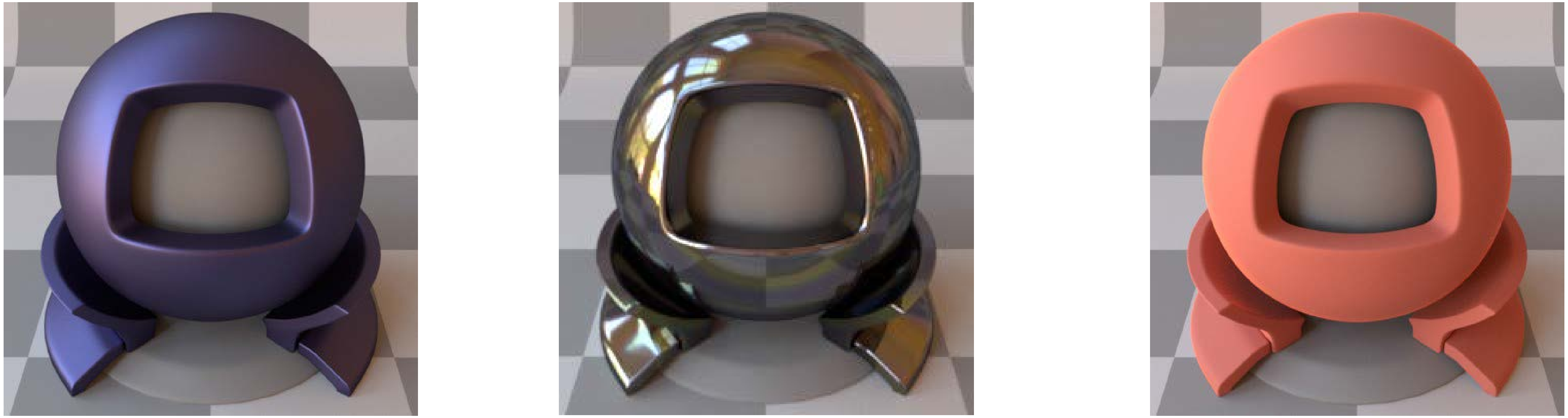
Microfacet model and Microfacet-based BRDF

2019.05.16
20186413 Murat Gaspard

Physically based microfacet BRDFs

$$L_o = \int_{\Omega_+} L_i \cdot f_r \cdot \cos\theta_i \cdot d\omega_i$$

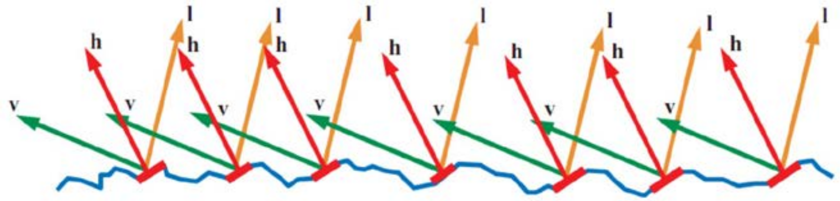


Microfacet Theory

From Hakyeong Kim's talk:

Materials = microsurfaces

Microsurfaces properties can be manipulated



$$f_r = \frac{F \cdot D \cdot G}{4 \cdot \cos\theta_i \cdot \cos\theta_o}$$

Papers

- **Extracting Microfacet-based BRDF Parameters from Arbitrary Materials with Power Iterations**

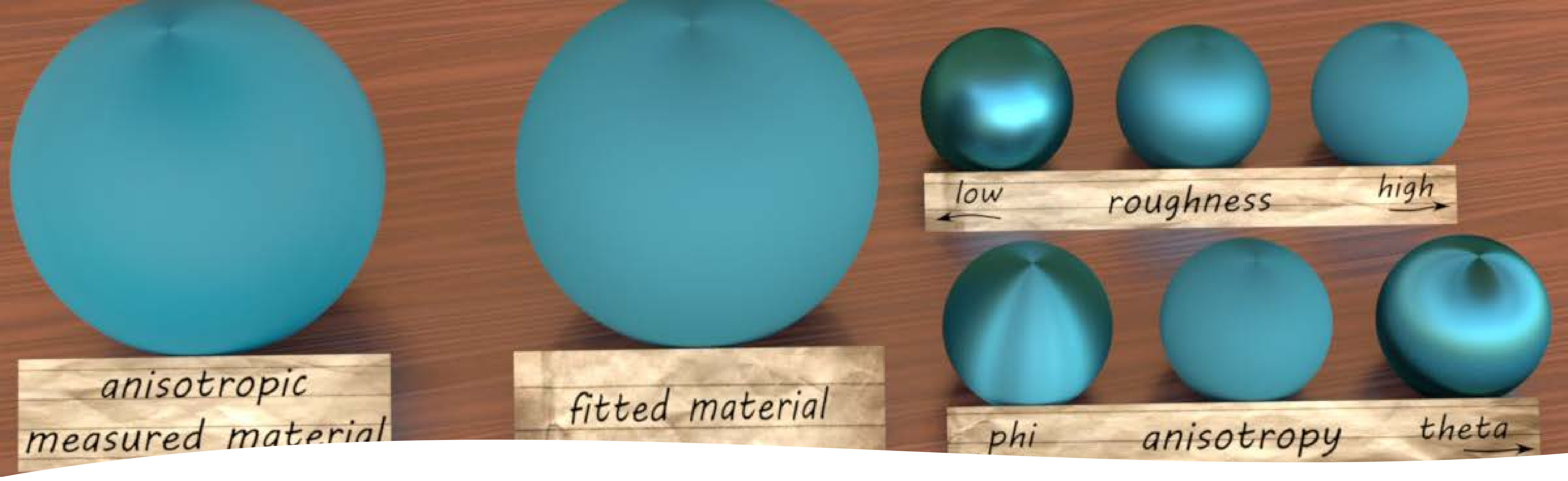
  Jonathan Dupuy   Eric Heitz   Pierre Poulin   Victor Ostromoukhov

  Eurographics Symposium on Rendering 2015

- **Fast Global Illumination with Discrete Stochastic Microfacets Using a Filterable Model**

  Beibei Wang   Lu Wang   Pierre Poulin   Nicolas Holzschuch

  Pacific Graphics 2018



Extracting Microfacet-based BRDF Parameters from Arbitrary Materials with Power Iterations

Jonathan Dupuy   Eric Heitz   Pierre Poulin   Victor Ostromoukhov

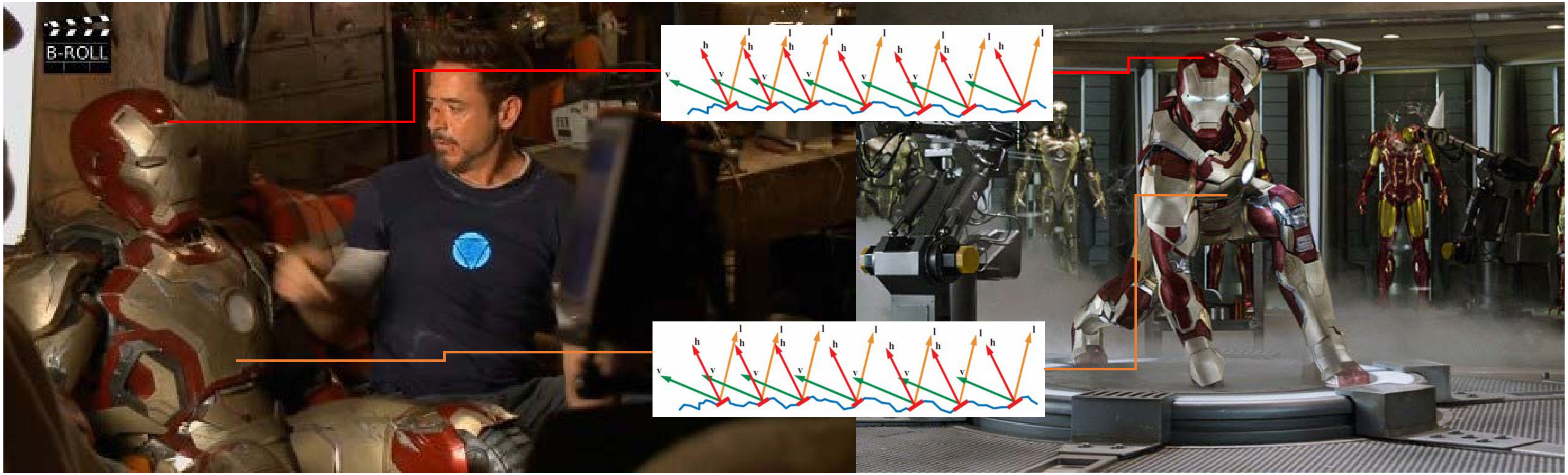
Eurographics Symposium on Rendering 2015

Context

Real
(real materials)

Digital
(microfacet BRDFs)

Context



Real
(real materials)

Digital
(microfacet BRDFs)

How to retrieve the microsurface from real material?

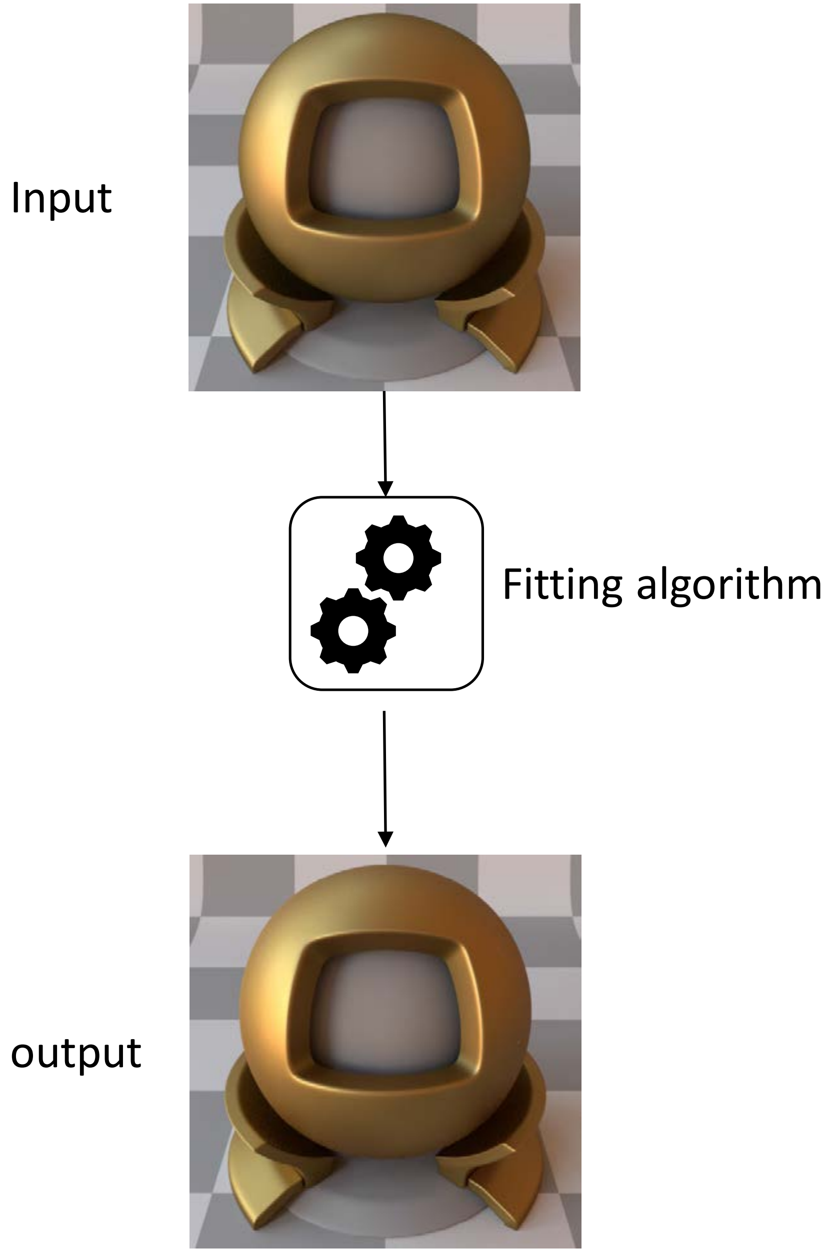
Microfacet BRDF Fitting

Approach:

- Fitted microsurface
- Minimize fitting metrics

Current limitations:

- Robustness / Speed
- Arbitrary metrics
- Reproducibility



Contribution

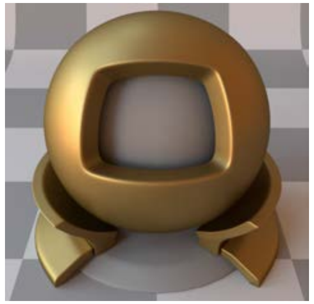
Idea:

- Find the NDF
- Approximize the Fresnel term

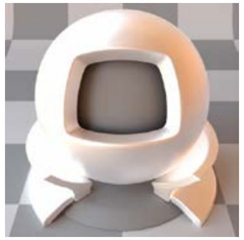
Properties:

- Robustness
- Simplicity
- Speed
- Reproducibility

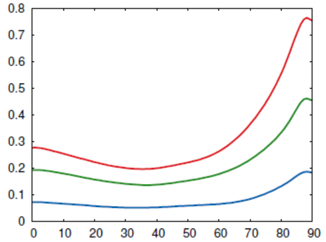
Input



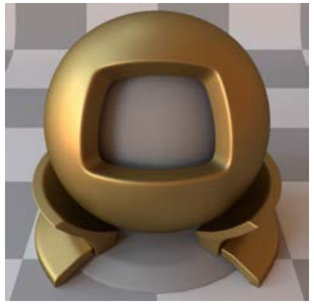
NDF



Fresnel



- Tabulated
- GGX
- Beckmann

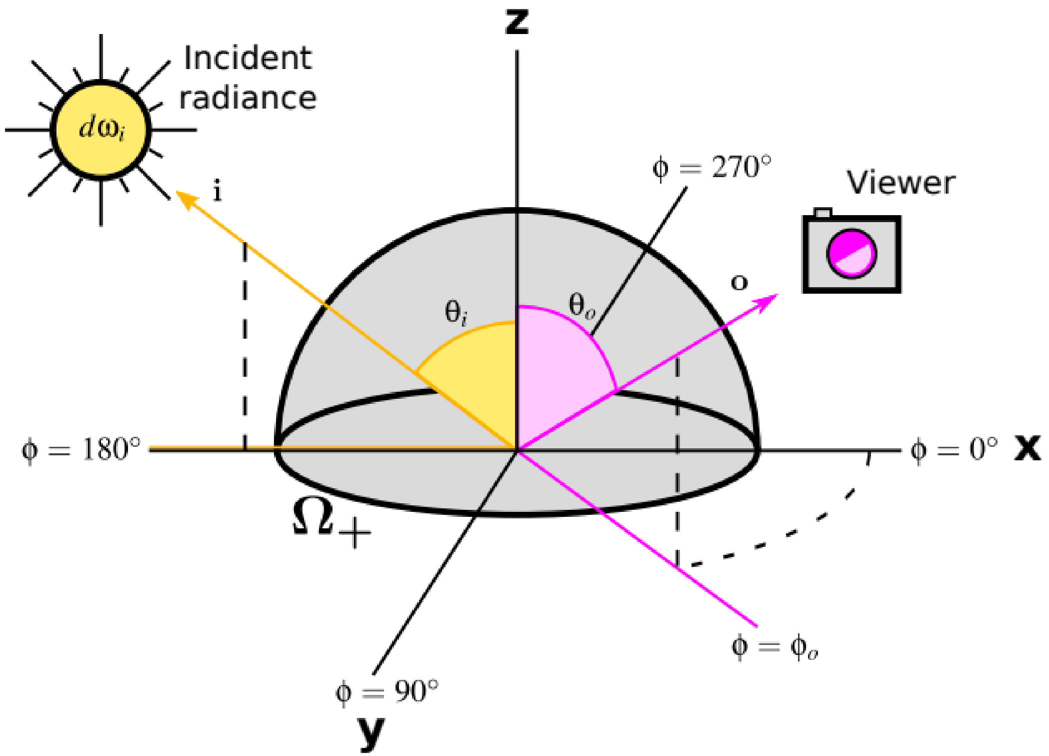


Microfacet theory

Assumption:

Single-bounce mirror reflection dominates on the microsurface

$$f_r = \frac{F(\theta_d)D(\boldsymbol{h})G(\boldsymbol{i},\boldsymbol{o})}{4\ cos\theta_i\ cos\theta_0}$$



Microfacet theory

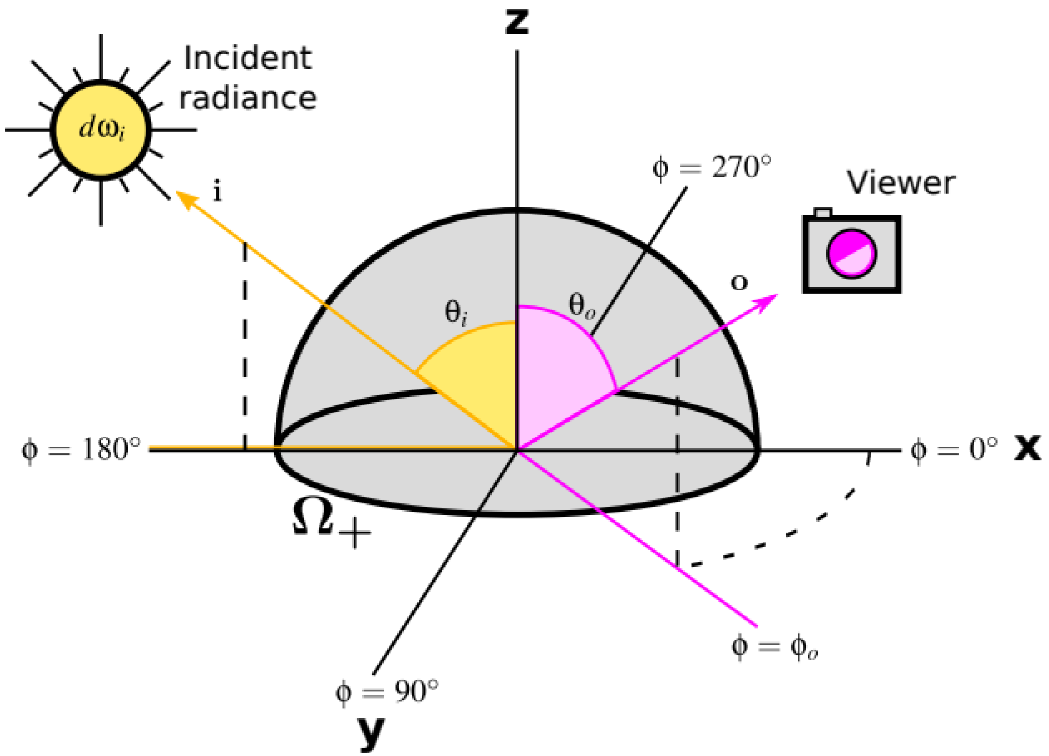
Assumption:

Single-bounce mirror reflection dominates on the microsurface

$$f_r = \frac{F(\theta_d)D(\boldsymbol{h})G(\boldsymbol{i},\boldsymbol{o})}{4\cos\theta_i \cos\theta_0}$$

Halfway vector,

$$\boldsymbol{h} = \frac{\boldsymbol{i}+\boldsymbol{o}}{||\boldsymbol{i}+\boldsymbol{o}||}$$

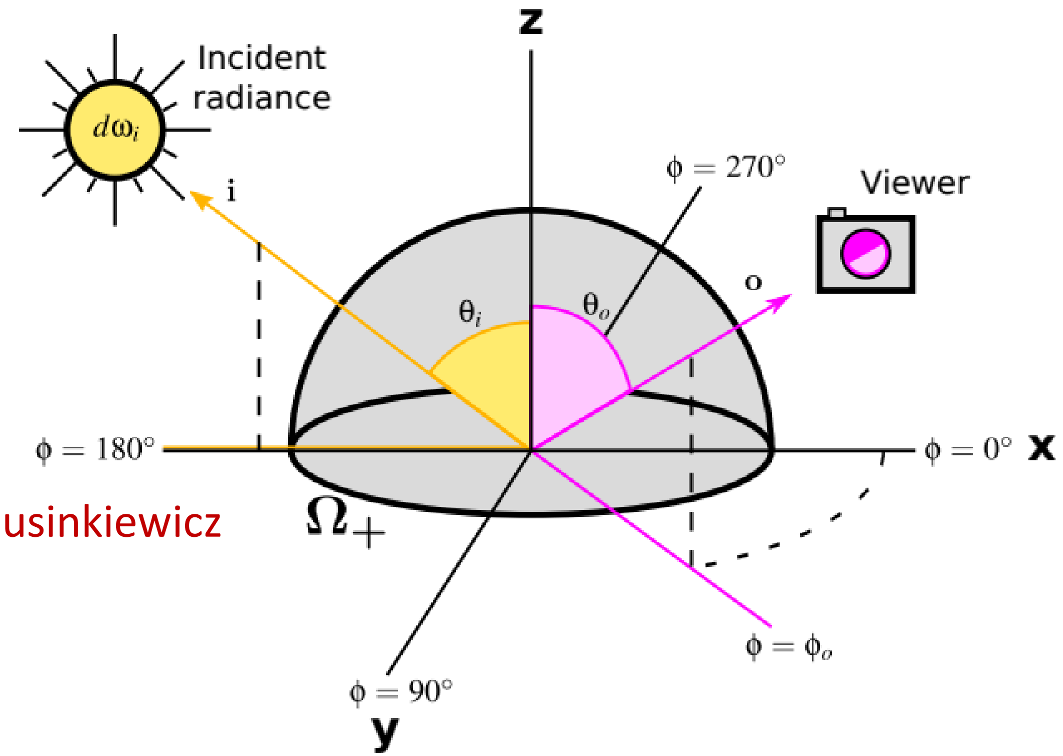


Microfacet theory

Assumption:

Single-bounce mirror reflection dominates on the microsurface

$$f_r = \frac{F(\theta_d)D(\boldsymbol{h})G(\boldsymbol{i},\boldsymbol{o})}{4\cos\theta_i\cos\theta_0}$$



θ_d = difference angle in the BRDF parameterization of Rusinkiewicz

$$\theta_d = \arccos(\mathbf{i} \cdot \mathbf{h}) \in [0, \pi/2]$$

Microfacet slopes

Goal:

Simplified the search of the NDF

Normalization constraint on the NDF:

$$\int_{\Omega_+} D(\boldsymbol{h}) \cos\theta_h \mathrm{d}\,\omega_h$$

Microfacet slopes

Goal:

Simplified the search of the NDF

Idea:

Instead of searching in the horizontal space (Ω_+), we search into the slopes space ($\mathbb{R}^2$)

In the Ω+ set, normals and slopes are linked through the bijection

$$\tilde{\boldsymbol{h}} = \begin{bmatrix} -\tan\theta h \cos\phi h = \tilde{\boldsymbol{x}}_k \\ -\tan\theta h \sin\phi h = \tilde{\boldsymbol{y}}_k \end{bmatrix}, \qquad \tilde{\boldsymbol{h}} \in \mathbb{R}^2$$

Microfacet slopes

Goal:

Simplified the search of the NDF

Idea:

Instead of searching in the horizontal space (Ω_+), we search into the slopes space ($\mathbb{R}^2$)

In the Ω+ set, normals and slopes are linked through the bijection

$$\tilde{\boldsymbol{h}} = \begin{bmatrix} -\tan\theta h \cos\phi h = \tilde{\boldsymbol{x}}_k \\ -\tan\theta h \sin\phi h = \tilde{\boldsymbol{y}}_k \end{bmatrix}, \qquad \tilde{\boldsymbol{h}} \in \mathbb{R}^2$$

Normal Distribution fonction

$$f_r = \frac{F(\theta_d)D(h)G(i,o)}{4\ cos\theta_i\ cos\theta_0}$$

$$D(\boldsymbol{h}) = P(\tilde{\boldsymbol{h}})sec^4\theta_h$$

Probability distribution function P

Normalisatio constraint:

$$\int_{\mathbb{R}^2} P(\tilde{\boldsymbol{h}})d\tilde{\boldsymbol{h}} = 1$$

Microfacet theory

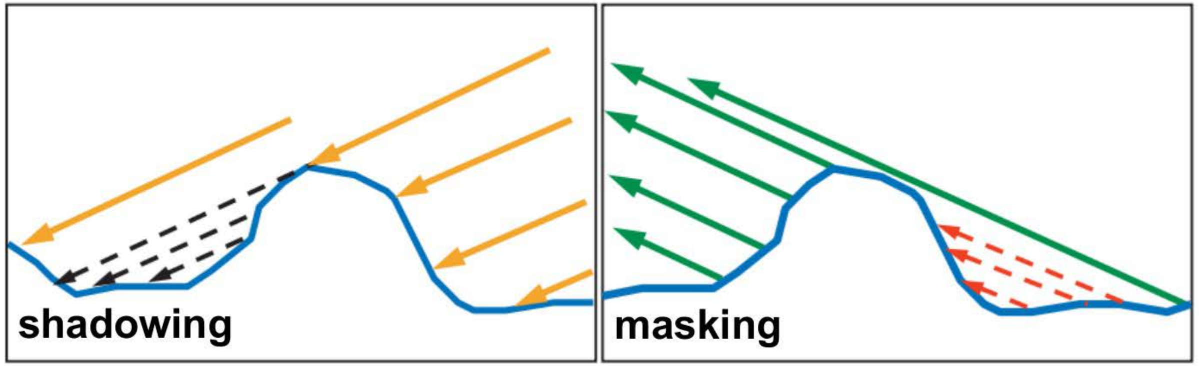
Geometric attenuation factor

$$f_r = \frac{F(\theta_d)D(h)G(i,o)}{4\ cos\theta_i\ cos\theta_0}$$

$$G(i,o) = \frac{G_1(i)G_1(0)}{G_1(i)+G_1(0)-G_1(i)G_1(0)}$$ G ∈ [0,1]

Smith monostatic shadowing function:

$$G_1(\boldsymbol{k}) = \frac{cos\theta_k}{\int_{\Omega+} \boldsymbol{kh}\ D(\boldsymbol{h})\ d\omega_k}$$ $G_1 \in [0,1]$



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

Backscattering Equation

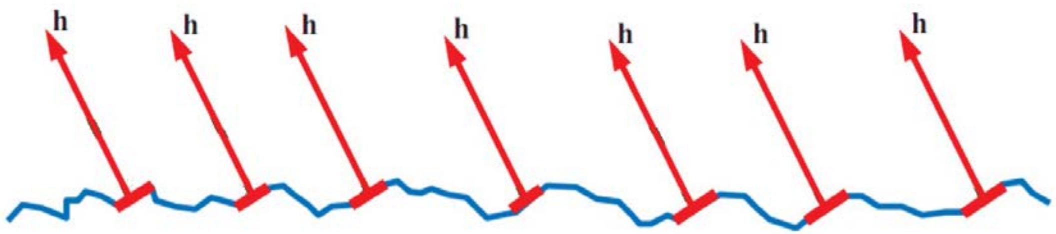
Mathematical (previous work)

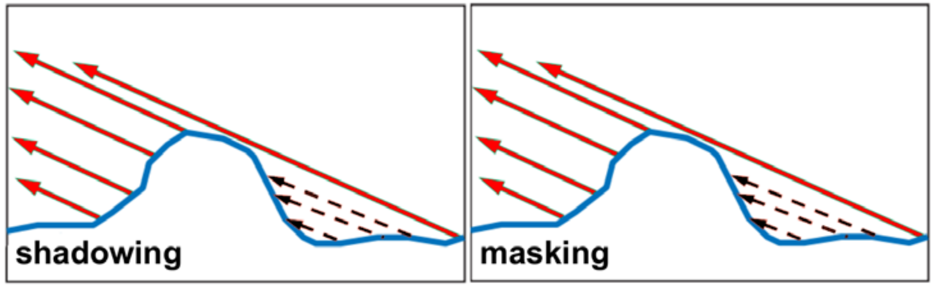
We focus on backscattering configuration which reduce the dimensionality of the BRDF

l = o = h $\theta_d = 0$

$$f_r = \frac{F_o D(\boldsymbol{o}) G(\boldsymbol{o},\boldsymbol{o})}{4\ cos^2\theta_0} \Rightarrow f_r = \frac{F_o D(\boldsymbol{o}) G_1(\boldsymbol{o})}{4\ cos^2\theta_0}$$

$$G(\boldsymbol{o},\boldsymbol{o}) = G_1(\boldsymbol{o})$$





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

Eigensystem construction

$$f_r = \frac{F_o D(o) G_1(o)}{4\ cos^2 \theta_0}$$

Inverted equation

$$F_0 P(\tilde{o}) = \int_{\Omega+} \mathrm{K(o,h)P}(\tilde{h})\ \mathrm{d}\omega_h$$

(Fredholm equation of the second kind)

$$K(\mathbf{o},\mathbf{h}) = 4 f_r(\mathbf{o},\mathbf{o}) \cos^5\theta_o\, \mathbf{oh} \sec^4\theta_h .$$

Eigensystem construction

$$f_r = \frac{F_o D(o) G_1(o)}{4\ cos^2\theta_0}$$

$$\Rightarrow F_0 P(\tilde{o}) = \int_{\Omega+} \mathrm{K(o,h)P}(\tilde{h})\ \mathrm{d}\omega_h$$

Numerically sovled by discretizing the equation with a quadrature rule

$$\Rightarrow F_0 P(\tilde{o}_i) = \sum_{j=1}^{N} w_j K(o_i, \boldsymbol{h_j}) P(\tilde{\boldsymbol{h}}_j)$$

Eigensystem construction

$$F_0 P(\tilde{o}_i) = \sum_{j=1}^{N} w_j K(o_i, \boldsymbol{h_j}) P(\tilde{\boldsymbol{h}}_j)$$

$\Rightarrow$

$$F_0 \boldsymbol{p} = \boldsymbol{K} \cdot \boldsymbol{p}$$

$$\mathbf{p} = (P(\tilde{\mathbf{o}}_1), \cdots, P(\tilde{\mathbf{o}}_N))^t$$

$$\mathbf{K} = \begin{bmatrix} w_1 K(\mathbf{o}_1, \mathbf{h}_1) & \cdots & w_N K(\mathbf{o}_1, \mathbf{h}_N) \\ \vdots & \ddots & \vdots \\ w_1 K(\mathbf{o}_N, \mathbf{h}_1) & \cdots & w_N K(\mathbf{o}_N, \mathbf{h}_N) \end{bmatrix}$$

Eigensystem construction

$$f_r = \frac{F_o D(\boldsymbol{o}) G_1(\boldsymbol{o})}{4\ cos^2 \theta_0}$$

$$\Rightarrow$$

$$F_0.\boldsymbol{p}(\tilde{\boldsymbol{o}}) = \boldsymbol{K}.\boldsymbol{p}$$

Discretize PDF vector

nonnegative matrix

Perron-Frobenius theorem
the solution is always the eigenvector
with the largest magnitude

Backscattering Equation

$$F_0 P(\widetilde{\boldsymbol{o}}) = \int_{\Omega+} \mathrm{K}(\mathbf{o},\mathbf{h})\mathrm{P}(\widetilde{\boldsymbol{h}})\, \mathrm{d}\omega_h$$

$$f_r = \frac{F_o D(\boldsymbol{o}) G(\boldsymbol{o},\boldsymbol{o})}{4 \cos^2 \theta_0}$$

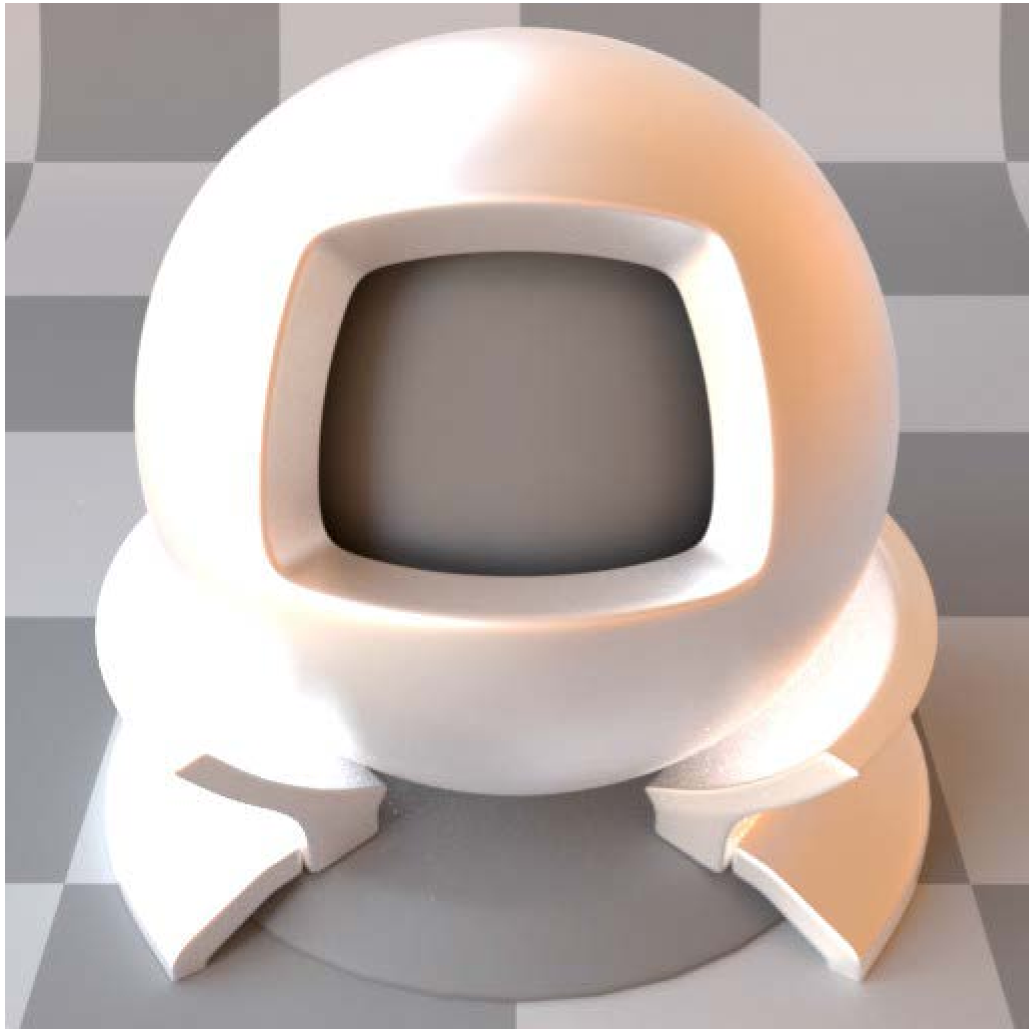
Algorithm 1 Extract P

function EXTRACT_$P(f_r, N)$
  for each $i, j \in [1, N]$ **do** ▷ Build kernel matrix
    $K_{i,j} \leftarrow w_j 4 f_r(\mathbf{o}_i, \mathbf{o}_i) \cos^5 \theta_{o_i}\, \mathbf{o}_i \mathbf{h}_j \sec^4 \theta_{h_j}$
  end for
  $\mathbf{p} \leftarrow (1, \cdots, 1)^t$
  for $0 \leq i < M$ **do** ▷ Power iterations (we set $M = 4$)
    $\mathbf{p} \leftarrow \mathbf{K} \cdot \mathbf{p}$
  end for
  $P \leftarrow \text{normalize}(\mathbf{p})$
end function

Ideal Mirrors

Special microfacet BRDF:
- Fresnel term is 1
- Independent of wavelength

$$f_{r,id} = \frac{D(\boldsymbol{o})G_1(\boldsymbol{o})}{4\,cos\theta_0\,cos\theta_i}$$

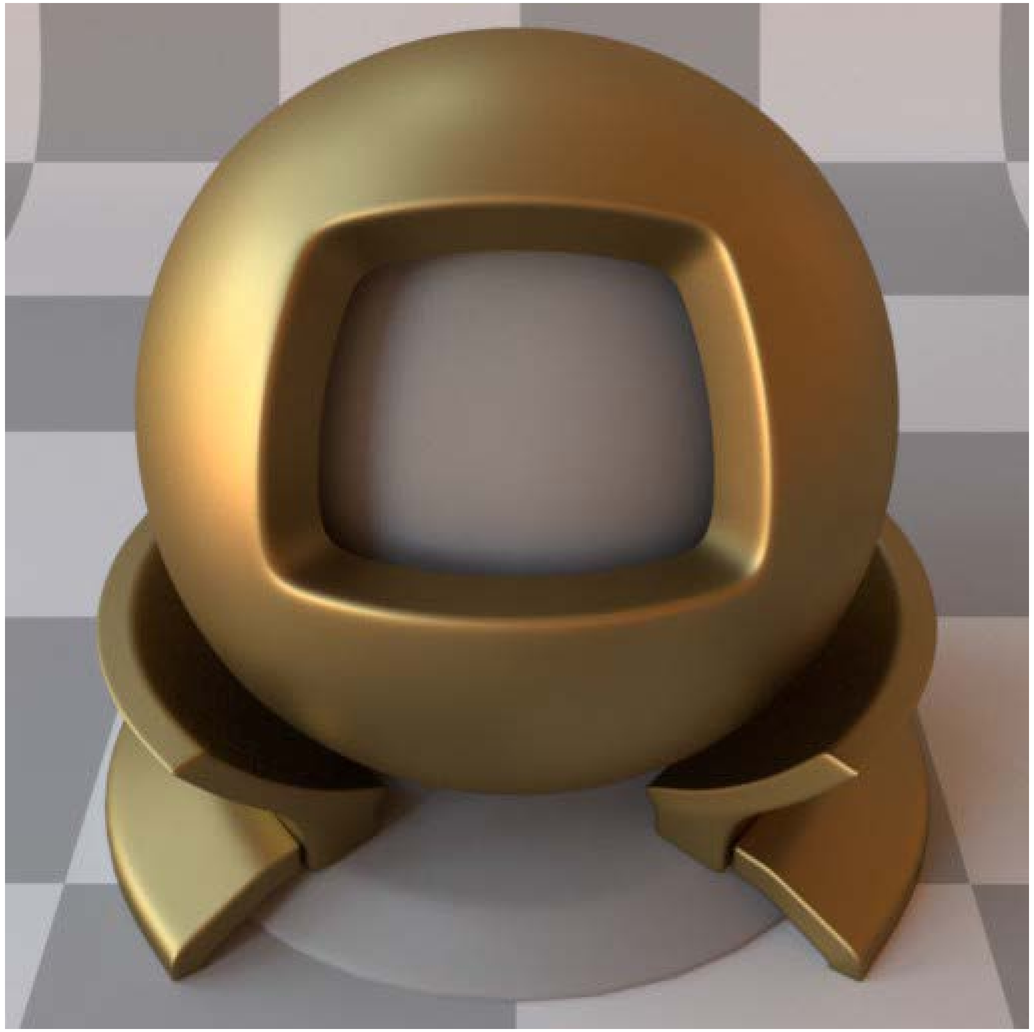


Fresnel Extraction

We compute an average response :

- Fully automatic
- Simple implementation
- Fast evaluation
- Works well in practice

$$F(\theta_d) = \mathbb{E}\left[\frac{f_r}{f_{r,id}}\middle|\boldsymbol{ih} = cos\theta_0\right]$$



Fresnel Extraction

We compute an average response :

- Fully automatic
- Simple implementation
- Fast evaluation
- Works well in practice

$$F(\theta_d) = \mathbb{E}\left[\frac{f_r}{f_{r,id}}\middle|\boldsymbol{ih} = cos\theta_0\right]$$

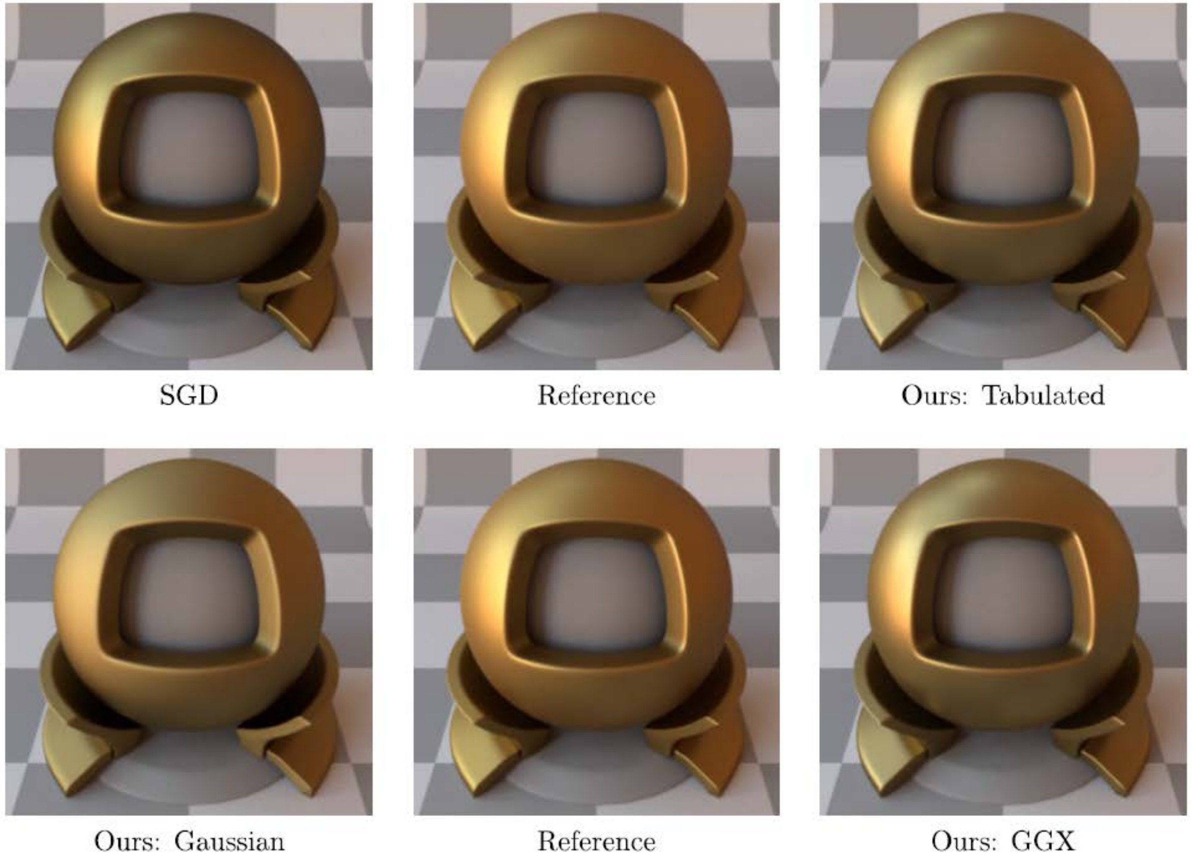
Algorithm 2 Extract F

```
function EXTRACT_F(f_r, f_r,id)
    for θ_d ∈ [0, π/2] do
        F(θ_d) ← 0
        N ← 0
        for φ_d, φ_h ∈ [0, 2π], θ_h ∈ [0, π/2] do
            i ← from_half_diff(h, d)
            o ← reflect(i, h)
            F(θ_d) ← F(θ_d) + f_r(i, o)/f_r,id(i, o)
            N ← N + 1
        end for
        F(θ_d) ← F(θ_d)/N
    end for
end function
```

Validation

29 gold-metallic-paint

Renderings

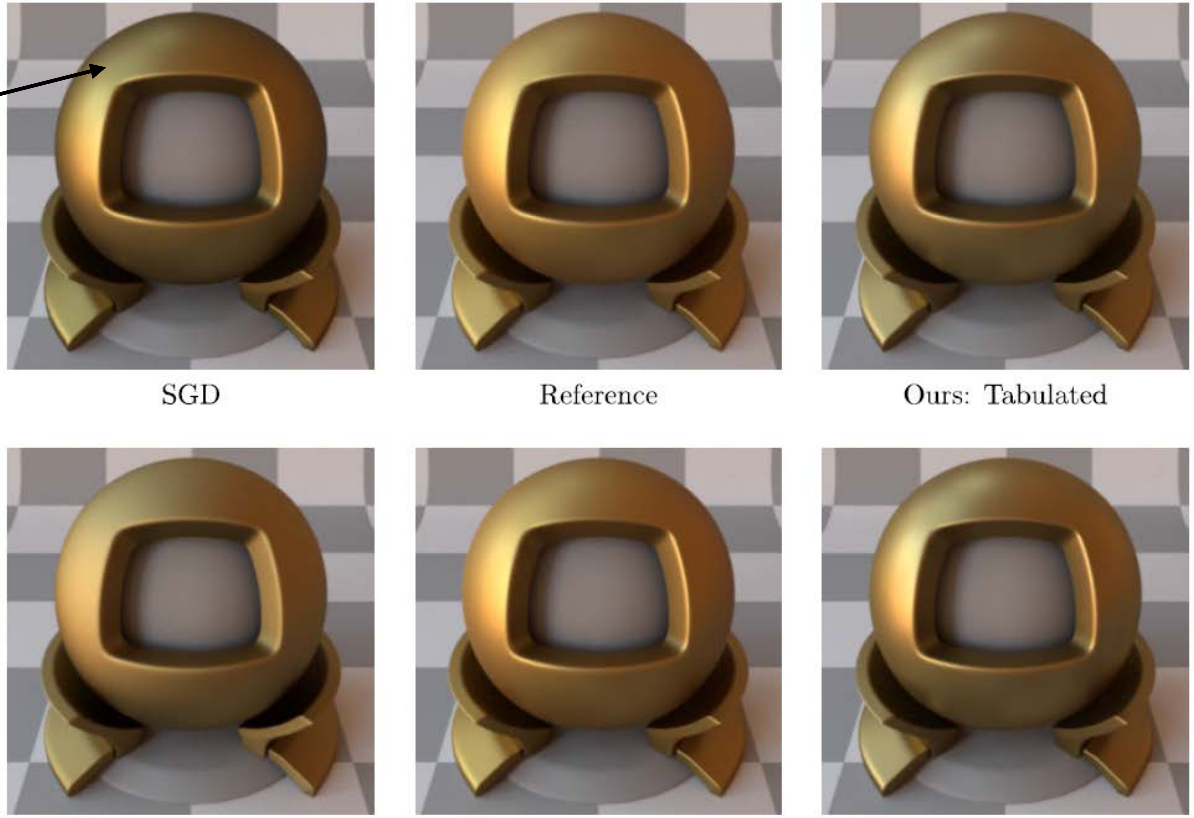


Validation

SGD fitting optimization failed and resulted in flawed images

29 gold-metallic-paint

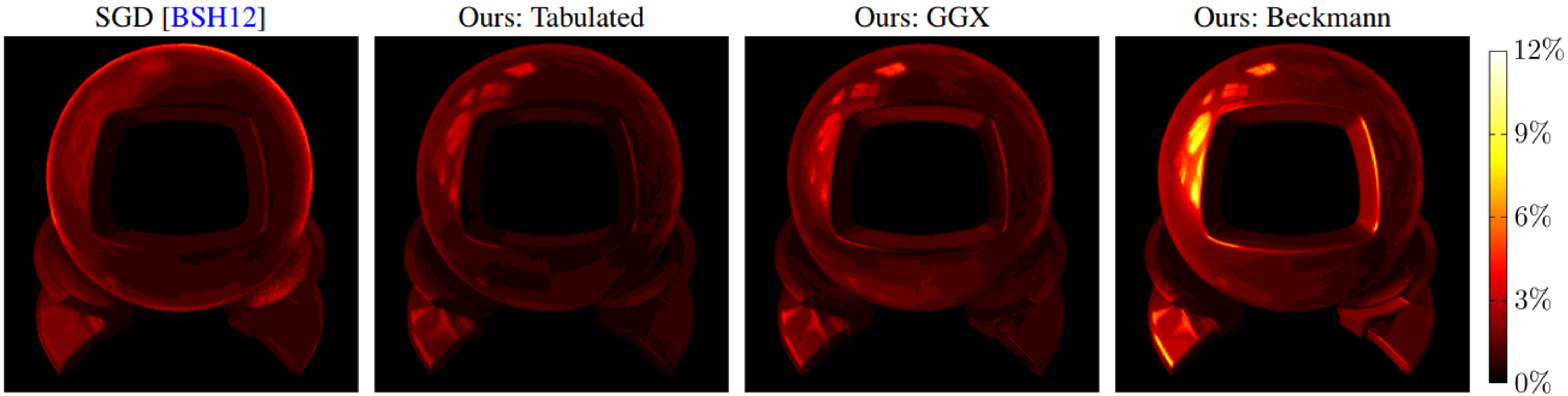
Renderings



SGD | Reference | Ours: Tabulated

Ours: Gaussian | Reference | Ours: GGX

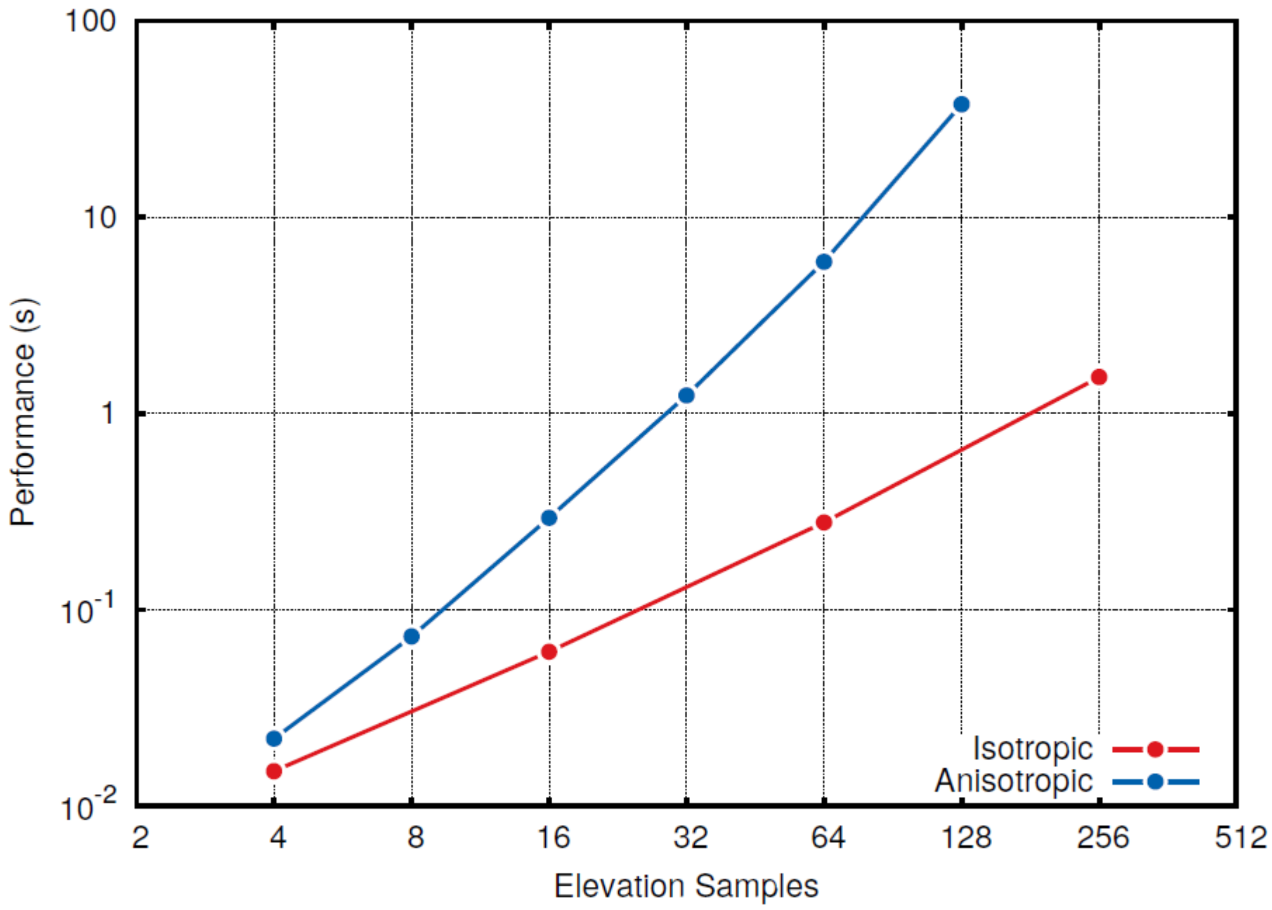
Accuracy



Mean delta-E difference image on the MERL database

Speed

(Intel i5-2500 @ 3.30 GHz)

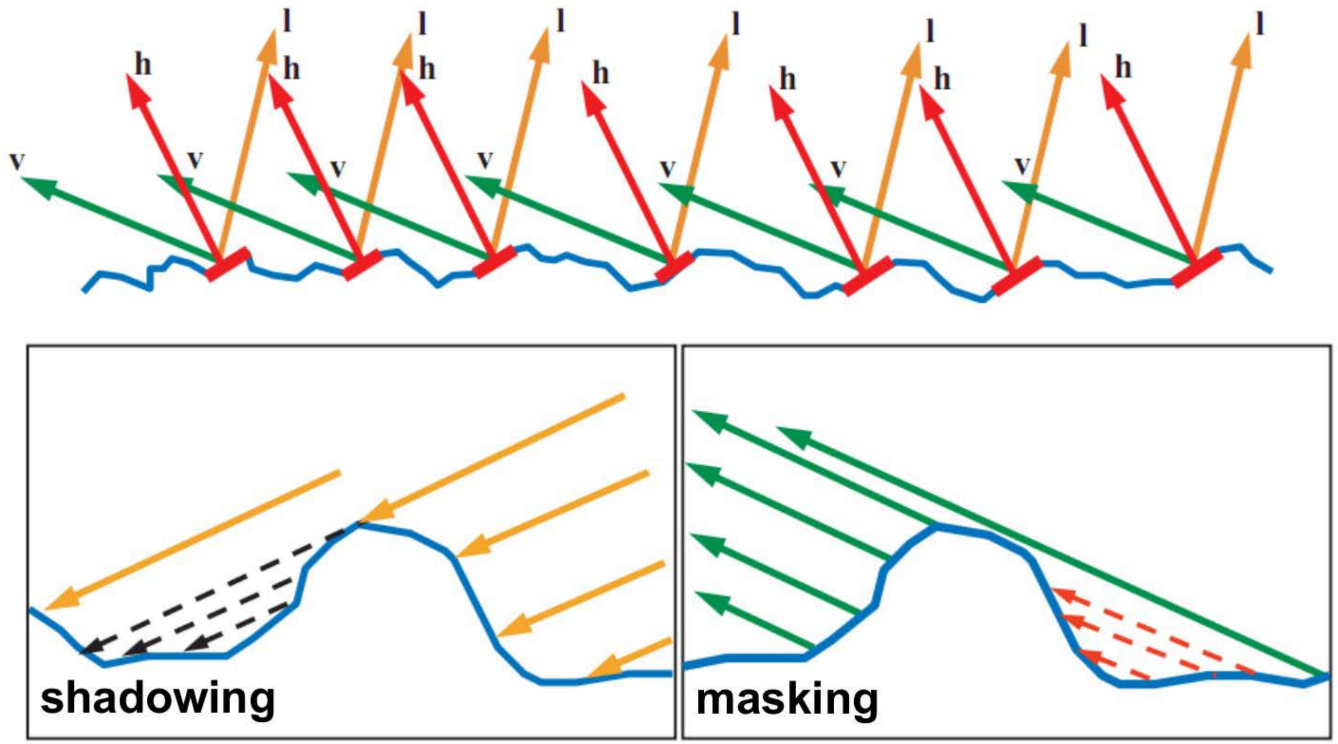


Microfacet BRDFs

$$f_r = \frac{F \cdot D \cdot G}{4 \cdot \cos\theta_i \cdot \cos\theta_o}$$

Modular components

- Fresnel term F
- Distribution of normals D
- Roughness α
- Geometric factor G



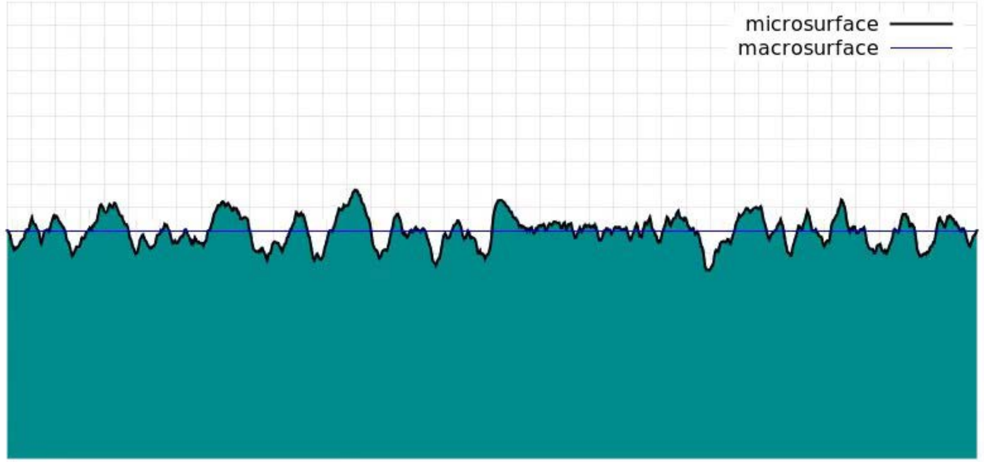
Images from "Real-Time Rendering, 3rd Edition", A K Peters 2008

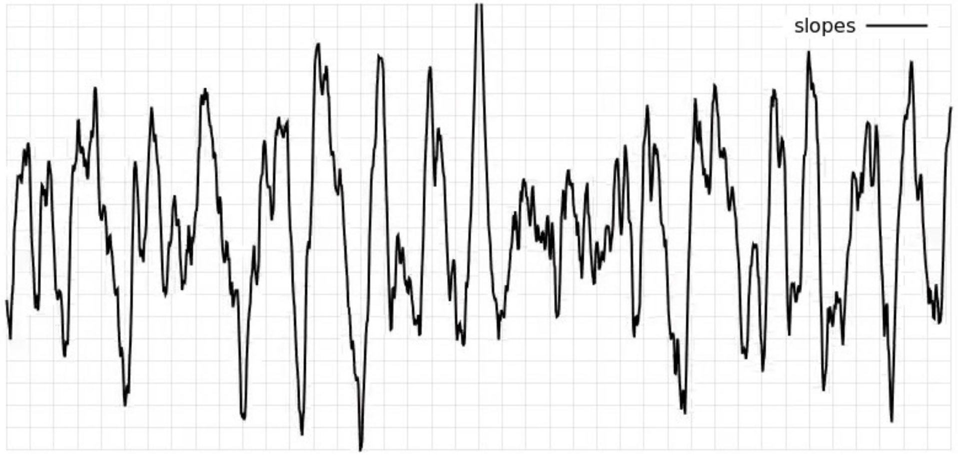
Artistic Control

$$D = P\left(\frac{\tilde{x}_h}{\alpha_x}, \frac{\tilde{y}_h}{\alpha_y}\right) \frac{\sec^4\theta_h}{\alpha_x\alpha_y}$$

Tabulate the slope PDF

- Roughness $\propto$ stretch^{-1}
- Efficient BRDF evaluation
- Efficient BRDF sampling





Fast Global Illumination with Discrete Stochastic Microfacets Using a Filterable Model

Beibei Wang Lu Wang Pierre Poulin Nicolas Holzschuch

Pacific Graphics 2018

Goal

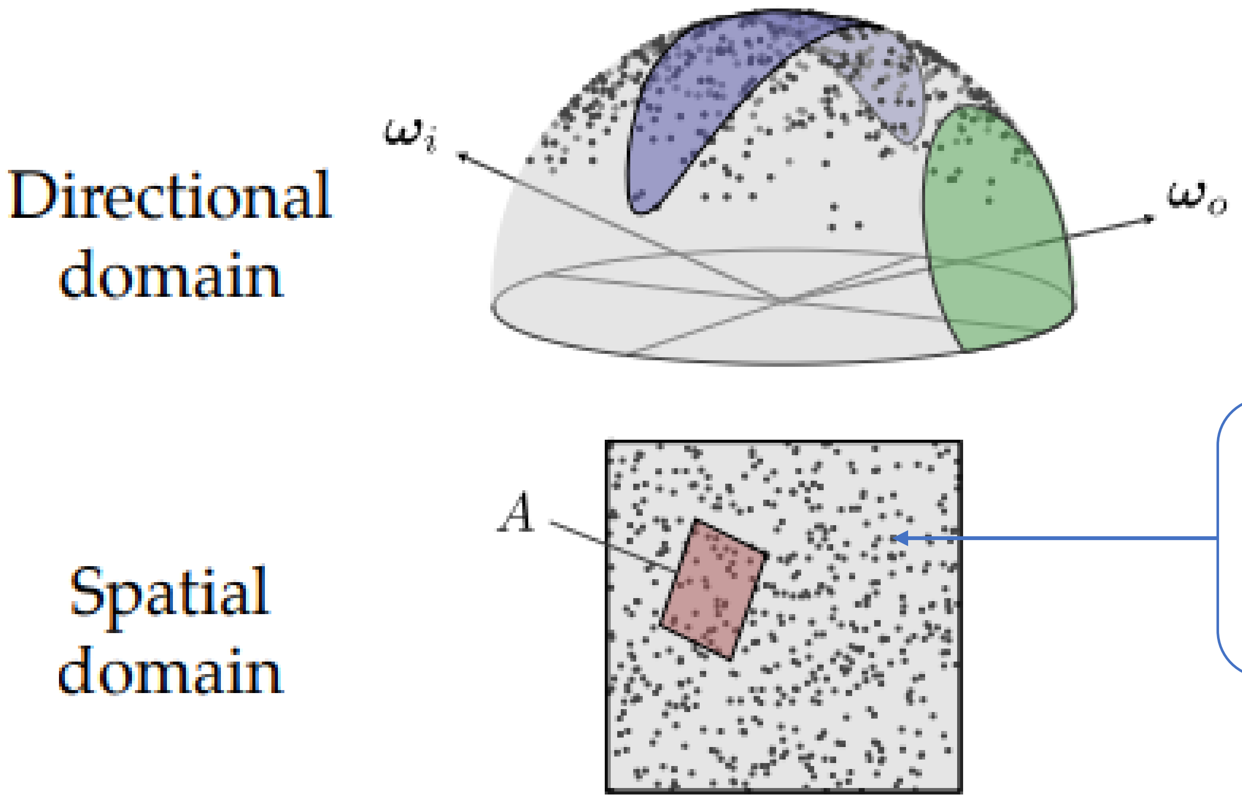
Original

Without Glints

Previous work

Jakob et al.

Discrete Stochastic Microfacet Models

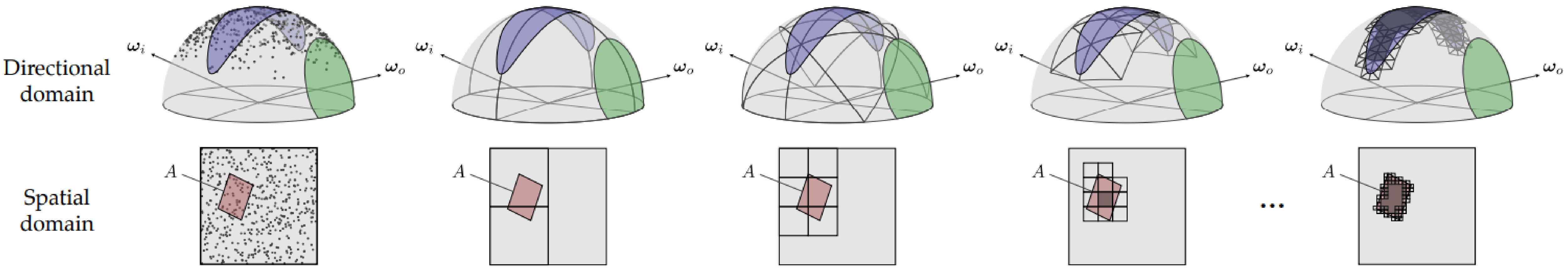


Flakes-like small mirror facets

Previous work

Jakob et al.

Discrete Stochastic Microfacet Models



These specular patches are organized in a hierarchy

Discrete Stochastic Microfacet Model

Extend the microsurface BRDF model to take into account a finite extend in space and angle

$$\hat{f}_r(A, \omega_i, \Omega_0) = \frac{1}{a(A)\sigma(\Omega_0)} \int_A \int_{\Omega_0} f_r(x, \omega_i, \omega_o) d\omega_o dx$$

A = finite area around x
Ω_0 = finite solid angle around outgoing dirrection ω_o

$$f_r(x, \omega_i, \omega_o) = \frac{F(\omega_i, \omega_o) D(x, \omega_h) G(\omega_i, \omega_o, \omega_h)}{4 \cos\theta_i \cos\theta_0}$$

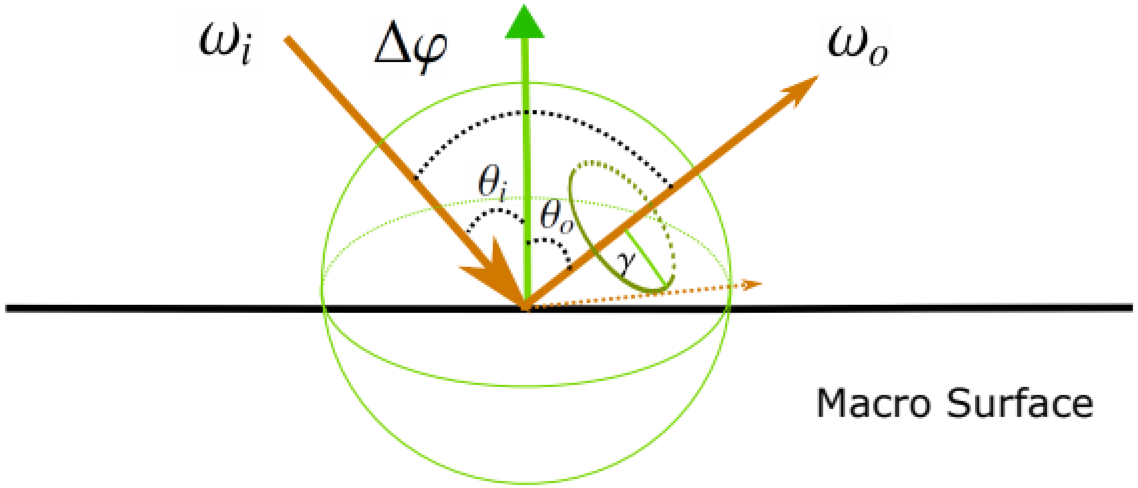
Discrete Stochastic Microfacet Model

We now assume the surface is made of discrete small mirror particles instead of a continuous microfacets

$$\hat{f}_r(A, \omega_i, \Omega_0) = \frac{1}{a(A)\sigma(\Omega_0)} \int_A \int_{\Omega_0} f_r(x, \omega_i, \omega_o) d\omega_o dx$$

$$\Rightarrow$$

$$\hat{f}_r(A, \omega_i, \Omega_0) = \frac{(\omega_i \cdot \omega_h)\, F(\omega_i, \omega_o) \widehat{D}(x, \omega_h) G(\omega_i, \omega_o, \omega_h)}{a(A)\sigma(\Omega_0) 4 (\omega_i \cdot n)(\omega_i \cdot n)}$$



$$\widehat{D}(x, \Omega_h) = \frac{1}{N} \sum_i^N 1_{\Omega_h}(\omega_h^k) 1_A(x^k)$$

Sum of a finite set of particles

A = pixel footprint

Discrete Stochastic Microfacet Model

$\widehat{D}(x,\Omega_h)=\frac{1}{N}\sum_i^N 1_{\Omega_h}(\omega_h^k)1_A(x^k)$

Sum of a finite set of particles

A = pixel footprint

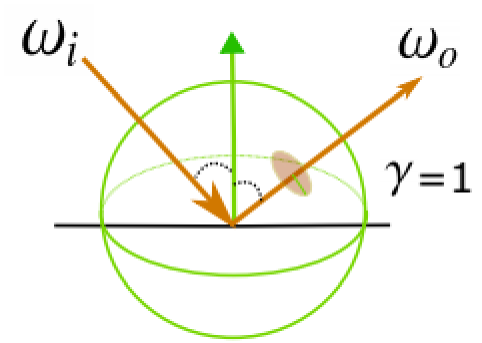
4 dimensional normal distribution

Count the number of particles in the 4 dimensional domain $A \times \Omega_h$

Filterable Discrete Stochastic Microfacet Model

Replace the particle count with a particle probability function
Introduce a Directional Probability Function (DPF)

$P(\omega_i, \omega_o, \gamma)$ = probability a particle exist that reflect lights incoming from direction ω_i into a coe centered around direction ω_o with half angle γ



x and (ωi , ωo) are independant variables

Filterable Discrete Stochastic Microfacet Model

Replace the particle count with a particle probability function
Introduce a Directional Probability Function (DPF)

$$P(\omega_i, \omega_o, \gamma) \approx \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{\Omega_h}(\omega_h^k)$$

$$\hat{D}(A, \omega_i, \omega_o) = \left(\frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_A(\boldsymbol{x}^k)\right) P(\omega_i, \omega_o, \gamma)$$

x and (ωi , ωo) are independant variables

$$\hat{D}(A, \omega_i, \omega_o) = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_A(\boldsymbol{x}^k) H\left(\lambda(\boldsymbol{x}) - P(\omega_i, \omega_o, \gamma)\right)$$

$$H(u) = \begin{cases} 1, & \text{if } u > 0, \\ 0, & \text{otherwise.} \end{cases}$$

Filterable Discrete Stochastic Microfacet Model

$$\hat{D}(A, \omega_i, \omega_o) = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_A(\boldsymbol{x}^k) H(\lambda(\boldsymbol{x}) - P(\omega_i, \omega_o, \gamma))$$

$$H(u) = \begin{cases} 1, & \text{if } u > 0, \\ 0, & \text{otherwise.} \end{cases}$$

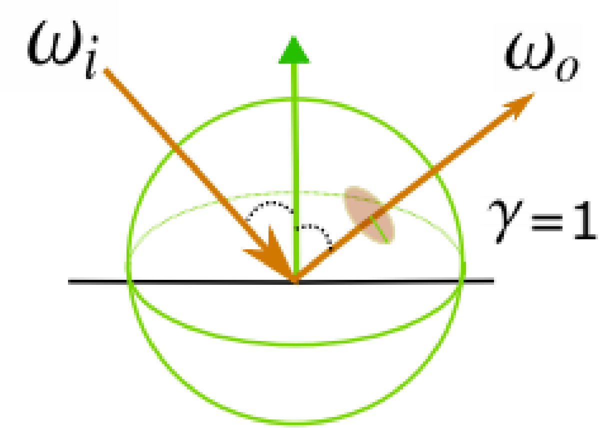
$\lambda(x)$ = uniformly Distributed random value from Tiny Encription Algorithm

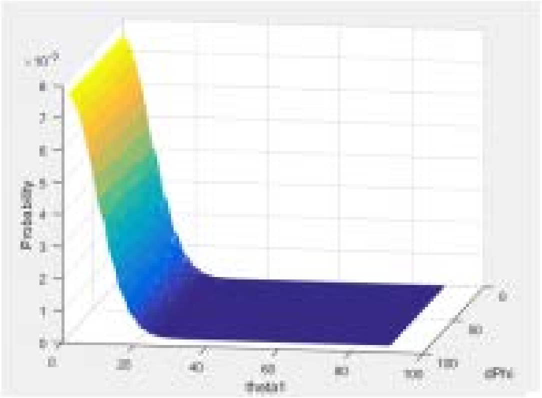
$\lambda(x)$ is used to assign a random value between 0 and 1 to each positio x

Filterable Discrete Stochastic Microfacet Model

Peprocess

compute and store P for $\gamma = 1^{\circ}$, by sampling regularly in each dimension, computing and storing the value for P.





Filterable Discrete Stochastic Microfacet Model

Peprocess

- Build a hierarchical representation of the Directional Probability Function, using Gaussian blur
- Each level is generated by blurring the finest level
- 9 hierarchical levels, for 1° , 2° , 5° , 10° , 20° , 30° , 45° , 60° and 90°

Filterable Discrete Stochastic Microfacet Model

2 steps hierachical traversal

- Angle
- Space

Look up in the precomputed table;
Select the approriate hierachical level;
Extract the prcomputed value for P

Algorithm 1 Spatial Traversal

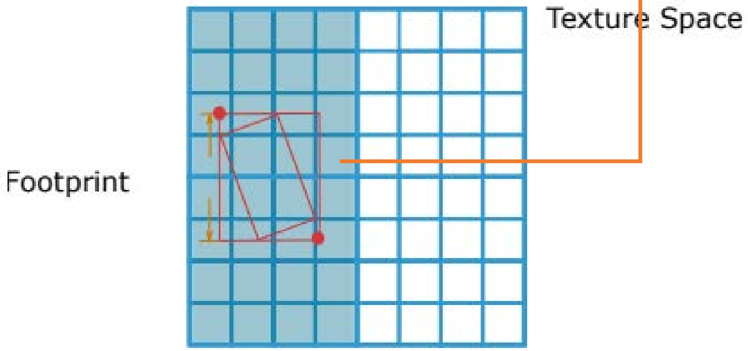
function $\hat{D}_s(A, \omega_i, \omega_o)$
    query $\leftarrow A$
    queue $\leftarrow node(N_{\text{start}})$
    count $\leftarrow 0$
    p $\leftarrow P(\omega_i, \omega_o)$
    while queue $\neq \varnothing$ **do**
        node $\leftarrow$ queue.pop()
        if node $\cap$ query $== \varnothing$ or $|$node$| = 0$ **then pass**
        else if node $\subseteq$ query **then** $n_k \leftarrow |$node$|$
            for i $< n_k$ **do** $\psi \leftarrow \lambda(k)$
                if $\psi > p$ **then** count $\leftarrow$ count + 1
                end if
            end for
        else if error criterion satisfied **then**
            overlap $\leftarrow$ (node $\cap$ query).vol()/node.vol()
            $n_k \leftarrow |$node$|$
            for i $< n_k$ **do** $\psi \leftarrow \lambda(k)$
                if $\psi > p$ **then** count $\leftarrow$ count + overlap
                end if
            end for
        else
            for c in node.split() **do** queue.push(c)
            end for
        end if
    end while
    return count
end function

Filterable Discrete Stochastic Microfacet Model

2 steps hierachical traversal

- Angle
- Space

Traversal in texture space;
Assume flakes are uniformly distributed in space;



Algorithm 1 Spatial Traversal

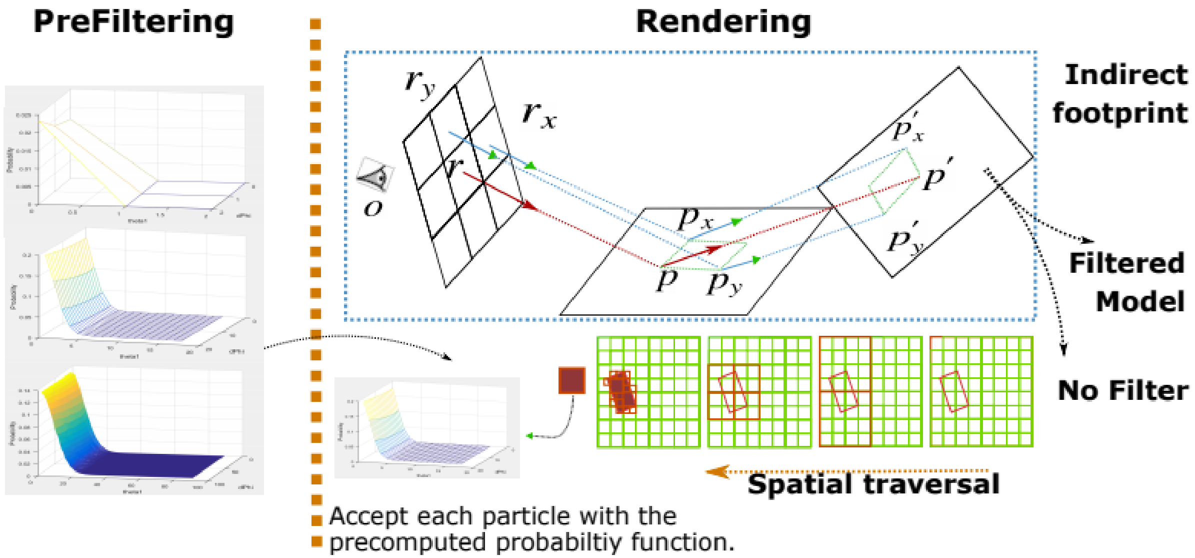
function $\hat{D}_s(A, \omega_i, \omega_o)$
  query $\leftarrow A$
  queue $\leftarrow$ *node*(N_{start})
  count $\leftarrow 0$
  p $\leftarrow P(\omega_i, \omega_o)$
  while queue $\neq \emptyset$ **do**
    node $\leftarrow$ queue.pop()
    if node $\cap$ query $== \emptyset$ or $|\text{node}| = 0$ **then pass**
    else if node $\subseteq$ query **then** $n_k \leftarrow |\text{node}|$
      for i $< n_k$ **do** $\psi \leftarrow \lambda(k)$
        if $\psi > p$ **then** count $\leftarrow$ count + 1
        end if
      end for
    else if error criterion satisfied **then**
      overlap $\leftarrow$ (node $\cap$ query).vol()/node.vol()
      $n_k \leftarrow |\text{node}|$
      for i $< n_k$ **do** $\psi \leftarrow \lambda(k)$
        if $\psi > p$ **then** count $\leftarrow$ count + overlap
        end if
      end for
    else
      for c in node.split() **do** queue.push(c)
      end for
    end if
  end while
  return count
end function

Filterable Glint Computation

- For each path, compute the path footprint at each light bounce
- Compute the average glint contribution at this footprint

$$\hat{D}(A, \omega_i, \omega_o) = a(A) \times P(\omega_i, \omega_o, \gamma),$$

- If the glint count is larger than a given threshold, use the average contribution.
- Otherwise, use the separable model described



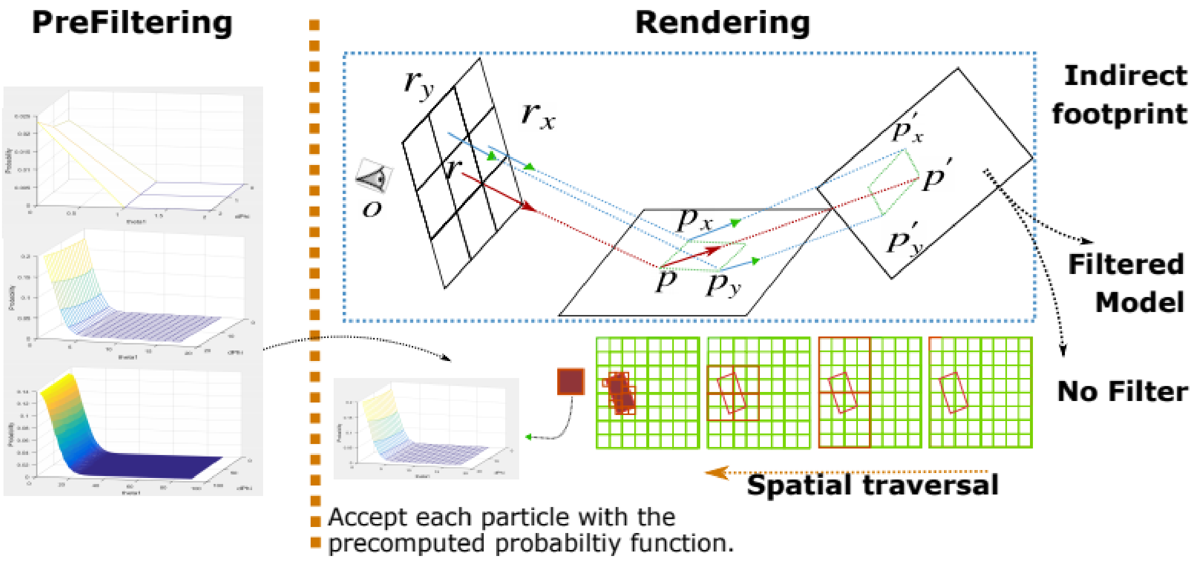
Discrete Stochastic Microfacets Using a Filterable Model

Idea:

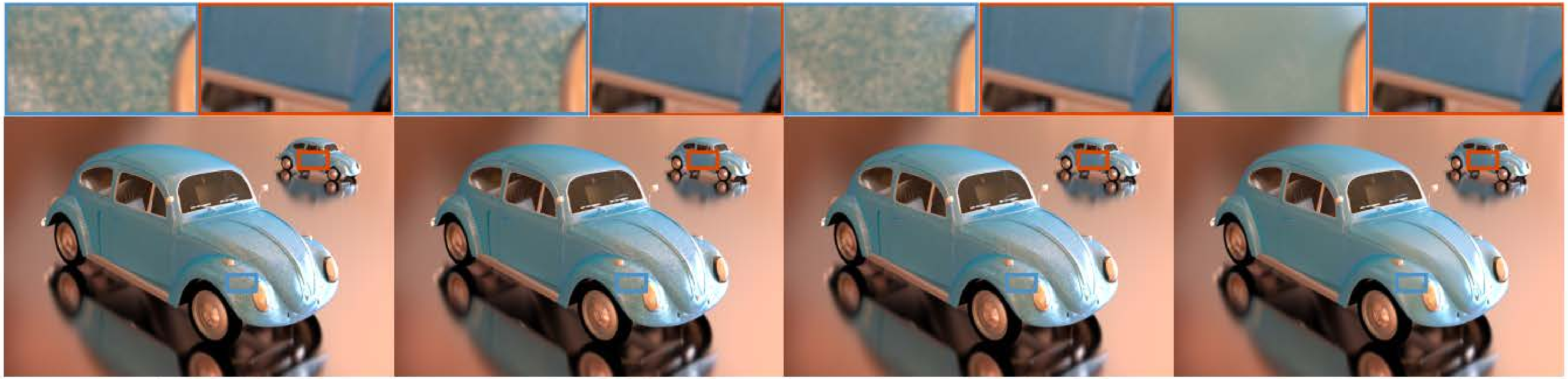
If a footprint cover a large surface,
- individual glint are not noticeable;
- average contribution

In practice:

Filterable model preafer for
- material far from camera;
- Several bounce in global illumination



Validation



Our model (with filtering)
Total: 4.6 min, Cost: +21%

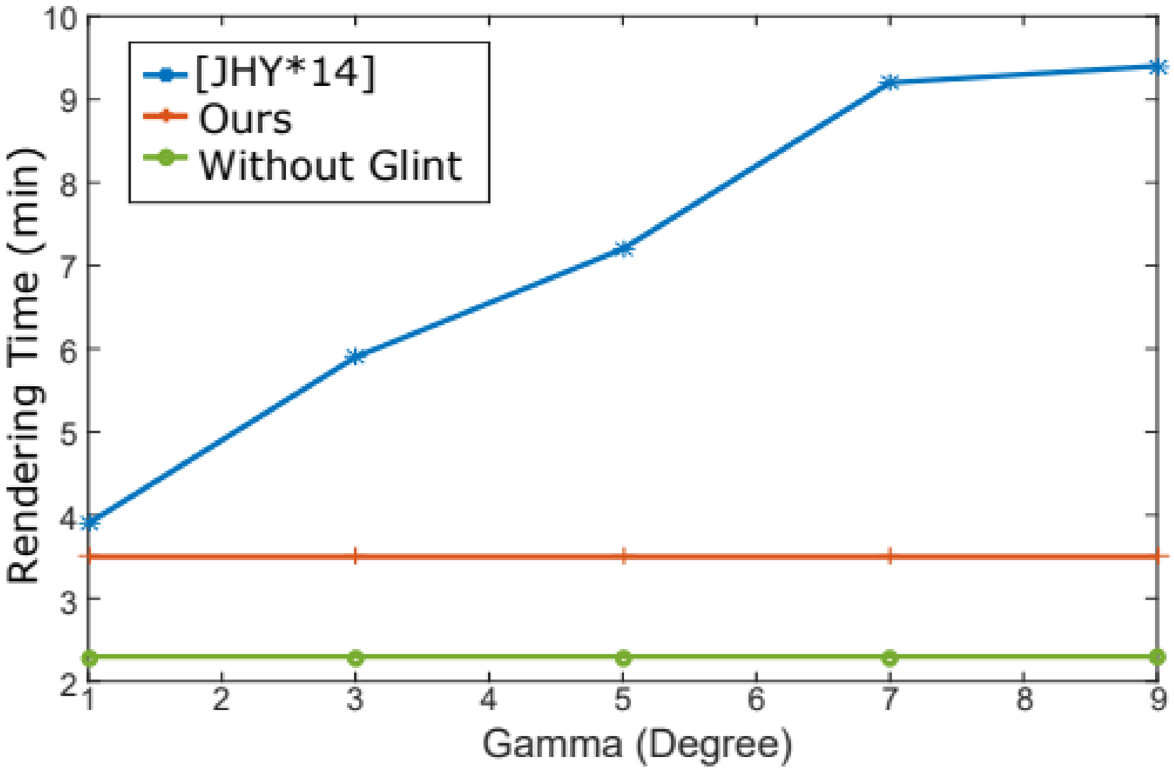
Our model (no filter)
Total: 7.4 min, Cost: +95%

*Original [JHY*14],*
Total: 24.4 min, Cost: +542%

Without Glints,
Total: 3.8 min

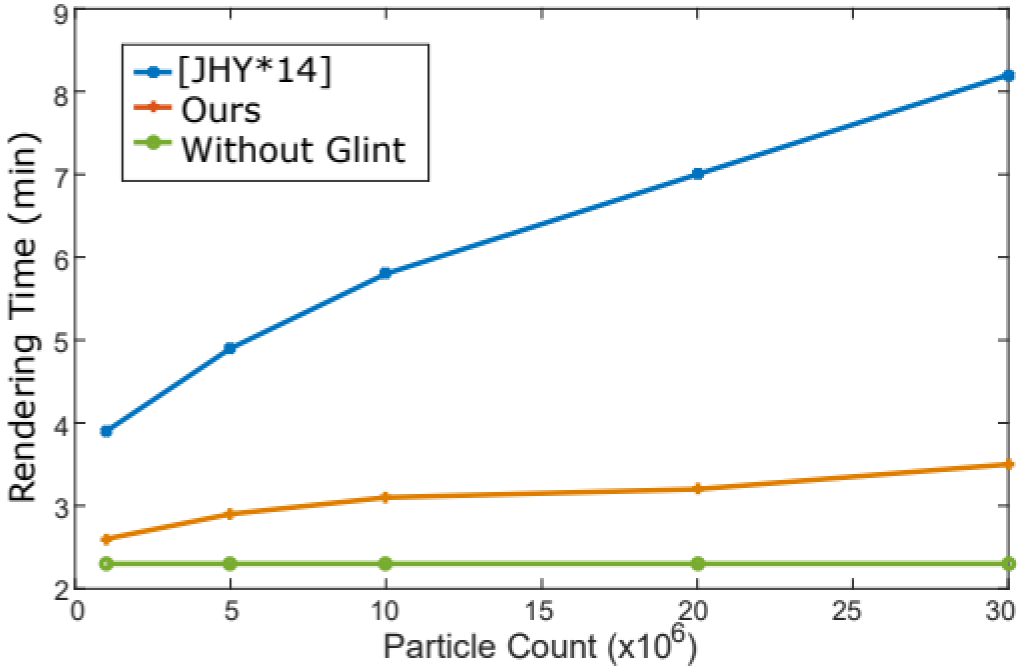
Speed

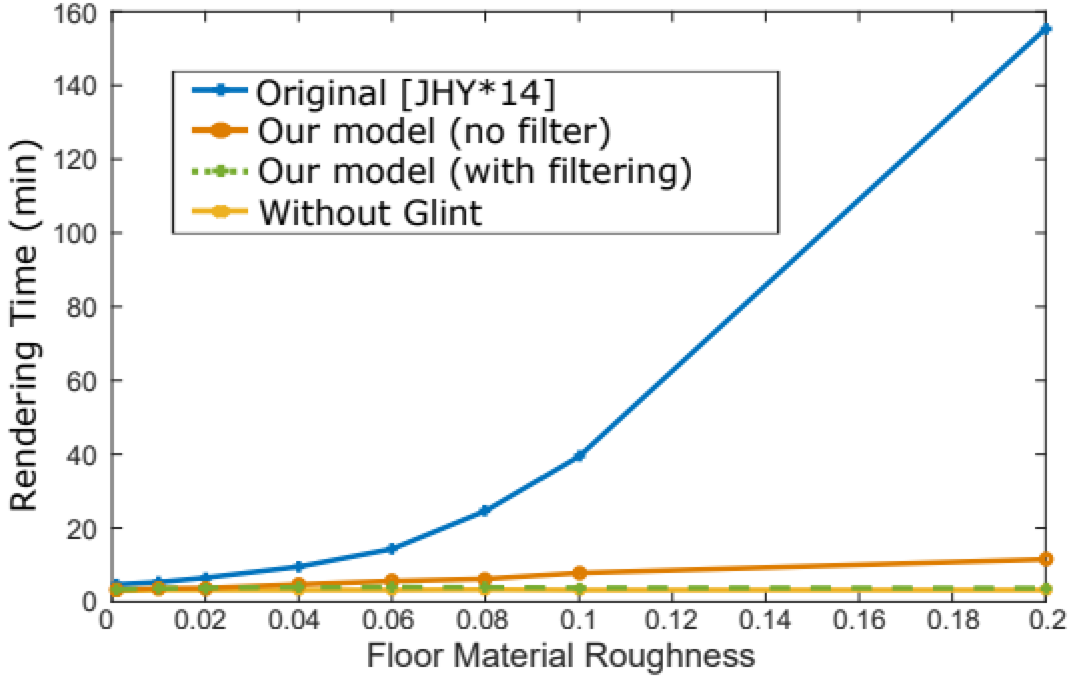
(Intel i7 (40 cores) with 32 GB of main memory @ a 2.20GHz)



Speed

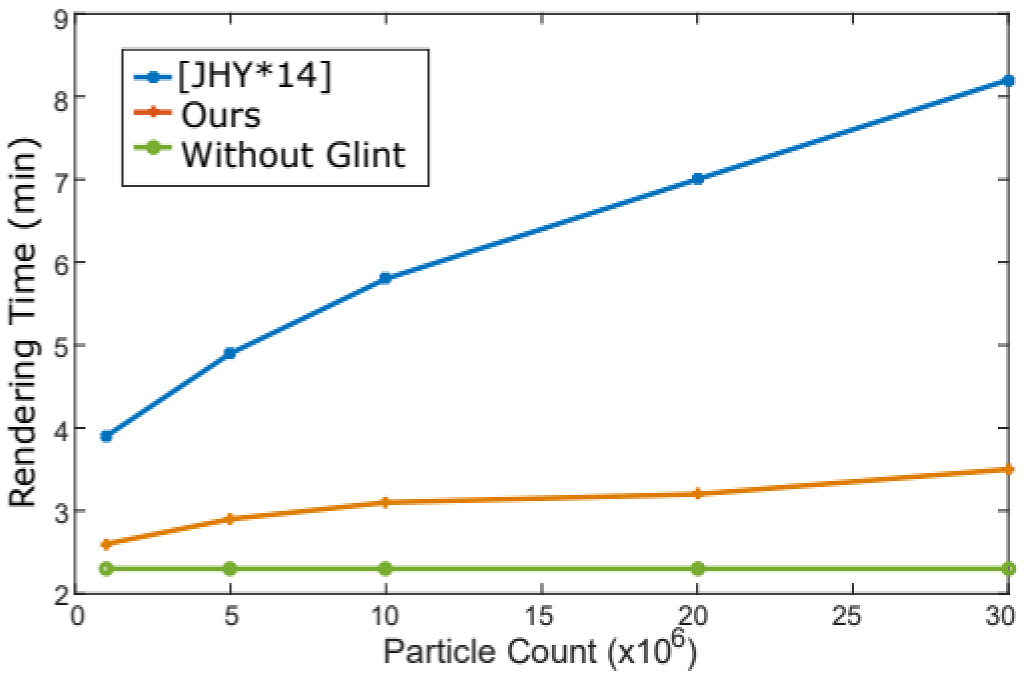
(Intel i7 (40 cores) with 32 GB of main memory @ a 2.20GHz)

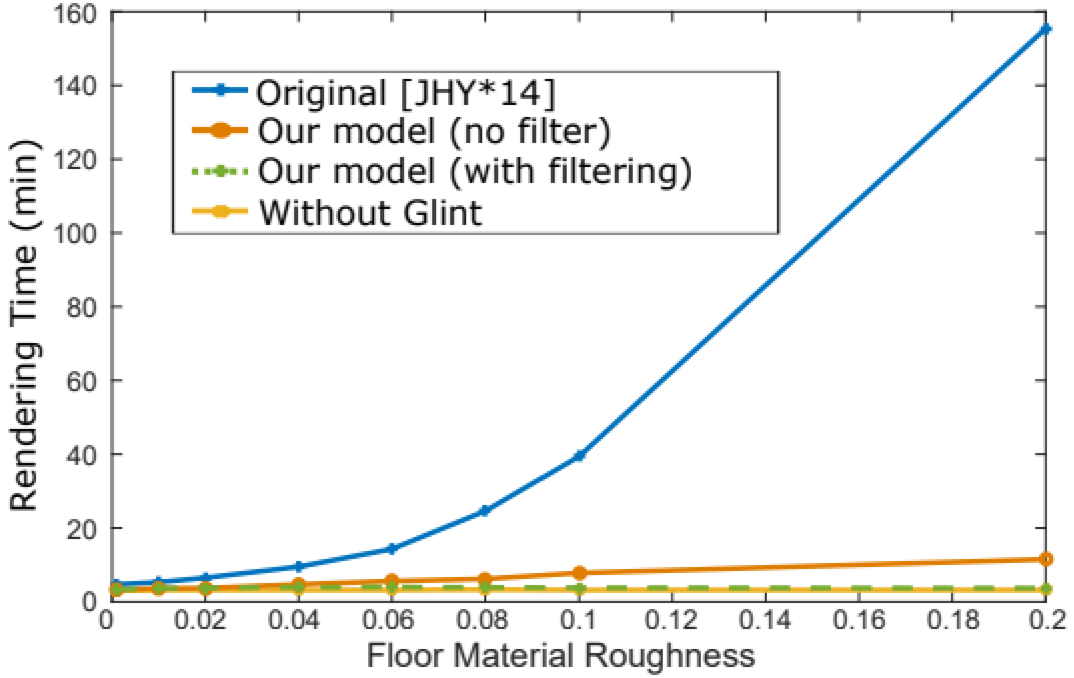




Speed

(Intel i7 (40 cores) with 32 GB of main memory @ a 2.20GHz)





Quiz

- In ‘Extracting Microfacet-based BRDF Parameters from Arbitrary Materials with Power Iterations’ , a bijection is used to link (   ) space and the slopes space (   ) .

  a. Ω_d and $\mathbb{R}^4$
  b. Ω_+ and $\mathbb{R}^2$
  c. $\mathbb{R}^2$ and Ω_+
  d. $\mathbb{R}^2$ and Ω_d

- In ‘Fast Global Illumination with Discrete Stochastic Microfacets Using a Filterable Model’, which variables do we consider for indépendance?

  a. x and (ω_i, ω_o)
  b. x and γ
  c. γ and (ω_i, ω_o)
  d. ω_h and γ