Interactive directional ambient occlusion and shadow computations for volume ray casting

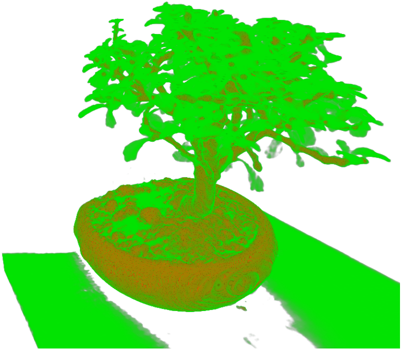
L. Q. Campagnolo W. Celes

lquatrin@tecgraf.puc-rio.br celes@tecgraf.puc-rio.br

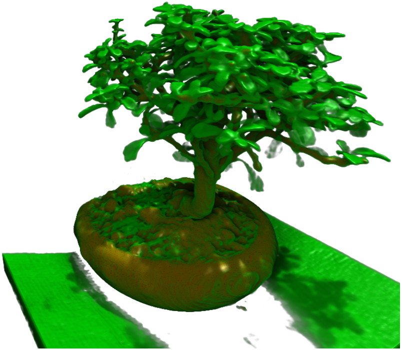
Tecgraf/PUC-Rio Institute, Rio de Janeiro, Brazil

SIBGRAPI 2019, Oct 29th

Goal



Emission and Absorption



Directional Ambient Occlusion and Shadows

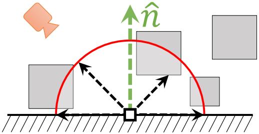
Introduction

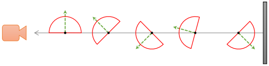
- Emission and Absorption **lacks** shape and depth details.
- Gradient-based shading.
  - Simplicity and neligible computational cost.
  - **Highly** sensible to noise.
  - **Local** lighting effects.
- Ambient Occlusion.
  - Simulates **global** illumination effects.
  - Evaluates the nontransparent structures in each point's vicinity.
  - Appears in three different flavors in the literature.

Introduction

- Types of Ambient Occlusion:

Surface
Ambient Occlusion

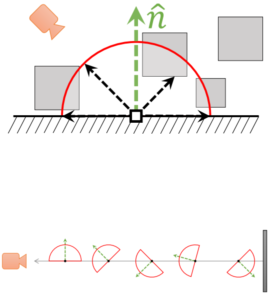




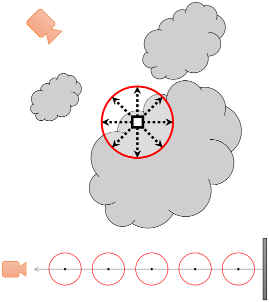
Introduction

- Types of Ambient Occlusion:

Surface
Ambient Occlusion



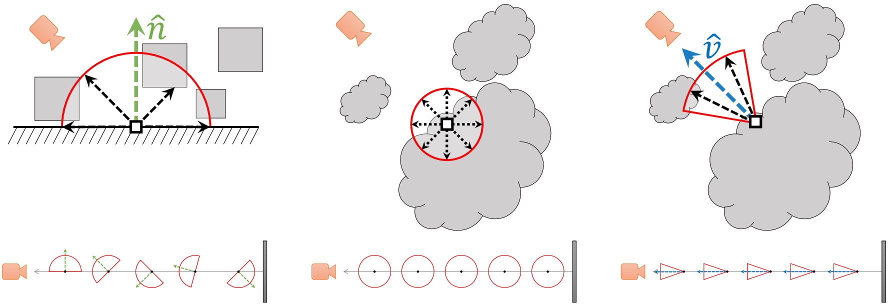
Local
Ambient Occlusion



Introduction

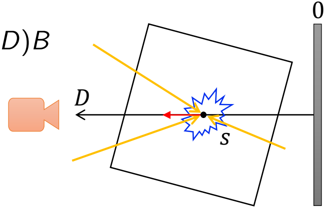
- Types of Ambient Occlusion:

Surface Ambient Occlusion	**Local** Ambient Occlusion	**Directional** Ambient Occlusion



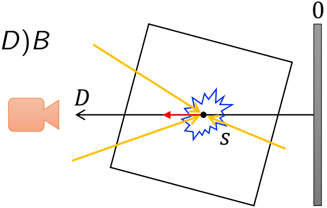
Directional Illumination Model

- Single scattering model:
  - $I(x, \hat{\omega}) = \int_0^D T(s)\tau(x(s))g(x(s), \hat{\omega})ds + T(D)B$



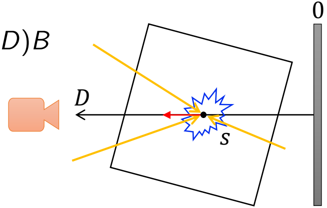
Directional Illumination Model

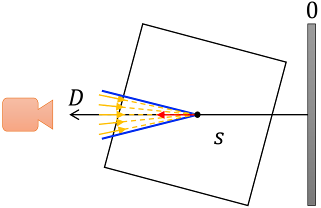
- Single scattering model:
  - $I(x,\hat{\omega}) = \int_0^D T(s)\tau(x(s))g(x(s),\hat{\omega})ds + T(D)B$
- In-scattered radiance:
  - $g(x,\hat{\omega}) = \int_\Omega r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$



Directional Illumination Model

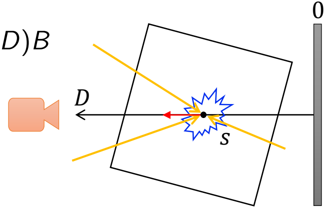
- Single scattering model:
  - $I(x, \hat{\omega}) = \int_0^D T(s)\tau(x(s))g(x(s), \hat{\omega})ds + T(D)B$
- In-scattered radiance:
  - $g(x, \hat{\omega}) = \int_\Omega r(\hat{\omega}, \hat{\omega}')L_0 e^{-\int_0^L \tau(x - s\hat{\omega}')ds} d\hat{\omega}'$
- Cone domain:
  - $g(x, \hat{\omega}) = \int_\Psi r(\hat{\omega}, \hat{\omega}')L_0 e^{-\int_0^L \tau(x - s\hat{\omega}')ds} d\hat{\omega}'$

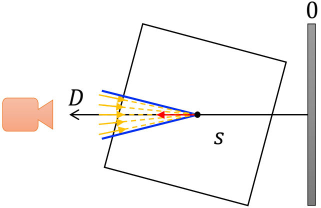




Directional Illumination Model

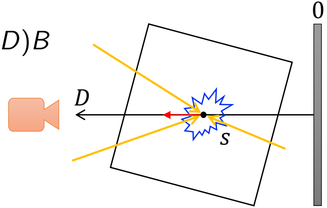
- Single scattering model:
  - $I(x,\hat{\omega}) = \int_0^D T(s)\tau(x(s))g(x(s),\hat{\omega})ds + T(D)B$
- In-scattered radiance:
  - $g(x,\hat{\omega}) = \int_\Omega r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
- Cone domain:
  - $g(x,\hat{\omega}) = \int_\Psi r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
- Also works for Shadows.

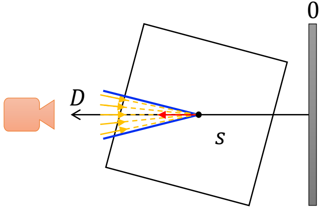




Directional Illumination Model

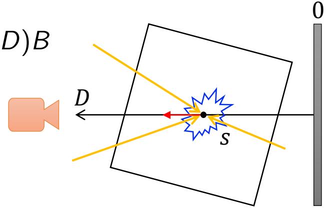
- Single scattering model:
  - $I(x,\hat{\omega}) = \int_0^D T(s)\tau(x(s))g(x(s),\hat{\omega})ds + T(D)B$
- In-scattered radiance:
  - $g(x,\hat{\omega}) = \int_\Omega r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
- Cone domain:
  - $g(x,\hat{\omega}) = \int_\Psi r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
- Also works for Shadows.
- Using Ray Tracing:
  - $g(x,\hat{\omega}) \approx \frac{\sum_{i=1}^n (\hat{\omega}\cdot\hat{\omega}_i)V_\tau(x,\hat{\omega}_i)}{\sum_{i=1}^n (\hat{\omega}\cdot\hat{\omega}_i)}$

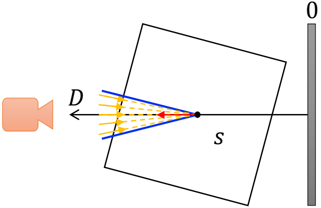




Directional Illumination Model

- Single scattering model:
  - $I(x,\hat{\omega}) = \int_0^D T(s)\tau(x(s))g(x(s),\hat{\omega})ds + T(D)B$
- In-scattered radiance:
  - $g(x,\hat{\omega}) = \int_\Omega r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
- Cone domain:
  - $g(x,\hat{\omega}) = \int_\Psi r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
- Also works for Shadows.
- Using Ray Tracing:
  - $g(x,\hat{\omega}) \approx \frac{\sum_{i=1}^n (\hat{\omega}\cdot\hat{\omega}_i)V_\tau(x,\hat{\omega}_i)}{\sum_{i=1}^n (\hat{\omega}\cdot\hat{\omega}_i)}$
- Ray Tracing is too costly for interactive visualizations.



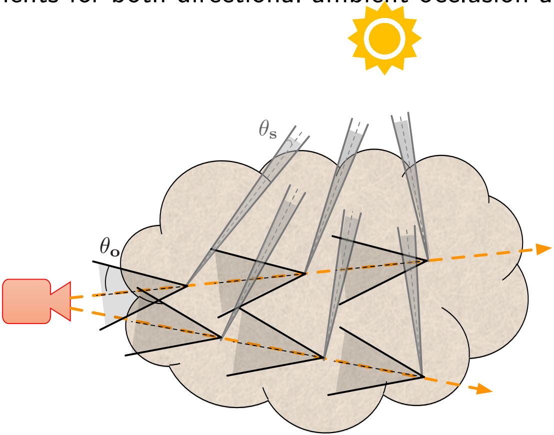


Our Proposal

- Cone tracing for **Directional Ambient Occlusion and Shadows**.
  - Replacing the large number of rays.
- Evaluate in-scattered light using a mathematical model based on **Gaussian integrals**.
  - Efficiently approximation.
- Comparison with past interactive specialized solutions.
  - **Competitive balance** between quality and performance.

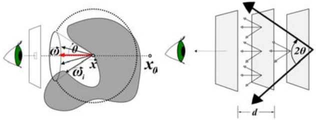
Our Proposal

- Cone placements for both directional ambient occlusion and shadows.

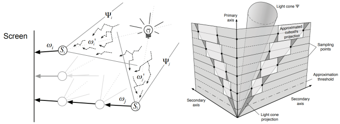


Past Proposed Techniques

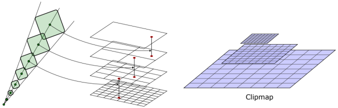
- Schott et al., 2009
  - Slice-based approach.
  - **Impractical** for shadow generation.
  - **Difficult** to apply optimizations.



- Schlegel et al., 2011
  - Based on Summed Area Table (SAT).
  - Cuboids **must** be volume aligned.

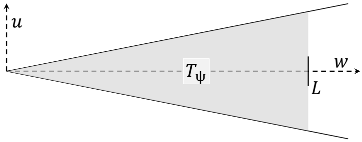


- Shih et al., 2016
  - Voxel Cone Tracing with super voxels.
  - Pre-filtered classification.
  - Large super voxels **miss** obstacle details.

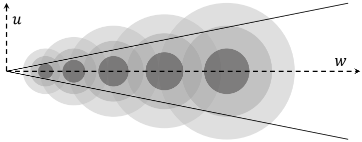


Proposed Transparency Computation

- Cone transparency estimation $T_\Psi(L)$:
  - $g(x,\hat{\omega}) = \int_\Psi r(\hat{\omega},\hat{\omega}')L_0 e^{-\int_0^L \tau(x-s\hat{\omega}')ds} d\hat{\omega}'$
  - $g(x,\hat{\omega}) \approx L_0 T_\Psi(L)$
  - $T_\Psi(L) = e^{-\iiint_\Psi \tau(u,v,w)dudvdw}$

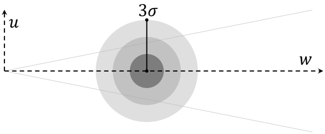


- Volumetric integral approximation:
  - Discrete samples along the cone axis.
  - Representative value of its vicinity.
  - Consecutive samples to compute $T_\Psi(L)$.



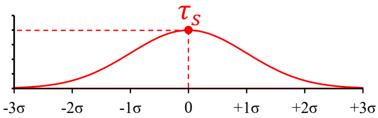
Proposed Transparency Computation

- Assume a Gaussian distribution in isolation.
  - Imposed σ.
  - Amplitude τ_s.



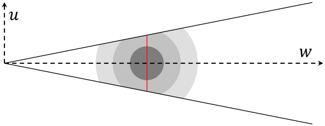
- Final associated function:

$$f(u,v,w) = \tau_s e^{-\frac{u^2+v^2+w^2}{2\sigma^2}}$$



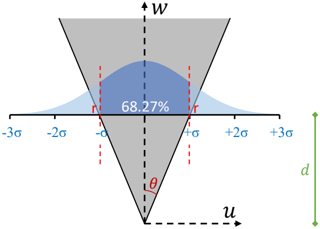
Proposed Transparency Computation

- Limit distribution inside the cone:
  - Truncated by cone surfaces.



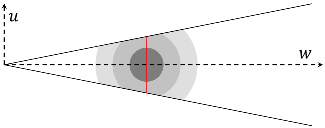
- Gaussian integral in the *uv* plane limited by the cone radius:
  - Truncate from $-r$ to r using p_r.

$$\int_{-r}^{r}\int_{-r}^{r}\tau_s e^{-\frac{t_u^2+t_v^2}{2\sigma^2}}\,dt_u dt_v = \tau_s\left(p_r\sigma\sqrt{2\pi}\right)^2$$



Proposed Transparency Computation

- Reduce to a representative 1D.
  - Along w direction.

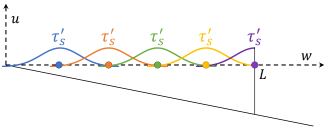


- Divide uv integral by the circular section area $A_c = \pi r^2$.

$$\iiint_{\Psi} \tau_s \, du dv dw \approx \int_0^L \frac{\left(p_r \sigma \sqrt{2\pi}\right)^2}{A_c} \tau_s \, dw$$

Proposed Transparency Computation

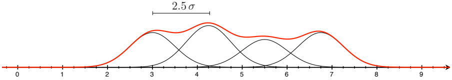
- Each sample:
  - 1D distribution with amplitude τ_s'.

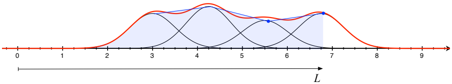


$$\tau_s' = \frac{\left(p_r \sigma \sqrt{2\pi}\right)^2}{A_c} \tau_s$$

- Linear approximation:

$$\iiint_\Psi \tau_s \, du dv dw \approx 0.5\, \sigma \sqrt{2\pi}\, \tau_{s(0)}' + \sum_0^{n-1} 2.5\sigma \frac{\tau_{s(i)}' + \tau_{s(i+1)}'}{2}$$

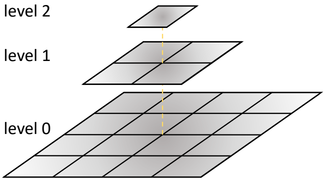




Implementation details: Extinction coefficient volume

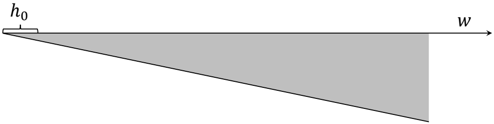
- Gaussian distribution amplitudes computed for each sample.
  - Using low-pass filter.
  - Time consuming.

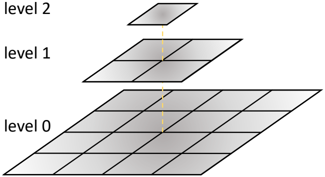
- Additional 3D texture to store precomputed amplitudes τ_s.
  - Imposed $\sigma = 1.0$ at the base level.
  - Successive convolutions to upper levels.
  - $\sigma_{i+1} = 2.0\sigma_i$.



Implementation details: Discrete cone sampling

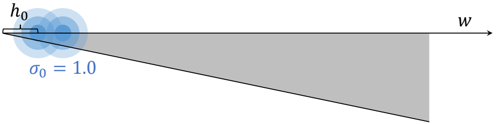
- Based on σ defined at extinction coefficient volume.
- Sample along the cone ray.
- Keep mipmap level until $r < r_c$, $r_c = 2\sigma$.

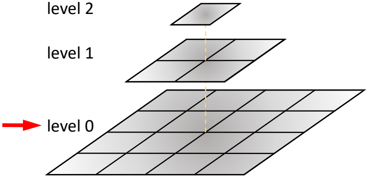




Implementation details: Discrete cone sampling

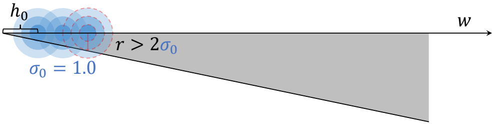
- Based on σ defined at extinction coefficient volume.
- Sample along the cone ray.
- Keep mipmap level until $r < r_c$, $r_c = 2\sigma$.

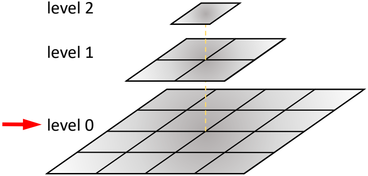




Implementation details: Discrete cone sampling

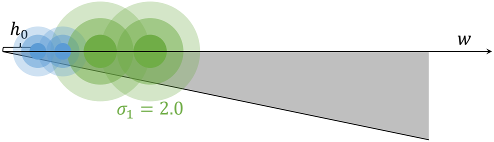
- Based on σ defined at extinction coefficient volume.
- Sample along the cone ray.
- Keep mipmap level until $r < r_c$, $r_c = 2\sigma$.

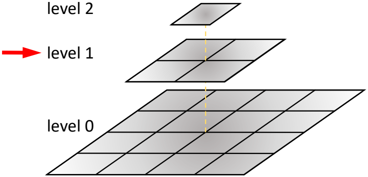




Implementation details: Discrete cone sampling

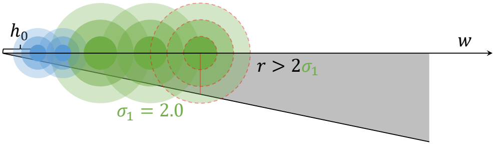
- Based on σ defined at extinction coefficient volume.
- Sample along the cone ray.
- Keep mipmap level until $r < r_c$, $r_c = 2\sigma$.

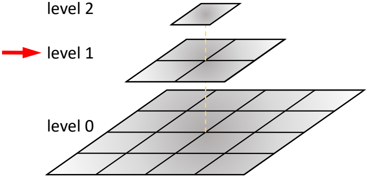




Implementation details: Discrete cone sampling

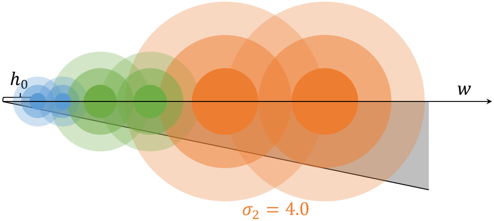
- Based on σ defined at extinction coefficient volume.
- Sample along the cone ray.
- Keep mipmap level until $r < r_c$, $r_c = 2\sigma$.

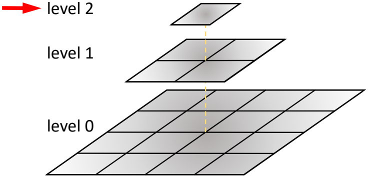




Implementation details: Discrete cone sampling

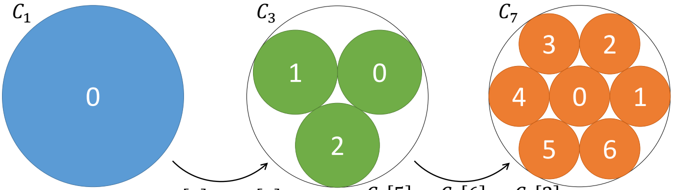
- Based on σ defined at extinction coefficient volume.
- Sample along the cone ray.
- Keep mipmap level until $r < r_c$, $r_c = 2\sigma$.





Implementation details: Cone splitting

- Cone split to prevent use of upper levels.
- Cosine-weighted average final transparency.
- Circle packing in a circle approach:



$C_3[0] = C_1[0]$
$C_3[1] = C_1[0]$
$C_3[2] = C_1[0]$

$C_7[5] = C_7[6] = C_3[2]$
$C_7[3] = C_7[4] = C_3[1]$
$C_7[1] = C_7[2] = C_3[0]$
$C_7[0] = (C_3[0] + C_3[1] + C_3[2])/3$

Results and Discussion

- Directional ambient occlusion comparison:
  - **Slice-based**: Schott et al.'s technique.
  - **Monte Carlo**: Ray Tracing implementation (Ground Truth).
- Shadow Comparison:
  - **SAT**: Schlegel et al.'s technique.
  - **VCT**: Single-GPU version of Shih et al.'s technique.
  - **Monte Carlo**: Ray Tracing implementation (Ground Truth).
- **image-space** and **object-space** illumination.
  - Image-space: Benefits from early termination.
  - Object-space: Avoids evaluating the same voxel multiple times.
  - **image-space** as default.

Results and Discussion

Dataset	Volume Size	Voxel Size
Synthetic	128x128x128	[1.00, 1.00, 1.00]
SyntheticBars	512x256x256	[1.00, 1.00, 1.00]
VisMale	128x256x256	[1.58, 0.99, 1.01]
Engine	256x256x256	[1.00, 1.00, 1.00]
Bonsai	256x256x256	[1.00, 1.00, 1.00]
Backpack	512x512x373	[0.98, 0.98, 1.25]
CT-Knee	379x229x305	[1.00, 1.00, 1.00]
Foot	256x256x256	[1.00, 1.00, 1.00]
Hazelnut	512x512x512	[1.00, 1.00, 1.00]
Flower	1024x1024x1024	[1.00, 1.00, 1.00]

Directional ambient occlusion comparison

- Inner regions better captured by our method.
- Darkness of oblique surfaces.
  - Expected from any cone tracing approximation.

VisMale Dataset

$\theta_o = 20.0$

Slice-based | Monte Carlo | Ours

Directional ambient occlusion comparison

- Overall occlusion is coherently captured.
- Limitation: semi-transparent regions with uniform opacity.

VisMale(T) Dataset

$\theta_o = 25.0$

Slice-based

Monte Carlo

Ours

Directional ambient occlusion comparison

- Overall, our method performed better than Slice-based.
- For models with semi-transparent regions: Object-space illumination.

		Frame Rate (*fps*)	
Dataset	$L = 0.5D$	Slice-based	Ours
VisMale	207.58	67	233
VisMale(T)	207.58	73	49
Bonsai	221.70	58	110
Backpack	423.50	27	83
CT-Knee	268.84	60	22
Foot	221.70	59	50

Directional ambient occlusion comparison

Performance for occlusion comparison using two viewports with sizes 768^2 and 1000^2

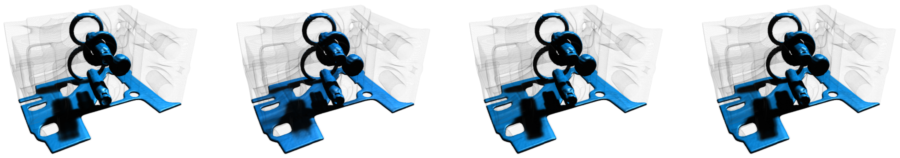
Dataset	Method	Dimension	Frame rate (fps) 768^2	Frame rate (fps) 1000^2
VisMale(T)	Slice-Based	–	73	45
	Ours image	–	49	30
	Ours object	128x256x256	159	140
	Ours object	128x128x128	376	290
CT-Knee	Slice-Based	–	60	39
	Ours image	–	22	13
	Ours object	256x256x256	51	49
	Ours object	200x200x200	96	88
Foot	Slice-Based	–	59	38
	Ours image	–	50	33
	Ours object	256x256x256	98	88
	Ours object	128x128x128	313	231

Shadow comparison

- Enlarged shadows from SAT.
- Inaccurate shadows from VCT.

Engine Dataset

$\theta_s = 2.0$



SAT | VCT | Monte Carlo | Ours

Shadow comparison

Performance comparison for shadow generation using a viewport of 768^2.

Dataset	$L = 0.75D$	Frame Rate (*fps*) SAT	VCT	Ours
SyntheticBars	470.30	89	178	83
Engine	332.55	74	97	63

Combining directional ambient occlusion and shadows

- Unable for past proposed methods.

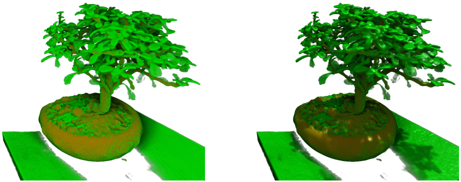
Performance results with directional occlusion and shadows using a viewport of 786^2.

Dataset	Ext. Coef. Volume Res.	Pre-illumination [1]	Parameters			Frame Rate (*fps*)		
			θ_o	C_{split}	θ_s	EA	AO	AO+SH
VisMale	128x256x256	128x128x128	25	7	2.0	783	376	263
Bonsai	256x256x256	—	20	3	0.5	458	155	64
Backpack	512x512x373	—	15	7	1.0	235	83	56
CT-Knee	256x256x256	200x200x200	10	3	2.0	630	183	103
Foot	256x256x256	256x256x256	30	3	1.0	511	163	58
Hazelnut	512x512x512	256x256x256	25	7	—	219	61	—
Flower	512x512x512	—	25	7	—	48	28	—

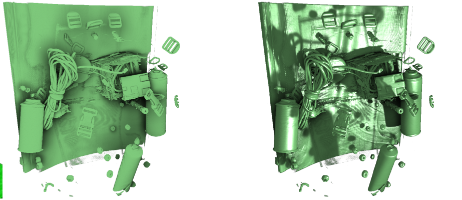
[1] Informed only if light is computed in Object Space.

Combining directional ambient occlusion and shadows

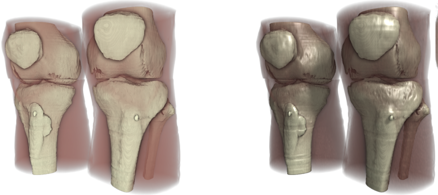
Bonsai



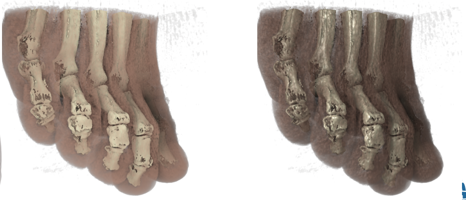
Backpack



CT-Knee



Foot



Conclusion

- Efficient method for Directional Ambient Occlusion and Shadows.
- Competitive quality and performance.
- First method focused on both Directional Ambient Occlusion and Shadows under the same structure.
- Limitations: Oblique surfaces and constant opacity regions.
- Future Work: Better representative preprocessed amplitudes and extension for unstructured meshes.

Thank You!

Interactive directional ambient occlusion and shadow computations for volume ray casting

Leonardo Quatrin Campagnolo
Waldemar Celes

Sponsored by CNPq & Tecgraf.

Questions?