Probabilistic PPM

Claude Knaus   Matthias Zwicker
University of Bern

State of the Art in Photon Density Estimation

Modified slides and presentation by Toshiya Hachisuka
Probabilistic PPM

- Alternative derivation of PPM
  - Fixed radius reduction, no need for statistics
  - Asymptotic convergence analysis
  - Trivial to implement
Bias-variance trade-off

Radiance estimation

Photons
Bias-variance trade-off

Radiance estimation

Radiance estimation kernel

Photons
Bias-variance trade-off

- Larger kernels
  - Lower variance
  - Higher bias

Radiance estimation

Radiance estimation kernel

Photons
Bias-variance trade-off

- Larger kernels
  - Lower variance
  - Higher bias

- Smaller kernels
  - Higher variance
  - Lower bias
Bias-variance trade-off

- Larger kernels
  - Lower variance
  - Higher bias

- Smaller kernels
  - Higher variance
  - Lower bias

- Vanishing variance and bias
  - Infinitely many photons
  - Infinitely small kernels

Radiance estimation

Radiance estimation kernel

Photons
Progressive photon mapping - recap

- Achieve reduction of variance and bias at the same time
Progressive photon mapping - recap

- Achieve reduction of variance and bias at the same time

- Basic algorithm
  - Iterate over photon mapping steps
  - Reduce kernel size in each step
  - Accumulate results
Progressive photon mapping - recap

- Achieve reduction of variance and bias at the same time

- Basic algorithm
  - Iterate over photon mapping steps
  - Reduce kernel size in each step
  - Accumulate results

- Key advantages
  - Error vanish over iterations (just like path tracing)
  - No memory bottleneck
  - Robust
Strategy to reduce kernel radius

- Original PPM [SIGGRAPH Asia 2008]
- Reduce kernel based on sample statistics
Strategy to reduce kernel radius

- Original PPM [SIGGRAPH Asia 2008]
  - Reduce kernel based on sample statistics

- Probabilistic PPM [ACM TOG 2011]
  - Reduce kernel based on expected statistics
Strategy to reduce kernel radius

- Original PPM [SIGGRAPH Asia 2008]
  - Reduce kernel based on sample statistics

- Probabilistic PPM [ACM TOG 2011]
  - Reduce kernel based on expected statistics
Strategic to reduce kernel radius

- Original PPM [SIGGRAPH Asia 2008]
  - Reduce kernel based on sample statistics

- Probabilistic PPM [ACM TOG 2011]
  - Reduce kernel based on expected statistics
  - Also known as recursive kernel density estimation
Probabilistic analysis

- Consider the expected behavior of radiance estimation
Probabilistic analysis

- Consider the expected behavior of radiance estimation

\[ \text{Noise} \propto \frac{1}{r^2} \]
Probabilistic analysis

- Consider the expected behavior of radiance estimation

\[ \text{Noise} \propto \frac{1}{r^2} \]

\[ \text{Bias} \propto r^2 \]

- Noise vanishes
- Bias increases
Radius reduction

- Iteration step $i$
- Radiance estimation radius $r_i$
- Parameter $0 < \alpha < 1$ (same as original PPM)

\[
\frac{r_{i+1}^2}{r_i^2} = \frac{i + \alpha}{i + 1}
\]

Theory and derivation [Knaus and Zwicker 2011]
Expected statistics

Bias of average

Bias per iteration

N
Expected statistics

\[ O\left(\frac{1}{N^{1-\alpha}}\right) \]
Expected statistics

\[ O\left(\frac{1}{N^{1-\alpha}}\right) \]
Expected statistics

\[ O\left(\frac{1}{N^{1-\alpha}}\right) \]

\[ O\left(\frac{1}{N^\alpha}\right) \]
Original PPM and Probabilistic PPM

Original
\[
\frac{r_{i+1}^2}{r_i^2} = \frac{N_i + \alpha M_i}{N_i + M_i}
\]

Probabilistic
\[
\frac{r_{i+1}^2}{r_i^2} = \frac{i + \alpha}{i + 1}
\]

Local Statistics
No Local Statistics!
Implementation
Implementation

\[ r \leftarrow r \sqrt{\frac{i + \alpha}{i + 1}} \]

Global Reference Radius

Photon Mapper

Black Box

Average Images
Implementation

Can be implemented via scripting (and indeed done with pbrt)!
Implementation

- Your photon mapper supports fixed-radius range query
- PPM ready
- Just change the radius according to the equation
Your photon mapper supports fixed-radius range query
  ▶️ PPM ready
  ▶️ Just change the radius according to the equation

Your photon mapper only does kNN query
  ▶️ PPM ready if there is a “max. radius” parameter
  ▶️ To emulate fixed-radius range query
    ▶️ “k” in kNN = # of stored photons per iteration
    ▶️ “max. radius” = radius for range query
Your photon mapper supports fixed-radius range query
- PPM ready
- Just change the radius according to the equation

Your photon mapper only does kNN query
- PPM ready if there is a “max. radius” parameter
- To emulate fixed-radius range query
  - “k” in kNN = # of stored photons per iteration
  - “max. radius” = radius for range query

In both cases, take the average of output images
Implementation
Arbitrary Kernels

Constant  Gaussian  SIGGRAPH
Stochastic Effects
Comparison with original PPM

Original     Probabilistic     20x Difference
Participating Media

\[
\frac{r^2_{i+1}}{r^2_i} = \frac{i + \alpha}{i + 1}
\]
Participating Media

\[
\frac{r_{i+1}^3}{r_i^3} = \frac{i + \alpha}{i + 1}
\]
Participating Media

1 iteration (2 million photons)
Participating Media

- 10 iteration (20 million photons)
Participating Media

- 100 iteration (200 million photons)
Participating Media

- 1000 iteration (2 billion photons)
Original PPM vs Probabilistic PPM

- Original PPM
  - Based on *sample* statistics
  - *Need some modification* to the existing code
  - Accumulated statistics are *useful* for some applications
Original PPM
- Based on *sample* statistics
- *Need some modification* to the existing code
- Accumulated statistics are *useful* for some applications

Probabilistic PPM
- Based on *expected* statistics
- *No modification* to the existing code
- *Cannot use* accumulated statistics
Original PPM vs Probabilistic PPM

- **Original PPM**
  - Based on *sample* statistics
  - Need some modification to the existing code
  - Accumulated statistics are *useful* for some applications

- **Probabilistic PPM**
  - Based on *expected* statistics
  - No modification to the existing code
  - Cannot use accumulated statistics

- **My recommendation**
  - Start with probabilistic PPM
  - Gradually incorporate original PPM for more features
Various extensions on the basic PPM algorithm