Real-Time Rendering of Wave-Optical Effects on Scratched Surfaces

Zdravko Velinov¹, Sebastian Werner¹, Matthias Hullin¹

¹ University of Bonn, Germany
Motivation

• Iridescent scratches defined by wave-optical phenomena appear on many everyday items
Related Work

• Scratched surfaces

• Area lighting
Goals

• Area lighting of worn surfaces covered by iridescent scratches

• Anti-aliasing of scratches based on camera pixel footprint

• Everything in real-time with single sample per pixel both in spatial and angular domain
Scratch Iridescence Model
Combined Surface BRDF Model

• Superposition of scratch S and masked base BRDF $\mathcal{B}$ - parameterized by the direction cosine $\xi$, spatial position $\mathbf{x}$

$$f(\xi, \mathbf{x}) = \frac{1}{\pi \sigma^2} |\mathcal{B}(\xi) - S(\xi, \mathbf{x})|^2$$

• Base BRDF in this case is limited to the smooth coherent model

• Scratch BRDF is defined according to [WVJH17]
Incoherent superposition

- Arbitrary base BRDF is enabled by neglecting interference

\[ f(\xi, \mathbf{x}) \approx \frac{1}{\pi \sigma^2} \left[ |B(\xi)|^2 \rho - |S(\xi, \mathbf{x})|^2 - |S(\xi, \mathbf{x})|^2 \right] \]

- Masking is performed according to the scratch density \( \rho \)

- Separates the base from the scratch response!
Scratch BRDF

- Defined by width function $W$ across the scratch, depth function $D$ and longitudinal integral term $\eta$
  
  $S(\xi, \mathbf{x}) = W(\xi) \cdot D \cdot \eta(\xi, \mathbf{x})$
  
  $= W(\xi) \cdot D \cdot \eta_a(\xi) \cdot \eta_s(\mathbf{x})$

- We use the small angles approximation
  
  $D \approx 1 - e^{-i4\pi d/\lambda}$

- Enables separability in angular domain!
Simplified integration in angular domain

• The separable BRDF in angular domain simplifies integration significantly

\[ L = \int_{\Omega^+} f(\xi) L_i \, d\xi \]

\[ \xi = \omega_{i,t} + \omega_{o,t} \]

• The projected outgoing direction \( \omega_{o,t} \) acts as offset of the light source projection defined by the projected incident direction \( \omega_{i,t} \)
Approximation motivation

- Response by three scratches in direction cosine domain
Spherical Light Sources
Spherical Light Source

1. *Check* for sphere underneath horizon
2. *Project* disk in direction cosine domain
3. *Intersect* horizon arc, if needed
4. *Intersect* projected ellipse
5. *Assemble* line segment
Approximation Comparison

a) Monte Carlo

b) Ours (Approximate Si)

c) Exact analytic approximation
Polygonal Light Sources
Polygonal Light Source Algorithm

1. *Clip* (split) triangle to the upper hemisphere
2. *Project* triangle into direction cosine domain as arcs
3. *Intersect* arcs with reflected band
4. *Cull* points outside of a triangle
5. *Sort* intersection points with bitonic sort and assemble line segments
6. *Evaluate* integral and superimpose on *Linearly Transformed Cosines (LTC)* [HDHN16]
Approximation Comparison

a) Monte Carlo

b) Ours (Approximate Si)

c) Exact analytic approximation
Anti-Aliasing and Scratch Density
Anti-Aliasing and Base Masking

• Taking the limit and correcting for small pixel footprint is good enough to approximate the integral in spatial domain

\[ \alpha = \min \left( \frac{1}{2 A_C}, \frac{A_\phi}{2 A_C}, 1 \right) \]

\[ |\eta_\phi|^2 \approx \alpha \int_{-\infty}^{+\infty} \int_{s_1}^{s_2} |\eta_s|^2 \, dx \, dy + (1 - \alpha) A_\phi |\eta_s|^2 \]

• Scratch density is similarly approximated

\[ \rho_\phi \approx \frac{2}{A_\phi} \sum_m W^{(m)} l^{(m)}_{\text{contained}} \]
Anti-Aliasing and Base Masking (Error)
Anti-Aliasing

No Anti-Aliasing  Anti-Aliasing (Ours)  Monte Carlo (Box)
Implementation and Results
Data Structure

- Bounding Volume Hierarchy – threaded BVH with skip pointers [Smi98] – simplified compared to the DAG data structure from original paper [WVJH17]

- Per Triangle Array
  - Traversed by using PrimitiveID from Visibility Buffer
  - Requires covering scratches associated with nearby triangles that fall within the coherence area
  - Similarly applicable in regular path tracing

Real-Time Rendering of Wave-Optical Effects on Scratched Surfaces, Velinov et al.
## Data Structure (Benchmark)

Benchmark was performed on NVIDIA GTX 970M

<table>
<thead>
<tr>
<th>Type</th>
<th>AA disabled</th>
<th>AA enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BVH 0.5x</td>
<td>1x</td>
</tr>
<tr>
<td>Zoom-In</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Intersection Only</td>
<td>4.1ms</td>
<td>4.4ms</td>
</tr>
<tr>
<td>Sphere</td>
<td>4.5ms</td>
<td>6.3ms</td>
</tr>
<tr>
<td>Triangle</td>
<td>13.9ms</td>
<td>26.5ms</td>
</tr>
</tbody>
</table>

Real-Time Rendering of Wave-Optical Effects on Scratched Surfaces, Velinov et al.
Interactive editing with complete model
Summary

• Improved performance several orders of magnitude compared to [WVJH17] and enabled real-time performance
  • Polygonal and sphere light sources with potential for generalizing to more simple shapes
  • Anti-aliasing through approximate integration in spatial domain
  • More performant specialized data structure which are also compatible with original model

• Properties of the original model are preserved by approximations
Thank you

Source Code:
Ground truth
Complete BRDF + Light Source Line Integral

\[ f(\xi, x) \approx \frac{1}{\pi \sigma^2} \left( (1 - \rho) |B(\xi)|^2 - |S(\xi, x)|^2 \right) \]

\[ S(\xi | q_1, q_2) = W^{(m)} D^{(m)} \eta_s^{(m)} \left( \mathcal{N}^{(m)}(\xi, q_1) - \mathcal{N}^{(m)}(\xi, q_0) \right) \]

\[ \mathcal{N}^{(m)}(\xi, q) = \frac{\sqrt{\pi}}{\sigma \pi} \left( 2 \frac{\text{Si}(kW^{(m)}q)}{kW^{(m)}} - 4 \frac{\sin^2(kW^{(m)}q/2)}{k^2W^{(m)}q^2} \right) \]