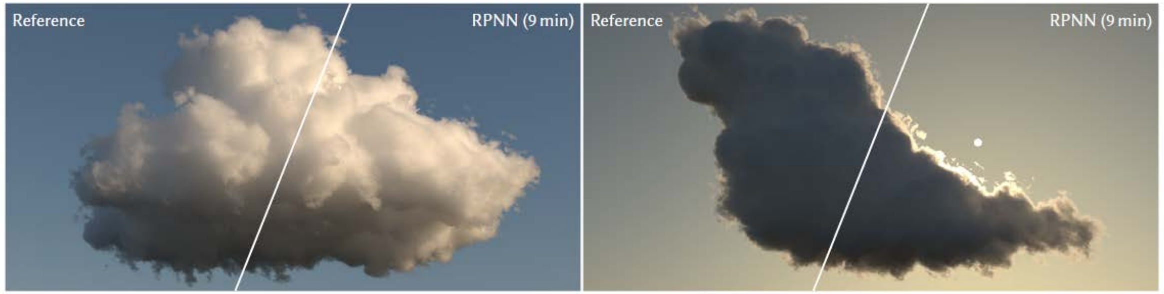
Learning-Based Rendering

2019.05.16.

20193138 Jaeyoon Kim

Deep Scattering: Rendering Atmospheric Clouds with Radiance-Predicting Neural Networks

SIMON KALLWEIT, Disney Research and ETH Zürich
THOMAS MÜLLER, Disney Research and ETH Zürich
BRIAN MCWILLIAMS, Disney Research
MARKUS GROSS, Disney Research and ETH Zürich
JAN NOVÁK, Disney Research



Bayesian online regression for adaptive direct illumination sampling

PETR VÉVODA*, Charles University, Prague and Render Legion, a. s.
IVO KONDAPANENI*, Charles University, Prague
JAROSLAV KŘIVÁNEK, Charles University, Prague and Render Legion, a. s.

No learning
RMSE 0.0031 (0.0014 for DI only)

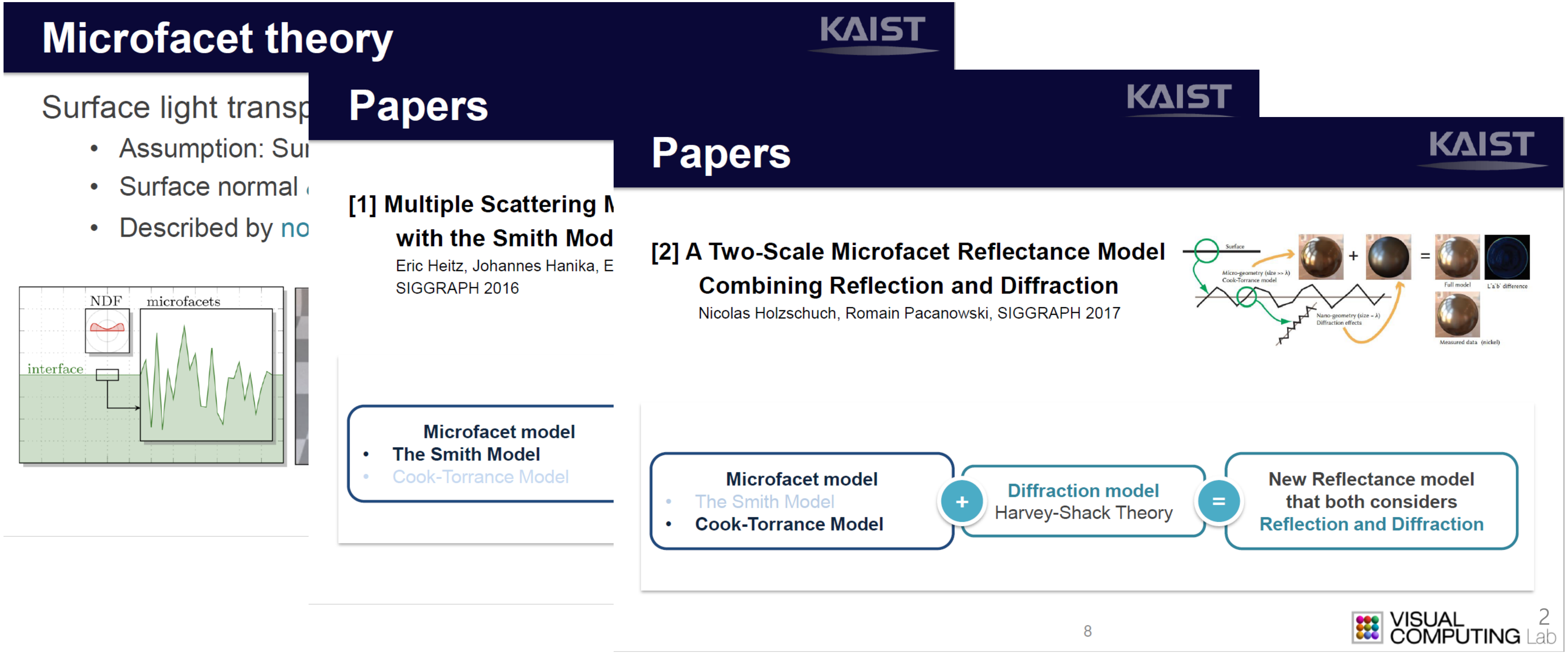
Our method
RMSE 0.0012, *6.7x speedup* (0.000062, *510x* for DI only)

No learning

Our method

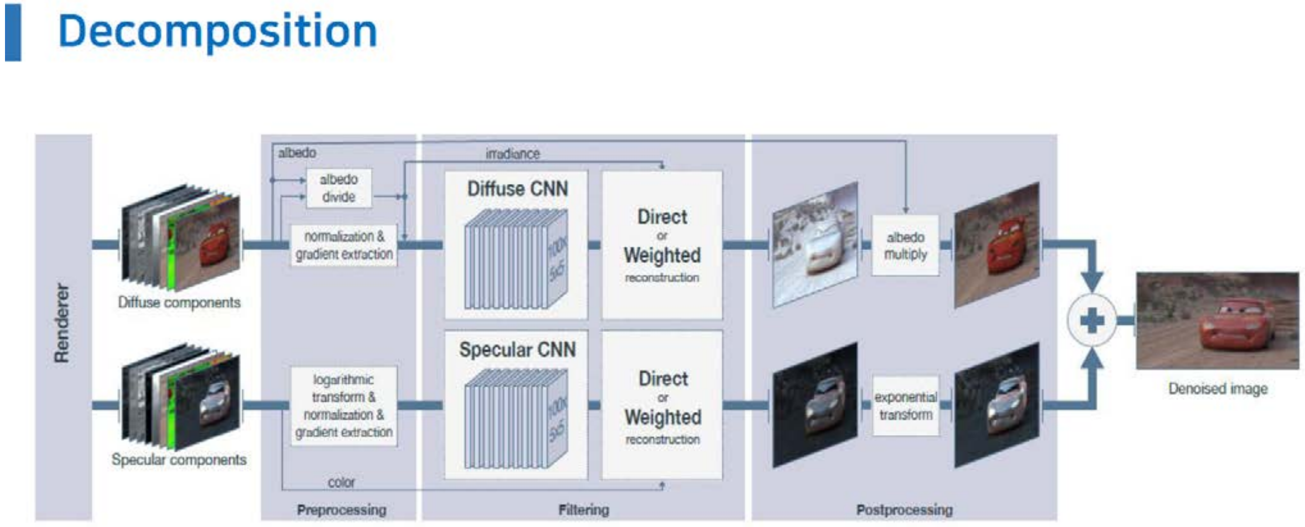
Review for previous presentation

- “Microfacet Model and Material Appearance” by Hakyeong Kim

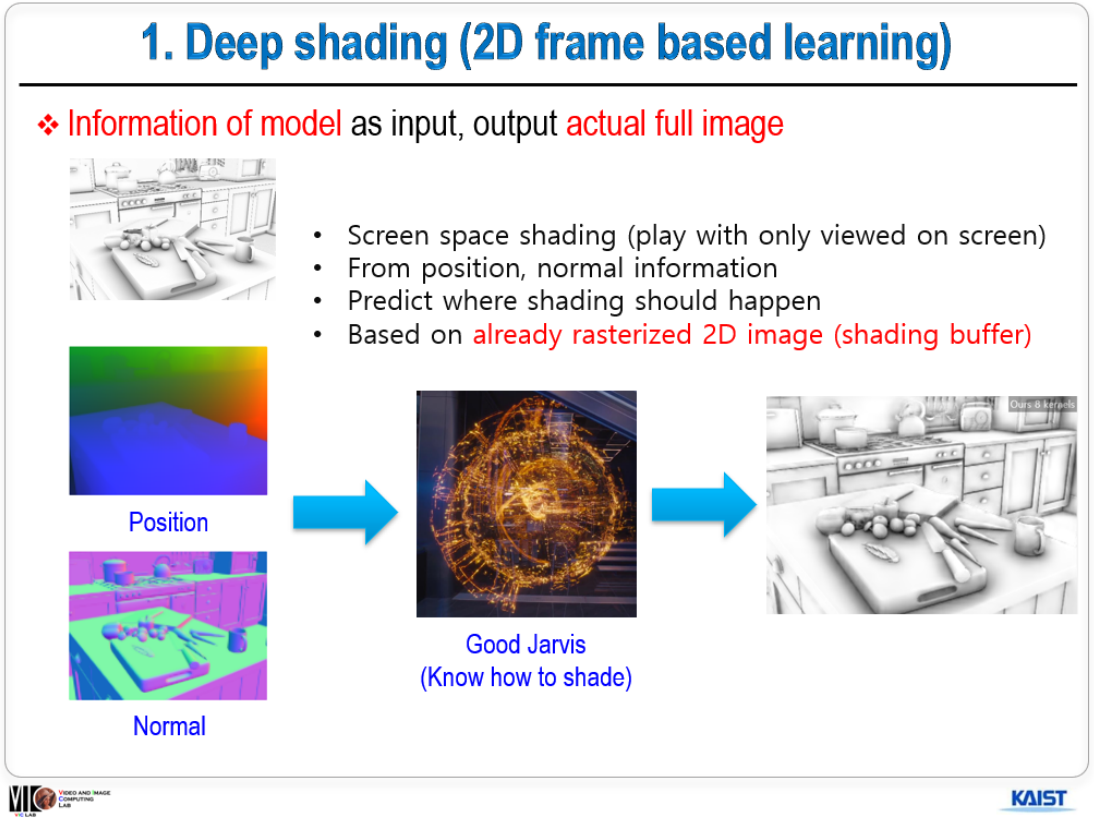


VISUAL COMPUTING Lab

Machine Learning in Rendering



From CheolMin's slides



From Saehun's slides

Deep Scattering:
Rendering Atmospheric Clouds with Radiance-Predicting Neural Networks

by S. Kallweit, T. Müller, B. McWilliams, M. Gross, J. Novák

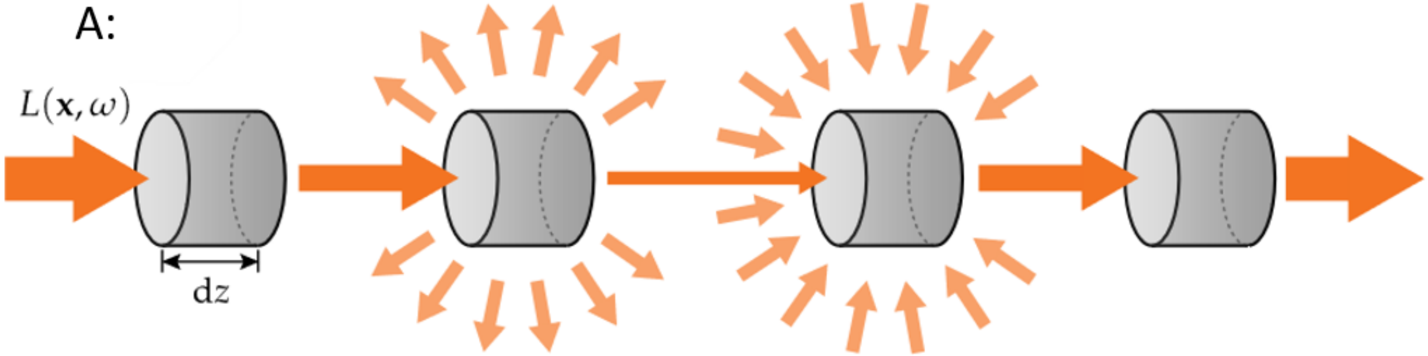
Slides and images from “Deep Learning for Light Transport” by Jan Novák, Disney Research

Remind previous presentation

Volume scattering process

Q: How do lights interact with volumes?

A:



(a) *Absorption* (b) *Out-scattering* (c) *In-scattering* (d) *Emission*

$$\frac{\mathrm{d}L}{\mathrm{d}z} = -\mu_a(\mathbf{x})L(\mathbf{x}, \omega) \qquad \frac{\mathrm{d}L}{\mathrm{d}z} = -\mu_s(\mathbf{x})L(\mathbf{x}, \omega) \qquad \frac{\mathrm{d}L}{\mathrm{d}z} = \mu_s(\mathbf{x})L_s(\mathbf{x}, \omega) \qquad \frac{\mathrm{d}L}{\mathrm{d}z} = \mu_a(\mathbf{x})L_e(\mathbf{x}, \omega)$$

μ_a - **absorption coefficient** μ_s - **scattering coefficient** L_s - **in-scattered radiance** L_e - **emitted radiance**

$\boxed{-\mu_t(\mathbf{x})L(\mathbf{x}, \omega) \quad \textbf{Losses}}$

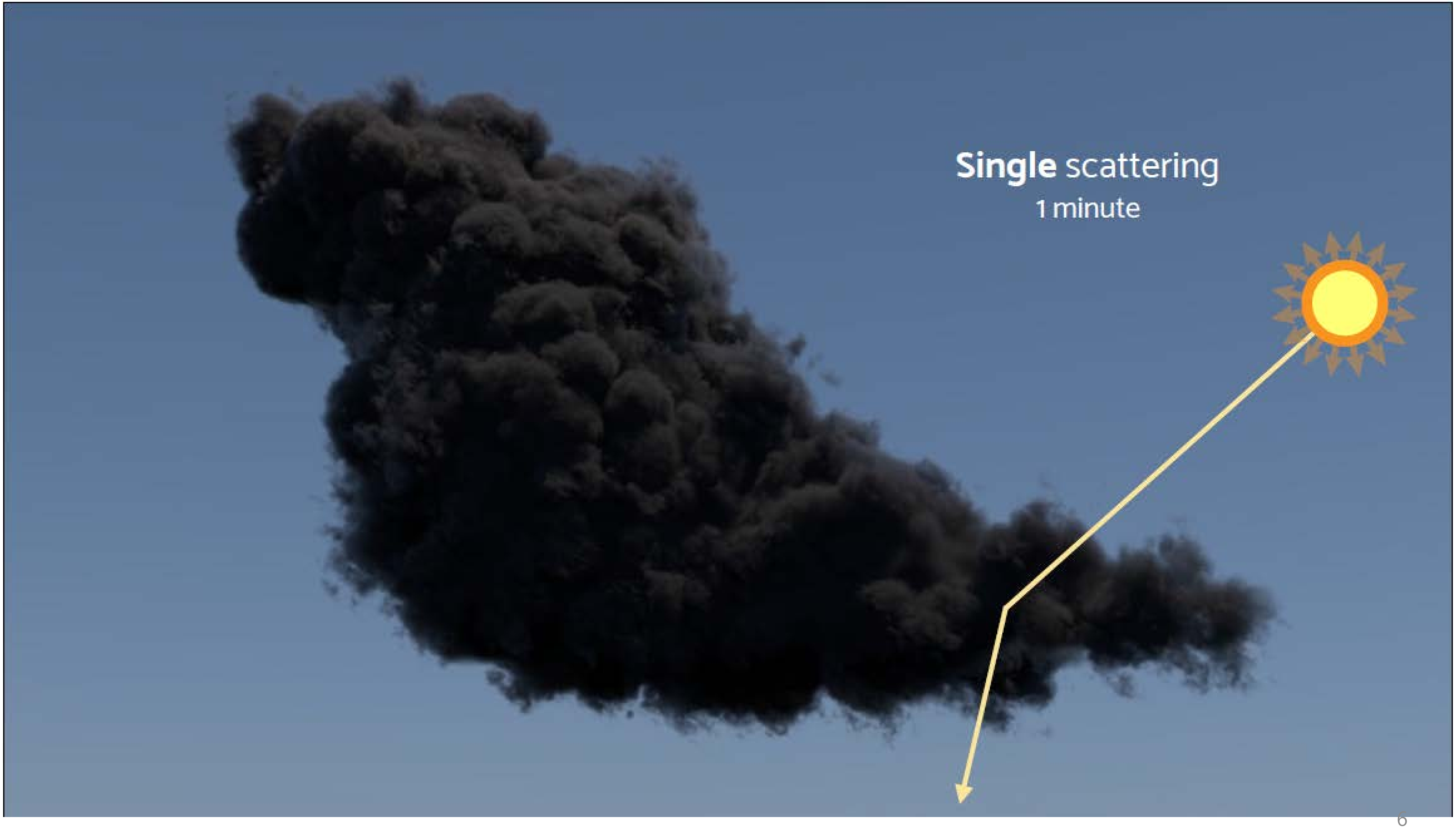
Extinction coefficient

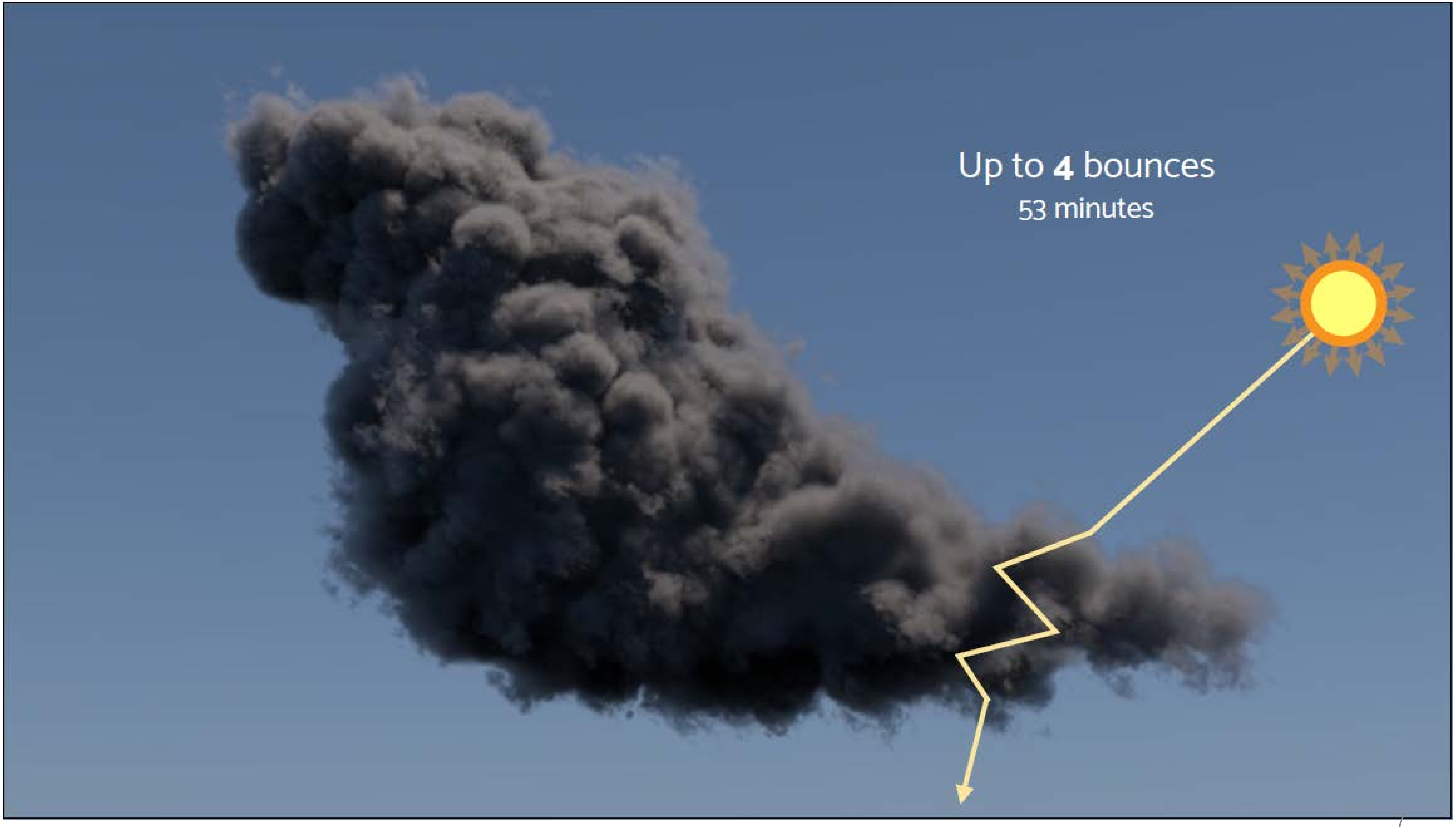
$\mu_t(\mathbf{x}) = \mu_a(\mathbf{x}) + \mu_s(\mathbf{x})$

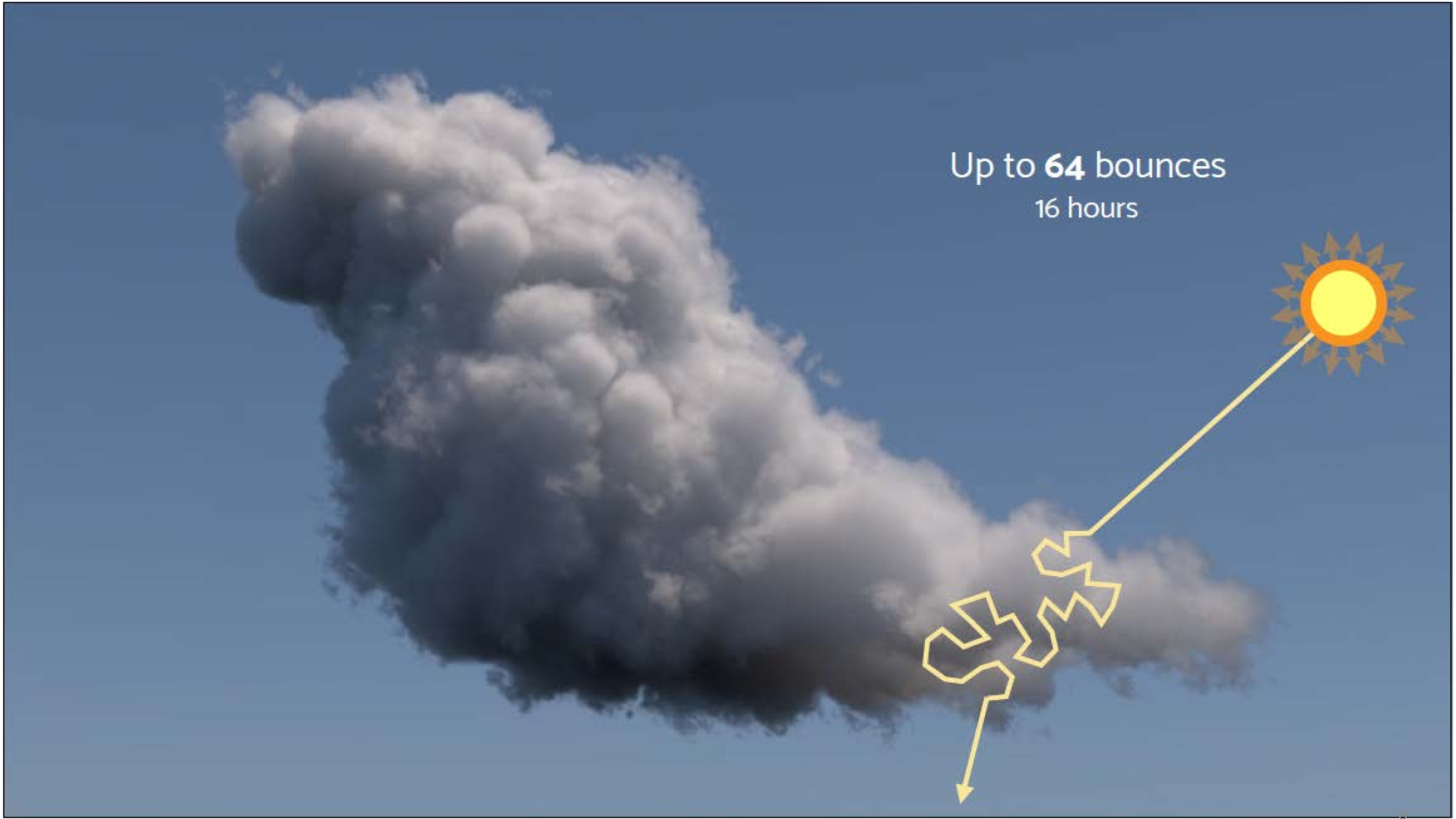
$$L_s(\mathbf{y}, \omega) = \int_{S^2} f_p(\omega, \bar{\omega})L(\mathbf{y}, \bar{\omega})\mathrm{d}\bar{\omega}$$

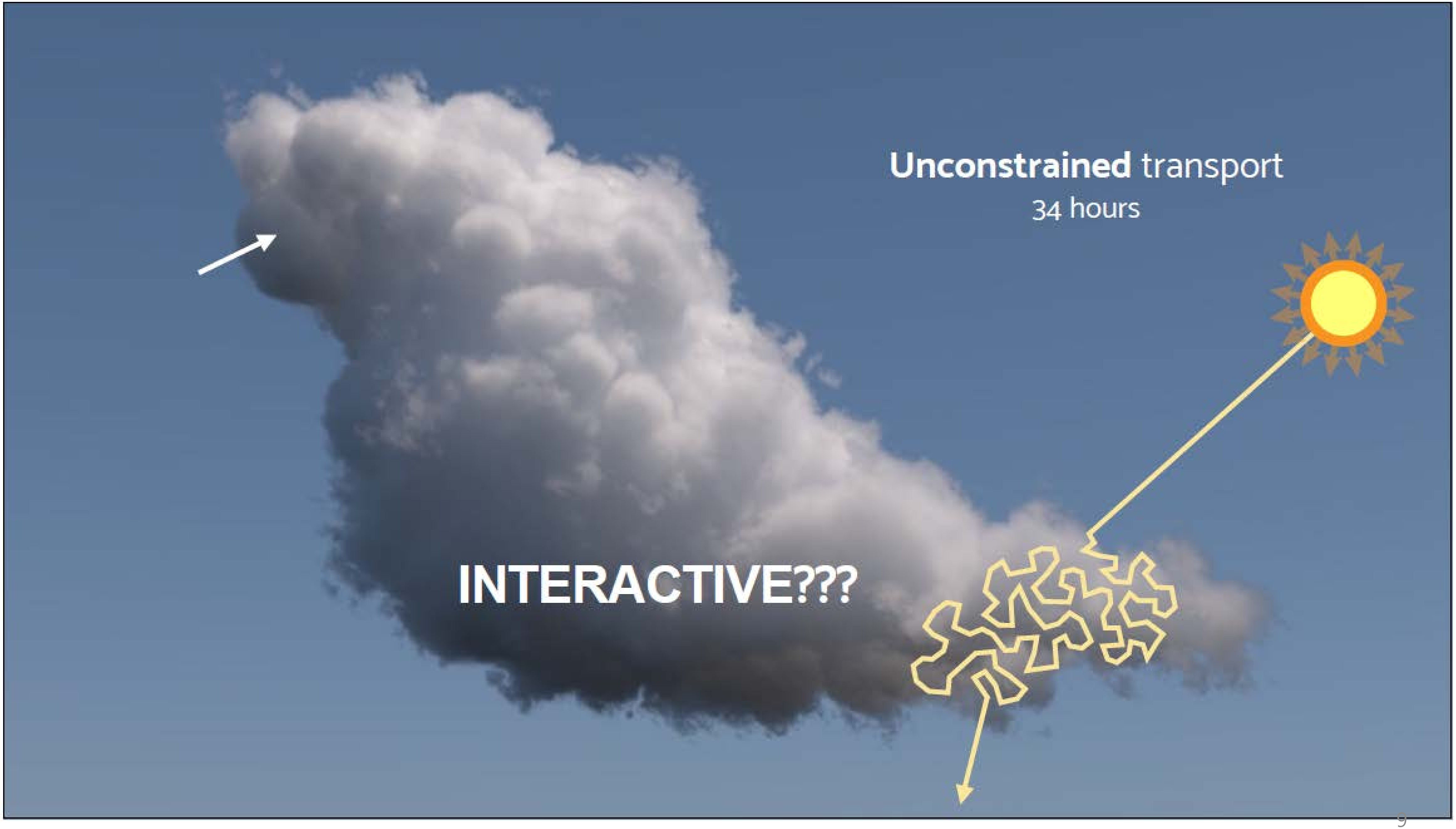
Phase function

From Hyunsoo's slides


Single scattering
1 minute


Up to **4** bounces
53 minutes

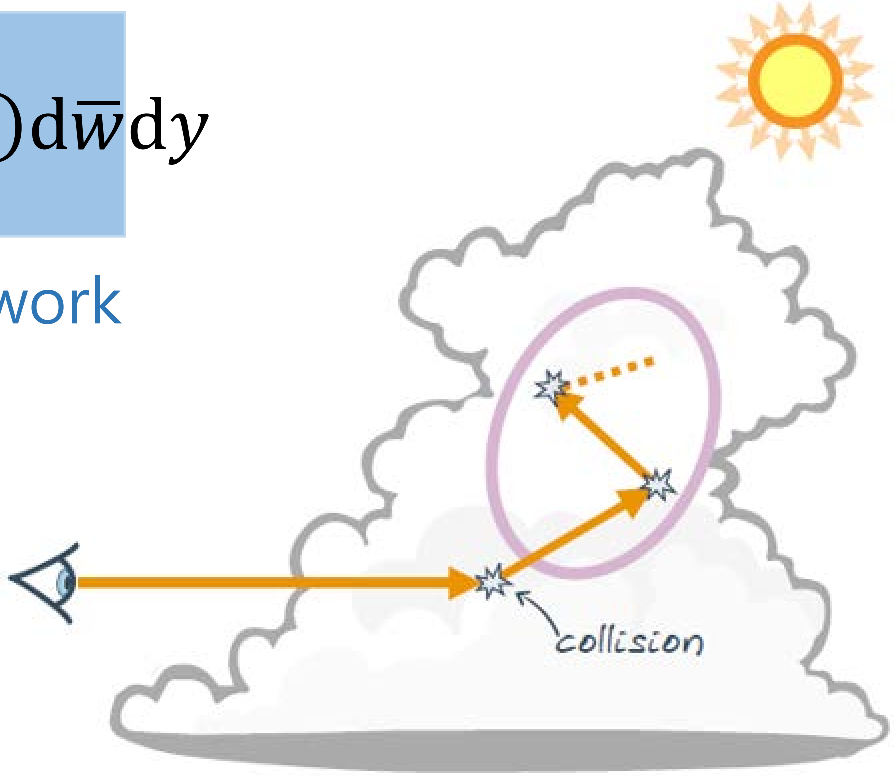

Up to **64** bounces
16 hours


Unconstrained transport
34 hours
INTERACTIVE???

Rendering Clouds

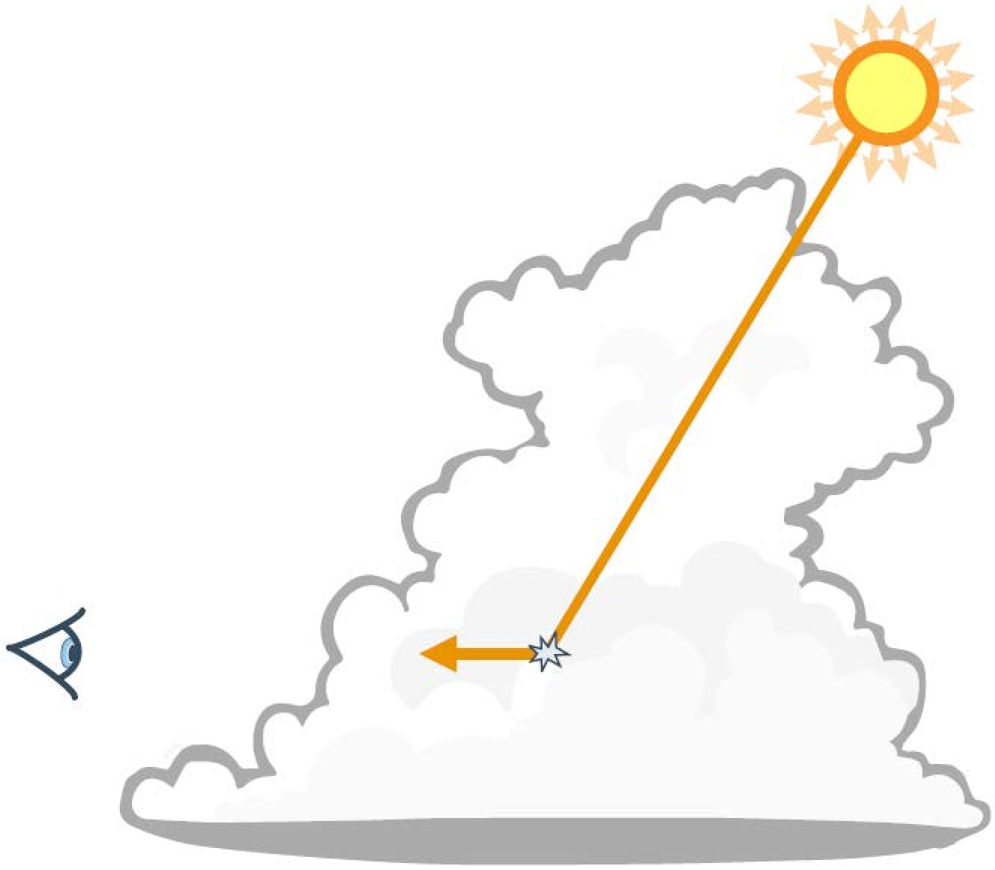
$$L(\mathbf{x}, w) = \int_0^\infty p(\mathbf{y}|\mathbf{x})\, \alpha \boxed{\int_S p(\overline{w}|w)\, L(\mathbf{y}, \overline{w}) \mathrm{d}\overline{w}} \mathrm{d}y$$

Predict using a network



Rendering Clouds

- Scattered direct illumination
  - Estimated using MC



Rendering Clouds

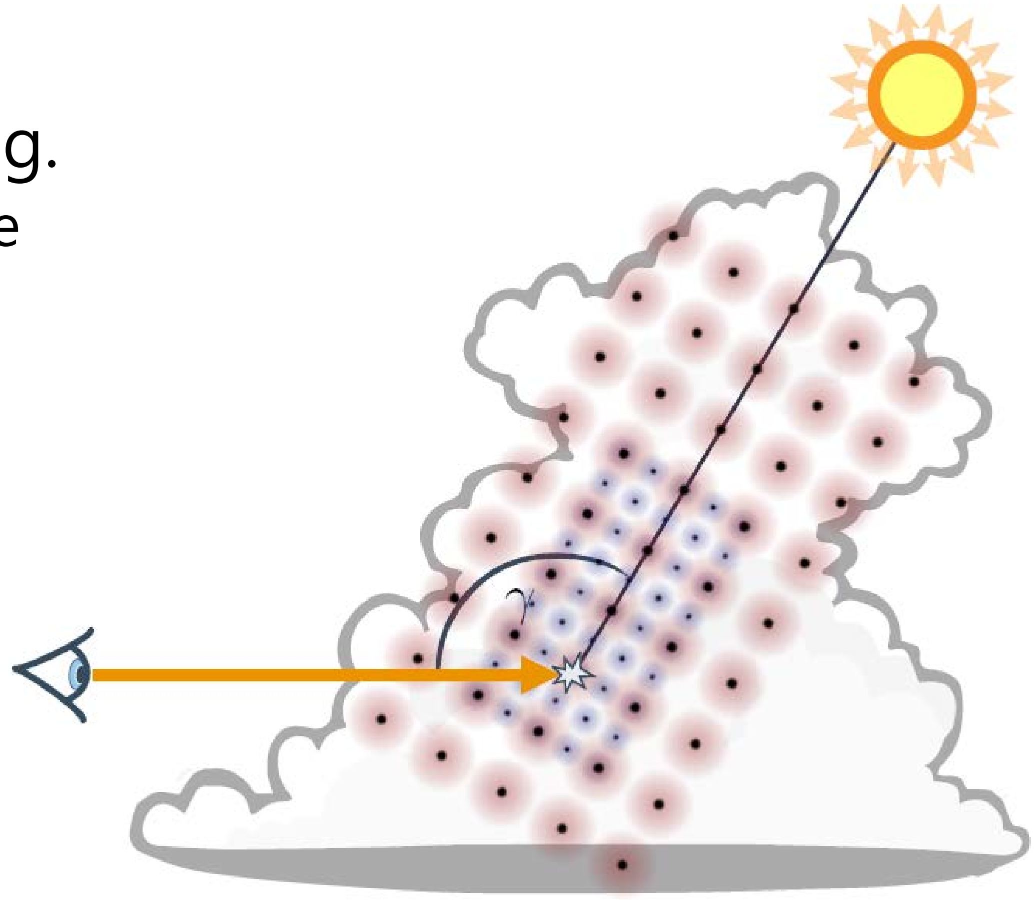
- Scattered direct illumination
  - Estimated using MC

- Scattered indirect illumination
  - Predicted using NN

Radiance Predicting MLP

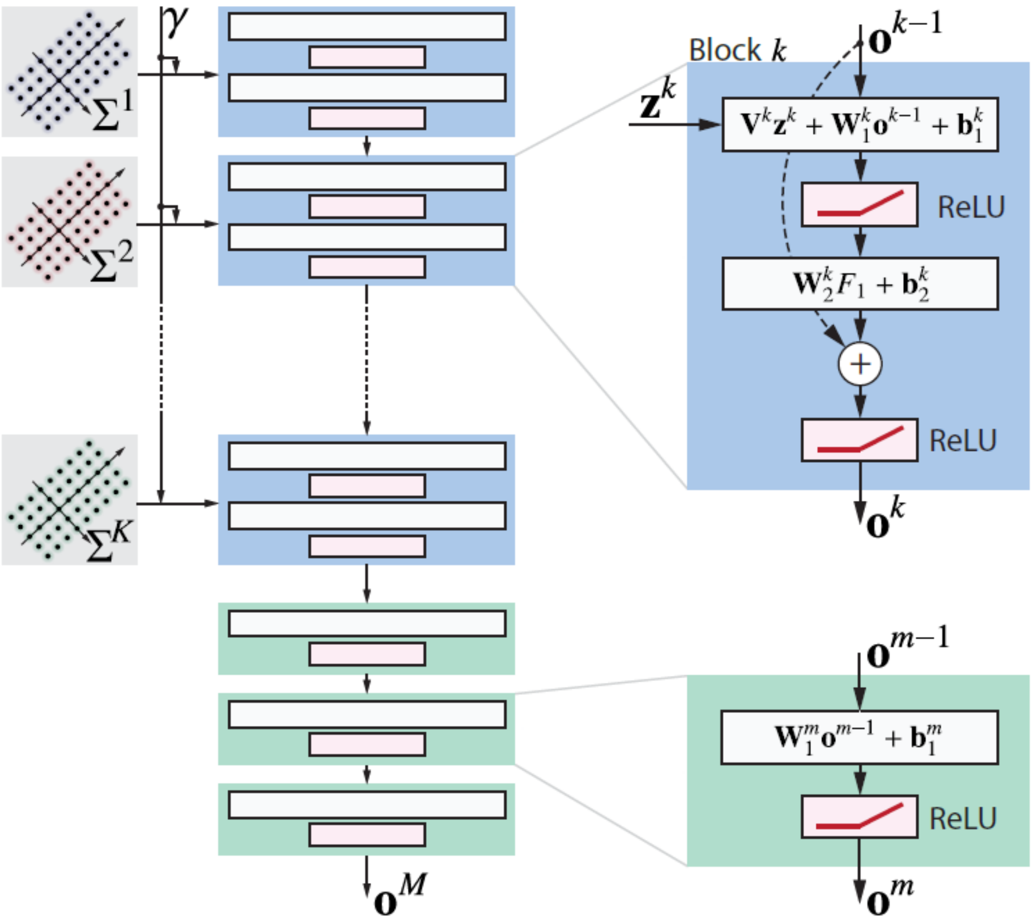
- Uses hierarchical stencil for sampling.
  - for both local details and overall shape

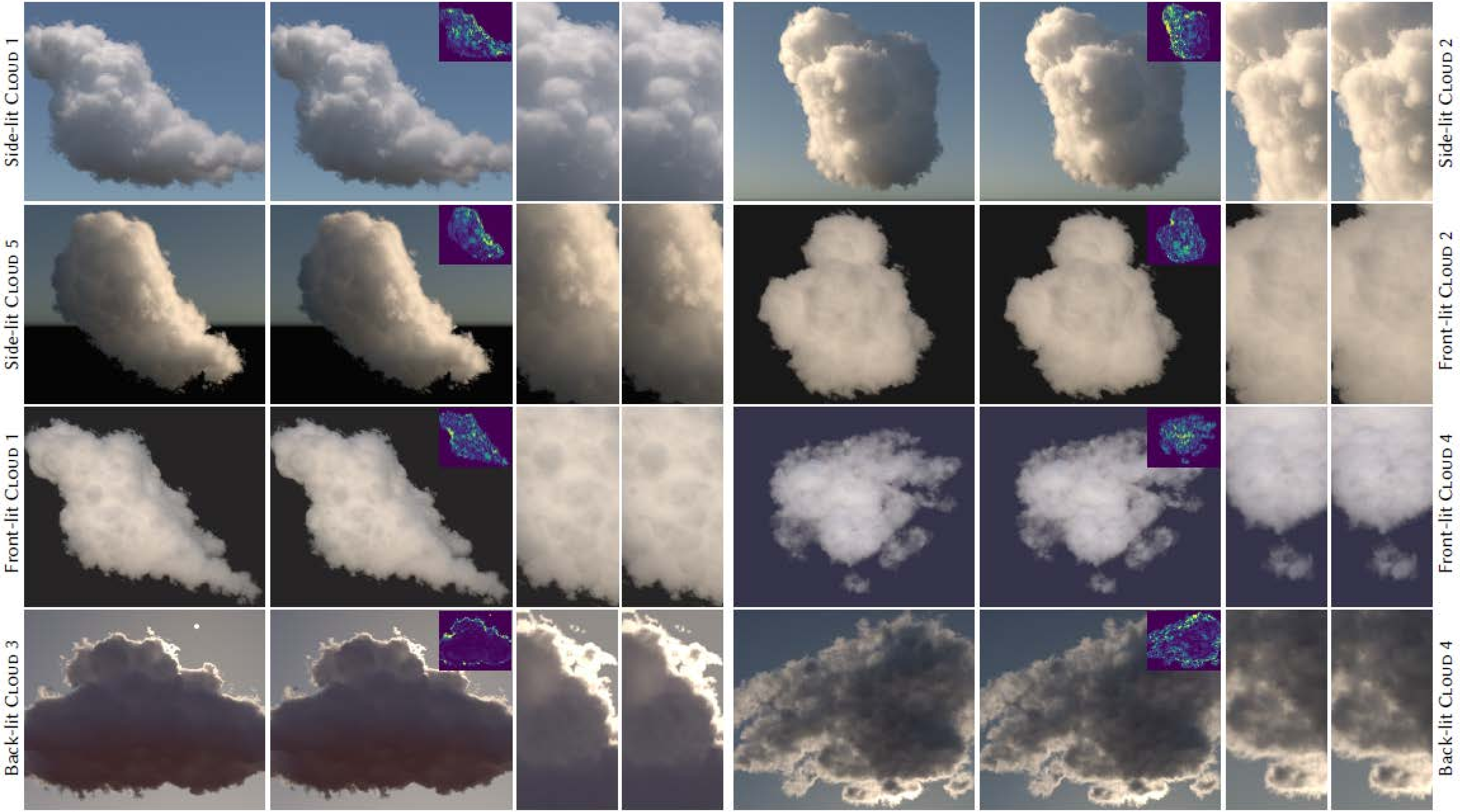
- Feeds into the network.



Radiance Predicting MLP

- Uses hierarchical stencil for sampling.
  - for both local details and overall shape
- Feeds into the network.
  - One block with two layers for each stencil.
  - Some FC layers at the end of the network.





(a) PT reference (b) RPNN (ours) (c) PT (d) RPNN (e) PT reference (f) RPNN (ours) (g) PT (h) RPNN

(a) PT reference (b) RPNN (ours) (c) Flux-limited Diffusion (d) PT reference (e) RPNN (f) FLD

Bayesian online regression for adaptive direct illumination sampling

Petr Vévoda, Ivo Kondapaneni, and Jaroslav Křivánek

Render Legion, a.s.
Charles University, Prague

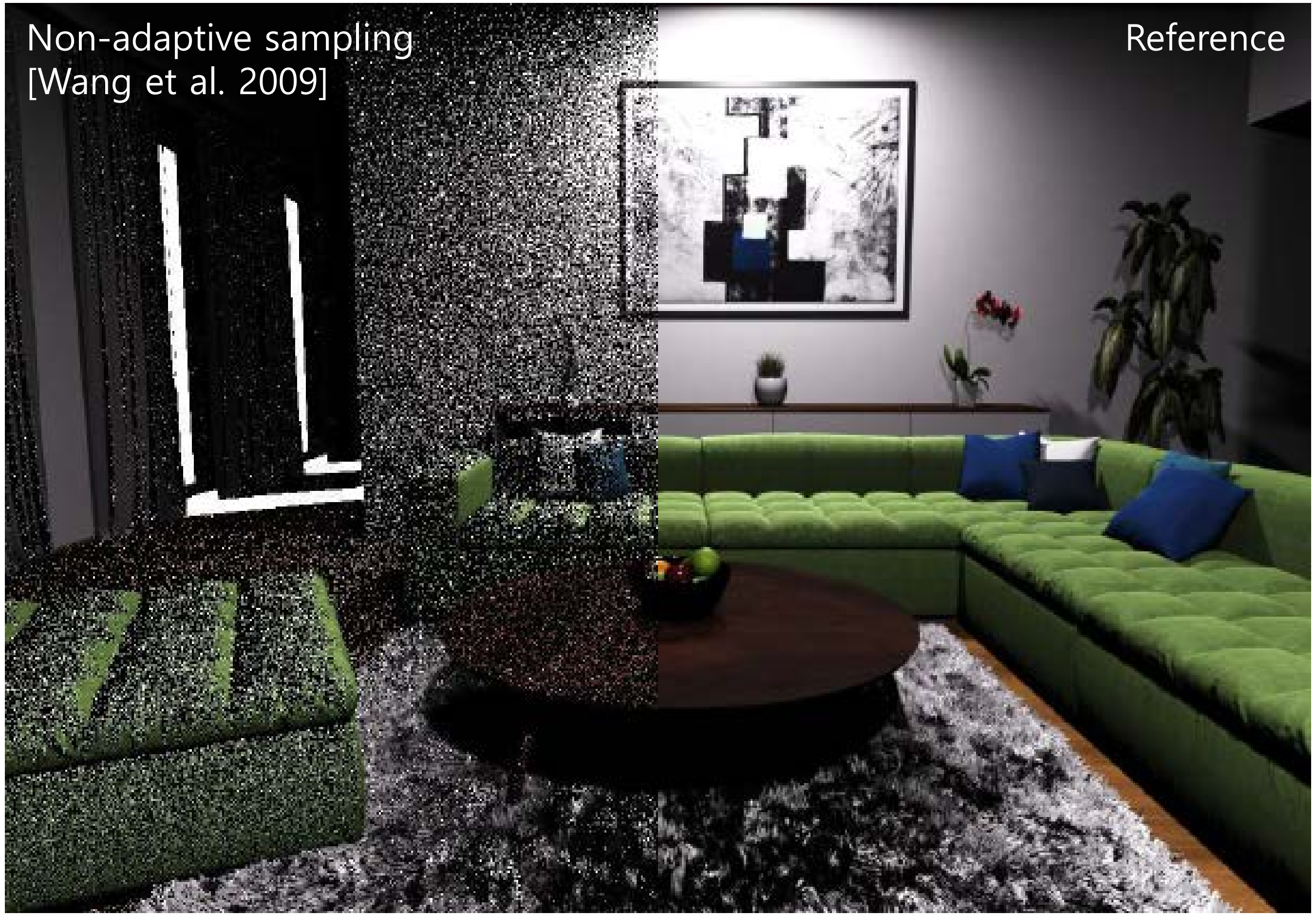
corona

Slides and images from “Bayesian online regression for adaptive direct illumination sampling” by Petr at al.

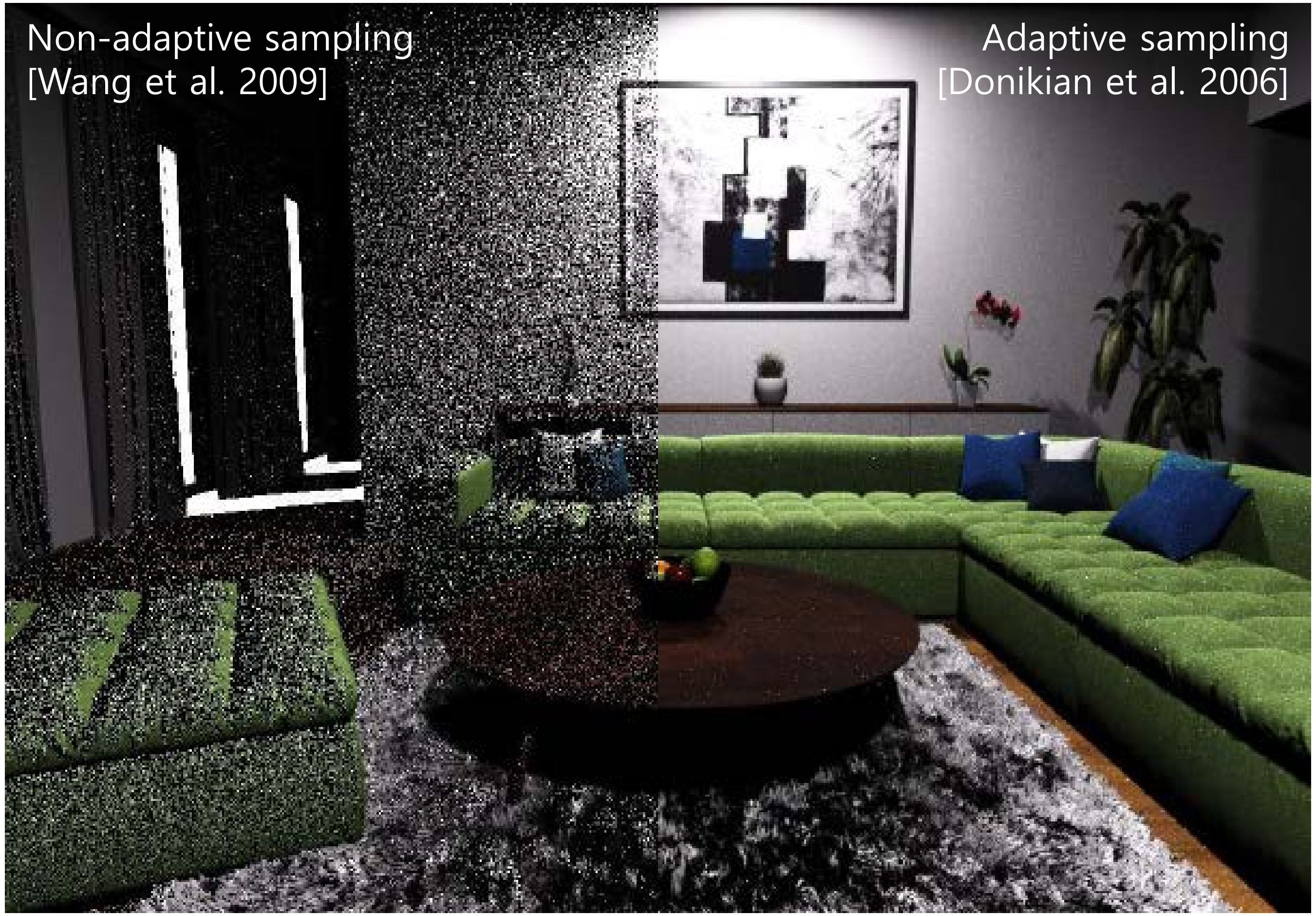
Computer Graphics Charles University

Noise in MC Rendering

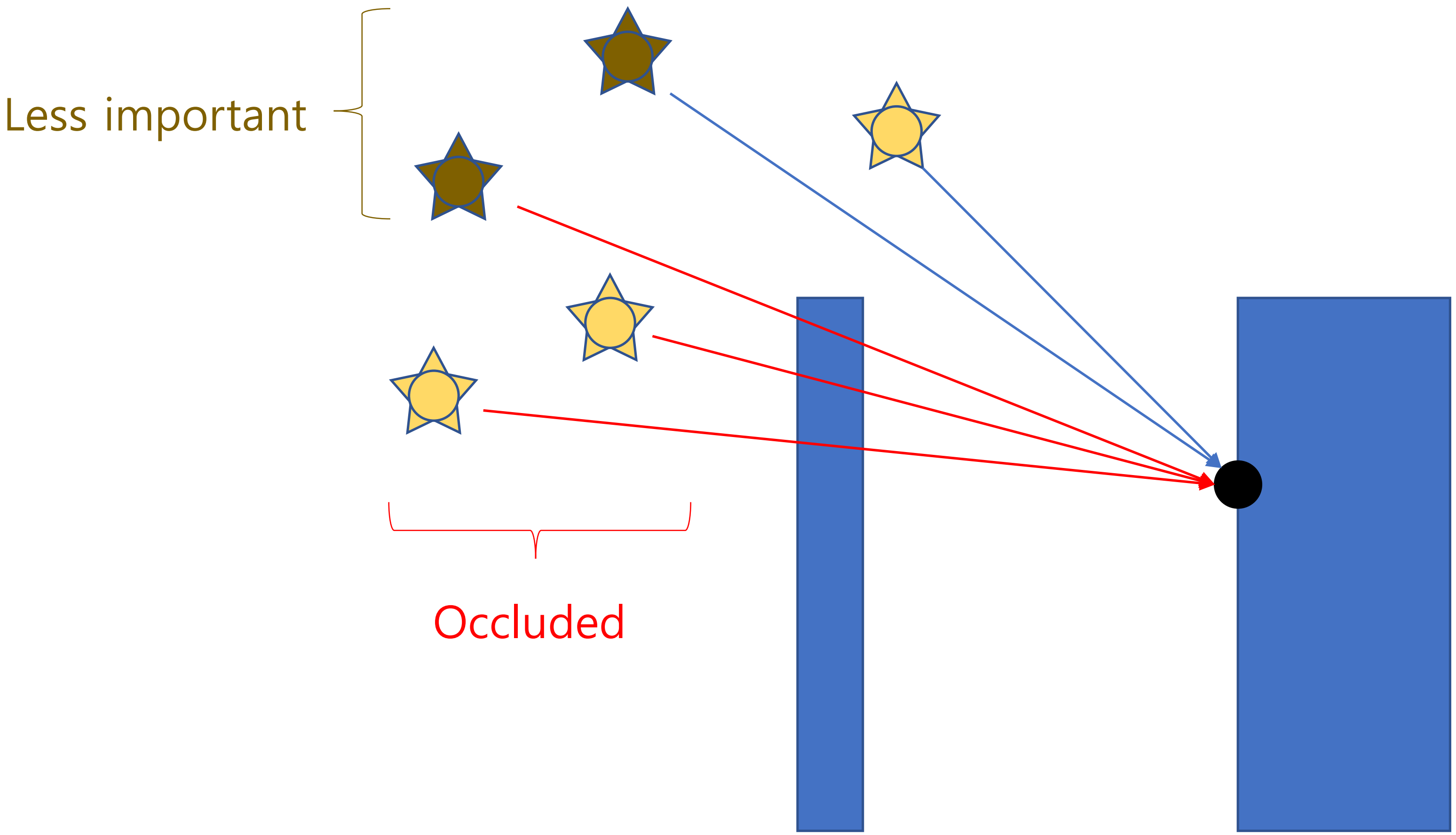
Noise in MC Rendering



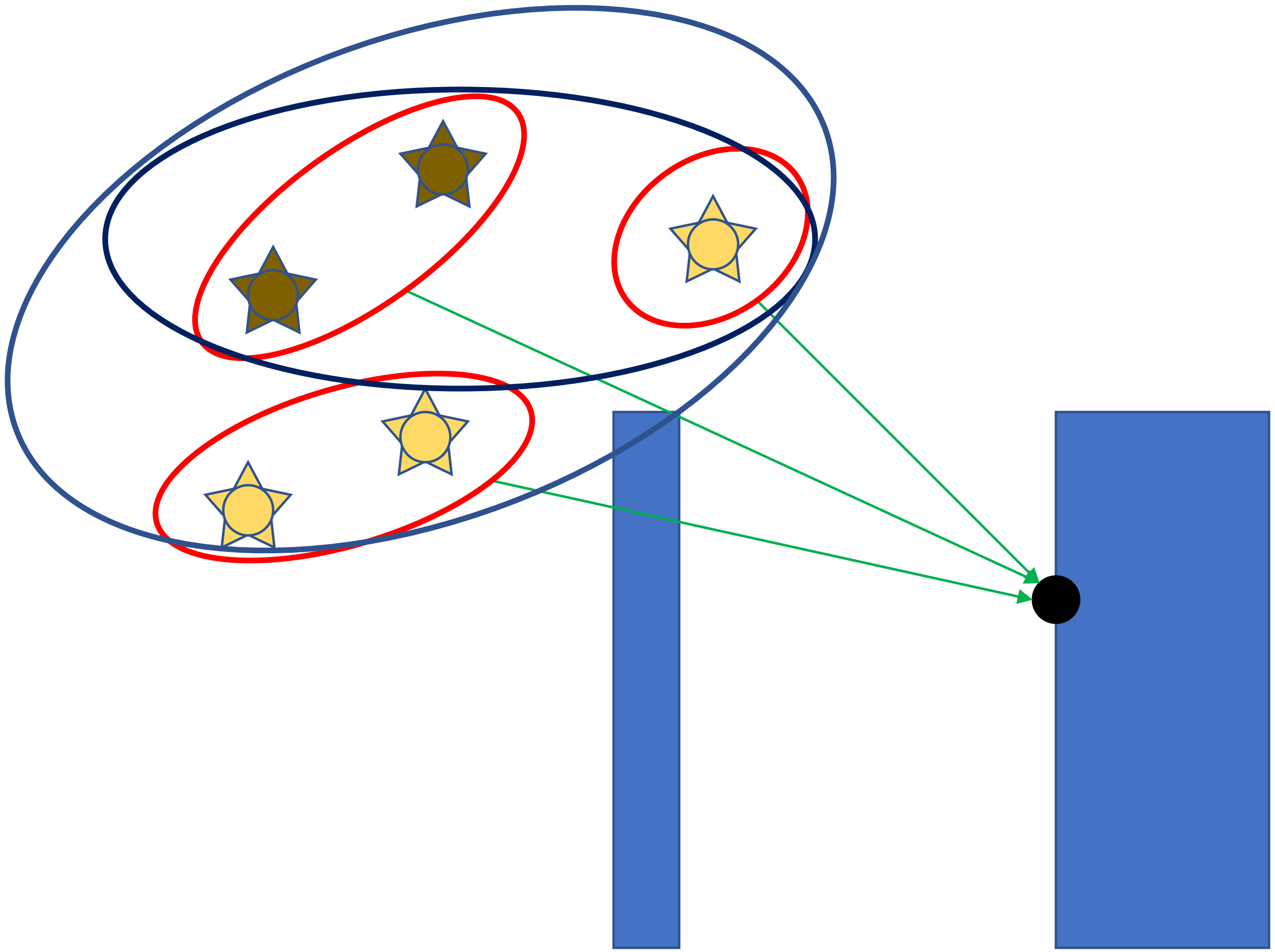
Noise in MC Rendering



Direct Illumination



Light Clustering

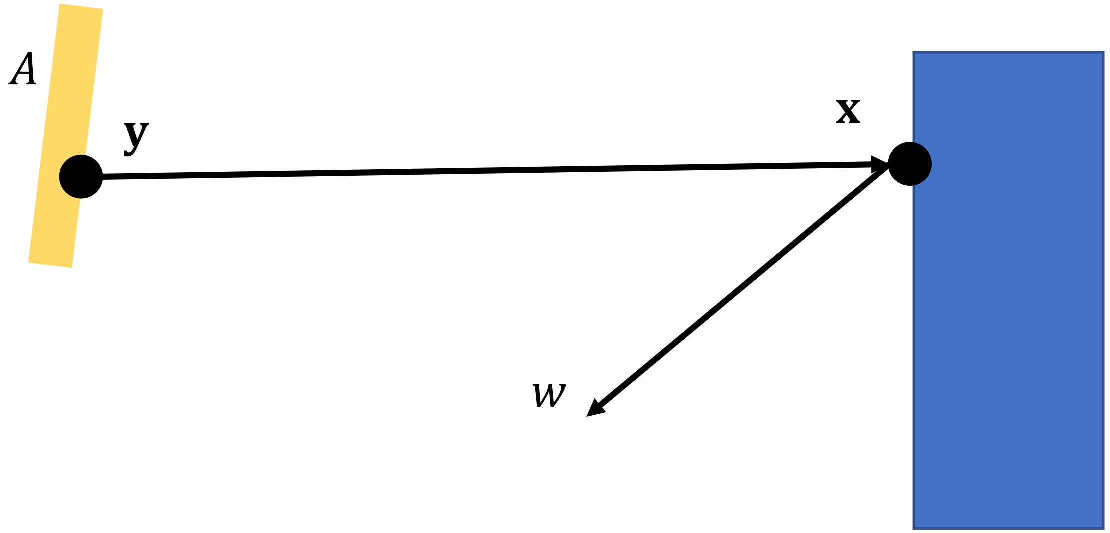


Direct Illumination

- $L(\mathbf{x}, w) = \int_A F(\mathbf{y} \rightarrow \mathbf{x} \rightarrow w) \mathrm{d}\mathbf{y}$

- $F(\mathbf{y} \rightarrow \mathbf{x} \rightarrow w) = L_e(\mathbf{y} \rightarrow \mathbf{x}) B(\mathbf{y} \rightarrow \mathbf{x} \rightarrow w) V(\mathbf{y} \leftrightarrow \mathbf{x}) G(\mathbf{y} \leftrightarrow \mathbf{x})$

- $\langle L(\mathbf{x}, w) \rangle = \frac{F(\mathbf{y} \rightarrow \mathbf{x} \rightarrow w)}{p(\mathbf{y} | \mathbf{x}, w)}$



Probability Density Function

- $p(\mathbf{y}|\mathbf{x}, w) = P(c|\mathbf{x})P(l|c)p(\mathbf{y}|l, w)$
  - $P(c|\mathbf{x})$: selecting a light cluster $\mathrm{c} \in \mathrm{C}$
  - $P(l|c)$: selecting a light $l \in c$, $P(l|c) = \Phi_l / \sum_{l' \in c} \Phi_{l'}$
  - $p(\mathbf{y}|l, w)$: selecting a point $y \in l$, using standard techniques in [Pharr et al. 2016; Shirley et al. 1996].

- Only need to predict $P(c|\mathbf{x})$.

Optimal Cluster Selection

- $Var[\langle L(\mathbf{x})\rangle] = -L(\mathbf{x})^2 + \sum_{c\in C} \frac{1}{P(c|\mathbf{x})} \int_{A_c} \frac{\left(F(\mathbf{y}\to\mathbf{x})\right)^2}{P(l|c)p(\mathbf{y}|l)} \mathrm{d}\mathbf{y}$

- $m_{2,c} = \int_{A_c} \frac{\left(F(\mathbf{y}\to\mathbf{x})\right)^2}{P(l|c)p(\mathbf{y}|l)} \mathrm{d}\mathbf{y}$: second moment of $\langle L_c(\mathbf{x})\rangle = \frac{F(\mathbf{y}\to\mathbf{x})}{P(l|c)p(\mathbf{y}|l)}$

- $P_{opt}(c|\mathbf{x}) \propto \sqrt{L_c^2(\mathbf{x}) + Var[\langle L_c(\mathbf{x})\rangle]}$

Bayesian Online Regression

- $\mathbf{D}$: training data
- θ: parameters
- $p(\mathbf{D}|\theta)$: a model describing the likelihood of $\mathbf{D}$ given θ

- Maximum likelihood (ML) estimate
  - Finding θ maximizing $p(\mathbf{D}|\theta)$
  - Prone to overfit when data is scarce
  - Gives poor approximations in early stages of rendering
- Maximum a posteriori (MAP) estimate
  - Finding θ maximizing $p(\theta|\mathbf{D}) \propto p(\mathbf{D}|\theta)p(\theta)$
  - More robust than ML estimate

Bayesian Online Regression

- $\hat{e} = \frac{L_e(\mathbf{y}\rightarrow\mathbf{x})V(\mathbf{y}\leftrightarrow\mathbf{x})\cos\theta_{\mathbf{y}}}{d^2(\mathbf{x},\mathbf{y})P(l|c)p(\mathbf{y}|l)}$, $\hat{e}_{\mathbf{x}} = \hat{e}\overline{\cos}\theta_{\mathbf{x}}$
- Data $\mathbf{D}$ consists of tuples $\left(\hat{e}_{\mathbf{x},i}, \hat{d}_i\right)$.
- $p(\mathbf{D}|\theta) = \prod_i^N p\left(\hat{e}_{\mathbf{x},i}|\hat{d}_i, \theta\right)$
- $p\left(\hat{e}_{\mathbf{x}}|\hat{d}, \theta\right) = \delta(\hat{e}_{\mathbf{x}})p_0 + (1 - p_0)N\left(\hat{e}_{\mathbf{x}}|\frac{k}{\hat{d}^2}, \frac{h}{\hat{d}^4}\right)$
- $\theta = (p_0, k, h)$
- $p(\theta) = B\left(p_0|\widehat{N}_o, \widehat{N}_v\right)N - \Gamma^{-1}\left(k, h|\mu_0, \widehat{N}, \widehat{N}_\alpha, \beta\right)$

Bayesian Online Regression

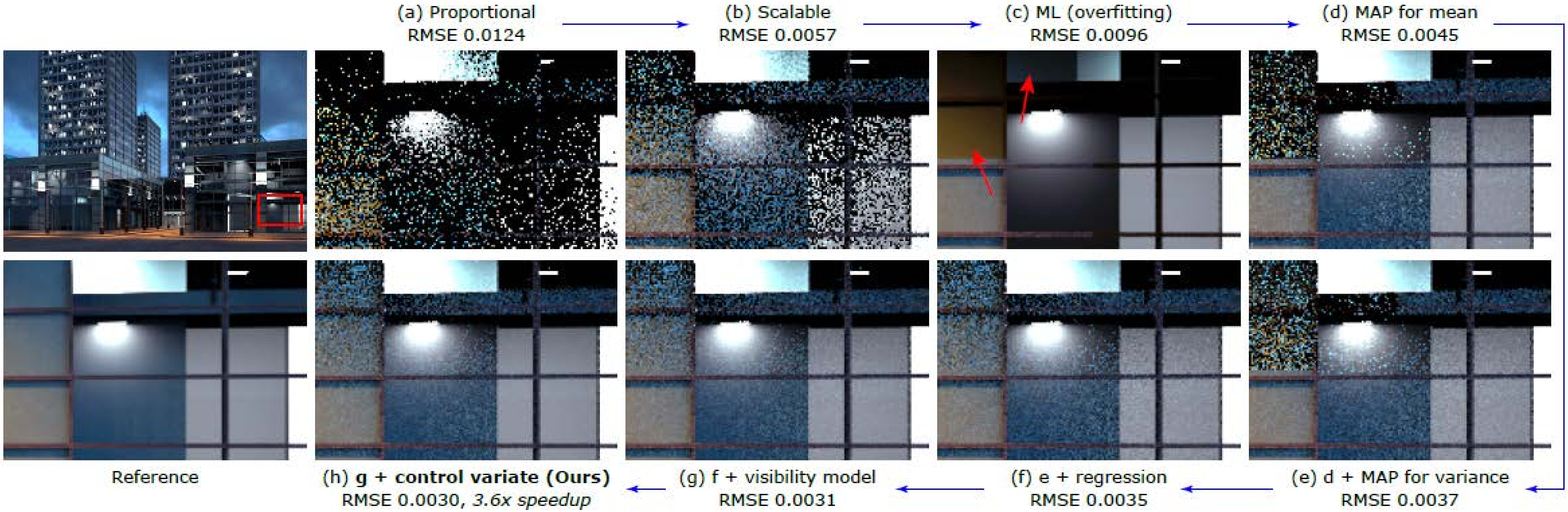
- $L_c(\mathbf{x}) \approx \frac{(1-p_0)k}{\hat{d}^2}$
- $Var[\langle L_c(\mathbf{x})\rangle] \approx \frac{(1-p_0)(p_0 k^2+h)}{\hat{d}^4}$

- $P^*(c|\mathbf{x}) \propto \frac{1}{\hat{d}^2}\sqrt{(1-p_0)^2 k^2 + (1-p_0)(p_0 k^2 + h)}$

Summary

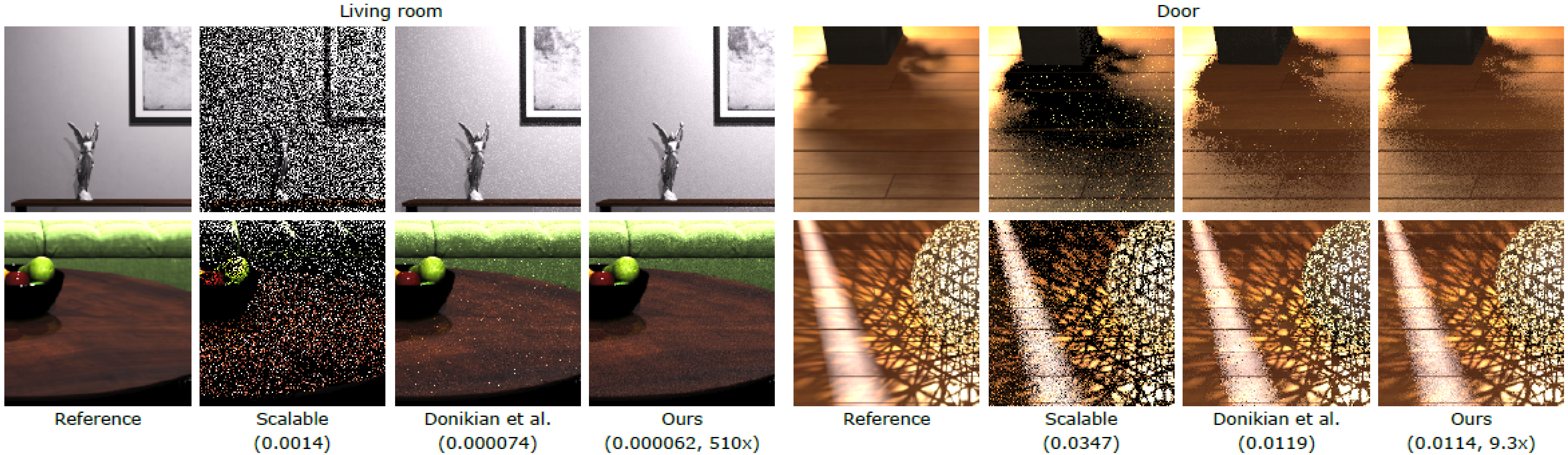
- Light preprocess (clustering)
- During each Next event estimation:
  - Obtain cached clustering.
  - Compute distributions of estimates for each cluster. (mean, variance)
  - Build distribution over clusters.
  - Sample direct illumination.
  - Record new data for sampled cluster.

Results



(a) Proportional
RMSE 0.0124
(b) Scalable
RMSE 0.0057
(c) ML (overfitting)
RMSE 0.0096
(d) MAP for mean
RMSE 0.0045
Reference
(h) g + control variate (Ours)
RMSE 0.0030, 3.6x speedup
(g) f + visibility model
RMSE 0.0031
(f) e + regression
RMSE 0.0035
(e) d + MAP for variance
RMSE 0.0037

Results


Living room
Door
Reference
Scalable
(0.0014)
Donikian et al.
(0.000074)
Ours
(0.000062, 510x)
Reference
Scalable
(0.0347)
Donikian et al.
(0.0119)
Ours
(0.0114, 9.3x)

Reference

- Simon Kallweit, Thomas Muller, Brian McWilliams, Markus Gross, and Jan Novak. 2017. Deep Scattering: Rendering Atmospheric Clouds with Radiance-Predicting Neural Networks. ACM Trans. Graph. 36, 6, Article 231 (November 2017), 11 pages.
- J. Novák at al. 2018. Deep Learning for Light Transport. Disney Research.
- Petr Vevoda, Ivo Kondapaneni, and Jaroslav Křivánek. 2018. Bayesian online regression for adaptive direct illumination sampling. ACM Trans. Graph. 37, 4, Article 125 (August 2018), 12 pages.

Thank you!

Any questions?

Quiz

Fill in the blanks with words in the box.

(a) Direct
(b) Indirect
(c) Diffuse
(d) Specular

1. In “Deep Scattering …“ paper, scattered (   ) illumination is predicted using a neural network.

2. In “Bayesian online regression …” paper, authors aim to reduce noise in (   ) illumination with Bayesian online learning.