University of British Columbia

CPSC 414 Computer Graphics

Scan Conversion

Week 6, Fri 10 Oct 2003

- sampling

News

- project 1
  - solution today
  - hall of fame next week
    - great work!!
- extra office hours
  - Fri 10-11 usual, 11-1:30 extra lab hours
  - Mon 10/13 no class, no office hours
  - Tue 11-1 extra lab hours,
    - 4-5:30 my office hours rescheduled (FSC2618)

News

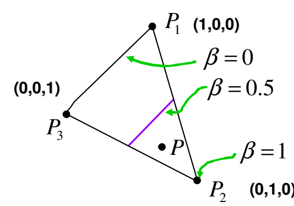
- Office hours reminder: FSC 2618
  - Mondays 10:30-11:30 or by appointment
  - exceptions: Oct 20, Nov 10
- Readings
  - Chap 8.9-8.11, Fri 10/3 slide notes

Barycentric Coordinates recap

- weighted combination of vertices

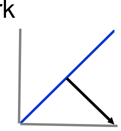
$$\begin{cases} P = \alpha \cdot P_1 + \beta \cdot P_2 + \gamma \cdot P_3 \\ \alpha + \beta + \gamma = 1 \\ 0 \le \alpha, \beta, \gamma \le 1 \end{cases}$$

"convex combination of points"



Transforming Normals

- nonuniform scaling does not work
- x-y=0 plane
  - line x=y
  - normal: [1,-1,0]
    - ignore normalization for now



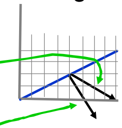
Transforming Normals

- apply nonuniform scale: stretch along x by 2
  - new plane x = 2y
- transformed normal

$$\begin{bmatrix} 2 \\ -1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}$$

2x – y = 0 (2x=y)

– not perpendicular!



$$\begin{bmatrix} .5 \\ -1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} .5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}$$

should be x-2y=0 (x = 2y)

Finding Correct Normal Transform

- transform a plane

$$P \qquad P' = MP$$
$$N \longrightarrow N' = QN$$

if we know M, what should Q be?

$N'^T P' = 0$ **stay perpendicular**

$(QN)^T (MP) = 0$ **substitute from above**

$N^T Q^T M P = 0$ $\qquad \mathbf{(AB)^T = B^T A^T}$

$N^T P = 0$ **true if** $Q^T M = I$

$$\boxed{Q = \left(M^{-1}\right)^T}$$

thus the normal to any surface has to be transformed by the inverse transpose of the modelling transformation

University of British Columbia

CPSC 414 Computer Graphics

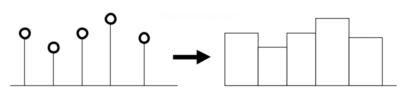
Sampling and Antialiasing

Samples

- most things in the real world are continuous
- everything in a computer is discrete
- the process of mapping a continuous function to a discrete one is called sampling
- the process of mapping a discrete function to a continuous one is called reconstruction
- the process of mapping a continuous variable to a discrete one is called quantization
- rendering an image requires sampling and quantization
- displaying an image involves reconstruction

Displays → Signal Reconstruction

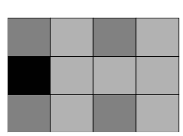
- All physical displays recreate a continuous image from a discrete sampled image by using a finite sized source of light for each pixel.



Imaging Devices Area Sample

- video camera : CCD array.

$$V = k \iint_{x,y} I\,dx\,dy$$

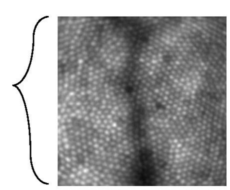


Imaging Devices Area Sample

- eye : photoreceptors

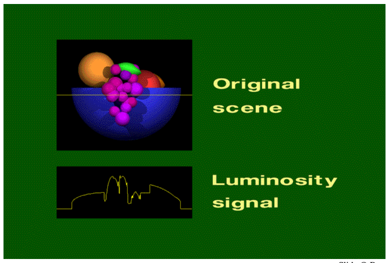
Film is similar : irregular array of receptors.

~ 0.47 deg.



J. Liang, D. R. Williams and D. Miller, "Supernormal vision and high- resolution retinal imaging through adaptive optics," J. Opt. Soc. Am. A 14, 2884- 2892 (1997)

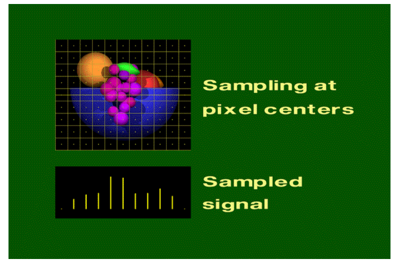
Continuous Luminosity Signal



Slide © Rosalee Nerheim-Wolfe

Week 6, Fri 10 Oct 03 © Tamara Munzner 13

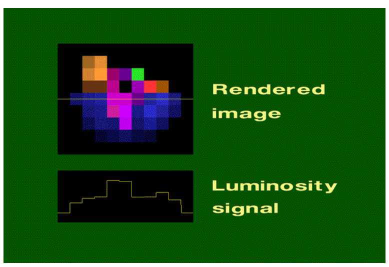
Sampled Luminosity



Slide © Rosalee Nerheim-Wolfe

Week 6, Fri 10 Oct 03 © Tamara Munzner 14

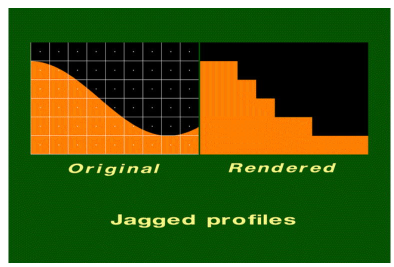
Reconstructed Luminosity



Slide © Rosalee Nerheim-Wolfe

Week 6, Fri 10 Oct 03 © Tamara Munzner 15

Reconstruction Artefact

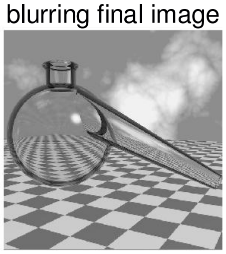


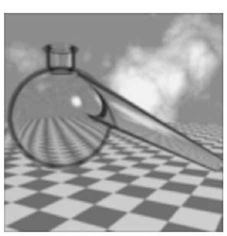
Slide © Rosalee Nerheim-Wolfe

Week 6, Fri 10 Oct 03 © Tamara Munzner 16

Bad Solution for Jaggies

- blurring final image

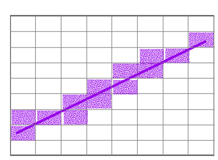




Week 6, Fri 10 Oct 03 © Tamara Munzner 17

Line Segments

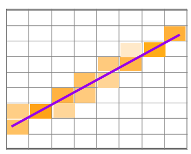
- we tried to sample a line segment so it would map to a 2D raster display
- we quantized the pixel values to 0 or 1
- we saw stair steps, or jaggies



Week 6, Fri 10 Oct 03 © Tamara Munzner 18

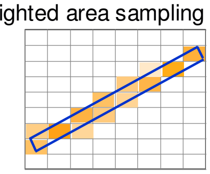
Line Segments

- instead, quantize to many shades
- but what sampling algorithm is used?



Area Sampling

- shade pixels according to the area covered by thickened line
- this is unweighted area sampling



- a rough approximation formulated by dividing each pixel into a finer grid of pixels

Unweighted Area Sampling

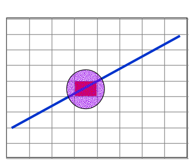
- primitive cannot affect intensity of pixel if it does not intersect the pixel
- equal areas cause equal intensity, regardless of distance from pixel center to area

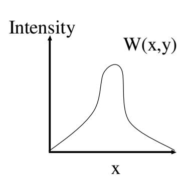
Weighted Area Sampling

- unweighted sampling colors two pixels identically when the primitive cuts the same area through the two pixels
- intuitively, pixel cut through the center should be more heavily weighted than one cut along corner

Weighted Area Sampling

- weighting function, W(x,y)
  - specifies the contribution of primitive passing through the point (x, y) from pixel center





Images

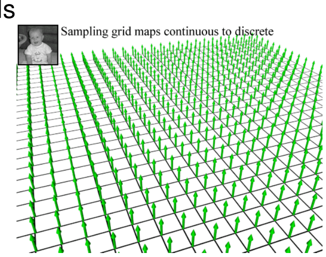
- an image is a 2D function I(x, y) that specifies intensity for each point (x, y)

Sampling and Image

- our goal is to convert the continuous image to a discrete set of samples
- the graphics system's display hardware will attempt to reconvert the samples into a continuous image: *reconstruction*

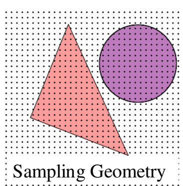
Point Sampling an Image

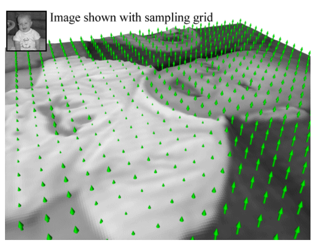
- simplest sampling is on a grid
- sample depends solely on value at grid points



Point Sampling

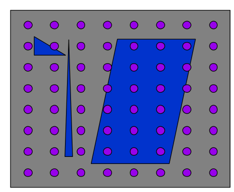
- multiply sample grid by image intensity to obtain a discrete set of points, or samples.





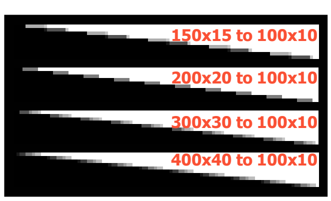
Sampling Errors

- some objects missed entirely, others poorly sampled



Fixing Sampling Errors

- *supersampling*
  - take more than one sample for each pixel and combine them
    - how many samples is enough?
    - how do we know no features are lost?

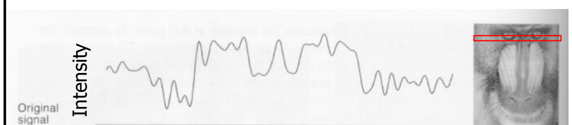


Spectral/Fourier Analysis

- spectral representation treats the function as a weighted sum of sines and cosines
- every function has two representations
  - spatial (time) domain - normal representation
  - frequency domain - spectral representation
- Fourier transform converts between the spatial and frequency domains.

Spatial Domain

- image as spatial signal



Pixel position across scanline

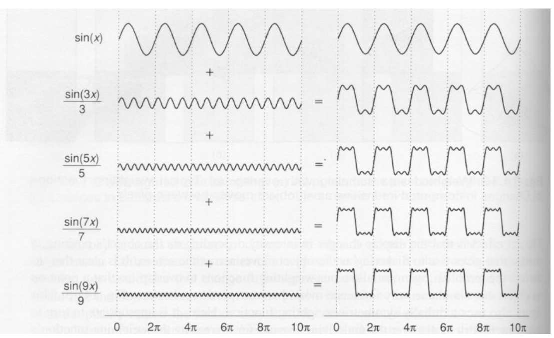
Examples from Foley, van Dam, Feiner, and Hughes

Spatial Frequency

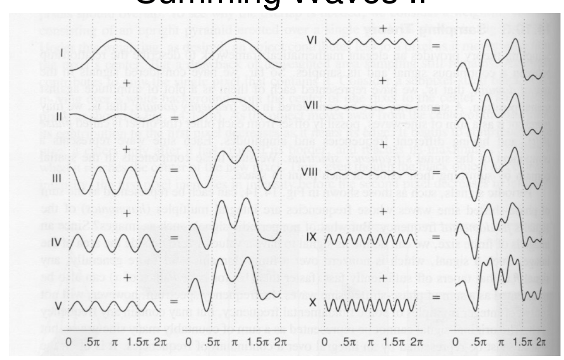
- in time - cycles per second
- in space - cycles per meter, degree, etc.

- Fourier view: sum of signals
  - pick frequency, phase shift
  - familiar example: sound spectrum

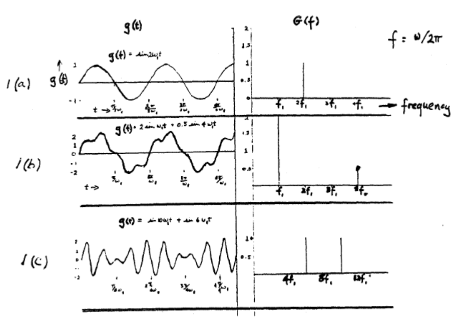
Summing Waves I



Summing Waves II

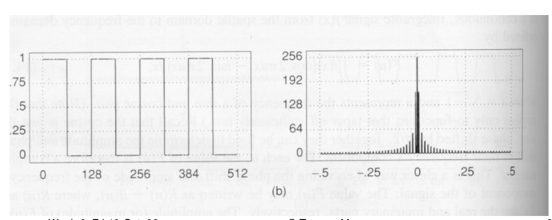


Waves as Frequencies



Frequency Domain

- height represents strength of each frequency
  - sine wave: impulse
  - square wave: infinite train of impulses



Spectral/Fourier Analysis

- *Fourier transform* converts between the spatial and frequency domain

Spatial domain

$$F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx$$

$$f(x) = \frac{1}{2\Pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega x} d\omega$$

Frequency domain.

- Euler formula : $e^{it} = \cos t + i \sin t$
  - real and imaginary components
- forward and reverse transforms very similar
  - reversal in sign of imaginary component, scale constant

Fourier Analysis

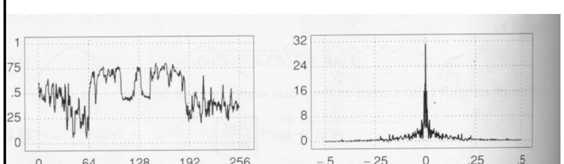
- convert spatial domain to frequency domain

$$F(u) = \int_{-\infty}^{+\infty} f(x)[\cos 2\pi ux - i \sin 2\pi ux] dx,$$

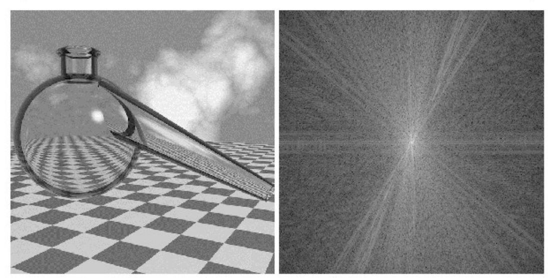
- let *f(x)* indicate the intensity at a location in space, *x (pixel value)*
- u is a complex number representing frequency and phase shift
  - i = sqrt (-1) … frequently not plotted
- F(u) is the amplitude of a particular frequency in a signal
  - in this case the signal is *f(x)*

Fourier Transform Example

spatial domain frequency domain



Spatial and Frequency Domain



Spatial Domain Frequency Domain

CS248 Fall 98 Lecture 6 Copyright © Pat Hanrahan

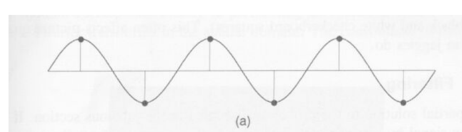
Sampling Theorem

the ideal samples of a continuous function contain all the information in the original function if and only if the continuous function is sampled at a frequency greater than twice the highest frequency in the function

- Claude Shannon

Nyquist Rate

- the lower bound on the sampling rate equals twice the highest frequency component in the image's spectrum
- this lower bound is the Nyquist Rate



Falling Below Nyquist Rate

- when sampling below Nyquist Rate, resulting signal looks like a lower-frequency one
  - this is aliasing!

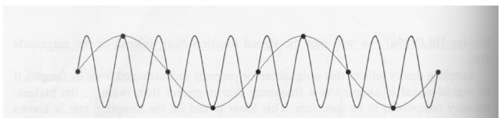


Fig. 14.17 Sampling below the Nyquist rate. (Courtesy of George Wolberg, Columbia University.)

Band-limited Signals

- if you know a function contains no components of frequencies higher than *x*
  - band-limited implies original function will not require any ideal functions with frequencies greater than x
  - facilitates reconstruction
  - avoids Nyquist Limit mistakes

Falling Below Nyquist Rate

- when sampling below Nyquist Rate, resulting signal looks like a lower-frequency one
  - safe with band-limits, guarantee that samples are not derived from signal of higher frequency

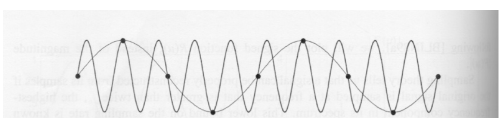


Fig. 14.17 Sampling below the Nyquist rate. (Courtesy of George Wolberg, Columbia University.)

Flaws with Nyquist Rate

- samples may not align with peaks

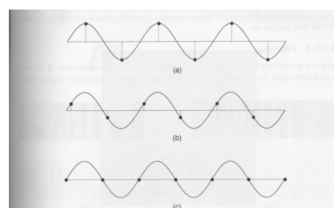
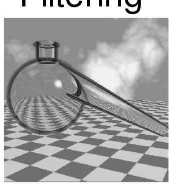
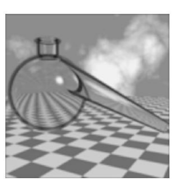


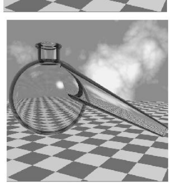
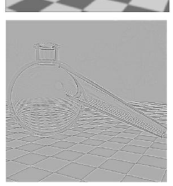
Fig. 14.16 Sampling at the Nyquist rate (a) at peaks, (b) between peaks, (c) at zero crossings. (Courtesy of George Wolberg, Columbia University.)

Filtering

- low pass

- high pass

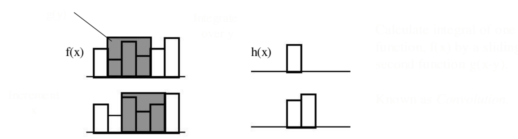



Filtering

- to lower Nyquist rate, remove high frequencies from image: *low-pass filter*
  - only low frequencies remain: band-limited

Filtering in Space Domain

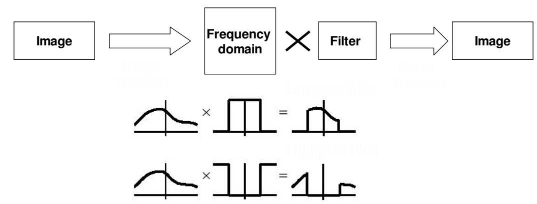
- blurring or averaging pixels together.



$$h(x) = f \otimes g = \int f(x) g(x-y) dy$$

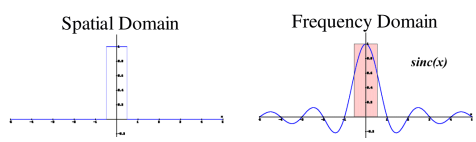
Filtering in Frequency Domain

- multiply signal's spectrum by pulse function



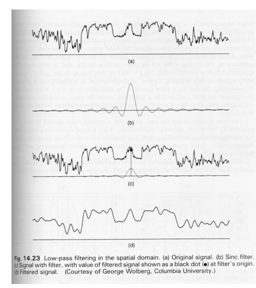
Filtering

- sinc (pulse) function is common filter:
  - sinc(x) = sin (πx)/πx



Sinc Filter

- Slide filter along spatial domain and compute new pixel value that results from convolution

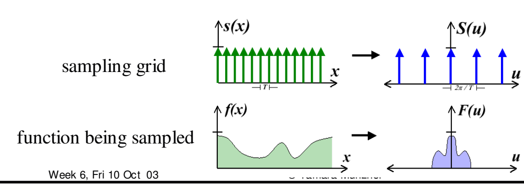


Convolution

- multiplying two Fourier Transforms (F(u)G(u)) in the frequency domain == convolution (represented as *) on their inverse Fourier transforms in the spatial domain
- f(x) * g(x) = h(x)
  - take the filter function, g(x) and center it at x
  - take a weighted average of f(x) in the neighborhood of x
    - weighting defined by g(x)

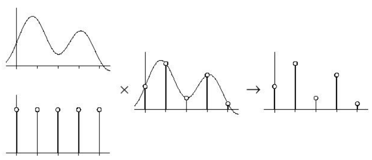
Sampling in Frequency Domain

- remember, sampling was defined as multiplying a grid of delta functions by the continuous image
- called a convolution in spatial domain



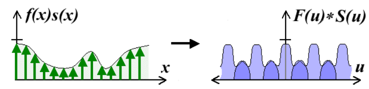
Sampling

- Multiplication of the sample with a regular train of delta functions.



Convolution

- This amounts to accumulating copies of the function’s spectrum sampled at the delta functions of the sampling grid

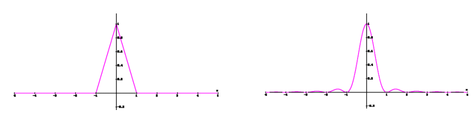


Convolution theorem.

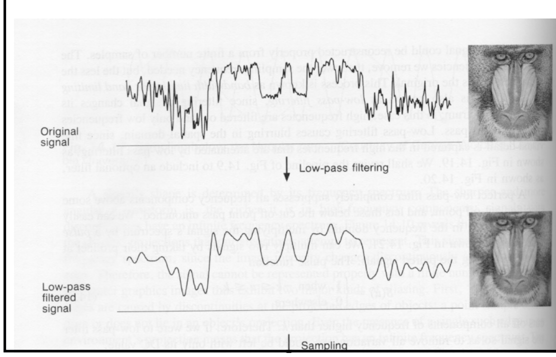
- *Theorem*: Multiplication in the frequency domain is equivalent to convolution in the space domain.
- *Symmetric Theorem*: Multiplication in the space domain is equivalent to convolution in the frequency domain.

Bilinear Filter

- sometimes called a tent filter
- easy to compute
  - just linearly interpolate between samples
- finite extent and no negative values
- still has artifacts



Sampling Pipeline



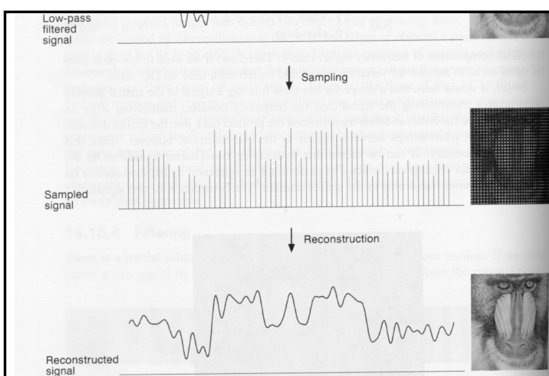


Fig. 14.20 The sampling pipeline with filtering. (Courtesy of George Wolberg, Columbia University.)