Interactive Volumetric Shadows in Participating Media with Single-Scattering

Chris Wyman
University of Iowa

Shaun Ramsey
Washington College

IEEE Symposium on Interactive Ray Tracing 2008

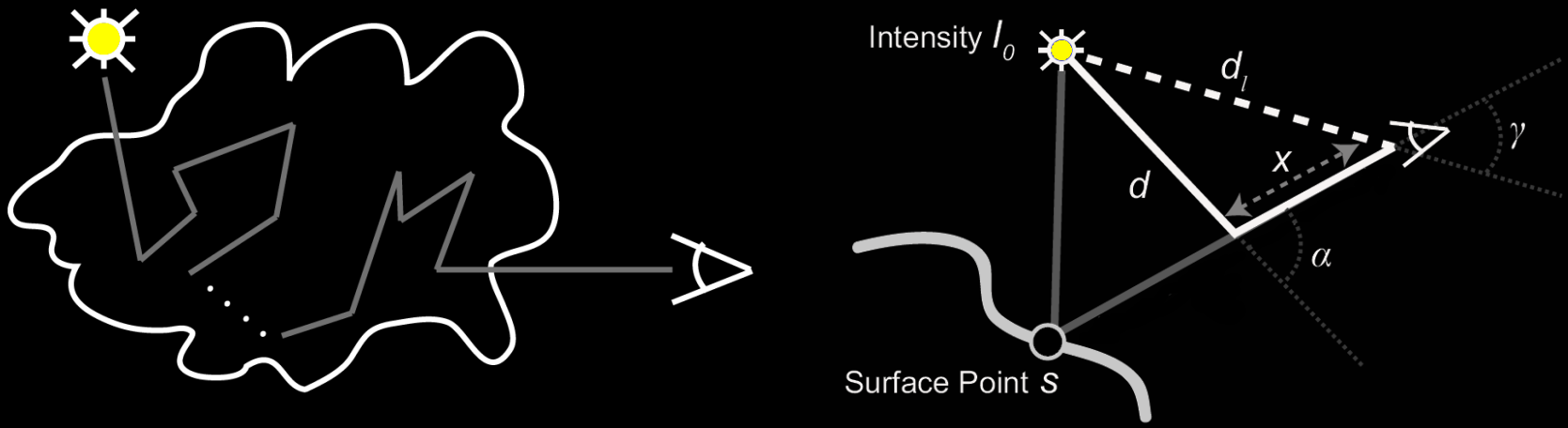
THE UNIVERSITY OF IOWA

What is Volumetric Scattering?

Creative Commons Image: Mila Zinkova

The University of Iowa

What is Volumetric Scattering?



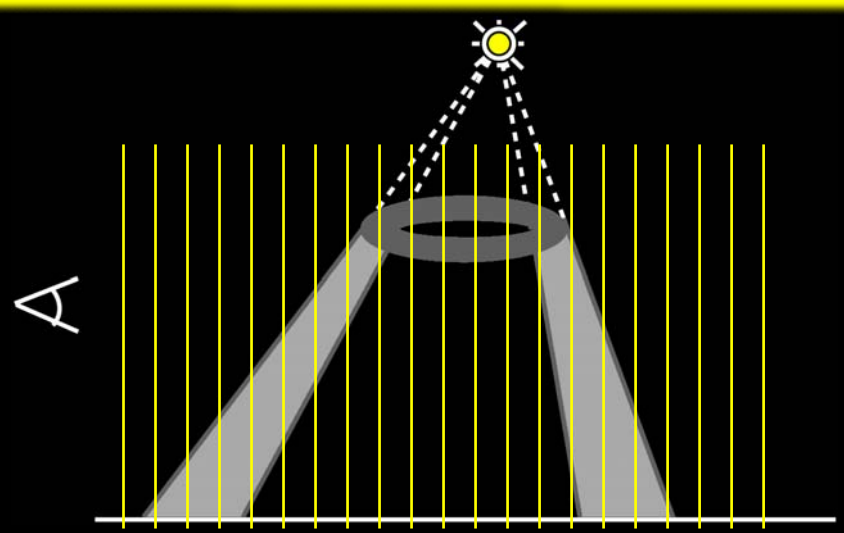
- Light interacts with particles
  - Absorption, emission, scattering
- Arbitrary number of interactions
  - Each scatter diffuses illumination
  - Many media well approximated with only 1 bounce
    - Simplifies computation

THE UNIVERSITY OF IOWA

Previous Work

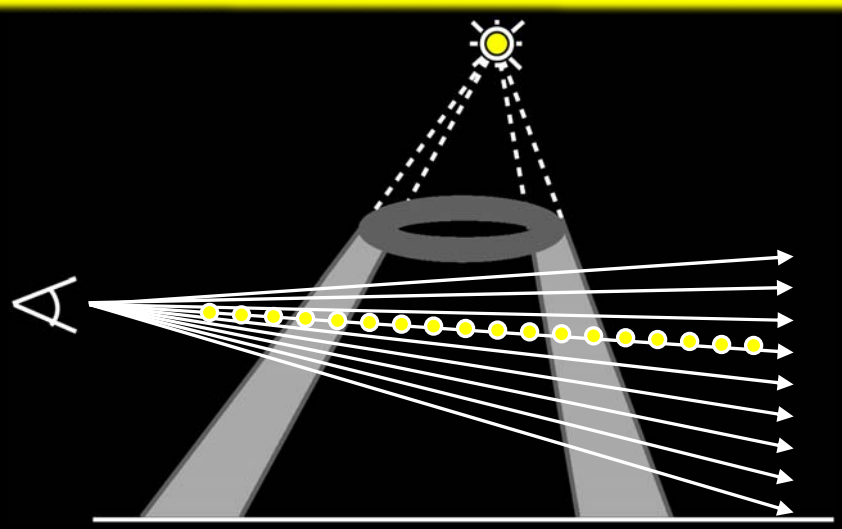
- (Generally) offline rendering techniques:
  - Random walk (e.g., [Rushmeier 1994])
  - Ray marching (e.g., [Kajiya and Von Herzen 1984])
  - Photon map (e.g., [Jarosz 2008])
- Interactive scattering models:
  - Single-scattering (e.g., [Sun 05], [Hoffman 03])
  - Multi-scattering (e.g., [Hegeman05])
- Interactive rendering for volume shadows:
  - Volume slicing and ray marching (e.g., [Imagire07])
  - Shadow volumes (e.g., [Biri06], [James03])

Real-time Rendering Challenges



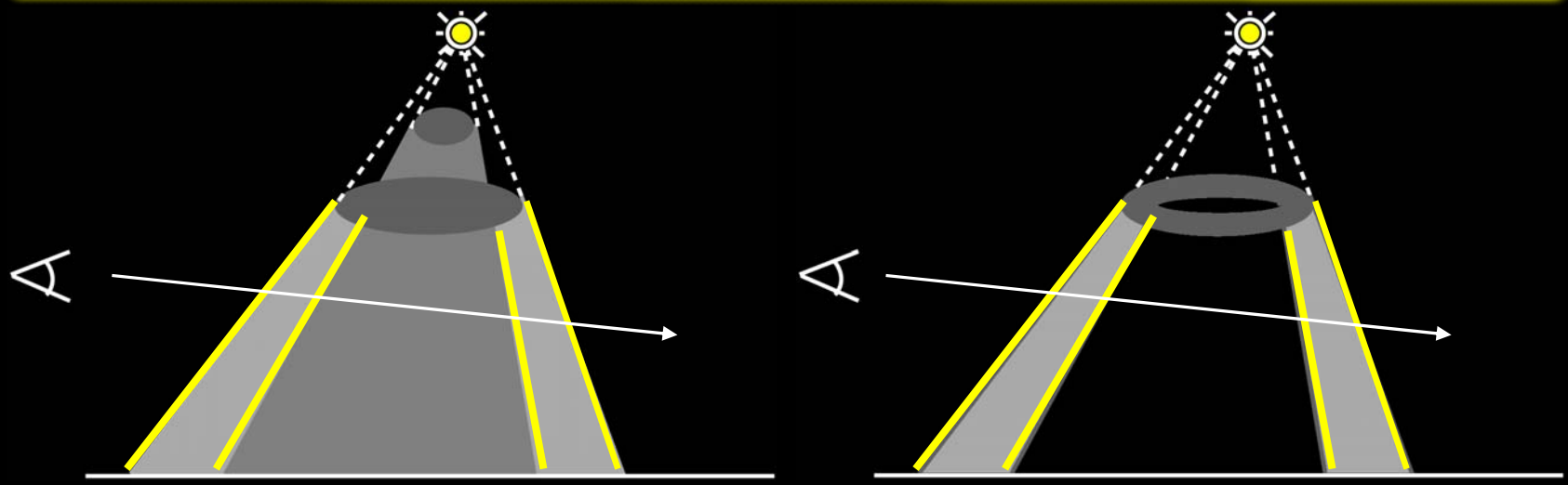
- Volume slicing
  - Combine scattering from planes parallel to image
  - Sampling locations same for all pixels
  - Forces tradeoff between:
    - Wasted samples for some pixels
    - Poor sampling around shadow boundaries for others

Real-time Rendering Challenges



- Ray marching
  - Loop over small steps for each ray
  - Still aliases shadow boundaries if poorly sampled
  - Independent samples possible
    - But how? Adaptive? Interleaved? Importance?
    - How efficient, esp. in streaming context?

Real-time Rendering Challenges



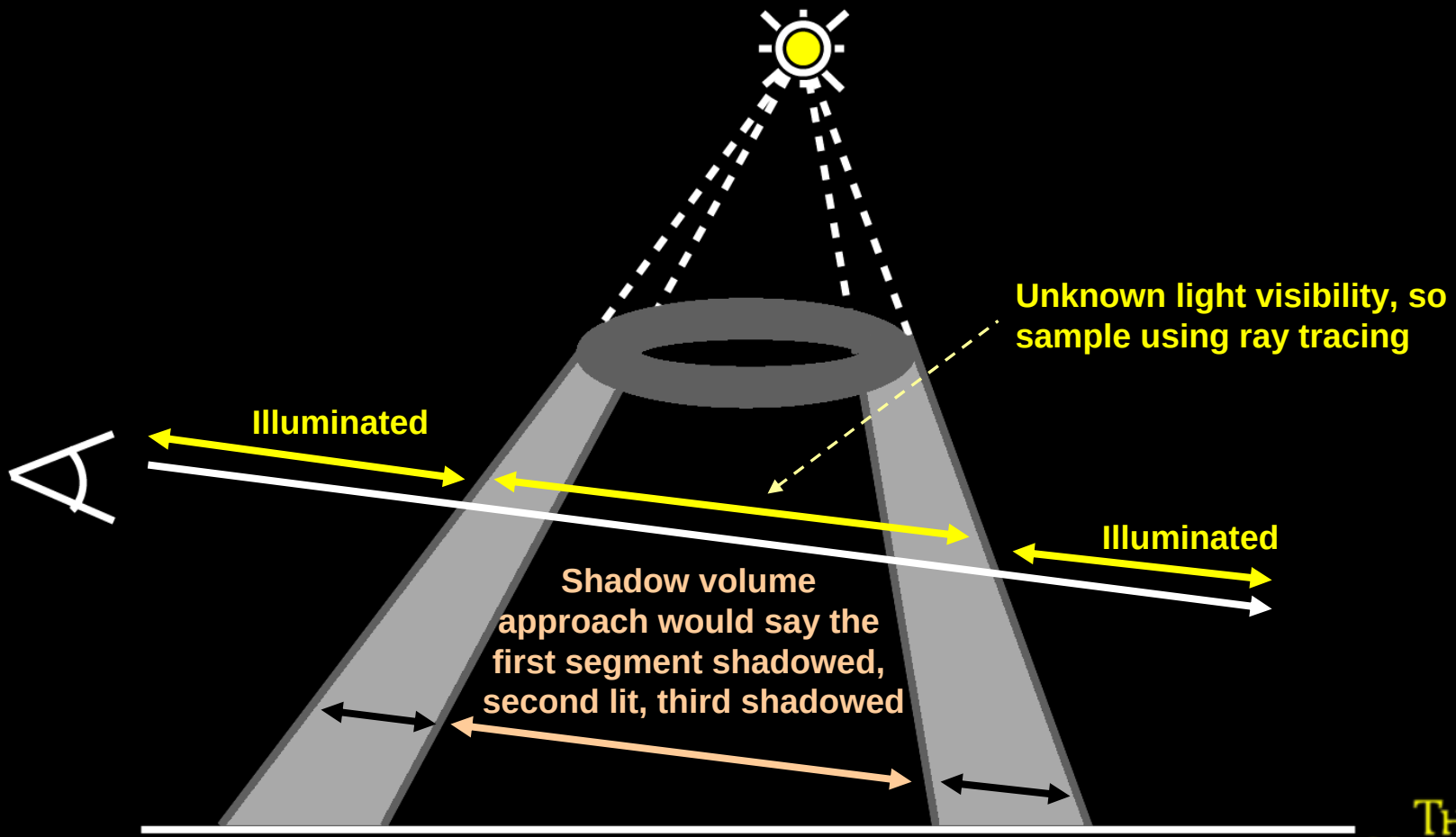
- Shadow volumes
  - Sample *only* at shadow boundaries → no aliasing
  - Knows nothing about light color
  - Requires ordering of polygons
    - Disambiguate cases, find front / back pairs
    - Not all shadow polygons matter

Our Motivation

- ***Not:***
  - Merely accelerate or improve brute force ray tracing

- Interactive volumetric shadows
  - Reduce aliasing along shadow boundaries
  - Use more intelligent sampling
    - Fewer wasted samples
    - Reduce coherent samples, esp. if they introduce banding
    - Sample light color and visibility separately
    - Use geometry info to pick samples! (i.e., a hybrid)

Algorithm Overview

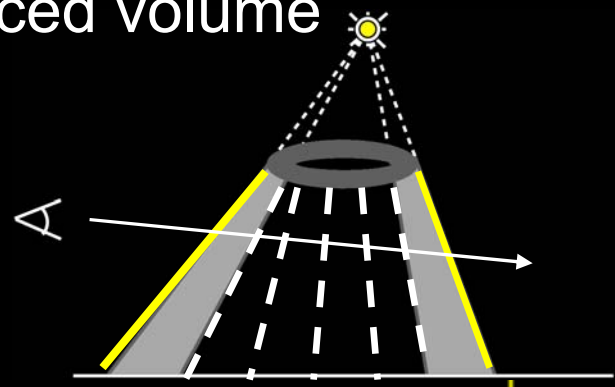


Basic Hybrid Algorithm

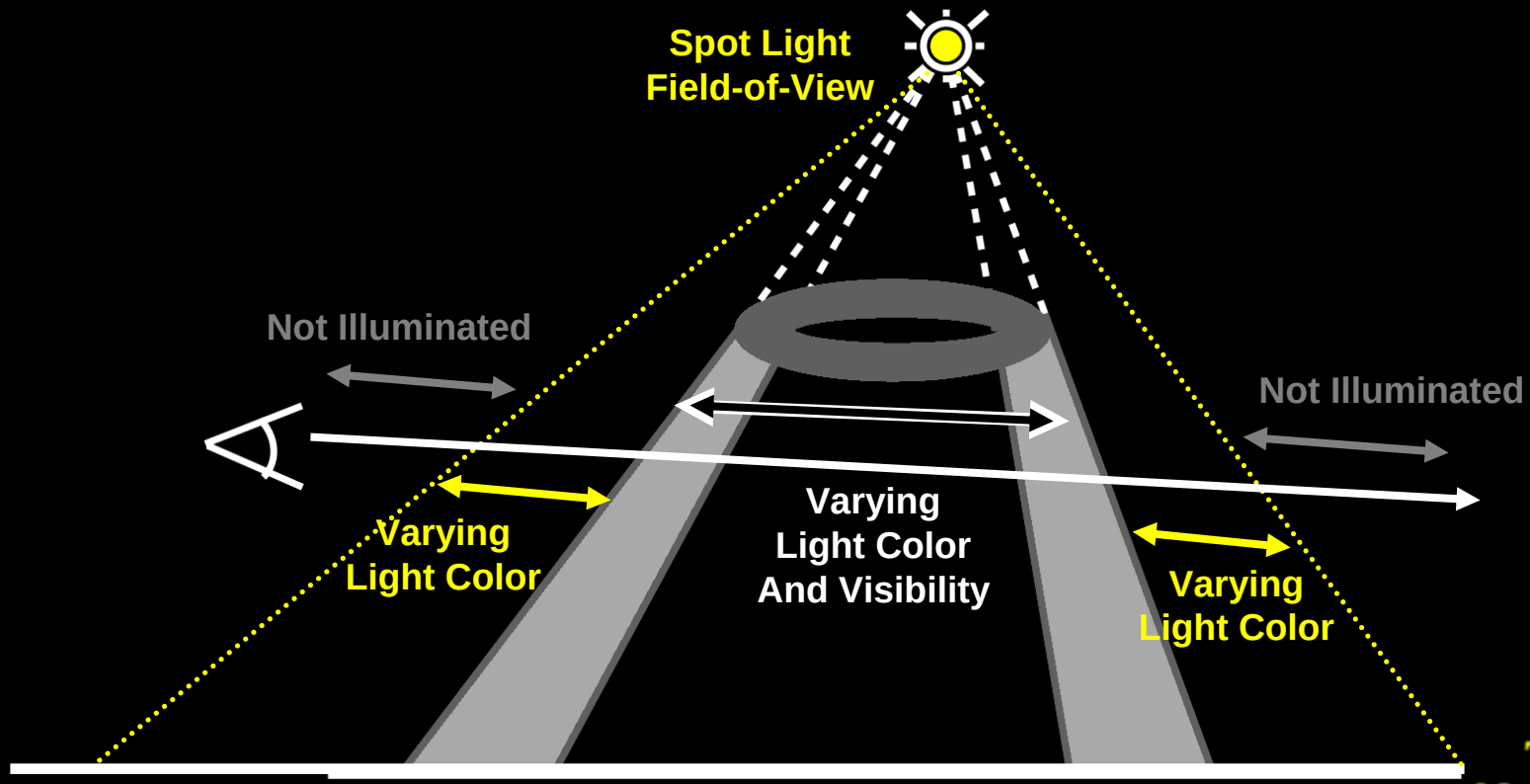
1. Render shadow map
   - Used instead of shadow rays when ray marching
2. Compute shadow volumes
   - Front & back bound region of varying visibility
3. Render scene without volume media
4. Compute scattering at every pixel
   - Preintegrated scattering for lit intervals (e.g., [Sun05])
   - Ray march to determine visibility between shadow volumes
   - No ray marching for pixels w/o shadow volumes
5. Combine into final image

Basic Hybrid Discussion

- Pluses:
  - Avoid ray marching in all pixels
  - Eliminate aliasing at front (and last) shadow boundary
  - Ray march smaller volume per ray → fewer samples
  - Uniform sampling works well in reduced volume
- Minuses:
  - Assumes constant color light
  - Assumes point light source
  - May not reduce sampling in all scenes



Adding Textured Spotlights



Textured Spotlight Hybrid

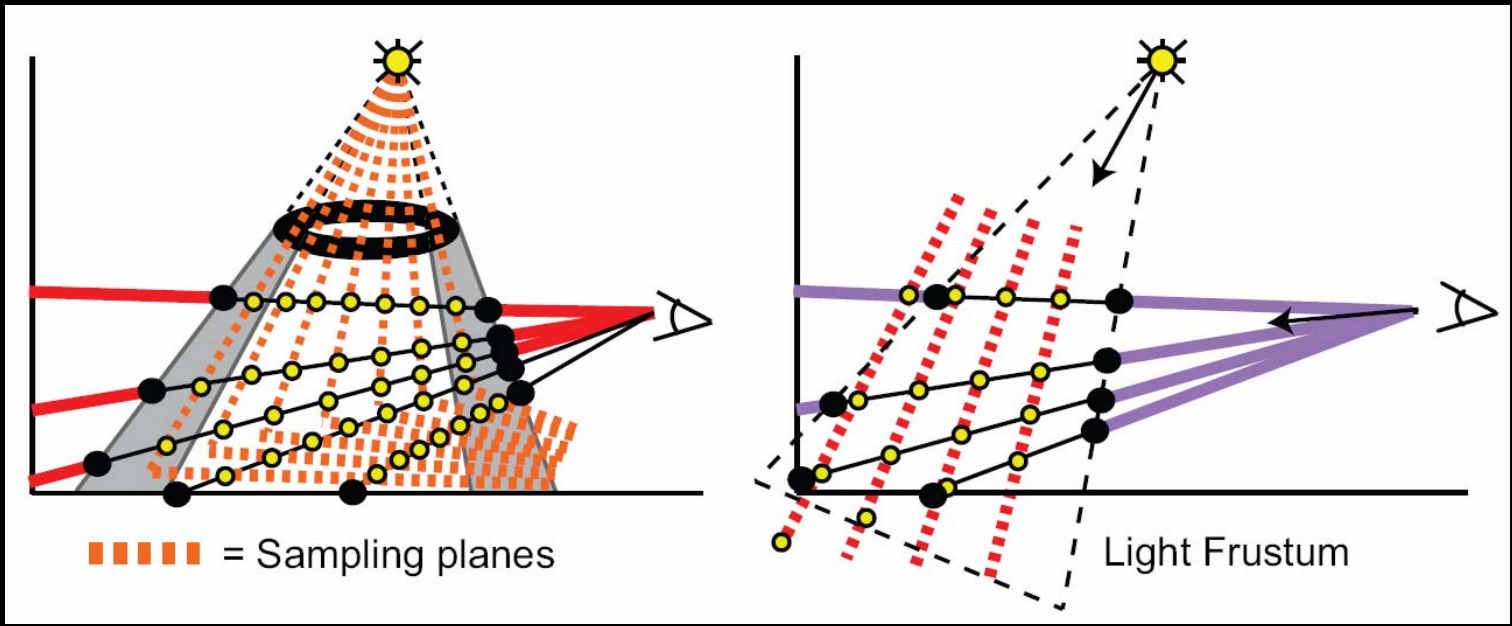
1. Render shadow map
2. Compute light frustum
3. Compute front & back shadow volume quads
4. Render scene without volume media
5. Compute scattering at every pixel
   - Where ray misses light frustum, scattering = 0
   - Where no shadows, march back to front in frustum
   - Where shadows, march in 3 segments
     - Fully lit, between shadow polygons, fully lit
6. Combine into final image

Sampling Issues

- Both lit & shadowed segments need samples
  - But sampling rates need not be the same!
  - Visibility has sharp edges
    - More samples
  - Lighting tends to get blurred by media
    - Especially with multiple scattering
    - Can prefilter lighting, e.g., mipmap
    - Use fewer samples

Sampling Issues

- Illumination samples add complications
  - Sampling plane discontinuities seen in image
- Align sample planes between view & light frusta
  - Similar to [Kniss03]

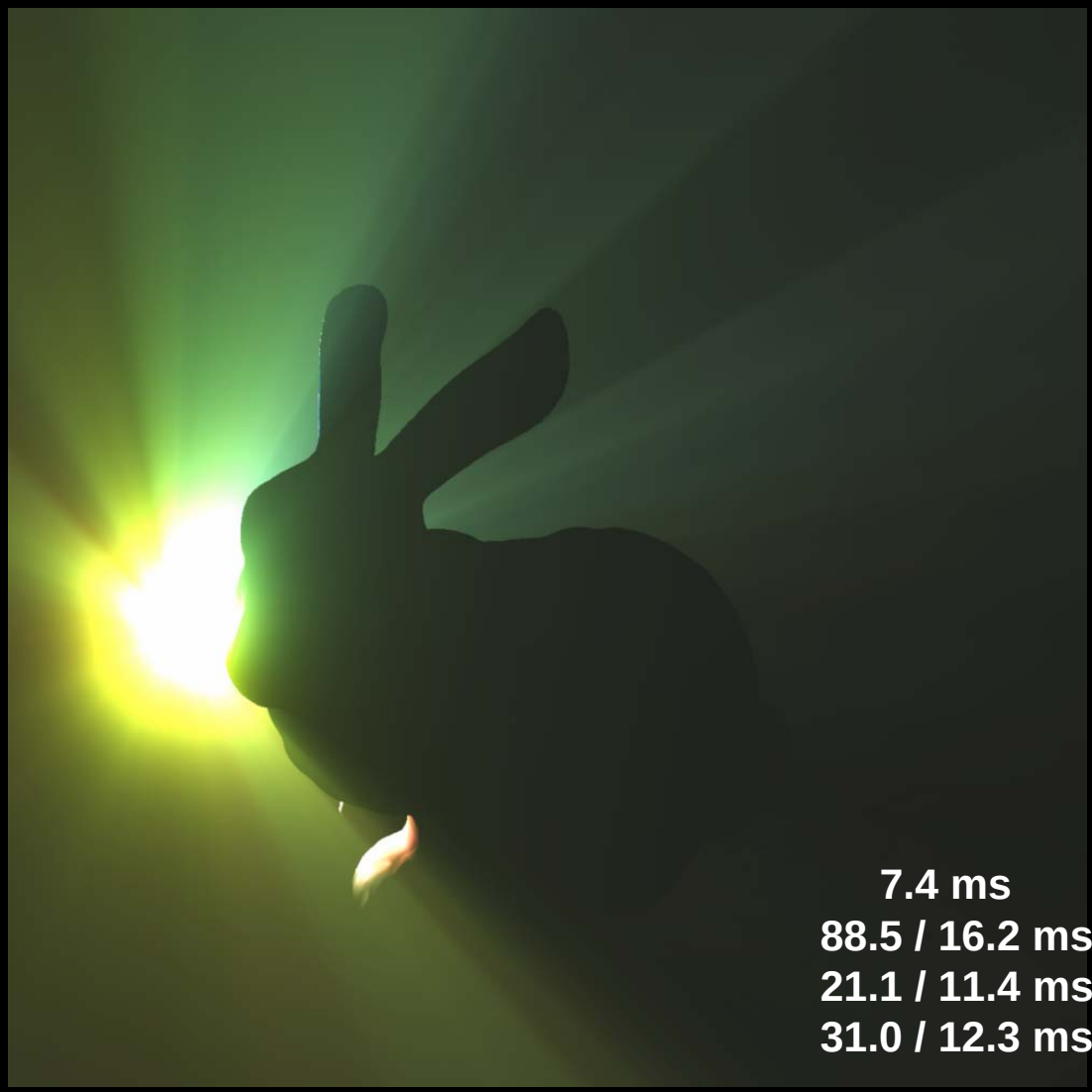


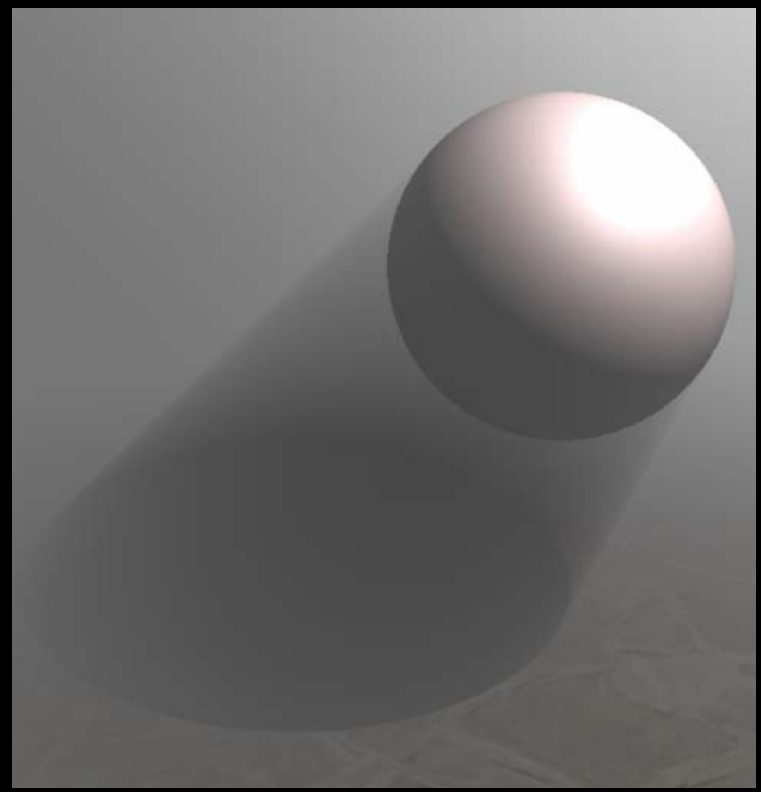
Other Optimizations

- Compute scattering @ lower resolution
  - Blurring effect of participating media
  - Upsample during final compositing
    - Use a bilateral filter (similar to [Sloan07])

- Simplified model for shadow volume extraction
  - Tradeoff between quality and speed
    - Aliasing along shadows with low poly-count models
    - More expensive extraction with higher poly-counts
  - We mainly use full-resolution models

Results (Runs entirely on nVidia 8800 GTX)



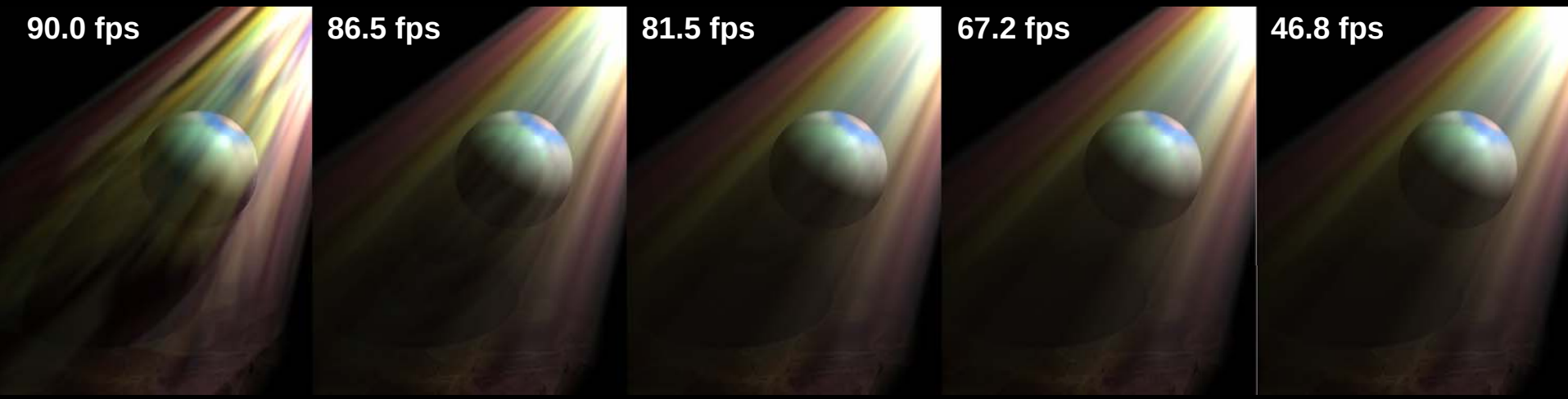


7.4 ms (no media)
88.5 / 16.2 ms (brute ray march)
21.1 / 11.4 ms (simple hybrid)
31.0 / 12.3 ms (texture hybrid)

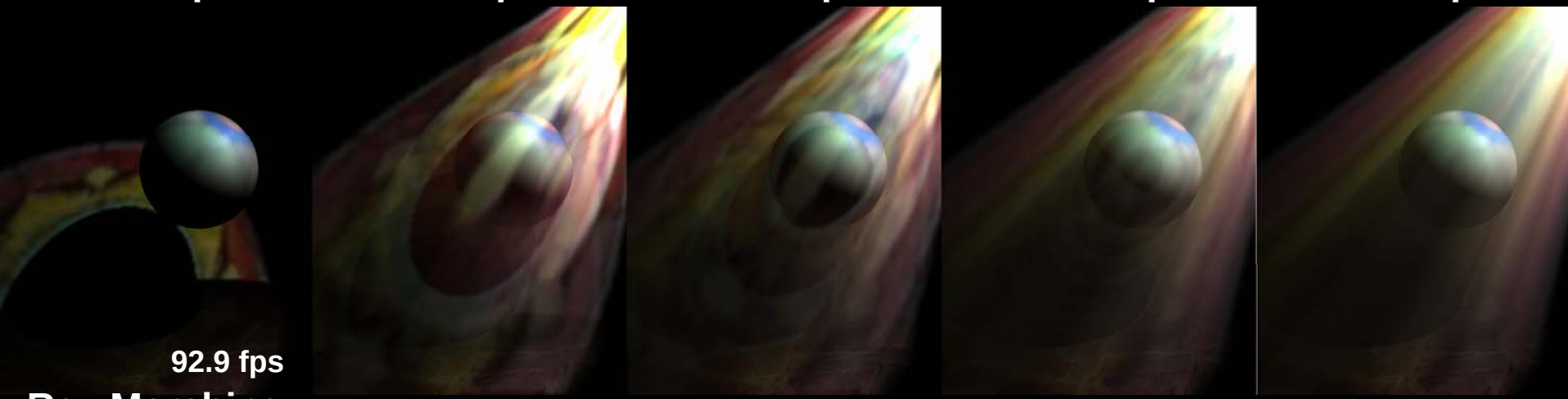
THE UNIVERSITY OF IOWA

Results

Our Results



10 samples | 50 samples | 100 samples | 300 samples | 800 samples



Ray Marching

60.2 fps
Texture Hybrid

61.0 fps
Ray Marching (Equal Speed)

27.1 fps
Ray Marching (Equal Quality)

THE UNIVERSITY OF IOWA

Video

Limitations

- Difficult cases:
  - Mostly shadowed scenes
    - Solution: March inside lit region
  - Frequent thin shadows
    - Possible solution: Combine visibility & light into texture
    - Only nearby scattering is clearly distinguishable

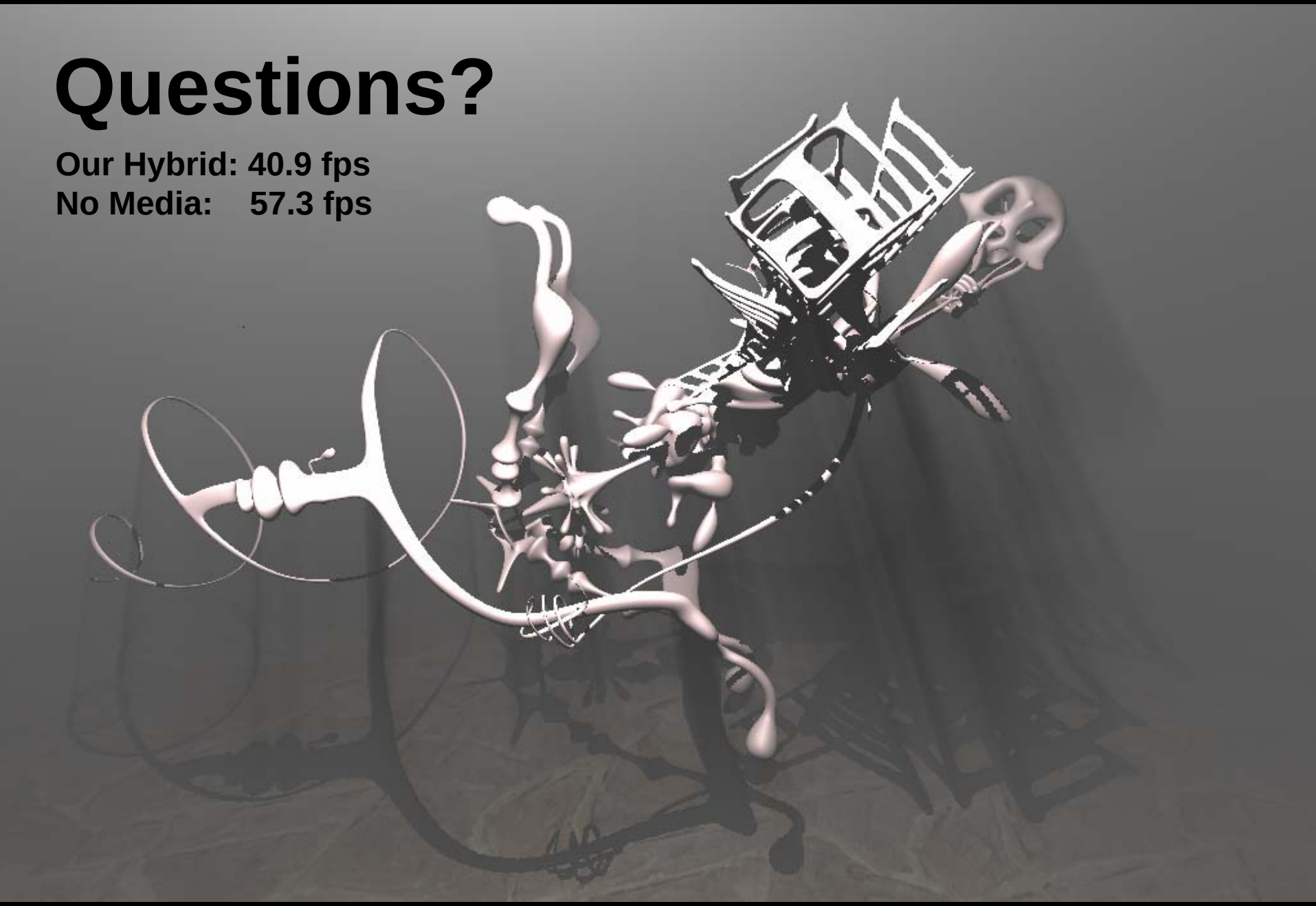
- Area lights not handled
  - Use penumbra wedges instead of shadow volumes?

Conclusions

- A hybrid algorithm for volumetric shadows
  - Combines ray marching and shadow volumes
  - Significantly reduces visibility samples needed
  - Separate visibility and illumination samples
    - Can sample them at different rates
- Future work
  - Shadow volume reconstruction
    - Eliminate extra shadow polys & further reduces samples
  - Build acceleration structure of shadow polys
  - Penumbra wedges for area light sources
  - Shadow volume culling techniques?

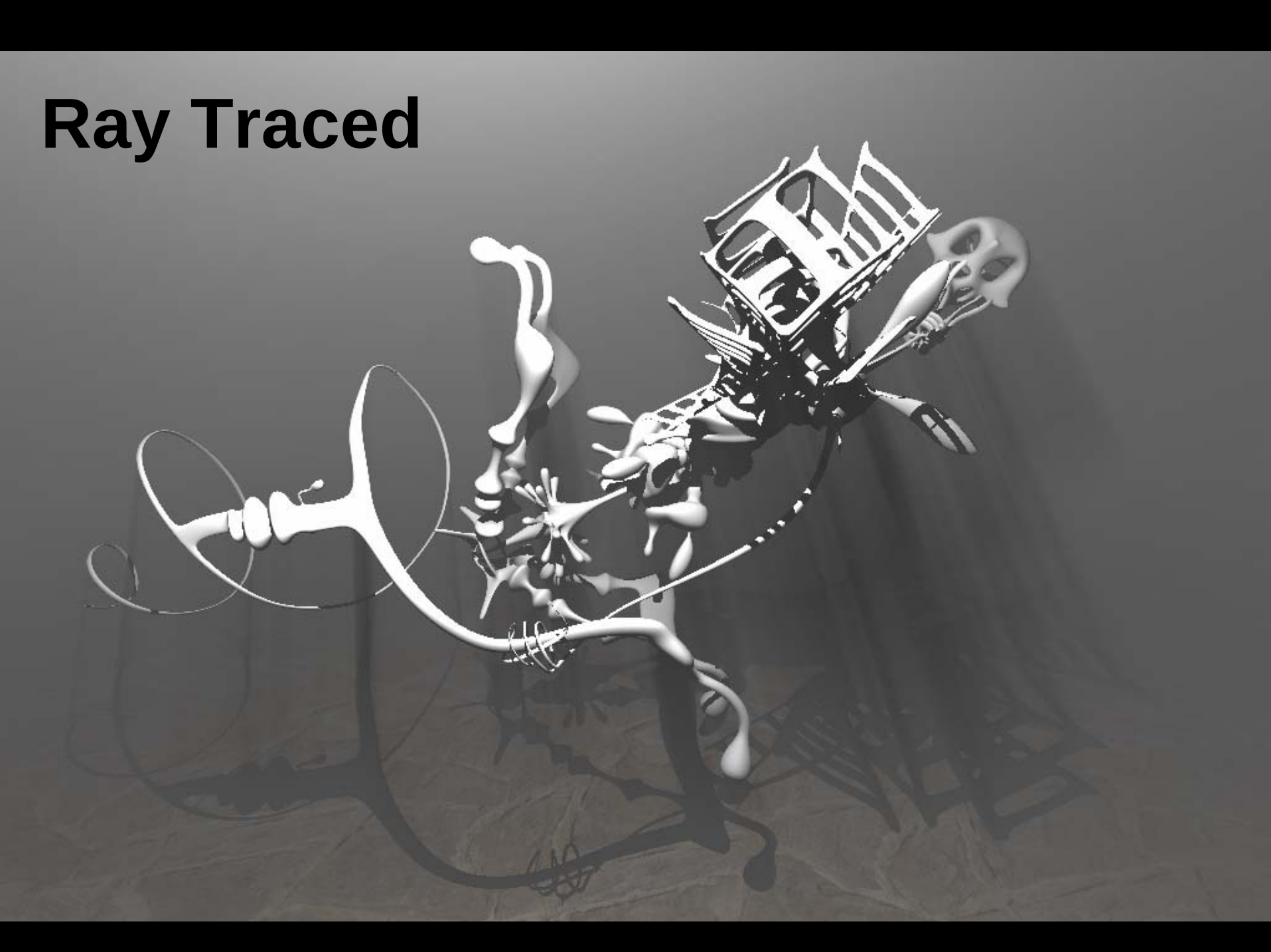
Questions?

Our Hybrid: 40.9 fps
No Media: 57.3 fps



Extra Slides…

Ray Traced



Fill Rate an Issue?

- Not really…
  - Shadow volumes take lots of fill rate
  - But have to touch all those pixels anyways
  - Can get all data in 1 pass thru shadow volumes
  - Depth test eliminates lots of work
    - Doesn't usually happen with surface shadows
- Yes, it takes fill rate
  - Made up by eliminating samples
  - Except in very simple scenes

More Extensive Timings

Scene & # Triangles (results in frames per second)	Timings for final images rendered at 1024^2						
	Single colored light source					Textured light	
	No Volumetric Shadows	Brute force ray marching, 150 spp		Ray marching in shadow volume, 50 spp		Ray marching in spotlight volume, 150 spp	
		@1024^2	@256^2	@1024^2	@256^2	@1024^2	@256^2
Sphere (20k)	120.5	12.9	59.8	85.0	99.6	45.5	83.5
Elephant (24k)	137.2	12.6	57.7	89.6	101.1	33.5	78.2
Spring (32k)	160.7	14.6	73.1	76.8	112.5	18.0	72.5
Bunny (70k)	135.0	11.9	61.6	47.5	87.5	32.3	81.3
Fairy (155k)	121.9	12.3	63.5	38.5	72.1	35.9	78.3
Buddha (250k)	96.1	12.2	46.9	50.2	61.4	28.3	61.7
YeahRight (755k)	57.3	11.1	32.8	26.3	33.8	16.1	40.9

THE UNIVERSITY OF IOWA