Achieving Visual Richness using Micro-scale BRDFs

CS580 Student Presentation
2019. 05. 21.

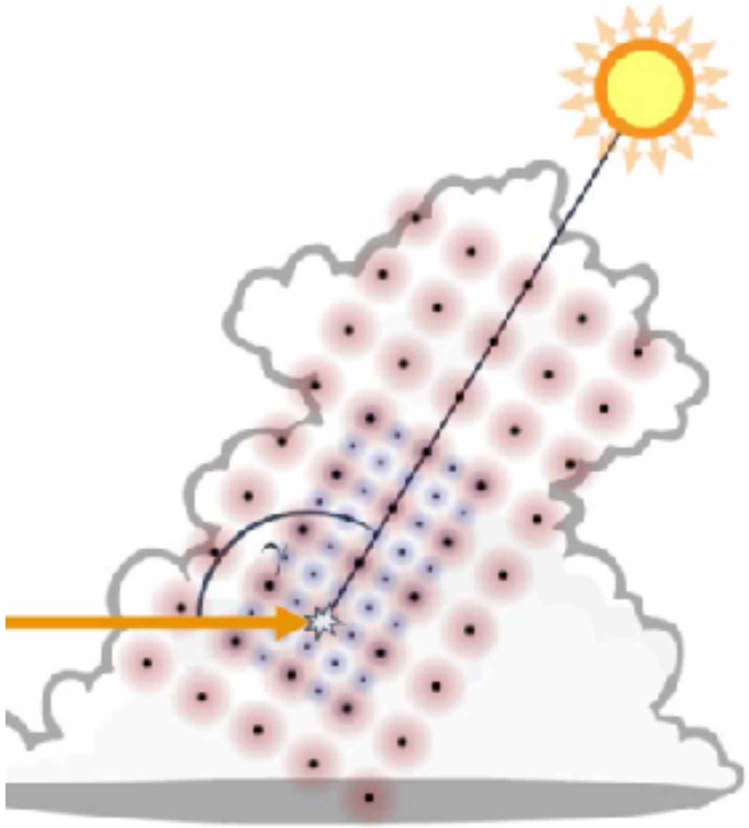
20193001 Dahyun Kang

Table of Contents

Review

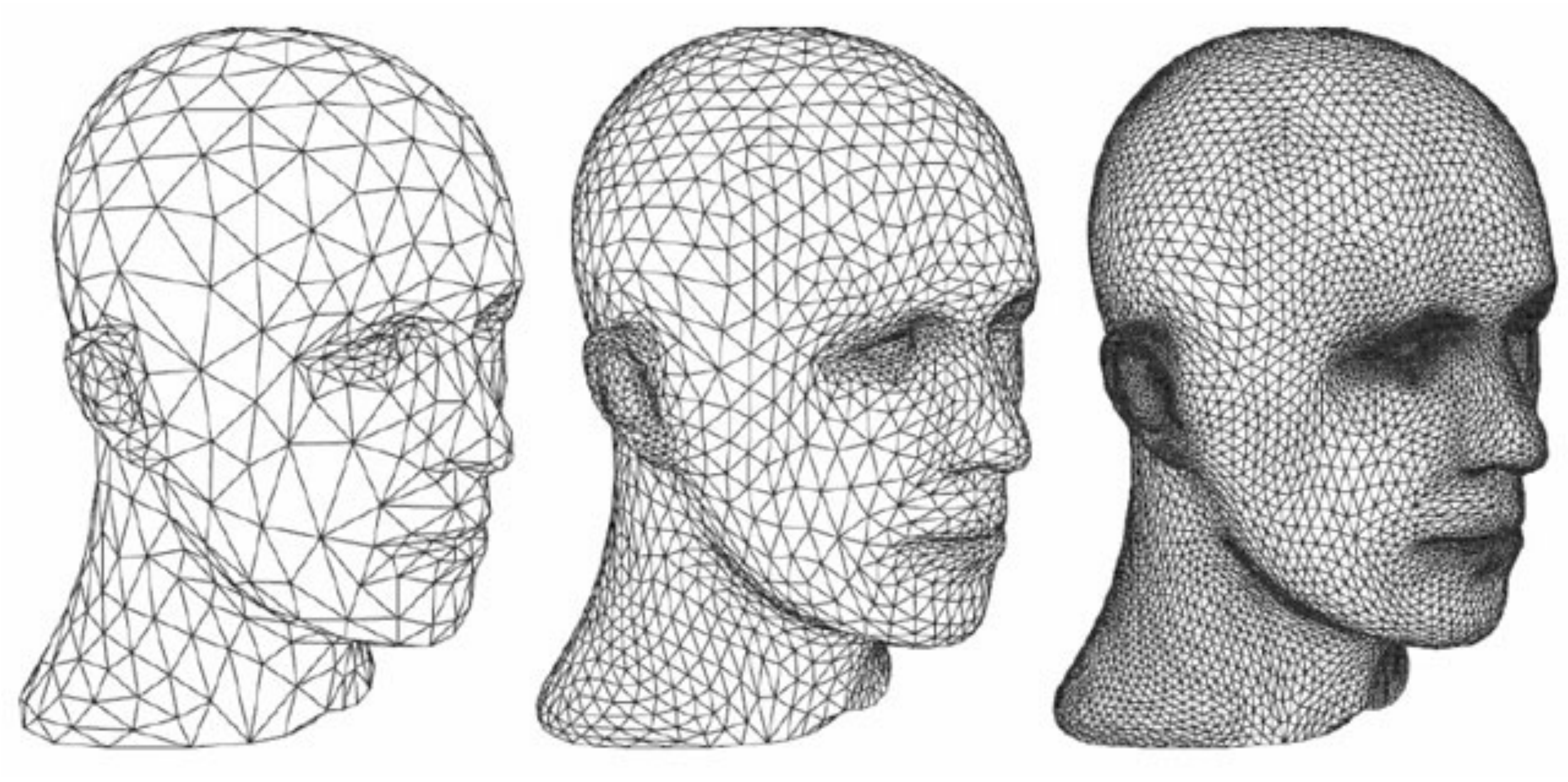
<Learning-based rendering> by Jaeyoon Kim

- Use hierarchical stencil for sampling and learn it to predict radiance
- Use light clustering and Bayesian online regression to reduce noise in adaptive direct illumination sampling.



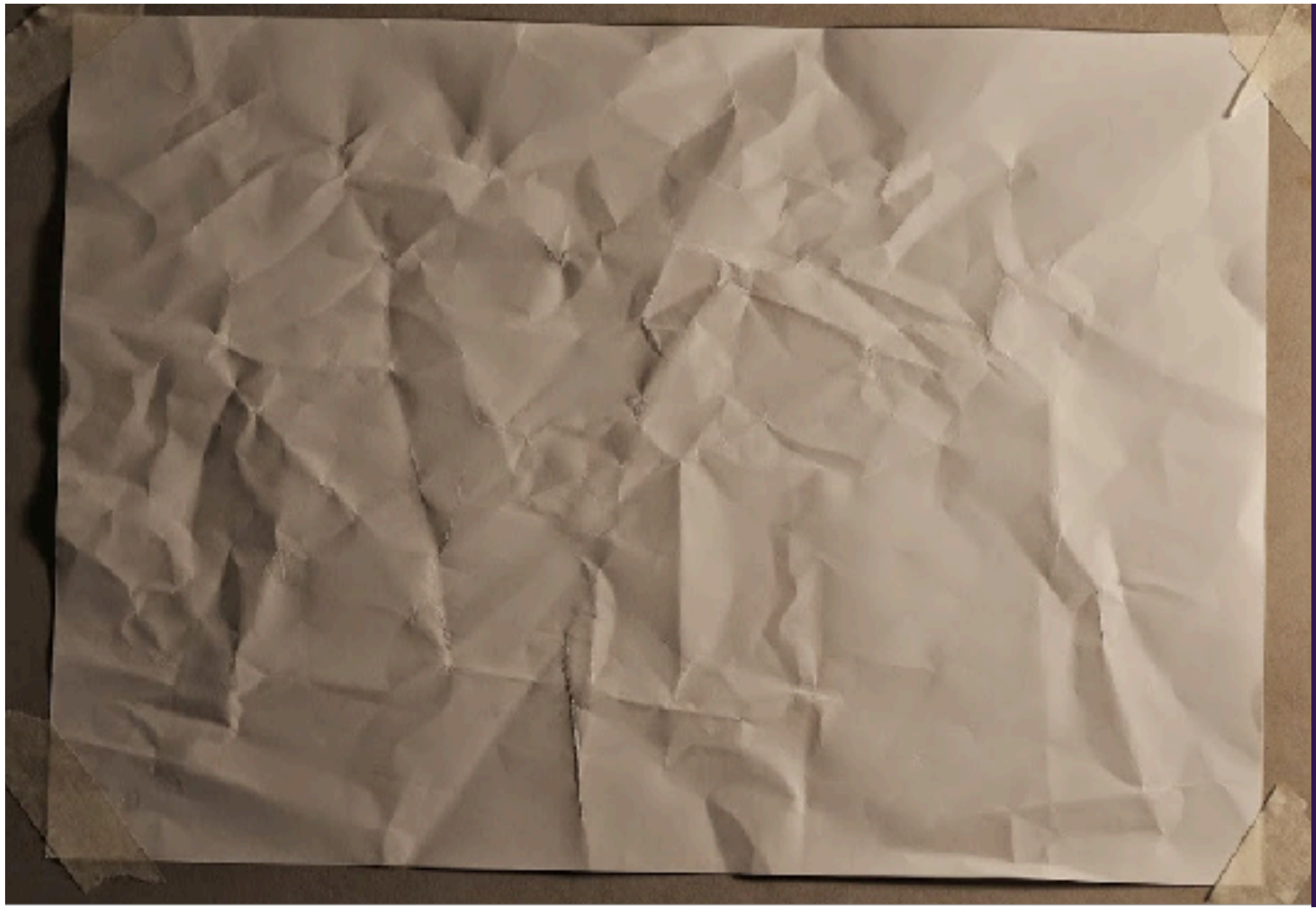
Overview

- Visual detail / richness / quality ..
  Increasing them is costly.



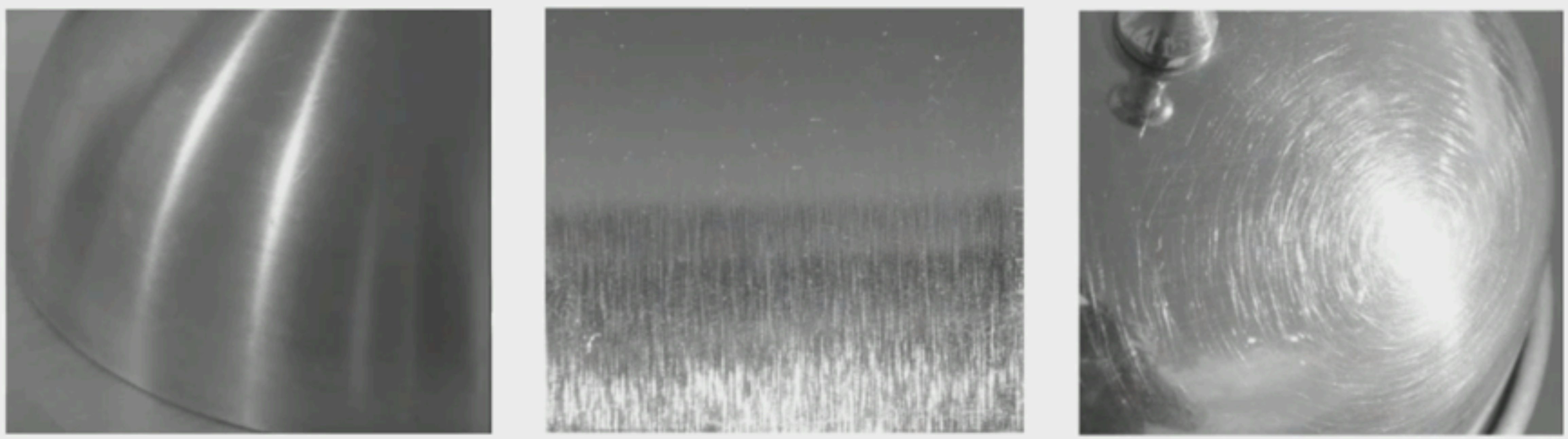
Overview

- Visual detail / richness / quality ..
  Increasing them is costly.



Overview

- Visual detail / richness / quality ..
  Increasing them is costly.

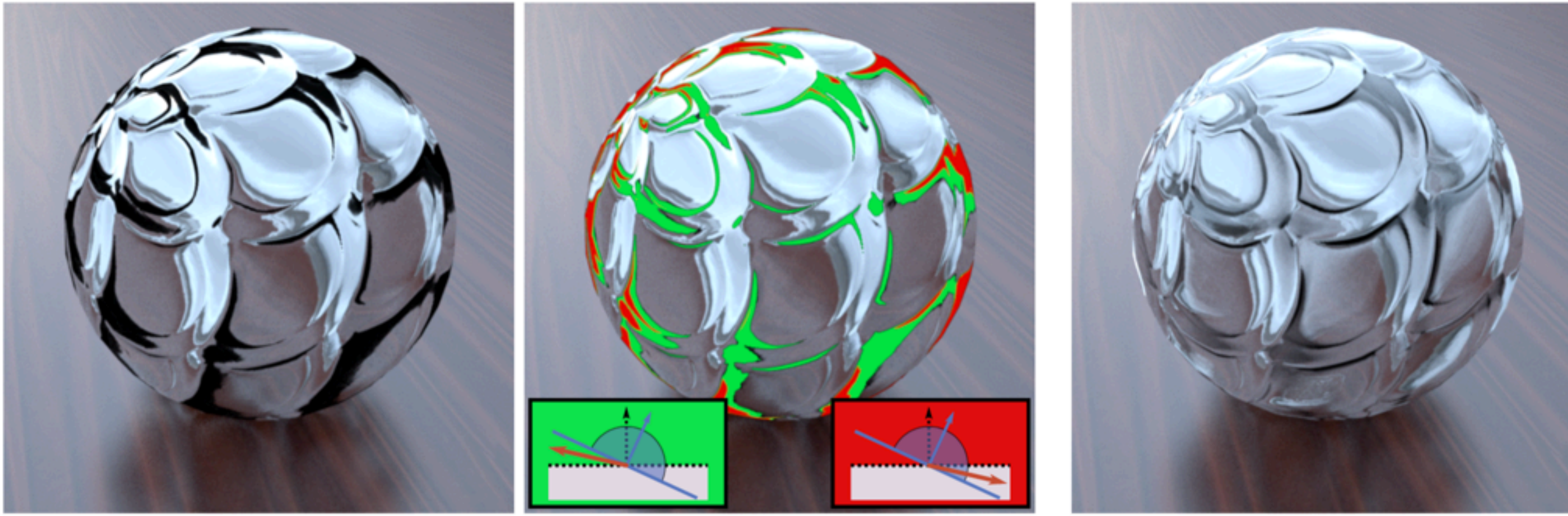


Smeared reflections *Patterned highlights* *Glint lines*

Overview

- Visual detail / richness / quality ..
  Increasing them is costly.

- *Cheaper techniques(tricks?)* should be used in practice.
- Today I bring two examples of them,
  especially **micro-scale BRDF related** issues:
  1. **Normal Mapping**
     Microfacet-based Normal Mapping for Robust Monte Carlo Path Tracing
  2. **Scratched Materials**
     Multi-Scale Rendering of Scratched Materials using a Structured SV-BRDF Model

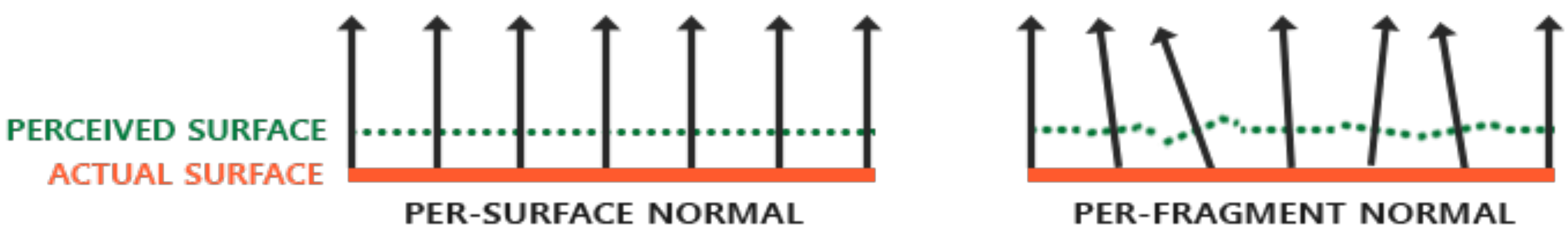


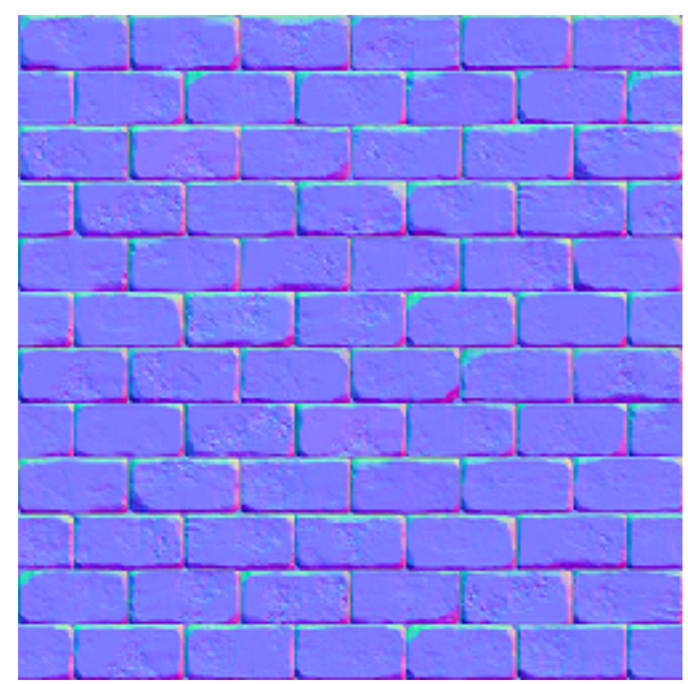
Microfacet-based Normal Mapping for Robust Monte Carlo Path Tracing

Vincent Schüssler et al.
SIGGRAPH Asia 2017

What is Normal Mapping?

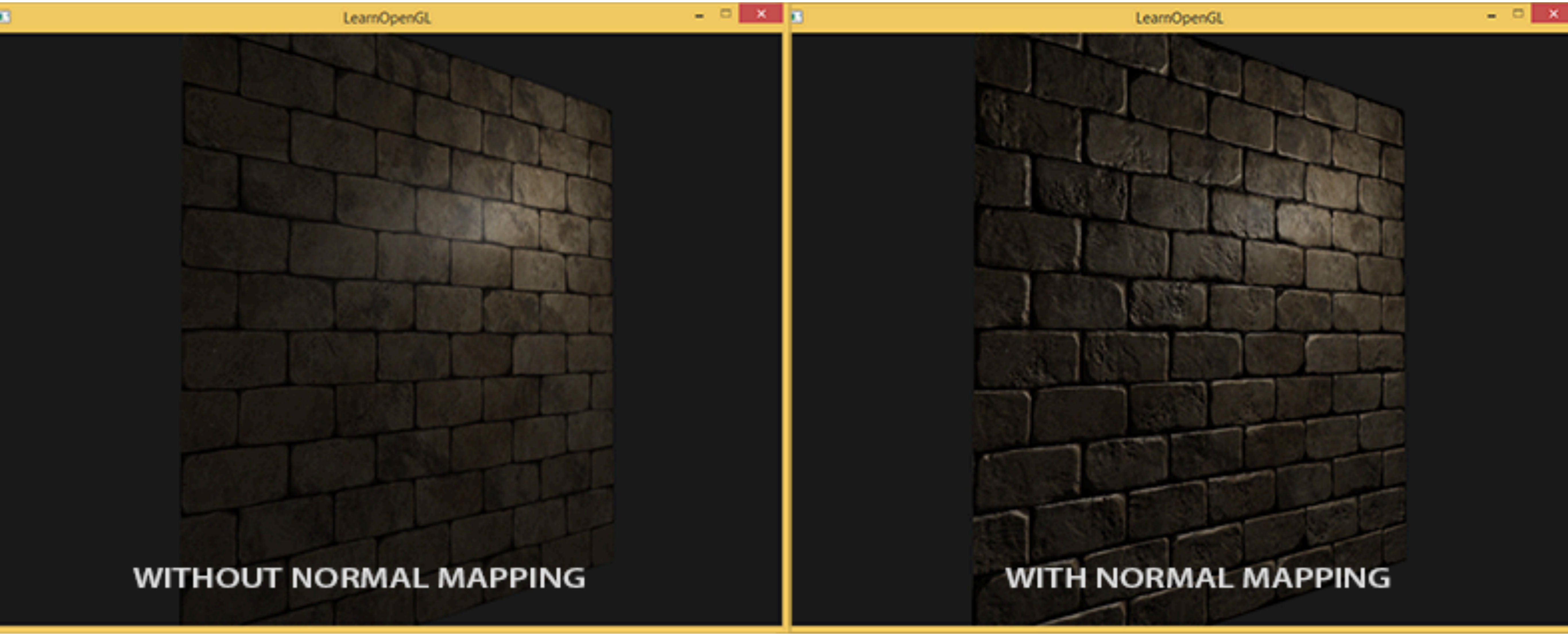
- Surface normal is critical in shading.
- Replace true geometric normal with new normal given by user which is designed to fake surface appearance details.





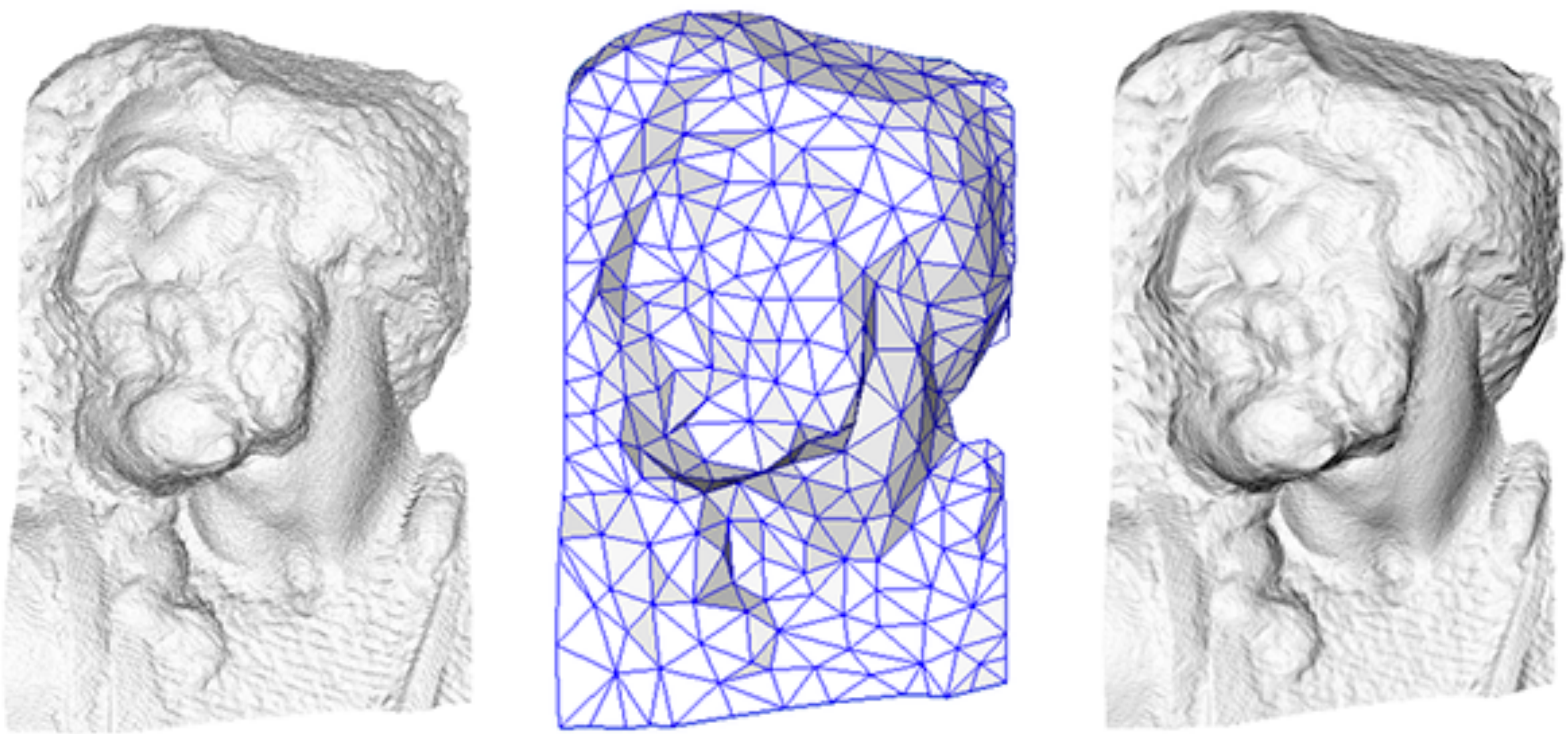
What is Normal Mapping?

- Unlike 'texture map',
  normal mapped surface varies as light moves.



What is Normal Mapping?

- Frequently used in practical modeling thanks to simplicity.



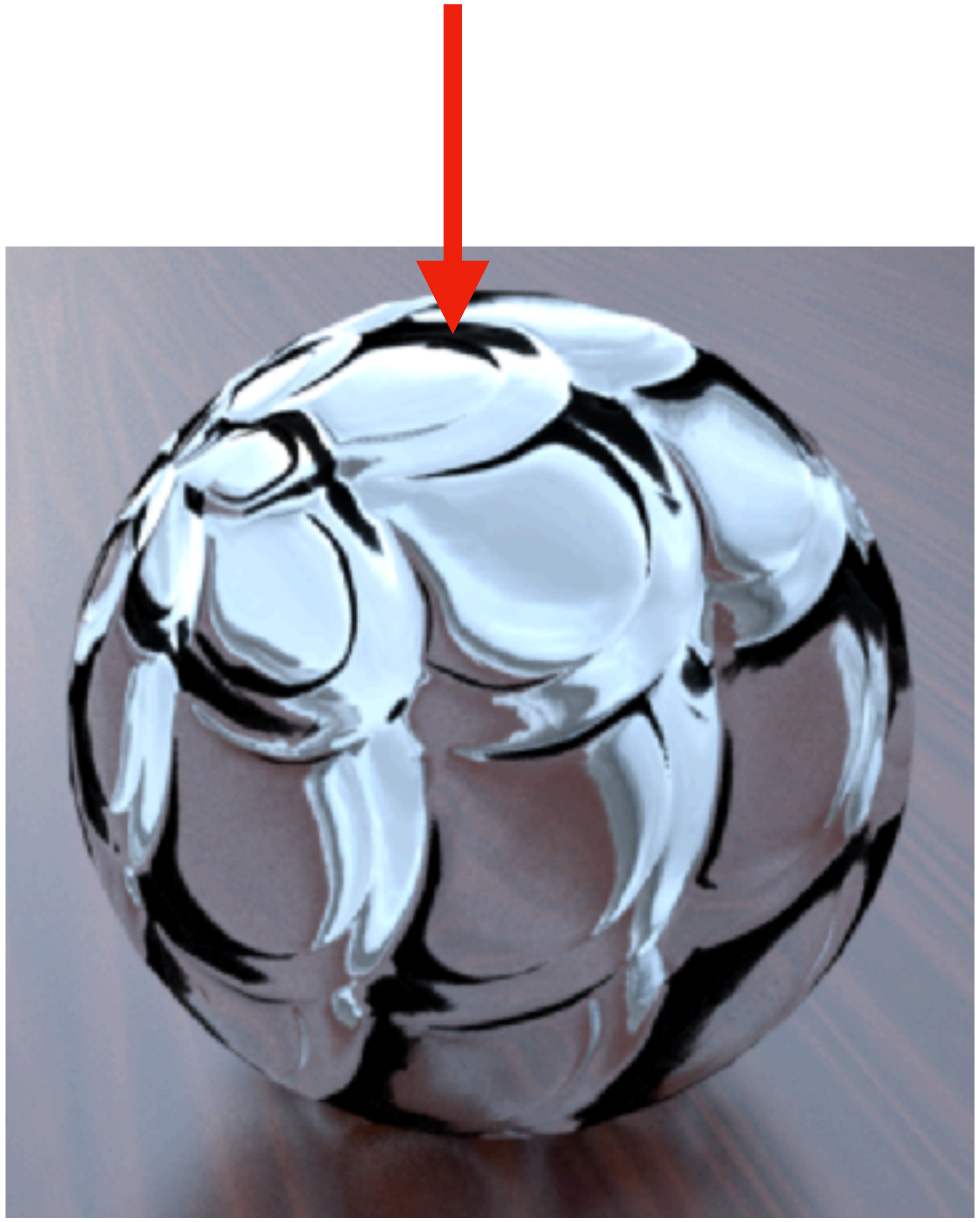
original mesh
4M triangles

simplified mesh
500 triangles

simplified mesh
and normal mapping
500 triangles

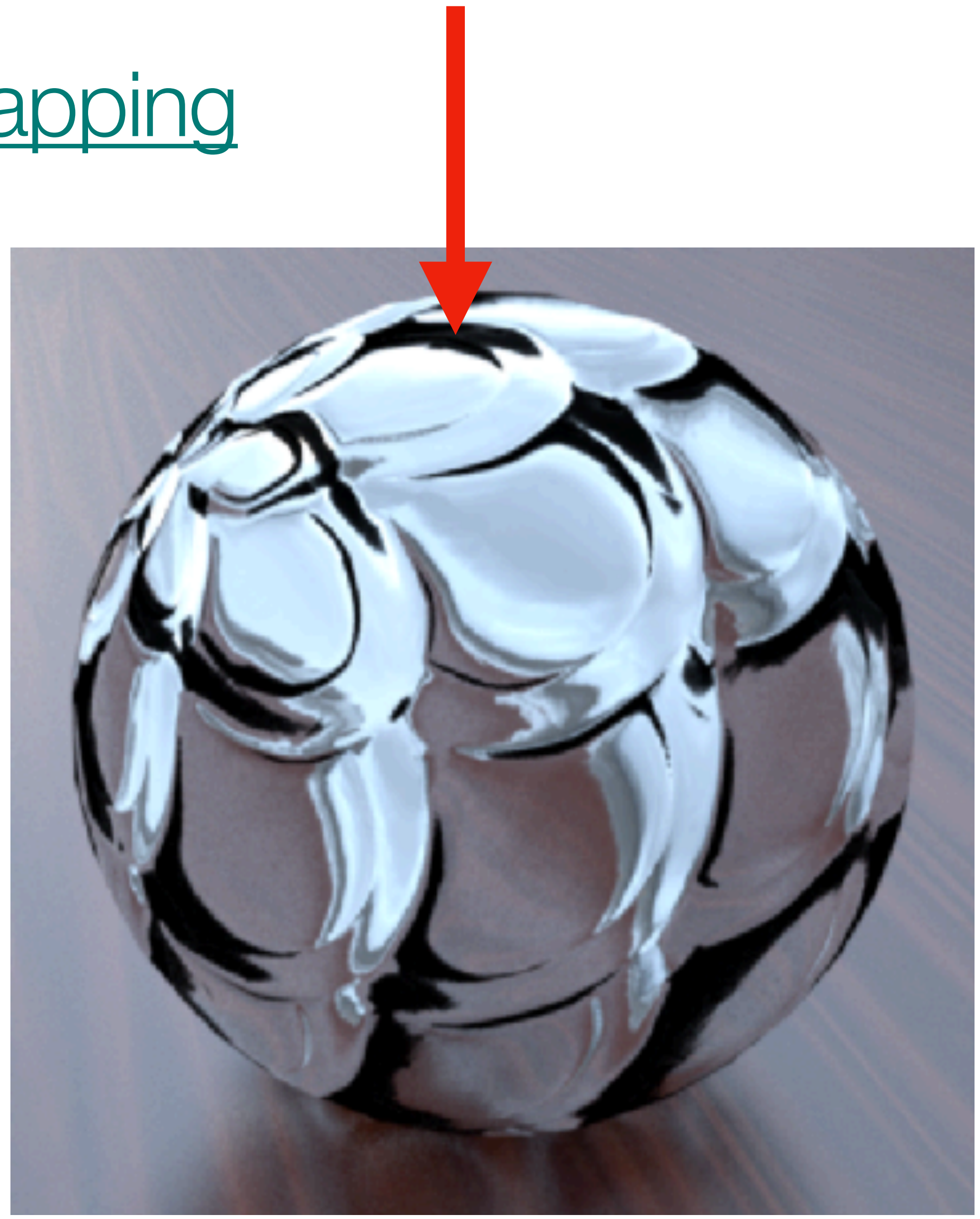
Problem of Normal Mapping

- It is something FAKING in principle and violates Physics!
- Thus PBRTs such as Monte Carlo Ray Tracing fails.



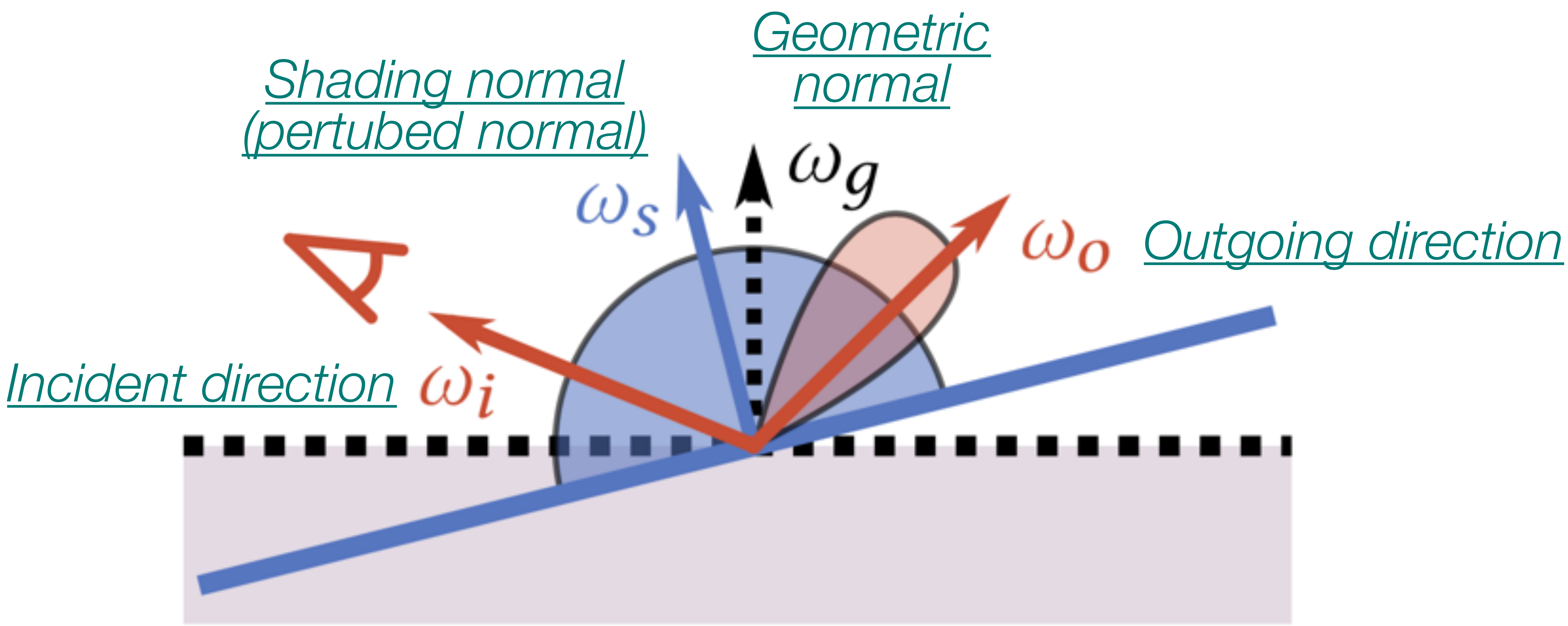
Problem of Normal Mapping

- It is something FAKING in principle and violates Physics!
- Thus PBRTs such as Monte Carlo Ray Tracing fails.
- This paper want to model normal mapping in physically correct manner so that normal mapping can be used in PBRT.
- To do that, this paper adopts **microfacet theory**.



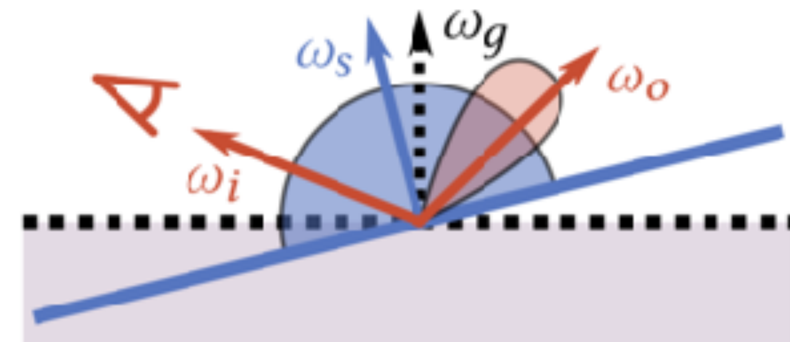
Problem of Normal Mapping

- Normal mapping tilts the BRDF hemisphere:



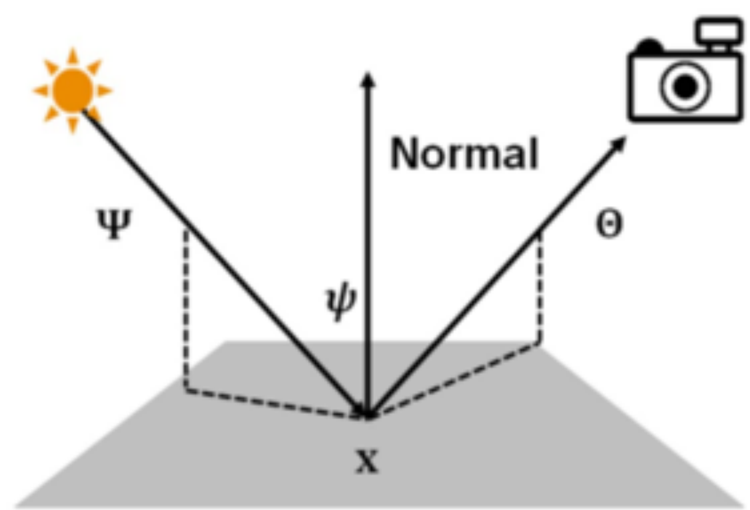
Problem of Normal Mapping

1. <u>Non-symmetry</u>



Assume we have BRDF f_{ω_s} evaluated w.r.t. the shading normal (instead of the geometric normal).

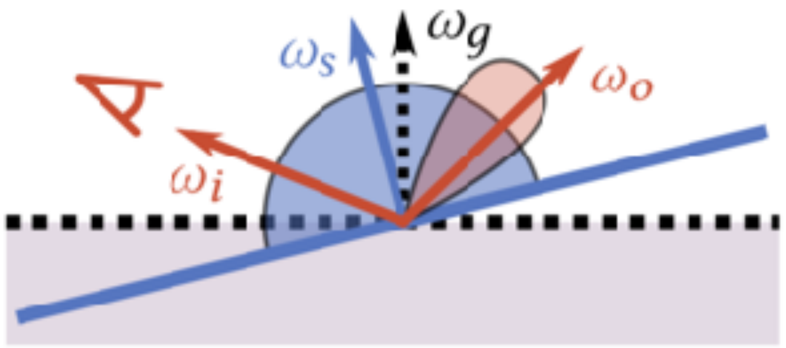
Definition of BRDF from the lecture slide



$$f_r(x, \Psi \to \Theta) = \frac{dL(x \to \Theta)}{dE(x \leftarrow \Psi)} = \frac{dL(x \to \Theta)}{L(x \leftarrow \Psi)\cos\psi d\omega_\Psi}$$

Problem of Normal Mapping

1. <u>Non-symmetry</u>



Assume we have BRDF f_{ω_s} evaluated w.r.t. the shading normal (instead of the geometric normal).

Because our integrator evaluates w.r.t. the geometric normal, we should modify the BRDF by:

$$\bar{f}(\omega_i, \omega_o) = f_{\omega_s}(\omega_i, \omega_o) \frac{\langle \omega_o, \omega_s \rangle}{\langle \omega_o, \omega_g \rangle}$$

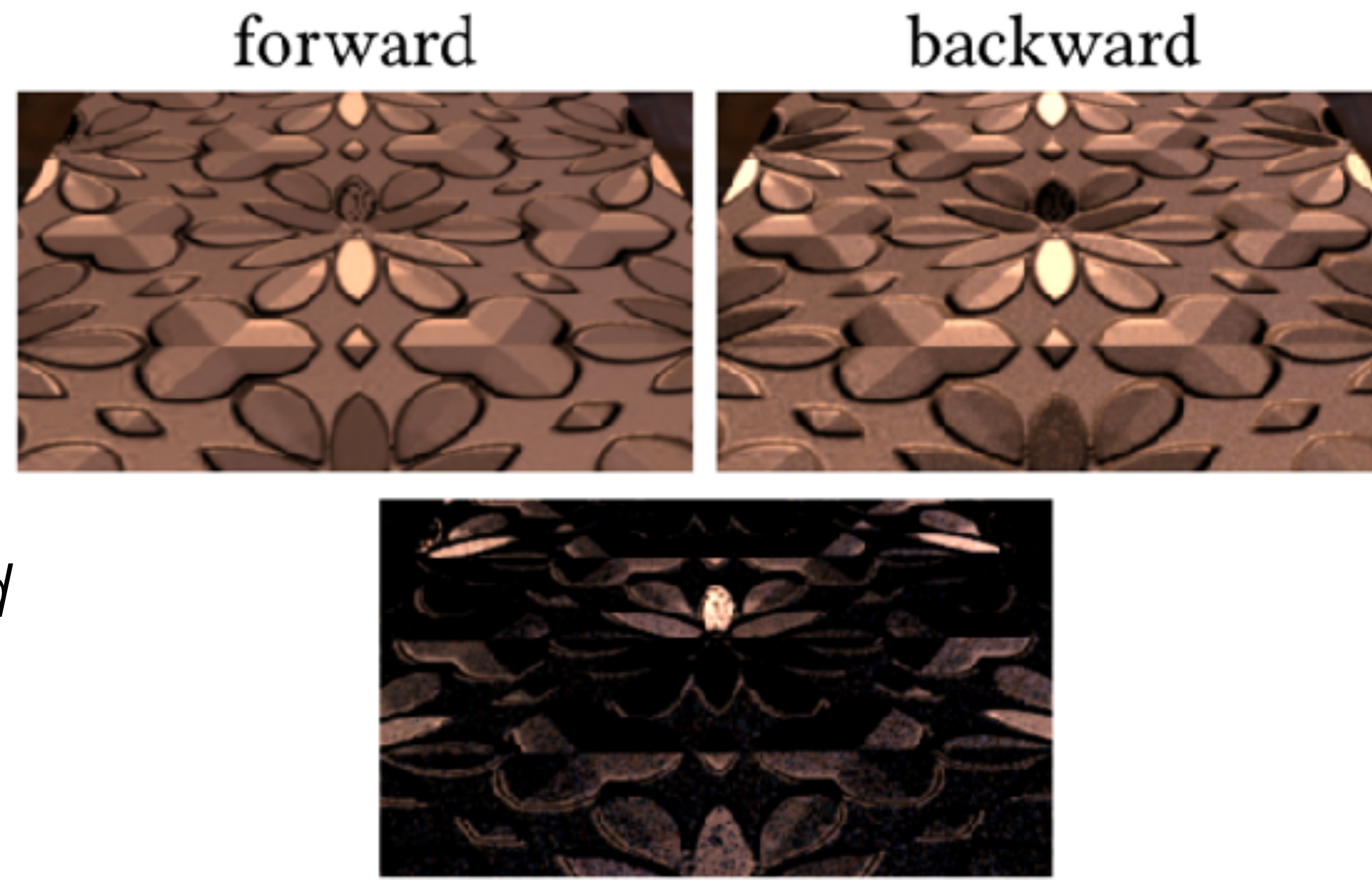
Problem of Normal Mapping

1. <u>Non-symmetry</u>

Modified BRDF is
not symmetric

→ *forward and backward path tracing differ*

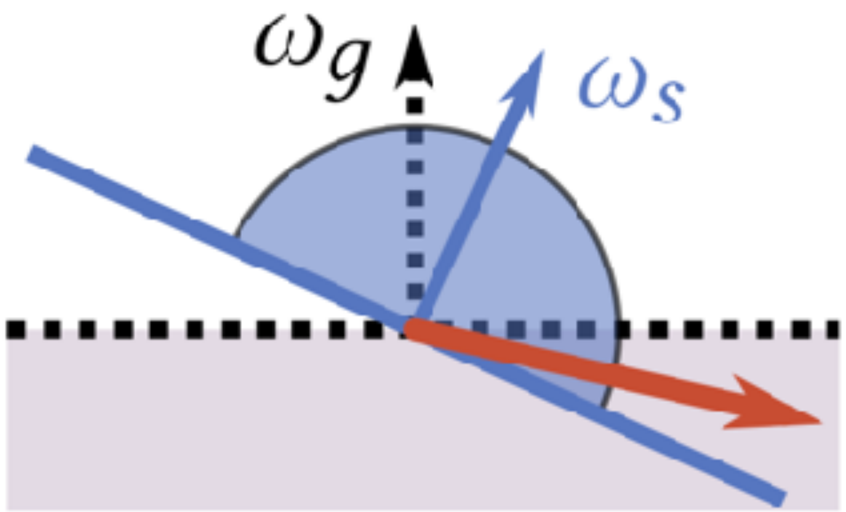
→ *cannot be used with bidirectional path tracing*



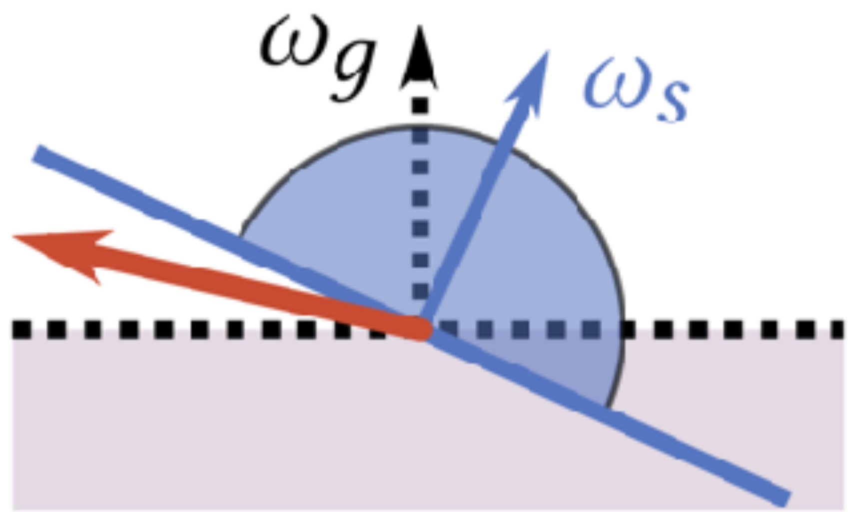
$$\bar{f}(\omega_i, \omega_o) = f_{\omega_s}(\omega_i, \omega_o) \frac{\langle \omega_o, \omega_s \rangle}{\langle \omega_o, \omega_g \rangle}$$

Problem of Normal Mapping

2. <u>Loss of energy and black fringes</u>



Light can leak through the surface



BRDF is undefined for directions below the tilted hemisphere

Black fringes due to energy loss

Problem of Normal Mapping

3. <u>Violation of energy conservation</u>

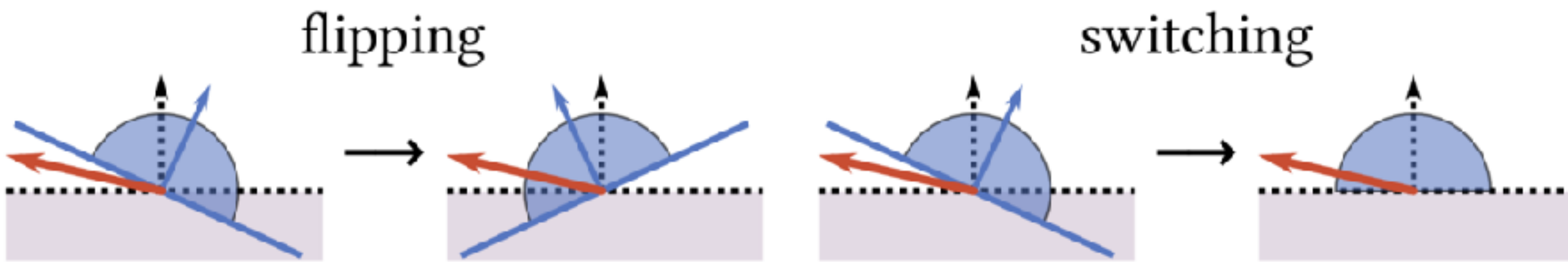
$$1 \not\geq \int_\Omega \bar{f}(\omega_i, \omega_o)(\omega_i \cdot \omega_g)\,\mathrm{d}\omega_i$$

$$= \frac{\omega_o \cdot \omega_s}{\omega_o \cdot \omega_g} \int_\Omega f_{\omega_s}(\omega_i, \omega_o)(\omega_i \cdot \omega_g)\,\mathrm{d}\omega_i$$

This can be arbitrarily large

Problem of Normal Mapping

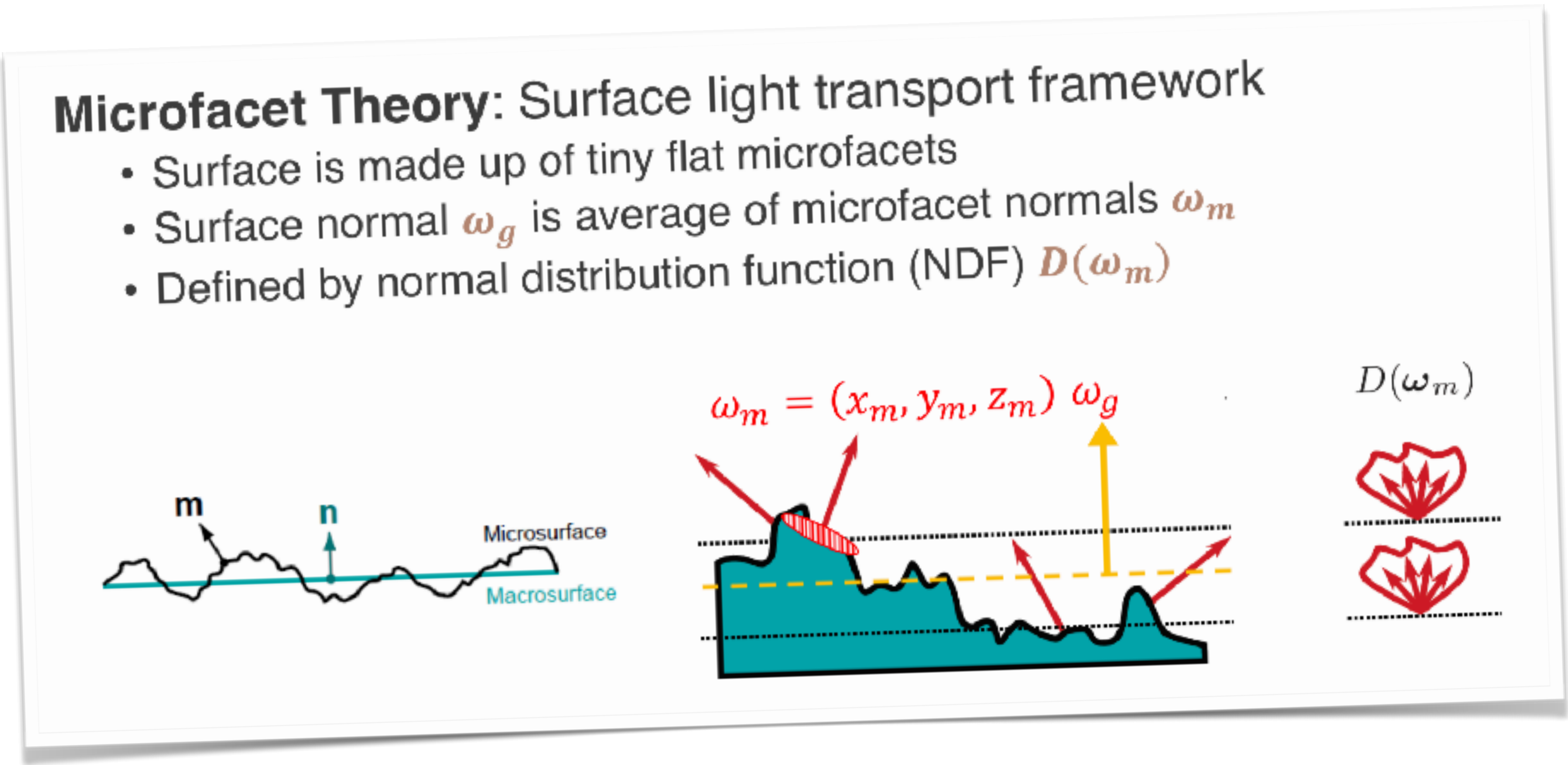
- Classic techniques to prevent undefined directions:



- .. Of course, even worsen the issues.

Microfacet Theory

- We have learnt a lot about this :)



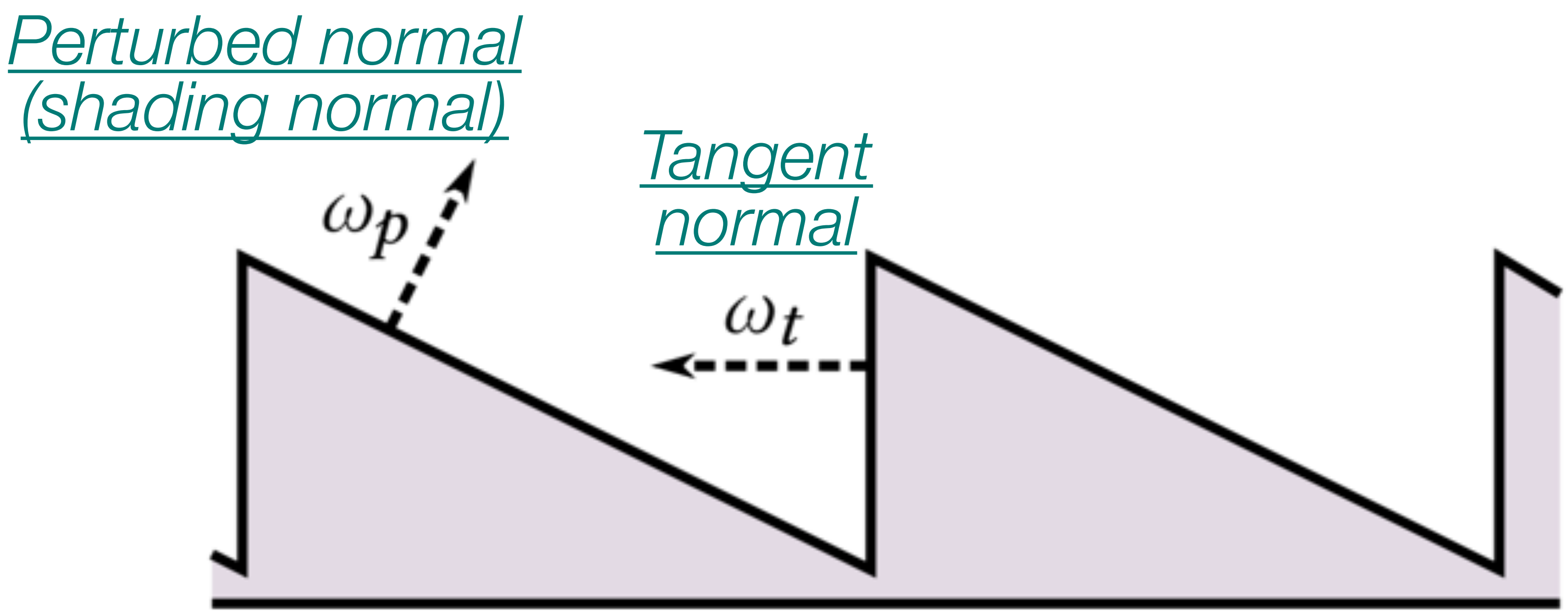
Microfacet Theory

- What they did using the microfacet theory?

1. Model(design) microsurfaces for normal mapping.
2. Evaluate BRDF of that microsurface.

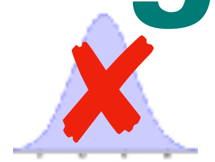
Modeling Microsurface

- Add tangent facet that compensates for the perturbed normal such that the average normal of the microsurface remains the geometric normal.



[1] Microfacet-based Normal Mapping

Modeling Microsurface

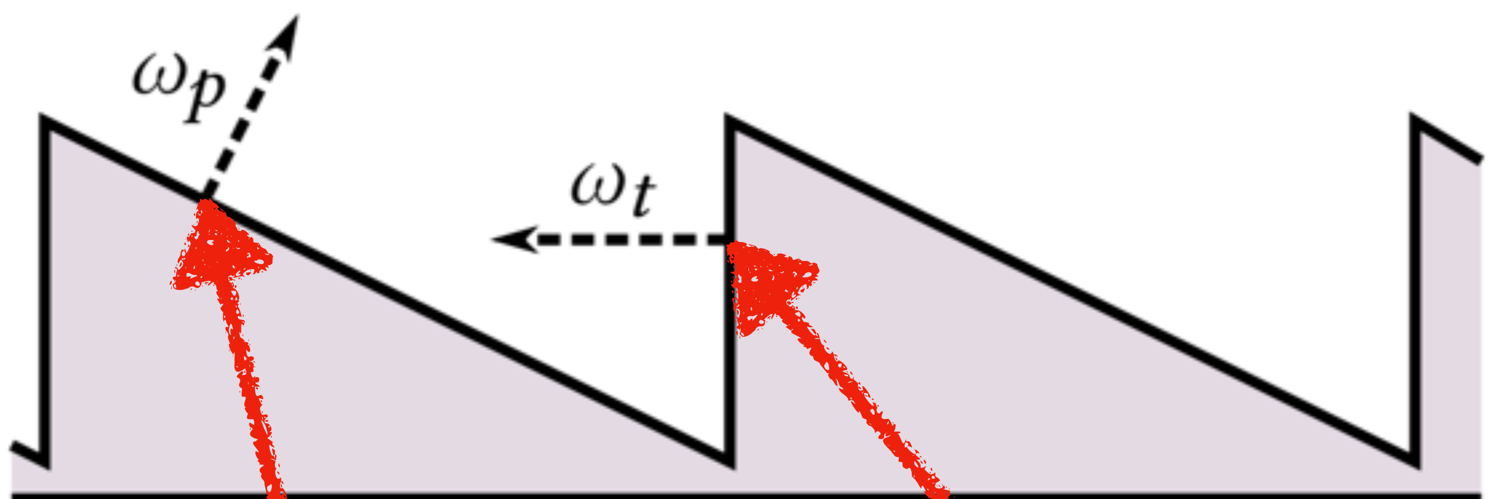


- *NDF(Distribution of Normal Function)* is designed to satisfy…

  i) Projected area = 1

  ii) Not smooth(discrete)… to avoid low-pass filter effect.

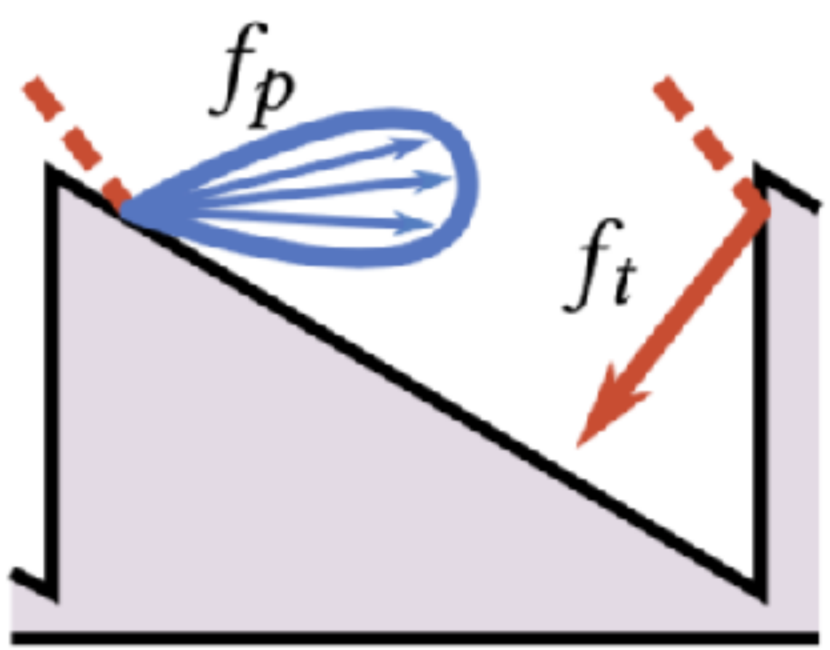
  iii) Maximizes the surface area with ω_p



$$D(\omega_m) = \frac{\delta_{\omega_p}(\omega_m)}{\omega_p \cdot \omega_g} + \frac{\delta_{\omega_t}(\omega_m)\sqrt{1-(\omega_p \cdot \omega_g)^2}}{\omega_p \cdot \omega_g}$$

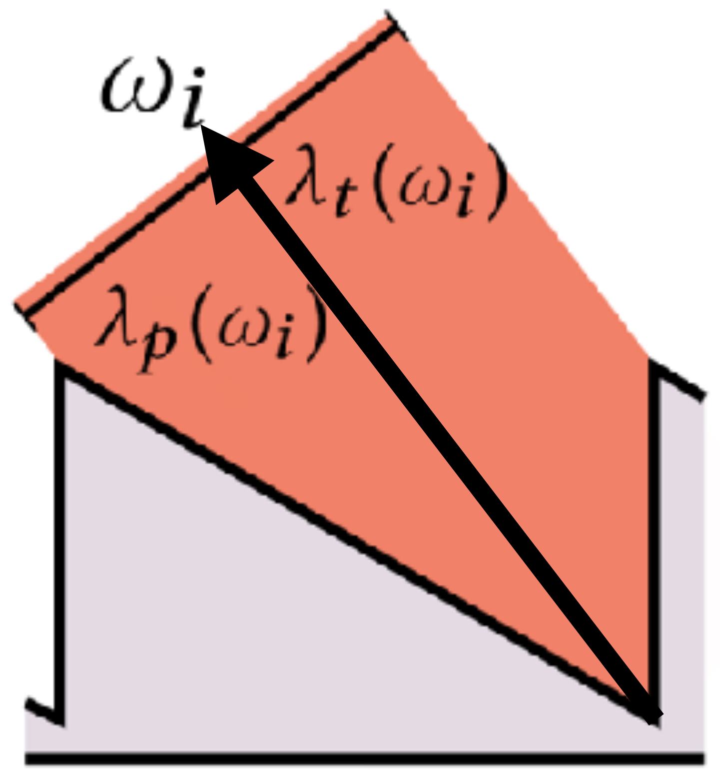
Evaluating BRDF

- Each microfacet has its own micro-BRDF:
  - f_p: microBRDF of perturbed facet
  - f_t: microBRDF of tangent facet
- BRDF of (macro)surface will be evaluated using those two.



Evaluating BRDF

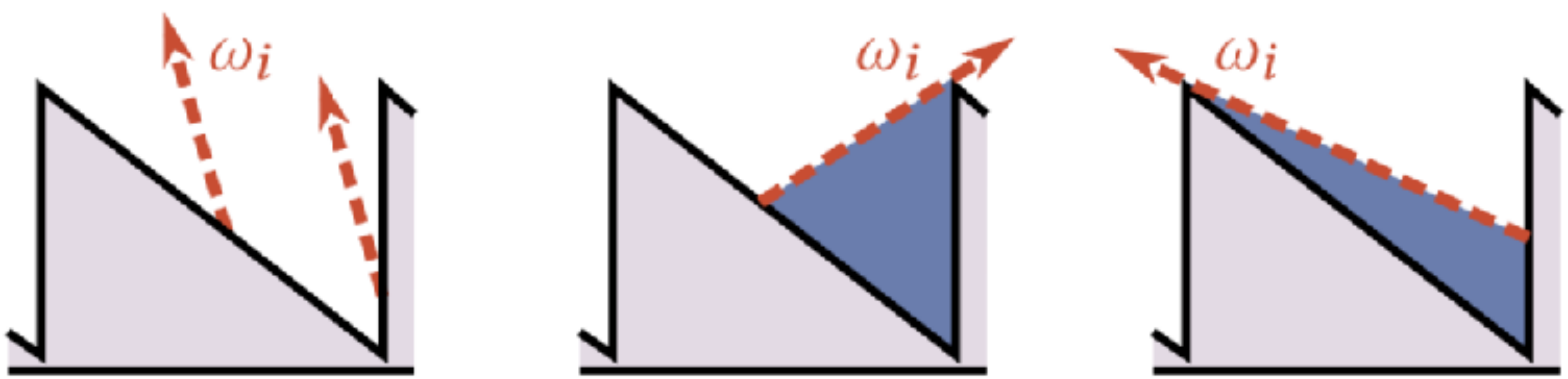
- Consider how probably of each facet is visible from incident ray direction ω_i



- Intuitively, consider **how much each facet will 'contributes' to the resulting appearance.**

Evaluating BRDF

- Consider how probably of each facet is visible from incident ray direction ω_i



Of course, sometimes one may be occluded by the other. Even we formulate this situation mathematically using Masking-shadowing function.

- Intuitively, consider **how much each facet will 'contributes' to the resulting appearance.**

Evaluating BRDF

- Consider how probably of each facet is visible from incident ray direction ω_i

Analytic form of single scattering:

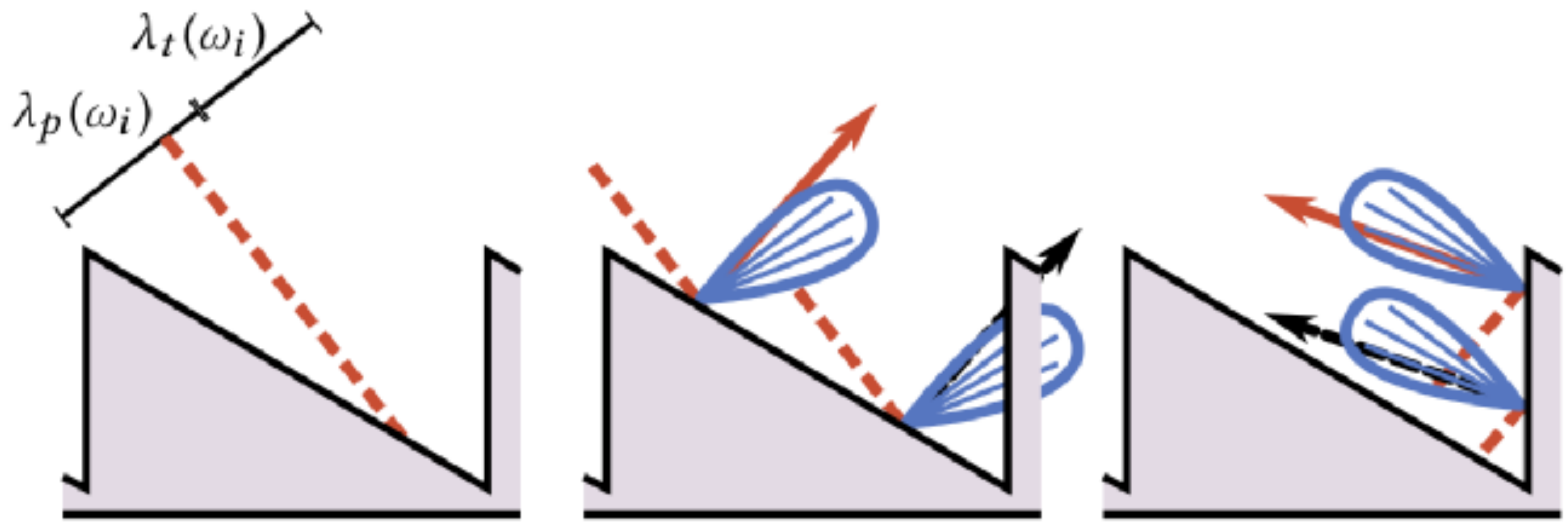
$$f_1(\omega_i, \omega_o)\langle\omega_o, \omega_g\rangle = \lambda_p(\omega_i)\, f_p(\omega_i, \omega_o)\langle\omega_o, \omega_p\rangle G_1(\omega_o, \omega_p) + \lambda_t(\omega_i)\, f_t(\omega_i, \omega_o)\langle\omega_o, \omega_t\rangle G_1(\omega_o, \omega_t),$$

- Intuitively, consider how much each facet will 'contributes' to the resulting appearance.

f_p: microBRDF of perturbed facet
f_t: microBRDF of tangent facet

Evaluating BRDF: Scattering Order

- Simulate multiple scattering using *Random Walk* algorithm.



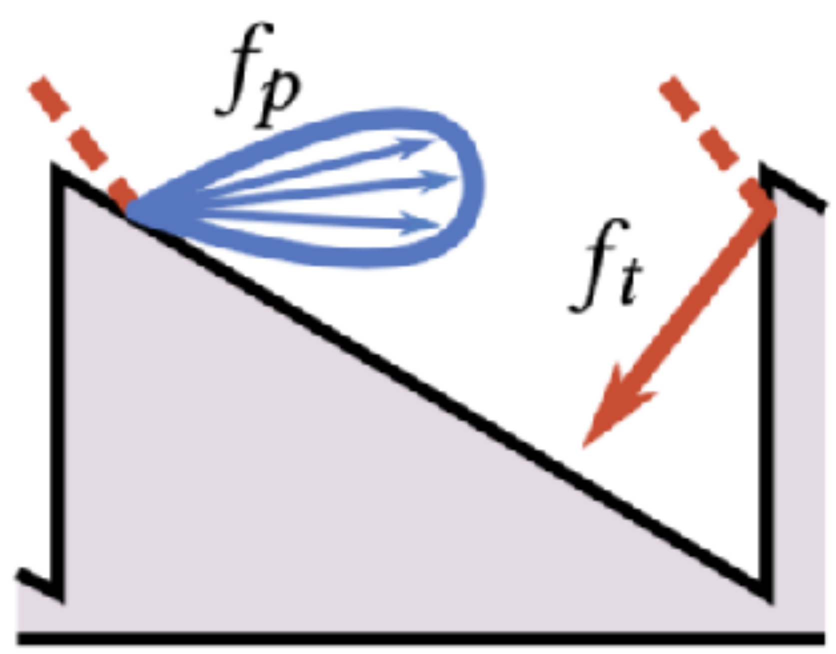
... *And so on*

- It gives an unbiased estimate of the cosine-weighted multiple-scattering BRDF $f_\infty(\omega_i, \omega_o)\,\langle\omega_g, \omega_o\rangle = E[L_o]$
- It is symmetric and energy conserving, (which is good).

Evaluating BRDF: Choosing f_t

Recall) Each microfacet has its own micro-BRDF

- f_p: microBRDF of perturbed facet. ← ***this is given by user***
- f_t: microBRDF of tangent facet. ← ***??? user never seen***



Evaluating BRDF: Choosing f_t

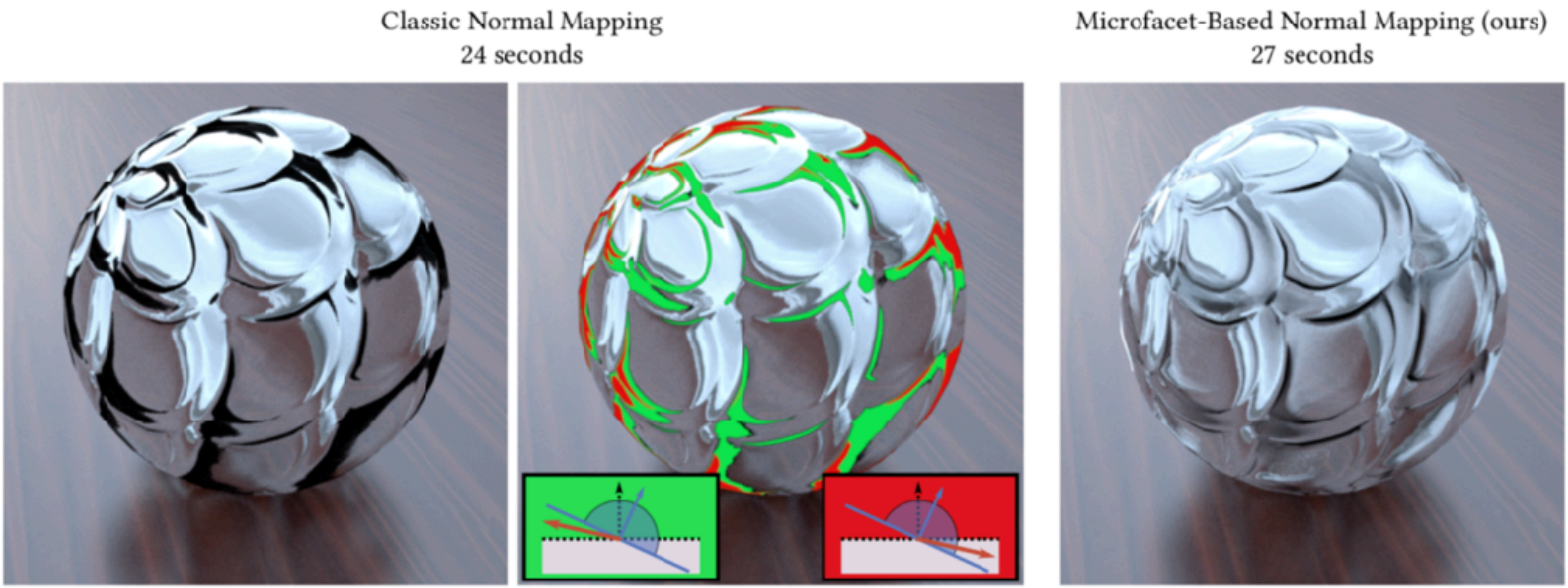
Recall) Each microfacet has its own micro-BRDF

- f_p: microBRDF of perturbed facet. ← ***this is given by user***
- f_t: microBRDF of tangent facet. ← ***??? user never seen***

- They provide three options for f_t:
  1. Same as f_p
  2. Diffuse
  3. Specular

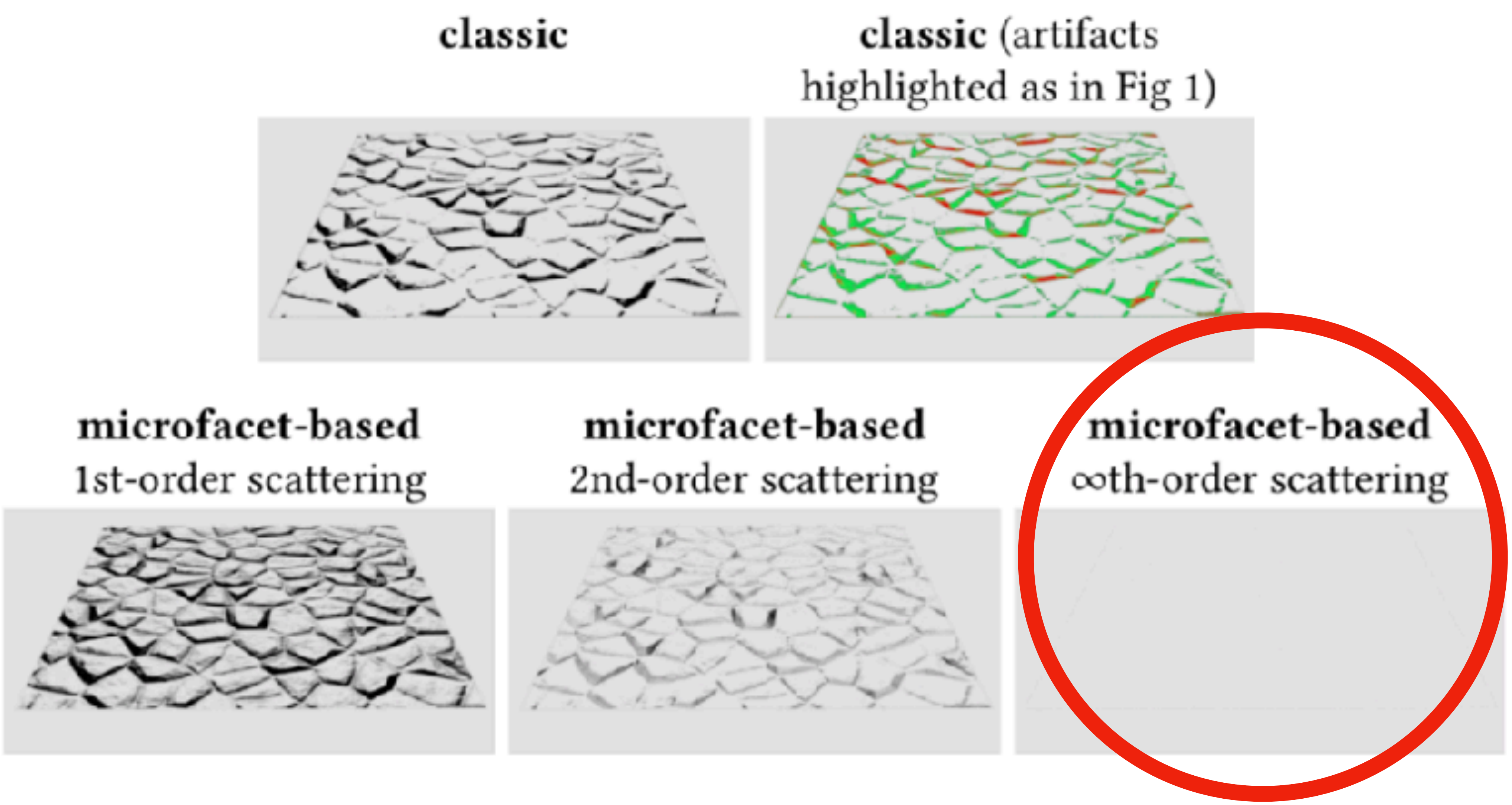
Result

- Resolved violation of energy conservation problem



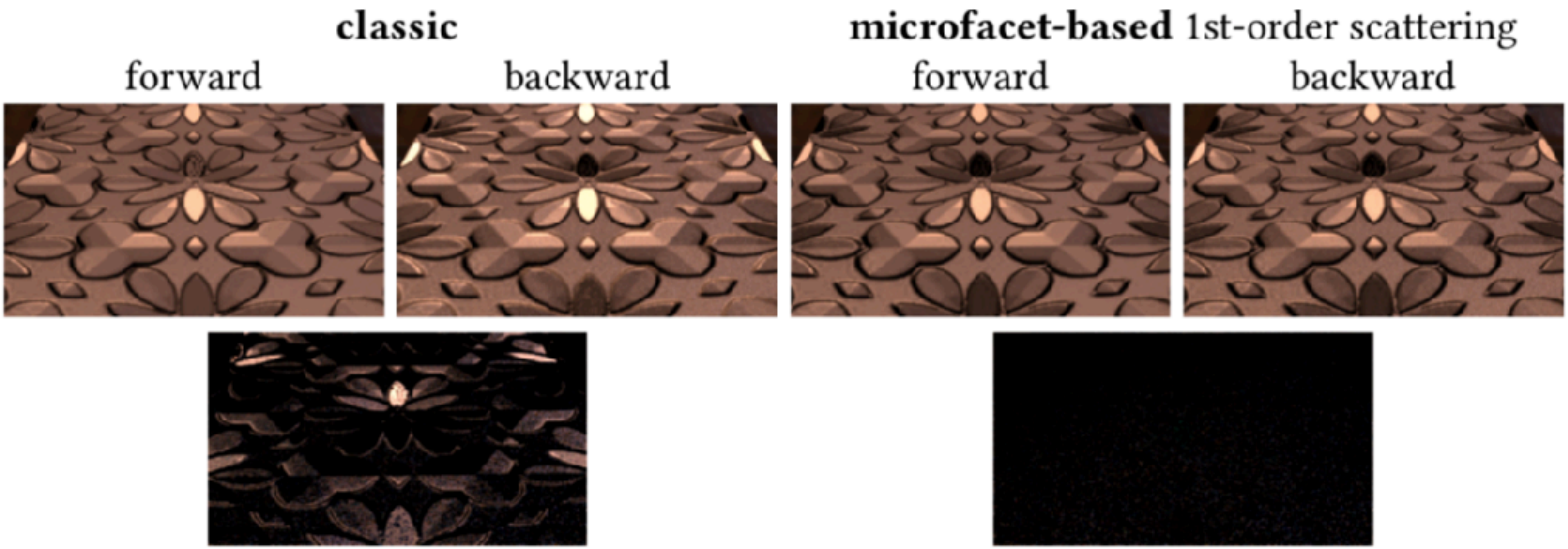
Results

- Resolved violation of energy conservation problem
- *White furnace test*: under white illumination, w/ 100% reflecting material, a scene should be white if energy is conserved.



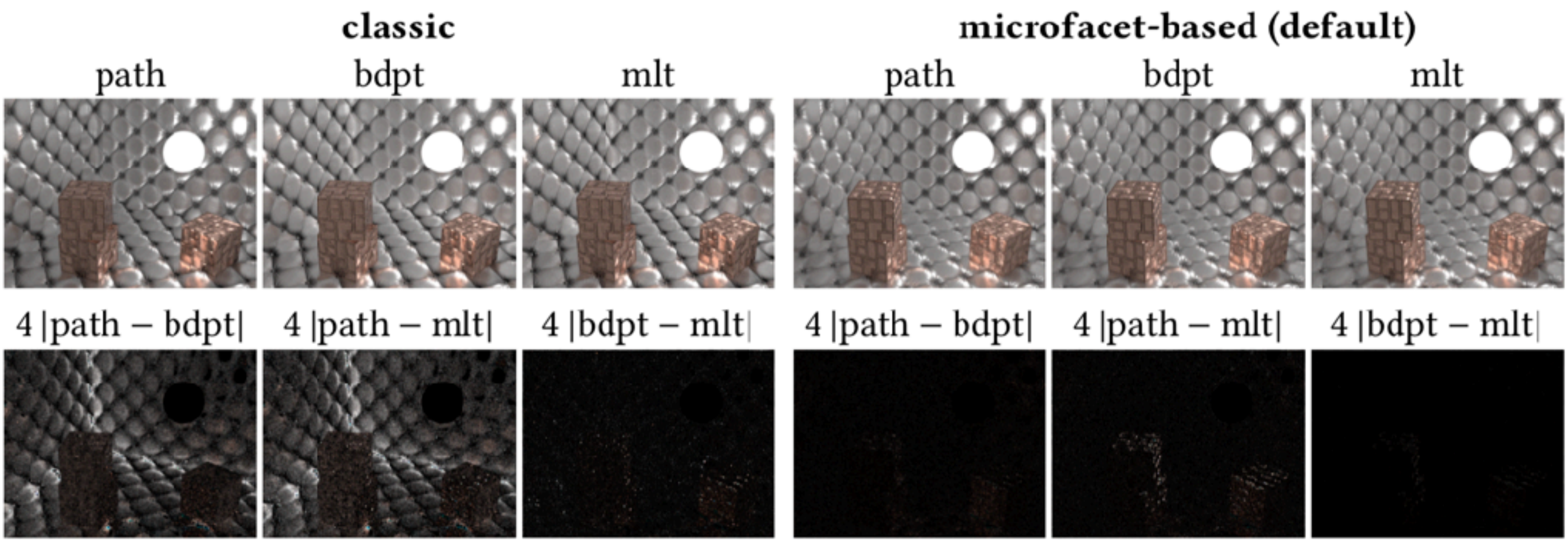
Result

- Resolved violation of symmetry of light transport problem



Result

- Resolved violation of symmetry of light transport problem
- Can be adopted into modern path tracing algorithms:



[1] Microfacet-based Normal Mapping

Result

- Performance: up to 70% more costly than the classic

classic			microfacet-based			
classic	switch (3.4)	flip (3.4)	same material tangent facet (6.1)		specular tangent facet (6.2)	
			2nd-order	∞th-order	2nd-order	∞th-order
29s	30s	30s	45s	49s	44s using Algo. 2 39s using Eq. (23)	47s

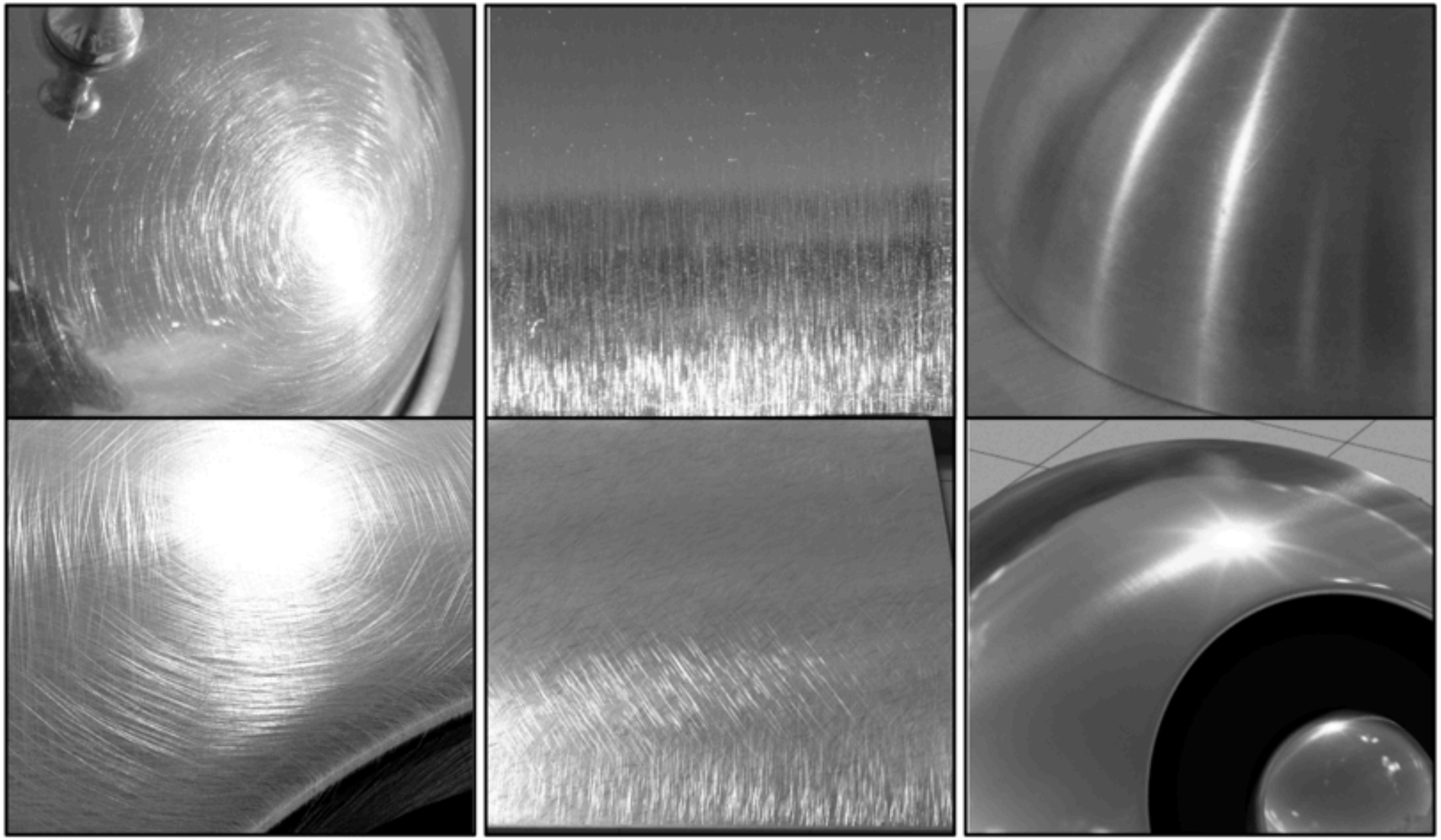
Multi-Scale Rendering of Scratched Materials using a Structured SV-BRDF Model

Boris Raymond et al.

SIGGRAPH 2016

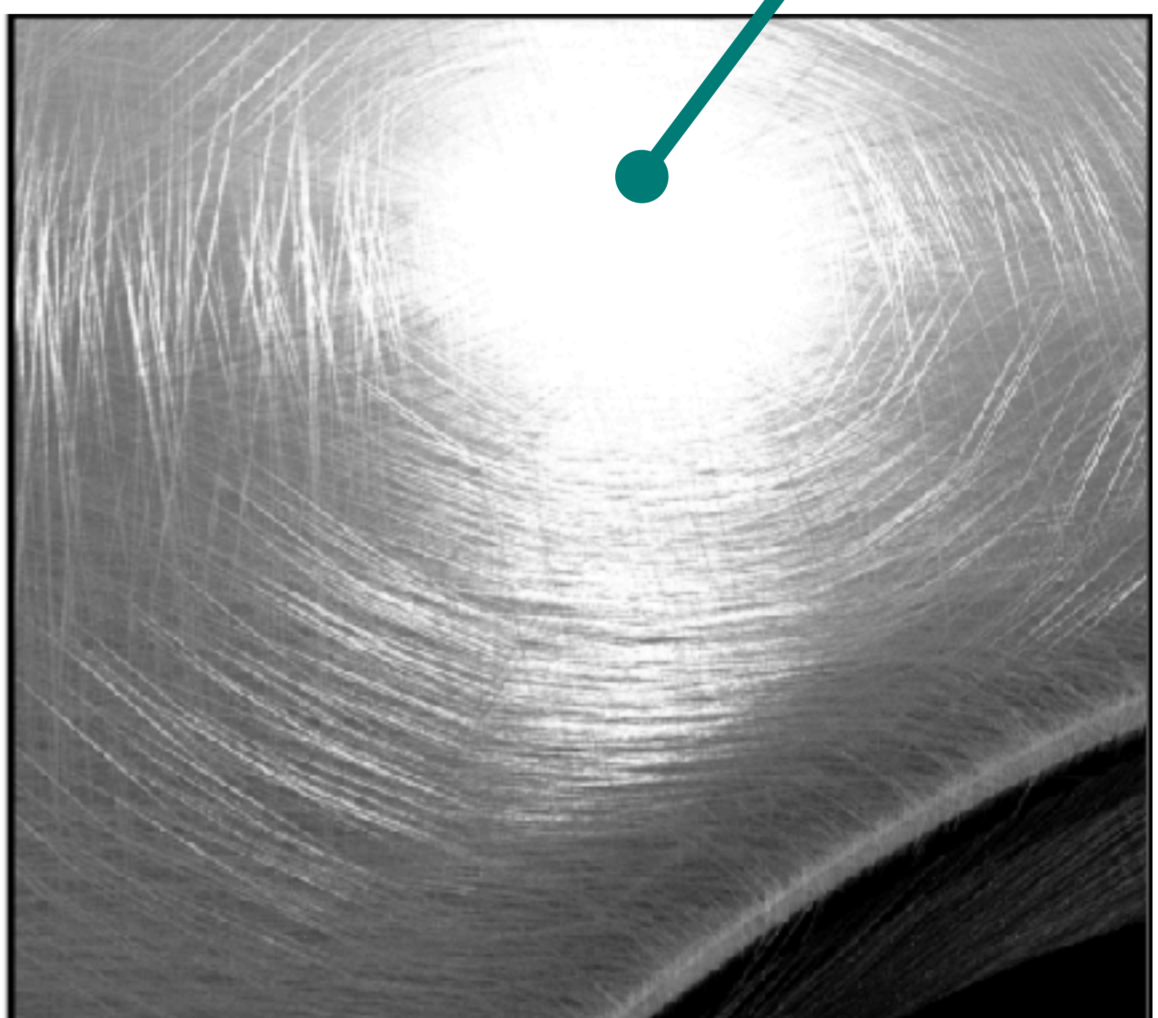
Scratched Materials

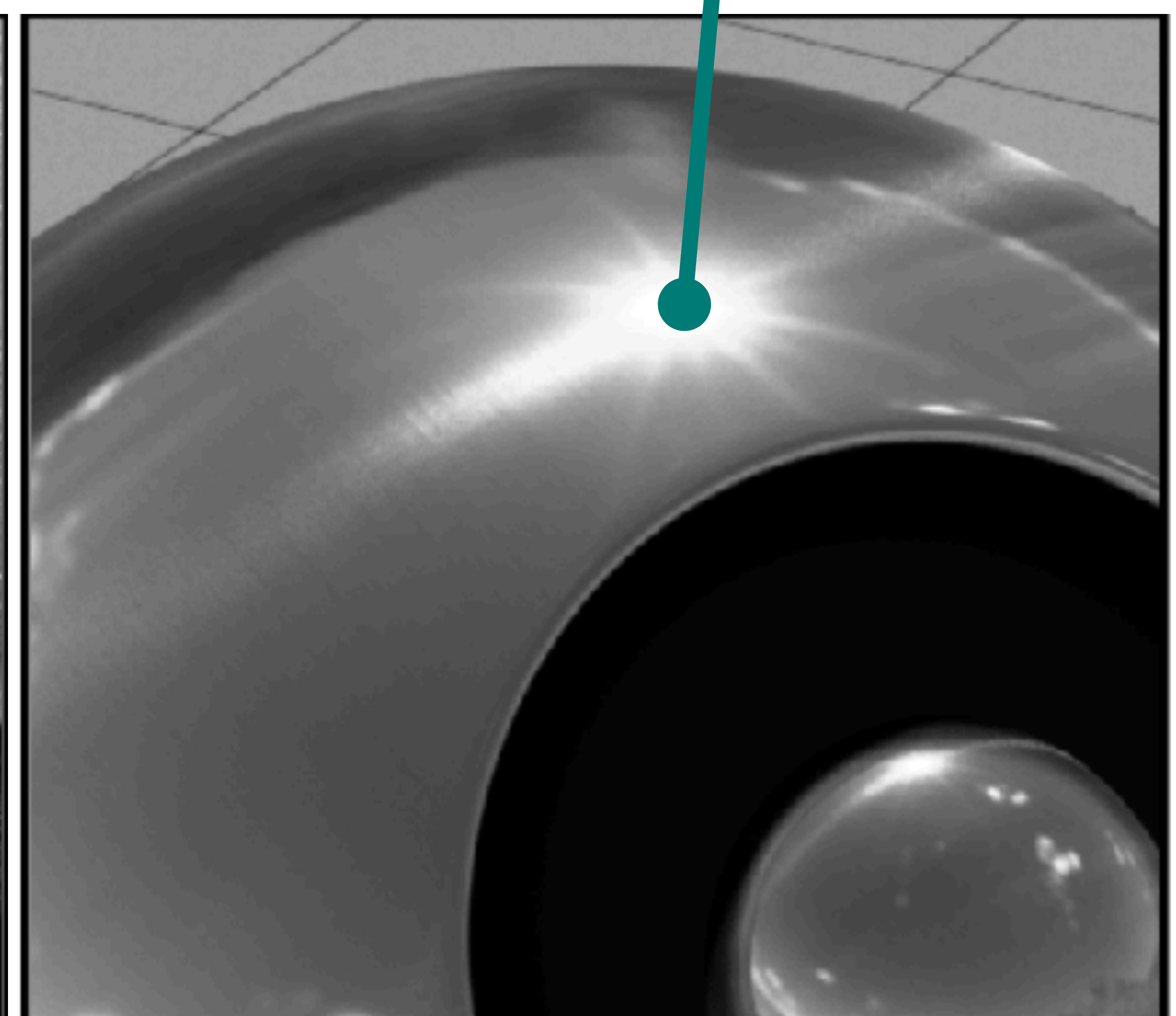
- Metals, plastics, finished woods, …



Scratched Materials

- **Affordable multi-scale approach** should be considered since scratch pattern is extremely high resolution but at the same time, also affects the appearance at farther distance.





Dimension Reduction

- Basically this is achievable by *SVBRDF(Spatially Varying-)*.

Dimension Reduction

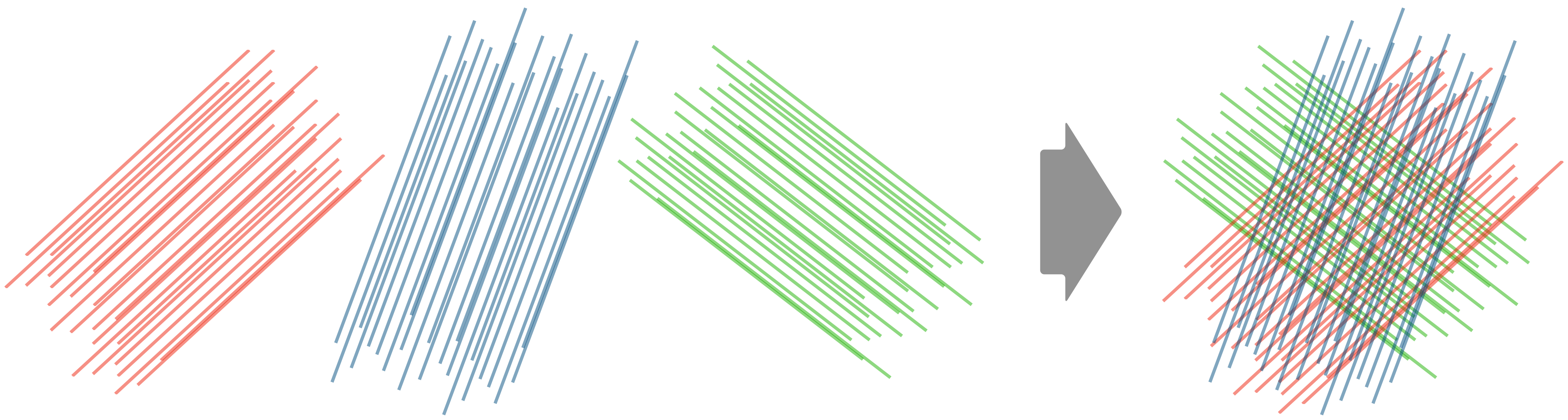
- Basically this is achievable by *SVBRDF(Spatially Varying-)*.

- … Which is super expensive 6D function.
- So they want less computation by **reducing dimension as much as possible, but still guaranteeing visual quality**.

- They make several convenient hypotheses to reduce dimension.

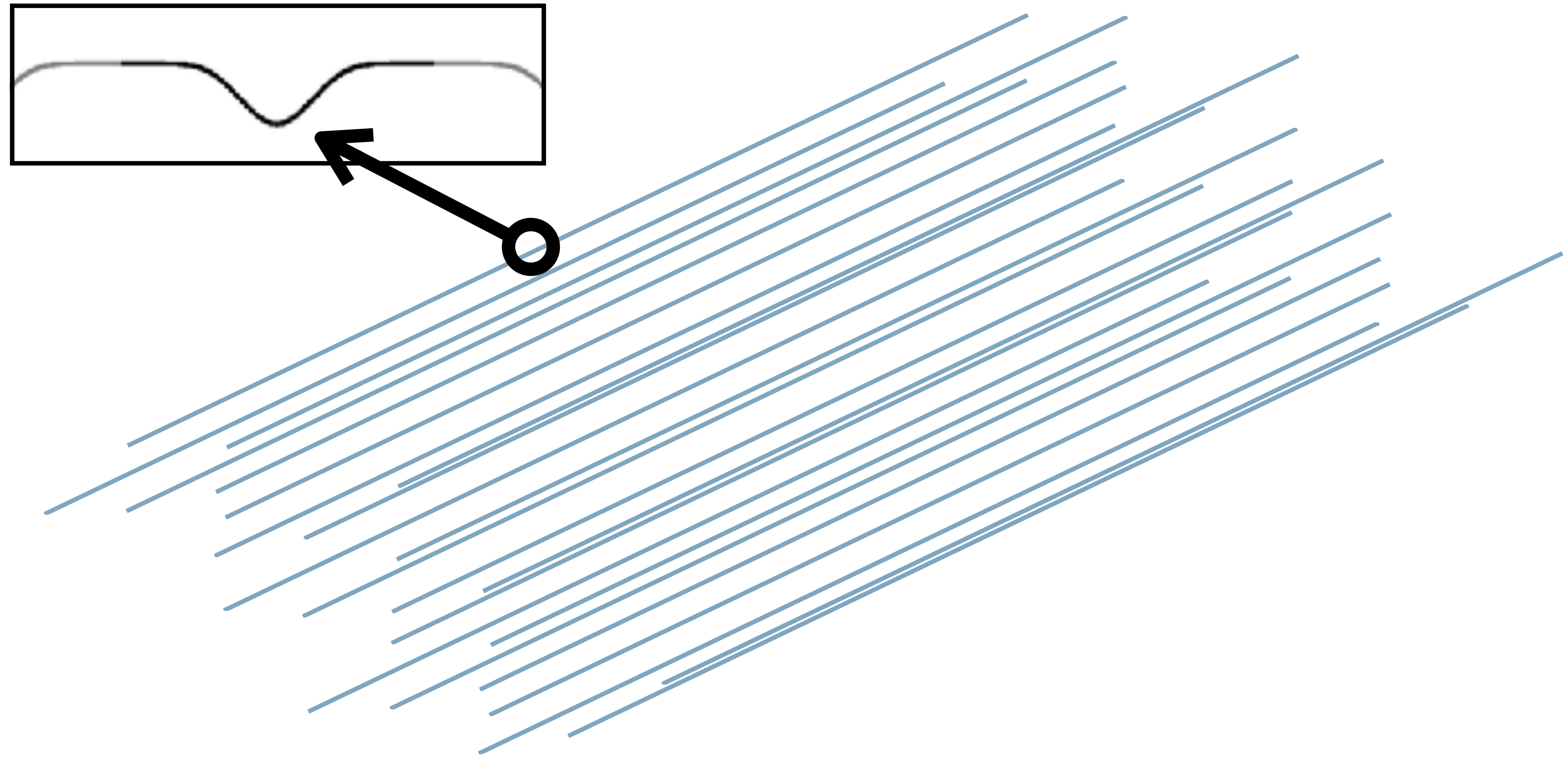
Model Hypotheses

i) “A scratched pattern is combination of several parallel scratch layers of various directions.”



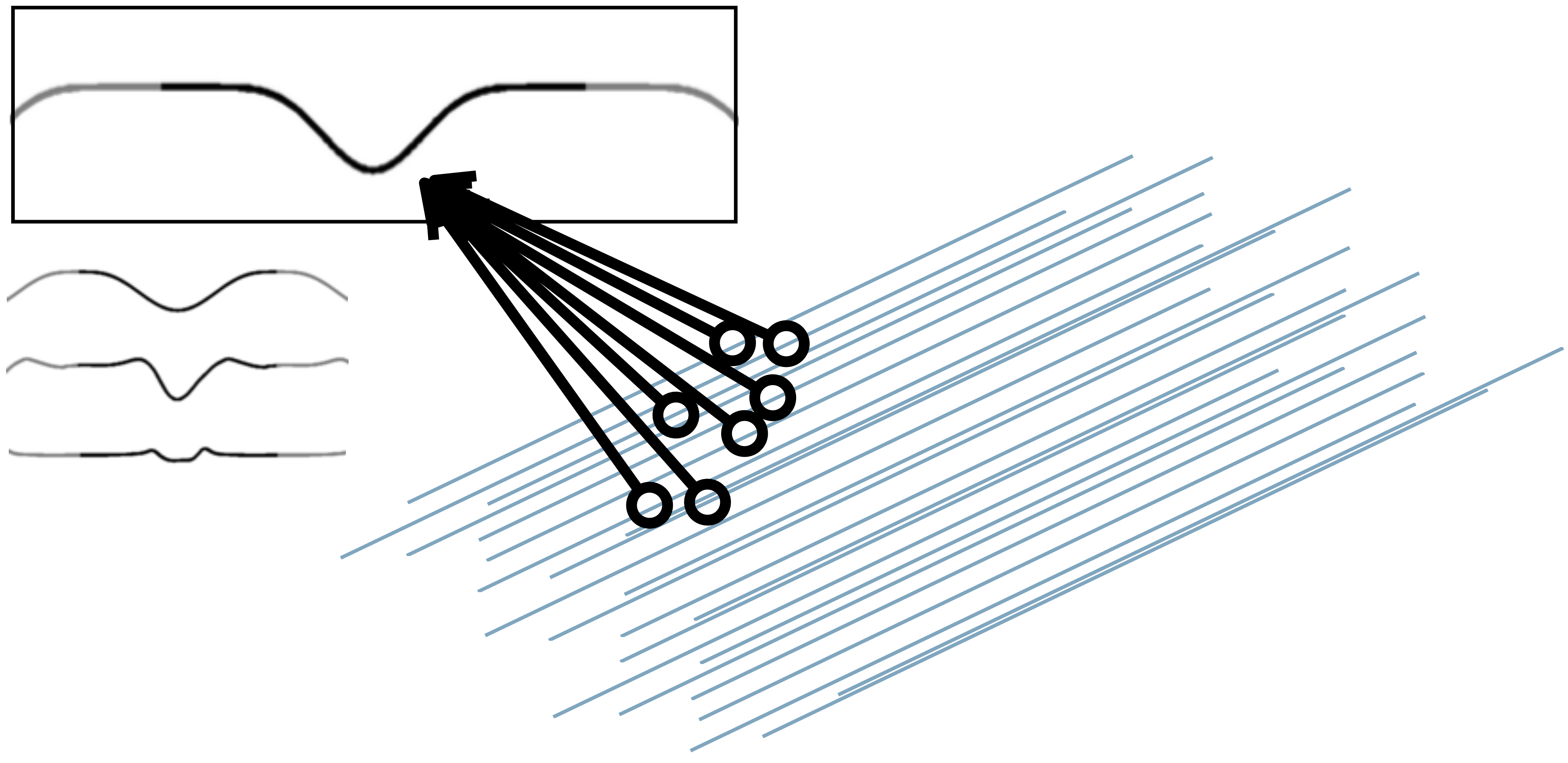
Model Hypotheses

ii) “In a parallel scratch layer, each scratches do not intersect and base surface between them are locally flat.”



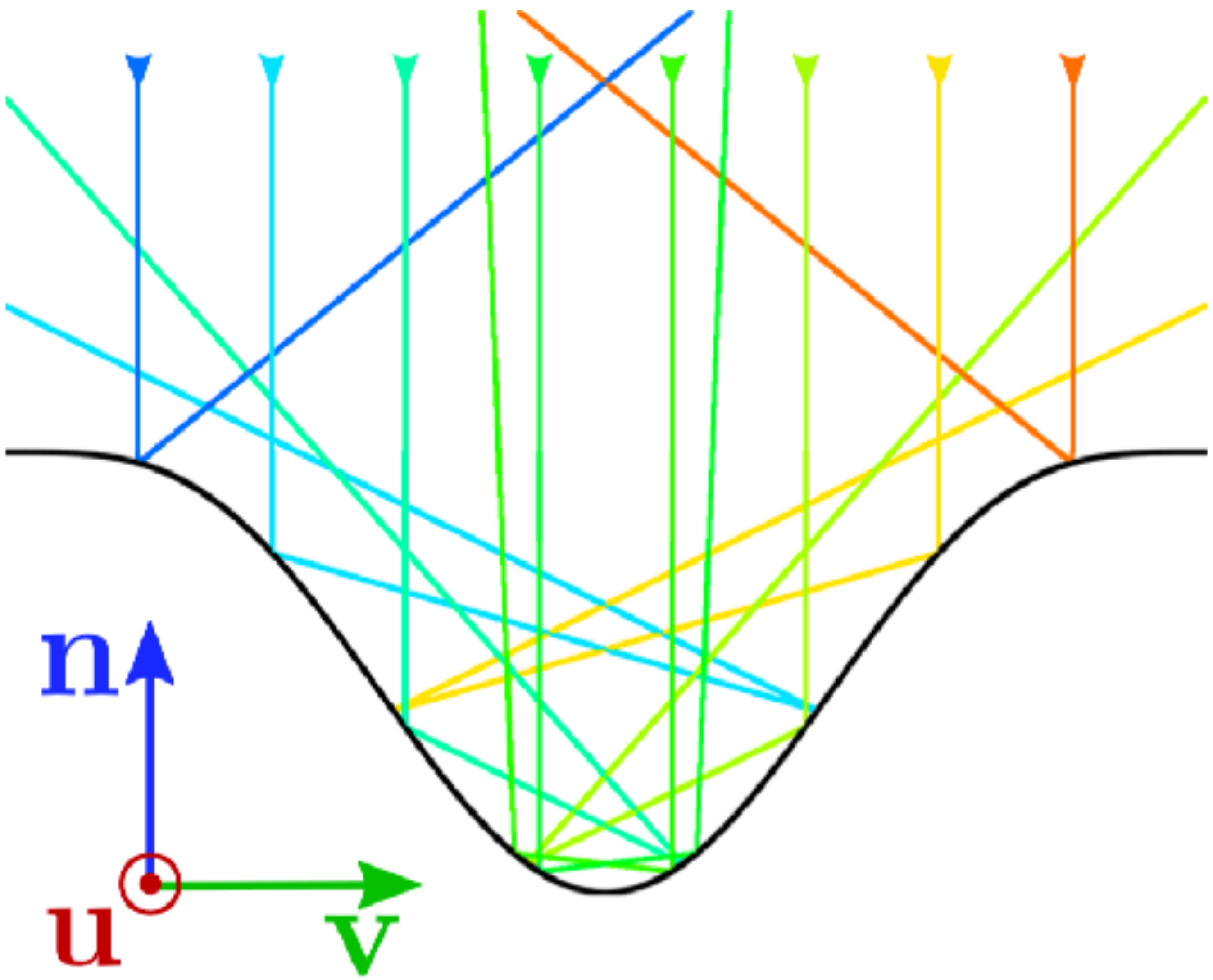
Model Hypotheses

iii) “All scratches share a 1D scratch profile.”



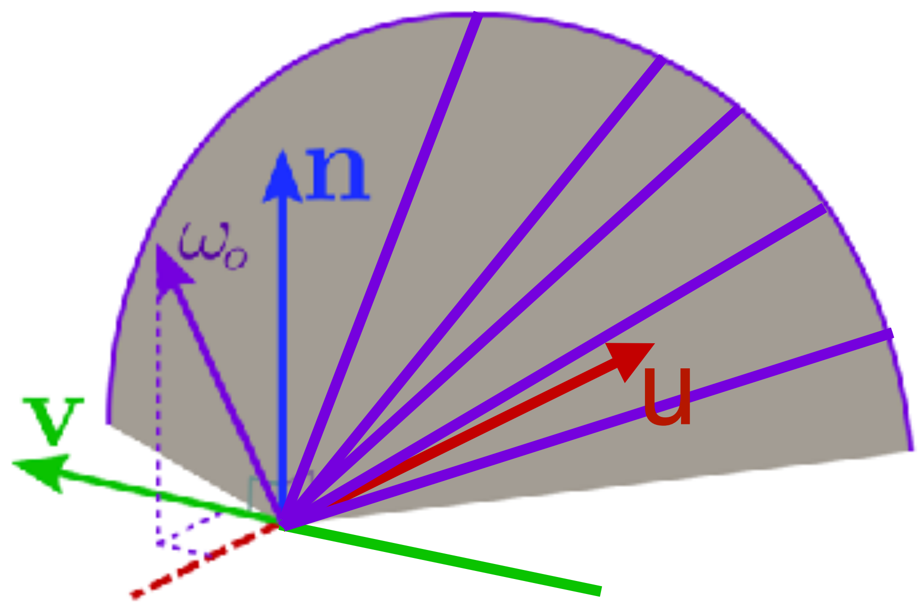
Model Hypotheses

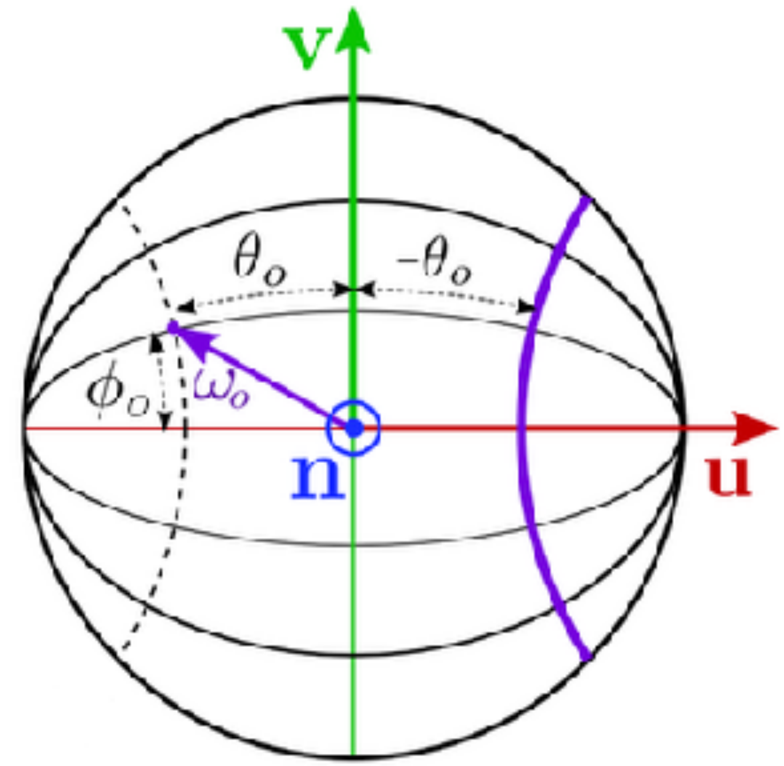
iv) "Inside a 1D scratch profile, reflection is perfect mirror."
(This assumption will be relaxed later.)



Model Hypotheses

iv) "Inside a 1D scratch profile, reflection is perfect mirror."





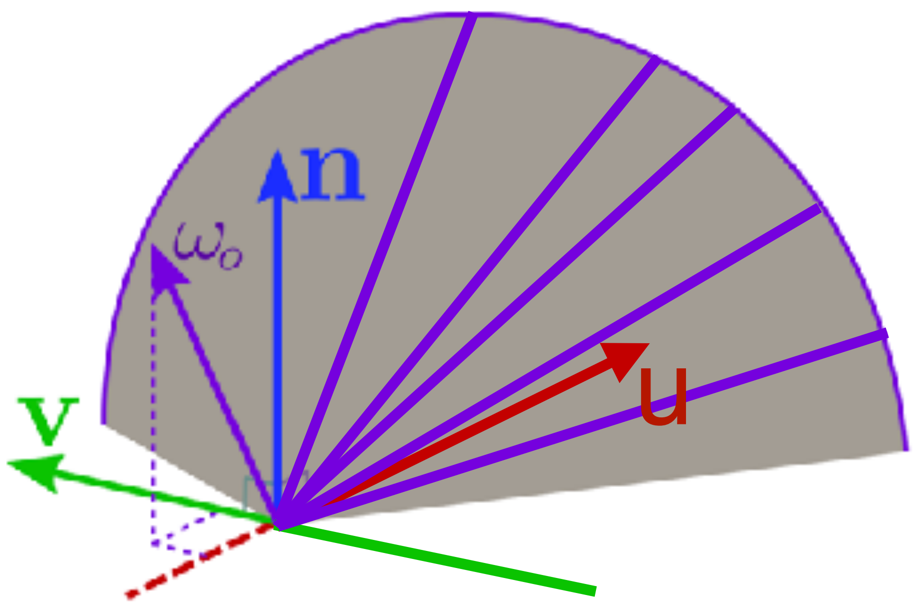
Upper view

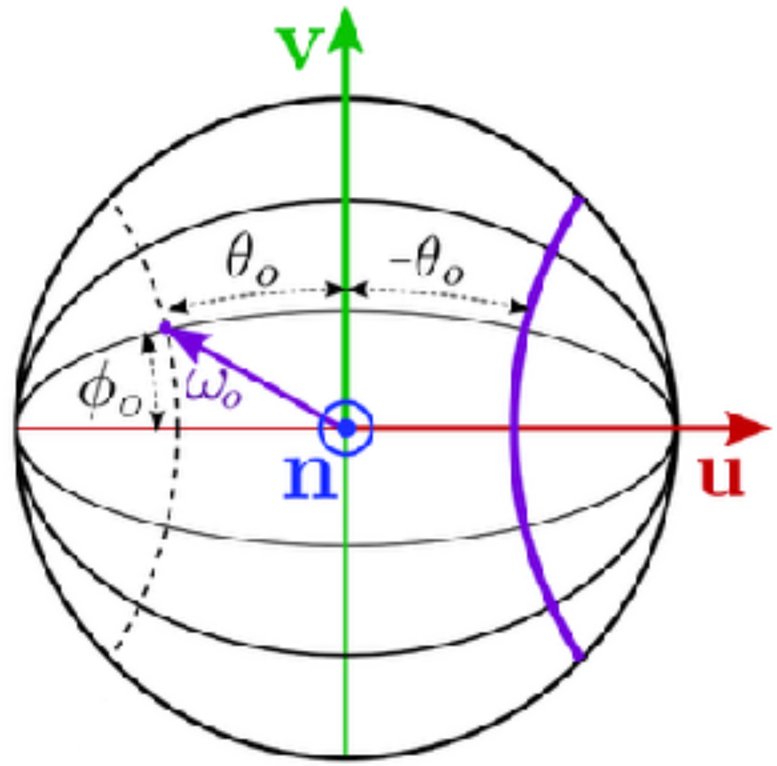
For a mirror scratch aligned with **u**, light reflected in the outgoing direction $\boldsymbol{\omega}_o$ lies on a half-cone of directions (θ_o-isocurve).

$\theta_o = \theta_i$, therefore 3D BRDF

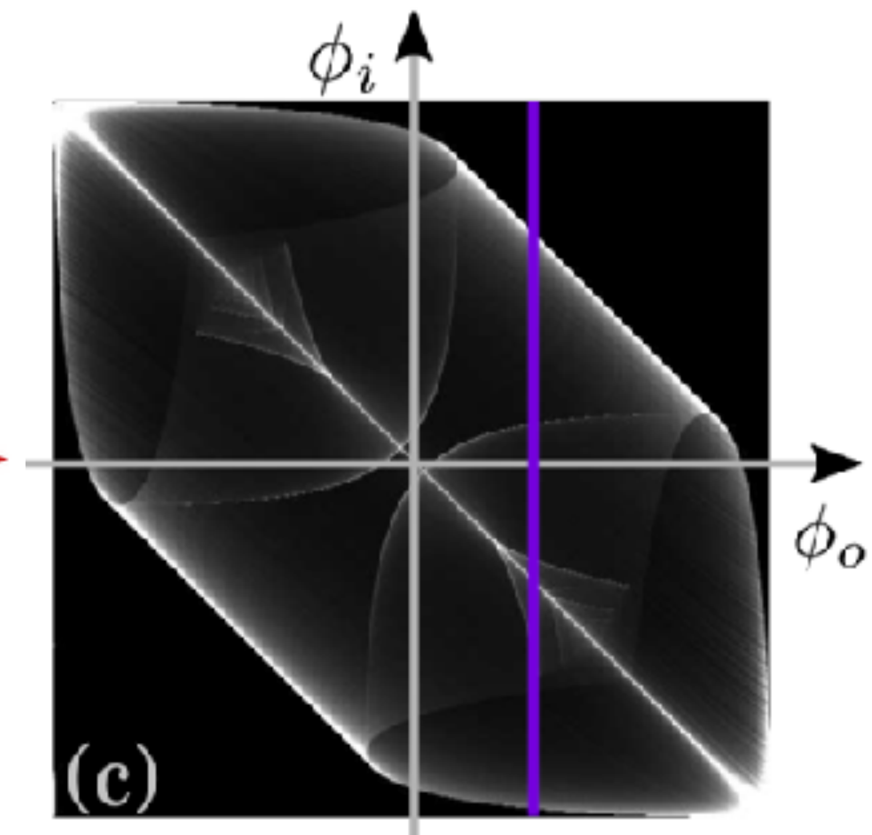
Model Hypotheses

iv) "Inside a 1D scratch profile, reflection is perfect mirror."





Upper view

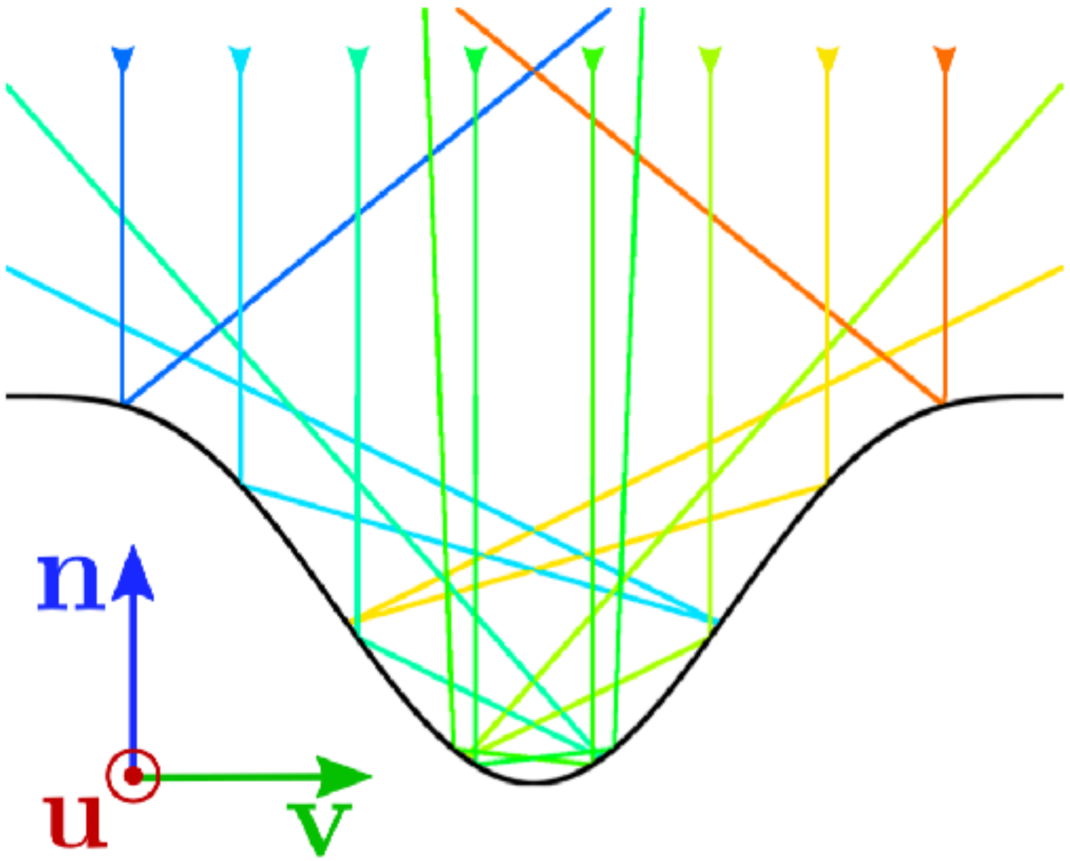


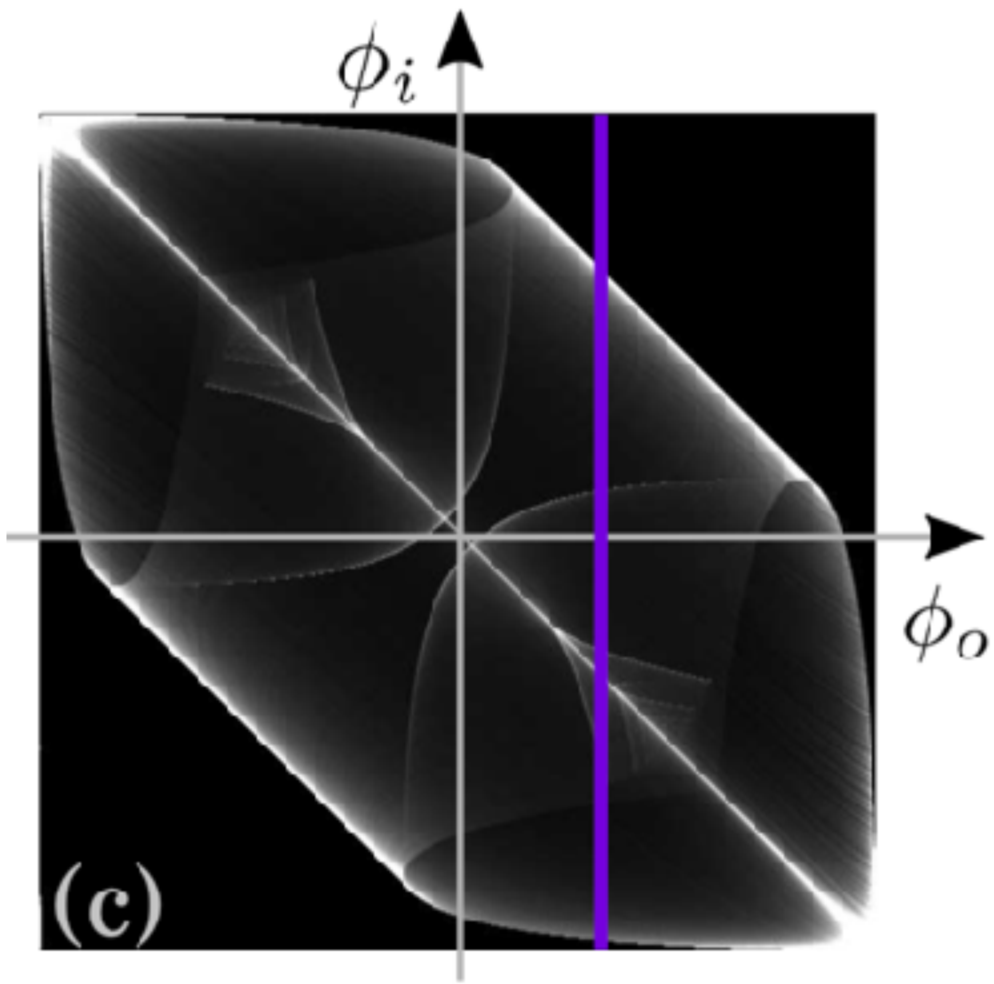
Moreover, since a scratch consists of an extruded profile, varying the elevation does not change BRDF

$\theta_o = \theta_i$, and invariant to θ_o , therefore 2D BRDF

Model Hypotheses

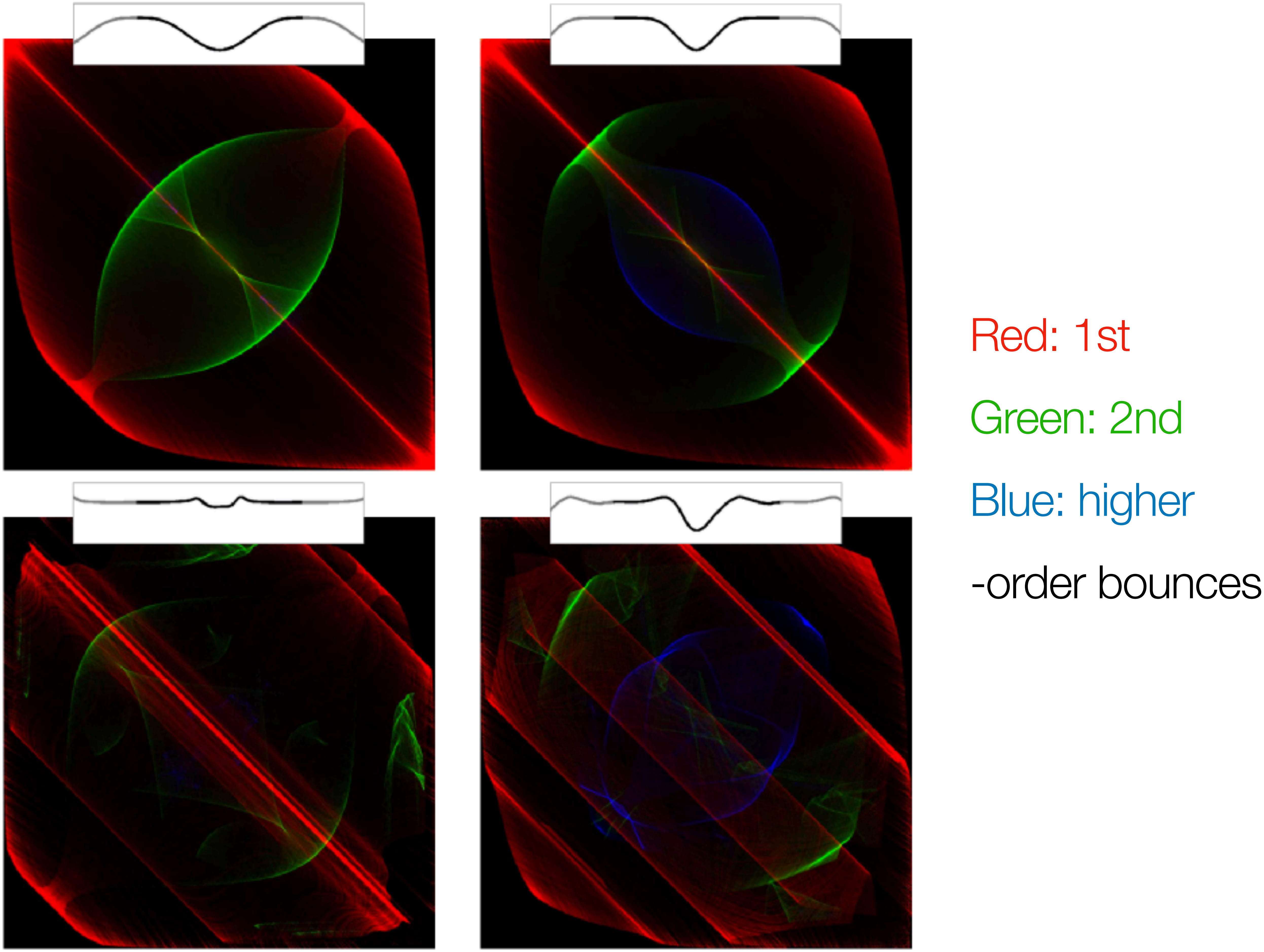
iv) “Inside a 1D scratch profile, reflection is perfect mirror.”





This 2D BRDF is evaluated by 2D ray tracing.

$$\rho_m(\theta_o,\phi_o,\theta_i,\phi_i) = \begin{cases} \rho_m(\phi_o,\phi_i) & \text{if } \theta_i = -\theta_o, \\ 0 & \text{otherwise.} \end{cases}$$



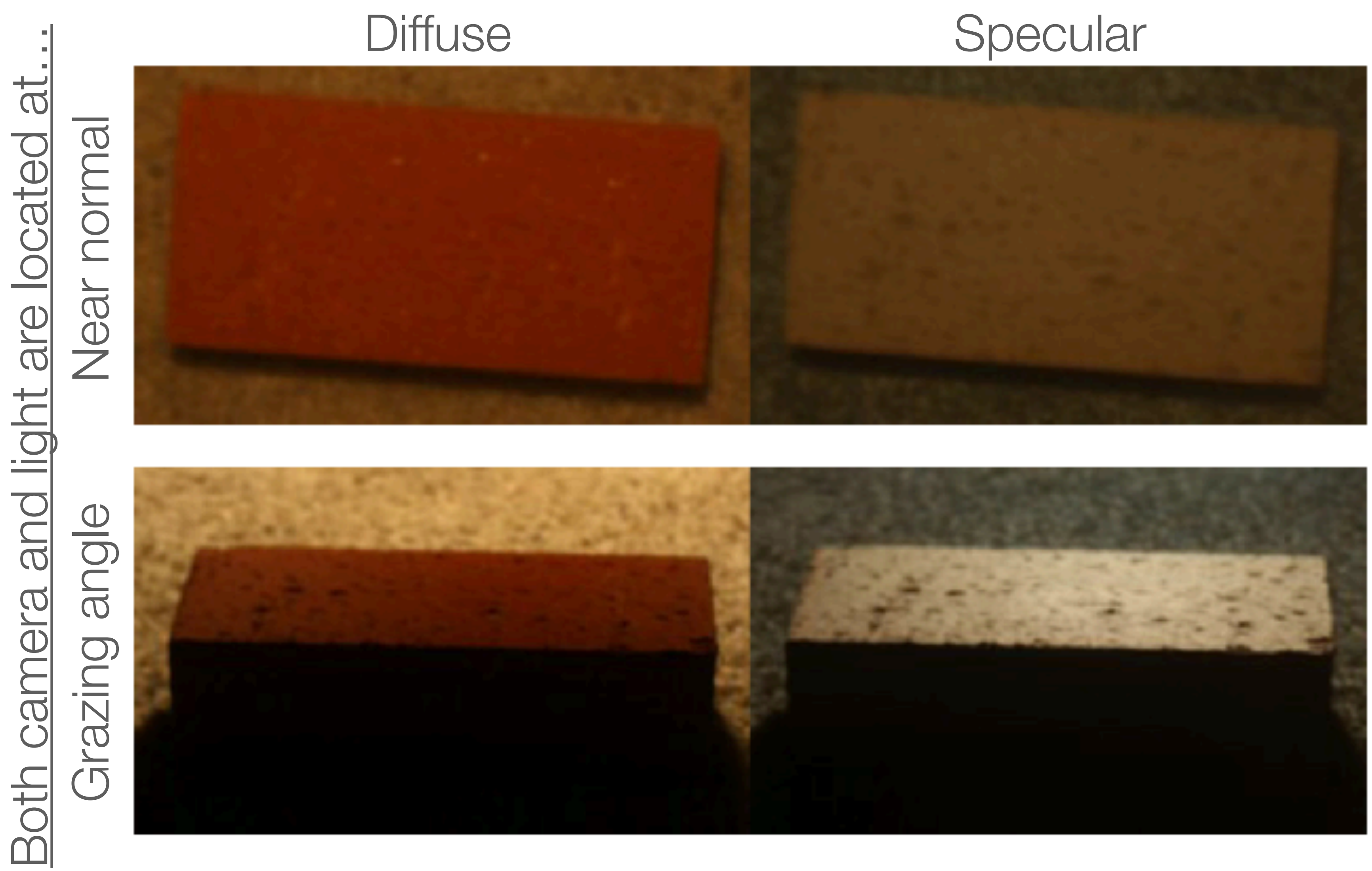
Red: 1st
Green: 2nd
Blue: higher
-order bounces

Fresnel Effect

iv) "Inside a 1D scratch profile, reflection is ~~perfect mirror~~."

Still specular reflection, but let's consider *Fresnel effect* now:

"At near-grazing incidence, media interfaces appear mirror-like."



Fresnel Effect

"At near-grazing incidence, media interfaces appear mirror-like."

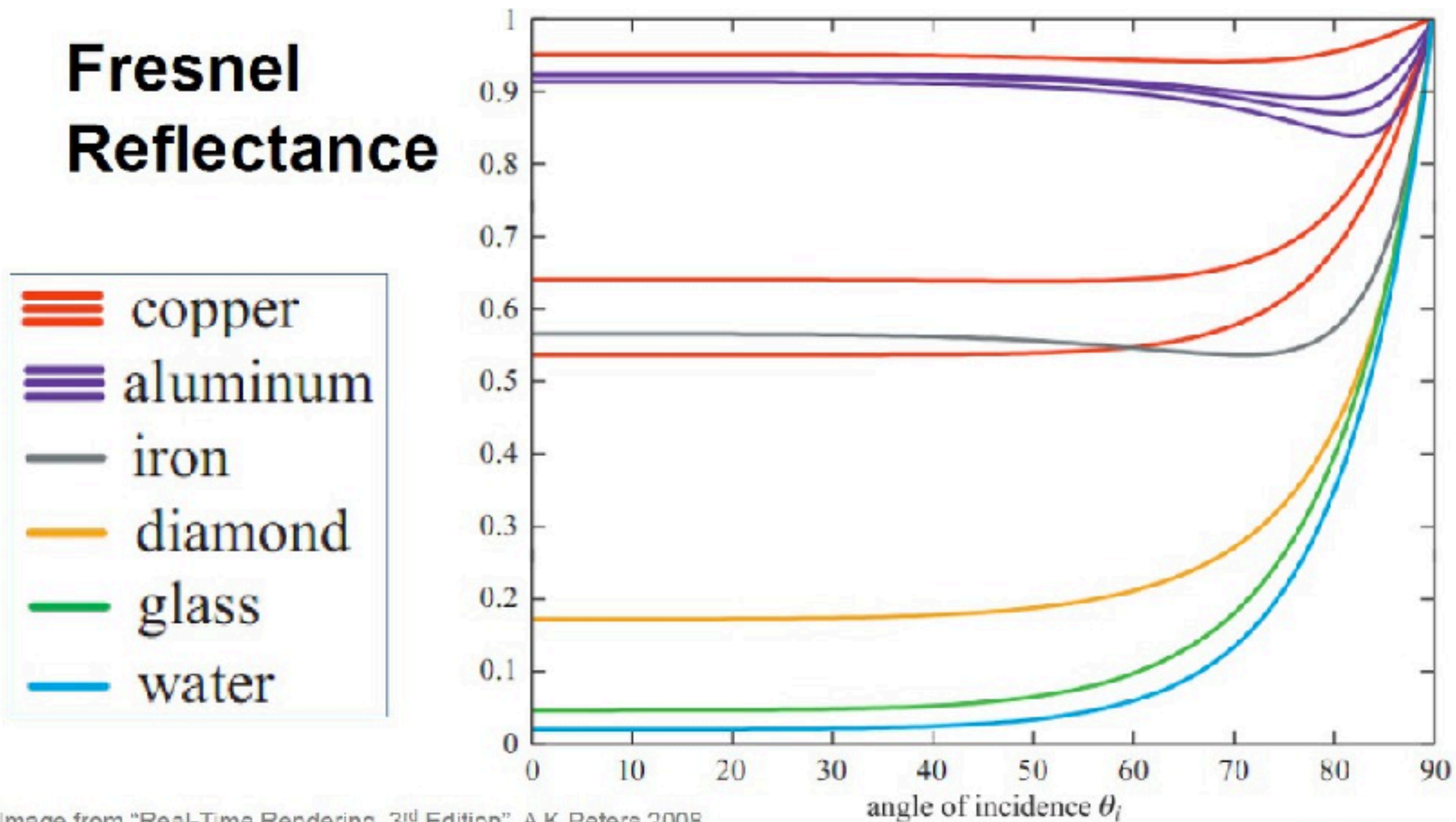
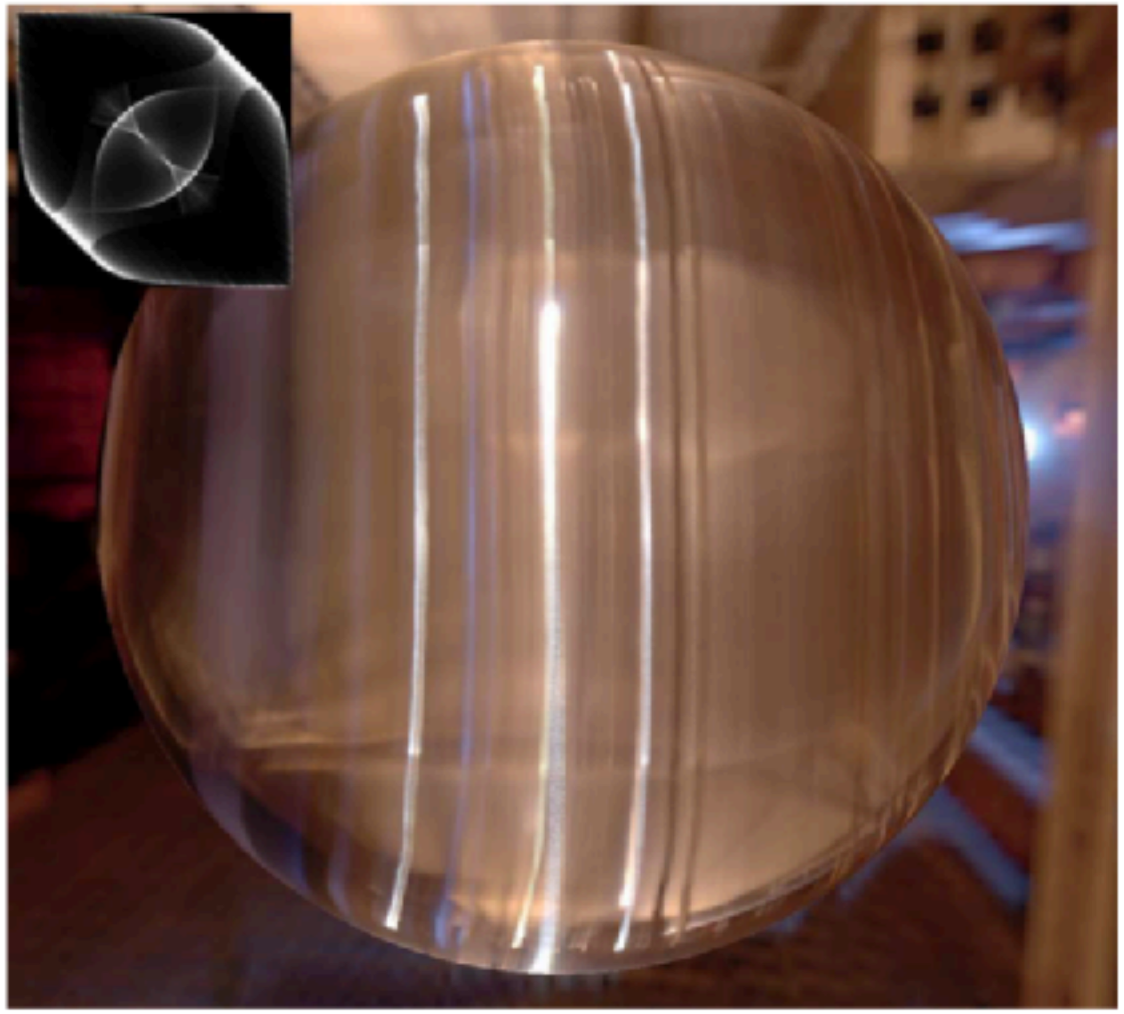


Image from "Real-Time Rendering, 3rd Edition", A K Peters 2008

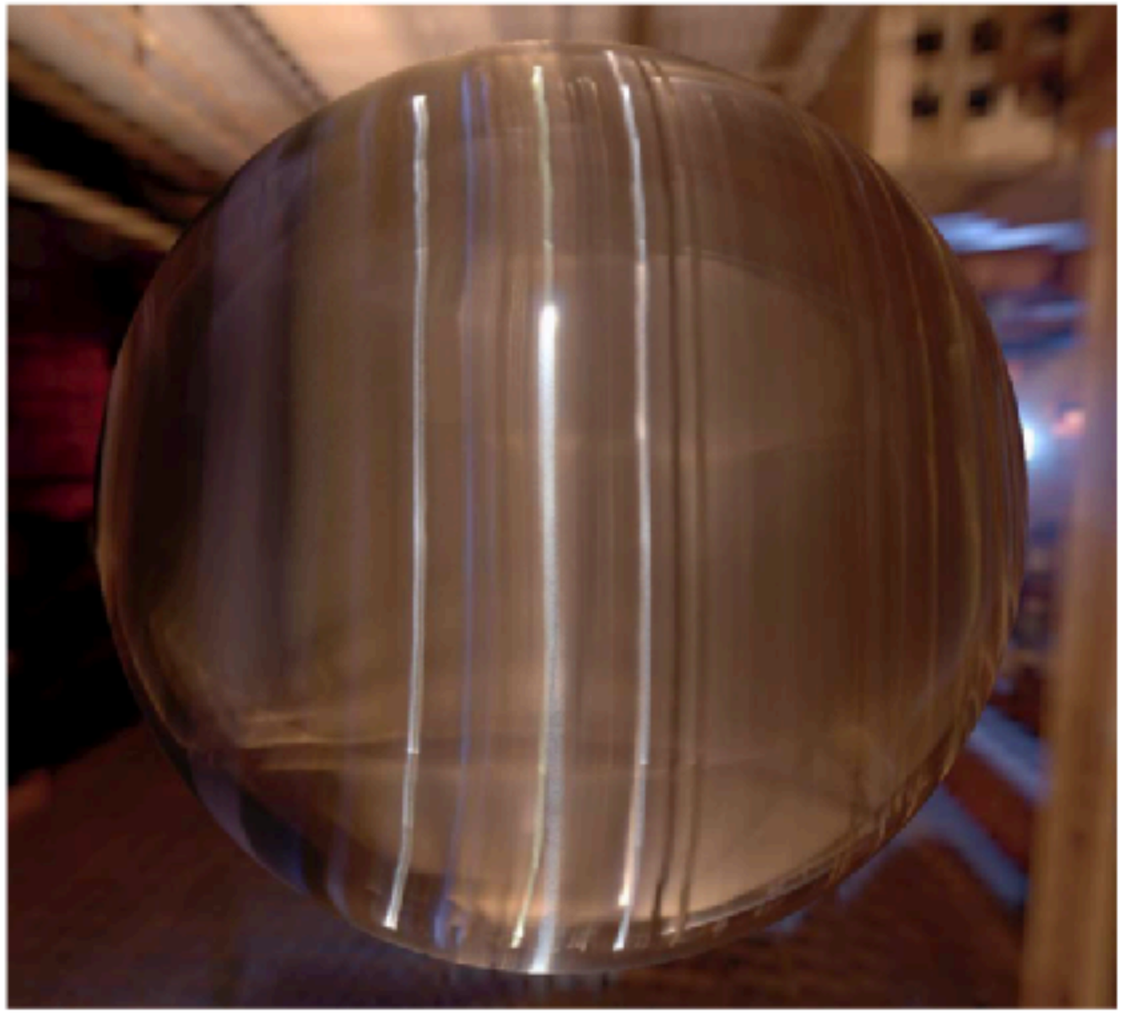
Fresnel Effect

Fresnel Effect is important in scratch BRDF since during a number of inter-reflections much energies should be lost.



(e) Ours (no Fresnel)

Wrong, it's too bright.

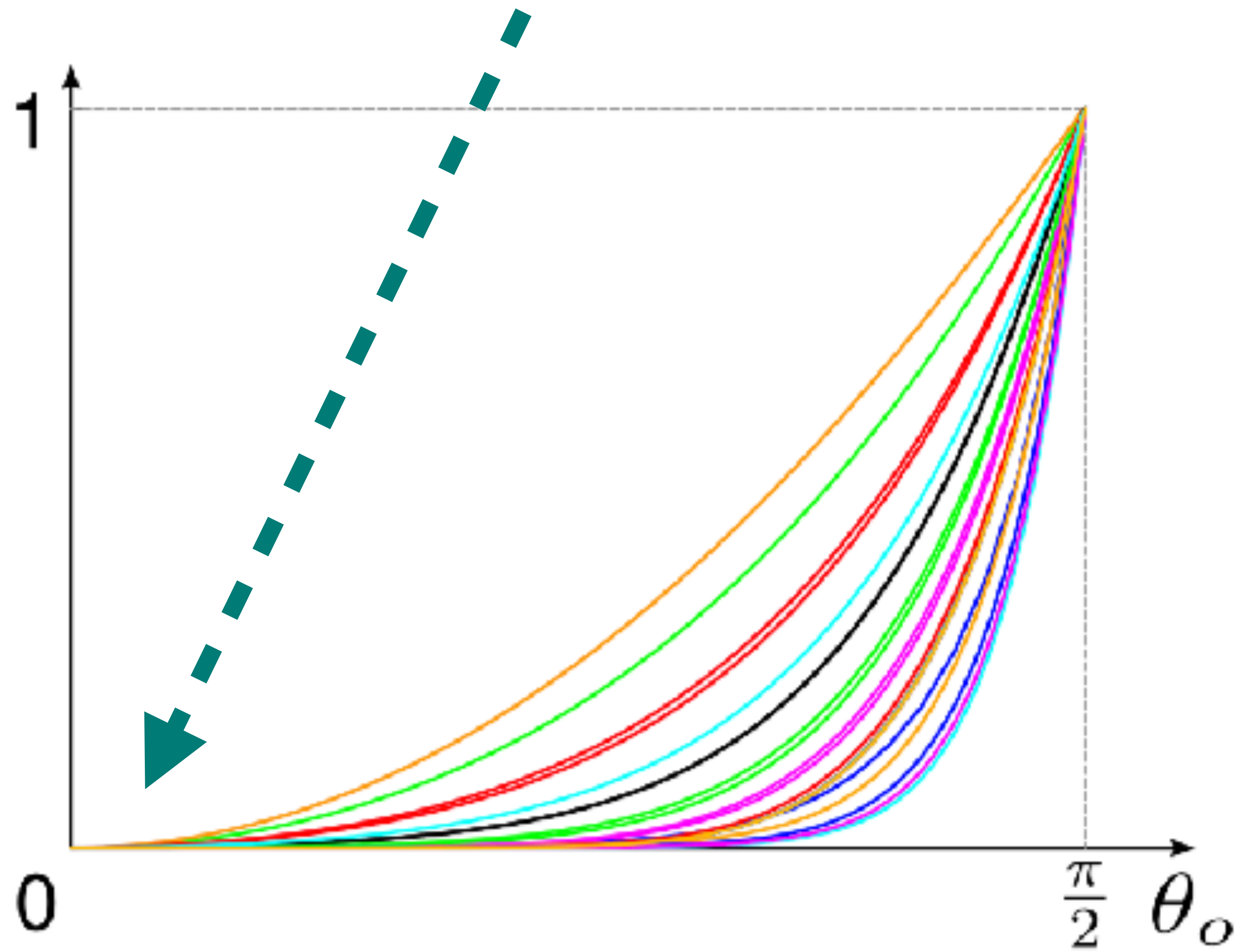


(f) Ground truth

Fresnel Effect

Considering Fresnel Effect during 2D ray tracing,

we can observe that with smaller θ_o, scratch BRDF is smaller.



[0,1] clamped BRDF for fixed (ϕ_o,ϕ_i) pair

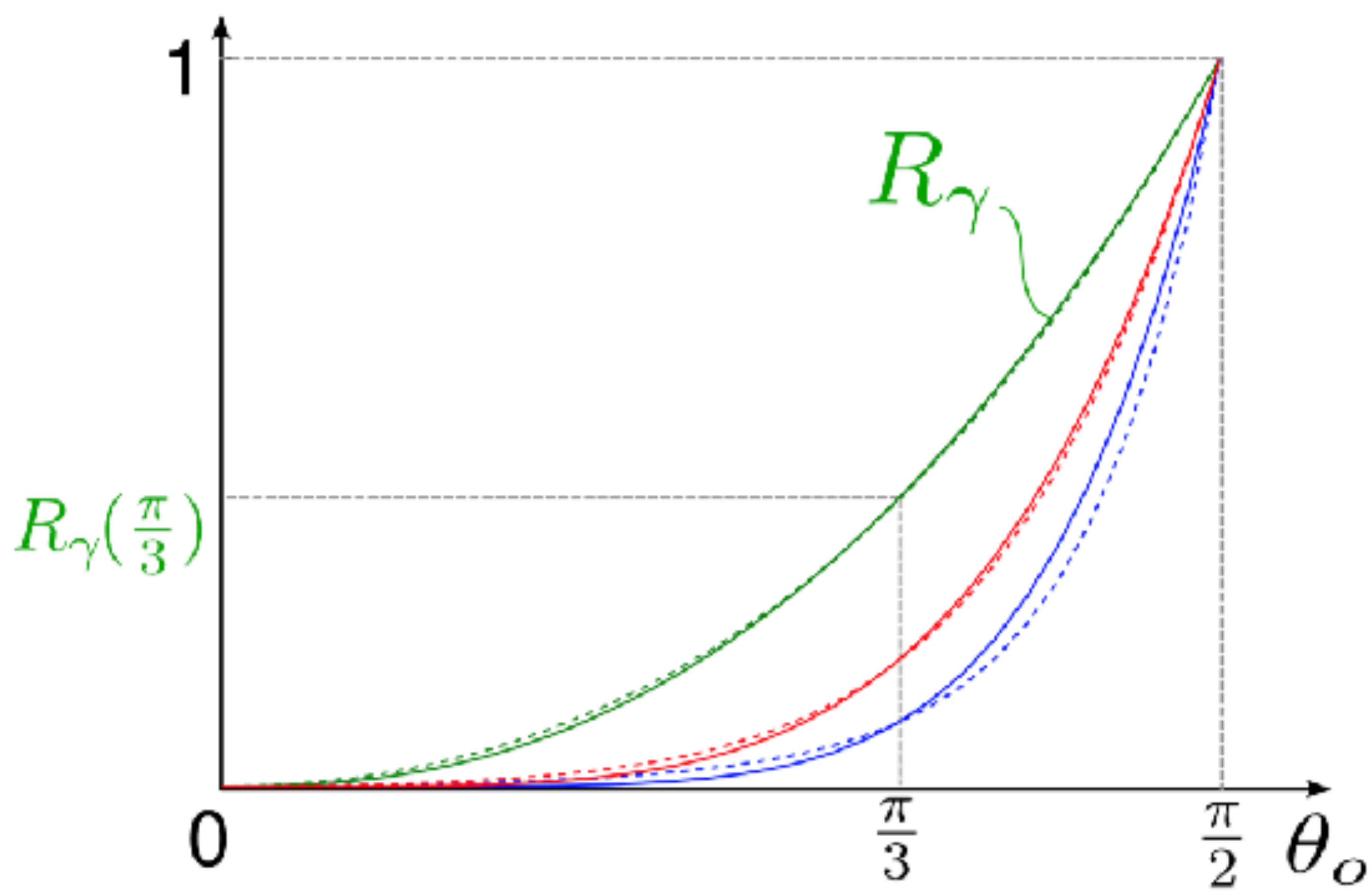
Fresnel Effect

We want to stay cheap.

Perform 2D ray tracing w/ Fresnel effect, **only three times:**

$\theta_o = 0°$, $\theta_o = 60°$, and $\theta_o = 90°$

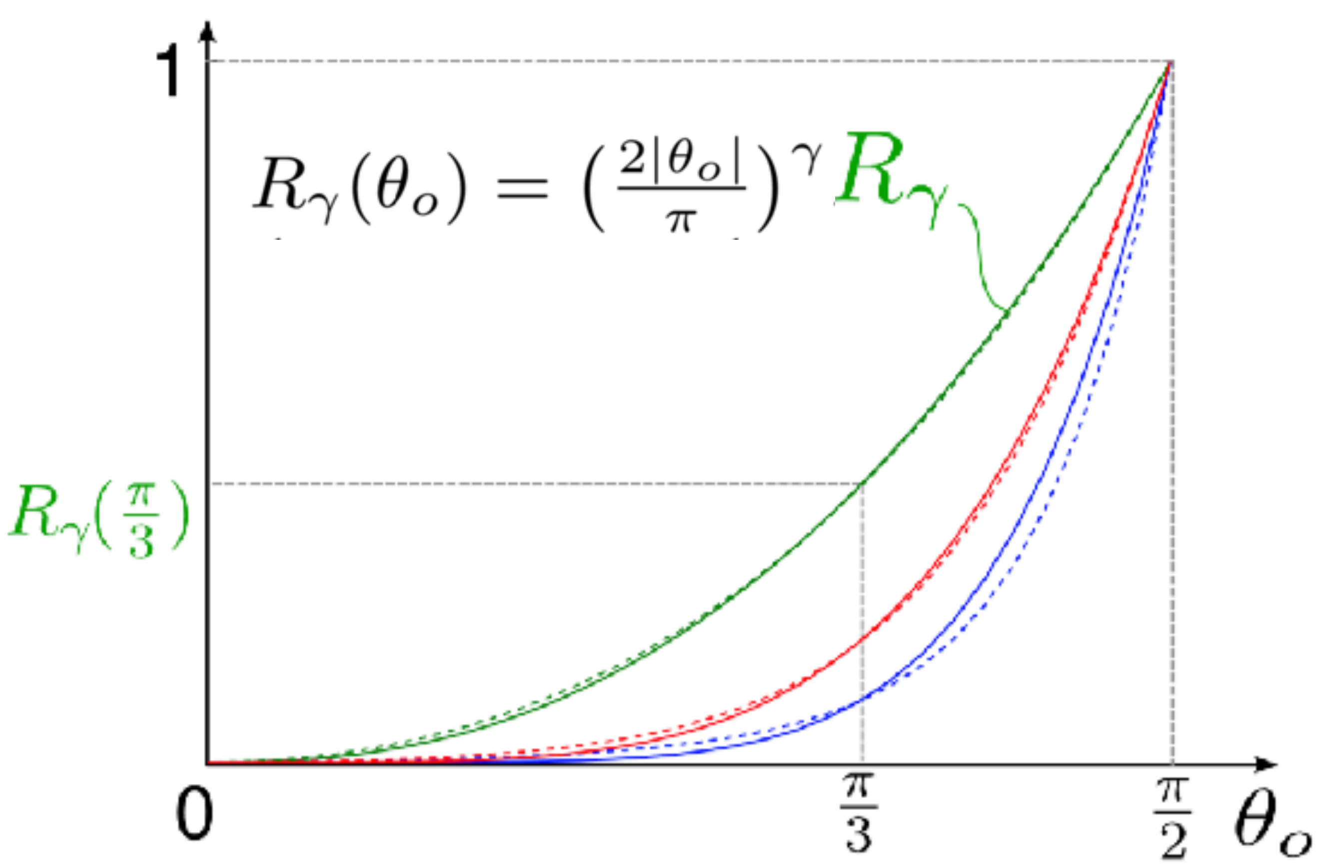
And then fit to Gamma curve: $R_\gamma(\theta_o) = \left(\frac{2|\theta_o|}{\pi}\right)^\gamma$

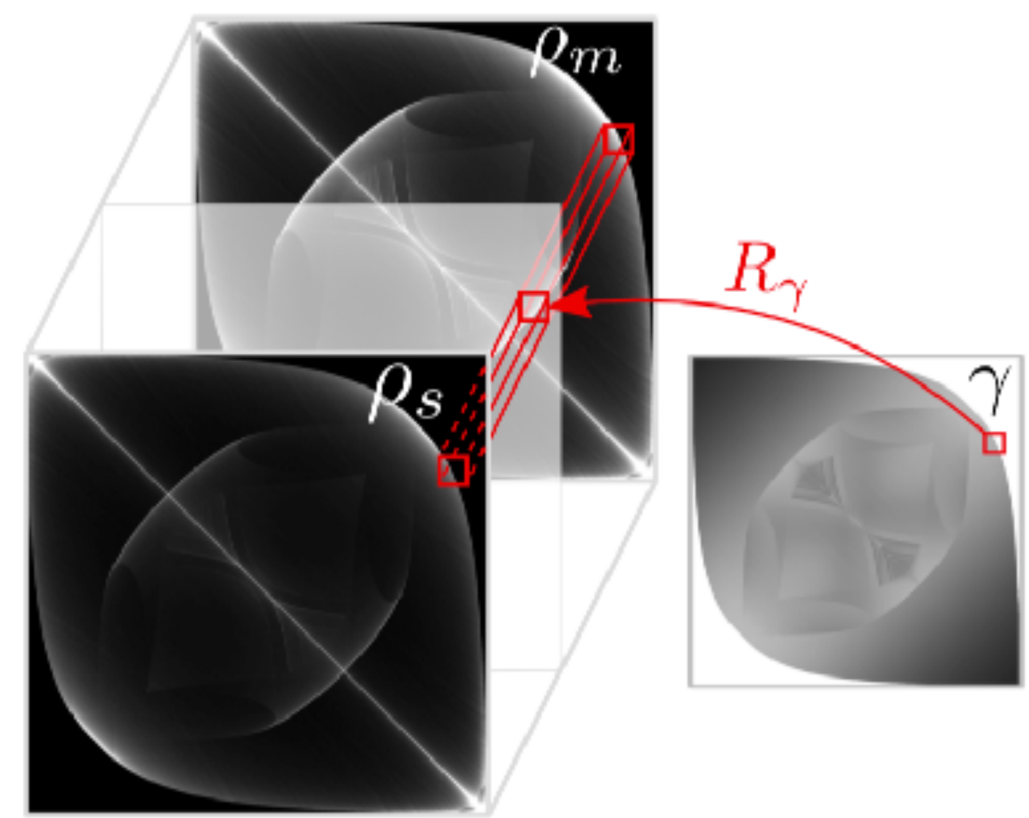


Fresnel Effect

We can approximate 3D scratch BRDF with cheap computation.

$$\rho_s(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) \approx R_\gamma(\theta_o)\, \rho_s(\tfrac{\pi}{2}, \phi_o, \phi_i) + (1 - R_\gamma(\theta_o))\, \rho_s(0, \phi_o, \phi_i)$$

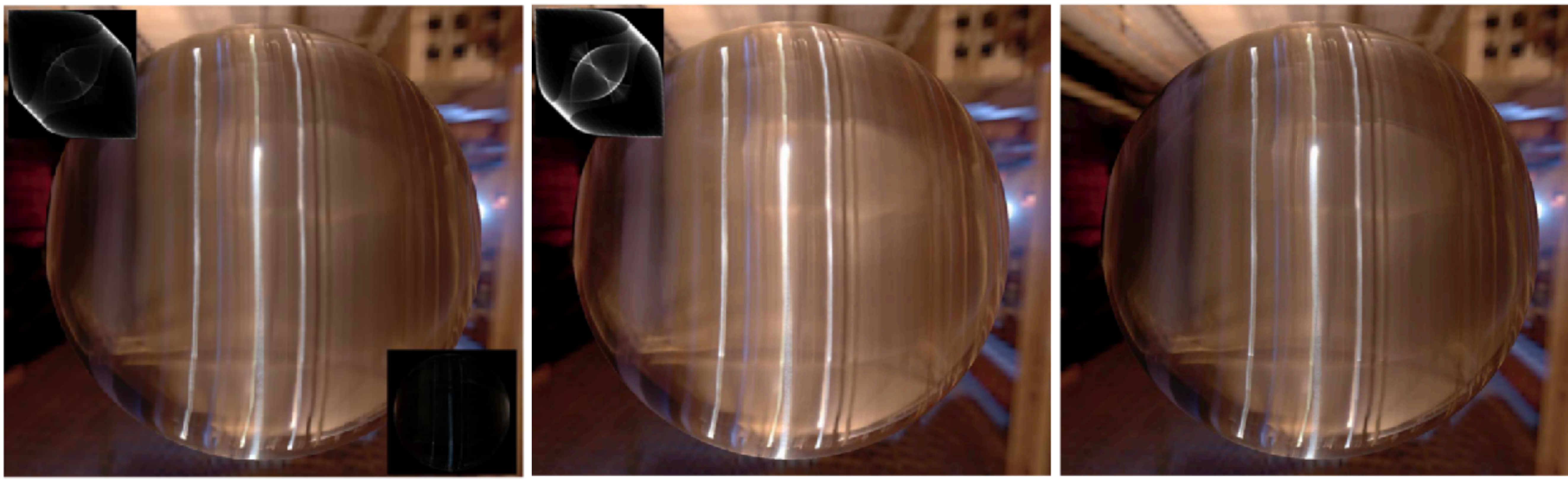




(b) *Scratch BRDF*

[2] Scratched Materials and SV-BRDF

Fresnel Effect



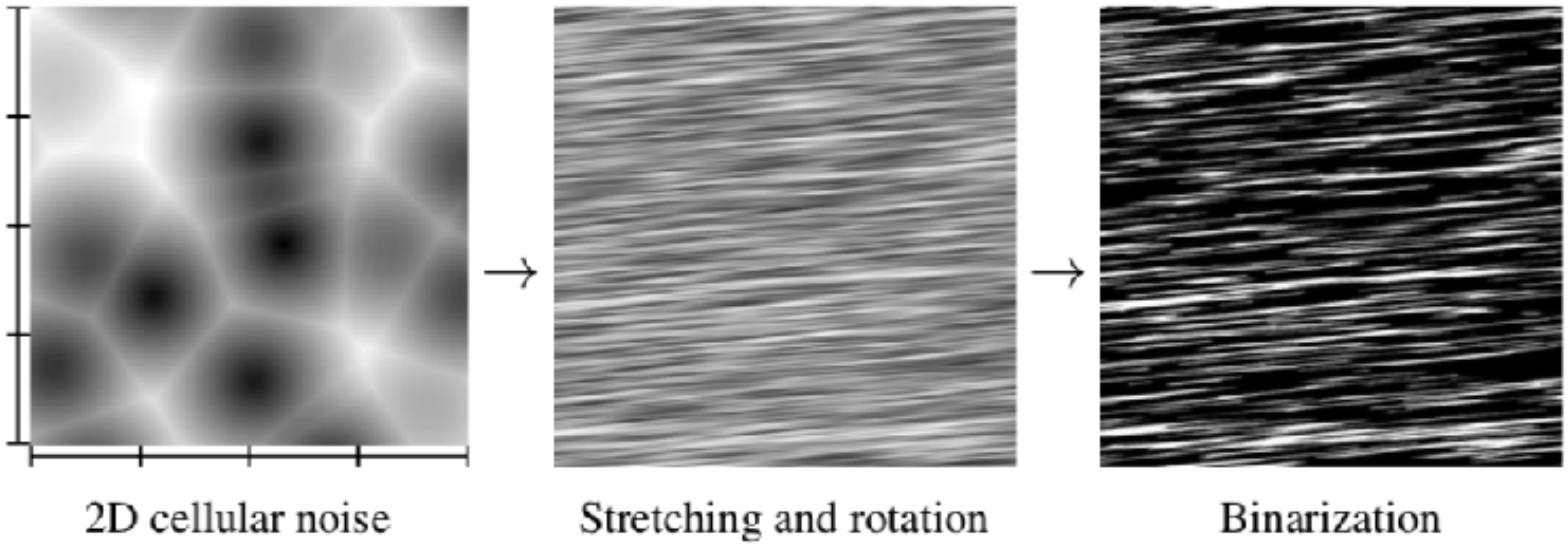
(d) Ours (all bounces) (e) Ours (no Fresnel) (f) Ground truth

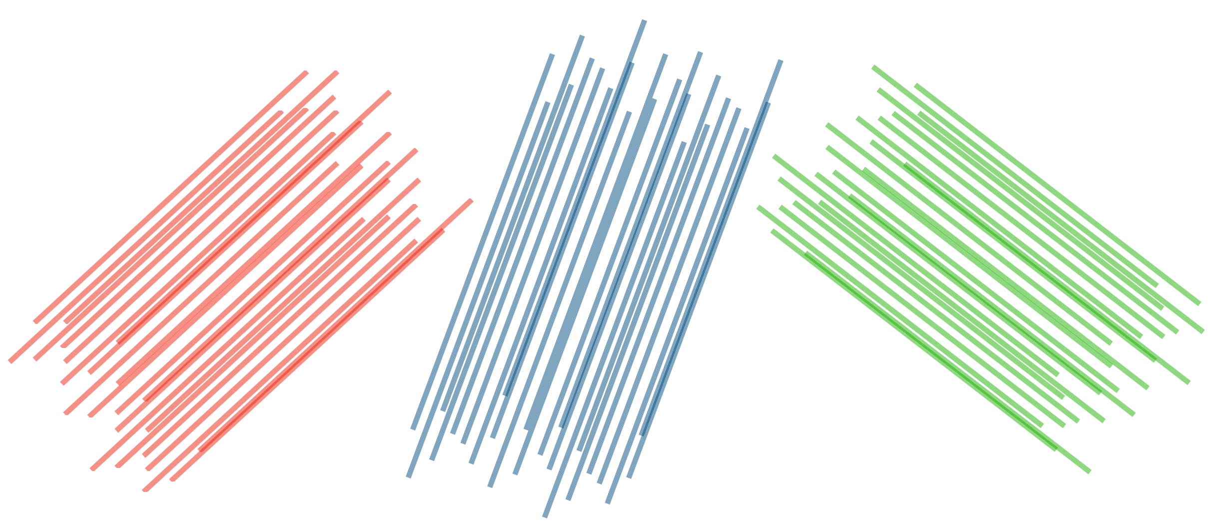
Fresnel Effect considered

Wrong, it's too bright.

SVBRDF

- Generate scratch indicator α(x) for each layer independently.

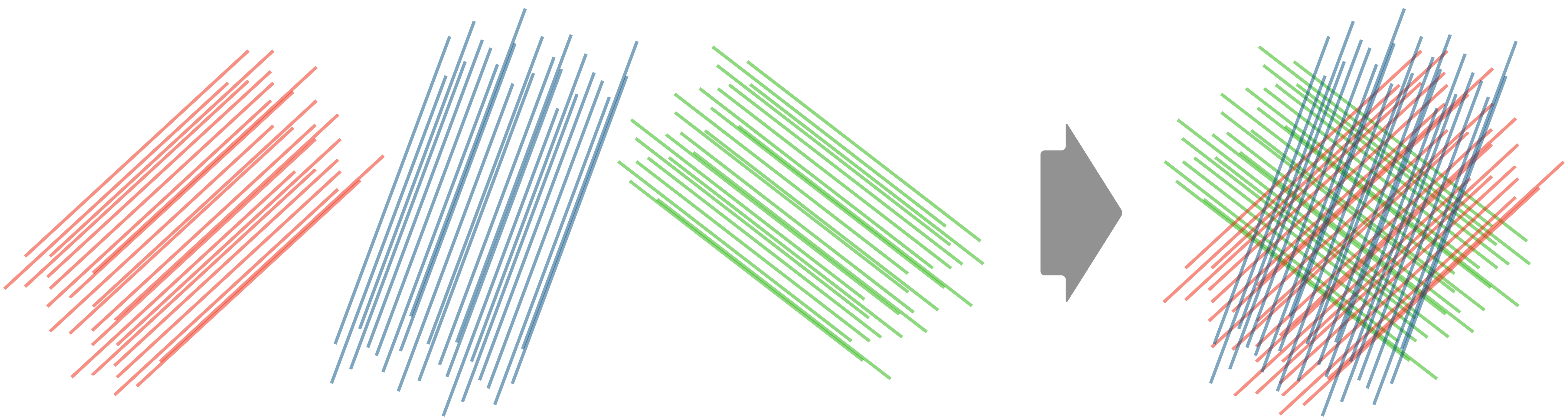




SVBRDF

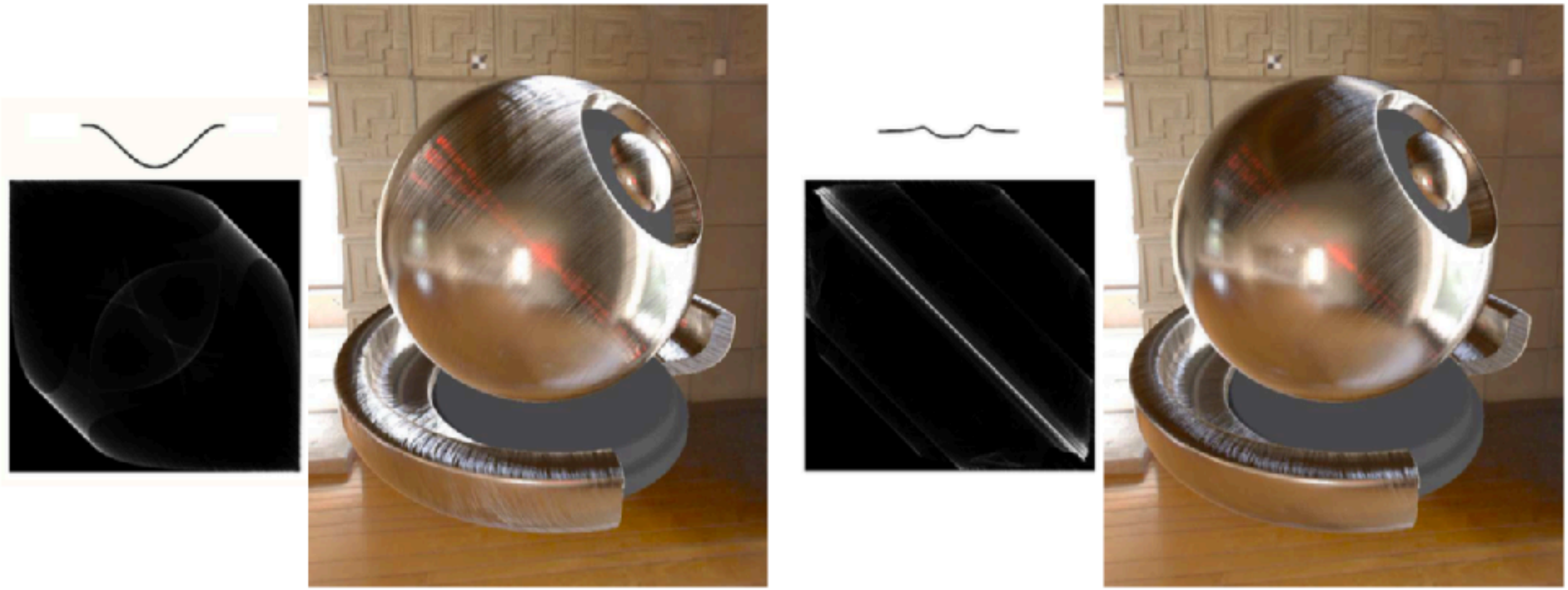
- Compute a combination of scratch BRDFs weighted by area:

$$\bar{\rho}(\mathbf{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = \sum \alpha_k(\mathbf{x}) \rho_{s,k}(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i)$$



$$\rho(\mathbf{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = \begin{cases} \bar{\rho}/\bar{\alpha}(\mathbf{x}) & \text{if } \bar{\alpha}(\mathbf{x}) > 1 \\ \bar{\rho} + (1 - \bar{\alpha}(\mathbf{x}))\rho_b & \text{otherwise.} \end{cases}$$

Results



(a) Quartic profile (b) Brushed w/ quartic (c) Measured profile (d) Brushed w/ measured

[2] Scratched Materials and SV-BRDF

Results

Thanks!