Fourier Transform in Image Processing

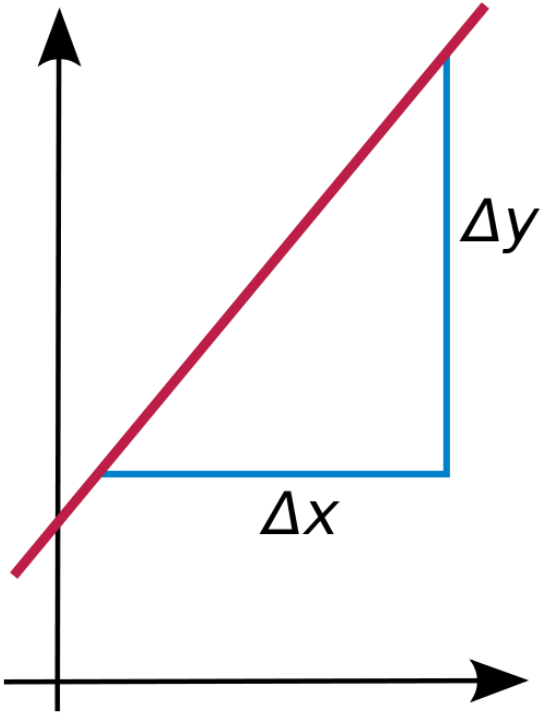
CS6640, Fall 2012

Guest Lecture

Marcel Prastawa, SCI Utah

Preliminaries

Function Representation



Linear function:

$$f(x) = mx + b \qquad m = \Delta x / \Delta y$$

Rewrite as:

$$f(x) = \tan(\theta) + b$$

Provides intuitive description of linear functions:

- Angles
- Shifts

How to do this for generic functions?

Basis Decomposition

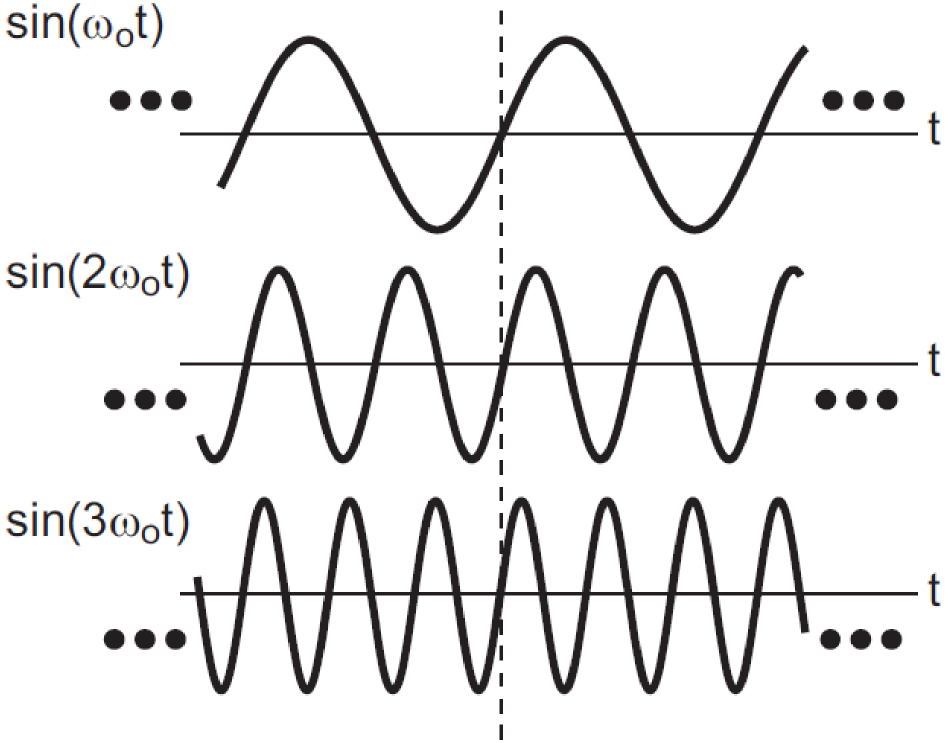
- Write a function as a weighted sum of basis functions

$$f(x) = \Sigma w_i B_i(x)$$

- What is a good set of basis functions?
- How do you determine the weights?

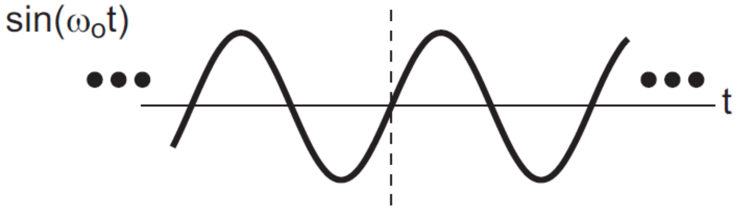
Sine Waves

- Use sine waves of different frequencies as basis functions?



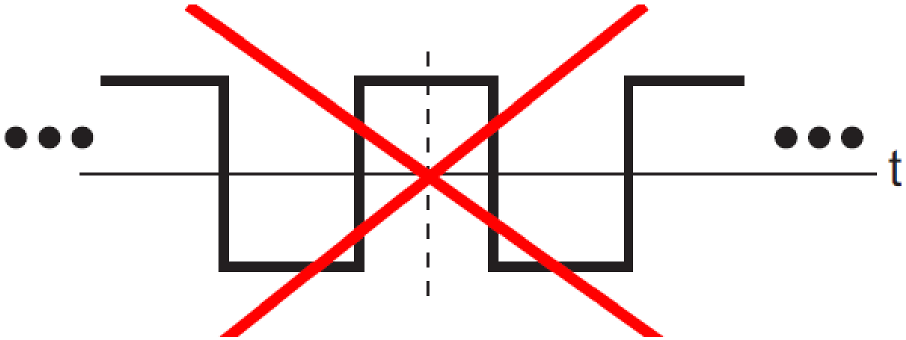
Limitation of Sines

- Sines are odd / anti-symmetric:



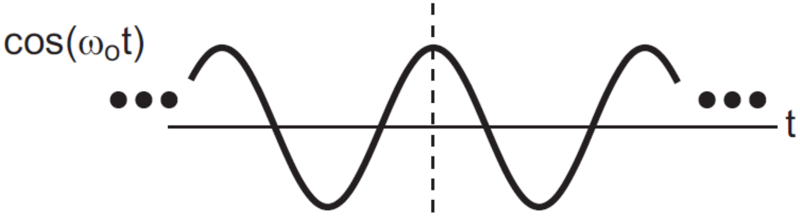
$$f(t) = -f(-t)$$

- Sine basis cannot create even functions:



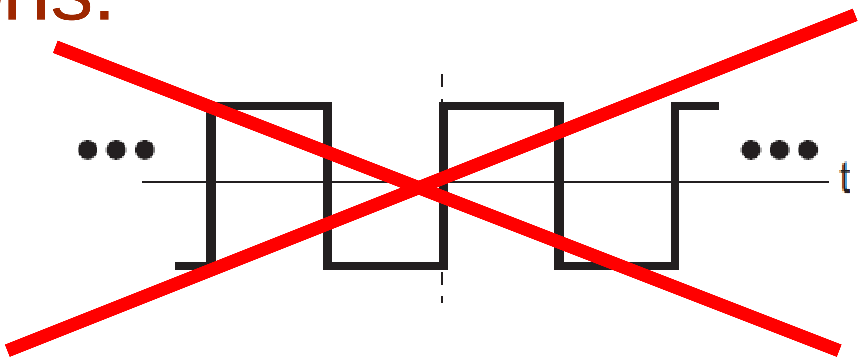
Limitation of Cosines

- Cosines are even / symmetric functions:



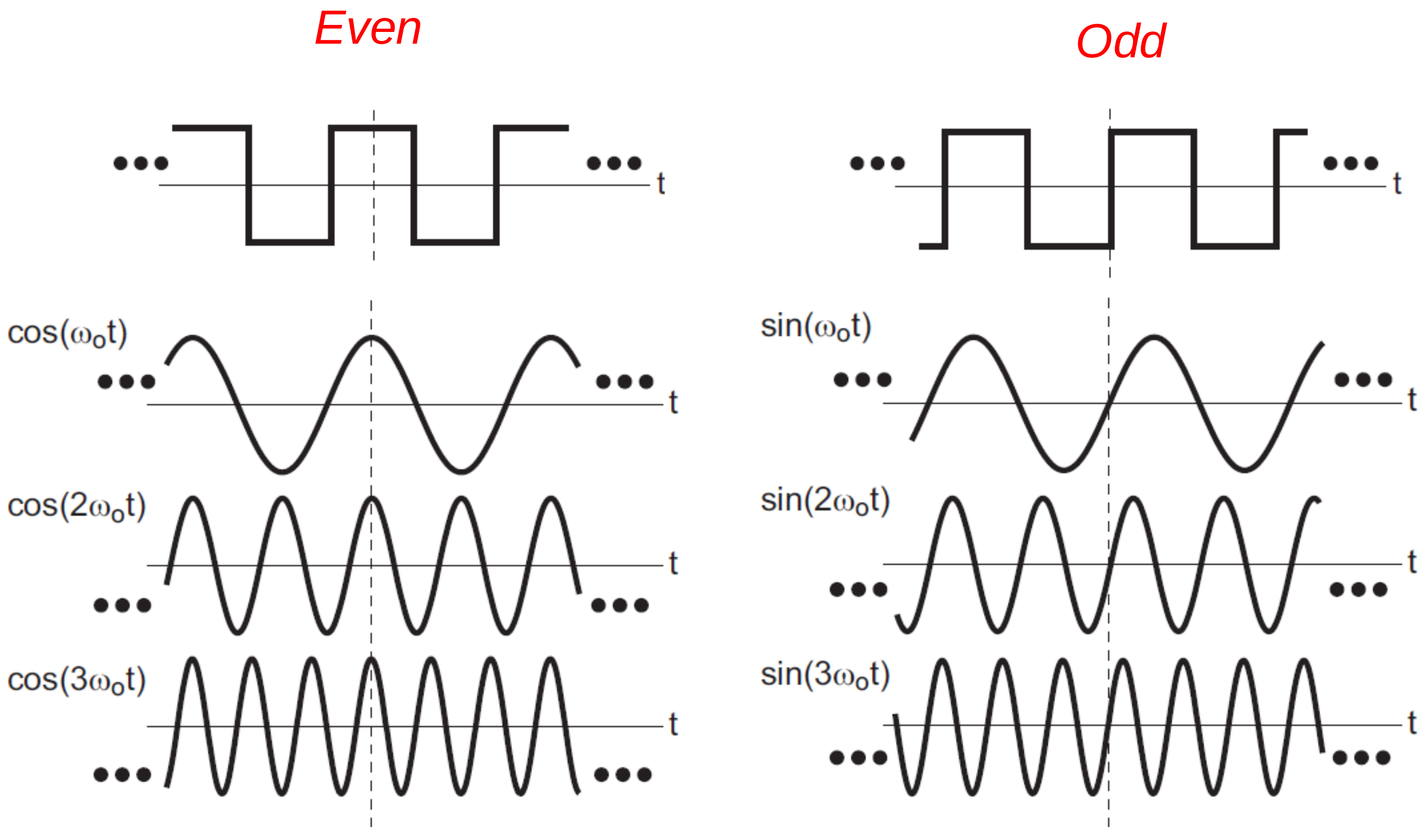
$$f(t) = f(-t)$$

- Cosine basis cannot create odd functions:



Combine Cosines and Sines

- Allow creation of both even and odd functions with different combinations:



Why Sines and Cosines?

- Represent functions as a combination of basis with different frequencies
- Intuitive description of signals / images:
  - how much high frequency content?
  - what do the low freq. content look like?
- Image processing “language”:
  - remove noise by reducing high freq content
  - explains sampling / perception phenomena

The Fourier Transform

Reminder: Euler's Identity

- From calculus

$$e^{jx} = \cos x + j\sin x$$

- j is the imaginary part of a complex number

Fourier Transform

- Forward, mapping to frequency domain:

$$F(s) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi st} dt$$

- Backward, inverse mapping to time domain:

$$f(t) = \int_{-\infty}^{\infty} F(s) e^{+j2\pi st} ds$$

Space and Frequency

Fourier Transform

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx$$

Fourier Synthesis

Fourier Analysis

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$$

Inverse Fourier Transform

Fourier Series

- Projection or change of basis
- Coordinates in Fourier basis:

$$c_n = \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t) e^{-j\frac{2\pi n}{T}t} dt$$

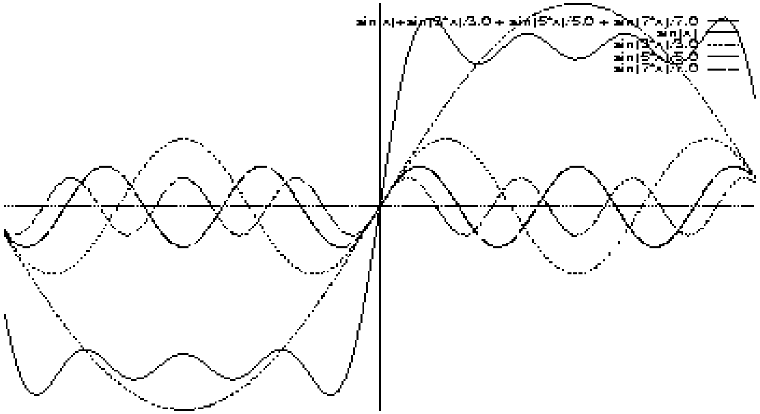
- Rewrite f as:

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi n}{T}t}$$

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \sin\left(j2\pi \frac{n}{T} t\right) + \sum_{n=1}^{\infty} b_n \cos\left(j2\pi \frac{n}{T} t\right)$$

Example: Step Function

Step function as sum of infinite sine waves



Discrete Fourier Transform

$$F_n = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} f_i e^{-j2\pi \frac{n}{N} t}$$

$$f_i = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} F_n e^{j2\pi \frac{n}{N} t}$$

Fourier Basis

- Why Fourier basis?
  - Can represent integrable functions with finite support
- Also
  - Orthonormal in [-pi, pi]
  - Periodic signals with different frequencies
  - Continuous, differentiable basis

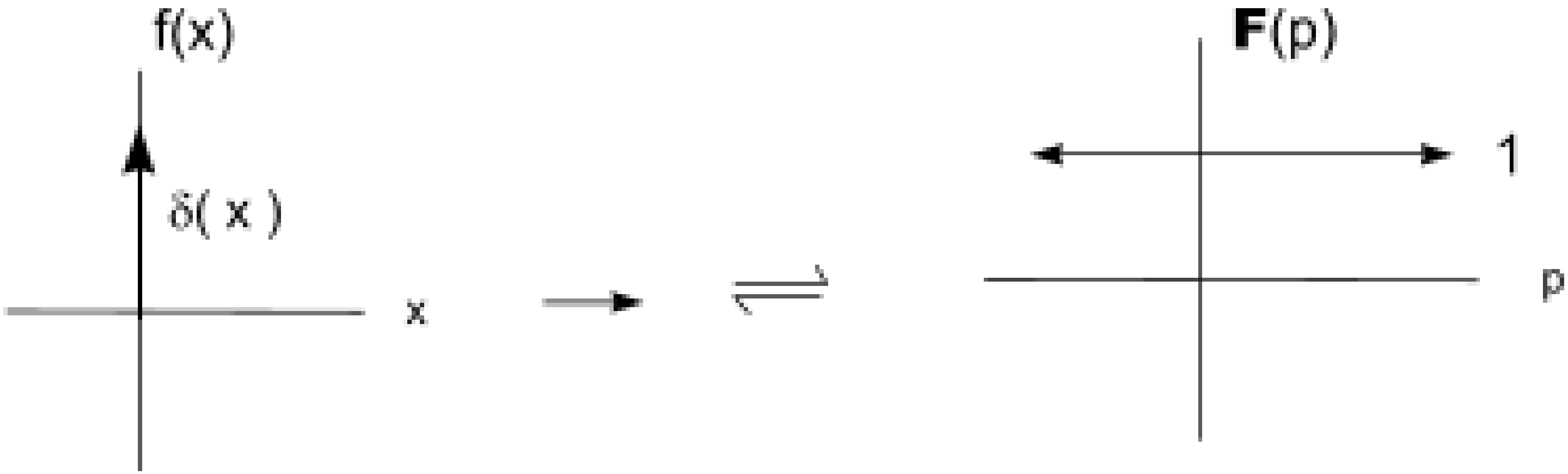
FT Properties

Linearity	$\alpha f(t) + \beta g(t)$	$\leftrightarrow$	$\alpha F(\omega) + \beta G(\omega)$
Time Translation	$f(t - t_0)$	$\leftrightarrow$	$e^{-j\omega t_0} F(\omega)$
Scale Change	$f(at)$	$\leftrightarrow$	$\frac{1}{\|a\|} F(\omega / a)$
Frequency Translation	$e^{j\omega_0 t} f(t)$	$\leftrightarrow$	$F(\omega - \omega_0)$
Time Convolution	$f(t) \star g(t)$	$\leftrightarrow$	$F(\omega) G(\omega)$
Frequency Convolution	$f(t) g(t)$	$\leftrightarrow$	$\frac{1}{2\pi} F(\omega) \star G(\omega)$

$$(f * g)(x) = \int_{\mathbf{R}^d} f(y) g(x - y)\, dy = \int_{\mathbf{R}^d} f(x - y) g(y)\, dy.$$

Common Transform Pairs

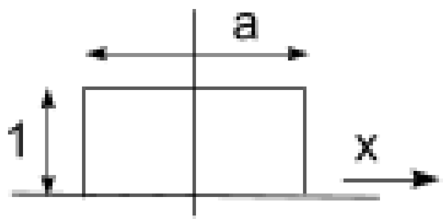
Dirac delta - constant



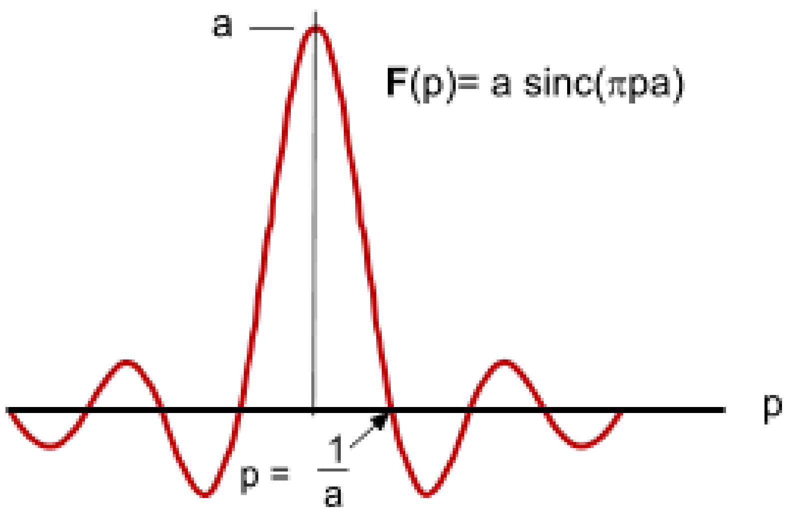
Common Transform Pairs

Rectangle – sinc

sinc(x) = sin(x) / x

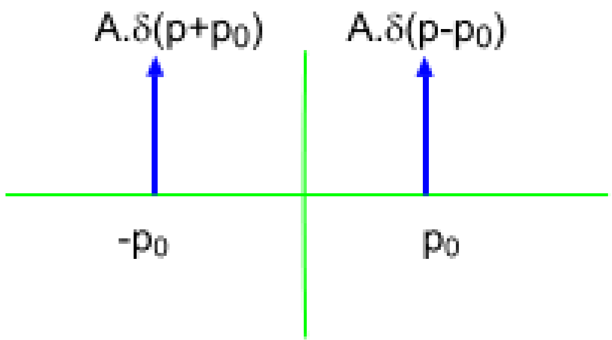


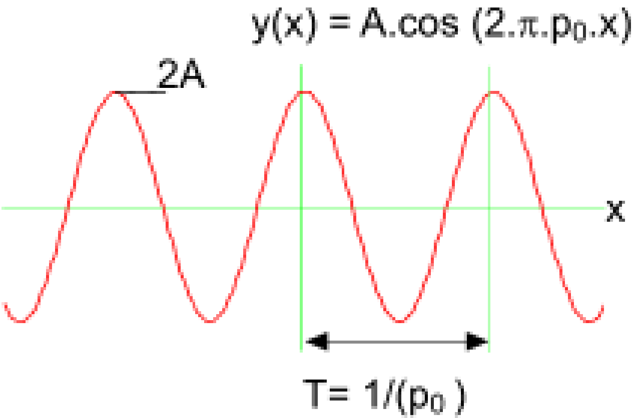
$$\Pi_a = 0, \; -\infty < x < -a/2$$
$$= 1, \; -a/2 < x < a/2$$
$$= 0, \; a/2 < x < \infty$$



Common Transform Pairs

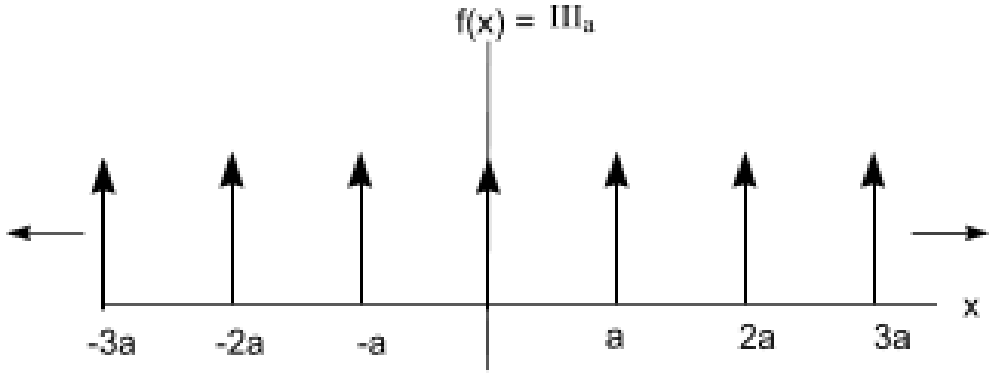
Two symmetric Diracs - cosine

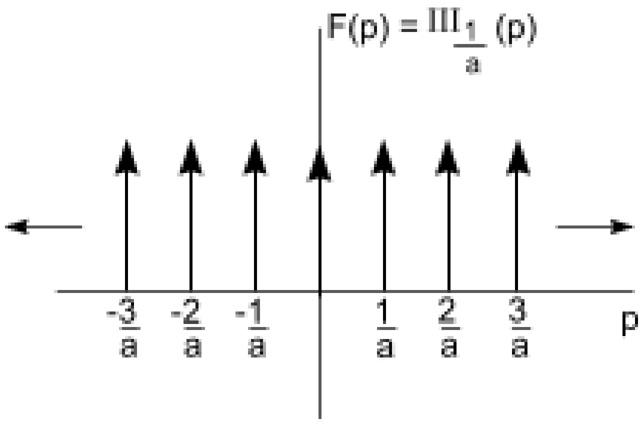




Common Transform Pairs

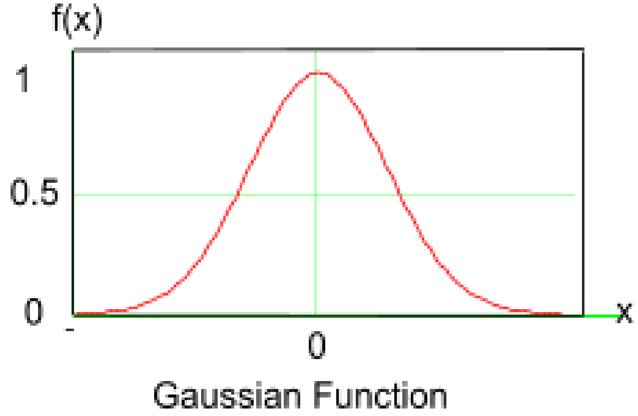
Comb – comb (inverse width)

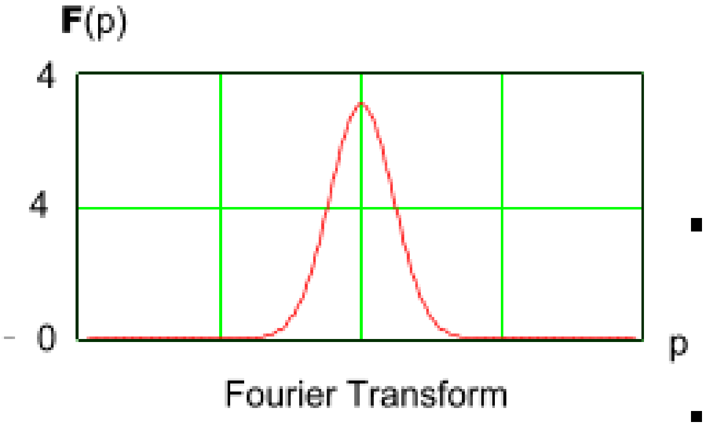




Common Transform Pairs

Gaussian – Gaussian (inverse variance)



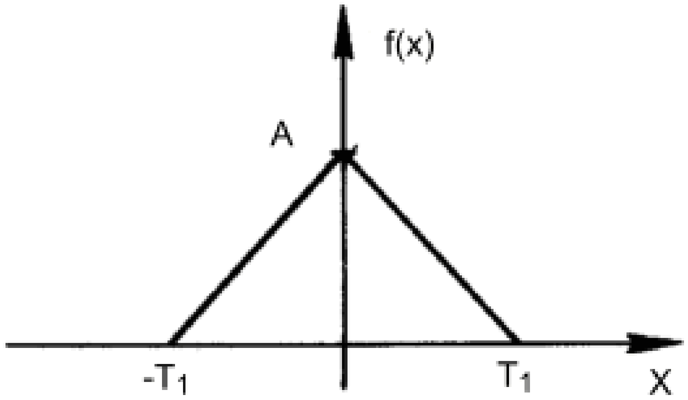


Common Transform Pairs Summary

Discrete unit impulse	$\delta(x, y) \Leftrightarrow 1$
Rectangle	$\text{rect}[a, b] \Leftrightarrow ab\frac{\sin(\pi ua)}{(\pi ua)}\frac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$
Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $j\frac{1}{2}\left[\delta(u + Mu_0, v + Nv_0) - \delta(u - Mu_0, v - Nv_0)\right]$
Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$ $\frac{1}{2}\left[\delta(u + Mu_0, v + Nv_0) + \delta(u - Mu_0, v - Nv_0)\right]$
Gaussian	$A2\pi\sigma^2 e^{-2\pi^2\sigma^2(t^2+z^2)} \Leftrightarrow Ae^{-(\mu^2+\nu^2)/2\sigma^2}$ (A is a constant)

Quiz

What is the FT of a triangle function?

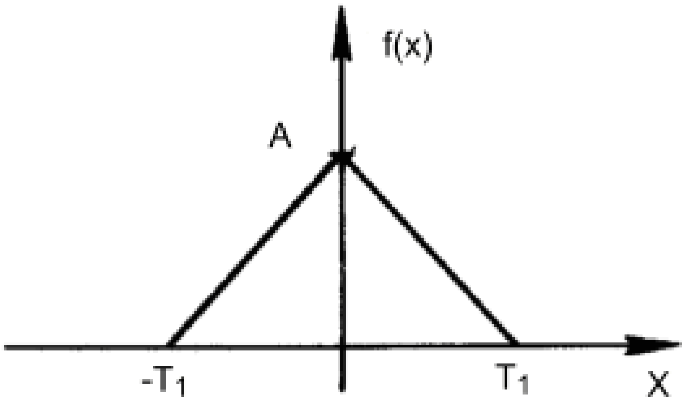


Hint: how do you get triangle function from the functions shown so far?

Answer

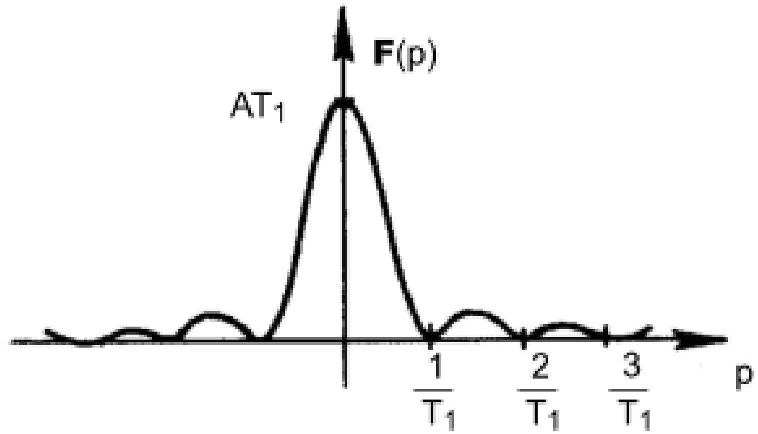
Triangle = box convolved with box

So its FT is sinc * sinc



$$f(x) = -\frac{A}{T_1}|x| + A$$

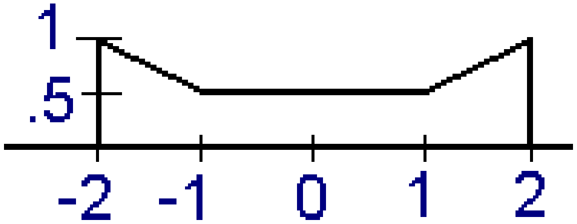
$f(x)=0 \quad |x| < T_1 \quad$ and $\quad |x| > T_1$



$$\mathbf{F}(p) = AT_1\left[\frac{\sin(\pi T_1 p)}{\pi T_1 p}\right]^2 = AT_1 \ \mathrm{sinc}^2(\pi T_1 p)$$

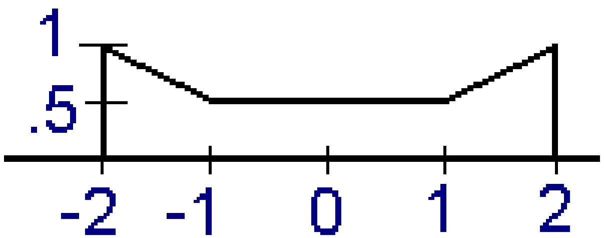
Quiz

- What is the FT?

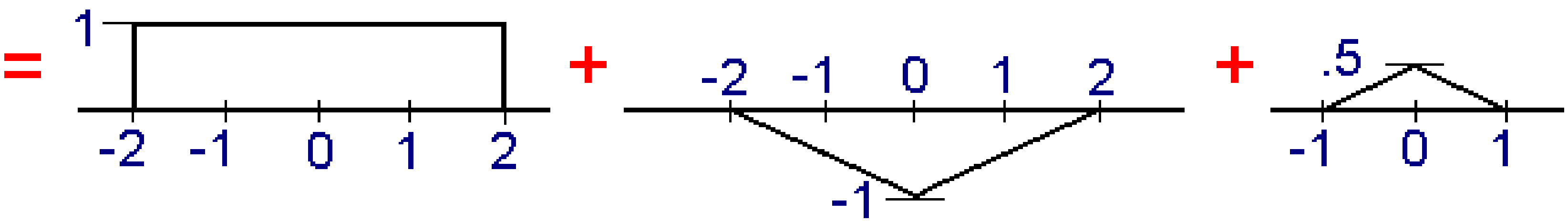


- Hint: use FT properties and express as functions with known transforms

Answer



$f(x) = \Pi(x/4) - \Lambda(x/2) + .5\Lambda(x)$



FT is linear, so

$F(w) = 4\mathrm{sinc}(4w) - 2\mathrm{sinc}^2(2w) + .5\mathrm{sinc}^2(w)$

Fourier Transform of Images

2D Fourier Transform

- Forward transform:

$$F(u,v) = \int \int_{-\infty}^{\infty} f(x,y) e^{-j2\pi(xu+yv)} dx\, dy$$

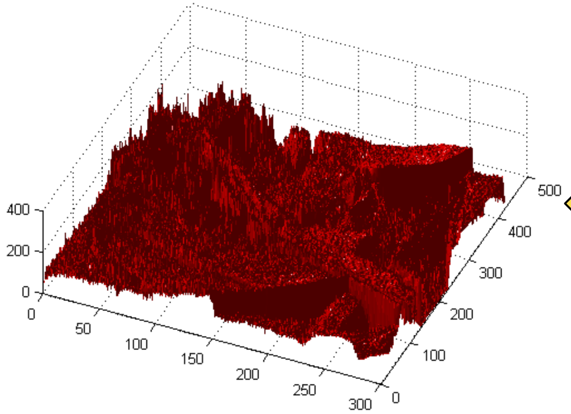
- Backward transform:

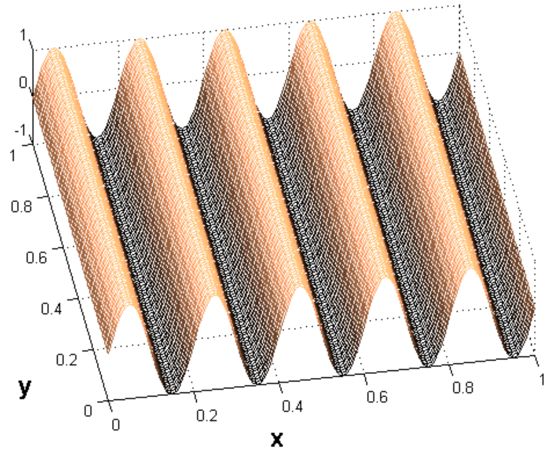
$$f(x,y) = \int \int_{-\infty}^{\infty} F(u,v) e^{j2\pi(xu+yv)} du\, dv$$

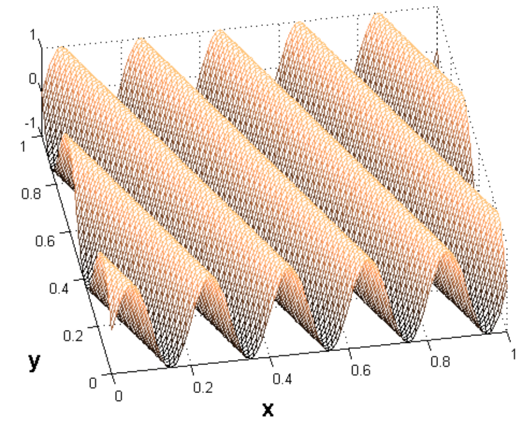
- Forward transform to freq. yields complex values (magnitude and phase):

$$F(u,v) = F_r(u,v) + jF_i(u,v) = |F(u,v)| e^{j\angle F(u,v)}$$

2D Fourier Transform



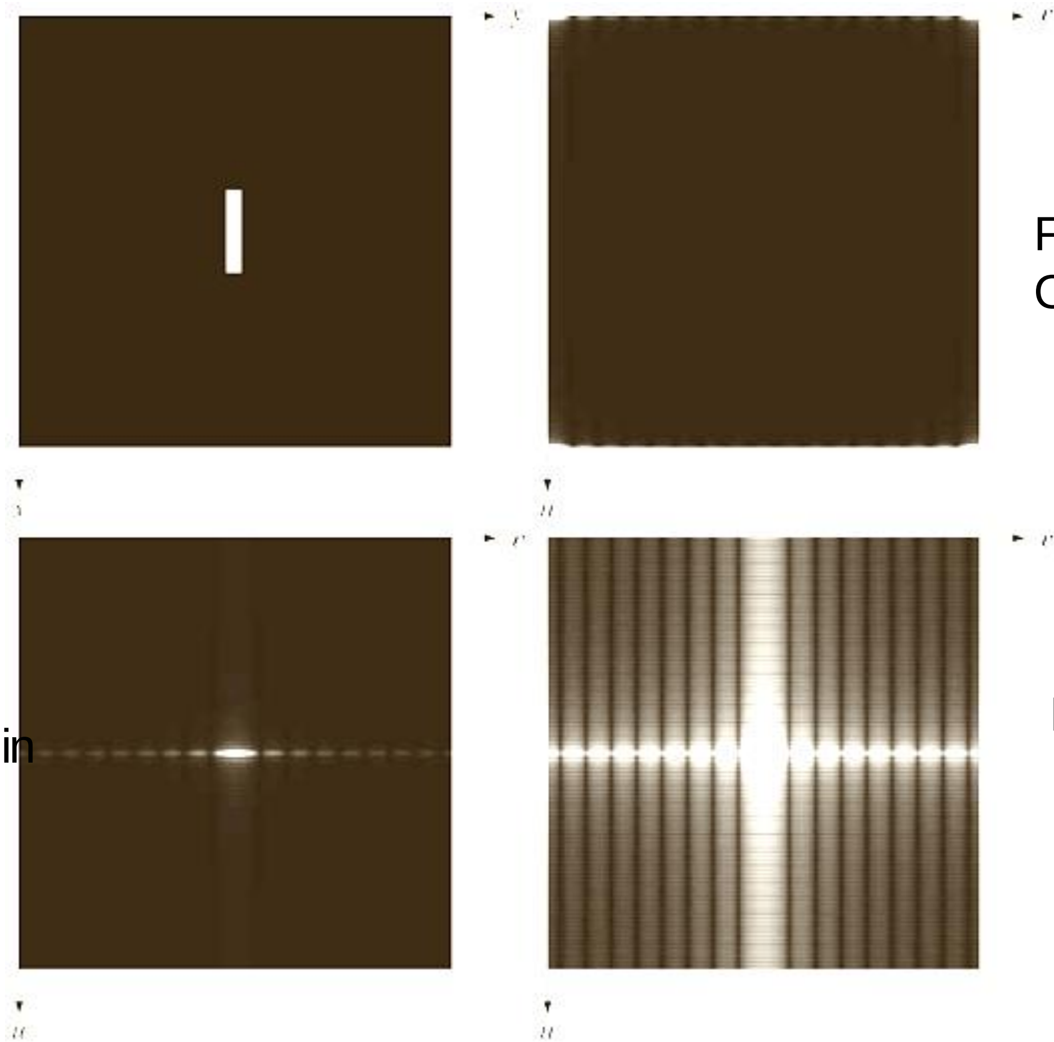




Fourier Spectrum

Image

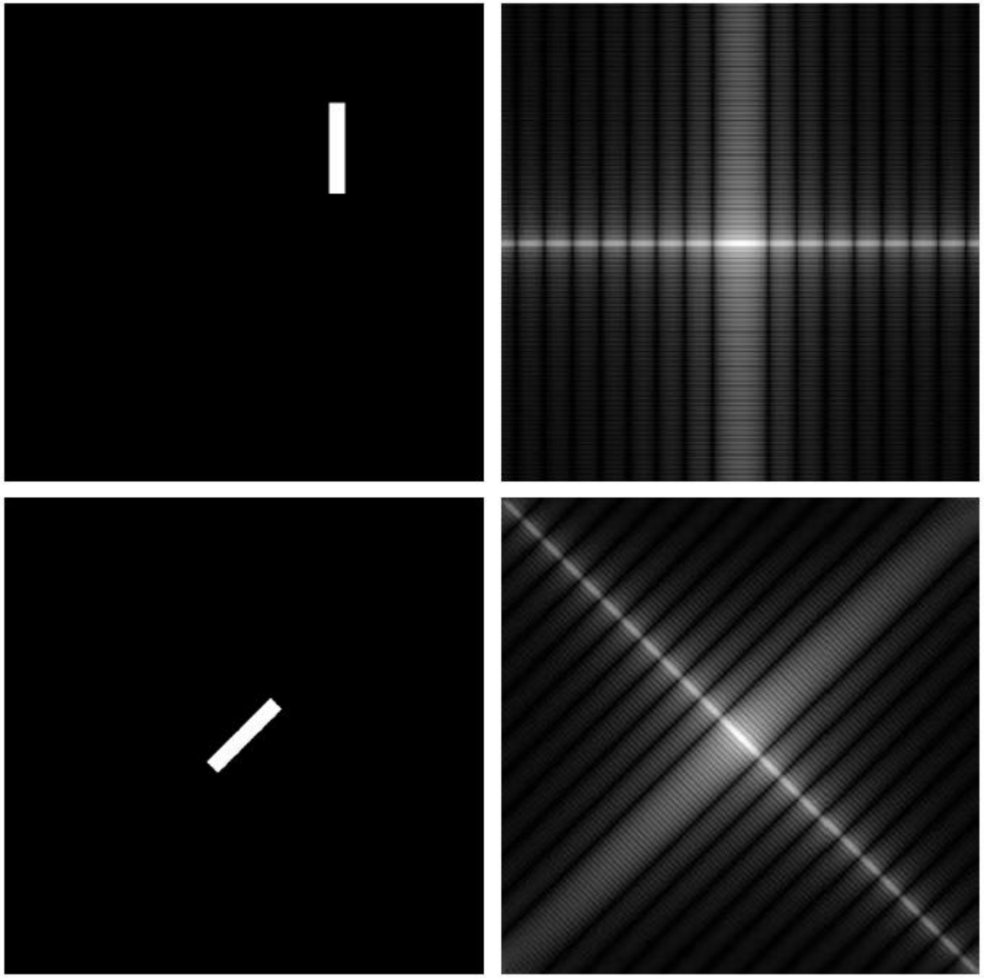
Fourier spectrum
Origin in corners



Retiled with origin
In center

Log of spectrum

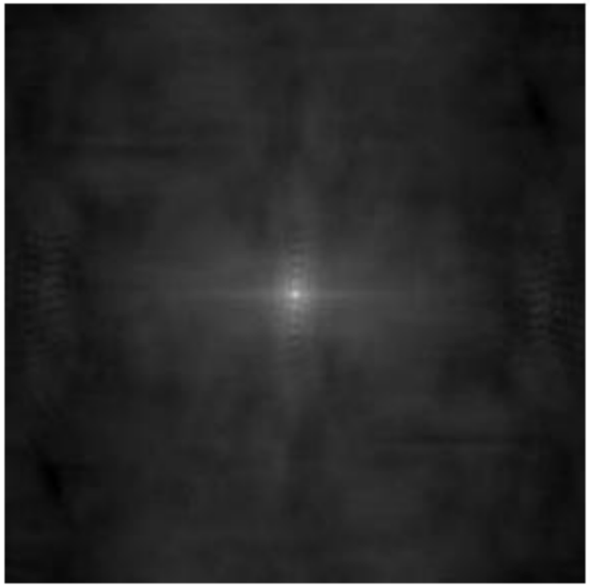
Fourier Spectrum – Translation and Rotation



Phase vs Spectrum

Image

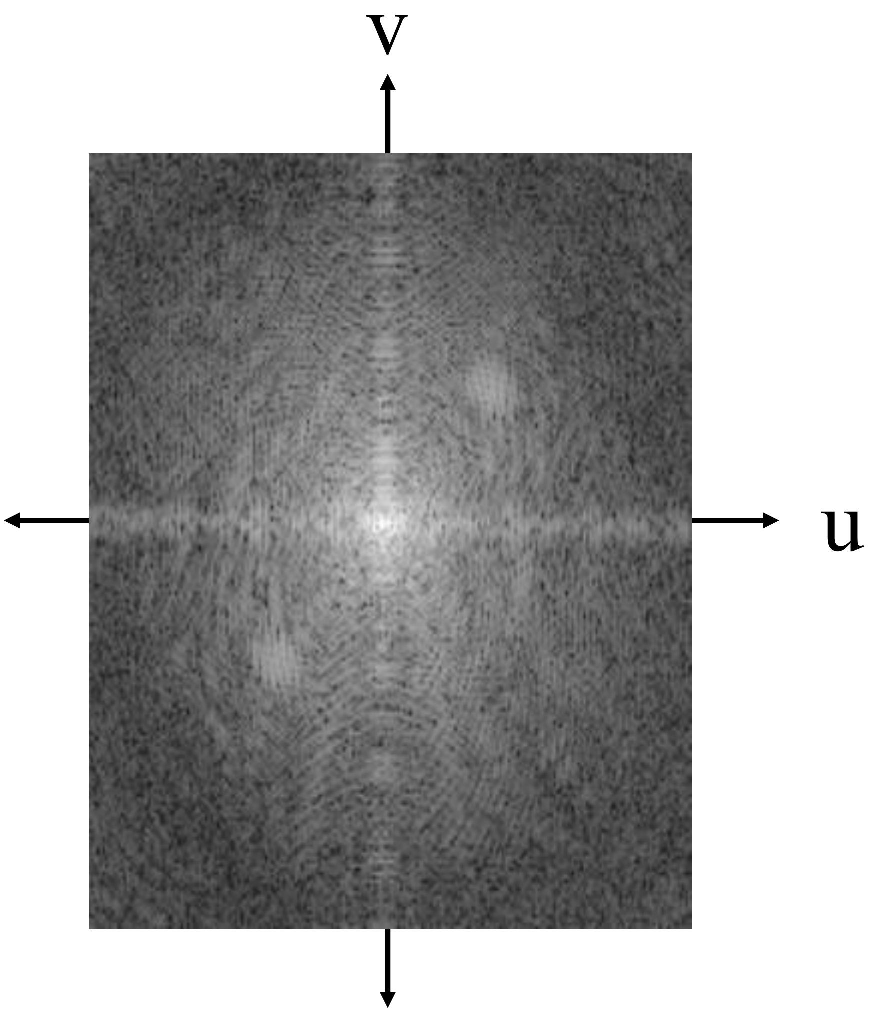
Reconstruction from
<u>phase map</u>

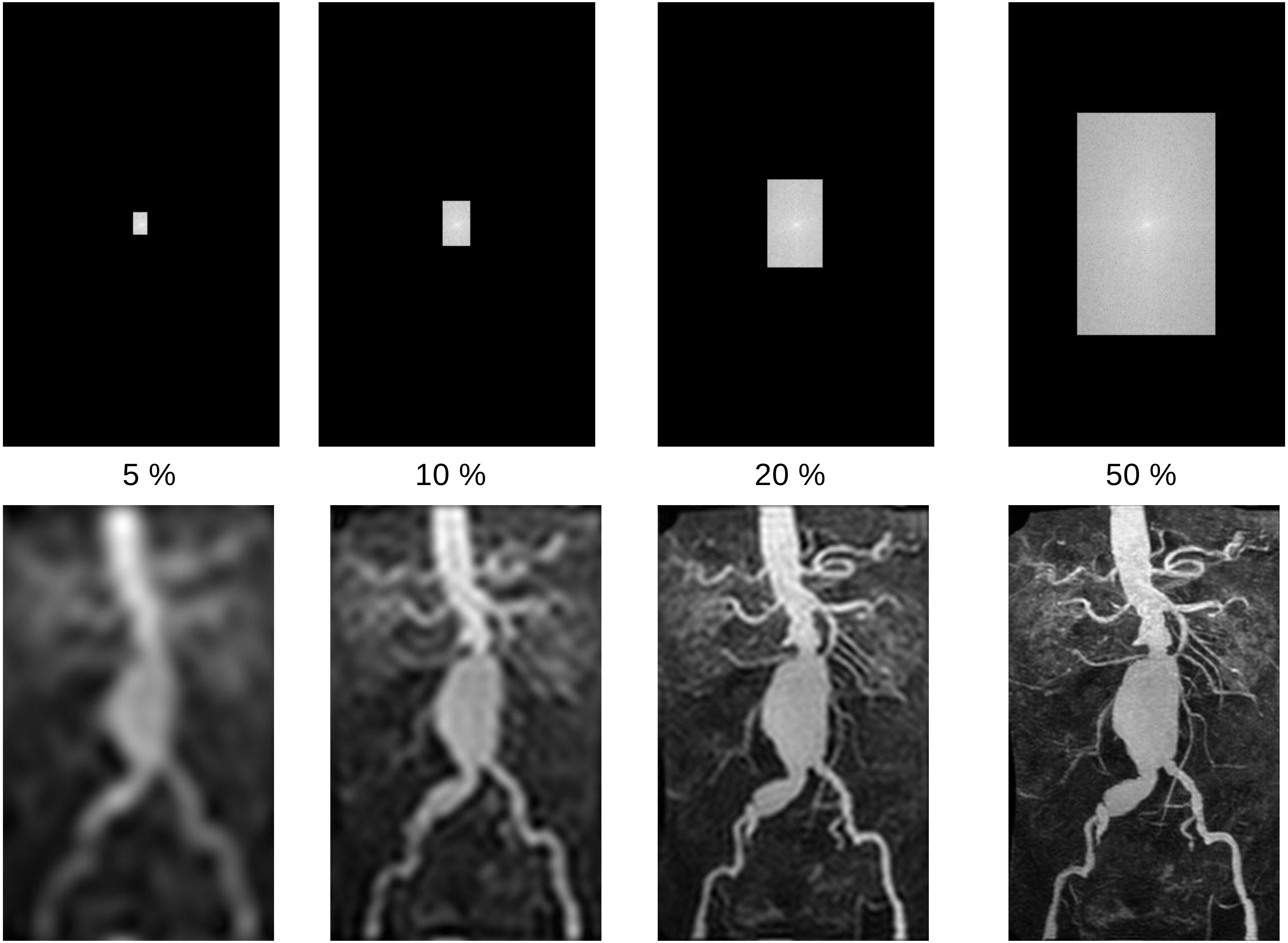


Reconstruction from
<u>spectrum</u>

Image

Fourier Space

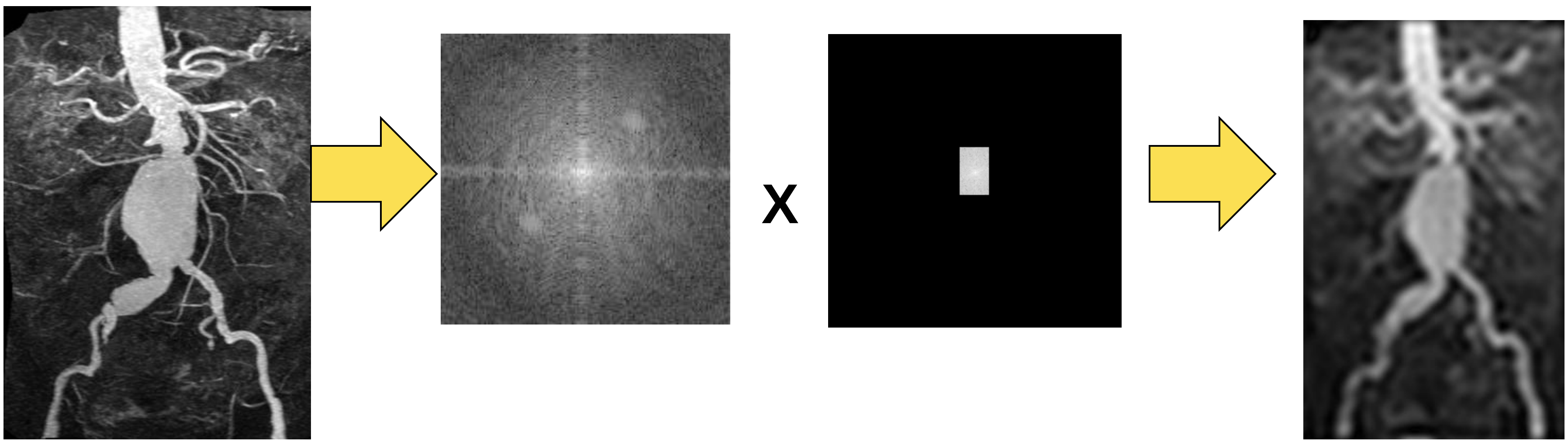



5 %
10 %
20 %
50 %

Fourier Spectrum Demo

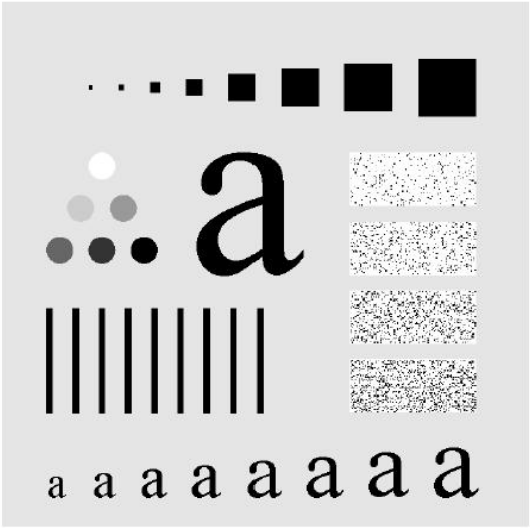
http://bigwww.epfl.ch/demo/basisfft/demo.html

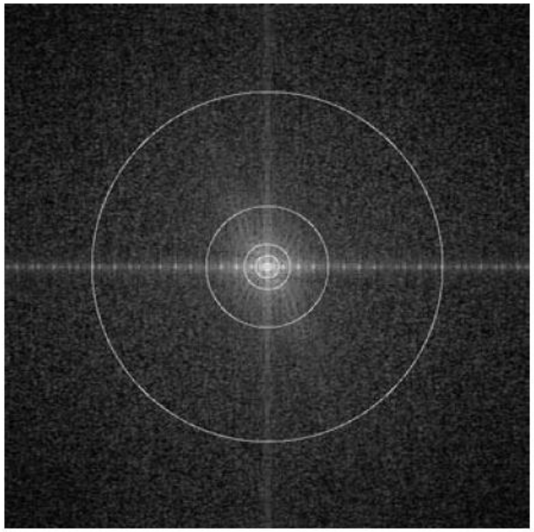
Filtering Using FT and Inverse

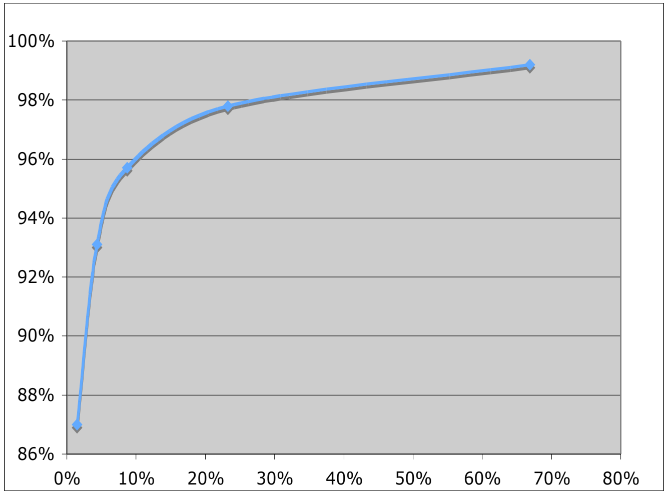


Low-Pass Filter

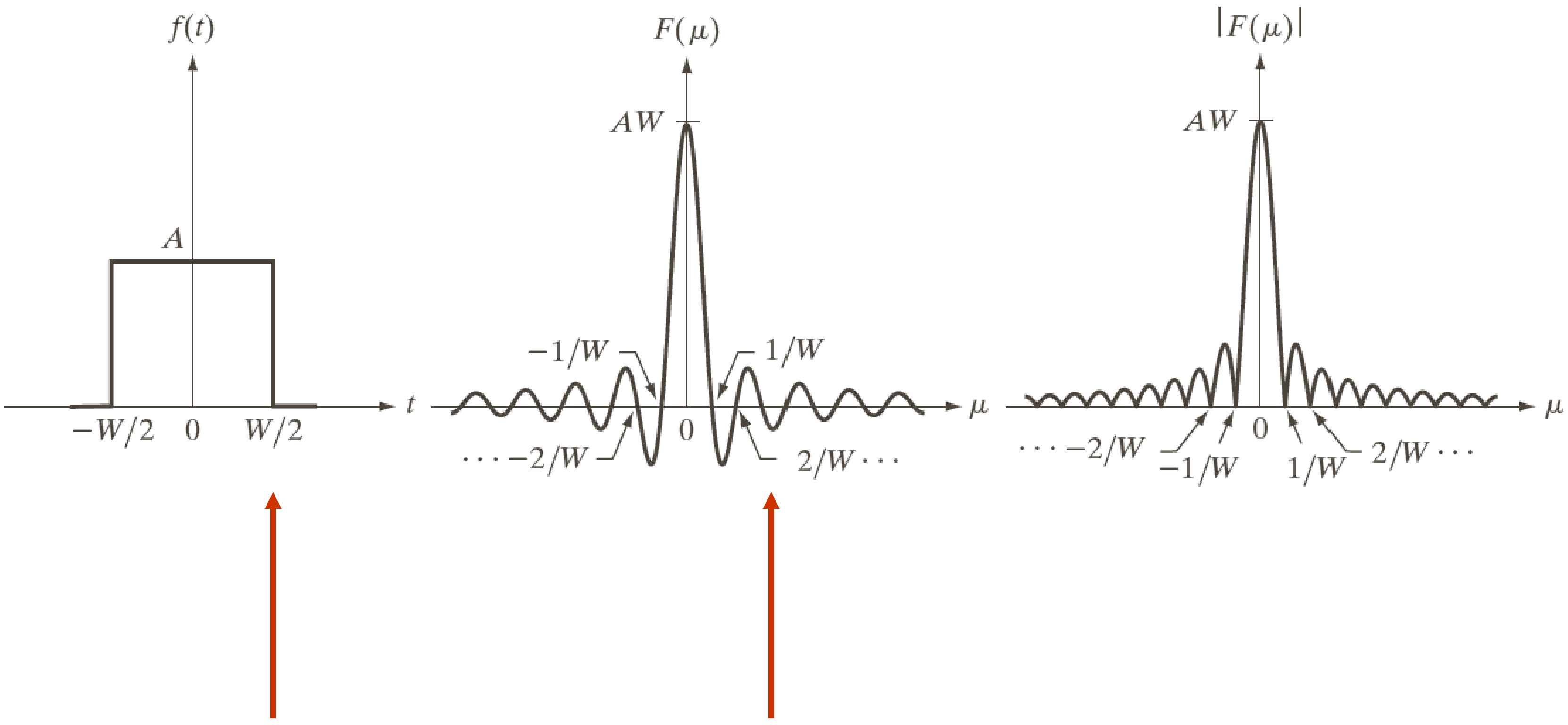
- Reduce/eliminate high frequencies
- Applications
  - – Noise reduction
    - uncorrelated noise is broad band
    - Images have sprectrum that focus on low







Ideal LP Filter – Box, Rect

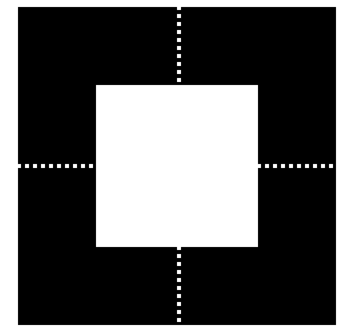


Cutoff freq

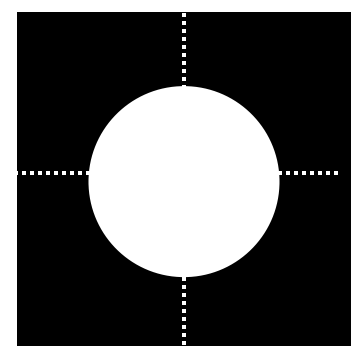
Ringing – Gibbs phenomenon

Extending Filters to 2D (or higher)

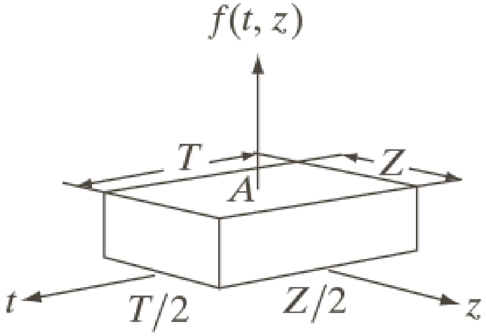
- Two options
  - Separable
    - H(s) -> H(u)H(v)
    - Easy, analysis

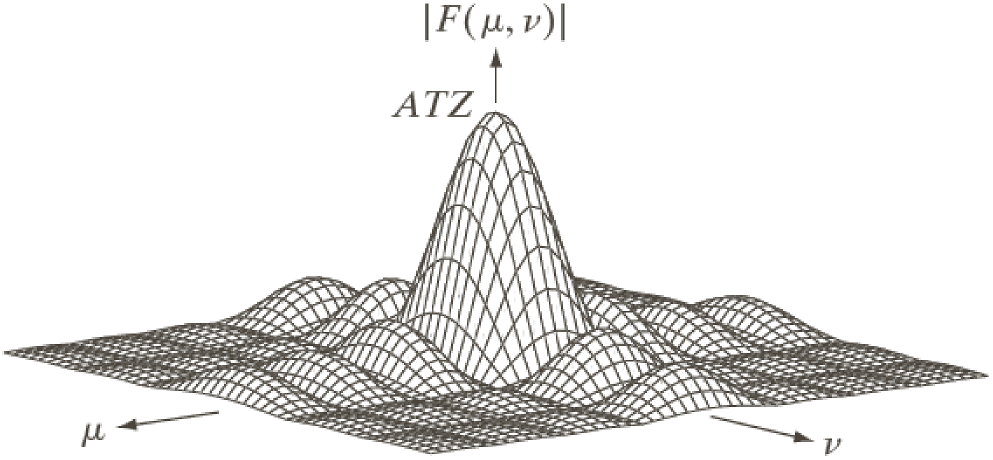


  - Rotate
    - H(s) -> H((u^2 + v^2)$^{1/2}$)
    - Rotationally invariant

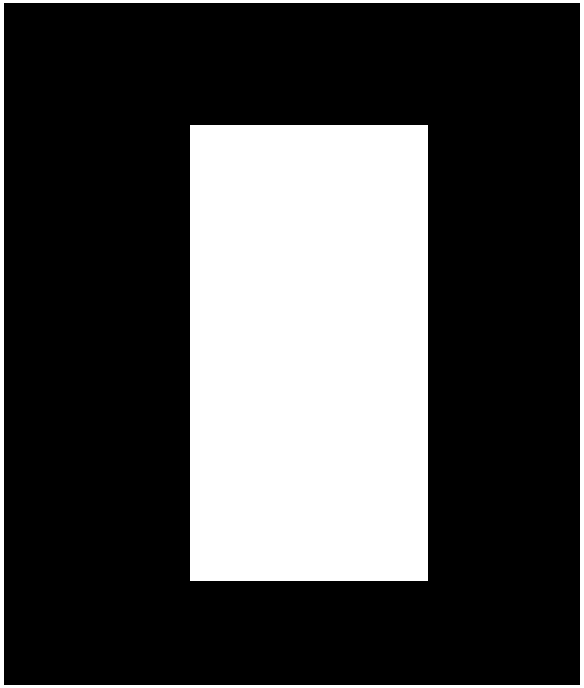


Ideal LP Filter – Box, Rect

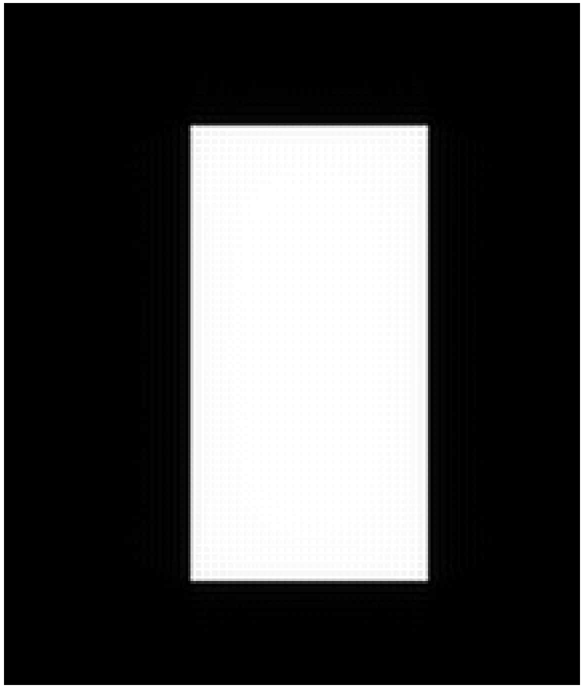




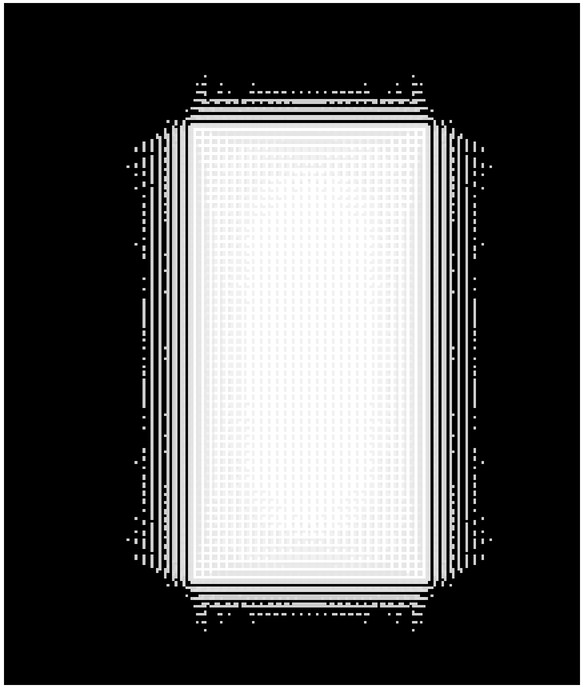
Ideal Low-Pass
Rectangle With Cutoff of 2/3



Image



Filtered



Filtered
+
Histogram Equalized

Ideal LP – 1/3

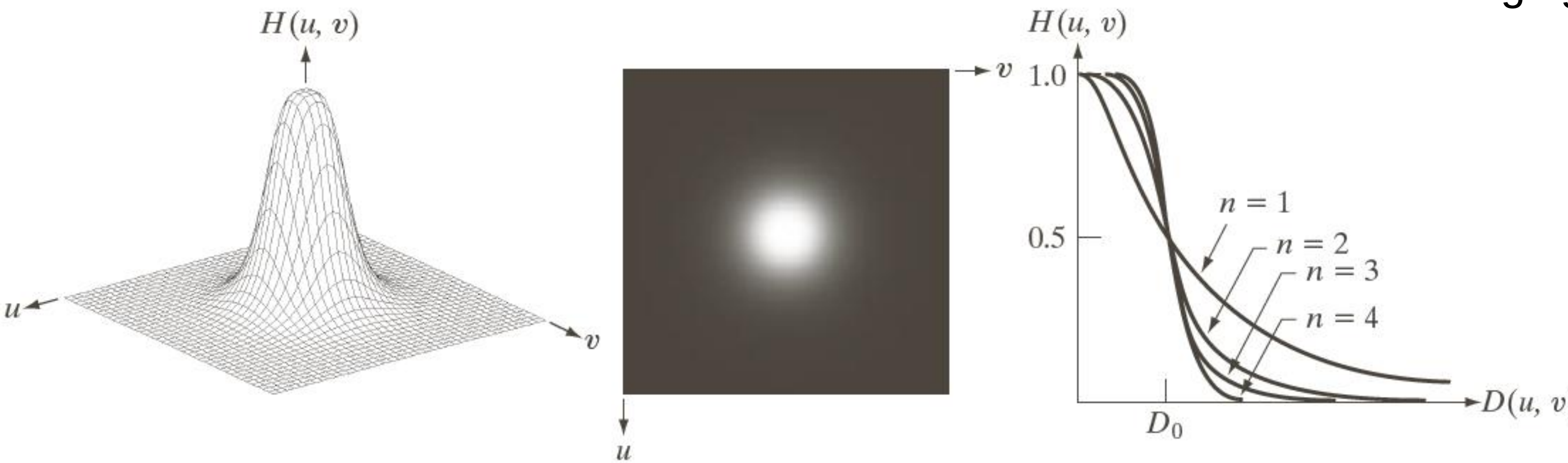
Ideal LP – 2/3

Butterworth Filter

Lowpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$	$H(u,v) = \dfrac{1}{1 + [D(u,v)/D_0]^{2n}}$	$H(u,v) = e^{-D^2(u,v)/2D_0^2}$

Control of cutoff and slope
Can control ringing

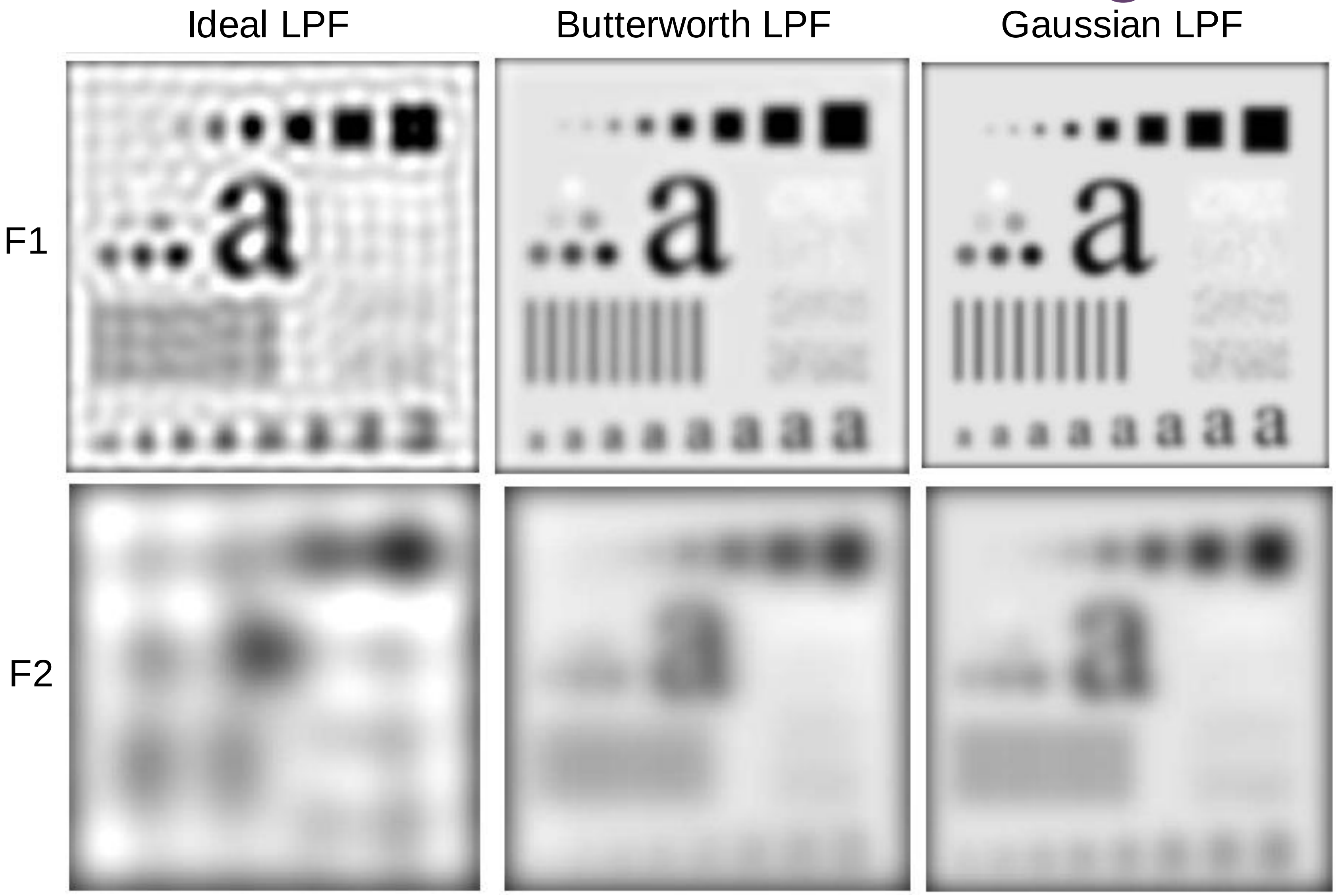


Butterworth - 1/3

Butterworth vs Ideal LP

Butterworth – 2/3

Gaussian LP Filtering



High Pass Filtering

- HP = 1 - LP
  - All the same filters as HP apply
- Applications
  - Visualization of high-freq data (accentuate)
- High boost filtering
  - HB = (1- a) + a(1 - LP) = 1 - a*LP

High-Pass Filters

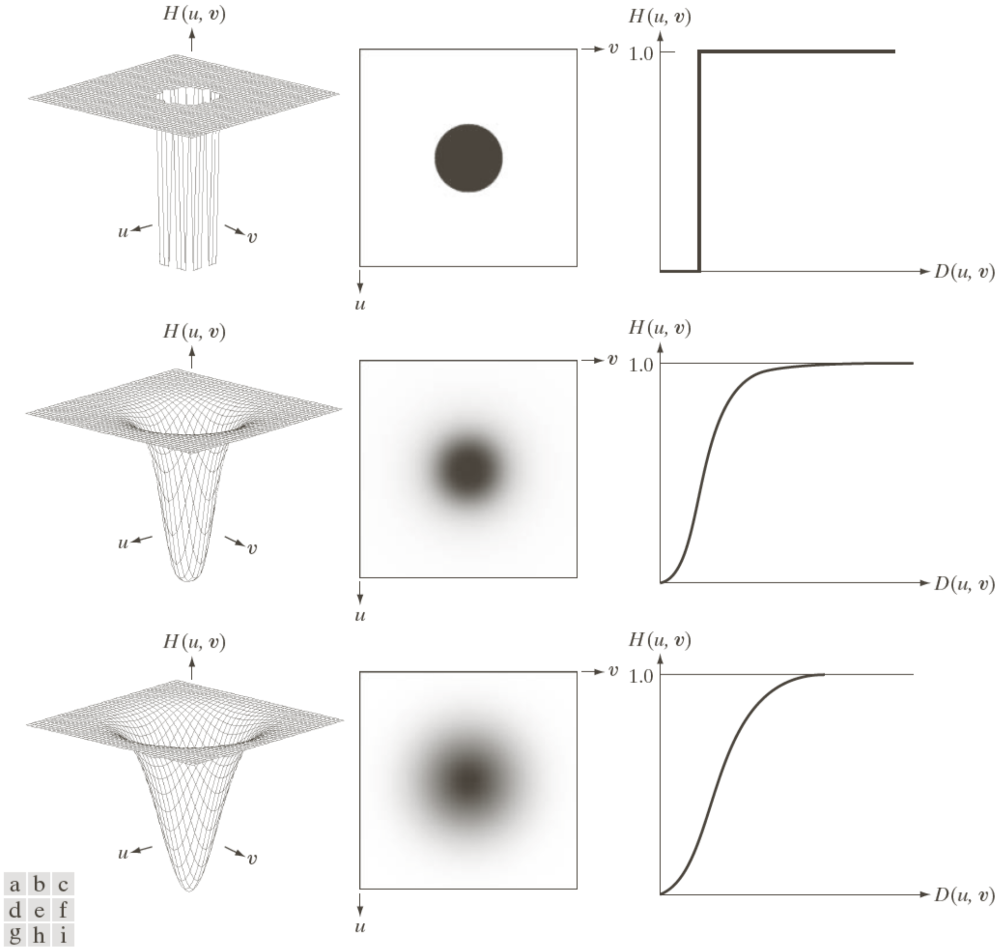


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

High-Pass Filters in Spatial Domain

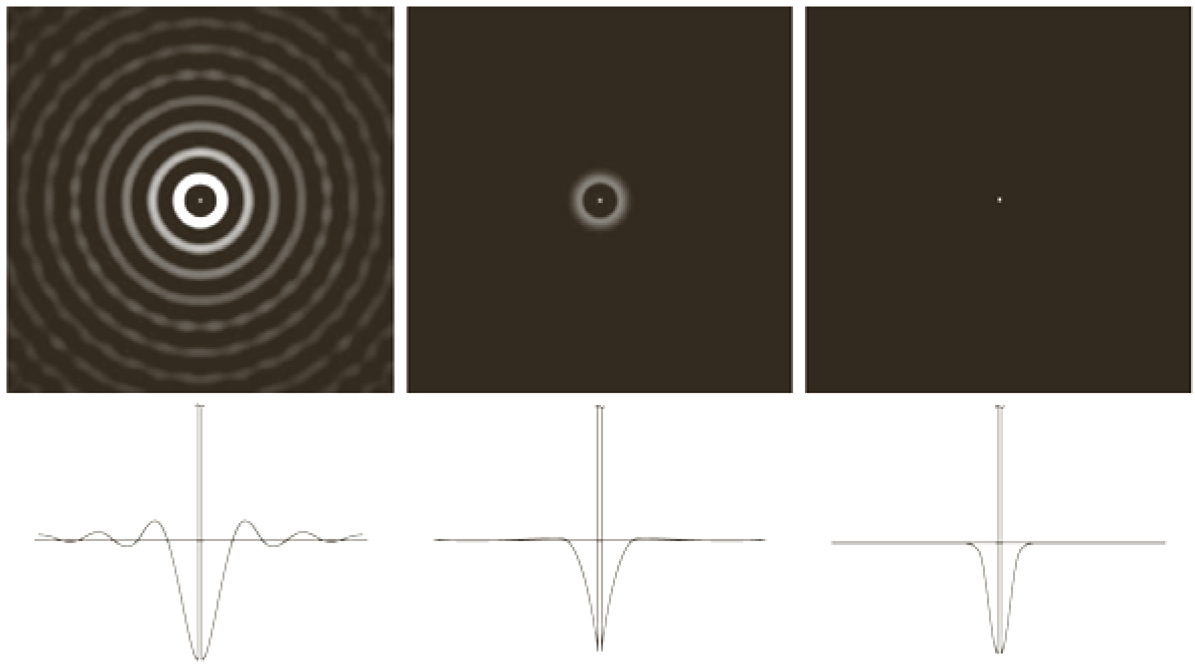


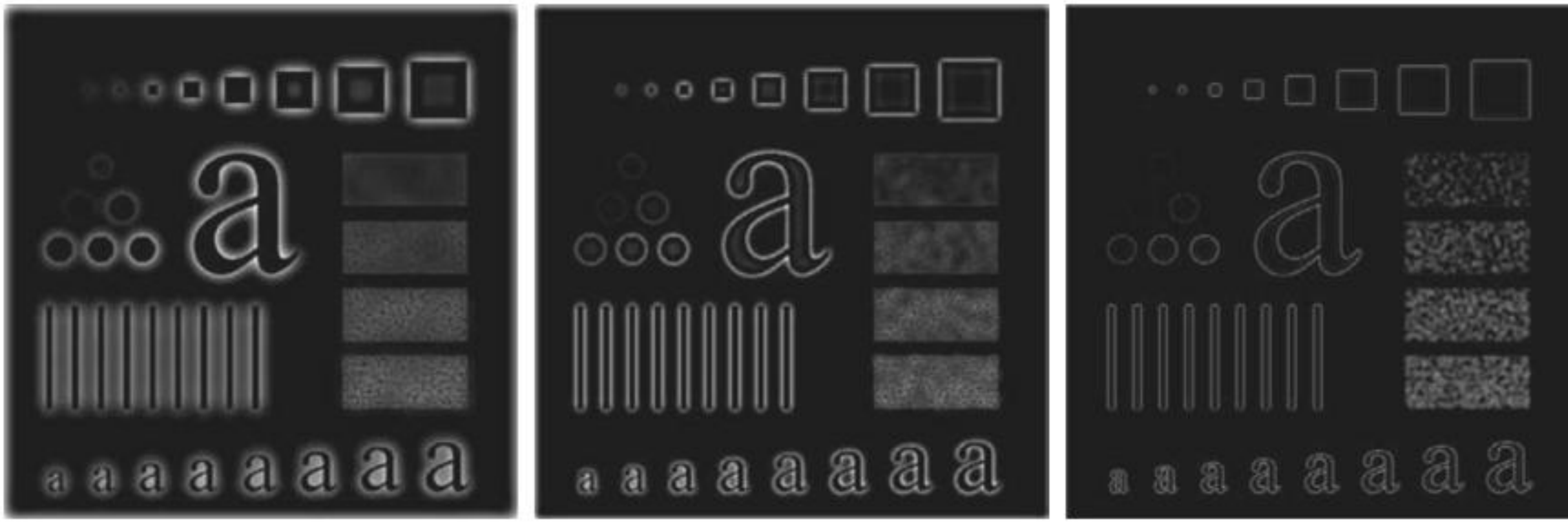
FIGURE 4.53 Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

High-Pass Filtering with IHPF

a b c

FIGURE 4.54 Results of highpass filtering the image in Fig. 4.41(a) using an IHPF with $D_0 = 30$, 60, and 160.

BHPF



a b c

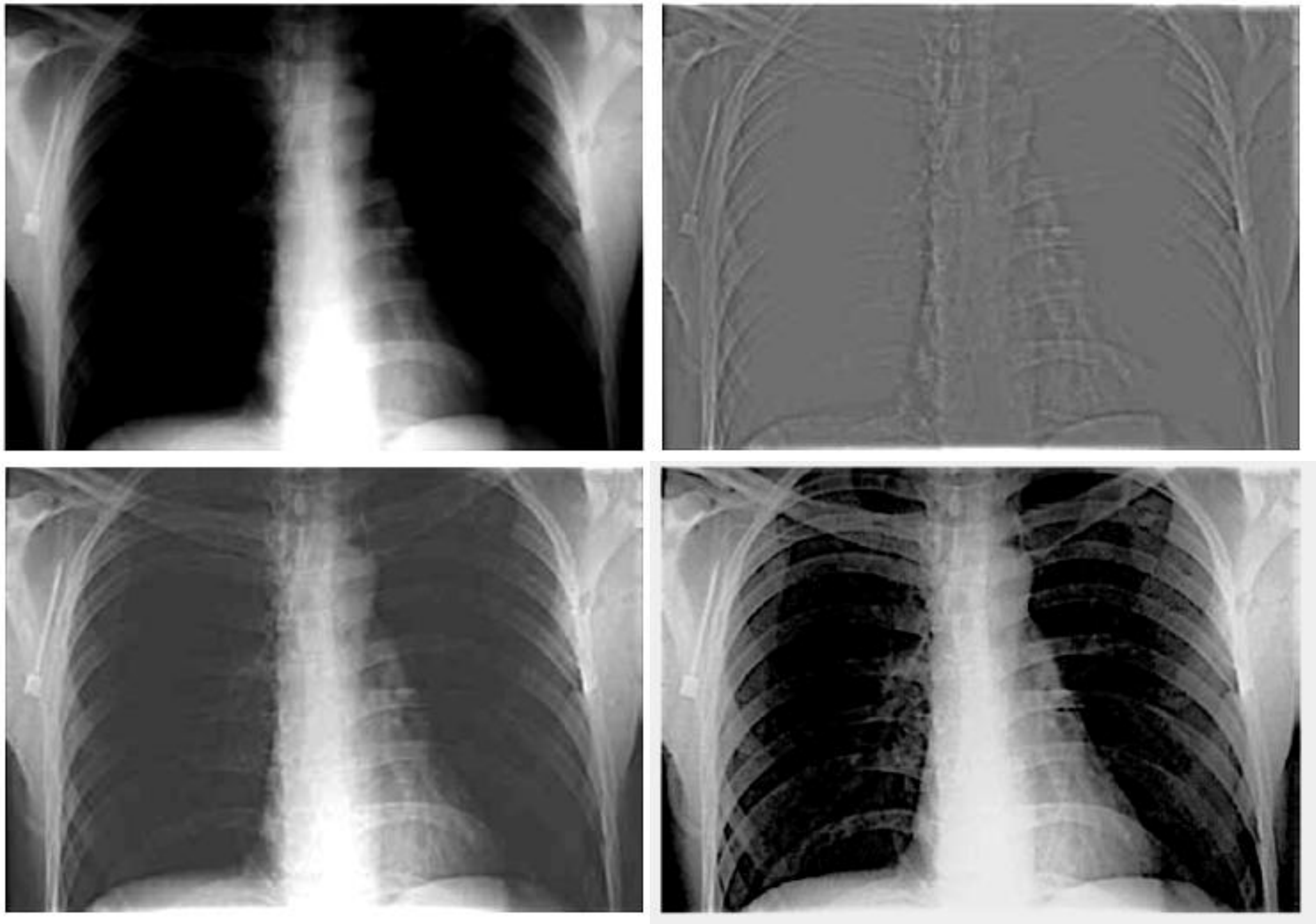
FIGURE 4.55 Results of highpass filtering the image in Fig. 4.41(a) using a BHPF of order 2 with $D_0 = 30, 60,$ and 160, corresponding to the circles in Fig. 4.41(b). These results are much smoother than those obtained with an IHPF.

GHPF

a b c

FIGURE 4.56 Results of highpass filtering the image in Fig. 4.41(a) using a GHPF with $D_0 = 30, 60,$ and 160, corresponding to the circles in Fig. 4.41(b). Compare with Figs. 4.54 and 4.55.

HP, HB, HE



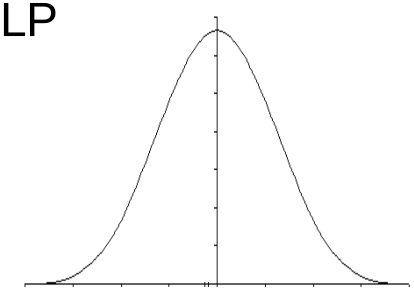
High Boost with GLPF

High-Boost Filtering

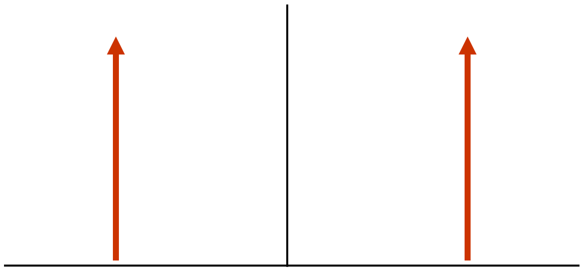
Band-Pass Filters

- Shift LP filter in Fourier domain by convolution with delta

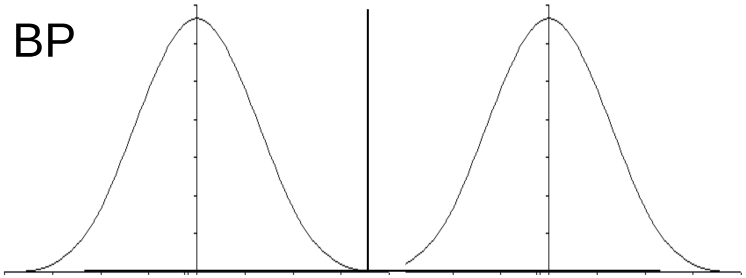
LP



$$\delta(s - s_0) + \delta(s + s_0)$$



BP

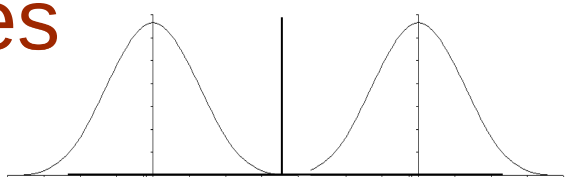


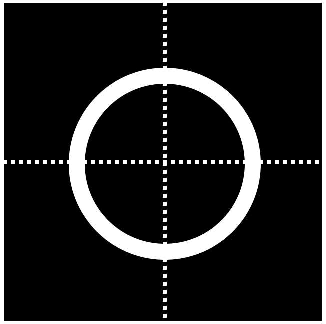
Typically 2-3 parameters

- -Width
- -Slope
- -Band value

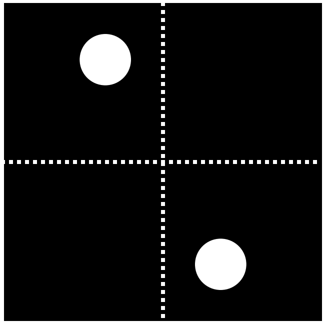
Band Pass - Two Dimensions

- Two strategies



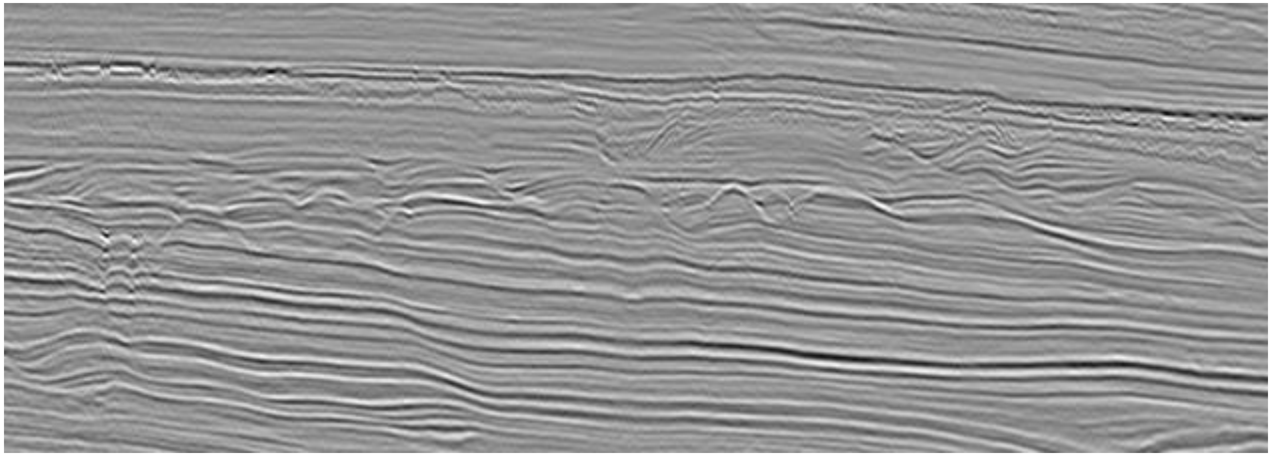


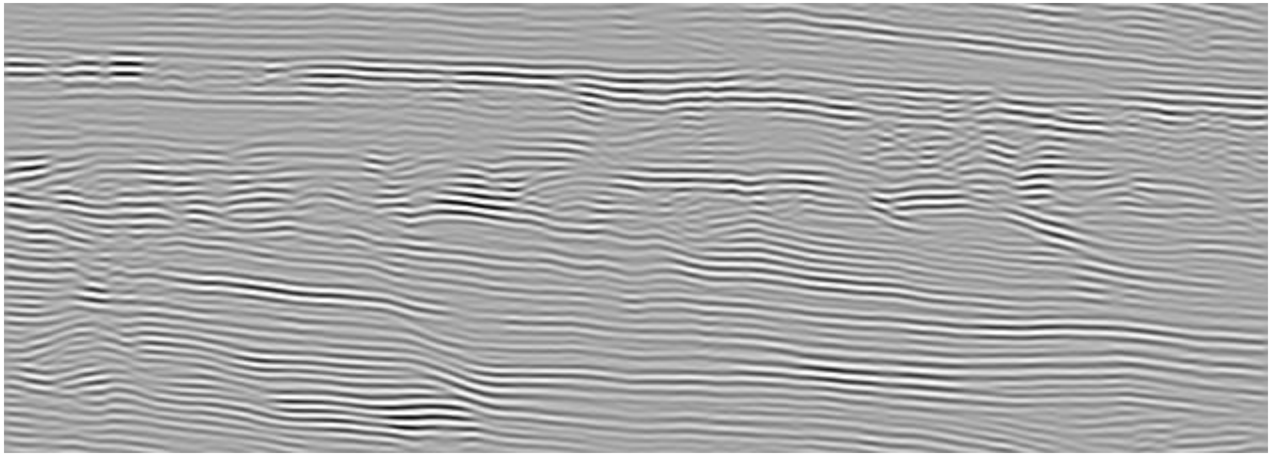
  - Rotate
    - Radially symmetric
  - Translate in 2D
    - Oriented filters



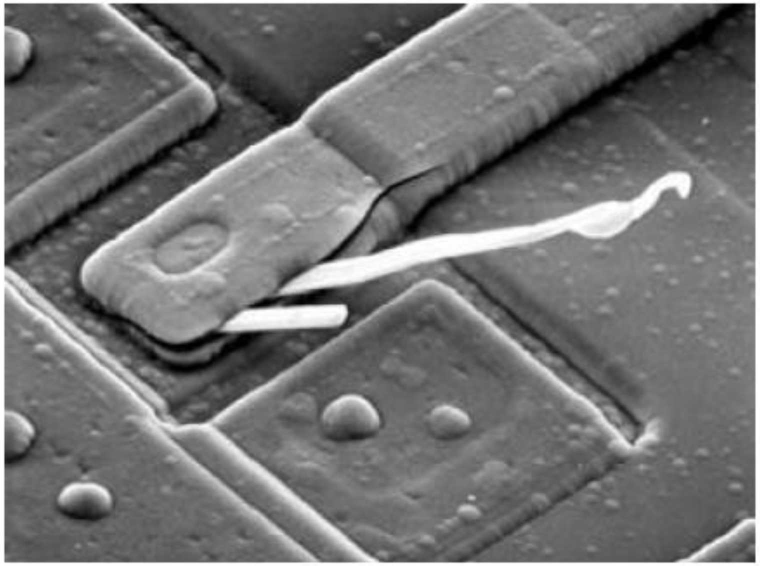
- Note:
  - Convolution with delta-pair in FD is multiplication with cosine in spatial domain

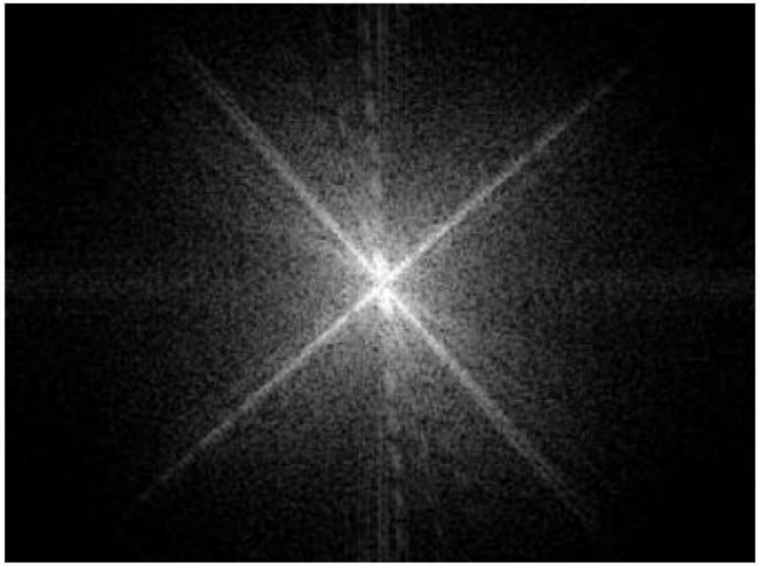
Band Bass Filtering



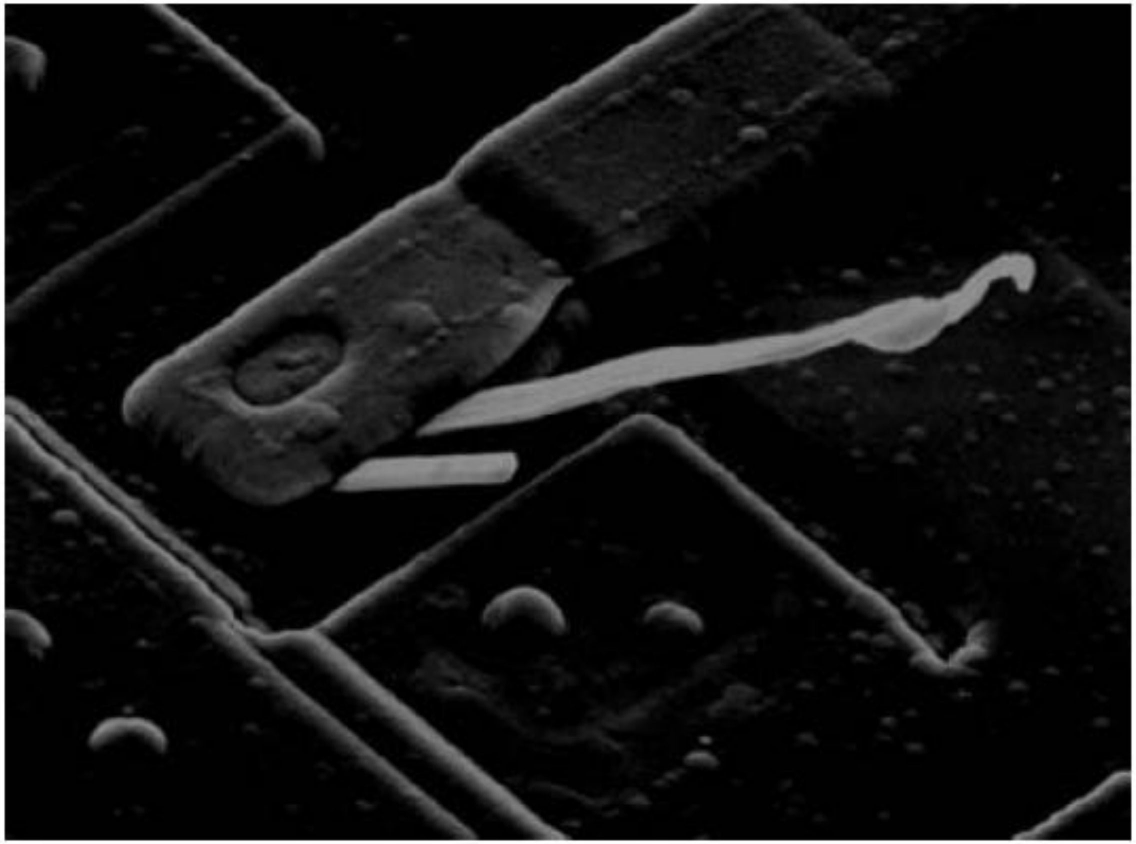


SEM Image and Spectrum



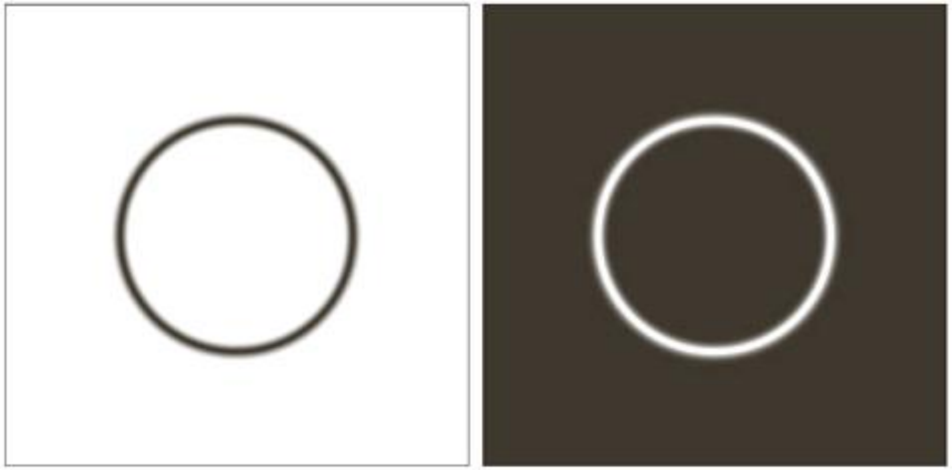


Band-Pass Filter

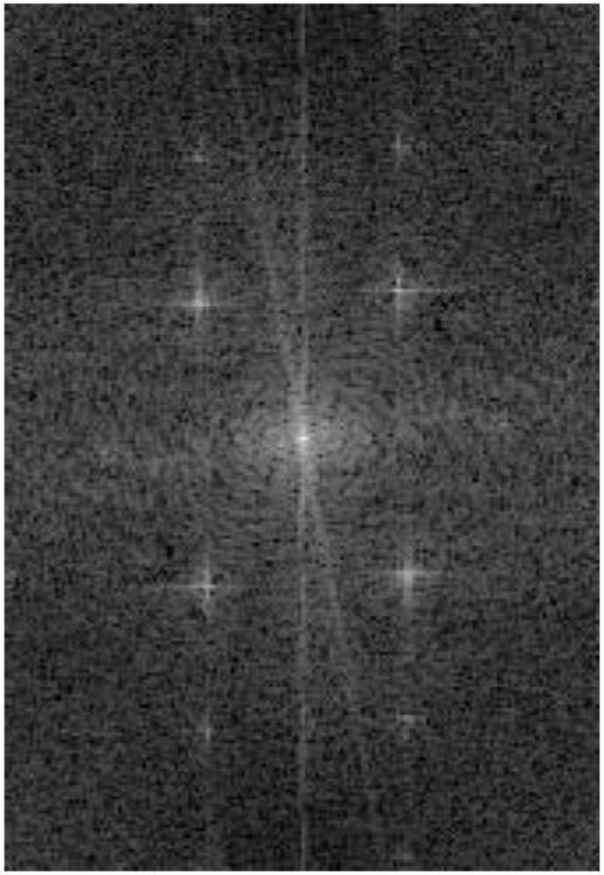


Radial Band Pass/Reject

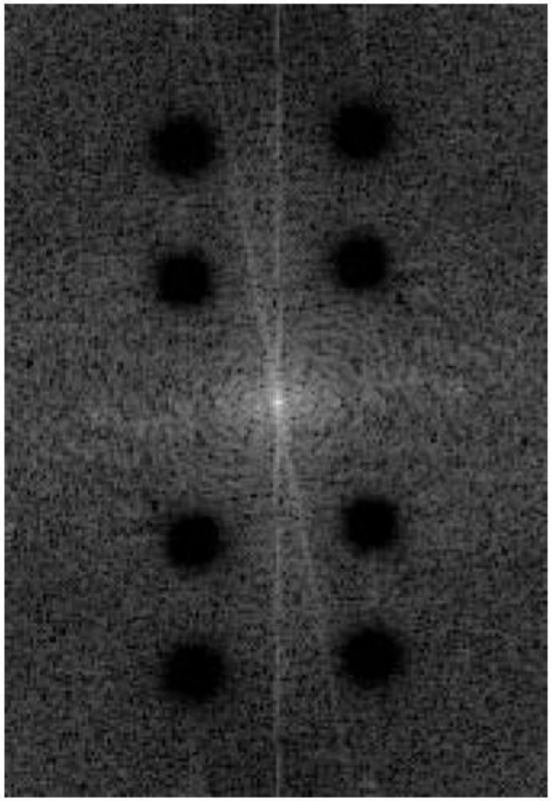
Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \le D \le D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u,v) = \frac{1}{1 + \left[\frac{DW}{D^2 - D_0^2}\right]^{2n}}$	$H(u,v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$



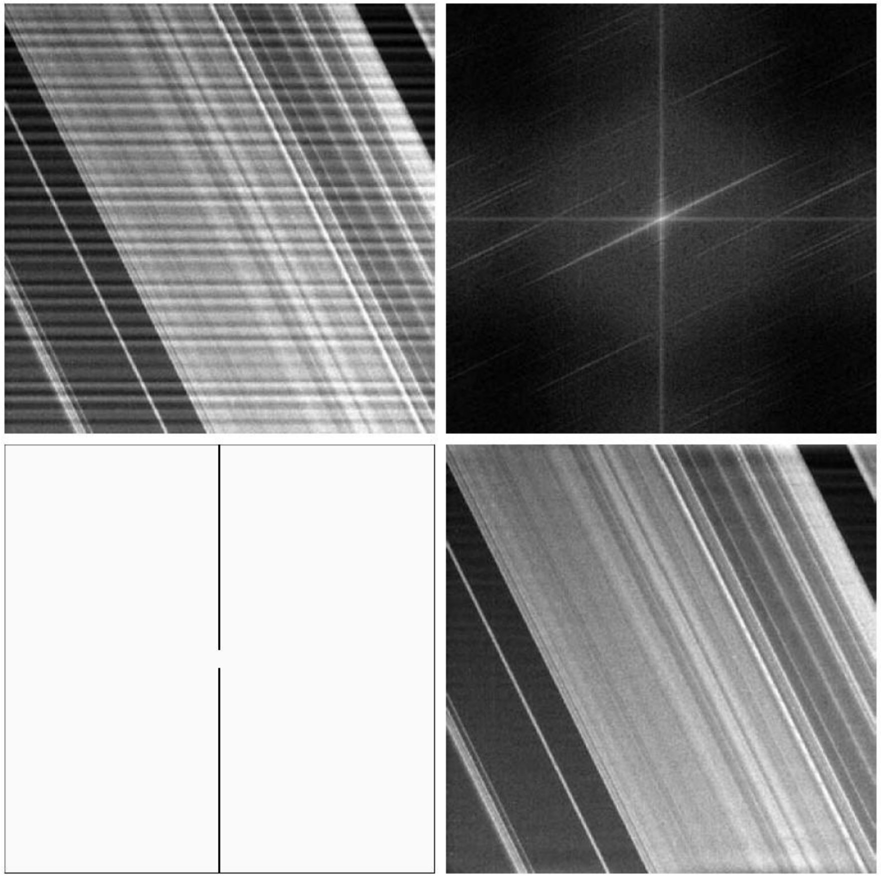
Band Reject Filtering



Band Reject Filtering



Band Reject Filtering

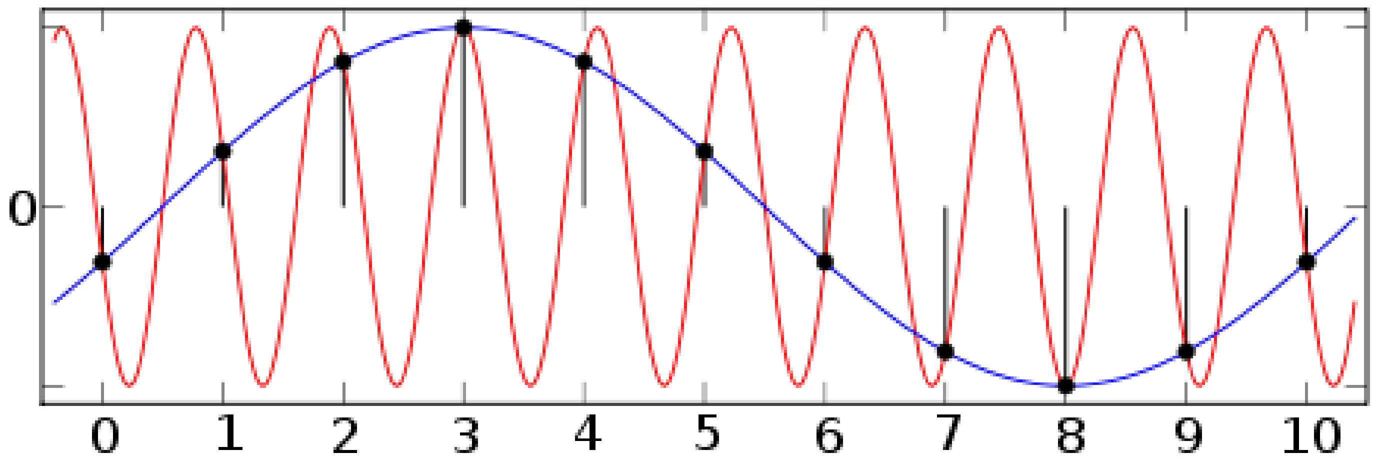


Aliasing

Discrete Sampling and Aliasing

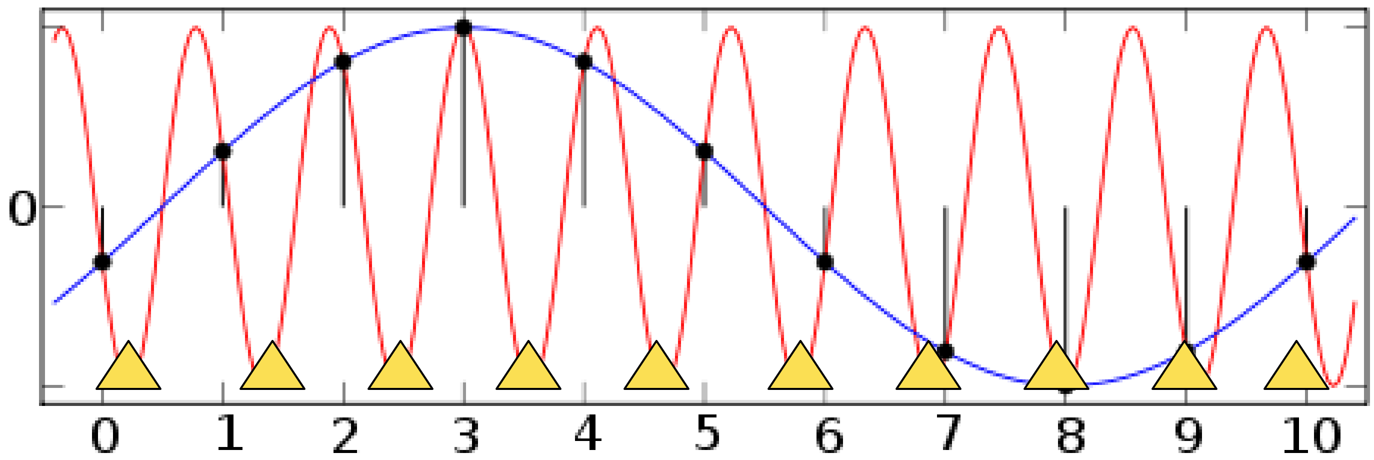
- Digital signals and images are discrete representations of the real world
  - Which is continuous
- What happens to signals/images when we sample them?
  - Can we quantify the effects?
  - Can we understand the artifacts and can we limit them?
  - Can we reconstruct the original image from the discrete data?

Sampling and Aliasing



- Given the sampling rate, CAN NOT distinguish the two functions
- High freq can appear as low freq

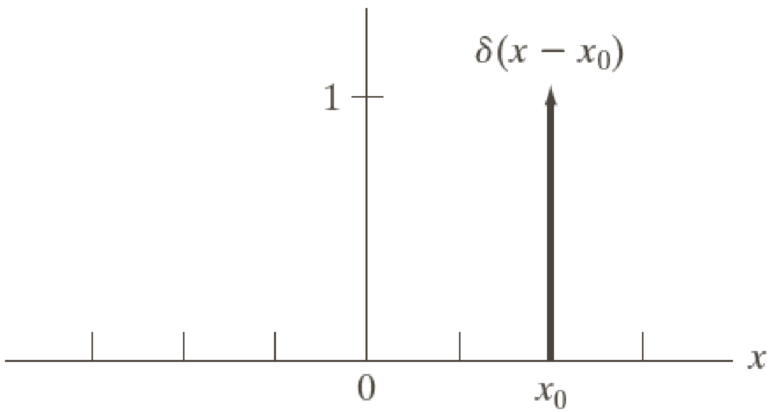
Ideal Solution: More Samples



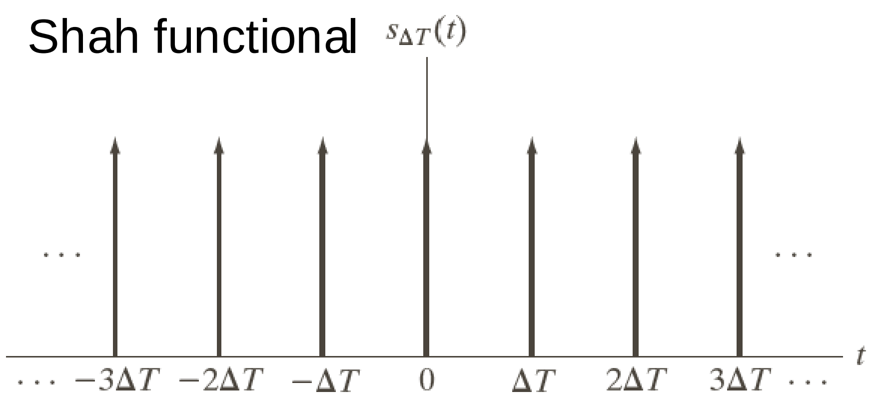
- Faster sampling rate allows us to distinguish the two signals
- Not always practical: hardware cost, longer scan time

A Mathematical Model of Discrete Samples

Delta functional



Shah functional



$$s_{\Delta T}(t) = \sum_{k=-\infty}^{\infty} \delta(t - k\Delta T)$$

A Mathematical Model of Discrete Samples

- Goal
  - To be able to do a continuous Fourier transform on a signal before and after sampling

Discrete signal

$$f_k \quad k = 0, \pm 1, \ldots$$

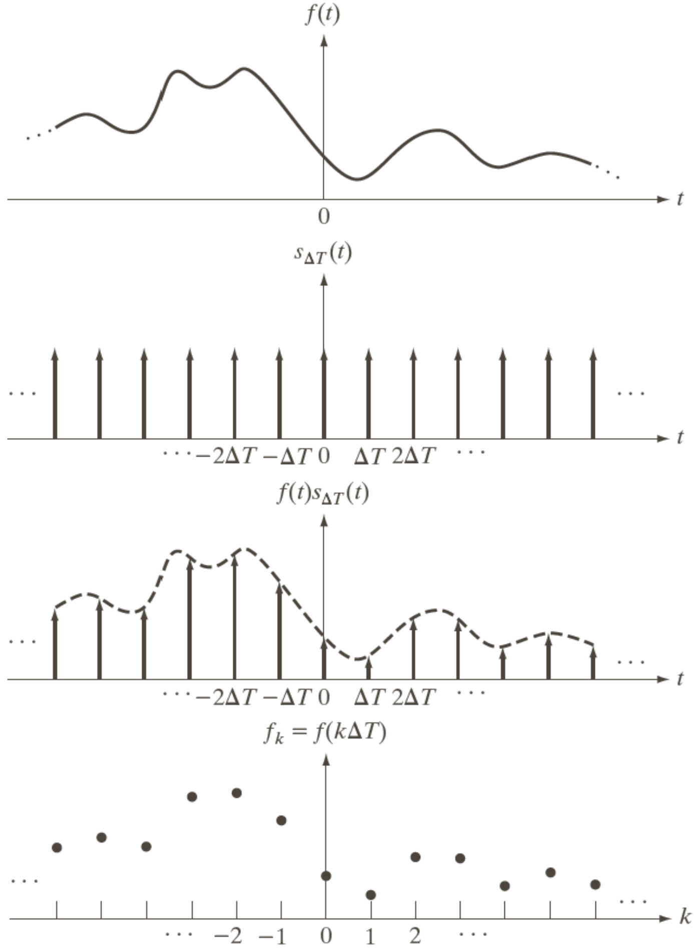
Samples from continuous function

$$f_k = f(k\Delta T)$$

Representation as a function of t

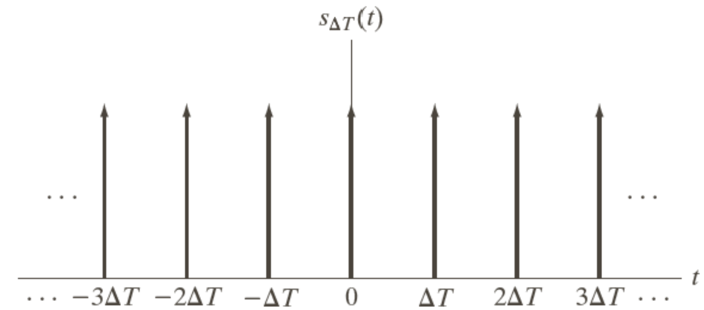
- Multiplication of f(t) with Shah

$$\tilde{f}(t) = f(t) s_{\Delta T}(t) = \sum_{k=-\infty}^{\infty} f_k \delta(t - k\Delta T)$$



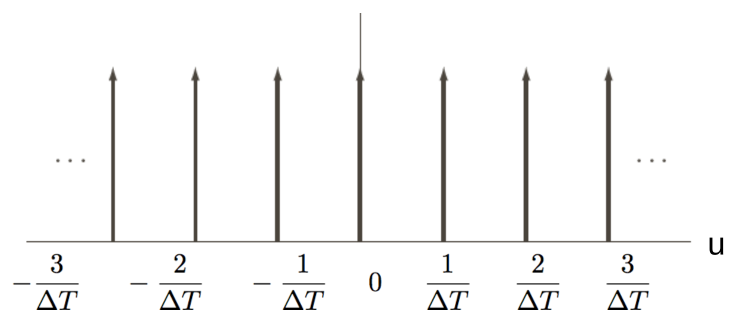
Fourier Series of A Shah Functional

$$s(t) = \sum_{k=-\infty}^{\infty} \delta(t - k\Delta T)$$



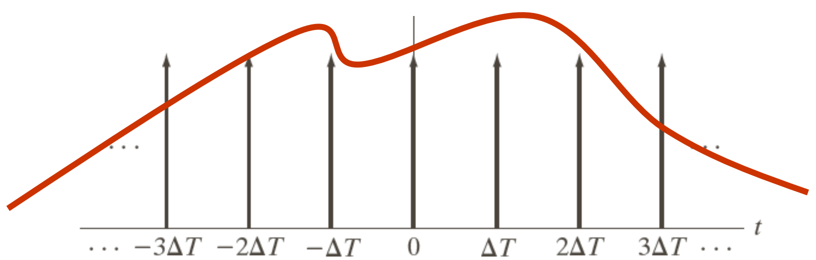
$$S(u) = \frac{1}{\Delta T} \sum_{k=-\infty}^{\infty} \delta(u - \frac{k}{\Delta T})$$

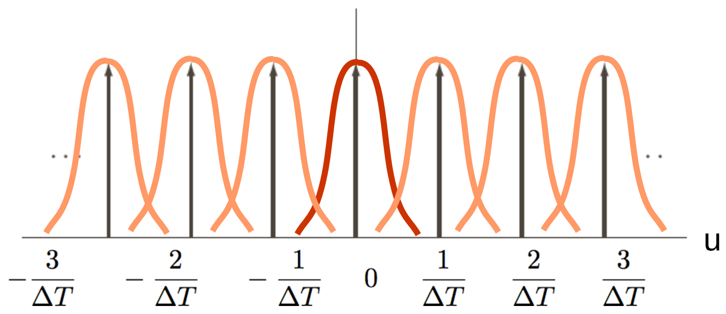
$$= \sum_{k=-\infty}^{\infty} \delta(\Delta T u - k)$$



Fourier Transform of A Discrete Sampling

$$\tilde{f}(t) = f(t)s(t) \longleftrightarrow \tilde{F}(u) = F(u) * S(u)$$

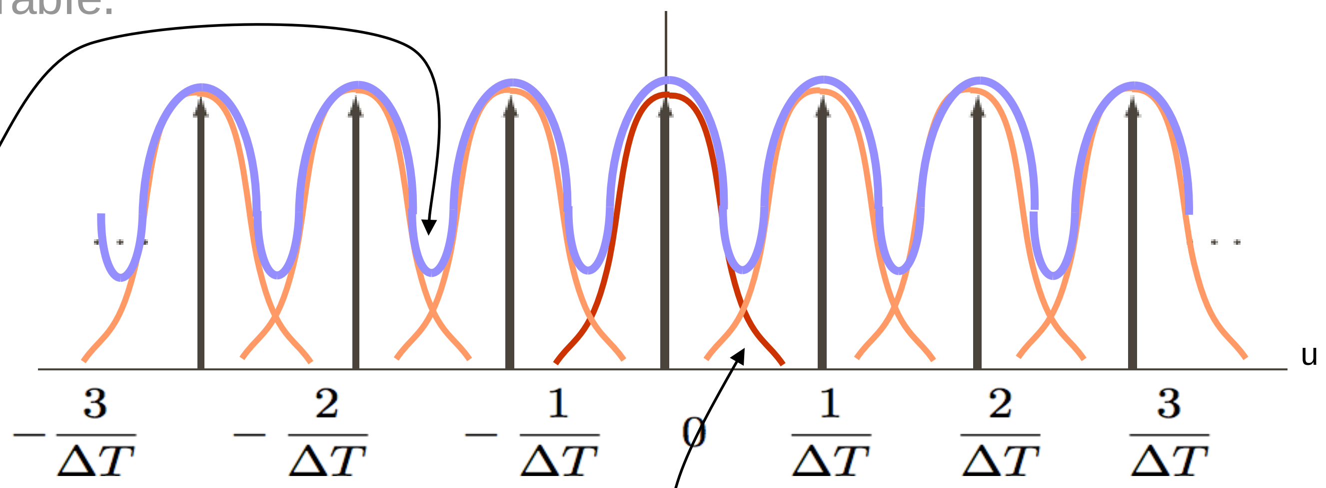




Fourier Transform of A Discrete Sampling

$$\tilde{F}(u) = F(u) * S(u)$$

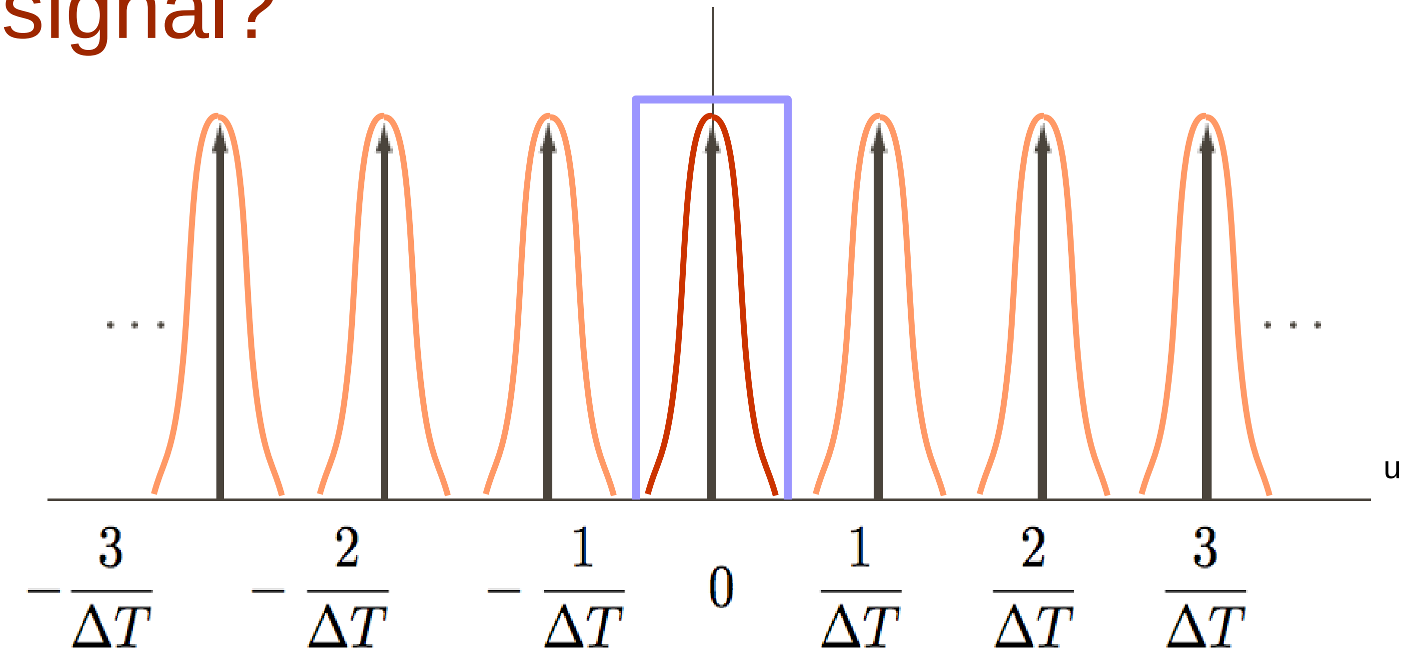
Frequencies get mixed. The original signal is not recoverable.



Energy from higher freqs gets folded back down into lower freqs – Aliasing

What if F(u) is Narrower in the Fourier Domain?

- No aliasing!
- How could we recover the original signal?



What Comes Out of This Model

- Sampling criterion for complete recovery
- An understanding of the effects of sampling
  - Aliasing and how to avoid it
- Reconstruction of signals from discrete samples

Shannon Sampling Theorem

- Assuming a signal that is band limited:

$$f(t) \longleftrightarrow F(u) \qquad |F(u)| = 0 \;\; \forall \;\; |u| > B$$

- Given set of samples from that signal

$$f_k = f(k\Delta T) \qquad \Delta T \leq \frac{1}{2B}$$

- Samples can be used to generate the original signal
  – Samples and continuous signal are equivalent

Sampling Theorem

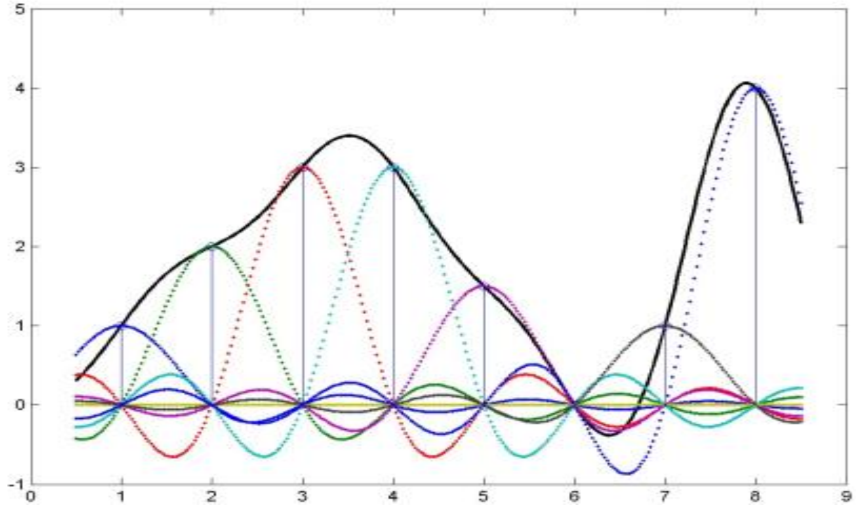
- Quantifies the amount of information in a signal
  - Discrete signal contains limited frequencies
  - Band-limited signals contain no more information then their discrete equivalents
- Reconstruction by cutting away the repeated signals in the Fourier domain
  - Convolution with sinc function in space/time

Reconstruction

- Convolution with sinc function

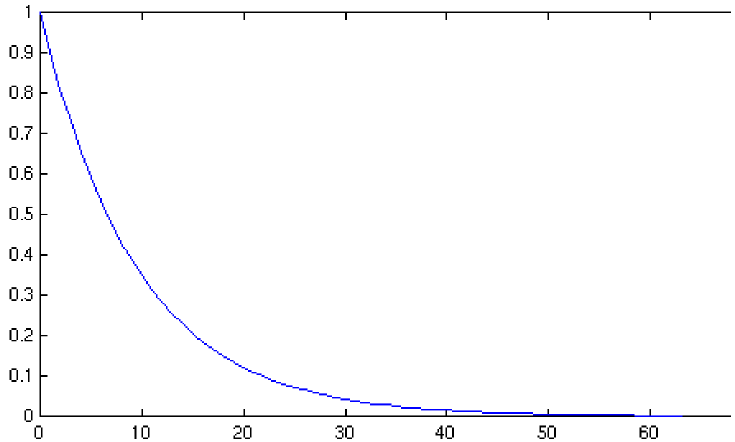
$$f(t) = \tilde{f}(t) * \mathbb{F}^{-1}\left[\mathrm{rect}\left(\Delta \mathrm{Tu}\right)\right]$$

$$= \left(\sum_k f_k \delta(t - k\Delta T)\right) * \mathrm{sinc}\left(\frac{\mathrm{t}}{\Delta \mathrm{T}}\right) = \sum_{\mathrm{k}} \mathrm{f_k}\ \mathrm{sinc}\left(\frac{\mathrm{t} - \mathrm{k}\Delta \mathrm{T}}{\Delta \mathrm{T}}\right)$$

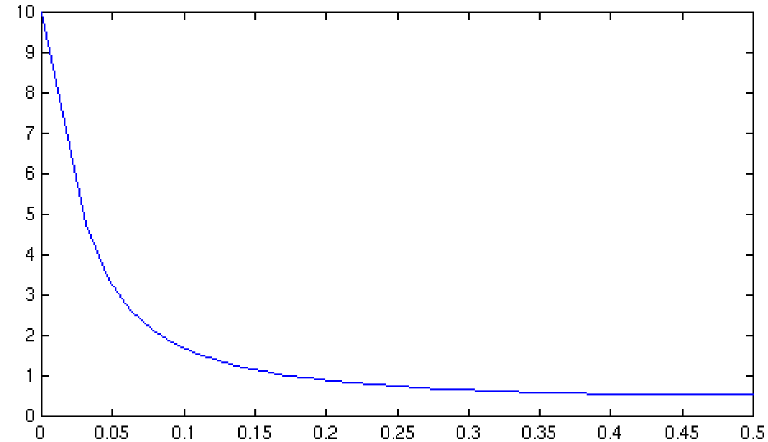


Sinc Interpolation Issues

- Must functions are not band limited
- Forcing functions to be band-limited can cause artifacts (ringing)

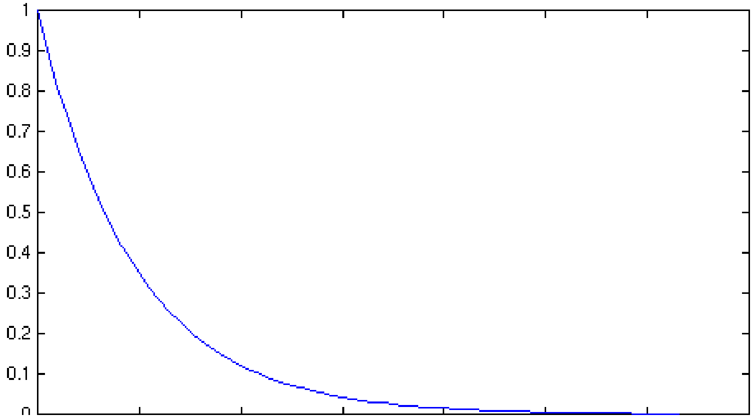


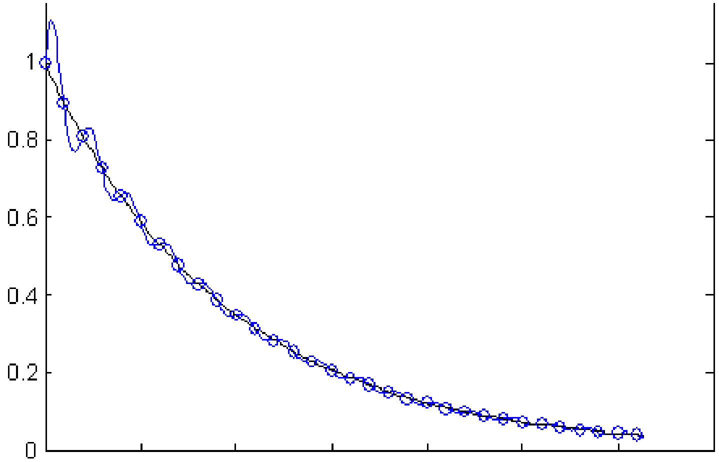
f(t)



|F(s)|

Sinc Interpolation Issues





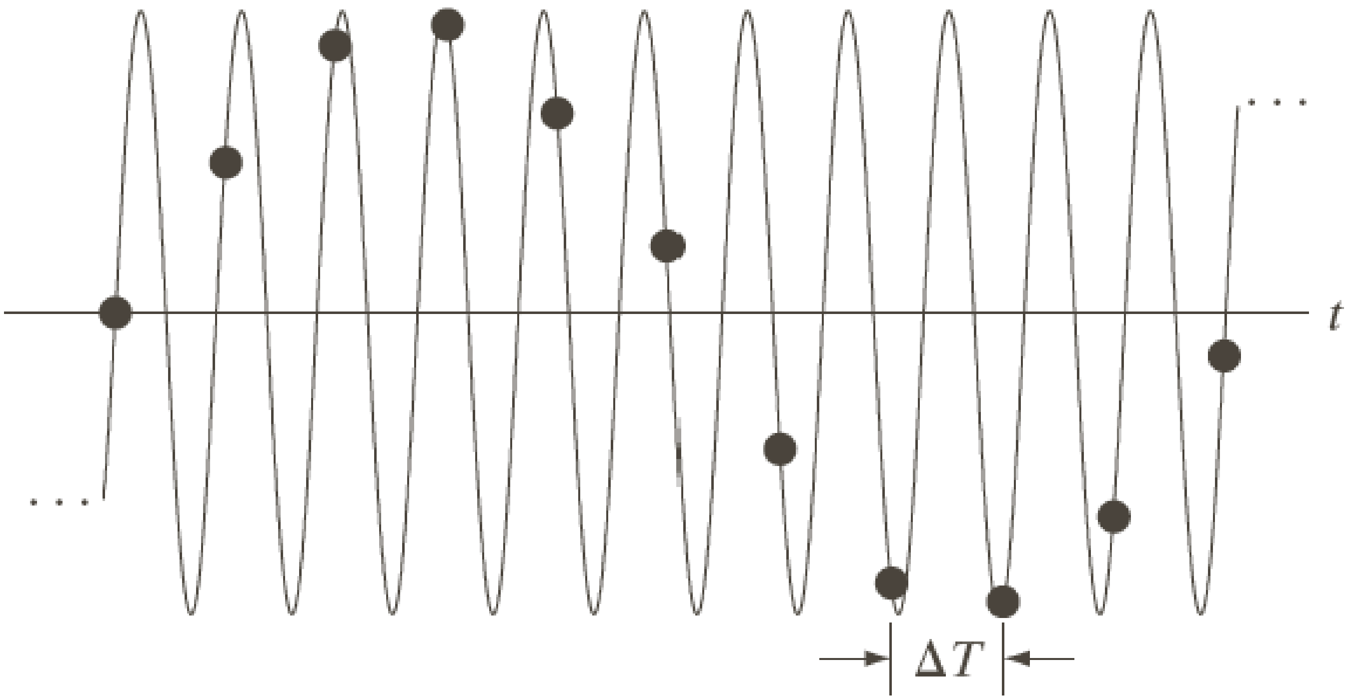
Ringing - Gibbs phenomenon

Other issues:

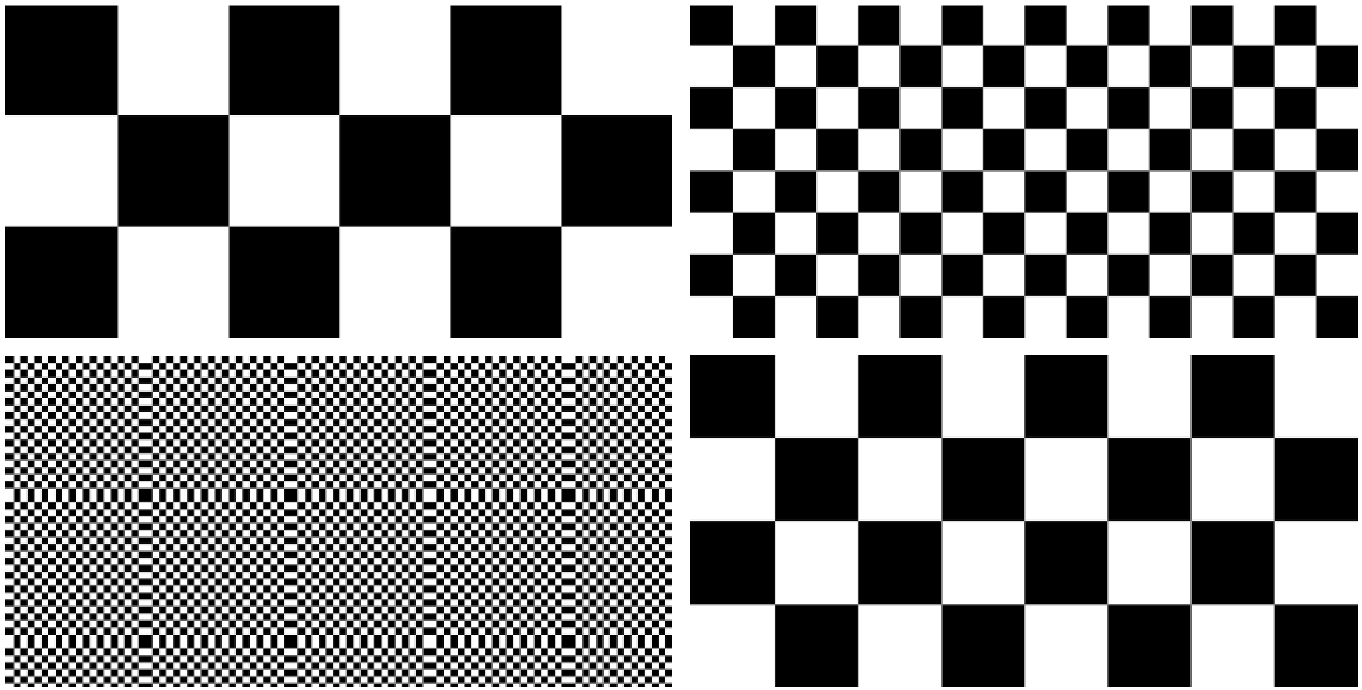
Sinc is infinite - must be truncated

Aliasing

- Reminder: high frequencies appear as low frequencies when undersampled



Aliasing



16 pixels

8 pixels

0.9174 pixels

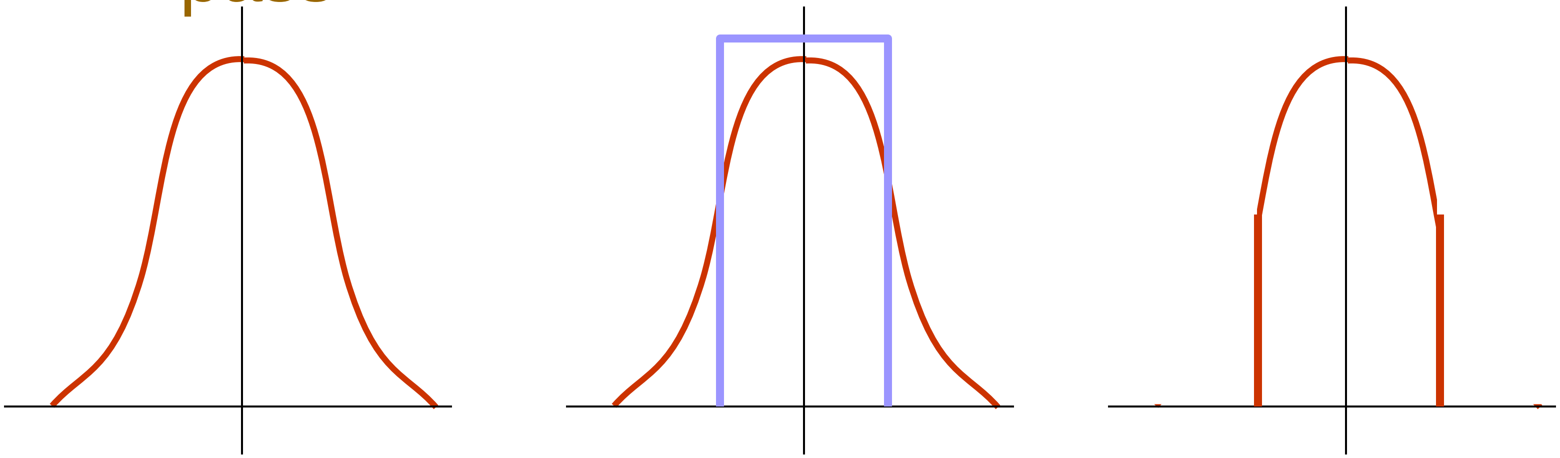
0.4798 pixels

Aliasing

Aliasing in digital videos

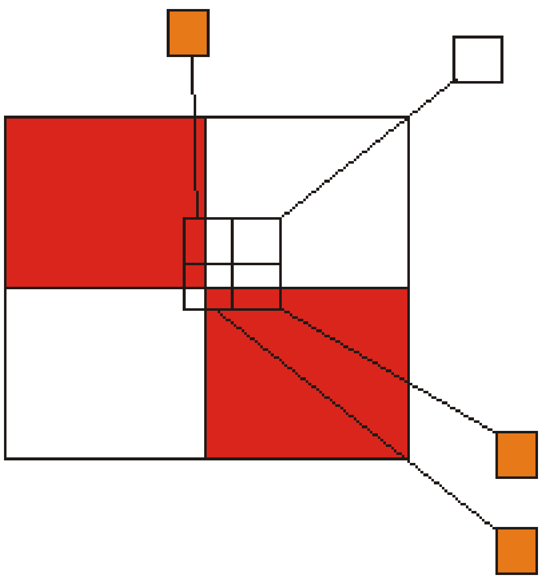
Overcoming Aliasing

- Filter data prior to sampling
  - Ideally - band limit the data (conv with sinc function)
  - In practice - limit effects with fuzzy/soft low pass



Antialiasing in Graphics

- Screen resolution produces aliasing on underlying geometry



Multiple high-res samples get averaged to create one screen sample

Antialiasing

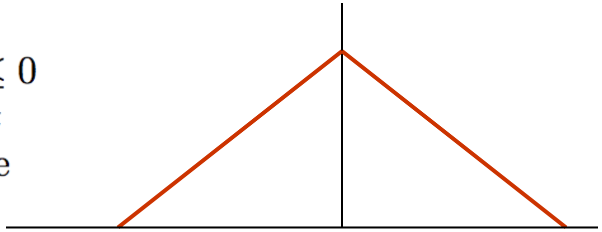
Interpolation as Convolution

- Any discrete set of samples can be considered as a functional

$$\tilde{f}(t) = \sum_k f_k \delta(t - k\Delta T)$$

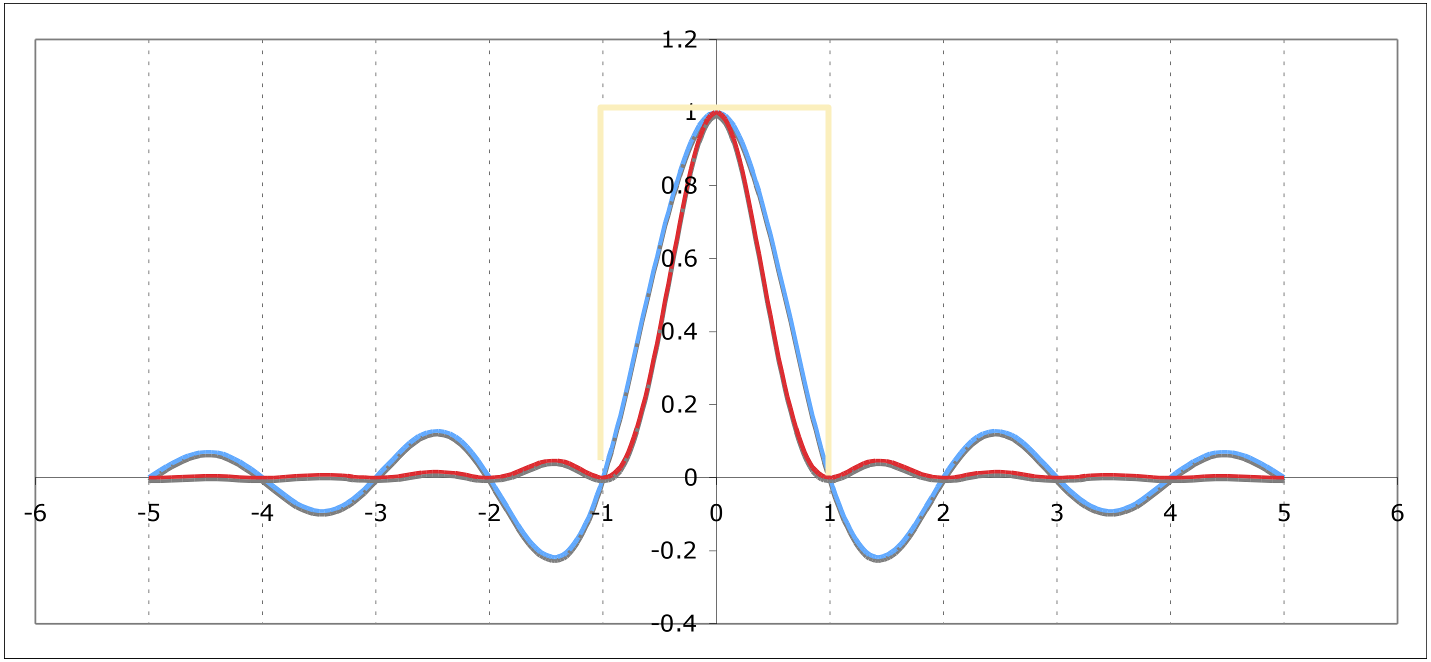
- Any linear interpolant can be considered as a convolution
  - Nearest neighbor - rect(t)
  - Linear - tri(t)

$$\mathrm{tri(t)} = \begin{cases} t+1 & -1 \leq t \leq 0 \\ 1-t & 0 \leq t \leq t \\ 0 & \text{otherwise} \end{cases}$$



Convolution-Based Interpolation

- Can be studied in terms of Fourier Domain
- Issues
  - Pass energy (=1) in band
  - Low energy out of band
  - Reduce hard cut off (Gibbs, ringing)



Fast Fourier Transform

With slides from Richard Stern, CMU

DFT

- Ordinary DFT is $O(N^2)$
- DFT is slow for large images

- Exploit periodicity and symmetricity
- Fast FT is O(N log N)
- FFT can be faster than convolution

Fast Fourier Transform

- Divide and conquer algorithm
- Gauss ~1805
- Cooley & Tukey 1965

- For $N = 2^K$

The Cooley-Tukey Algorithm

- Consider the DFT algorithm for an integer power of 2, $N = 2^{\nu}$

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} = \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}; \; W_N = e^{-j2\pi/N}$$

- Create separate sums for even and odd values of *n:*

$$X[k] = \sum_{n \; even} x[n] W_N^{nk} + \sum_{n \; odd} x[n] W_N^{nk}$$

- Letting $n = 2r$ for *n* even and $n = 2r+1$ for *n* odd, we obtain

$$X[k] = \sum_{r=0}^{(N/2)-1} x[2r] W_N^{2rk} + \sum_{r=0}^{(N/2)-1} x[2r+1] W_N^{(2r+1)k}$$

The Cooley-Tukey Algorithm

- Splitting indices in time, we have obtained

$$X[k] = \sum_{r=0}^{(N/2)-1} x[2r] W_N{}^{2rk} + \sum_{r=0}^{(N/2)-1} x[2r+1] W_N{}^{(2r+1)k}$$

- But $W_N^2 = e^{-j2\pi 2/N} = e^{-j2\pi/(N/2)} = W_{N/2}$ and $W_N^{2rk} W_N^k = W_N^k W_{N/2}^{rk}$

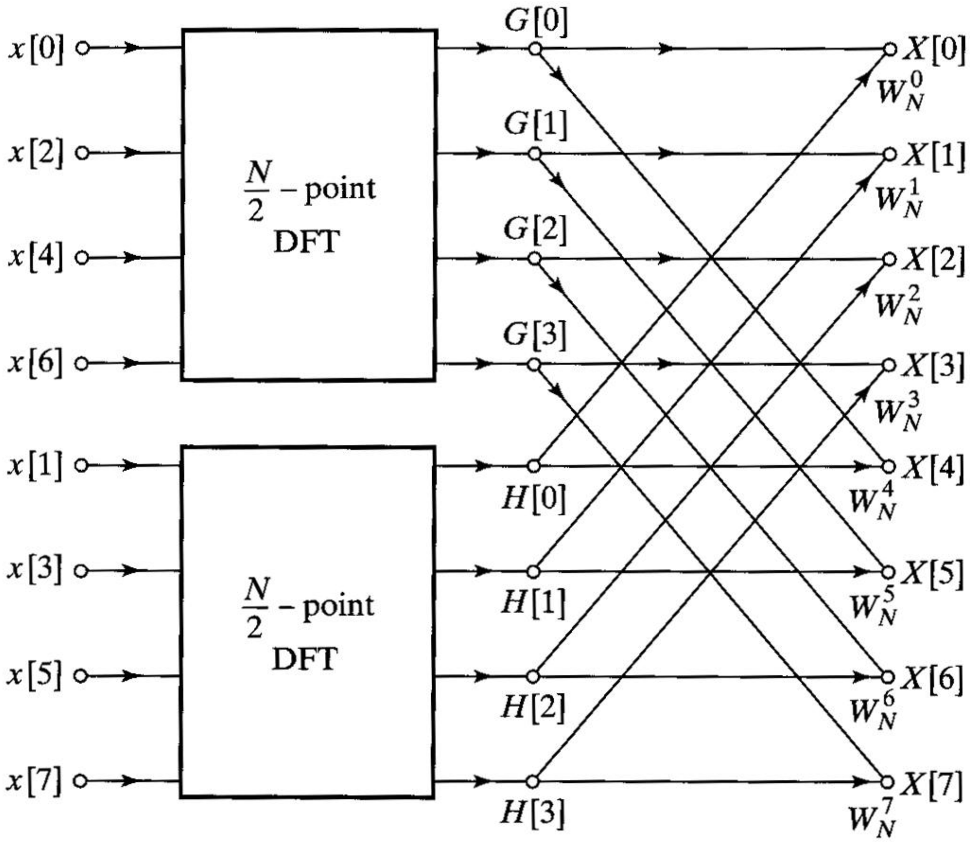
So …

$$X[k] = \underbrace{\sum_{n=0}^{(N/2)-1} x[2r] W_{N/2}^{rk}} + W_N^k \underbrace{\sum_{n=0}^{(N/2)-1} x[2r+1] W_{N/2}^{rk}}$$

N/2-point DFT of *x[2r]* *N/2*-point DFT of x*[2r+1]*

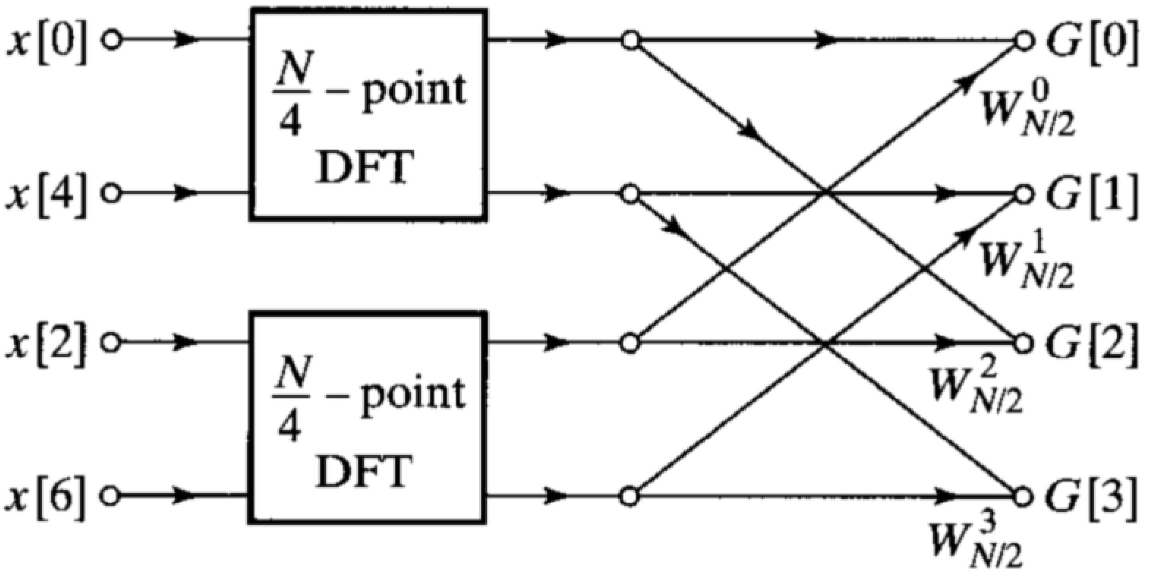
Example: N=8

- Divide and reuse



Example: N=8, Upper Part

- Continue to divide and reuse



Two-Point FFT

- The expression for the 2-point DFT is:

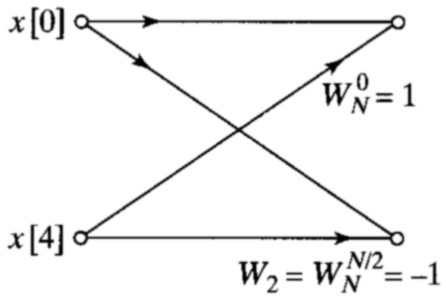
$$X[k] = \sum_{n=0}^{1} x[n] W_2^{nk} = \sum_{n=0}^{1} x[n] e^{-j2\pi nk/2}$$

- Evaluating for $k = 0, 1$ we obtain

$$X[0] = x[0] + x[1]$$

$$X[1] = x[0] + e^{-j2\pi 1/2} x[1] = x[0] - x[1]$$

which in signal flowgraph notation looks like ...



This topology is referred to as the basic butterfly

Modern FFT

- FFTW

  http://www.fftw.org/