I3D symposium

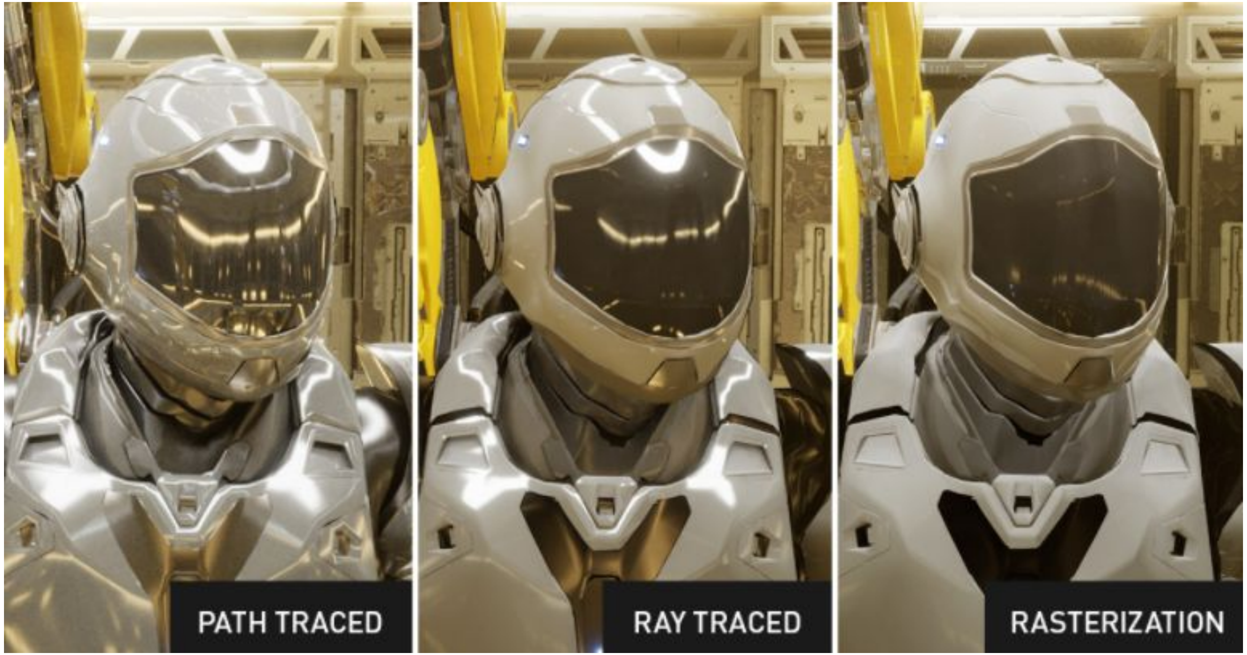
ACM SIGGRAPH SYMPOSIUM ON INTERACTIVE 3D GRAPHICS AND GAMES

Real-Time Markov Chain Path Guiding for Global Illumination and Single Scattering

Lucas Alber, Johannes Hanika and Carsten Dachsbacher

Karlsruhe Institute of Technology (KIT)

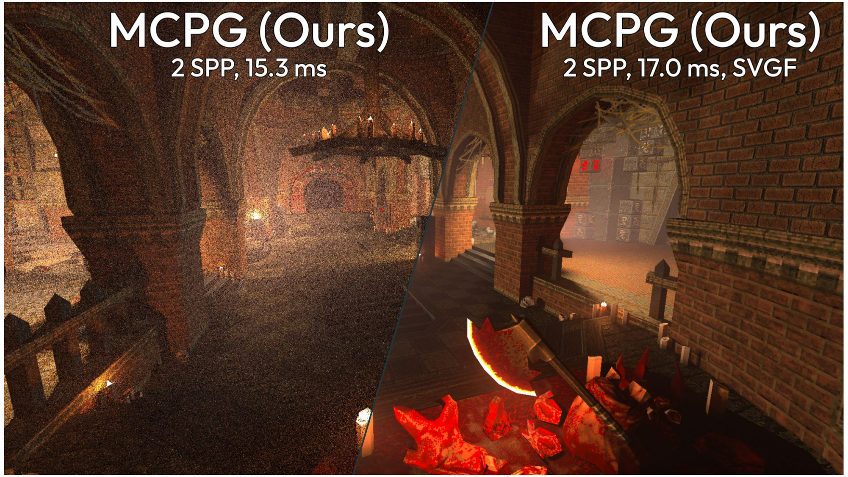
Motivation



https://blogs.nvidia.com/blog/what-is-path-tracing/

Motivation | Challenges

I3D symposium



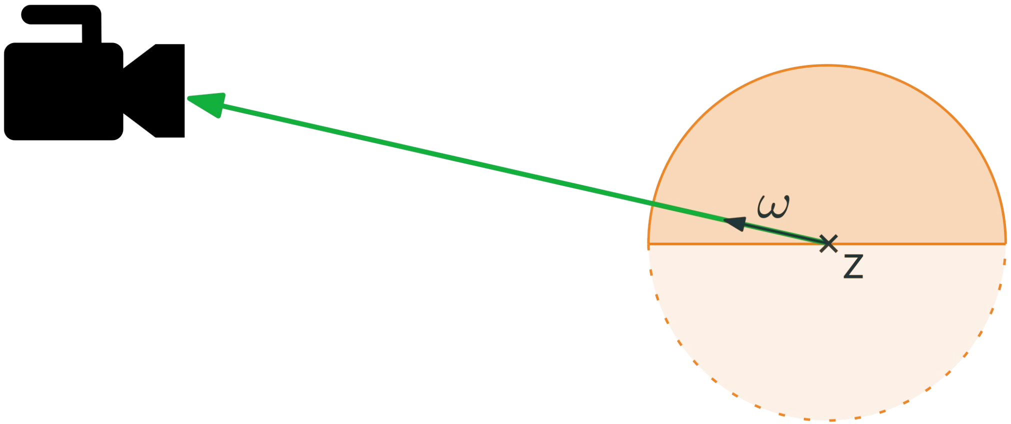
Motivation | Teaser

I3D symposium

Introduction | Render Equation [Kajiya, 1986]

I3D symposium

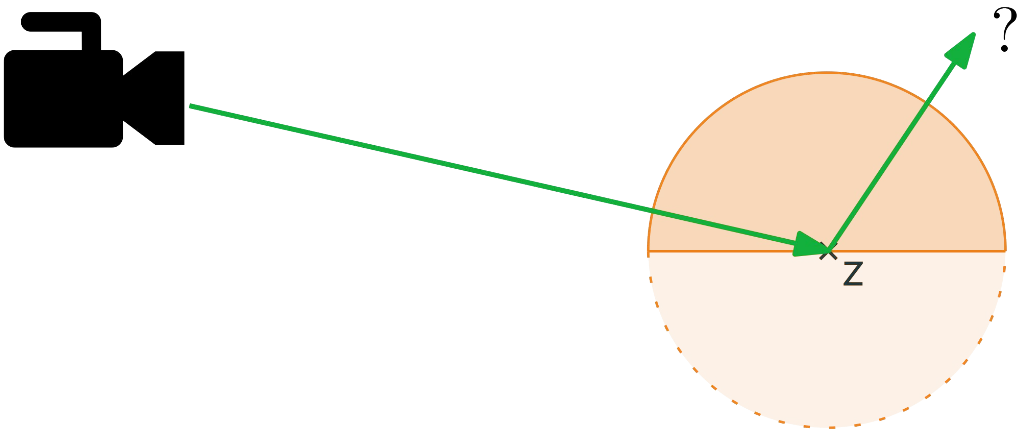
$$L(z, \omega) = L_e(z, \omega) + \int_{S^2} f_r(z, \omega_i, \omega) L_i(z, \omega_i) \, |n(z) \cdot \omega_i| \, d\omega_i \, ,$$



Introduction | Monte Carlo

I3D symposium

$$L(z,\omega) = L_e(z,\omega) + \int_{S^2} f_r(z,\omega_i,\omega) L_i(z,\omega_i)\, |n(z)\cdot\omega_i|\, d\omega_i\,,$$



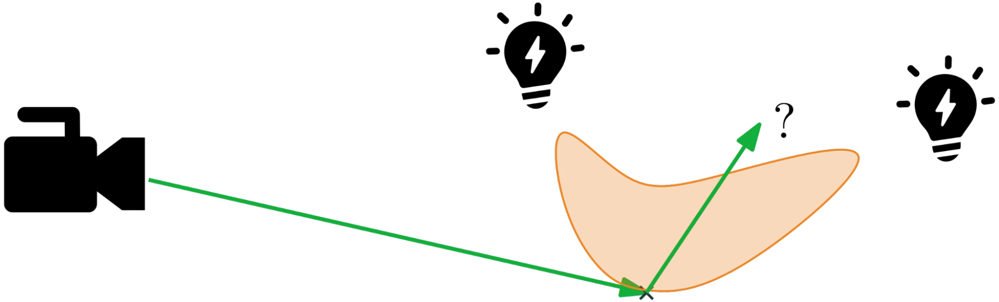
$$I = \int_\Omega f(x)\,dx$$

$$\langle I \rangle = \frac{1}{N}\sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)}$$

Introduction | BRDF Importance Sampling

Introduction | Path Guiding

I3D symposium



Learn the incident radiance to *guide* paths into important directions

- Many approaches exist for offline path tracing
- Limitations: non-adaptive, expensive fitting, dedicated learning phases, impractical for GPUs

Markov Chain Path Guiding (MCPG)

Markov Chain Path Guiding (MCPG)

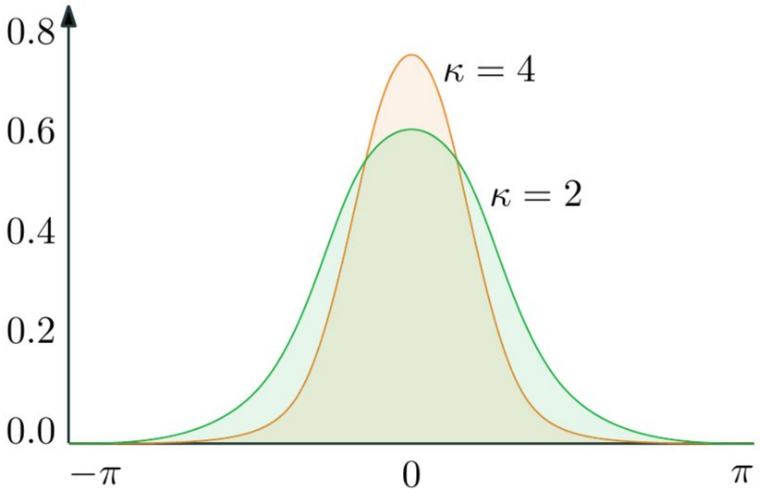
- Lightweight
- Unbiased
- Tailored to highly dynamic environments
- Extends naturally to single scattering

MCPG | von Mises-Fisher Distribution

I3D symposium

$$p(\boldsymbol{\omega} \mid t) = \frac{\kappa}{4\pi \sinh \kappa} \exp(\kappa \boldsymbol{\mu}^T \boldsymbol{\omega}) \,,$$

$$t = (\boldsymbol{\mu}, \kappa)$$

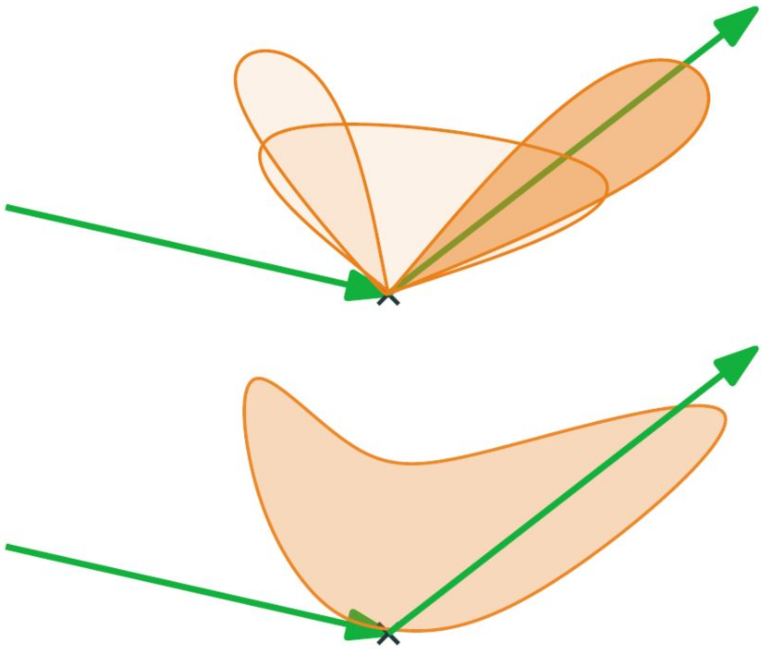


- Suited for directional data
- Efficient online fitting procedure [Ruppert et al., 2020]
- Simple and compact

MCPG | Mixture Model

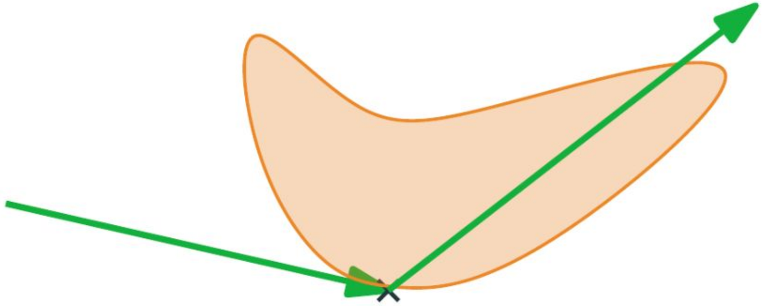
$$p(\omega) = \sum_{i=1}^{K} \pi_i p(\omega \mid t_i)\,,$$

$$p(\omega) = \int_{\mathcal{T}} p(t) p(\omega \mid t)\,\mathrm{d}t\,,$$



MCPG | Mixture Model

$$p(\omega) = \int_{\mathcal{T}} p(t) p(\omega \mid t) \, \mathrm{d}t \, ,$$



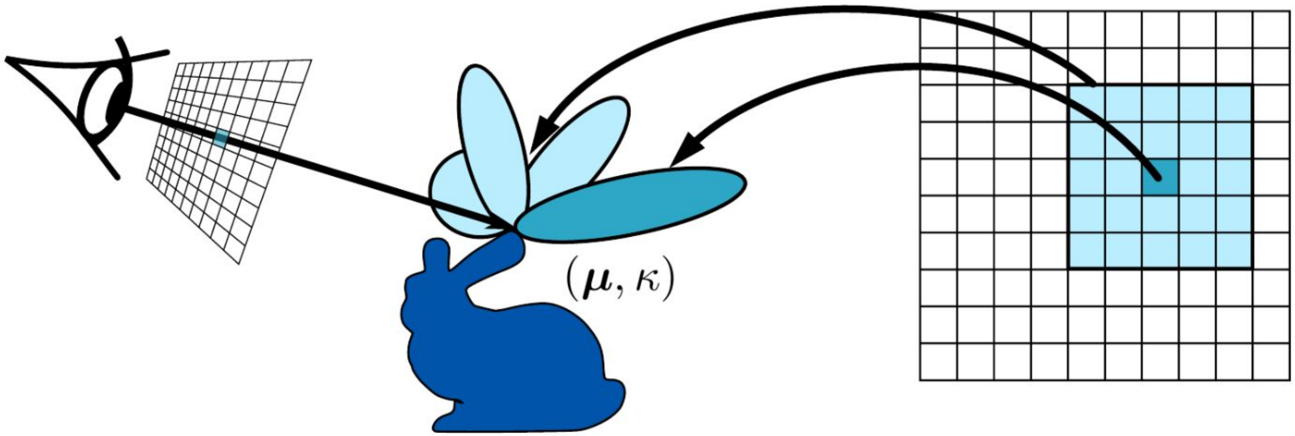
→ Stochastic Markov Chain Process

→ Evaluation using Stochastic Multiple Importance Sampling (SMIS) [West et al., 2020]

$$\langle I \rangle_{\text{SMIS}} = \sum_{i=1}^{N} \frac{f(\omega_i)}{\sum_{j=1}^{N} p(\omega_i \mid t_j)} \, , \qquad t_j \in \mathcal{T}$$

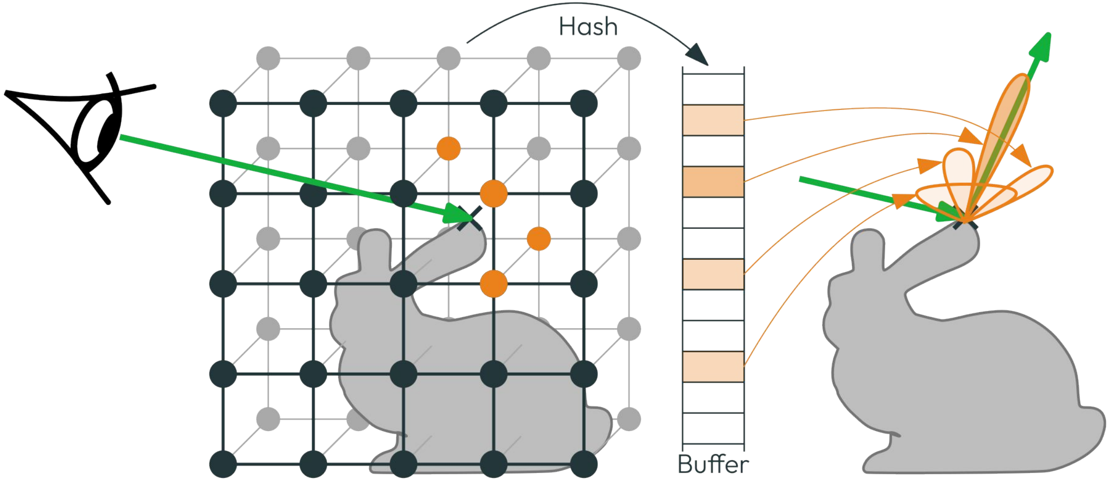
MCPG | Previous Work [Dittebrandt et al., 2023]

I3D symposium



- Run a Markov Chain per pixel
- Exchange vMF parameters with neighbor pixels

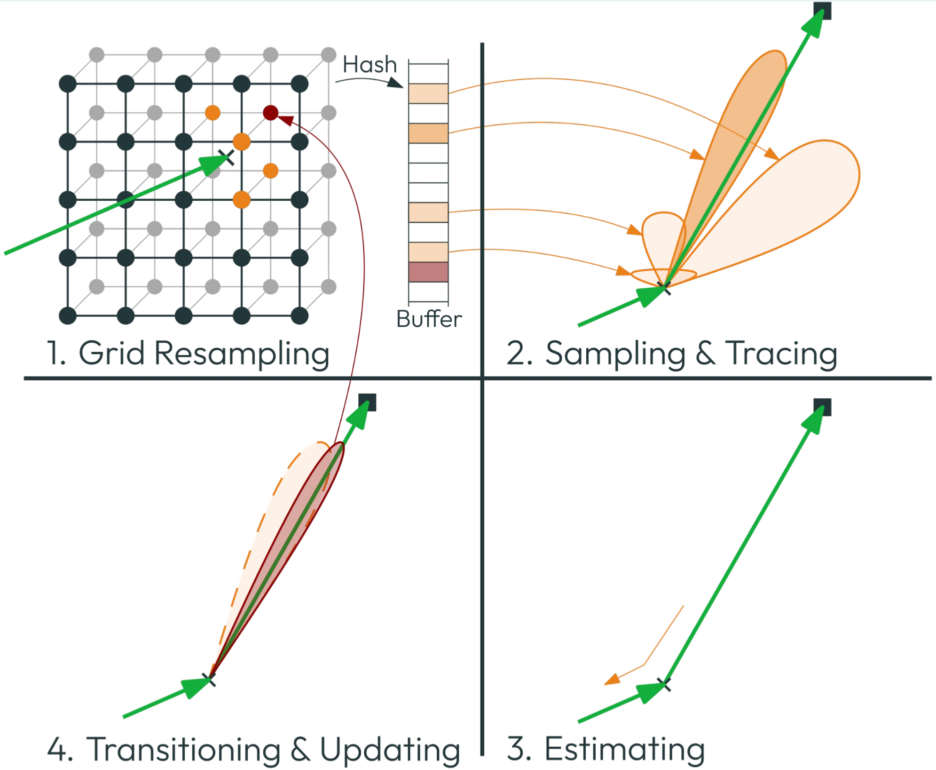
MCPG | Overview



- Run a Markov Chain per hash grid vertex
- Exchange vMF parameters with neighboring cells

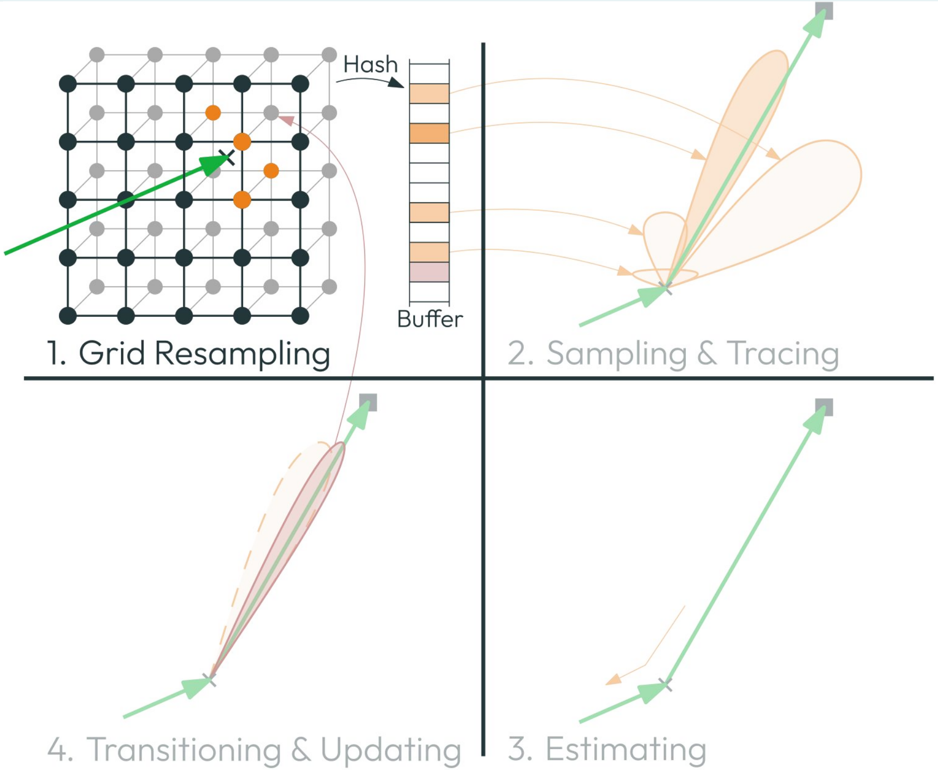
MCPG | Procedure Overview

I3D symposium

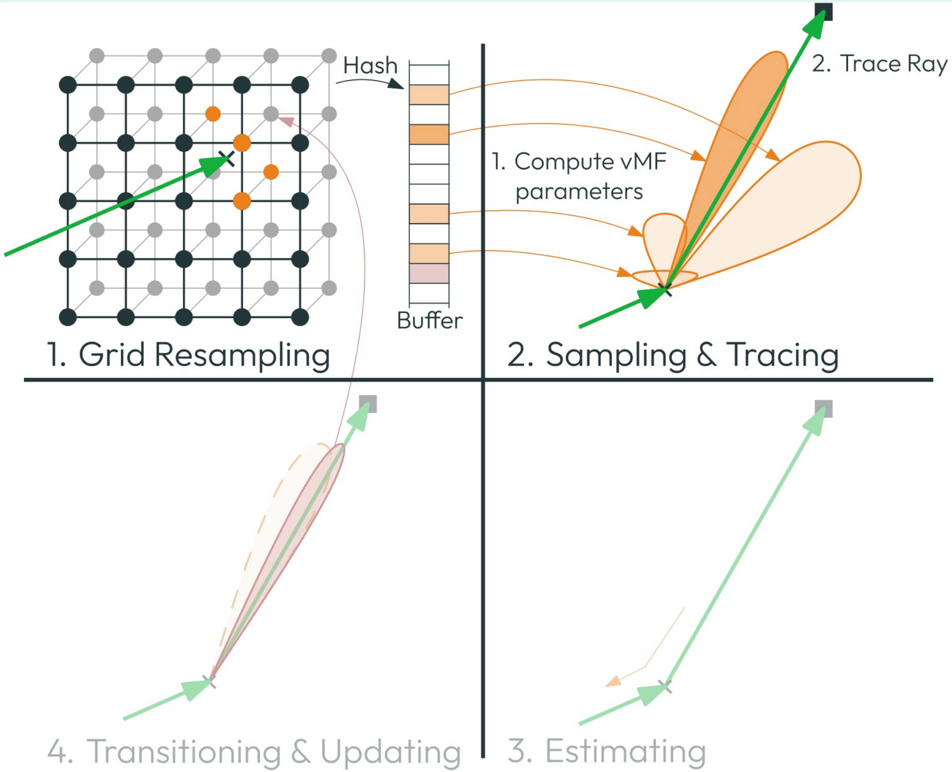


MCPG | Procedure

I3D symposium

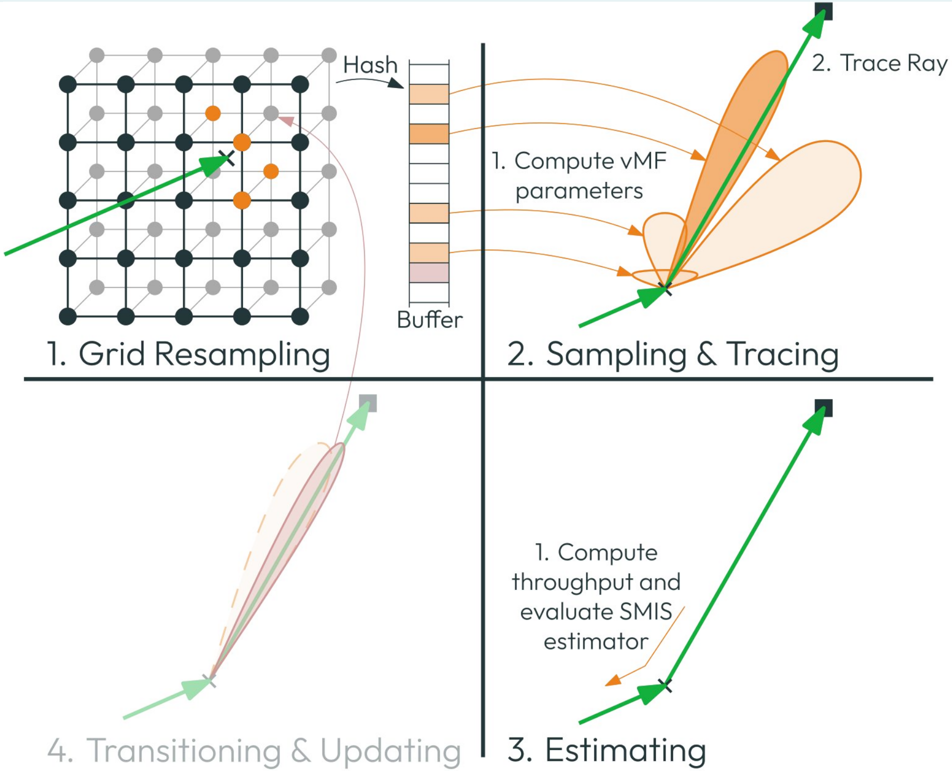


MCPG | Procedure



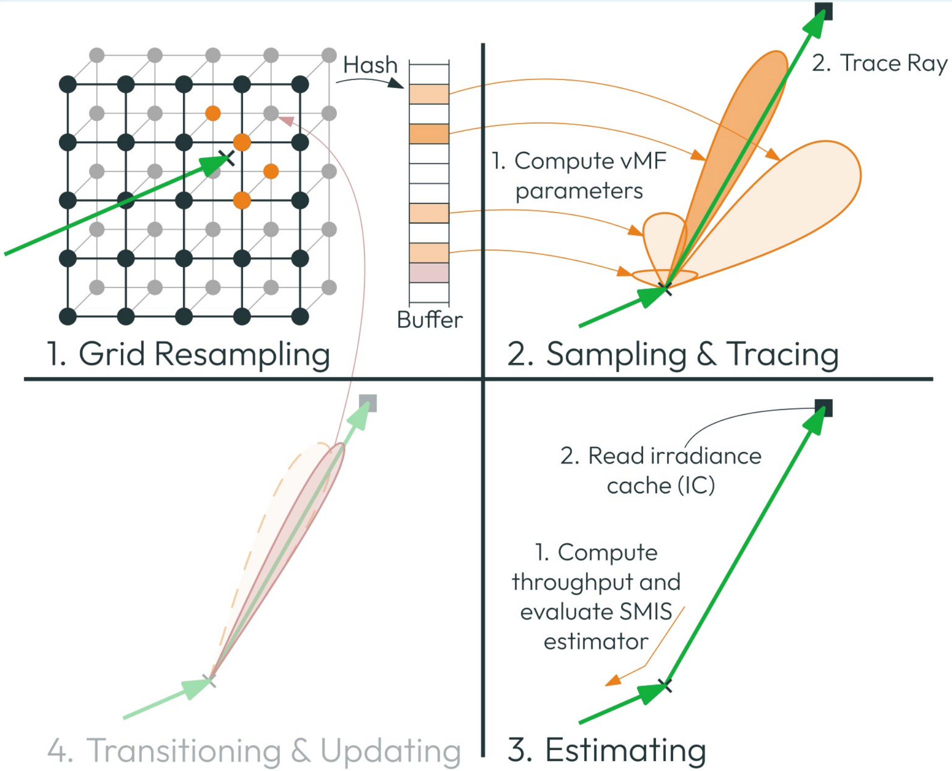
MCPG | Procedure

I3D symposium



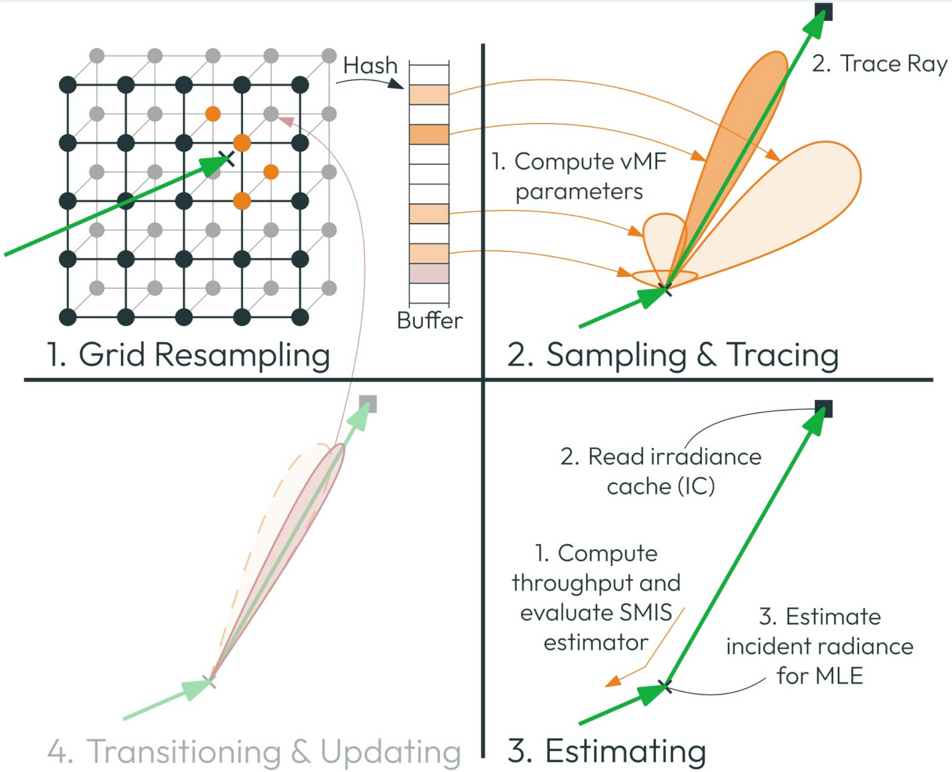
MCPG | Procedure

I3D symposium

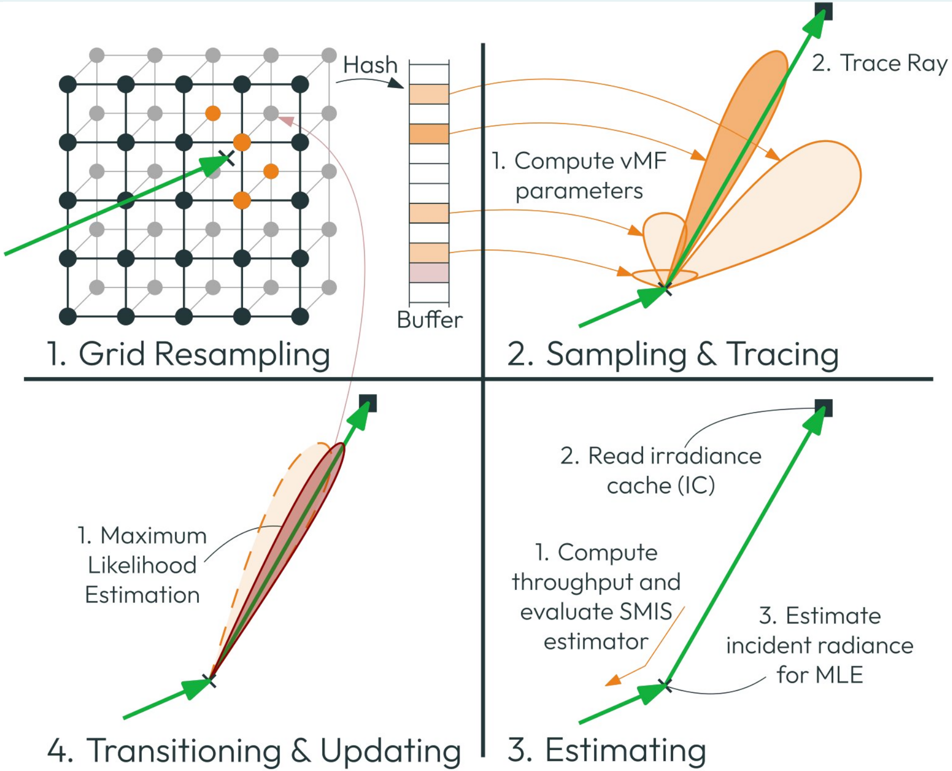


MCPG | Procedure

I3D symposium

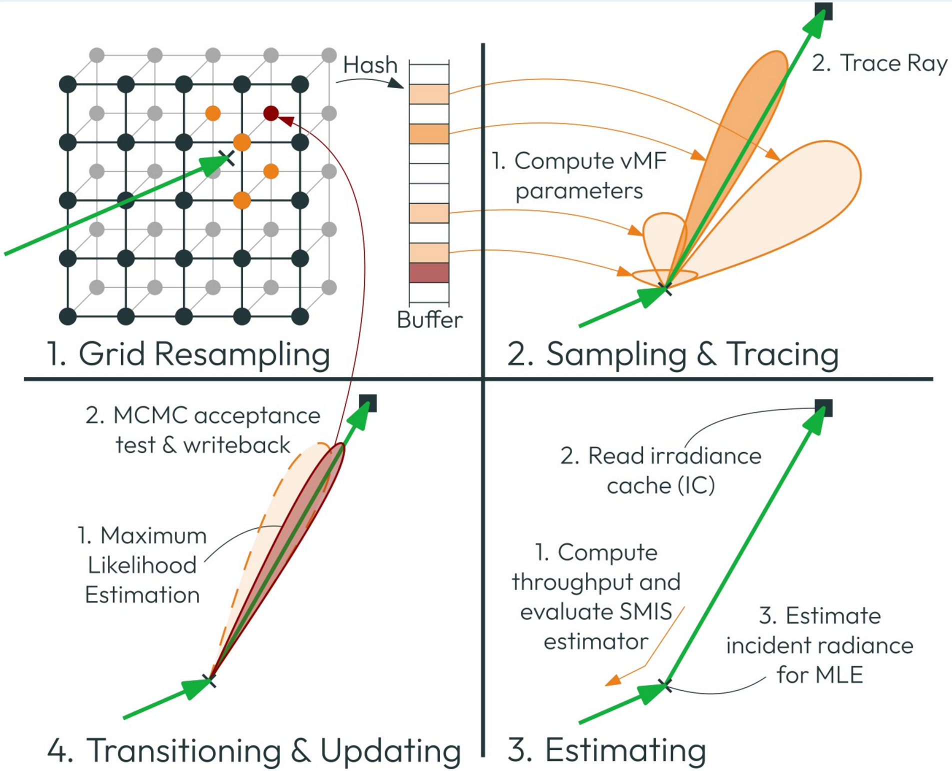


MCPG | Procedure



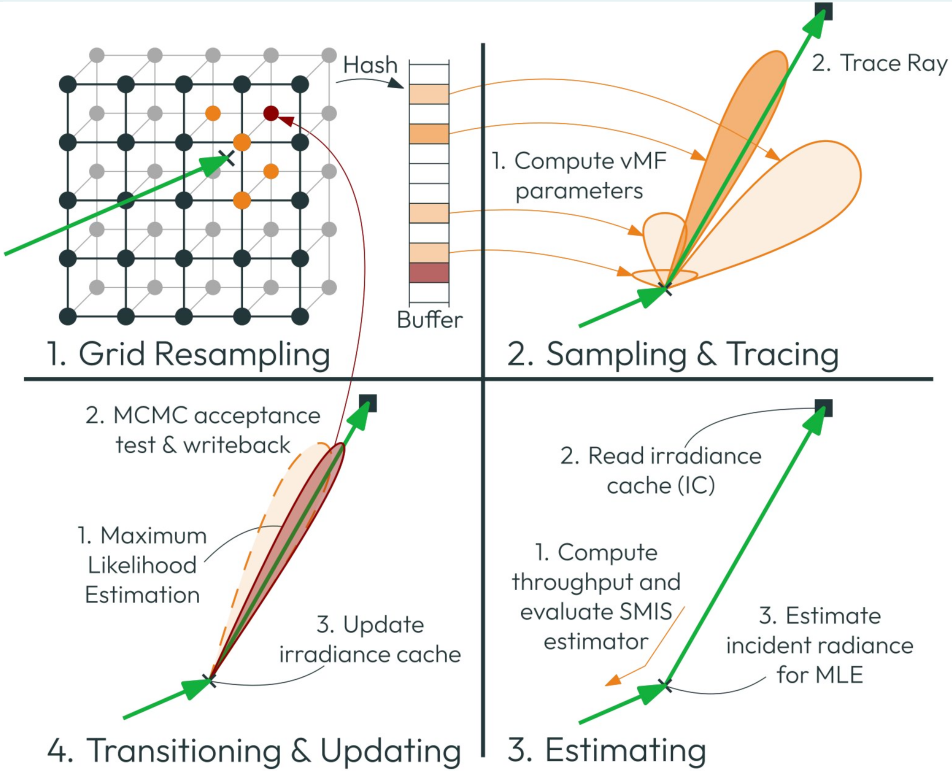
MCPG | Procedure

I3D symposium



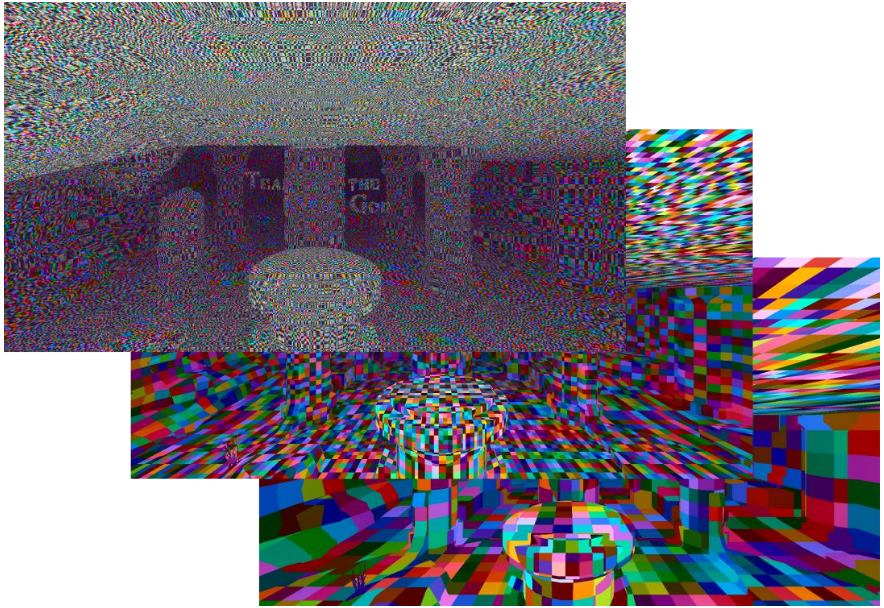
MCPG | Procedure

I3D symposium



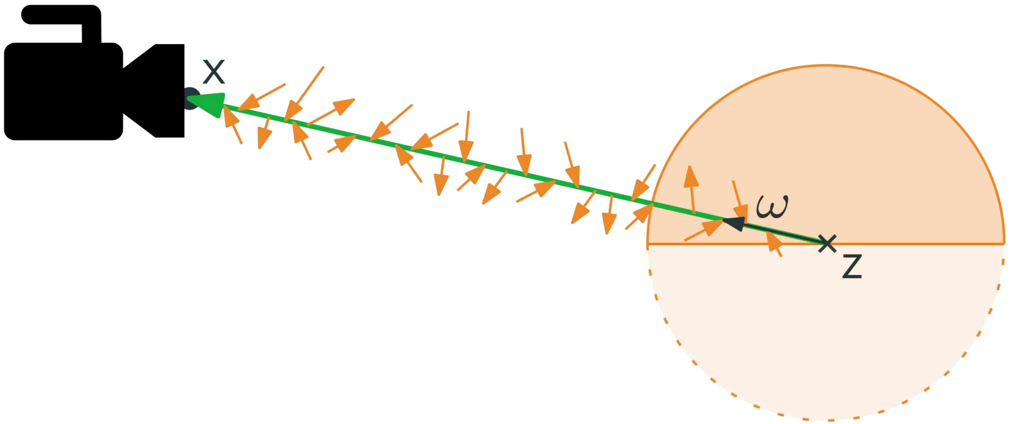
MCPG | Hash Grid

- Multi-resolution hash grids
- Efficient sharing
- Stochastically select level based on camera-distance



MCPG | Single-Scattering

I3D symposium

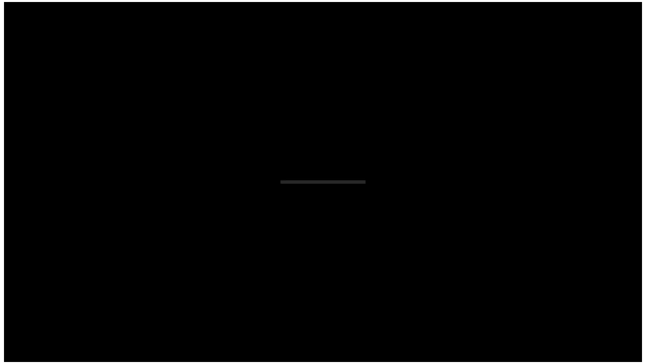


Changes to the guiding procedure:

- Generate points on the camera ray (e.g. transmittance sampling)
- Use phase function instead of BSDF for points in the volume

MCPG | Dynamic Content

- Store light source velocity in state
- Read grid at the world-space position of the previous frame
- Heuristic for invalidation of missing light sources



Results

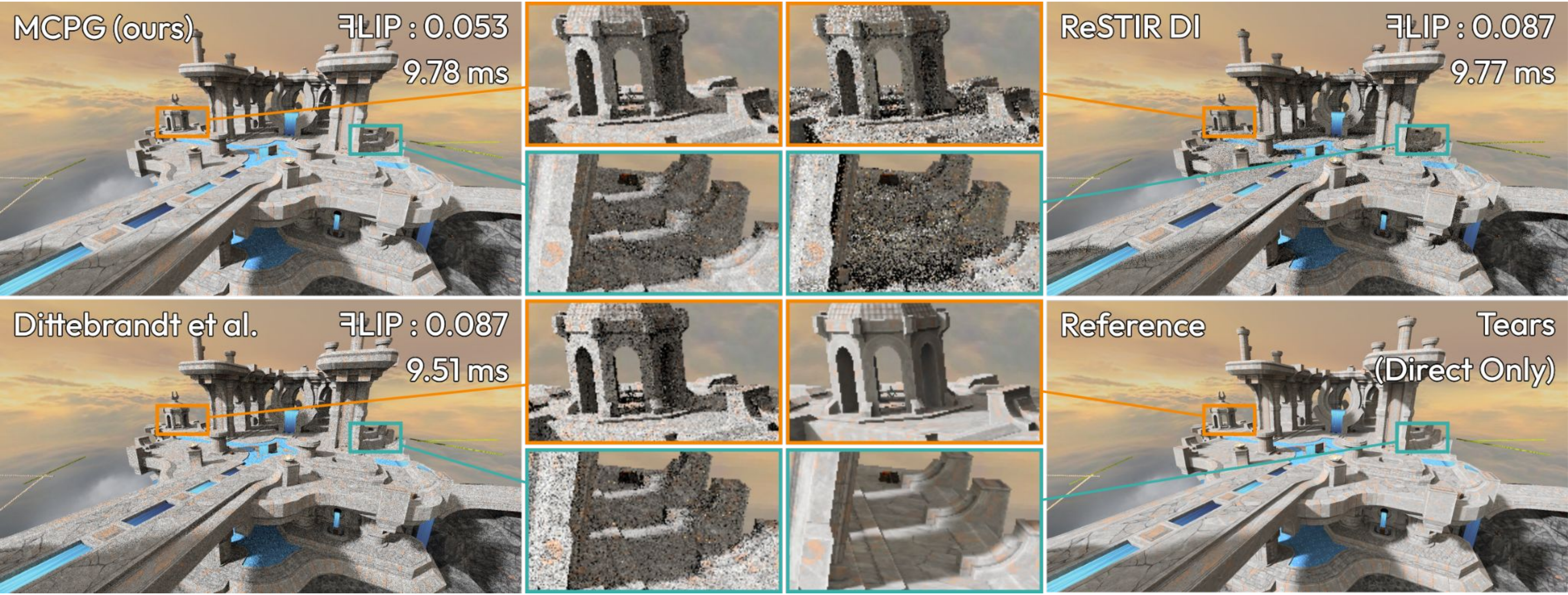
*Results were generated on an AMD Radeon™ RX 7900 XTX

Results | Demo

I3D symposium

Results | World-Space vs Screen-Space (Direct Light)

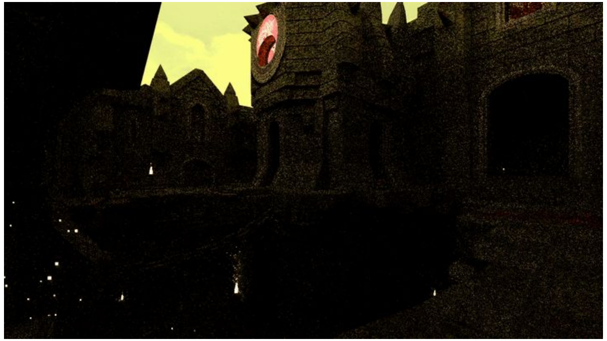
I3D symposium



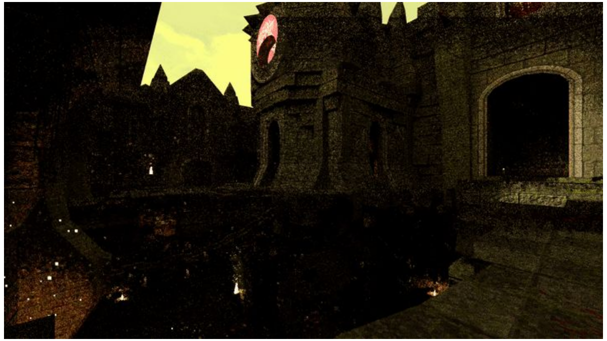
Results | Convergence

I3D symposium

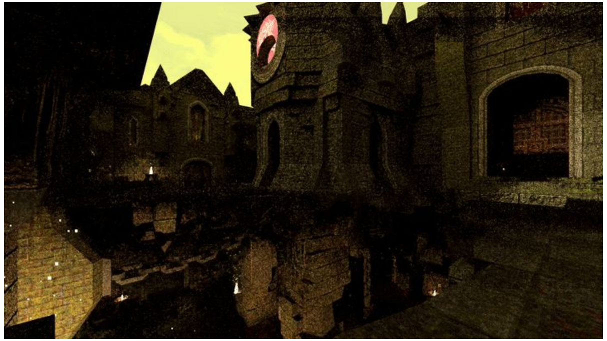
Dittebrandt et al., 2023



MCPG (ours)

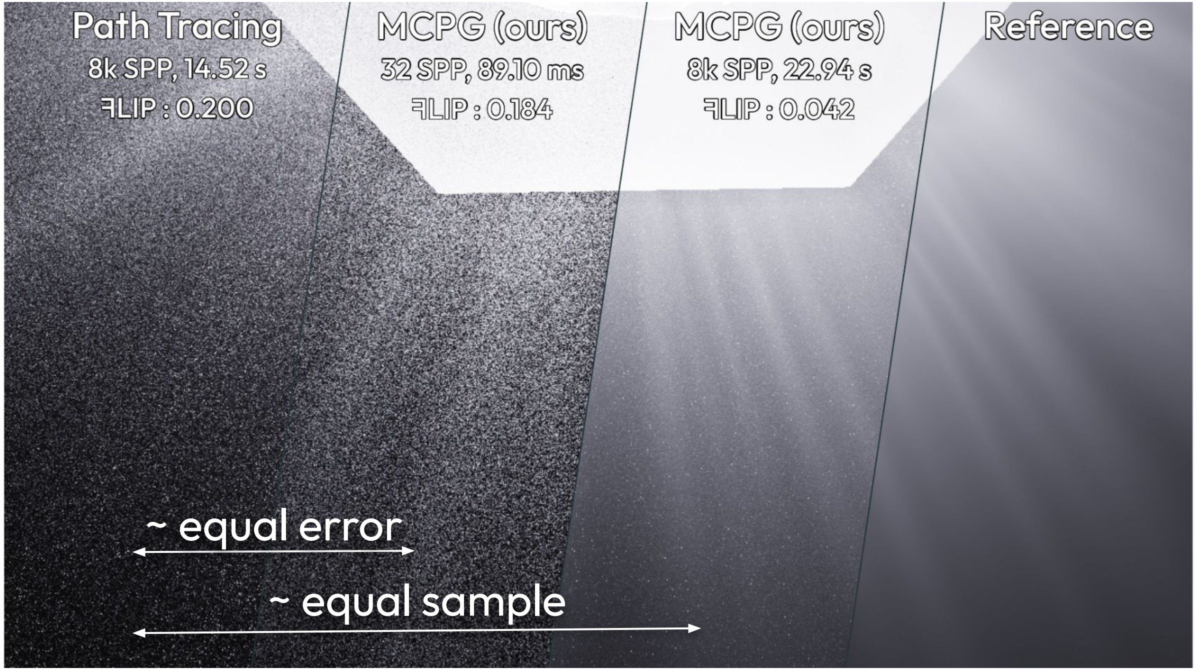


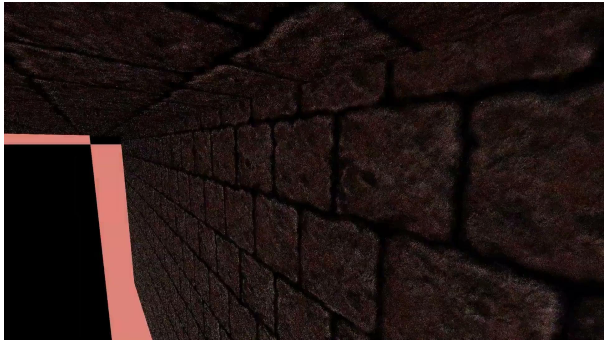
MCPG (ours, 2 bounces)



Results | Single-Scattering

I3D symposium

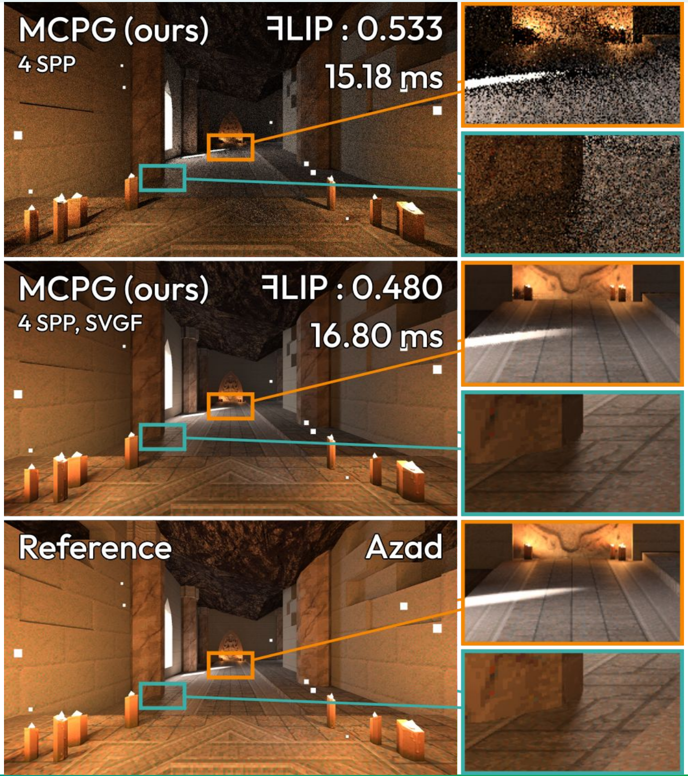




Results | Limitations

I3D symposium

- Limited SMIS size: increased variance in scenes with many light sources
- Proximity bias



Conclusion

I3D symposium

- Lightweight and unbiased path guiding for GPUs
- Tailored for highly dynamic content
- Guides direct and indirect illumination as well as single-scattering events

Contact: lucas.alber@kit.edu

Demo Project: Quake Path Tracer

www.lalber.org/markov-chain-path-guiding

Bibliography (1/2)

I3D symposium

Banerjee, A., Dhillon, I. S., Ghosh, J., and Sra, S. (2005).
Clustering on the unit hypersphere using von mises-fisher distributions.
Journal of Machine Learning Research, 6(46):1345–1382.

Dittebrandt, A., Schüß ler, V., Hanika, J., Herholz, S., and Dachsbacher, C. (2023).
Markov Chain Mixture Models for Real-Time Direct Illumination.
Computer Graphics Forum.

Kajiya, J. T. (1986).
The rendering equation.
In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '86, pages 143–150, New York, NY, USA. ACM.

Bibliography (2/2)

I3D symposium

Novák, J., Georgiev, I., Hanika, J., and Jarosz, W. (2018).
Monte Carlo methods for volumetric light transport simulation.
Computer Graphics Forum (Eurographics State of the Art Reports), 37(2):1–26.

Ruppert, L., Herholz, S., and Lensch, H. P. A. (2020).
Robust fitting of parallax-aware mixtures for path guiding.
ACM Trans. Graph., 39(4).

West, R., Georgiev, I., Gruson, A., and Hachisuka, T. (2020).
Continuous multiple importance sampling.
ACM Trans. Graph., 39(4).

Backup

Backup | Hash Grid Data

I3D symposium

Table 1: Overview of data that is stored for every Markov chain state at hash grid vertices.

Symbol	Meaning
$\overline{W}$	luminance estimate of eq. (1), used as resampling weight
N	number of samples
$\bar{y}$	luminance-weighted mean of light source positions
$\bar{r}$	luminance-weighted mean cosine of light source directions
T	time of last update
$\mathbf{v}$	last known velocity of the light source
hash	hashed grid position for collision detection

Backup | Grid Resampling

I3D symposium

Algorithm 1: Grid Resampling

S $\leftarrow$ array of N_{mc} Markov chain states
$s \leftarrow \{0\}$ `/* empty Markov chain state */`
$sum_{mc} \leftarrow 0$
foreach $i \leftarrow 0$ **to** N_{mc} - 1 **do**
    S[i] $\leftarrow$ grid.`load`(hit.x, hit.n)
    if hash grid collision **then**
        S[i].$\overline{w} = 0$
    $sum_{mc} \leftarrow sum_{mc} +$ S[i].$\overline{w}$
    $\xi \leftarrow$ uniform random in $[0, 1)$
    if $\xi < \frac{S[i].\overline{w}}{sum_{mc}}$ **then**
        $s \leftarrow$ S[i]

Backup | Sampling and Ray Casting

Algorithm 2: Sampling and Ray Casting

$\xi \leftarrow$ uniform random in $[0, 1)$

if $s.\overline{w} = 0$ `/* invalid */` or $\xi < p_{\text{bsdf}}$ **then**

  $p(\omega) \leftarrow f_s(\omega_i, \text{hit}.n)$ or $|\text{hit}.n \cdot \omega|$

  $s \leftarrow \{0\}$ `/* empty Markov chain state */`

else

  $\boldsymbol{\mu} \leftarrow \texttt{normalize}(s.\overline{y}/s.\overline{w} - \text{hit}.x)$

  $r \leftarrow (s.N^2 \cdot s.\overline{r}/s.\overline{w} + N_p \cdot r_p) \,/\, (s.N^2 + N_p)$

  $\kappa \leftarrow (3r - r^3)/(1 - r^2)$ `/* [Banerjee et al., 2005, eq. 4.4] */`

  $p(\omega) \leftarrow p(\omega \mid \boldsymbol{\mu}, \kappa)$ `/* ?? */`

sample $\omega \sim p(\omega)$

$\text{hit}_{\text{next}} \leftarrow$ trace ray $(\text{hit}.x, \omega)$

Backup | Update

Algorithm 3: State Transition and Maximum Likelihood Estimation

$f_{mc} \leftarrow$ `lum(`$\langle L_i(\text{hit}.x, \omega)\rangle_{SMIS}$`)`

$\xi \leftarrow$ uniform random in $[0, 1)$

if $\xi < f_{mc}/(sum_{mc}/N_{mc})$ `/* MCMC acceptance test */` **then**

  $s.N \leftarrow \min(s.N + 1, N_{max})$

  $\alpha \leftarrow \max(1/s.N, \alpha_{min})$ `/* blend factor, cf. [Dittebrandt et al., 2023] */`

  $\boldsymbol{\mu} \leftarrow$ `normalize(`$s.\overline{y}/s.\overline{w} - \text{hit}.x$`)`

  $s.\overline{w} \leftarrow$ `mix(`$s.\overline{w}, f_{mc}, \alpha$`)`

  $s.\overline{y} \leftarrow$ `mix(`$s.\overline{y}, f_{mc} \cdot \text{hit}_{next}.x, \alpha$`)`

  $s.\overline{r} \leftarrow$ `mix(`$s.\overline{r}, f_{mc} \cdot$ `dot(normalize(`$\text{hit}_{next}.x - \text{hit}.x$`)`$, \boldsymbol{\mu}$`)`$, \alpha$`)`

  $s.T \leftarrow T$ `/* current time */`

  $s.v \leftarrow (\text{hit}_{next}.x - \text{hit}_{next}.x_{prev})/(T - T_{prev})$

  grid.`store(`s`)`

Backup | MCMC Acceptance Probability

I3D symposium

- Use estimate for increased exploration of outlier samples
- Use mean score for learning of the mixture across states

$$p_{\text{accept}} = \min\left(\frac{f_{\text{mc}}}{(sum_{\text{mc}}/N_{\text{mc}})}, 1\right)$$

f_{mc} : Luminance estimate of the incident light $\langle L_{\text{i}}(x, \omega)\rangle_{\text{SMIS}}$

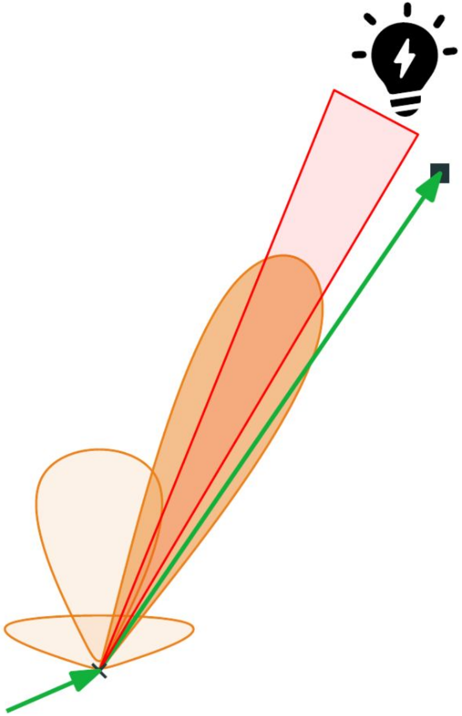
sum_{mc} : Sum of resampling weights

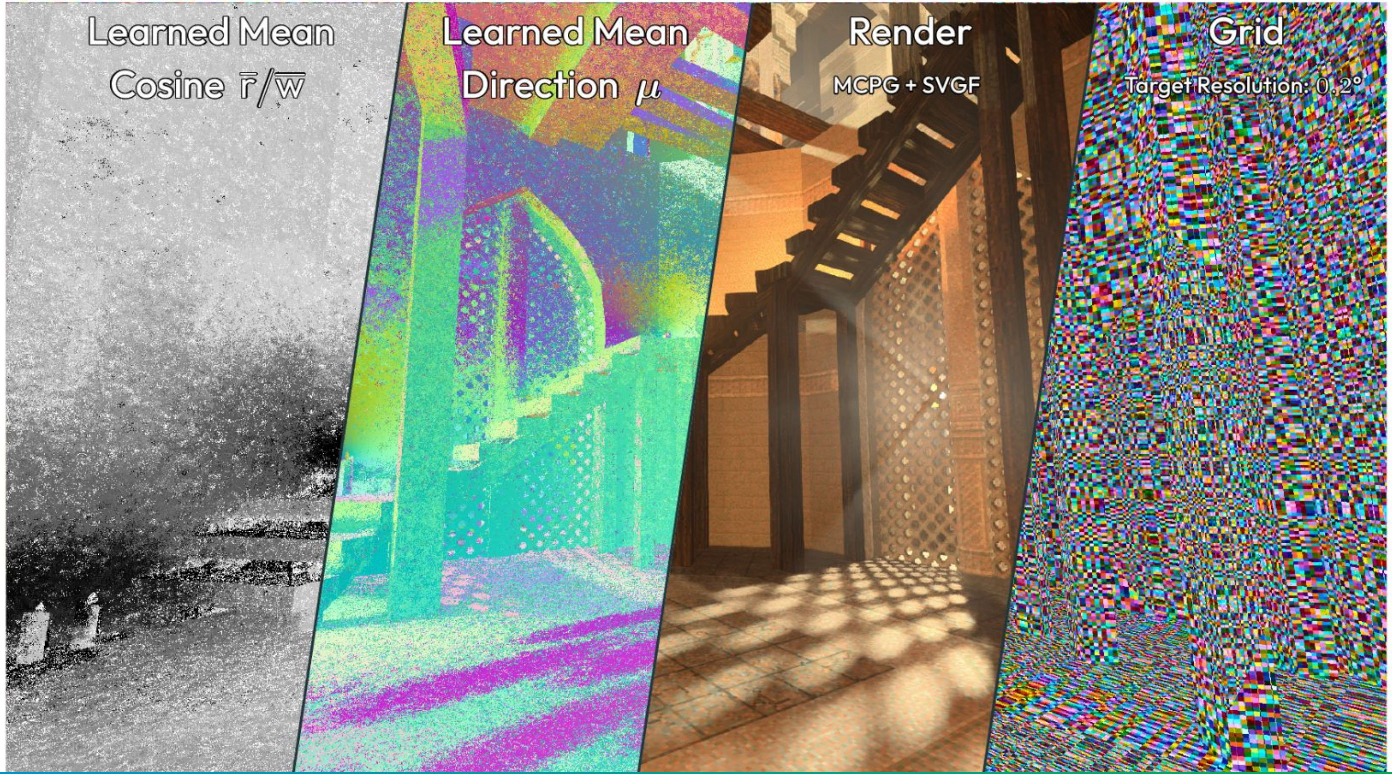
N_{mc} : Number of resampled states

Backup | Heuristic

I3D symposium

- Based on mean direction and cosine, we define a trust-region
- States are invalidated, if samples inside the trust-region return less-than-expected radiance




Learned Mean
Cosine $\bar{r}/\bar{w}$
Learned Mean
Direction μ
Render
MCPG + SVGF
Grid
Target Resolution: 0.2°