PG 2024
Huangshan, China

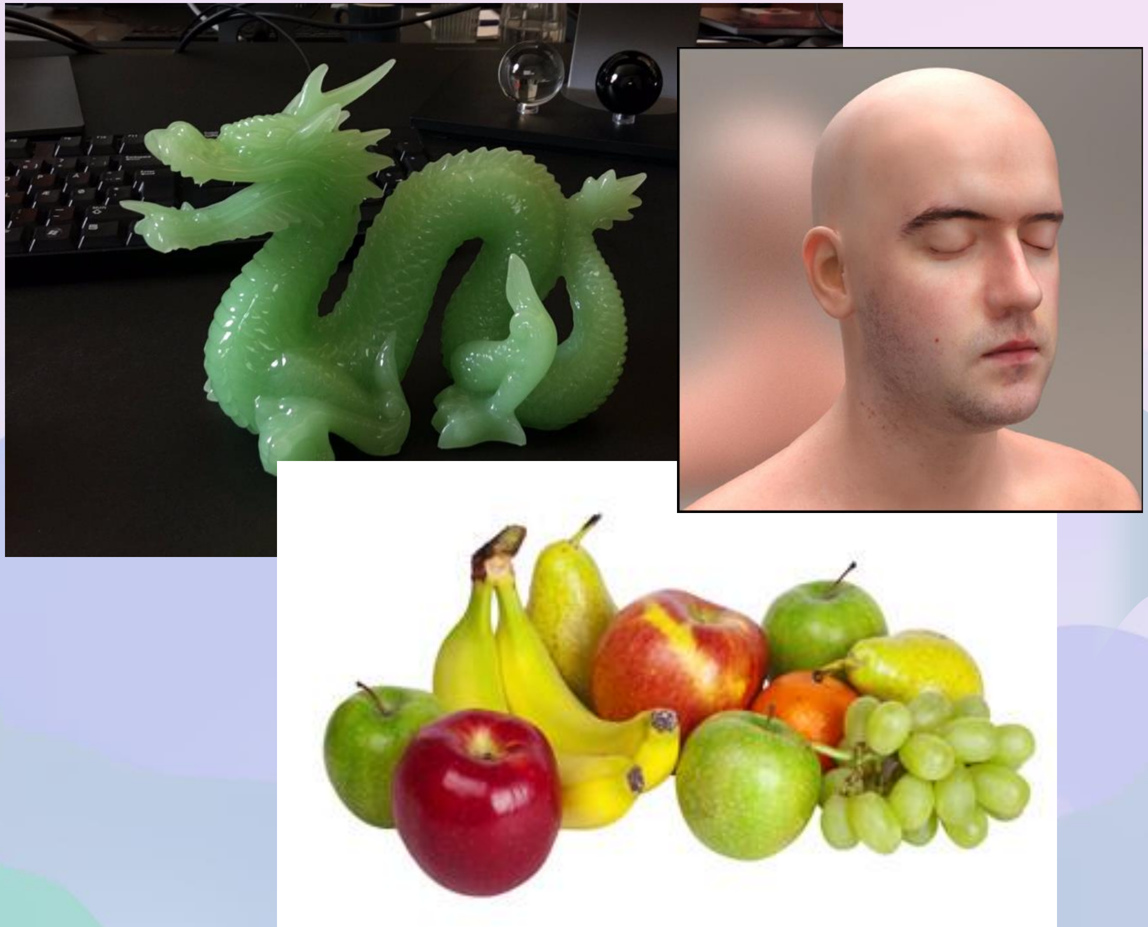
Inverse Rendering of Translucent Objects with Shape-Adaptive Importance Sampling

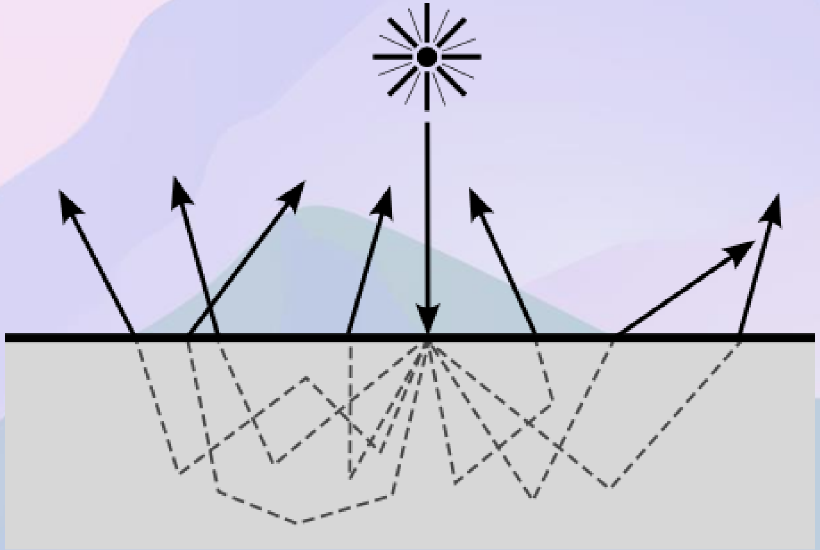
Jooeun Son, Yucheol Jung, Gyeongmin Lee, Soongjin Kim, Joo Ho Lee, Seungyong Lee

POSTECH COMPUTER GRAPHICS

SOGANG UNIVERSITY

Ubiquity of Translucent Objects

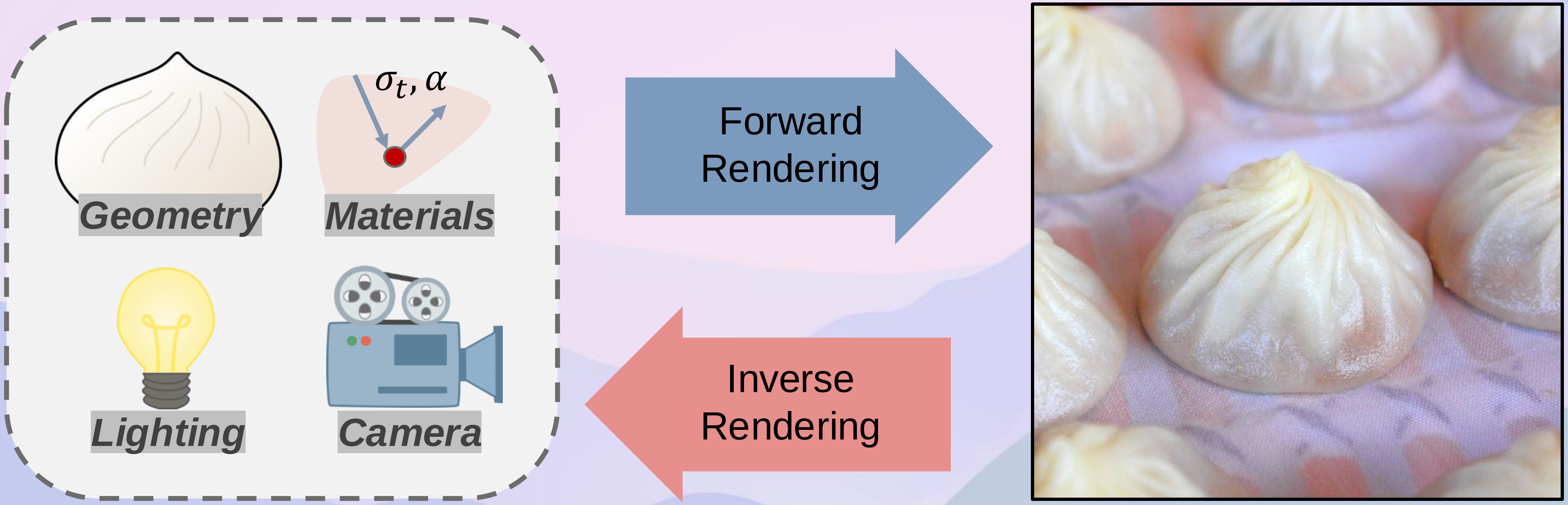




Subsurface Scattering

[Vicini 2019] “A Learned Shape-Adaptive Subsurface Scattering Model.”

Inverse Rendering of Translucent Objects



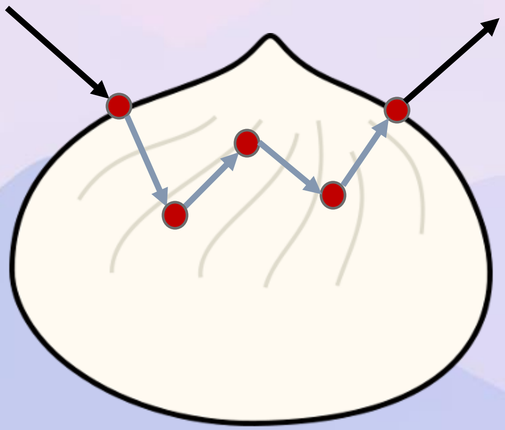
Rendered Image

Need a Differentiable Rendering Algorithm!

Related Work

Inverse Rendering of Translucent Objects

Volumetric Path Tracing



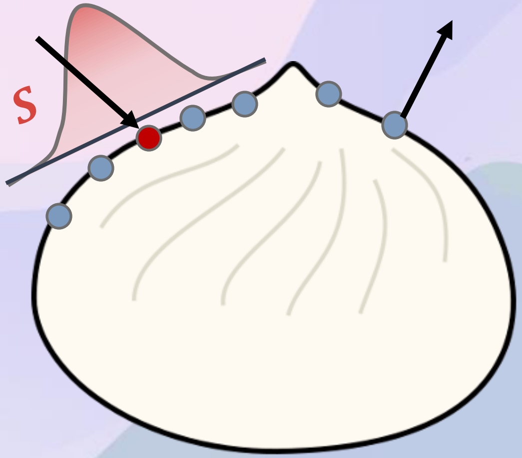
👍 *Physical Accuracy*

👎 *Expensive*

Radiative Transfer Equation (RTE)

$$\frac{\partial}{\partial t} L_o(p,\omega) = -\sigma_t(p,\omega) L_i(p,-\omega) + \sigma_t(p,\omega) L_s(p,\omega)$$

BSSRDF Methods



👍 *Efficient*

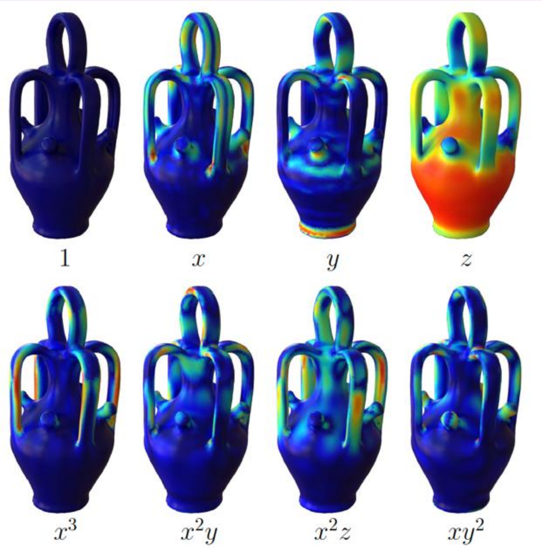
👎 *Less Realistic*

Rendering Equation

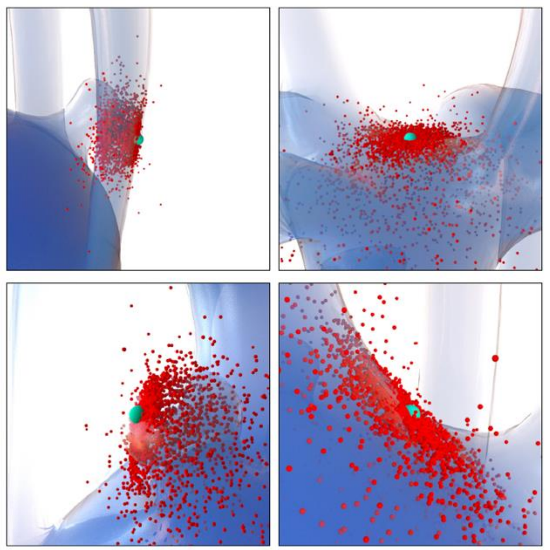
$$L_o(p,\omega) = \int_A \int_\Omega L_i(p,\omega')\, S(p,p',\omega,\omega') \cos\theta \; d\omega' dp'$$

Related Work

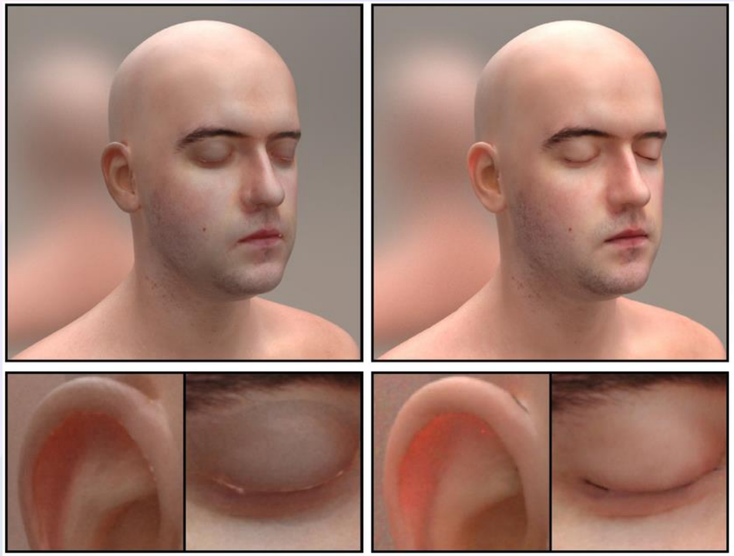
Shape-adaptive BSSRDF



Shape Descriptors



Learned Exit Position Distribution



Beam Dipole *Shape-adaptive*

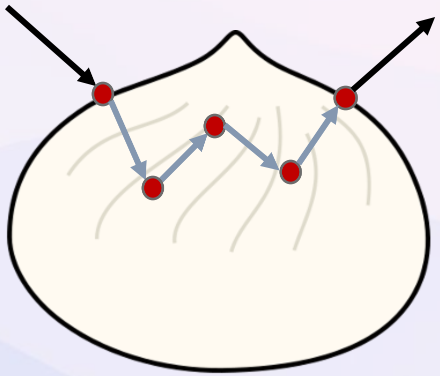
Efficient & More Accurate!

[Vicini 2019] "A Learned Shape-Adaptive Subsurface Scattering Model."

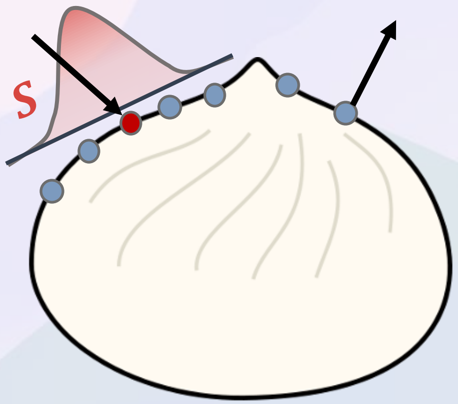
Our Contribution

Differentiable Shape-adaptive BSSRDF!

→ *Reconstruction of Optical Parameters with* *Differentiable Shape-adaptive BSSRDF*



👍 ***Accuracy* of**
Volumetric Path Tracing



👍 ***Efficiency* of**
BSSRDF Methods

Preliminaries

The Shape-adaptive BSSRDF

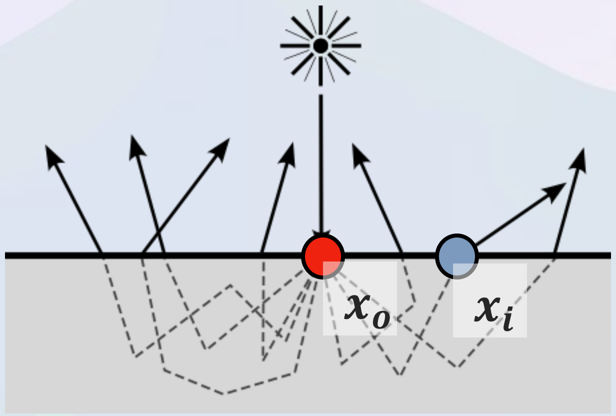
PG 24
Huangshan
China

BSSRDF

- → **B**idirectional **S**cattering **S**urface **R**eflectance **D**istribution **F**unction
  - → Returns ratio of reflected radiance (x_o, ω_o) to incident irradiance (x_i, ω_i)
  - → Ex. Dipole Model [Jensen 2001]

$$S(x_i, \omega_i, x_o, \omega_o) = S_\omega(\omega_i)\ \boldsymbol{S_p}(x_i, x_o)\ S_\omega(\omega_o)$$

- → ***Not physically accurate*** **due to assumptions**
  - → $\boldsymbol{S_p}$ is a radial function $\boldsymbol{S_p}(||\boldsymbol{x_i} - \boldsymbol{x_o}||)$
  - → The local geometry is ***planar***



Shape-adaptive BSSRDF

Keypoint #1. Shape-Adaptivity

$$S(x_i, \omega_i, x_o, \omega_o) = S_\omega(\omega_i)\, \boldsymbol{S_p}\left(\left|\left|x_i - x_o\right|\right|; \theta_{med}\right) S_\omega(\omega_o)$$

Classical BSSRDF (Planar)

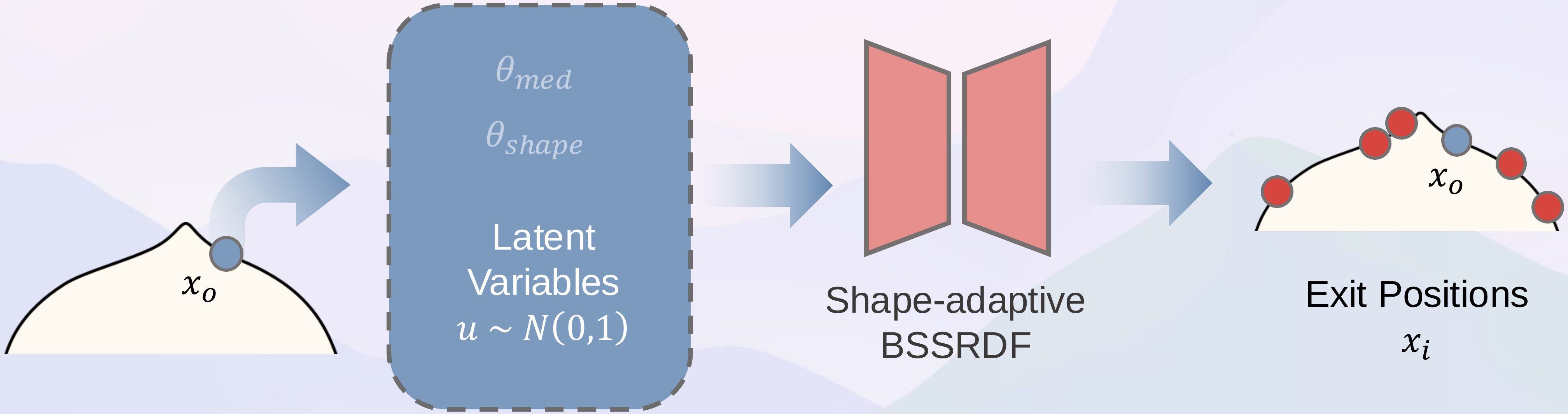
$$S(x_i, \omega_i, x_o, \omega_o) = \boldsymbol{S_p}\left(x_i, x_o, \omega_i;\ \theta_{med}, \boldsymbol{\theta_{shape}}\right) S_\omega(\omega_o)$$

Shape-adaptive BSSRDF

$\boldsymbol{\theta_{shape}}$

Shape-adaptive BSSRDF

Keypoint #2. Importance Sampling Framework



→ ***NO** analytic BSSRDF or Sampling PDF!*

Method

Differentiable Shape-adaptive BSSRDF

Problem Overview

Challenges

Differentiable Rendering with Shape-adaptive BSSRDF

Solution : Just use Automatic Differentiation?

Why?

No explicit BSSRDF !

Cannot Invert Sampling Transform

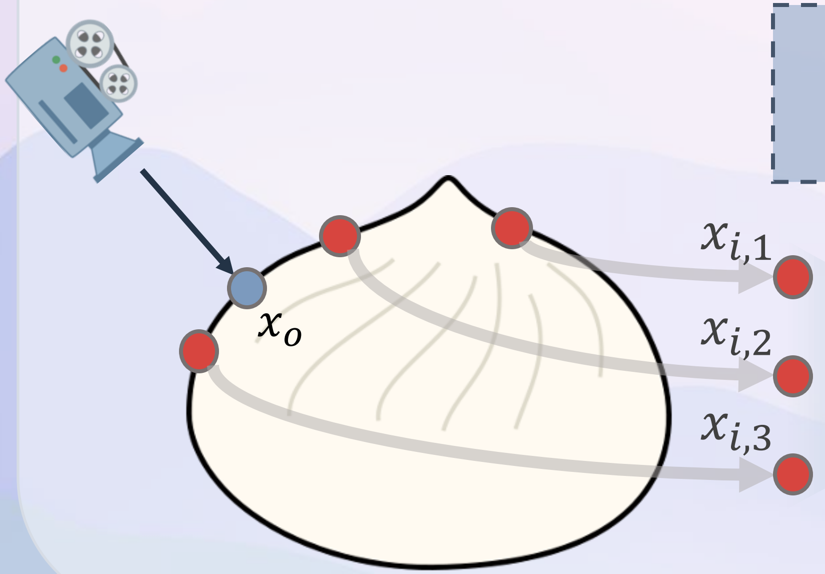
$$L_o(x_o, \omega_o) = \sum_k L_i(x_{i,k}, \omega_{i,k})\, weight(x_{i,k})$$

Challenge #1

Implicitly Defined BSSRDF

→ **Weights determine contribution of each sample to pixel intensity**

$$weight \propto \frac{\cancel{S_p(x_o, x_i, \omega_o)}}{\cancel{pdf(x_i; x_o)}} = w$$



$$w(x_{i,1}) = w(x_{i,2}) = w(x_{i,3})$$

"Uniform Weights"

$$\sum_k L_i * weight$$

Pixel Intensity

Challenge #1

Implicitly Defined BSSRDF

Primal Rendering

$$weight(\bullet) \propto \frac{\cancel{S_p(x_o, x_i, \omega_o; \theta)}}{\cancel{pdf(x_i; x_o, \theta)}}$$

Do not need S_p or p !

Differential Rendering

$$\frac{d\, weight(\bullet)}{d\theta} \propto \frac{S_p' \cdot pdf - S_p \cdot pdf'}{pdf^2}$$

Cannot compute derivatives without S_p and pdf !

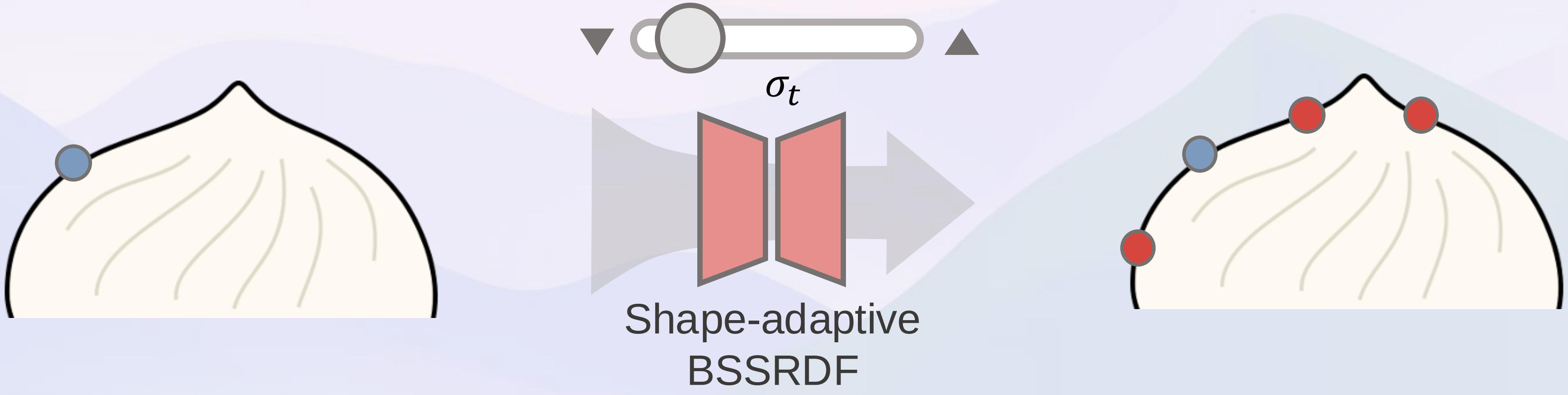
Challenge #1

Implicitly Defined BSSRDF

$$weight \propto \frac{\cancel{S_p(x_o, x_i, \omega_o)}}{\cancel{pdf(x_i; x_o)}}$$

→ **Use same weights as in Primal Phase for Differential Phase**

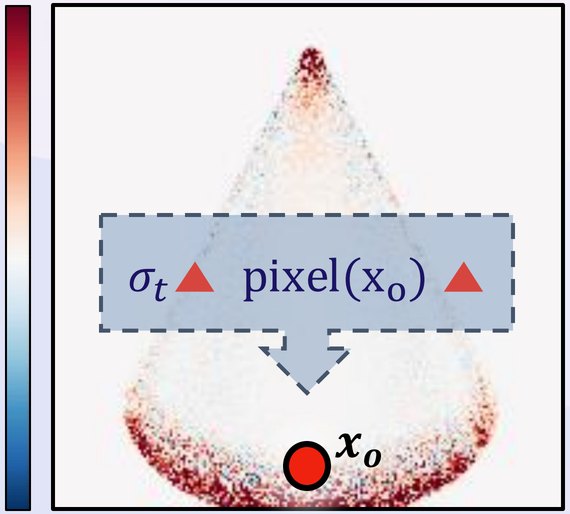
→ **...*BIASED!* Does NOT account for *how medium affects distribution of samples***



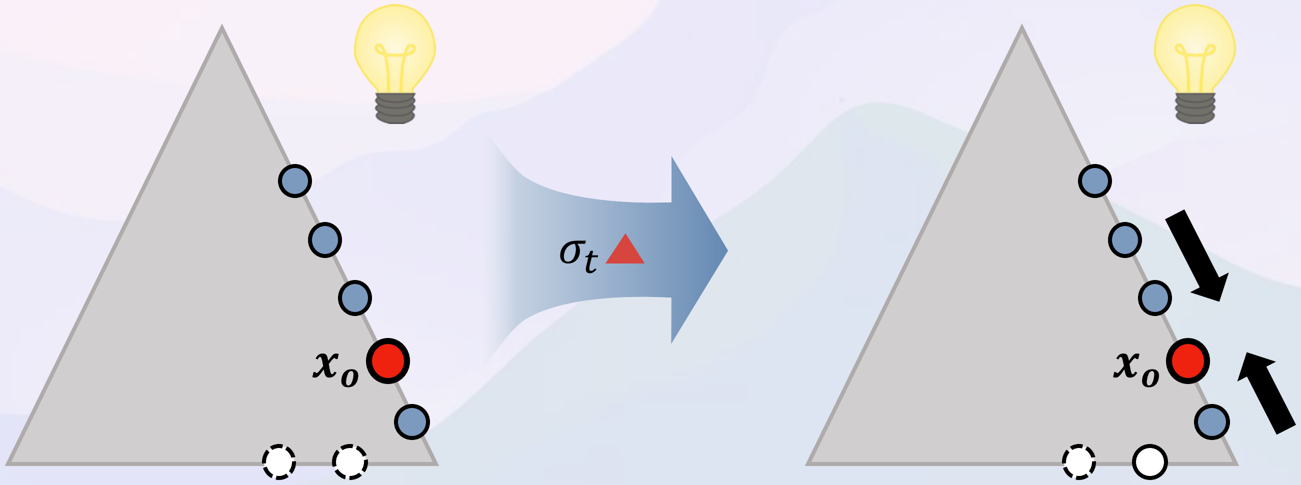
Problem

Parameter-dependent Visibility Discontinuities (σ_t)

→ ***Visibility discontinuities MOVE*** **when importance sampling w.r.t. optimizing parameters**



FD Reference

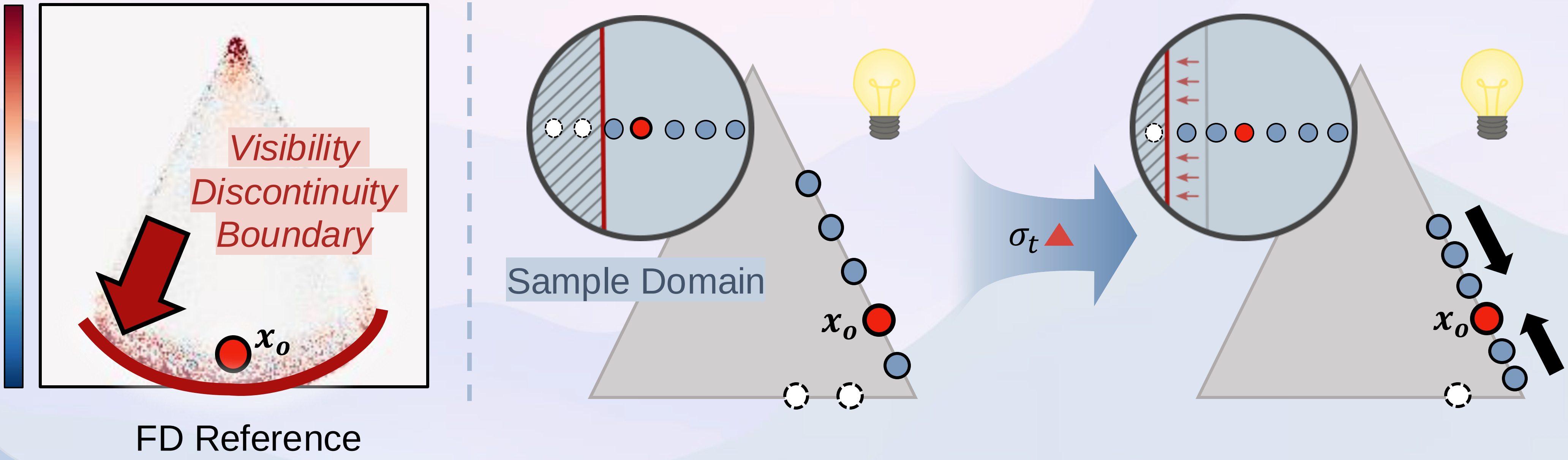


Occluded samples become visible

Problem

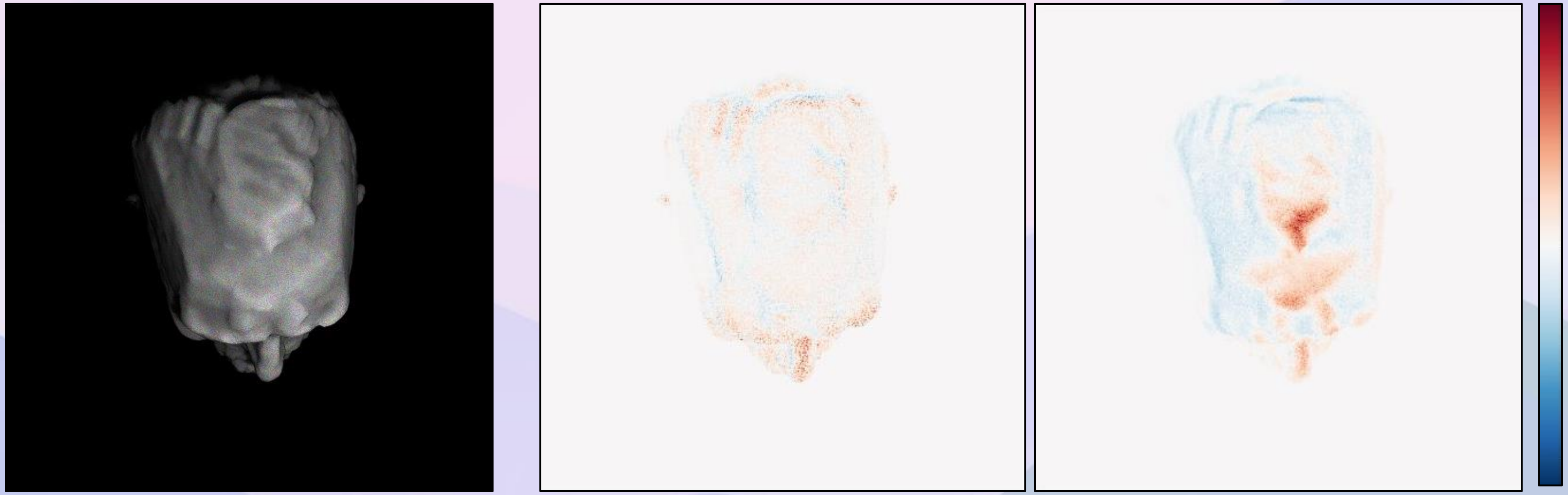
Parameter-dependent Visibility Discontinuities (σ_t)

→ *Visibility discontinuities MOVE* **when importance sampling w.r.t. optimizing parameters**



Problem

Parameter-dependent Visibility Discontinuities (σ_t)



(a) Primal Rendering (b) FD Reference (c) Naïve Autograd

Biased Gradients without handling moving visibility discontinuities!

Previous Methods

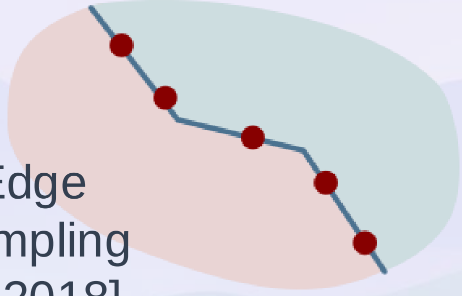
Differentiating Moving Discontinuities

- Reynolds Transport Theorem

$$\frac{\partial}{\partial \pi}\int_{\Omega} f(p)\, d\mu(p) = \int_{\Omega} \frac{\partial f(p)}{\partial \pi}\, d\mu(p) + \int_{\partial\Omega} \Delta\, f(p) v_B d\mu'(p)$$

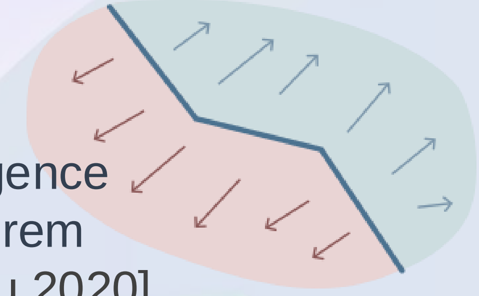
Interior Term   Boundary Term

Boundary-based



Edge Sampling [Li 2018]

Area-based



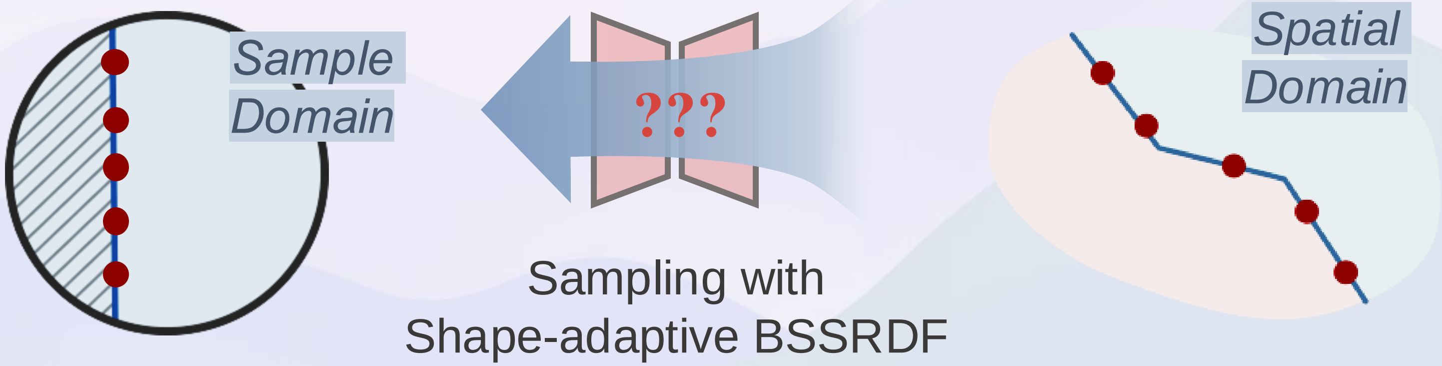
Divergence Theorem [Bangaru 2020]

Challenge #2

Non-Invertibility of Sampling Transform

→ Boundary derivative methods rely on ***invertibility of sampling transform***

→ ***Non-invertible layers*** **in sampling network (e.g. ReLU) !**



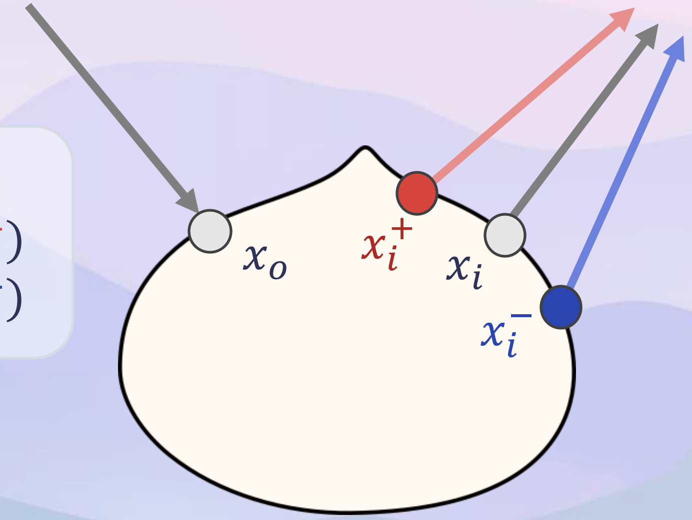
→ **We need to *implicitly* handle boundary effects *without inversion***

Our Solution

Derivatives using Offset Samples (σ_t)

- ► Offset Samples: samples obtained by slightly perturbing medium parameters of the primal sample

$$x_i = T(u_i; \theta)$$
$$x_i^+ = T(u_i; \theta^+)$$
$$x_i^- = T(u_i; \theta^-)$$



Finite Differences in Sample Domain

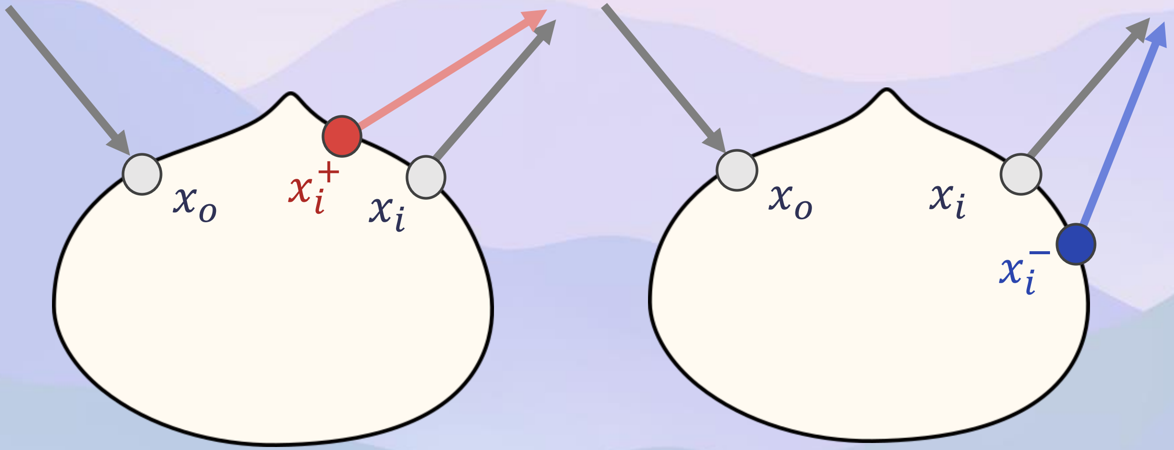
$$\partial_\theta L_o \approx \sum_{x_i} \frac{weight(x_i^+) - weight(x_i^-)}{2\Delta\theta}$$

Our Solution

Comparison To Finite Differences

Finite Differences

$$\partial_\theta L_o \approx \frac{1}{2\Delta\theta}\left(\sum_{x}^{N} weight(\boldsymbol{x_i^+}) - \sum_{x}^{N} weight(\boldsymbol{x_i^-})\right)$$



$$N\ paths + N\ paths = \boldsymbol{2N\ paths}$$

No correlation of paths!

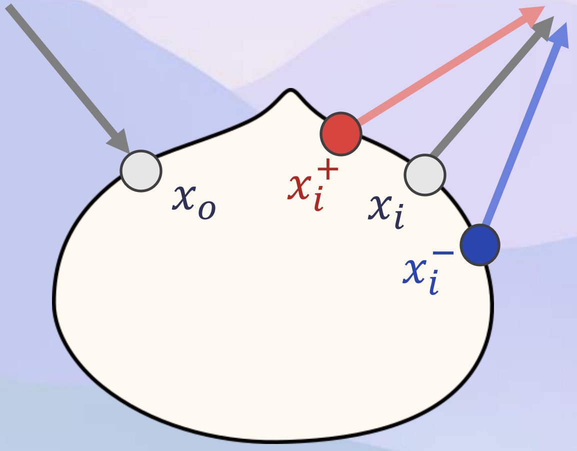
- ***Independent sampling*** **for positive / negative paths**
- ***High-variance*** **gradients**

Our Solution

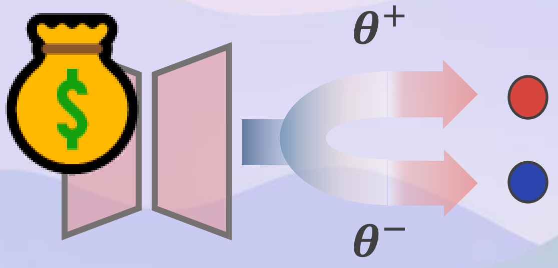
Comparison To Finite Differences

Ours

$$\partial_\theta L_o \approx \sum_{x_i} \frac{weight(\,x_i^{+}) - weight(\,x_i^{-})}{2\Delta\theta}$$



N paths



Single Network Evaluation for Both Samples

Maximized Efficiency

- ***Shared components*** for positive / negative paths
- ***Low-variance*** gradients

Summary

→ Challenges in Differentiation due to Importance Sampling
  → ***Implicitly defined BSSRDFs*** hinder weight differentiation
  → ***Non-invertibility*** of sampling transform hinder addressing discontinuity issues

→ ***Solution: Offset Sample*-based Finite Differences**
  → ***Efficient*** reuse of paths compared to Naïve FD
  → ***Inherent handling*** of boundary effects without sampling

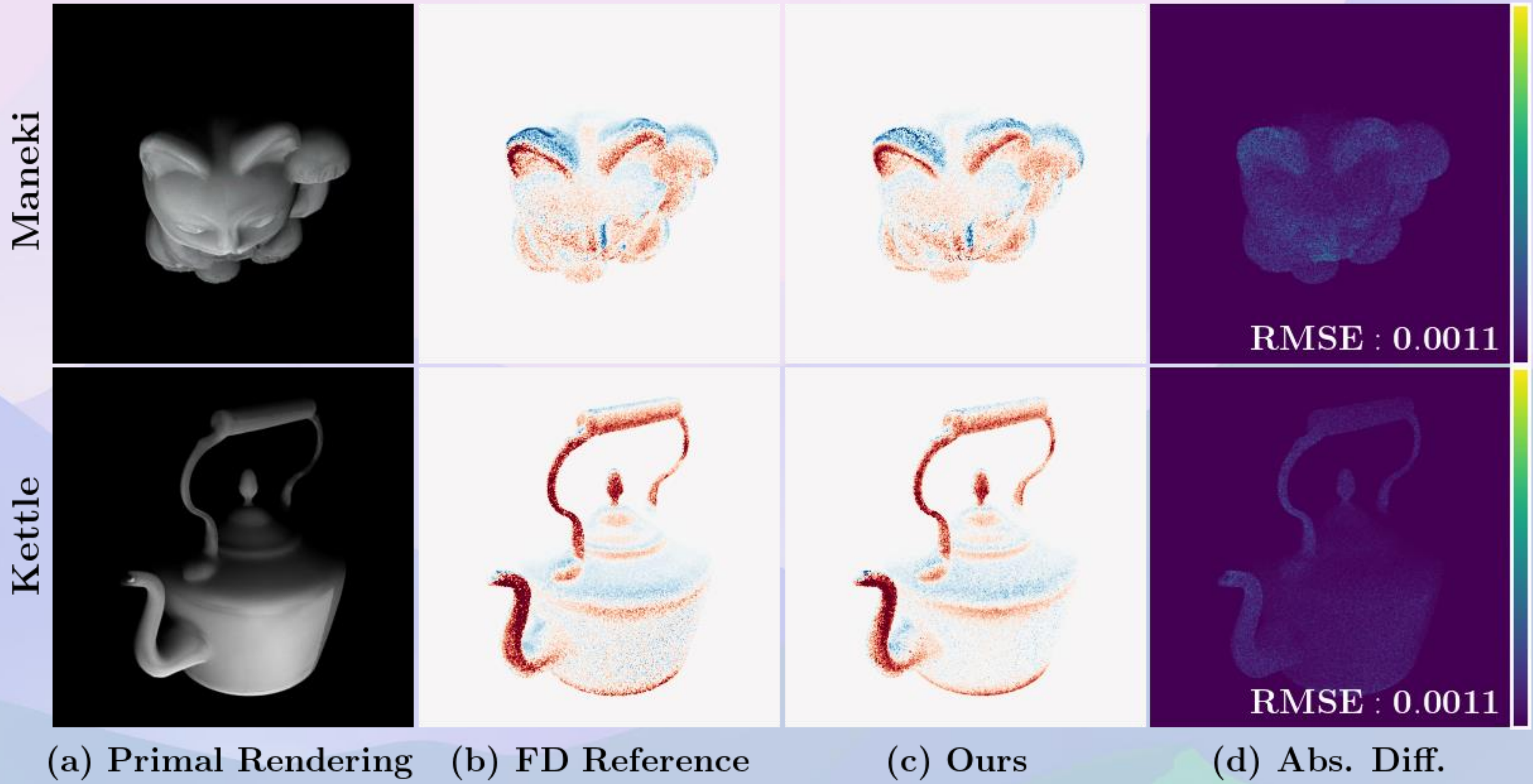
→ Absorption Probability as *Differentiable Scaling Factor* (α)
  → Details in Main Paper!

Results

PG 24
Huangshan
China

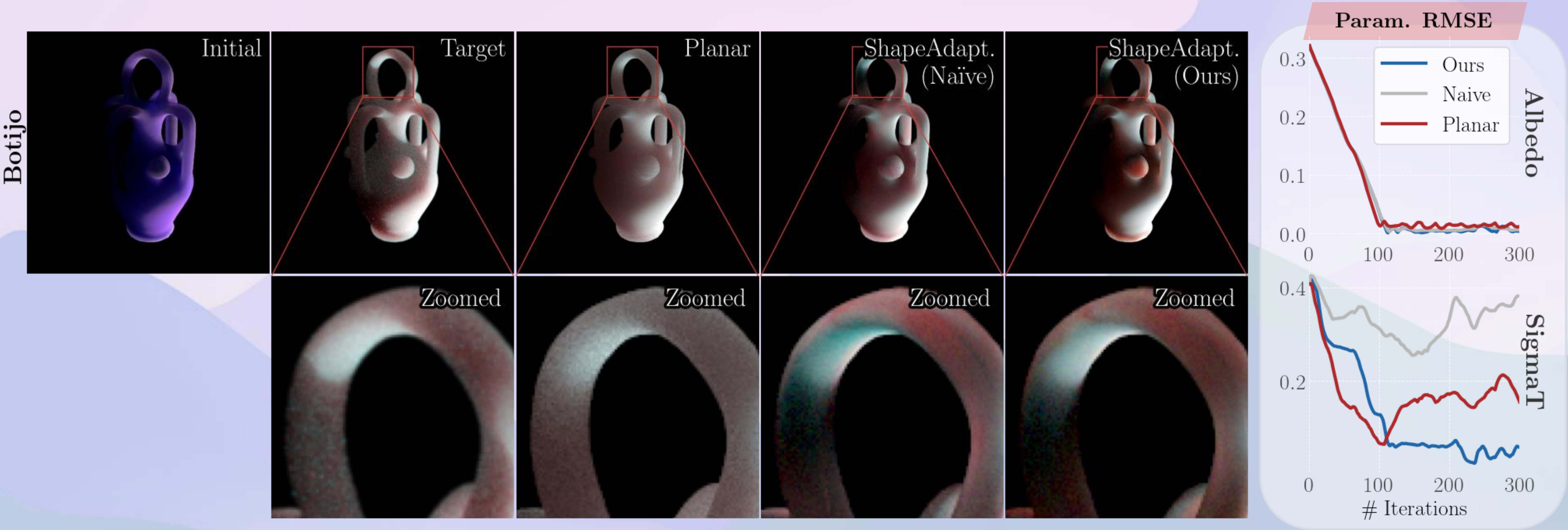
Validation

Comparison with Finite Differences (σ_t)



Comparison with BSSRDF Methods

Differentiable Rendering with Planar BSSRDF

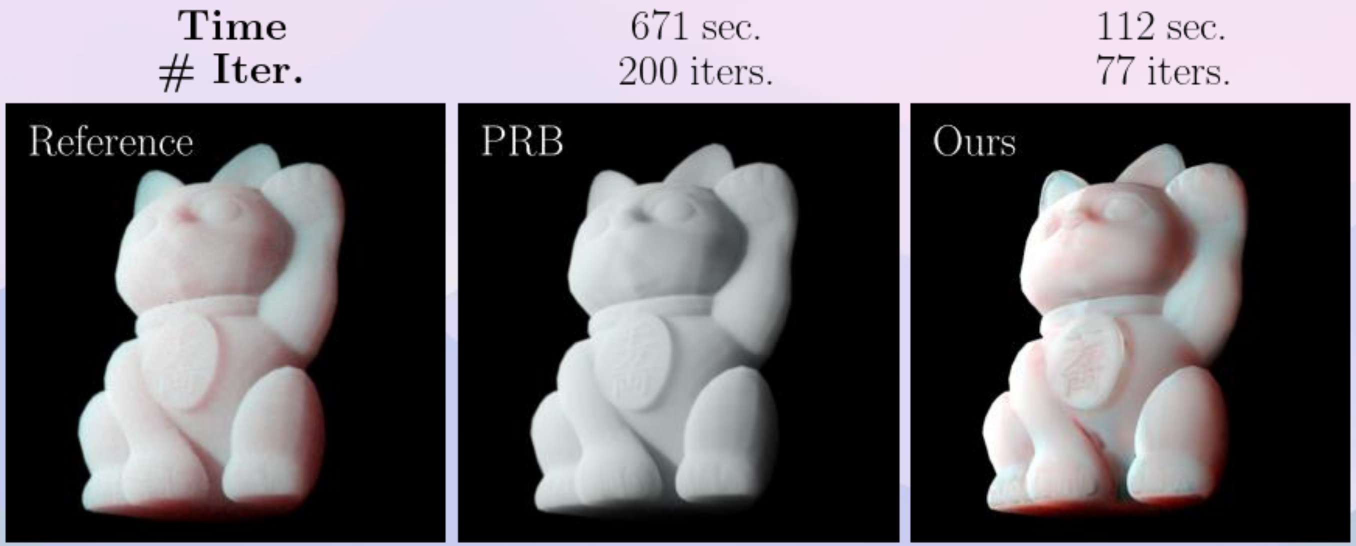


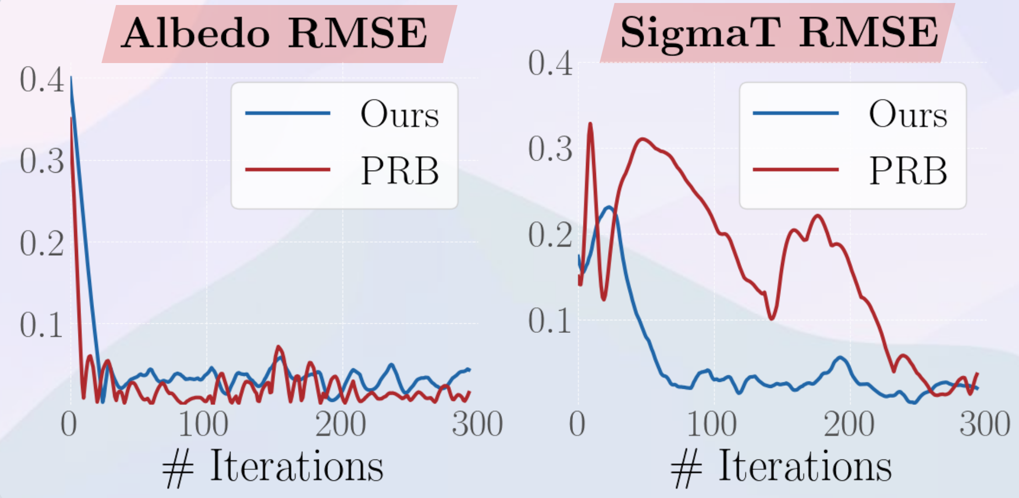
"Planar" : [Deng 2022] "Reconstructing Translucent Objects Using Differentiable Rendering"

"Naïve" : Ours w/o offset sampling

Comparison with Volumetric Methods

Path Replay Backpropagation (PRB)

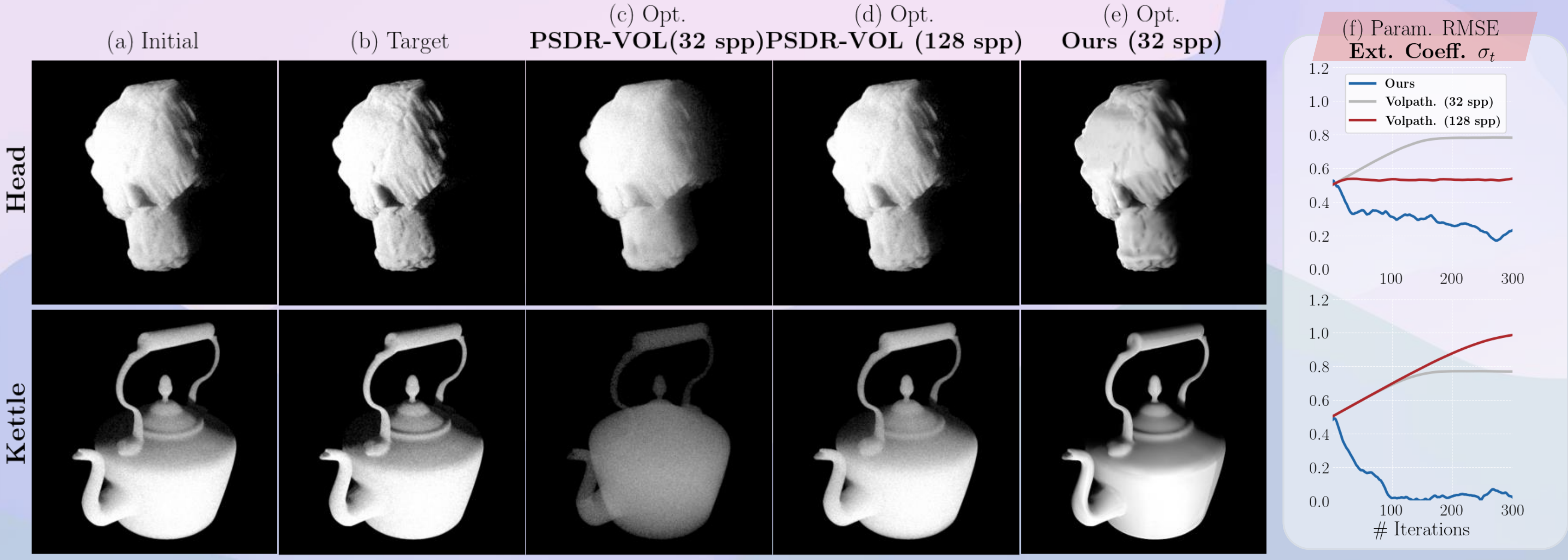




[Vicini 2021] "Path Replay Backpropagation: Differentiating Light Paths Using Constant Memory and Linear Time."

Comparison with Volumetric Methods

PSDR-VOL



[Zhang 2021] “Path-Space Differentiable Rendering of Participating Media.”

Comparison with Volumetric Methods

Time Efficiency

Ours (32 spp)	Ours (64 spp)	PRB (32 spp)	PSDR-VOL (32 spp)
1.40	2.80	2.28	4.07

[Table 1] Average time per iteration (sec.) of different approaches

Conclusion

PG 24
Huangshan
China

Summary

- Differentiable rendering based on ***finite differences in sample domain***
  - **Effective** due to inherent capturing of boundary integral effects.
  - **Efficient** due to utilizing correlation of sampled paths

- Comparison
  - Better sampling efficiency & time-efficiency compared to **volumetric methods**
  - Better accuracy compared to **planar BSSRDF model**

Limitations & Future Work

- Suitable for low-dimensional parameters
- Extending to heterogeneous medium
  - Can be approximated with textured spatial variation
- Extending to other sampling network architectures
  - Invertible sampling networks
  - Models with an explicit pdf for weight differentiation

Thank You for Listening!

Email me at jeson@postech.ac.kr

PG 24 Huangshan China

References

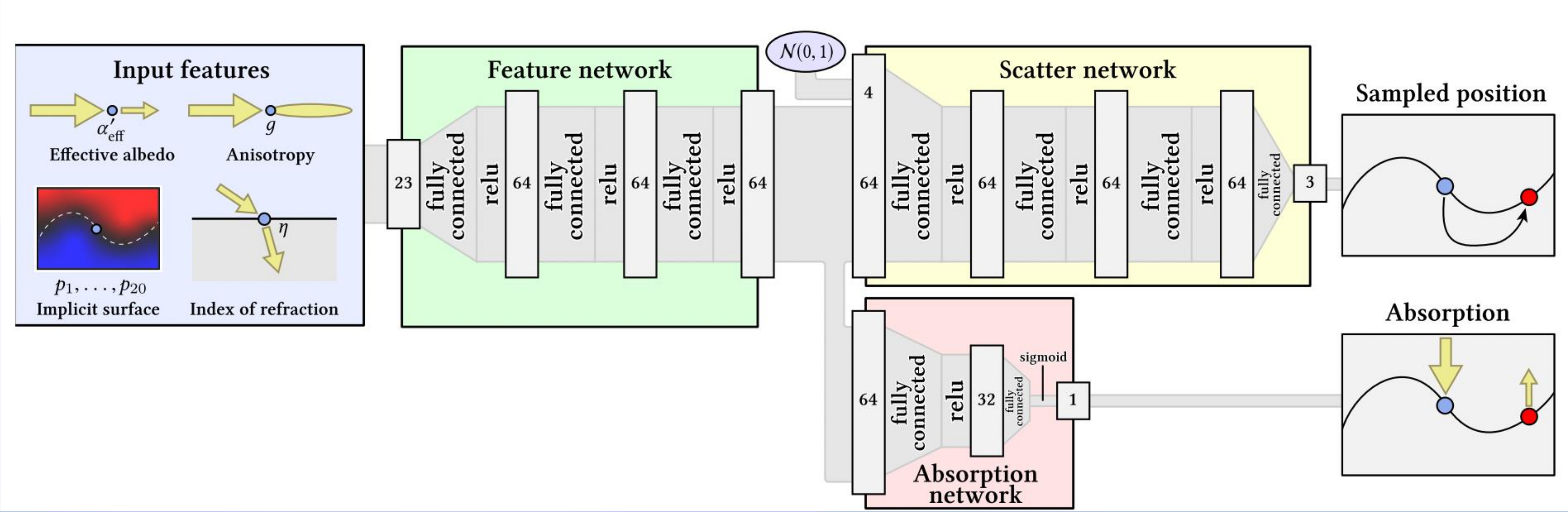
- **[Jensen 2001]** Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. 2001. A practical model for subsurface light transport. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH '01). Association for Computing Machinery. https://doi.org/10.1145/383259.383319
- **[Vicini 2019]** Vicini, Delio, Vladlen Koltun, and Wenzel Jakob. 2019. “A Learned Shape-Adaptive Subsurface Scattering Model.” ACM Transactions on Graphics 38 (4). https://doi.org/10.1145/3306346.3322974
- **[Bangaru 2020]** Bangaru, Sai Praveen, Tzu Mao Li, and Frédo Durand. 2020. “Unbiased Warped-Area Sampling for Differentiable Rendering.” ACM Transactions on Graphics 39 (6). https://doi.org/10.1145/3414685.3417833.
- **[Li 2018]** Li, Tzu-Mao, et al. "Differentiable monte carlo ray tracing through edge sampling." *ACM Transactions on Graphics (TOG)* 37.6 (2018): 1-11.

References

- **[Deng 2022]** Deng, Xi, Fujun Luan, Bruce Walter, Kavita Bala, and Steve Marschner. 2022. "Reconstructing Translucent Objects Using Differentiable Rendering." In , 1–10. Association for Computing Machinery (ACM). https://doi.org/10.1145/3528233.3530714.
- **[Vicini 2021]** Vicini, Delio, Sébastien Speierer, and Wenzel Jakob. 2021. "Path Replay Backpropagation: Differentiating Light Paths Using Constant Memory and Linear Time." ACM Transactions on Graphics 40 (4). https://doi.org/10.1145/3450626.3459804.
- **[Zhang 2021]** Zhang, Cheng, Zihan Yu, and Shuang Zhao. 2021. "Path-Space Differentiable Rendering of Participating Media." *ACM Transactions on Graphics* 40 (4). https://doi.org/10.1145/3450626.3459782.

Appendix

A. Shape-adaptive BSSRDF Architecture



B. Sampling Transform

Importance Sampling Transform

$$x_i^+ = \Pi(x_o + \sigma_n(\theta^+) \cdot \ x_{VAE}(u))$$

Canonical Sample
$u \sim N(0,1)$

eed to VAE
$\boldsymbol{x_{VAE}(u)}$

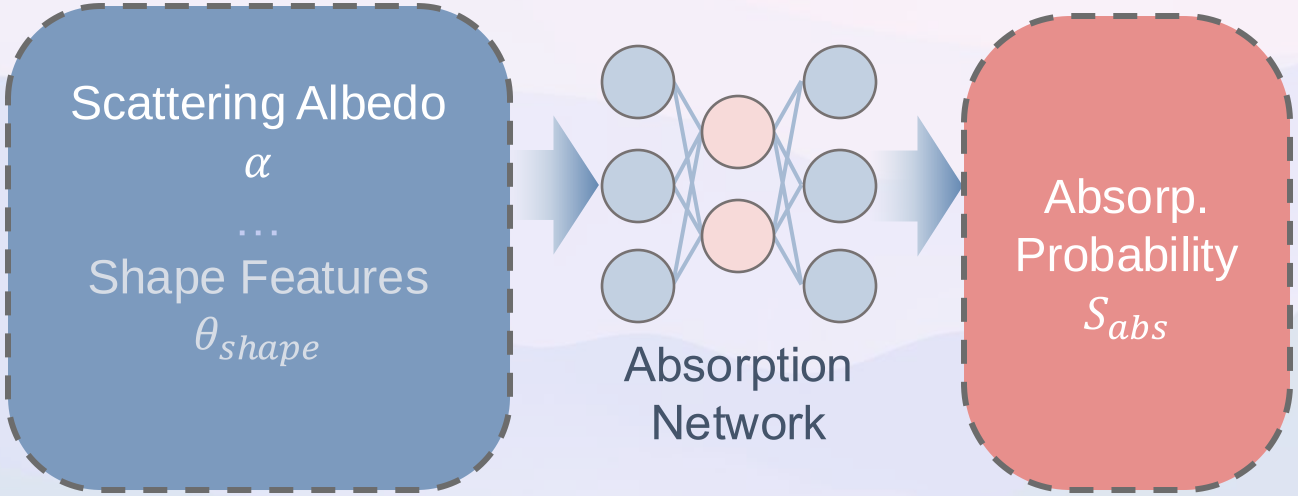
Scale by neighborhood size
$\boldsymbol{\sigma_n(\theta)}$

Project to surface
$\boldsymbol{\Pi}$

Final exit position
$\mathrm{x_i}$

C. Derivative of Scattering Albedo α

Primal Rendering



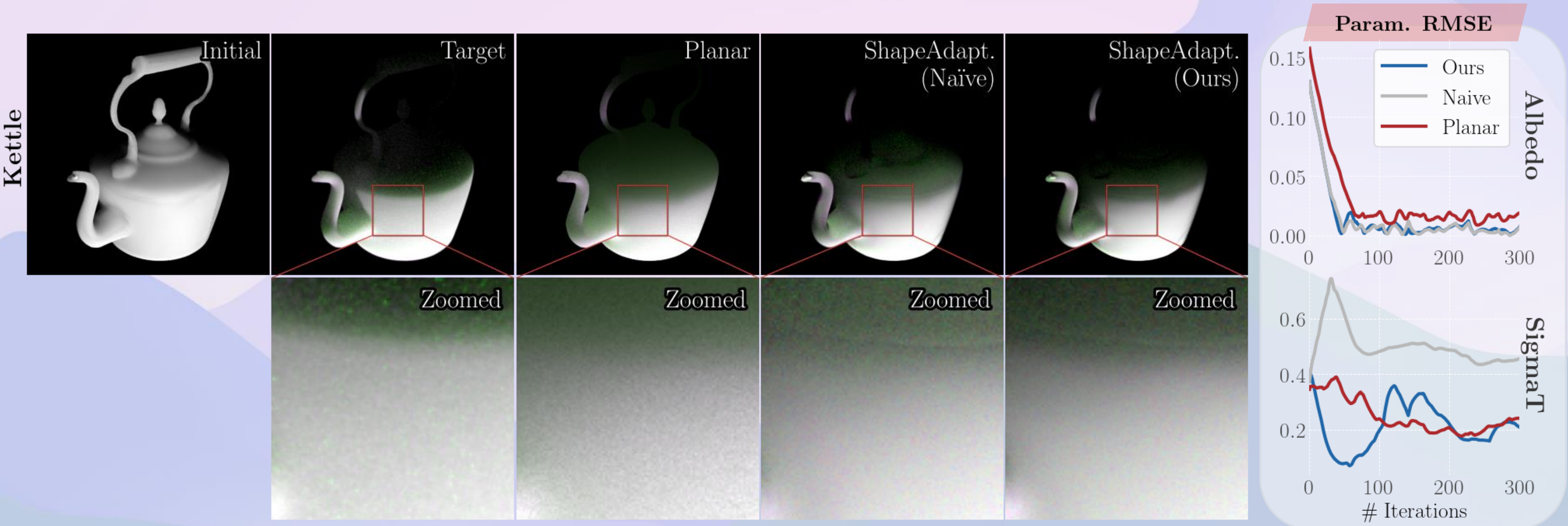
Differentiable Scaling Factor

$$weight(x_i) \propto \frac{S_p(x_o, x_i, \omega_o)}{p(x_i; x_o)}$$

$$\propto \boldsymbol{S_{abs}}$$

→ ***Can use standard Auto-diff***

D. More Comparisons with Planar BSSRDF



"Planar" : [Deng 2022] "Reconstructing Translucent Objects Using Differentiable Rendering"
"Naïve" : Ours w/o offset sampling