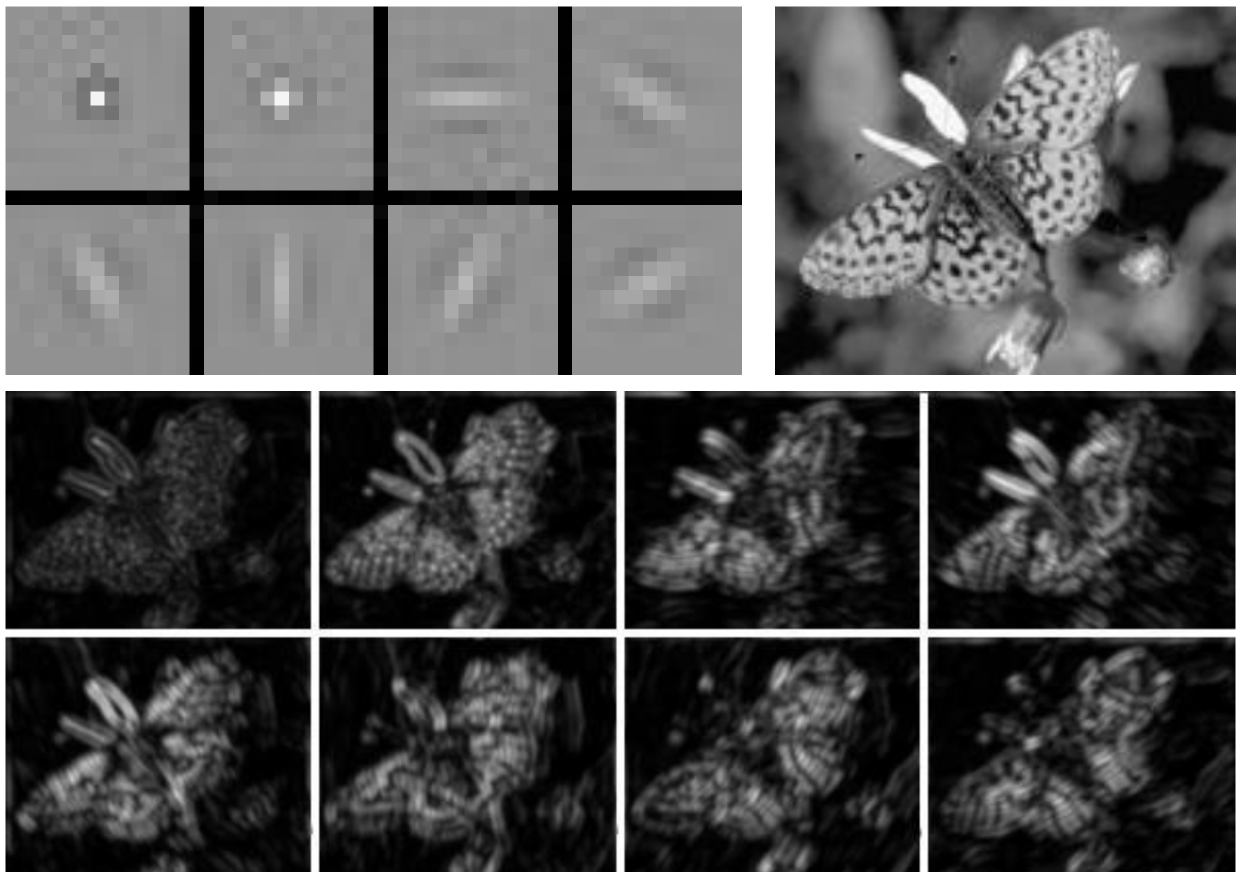


Representing texture by mean abs response

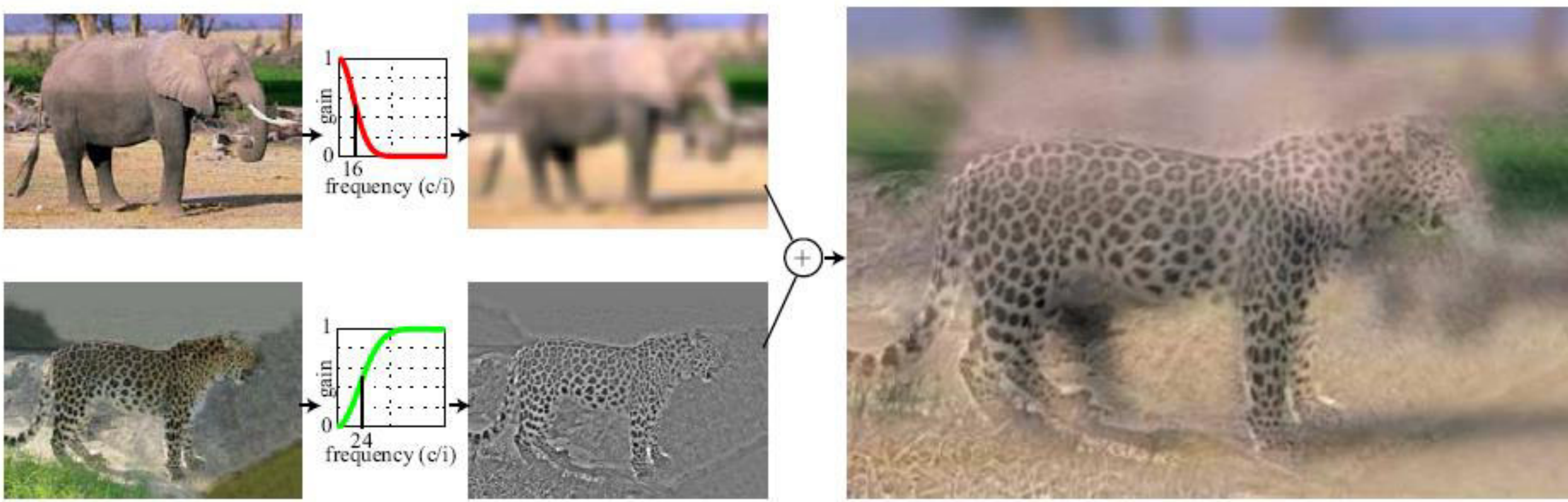


Representing texture

- Idea 2: take vectors of filter responses at each pixel and cluster them, then take histograms (more on this in coming weeks)



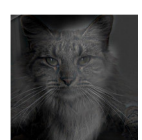
Hybrid Images



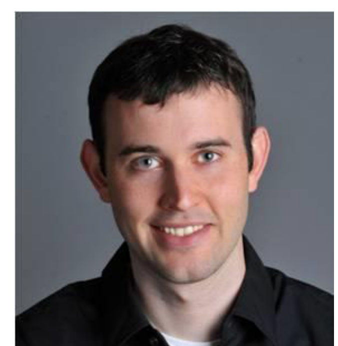
- A. Oliva, A. Torralba, P.G. Schyns, [“Hybrid Images,”]() SIGGRAPH 2006

Slide credit: Derek Hoiem

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



?



Slide credit: Derek Hoiem

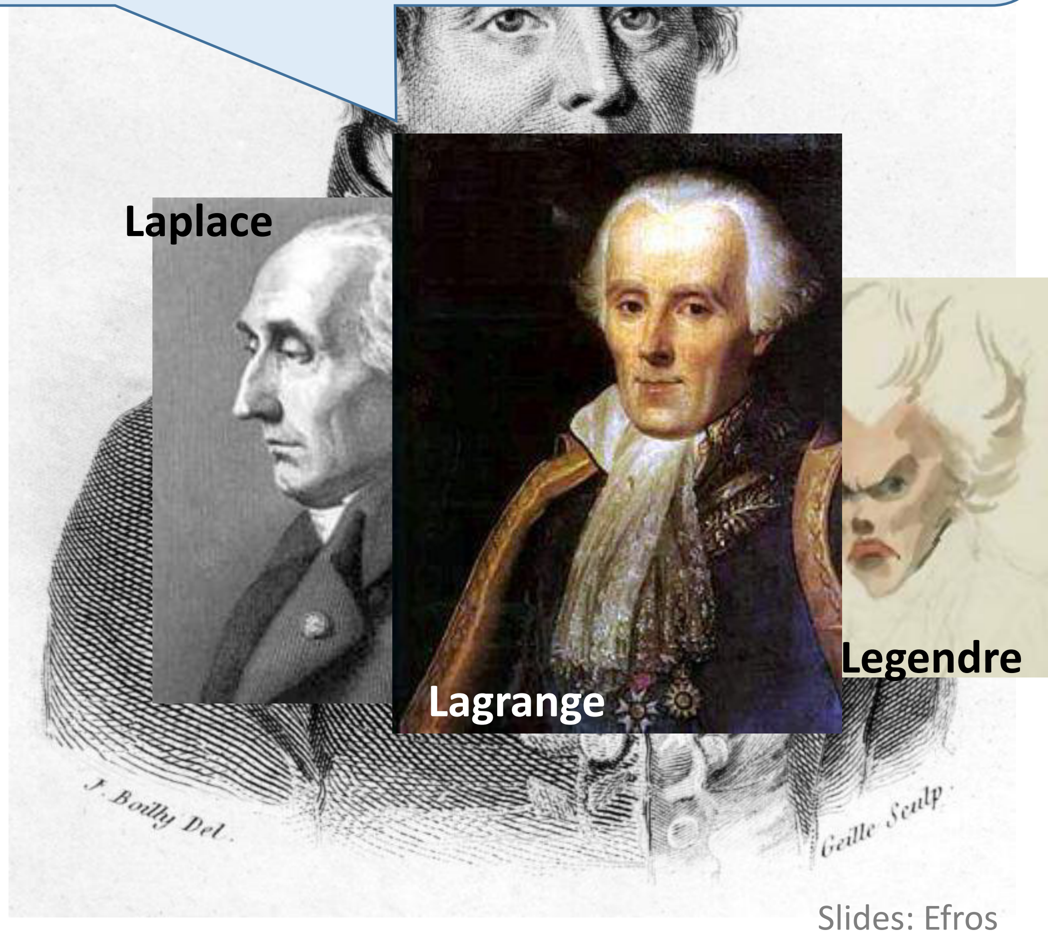
Jean Baptiste Joseph Fourier (1768-1830)

...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.

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



Slides: Efros

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?

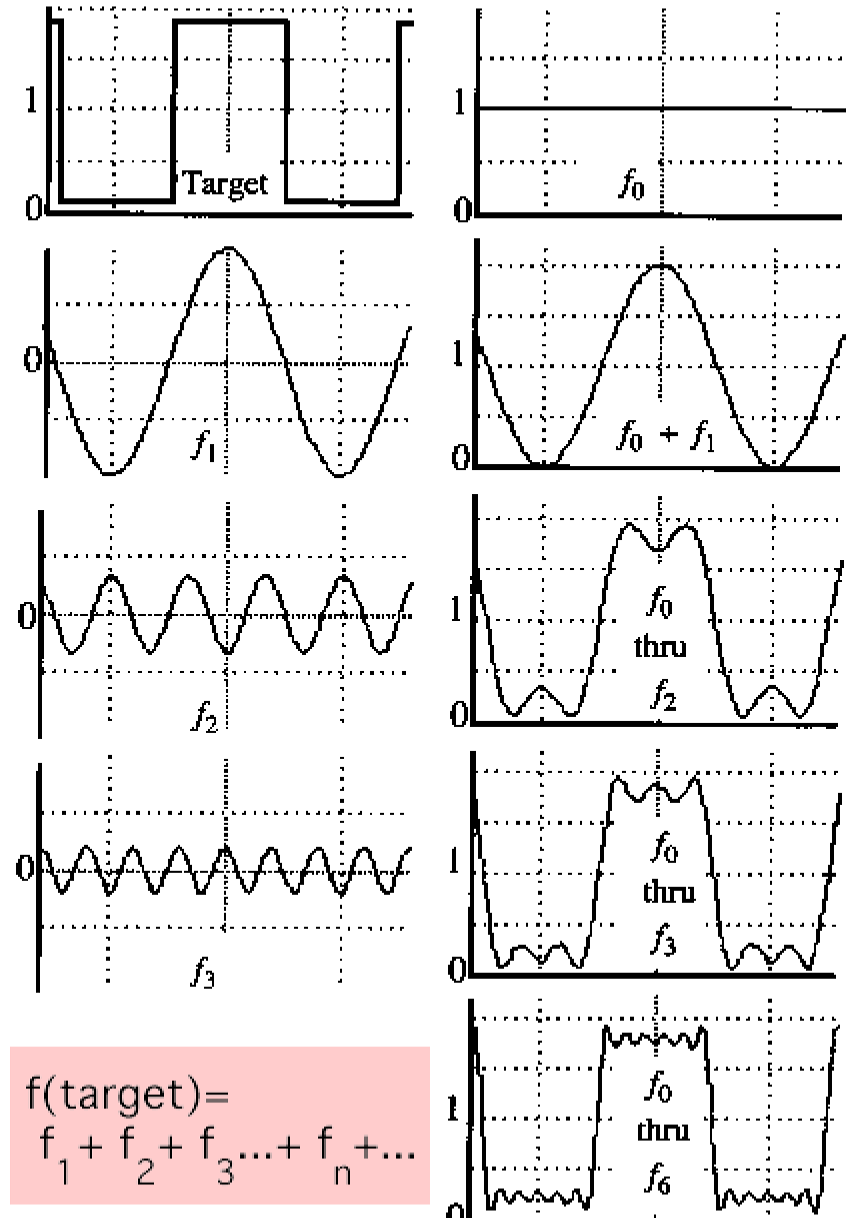
Slide credit: James Hays

A sum of sines

Our building block:

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

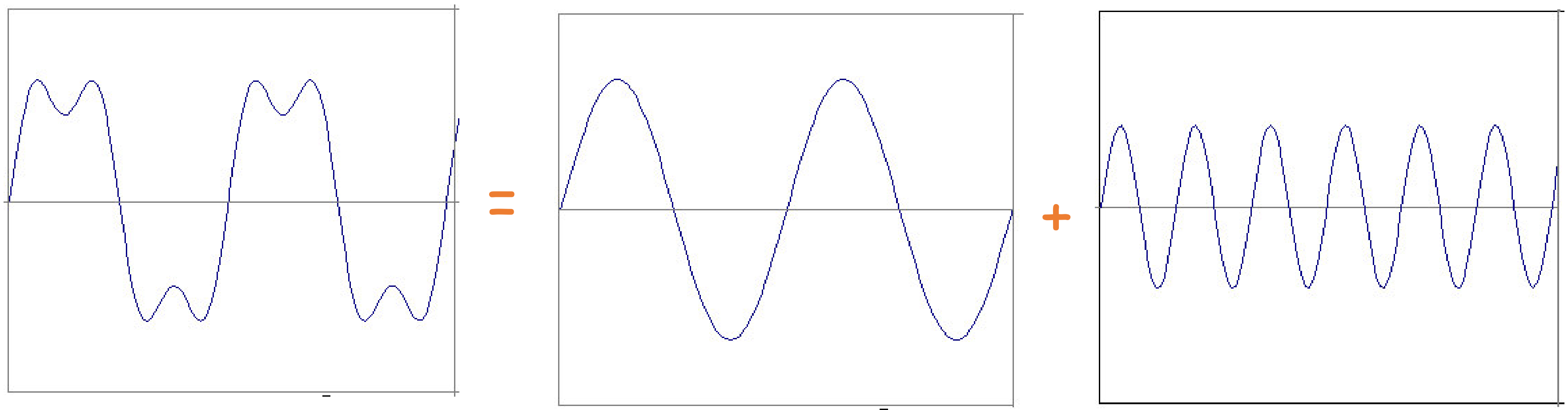
Add enough of them to get any signal *f(x)* you want!

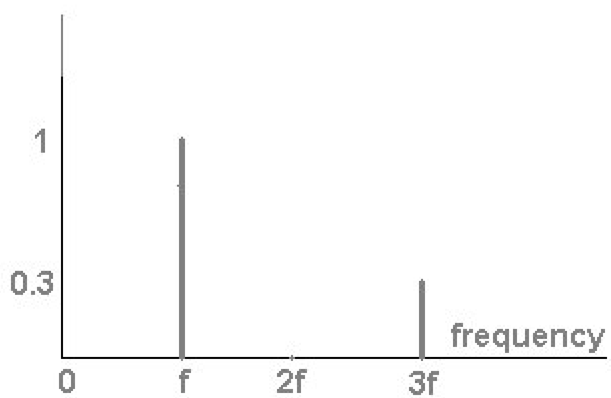


f(target)=
$f_1 + f_2 + f_3 \ldots + f_n + \ldots$

Frequency Spectra

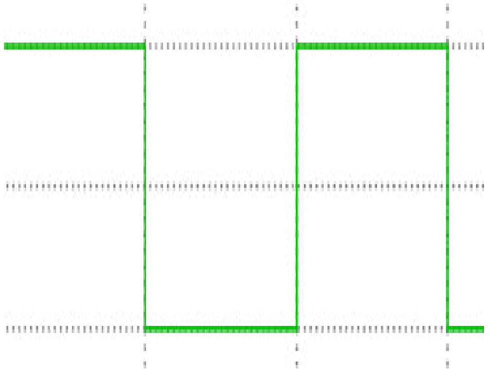
- example : $g(t) = \sin(2\pi f\, t) + (1/3)\sin(2\pi(3f)\, t)$



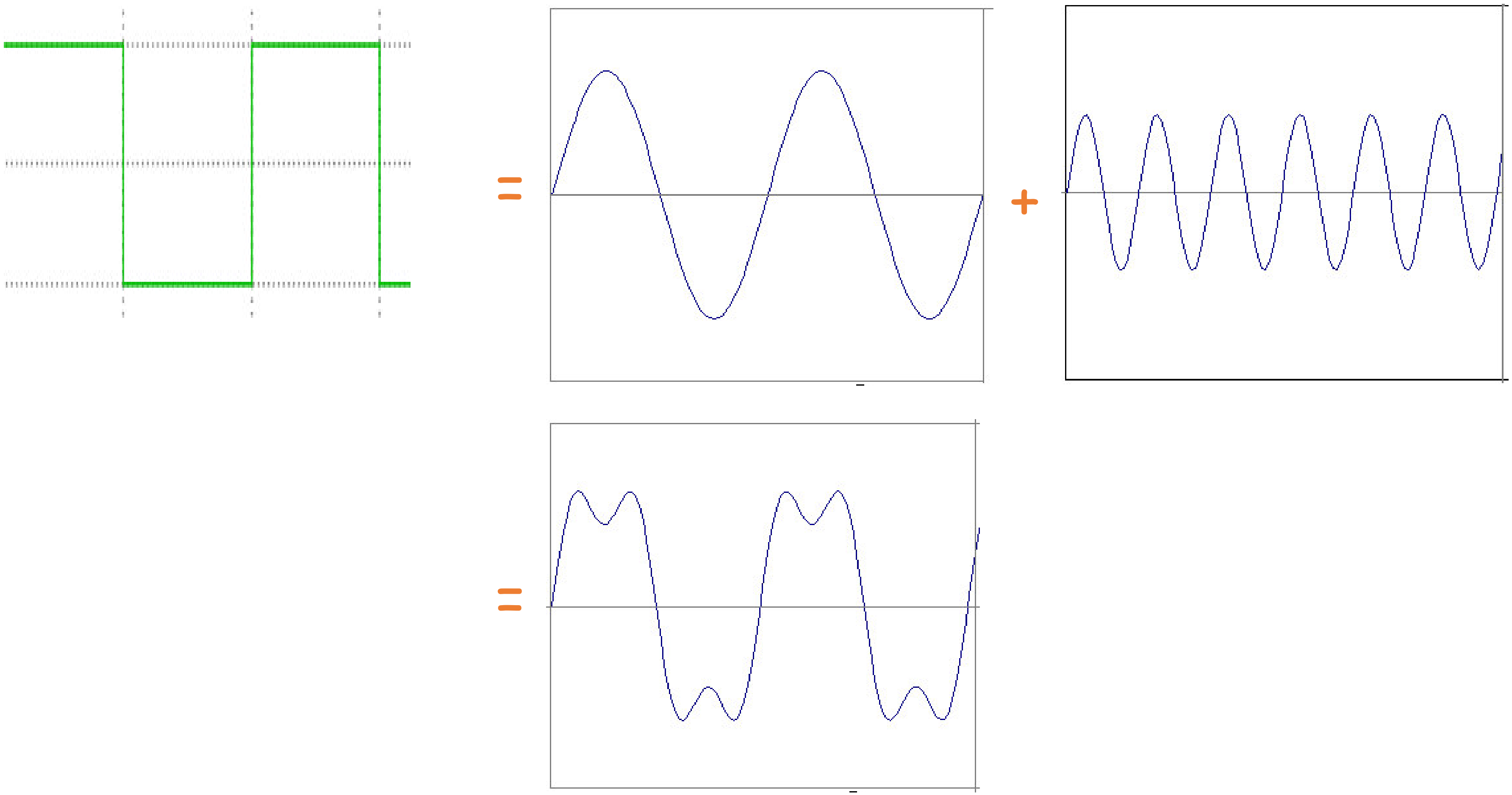


Slides: Efros

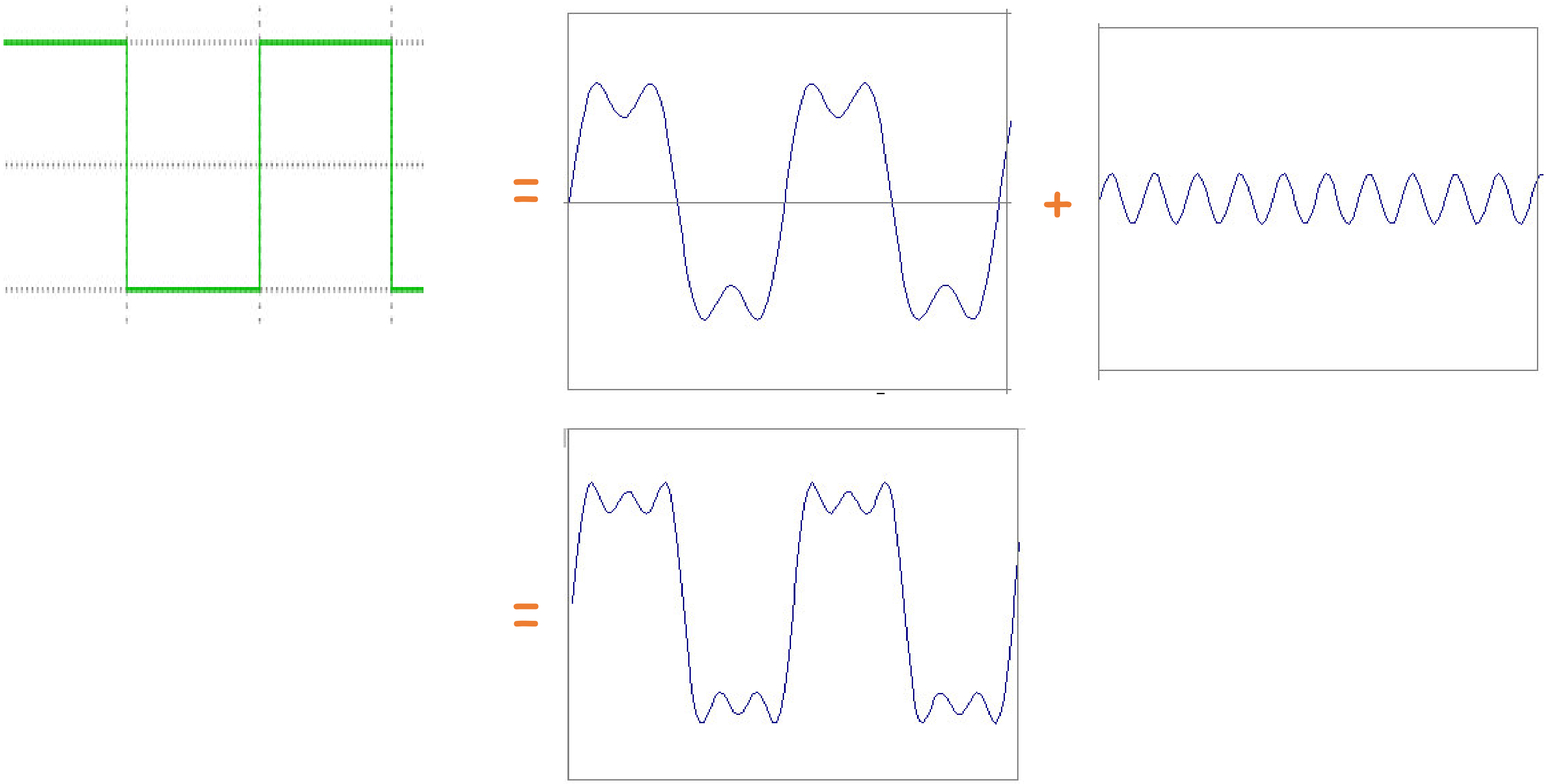
Frequency Spectra



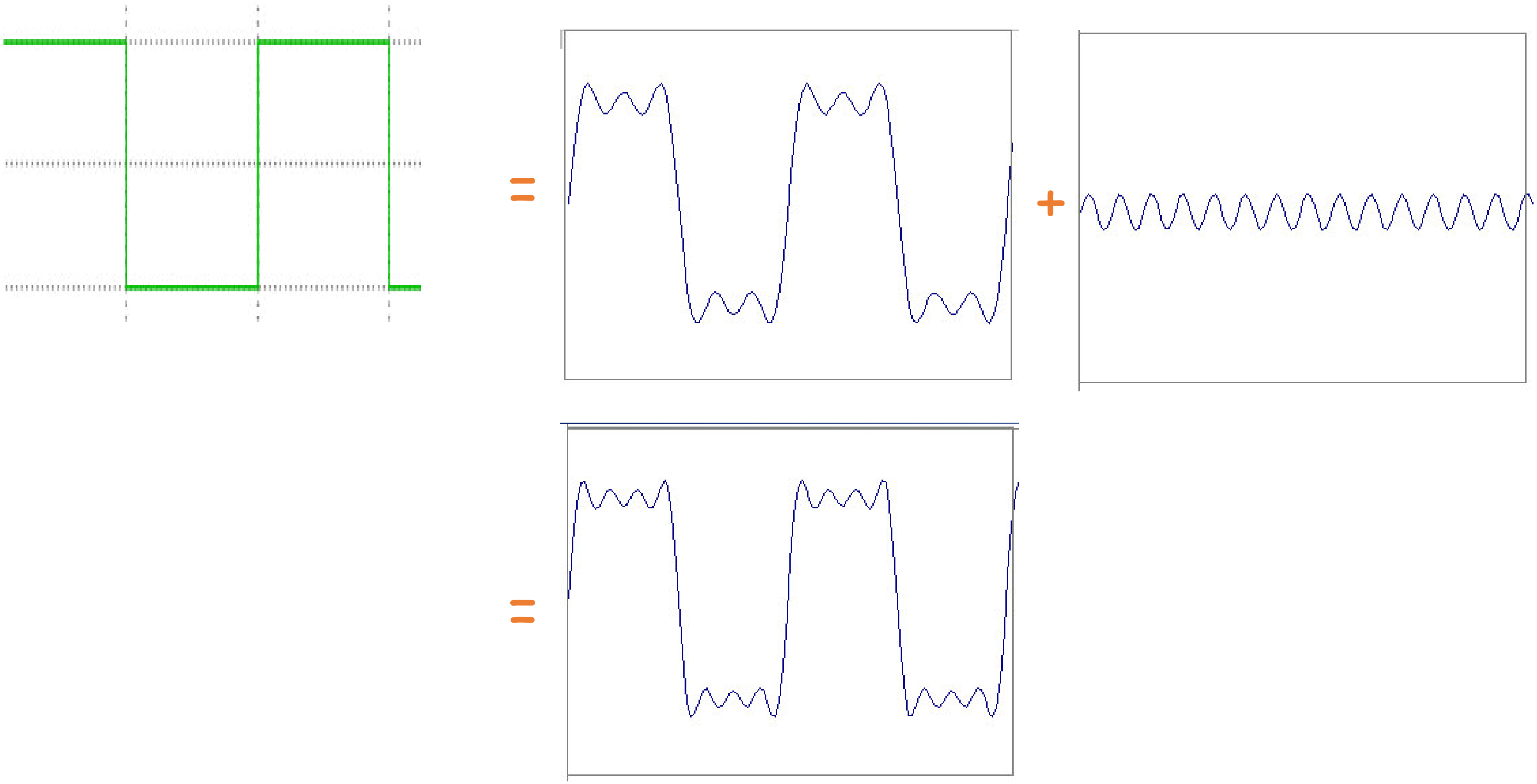
Frequency Spectra



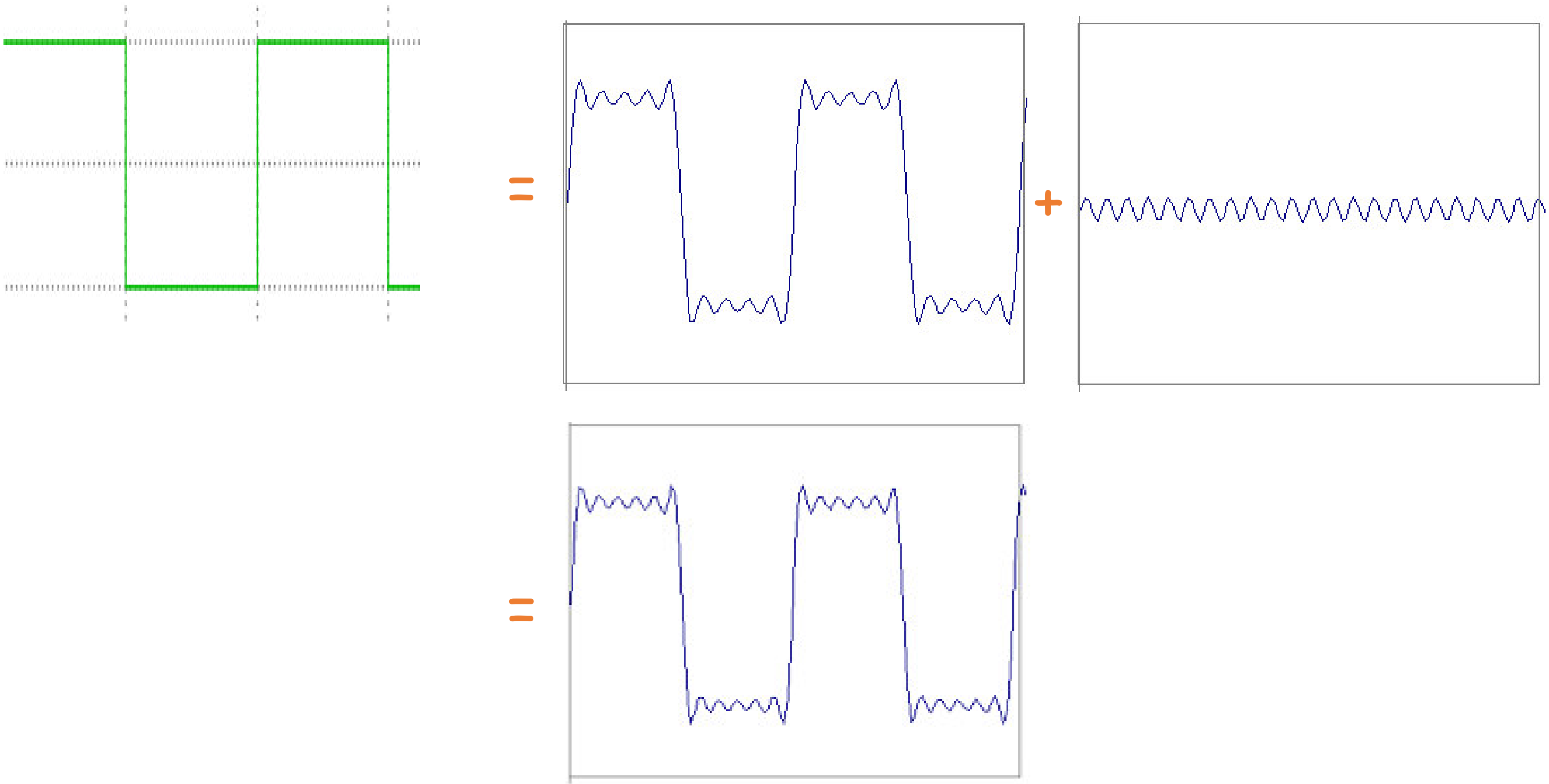
Frequency Spectra



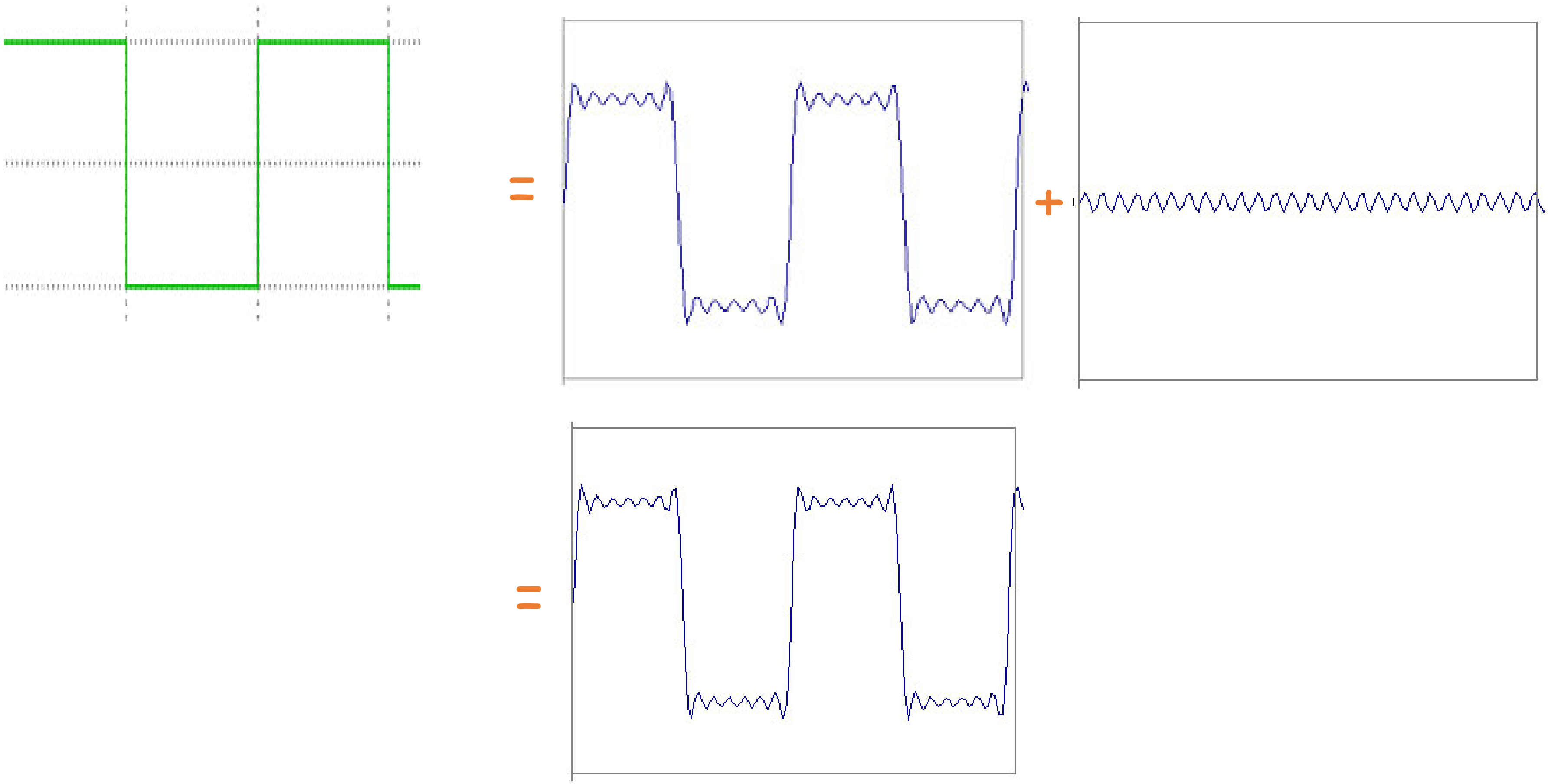
Frequency Spectra



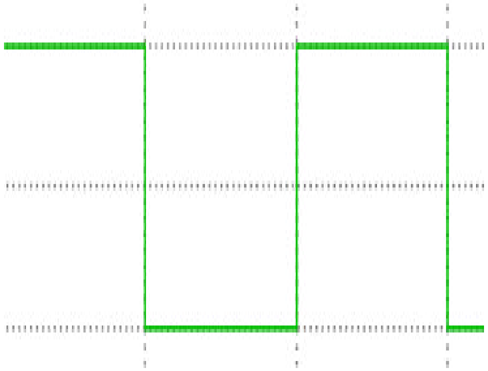
Frequency Spectra



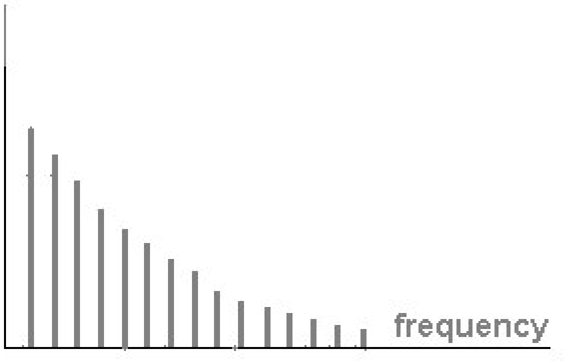
Frequency Spectra



Frequency Spectra

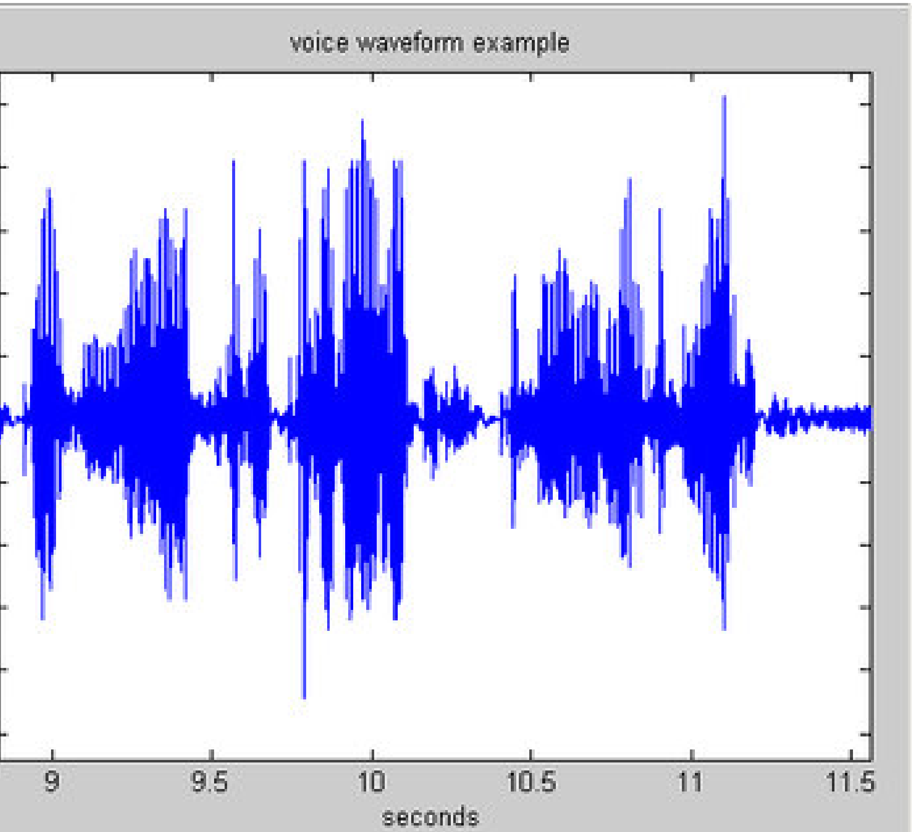


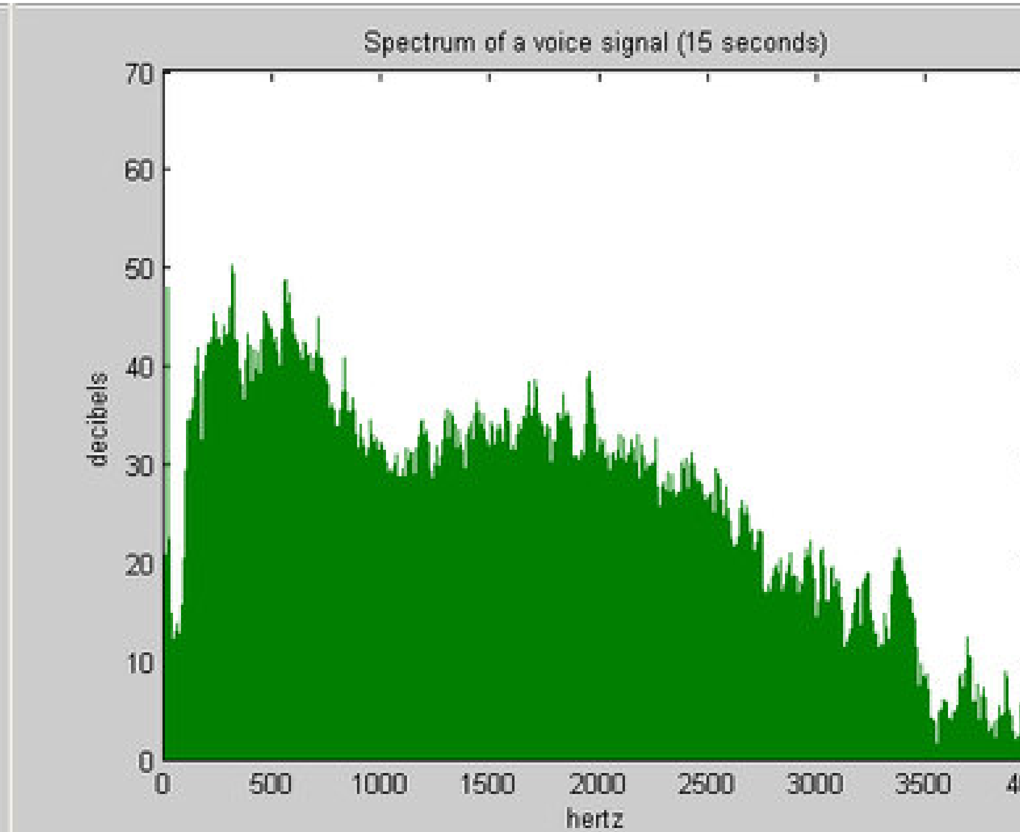
$$= \boxed{A\sum_{k=1}^{\infty}\frac{1}{k}\sin(2\pi kt)}$$



Example: Music

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

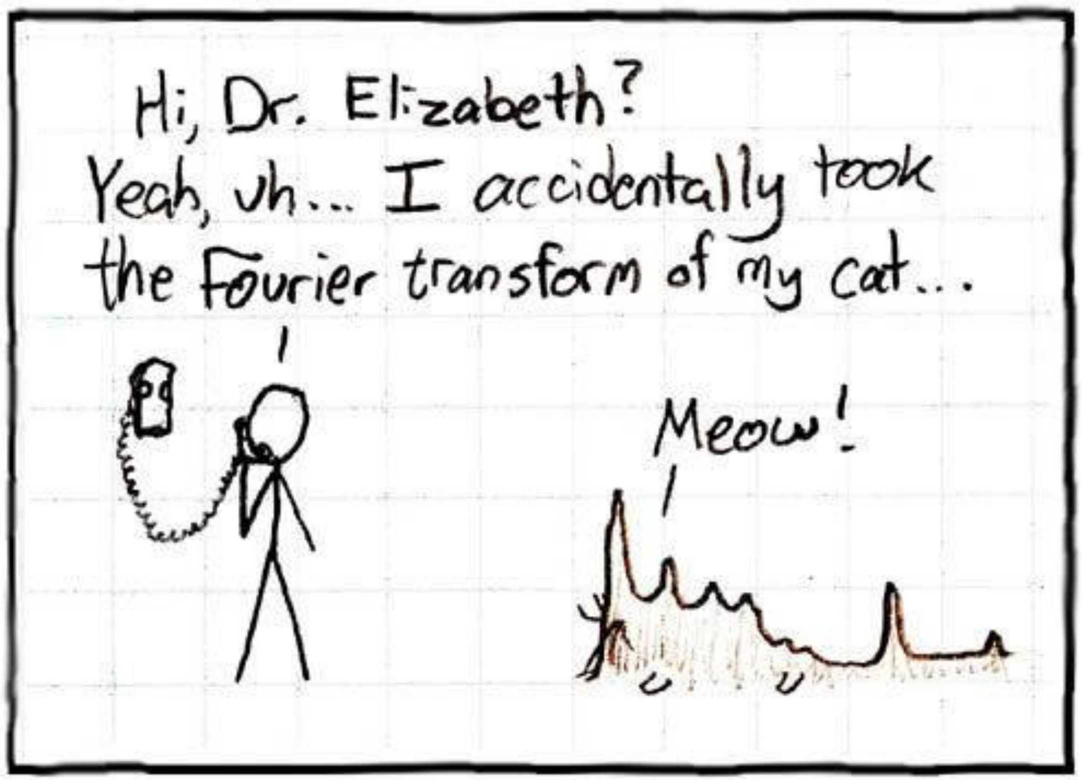




Other signals

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

Cats(?)

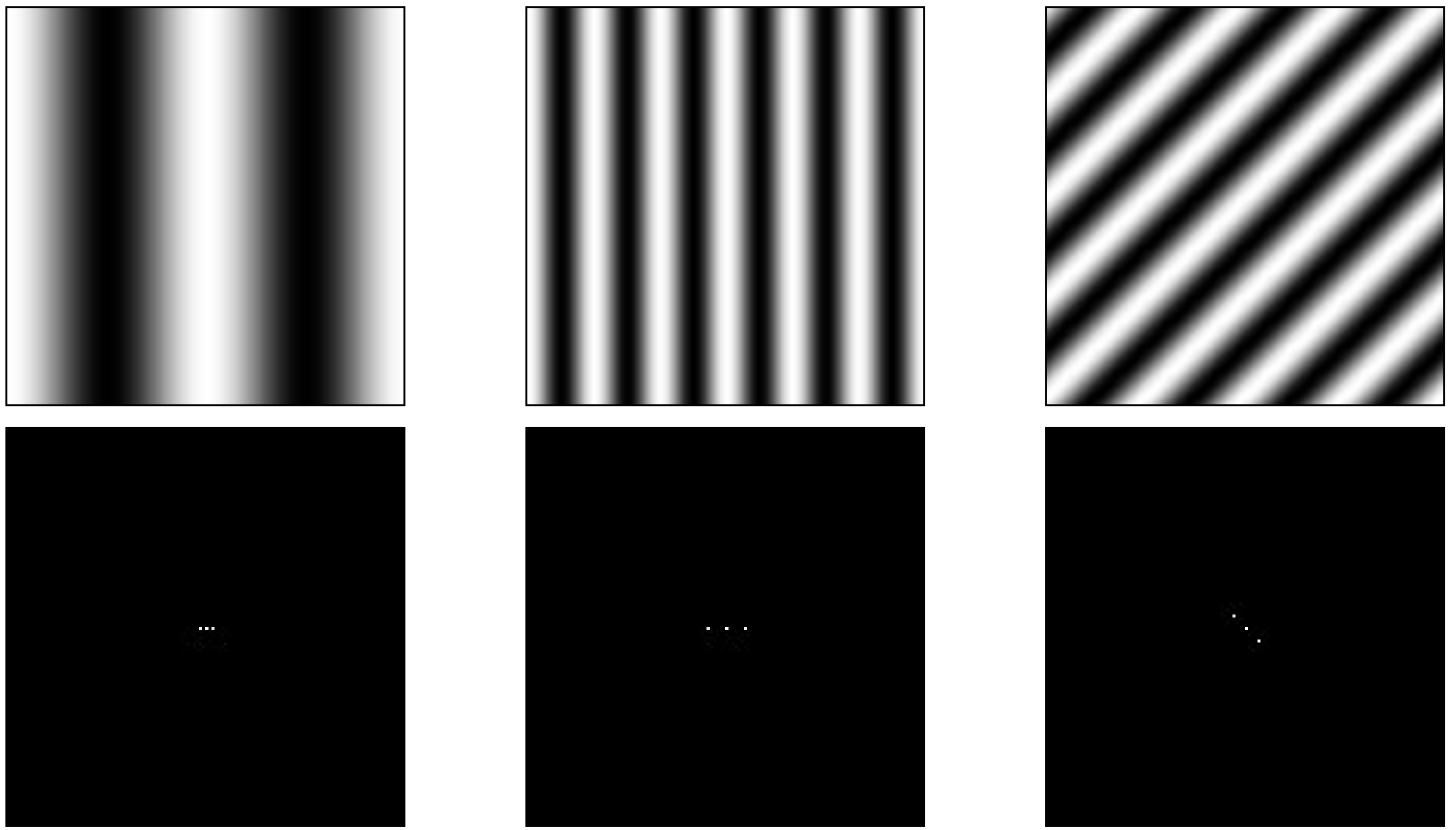


xkcd.com

Fourier analysis in images

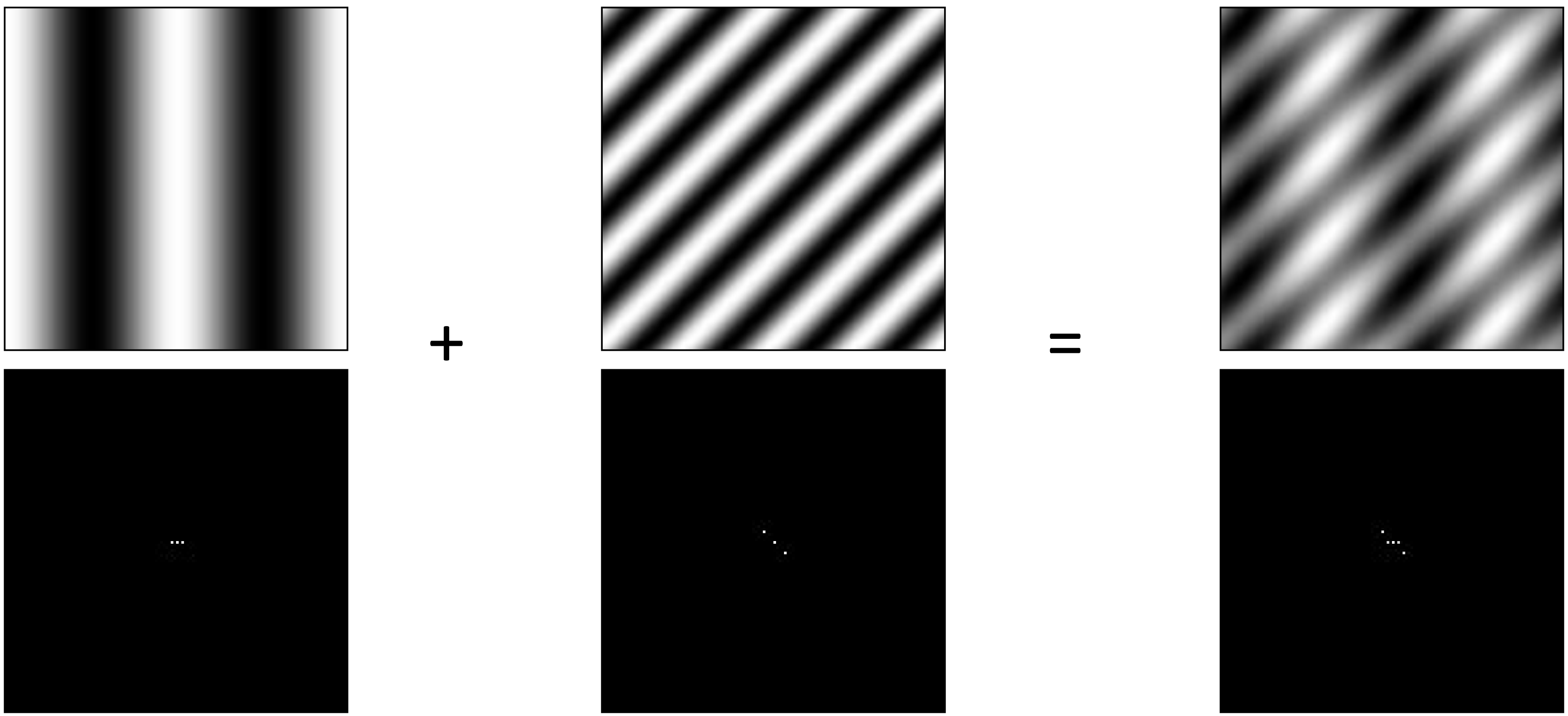
Intensity Image

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

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

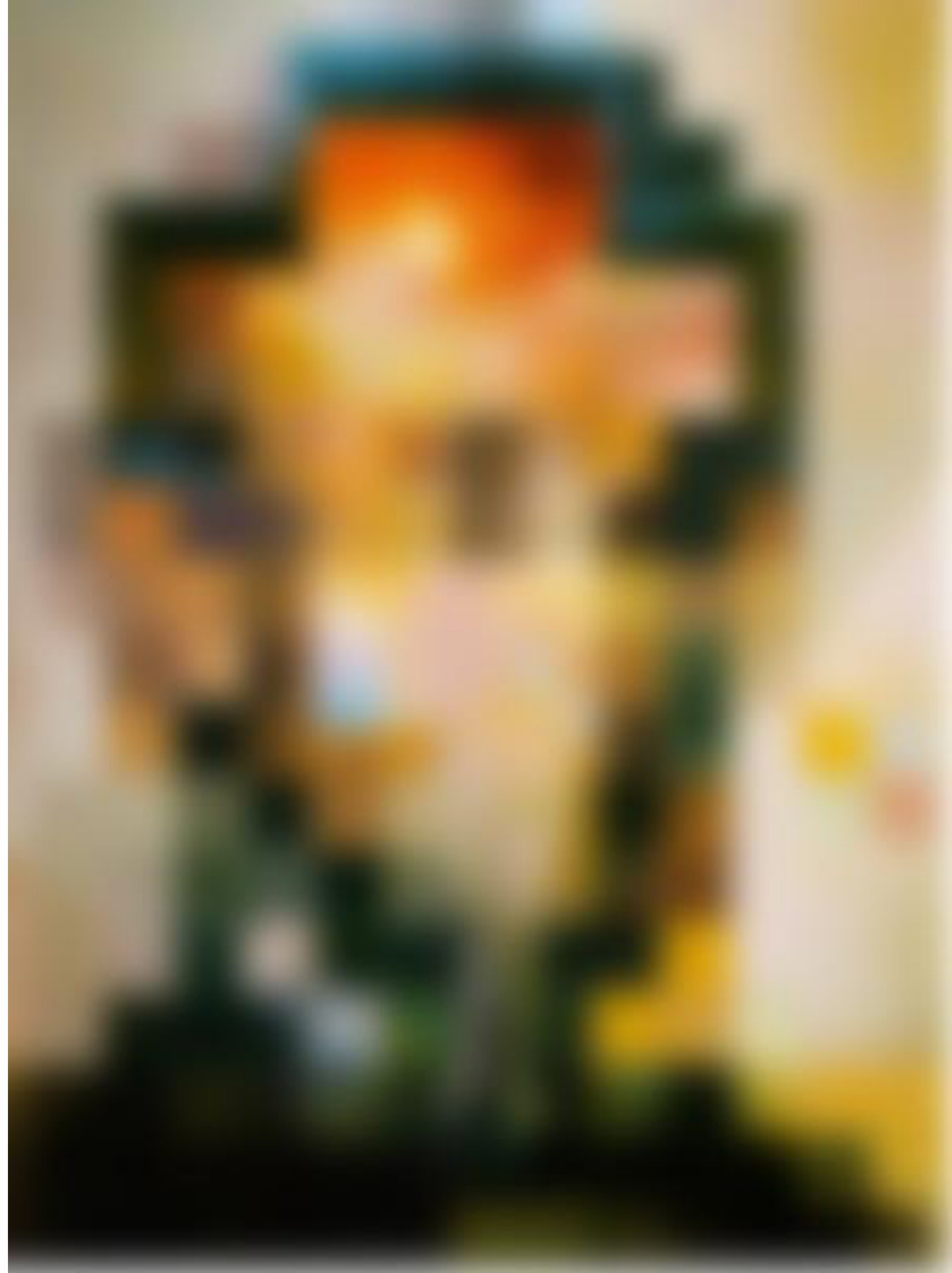
Amplitude: $A = \pm\sqrt{R(\omega)^2 + I(\omega)^2}$

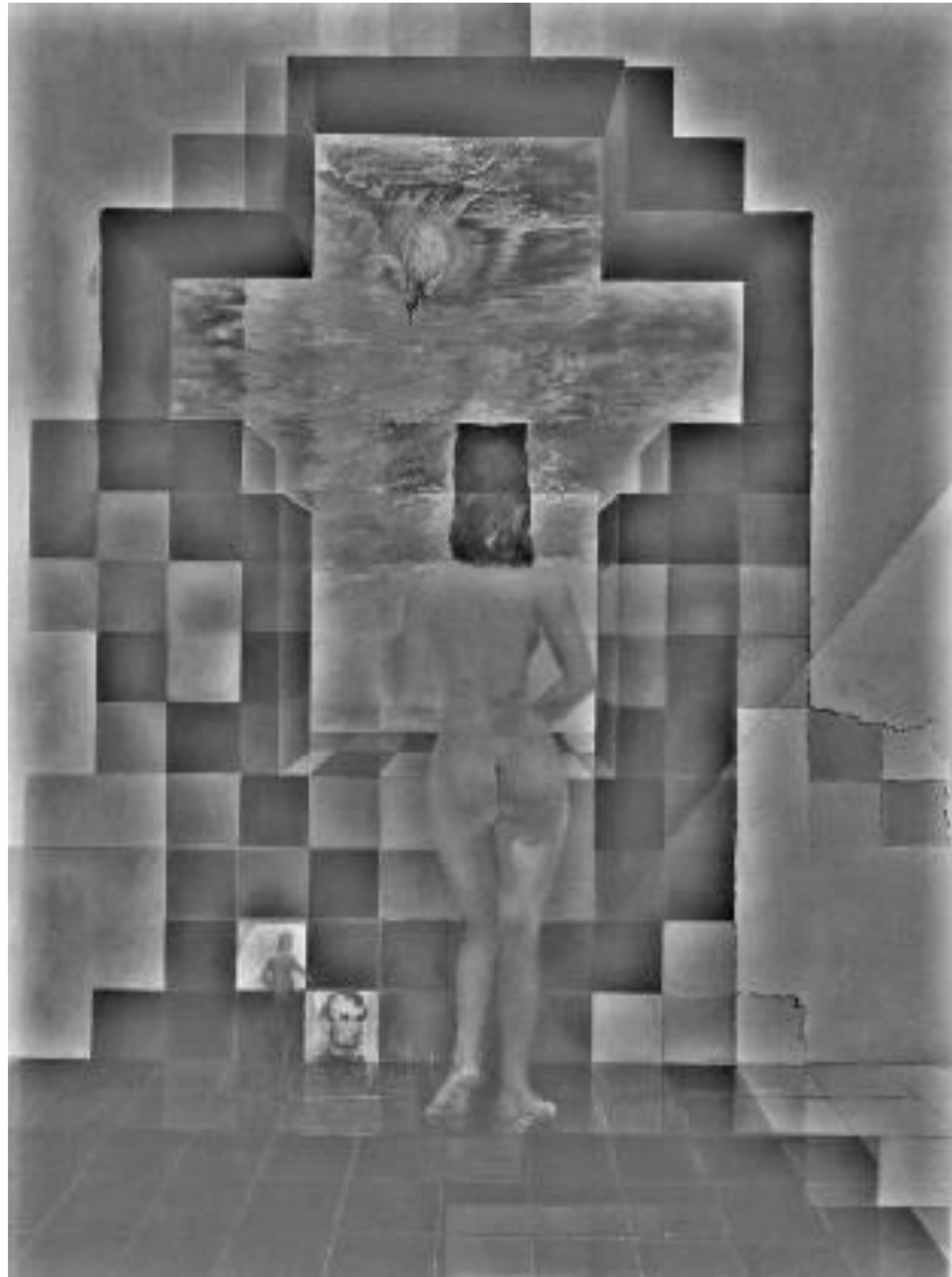
Phase: $\phi = \tan^{-1} \dfrac{I(\omega)}{R(\omega)}$

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

Salvador Dali invented Hybrid Images?

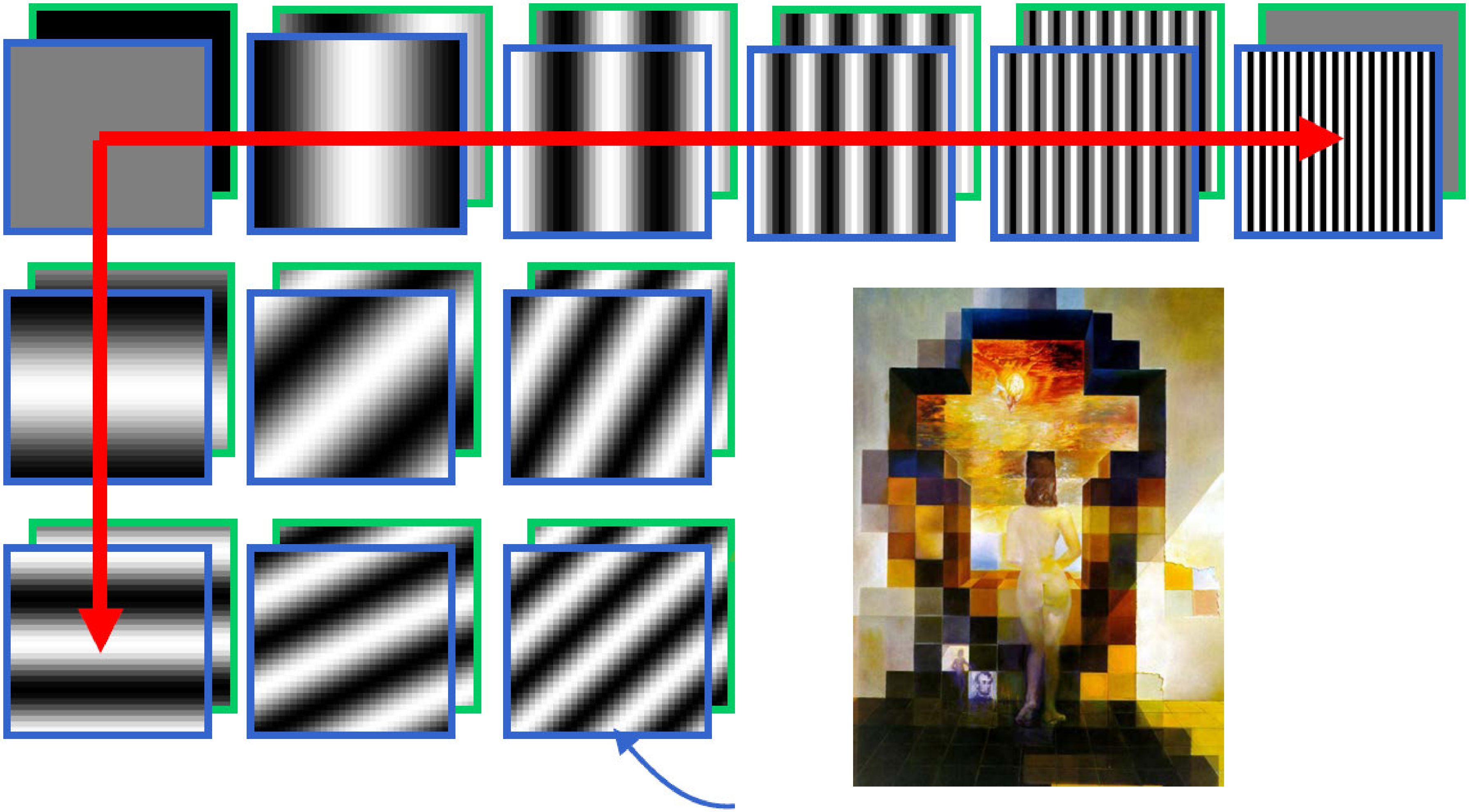
Salvador Dali
“Gala Contemplating the Mediterranean Sea, which at 30 meters becomes the portrait of Abraham Lincoln”, 1976





Fourier: A nice set of basis

Teases away fast vs. slow changes in the image.



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)***

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

For every ω from 0 to inf, (actually –inf to inf), ***F(ω)*** holds the amplitude *A* and phase ϕ of the corresponding sine

- How can *F* hold both? Complex number trick!

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

$$F(\omega) = \underbrace{R(\omega)}_{\textit{Even}} + i\underbrace{I(\omega)}_{\textit{Odd}}$$

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

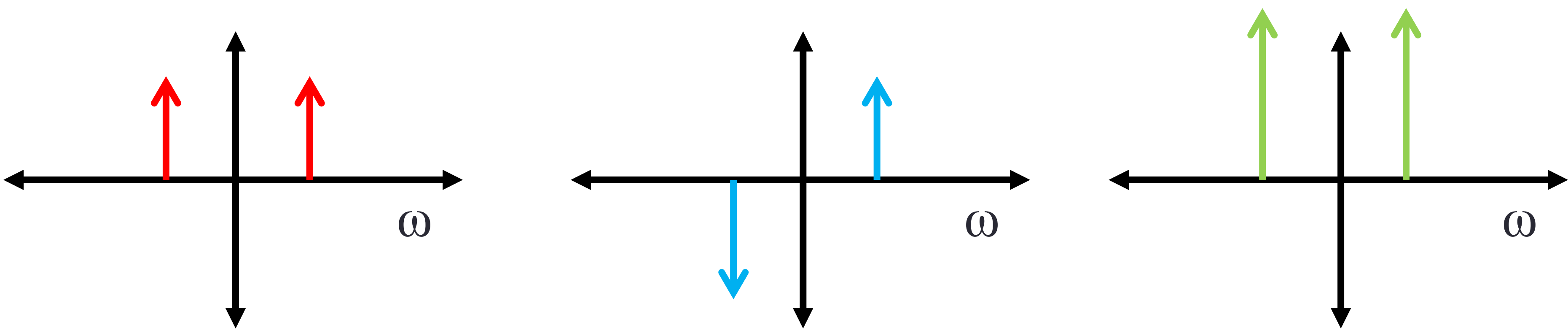
And we can go back:

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

Frequency Spectra – Even/Odd

Frequency actually goes from –inf to inf.
Sinusoid example:

Even (cos) | *Odd* (sin) | *Magnitude*



Real | ***Imaginary*** | ***Power***

2D Fourier Transforms

- The two dimensional version: .

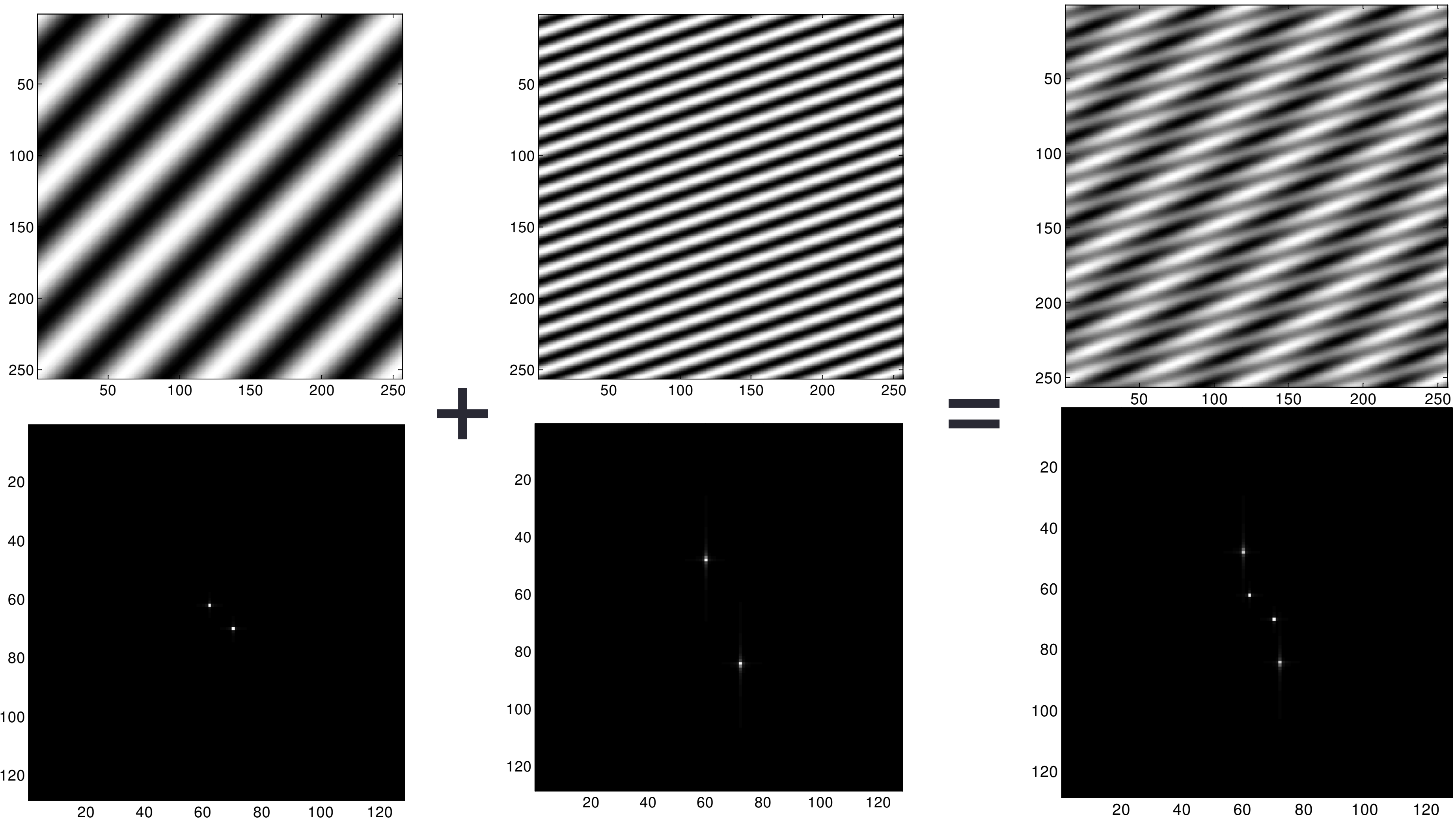
$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\, e^{-i2\pi(ux+vy)}\, dx\, dy$$

- And the 2D ***Discrete FT***:

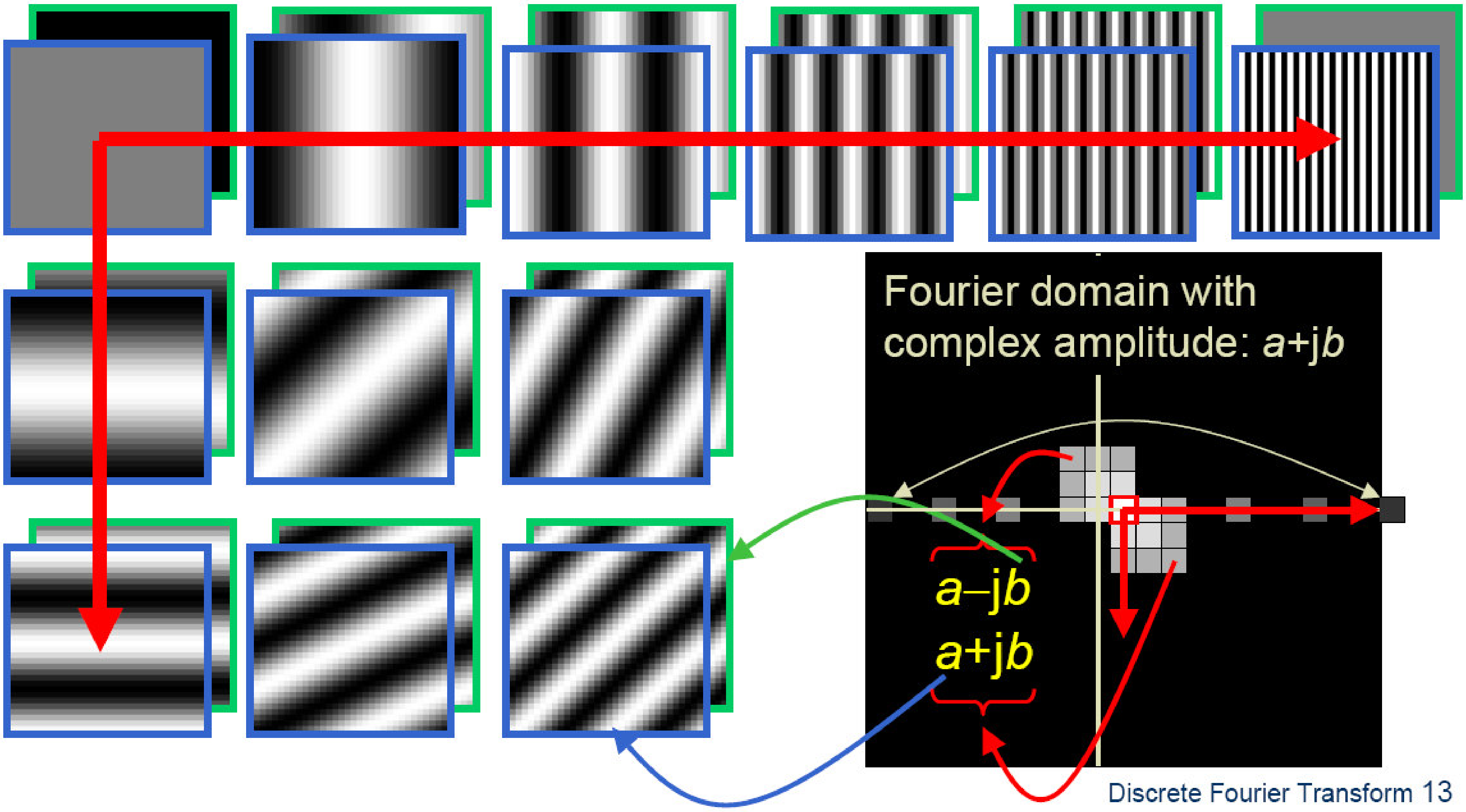
$$F(k_x,k_y) = \frac{1}{N}\sum_{x=0}^{x=N-1}\sum_{y=0}^{y=N-1} f(x,y)\, e^{-i\frac{2\pi(k_x x+k_y y)}{N}}$$

- Works best when you put the origin of *k* in the middle….

Linearity of Sum

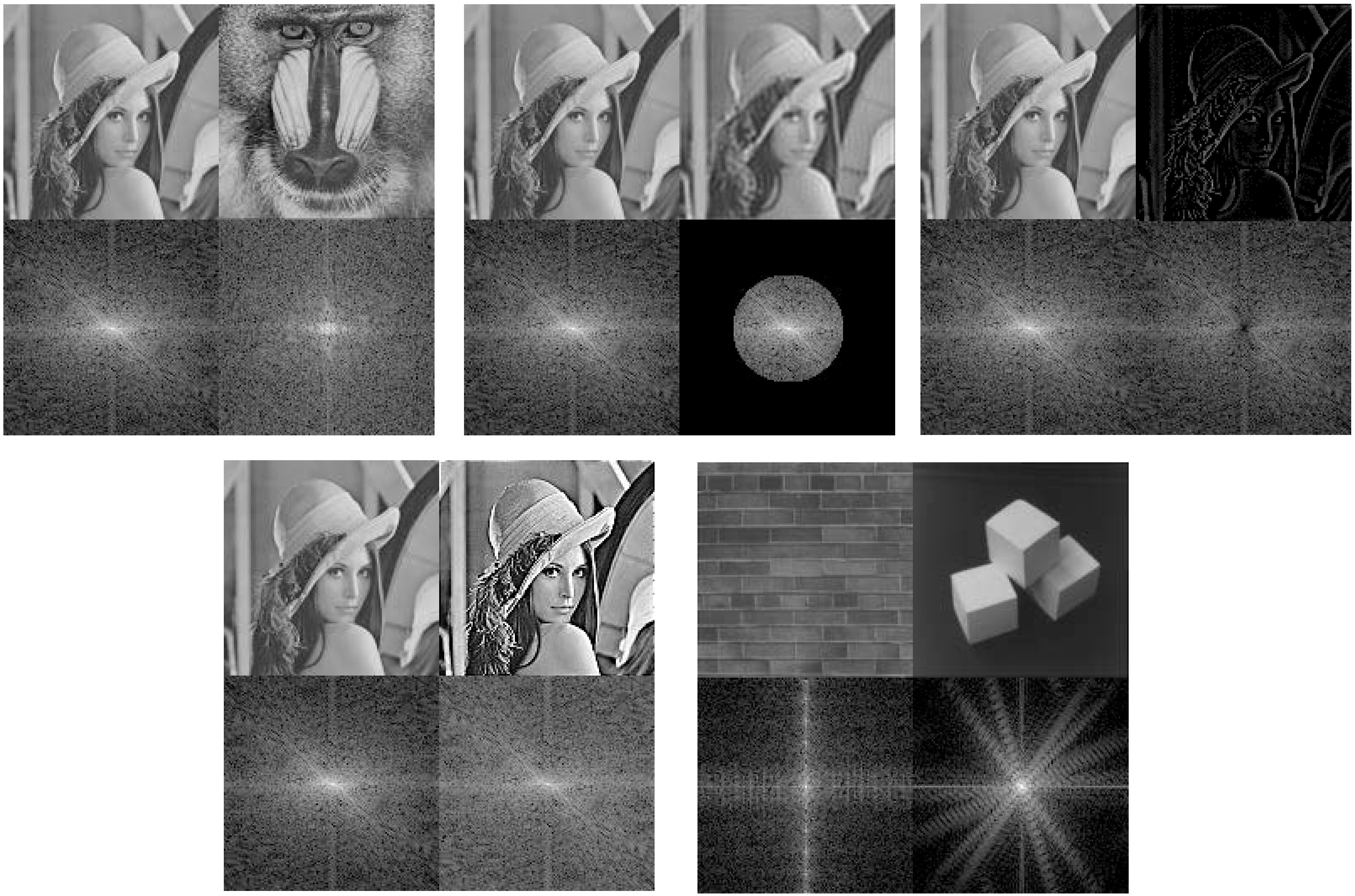


Extension to 2D – Complex plane



Both a Real and Im version

Examples



B.K. Gunturk

Fourier Transform and Convolution

Let $g = f * h$

Then $G(u) = \int_{-\infty}^{\infty} g(x) e^{-i2\pi ux} dx$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\tau) h(x-\tau) e^{-i2\pi ux} d\tau dx$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left[f(\tau) e^{-i2\pi u\tau} d\tau\right]\left[h(x-\tau) e^{-i2\pi u(x-\tau)} dx\right]$$

$$= \int_{-\infty}^{\infty} \left[f(\tau) e^{-i2\pi u\tau} d\tau\right]\int_{-\infty}^{\infty} \left[h(x') e^{-i2\pi ux'} dx'\right]$$

$$= F(u) H(u)$$

Convolution in spatial domain

$\Leftrightarrow$ *Multiplication in frequency domain*

Fourier Transform and Convolution

Spatial Domain *(x)*		Frequency Domain *(u)*
$g = f * h$	$\longleftrightarrow$	$G = FH$
$g = fh$	$\longleftrightarrow$	$G = F * H$

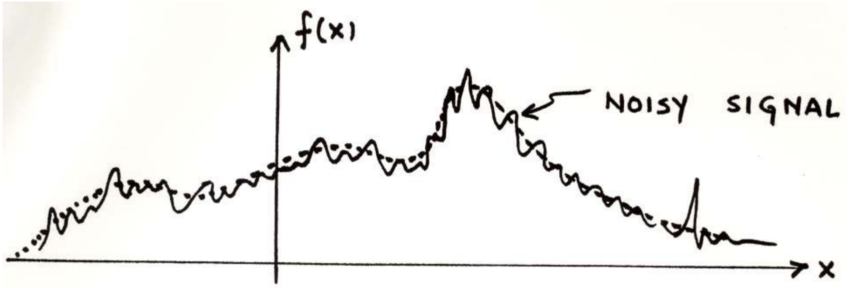
So, we can find *g(x)* by Fourier transform

$$\begin{tikzcd} g & = & f \arrow[d, "\boxed{\text{FT}}"] & * & h \arrow[d, "\boxed{\text{FT}}"] \\ G \arrow[u, "\boxed{\text{IFT}}"] & = & F & \times & H \end{tikzcd}$$

Example use: Smoothing/Blurring

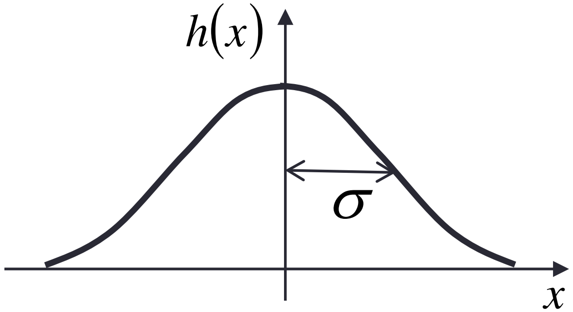
- We want a smoothed function of *f(x)*

$$g(x) = f(x) * h(x)$$



- Let us use a Gaussian kernel

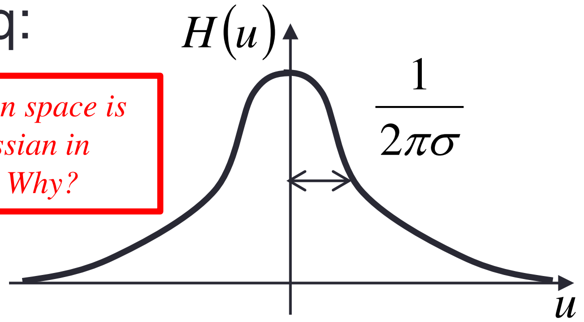
$$h(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2} \frac{x^2}{\sigma^2} \right]$$



- Convolution in space is multiplication in freq:

$$G(u) = F(u) H(u)$$

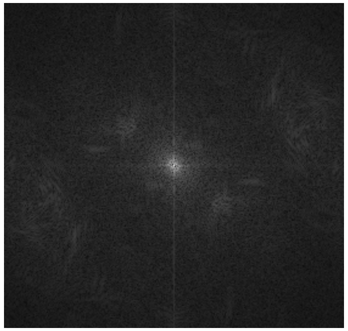
Fat Gaussian in space is skinny Gaussian in frequency. Why?



H(u) ***attenuates*** high frequencies in *F(u)* (Low-pass Filter)!

2D convolution theorem example

$f(x,y)$

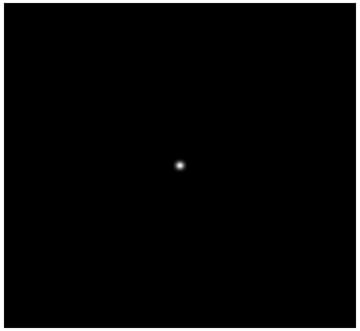


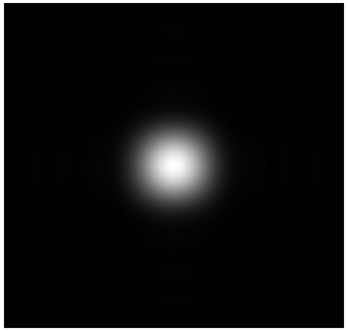
$|F(s_x,s_y)|$

$(\text{ or } |F(u,v)|\ )$

$*$ $\times$

$h(x,y)$

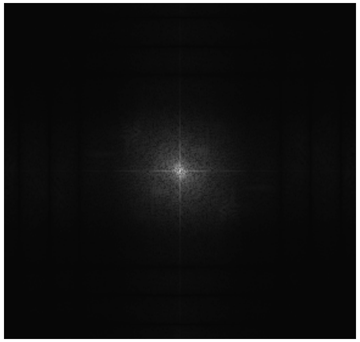




$|H(s_x,s_y)|$

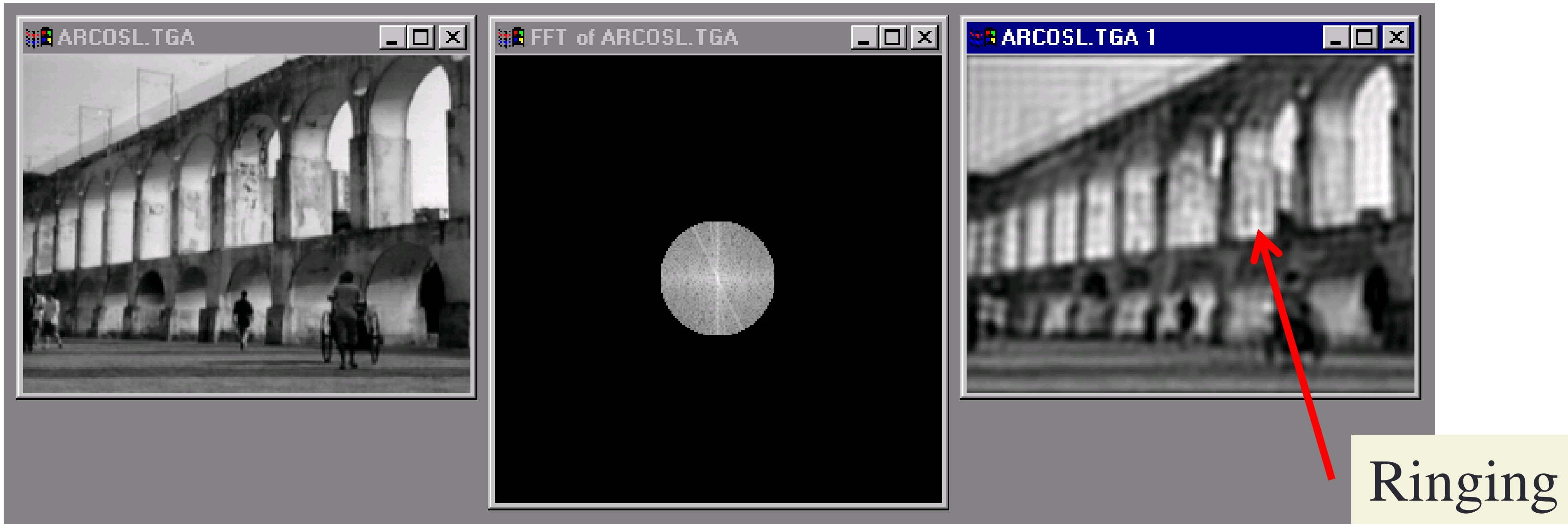
$\Downarrow$ $\Downarrow$

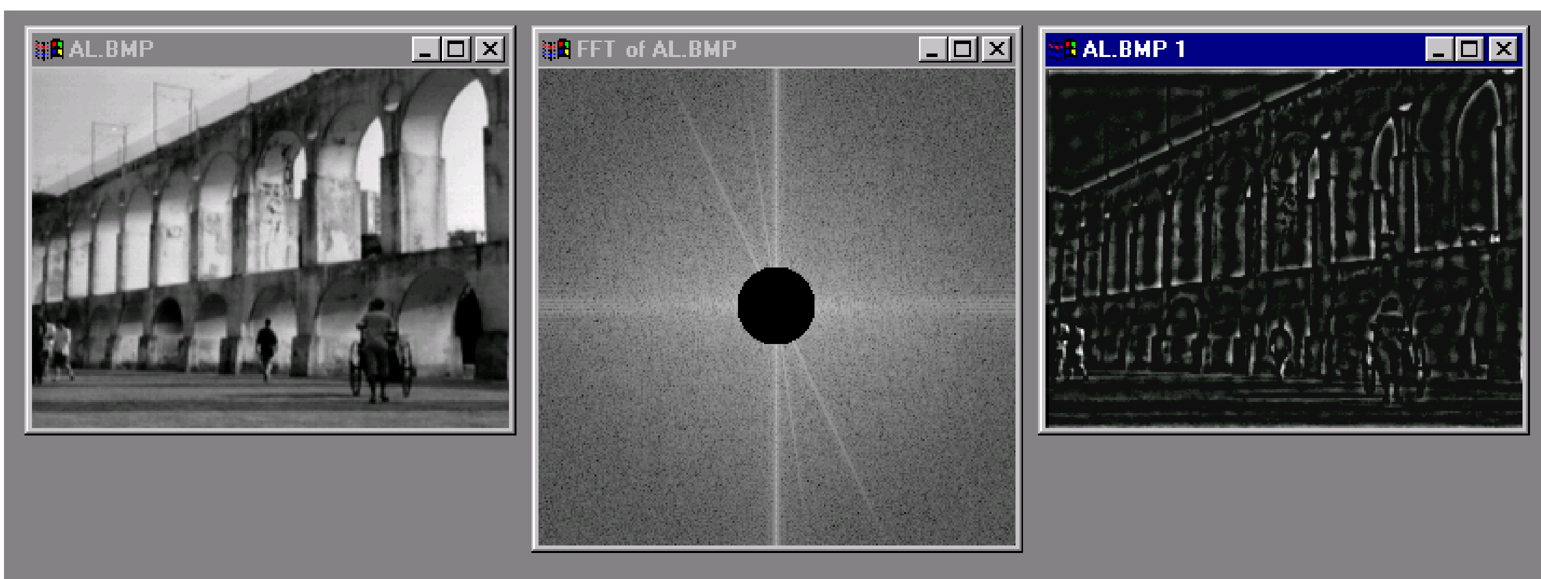
$g(x,y)$



$|G(s_x,s_y)|$

Low and High Pass filtering





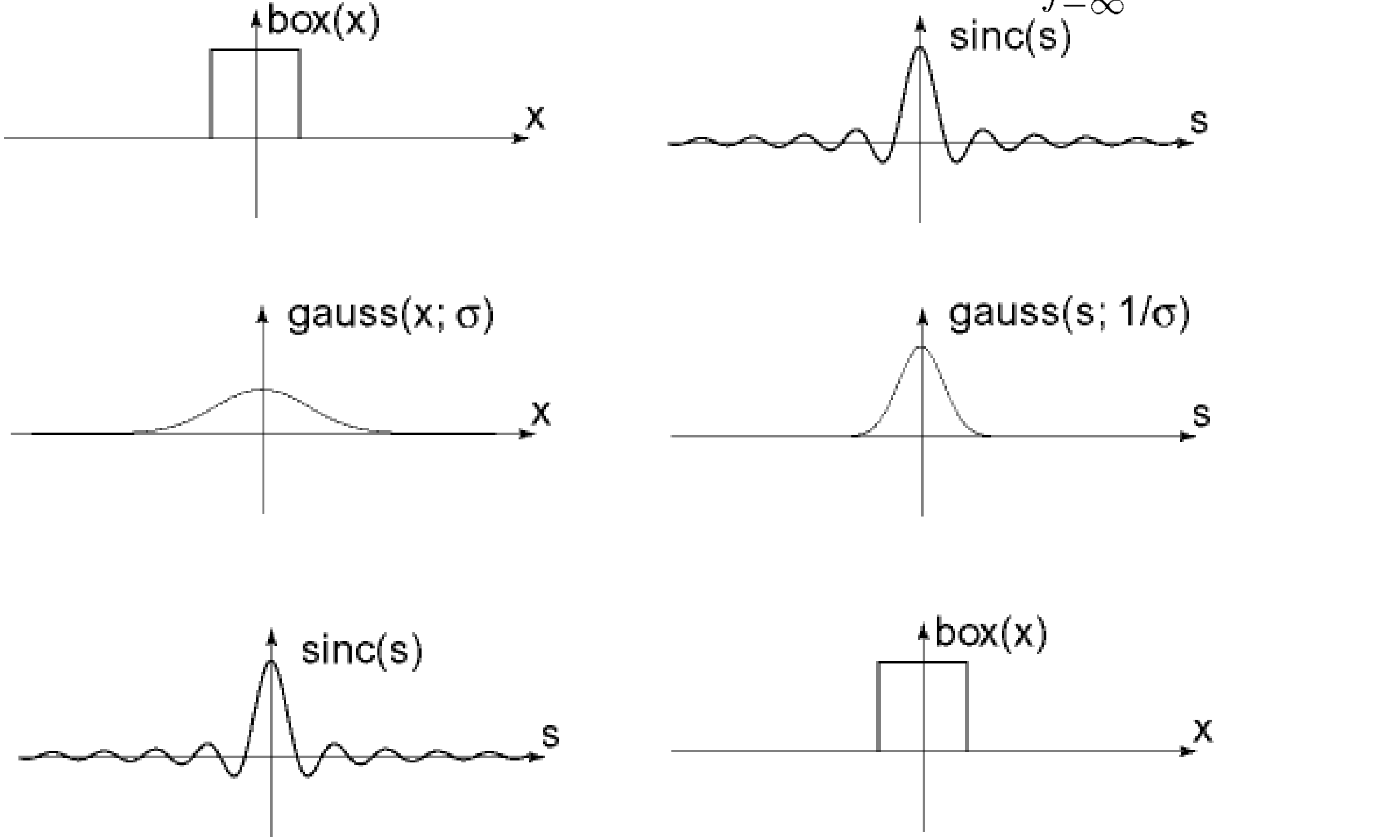
Fourier Transform smoothing pairs

Spatial domain

$f(x)$

Frequency domain

$$F(s) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi sx} dx$$



Properties of Fourier Transform

	Spatial Domain *(x)*	Frequency Domain *(u)*
Linearity	$c_1 f(x) + c_2 g(x)$	$c_1 F(u) + c_2 G(u)$
Scaling	$f(ax)$	$\frac{1}{\lvert a \rvert} F\left(\frac{u}{a}\right)$
Shifting	$f(x - x_0)$	$e^{-i2\pi u x_0} F(u)$
Symmetry	$F(x)$	$f(-u)$
Conjugation	$f^*(x)$	$F^*(-u)$
Convolution	$f(x) * g(x)$	$F(u)G(u)$
Differentiation	$\frac{d^n f(x)}{dx^n}$	$(i2\pi u)^n F(u)$

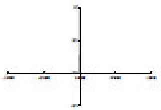
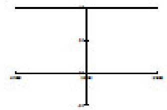
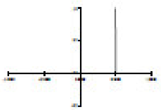
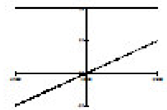
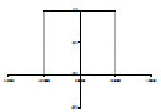
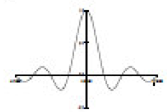
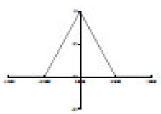
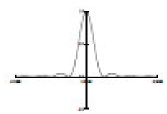
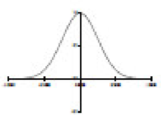
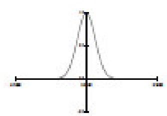
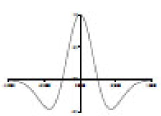
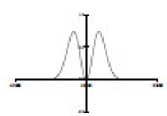
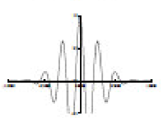
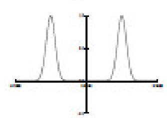
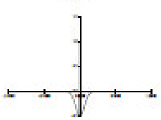
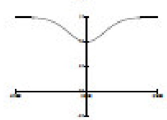
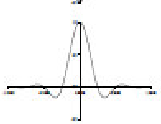
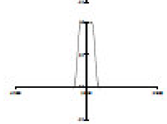
Shrink

Stretch

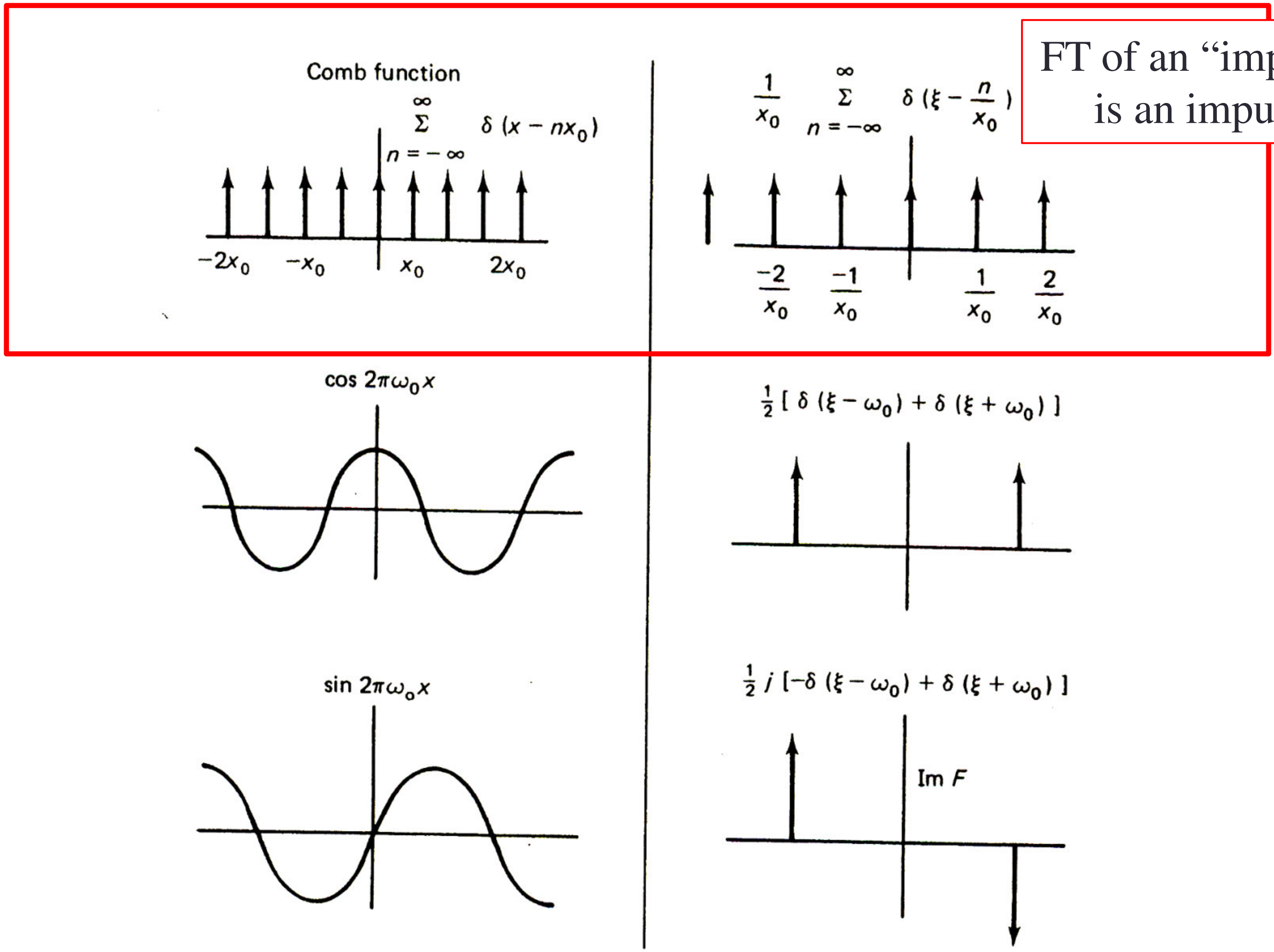
Differentiate

Multiply by u

Fourier Pairs (from Szeliski)

Name		Signal		Transform	
impulse		$\delta(x)$	$\Leftrightarrow$	1	
shifted impulse		$\delta(x-u)$	$\Leftrightarrow$	$e^{-j\omega u}$	
box filter		$\text{box}(x/a)$	$\Leftrightarrow$	$a\text{sinc}(a\omega)$	
tent		$\text{tent}(x/a)$	$\Leftrightarrow$	$a\text{sinc}^2(a\omega)$	
Gaussian		$G(x;\sigma)$	$\Leftrightarrow$	$\frac{\sqrt{2\pi}}{\sigma}G(\omega;\sigma^{-1})$	
Laplacian of Gaussian		$(\frac{x^2}{\sigma^4}-\frac{1}{\sigma^2})G(x;\sigma)$	$\Leftrightarrow$	$-\frac{\sqrt{2\pi}}{\sigma}\omega^2 G(\omega;\sigma^{-1})$	
Gabor		$\cos(\omega_0 x)G(x;\sigma)$	$\Leftrightarrow$	$\frac{\sqrt{2\pi}}{\sigma}G(\omega\pm\omega_0;\sigma^{-1})$	
unsharp mask		$(1+\gamma)\delta(x)$ $-\gamma G(x;\sigma)$	$\Leftrightarrow$	$(1+\gamma)-$ $\frac{\sqrt{2\pi}\gamma}{\sigma}G(\omega;\sigma^{-1})$	
windowed sinc		$\text{rcos}(x/(aW))$ $\text{sinc}(x/a)$	$\Leftrightarrow$	(see Figure 3.29)	

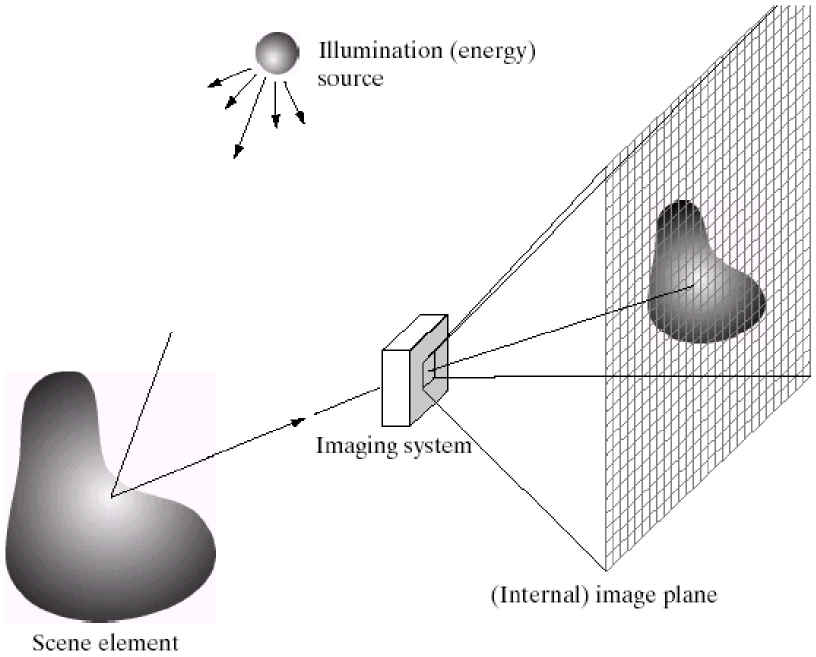
Fourier Transform Sampling Pairs

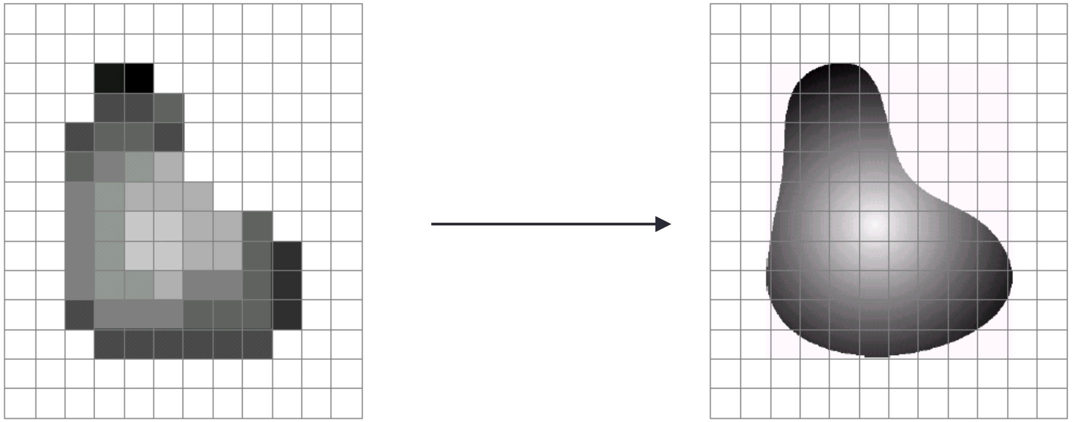


FT of an "impulse train" is an impulse train

Sampling and Aliasing

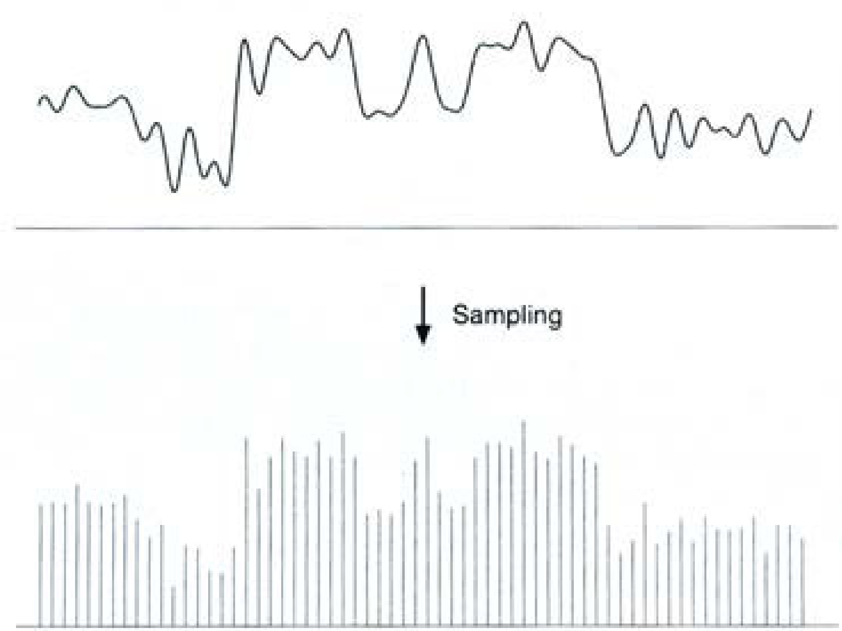
Sampling and Reconstruction





Sampled representations

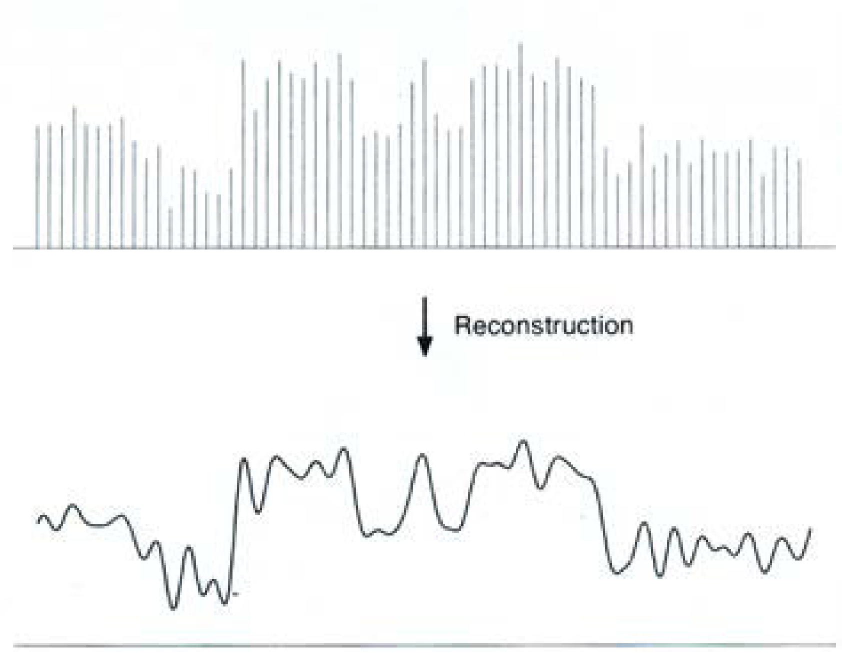
- How to store and compute with continuous functions?
- Common scheme for representation: samples
  - write down the function's values at many points



S. Marschner

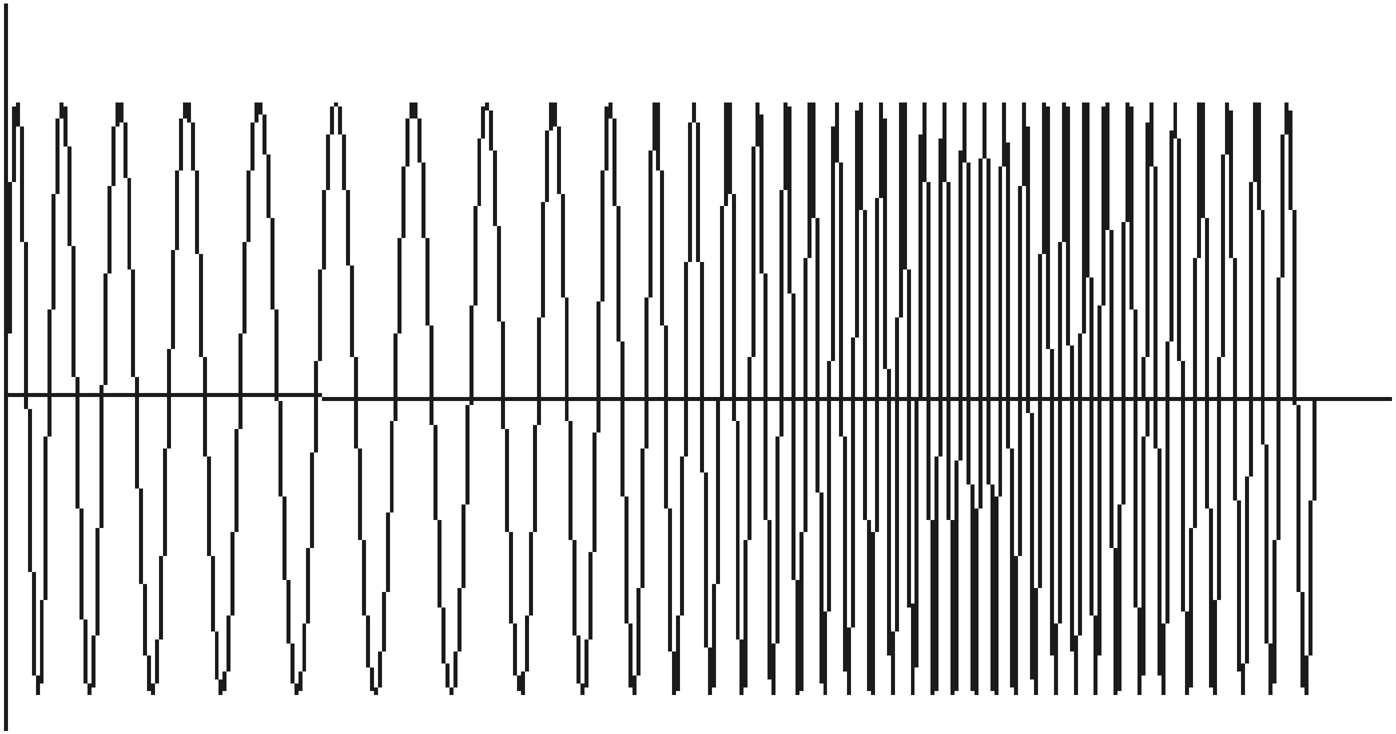
Reconstruction

- Making samples back into a continuous function
  - for output (need realizable method)
  - for analysis or processing (need mathematical method)
  - amounts to "guessing" what the function did in between



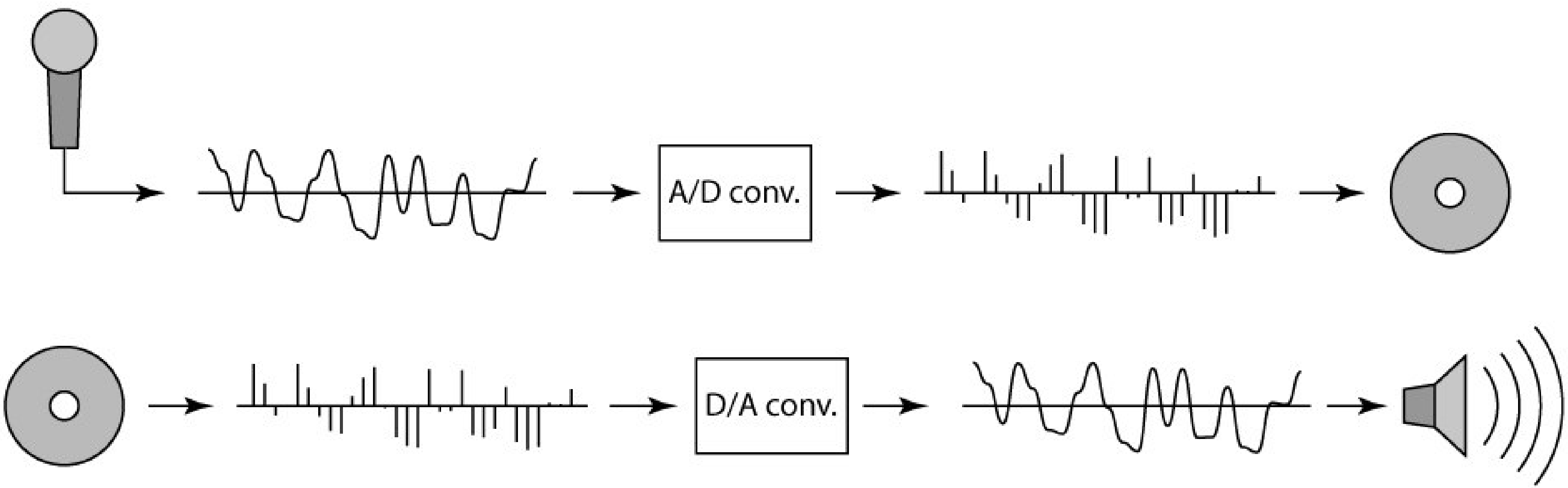
S. Marschner

1D Example: Audio



Sampling in digital audio

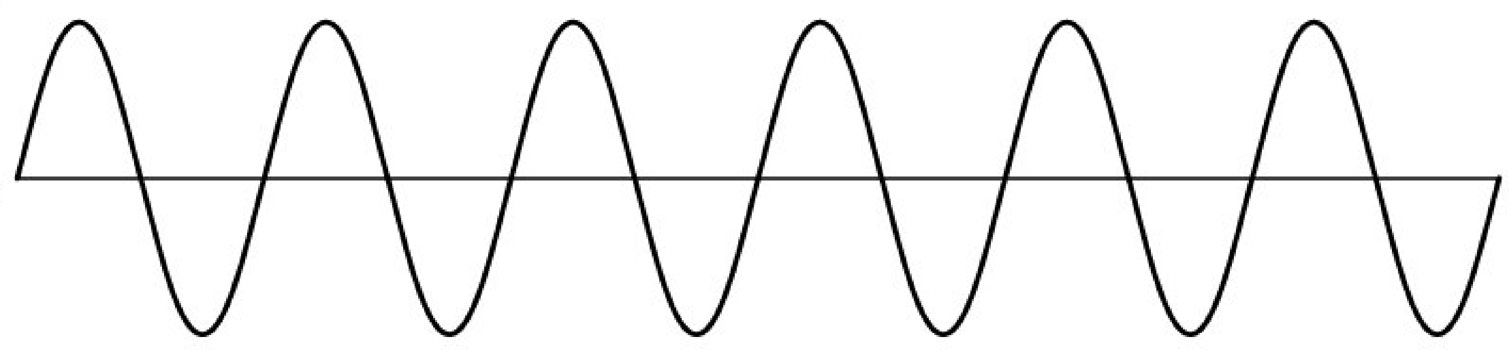
- Recording: sound to analog to samples to disc
- Playback: disc to samples to analog to sound again
  - how can we be sure we are filling in the gaps correctly?



S. Marschner

Sampling and Reconstruction

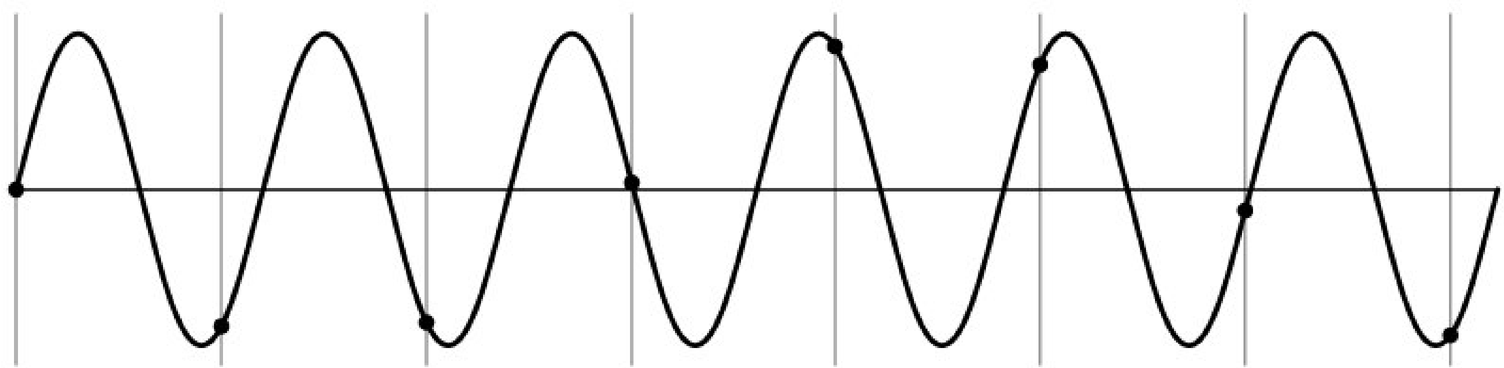
- Simple example: a sign wave



S. Marschner

Undersampling

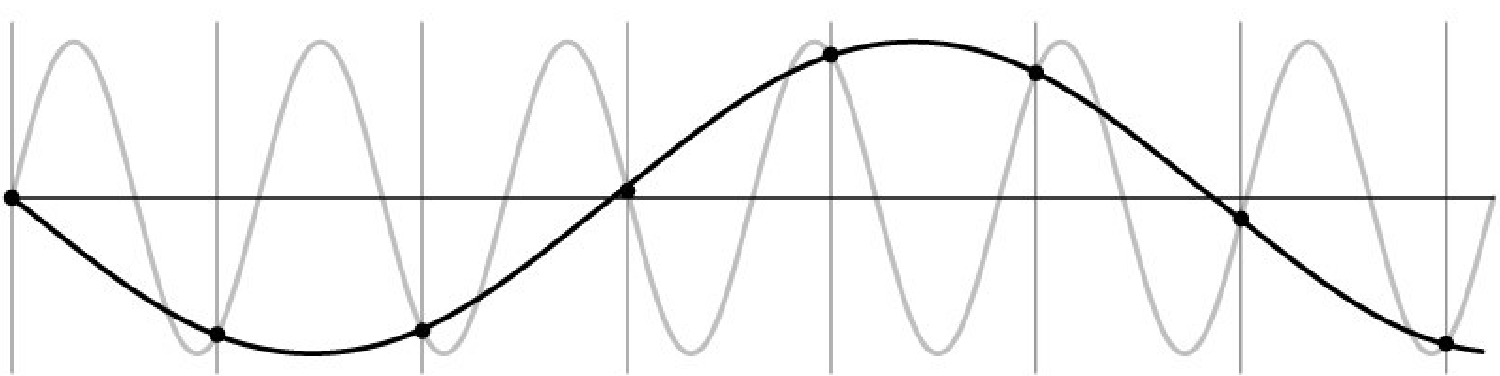
- What if we "missed" things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost



S. Marschner

Undersampling

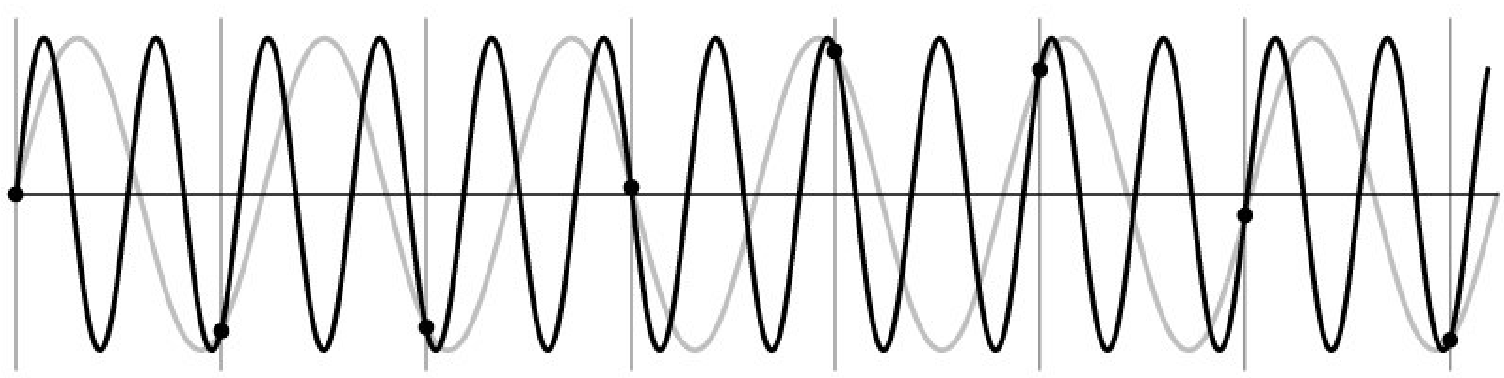
- What if we “missed” things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost
  - surprising result: indistinguishable from lower frequency



S. Marschner

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

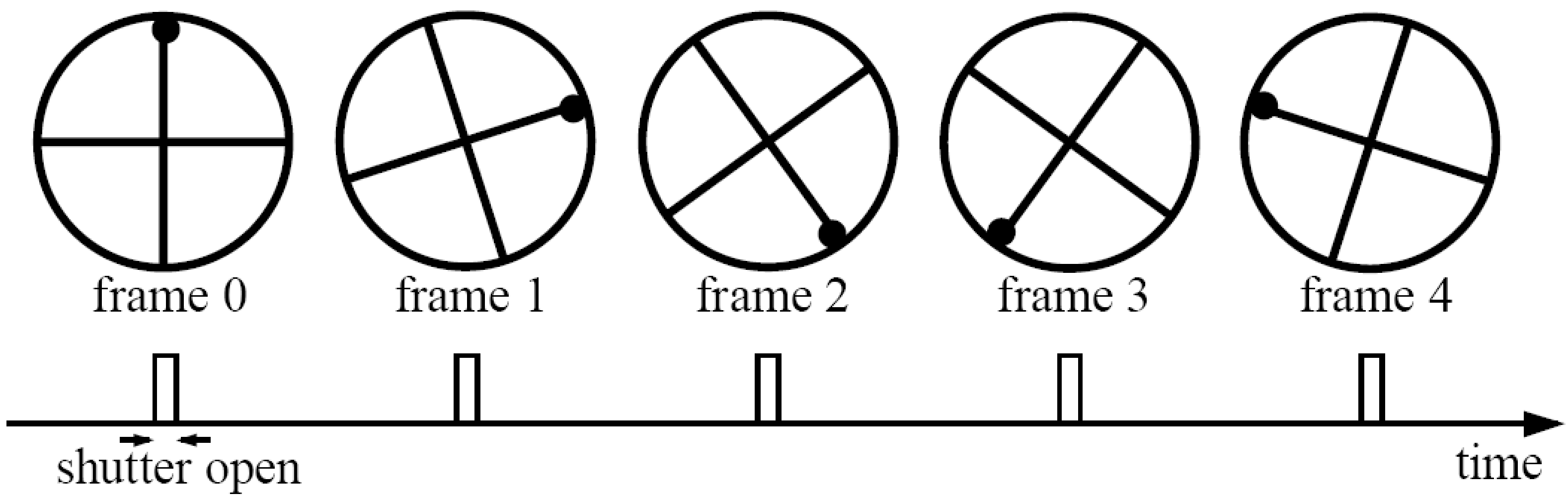


S. Marschner

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)

S. Seitz

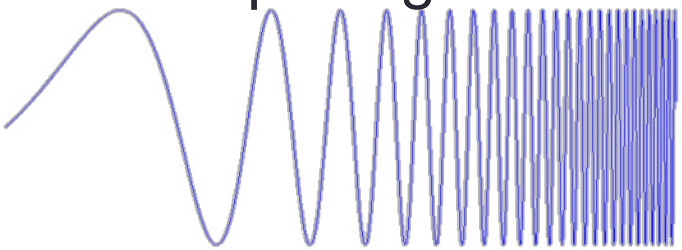
Image sub-sampling

Source: F. Durand

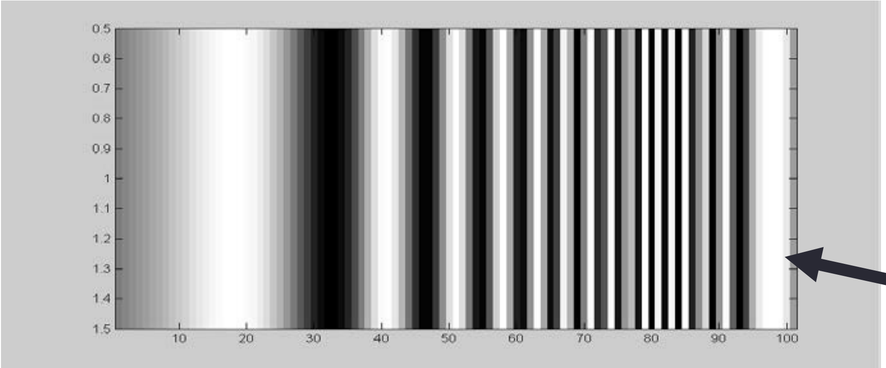
Aliasing in images

What's happening?

Input signal:

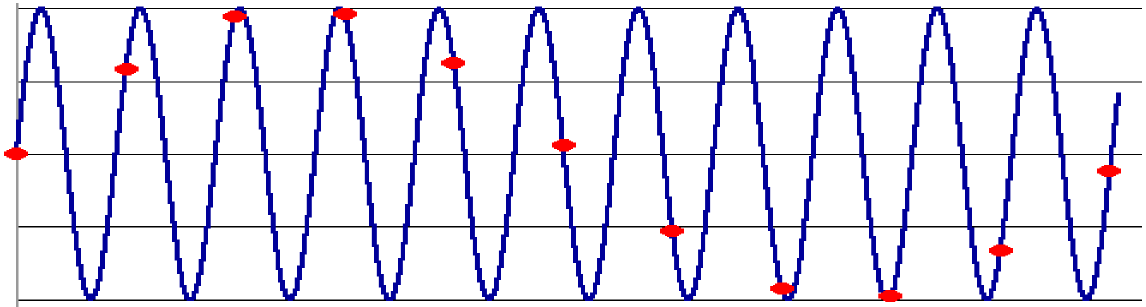


Plot as image:

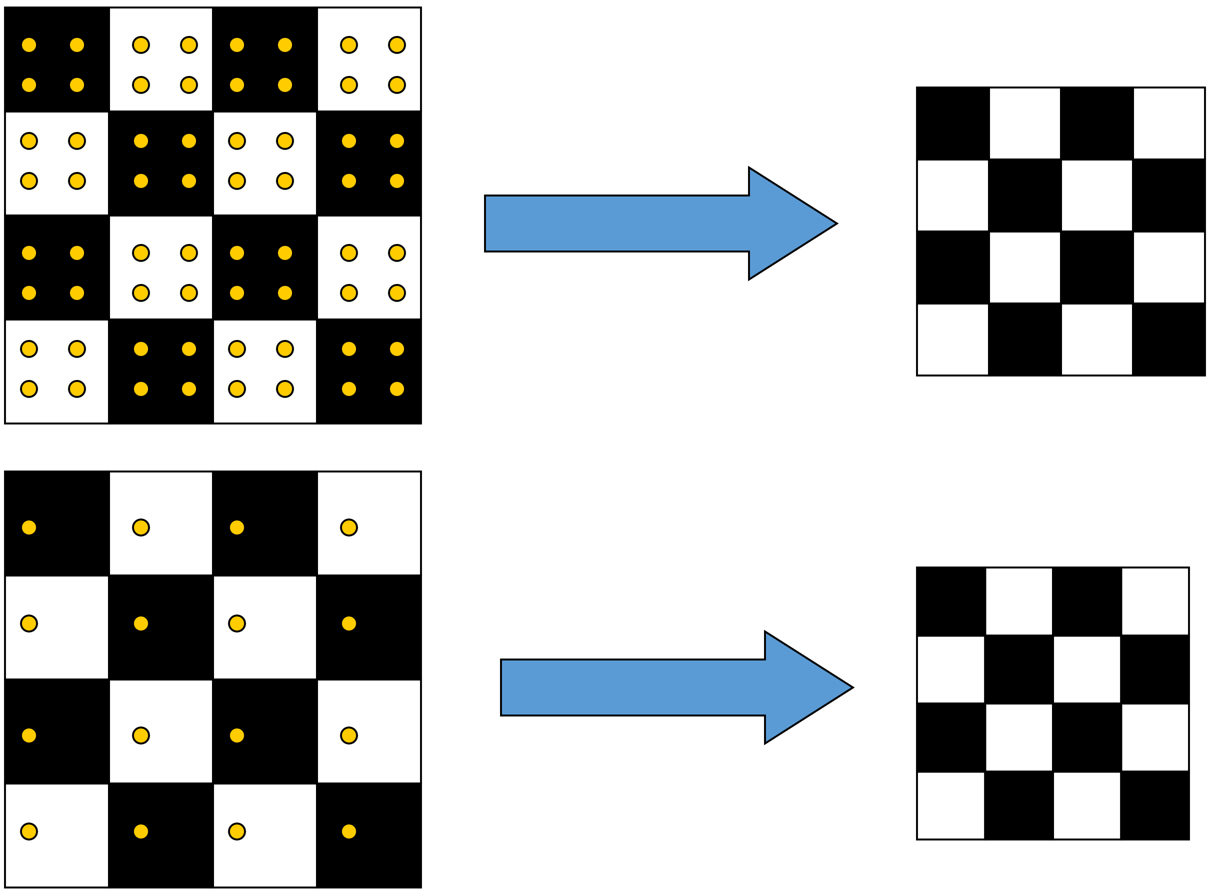


x = 0:.05:5; imagesc(sin((2.^x).*x))

Alias!
Not enough samples

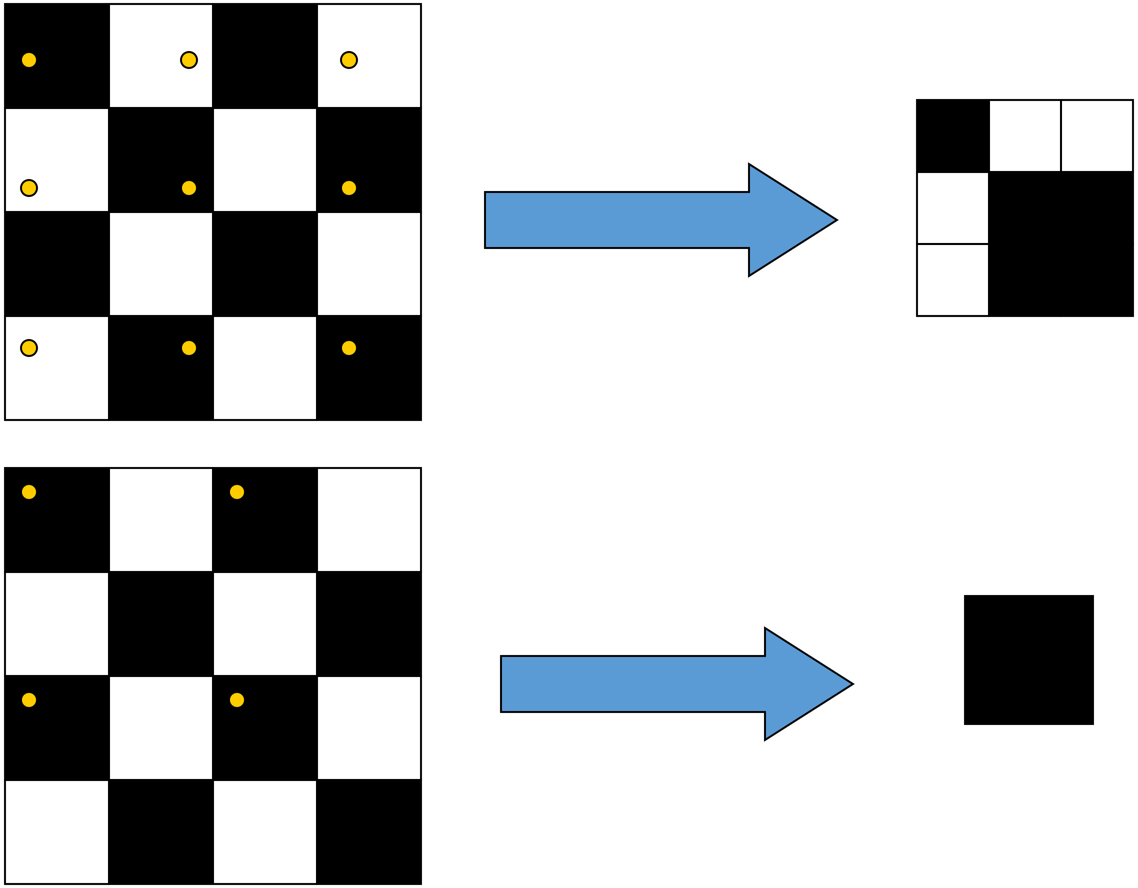


Sampling an image



Examples of GOOD sampling

Undersampling



Examples of BAD sampling -> Aliasing

Antialiasing

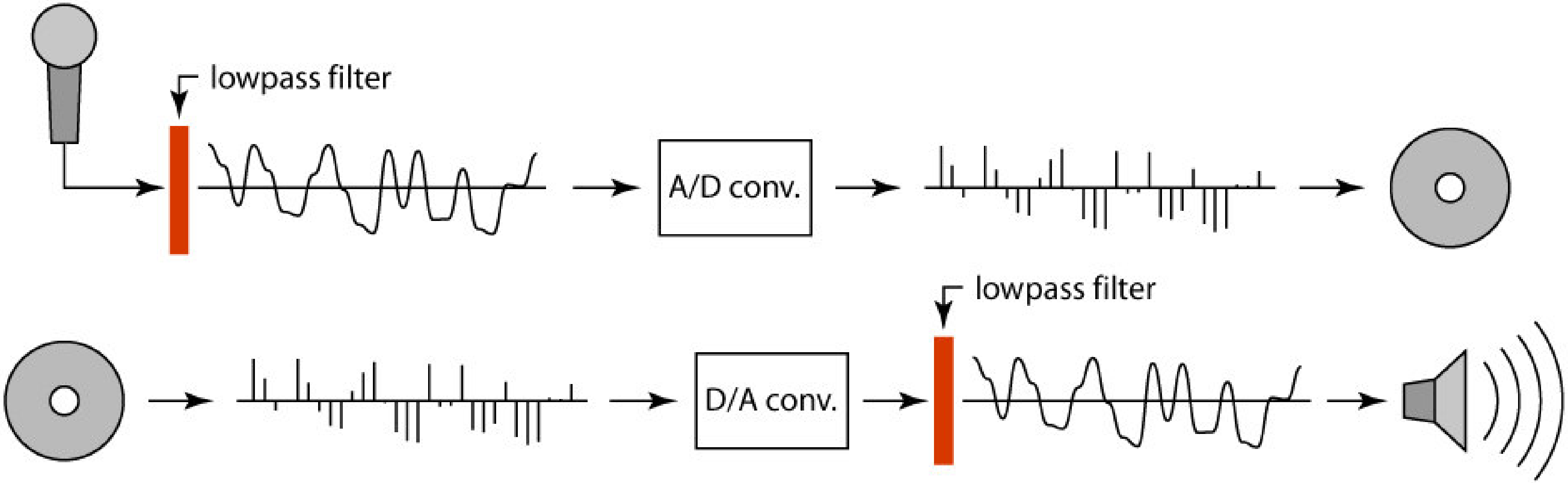
- What can we do about aliasing?

- Sample more often
  - Join the Mega-Pixel craze of the photo industry
  - But this can't go on forever

- Make the signal less "wiggly"
  - Get rid of some high frequencies
  - Will loose information
  - But it's better than aliasing

Preventing aliasing

- Introduce lowpass filters:
  - remove high frequencies leaving only safe, low frequencies
  - choose lowest frequency in reconstruction (disambiguate)



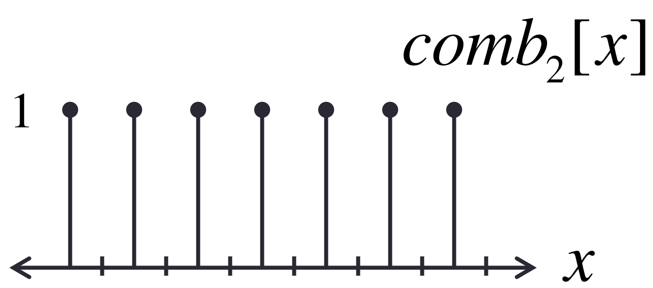
S. Marschner

Impulse Train

- Define a *comb* function (impulse train) in 1D as follows

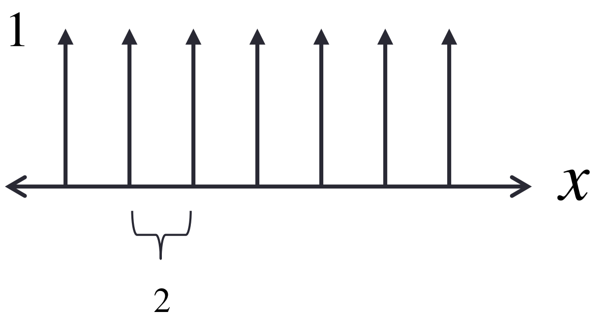
$$comb_M[x] = \sum_{k=-\infty}^{\infty} \delta[x - kM]$$

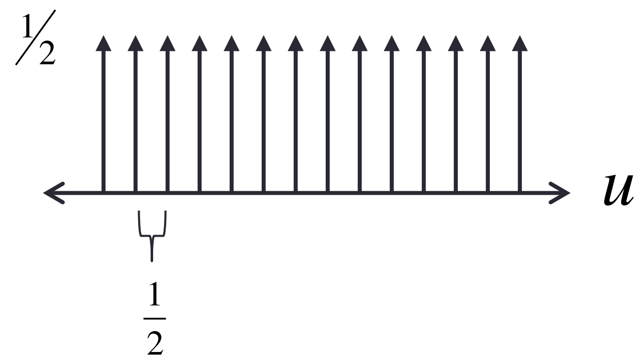
where *M* is an integer



Impulse Train in 1D

$$comb_2(x) \quad \Leftrightarrow \quad \frac{1}{2} comb_{\frac{1}{2}}(u)$$





Remember:

Scaling $\quad f(ax) \quad \quad \frac{1}{|a|} F\left(\frac{u}{a}\right)$

B.K. Gunturk

Impulse Train in 2D *(bed of nails)*

$$comb_{M,N}(x,y) \equiv \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty}\delta\left(x-kM, y-lN\right)$$

- Fourier Transform of an impulse train is also an impulse train:

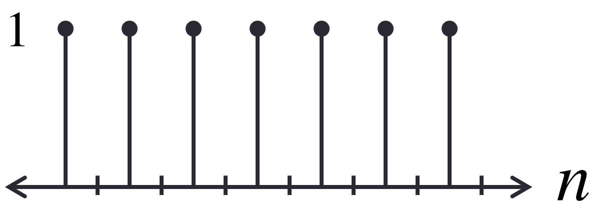
$$\underbrace{\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty}\delta\left(x-kM, y-lN\right)}_{comb_{M,N}(x,y)} \Leftrightarrow \underbrace{\frac{1}{MN}\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty}\delta\left(u-\frac{k}{M}, v-\frac{l}{N}\right)}_{comb_{\frac{1}{M},\frac{1}{N}}(u,v)}$$

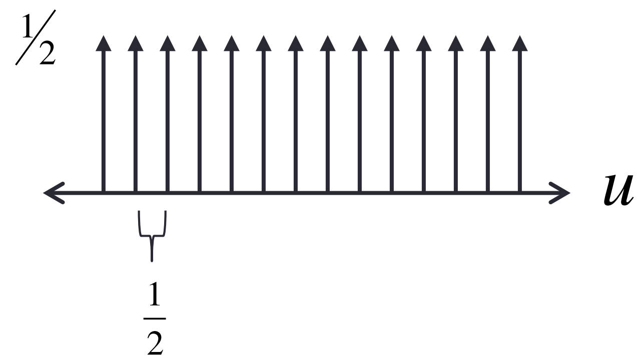
As the comb samples get further apart, the spectrum samples get closer together!

B.K. Gunturk

Impulse Train

$$comb_2[n] \quad \Leftrightarrow \quad \frac{1}{2} comb_{\frac{1}{2}}(u)$$



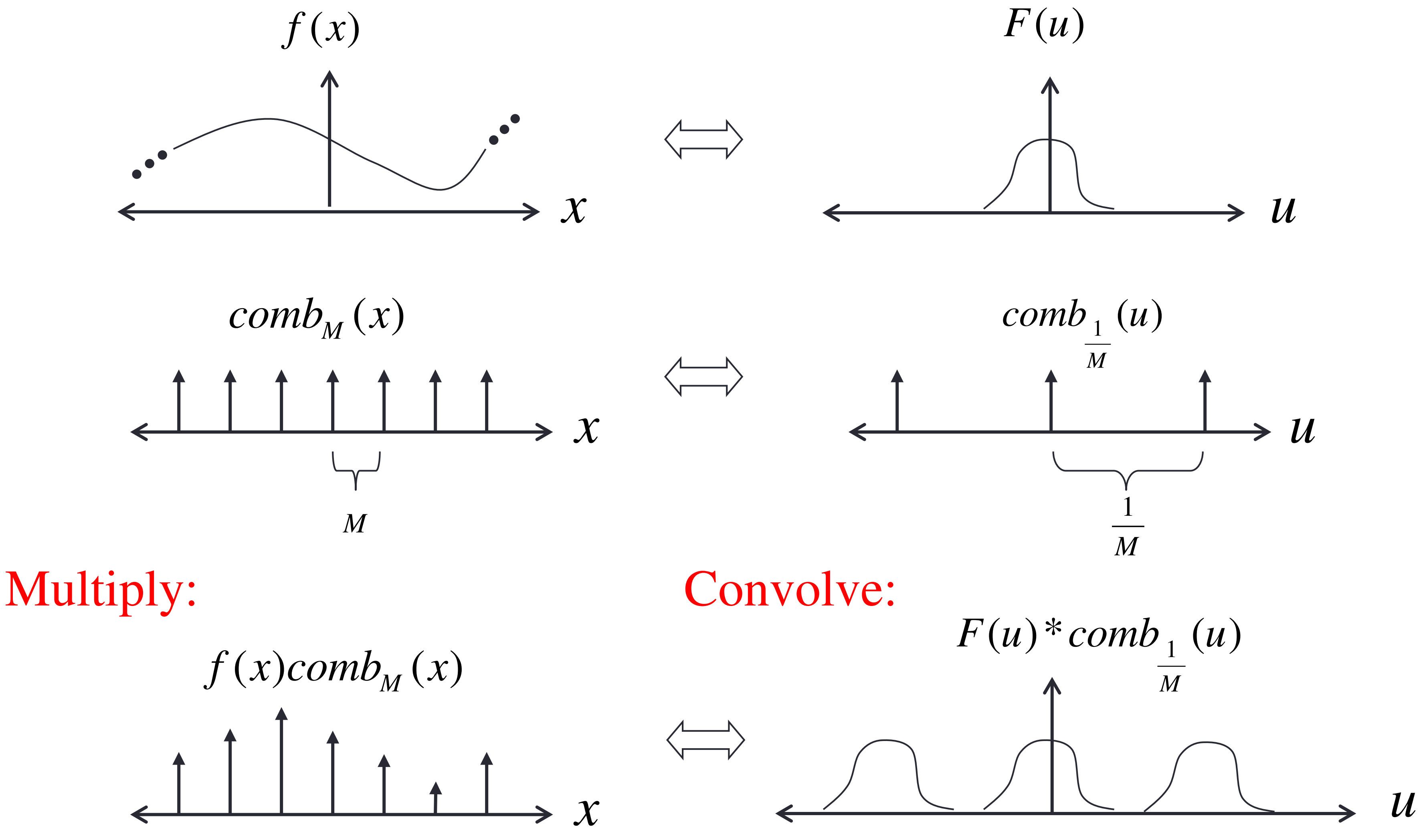


Remember:

Scaling $\quad f(ax) \quad \Leftrightarrow \quad \frac{1}{|a|} F\left(\frac{u}{a}\right)$

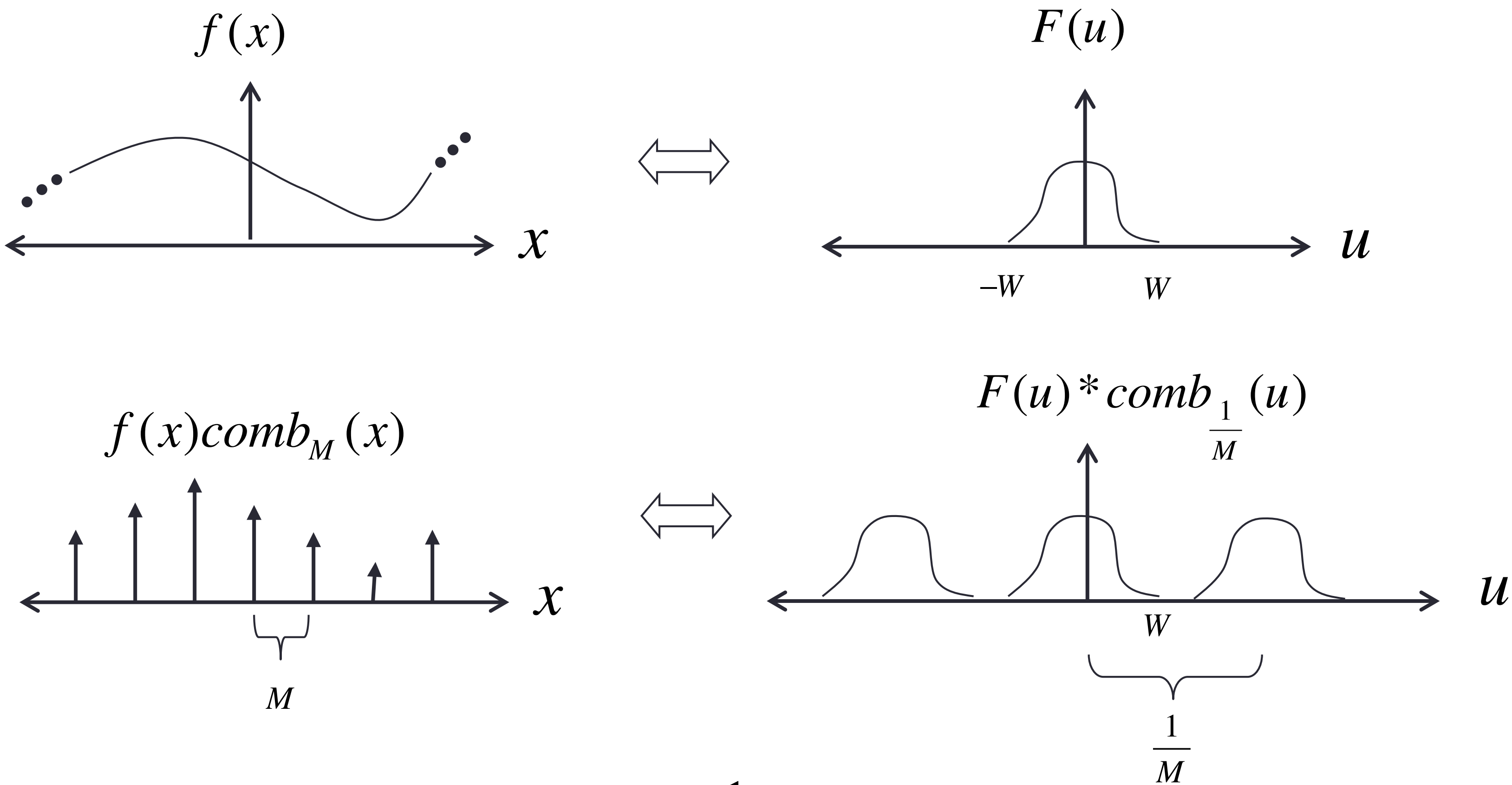
B.K. Gunturk

Sampling low frequency signal



B.K. Gunturk

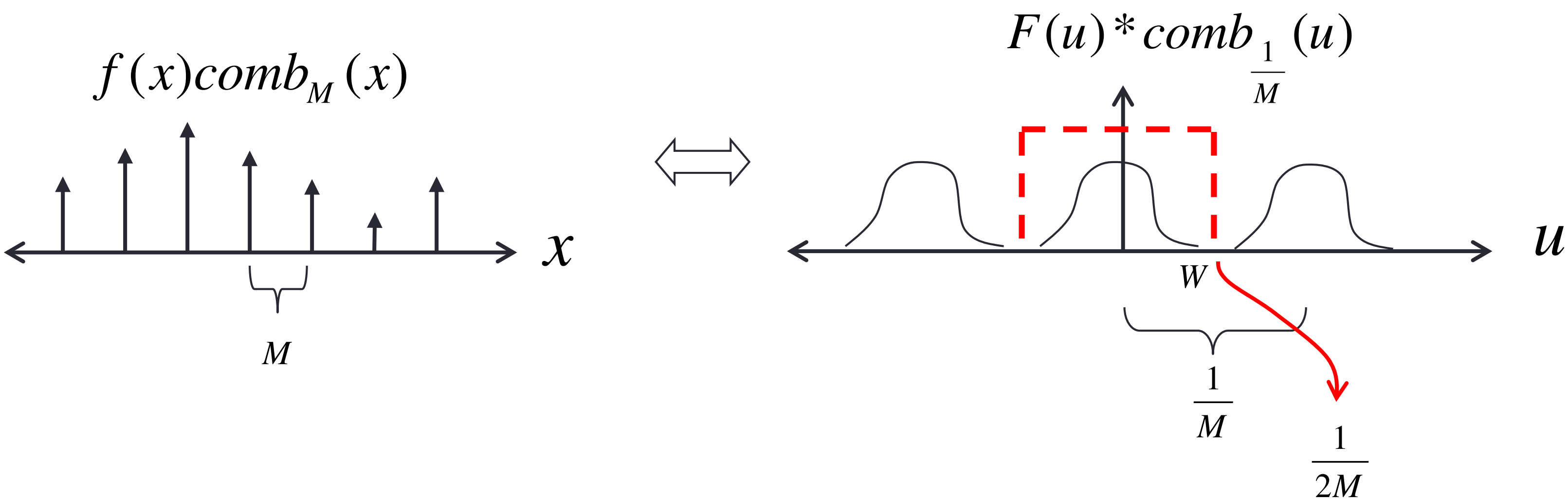
Sampling low frequency signal



No "problem" if $\frac{1}{M} > 2W$

B.K. Gunturk

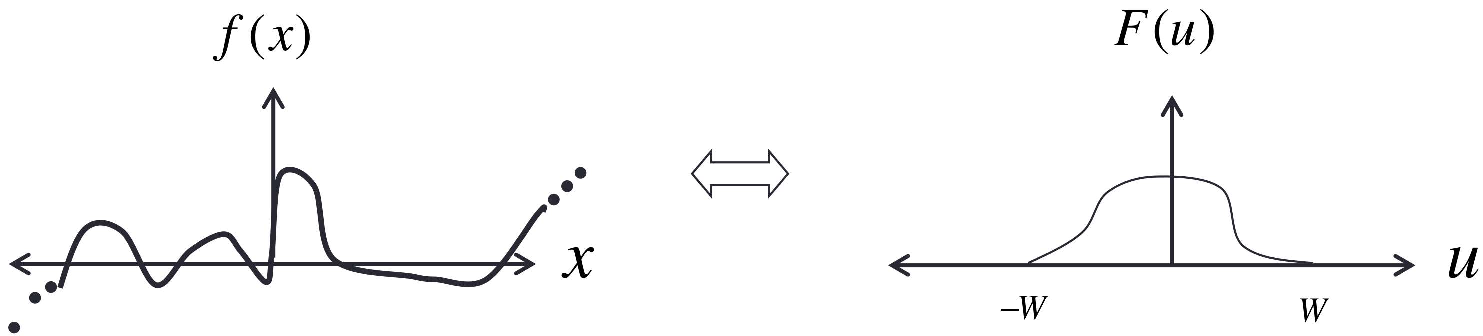
Sampling low frequency signal

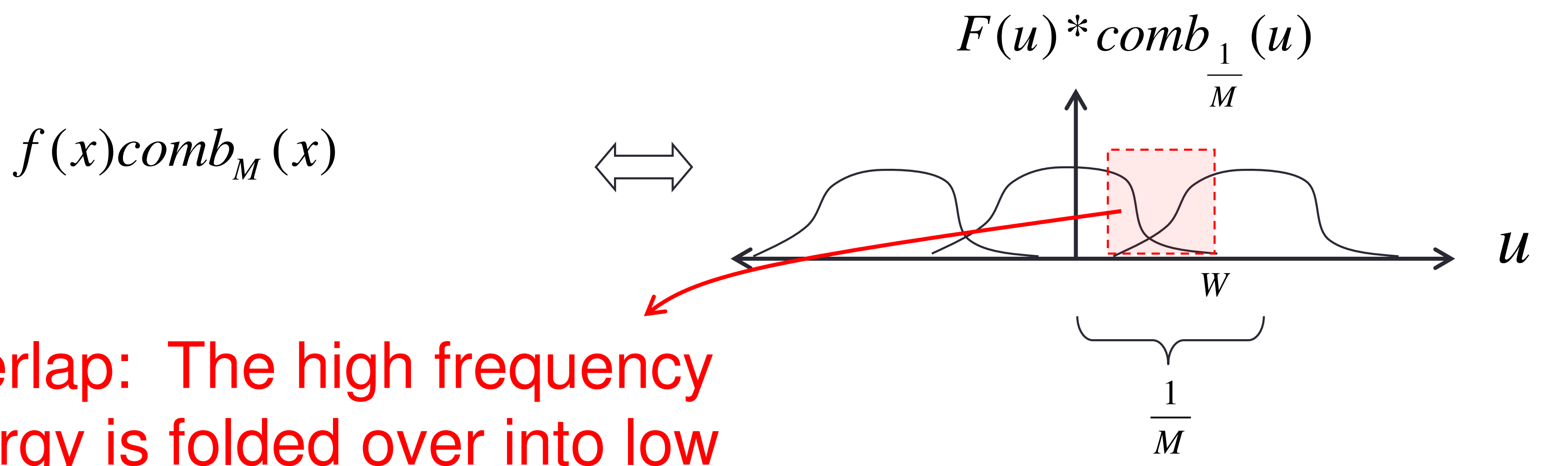


If there is no overlap, the original signal can be recovered from its samples by low-pass filtering.

B.K. Gunturk

Sampling high frequency signal

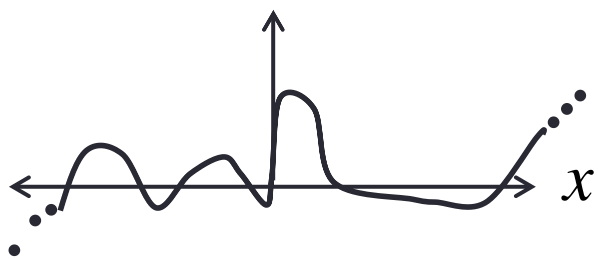




Overlap: The high frequency energy is folded over into low frequency. It is "aliasing" as lower frequency energy. And you cannot fix it once it has happened.

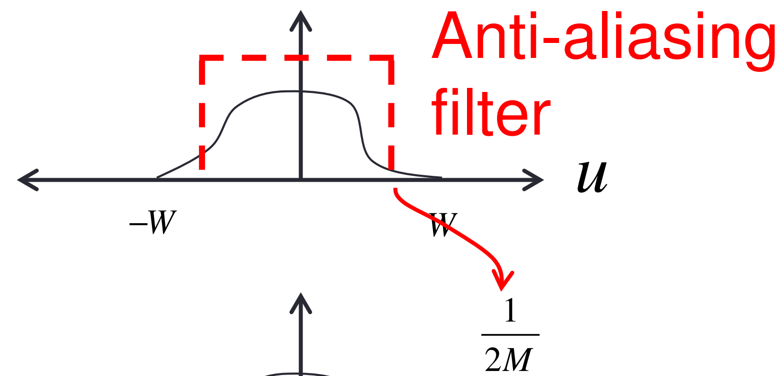
Sampling high frequency signal

$f(x)$



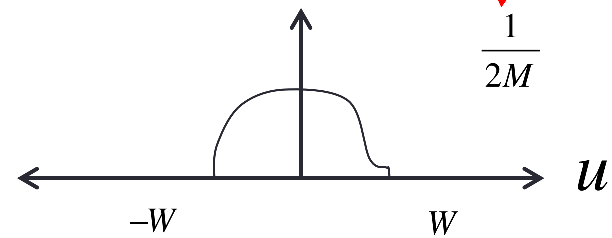
$\Longleftrightarrow$

$F(u)$



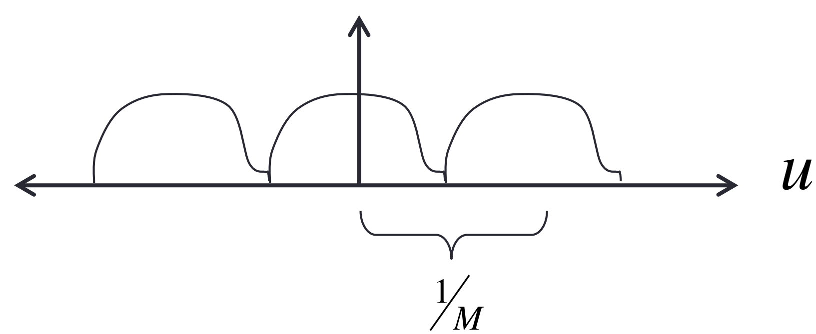
$$f(x) * h(x)$$

$\Longleftrightarrow$



$$\left[f(x) * h(x)\right] comb_M(x)$$

$\Longleftrightarrow$

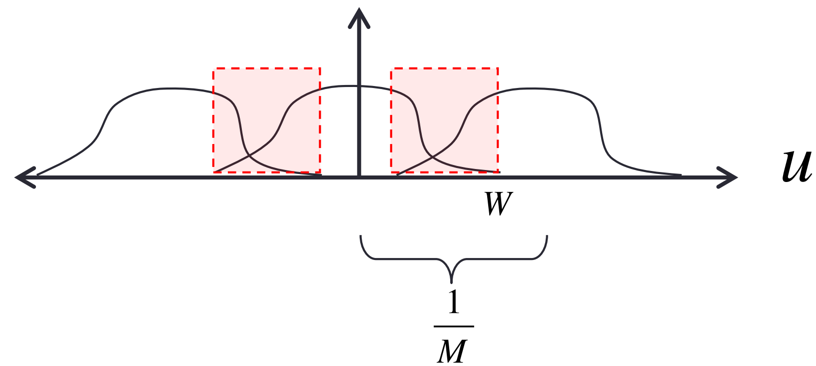


B.K. Gunturk

Sampling high frequency signal

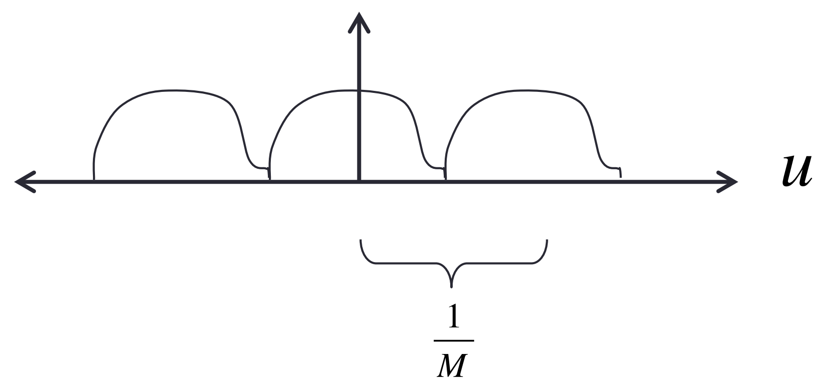
- Without anti-aliasing filter:

$$f(x)comb_M(x) \Longleftrightarrow$$



- With anti-aliasing filter:

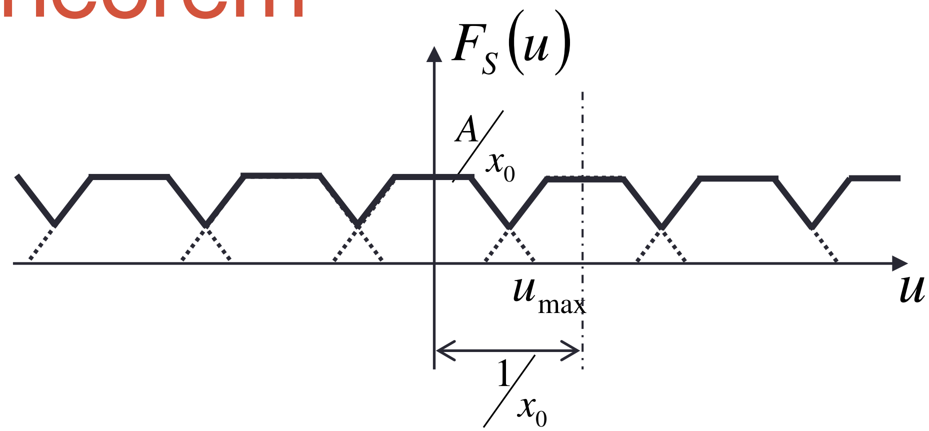
$$[f(x)*h(x)]comb_M(x) \Longleftrightarrow$$



B.K. Gunturk

Nyquist Theorem

If $u_{\max} > \dfrac{1}{2x_0}$



Aliasing

When can we recover $F(u)$ from $F_S(u)$?

$\boxed{\text{Only if } u_{\max} \le \dfrac{1}{2x_0} \text{ (Nyquist Frequency)}}$

We can use

$$C(u) = \begin{cases} x_0 & |u| < 1/2x_0 \\ 0 & \text{otherwise} \end{cases}$$

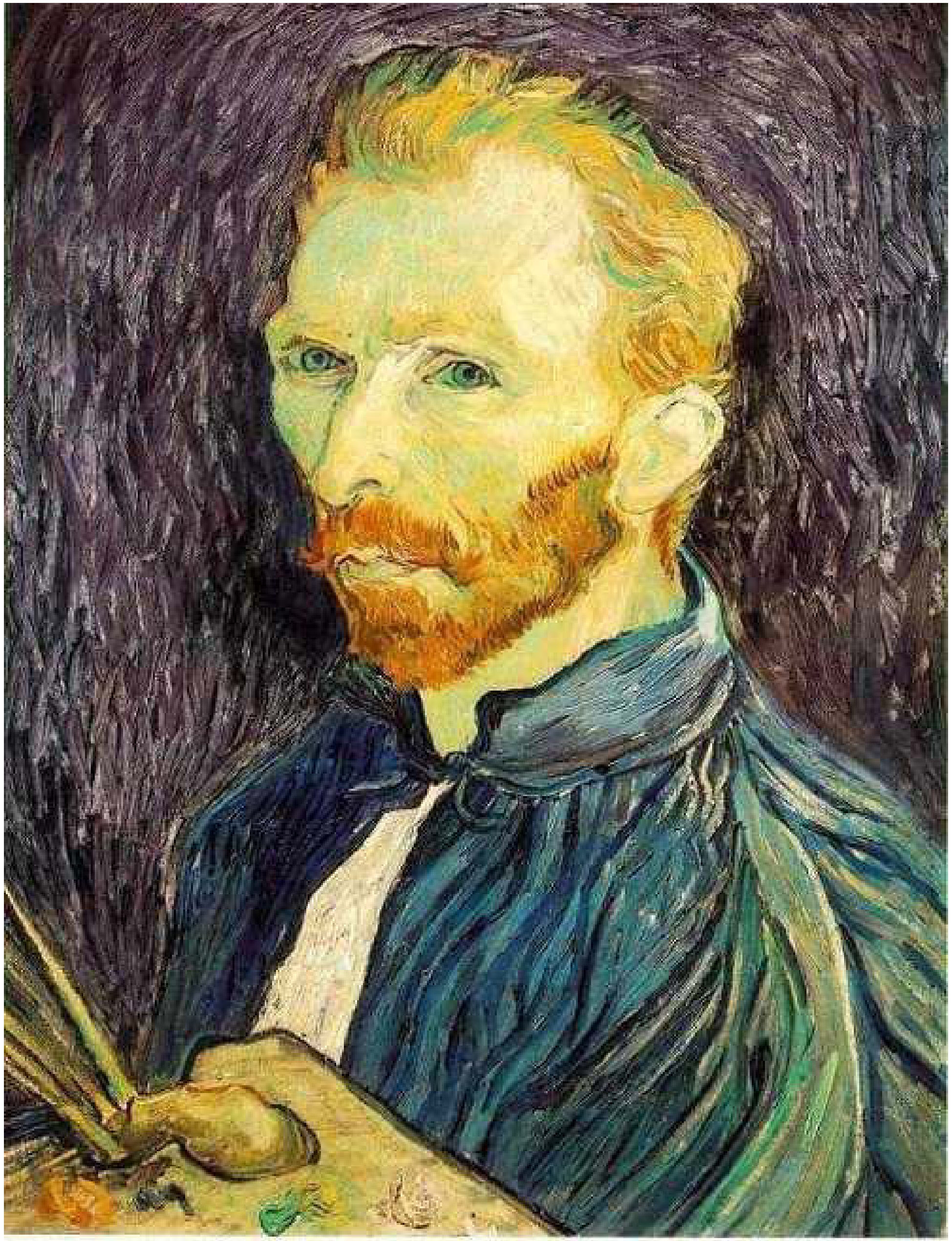
Then $F(u) = F_S(u)C(u)$ and $f(x) = \mathrm{IFT}[F(u)]$

$\boxed{\text{Sampling frequency must be greater than } 2u_{\max}}$

Image half-sizing

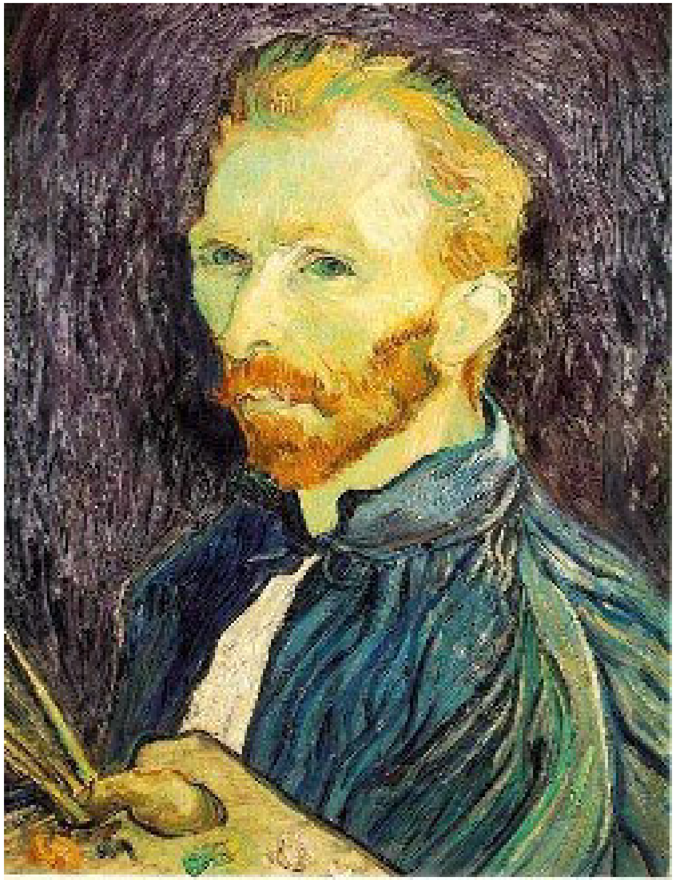
This image is too big to fit on the screen. How can we reduce it?

How to generate a half-sized version?



S. Seitz

Image sub-sampling



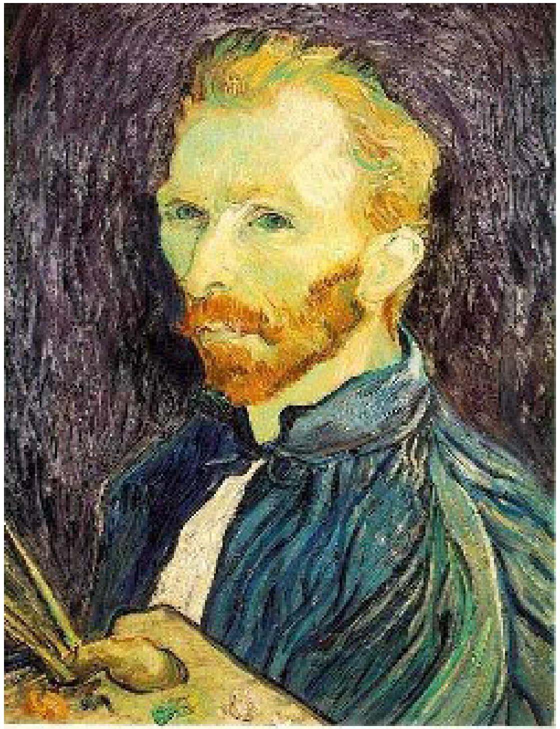
1/4

1/8

Throw away every other row and column to create a 1/2 size image
- called *image sub-sampling*

S. Seitz

Image sub-sampling



1/2

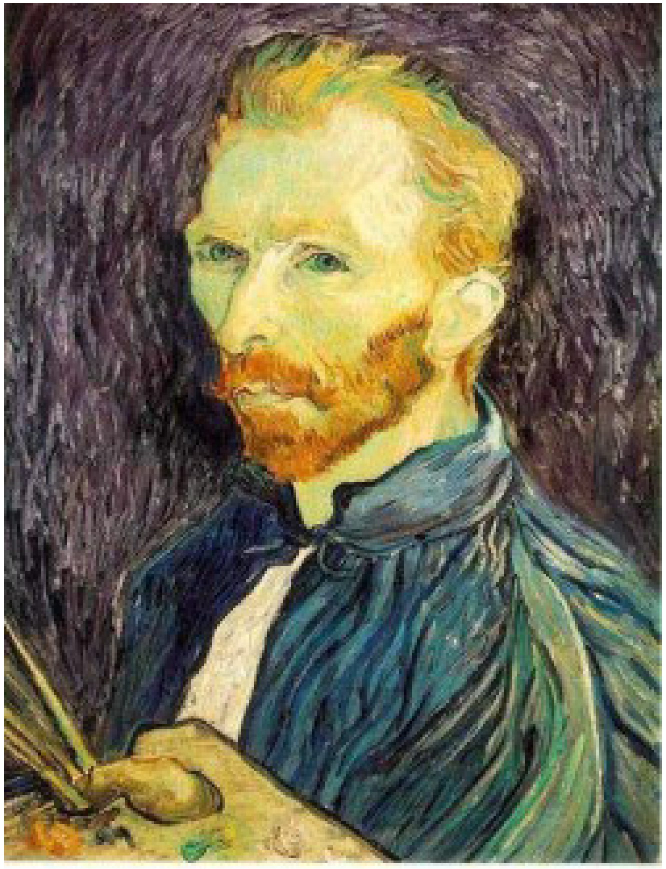
1/4 (2x zoom)

1/8 (4x zoom)

Aliasing! What do we do?

S. Seitz

Gaussian (lowpass) pre-filtering



Gaussian 1/2

G 1/4

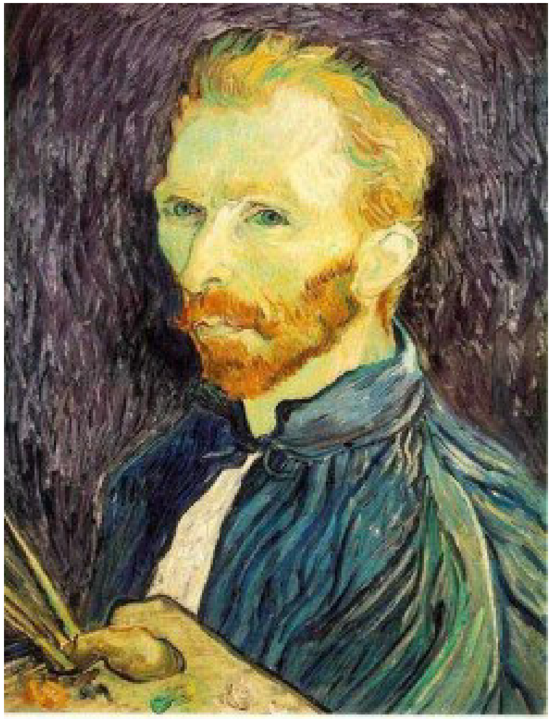
G 1/8

Solution: filter the image, *then* subsample

- Filter size should double for each ½ size reduction. Why?

S. Seitz

Subsampling with Gaussian pre-filtering



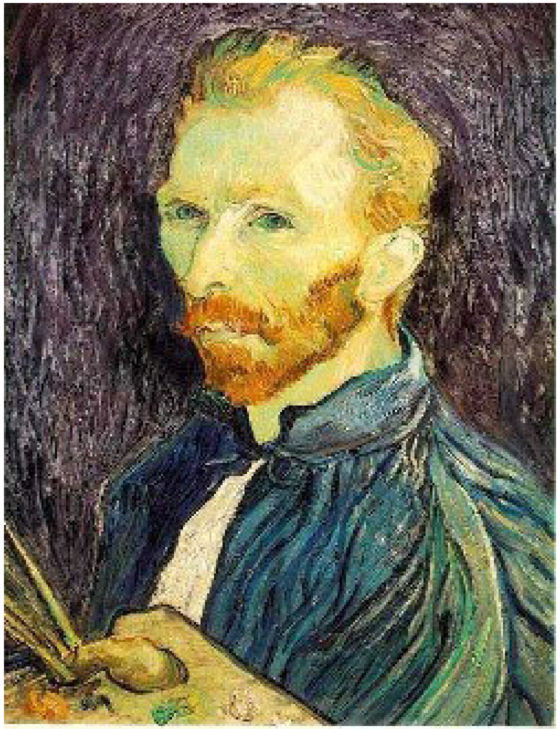
Gaussian 1/2

G 1/4

G 1/8

S. Seitz

Compare with...



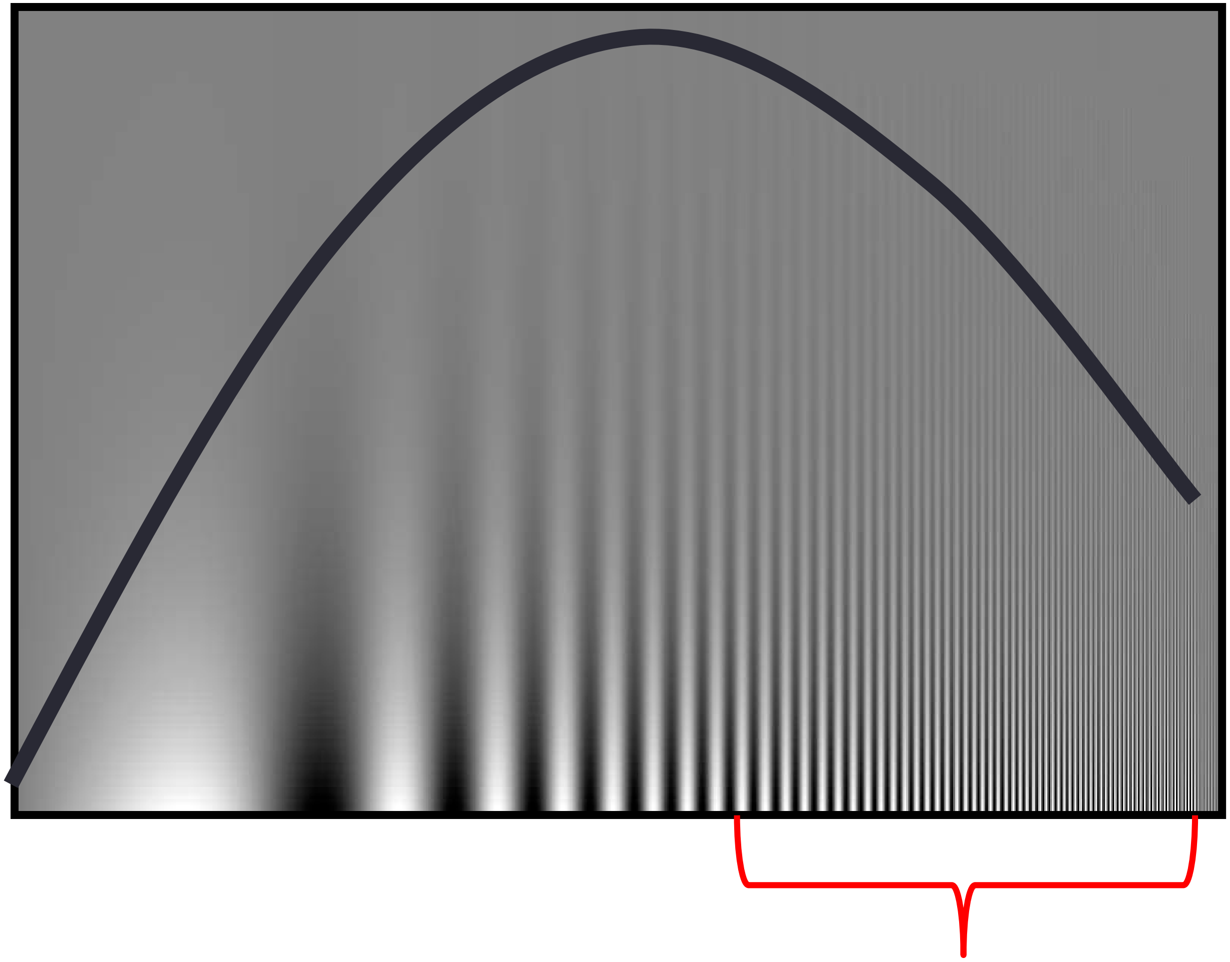
1/2

1/4 (2x zoom)

1/8 (4x zoom)

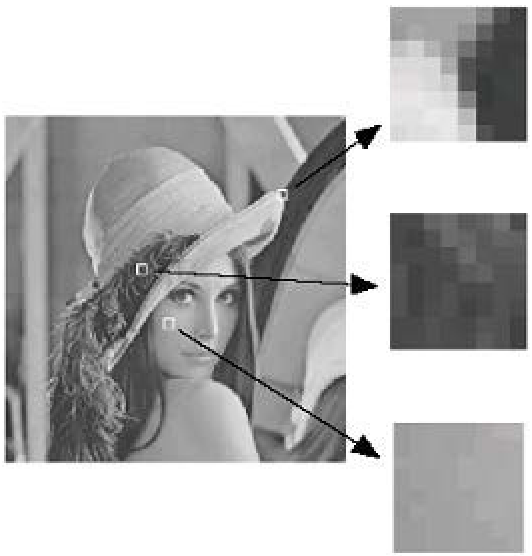
S. Seitz

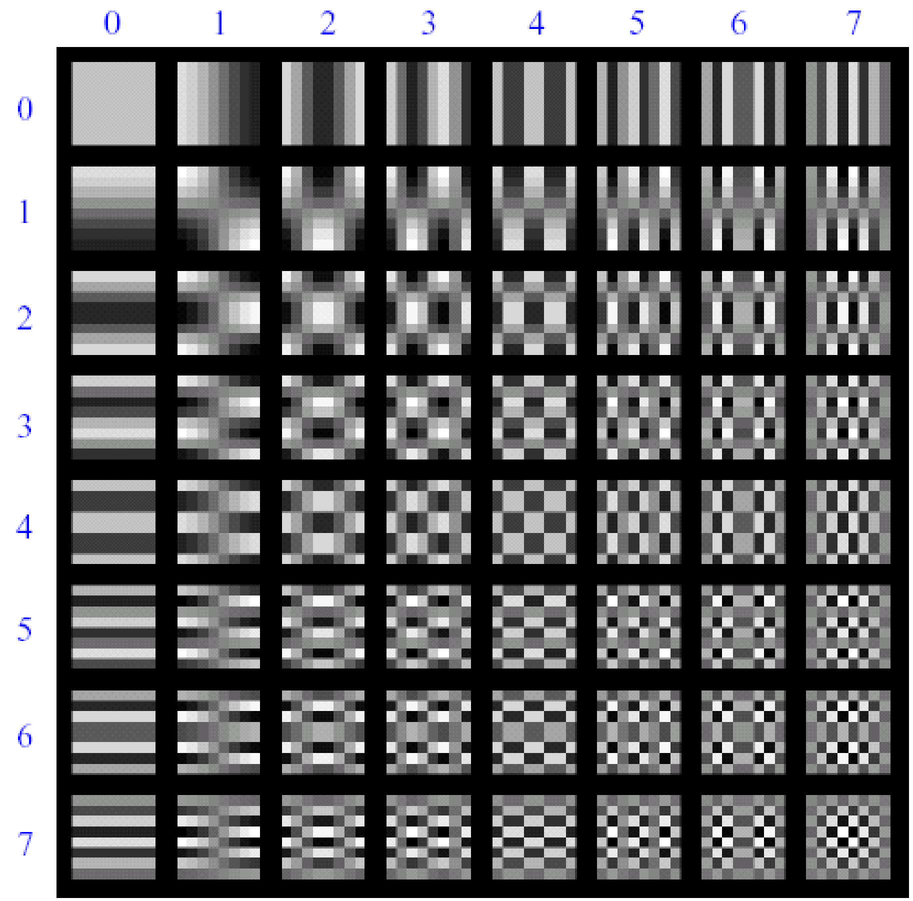
Campbell-Robson contrast sensitivity curve



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

Lossy Image Compression (JPEG)





Block-based Discrete Cosine Transform (DCT) on 8x8