Sampling, Aliasing, & Mipmaps
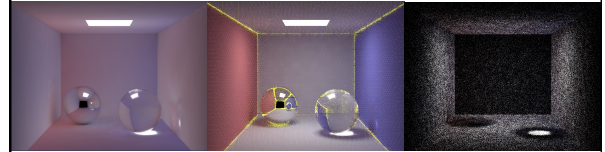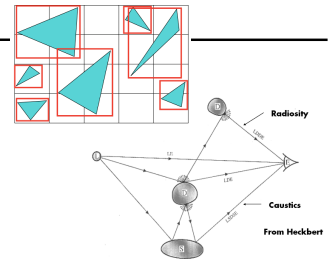
Last Time?
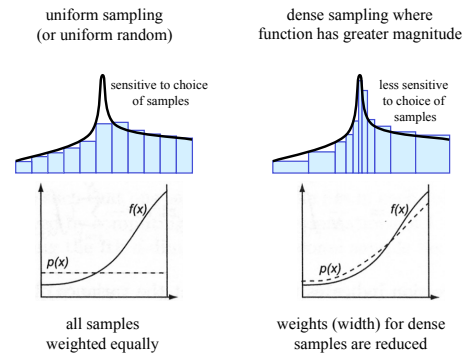
- Irradiance Caching
- Photon Mapping
- Acceleration Data Structures
- Ray Grammar

Radiosity

Caustics

From Heckbert

Today

- Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
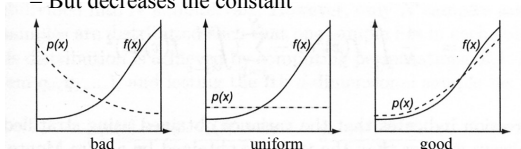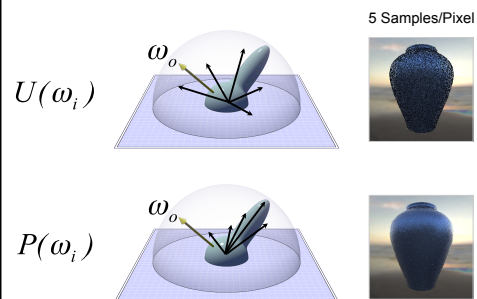- Anti-Aliasing for Texture Maps

Sampling

uniform sampling (or uniform random)

dense sampling where function has greater magnitude

sensitive to choice of samples

less sensitive to choice of samples

$f(x)$

$f(x)$

$p(x)$

$p(x)$

all samples weighted equally

weights (width) for dense samples are reduced

Importance Sampling

$$\langle I \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)}$$

- Choose p wisely to reduce variance
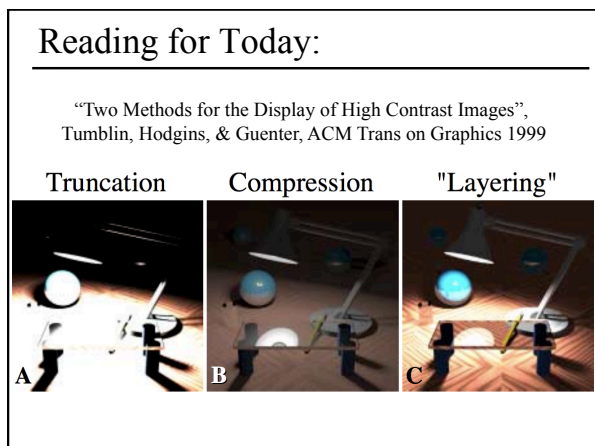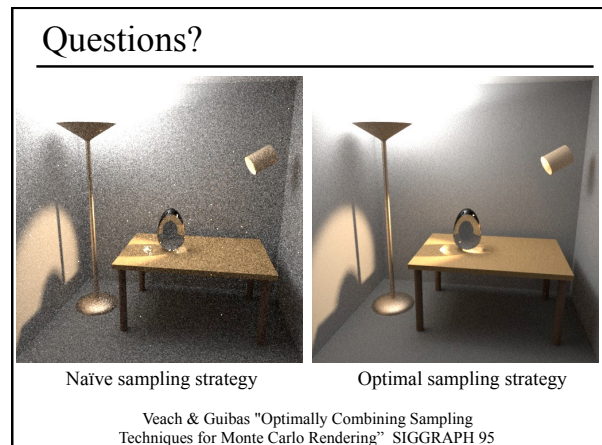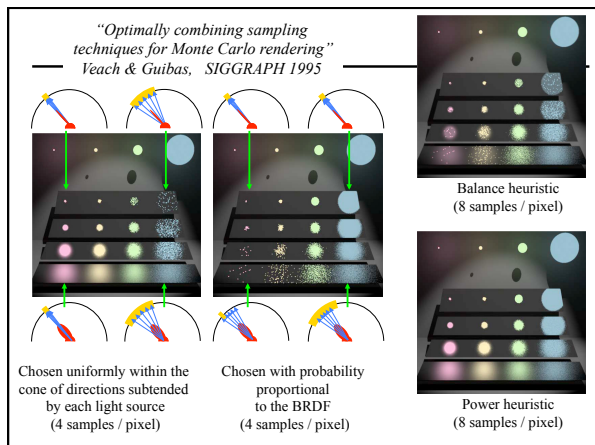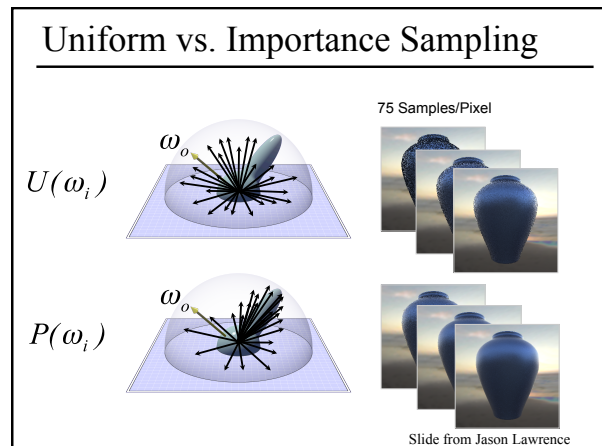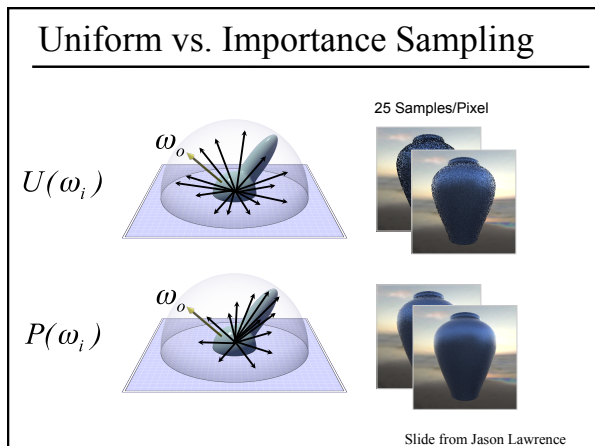 - Want to use a p that resembles f
  - Does not change convergence rate (still sqrt)
  - But decreases the constant

$p(x)$ $f(x)$

bad

$f(x)$ $p(x)$

uniform

$f(x)$ $p(x)$

good

Uniform vs. Importance Sampling

5 Samples/Pixel

ω_o

$U(\omega_i)$

ω_o

$P(\omega_i)$

Slide from Jason Lawrence

1

Uniform vs. Importance Sampling

25 Samples/Pixel

$U(\omega_i)$

ω_o

$P(\omega_i)$

ω_o

Slide from Jason Lawrence

Uniform vs. Importance Sampling

75 Samples/Pixel

$U(\omega_i)$

ω_o

$P(\omega_i)$

ω_o

Slide from Jason Lawrence

"Optimally combining sampling techniques for Monte Carlo rendering"
Veach & Guibas, SIGGRAPH 1995

Balance heuristic
(8 samples / pixel)

Chosen uniformly within the cone of directions subtended by each light source
(4 samples / pixel)

Chosen with probability proportional to the BRDF
(4 samples / pixel)

Power heuristic
(8 samples / pixel)

Questions?

Naïve sampling strategy   Optimal sampling strategy

Veach & Guibas "Optimally Combining Sampling
Techniques for Monte Carlo Rendering" SIGGRAPH 95

Reading for Today:

"Two Methods for the Display of High Contrast Images",
Tumblin, Hodgins, & Guenter, ACM Trans on Graphics 1999

Truncation   Compression   "Layering"

A   B   C

"Fast Bilateral Filtering for the Display of High-Dynamic
Range Images", Durand & Dorsey, SIGGRAPH 2002

"Fast Bilateral Filtering for the Display of High-Dynamic Range Images", Durand & Dorsey, SIGGRAPH 2002
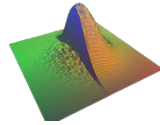
Before   After

The Secret:

Scene contrast is high
Display contrast is low

Contrast reduced
Details preserved

Bilateral filter

Today

- Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps

What is a Pixel?

- A pixel is not:
  - a box
  - a disk
  - a teeny tiny little light
- A pixel "looks different" on different display devices
- A pixel is a sample
  - it has no dimension
  - it occupies no area
  - it cannot be seen
  - it has a coordinate
  - it has a value

not a box!

not a circle!
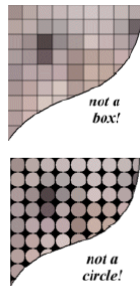
More on Samples

- Most things in the real world are *continuous*, yet 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 continuous variable to a discrete one is called *quantization*
- To represent or render an image using a computer, we must both sample and quantize

$f(x)$

$f[x]$

discrete value

x

discrete position

x
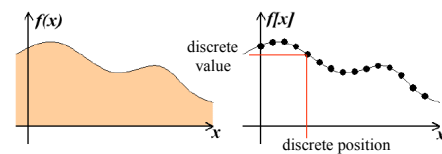
An Image is a 2D Function

- An *ideal image* is a continuous function I(x,y) of intensities.
- It can be plotted as a height field.
- In general an image cannot be represented as a continuous, analytic function.
- Instead we represent images as tabulated functions.
- How do we fill this table?

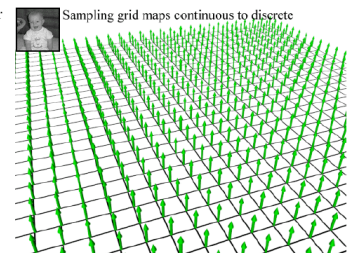An image seen as a continuous 2D function

Sampling Grid

- We can generate the table values by multiplying the continuous image function by a sampling grid of Kronecker delta functions.

The definiton of the 2-D Kronecker delta is:

Sampling grid maps continuous to discrete

$$\delta(x,y) = \begin{cases} 1, & (x,y) = (0,0) \\ 0, & \text{otherwise} \end{cases}$$

And a 2-D sampling grid:
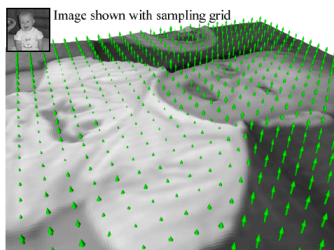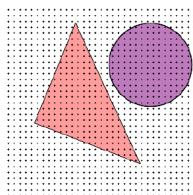
$$\sum_{j=0}^{h-1}\sum_{i=0}^{w-1} \delta(u-i, v-j)$$

Sampling an Image

- The result is a set of point samples, or pixels.

The same analysis can be applied to geometric objects:
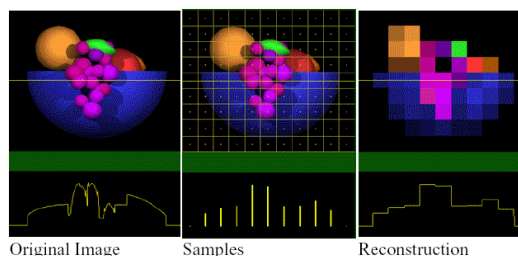
Image shown with sampling grid

Questions?

Today

- Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
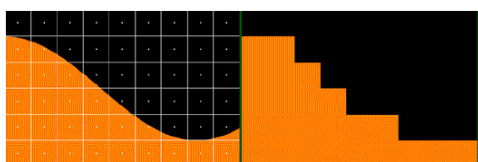- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps

Examples of Aliasing

Original Image   Samples   Reconstruction

- Aliasing occurs because of *sampling* and *reconstruction*

Examples of Aliasing

Jagged boundaries

Examples of Aliasing

Improperly rendered detail
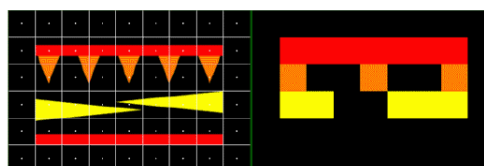
Examples of Aliasing
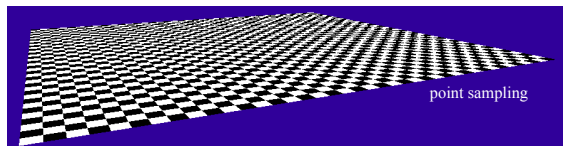
Texture Errors

point sampling

Questions?

Today

- Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
  - Sampling Density
  - Fourier Analysis & Convolution
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps

Sampling Density

- How densely must we sample an image in order to capture its essence?

- If we under-sample the signal, we won't be able to accurately reconstruct it...

Image shown with sampling grid
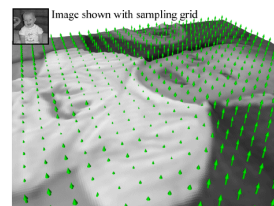
Sampling Density

- If we insufficiently sample the signal, it may be mistaken for something simpler during reconstruction (that's aliasing!)

Image from Robert L. Cook, "Stochastic Sampling and Distributed Ray Tracing", An Introduction to Ray Tracing, Andrew Glassner, ed., Academic Press Limited, 1989.
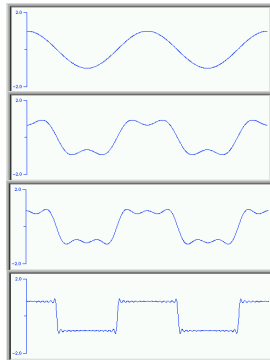
(a) Point sampling within the Nyquist limit

(b) Point sampling beyond the Nyquist limit
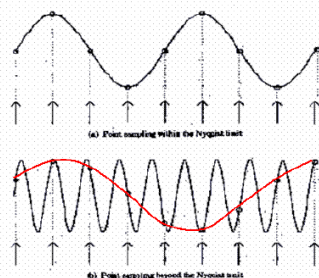
Sampling Density

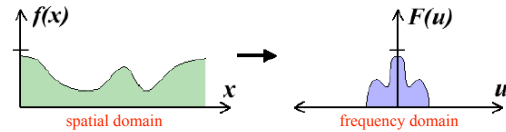- Aliasing in 2D because of insufficient sampling density
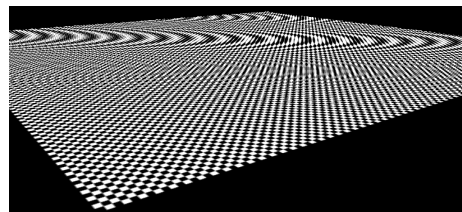
Remember Fourier Analysis?

- All periodic signals can be represented as a summation of sinusoidal waves.

Remember Fourier Analysis?

- Every periodic signal in the *spatial domain* has a dual in the *frequency domain*.

- This particular signal is *band-limited*, meaning it has no frequencies above some threshold
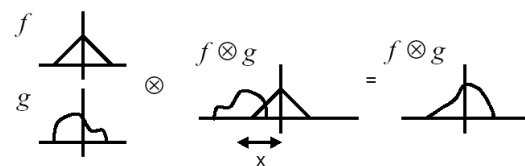
Remember Fourier Analysis?

- We can transform from one domain to the other using the Fourier Transform.

frequency domain   spatial domain

Fourier Transform
$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\, e^{-i2\pi(ux+vy)} dx dy$$

Inverse Fourier Transform
$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)\, e^{i2\pi(ux+vy)} du dv$$

Remember Convolution?

Convolution describes how a system with impulse response, *h(x)*, reacts to a signal, *f(x)*.

$$f(x) * h(x) = \int_{-\infty}^{\infty} f(\lambda) h(x - \lambda) d\lambda$$

f

g

$f \otimes g$ $f \otimes g$

$\otimes$ =

X

CS174 Fall 99 Lecture 7   Copyright © Mark Meyer

Images from Mark Meyer
http://www.gg.caltech.edu/~cs174ta/

Remember Convolution?

- Some operations that are difficult to compute in the spatial domain can be simplified by transforming to its dual representation in the frequency domain.
- For example, convolution in the spatial domain is the same as multiplication in the frequency domain.

$$f(x) * h(x) \rightarrow F(u)H(u)$$

- And, convolution in the frequency domain is the same as multiplication in the spatial domain
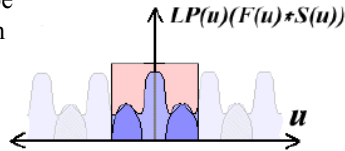
$$F(u) * H(u) \rightarrow f(x)h(x)$$

Sampling in the Frequency Domain
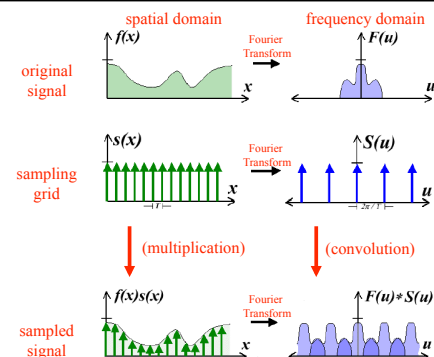
Reconstruction

- If we can extract a copy of the original signal from the frequency domain of the sampled signal, we can reconstruct the original signal!

- But there may be overlap between the copies.

$LP(u)(F(u) \ast S(u))$

u

Guaranteeing Proper Reconstruction

- Separate by removing high frequencies from the original signal (low pass pre-filtering)

$F(u)LP(u) \ast S(u)$

u

- Separate by increasing the sampling density

$f(x)s(x)$

x

$F(u) \ast S(u)$

u

- If we can't separate the copies, we will have overlapping frequency spectrum during reconstruction → *aliasing*.
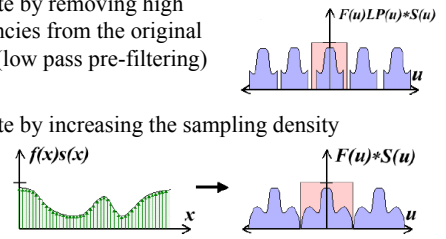
Sampling Theorem

- When sampling a signal at discrete intervals, the sampling frequency must be *greater than twice* the highest frequency of the input signal in order to be able to reconstruct the original perfectly from the sampled version (Shannon, Nyquist)

Questions?

Today

- Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
  - Ideal, Gaussian, Box, Bilinear, Bicubic
- Anti-Aliasing for Texture Maps

Filters

- Weighting function (convolution kernel)
- Area of influence often bigger than "pixel"
- Sum of weights = 1
  - Each sample contributes the same total to image
  - Constant brightness as object moves across the screen.
- No negative weights/colors (optional)

Source: Foley, VanDam, Feiner, Hughes - Computer Graphics, Second Edition,

7
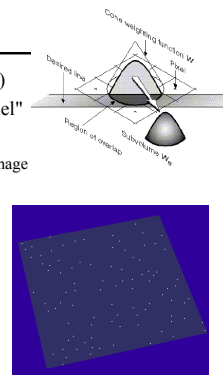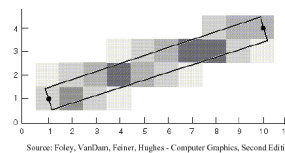
Filters

- Filters are used to
  - reconstruct a continuous signal from a sampled signal (reconstruction filters)
  - band-limit continuous signals to avoid aliasing during sampling (low-pass filters)
- Desired frequency domain properties are the same for both types of filters
- Often, the same filters are used as reconstruction and low-pass filters

The Ideal Filter

- Unfortunately it has *infinite* spatial extent
  - Every sample contributes to every interpolated point
- Expensive/impossible to compute

sinc(x)

spatial

$LP(u)(F(u)*S(u))$

u
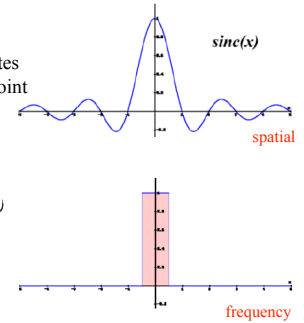
frequency

Problems with Practical Filters

- Many visible artifacts in re-sampled images are caused by poor reconstruction filters
- Excessive pass-band attenuation results in blurry images
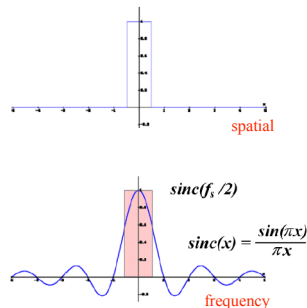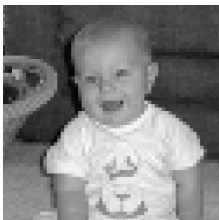- Excessive high-frequency leakage causes "ringing" and can accentuate the sampling grid (anisotropy)

"Pass-Band" Attenuation

High-frequency leakage

frequency
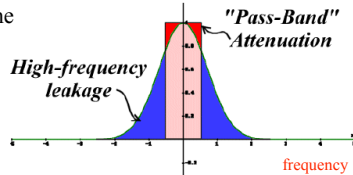
Gaussian Filter

- This is what a CRT does for free!

lp(x)

spatial

LP(u)

frequency
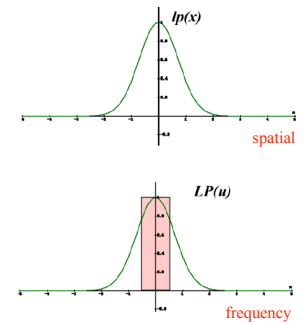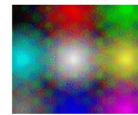
Box Filter / Nearest Neighbor

- Pretending pixels are little squares.

spatial

$sinc(f_s /2)$

$sinc(x) = \dfrac{sin(\pi x)}{\pi x}$

frequency

Tent Filter / Bi-Linear Interpolation

- Simple to implement
- Reasonably smooth

spatial

frequency

8
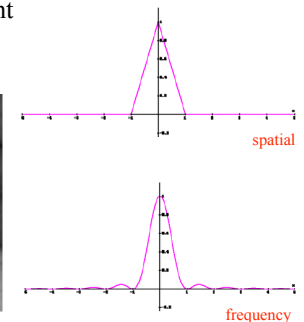
Bi-Cubic Interpolation

- Begins to approximate the ideal spatial filter, the sinc function
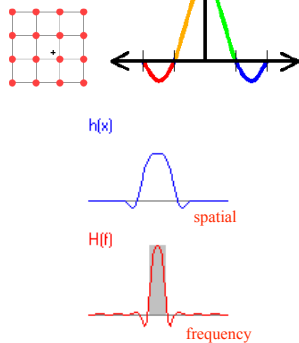
h(x)

spatial

H(f)

frequency

Questions?

Today

- Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
  - Magnification & Minification
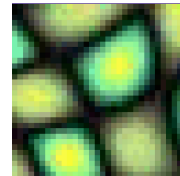 - Mipmaps
  - Anisotropic Mipmaps

Sampling Texture Maps

- When texture mapping it is rare that the screen-space sampling density matches the sampling density of the texture.

64x64 pixels

Original Texture

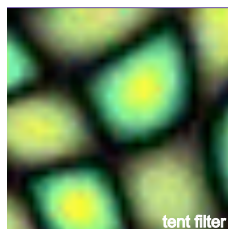Magnification for Display
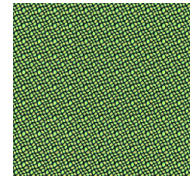
Minification for Display

for which we must use a reconstruction filter

Linear Interpolation

- Tell OpenGL to use a tent filter instead of a box filter.
- Magnification looks better, but blurry
  - (texture is under-sampled for this resolution)
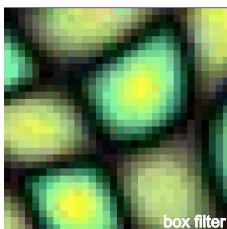
box filter

tent filter

Spatial Filtering

- Remove the high frequencies which cause artifacts in texture minification.
- Compute a spatial integration over the extent of the pixel
- This is equivalent to convolving the texture with a filter kernel centered at the sample (i.e., pixel center)!
- Expensive to do during rasterization, but an approximation it can be precomputed

projected texture in image plane

box filter in texture plane

MIP Mapping

- Construct a pyramid of images that are pre-filtered and re-sampled at 1/2, 1/4, 1/8, etc., of the original image's sampling

- During rasterization we compute the index of the decimated image that is sampled at a rate closest to the density of our desired sampling rate
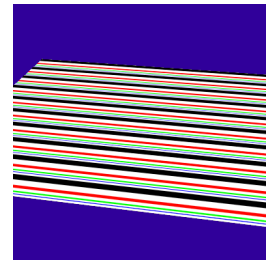- MIP stands for *multum in parvo* which means *many in a small place*
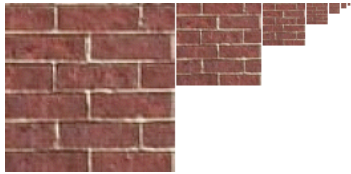
MIP Mapping Example

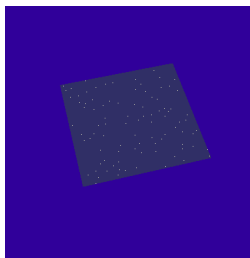- Thin lines may become disconnected / disappear
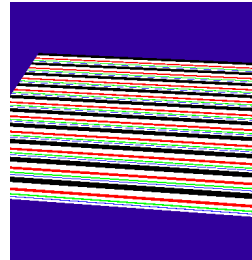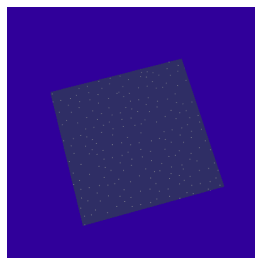

Nearest Neighbor   MIP Mapped (Bi-Linear)

MIP Mapping Example
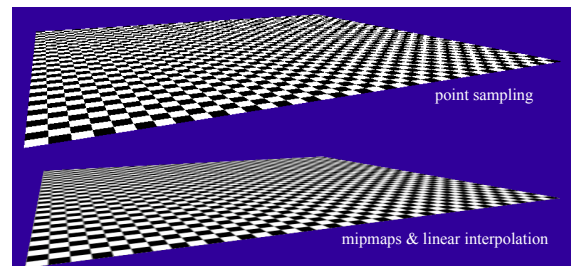
- Small details may "pop" in and out of view

Nearest Neighbor   MIP Mapped (Bi-Linear)

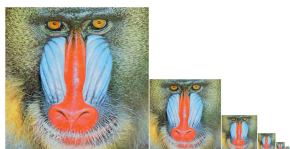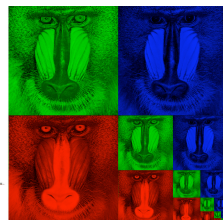## Examples of Aliasing

Texture Errors

point sampling

mipmaps & linear interpolation

Storing MIP Maps

- Can be stored compactly
- Illustrates the 1/3 overhead of maintaining the MIP map
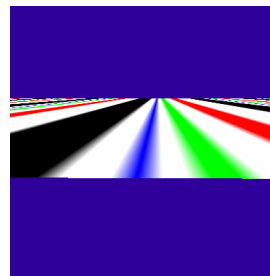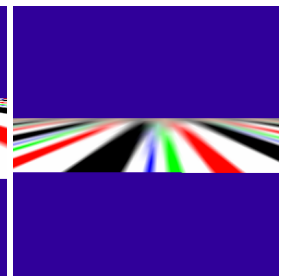

10-level mip map   Memory format of a mip map

Anisotropic MIP-Mapping

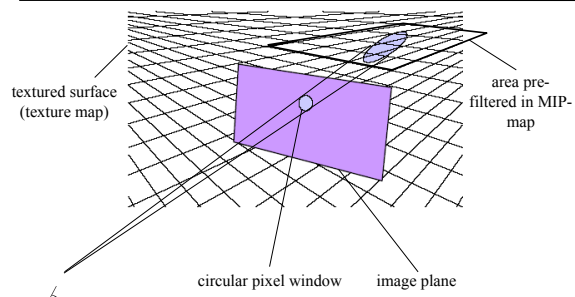- What happens when the surface is tilted?

Nearest Neighbor   MIP Mapped (Bi-Linear)

Anisotropic MIP-Mapping

textured surface
(texture map)

area pre-
filtered in MIP-
map

circular pixel window   image plane

• Square MIP-map area is a bad approximation

Anisotropic MIP-Mapping

• We can use different mipmaps for the 2 directions
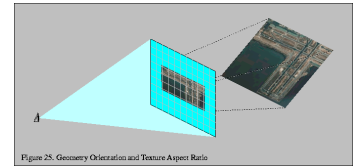
• Additional extensions can handle non axis-aligned views

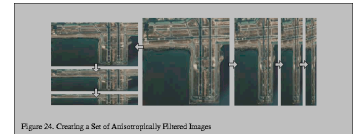Figure 25. Geometry Orientation and Texture Aspect Ratio

Figure 24. Creating a Set of Anisotropically Filtered Images

Images from http://www.sgi.com/software/opengl/advanced98/notes/node37.html

Questions?

Reading for Friday 4/2:

• "Radiance Caching for Participating Media", Jarosz, Donner, Zwicker, & Jensen, 2008.