Sampling, Aliasing, & Mipmaps

Max: 50

Avg 35

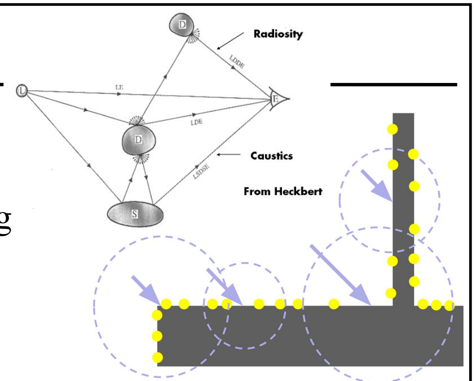
Std dev 7

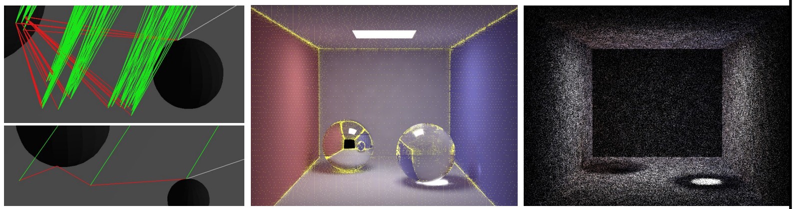
A- = 37 & up

B- = 30 & up

Last Time?

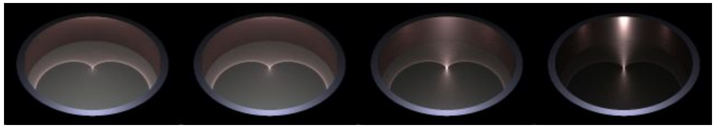
- Path Tracing vs. Ray Tracing
- Irradiance Caching
- Photon Mapping
- Ray Grammar



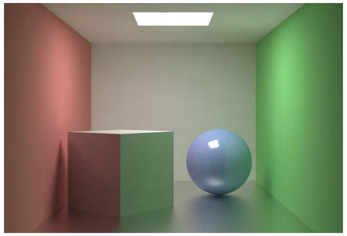


Readings for Today!

- "Rendering Caustics on Non-Lambertian Surfaces", Henrik Wann Jensen, *Graphics Interface* 1996.

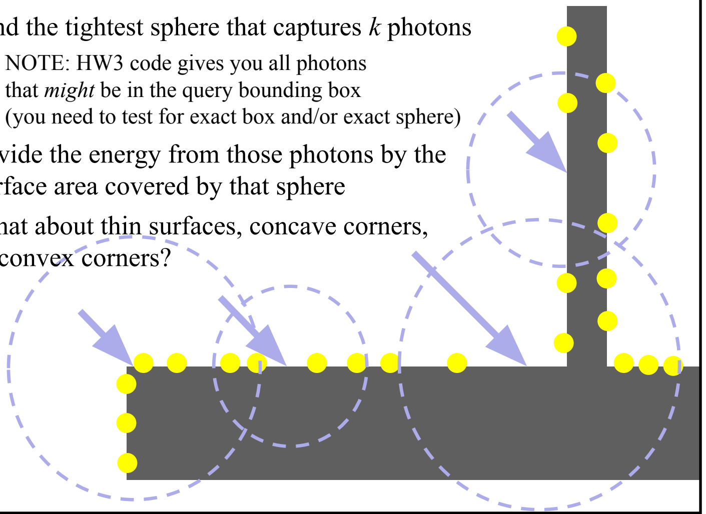


- "Global Illumination using Photon Maps", Henrik Wann Jensen, *Rendering Techniques* 1996.

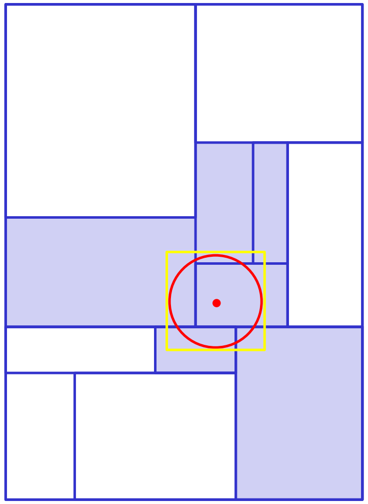


Closest Photon Details

- Find the tightest sphere that captures *k* photons
  - NOTE: HW3 code gives you all photons that *might* be in the query bounding box (you need to test for exact box and/or exact sphere)
- Divide the energy from those photons by the surface area covered by that sphere
- What about thin surfaces, concave corners, & convex corners?



Photons in the k-d tree details



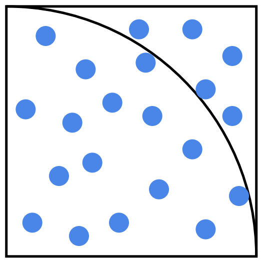
- You start with query point & radius (red)
- You give the `KDTree::CollectPhotonsInBox` function a bounding box (yellow)
- The algorithm finds all k-d tree cells that overlap with bounding box (blue)
- The function returns all photons in those cells
- *You need to discard all photons not in your original query radius*

Today

- Monte-Carlo Integration
  - Probabilities and Variance, Analysis
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

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 $= \pi/4$



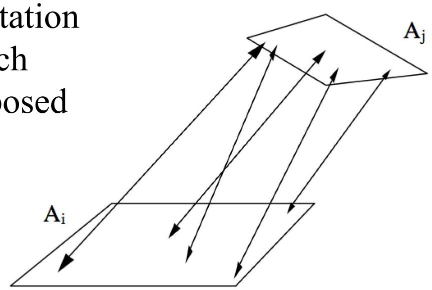
- $\pi \approx 4$ * number inside disc / total number
- The error depends on the number of trials

Use MC to calculate Form Factor

- 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

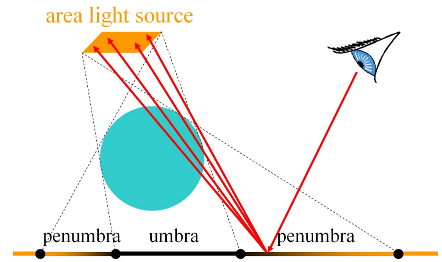
Use this for HW3!

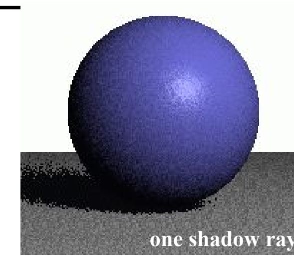
- Permits the computation of the patch-to-patch form factor, as opposed to point-to-patch

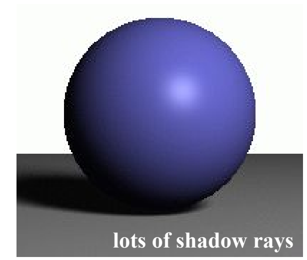


MC for Distributed Ray Tracing

- multiple shadow rays to sample area light source







Convergence & Error

- Let's "compute 0.5" by flipping a coin:
  - 1 flip: 0 or 1
    $\rightarrow$ average error = 0.5
  - 2 flips: 0, 0.5, 0.5 or 1
    $\rightarrow$ average error = 0. 25
  - 4 flips: 0 (*1),0.25 (*4), 0.5 (*6), 0.75(*4), 1(*1)
    $\rightarrow$ average error = 0.1875
- Unfortunately, doubling the number of samples does not double accuracy

Review of (Discrete) Probability

- Random variable can take discrete values x_i
- Probability p_i for each x_i

  $0 < p_i < 1, \quad \Sigma\, p_i = 1$
- Expected value $$E(x) = \sum_{i=1}^{n} p_i x_i$$
- Expected value of function of random variable
  - $f(x_i)$ is also a random variable

$$E[f(x)] = \sum_{i=1}^{n} p_i f(x_i)$$

Variance & Standard Deviation

- Variance σ^2: deviation from expected value
- Expected value of square difference

$$\sigma^2 = E[(x - E[x])^2] = \sum_i (x_i - E[x])^2 p_i$$

- Also

$$\sigma^2 = E[x^2] - (E[x])^2$$

- Standard deviation σ:
  square root of variance (notion of error, RMS)

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 MC 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 MC Integration

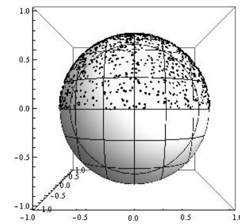
- Noisy
- Slow convergence
- Good implementation is hard
  - Debugging code
  - Debugging math
  - Choosing appropriate techniques
- Punctual technique, no notion of smoothness of function (e.g., between neighboring pixels)

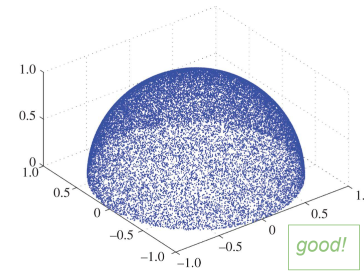
Today

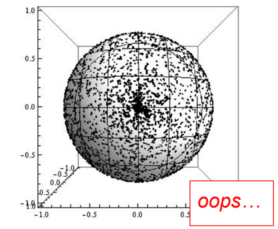
- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

Domains of Integration

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





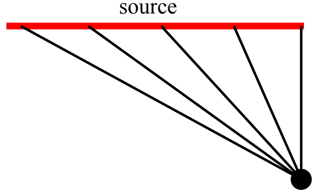


Example: Light Source

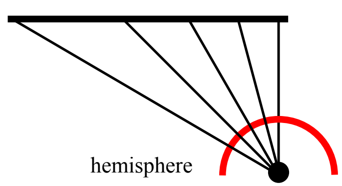
- We can integrate over surface *or* over angle
- But we must be careful to get probabilities and integration measure right!

Sampling the source uniformly

source

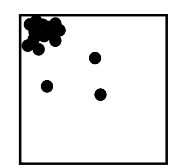


Sampling the hemisphere uniformly

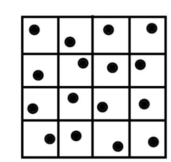


Stratified Sampling

- 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 Example

$f(x) = e^{\sin(3x^2)}$	
N	I
1	2.75039
10	1.9893
100	1.79139
1000	1.75146
10000	1.77313
100000	1.77862

$f(x) = e^{\sin(3x^2)}$	
N	I
1	2.70457
10	1.72858
100	1.77925
1000	1.77606
10000	1.77610
100000	1.77610

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

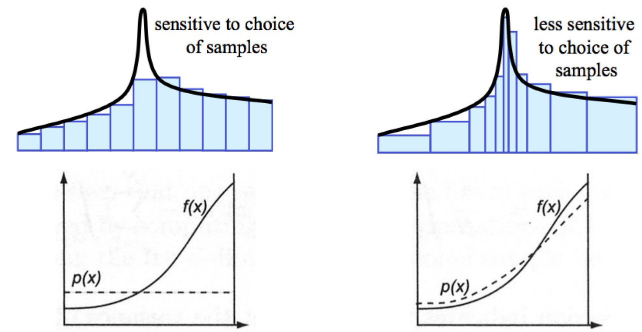
Stratified $O(1/N)$

Slide from Henrik Wann Jensen

Sampling

uniform sampling
(or uniform random)

dense sampling where
function has greater magnitude



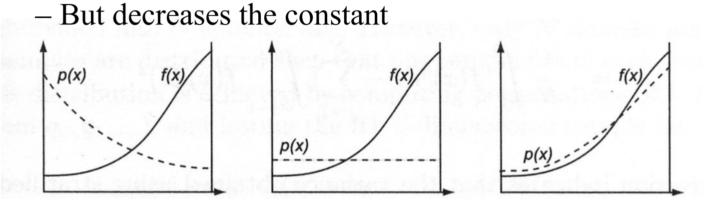
all samples
weighted equally

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



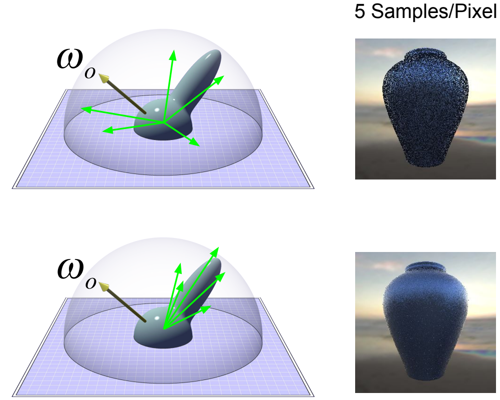
bad uniform good

Uniform vs. Importance Sampling

5 Samples/Pixel

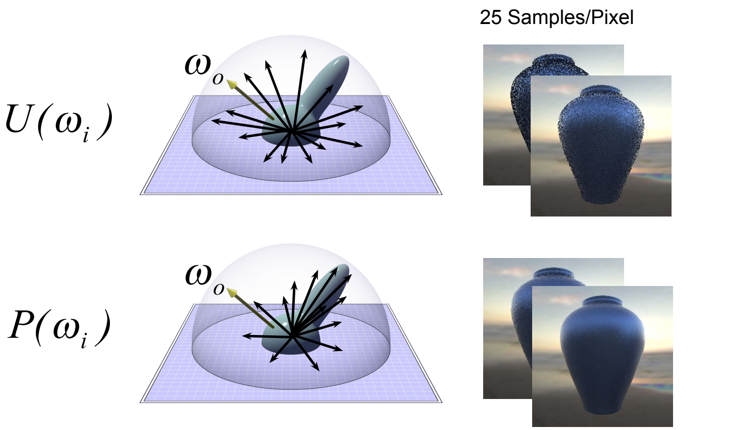
$U(\omega_i)$

$P(\omega_i)$



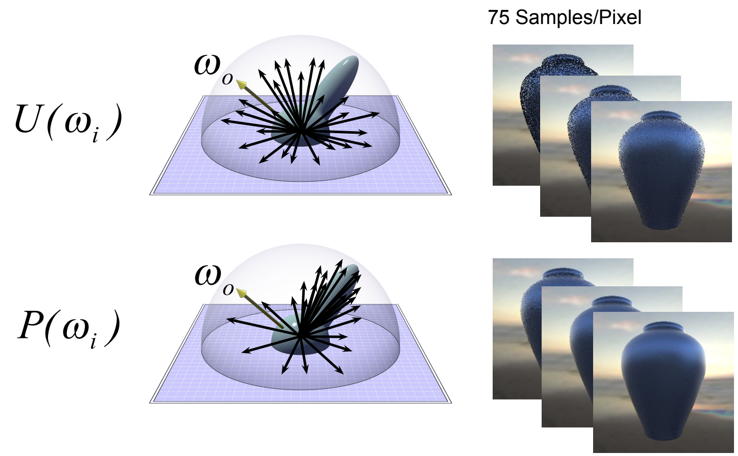
Slide from Jason Lawrence

Uniform vs. Importance Sampling



Slide from Jason Lawrence

Uniform vs. Importance Sampling



Slide from Jason Lawrence

Bidirectional Path Tracing

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

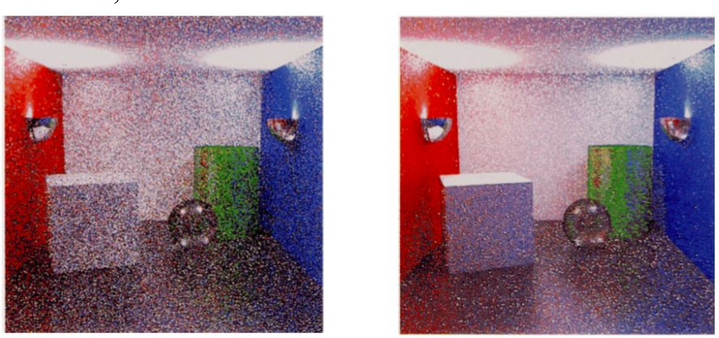


Figure B: *An indirectly illuminated scene rendered using path tracing and bidirectional path tracing respectively. The latter method results in visibly less noisefor the same amount of work.*

Questions?

Naïve sampling strategy Optimal sampling strategy

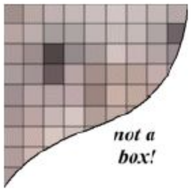
Veach & Guibas "Optimally Combining Sampling Techniques for Monte Carlo Rendering" SIGGRAPH 95

Today

- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

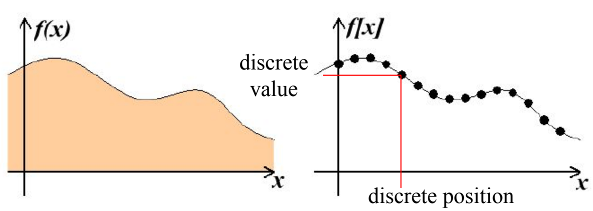
What is a Pixel?

- 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



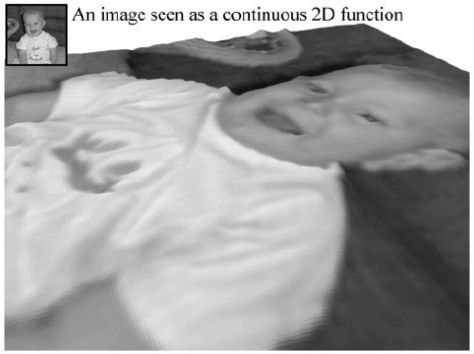
More on Samples

- 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



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

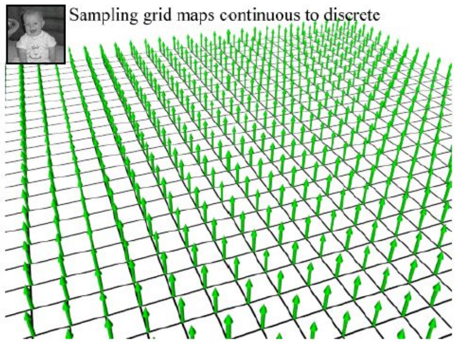
- We can generate the table values by multiplying the continuous image function by a sampling grid of Kronecker delta functions.

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:

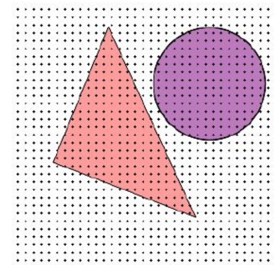
$$\sum_{j=0}^{h-1} \sum_{i=0}^{w-1} \delta(u-i, v-j)$$

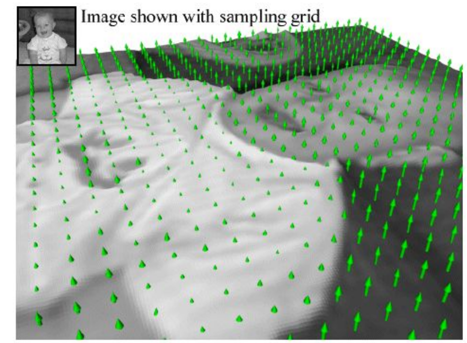


Sampling an Image

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

The same analysis can be applied to geometric objects:

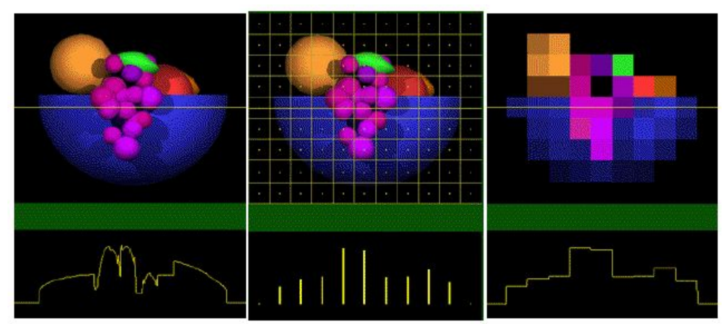




Today

- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

Examples of Aliasing

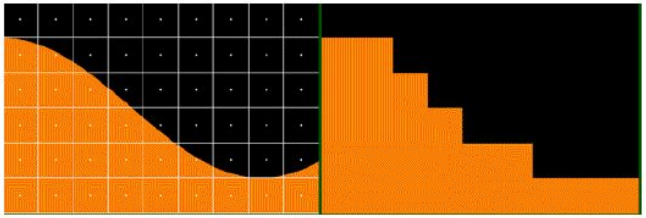


Original Image Samples Reconstruction

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

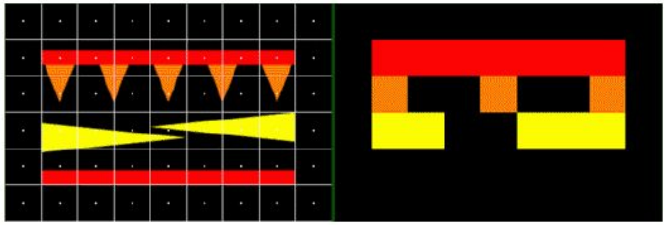
Examples of Aliasing

Jagged boundaries



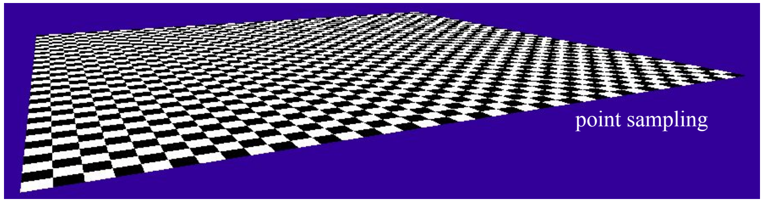
Examples of Aliasing

Improperly rendered detail



Examples of Aliasing

Texture Errors

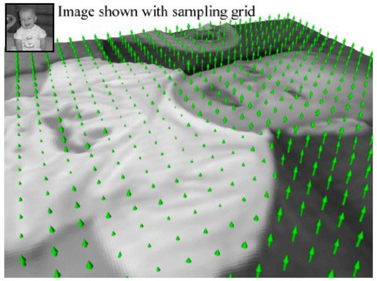


Today

- Monte-Carlo Integration
- Stratified Sampling & 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
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

Sampling Density

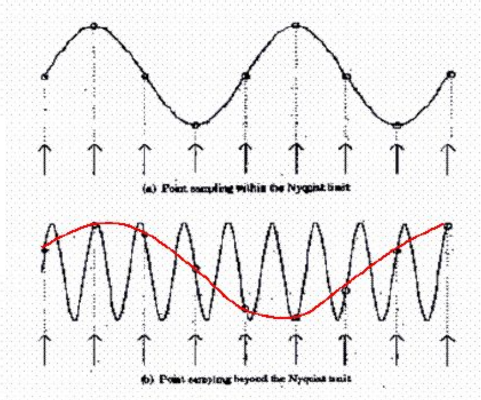
- How densely must we sample an image in order to capture its essence?
- If we under-sample the signal, we won't be able to accurately reconstruct it...



Sampling Density

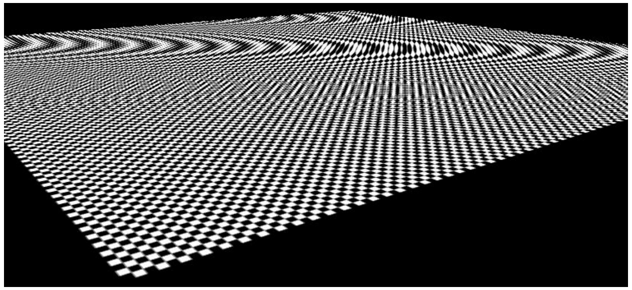
- If we insufficiently sample the signal, it may be mistaken for something simpler during reconstruction (that's aliasing!)

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



Sampling Density

- Aliasing in 2D because of insufficient sampling density

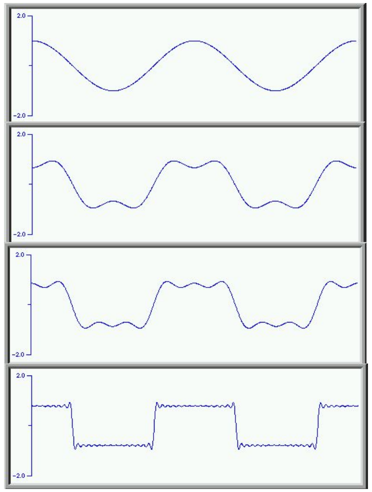


Remember Fourier Analysis?

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

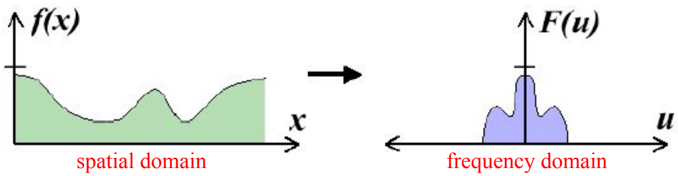
It's a shame that Signals & Systems is not required for CSCI majors...

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



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

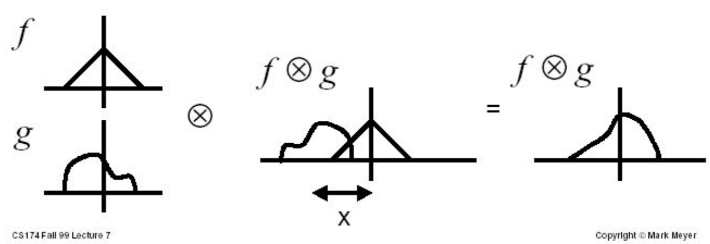
Fourier Transform $$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\, e^{-i2\pi(ux+vy)}\, dx\, dy$$

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, $f(x)$.

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



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

Remember Convolution?

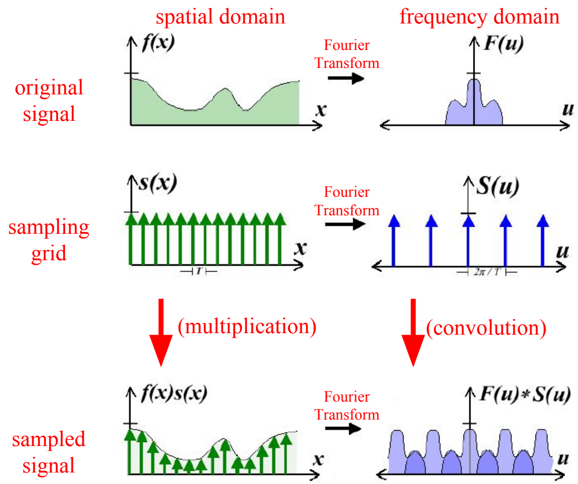
- Some operations that are difficult to compute in the spatial domain can be simplified by transforming to its dual representation in the frequency domain.
- For example, convolution in the spatial domain is the same as multiplication in the frequency domain.

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

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

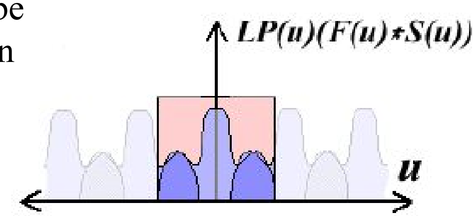
$$F(u) * H(u) \rightarrow f(x)h(x)$$

Sampling in the Frequency Domain



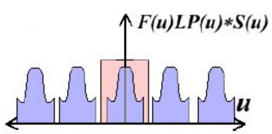
Reconstruction

- 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.

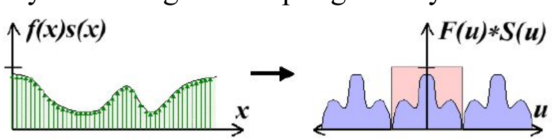


Guaranteeing Proper Reconstruction

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



- Separate by increasing the sampling density



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

Sampling Theorem

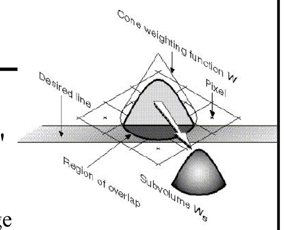
- When sampling a signal at discrete intervals, the sampling frequency must be *greater than twice* the highest frequency of the input signal in order to be able to reconstruct the original perfectly from the sampled version (Shannon, Nyquist)

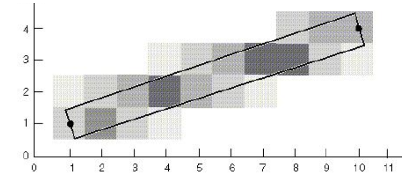
Today

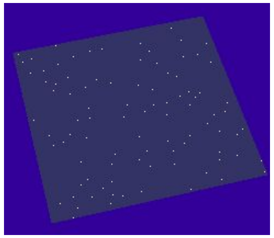
- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
  - Ideal, Gaussian, Box, Bilinear, Bicubic
- Anti-Aliasing for Texture Maps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

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)







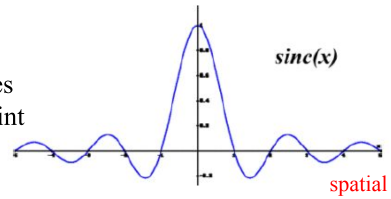
Source: Foley, VanDam, Feiner, Hughes - Computer Graphics, Second Edition,

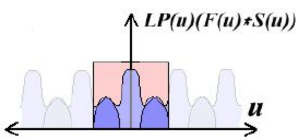
Filters

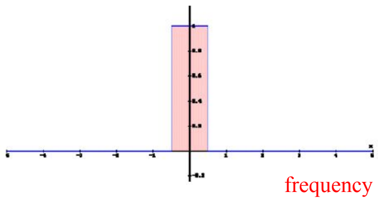
- Filters are used to
  - reconstruct a continuous signal from a sampled signal (reconstruction filters)
  - band-limit continuous signals to avoid aliasing during sampling (low-pass filters)
- Desired frequency domain properties are the same for both types of filters
- Often, the same filters are used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

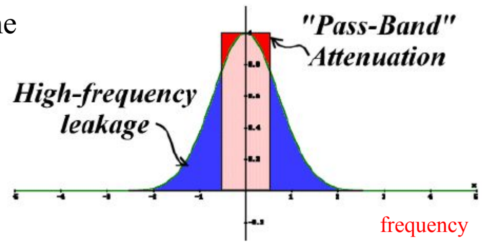






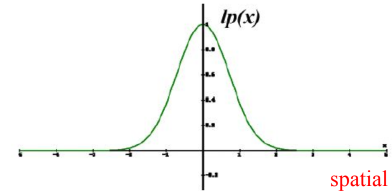
Problems with Practical Filters

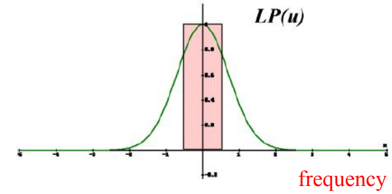
- Many visible artifacts in re-sampled images are caused by poor reconstruction filters
- Excessive pass-band attenuation results in blurry images
- Excessive high-frequency leakage causes "ringing" and can accentuate the sampling grid (anisotropy)



Gaussian Filter

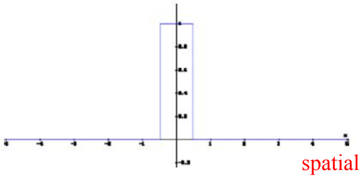
- This is what a CRT does for free!

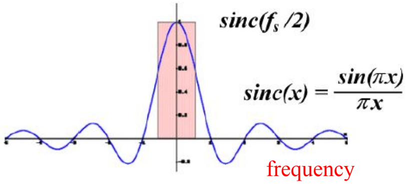




Box Filter / Nearest Neighbor

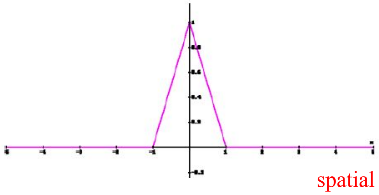
- Pretending pixels are little squares.

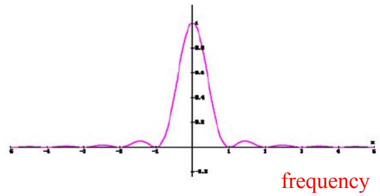




Tent Filter / Bi-Linear Interpolation

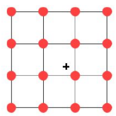
- Simple to implement
- Reasonably smooth

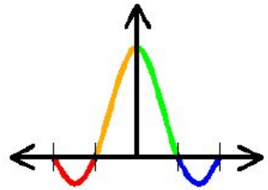


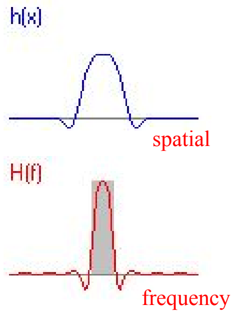


Bi-Cubic Interpolation

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





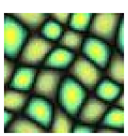


Today

- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
  - Magnification & Minification, Mipmaps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

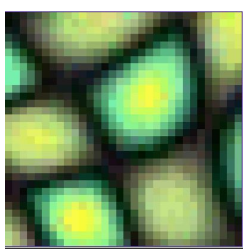
Sampling Texture Maps

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

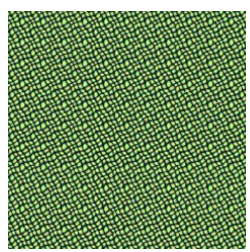


64x64 pixels

Original Texture



Magnification for Display

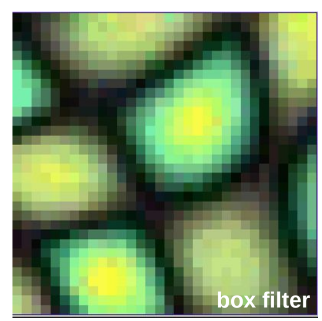


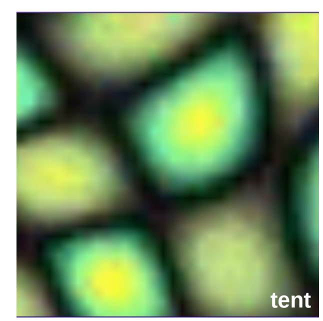
Minification for Display

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)

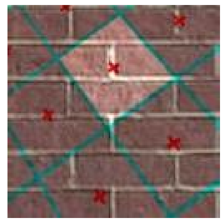




Spatial Filtering

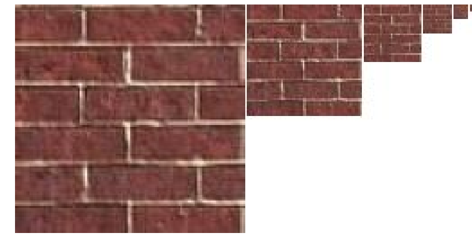
- Remove the high frequencies which cause artifacts in texture minification.
- Compute a spatial integration over the extent of the pixel
- This is equivalent to convolving the texture with a filter kernel centered at the sample (i.e., pixel center)!
- Expensive to do during rasterization, but an approximation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

projected texture in image plane



box filter in texture plane

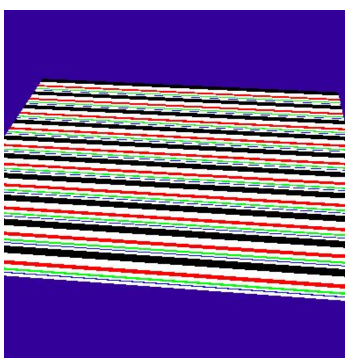
MIP Mapping



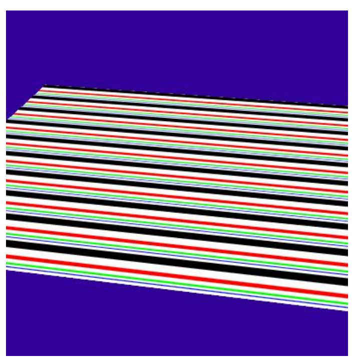
- Construct a pyramid of images that are pre-filtered and re-sampled at 1/2, 1/4, 1/8, etc., of the original image's sampling
- During rasterization we compute the index of the decimated image that is sampled at a rate closest to the density of our desired sampling rate
- MIP stands for *multum in parvo*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

MIP Mapping Example

- Thin lines may become disconnected / disappear



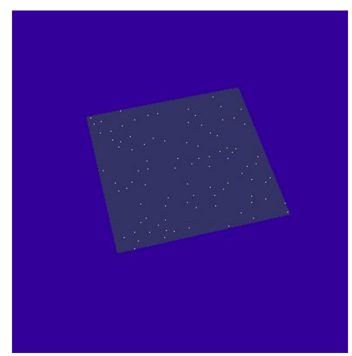
Nearest Neighbor



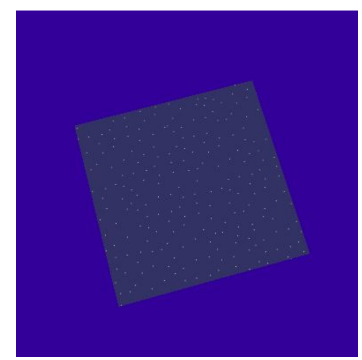
MIP Mapped (Bi-Linear)

MIP Mapping Example

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



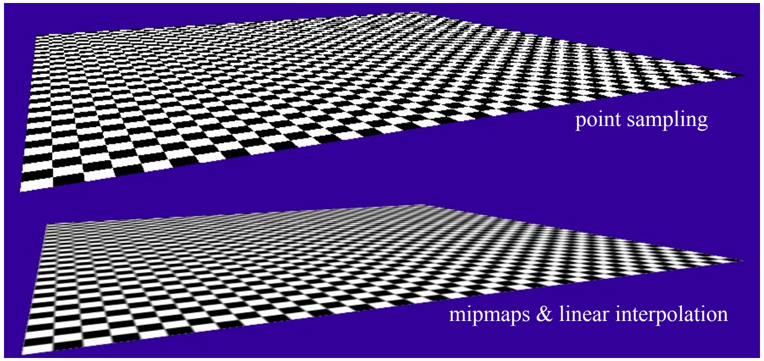
Nearest Neighbor



MIP Mapped (Bi-Linear)

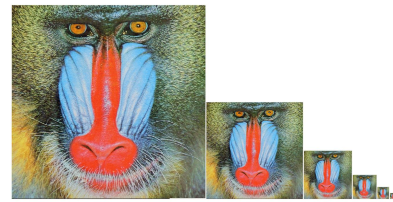
Examples of Aliasing

Texture Errors

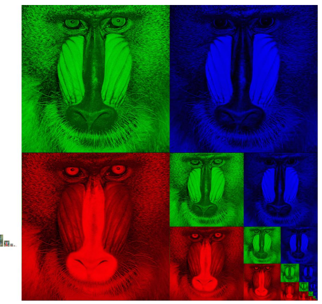


Storing MIP Maps

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



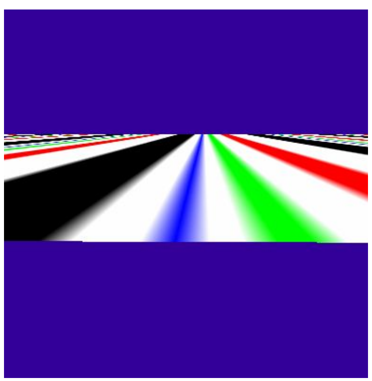
10-level mip map



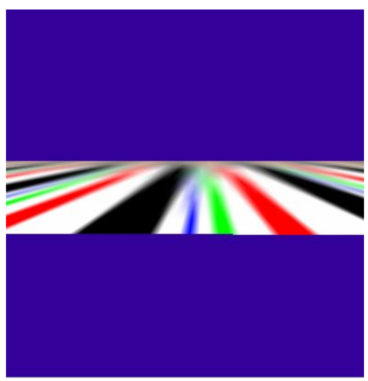
Memory format of a mip map

Anisotropic MIP-Mapping

- What happens when the surface is tilted?

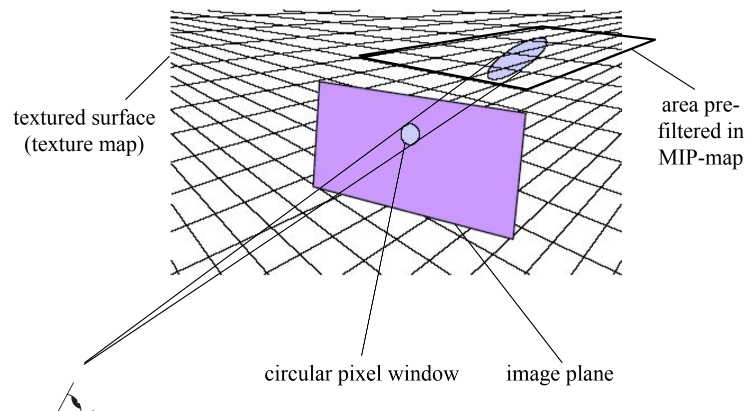


Nearest Neighbor



MIP Mapped (Bi-Linear)

Anisotropic MIP-Mapping



- Square MIP-map area is a bad approximation

Anisotropic MIP-Mapping

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

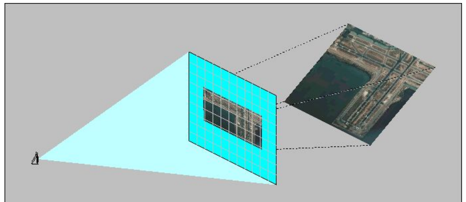


Figure 25. Geometry Orientation and Texture Aspect Ratio

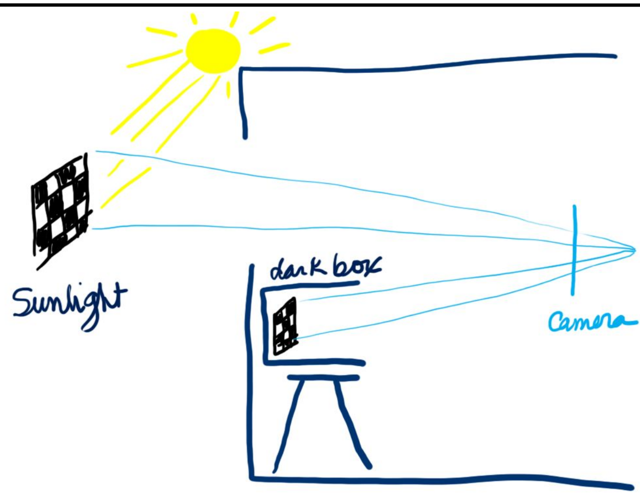
Figure 24. Creating a Set of Anisotropically Filtered Images

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

Today

- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is a Pixel?
- Examples of Aliasing
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Intro to High Dynamic Range (HDR) & Tone Mapping Algorithms

High Dynamic Range Example:



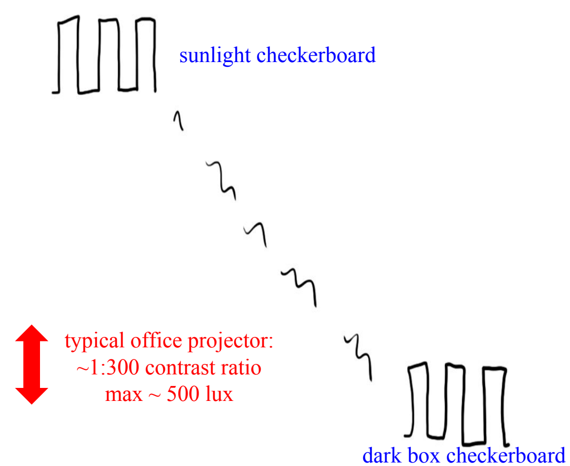
Illuminance & typical Lux values:

Direct sunlight:
> 100,000 lux

Overcast day/
TV studio lighting:
~ 1,000 lux

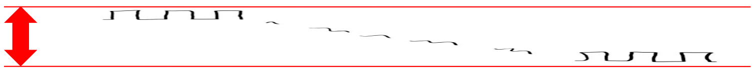
Office lighting:
~500 lux

Moonlight: 1 lux

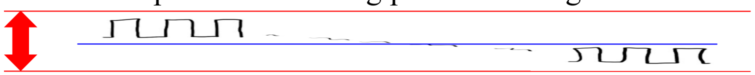


Tone Mapping

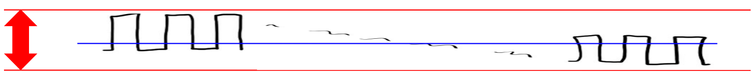
- Convert high dynamic range (HDR) data to low dynamic range (LDR)
  - Linear Scale: loss of contrast & precision



  - Nonlinear Scale: preserve more contrast & precision in important/interesting/prominent ranges

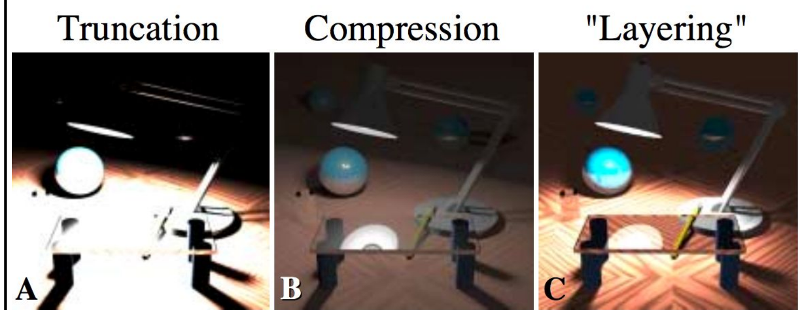


  - Spatially-varying Scaling:



Readings for Friday: *(pick one)*

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



Readings for Friday: *(pick one)*

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