Robust adaptive sampling for Monte-Carlo-based rendering
Anthony Pajot, Loïc Barthe, Mathias Paulin

To cite this version:
picked=prox 10.1145/2073304.2073359 hal-01518537

HAL Id: hal-01518537
https://hal.archives-ouvertes.fr/hal-01518537
Submitted on 4 May 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Robust Adaptive Sampling for Monte-Carlo-based rendering

Anthony Pajot, Loïc Barthe, Mathias Paulin

IRIT - University of Toulouse - France

Monte-Carlo rendering
- Value of each pixel defined as the expected value of a random variable \( X \)
  \[
  I = E[X]
  \]
- Estimated using samples of \( X \)
  \[
  \bar{I} = \frac{1}{N} \sum_{i=1}^{N} x_i
  \]

Previous work and their limitations
- Adaptive sampling based on the statistical nature of the estimation [Purgathofer 1987]
  Not a relative error: does not take into account dynamic reduction during tonemapping
- Adaptive sampling based on information-theoretic approaches and entropy measures [Xu et al 2007]
  Does not make the error uniform during rendering, thus less adapted for progressive or time-constrained rendering

Both approaches can lead to poor sampling due to low-samples error estimations which underestimate the actual error

Our approach
- Use a relative error to avoid focusing all processing power on areas receiving more energy. These zones are not necessarily the most noticeable after tonemapping, e.g., undersampling in shadows leads to highly visible noise
- Alternate between uniform and adaptive sampling, to ensure that error estimations of all pixels improve during rendering

Our goals
- Focus processing power where convergence is harder to reach during Monte-Carlo based rendering
- Make the error over the pixels uniform at any moment for progressive or time-constrained rendering

Monte-Carlo rendering
- Value of each pixel defined as the expected value of a random variable \( X \)
  \[
  I = E[X]
  \]
- Estimated using samples of \( X \)
  \[
  \bar{I} = \frac{1}{N} \sum_{i=1}^{N} x_i
  \]

Our proposal: approximate median

For each pixel, relative error:
\[
\epsilon_r(I) = \frac{\text{estimated error}}{\text{expected error}}
\]

Standard estimation:
\[
\epsilon_a(I) = \frac{\sum_{i=1}^{N} |x_i - \bar{x}|}{N} 
\]

Not robust to outliers

Goal: focus on bright spots

Relative error and robustness to outliers

Our proposal: approximate median

\[
\epsilon_m(I) = \frac{\sum_{i=1}^{N} |x_i - \bar{x}|}{M_x}
\]

\( M_x \) = average of median of chunks of samples

Alternation: avoid poor low-samples error estimation

Adaptive sampling based on error measure.

poor error estimate → poor sampling

Our proposal: alternate adaptive and uniform sampling

Complete scene.
Penumbra prone to poor sampling

Complete scene. Penumbra prone to poor sampling

Uniform sampling
Adaptive sampling, no alternation.
Patterns in the penumbra
Adaptive sampling, alternation.
No patterns in the penumbra

Comparison with Tsallis entropy [Xu, Sbert, Xinh and Zhan 2007]

Test scene
Noise measure for uniform sampling
Noise measure for Tsallis entropy
Noise measure for our method

Comparison with Tsallis entropy [Xu, Sbert, Xinh and Zhan 2007]

Test scene
Noise measure for uniform sampling
Noise measure for Tsallis entropy
Noise measure for our method