

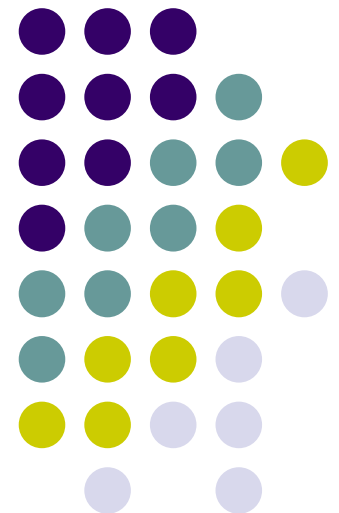
6.189 IAP 2007

Student Project Presentation

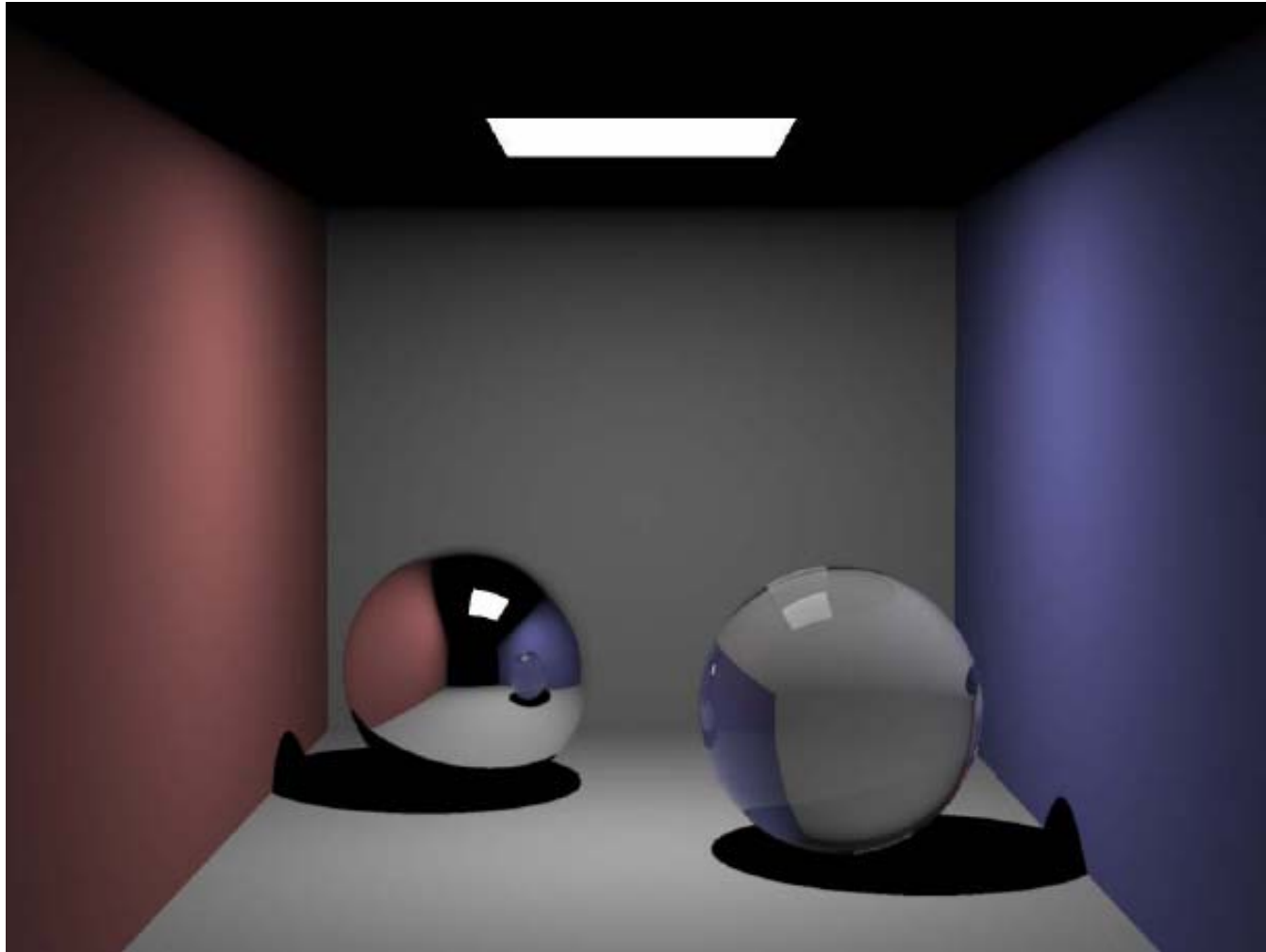
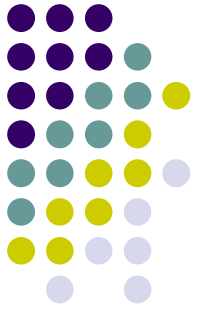
Global Illumination

Global Illumination

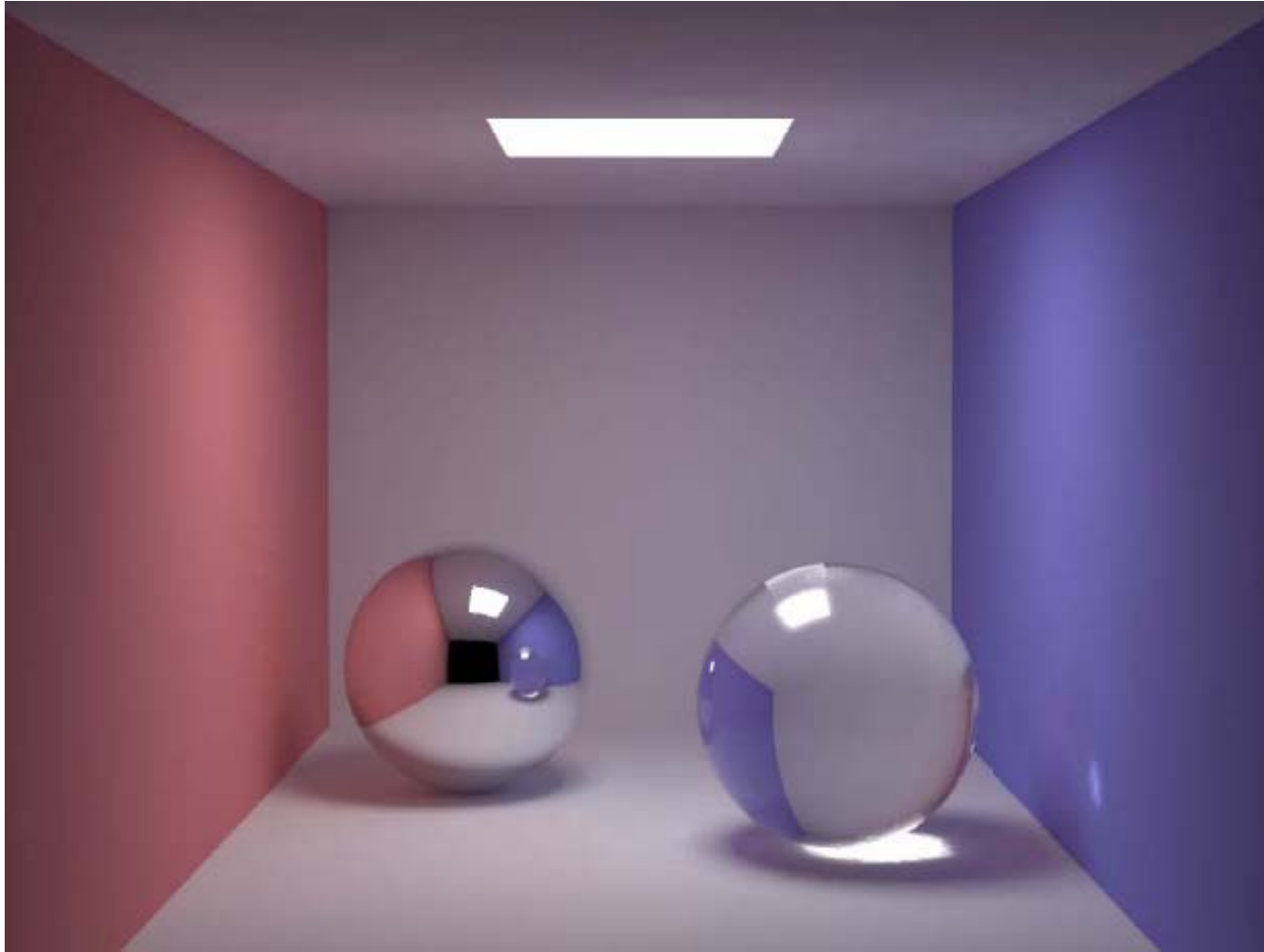
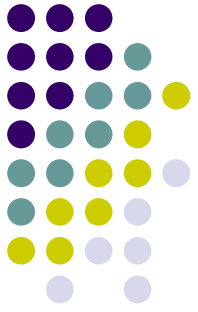
Pramook Khungurn



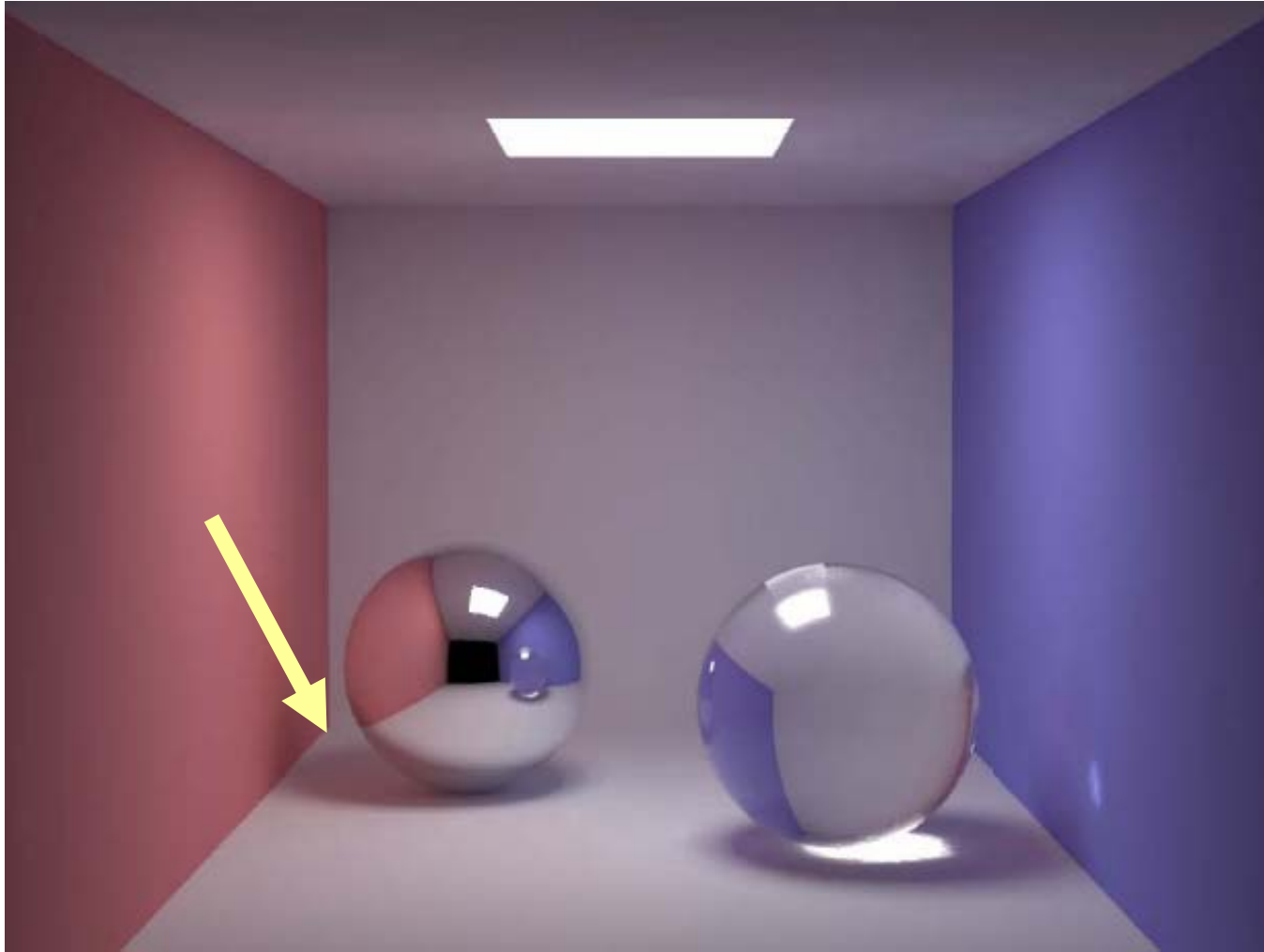
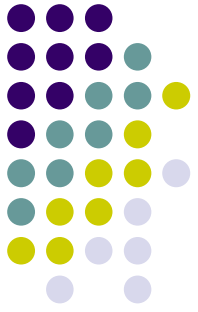
Direct Illumination



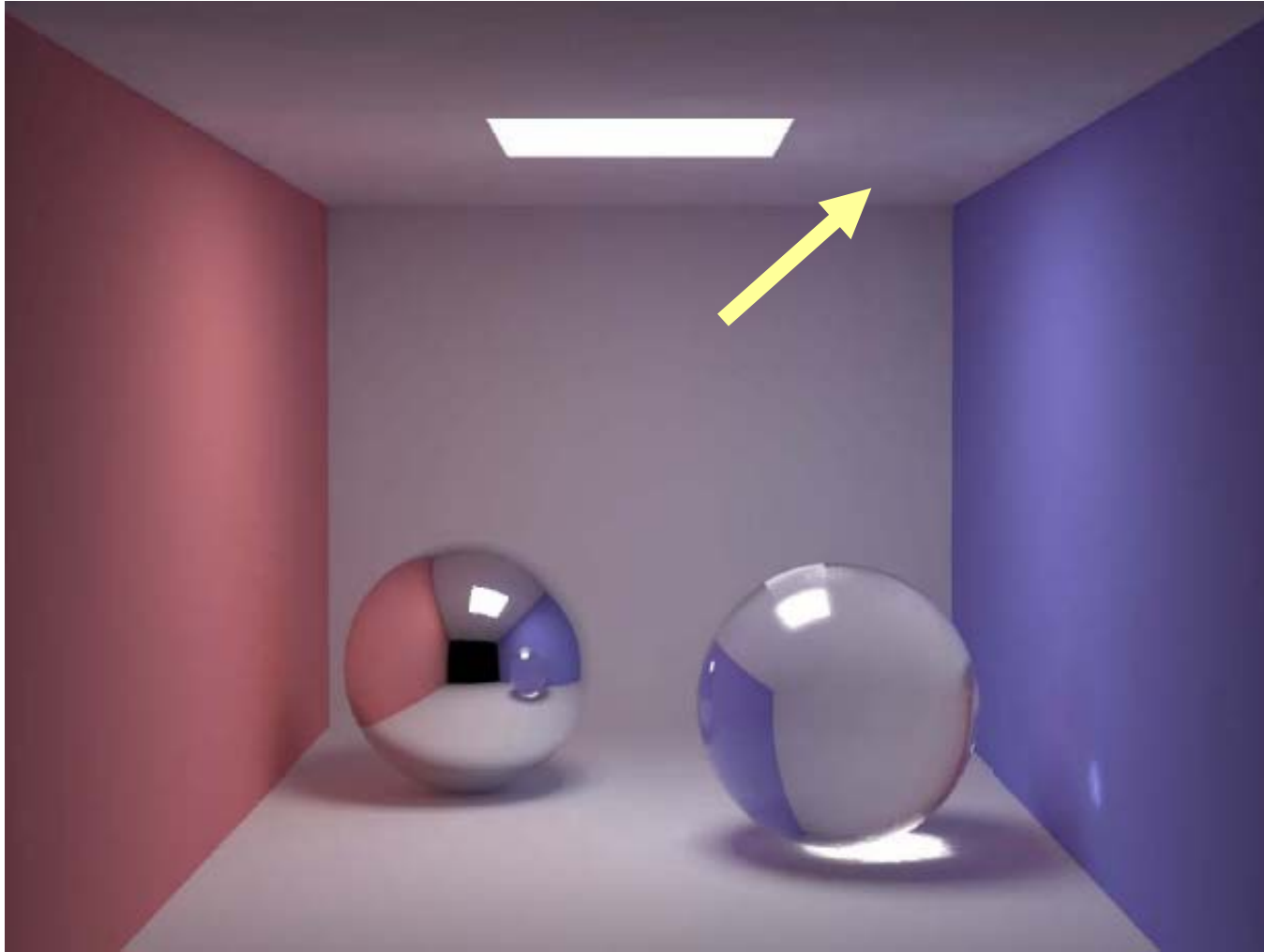
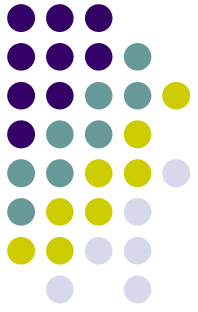
Global Illumination



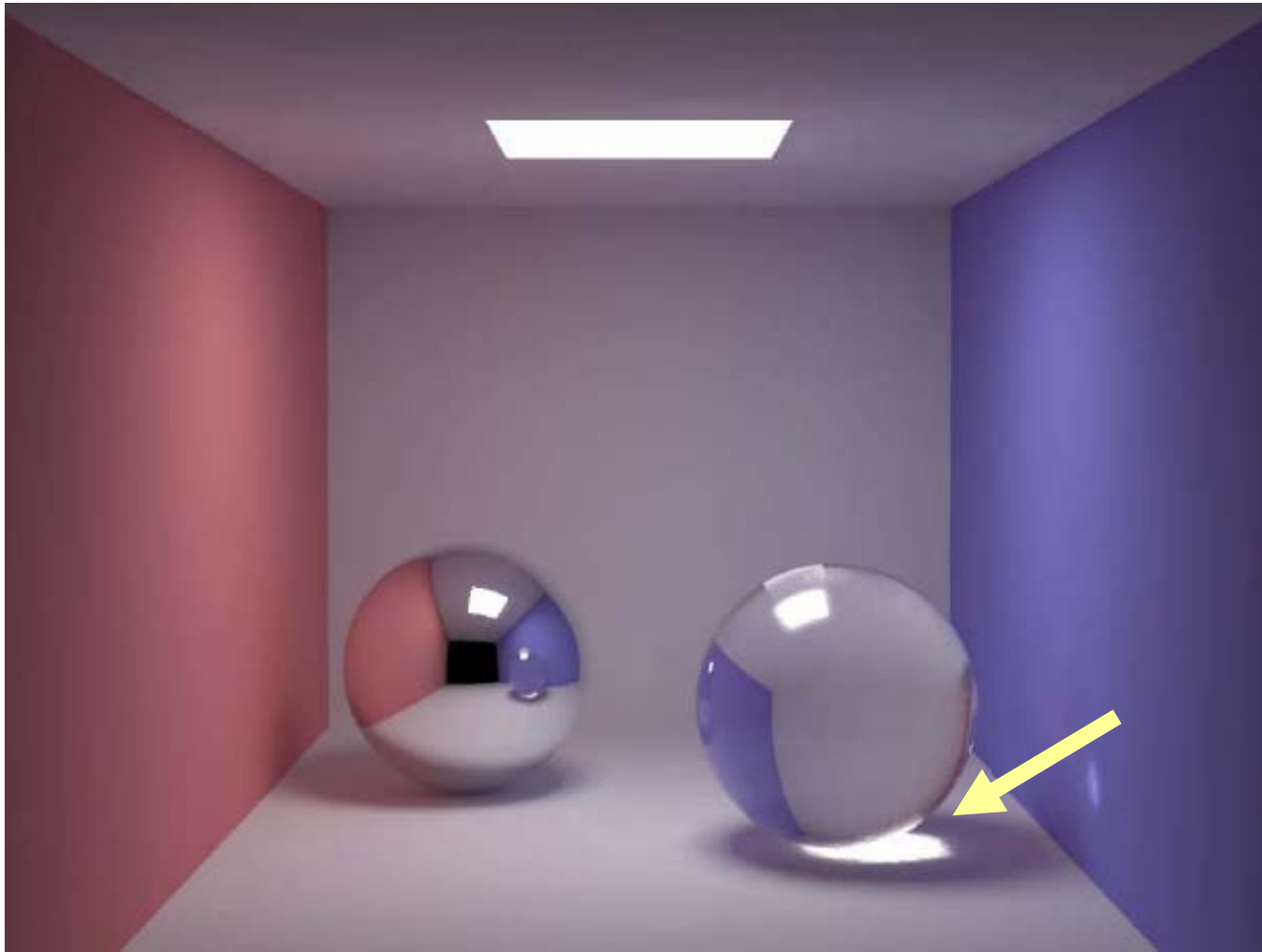
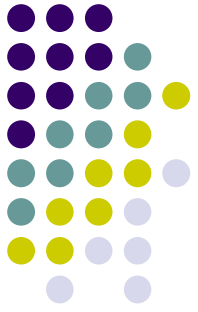
Soft Shadows

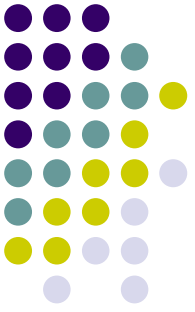


Color Bleeding



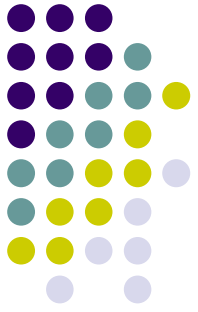
Caustics





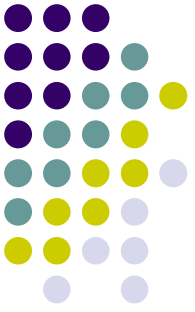
Goals

- Globally illuminate the Cornell Box.
- In real time!
- With caustics.
 - Have to use photon mapping.



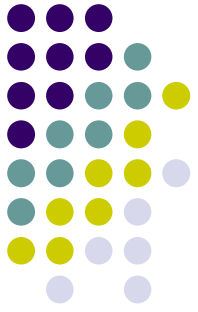
Accomplished Goals

- Globally illuminate the Cornell Box.
- ~~In real time! (Too slow...)~~
- ~~With caustics~~
 - ~~Have to use photon mapping~~
- Sadly, there's still a lot of rooms for optimization.



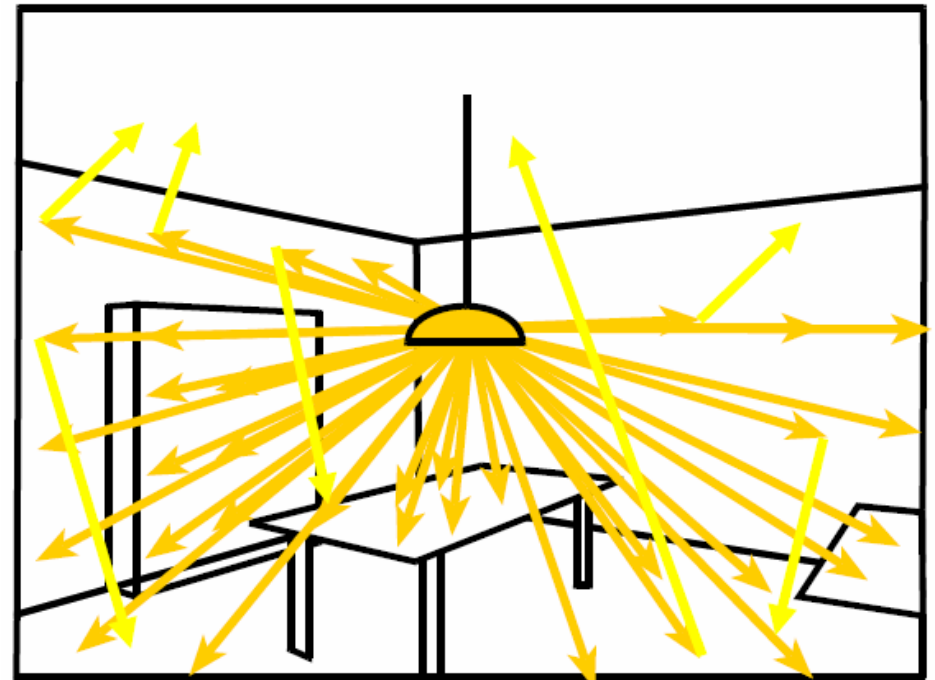
Instant Radiosity

- Alexander Keller, “Instant Radiosity,” SIGGRAPH, 1997
- Good for scenes with diffuse objects.
- Can’t do caustics though.
- Very simple.

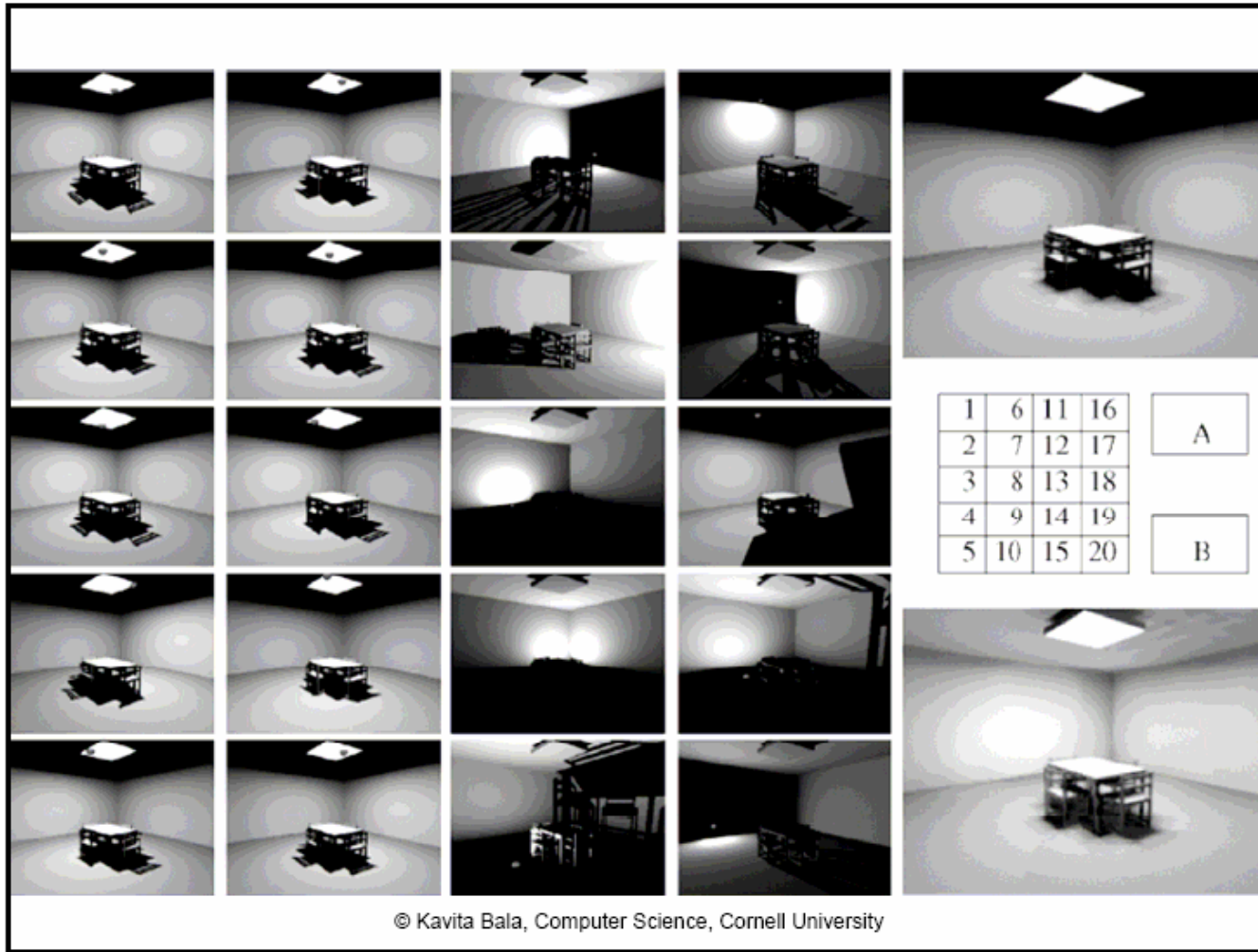
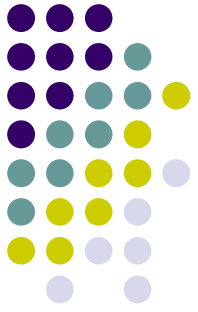


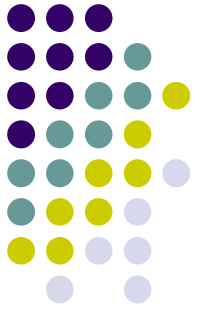
Instant Radiosity (cont.)

- Emit “artificial light sources.”
- Each has a fixed number of “bounces.”
- Raytrace the scene with these new light sources.



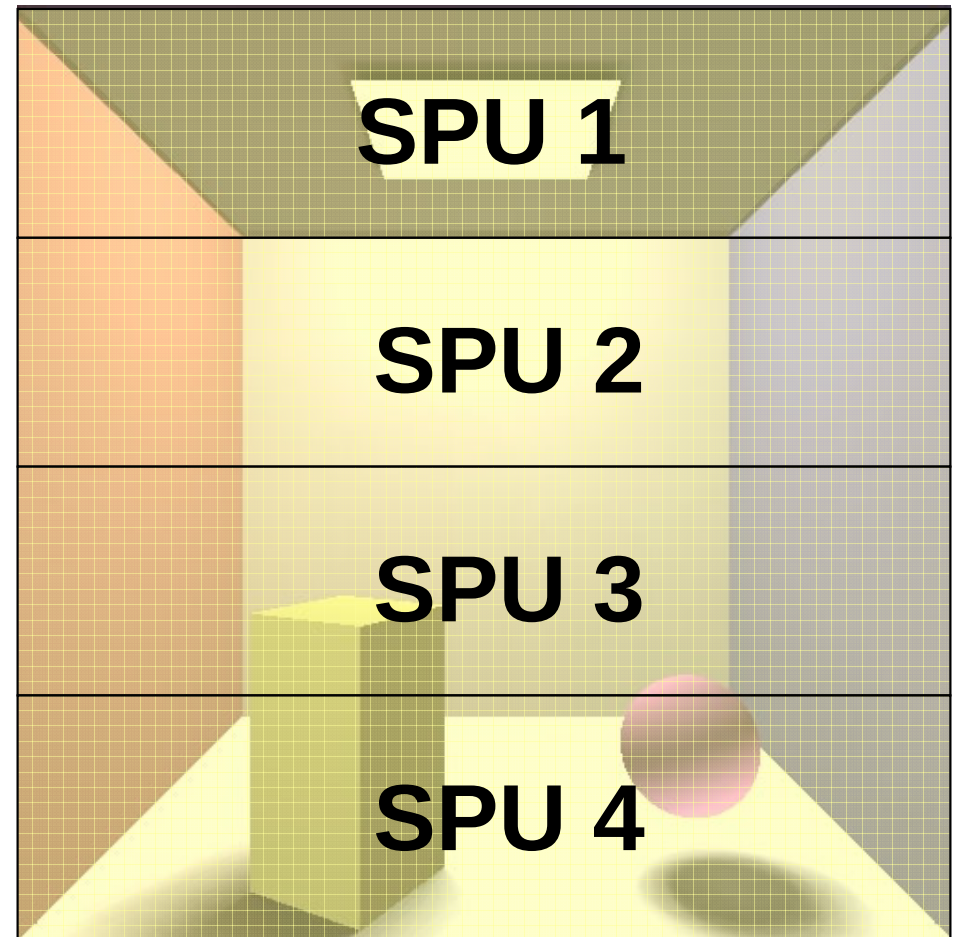
Instant Radiosity (cont.)

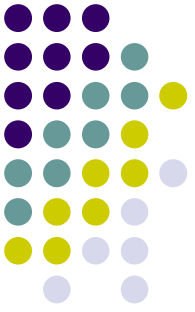




Implementation

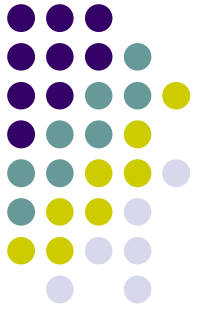
- Have 1 SPU scatter the artificial light sources.
- Embarassingly parallel rendering.
- Scene is small, so no worries about memory.
- Bruteforce ray shooting.





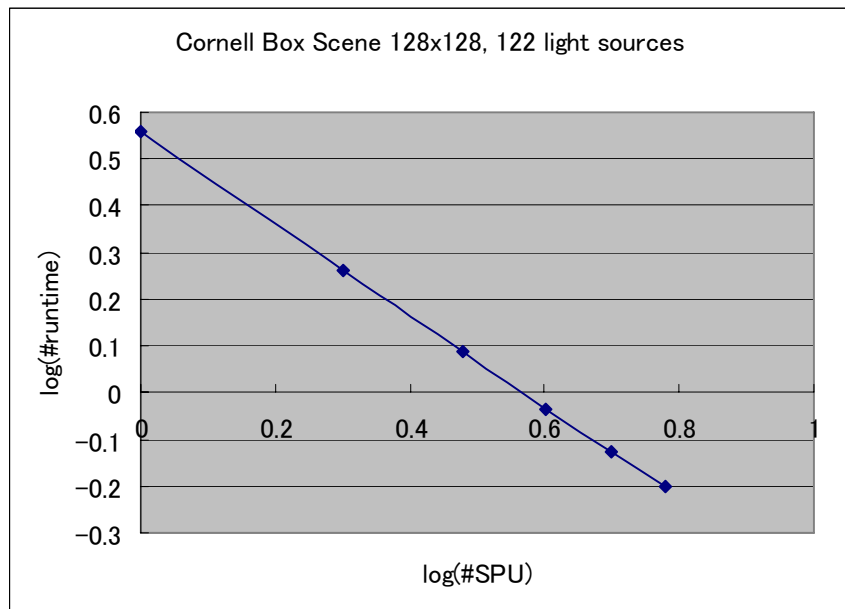
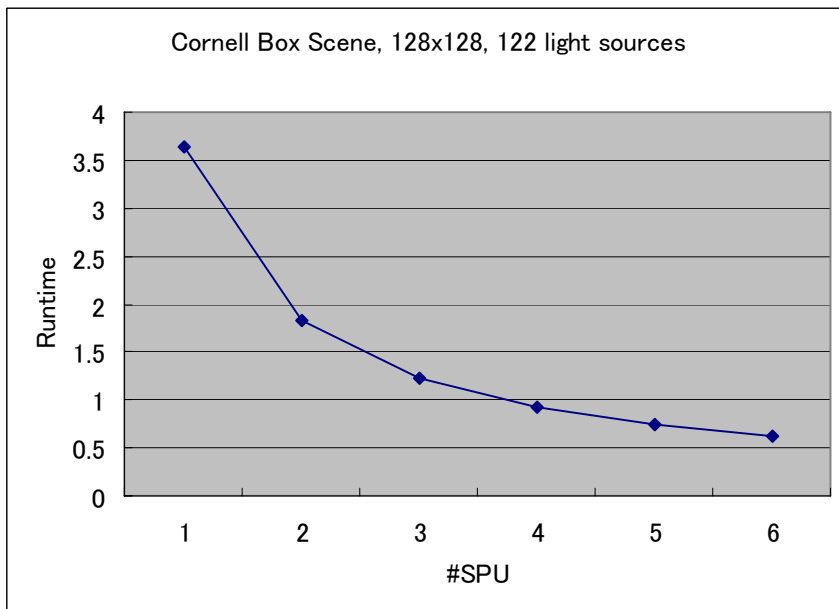
Performance

- $O(\text{image size} \times \text{lights} \times \text{objects})$
 - Instant radiosity fast on GPU, not on Cell.
- I wish I could have done better.
 - 20 seconds to render a simple scene at 512x512
 - Still far from interactive rate.

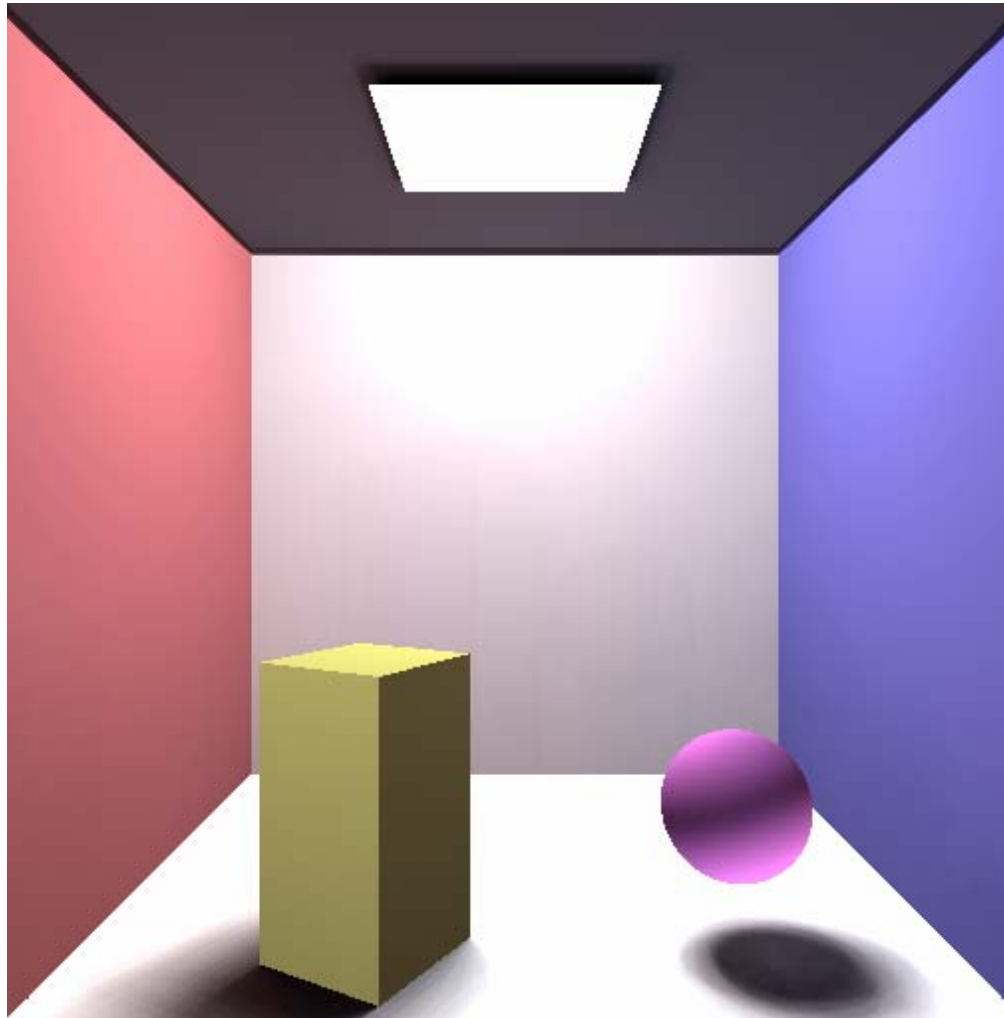
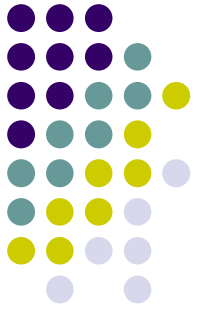


Performance (cont.)

- Speed up = number of processors used.

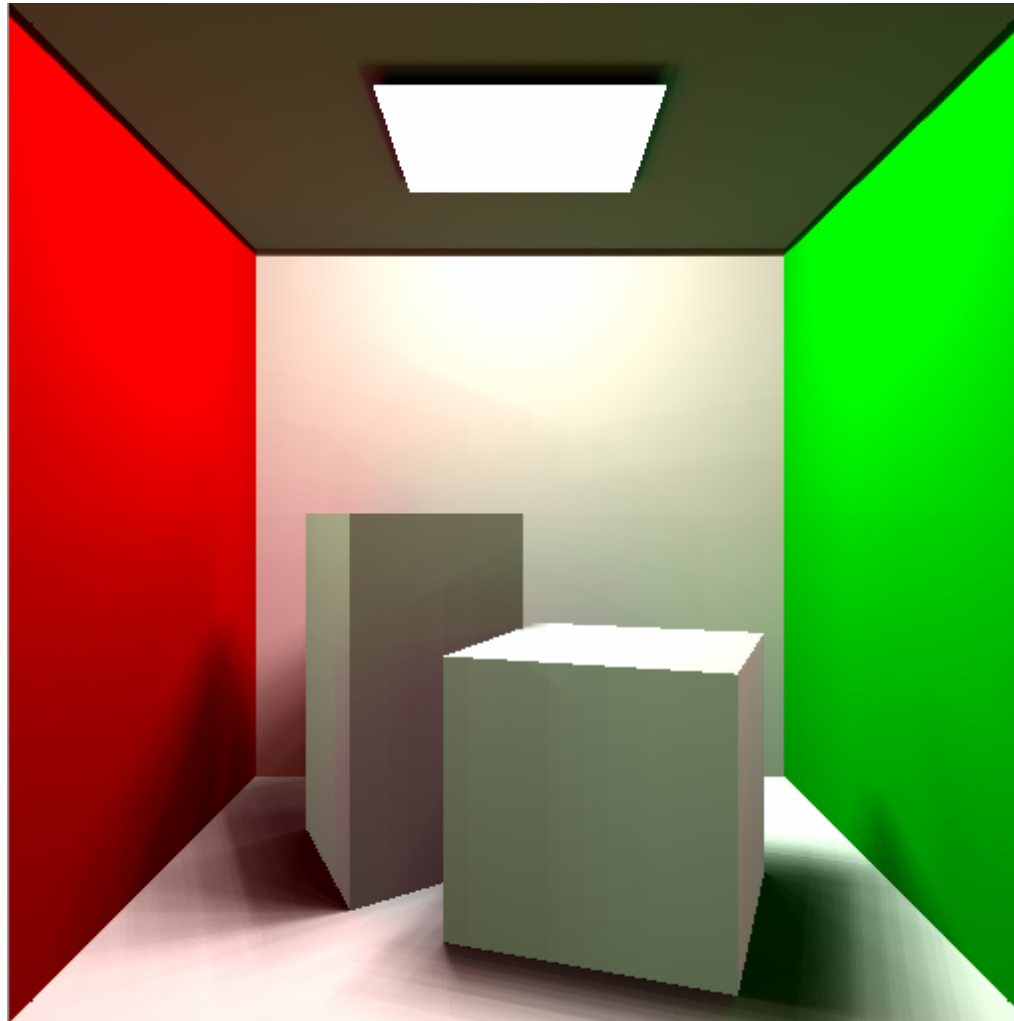
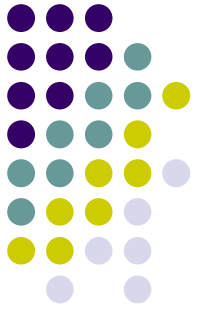


Results



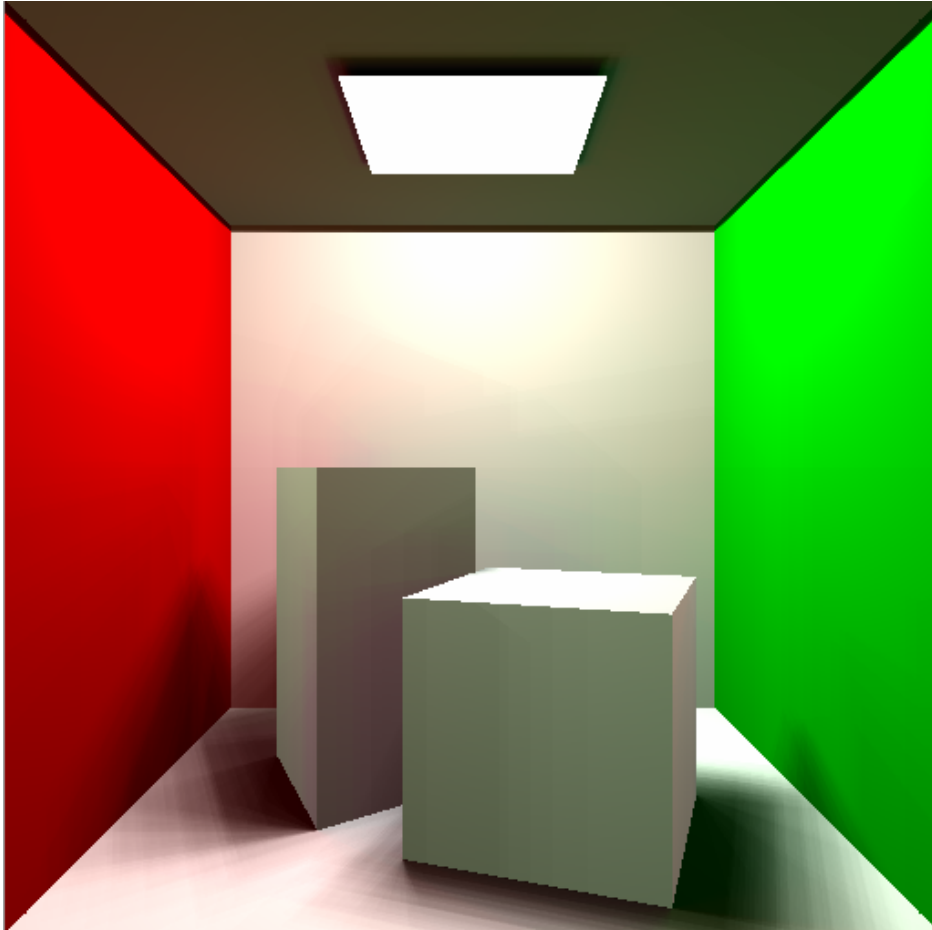
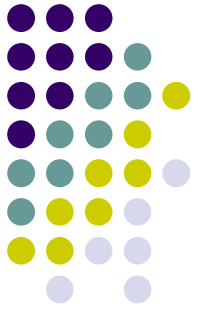
512x512, 8 objects, 251 light sources, 20.5 seconds with 6 SPUs

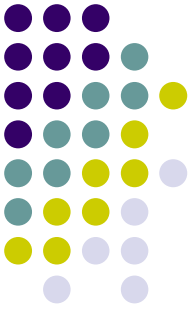
Results (cont.)



512x512, 8 objects, 253 light sources, 20.2 seconds with 6 SPUs

Results (cont.)





Further Optimization

- Accelerated structure for geometries.
 - $\#objects \rightarrow \log(\#objects)$
 - In my case, $8 \rightarrow 3$. So about 3x speed up.
- Tracing packet of rays.
- Triangles as primitives.
 - Transforms are costly.



Thank you.