Computer Graphics

- Anti-Aliasing & Super-Sampling -

Hendrik Lensch

Organization

- **Registration HISPOS:**
  - deadline: **December 1st**
- **Final Exam**
  - date change: **February 19th**
- **Rendering Competition**
  - final deadline: **January 31st**
  - pre-deadline: January 24th

Overview

- **Last time**
  - Filtering
  - Signal processing
- **Today**
  - Sampling
  - Anti-aliasing & supersampling
- **Next lecture**
  - The Human Visual System

Discrete Fourier Transform

- **Equally-spaced function samples**
  - Function values known only at discrete points
    - Physical measurements
    - Pixel positions in an image !
- **Fourier Analysis**

$$\mathbf{a_k = 1/ N \;\; \sum_i \sin(2\pi \; k \; i \; / \; N) \;\; f_i \; , \qquad b_k = 1/ N \;\; \sum_i \cos(2\pi \; k \; i \; / \; N) \;\; f_i}$$

  - yields amplitude and phase per frequency
  - Sum over all measurement points N
  - k=0,1,2, …, ? Highest possible frequency ?

  $\Rightarrow$ **Nyquist frequency**
    - Sampling rate N_i
    - 2 samples / period $\Leftrightarrow$ 0.5 cycles per pixel

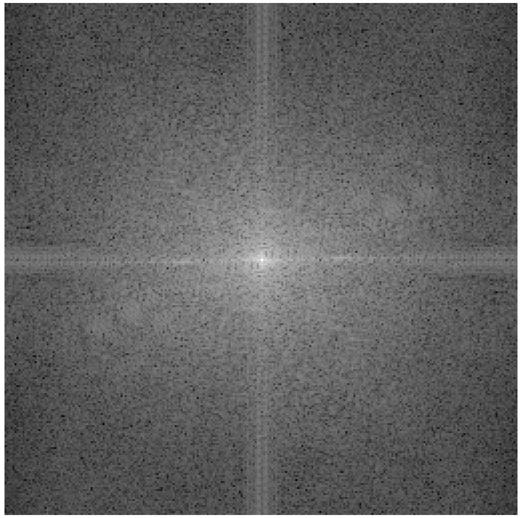
    $\Rightarrow k \leq N / 2$

An Exam

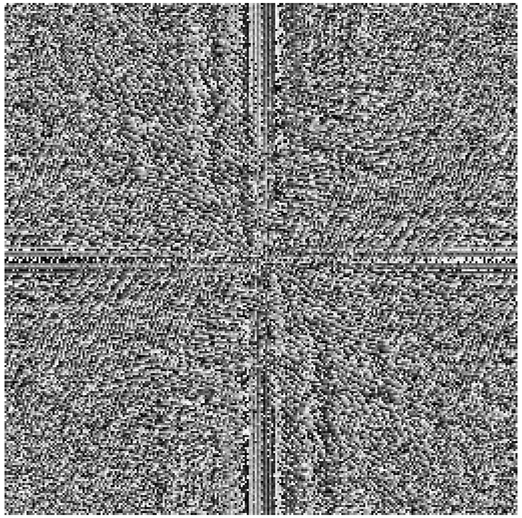
Fourier transformed

reconstructed

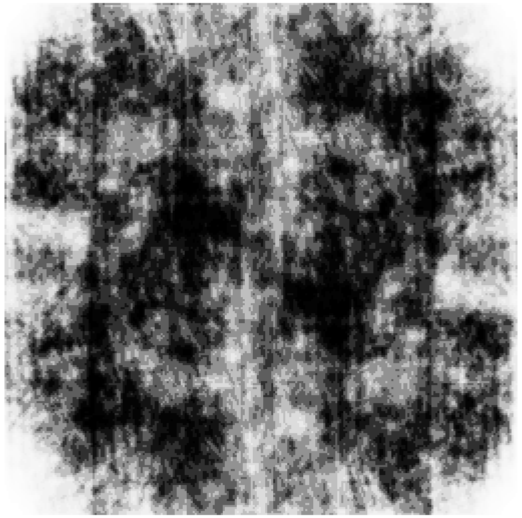
f(x)



Amplitude



Phase



ignoring Phase

using Phase+Amplitude

Spatial vs. Frequency Domain

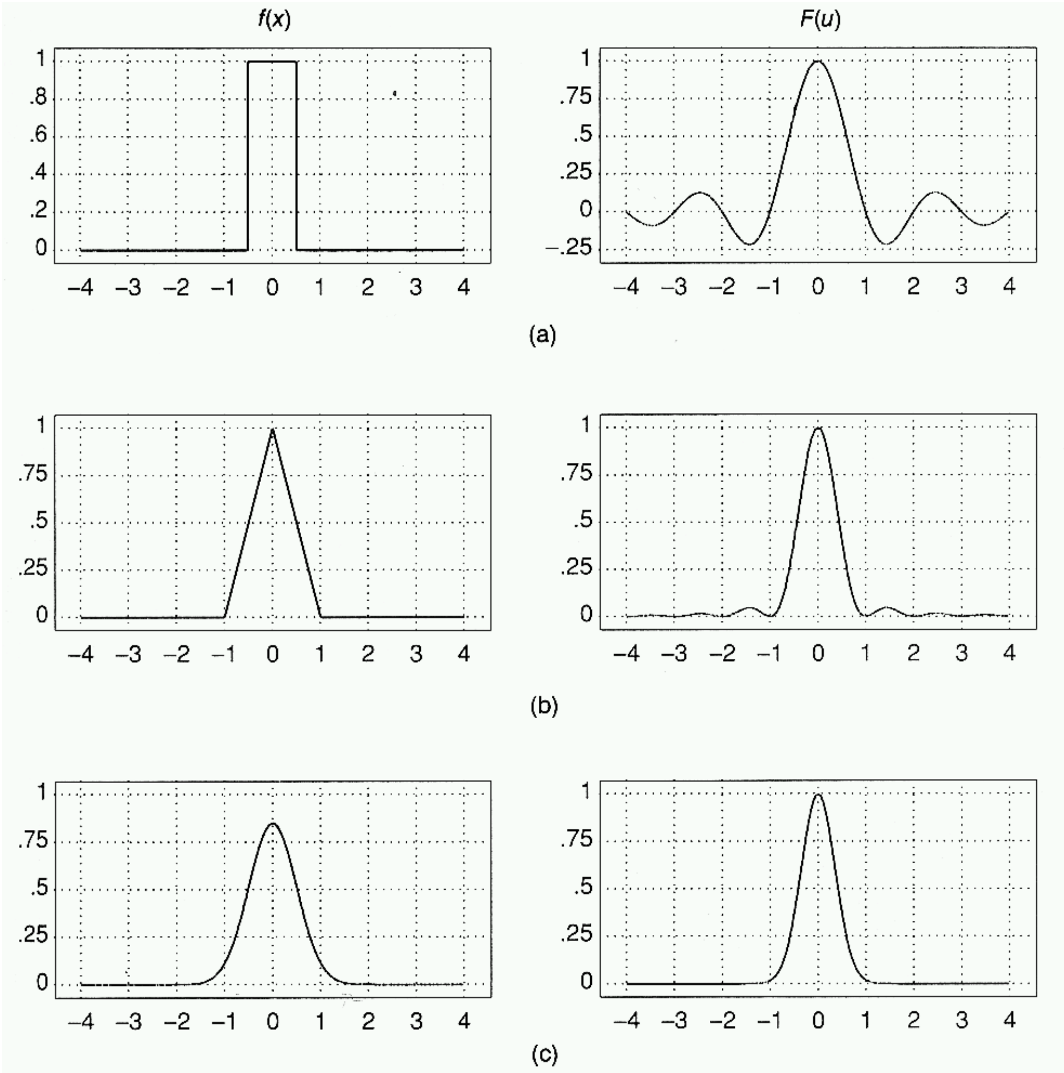
- **Important basis functions**
  - Box ←→ sinc

$$\mathrm{sinc}(x) = \frac{\sin(x\pi)}{x\pi}$$

$$\mathrm{sinc}(0) = 1$$

$$\int \mathrm{sinc}(x)\,dx = 1$$

    - Wide box → small sinc
    - Negative values
    - Infinite support
  - Triangle ←→ sinc²
  - Gauss ←→ Gauss

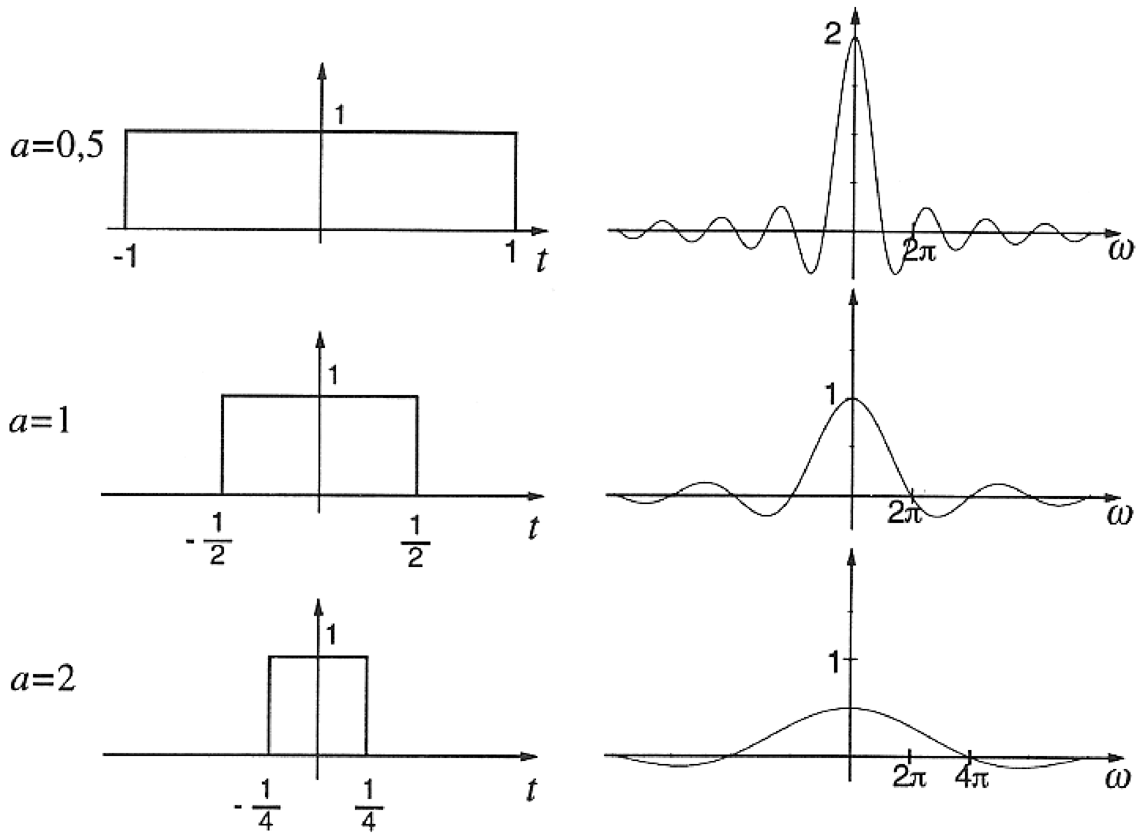


Spatial vs. Frequency Domain

- **Transform behavior**
- **Example: box function**

$$\mathrm{rect}(at) \quad \circ\!\!-\!\!\bullet \quad \frac{1}{|a|}\,\mathrm{si}\left(\frac{\omega}{2a}\right)$$

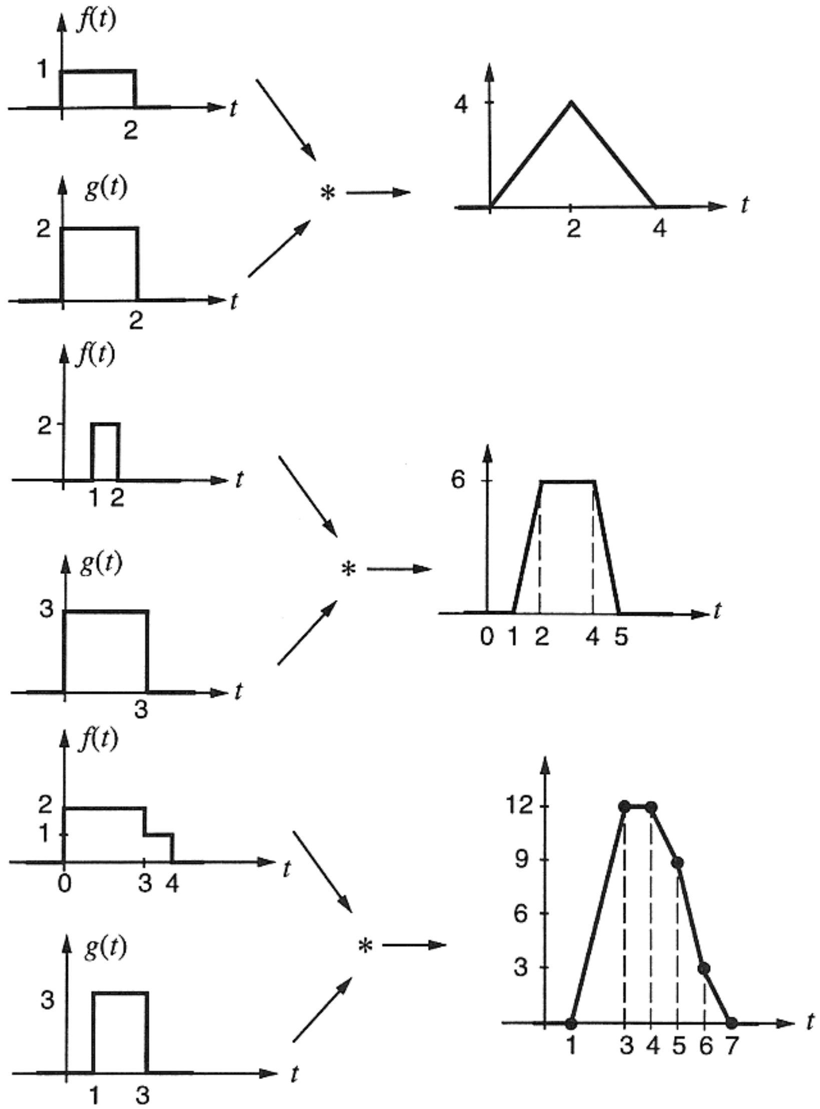
  - Fourier transform: sinc
  - Wide box: narrow sinc
  - Narrow box: wide sinc



Convolution

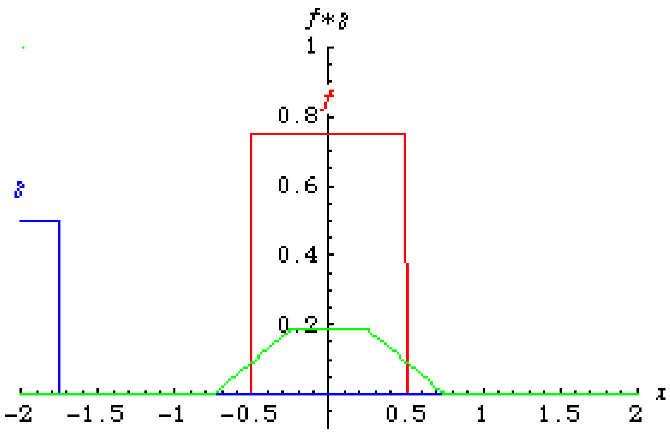
$$f(x) \otimes g(x) = \int_{-\infty}^{\infty} f(\tau) g(x-\tau) d\tau$$

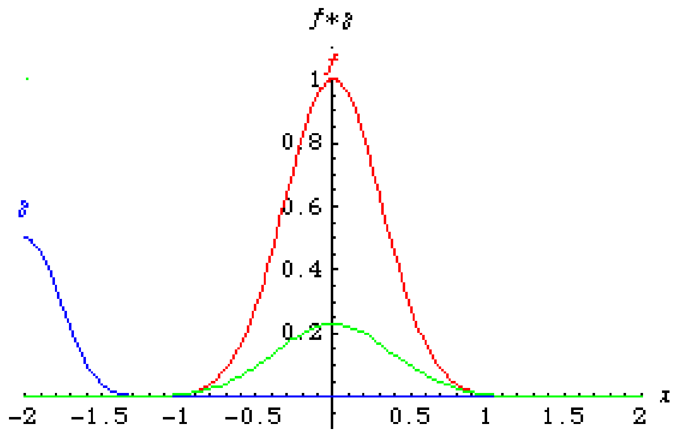
- **Two functions *f*, *g***
- **Shift one function against the other by *x***
- **Multiply function values**
- **Integrate overlapping region**
- **Numerical convolution: Expensive operation**
  - For each *x*: integrate over non-zero domain



Convolution

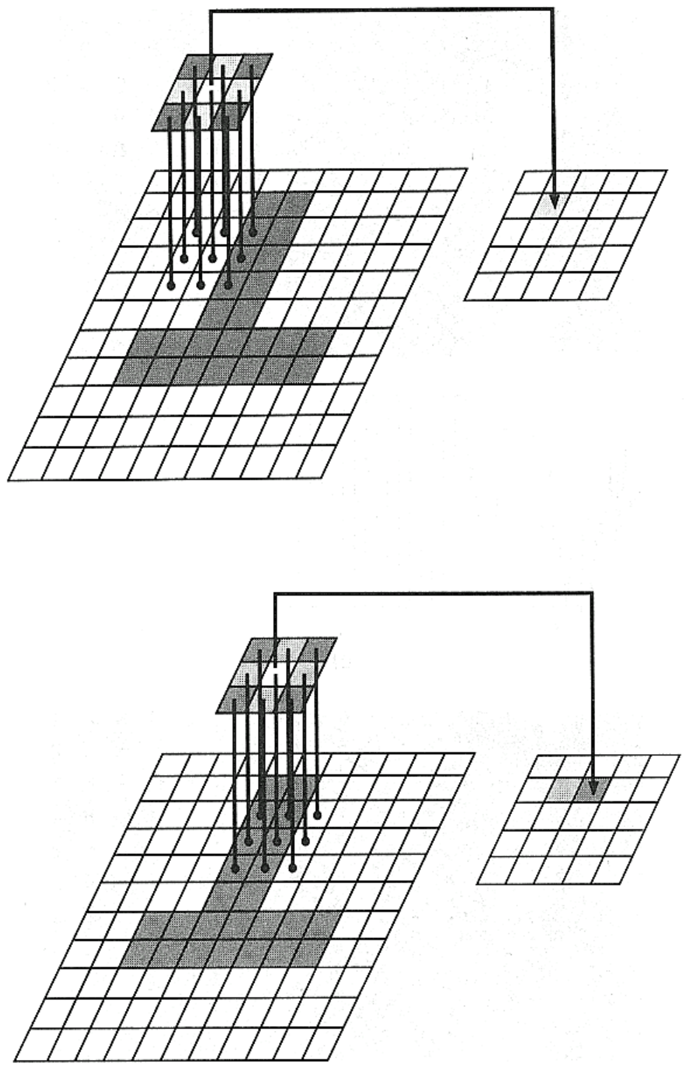
- **Examples**
  – Box functions
  – Gauss functions





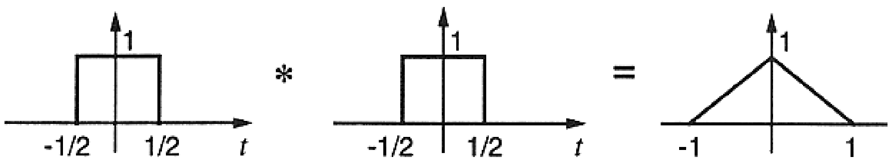
Convolution and Filtering

- **Technical Realization**
  - In image domain
  - Pixel mask with weights
  - OpenGL: Convolution extension
- **Problems (e.g. sinc)**
  - Large filter support
    - Large mask
    - A lot of computation
  - Negative weights
    - Negative light?



Convolution Theorem

- **Convolution in image domain:**
  multiplication in Fourier domain
- **Convolution in Fourier domain:**
  multiplication in image domain
  - Multiplication much cheaper than convolution !

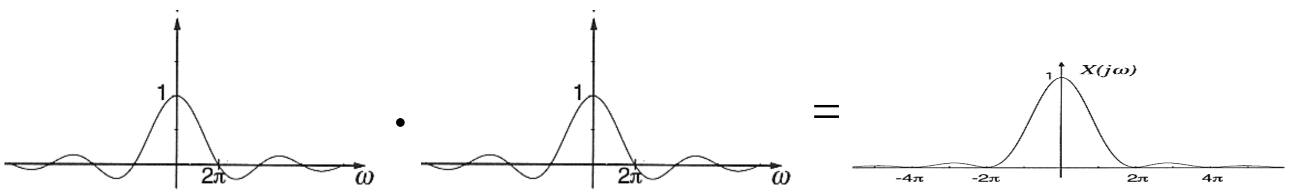


$$x(t) = \begin{cases} t+1 & \text{für } -1 \le t < 0 \\ -t+1 & \text{für } 0 \le t < 1 \\ 0 & \text{sonst} \end{cases}$$

$$\mathrm{rect}(t) * \mathrm{rect}(t) = x(t)$$

$$\circ\!\!-\!\!\bullet \qquad \circ\!\!-\!\!\bullet \qquad \circ\!\!-\!\!\bullet$$

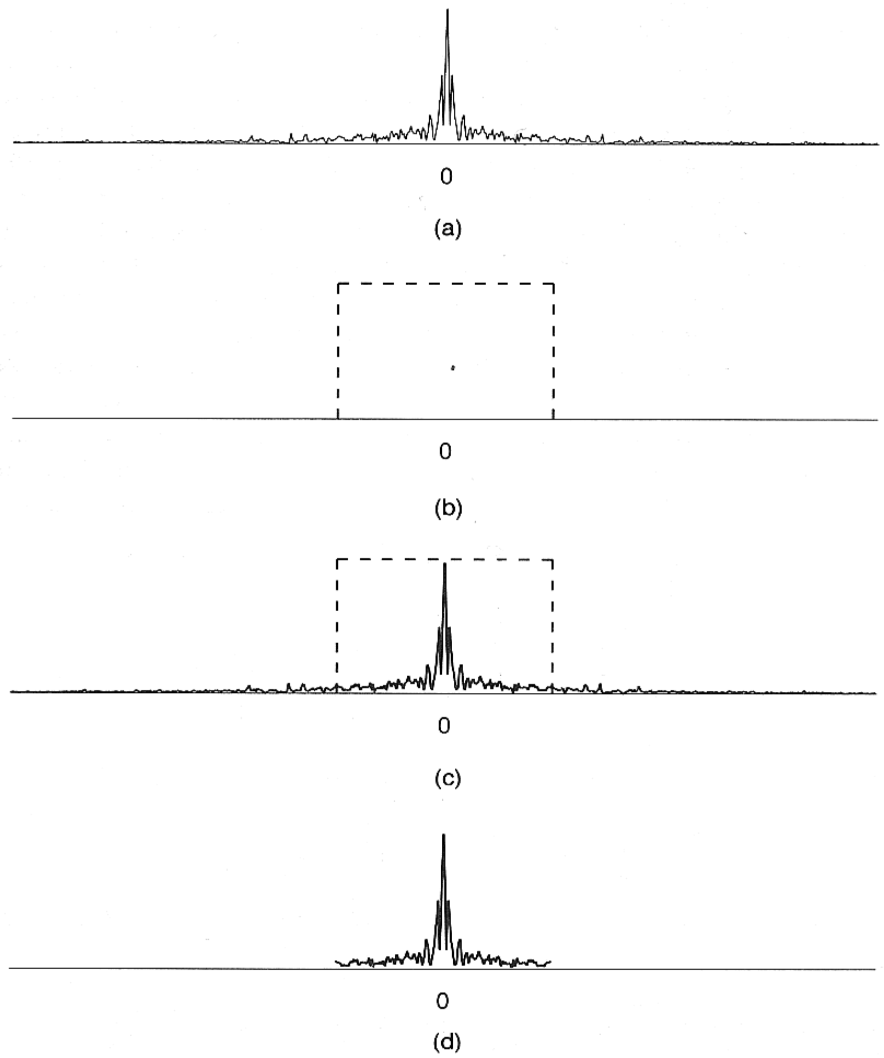
$$\mathrm{si}\left(\frac{\omega}{2}\right) \cdot \mathrm{si}\left(\frac{\omega}{2}\right) = X(j\omega)\,.$$



$$X(j\omega) = \mathrm{si}^2\left(\frac{\omega}{2}\right).$$

Filtering

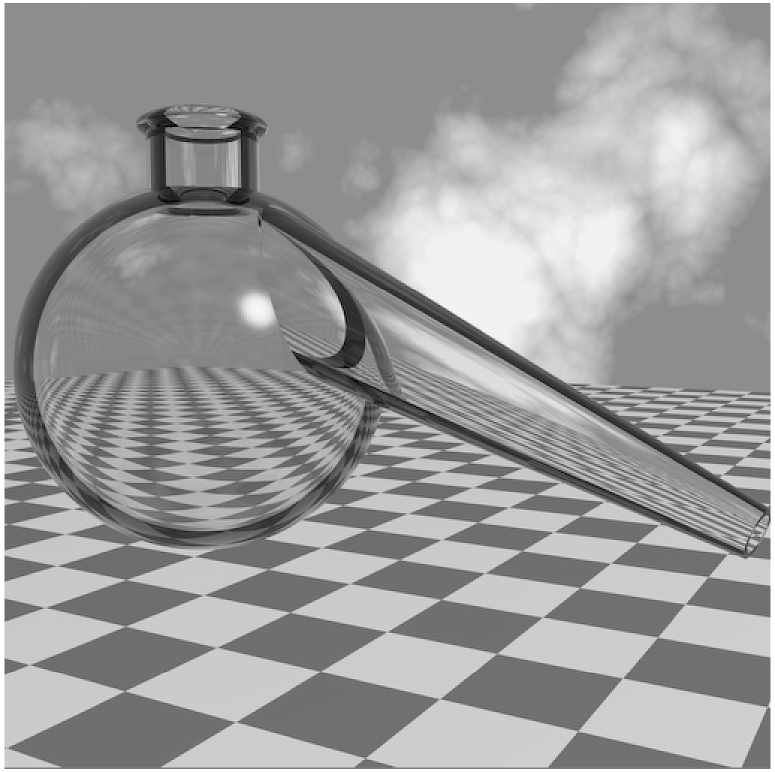
- **Low-pass filtering**
  - Convolution with sinc in spatial domain, or
  - Multiplication with box in frequency domain
- **High-pass filtering**
  - Only high frequencies
- **Band-pass filtering**
  - Only intermediate

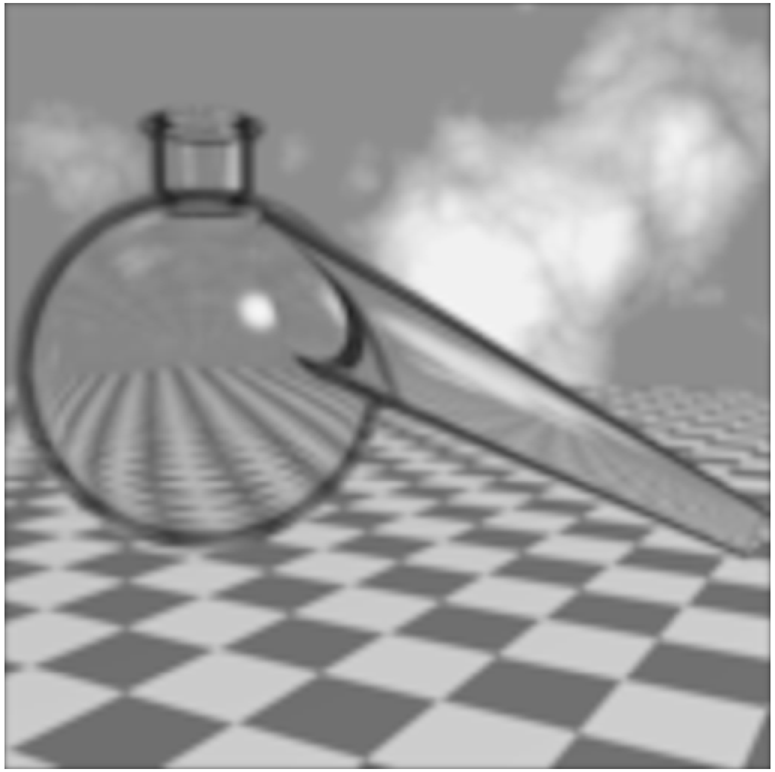


Low-pass filtering in frequency domain: multiplication with box

Low-Pass Filtering

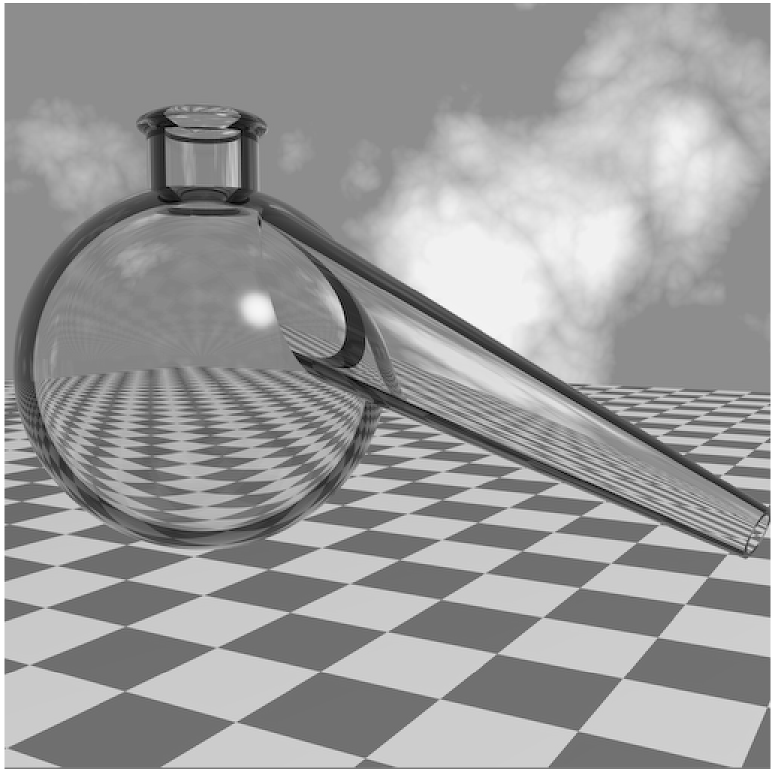
- **„Blurring“**

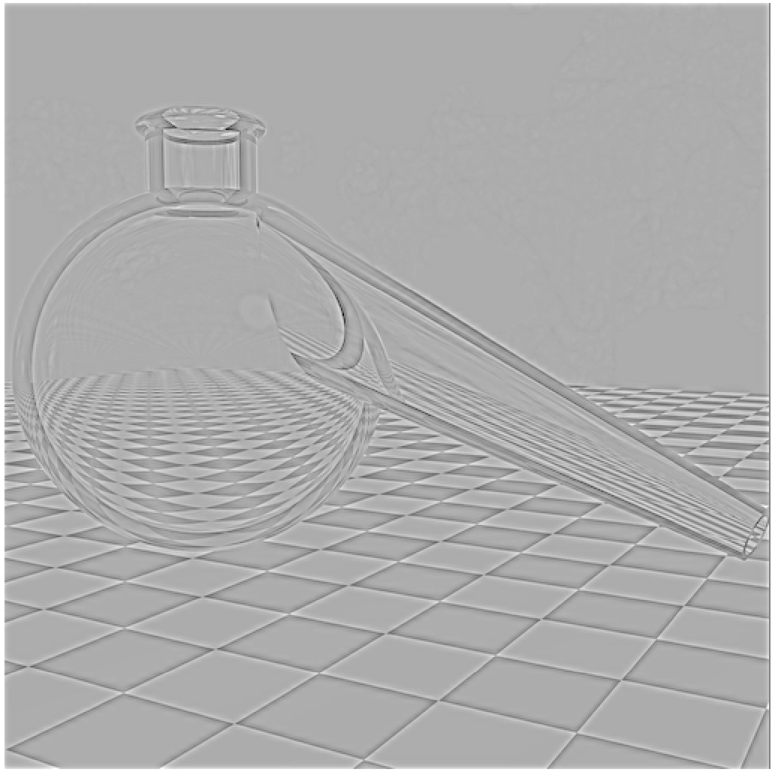




High-Pass Filtering

- **Enhances discontinuities in image**
  - Useful for edge detection

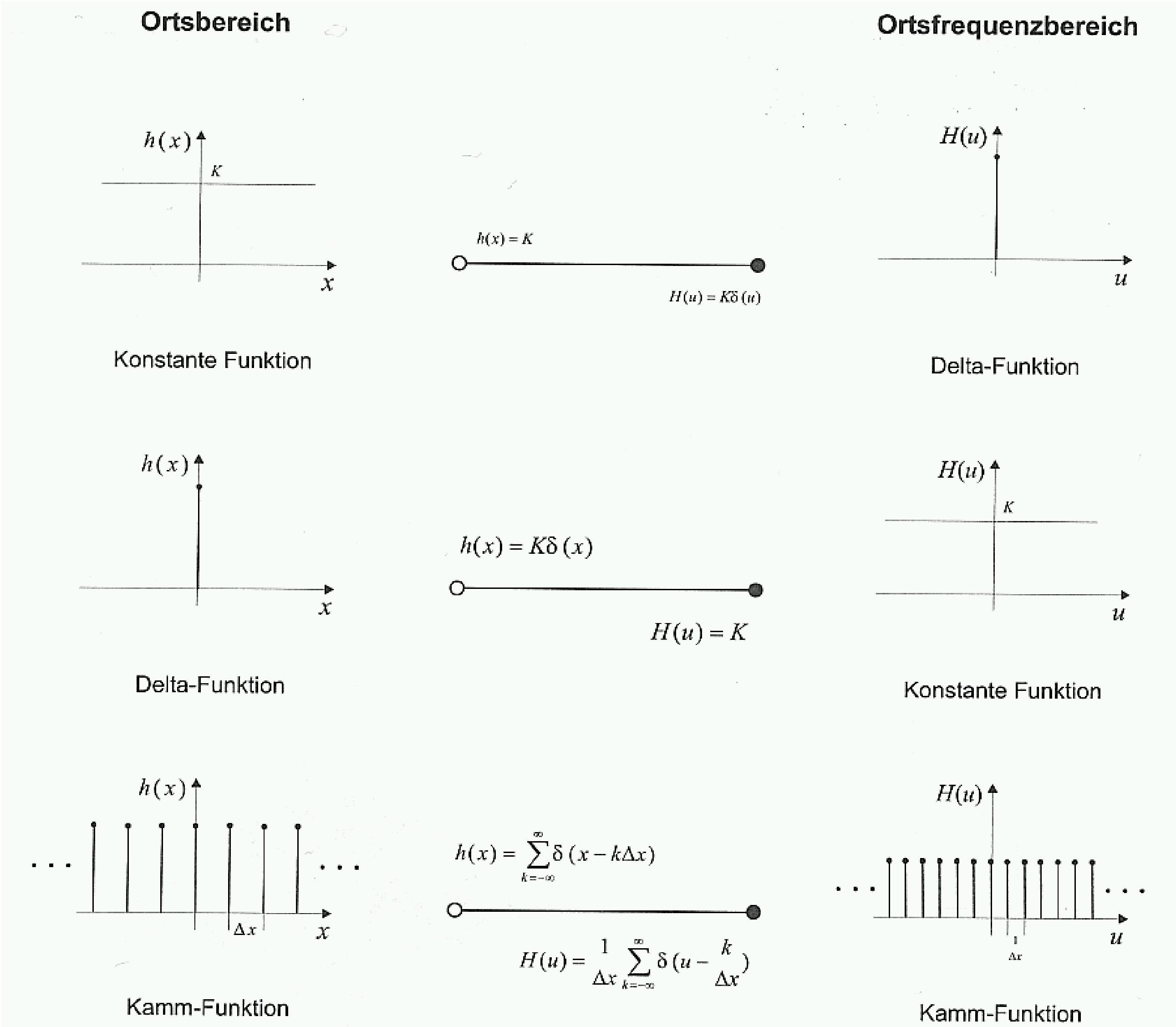




Sampling

Sampling

- **Constant & δ-Function**
  - flash
- **Comb/Shah function**



Sampling

- **Constant & δ-Function**
  - Duality

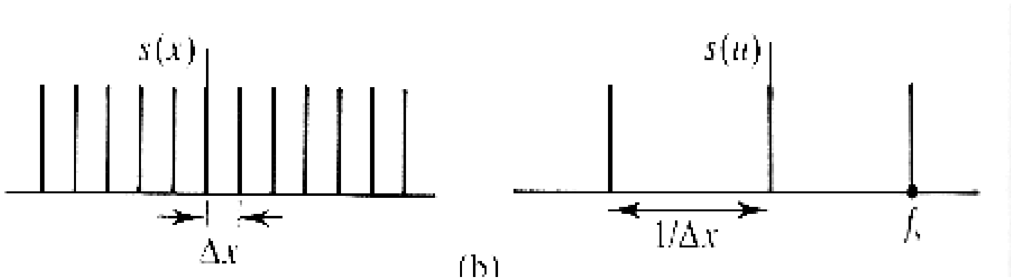
$$f(x) = K$$

$$F(\omega) = K\delta(\omega)$$

  - And vice versa
- **Comb function**
  - Duality: The dual of a comb function is again a comb function
    - Inverse wave length, amplitude scales with inverse wave length

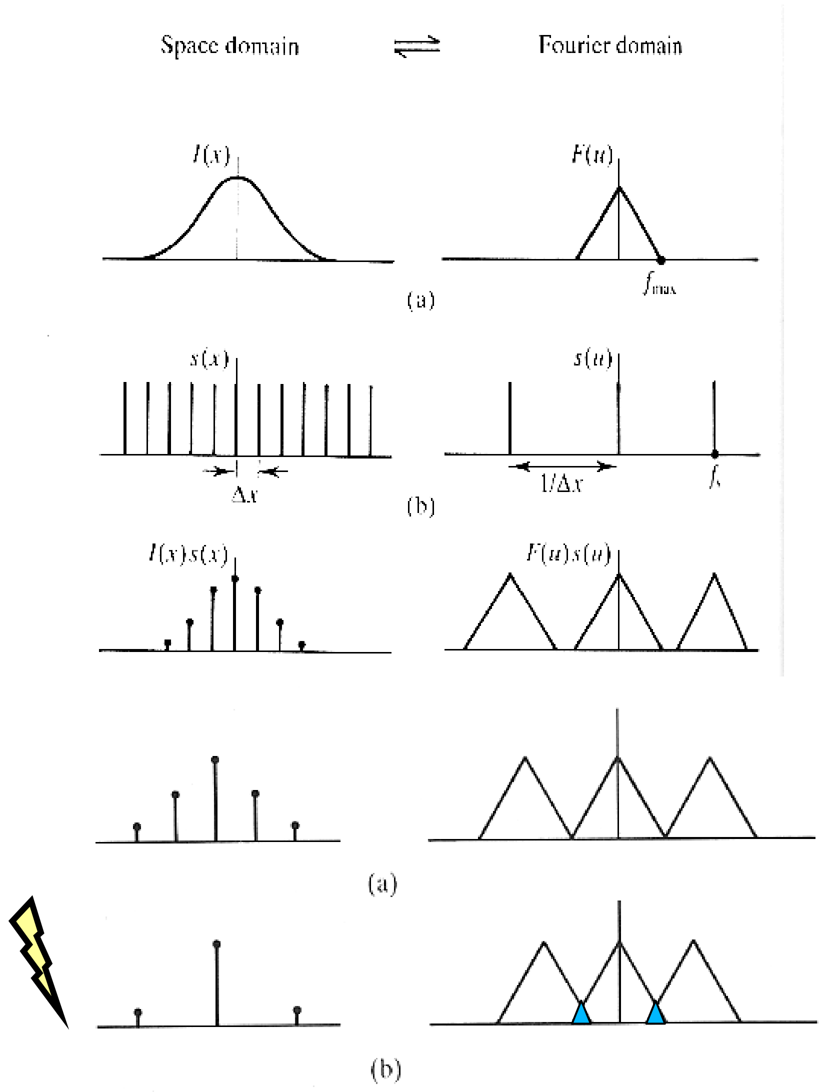
$$f(x) = \sum_{k=-\infty}^{\infty} \delta(x - k\Delta x)$$

$$F(w) = \frac{1}{\Delta x} \sum_{k=-\infty}^{\infty} \delta(w - k\frac{1}{\Delta x})$$



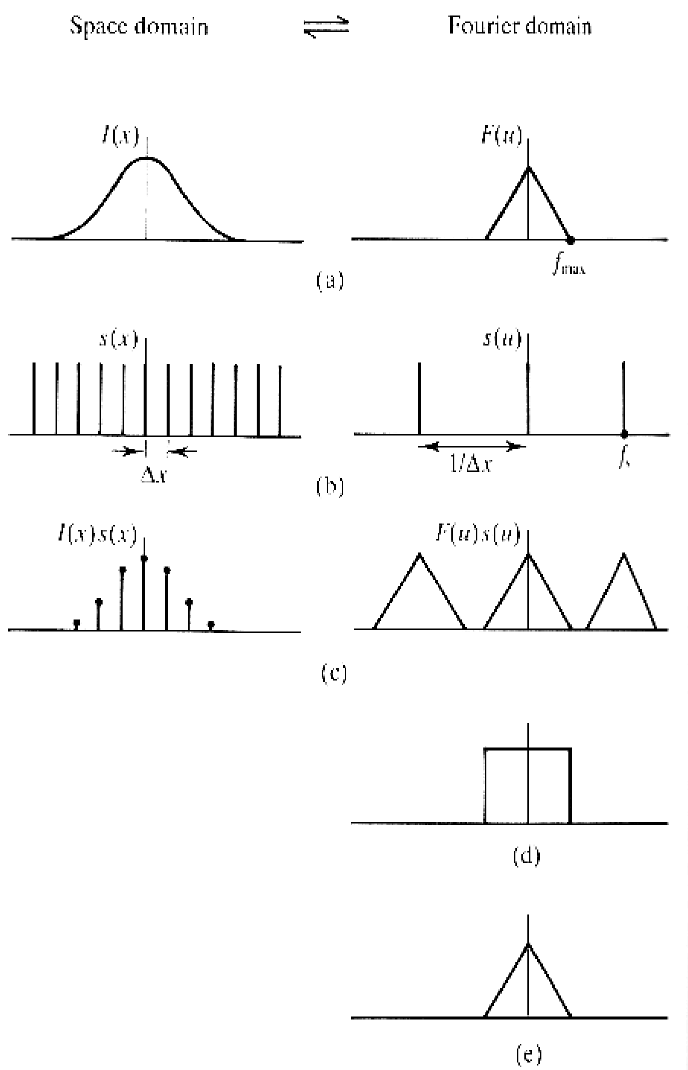
Sampling

- **Continuous function**
  - Band-limited Fourier transform
- **Sampled at discrete points**
  - Multiplication with Comb function in space domain
  - Corresponds to convolution in Fourier domain
  - Multiple copies of the original spectrum
- **Frequency bands overlap ?**
  - No: good
  - Yes: bad, ***aliasing***



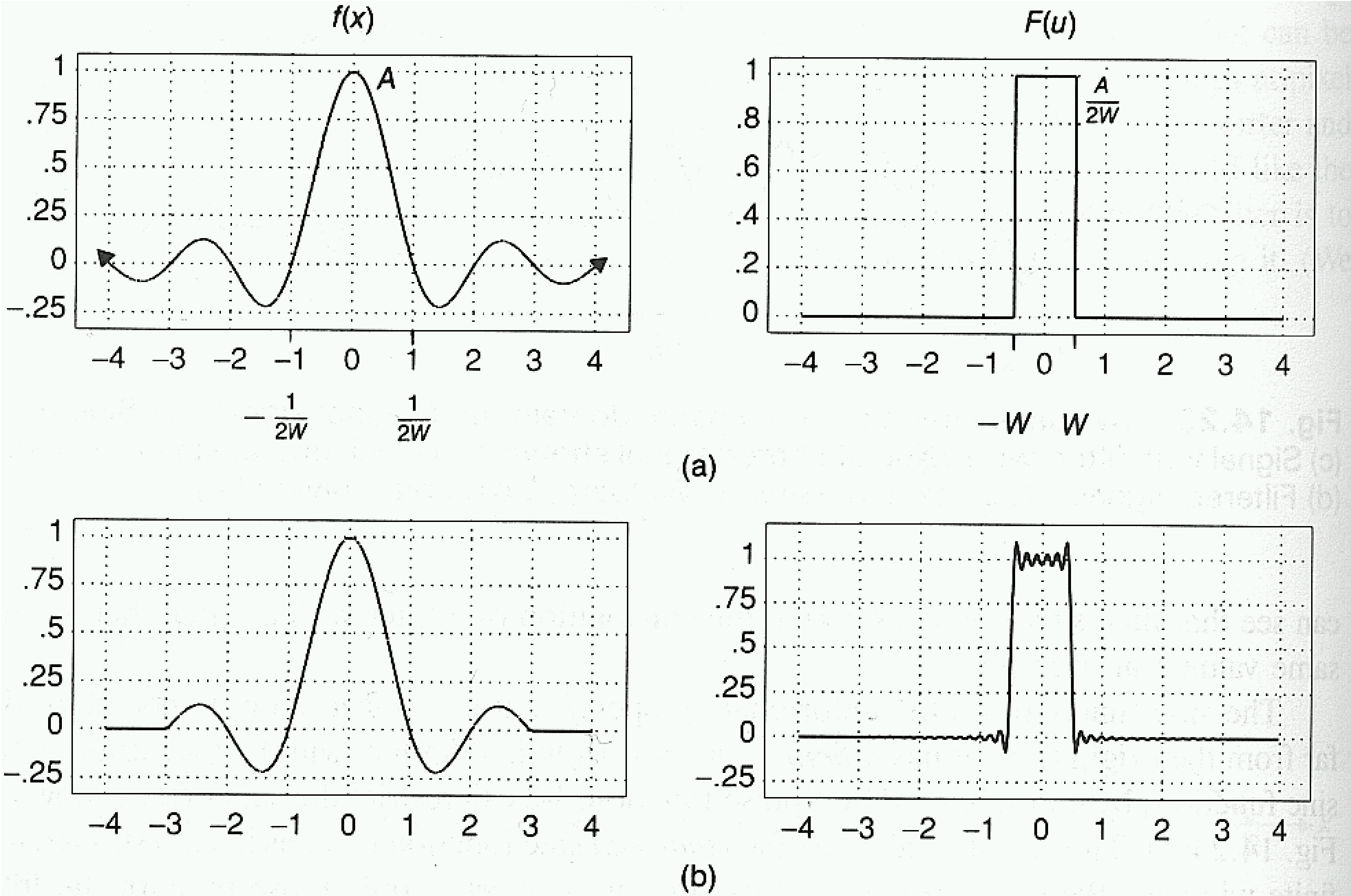
Reconstruction

- **Only original frequency band desired**
- **Filtering**
  - In Fourier domain: multiplication with windowing function around origin
  - In spatial domain: convolution with Fourier transform of windowing function
- **Optimal filtering function**
  - Box function in Fourier domain
  - Corresponds to sinc in space domain
    - Unlimited region of support
  - Spatial domain only allows approximations (due to finite support)



Reconstruction Filter

- **Cutting off the support is *not* a good solution**



Sampling and Reconstruction

Original function and its band-limited frequency spectrum

Signal sampling:

Mult./conv. with comb

Comb dense enough (sampling≥2*bandlimit)

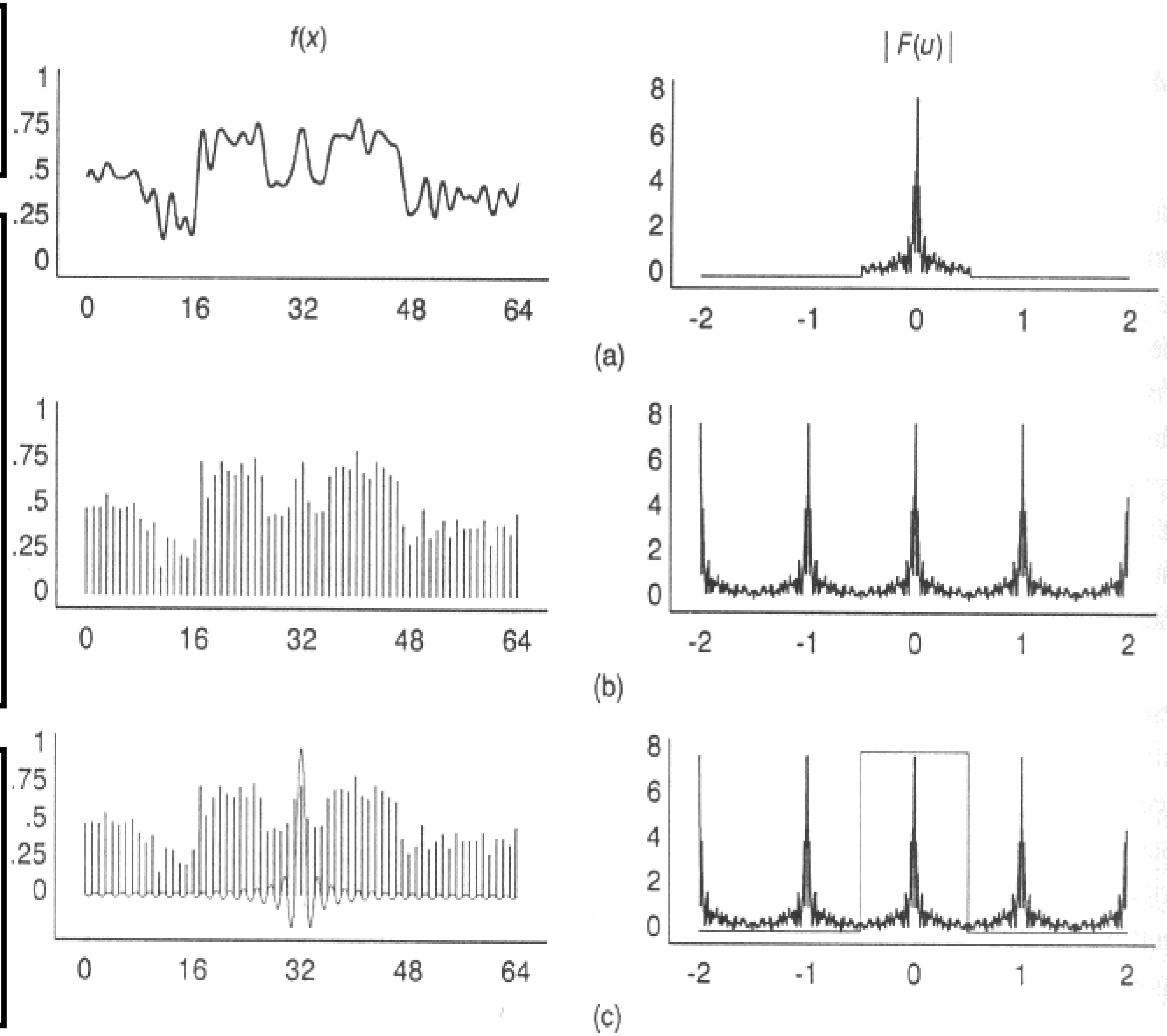
Frequency spectrum is replicated

Bands do not overlap

Correct filtering

Fourier: Box (mult.)
Image: sinc (conv.)

Only one copy



Sampling and Reconstruction

Reconstruction with ideal sinc

Identical signal

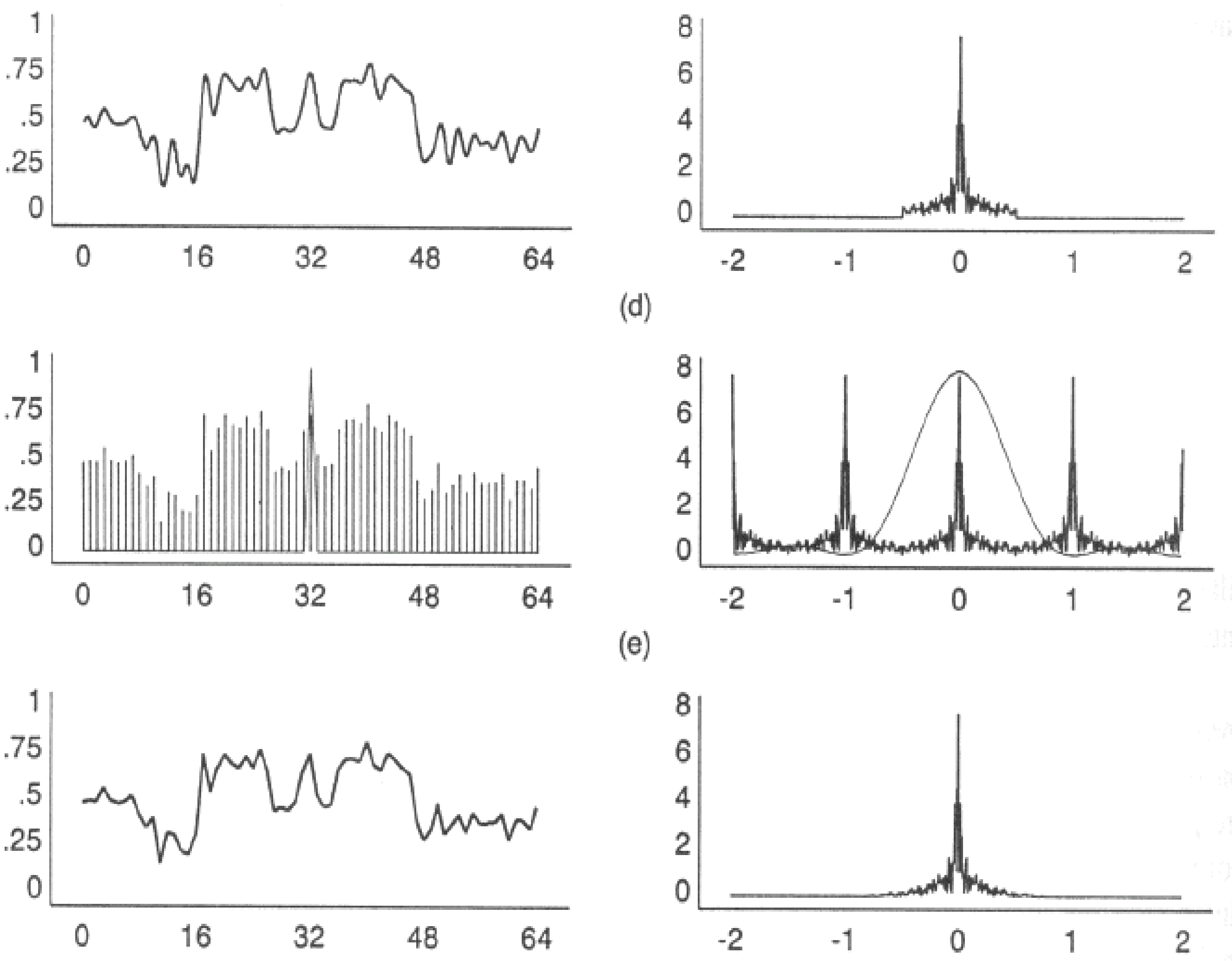
Approximate filtering

Space: tri (conv.)
Fourier: sinc^2 (mult.)

High frequencies are not ignored

➔ Aliasing

Reconstruction with tri function (= piecewise linear interpolation)



Sampling with Too Low Frequency

Original function

Sampling below Nyquist:

Comb spaced to far (sampling<2*bandlimit)

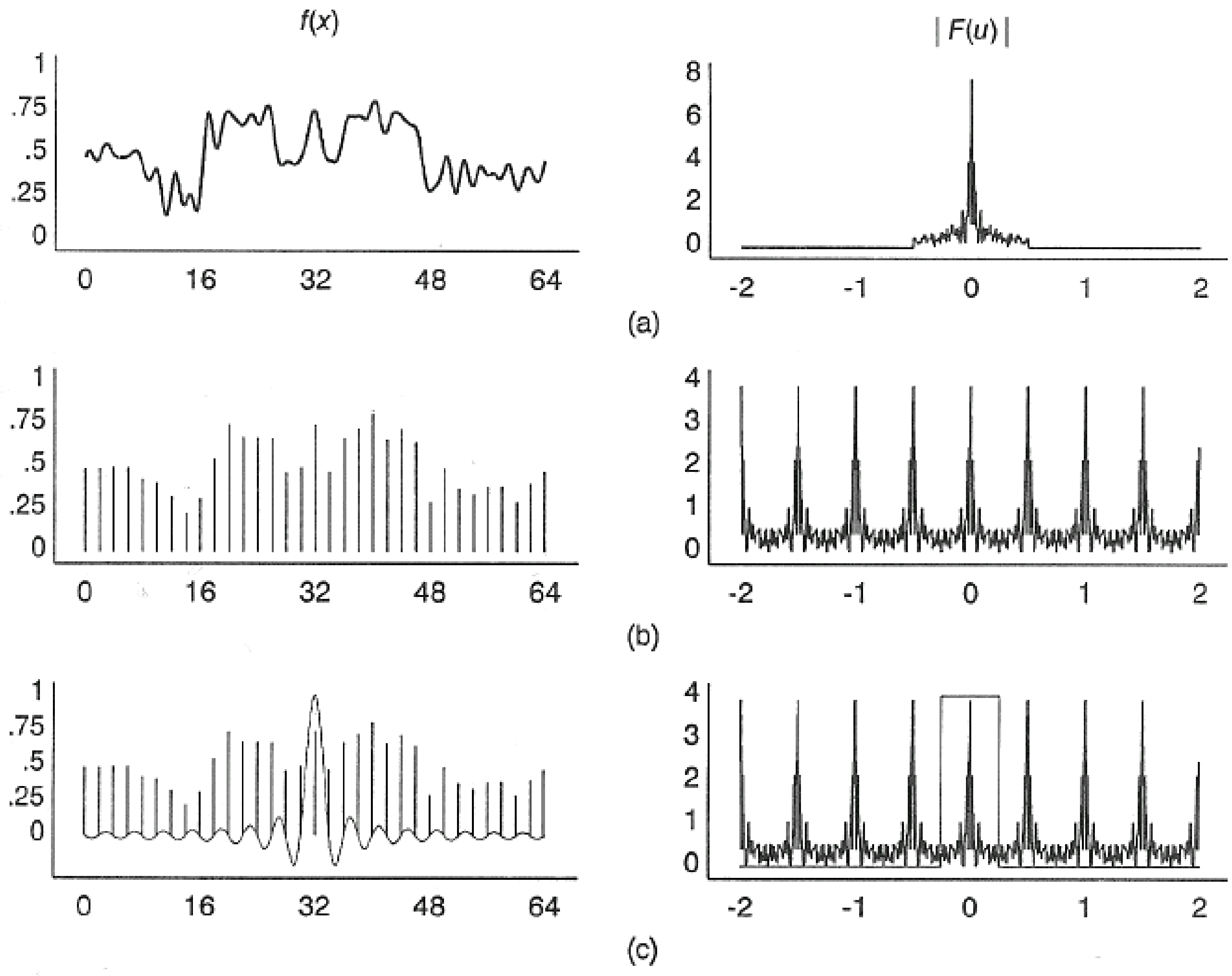
Frequency bands overlap

Correct filtering

Image: sinc (conv.)

Fourier: box (mult.)

Band overlap in frequency domain cannot be corrected - aliasing



Sampling with Too Low Frequency

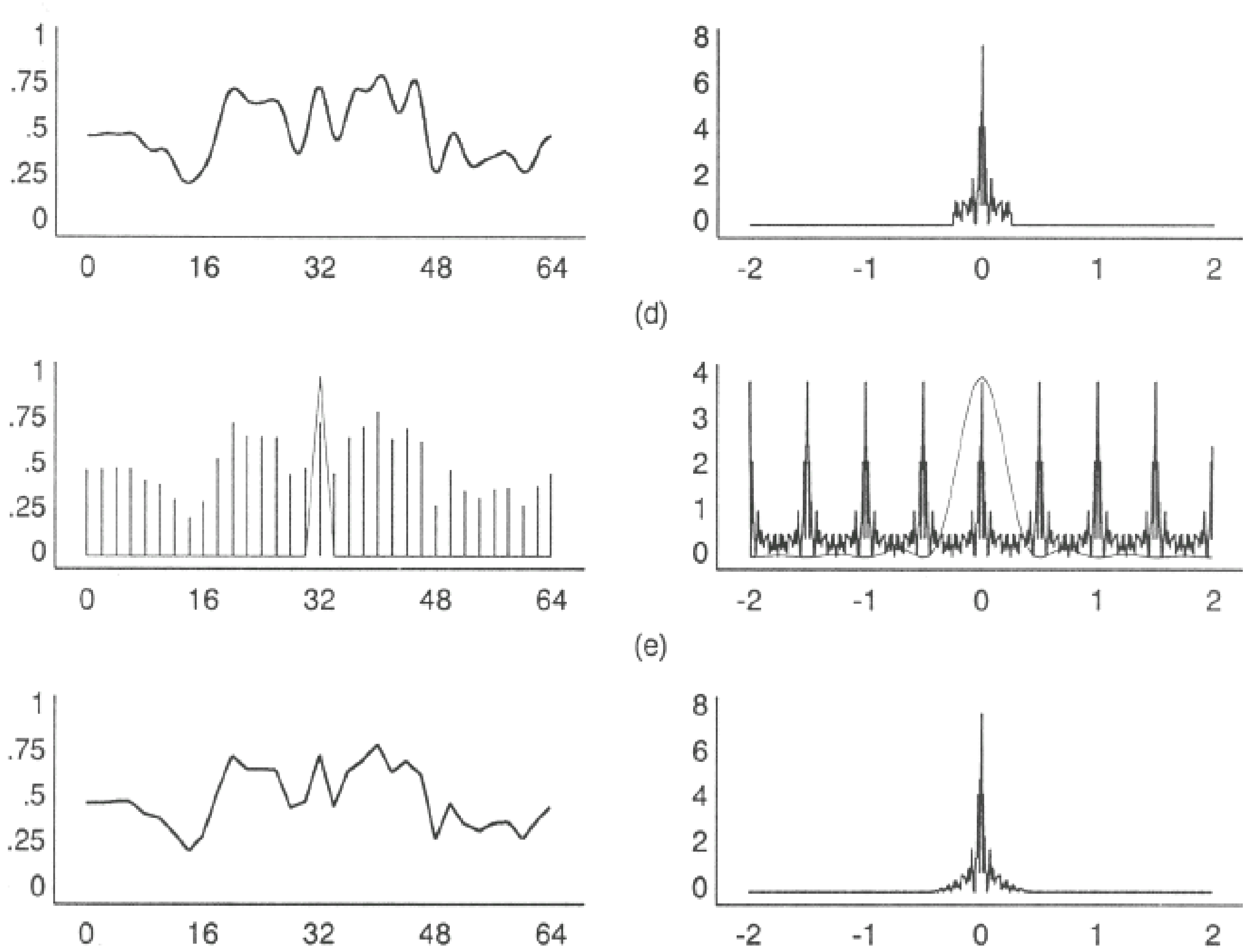
Reconstruction with ideal sinc

Reconstruction fails (frequency components wrong due to aliasing !)

Filtering with sinc² function

Reconstruction with tri function (= piecewise linear interpolation)

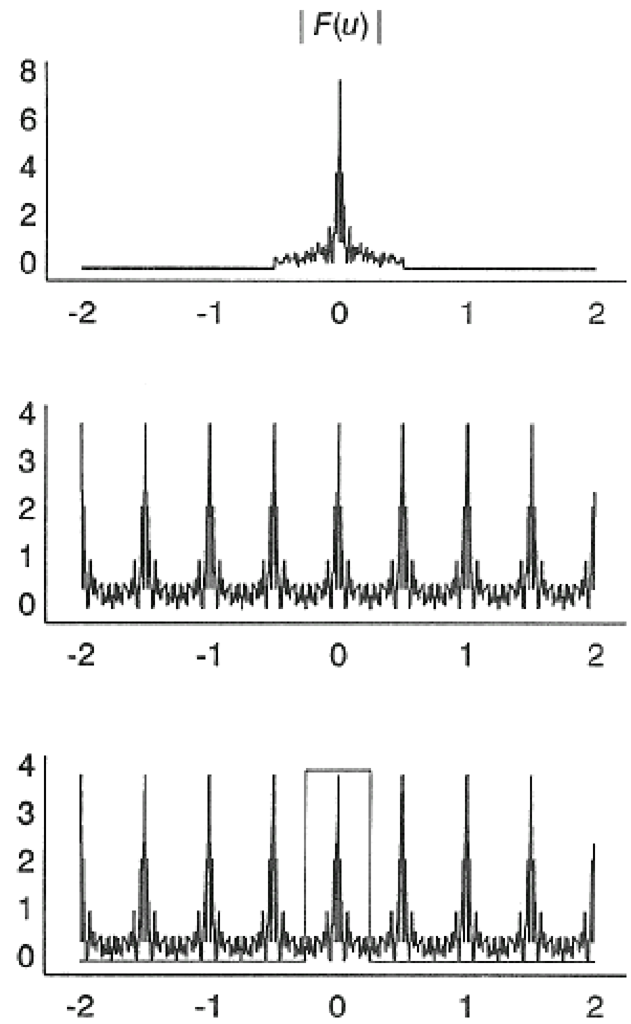
Even worse reconstruction



Aliasing

- **Overlap between replicated copies in frequency spectrum**

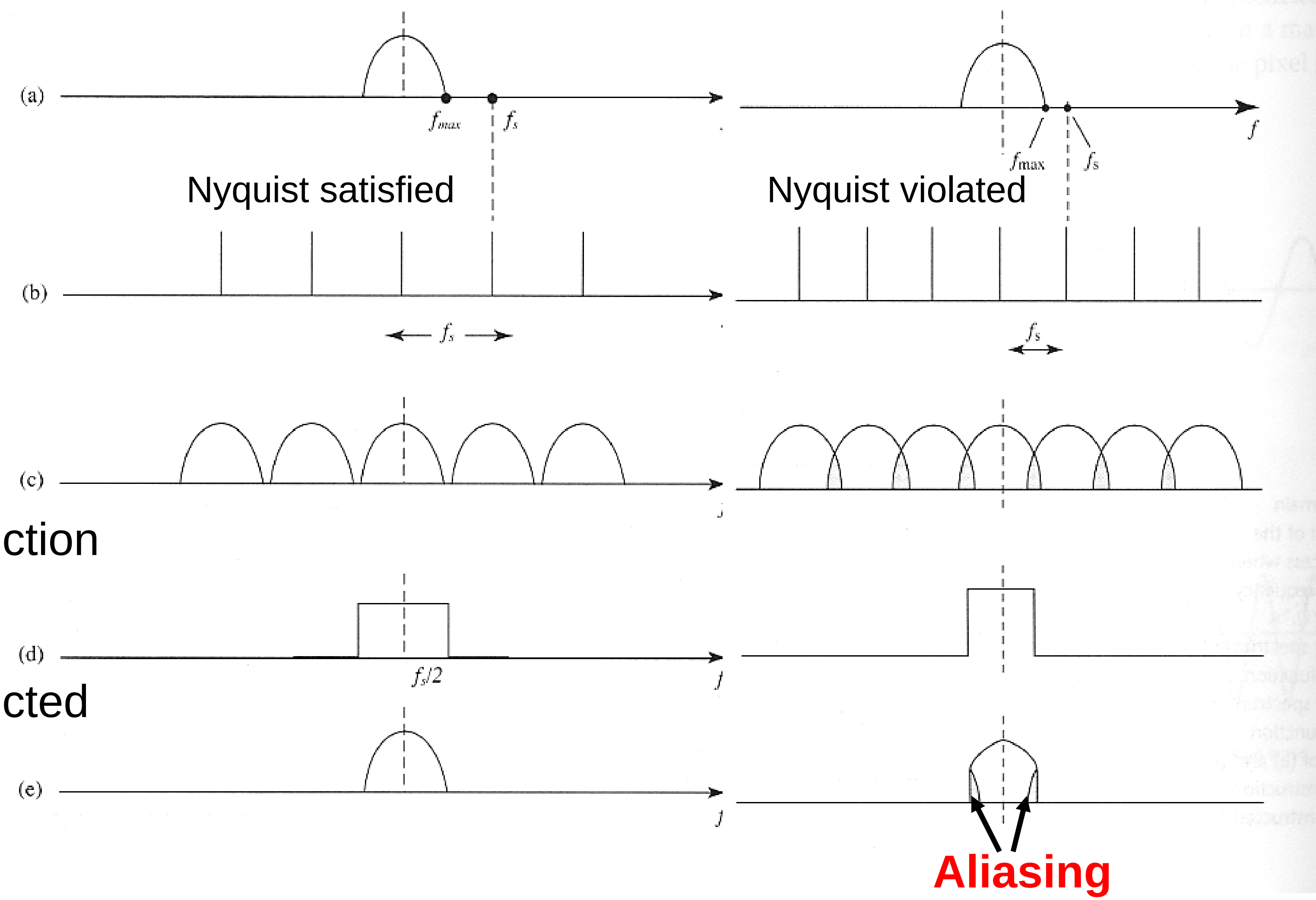
  High frequency components from the replicated copies are treated like low frequencies during the reconstruction process



Aliasing

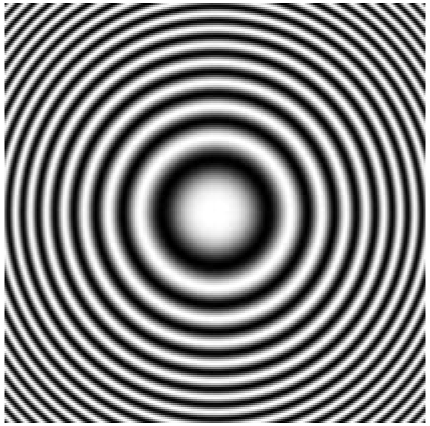
- **In Fourier space**

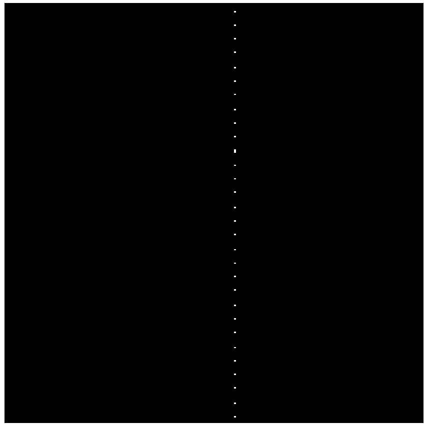
- Original spectrum
- Sampling comb
- Resulting spectrum
- Reconstruction Filter
- Reconstructed spectrum

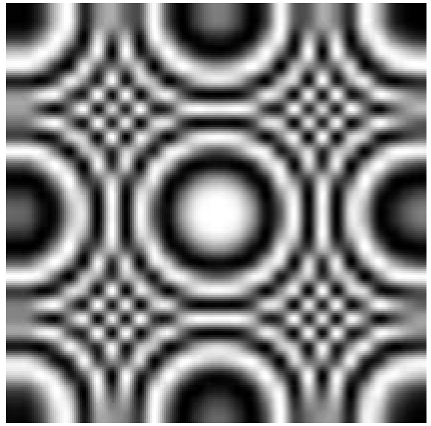


Aliasing 2D

[wikipedia]







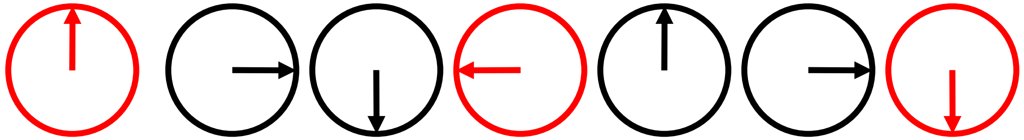
original image   sampled at these location   yields this reconstruction.

Sampling Artifacts

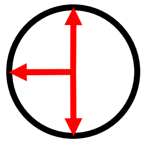
- **Spatial aliasing:**
  - Stair cases, Moiré patterns, etc.
- **Solutions:**
  - Increasing the sampling rate
    - Ok, but infinite frequencies at sharp edges
  - Post-filtering (after reconstruction)
    - Does not work - only leads to blurred stair cases
  - Pre-filtering (Blurring) of sharp geometry features
    - Slowly make geometry transparent at the edges
    - Correct solution in principal
    - Analytic low-pass filtering hard to implement
    - Super-sampling

Sampling Artifacts

- **Temporal Aliasing**
  - Cart wheels, ...



- **Solutions**
  - Increasing the frame rate
    - OK
  - Pre-filtering (Motion Blur)
    - Yes, possible for simple geometry (e.g., Cartoons)
    - Problems with texture, etc.
  - Post-filtering (Averaging several frames)
    - Does not work – only multiple detail



- **Important**
  - Distinction between aliasing errors and reconstruction errors

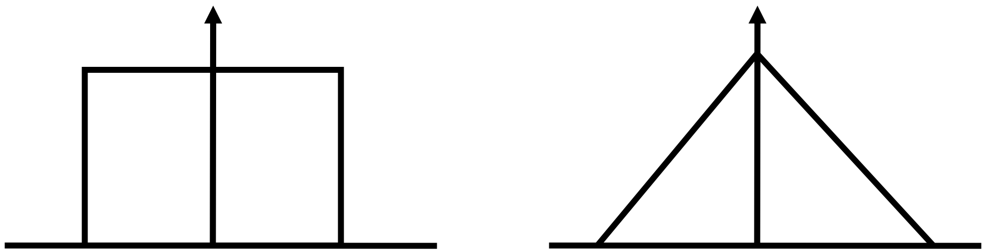
Aliasing

- **It all comes from sampling at discrete points**
  - Multiplied with comb function, no smoothly weighted filters
  - Comb function: repeats frequency spectrum
- **Or, from using non band limited primitives**
  - Hard edges $\Rightarrow$ infinitely high frequencies
- **In reality, integration over finite region necessary**
  - E.g., finite CCD pixel size
- **Computer: Analytic integration often not possible**
  - No analytic description of radiance or visible geometry available
- **Only way: numerical integration**
  - Estimate integral by taking multiple point samples, average
    - Leads to aliasing
  - Computationally expensive
  - Approximate

Sources of High Frequencies

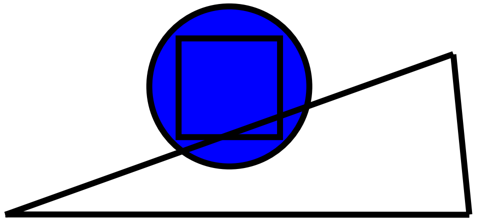
- **Geometry**
  - Edges, vertices, sharp boundaries
  - Silhouettes (view dependent)
  - …
- **Texture**
  - E.g., checkerboard pattern, other discontinuities, …
- **Illumination**
  - Shadows, lighting effects, projections, …

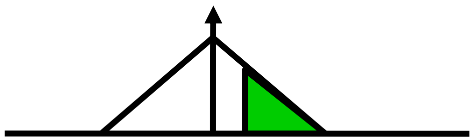
➔**Analytic filtering almost impossible**
  - Even with the most simple filters



Comparison

- **Analytic low-pass filtering**
  - Ideally eliminates aliasing completely
  - Hard to implement
    - Weighted or unweighted area sampling
    - Compute distance from pixel to a line
    - Filter values can be stored in look-up tables
      - Possibly taking into account slope
      - Distance correction
      - Non rotationally symmetric filters
    - Does not work at corners
- **Over-/Super-sampling**
  - Very easy to implement
  - Does not eliminate aliasing completely
    - Sharp edges contain *infinitely* high frequencies

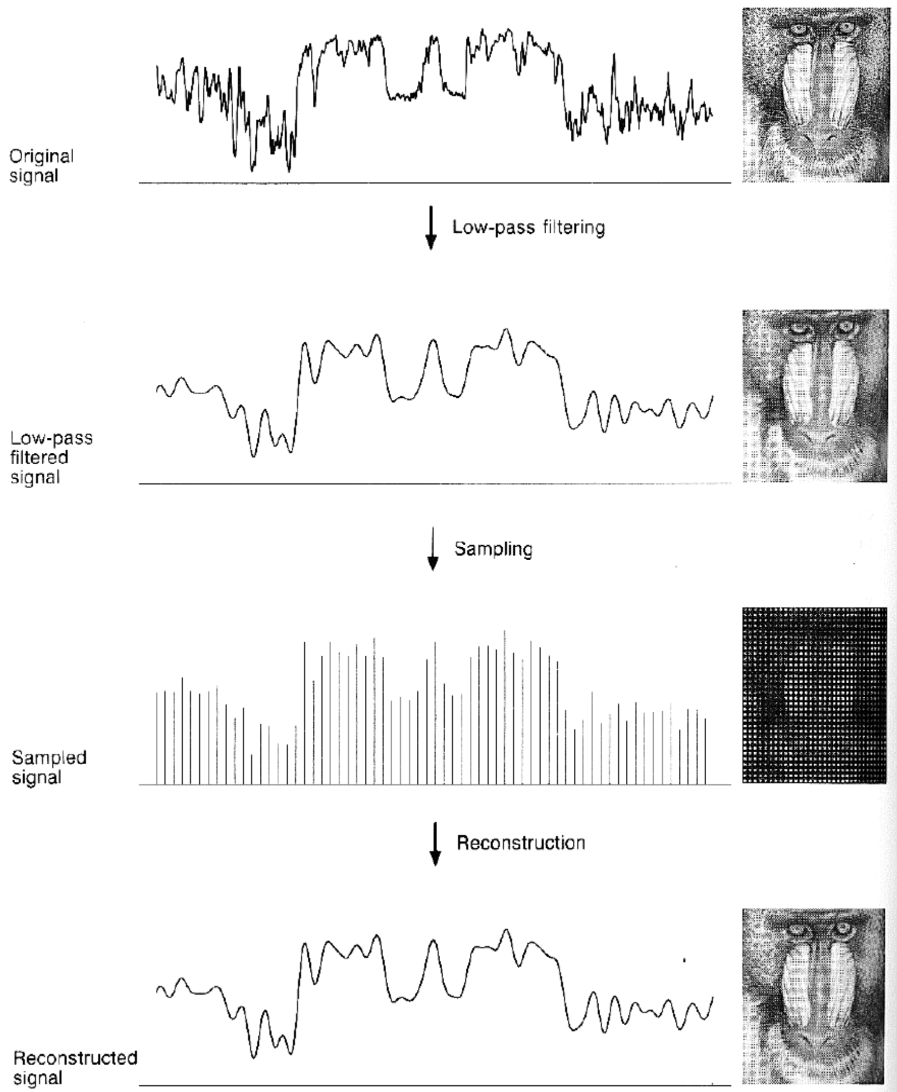




Antialiasing by Pre-Filtering

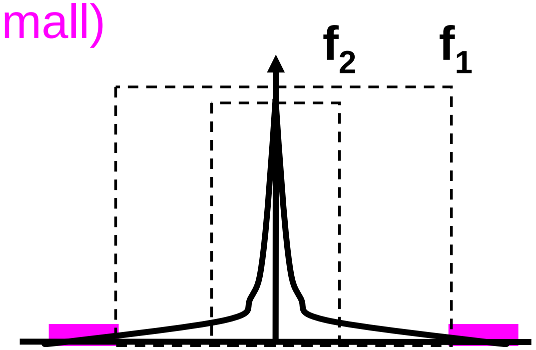
- **Filtering before sampling**
  - Band-limiting signal
  - Analog/analytic or
  - Reduce Nyquist frequency for chosen sampling-rate
- **Ideal reconstruction**
  - Convolution with sinc
- **Practical reconstruction**
  - Convolution with
    - Box filter, Bartlett (Tent)

  → Reconstruction error



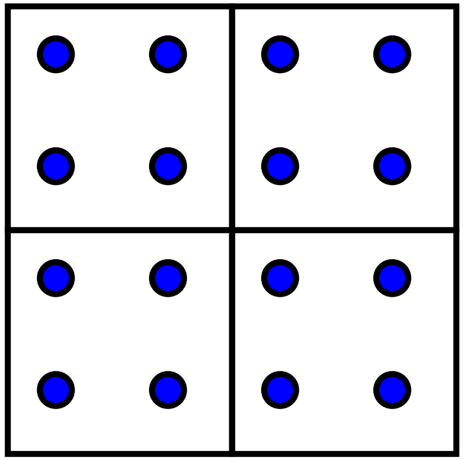
Re-Sampling Pipeline

- **Assumption**
  - Energy in high frequencies decreases quickly
  - Reduced aliasing by intermediate sampling with higher frequencies
- **Algorithm**
  - Super-sampling
    - Sample continuous signal with high frequency f_1
    - Aliasing with energy beyond f_1 (assumed to be small)
  - Reconstruction of signal
    - Filtering with $g_1(x)$: e.g. convolution with sinc_f_1
    - Exact representation with sampled values !!
  - Analytic low-pass filtering of signal
    - Filtering with filter $g_2(x)$ with $f_2 << f_1$
    - Signal is now band limited w.r.t. f_2
  - Re-sampling with a sampling frequency that is compatible with f_2
    - No additional aliasing
  - Filters $g_1(x)$ and $g_2(x)$ can be combined



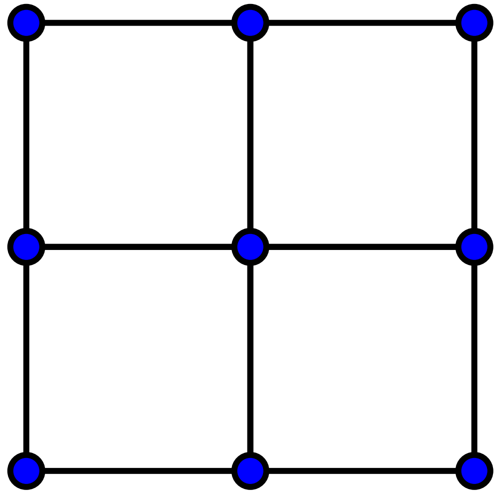
Super-Sampling in Practice

- **Regular super-sampling**
  - Averaging of N samples per pixel on a grid
  - N:
    - 4 quite good
    - 16 almost always sufficient
  - Samples
    - Rays, z-buffer, reflection, motion, ...
  - Filter Weights
    - Box filter
    - Others: B-spline, Pyramid (Bartlett), Hexagonal, ...
  - Regular super-sampling
    - Nyquist frequency for aliasing only shifted
      - ➔ Irregular sampling patterns

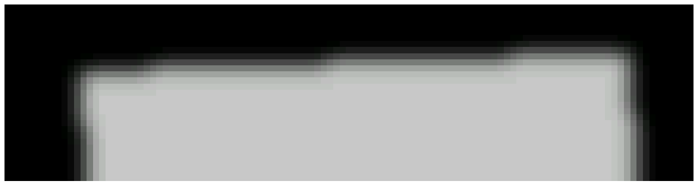


Super-Sampling Caveats

- **Popular mistake**
  - Sampling at the corners of every pixel
  - Pixel color by averaging
  - Free super-sampling ???
- **Problem**
  - Wrong reconstruction filter !!!
  - Same sampling frequency,
    but post-filtering with a hat function
  - Blurring: Loss of information
- **Post-Reconstruction Blur**



1x1 Sampling, 3x3 Blur



1x1 Sampling, 7x7 Blur

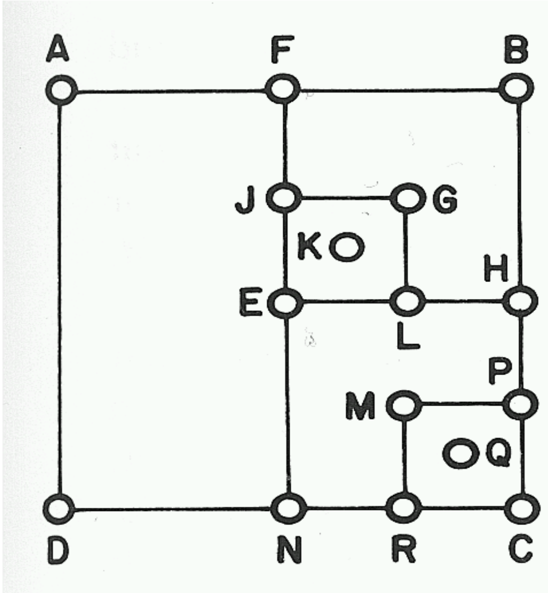
➔ **„Super-sampling“ does not come for free**

Adaptive Super-Sampling

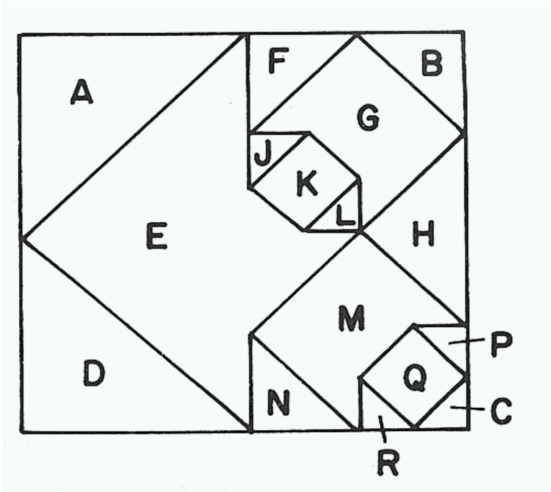
- **Adaptive super-sampling**
  - Idea: locally adapt sampling density
    - Slowly varying signal: low sampling rate
    - Strong changes: high sampling rate
  - Decide sampling density locally
  - Decision criterion needed
    - Differences of pixel values
    - Contrast (relative difference)
      - $|A-B| / |A|+|B|$

Adaptive Super-Sampling

- **Algorithm**
  - Sampling at corners and mid points
  - Recursive subdivision of each quadrant
  - Decision criterion
    - Differences, contrast, object-IDs, ray trees, ...
  - Filtering with weighted averaging
    - ¼ from each quadrant
    - Quadrant: ½ (midpoint + corner)
      - Recursion



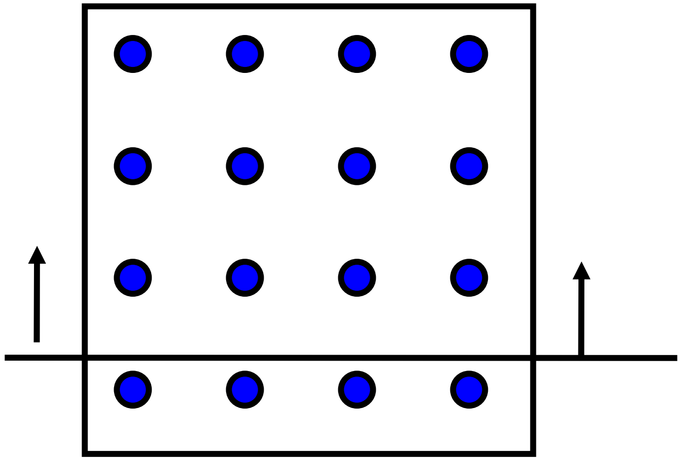
$$\frac{1}{4}\left(\frac{A+E}{2}+\frac{D+E}{2}+\frac{1}{4}\left[\frac{F+G}{2}+\frac{B+G}{2}+\frac{H+G}{2}+\frac{1}{4}\left\{\frac{J+K}{2}+\frac{G+K}{2}+\frac{L+K}{2}+\frac{E+K}{2}\right\}\right]\right.$$
$$\left.+\frac{1}{4}\left[\frac{E+M}{2}+\frac{H+M}{2}+\frac{N+M}{2}+\frac{1}{4}\left\{\frac{M+Q}{2}+\frac{P+Q}{2}+\frac{C+Q}{2}+\frac{R+Q}{2}\right\}\right]\right)$$



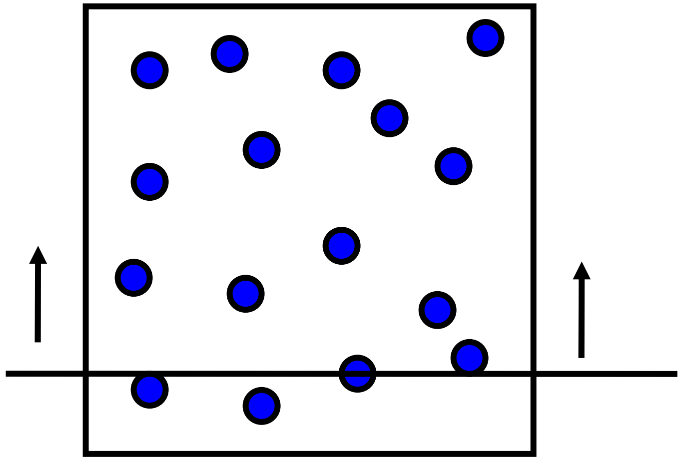
- **Extension**
  - Jittering of sample points

Super-Sampling in Practice

- **Problems with regular super-sampling**
  - Expensive: 4-fold to 16-fold effort
  - Non-adaptive: Same effort everywhere
  - Too regular: Apparent reduction of number of levels
- **Introduce irregular sampling pattern**



0 → 4/16 → 8/16 → 12/16 → 16/16



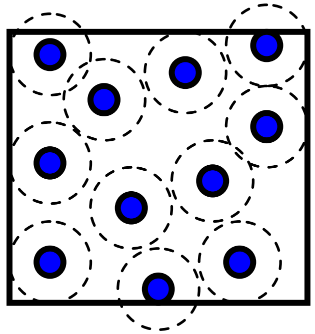
Better, but noisy

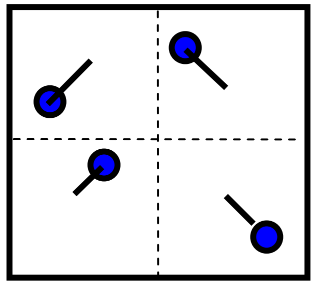
➔ Stochastic super-sampling

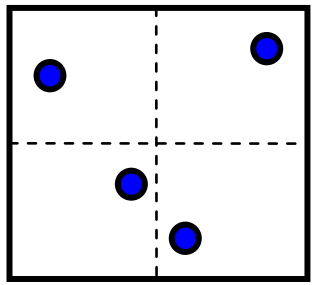
- Or analytic computation of pixel coverage and pixel mask

Stochastic Sampling

- **Requirements**
  - Even distribution
  - Little correlation between samples
  - Incremental generation
- **Generation of samples**
  - Poisson-disk sampling
    - Fixes a minimum distance between samples
    - Random generation of samples
      - Rejection, if too close to other samples
  - Jittered sampling
    - Random perturbation from regular positions
  - Stratified Sampling
    - Subdivision into areas with one random sample each
    - Improves even distribution
  - Quasi-random numbers (Quasi-Monte Carlo)
    - E.g. Halton Sequence
    - Advanced feature

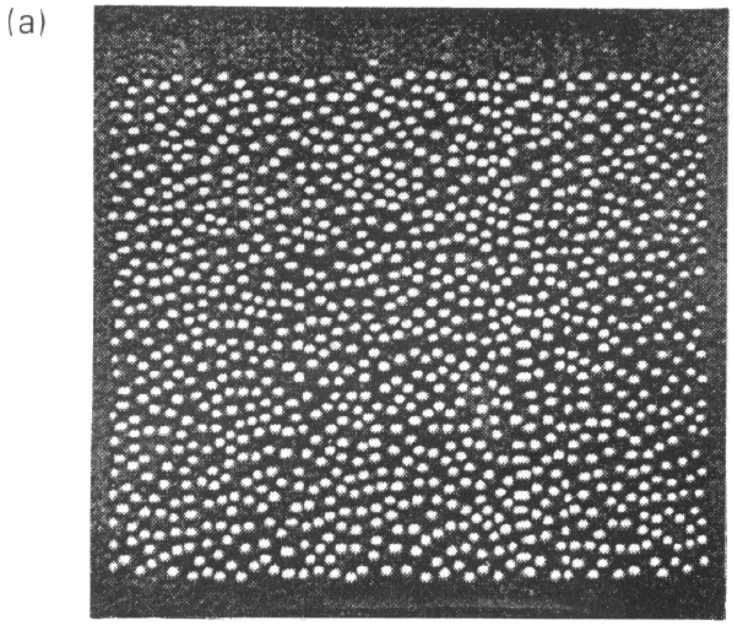




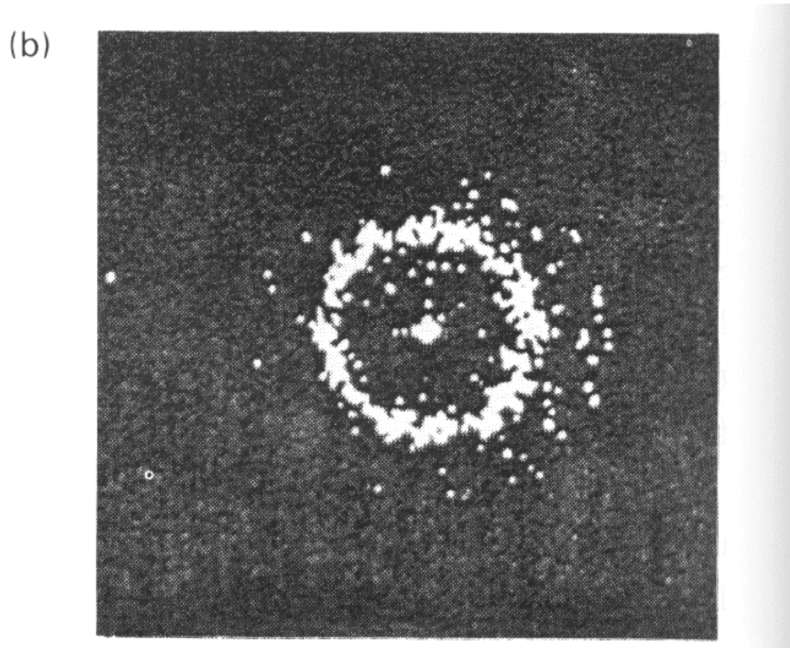


Poisson-Disk Sample Distribution

- **Motivation**
  - Distribution of the optical receptors on the retina (here: ape)



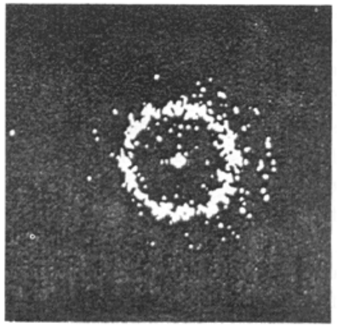
Distribution of the receptors

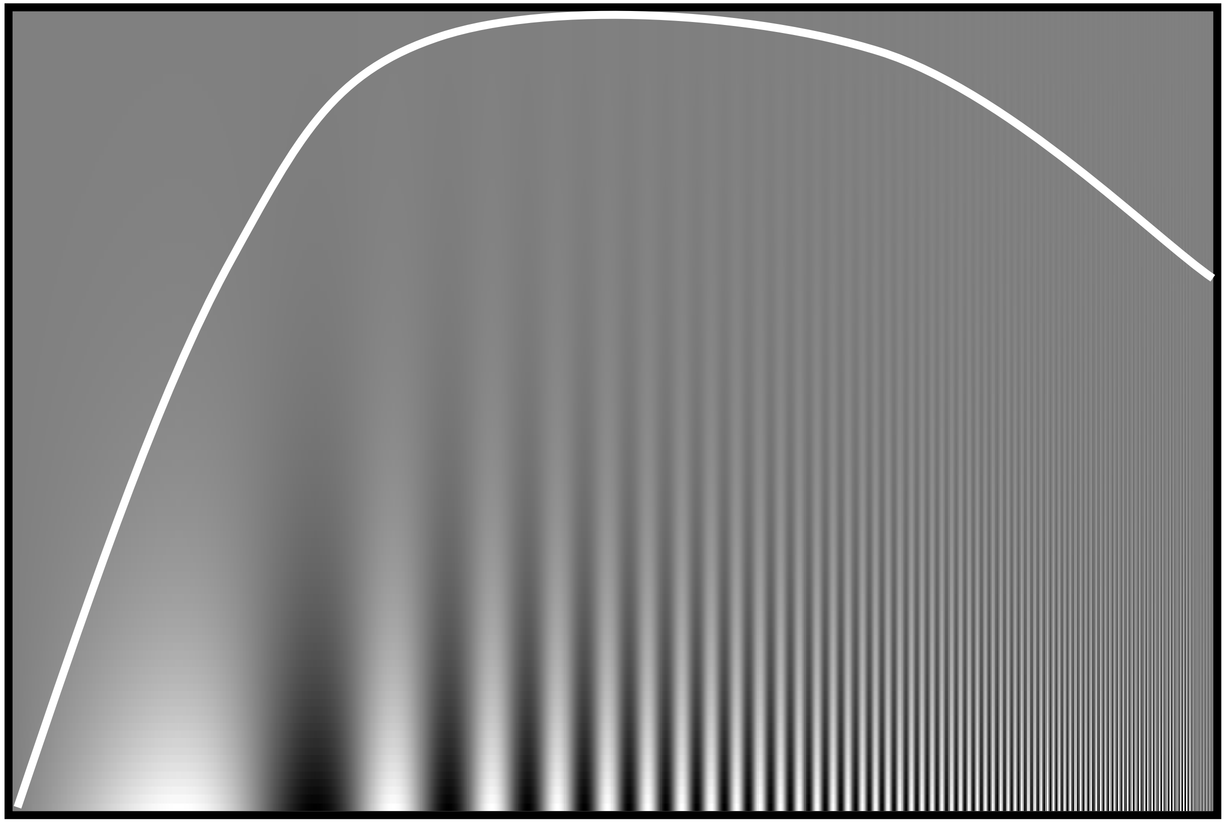


Fourier analysis

HVS: Poisson Disk Experiment

- **Human Perception**
  - Very sensitive to regular structures
  - Insensitive against (high frequency) noise

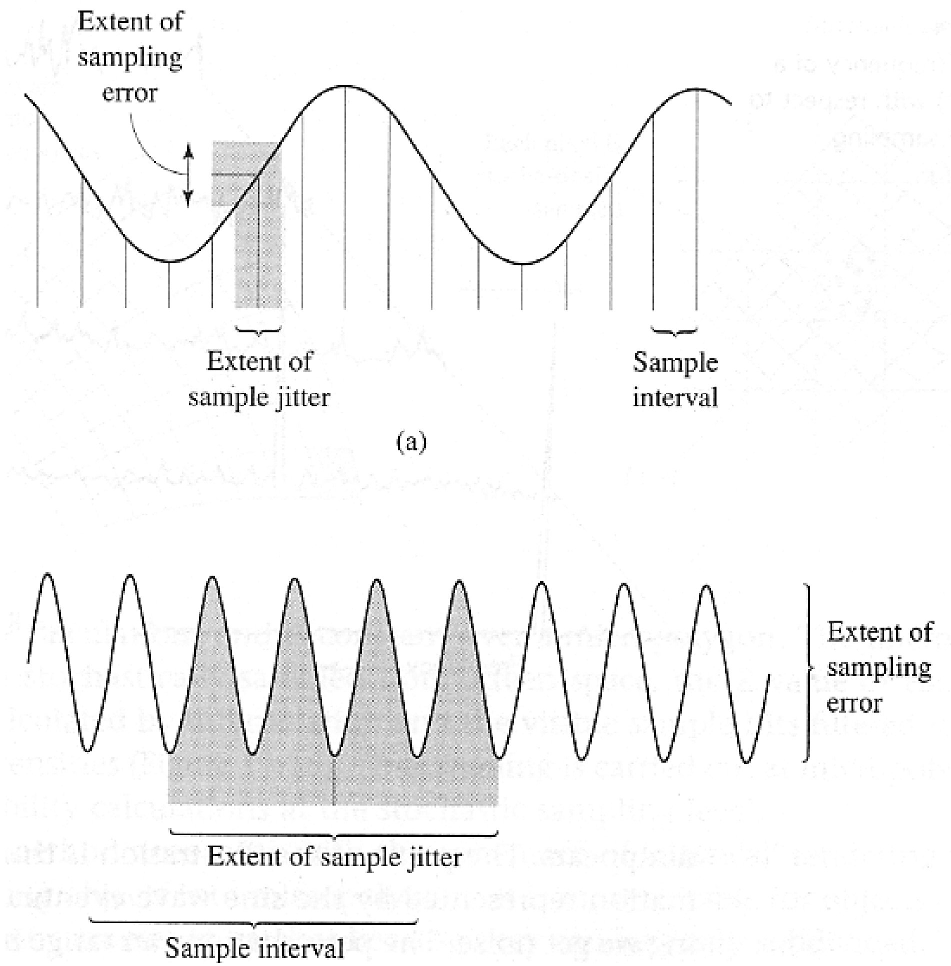




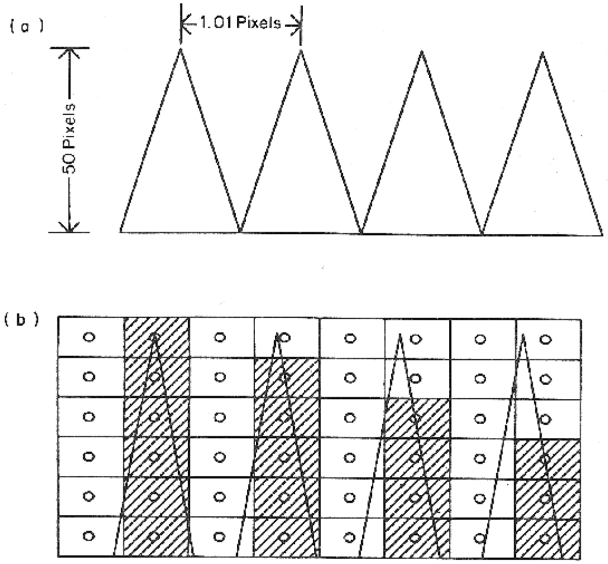
Campbell-Robson contrast sensitivity chart

Stochastic Sampling

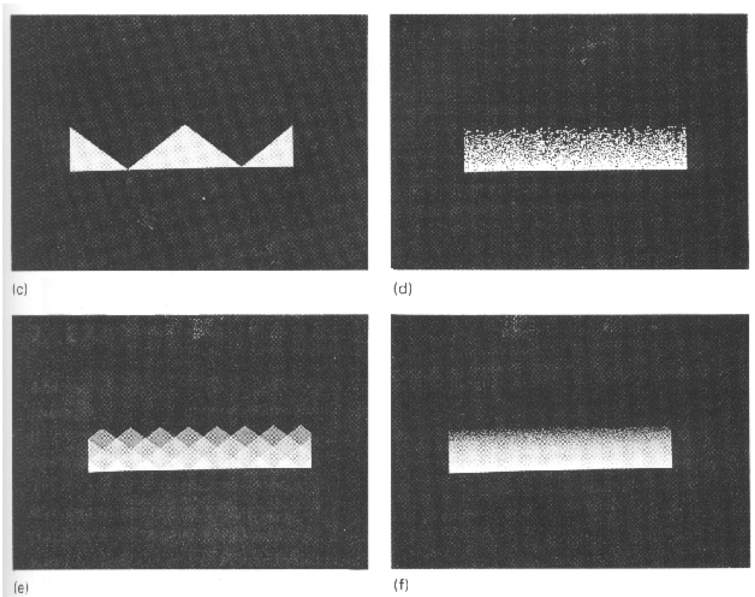
- **Stochastic Sampling**
  - Transforms energy in high frequency bands into noise
  - Low variation in sample domain
    - Closely reconstructs target value
  - High variation
    - Reconstructs average value



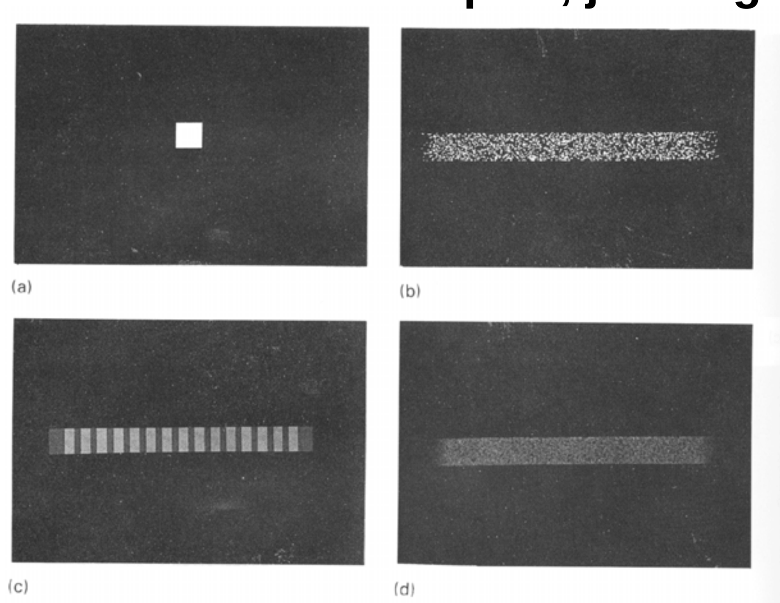
Examples



Triangle comb:
(Width: 1.01 pix, Heigth: 50 pix):
1 sample, no jittering
1 sample, jittering
16 samples, no jittering
16 samples, jittering



Motion Blur:
1 sample, no jittering
1 sample, jittering
16 samples, no jittering
16 samples, jittering



Comparison

Regular, 1x1

Regular 3x3

Regular, 7x7

Jittered, 3x3

Jittered, 7x7