CSCI 4530/6530 Advanced Computer Graphics

https://www.cs.rpi.edu/~cutler/classes/advancedgraphics/S25/

Lecture 15: Monte Carlo, Sampling, Aliasing, & Mipmaps

The Parthenon, Paul Debevec et al., 2004


2

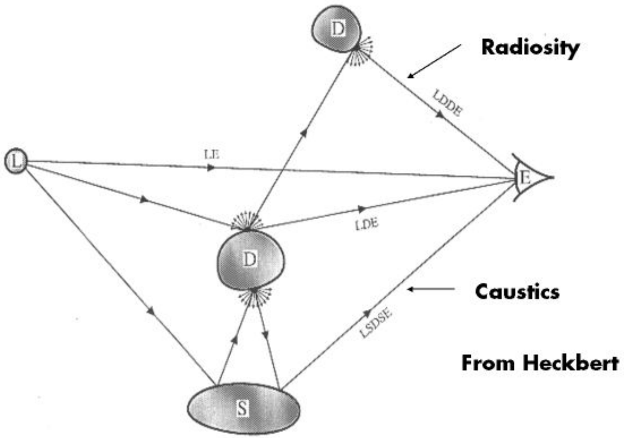

3

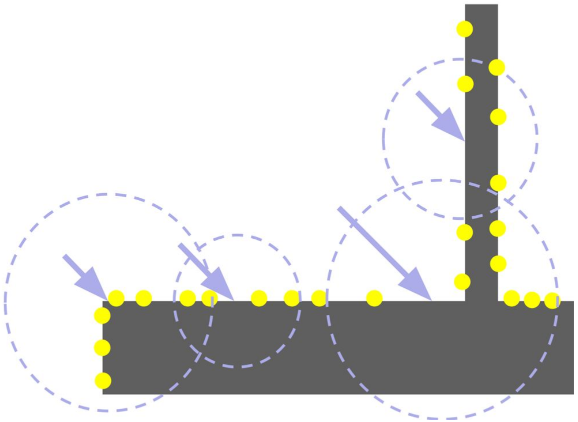

4

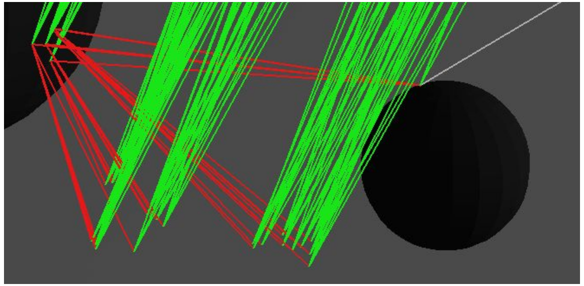
Worksheet: Photon Mapping

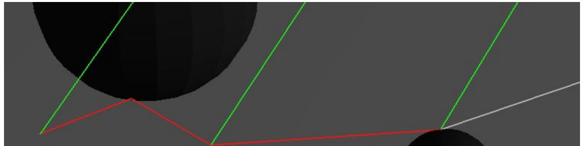
Last Time?

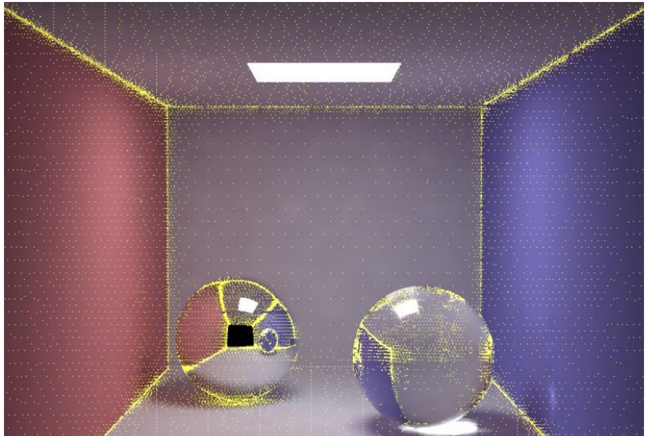
- Path Tracing vs. Ray Tracing
- Irradiance Caching
- Photon Mapping
- Ray Grammar











Notes & Announcements…

- **Final Project Brainstorming Part 2: Peer Feedback**
  Due Tuesday 3/11 @ 11:59pm
- Reply to 3 of your classmates' ideas posts!

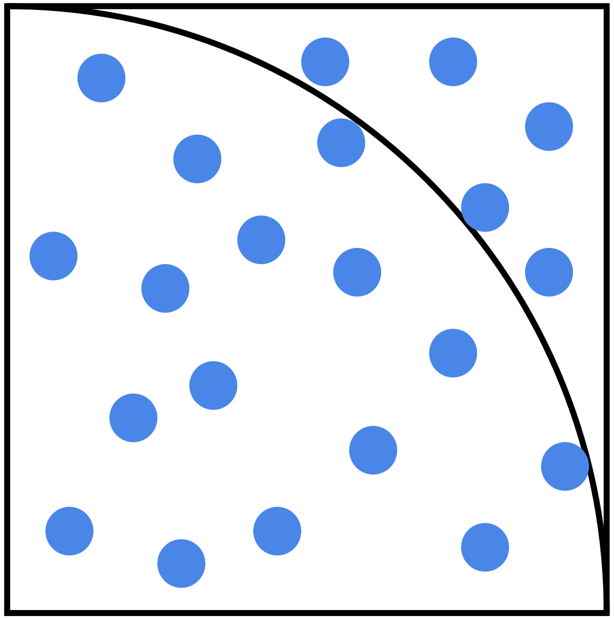
- Don't use C/C++: `abs`
  On linux, this is will cast to int
- You probably want: `fabs`

Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
  - Examples, Convergence, & Error
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

Monte Carlo Computation of π

- Take a random point (x,y) in unit square
- Test if it is inside the ¼ disc
  - Is $x^2 + y^2 < 1$?
- Probability of being inside disc?
  - area of ¼ unit circle / area of unit square = π /4



- π ≈ 4 * number inside disc / total number
- The error depends on the number of trials

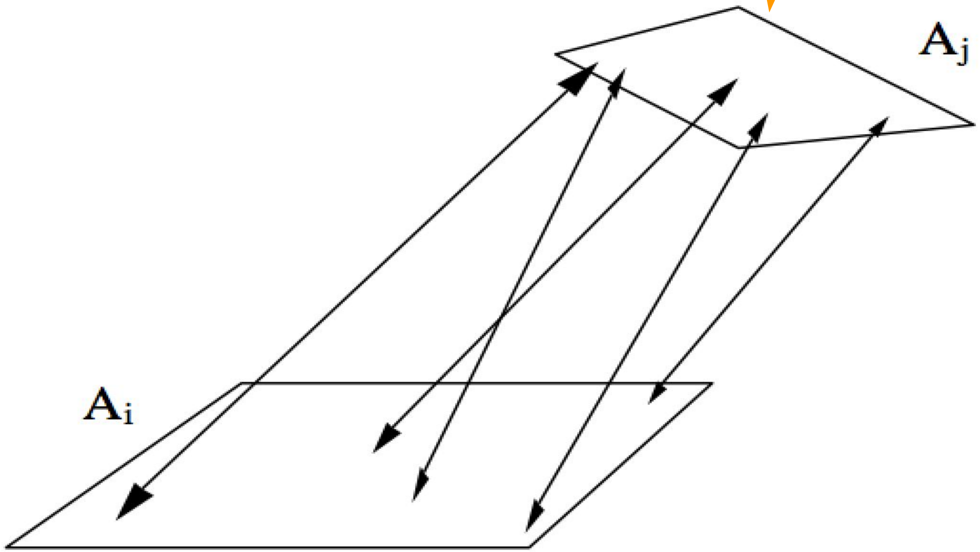
16/21 = 0.7619 ≈ π / 4 = 0.7854

π ≈ 3.1416

Use Monte Carlo to Calculate Form Factors

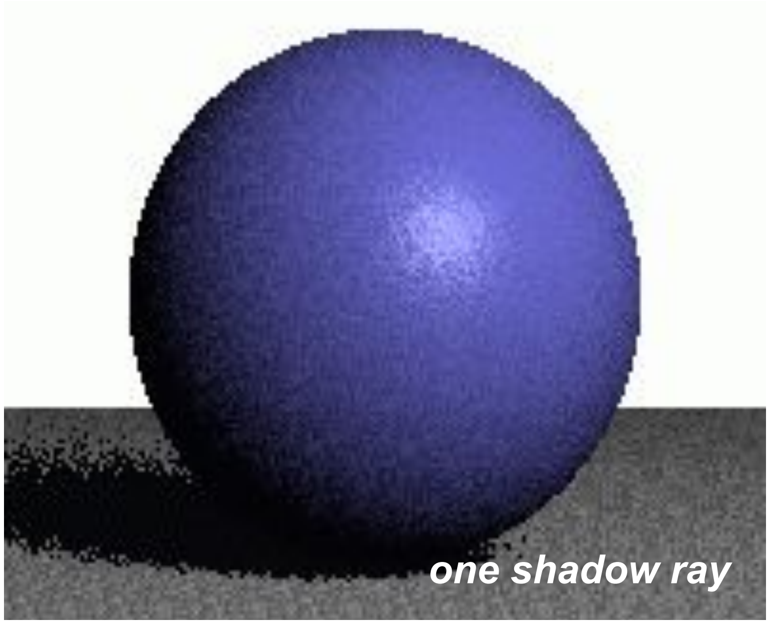
- Cast *n* rays between the two patches
  - Compute visibility (what fraction of rays do not hit an occluder)
  - Integrate the point-to-point form factor
- Permits the computation of the patch-to-patch form factor, as opposed to point-to-patch

Use this for HW3!



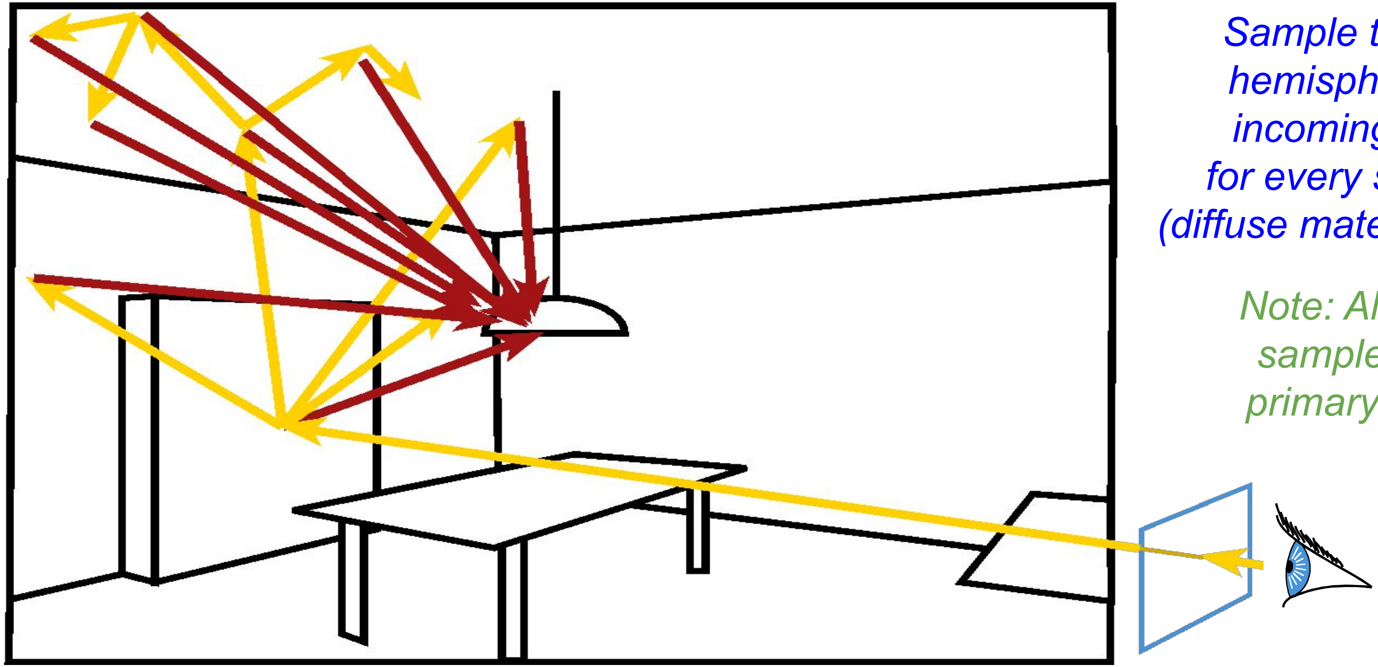
Monte Carlo for Distributed Ray Tracing

- Multiple shadow rays to sample area light source



Monte Carlo Ray Tracing

- Cast a ray from the eye through each pixel
- Cast *lots and lots* of random rays to accumulate radiance contribution
    - Recurse to solve the *full Rendering Equation*

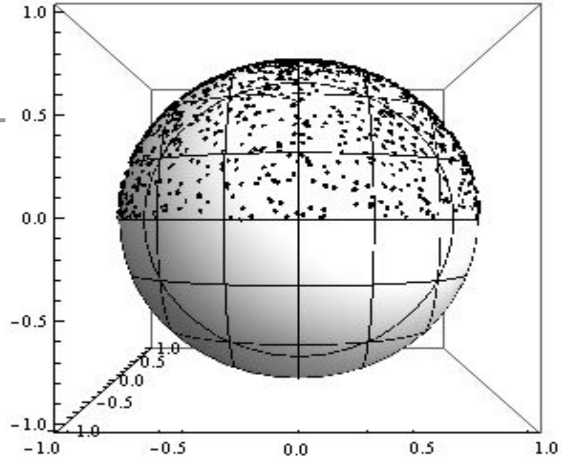


Sample the full hemisphere of incoming light for every surface (diffuse materials too!)

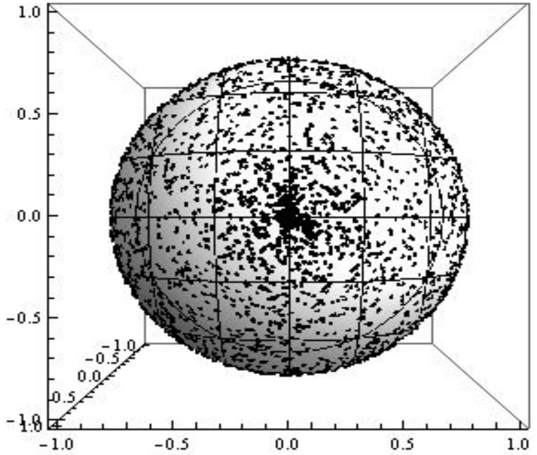
Note: Always sample the primary light

Domains of Integration

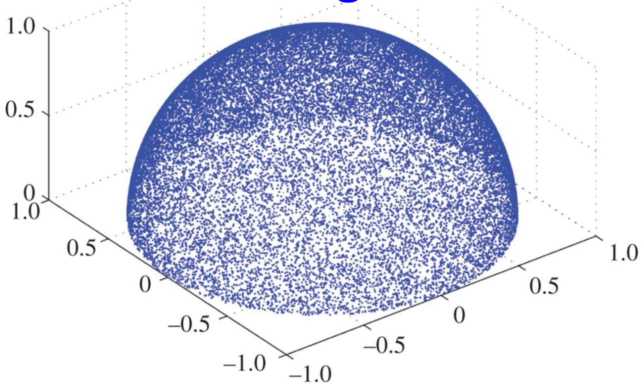
- Pixel, lens
  (Euclidean 2D domain)
- Time (1D)
- Hemisphere: Work needed to ensure *uniform* probability



This is oops…



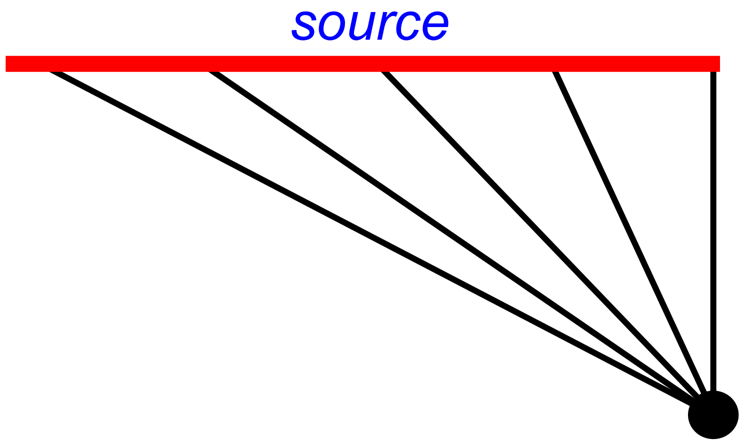
This is good!



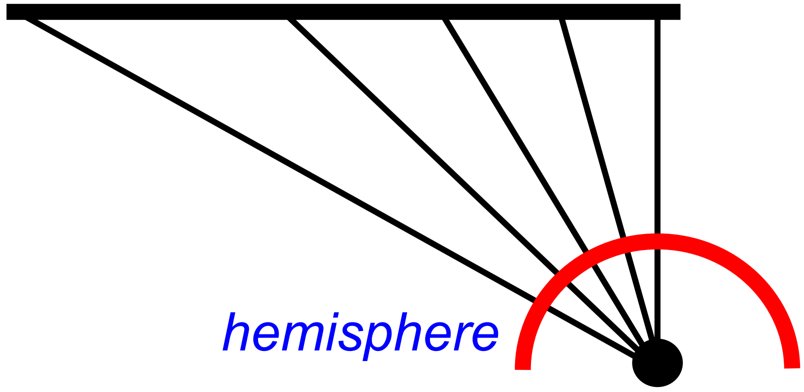
Example: Energy from an Area Light Source

- We can correctly integrate over surface *or* over angle
- But we must be careful to get probabilities and integration measure right!
  - It might require re-weighting/normalizing samples

Sampling the source uniformly



Sampling the hemisphere uniformly



Monte Carlo Convergence & Error

- Let’s use Monte Carlo to “compute 0.5” by flipping a coin:
  - 1 flip: 0 or 1
    → average error = 0.5
  - 2 flips: 0, 0.5, 0.5 or 1
    → average error = 0. 25
  - 4 flips: 0 (*1),0.25 (*4), 0.5 (*6), 0.75(*4), 1(*1)
    → average error = 0.1875

- Unfortunately, doubling the number of samples does not double the accuracy (a.k.a. *half the error*) of Monte Carlo approximations

Monte Carlo Integration

- Turn integral into finite sum
- Use *n* random samples
- As *n* increases…
  - Expected value remains the same
  - Variance decreases by *n*
  - Standard deviation (error) decreases by $\frac{1}{\sqrt{n}}$

- Thus, converges with $\frac{1}{\sqrt{n}}$

Advantages of Monte Carlo Integration

- Few restrictions on the integrand
  - Doesn't need to be continuous, smooth, ...
  - Only need to be able to evaluate at a point
- Extends to high-dimensional problems
  - Same convergence
- Conceptually straightforward
- Efficient for solving at just a few points

Disadvantages of Monte Carlo Integration

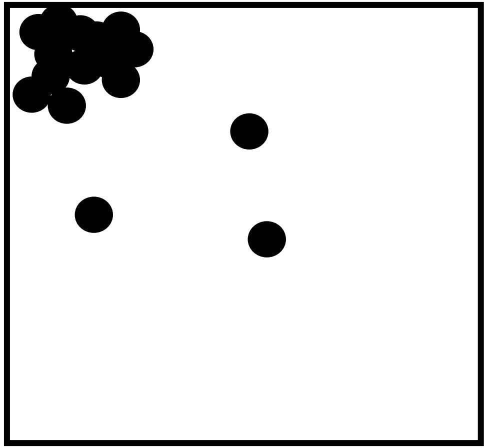
- Noisy
- Slow convergence
- Good implementation is hard
  - Debugging code
  - Debugging math
  - Choosing appropriate techniques
- Punctual technique, no notion of smoothness of function (e.g., between neighboring pixels)

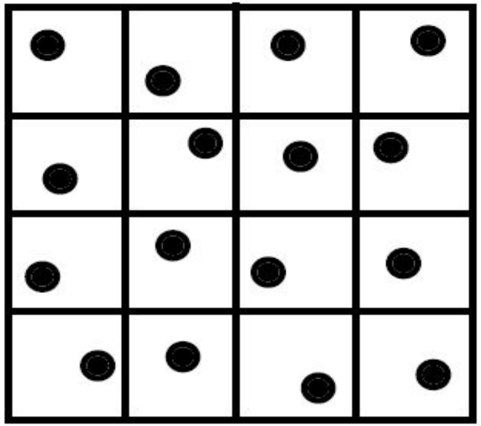
Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

Stratified Sampling

- With uniform sampling, we can get unlucky
  - E.g. all samples in a corner
- To prevent it, subdivide domain Ω into non-overlapping regions Ω_i
  - Each region is called a stratum
- Take one random samples per Ω_i





Stratified Sampling Convergence Comparison

$$f(x) = e^{\sin(3x^2)}$$

N	I
1	2.75039
10	1.9893
100	1.79139
1000	1.75146
10000	1.77313
100000	1.77862

Unstratified

$$O(1/\sqrt{N})$$

$$f(x) = e^{\sin(3x^2)}$$

N	I
1	2.70457
10	1.72858
100	1.77925
1000	1.77606
10000	1.77610
100000	1.77610

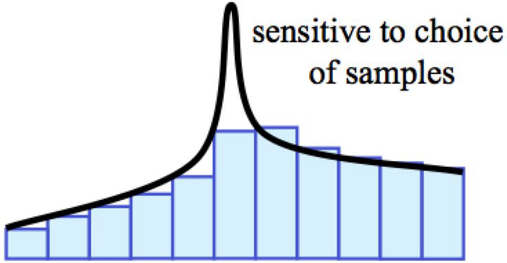
Stratified

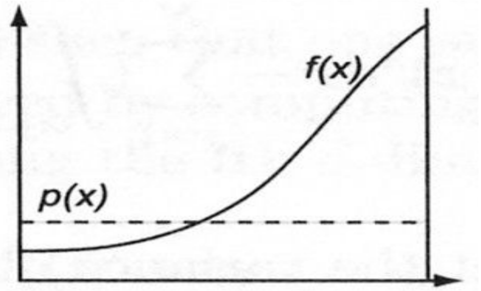
$$O(1/N)$$

Slide from Henrik Wann Jensen

Options: Uniform vs. Non-Uniform Sampling

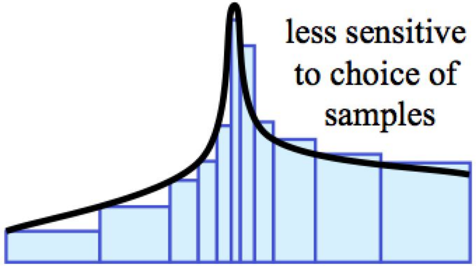
uniform sampling
(or uniform random)

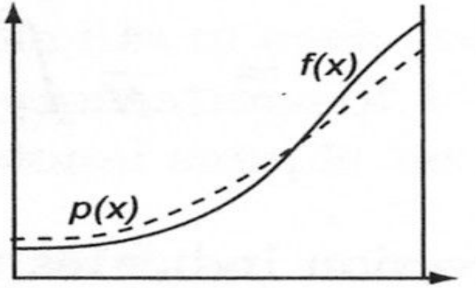




*all samples
weighted equally*

dense sampling where
function has greater magnitude



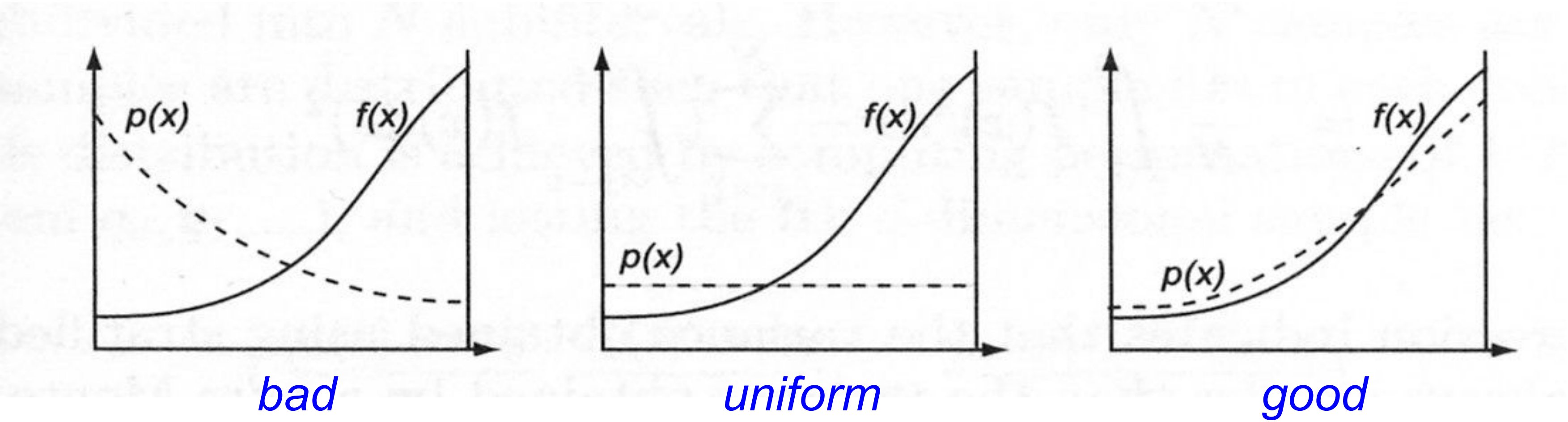


*weights (width) for dense
samples are reduced*

Importance Sampling

- 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

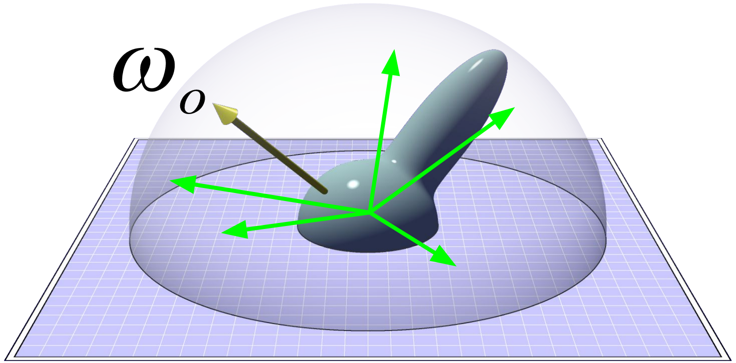
$$\langle I \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)}$$



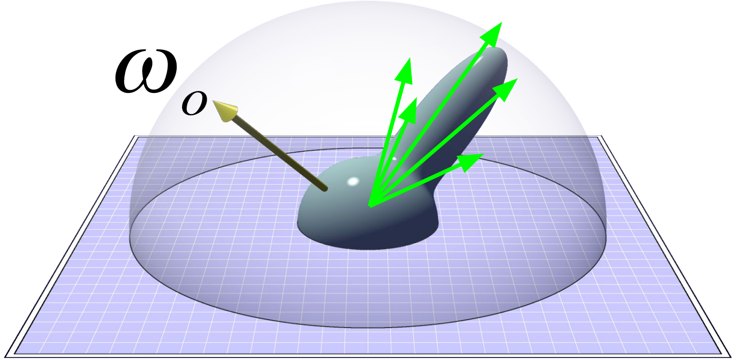
Uniform vs. Importance Sampling

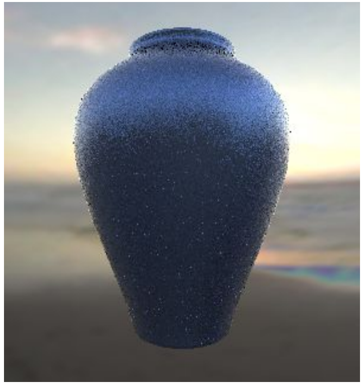
5 Samples/Pixel

$U(\omega_i)$



$P(\omega_i)$



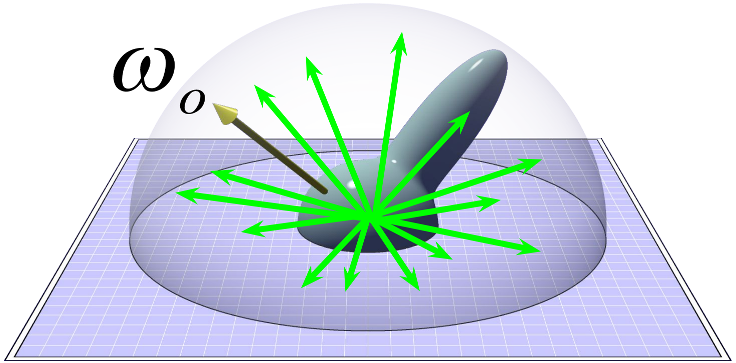


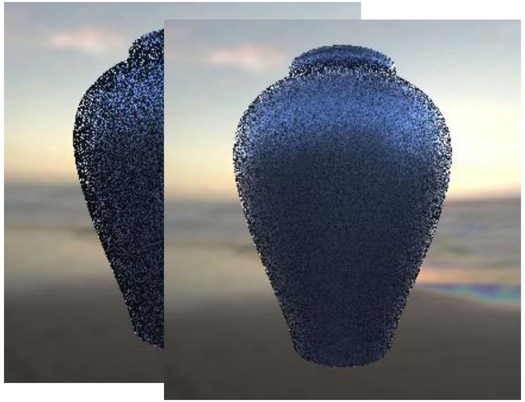
Slide from Jason Lawrence

Uniform vs. Importance Sampling

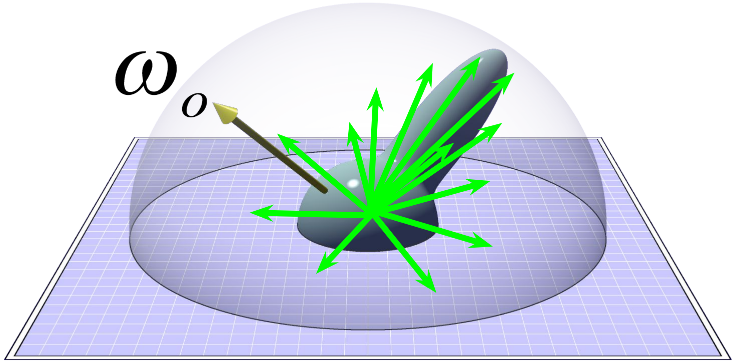
25 Samples/Pixel

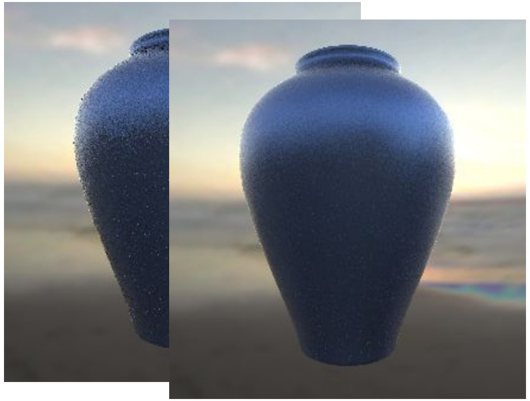
$U(\omega_i)$





$P(\omega_i)$



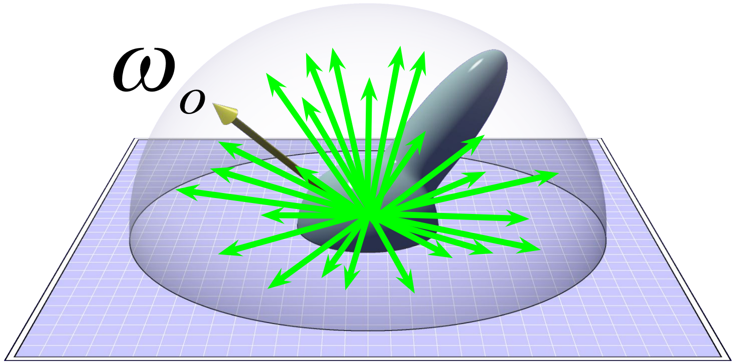


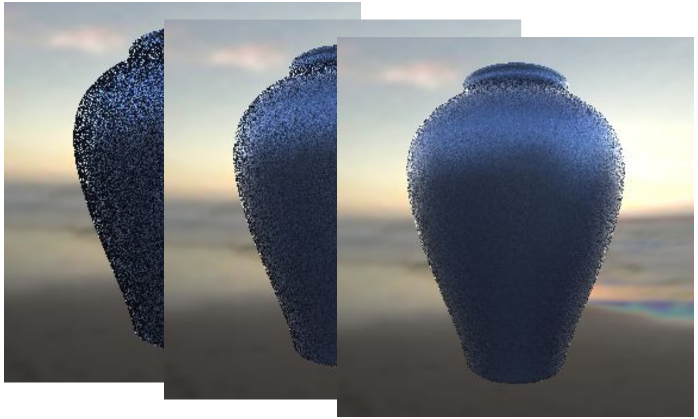
Slide from Jason Lawrence

Uniform vs. Importance Sampling

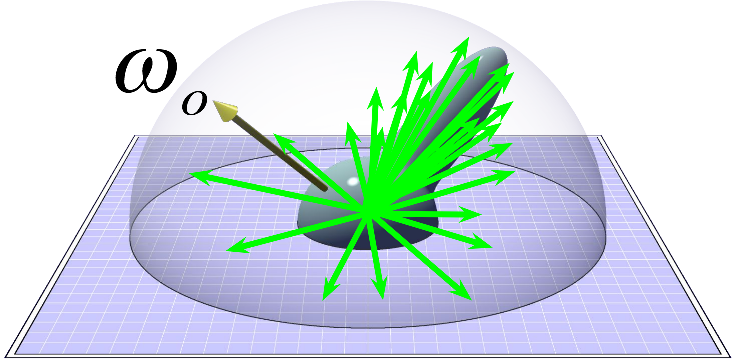
75 Samples/Pixel

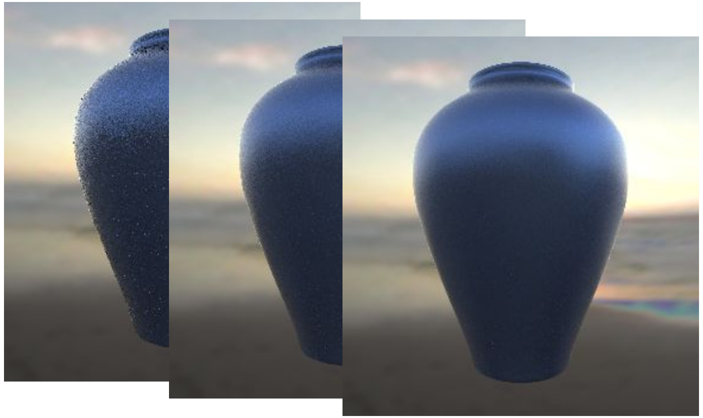
$$U(\omega_i)$$





$$P(\omega_i)$$





Slide from Jason Lawrence

Bidirectional Path Tracing

- "A Theoretical Framework for Physically Based Rendering", Lafortune and Willems, Computer Graphics Forum, 1994.

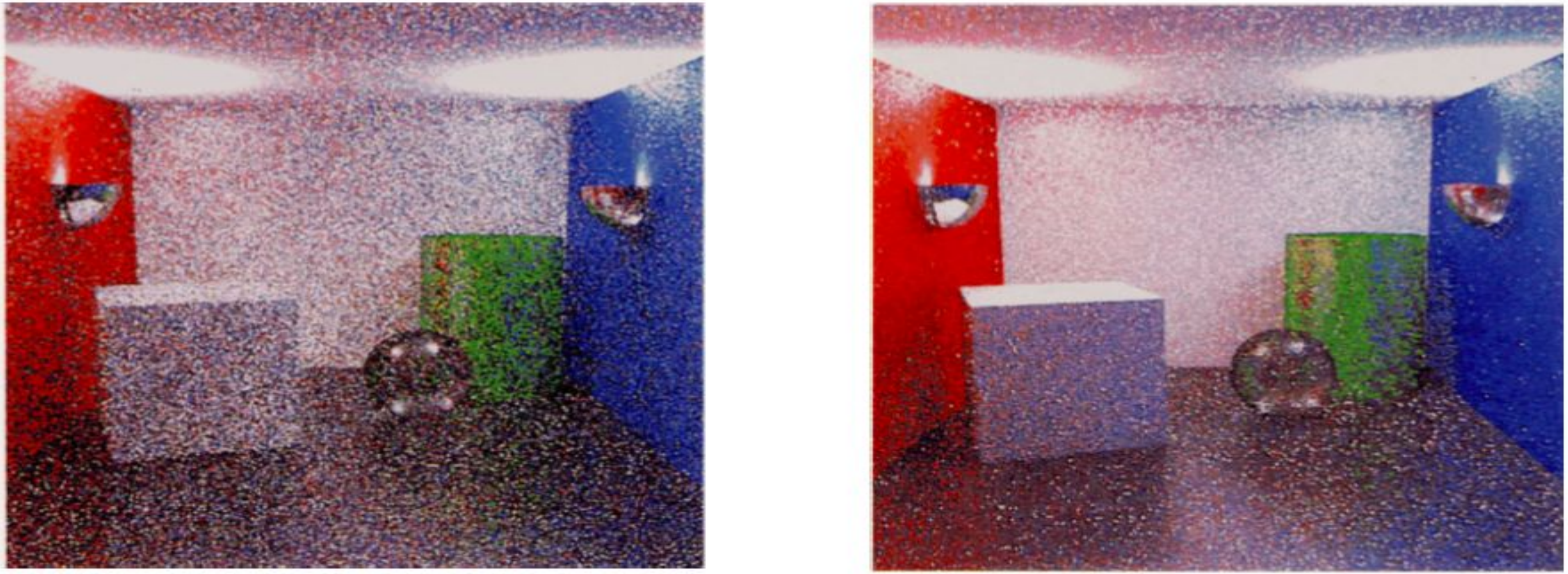


Figure B: *An indirectly illuminated scene rendered using path tracing and bidirectional path tracing respectively. The latter method results in visibly less noisefor the same amount of work.*

Questions?

Naïve sampling strategy

Optimal sampling strategy

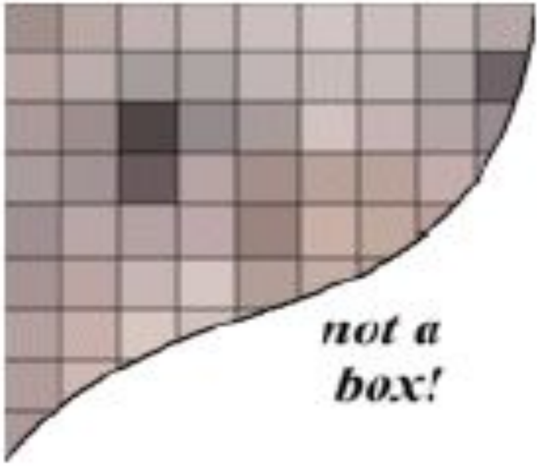
Veach & Guibas "Optimally Combining Sampling Techniques for Monte Carlo Rendering" SIGGRAPH 95

Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

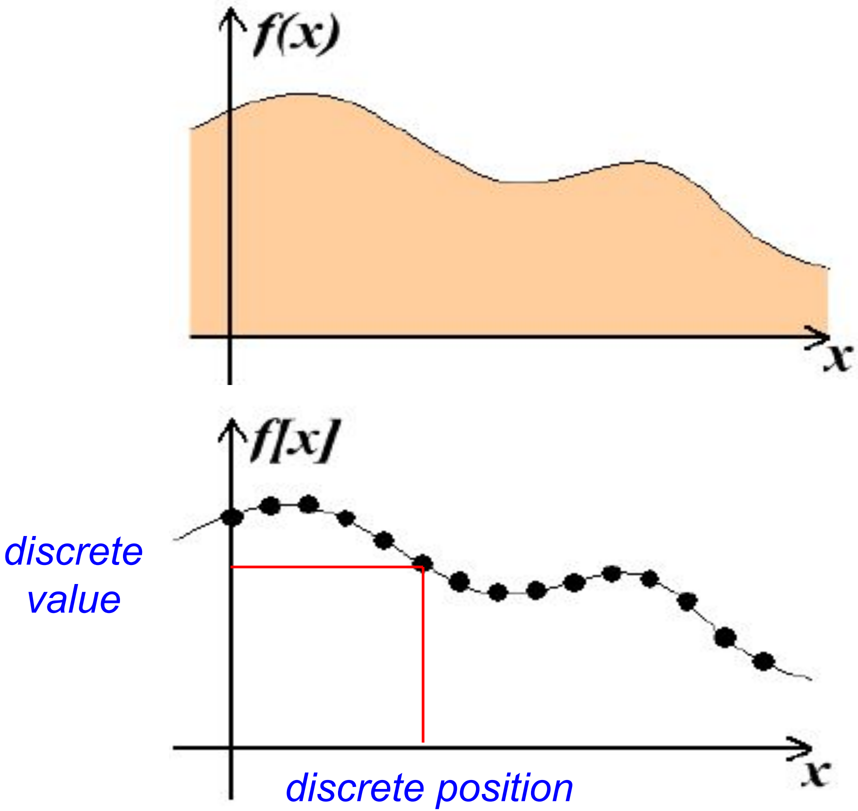
What is a Pixel?

- 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



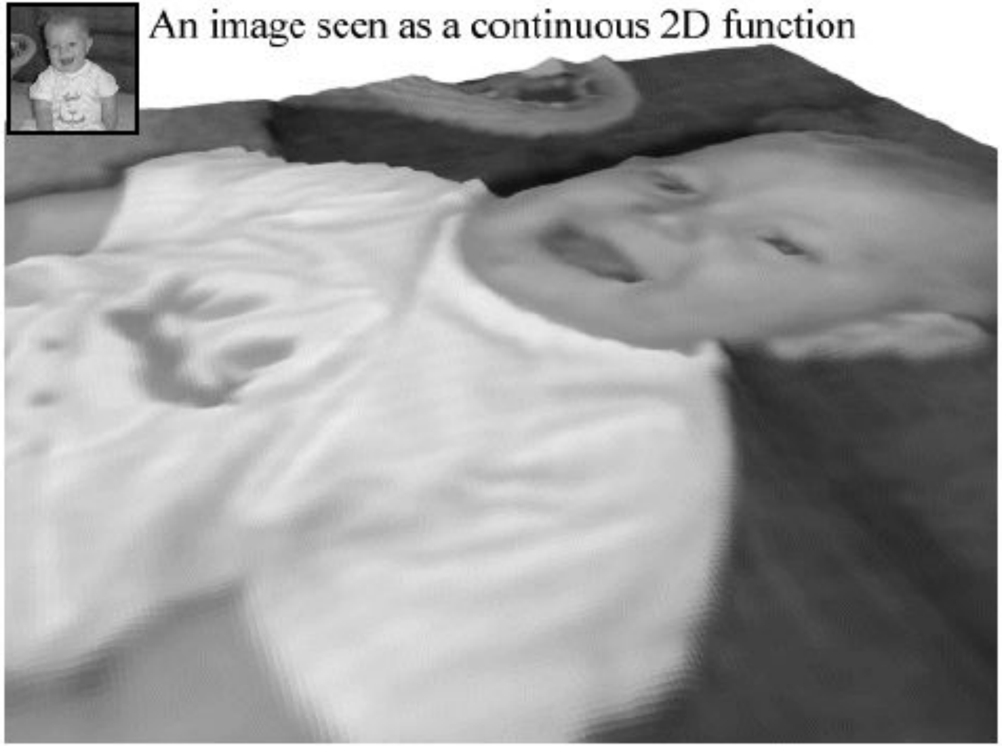
How & What do we Sample?

- Most things in the real world are *continuous*, yet everything in a computer is *discrete*
- Mapping a continuous function to a discrete one is called *sampling*
- 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



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?



Sampling Grid

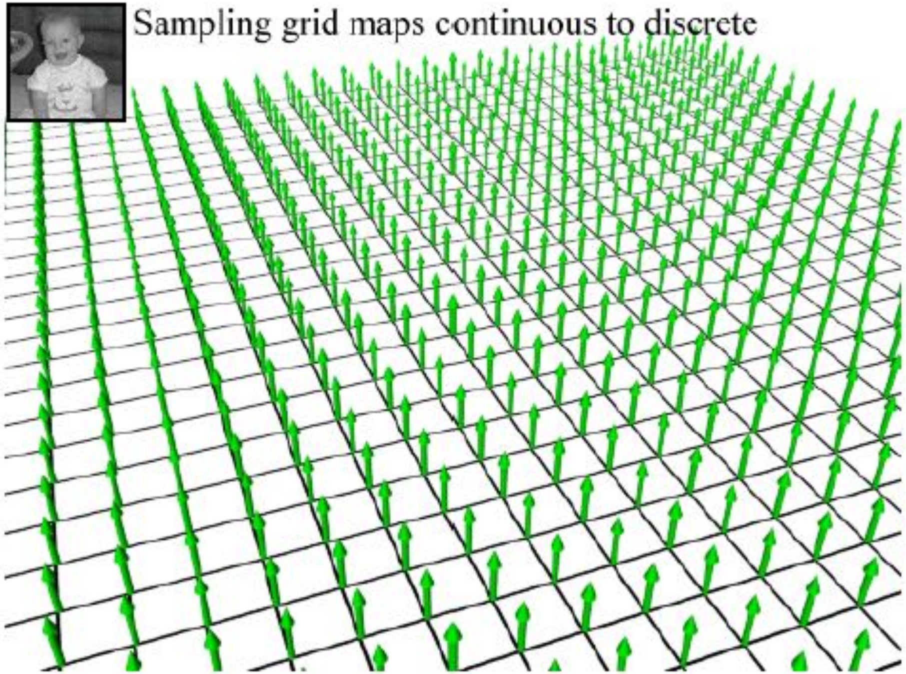
- We can generate the table values by multiplying the continuous image function by a sampling grid of Kronecker delta functions.

The definiton of the 2-D Kronecker delta is:

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

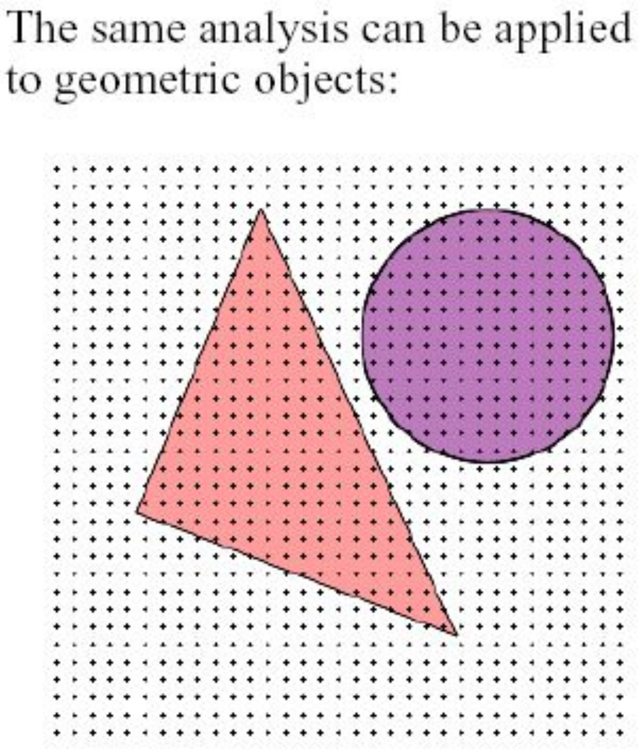
And a 2-D sampling grid:

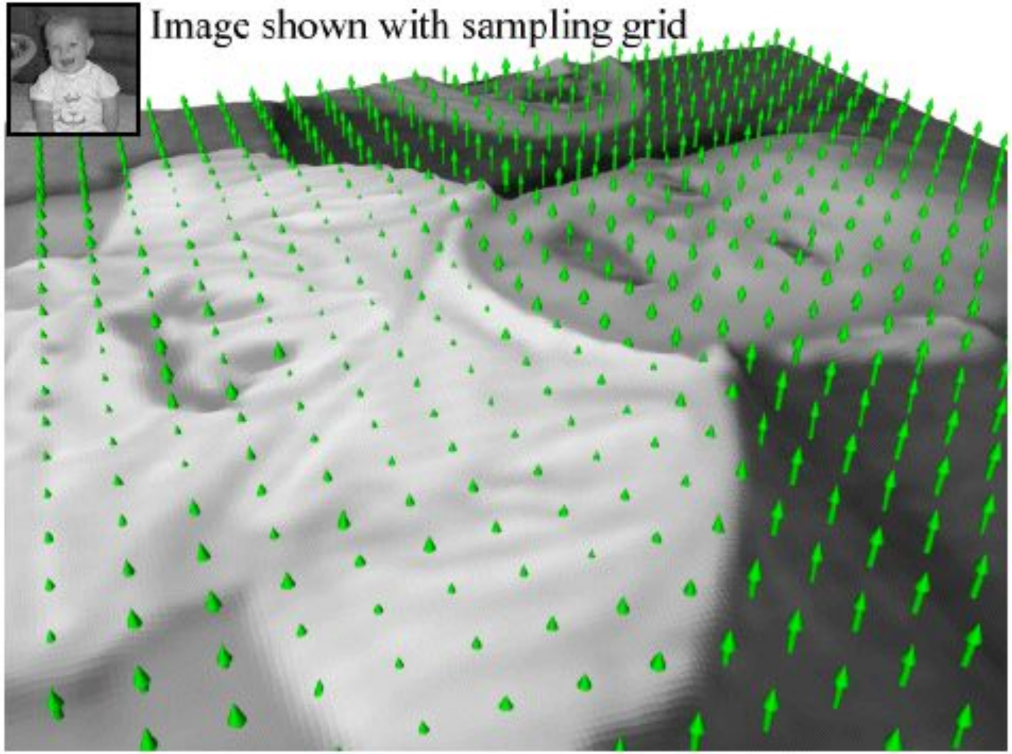
$$\sum_{j=0}^{h-1} \sum_{i=0}^{w-1} \delta(u-i, v-j)$$



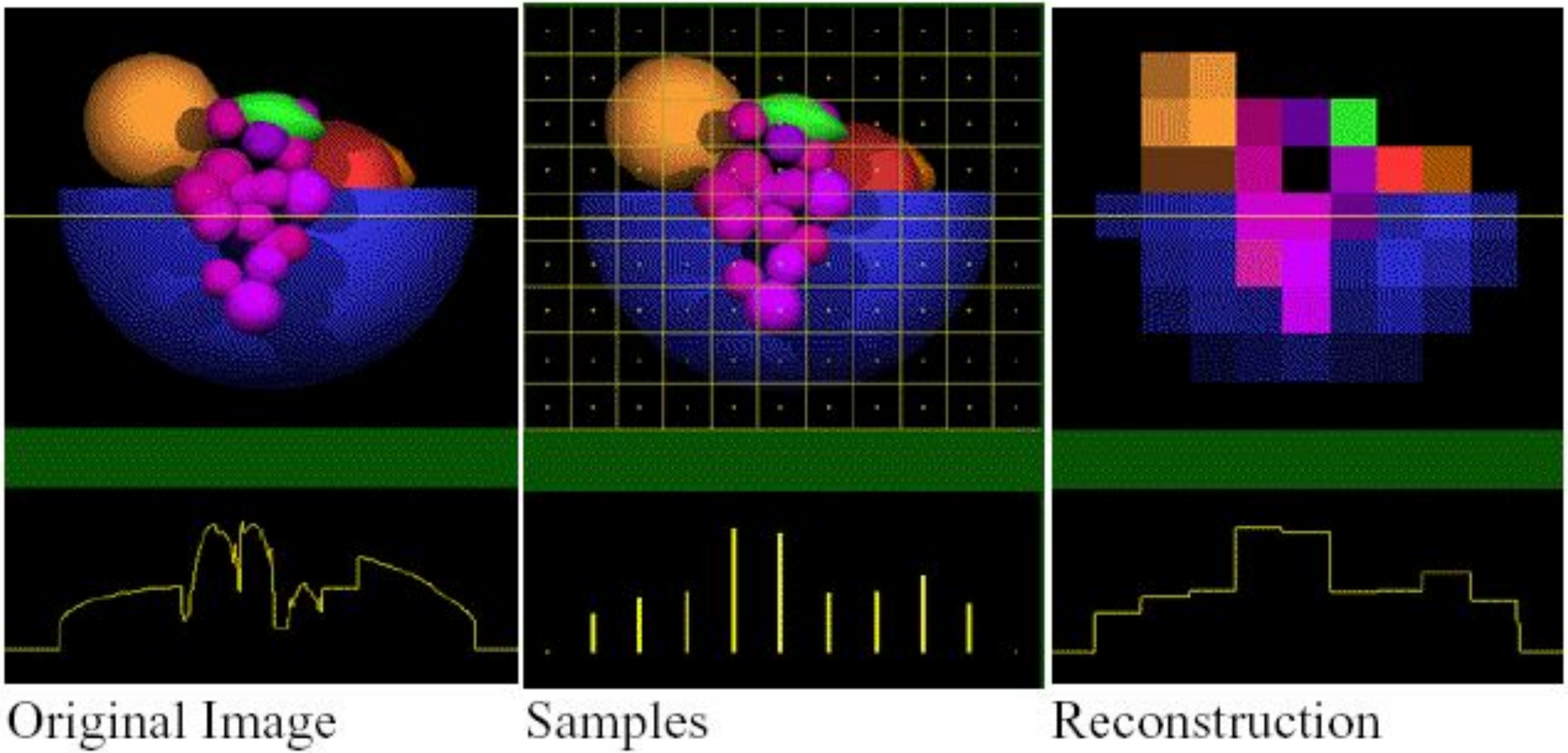
Sampling an Image

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

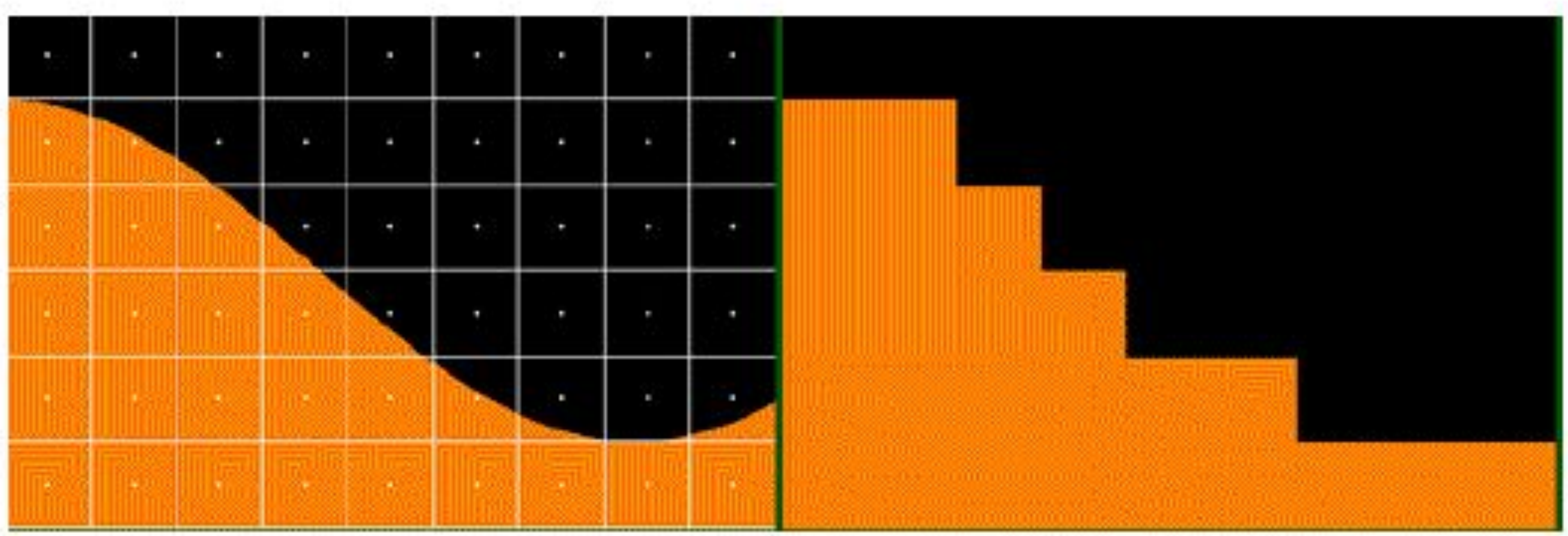




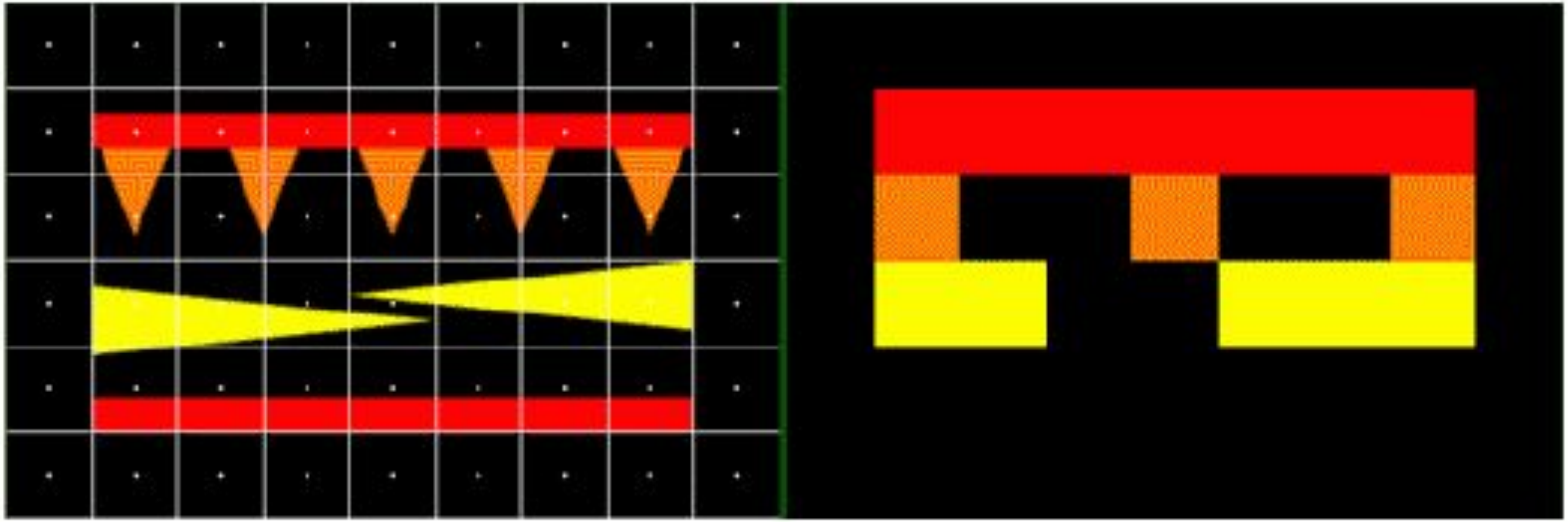
Aliasing occurs from *Sampling* and *Reconstruction*



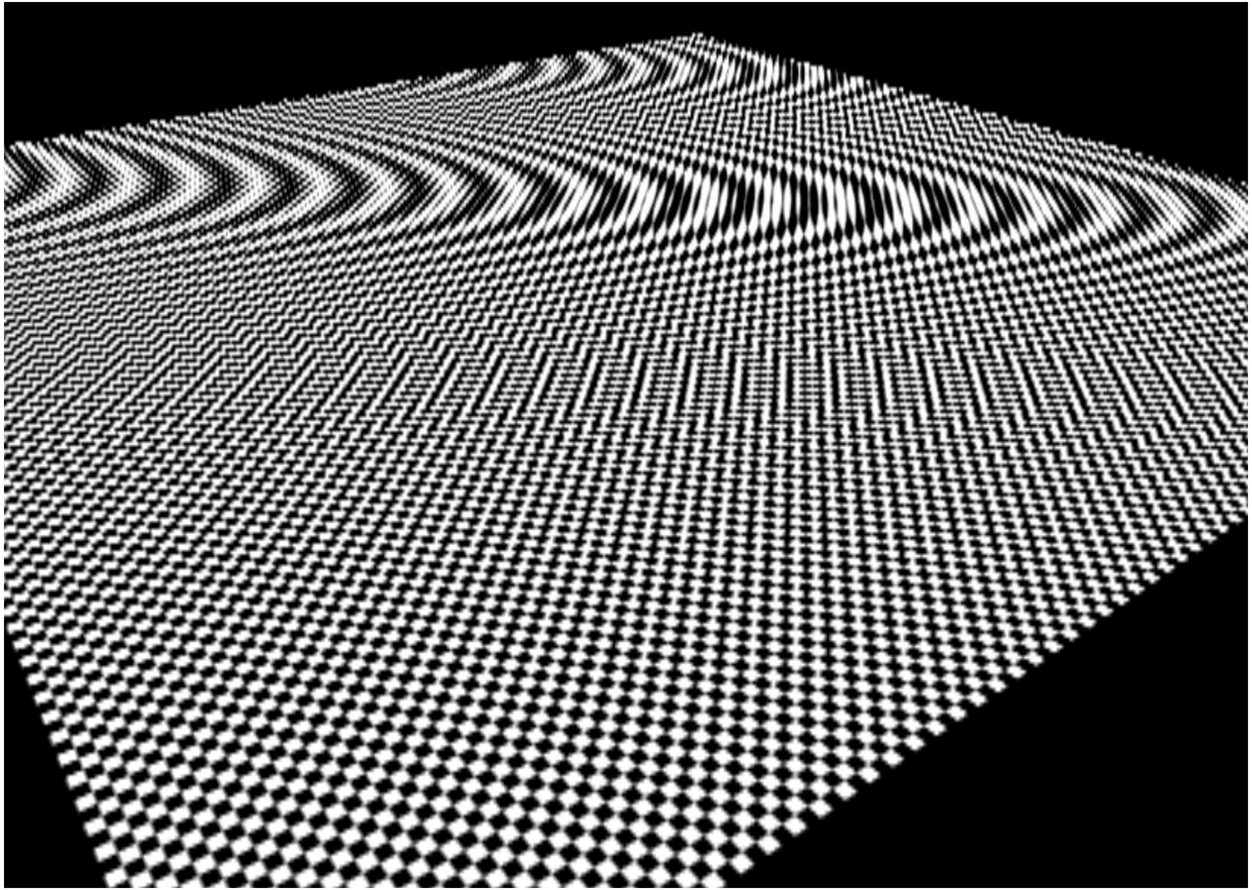
Examples of Aliasing: Jagged Boundaries



Examples of Aliasing: Improperly Rendered Detail



Examples of Aliasing: Insufficient Sampling Density



Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
  - ECSE Signals & Systems
  - Sampling Density, Fourier Analysis & Convolution
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

Sampling Density

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

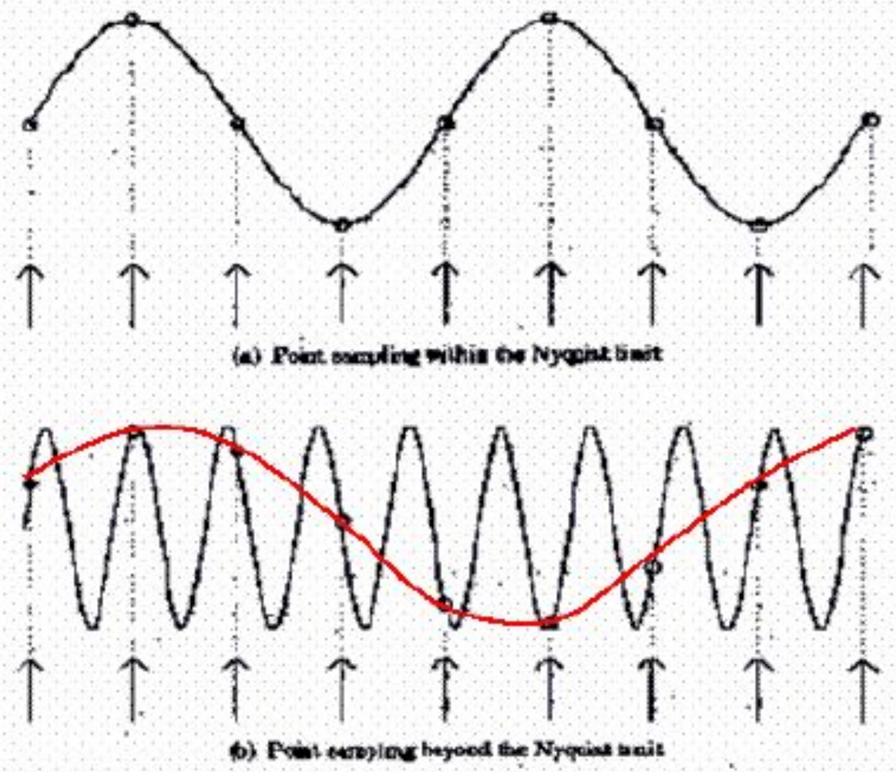


Image from Robert L. Cook, "Stochastic Sampling and Distributed Ray Tracing", An Introduction to Ray Tracing, Academic Press Limited, 1989.

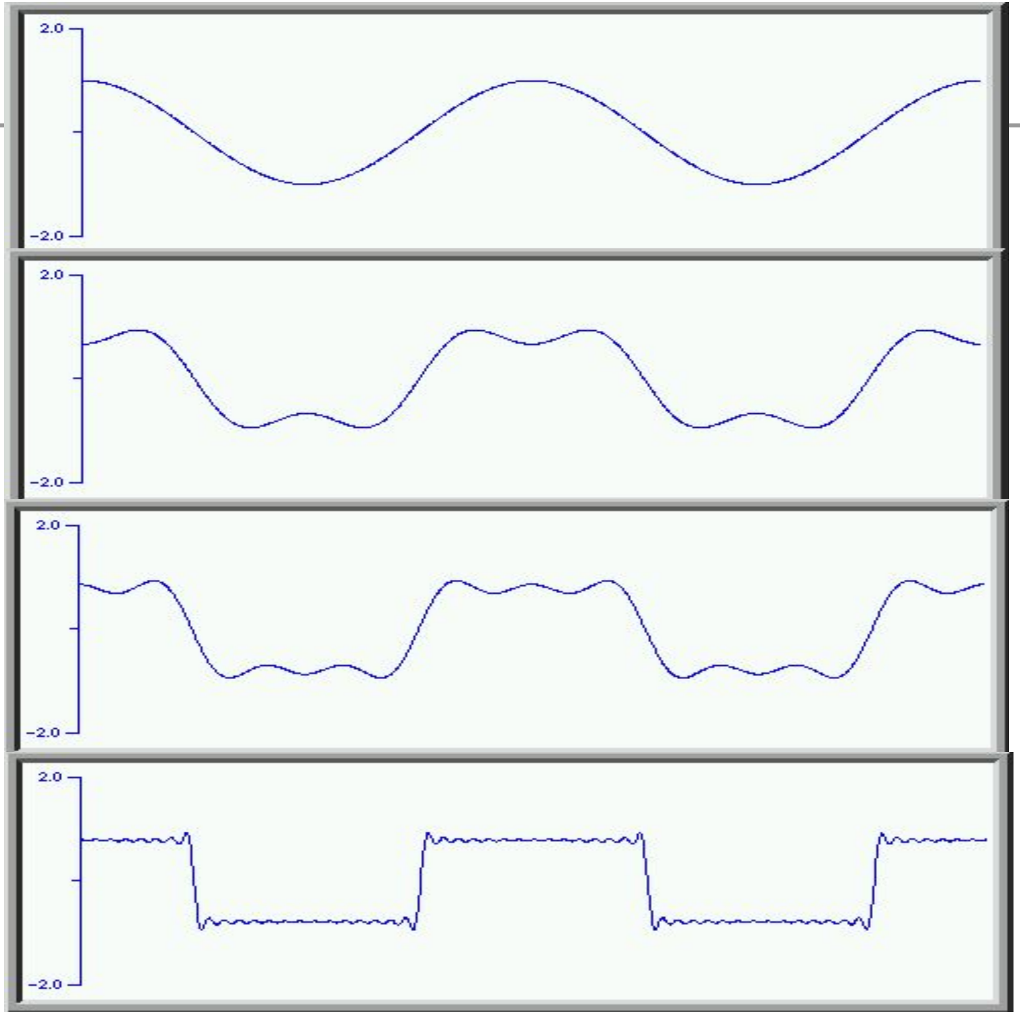
The simplest explanation for these samples is a low frequency sine wave

Signals & Systems

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

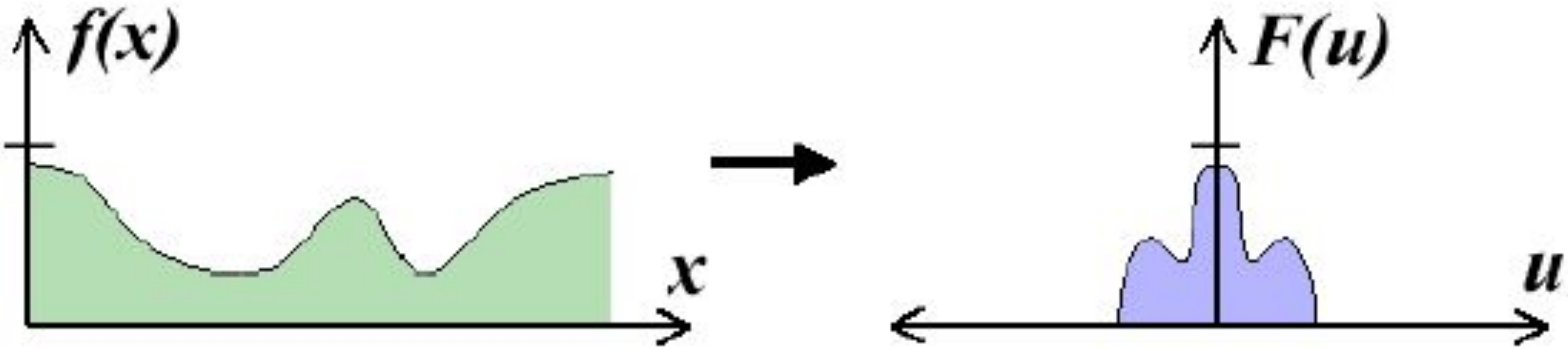
It's a shame that Signals & Systems is not required for CSCI majors...

Images from http://axion.physics.ubc.ca/341-02/fourier/fourier.html



Frequency Analysis

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



spatial domain *frequency domain*

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

Fourier Transform

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

frequency domain → $F(u,v)$; spatial domain → $f(x,y)$

Fourier Transform

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

Inverse Fourier Transform

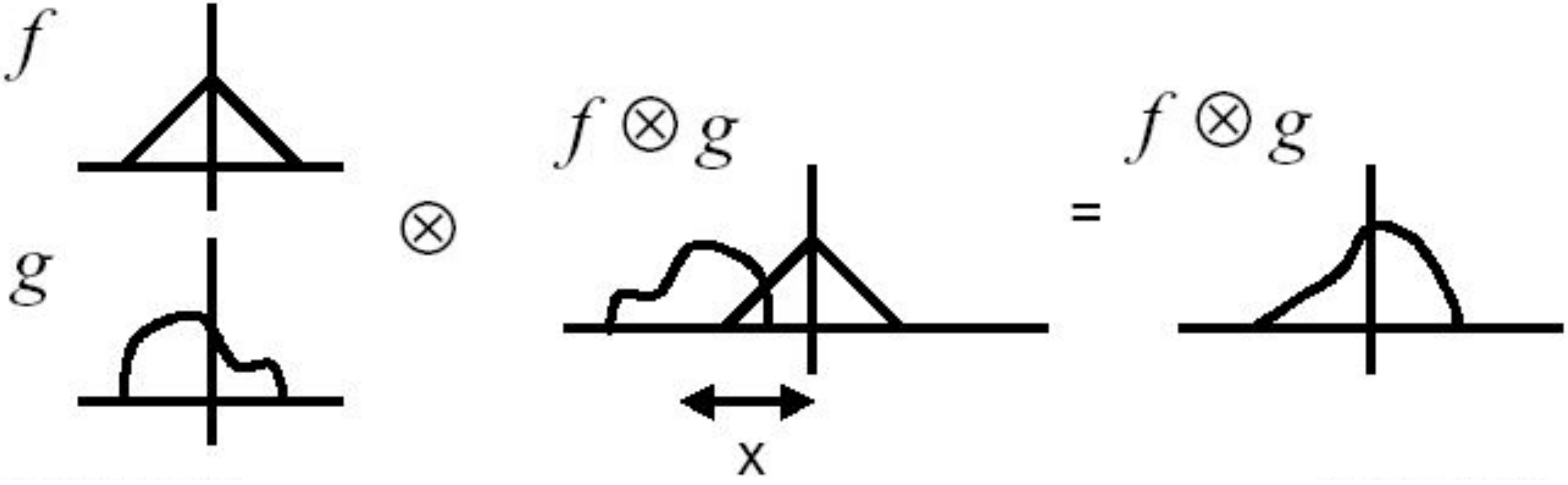
$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)\, e^{i2\pi(ux+vy)}\,du\,dv$$

Convolution

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

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

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



Fourier Transform & Convolution

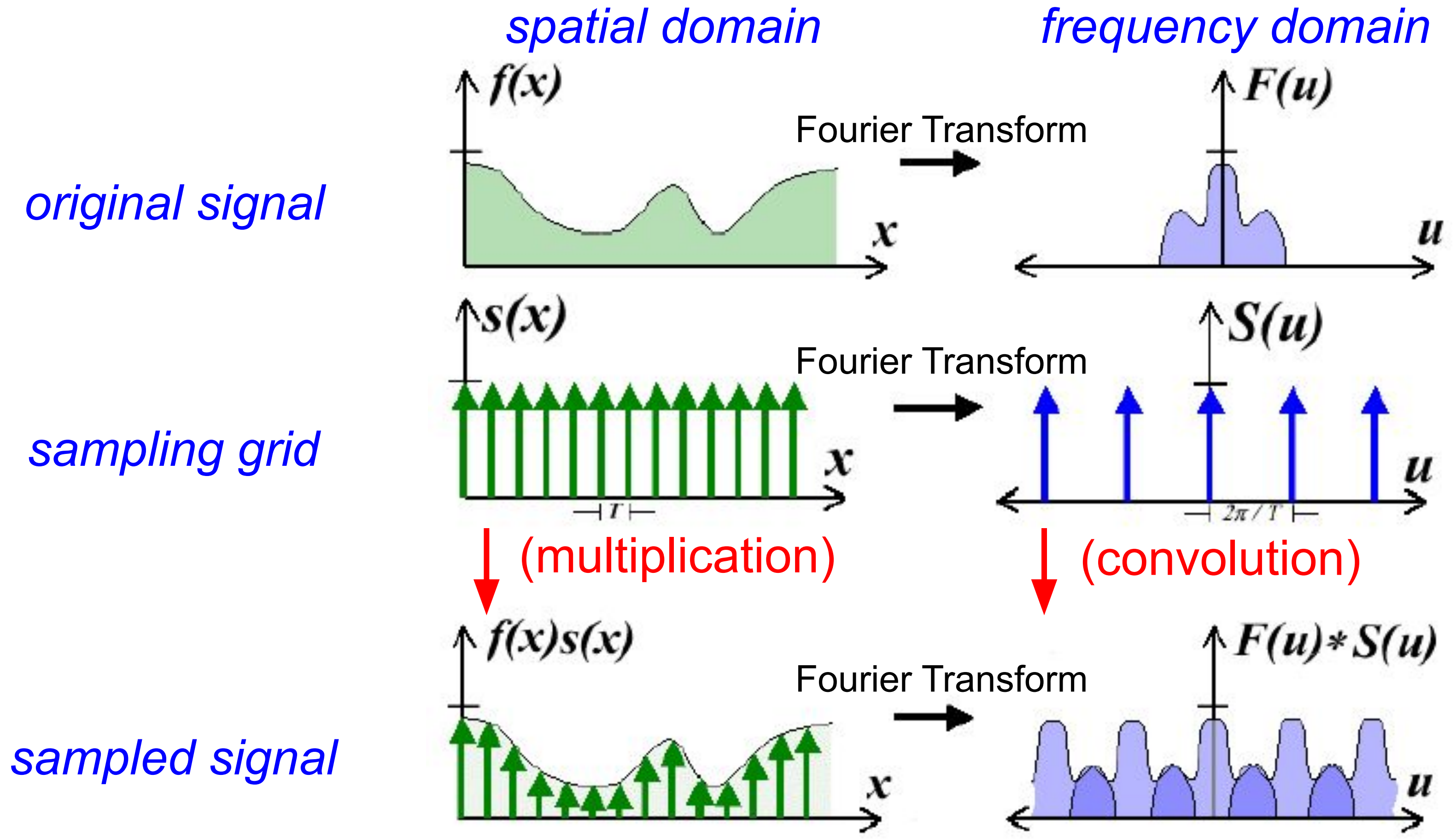
- Some operations that are difficult to compute in the spatial domain can be simplified by transforming to its dual representation in the frequency domain.
- For example, convolution in the spatial domain is the same as multiplication in the frequency domain.

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

- And, convolution in the frequency domain is the same as multiplication in the spatial domain

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

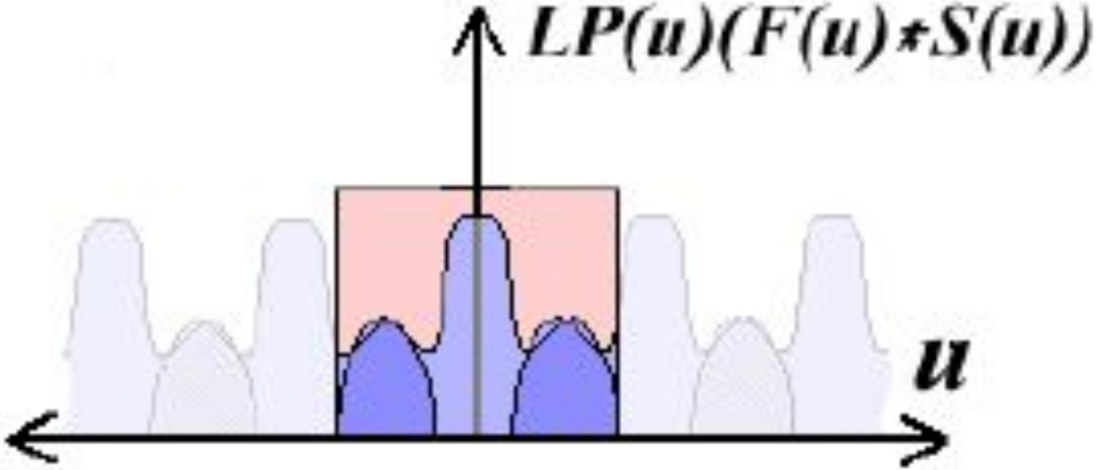
Sampling in the Frequency Domain



Reconstruction

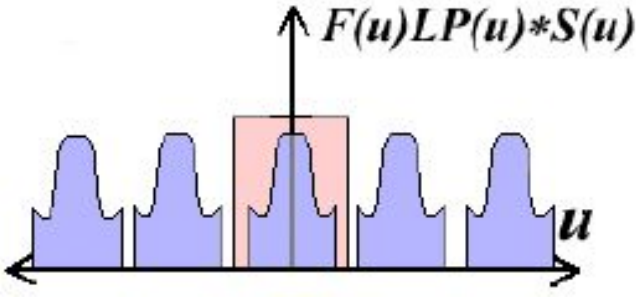
- If we can extract a copy of the original signal from the frequency domain of the sampled signal, we can reconstruct the original signal!

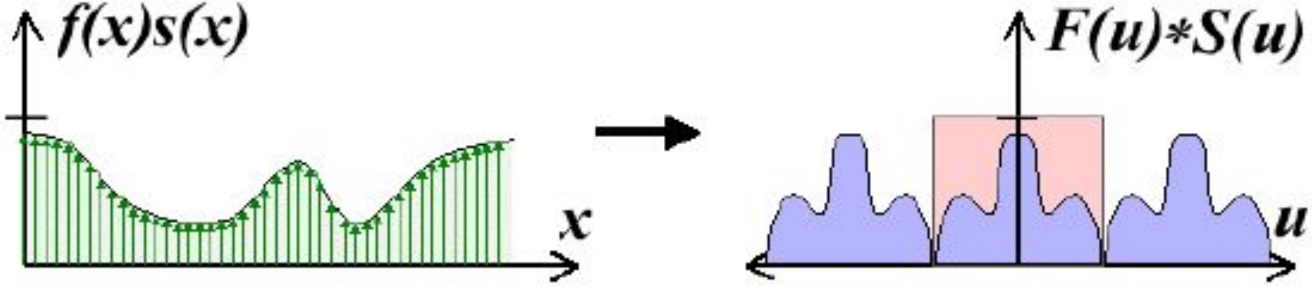
- But there may be overlap between the copies.



Guaranteeing Proper Reconstruction

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





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

Sampling Theorem

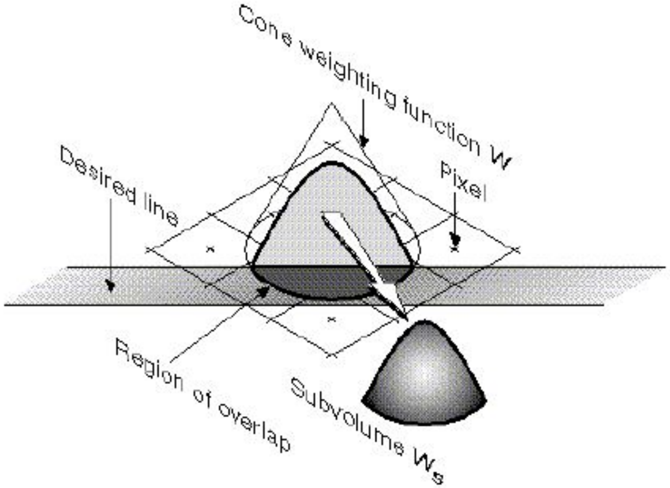
- When sampling a signal at discrete intervals,
  the sampling frequency must be *greater than twice*
  the highest frequency of the input signal
  in order to be able to reconstruct the original perfectly
  from the sampled version (Shannon, Nyquist)

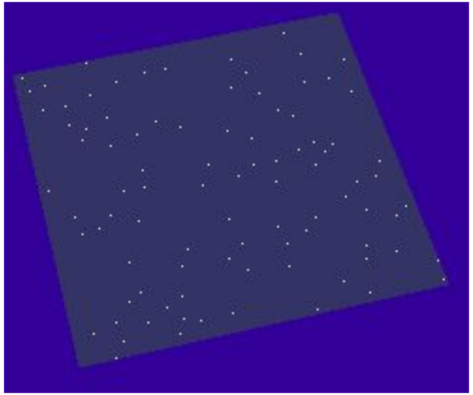
Today

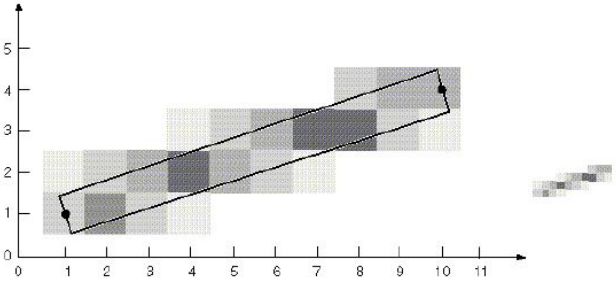
- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
  - Ideal, Gaussian, Box, Bilinear, Bicubic
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

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)







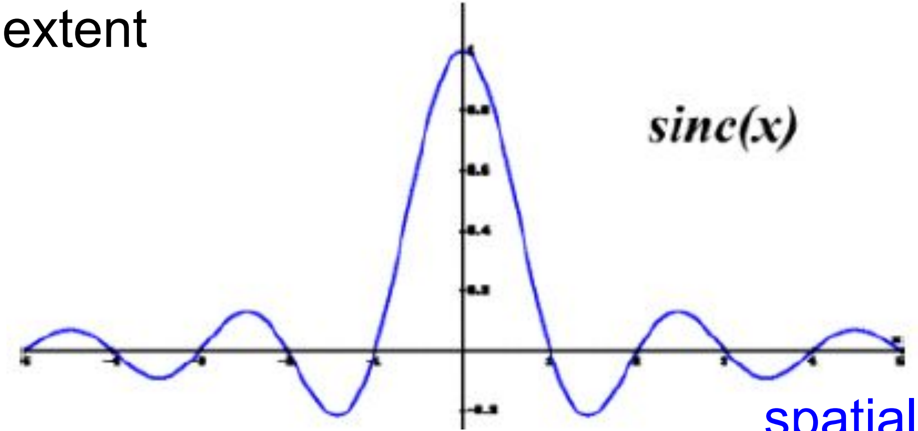
Source: Foley, VanDam, Feiner, Hughes - Computer Graphics, Second Edition, Addison Wesley

Filters

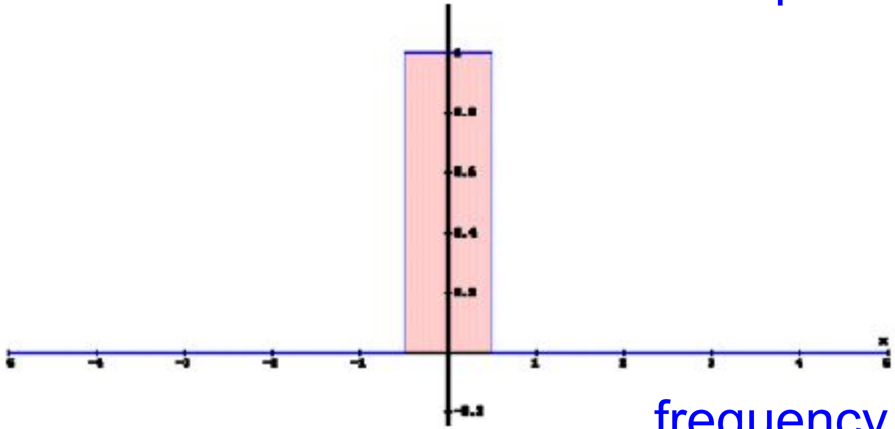
- Filters are used to
  - reconstruct a continuous signal from a sampled signal (reconstruction filters)
  - band-limit continuous signals to avoid aliasing during sampling (low-pass filters)
- Desired frequency domain properties are the same for both types of filters
- Often, the same filters are used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

The Ideal Filter

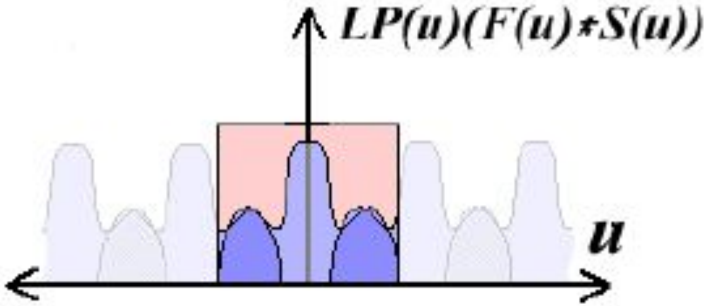
- Unfortunately it has *infinite* spatial extent
  - Every sample contributes to every interpolated point
- Expensive/impossible to compute



spatial

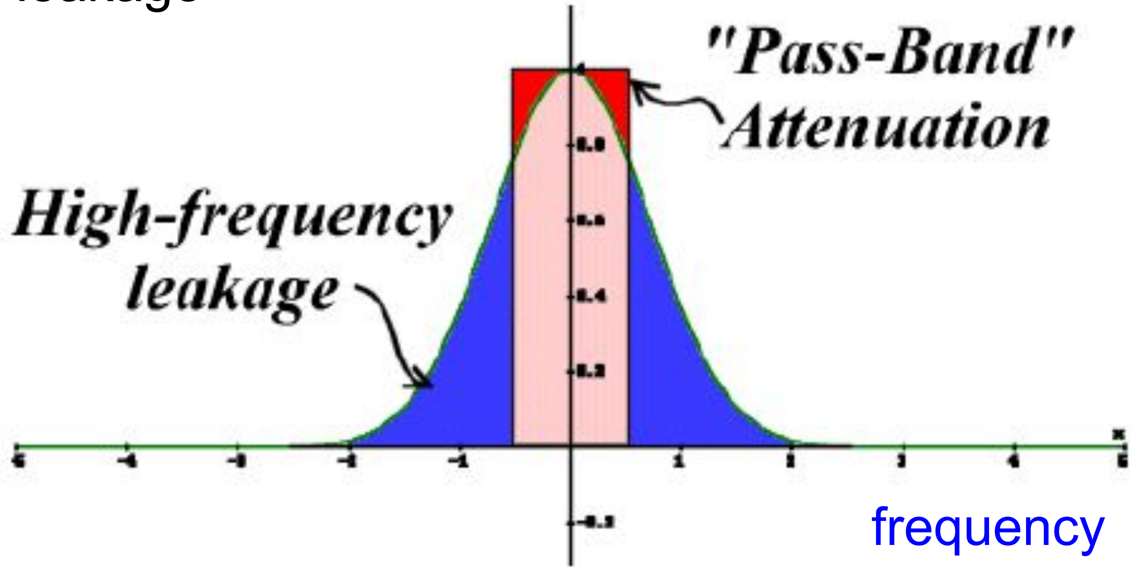


frequency



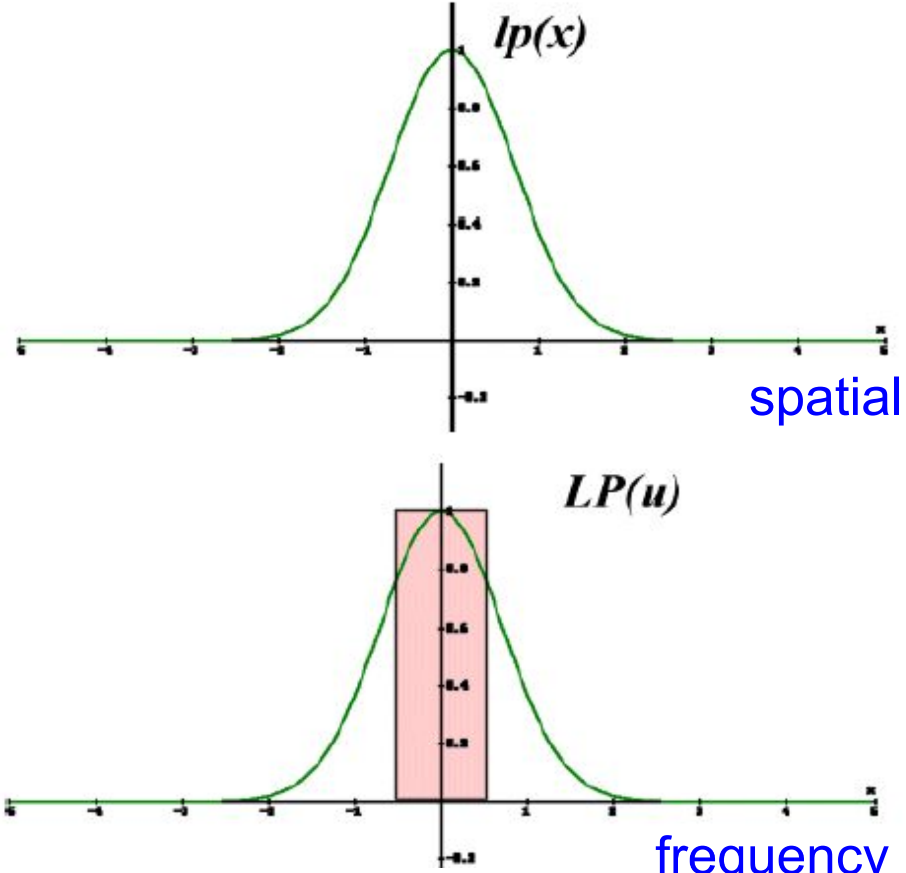
Problems with Practical Filters

- 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)



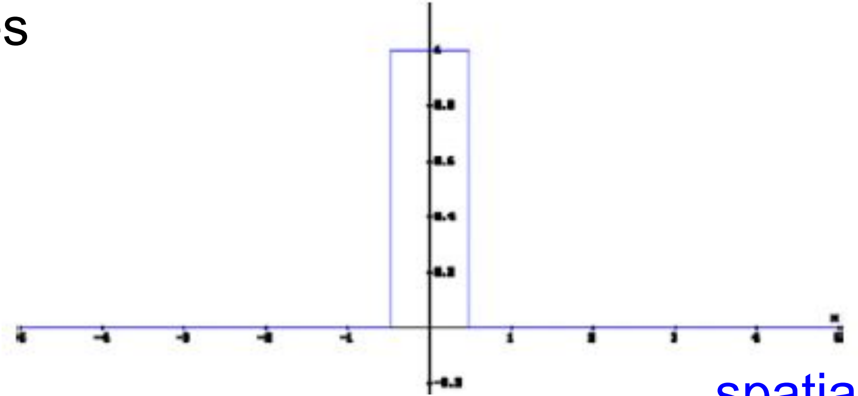
Gaussian Filter

- This is what old Cathode Ray Tube (CRT) monitors did for free!

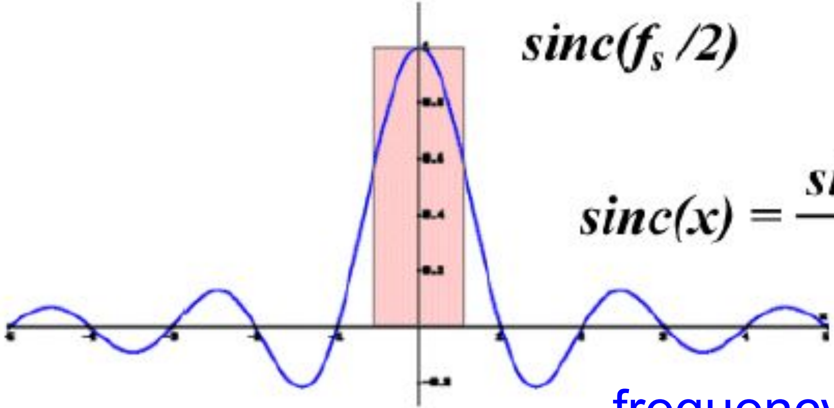


Box Filter / Nearest Neighbor

- Pretending pixels are little squares



spatial



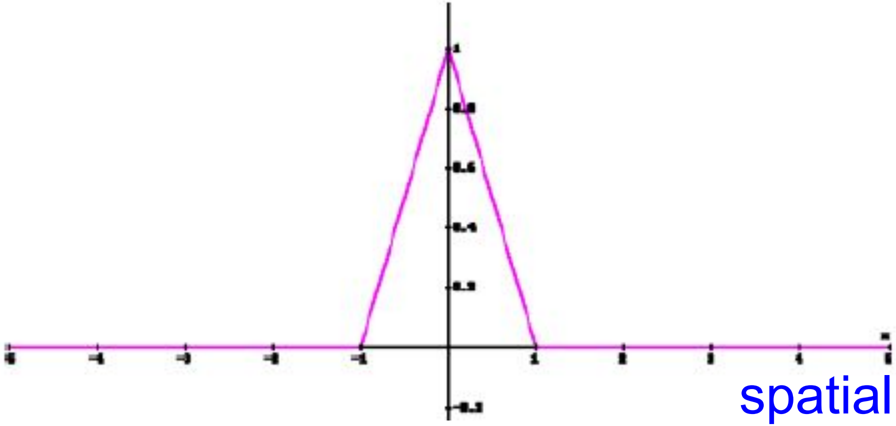
$sinc(f_s/2)$

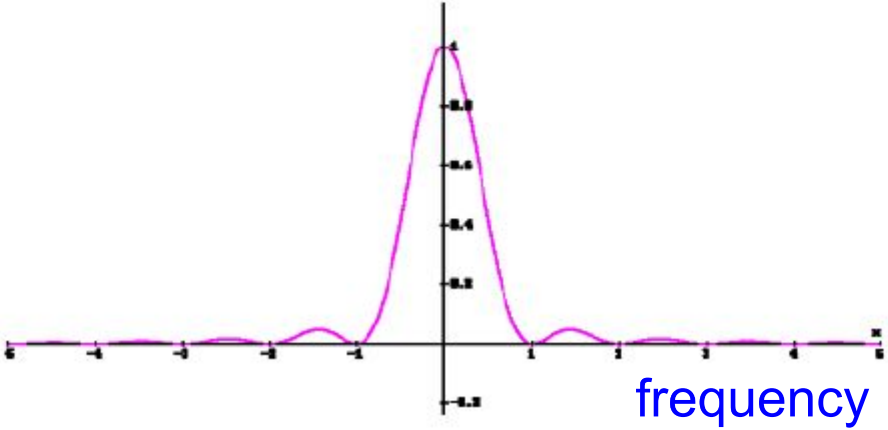
$$sinc(x) = \frac{\sin(\pi x)}{\pi x}$$

frequency

Tent Filter / Bi-Linear Interpolation

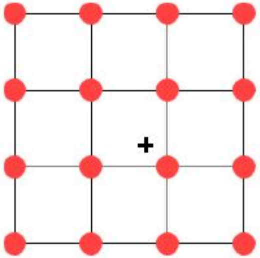
- Simple to implement
- Reasonably smooth

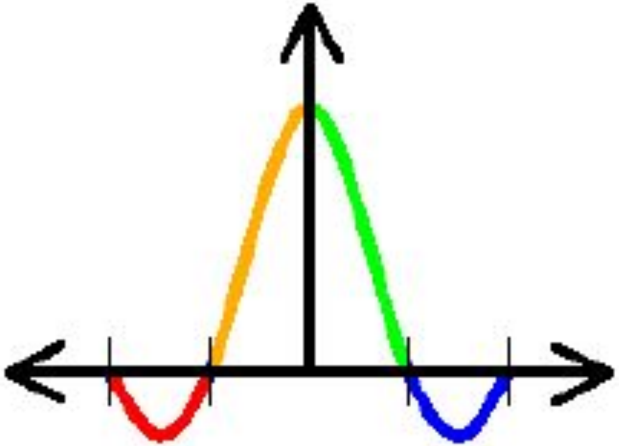


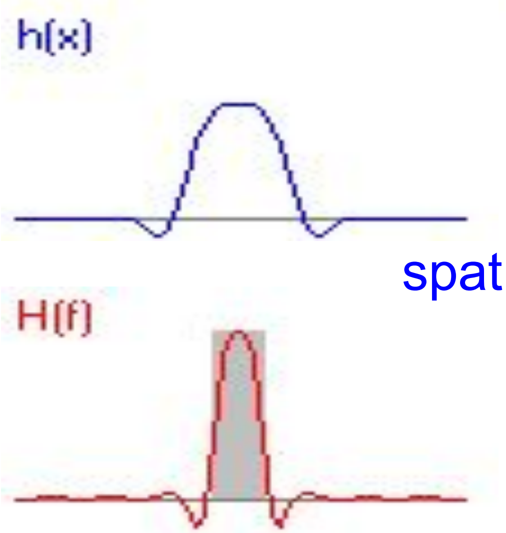


Bi-Cubic Interpolation

- Begins to approximate the ideal spatial filter, the sinc function







spatial

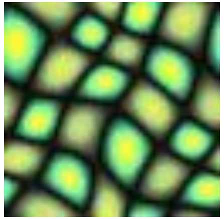
frequency

Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
  - Magnification & Minification, Mipmaps
- Papers for Today
- Papers for Next Time

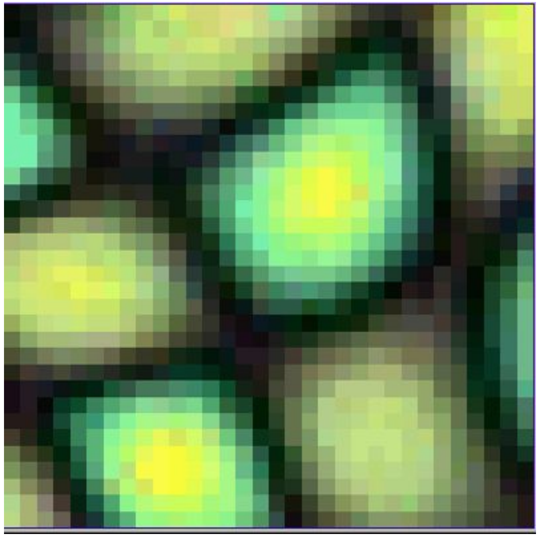
Sampling Texture Maps

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

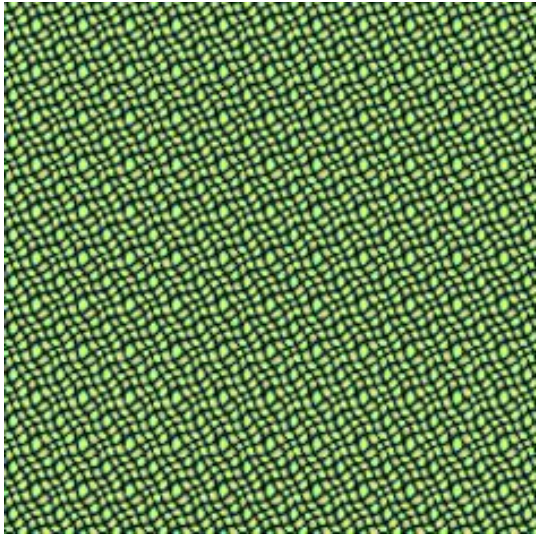


64x64 pixels

Original Texture



Magnification for Display



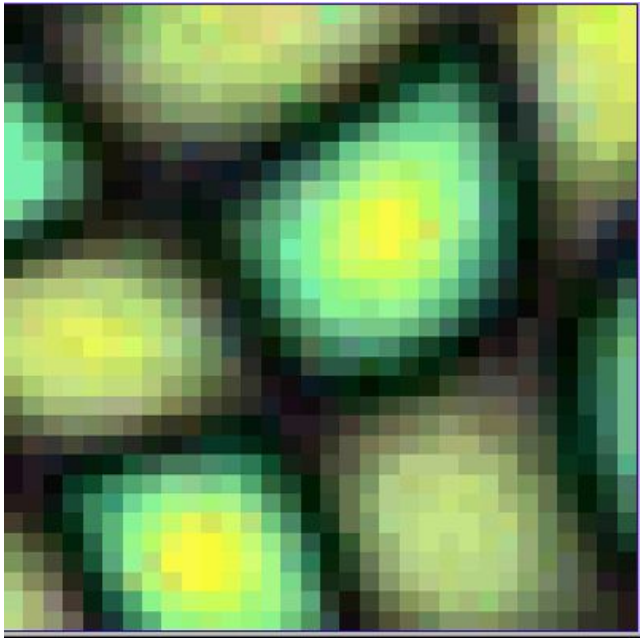
Minification for Display

for which we must use a reconstruction filter

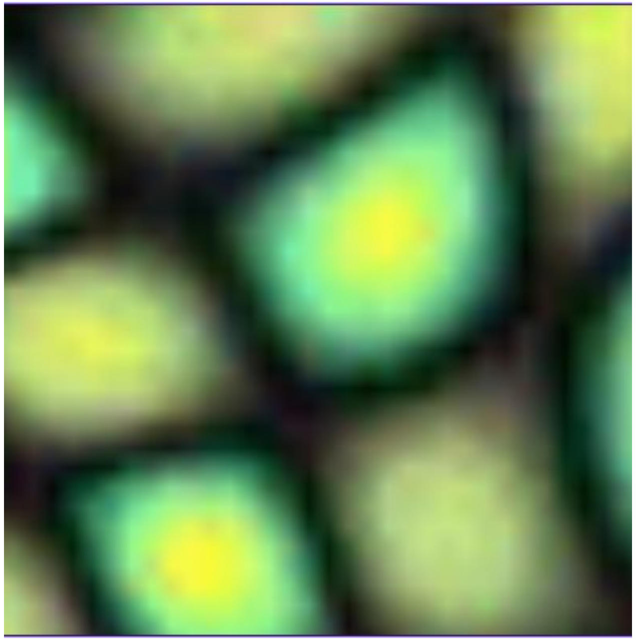
Linear Interpolation

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

box filter



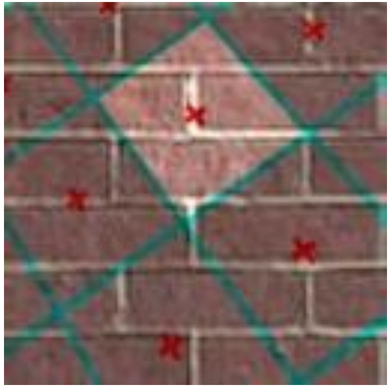
tent filter



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



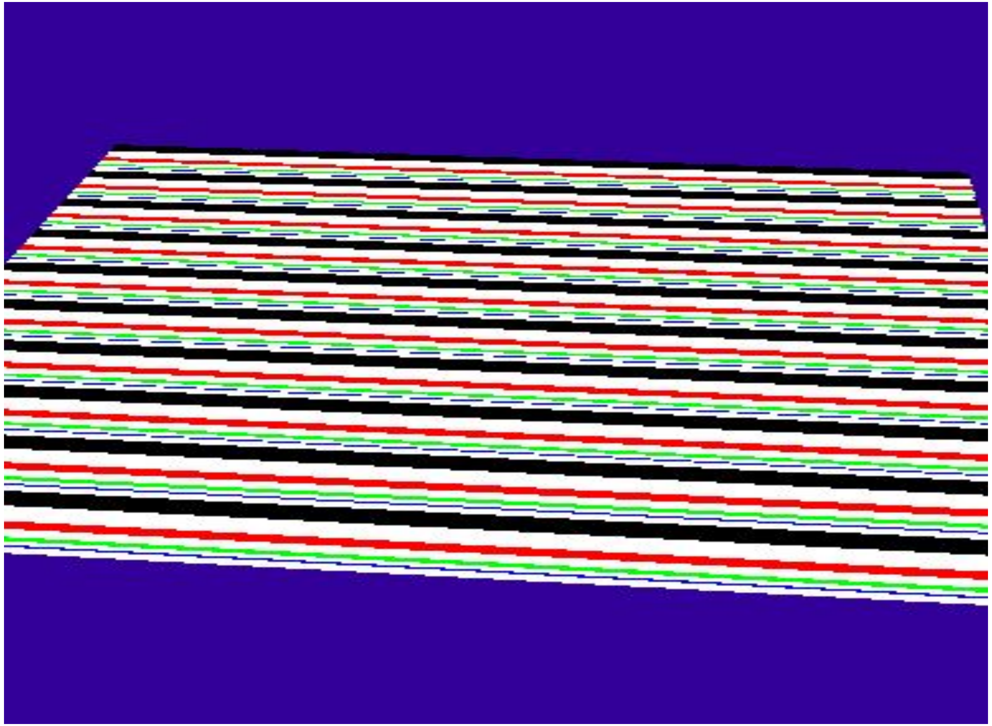
box filter in texture plane

What is MIP Mapping?

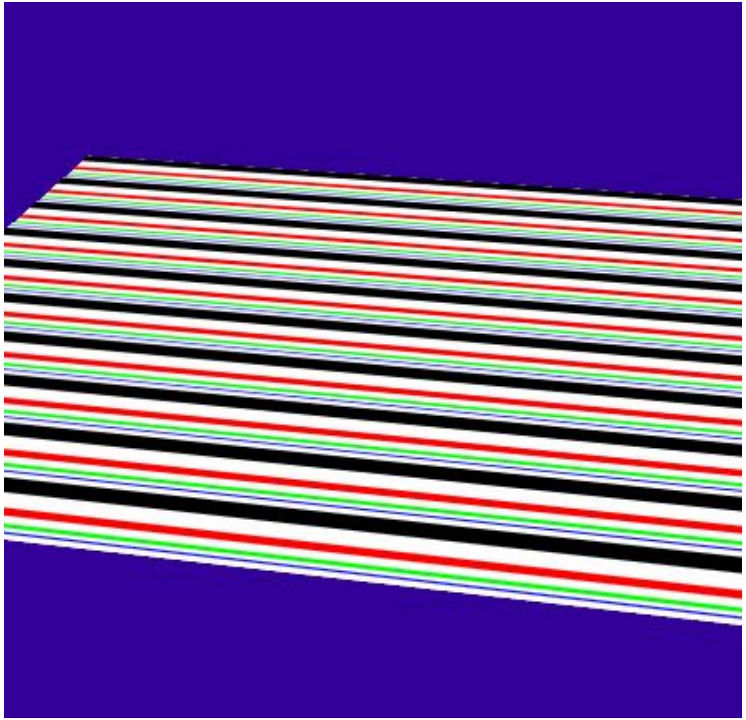
- Construct a pyramid of images that are pre-filtered and re-sampled at 1/2, 1/4, 1/8, etc., of the original image's sampling
- During rasterization we compute the index of the decimated image that is sampled at a rate closest to the density of our desired sampling rate
- MIP stands for *multum in parvo* which means *many in a small place*

MIP Mapping to Reduce/Eliminate Aliasing

- Thin lines may become disconnected / disappear



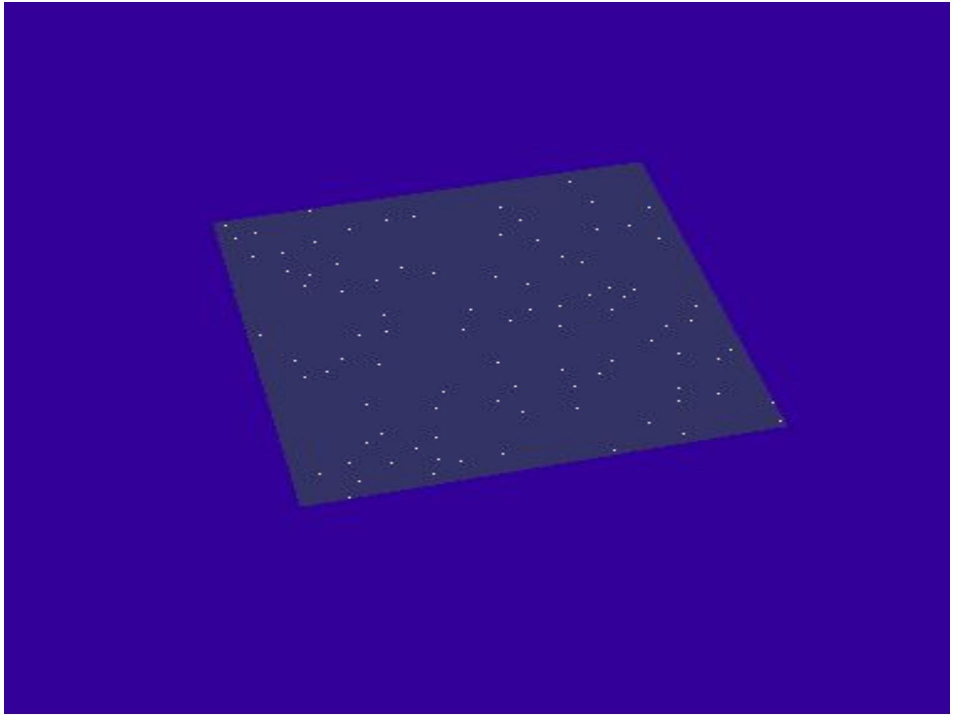
Nearest Neighbor



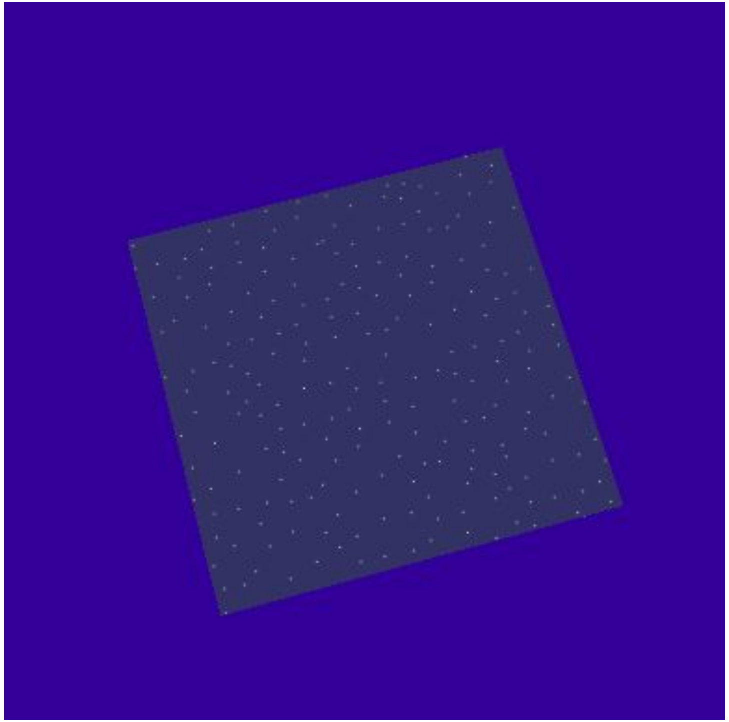
MIP Mapped (Bi-Linear)

MIP Mapping to Reduce/Eliminate Aliasing

- Small details may "pop" in and out of view

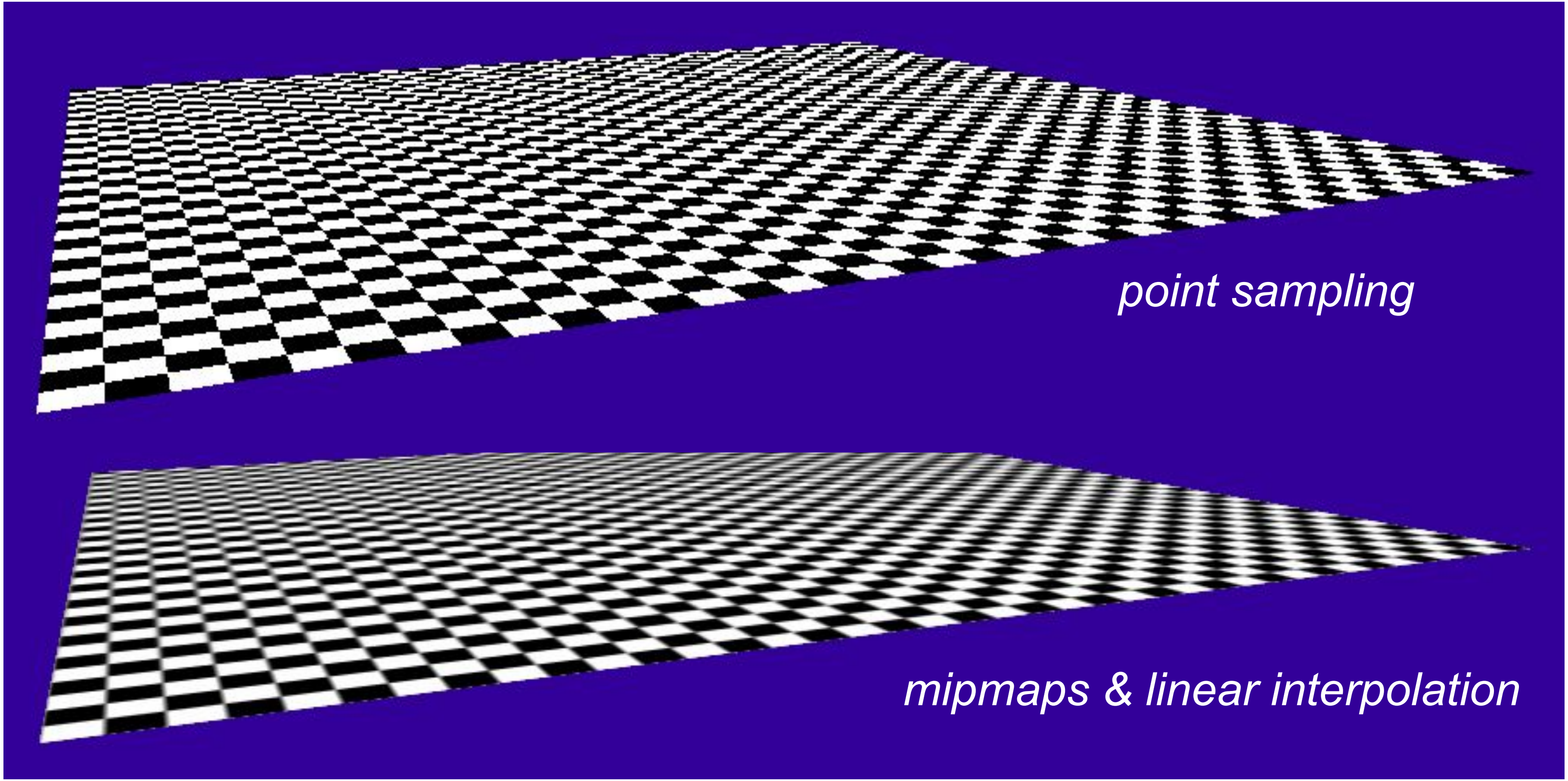


Nearest Neighbor



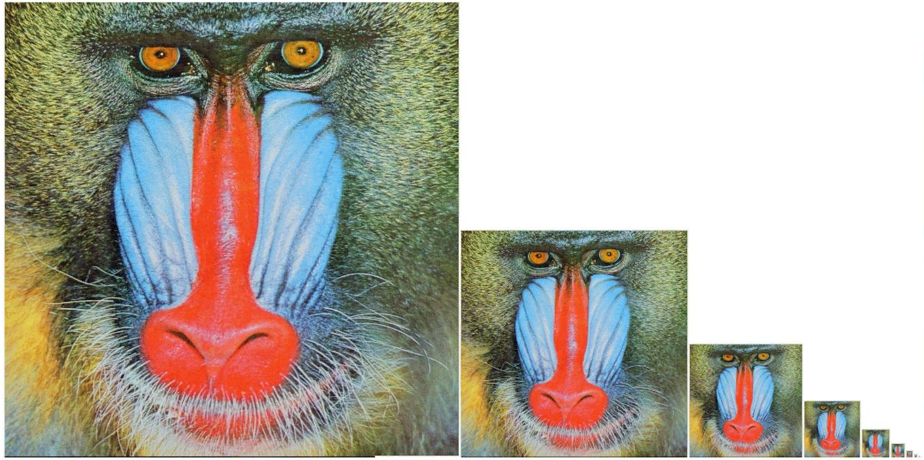
MIP Mapped (Bi-Linear)

Examples of Aliasing: Texture Errors

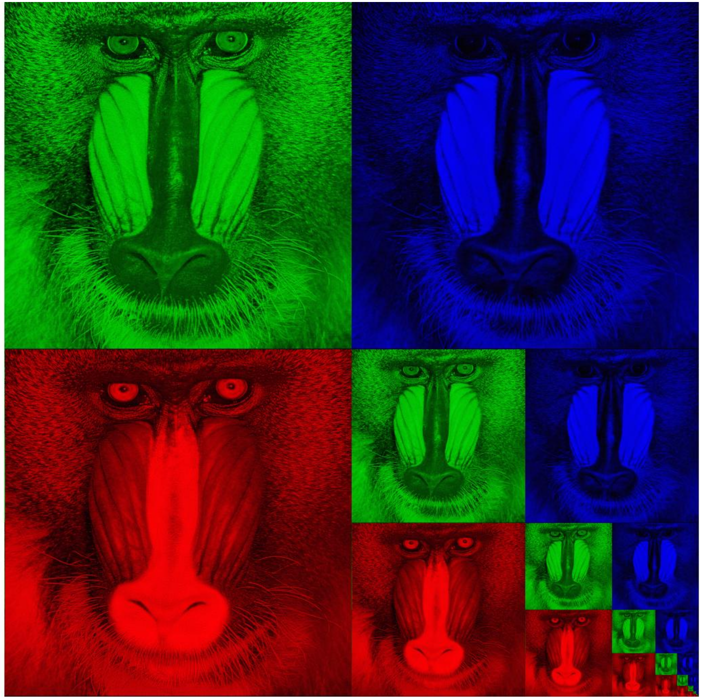


Storing MIP Maps

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



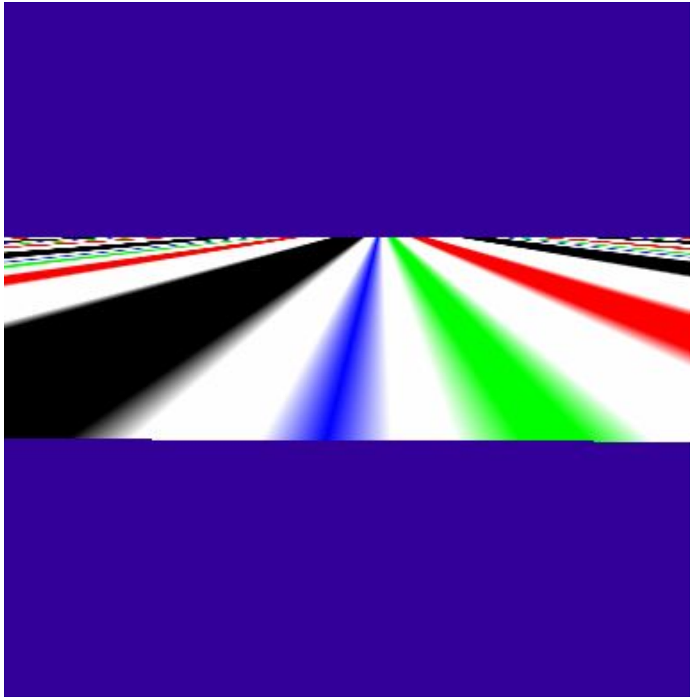
10-level mip map



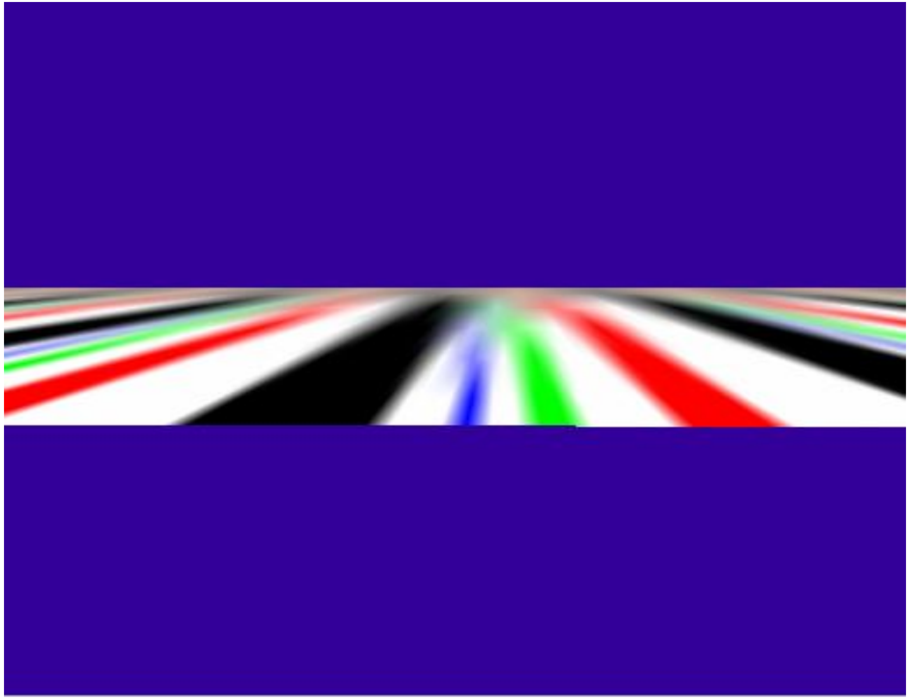
Memory format of a mip map

Anisotropic MIP Mapping

- What happens when the surface is tilted?



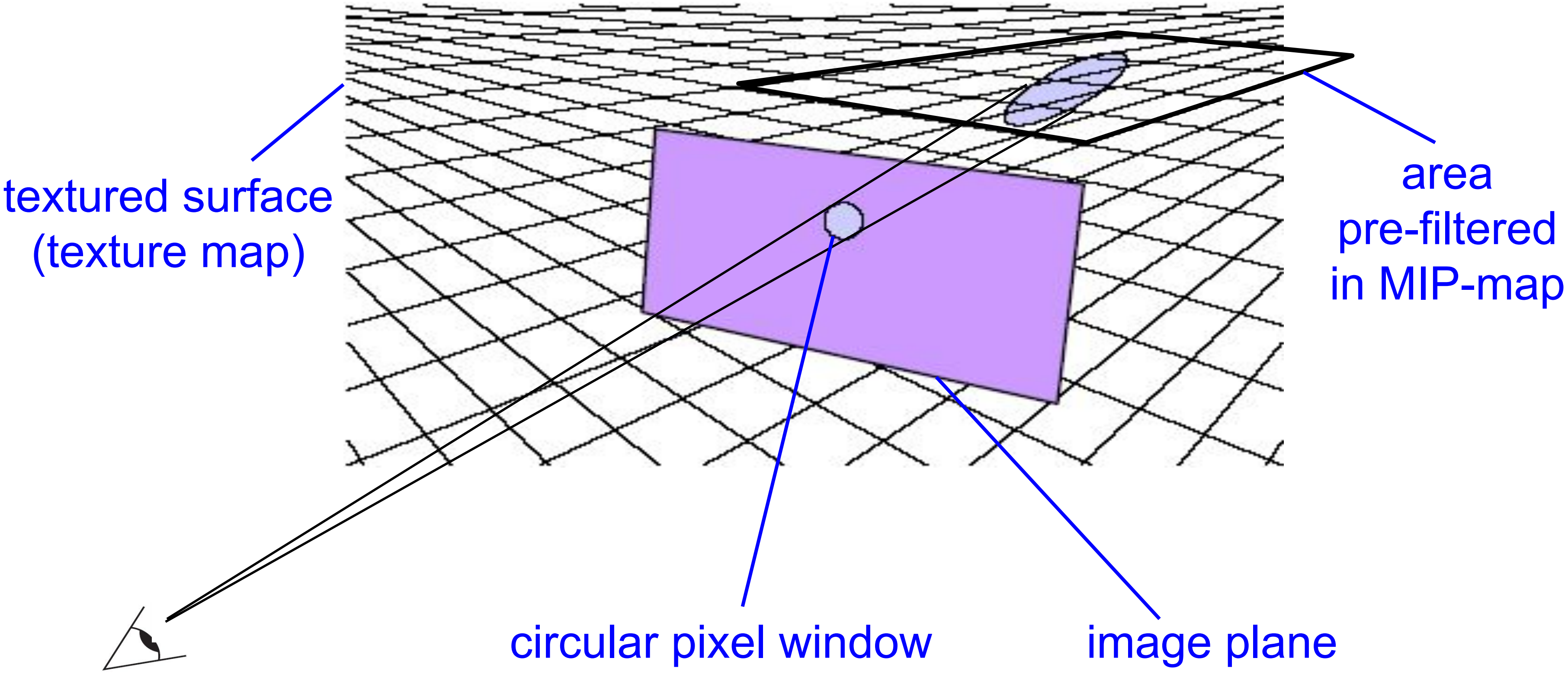
Nearest Neighbor



MIP Mapped (Bi-Linear)

Anisotropic MIP Mapping

- Square MIP map area can be a bad approximation



Anisotropic MIP Mapping

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

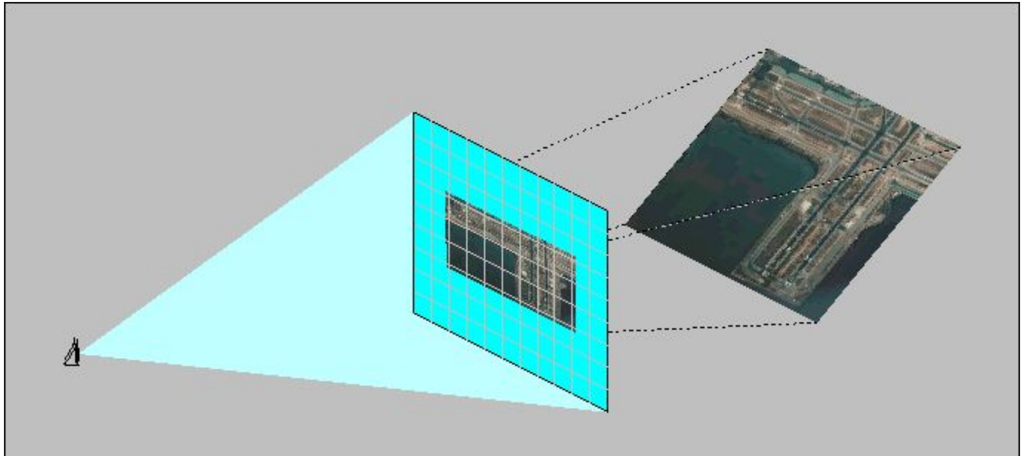


Figure 25. Geometry Orientation and Texture Aspect Ratio

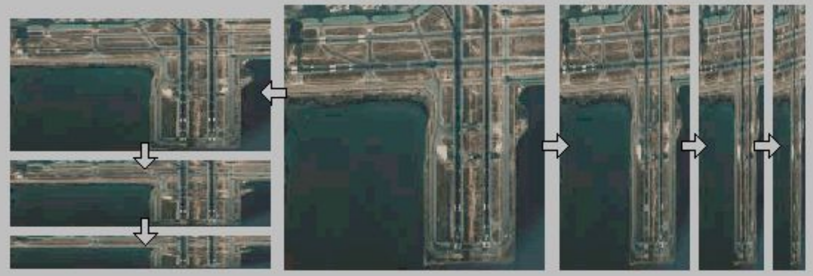


Figure 24. Creating a Set of Anisotropically Filtered Images

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

Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

Readings for Today

- “Correlated Multi-Jittered Sampling”,
  Andrew Kensler, Pixar Technical Memo, 2013

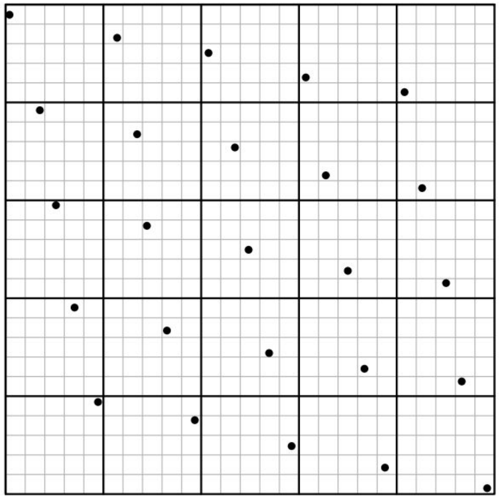


Figure 1: The canonical arrangement. Heavy lines show the boundaries of the 2D jitter cells. Light lines show the horizontal and vertical substrata of N-rooks sampling. Samples are jittered within the subcells.

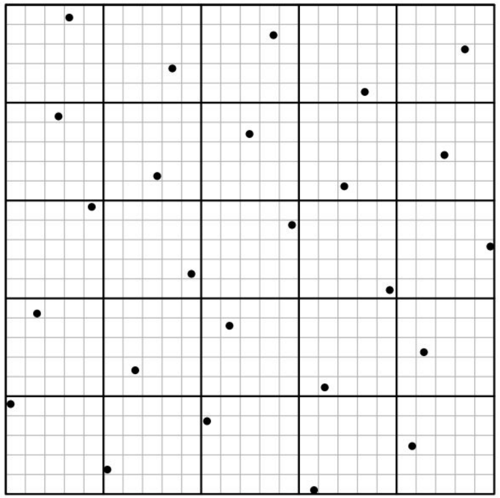


Figure 3: With correlated shuffling.

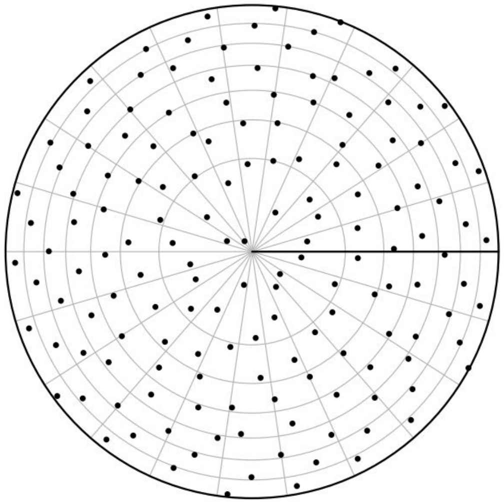


Figure 9: Polar warp with $m = 22, n = 7$.

[9]G. J. Ward and P. S. Heckbert. Irradiance gradients. In *Third Eurographics Rendering Workshop*, pages 85–98, May 1992.

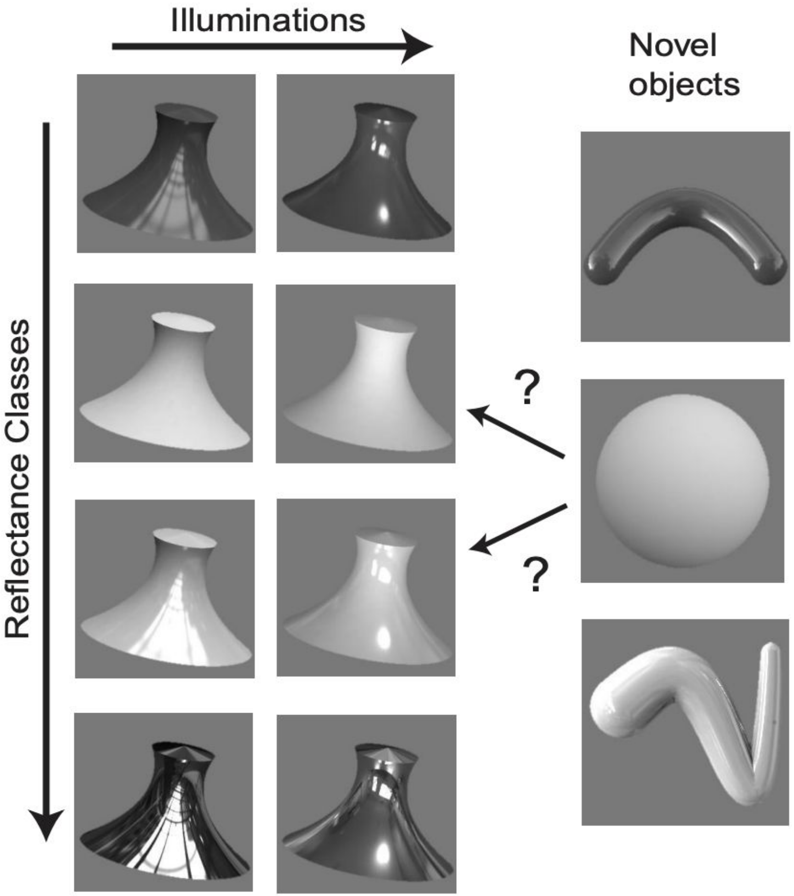
Readings for Today

- “Implicit Visibility and Antiradiance for Interactive Global Illumination”, Dachsbacher, Stamminger, Drettakis, and Durand Siggraph 2007

Readings for Today

- "Recognition of Surface Reflectance Properties from a Single Image under Unknown Real-World Illumination", Dror, Adelson, & Willsky, 2001.

Figure 1. The task addressed by our classifier. Using images of several surface materials under various illuminations as a training set, we wish to classify novel objects under novel illumination according to their surface material.



Today

- Worksheet: Photon Mapping
- Monte Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Today
- Papers for Next Time

Reading for Next Time *(pick one)*

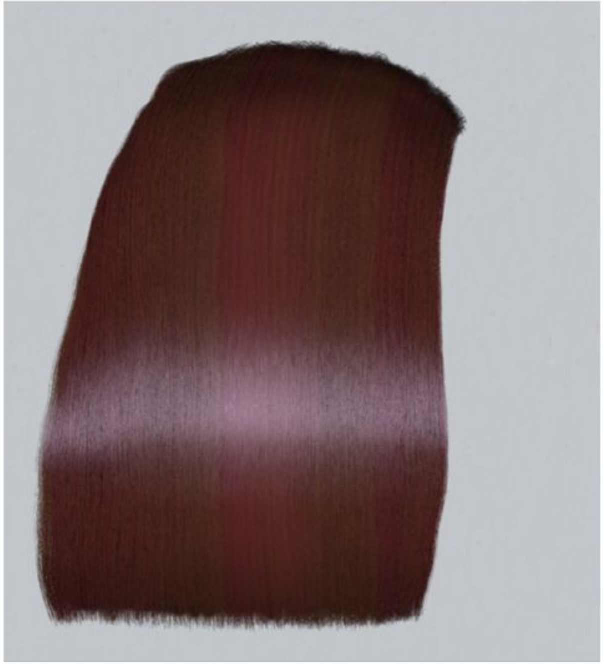
"A Practical Model for Subsurface Light Transport",
Jensen, Marschner, Levoy, & Hanrahan, SIGGRAPH 2001





Reading for Next Time *(pick one)*

Old Method



New Method

Photo

"Light Scattering from Human Hair Fibers"
Marschner et al., SIGGRAPH 2003

AND… everyone should read

- "Countering Racial Bias in Computer Graphics Research"
  Kim et al., SIGGRAPH 2022

- "More than Killmonger Locs - a style guide to Black Hair (in computer graphics)",
  Slides from A.M.Darke, 2024

Optional reading:

- "Curly-Cue: Geometric Methods for Highly Coiled Hair",
  Wu, Shi, Darke, & Kim, Siggraph Asia 2024