Part II

Fourier Transform in Image Processing

CS/BIOEN 6640 U of Utah
Guido Gerig
(slides modified from
Marcel Prastawa 2012)

1D: Common Transform Pairs Summary

Fourier Transform Pairs

Pair Number	$x(t)$	$X(f)$
1.	$\Pi\left(\frac{t}{\tau}\right)$	$\tau \operatorname{sinc} \tau f$
2.	$2W \operatorname{sinc} 2Wt$	$\Pi\left(\frac{f}{2W}\right)$
3.	$\Lambda\left(\frac{t}{\tau}\right)$	$\tau \operatorname{sinc}^2 \tau f$
4.	$\exp(-\alpha t)u(t),\ \alpha > 0$	$\frac{1}{\alpha + j2\pi f}$
5.	$t\exp(-\alpha t)u(t),\ \alpha > 0$	$\frac{1}{(\alpha + j2\pi f)^2}$
6.	$\exp(-\alpha\|t\|),\ \alpha > 0$	$\frac{2\alpha}{\alpha^2 + (2\pi f)^2}$
7.	$e^{-\pi(t/\tau)^2}$	$\tau e^{-\pi(f\tau)^2}$
8.	$\delta(t)$	1
9.	1	$\delta(f)$
10.	$\delta(t - t_0)$	$\exp(-j2\pi f t_0)$
11.	$\exp(j2\pi f_0 t)$	$\delta(f - f_0)$
12.	$\cos 2\pi f_0 t$	$\frac{1}{2}\delta(f - f_0) + \frac{1}{2}\delta(f + f_0)$
13.	$\sin 2\pi f_0 t$	$\frac{1}{2j}\delta(f - f_0) - \frac{1}{2j}\delta(f + f_0)$
14.	$u(t)$	$(j2\pi f)^{-1} + \frac{1}{2}\delta(f)$
15.	$\operatorname{sgn} t$	$(j\pi f)^{-1}$
16.	$\frac{1}{\pi t}$	$-j \operatorname{sgn}(f)$
17.	$\hat{x}(t) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{x(\lambda)}{t - \lambda}\, d\lambda$	$-j \operatorname{sgn}(f) X(f)$
18.	$\sum_{m=-\infty}^{\infty} \delta(t - mT_s)$	$f_s \sum_{m=-\infty}^{\infty} \delta(f - mf_s),$ $f_s = T_s^{-1}$

source

FT Properties: Convolution

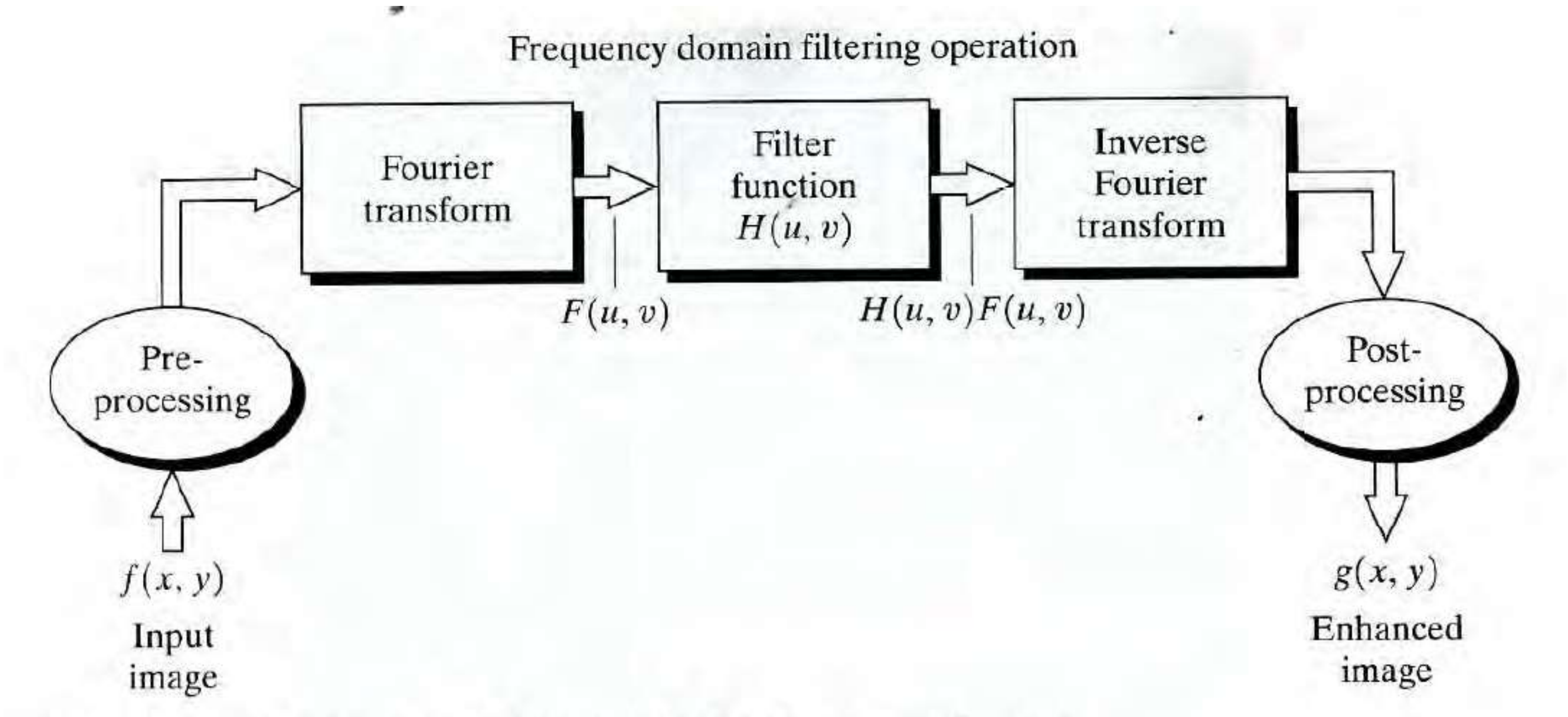
- See book DIP 4.2.5:

$$\mathcal{F}[f(t) \otimes g(t)] = F(s).G(s)$$

- Convolution in space/time domain is equiv. to multiplication in frequency domain.

Time Convolution	$f(t) \star g(t)$	$\leftrightarrow$	$F(\omega)G(\omega)$
Frequency Convolution	$f(t)g(t)$	$\leftrightarrow$	$\frac{1}{2\pi}F(\omega) \star G(\omega)$

Important Application



Filtering in frequency Domain

FT Properties

Functional relationships [edit]

The Fourier transforms in this table may be found in Erdélyi (1954) or Kam

	Function	Fourier transform unitary, ordinary frequency
	$f(x)$	$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi}\, dx$
101	$a \cdot f(x) + b \cdot g(x)$	$a \cdot \hat{f}(\xi) + b \cdot \hat{g}(\xi)$
102	$f(x-a)$	$e^{-2\pi i a \xi} \hat{f}(\xi)$
103	$e^{2\pi i a x} f(x)$	$\hat{f}(\xi - a)$
104	$f(ax)$	$\frac{1}{\|a\|} \hat{f}\left(\frac{\xi}{a}\right)$
105	$\hat{f}(x)$	$f(-\xi)$
106	$\frac{d^n f(x)}{dx^n}$	$(2\pi i \xi)^n \hat{f}(\xi)$
107	$x^n f(x)$	$\left(\frac{i}{2\pi}\right)^n \frac{d^n \hat{f}(\xi)}{d\xi^n}$
108	$(f * g)(x)$	$\hat{f}(\xi)\hat{g}(\xi)$
109	$f(x)g(x)$	$(\hat{f} * \hat{g})(\xi)$

Wikipedia Fourier Transforms

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	$f(at)$	$\leftrightarrow$	$\frac{1}{\lvert\lvert a\rvert\rvert} F(\omega / a)$
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	$f(t) g(t)$	$\leftrightarrow$	$\frac{1}{2\pi} F(\omega) \star G(\omega)$

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

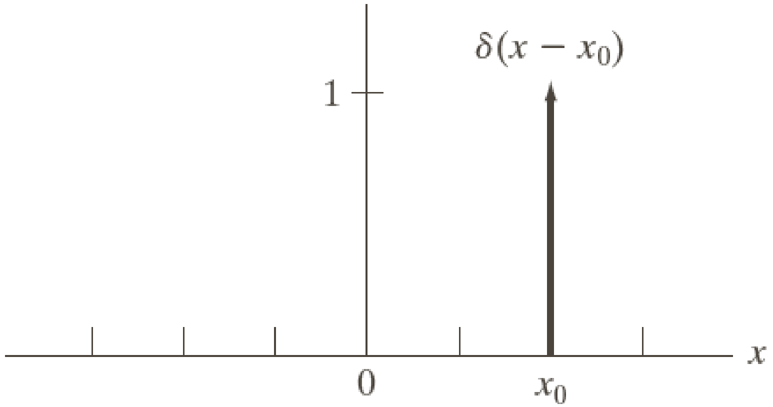
Aliasing

Discrete Sampling and Aliasing

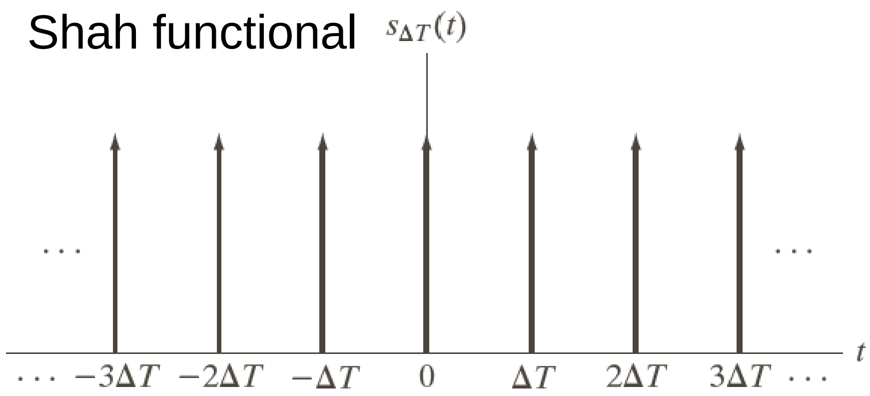
- Digital signals and images are discrete representations of the real world
  - Which is continuous
- What happens to signals/images when we sample them?
  - Can we quantify the effects?
  - Can we understand the artifacts and can we limit them?
  - Can we reconstruct the original image from the discrete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

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

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

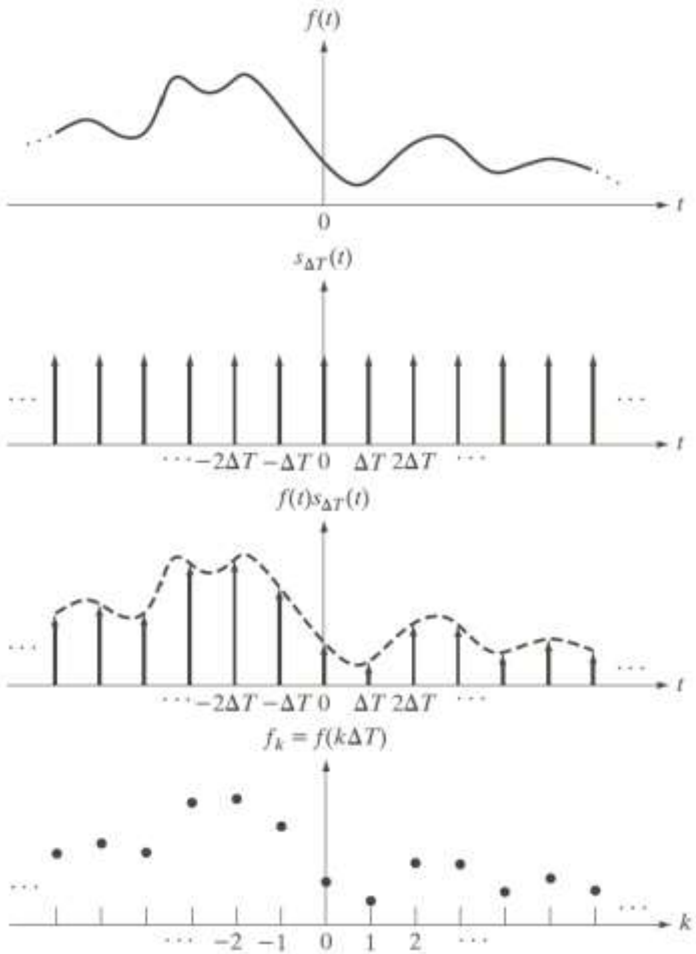
Samples from continuous function

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

Representation as a function of t

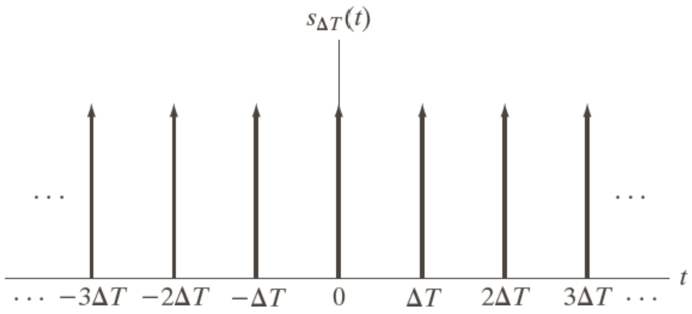
- Multiplication of f(t) with Shah

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



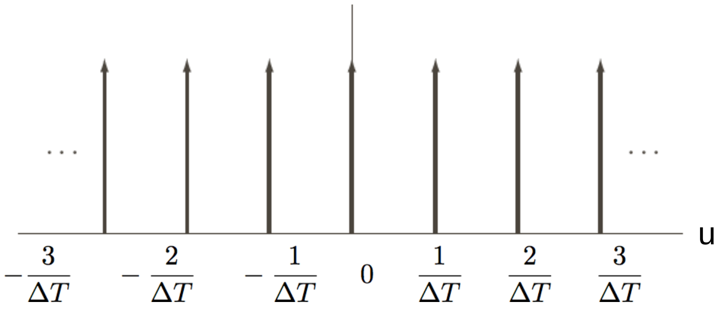
Fourier Series of A Shah Functional

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



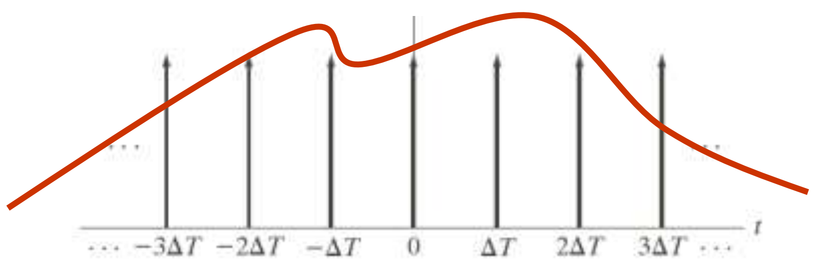
$$S(u) = \frac{1}{\Delta T} \sum_{k=-\infty}^{\infty} \delta(u - \frac{k}{\Delta T})$$

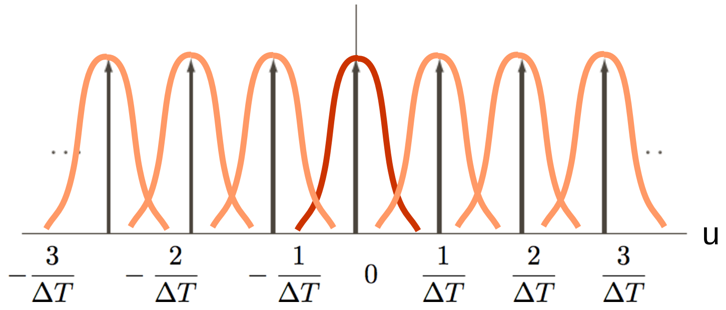
$$= \sum_{k=-\infty}^{\infty} \delta(\Delta T u - k)$$



Fourier Transform of A Discrete Sampling

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

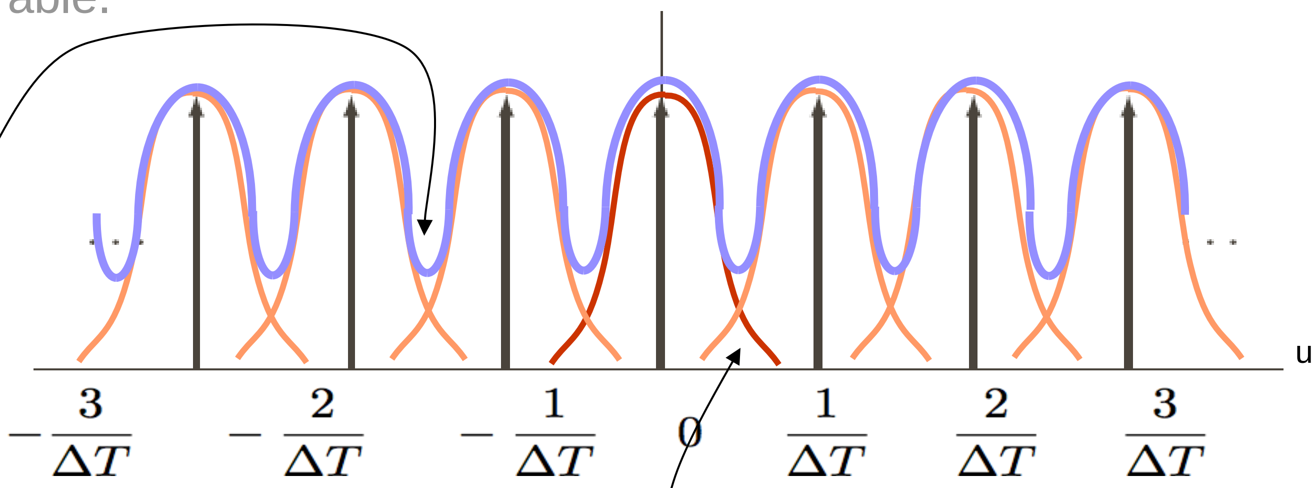




Fourier Transform of A Discrete Sampling

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

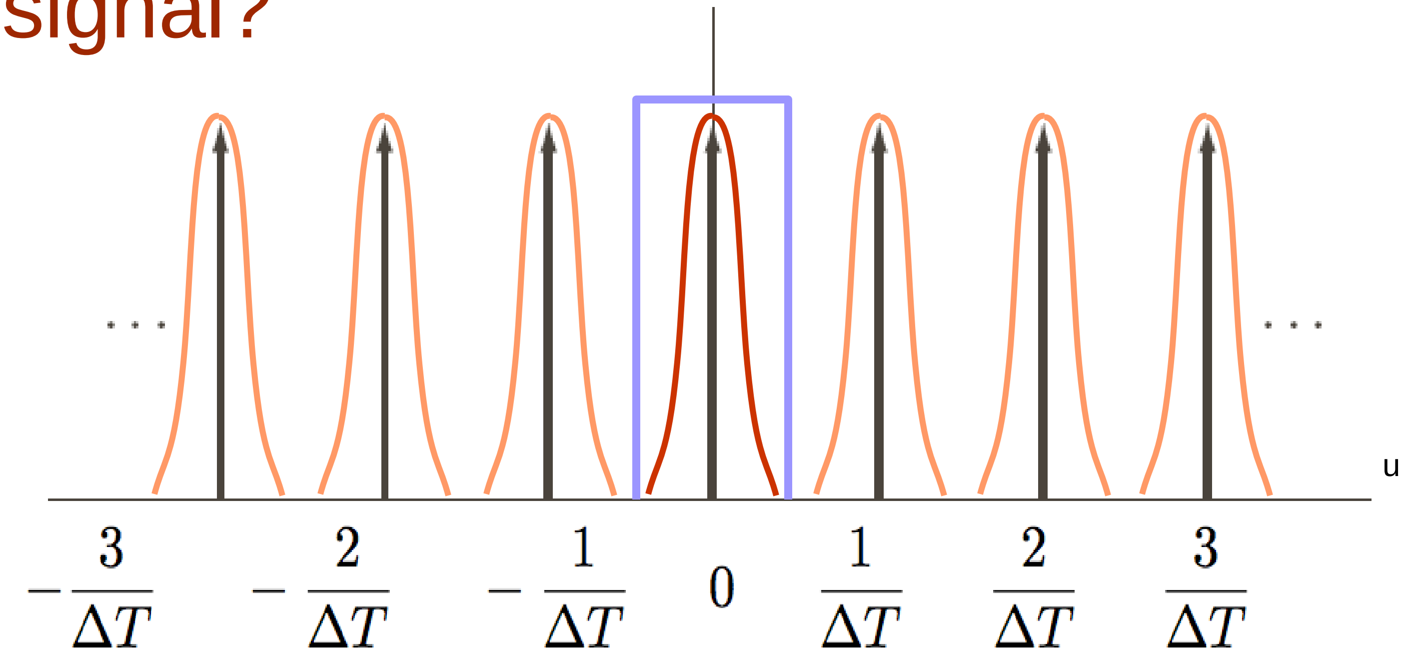
Frequencies get mixed. The original signal is not recoverable.



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

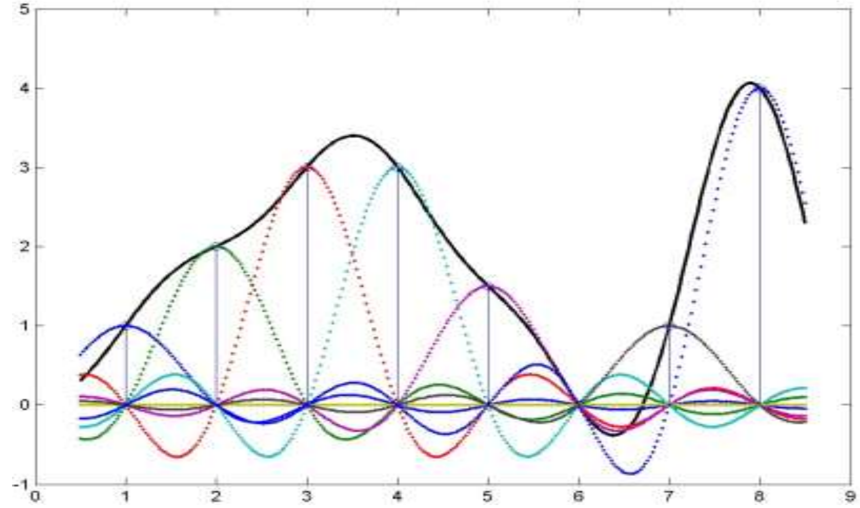
- Quantifies the amount of information in a signal
  - Discrete signal contains limited frequencies
  - Band-limited signals contain no more information then their discrete equivalents
- Reconstruction by cutting away the repeated signals in the Fourier domain
  - Convolution with sinc function in space/time

Reconstruction

- Convolution with sinc function

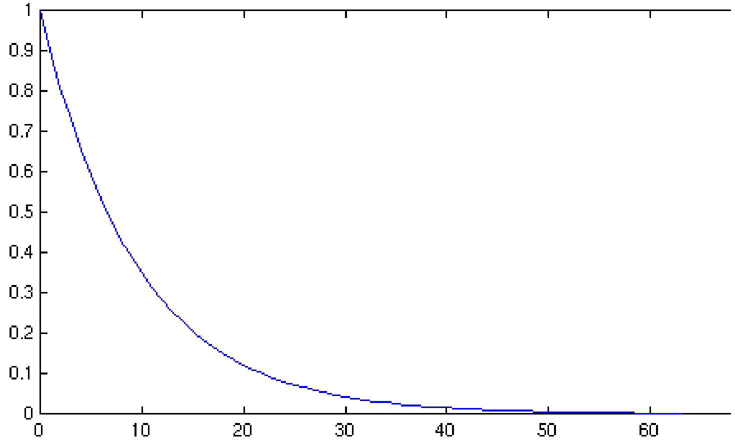
$$f(t) = \tilde{f}(t) * \mathbb{F}^{-1}\left[\mathrm{rect}\left(\Delta \mathrm{Tu}\right)\right]$$

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

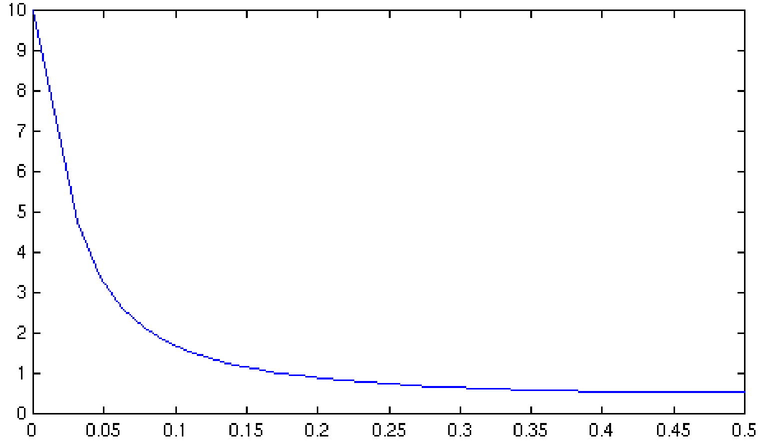


Sinc Interpolation Issues

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

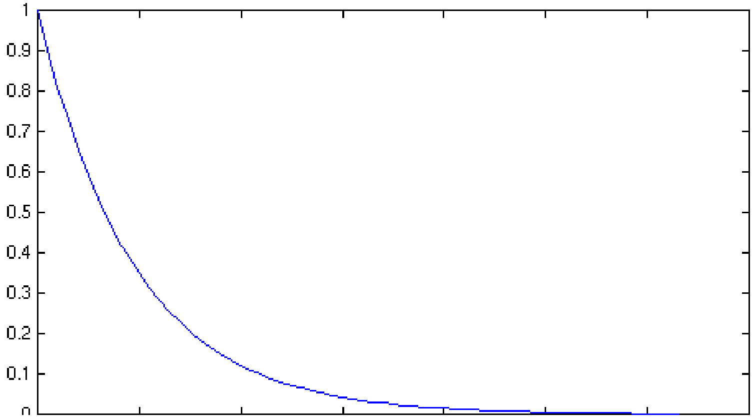


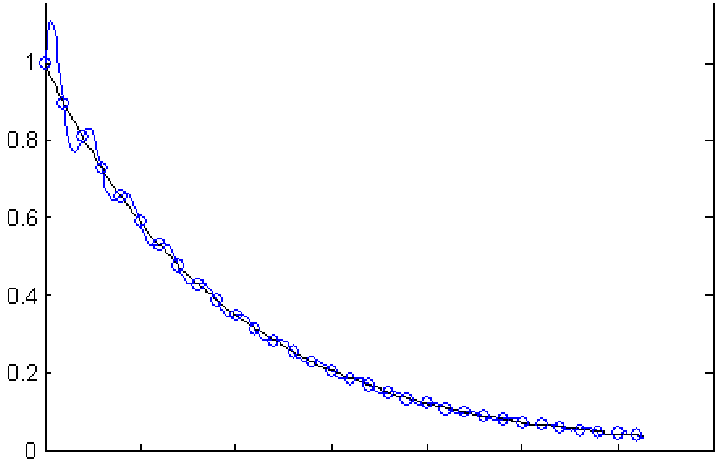
f(t)



|F(s)|

Sinc Interpolation Issues





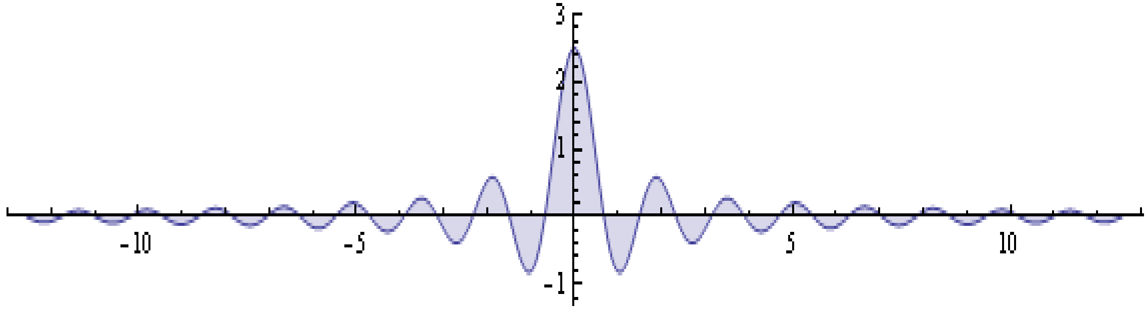
Ringing - Gibbs phenomenon

Other issues:

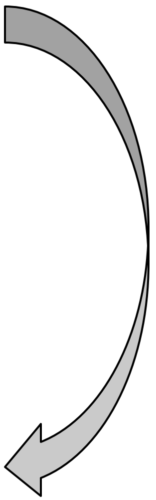
Sinc is infinite - must be truncated

Fourier Transform

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

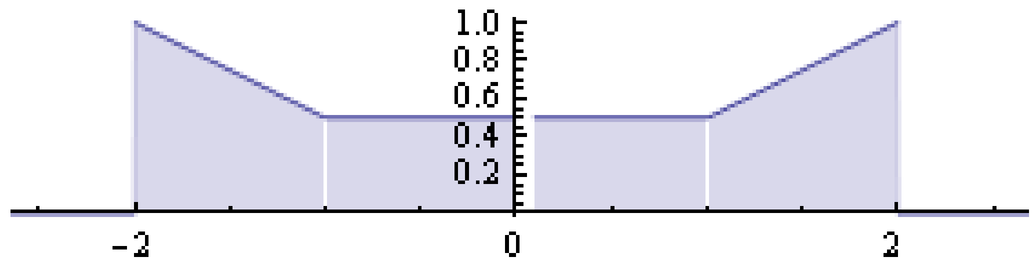


F(s)



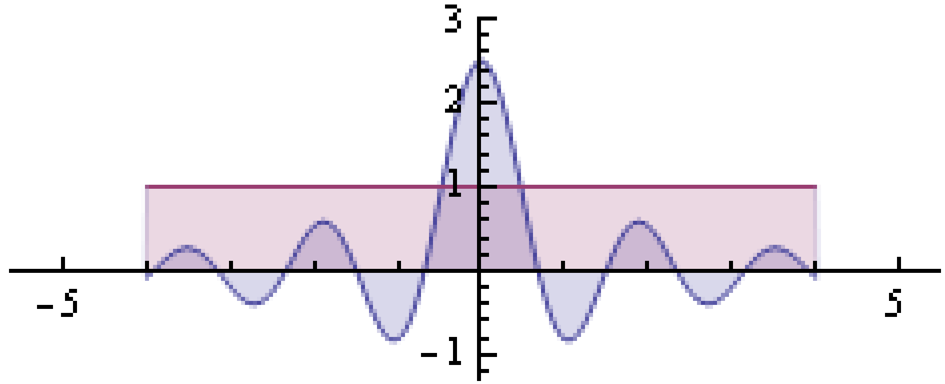
f(x)

InverseFourier

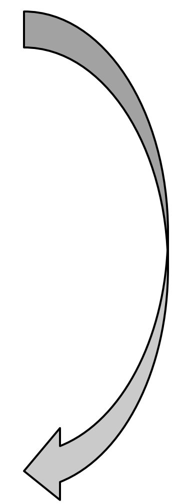


Cut-off High Frequencies

$F(s) = (4\text{sinc}(4s) - 2\text{sinc}^2(2s) + .5\text{sinc}^2(s))*(\text{HeavisidePi}(w/8))$

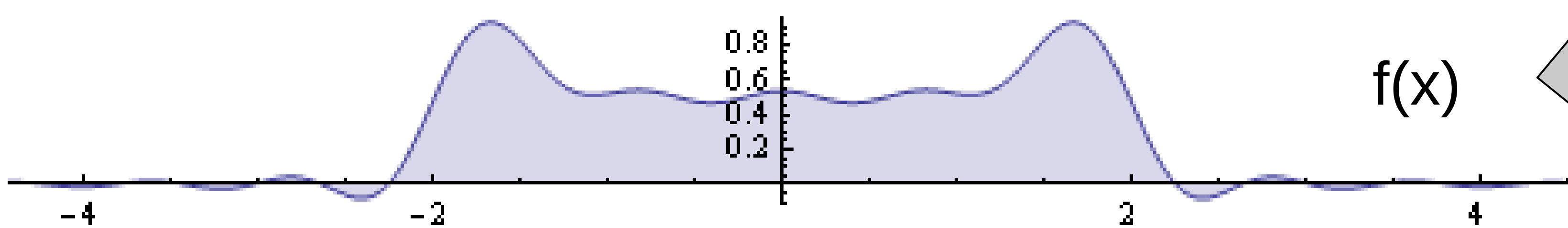


F(s)



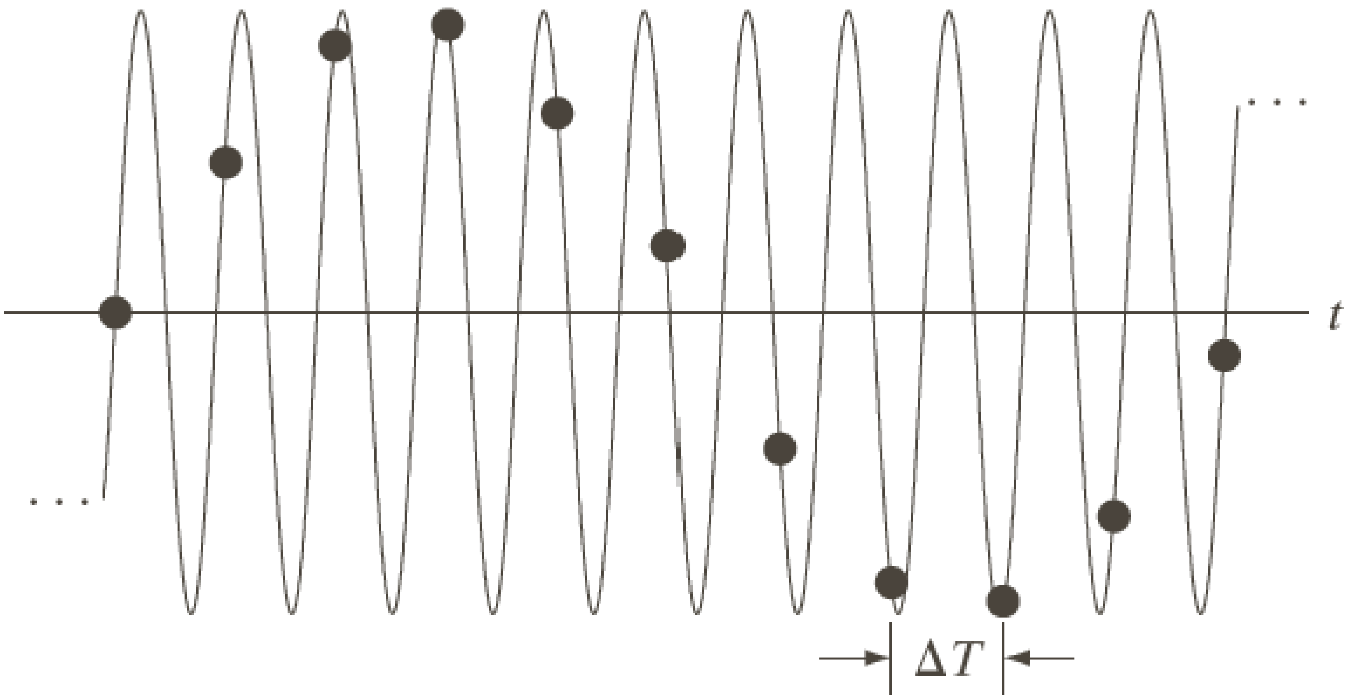
f(x)

InverseFourier

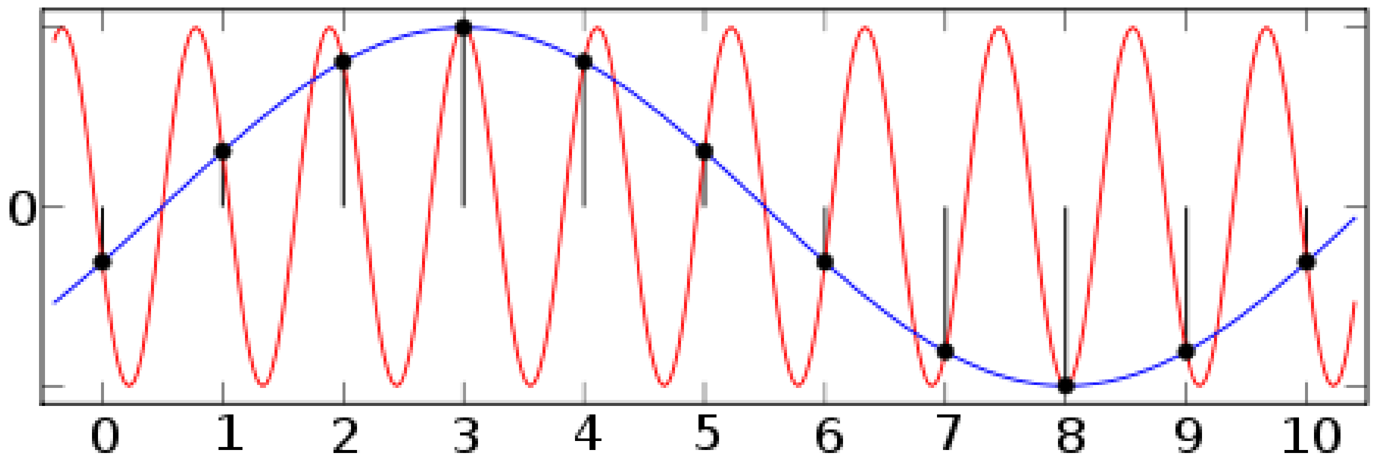


Aliasing

- Reminder: high frequencies appear as low frequencies when undersampled

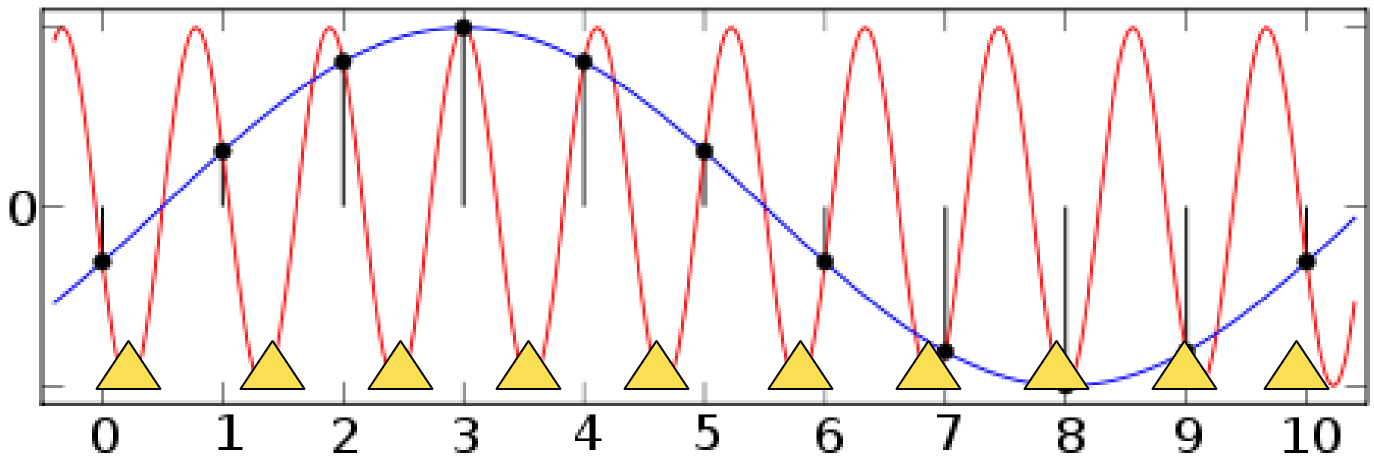


Sampling and Aliasing



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

Ideal Solution: More Samples

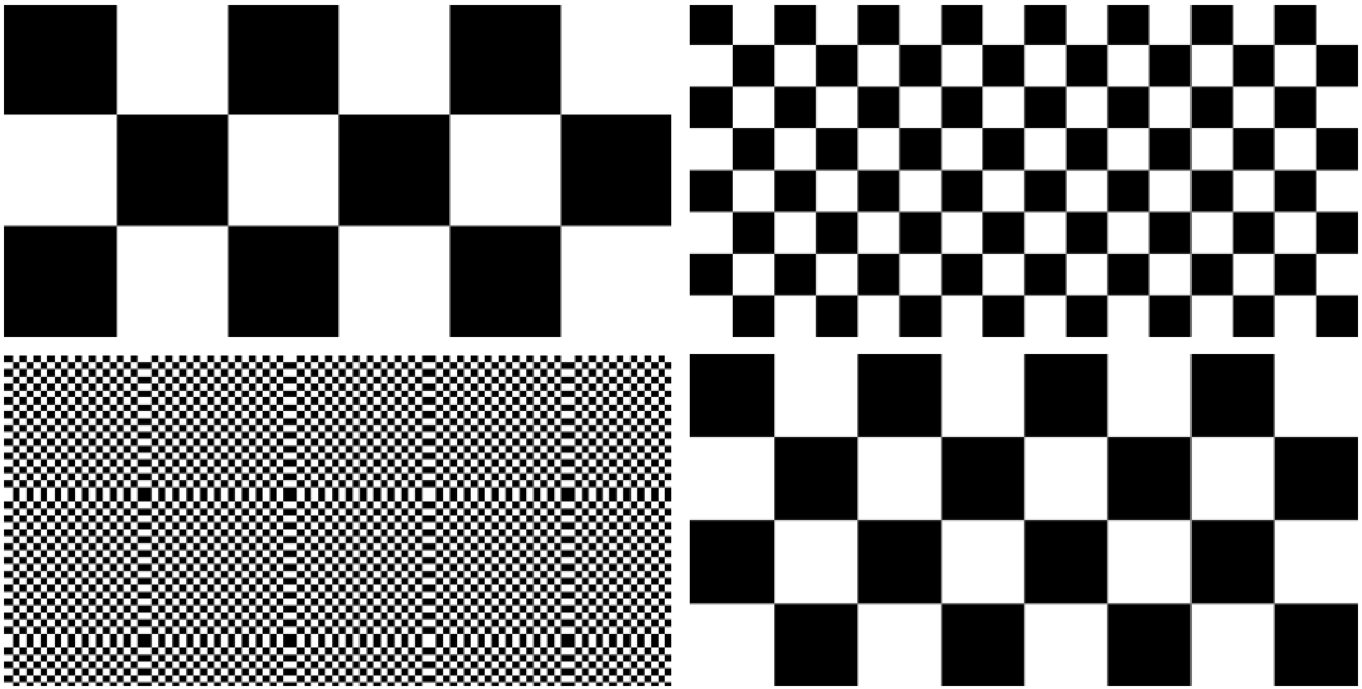


- Faster sampling rate allows us to distinguish the two signals
- Not always practical: hardware cost, longer scan time

Aliasing

16 pixels

8 pixels



0.9174 pixels

0.4798 pixels

Aliasing

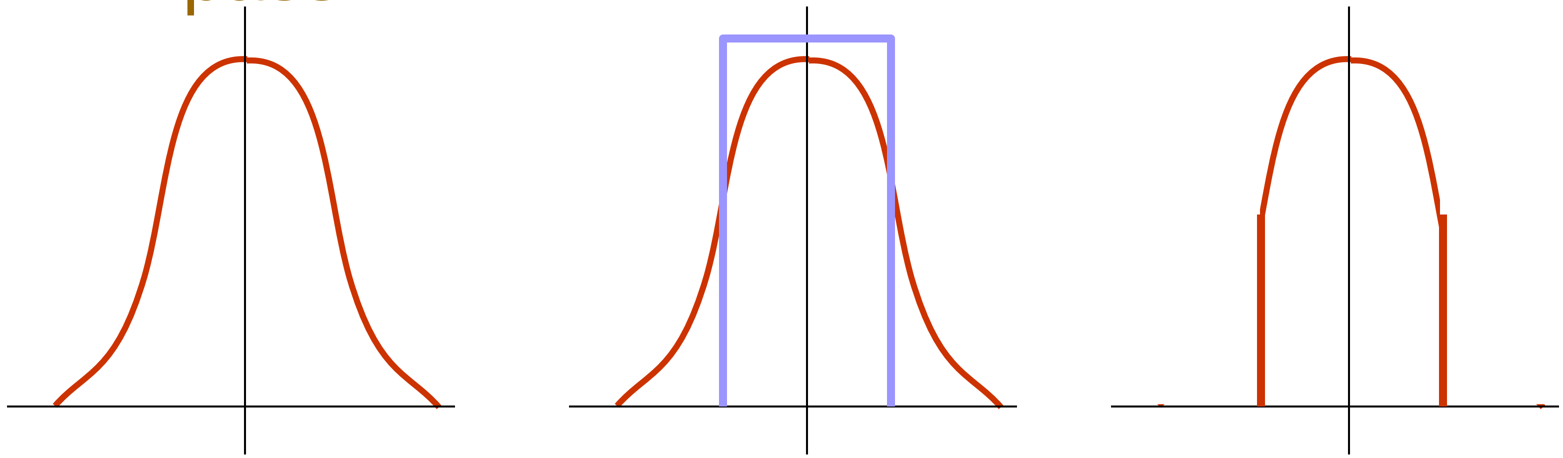
Aliasing in digital videos

Video1

Video2

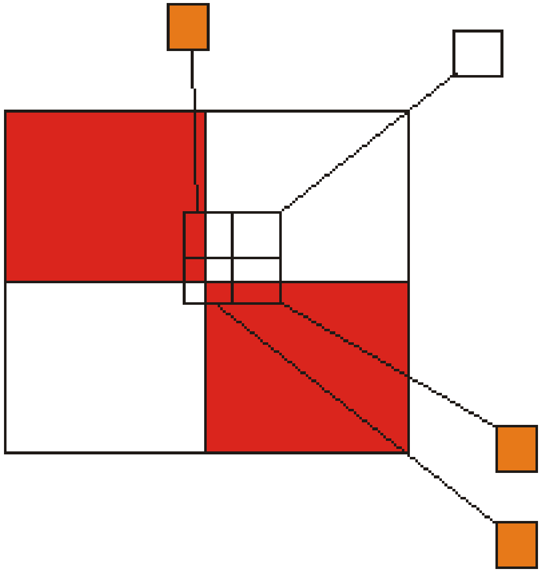
Overcoming Aliasing

- 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

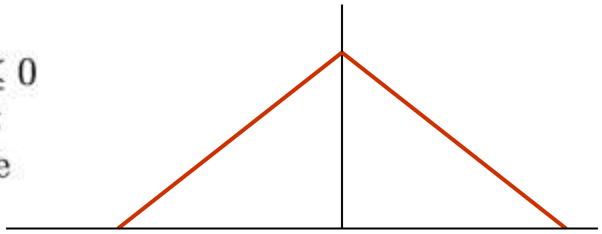
Interpolation as Convolution

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

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

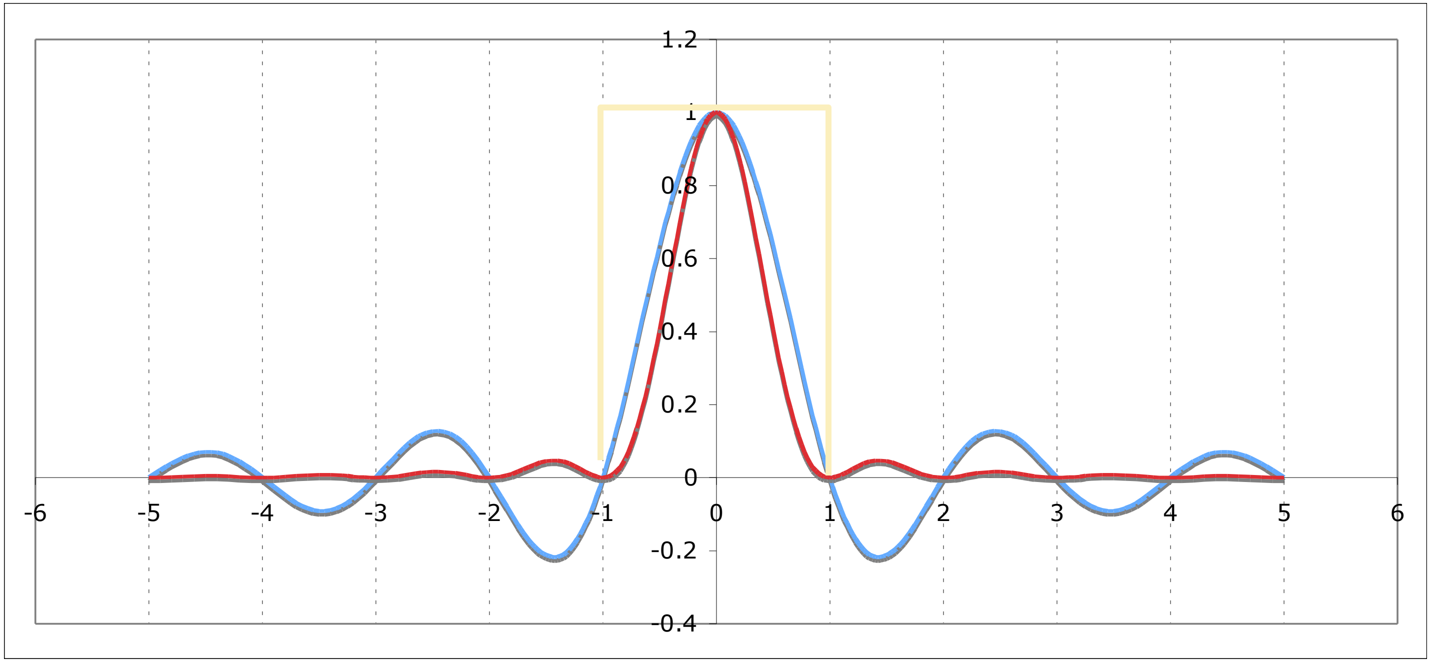
- Any linear interpolant can be considered as a convolution
  - Nearest neighbor - rect(t)
  - Linear - tri(t)

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



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)



Fourier Transform of Images

2D Fourier Transform

- Forward transform:

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

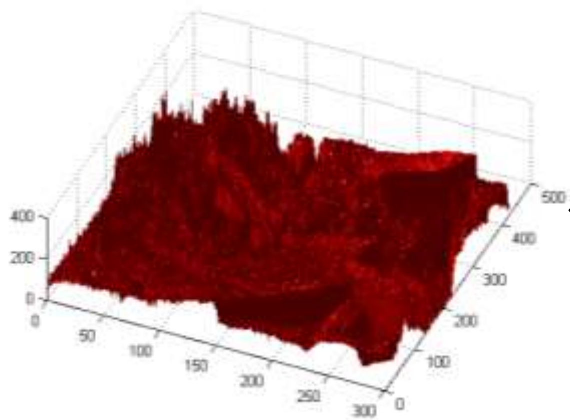
- Backward transform:

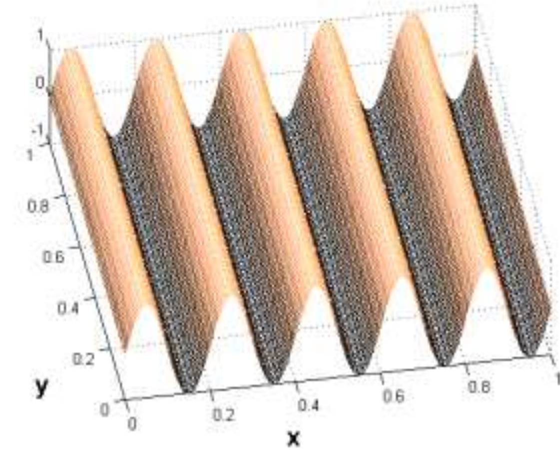
$$f(x,y) = \int \int_{-\infty}^{\infty} F(u,v) e^{j2\pi(xu+yv)} du\, dv$$

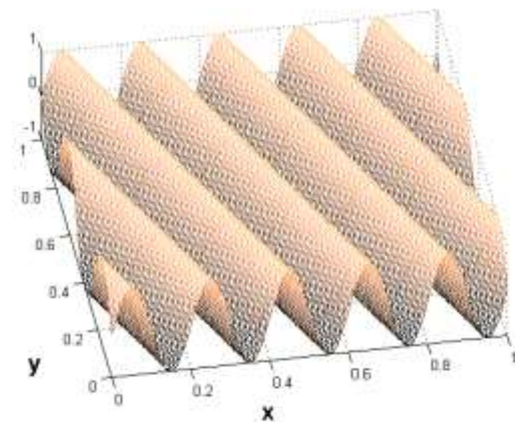
- Forward transform to freq. yields complex values (magnitude and phase):

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

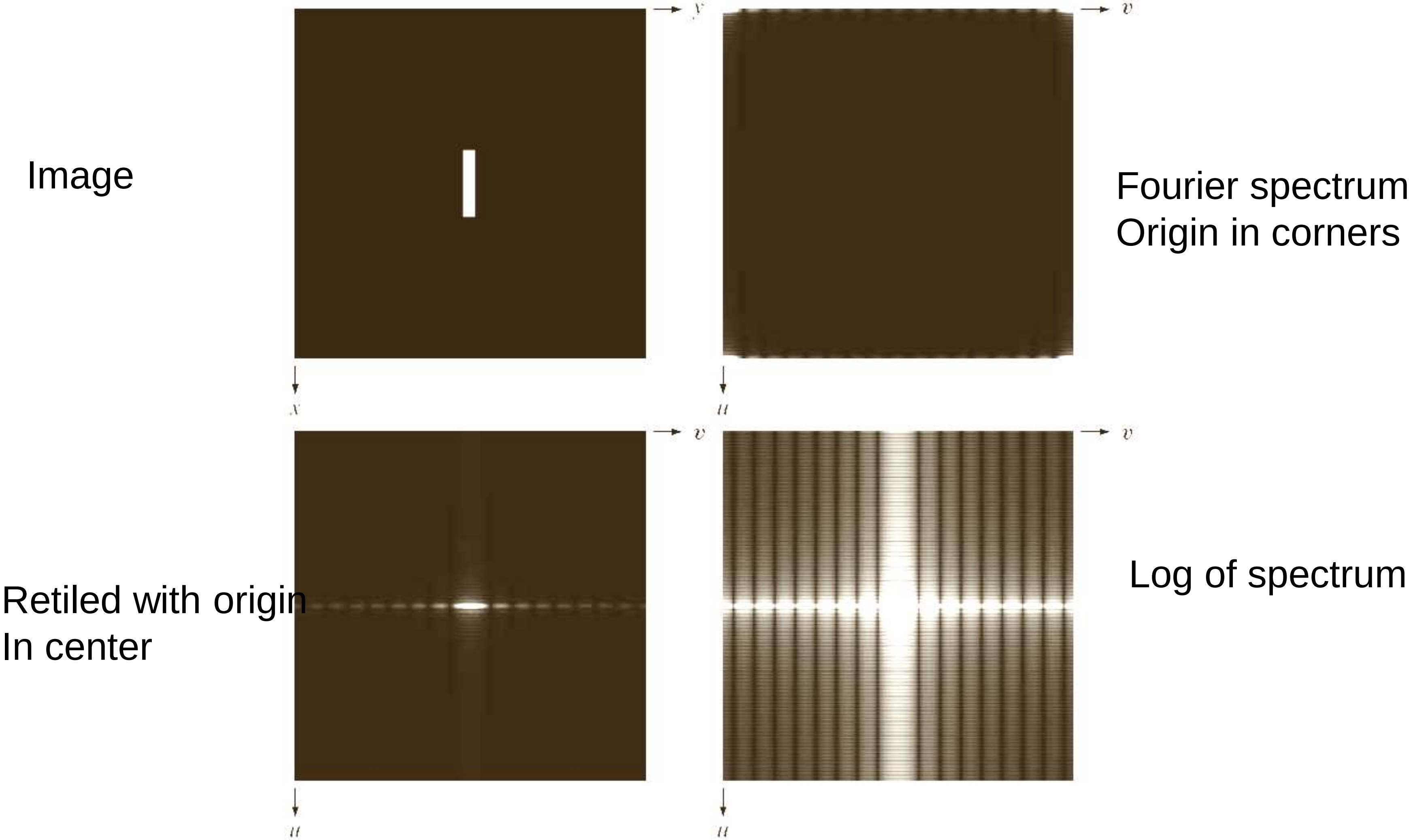
2D Fourier Transform



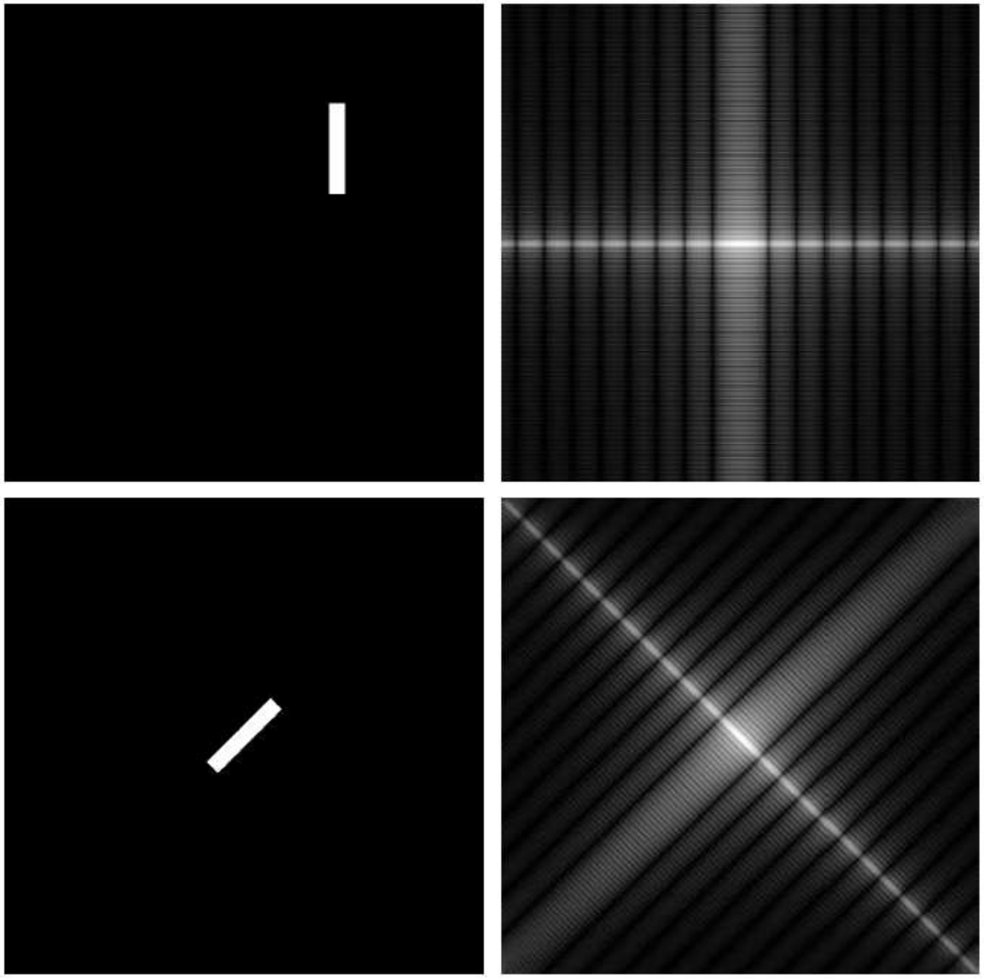




Fourier Spectrum



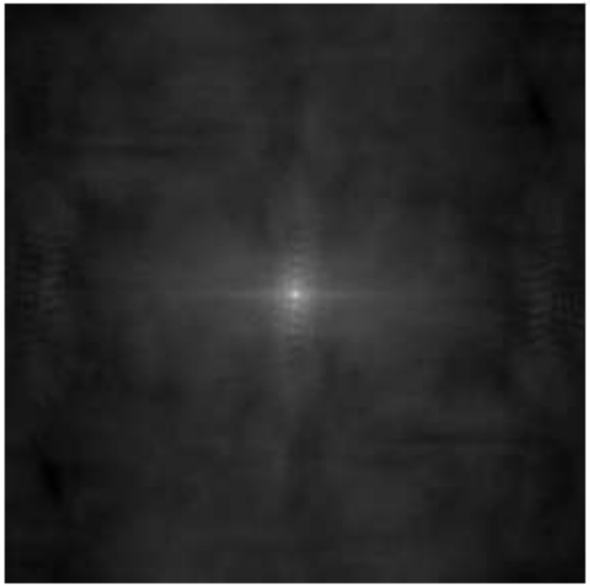
Fourier Spectrum – Translation and Rotation



Phase vs Spectrum

Image

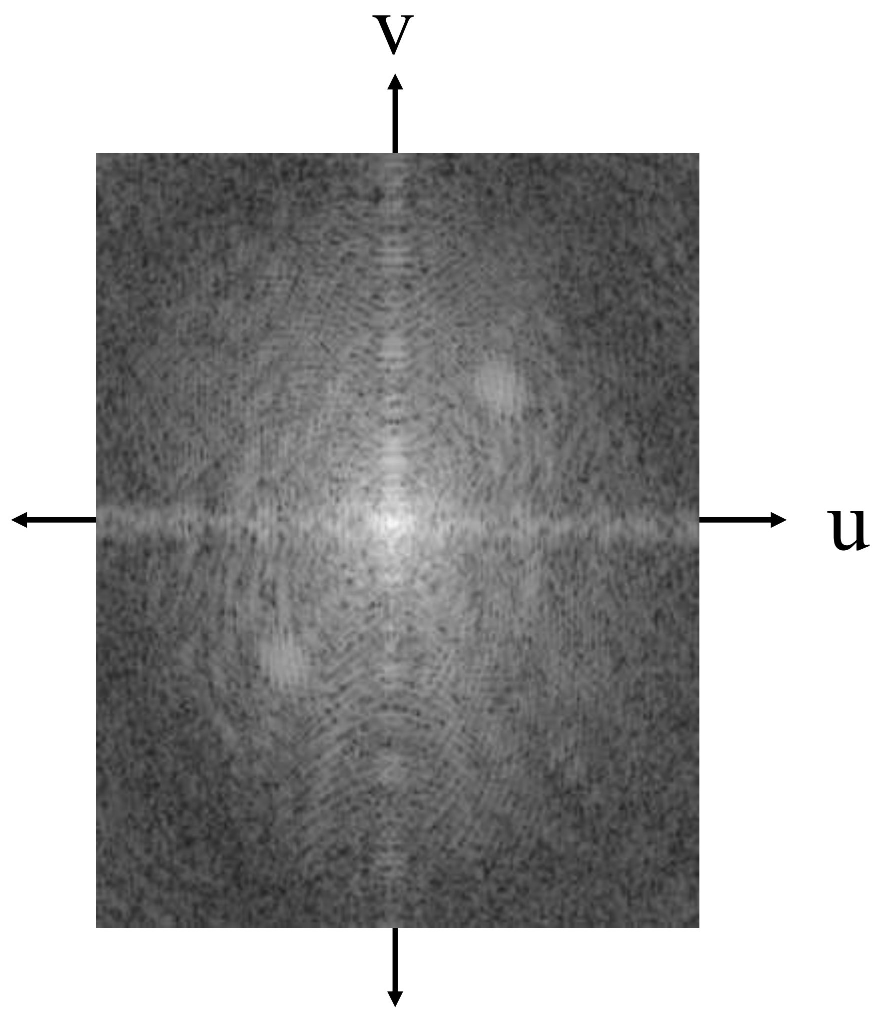
Reconstruction from phase map

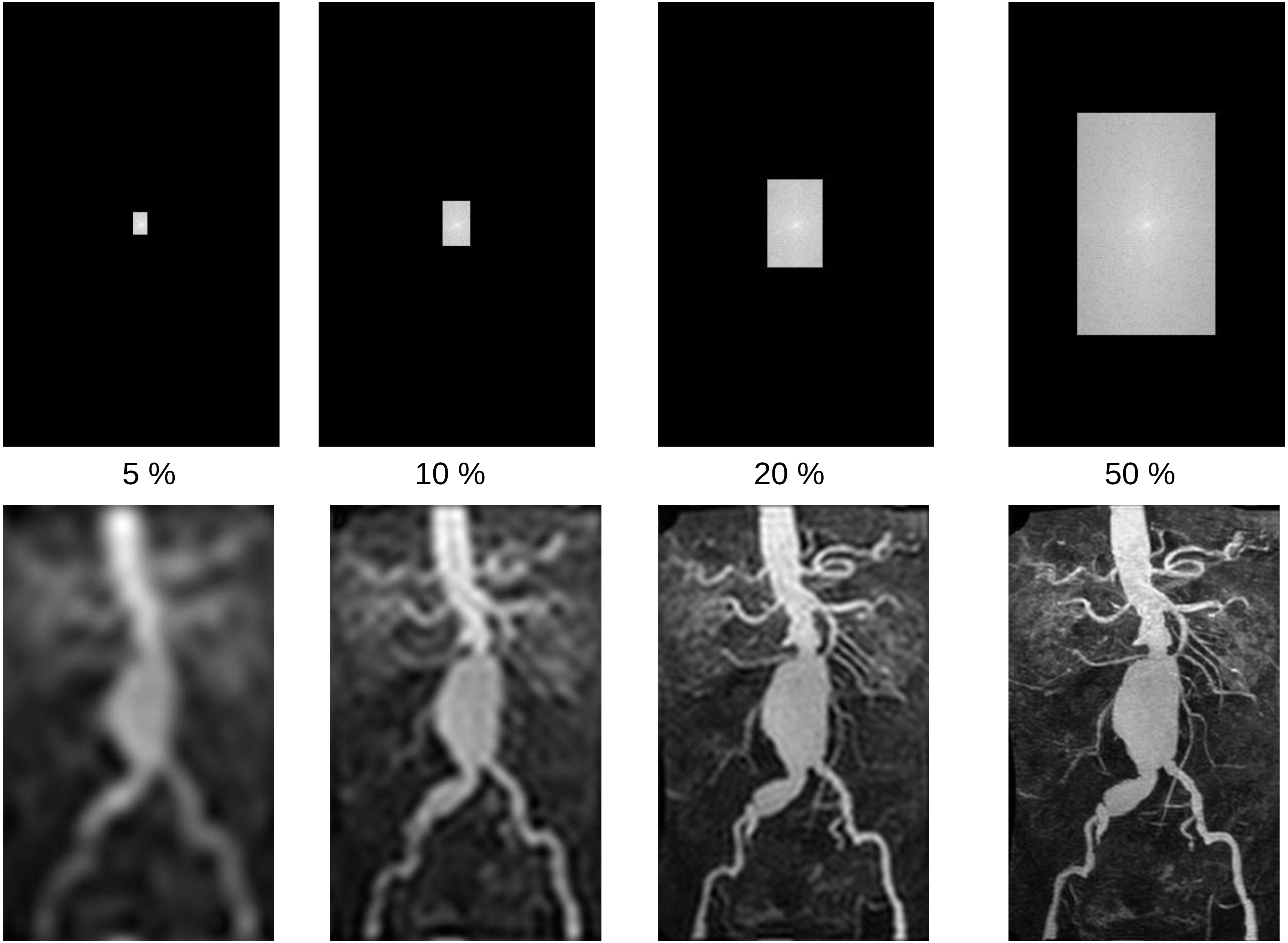


Reconstruction from spectrum

Image

Fourier Space



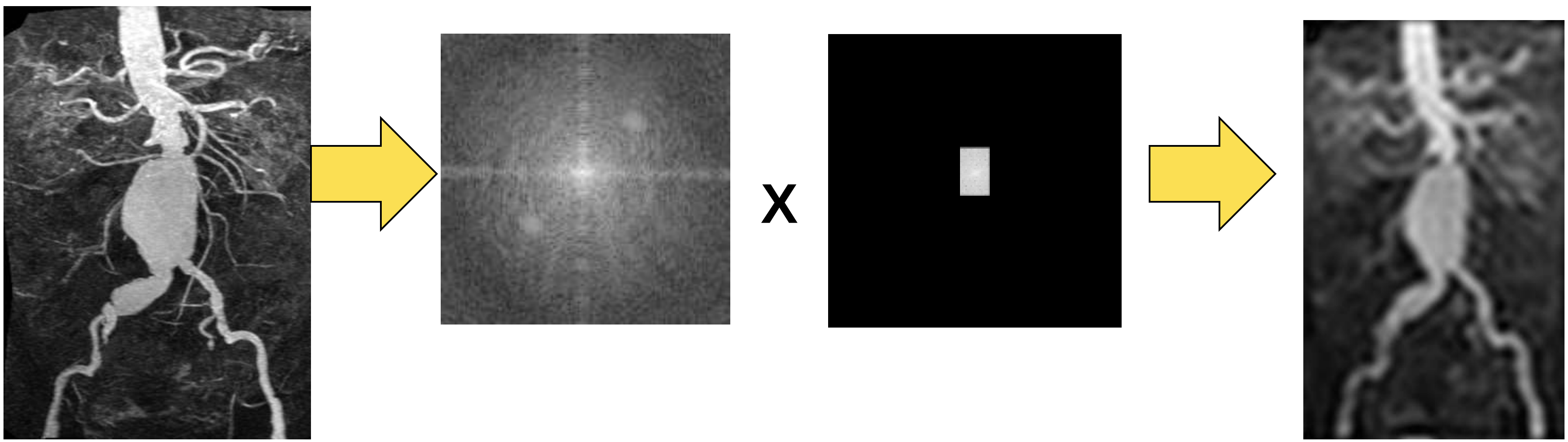


5 %
10 %
20 %
50 %

Fourier Spectrum Demo

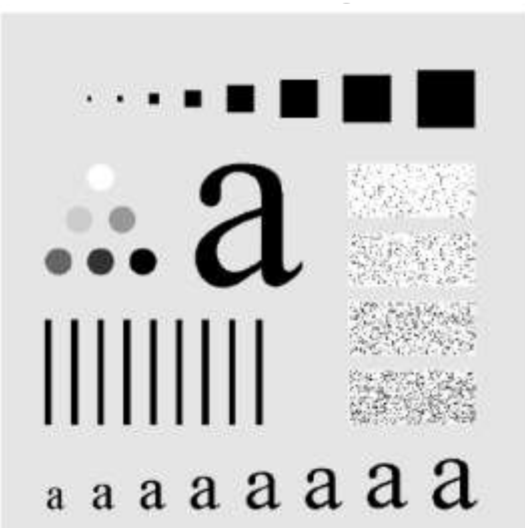
http://bigwww.epfl.ch/demo/basisfft/demo.html

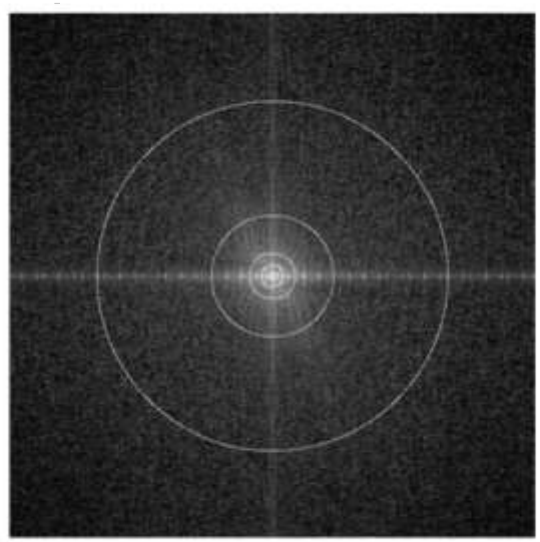
Filtering Using FT and Inverse

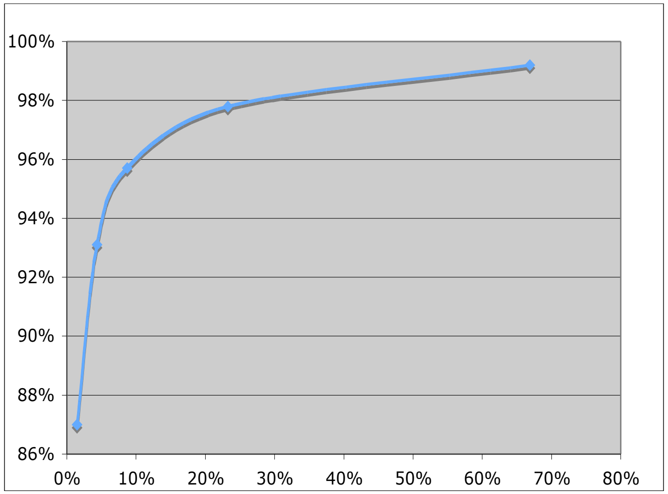


Low-Pass Filter

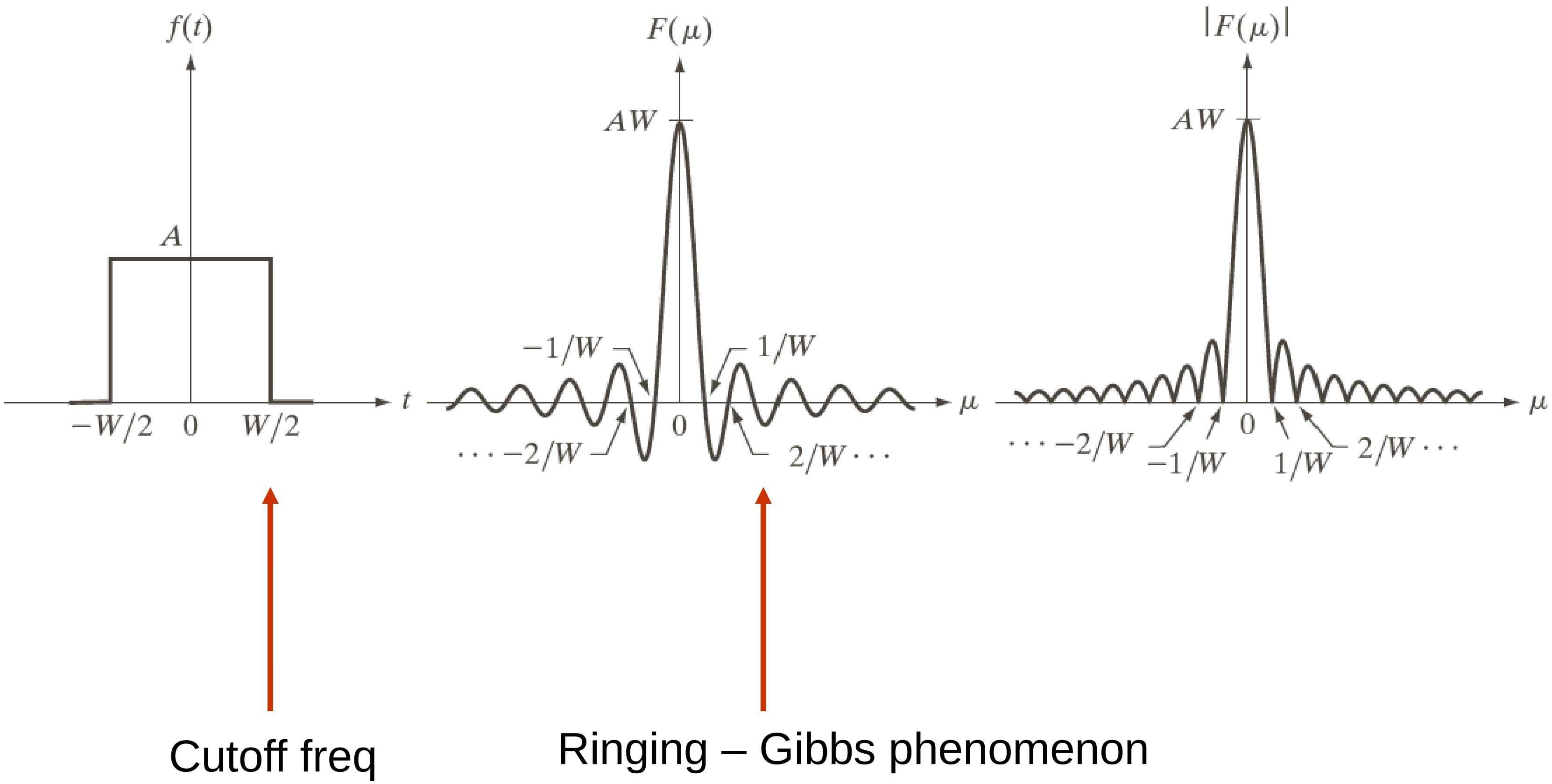
- Reduce/eliminate high frequencies
- Applications
  - Noise reduction
    - uncorrelated noise is broad band
    - Images have spectrum that focus on low





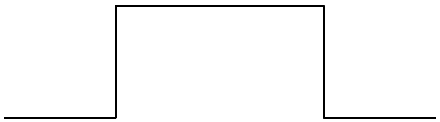


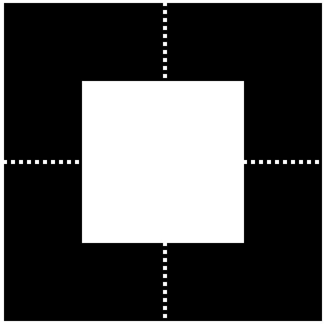
Ideal LP Filter – Box, Rect

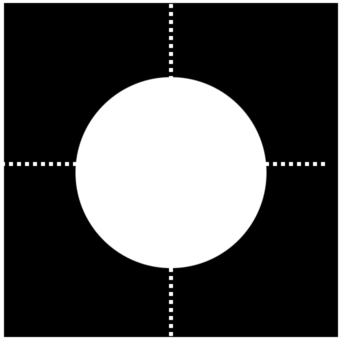


Extending Filters to 2D (or higher)

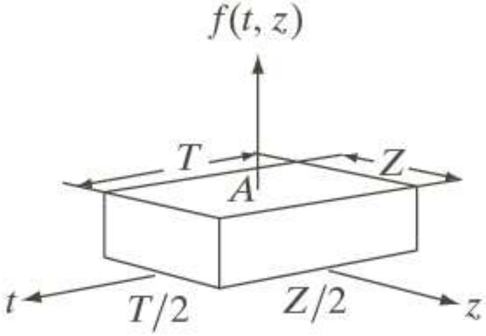
- Two options
  - Separable
    - H(s) -> H(u)H(v)
    - Easy, analysis
  - Rotate
    - $H(s) \rightarrow H((u^2 + v^2)^{1/2})$
    - Rotationally invariant

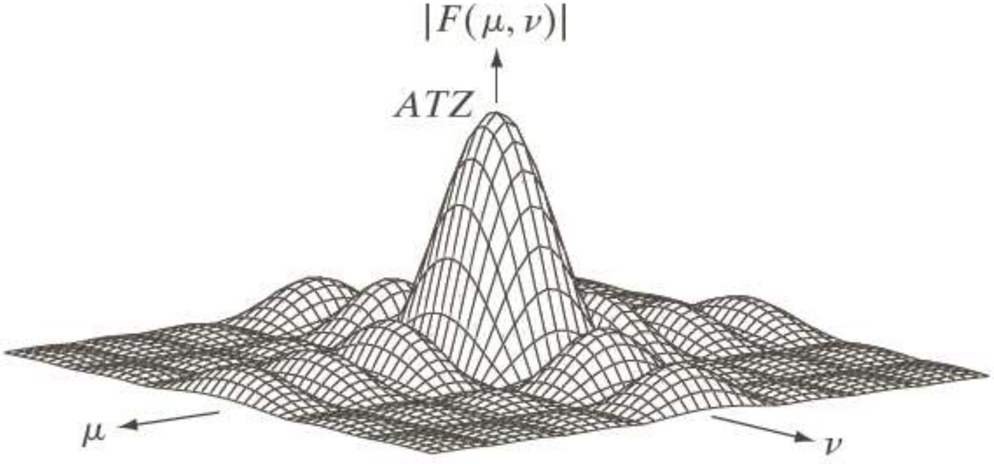




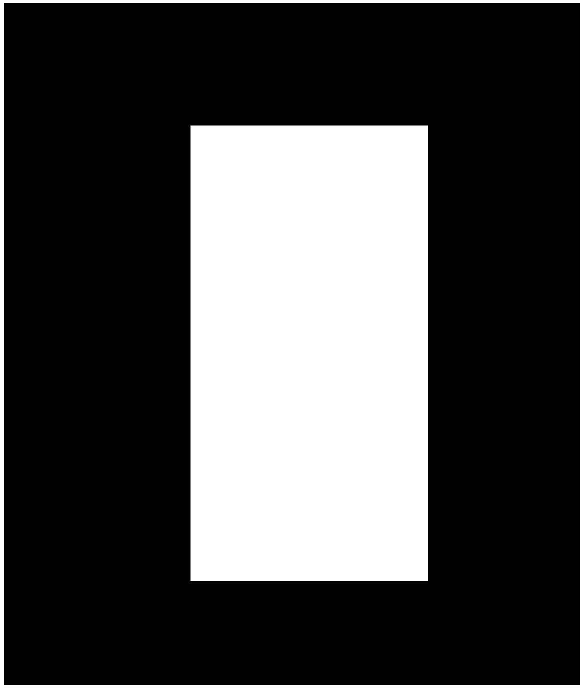


Ideal LP Filter – Box, Rect

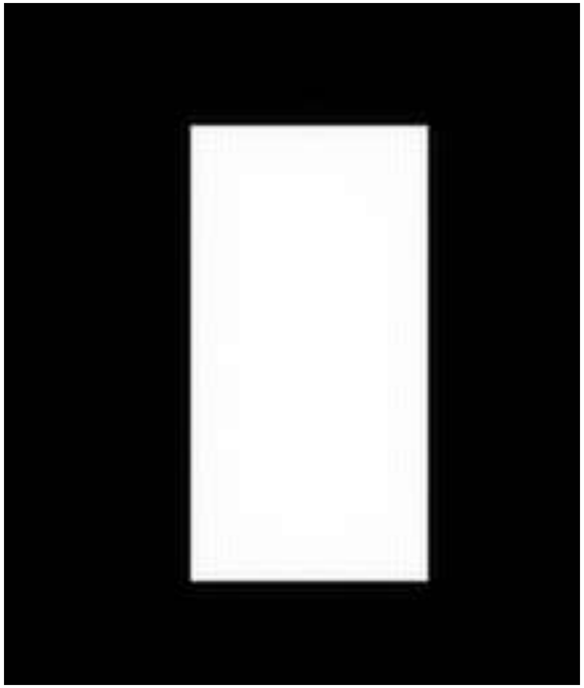




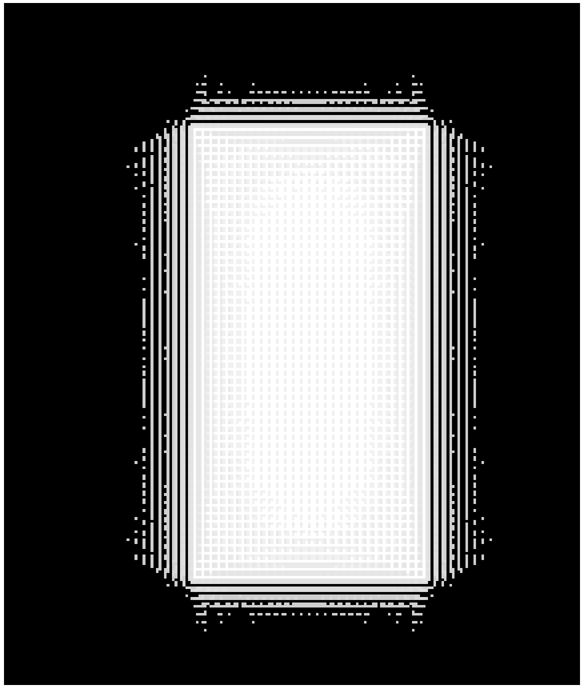
Ideal Low-Pass
Rectangle With Cutoff of 2/3



Image



Filtered



Filtered
+
Histogram Equalized

Ideal LP – 1/3

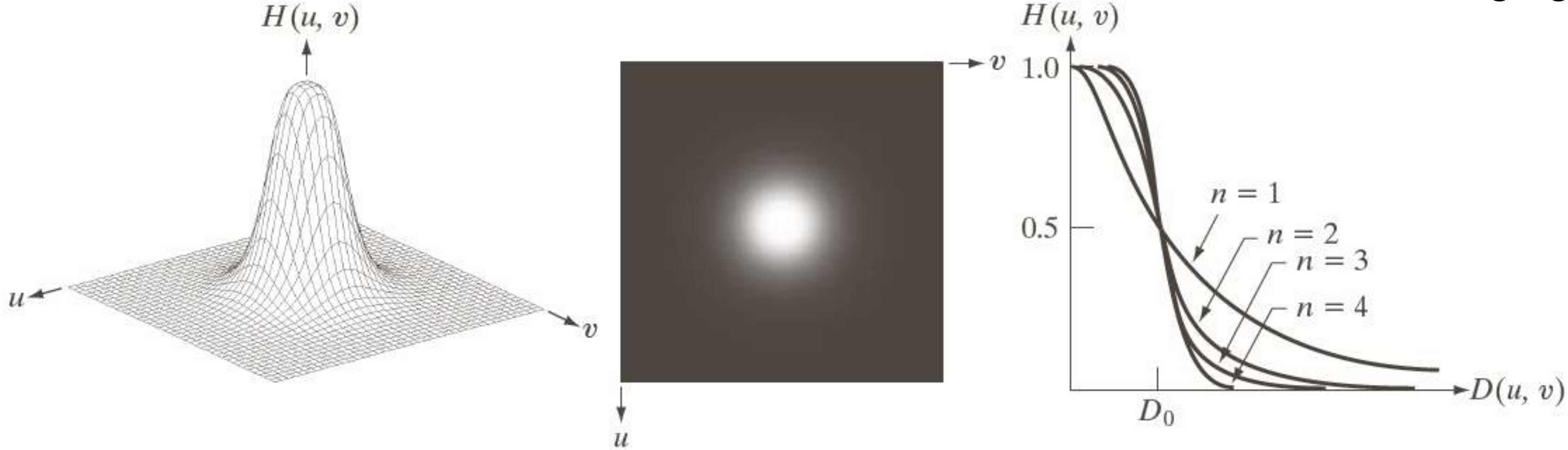
Ideal LP – 2/3

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

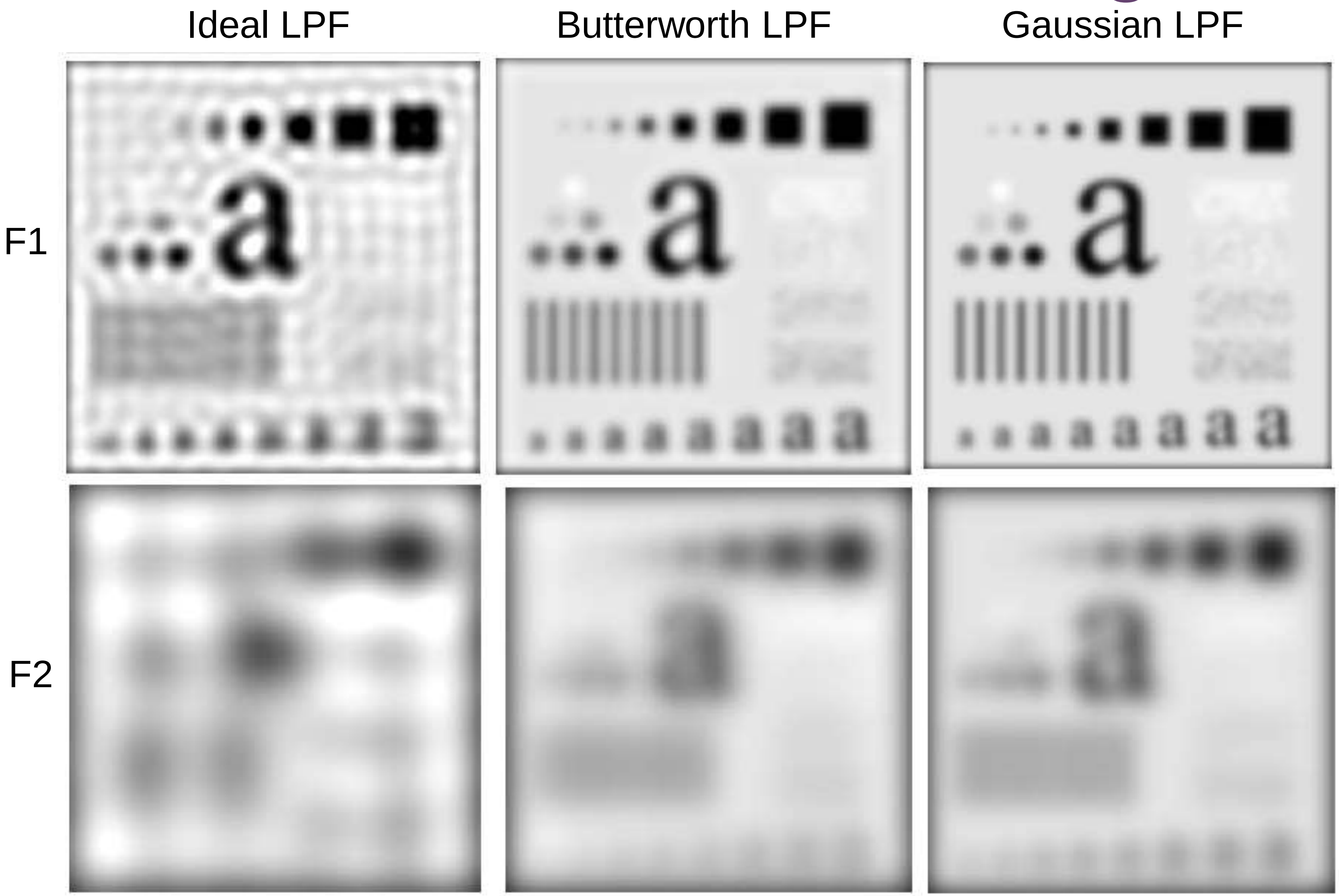


Butterworth - 1/3

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

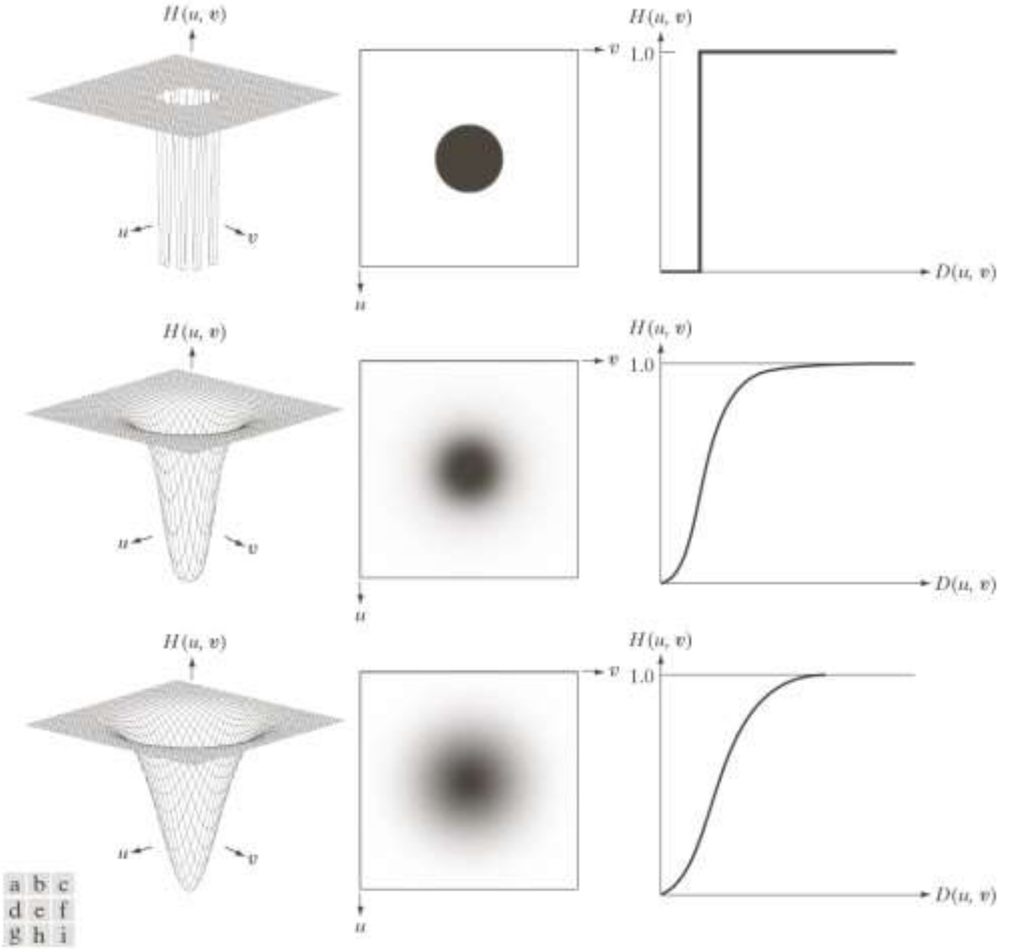
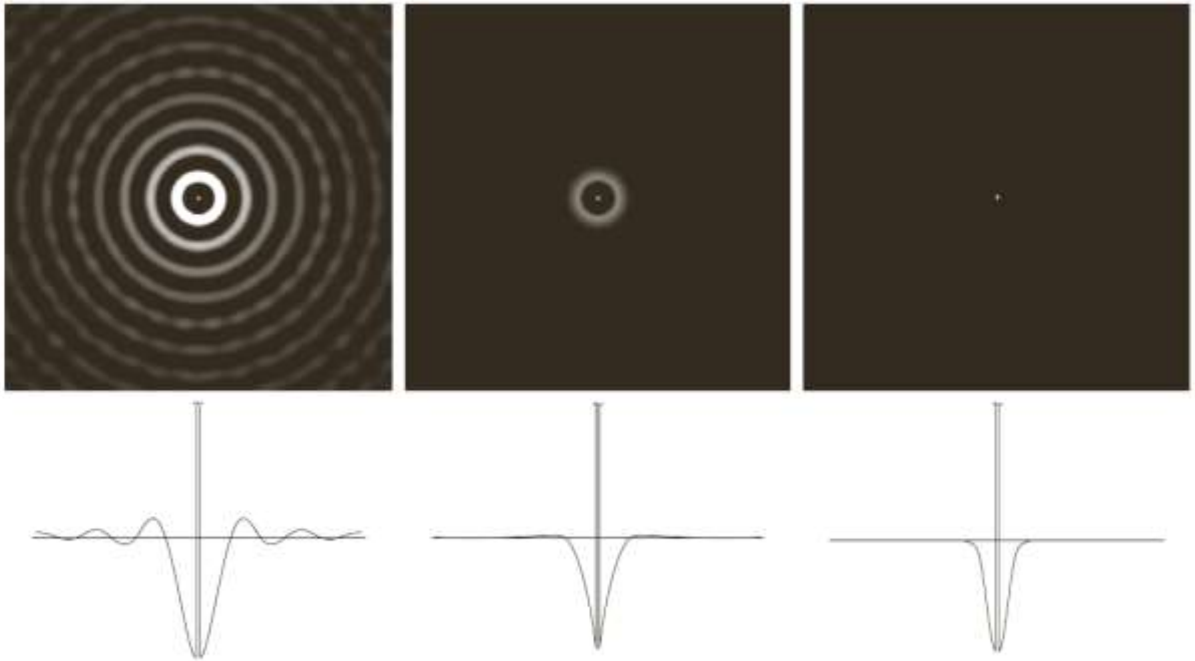


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

High-Pass Filters in Spatial Domain



a b c

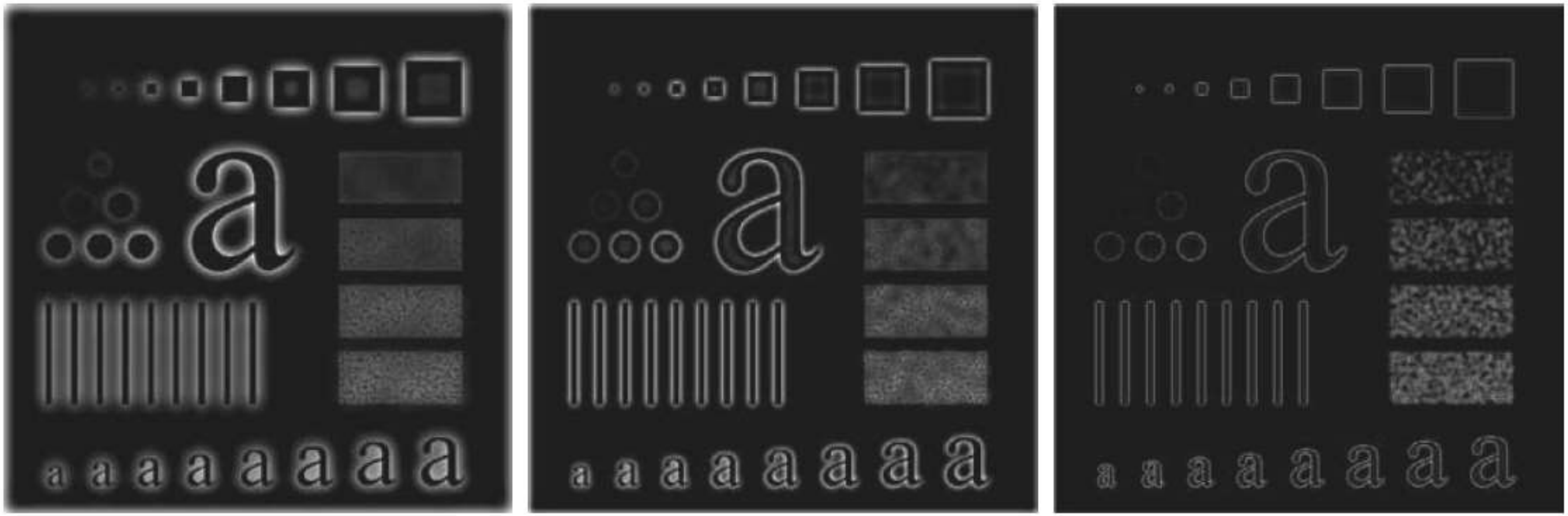
FIGURE 4.53 Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

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.

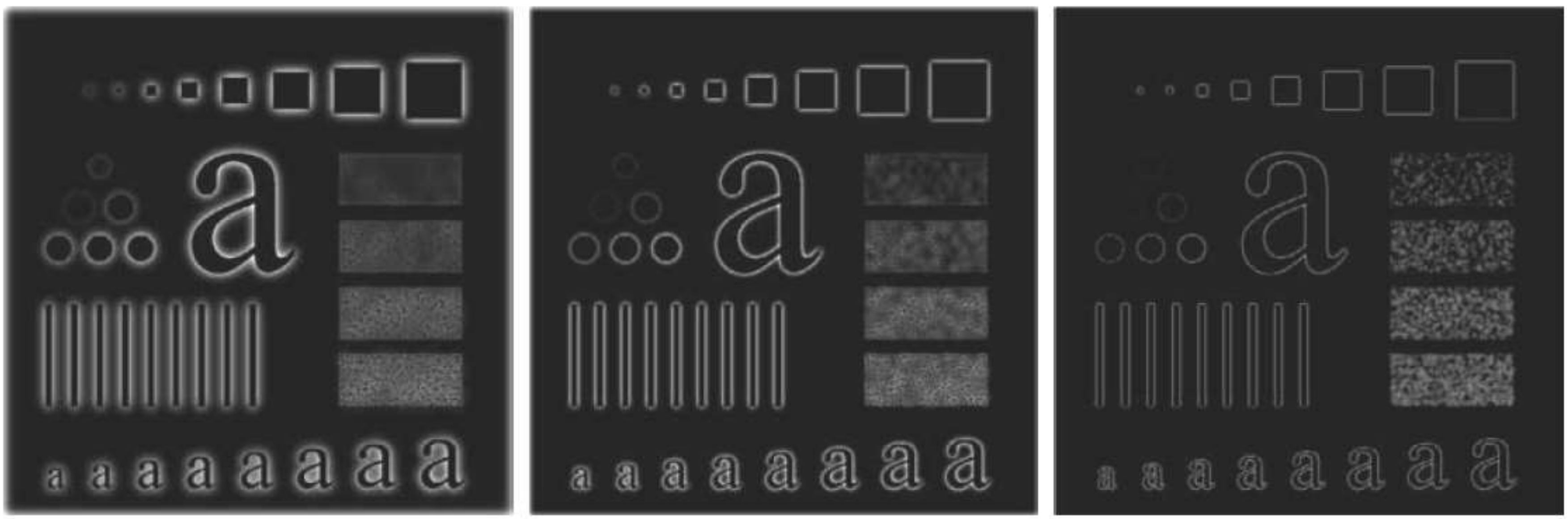
BHPF



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.

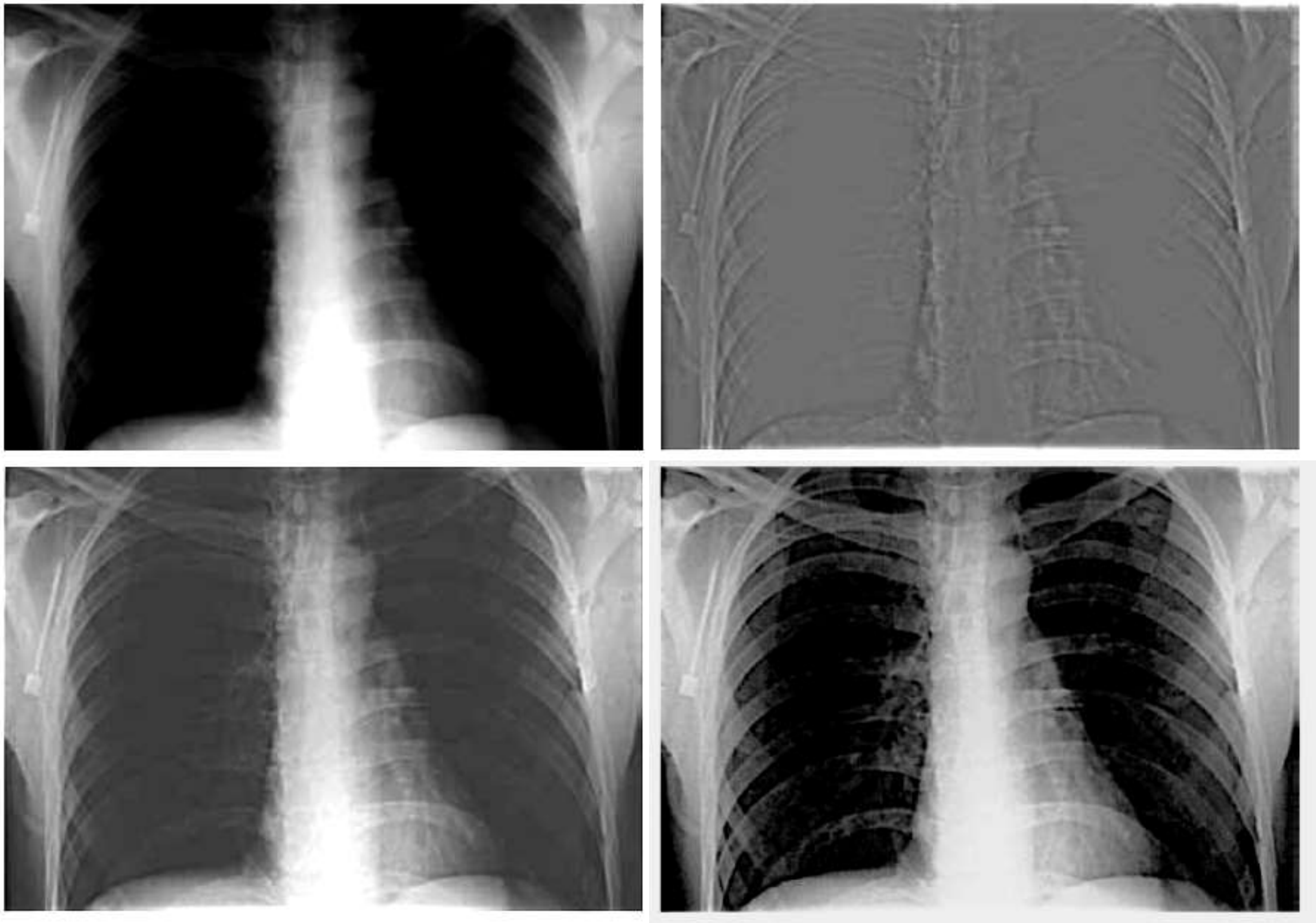
GHPF



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



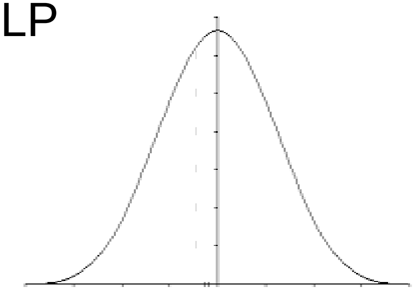
High Boost with GLPF

High-Boost Filtering

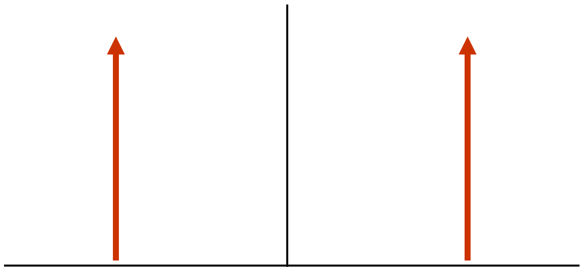
Band-Pass Filters

- Shift LP filter in Fourier domain by convolution with delta

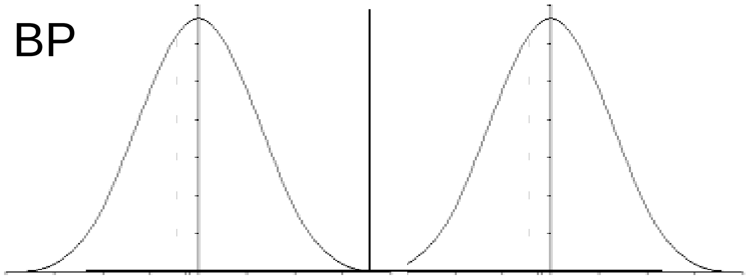
LP



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



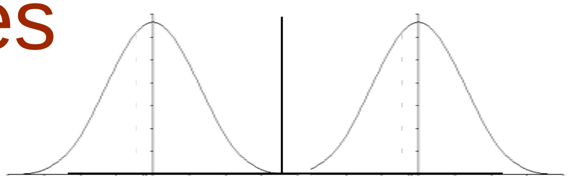
BP

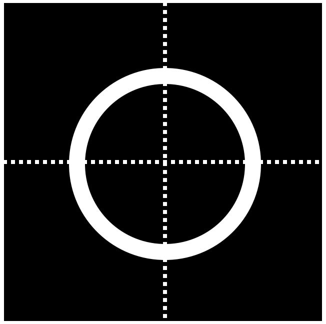


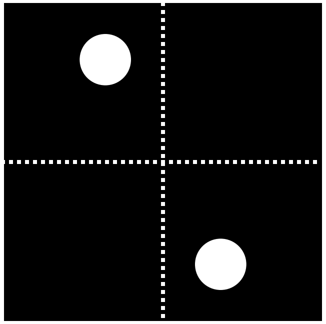
Typically 2-3 parameters
- -Width
- -Slope
- -Band value

Band Pass - Two Dimensions

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

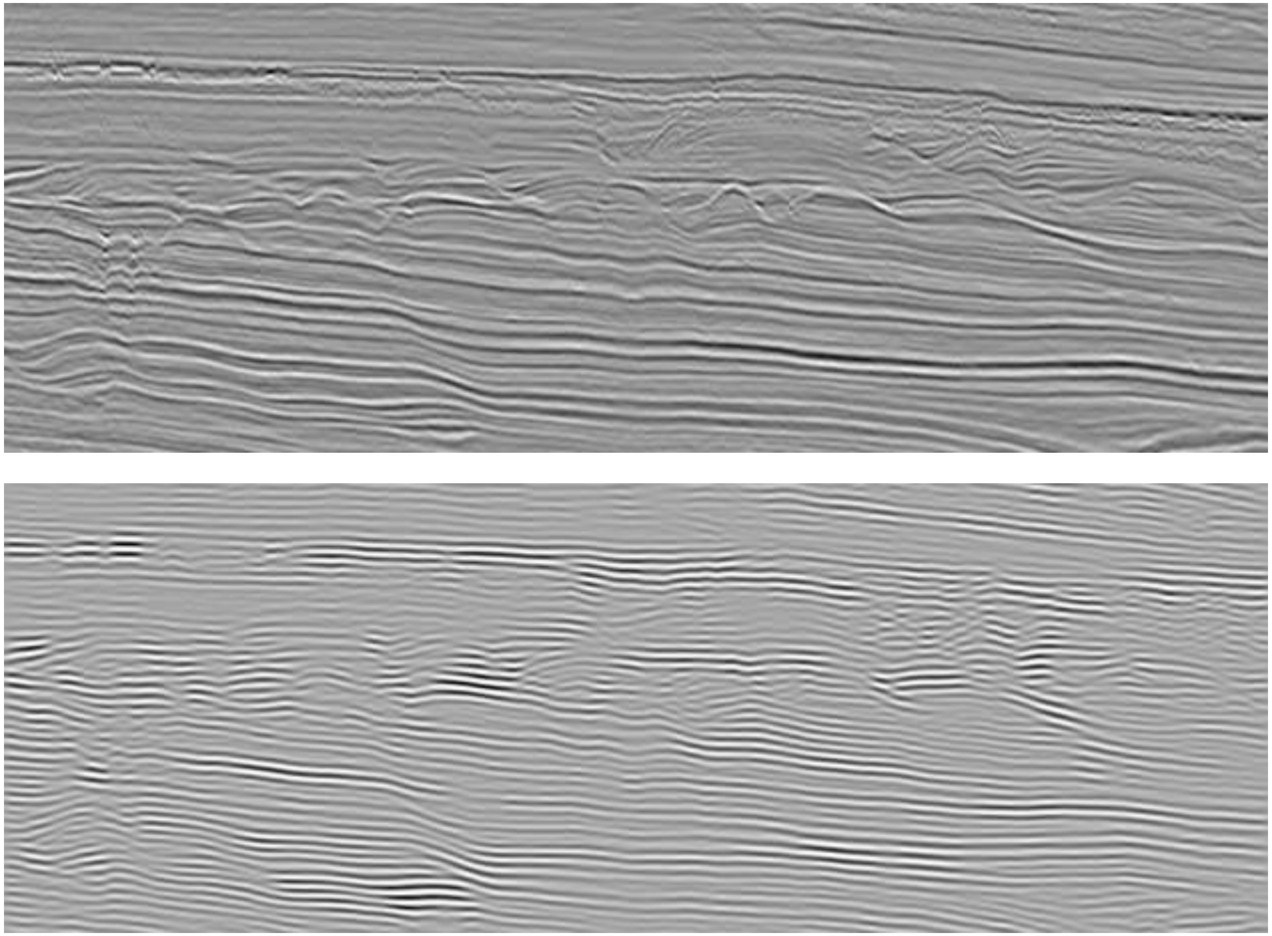




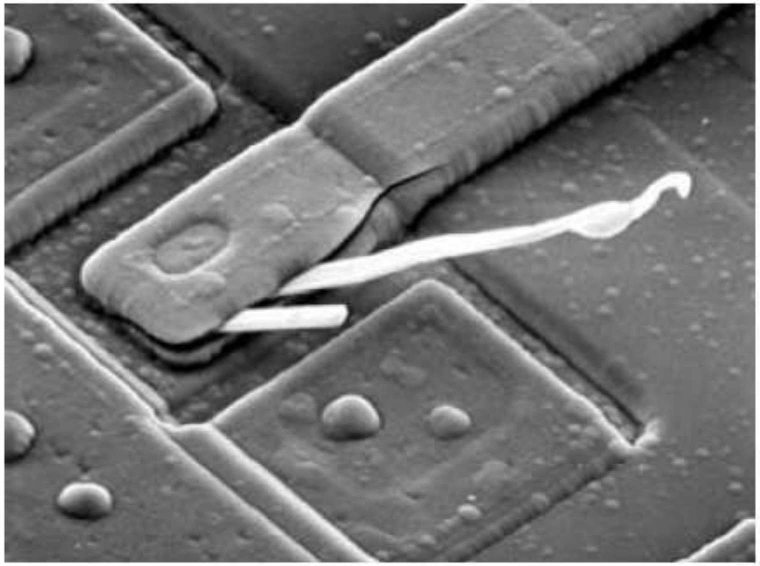


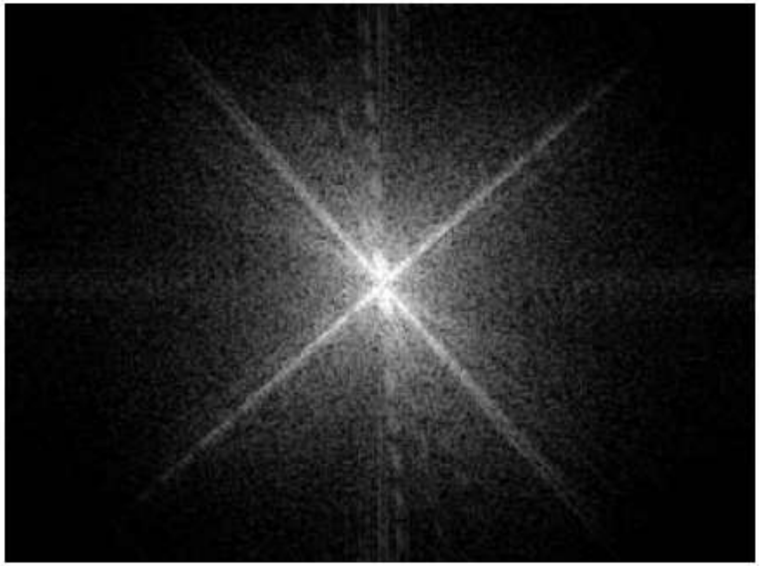
- Note:
  - Convolution with delta-pair in FD is multiplication with cosine in spatial domain

Band Bass Filtering

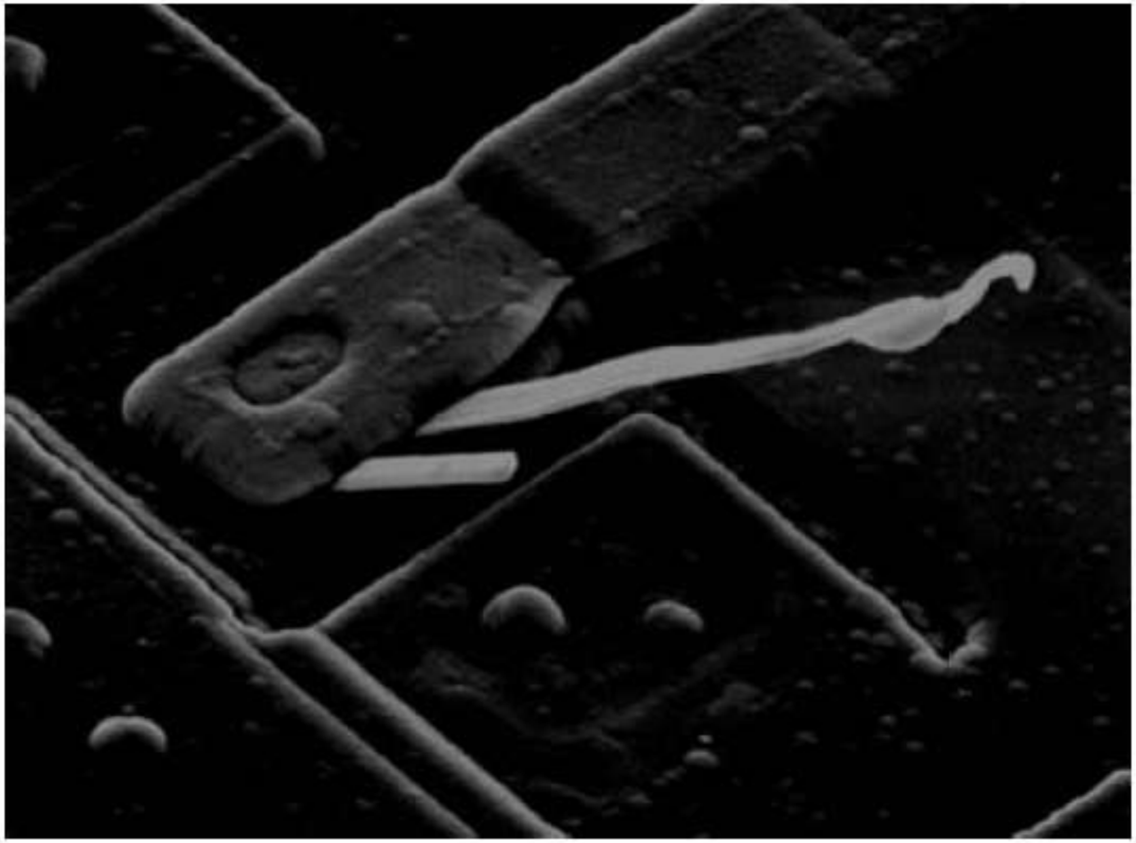


SEM Image and Spectrum



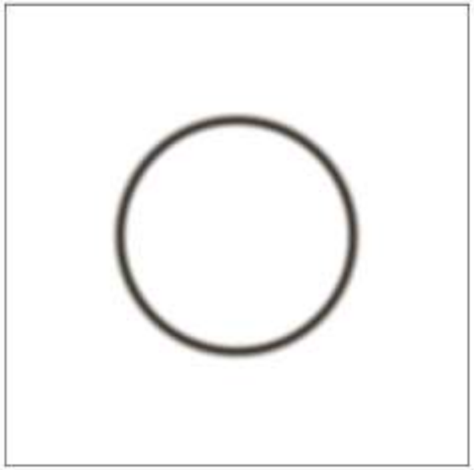


Band-Pass Filter

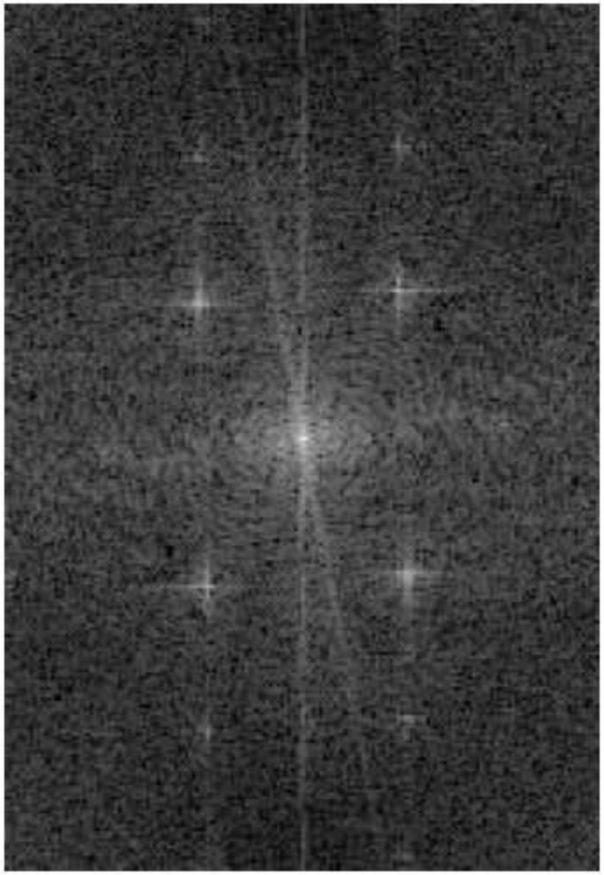


Radial Band Pass/Reject

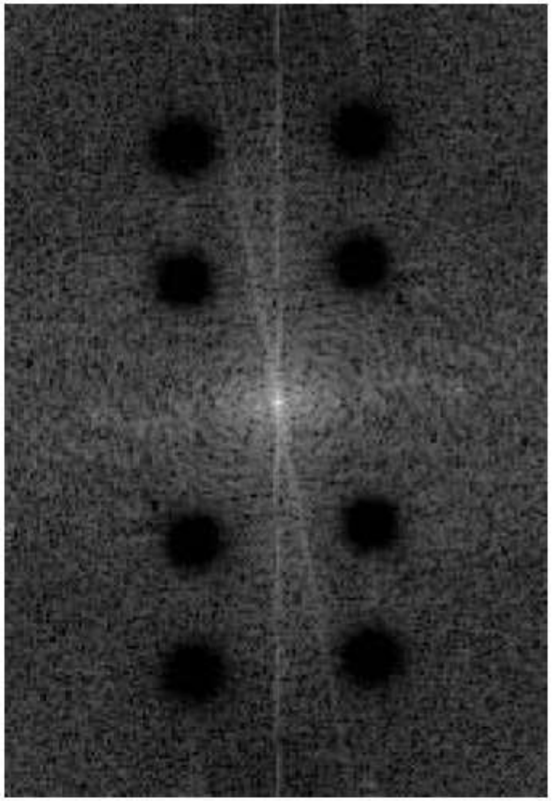
Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \leq D \leq D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u,v) = \dfrac{1}{1 + \left[\dfrac{DW}{D^2 - D_0^2}\right]^{2n}}$	$H(u,v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$



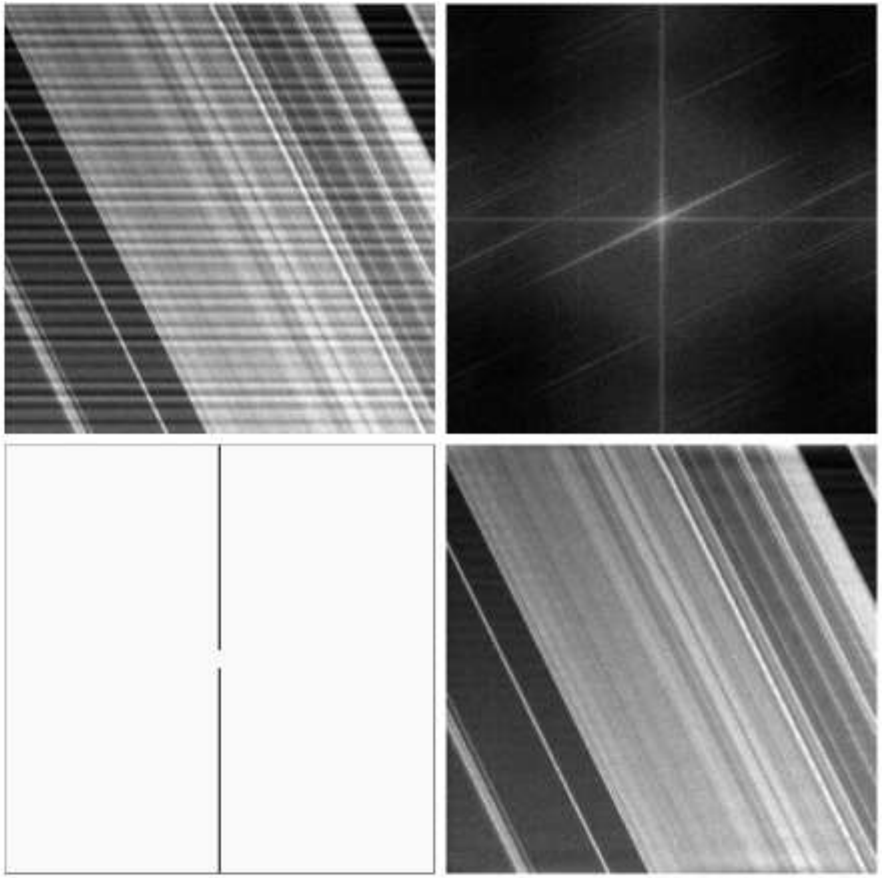

Band Reject Filtering



Band Reject Filtering



Band Reject Filtering



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 $W_N^{2rk} W_N^k = W_N^k W_{N/2}^{rk}$

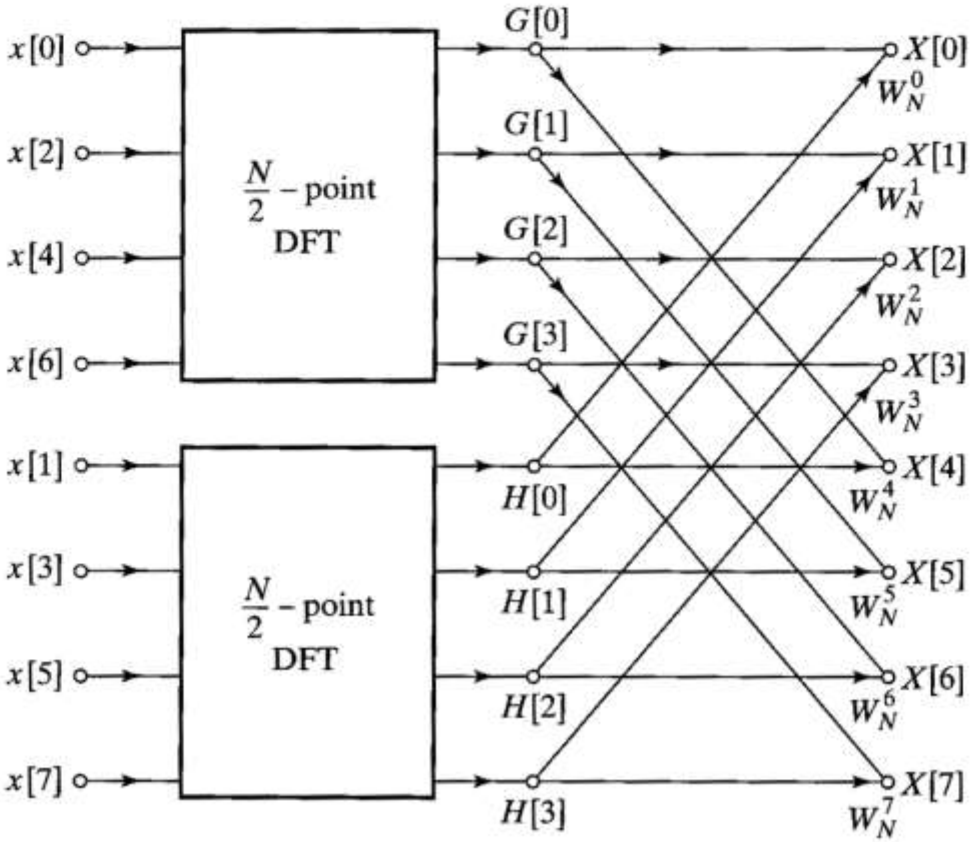
So …

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

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

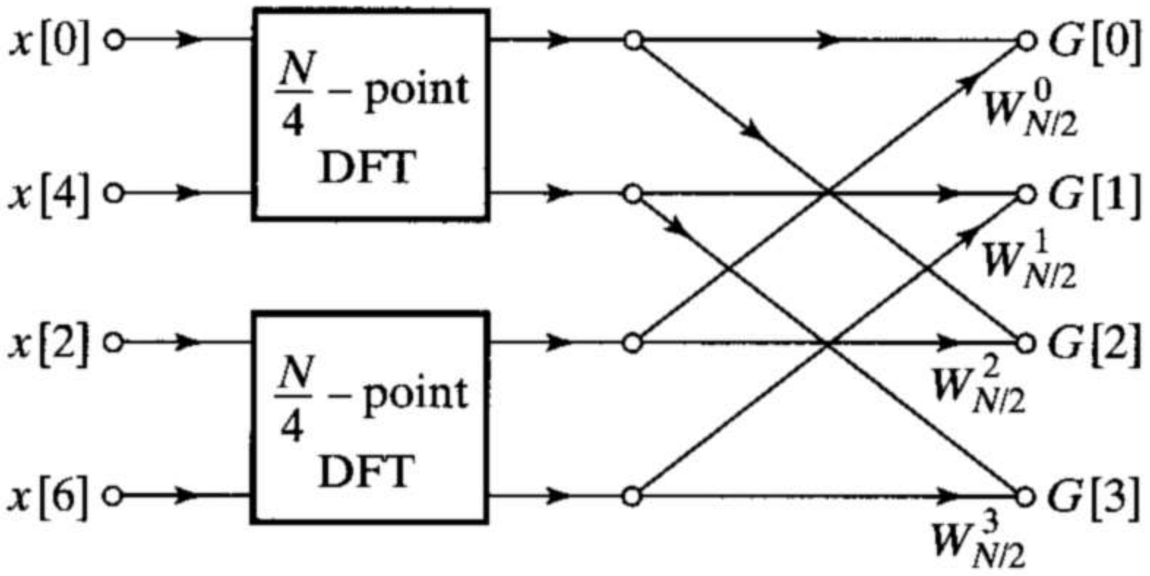
Example: N=8

- Divide and reuse



Example: N=8, Upper Part

- Continue to divide and reuse



Two-Point FFT

- The expression for the 2-point DFT is:

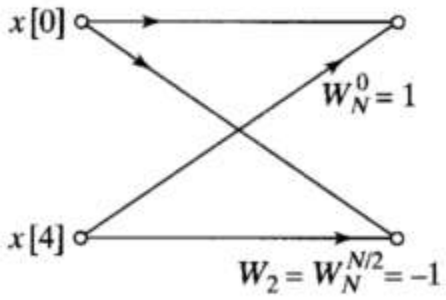
$$X[k] = \sum_{n=0}^{1} x[n] W_2^{nk} = \sum_{n=0}^{1} x[n] e^{-j2\pi nk/2}$$

- 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/