… Sampling …
… Filtering …
… Reconstruction …

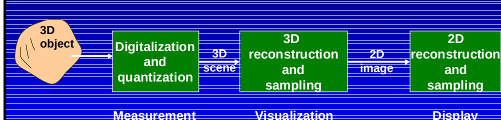
Leonid I. Dimitrov & Miloš Šrámek
Commission for Scientific Visualization
Austrian Academy of Sciences

Overview

- Sampling of continuous signals
- Filtering of discrete signals
- Reconstruction of continuous signals
- Sampling aspects of volume rendering
- Voxelization of geometric objects

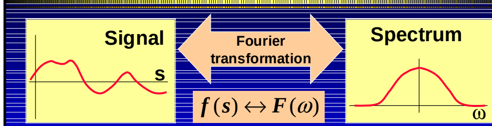
Why Sampling and Reconstruction?

- Real-world signals are continuous



- Computer representations are discrete

Signal and Spectrum



- FT: decomposition of a signal into a sum of sinusoids, determined by frequency, amplitude and phase
- *f(x)* - representation in the spatial domain
- *F(ω)* - representation in the frequency domain

Fourier Transform

$$f(s) \leftrightarrow F(\omega)$$

- Direct FT: $F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(s) e^{-i\omega s} ds$
- Inverse FT: $f(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega) e^{i\omega s} d\omega$

Convolution

- Continuous case:

$$f(s) * h(s) = \int_{-\infty}^{\infty} f(s) \cdot h(s-\tau) d\tau$$

- Discrete case:

$$f[n] * h[n] = \sum_{k} f[k] \cdot h[n-k]$$

Convolution Theorem

- Continuous case:

$$FT\{f(t)*h(t)\}=F(\omega)\cdot H(\omega)$$

$$FT\{f(t)\cdot h(t)\}=F(\omega)*H(\omega)$$

- Discrete case:

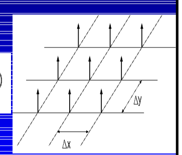
$$FT\{f[n]*h[n]\}=F(\Omega)\cdot H(\Omega)$$

$$FT\{f[n]\cdot h[n]\}=F(\Omega)*H(\Omega)$$

Sampling

- Input signal: $f_I(x,y)$
- Ideal sampling function:

$$s(x,y)=\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}\delta(x-i\Delta_x, y-j\Delta_y)$$



- Sampled signal:

$$f_s(x,y)=f_I(x,y)s(x,y)$$

Sampling in the Frequency Domain

Spectrum of the sampled signal:

$$F_s(\omega_x,\omega_y)=\frac{1}{4\pi^2}F_I(\omega_x,\omega_y)*S(\omega_x,\omega_y)$$

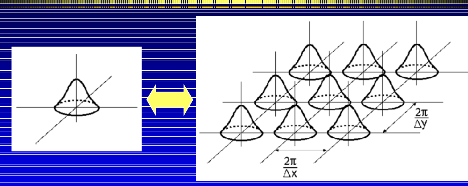
where

$$S(\omega_x,\omega_y)=\frac{4\pi^2}{\Delta_x\Delta_y}\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty}\delta(\omega_x-i\omega_{xs},\omega_y-i\omega_{ys})$$

and

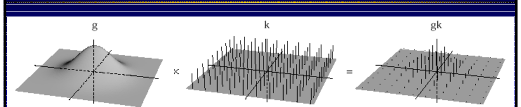
$$\omega_{xs}=\frac{2\pi}{\Delta_x},\ \omega_{ys}=\frac{2\pi}{\Delta_y}$$

Sampling in the Frequency Domain

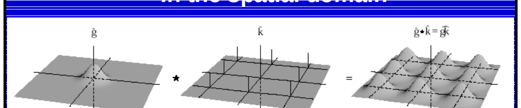


- Sampling a signal causes replication of its spectrum in the frequency domain
- Nyquist criterion: $\omega_s > 2\omega_m$

Sampling of a Signal



In the spatial domain



In the frequency domain

Real-world Sampling Process

- The sampling function has a volume
- It is approximated by a Gaussian:

$$s_r(x,y,z)=\exp\left(-\left(\frac{x^2}{2\sigma_x}+\frac{y^2}{2\sigma_y}+\frac{z^2}{2\sigma_z}\right)\right)$$

- A Gaussian is a low-pass filter, i.e. it causes blurred image, blunt edges

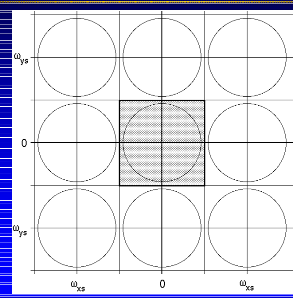
Reconstruction

Direct reconstruction of a continuous signal from discrete samples:

- The spectrum replicas have to be suppressed by a reconstruction filter
- The ideal reconstruction filter is a box function:

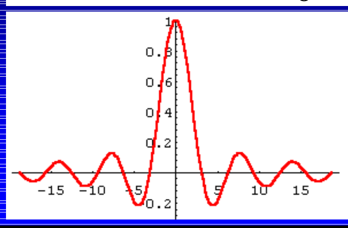
$$R(\omega_x, \omega_y) = \begin{cases} K \text{ if } |\omega_x| \le \omega_{xc} \wedge |\omega_y| \le \omega_{yc} \\ 0 \text{ else} \end{cases}$$

Reconstruction



Ideal Reconstruction Filter

$$\text{box}(\omega) \leftrightarrow \text{sinc}(s) = \frac{\sin(s)}{s}$$



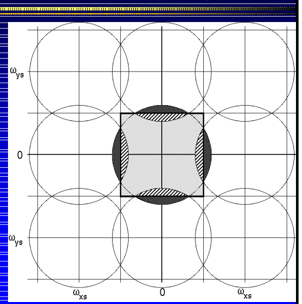
Ideal Reconstruction Filter

$$\text{box}(\omega) \leftrightarrow \text{sinc}(s) = \frac{\sin(s)}{s}$$

- Bounded support in the frequency domain, but unbounded in the spatial domain
- Can't be realized practically
- It has to be approximated (bounded):
  - The approximations have unbounded support in the frequency domain

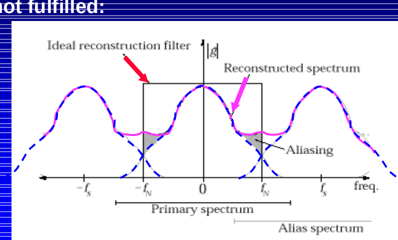
Problems with the Reconstruction (aliasing)

- If the Nyquist criterion is not fulfilled
- If the reconstruction filter is too big or too small



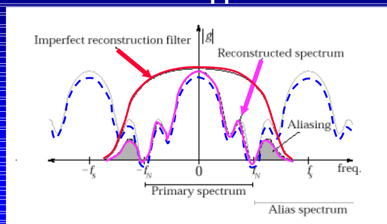
Prealiasing

- Wrong frequencies appear if the Nyquist criterion is not fulfilled:



Postaliasing

- Wrong frequencies appear also if the reconstruction filter support is too broad



Classification of 3D reconstruction filters

- Separable:

$$h(x,y,z) = h_s(x)\ h_s(y)\ h_s(z)$$

  - Sequential application along the axes
  - Computational complexity: *3n*
- Spherically symmetrical:
  - Computational complexity: n^3

Separable Filters

- Order 0: nearest neighbor

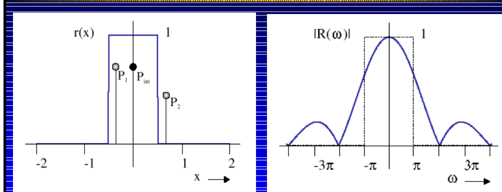
$$h_s = 1 \text{ if } |x| < 0.5$$

- Order 1: linear interpolation

$$h_s = 1 - |x| \text{ if } |x| < 1$$

- Order 3: cubic filters:
  - Cubic B-spline
  - Catmull-Rom spline

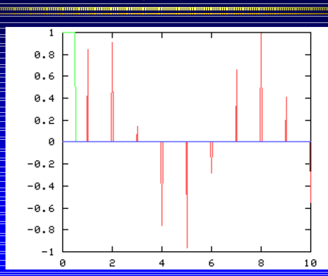
Nearest Neighbor



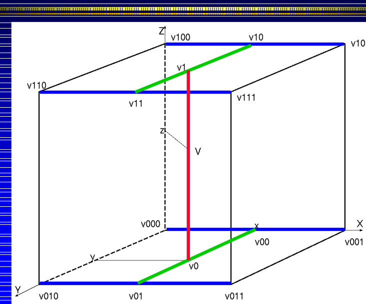
Filter

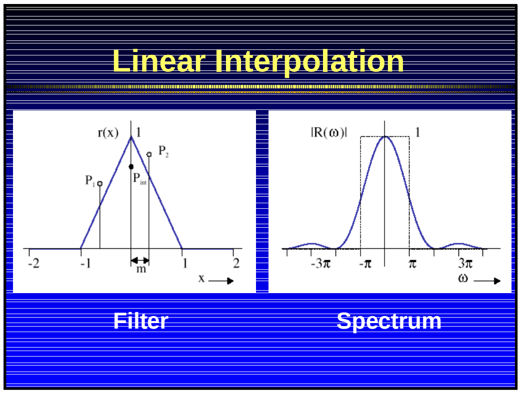
Spectrum

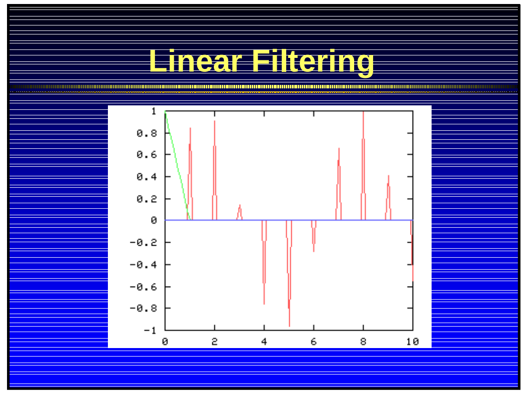
Nearest Neighbor Filtering

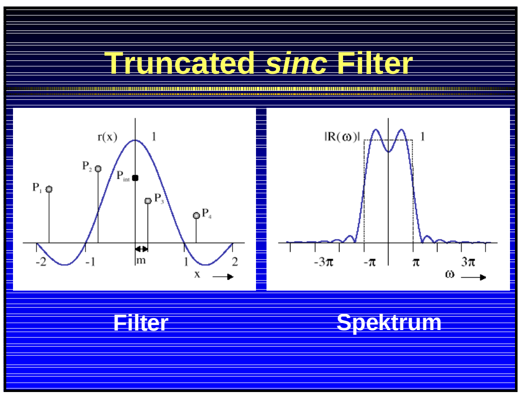


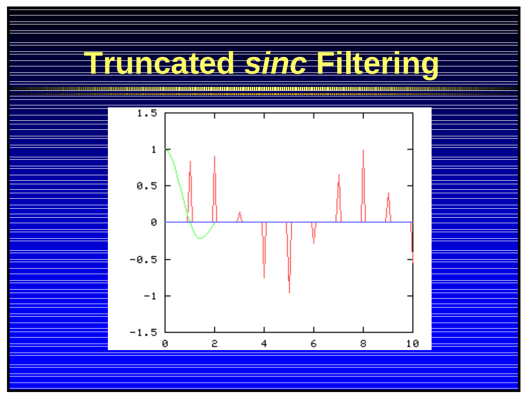
Trilinear Reconstruction Filter

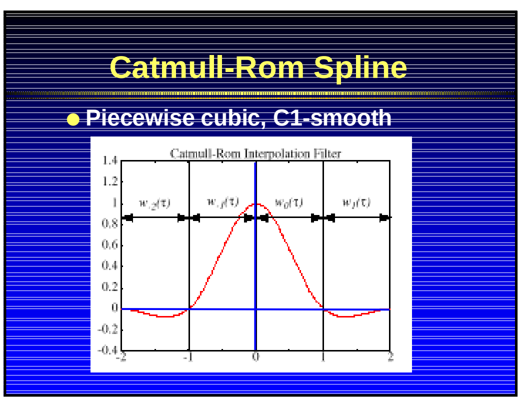


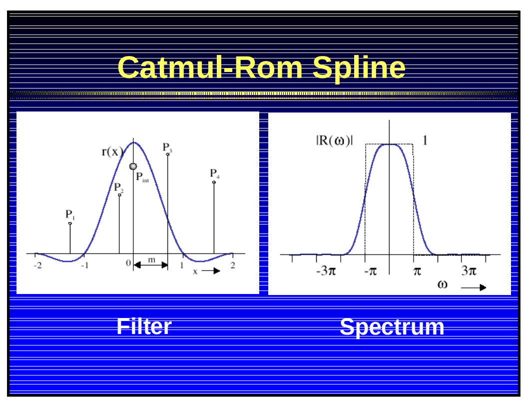

Linear Interpolation
Filter
Spectrum


Linear Filtering

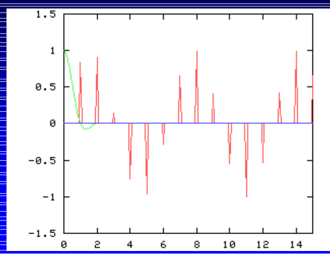

Truncated *sinc* Filter
Filter
Spektrum


Truncated *sinc* Filtering


Catmull-Rom Spline
Piecewise cubic, C1-smooth
Catmull-Rom Interpolation Filter


Catmul-Rom Spline
Filter
Spectrum

Catmull-Rom Spline Filtering



Other Separable Filters

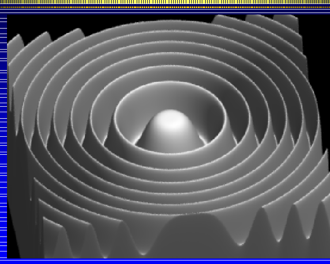
- Gaussian filter

$$h_s(x) = \exp(-\frac{x^2}{2\sigma^2}), \quad |x| < x_m$$

- Windowed *sinc* filter

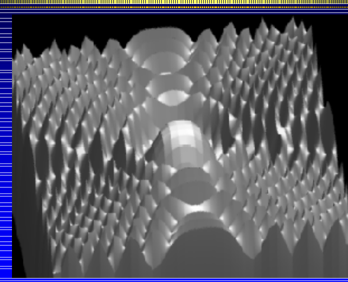
$$h_s(x) = \left(1 + \cos(\frac{\pi x}{x_m})\right) \operatorname{sinc}(\frac{4x}{x_m}), \quad |x| < x_m$$

An Example

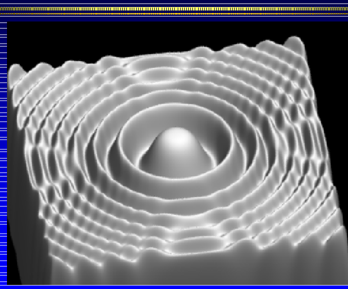


$F_N = 10$, sampling 40 x 40 x 40

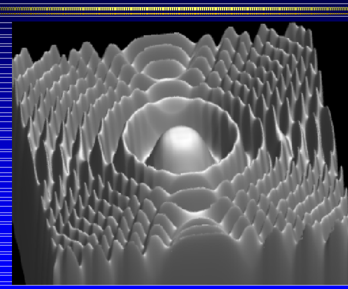
Reconstruction by the Trilinear Filter



Reconstruction by the Cubic B-spline Filter



Reconstruction by the Catmull-Rom Spline Filter




Reconstruction by the
Windowed *sinc* Filter