

Hierarchical Multi-Layer Screen-Space Ray Tracing

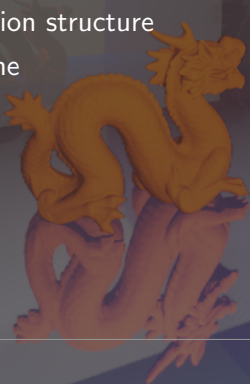
Nikolai Hofmann Phillip Bogendörfer Marc Stamminger Kai Selgrad

Computer Graphics Group, University of Erlangen-Nuremberg

July 29, 2017

Overview

- Hack
- Multiple depth layers
- 2.5D screen-space acceleration structure
- Multiple bounces in real-time
- Did I mention it's a hack?



Overview

Render passes:

- Collection
- Hierarchy construction
- Traversal

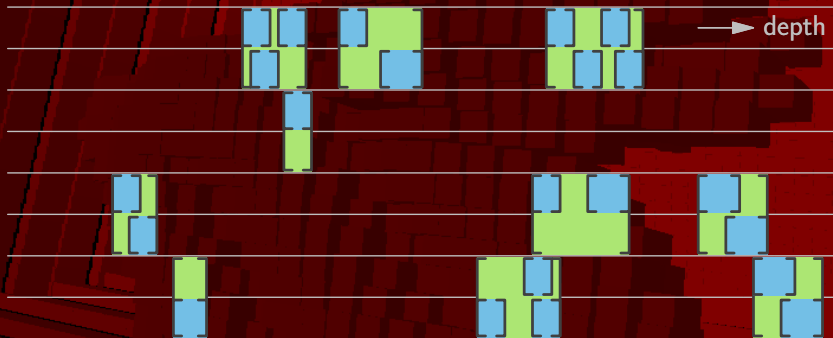


Collection pass

- Collect multiple layers of geometry in a single pass
- We used Per-Pixel Linked-Lists (PPLL)
- Works just as well with depth-peeling based approaches
- Details omitted here, see paper for technical details

Hierarchy construction

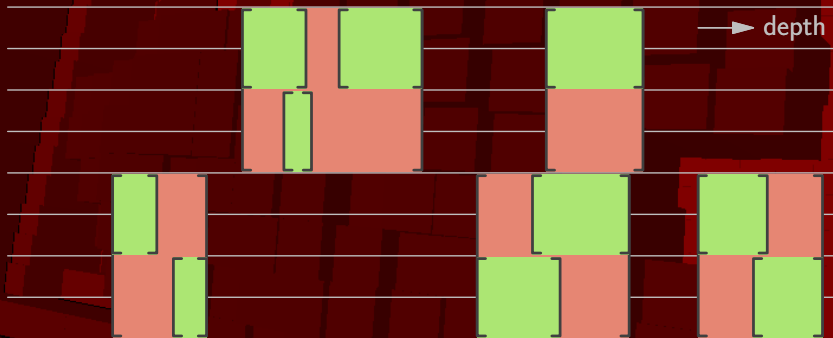
Level 1:



- Merge intervals closer than parameter τ from 2×2 pixels
- Consecutively relax per hierarchy level: $\tau \times 2^{\text{level}}$
- τ is multiple of (world space) pixel width

Hierarchy construction

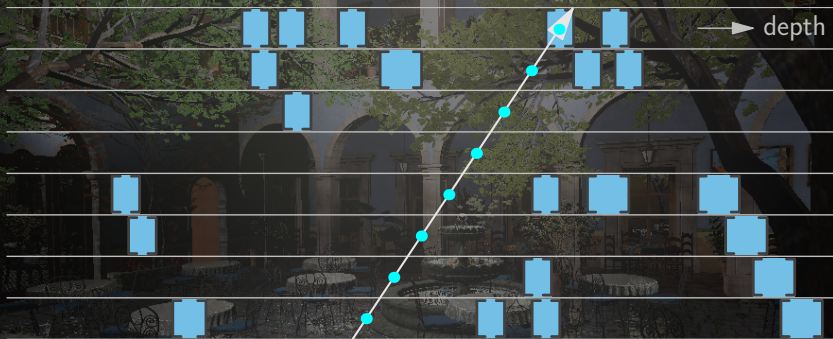
Level 2:



- Merge intervals closer than parameter τ from 2×2 pixels
- Consecutively relax per hierarchy level: $\tau \times 2^{\text{level}}$
- τ is multiple of (world space) pixel width

Traversal: Multi-Layer SSR (MLSSR)

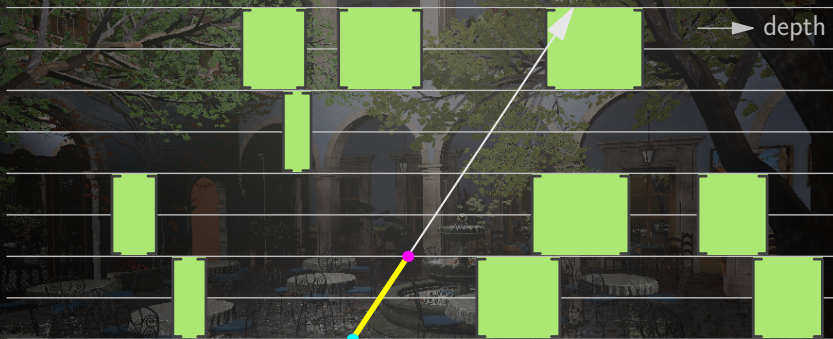
Level 0:



- Bresenham-style traversal, possibly striding
- Intersect ray with pixel depth intervals
- Applicable to one resolution

Traversal: Multi-resolution DDA

Level 1:



- DDA-style traversal, march ray on pixel grid
- Intersect ray with pixel depth intervals
- Applicable to multiple resolutions

Traversal: Multi-resolution DDA

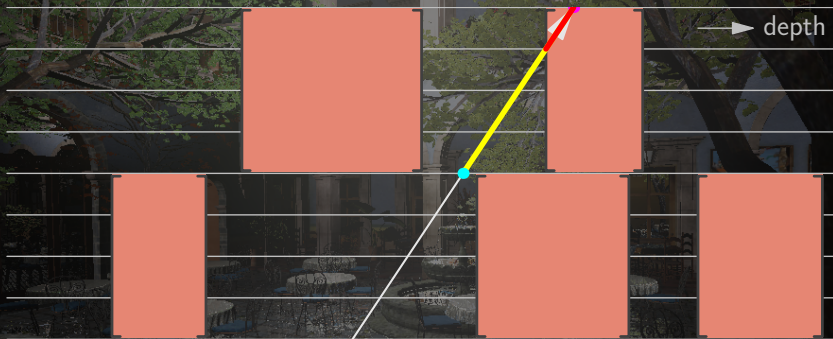
Level 2:



- DDA-style traversal, march ray on pixel grid
- Intersect ray with pixel depth intervals
- Applicable to multiple resolutions

Traversal: Multi-resolution DDA

Level 2:



- DDA-style traversal, march ray on pixel grid
- Intersect ray with pixel depth intervals
- Applicable to multiple resolutions

Traversal: Multi-resolution DDA

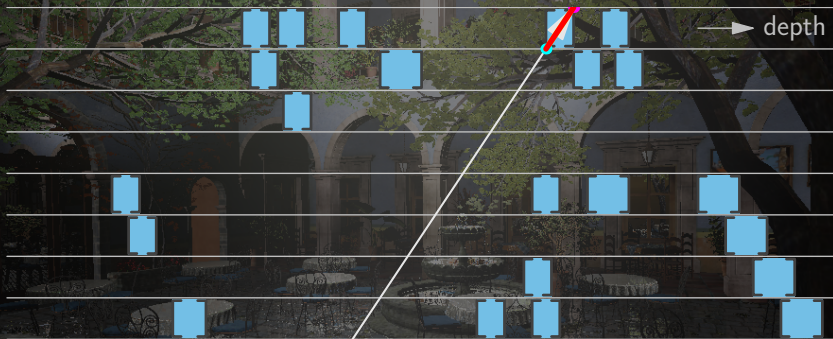
Level 1:



- DDA-style traversal, march ray on pixel grid
- Intersect ray with pixel depth intervals
- Applicable to multiple resolutions

Traversal: Multi-resolution DDA

Level 0:



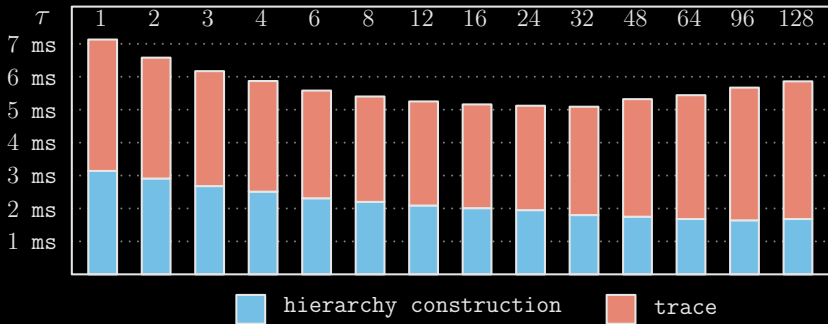
- DDA-style traversal, march ray on pixel grid
- Intersect ray with pixel depth intervals
- Applicable to multiple resolutions

Performance

- Everything bandwidth bound
- We used minimal material properties for the base level, modern physically based rendering might require more
- Will be focussing on hierarchy construction and traversal

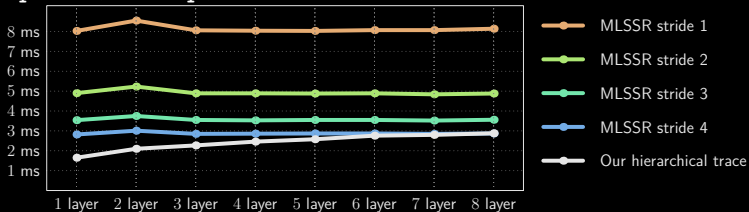
Tau

- We found it to be very robust in [15, 35]
- Note the logarithmic x-axis

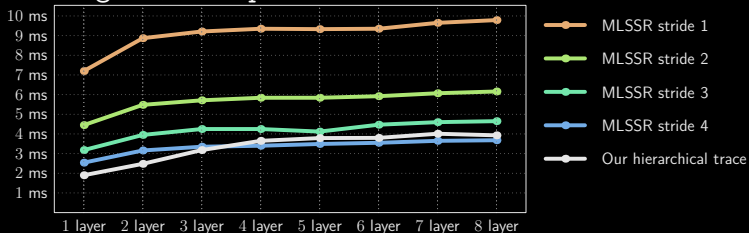


Traversal performance

Sponza, 1080p, GTX 1070, half-res trace:

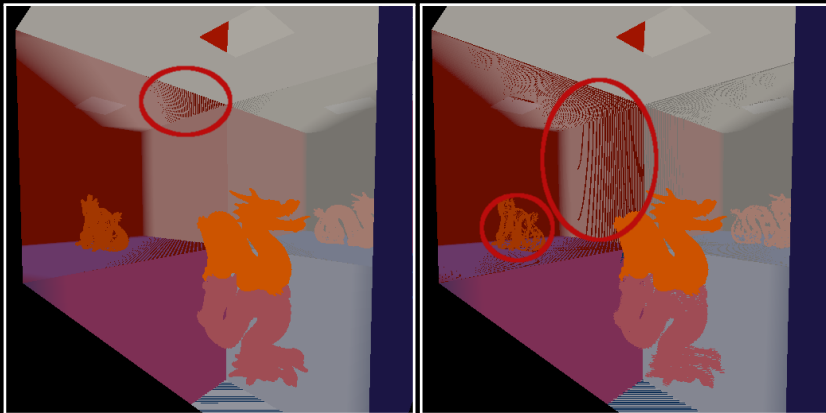


San Miguel, 1080p, GTX 1070, half-res trace:



Quality

Geometry misses by virtue of traversal:

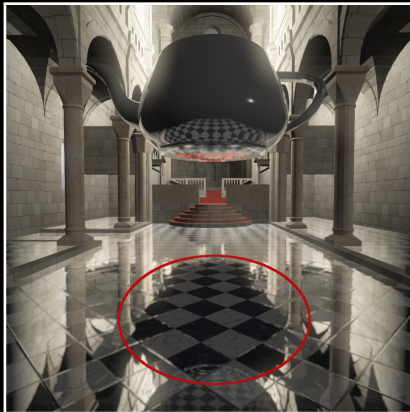


(Ours, MLSSR stride 1)

(MLSSR stride 4)

Quality

Quality increase from multiple depth layers:



(Single depth layer)



(Multiple depth layers)

Memory

Memory requirements for Sponza, 4 Layers, 5 Levels in MB:

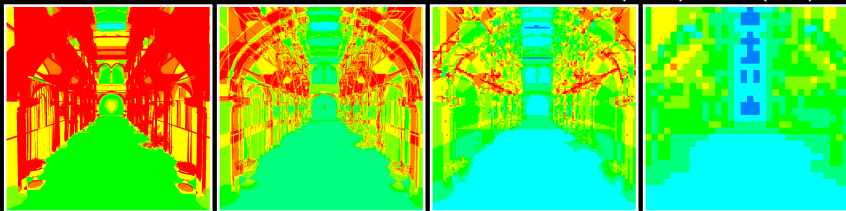
Resolution	PPLL	Deep G-Buffer	Hierarchy (worst-case)
720p	96	42	40 (147)
1080p	217	95	89 (332)

Memory

Memory requirements for Sponza, 4 Layers, 5 Levels in MB:

Resolution	PPLL	Deep G-Buffer	Hierarchy (worst-case)
720p	96	42	40 (147)
1080p	217	95	89 (332)

Population of hierarchy levels for $\tau = 25$ from 1 (blue) to 8 (red):



Level 0

Level 1

Level 3

Level 5

⇒ memory requirements lower than worst-case

Application: Ambient Occlusion



AO from 16 samples, Ours: 18.4ms, MLSSR: 12.1ms

Application: Shadow rays



Shadows from 3 point lights, Ours: 10.1ms, MLSSR: 24.4ms

Application: Reflections

1080p / 60fps / GTX 1070



Conclusion

Pros:

- Accelerated ray marching without trading image quality
- Faster than MLSSR for long rays
- Multiple bounces in real-time

Cons:

- Slower than MLSSR for short rays
- High memory footprint

Thanks!

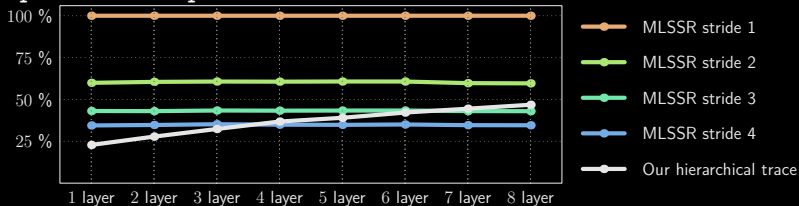
1080p / 60fps / GTX 1070



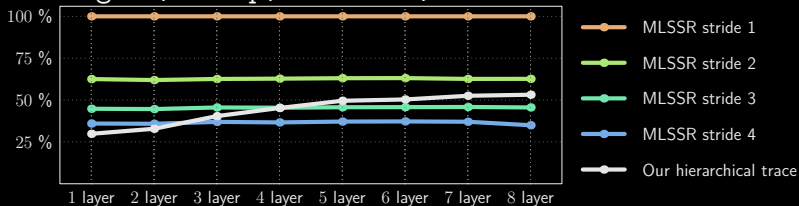
:wq

Construct & Traversal performance

Sponza, 1080p, GTX 1070, half-res trace:

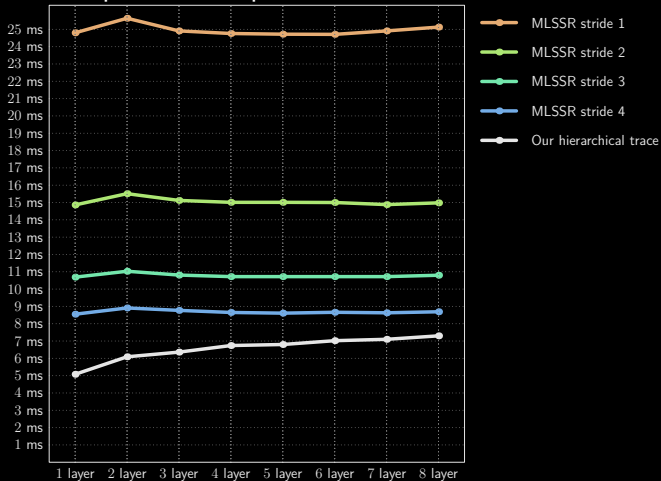


San Miguel, 1080p, GTX 1070, half-res trace:



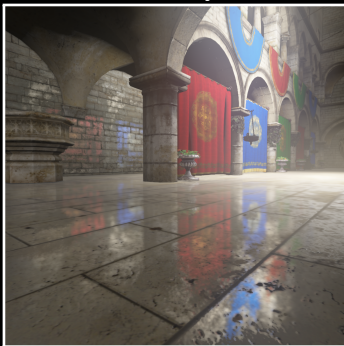
Traversal performance

Sponza, 1080p, GTX 1070, full-res trace:



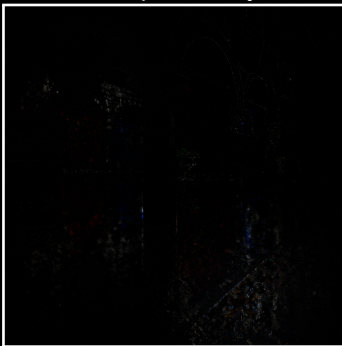
Terminating long rays

Kill 0 rays:



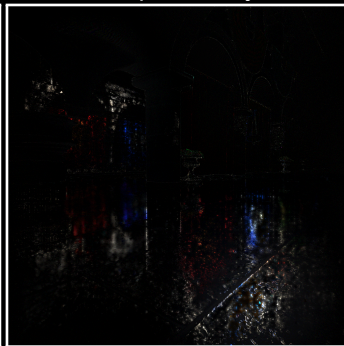
6.0ms

Kill up to 1 ray:



5.4ms

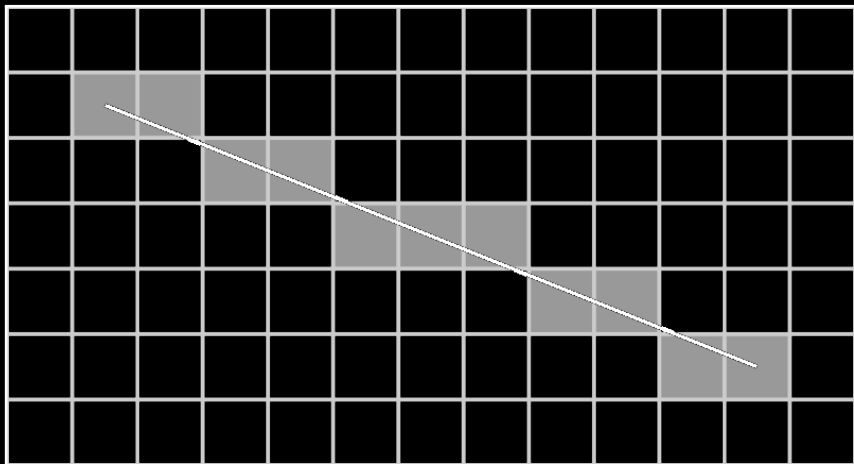
Kill up to 4 rays:



4.7ms

(Difference x8)

Traversal: Bresenham

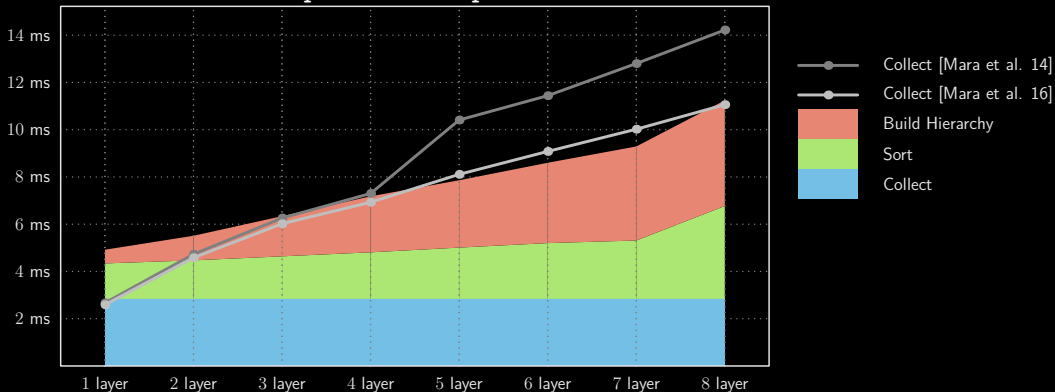


⇒ slight misses problematic when traversing the hierarchy

Traversal: Multi-resolution DDA

Collection performance

Sponza, 1080p, GTX 1070, collect



Our renderer is not quite up to date though..

