Monte-Carlo, Sampling, Aliasing, & Mipmaps

The Parthenon, 2004

Paul Debevec et al.
The Parthenon, 2004

Paul Debevec et al.

Last Time?

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

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

Monte Carlo Ray Tracing

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

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

Note: Always sample the primary light
Irradiance Cache

- The indirect illumination is smooth
- Store the indirect illumination

Interpolate nearby cached values
- But do full calculation for direct lighting
Irradiance Cache

Photon Mapping

• Preprocess: cast rays from light sources
  – independent of viewpoint
Photon Mapping

- Store photons
  - position + light power + incoming direction

Storing the Photon Map

- Efficiently store photons for fast access
- Use hierarchical spatial structure (kd-tree)
Rendering with Photon Map

- Cast primary rays
- For secondary rays: reconstruct irradiance using k closest photons
- Combine with irradiance caching and other techniques

Photon Map Results
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 a query point & radius (red circle)
- KDTree::CollectPhotonsInBox
  - Input: a bounding box (yellow)
  - Finds all k-d tree cells (blue) that overlap with the bounding box
  - Returns all photons in those cells
- You must discard all photons not in your original query radius
- Should it return only return the photons in the yellow box? Why not?
Today

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

Readings for Today: *(pick one)*

“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 substra for N-rooks sampling. Samples are jittered within the subcells.*

*Figure 2: With correlated shuffling.*

*Figure 3: Polar warp with $n = 22$, $a = 7$.*

Readings for Today: *(pick one)*

“Implicit Visibility and Antiradiance for Interactive Global Illumination”

Dachsbacher, Stamminger, Drettakis, and Durand
Siggraph 2007

---

Today

- Worksheet for Today
- Papers for Today
  - Monte-Carlo Integration
    - Probabilities and Variance, Analysis
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- Filters in Computer Graphics
- Anti-Aliasing for Texture Maps
- Papers for Tuesday
Monte-Carlo Computation of $\pi$  

• 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  
  = $\pi / 4$

- $\pi \approx 4 \times \text{number inside disc} / \text{total number}$  
- 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  

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

Use this for HW3!
MC for Distributed Ray Tracing

- multiple shadow rays to sample area light source

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!
  - It might require re-weighting/normalizing samples

Sampling the source uniformly

Sampling the hemisphere uniformly

Convergence & Error

- Let’s “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 accuracy
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

• Worksheet for Today
• Papers for Today
• Monte-Carlo Integration
• Stratified Sampling & Importance Sampling
• What is Aliasing?
• Sampling & Reconstruction
• Filters in Computer Graphics
• Anti-Aliasing for Texture Maps
• Papers for Tuesday
Stratified Sampling

- With uniform sampling, we can get unlucky
  - E.g. all samples in a corner

- To prevent it, subdivide domain $\Omega$ into non-overlapping regions $\Omega_i$
  - Each region is called a stratum

- Take one random samples per $\Omega_i$

Stratified Sampling Example

<table>
<thead>
<tr>
<th>$N$</th>
<th>$I$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.75039</td>
</tr>
<tr>
<td>10</td>
<td>1.9893</td>
</tr>
<tr>
<td>100</td>
<td>1.79139</td>
</tr>
<tr>
<td>1000</td>
<td>1.75146</td>
</tr>
<tr>
<td>10000</td>
<td>1.77313</td>
</tr>
<tr>
<td>100000</td>
<td>1.77862</td>
</tr>
</tbody>
</table>

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

Stratified
$O(1/N)$

Slide from Henrik Wann Jensen
**Sampling Options**

- **uniform sampling** (or uniform random)
  - all samples weighted equally

- **dense sampling** where function has greater magnitude
  - less sensitive to choice of samples
  - 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

---
Uniform vs. Importance Sampling

$U(\omega_i)$

$P(\omega_i)$

5 Samples/Pixel

25 Samples/Pixel

Slide from Jason Lawrence
Uniform vs. Importance Sampling

- Bidirectional Path Tracing


Figure B: An indirectly illuminated scene rendered using path tracing and bidirectional path tracing respectively. The latter method results in visibly less noise for 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 for Today
• Papers for Today
• Monte-Carlo Integration
• Stratified Sampling & Importance Sampling
• What is Aliasing?
• Sampling & Reconstruction
• Filters in Computer Graphics
• Anti-Aliasing for Texture Maps
• Papers for Tuesday
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
- 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 definition 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:

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

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

The same analysis can be applied to geometric objects:

Examples of Aliasing

- Aliasing occurs because of sampling and reconstruction
Examples of Aliasing

Jagged boundaries

Examples of Aliasing

Improperly rendered detail
Examples of Aliasing

Texture Errors

Today

• Worksheet for Today
• Papers for Today
• 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 Tuesday
Sampling Density

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


Sampling Density

- Aliasing in 2D because of insufficient sampling density
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.

• 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.

\[ F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) e^{-i2\pi(ux+vy)} dxdy \]

\[ f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) e^{i2\pi(ux+vy)} dudv \]

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/
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

- Original signal is transformed to its frequency domain representation using the Fourier Transform.
- Sampling in the spatial domain results in a sampled signal.
- Sampling the original signal in the spatial domain is equivalent to multiplying its Fourier Transform by a sampling grid in the frequency domain.
- Convolution in the frequency domain corresponds to multiplication in the spatial domain.
- Multiplication in the frequency domain corresponds to convolution in the spatial 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 \textit{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 for Today
- Papers for Today
- Monte-Carlo Integration
- Stratified Sampling & Importance Sampling
- What is Aliasing?
- Sampling & Reconstruction
- \textbf{Filters in Computer Graphics}
  - Ideal, Gaussian, Box, Bilinear, Bicubic
- Anti-Aliasing for Texture Maps
- Papers for Tuesday
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)

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

- Worksheet for Today
- Papers for Today
- 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 Tuesday

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

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
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.
MIP Mapping Example

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

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](image1)
![Memory format of a mip map](image2)

Anisotropic MIP-Mapping

- What happens when the surface is tilted?

![Nearest Neighbor](image3)
![MIP Mapped (Bi-Linear)](image4)
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

Today

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