Robust Adaptive Photon Tracing using Photon Path Visibility

Toshiya Hachisuka\textsuperscript{1,2} Henrik Wann Jensen\textsuperscript{2}

\textsuperscript{1}Aarhus University \textsuperscript{2}UC San Diego

Invisible paths = wasted computation
Metropolis Light Transport

[Veach and Guibas 97]
Metropolis Light Transport

Reference

Metropolis Light Transport [100 min]
Specular-Diffuse-Specular

- Fundamental limitation of MC integration [Veach 98]
- Missing “reflections of caustics from a point light”
- Applicable to all local path sampling methods
- True for small area lights in practice [Hachisuka et al. 08]
Specular-Diffuse-Specular
Specular-Diffuse-Specular
Specular-Diffuse-Specular
Specular-Diffuse-Specular
Progressive Photon Mapping
Inefficient Case

Progressive Photon Mapping [100 min]
Ideal

• Can we combine these two algorithms?
  • **MLT**: Efficient for difficult lighting scenarios
  • **PPM**: Robust to complex types of light paths
Contributions

MLT + PPM
Contributions

MLT + PPM

Simple, fast, general, and easy to use
Method
Progressive Photon Mapping

Eye pass

Photon pass
Eye Pass
Eye Pass
Photon Pass

Photons
Photon Pass

Range query
Photon Pass

Radius reduction
Next Photon Pass
Next Photon Pass
Next Photon Pass
Next Photon Pass
Rendering

Use accumulated statistics
Path Visibility

Wasted light paths
Key Observation

- We can determine whether a photon path is visible or not
- Because PPM stores visible points from the eye
- Contributed to at least one visible point = visible
Path Visibility

Visible path

Invisible paths
Our Method
Our Method

Initial visible path
Our Method

Perturbation ("Mutation")
Our Method

Visibility check by Range query
Our Method
Our Method

Reject
(Restore the original)
Our Method
Our Method
Our Method

Accept
(Replace the original)
Our Method
Our Method
Our Method
Our Method
Sampling using Path Visibility

- Three technical components
- Mapping to the primary space
- Definition of the visibility function
- Sampling via Markov chain Monte Carlo
Primary Space

- Mapping a path to a point in hypercube [Kelemen et al. 2002]
- Path = vector of random numbers $\vec{u} = (\xi_1, \cdots, \xi_N) \in (0, 1)^N$
Visibility Function

- Binary function in the primary space

\[ V(\vec{u}) = \begin{cases} 
0 & \text{No contribution} \\
1 & \text{Contributed to at least one measurement point} 
\end{cases} \]
Markov Chain Monte Carlo

- Importance sampling the visibility function
- Based on perturbation & accept/reject
- Sampling visible paths only = Sampling only points with $V(\vec{u}) = 1$
Two Issues

- Markov chain Monte Carlo tends to...
  - be sensitive with parameter tuning
  - get trapped in small regions of the image
Problem 1: Parameter Tuning

- Amount of perturbation (mutation size) affects the efficiency
- Scene dependent and unintuitive to tweak

Too small

Appropriate

Too large
Solution 1: Adaptive MCMC

- Self-tuning Markov chain Monte Carlo methods
- Relatively new techniques in statistics [Andrieu and Thomas 08]
- Uses all the past samples to tweak parameters
Problem 2: Insufficient Exploration

- Markov chain can get trapped to a small region in the domain
- “Sampling one of the windows too many times”
Solution 2: Replica Exchange

- Sampling multiple distributions simultaneously
- Chain moves across distributions (= exchange)
- Bridging peaks via “easy” distributions

\[ p(\vec{u}) \propto V(\vec{u}) \quad q(\vec{u}) = \text{const.} \]
Solution 2: Replica Exchange

- Sampling multiple distributions simultaneously
- Chain moves across distributions (= exchange)
- Bridging peaks via “easy” distributions

\[ p(\vec{u}) \propto V(\vec{u}) \quad q(\vec{u}) = \text{const.} \]
**Solution 2: Replica Exchange**

- Sampling multiple distributions simultaneously
- Chain moves across distributions (exchange)
- Bridging peaks via "easy" distributions

\[ p(\vec{u}) \propto V(\vec{u}) \]

\[ q(\vec{u}) = \text{const.} \]
Results
Implementation

• Simple extension over a standard PPM code
  • Just enable generation of a path from a random vector
  • Successfully used as an assignment in a graphics course
Our method

Uniform

1 min
15 min
30 min
60 min
Our method

(Equal time comparisons)
Number of Visible Photons

Number of Visible Photons [M photons]

- **Cognac**
- **Door**
- **Watch**
- **Room**

Our method
“Easy” Scenes

- Does not hurt rendering times of “easy” scenes
Limitations

- Does not resolve noise due to BRDFs
Limitations

- Does not resolve noise due to BRDFs

See our paper for a solution:
"A Path Space Extension for Robust Light Transport Simulation"
(a.k.a. “Vertex Connection and Merging” by Georgiev et al.)
Limitations

- No proof of convergence
Limitations

- No proof of convergence

- Kaplanyan and Dachsbacher proved convergence on essentially the same problem: “Path Space Regularization for Holistic and Robust Light Transport”, Eurographics 2013
Conclusions

- Adaptive photon tracing based on visibility
- Samples only visible photon paths
- Completely parameter-free
- Efficient, simple, and practical
Conclusions

• Adaptive photon tracing based on visibility
• Samples only visible photon paths
• Completely parameter-free
• Efficient, simple, and practical
• You should have already tested it two years ago (if not, do so tonight)
Thank You

- Youichi Kimura (Studio Azurite) for providing us the room model and some inspiration images

- Marko Dabrovic for the Sibenik Cathedral model

- VC-ISTI for the Budda model