Stochastic Light Culling for VPLs on GGX Microsurfaces

YUSUKE TOKUYOSHI (SQUARE ENIX CO., LTD.)

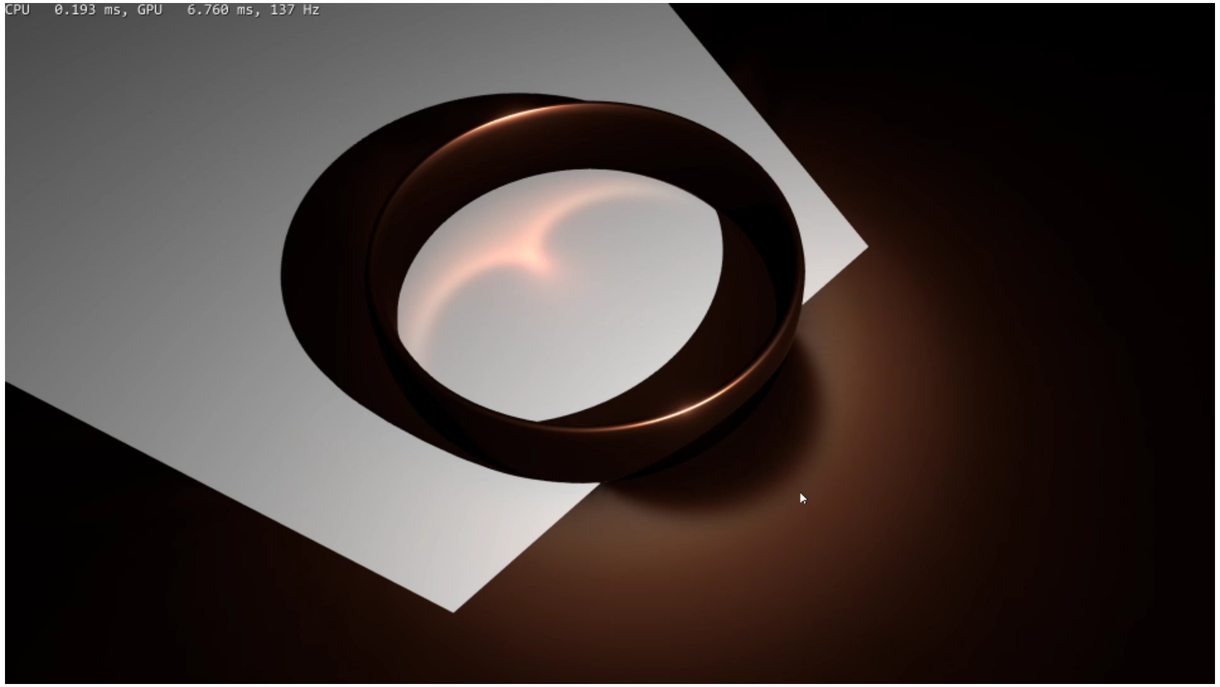
TAKAHIRO HARADA (ADVANCED MICRO DEVICES, INC.)

Virtual Point Lights (VPLs) [Keller97]

- Represent indirect illumination
- VPLs should be sampled according to the contribution of light for each shading point
- Light culling for real-time rendering
  - Splatting[Dachsbacher06], tiled culling [Stewart15], clustered shading [Olsson12, Ortegren16]

Challenge: glossy caustics at real-time frame rates

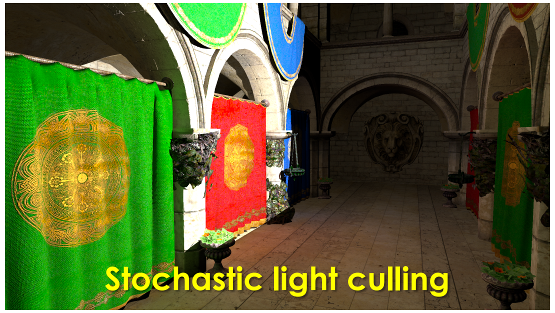
Single-bounce Glossy Caustics (65536 VPLs)



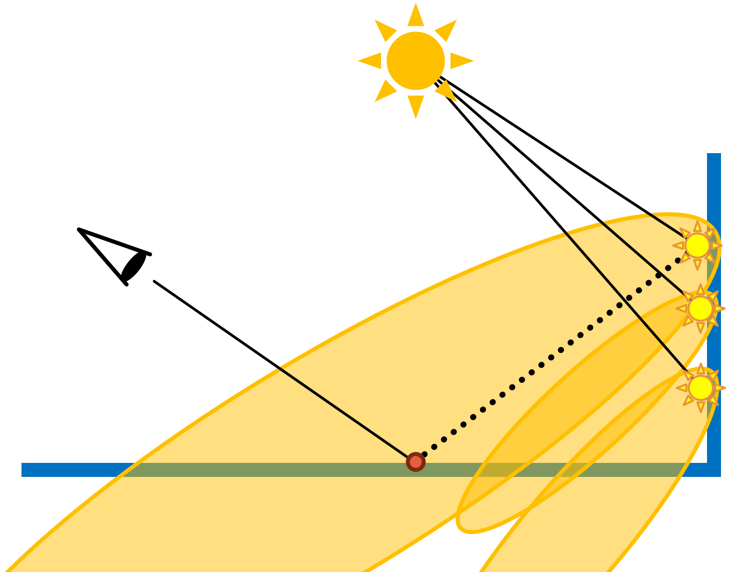
1920×1080 screen resolution, GPU: NVIDIA® GeForce® GTX 970

Stochastic Light Culling for VPLs [Tokuyoshi16]

- Unbiased culling based on *Russian roulette*
  - Avoid darkening bias for light culling

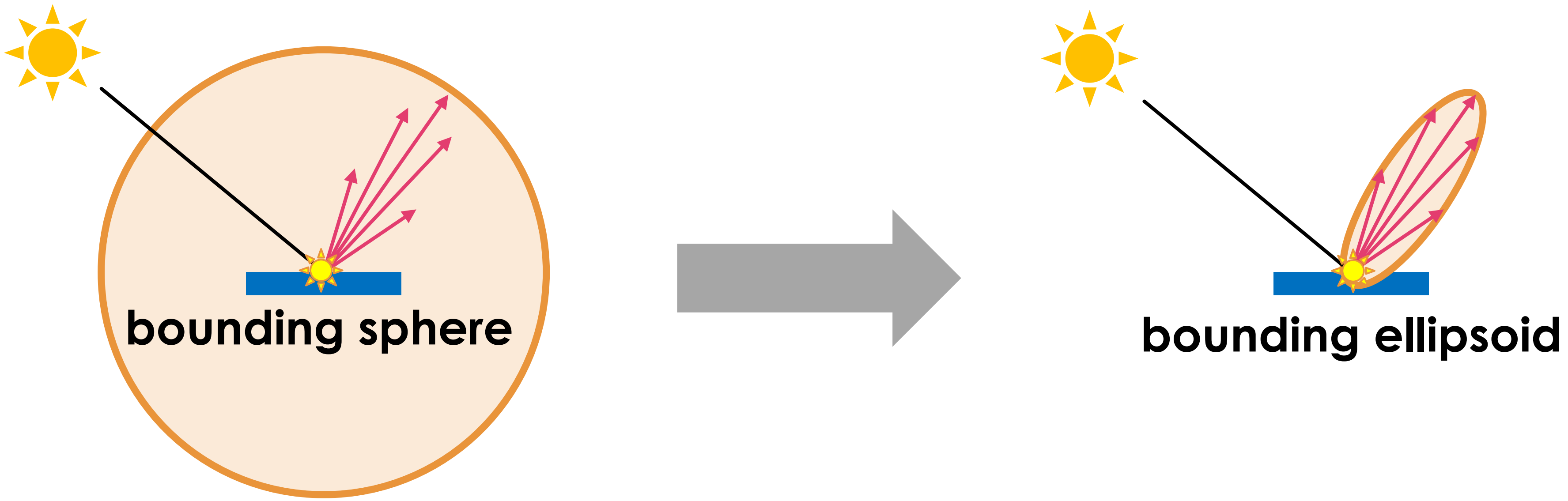


- Algorithm:
  - (1) Random light range based on Russian roulette for each VPL
  - (2) Culling using a bounding volume of that range
    - E.g., tiled culling for bounding spheres
  - (3) Shading with Russian roulette



Bounding volume should be tight to reduce false positives

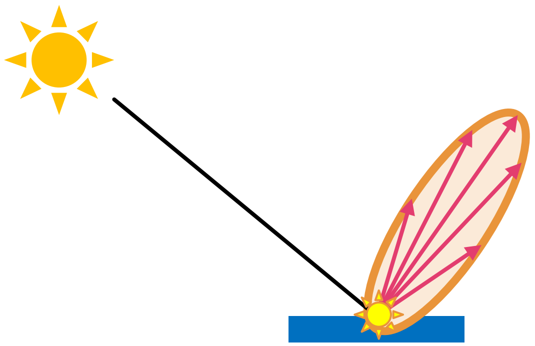
Tighter Bounding Ellipsoid [Dachsbacher06]



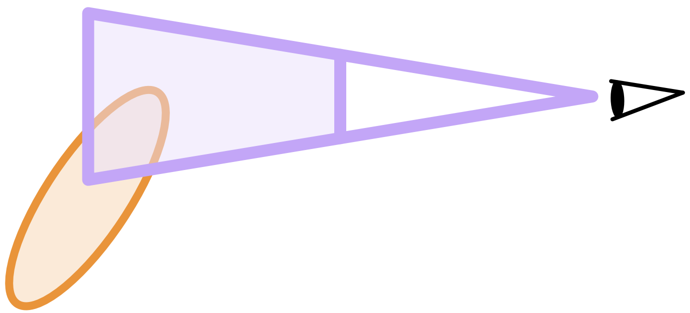
- Can be used for the classic Phong reflection model [Phong75]
- Cannot be applied to physically plausible materials

Our Contributions

- Tighter bounding ellipsoid for the GGX microfacet BRDF [Walter07]
  - Simple analytical calculation
- Efficient tiled culling implementation for bounding ellipsoids



Derivation of the bounding ellipsoid



Tiled culling for bounding ellipsoids (rough intersection test)

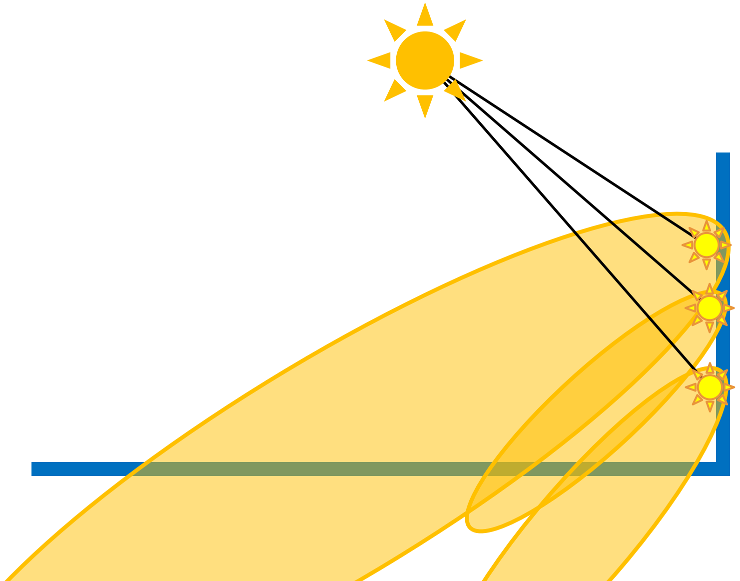
Bounding Ellipsoid for GGX Reflection

Random Light Range

- Russian roulette according to the radiant intensity × fall-off for each VPL
- Single random number for each VPL
- Can bound the light range in an unbiased fashion
- Random sampling of an isosurface of the reflected radiance
  - Inverse of the probability of Russian roulette

$$l_{max}(\boldsymbol{\omega}_o) = p^{-1}(\xi) = \sqrt{\frac{\Phi f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \max(\boldsymbol{\omega}_o \cdot \mathbf{n}, 0)}{\delta \xi}}$$

uniform random number $\in [0,1)$



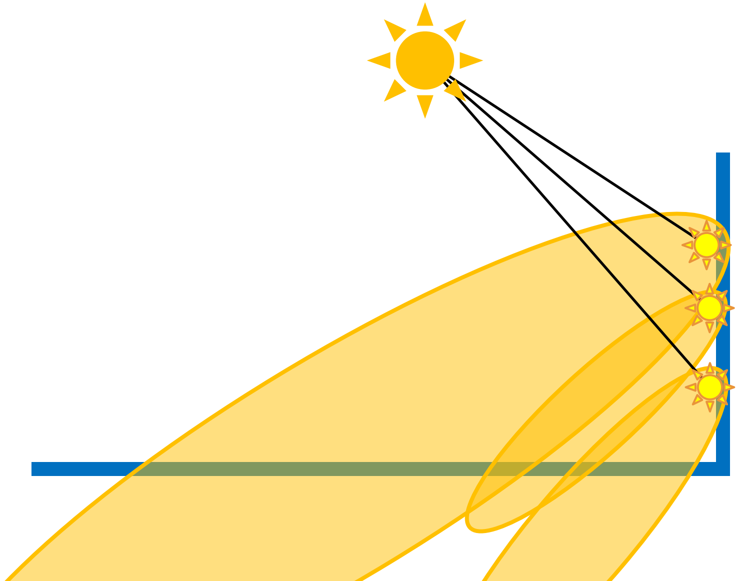
Random Light Range

- Russian roulette according to the radiant intensity × fall-off for each VPL
- Single random number for each VPL
- Can bound the light range in an unbiased fashion
- Random sampling of an isosurface of the reflected radiance
  - Inverse of the probability of Russian roulette

BRDF

$$l_{max}(\boldsymbol{\omega}_o) = p^{-1}(\xi) = \sqrt{\frac{\Phi f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \max(\boldsymbol{\omega}_o \cdot \mathbf{n}, 0)}{\delta \xi}}$$

uniform random number $\in [0,1)$



Microfacet BRDF

masking-shadowing (low-frequency) — Fresnel (low-frequency) — NDF (GGX) (high-frequency)

$$f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{G_2(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h) F(\boldsymbol{\omega}_h \cdot \boldsymbol{\omega}_i) D(\boldsymbol{\omega}_h \cdot \mathbf{n})}{4|\boldsymbol{\omega}_i \cdot \mathbf{n}||\boldsymbol{\omega}_o \cdot \mathbf{n}|}$$

- Consider a simpler function for the bounding volume

Maximum masking-shadowing — Maximum Fresnel

$$f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \max(\boldsymbol{\omega}_o \cdot \mathbf{n}, 0) \leq \frac{G_1^{\text{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i) D(\boldsymbol{\omega}_h \cdot \mathbf{n})}{4|\boldsymbol{\omega}_i \cdot \mathbf{n}|}$$

Microfacet BRDF

masking-shadowing (low-frequency) — Fresnel (low-frequency) — NDF (GGX) (high-frequency)

$$f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{G_2(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h) F(\boldsymbol{\omega}_h \cdot \boldsymbol{\omega}_i) D(\boldsymbol{\omega}_h \cdot \mathbf{n})}{4|\boldsymbol{\omega}_i \cdot \mathbf{n}||\boldsymbol{\omega}_o \cdot \mathbf{n}|}$$

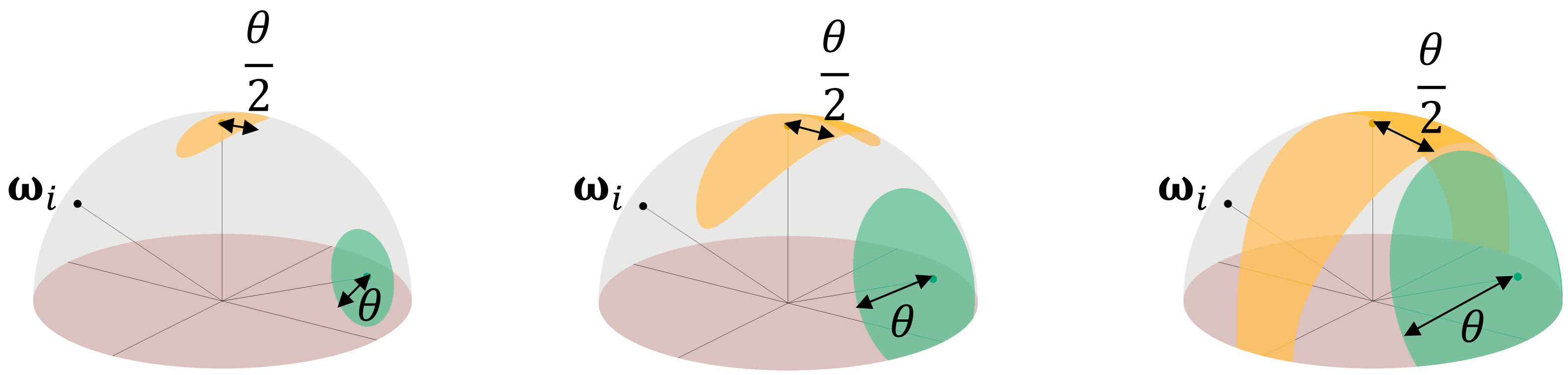
- Consider a simpler function for the bounding volume

Maximum masking-shadowing — Maximum Fresnel

$$f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \max(\boldsymbol{\omega}_o \cdot \mathbf{n}, 0) \leq \frac{G_1^{\text{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i) D(\boldsymbol{\omega}_h \cdot \mathbf{n})}{4|\boldsymbol{\omega}_i \cdot \mathbf{n}|}$$

only the $\boldsymbol{\omega}_o$-dependent term

Angular Lower Bound for the Halfvector

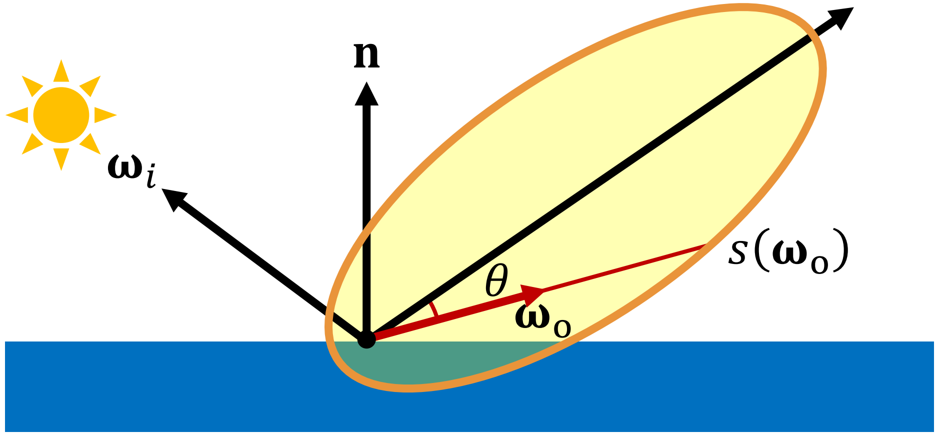


θ: angle between the outgoing direction $\boldsymbol{\omega}_o$ and perfect specular reflection vector

- Angular lower bound between the halfvector $\boldsymbol{\omega}_h$ and normal $\mathbf{n}$ [Jakob14]
- GGX is monotonically decreasing for the angle, when roughness parameter $\alpha \in (0,1]$

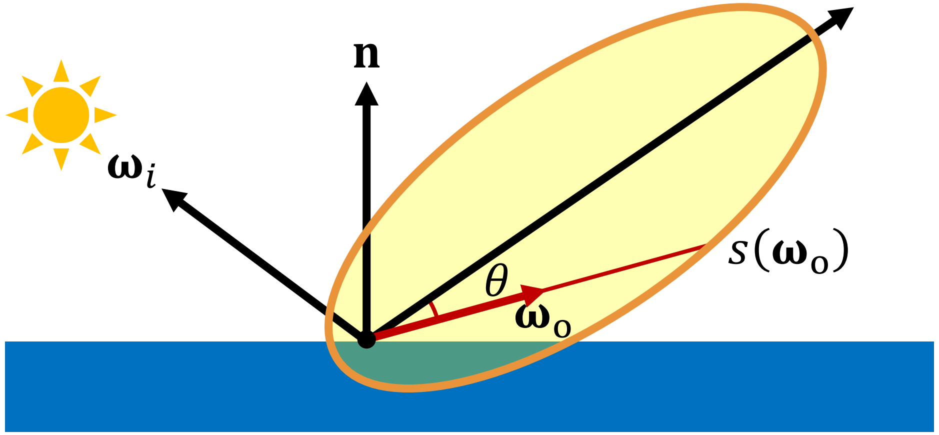
$$D(\boldsymbol{\omega}_h \cdot \mathbf{n}) \leq D\left(\cos\frac{\theta}{2}\right)$$

Bounding Volume for the Light Range



$$l_{max}(\boldsymbol{\omega}_o) \leq s(\boldsymbol{\omega}_o) = \sqrt{\frac{\Phi G_1^{\text{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i)}{4\delta\xi |\boldsymbol{\omega}_i \cdot \mathbf{n}|} D\left(\cos\frac{\theta}{2}\right)}$$

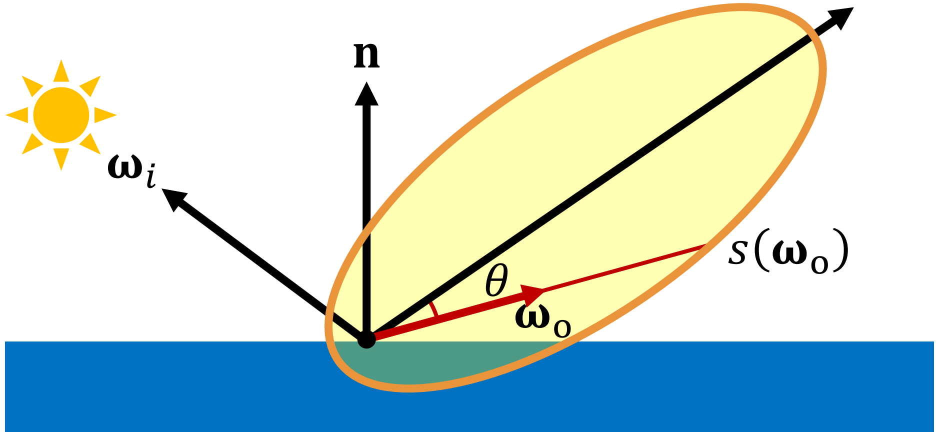
Bounding Volume for the Light Range



$$l_{max}(\boldsymbol{\omega}_o) \leq s(\boldsymbol{\omega}_o) = \sqrt{\frac{\Phi G_1^{\mathrm{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i)}{4\delta\xi|\boldsymbol{\omega}_i \cdot \mathbf{n}|} D\left(\cos\frac{\theta}{2}\right)}$$

Independent from $\boldsymbol{\omega}_o$

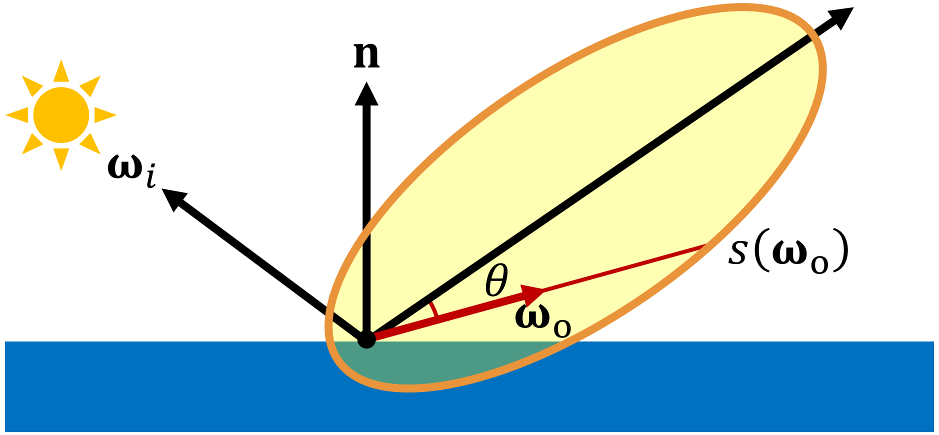
Bounding Volume for the Light Range



$$l_{max}(\boldsymbol{\omega}_\mathrm{o}) \leq s(\boldsymbol{\omega}_\mathrm{o}) = \sqrt{\frac{\Phi G_1^{\mathrm{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i)}{4\delta\xi|\boldsymbol{\omega}_i \cdot \mathbf{n}|} D\left(\cos\frac{\theta}{2}\right)} = r\sqrt{\pi D\left(\cos\frac{\theta}{2}\right)}$$

Independent from $\boldsymbol{\omega}_\mathrm{o}$

Bounding Volume for the Light Range

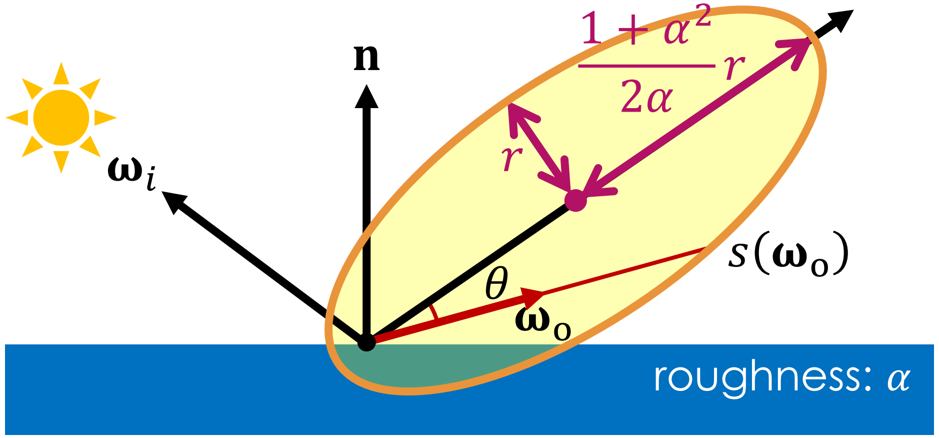


$$l_{max}(\boldsymbol{\omega}_{\mathrm{o}}) \le s(\boldsymbol{\omega}_{\mathrm{o}}) = \sqrt{\frac{\Phi G_1^{\mathrm{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i)}{4\delta\xi|\boldsymbol{\omega}_i \cdot \mathbf{n}|} D\left(\cos\frac{\theta}{2}\right)} = r\sqrt{\pi D\left(\cos\frac{\theta}{2}\right)}$$

Independent from $\boldsymbol{\omega}_{\mathrm{o}}$

Spheroid

Bounding Volume for the Light Range



$$l_{max}(\boldsymbol{\omega}_\mathrm{o}) \leq s(\boldsymbol{\omega}_\mathrm{o}) = \sqrt{\frac{\Phi G_1^{\mathrm{dist}}(\boldsymbol{\omega}_i) F_{\max}(\boldsymbol{\omega}_i)}{4\delta\xi|\boldsymbol{\omega}_i \cdot \mathbf{n}|} D\left(\cos\frac{\theta}{2}\right)} = r\sqrt{\pi D\left(\cos\frac{\theta}{2}\right)}$$

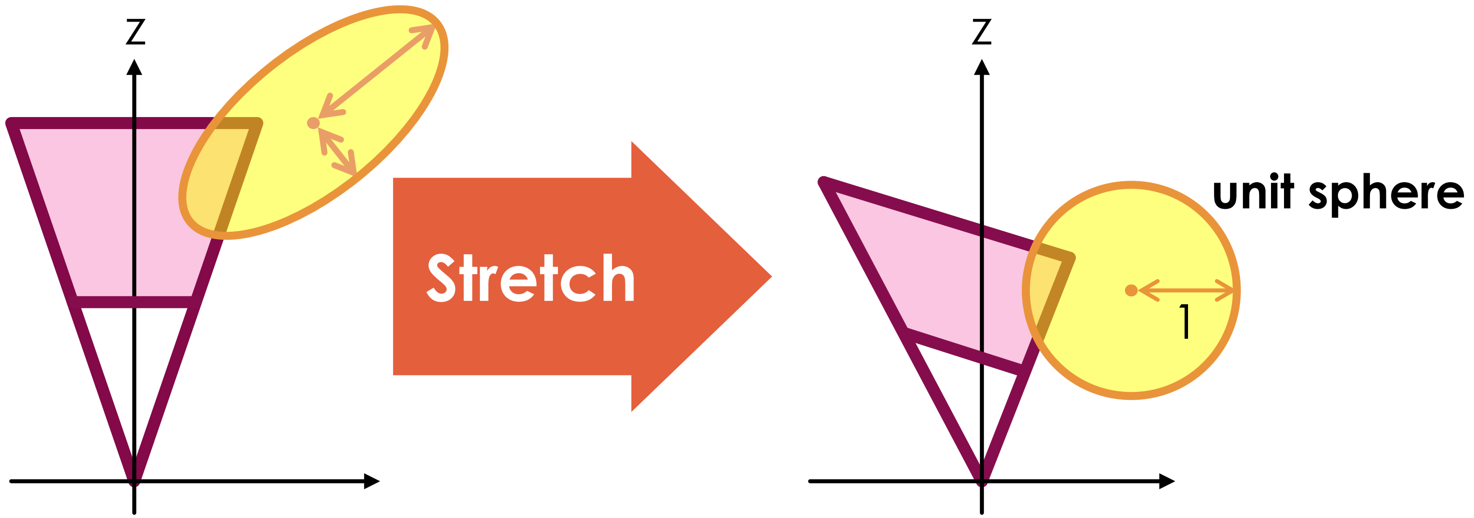
Independent from $\boldsymbol{\omega}_\mathrm{o}$

Spheroid

Tiled Culling

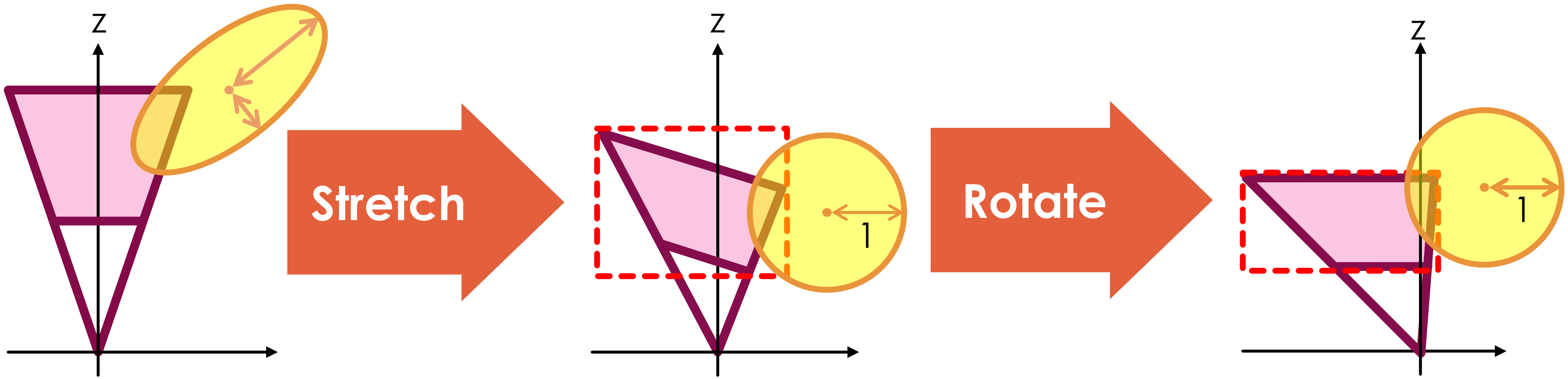
Tiled Culling for Bounding Ellipsoids

- Rough intersection test between the ellipsoid and a frustum in each screen-space tile
- Calculate using a sphere-frustum intersection test in the stretched space
  - Implemented based on sphere-based *Modified HalfZ Culling* [Stewart15]



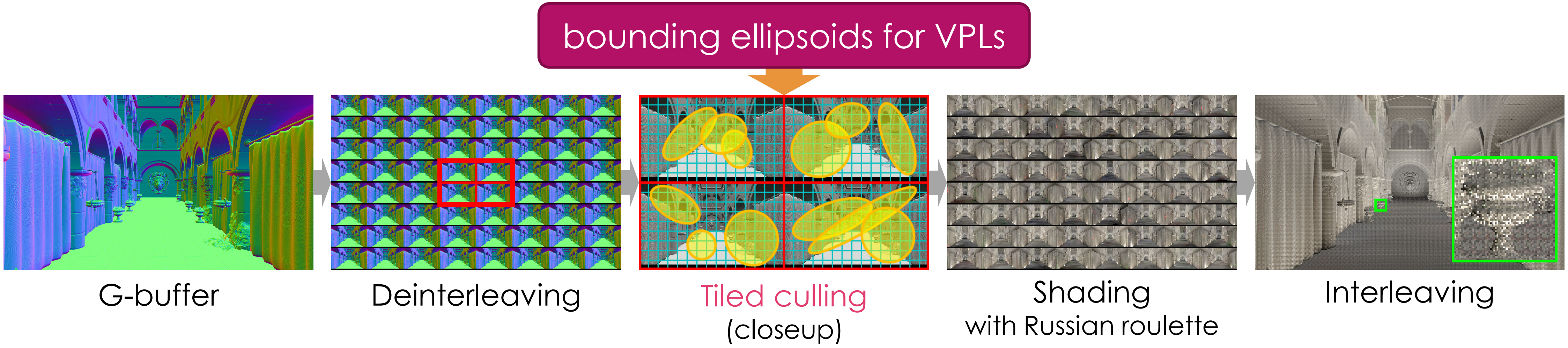
Optimization

- Depth plane of the frustum should be perpendicular to the z-axis for code optimization
- Mismatch between the stretched frustum and AABB(used in Modified HalfZ Culling)
- Solution: Rotate the test space



Interleaved Sampling of VPLs

- Combination of stochastic light culling & interleaved sampling [Segovia06]
- Interleaved sampling reduces both culling time & shading time
  - 8x8 interleaving for 65536 VPLs → 1024 VPLs/tile
- Variance is produced as noise ⟵ Denoise in post-processing (cross bilateral filter)



Results

Single-bounce indirect illumination

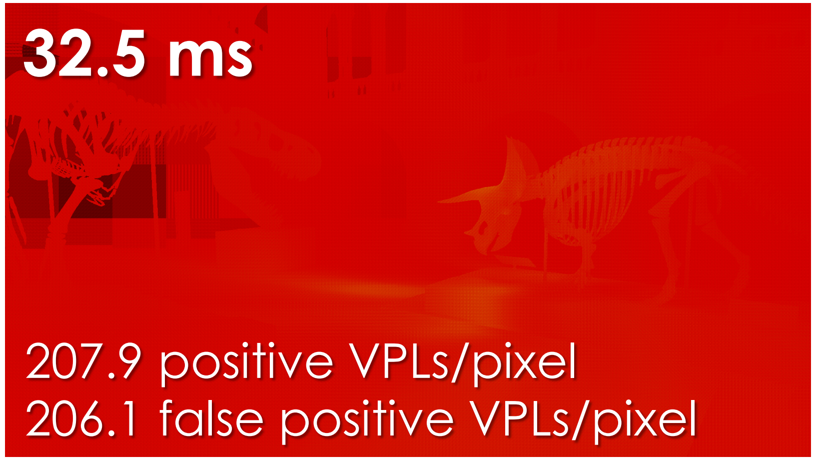
- Diffuse-to-diffuse light paths
- Glossy-to-diffuse light paths

65536 VPLs

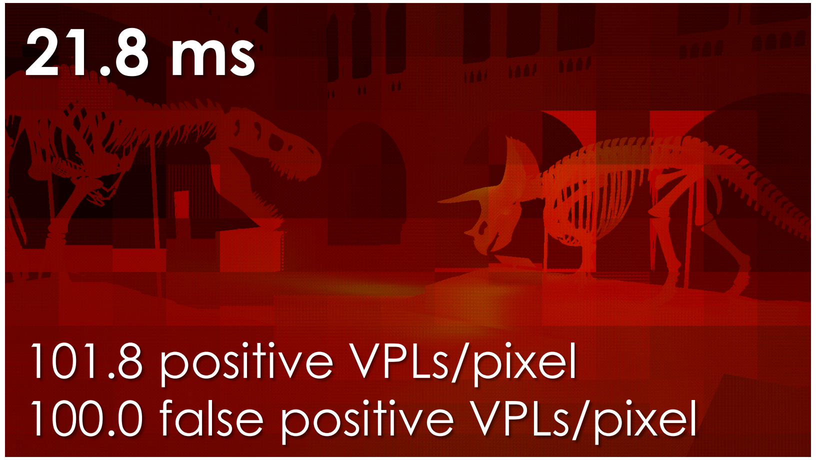
1920x1080 screen resolution

GPU: NVIDIA® GeForce® GTX 970

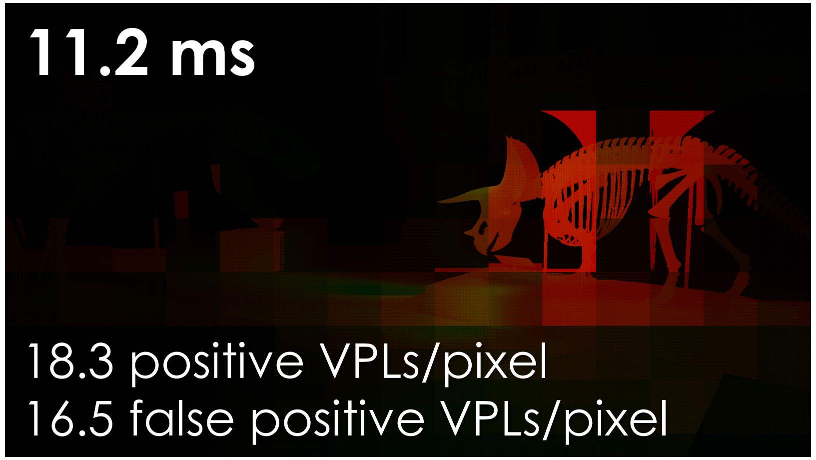
False Positives



Bounding sphere
centered at the VPL position



Tighter bounding sphere
enclosing our bounding ellipsoid



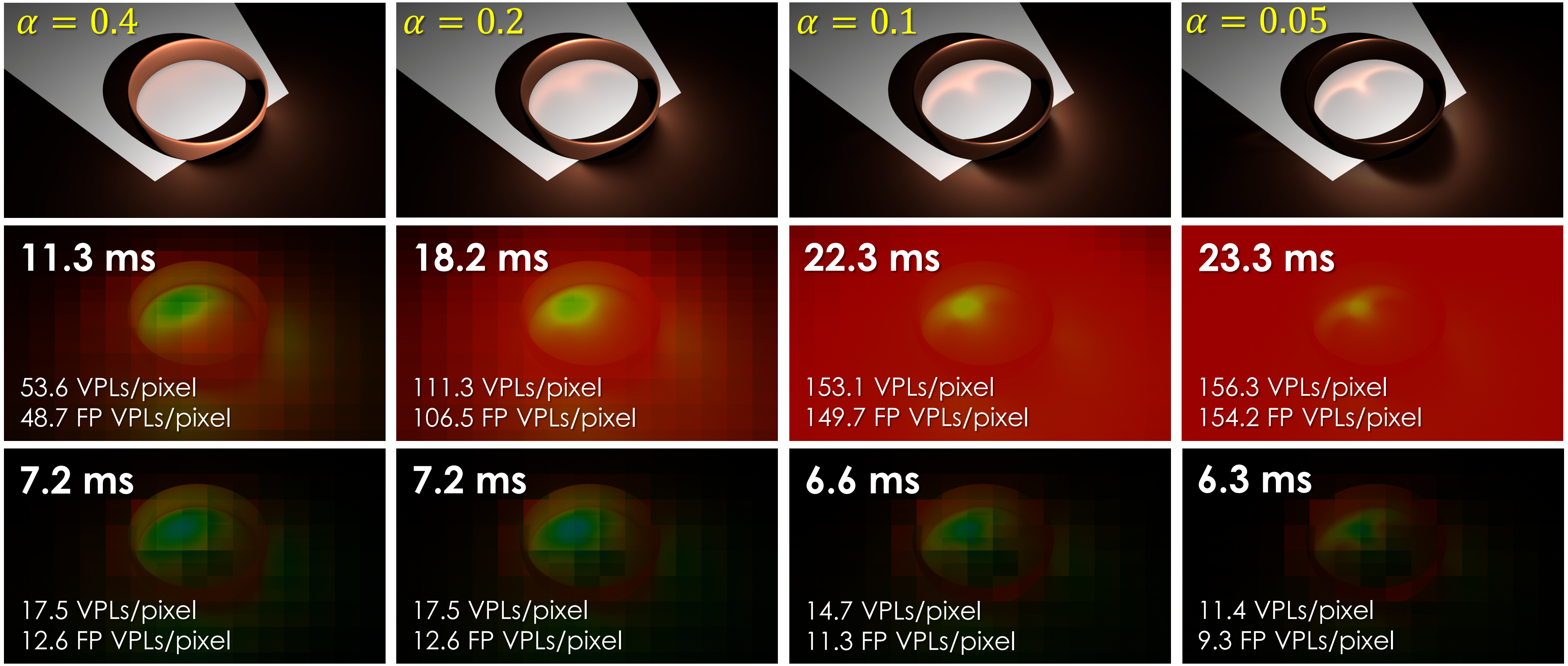
Our bounding ellipsoid

Brightness: number of positive glossy VPLs. Spectrum: false discovery rate

Different Roughnesses

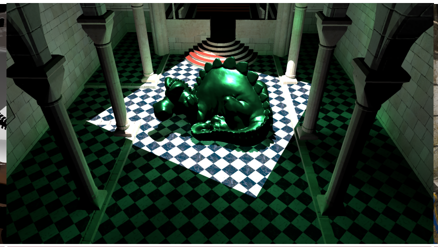
Bounding sphere centered at the VPL position:

Our bounding ellipsoid:



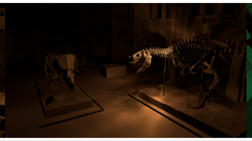
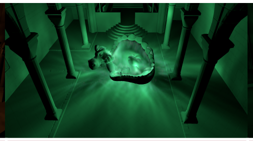
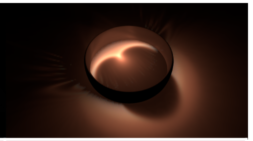
Brightness: number of positive glossy VPLs. Spectrum: false discovery rate

Computation Time

			
G-buffer	0.986 ms	0.618 ms	0.628 ms
Reflective shadow map	0.610 ms	0.270 ms	0.121 ms
VPL generation & range determination	**0.136 ms**	**0.140 ms**	**0.140 ms**
Culling & shading (diffuse VPLs + glossy VPLs)	**7.398 ms**	**6.960 ms**	**9.745 ms**
Cross bilateral filtering	1.320 ms	1.328 ms	1.320 ms

1920×1080 screen resolution, GPU: NVIDIA® GeForce® GTX 970

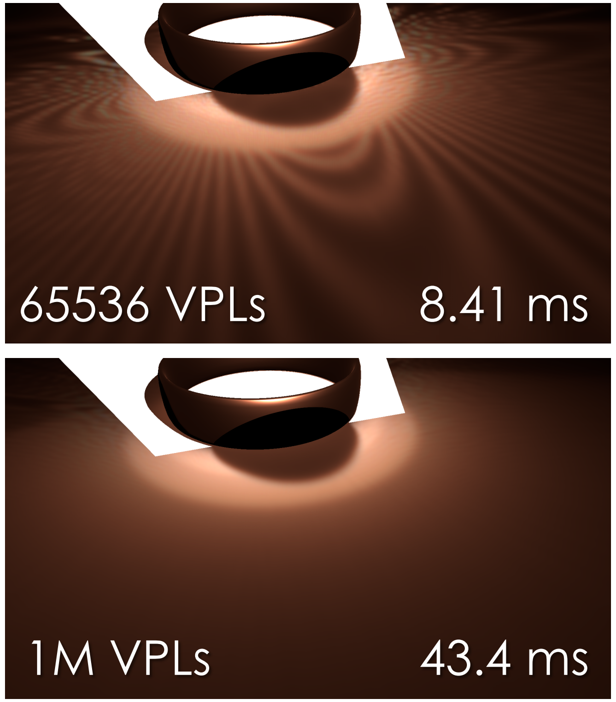
Culling Methods for Bounding Ellipsoids

Glossy VPLs only:							
Splatting [Dachsbacher06]	10.12 ms	8.91 ms	17.67 ms	5.93 ms	58.81 ms	14.68 ms	13.02 ms
Clustered shading [Ortegren16]	5.92 ms	4.99 ms	8.13 ms	3.92 ms	15.66 ms	10.99 ms	5.40 ms
Our culling & shading	**3.73 ms**	**2.87 ms**	**3.94 ms**	**2.15 ms**	**7.28 ms**	**5.14 ms**	**2.47 ms**

- Comparison with existing rasterization-based culling methods which support bounding ellipsoids
- Tiled culling is more suitable to interleaved sampling than rasterization-based culling

Limitations and Future Work

- Room to shrink the bounding ellipsoid
  - Anisotropic reflection lobe at grazing angles
  - Anisotropic NDFs
- Glossy-to-glossy interreflections
  - Sampling probability still ignores the BRDF at a shading point
  - Variance ☹
- Quality is limited by the density of VPLs before culling
  - Necessary to generate more VPLs for higher-frequency BRDFs
  - Need an efficient culling method for millions of VPLs



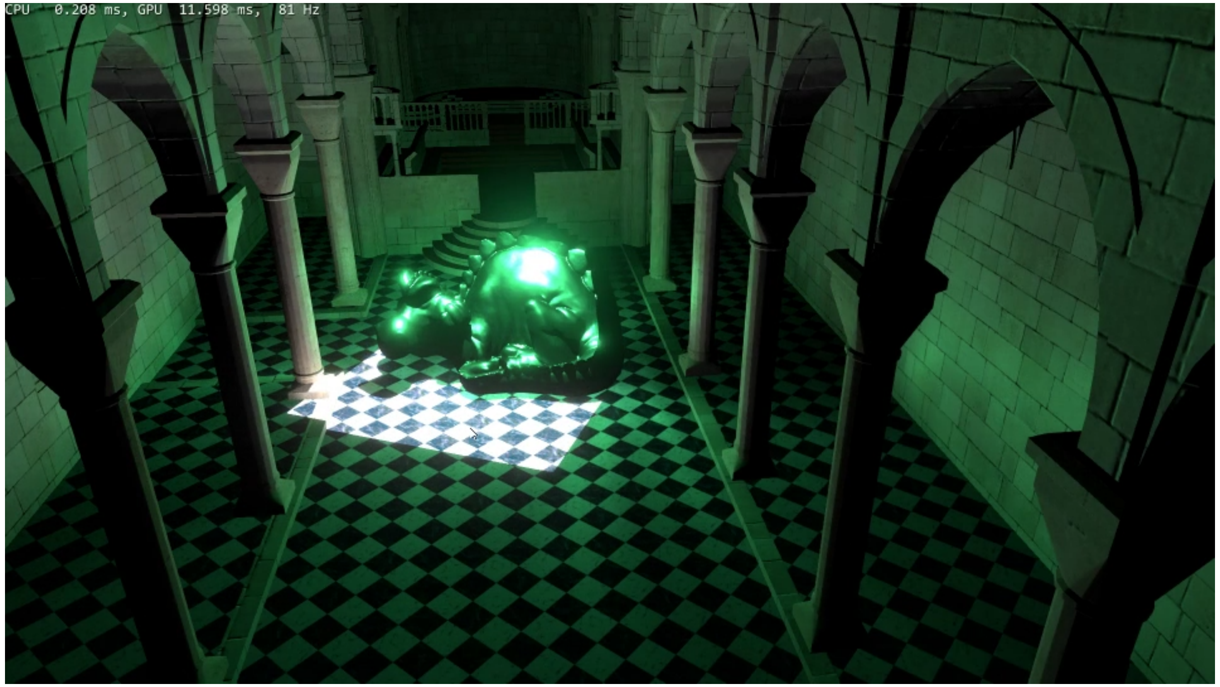
roughness: $\alpha = 0.02$

Conclusions

- $\sqrt{D\left(\cos\frac{\theta}{2}\right)}$ is a spheroid, if $D(\cos\theta)$ is the GGX distribution
- Bounding spheroid for the randomly sampled light range of a glossy VPL
- Tiled culling using bounding ellipsoids

Real-time glossy caustics created by the GGX microfacet BRDF

Thank you for your attention



References

- DACHSBACHER C., STAMMINGER M.: Splatting indirect illumination. In I3D'06 (2006), pp. 93–100
- JAKOB W., HAŠAN M., YAN L.-Q., LAWRENCE J., RAMAMOORTHI R., MARSCHNER S.: Discrete stochastic microfacet models. ACM Trans. Graph. 33, 4 (2014), 115:1–115:10.
- KELLER A.: Instant radiosity. In SIGGRAPH'97 (1997), pp. 49–56.
- OLSSON O., BILLETER M., ASSARSSON U.: Clustered deferred and forward shading. In HPG '12 (2012), pp. 87–96.
- ÖRTEGREN K., PERSSON E.: Clustered shading: Assigning lights using conservative rasterization in DirectX 12. In GPU Pro 7: Advanced Rendering Techniques. A K Peters/CRC Press, 2016, pp. 43–81.
- PHONG B. T.: Illumination for computer generated pictures. Commun. ACM 18, 6 (1975), 311–317.
- SEGOVIA B., IEHL J.-C., MITANCHEY R., PÉROCHE B.: Non-interleaved deferred shading of interleaved sample patterns. In Graph. Hardw. '06 (2006), pp. 53–60.
- STEWART J.: Compute-based tiled culling. In GPU Pro 6: Advanced Rendering Techniques. A K Peters/CRC Press, 2015, pp. 435–458.
- TOKUYOSHI Y., HARADA T.: Stochastic light culling. J. Comput. Graph. Tech. 5, 1 (2016), 35–60.
- WALTER B., MARSCHNER S. R., LI H., TORRANCE K. E.: Microfacet models for refraction through rough surfaces. In EGSR'07 (2007), pp. 195–206.