Lightcuts: A Scalable Approach to Illumination

Bruce Walter, Sebastian Fernandez,
Adam Arbree, Mike Donikian,
Kavita Bala, Donald Greenberg

Program of Computer Graphics, Cornell University

Lightcuts

- Efficient, accurate complex illumination

Environment map lighting & indirect
Time 111s

Textured area lights & indirect
Time 98s

(640x480, Anti-aliased, Glossy materials)

Scalable

- Scalable solution for many point lights
  - Thousands to millions
  - Sub-linear cost

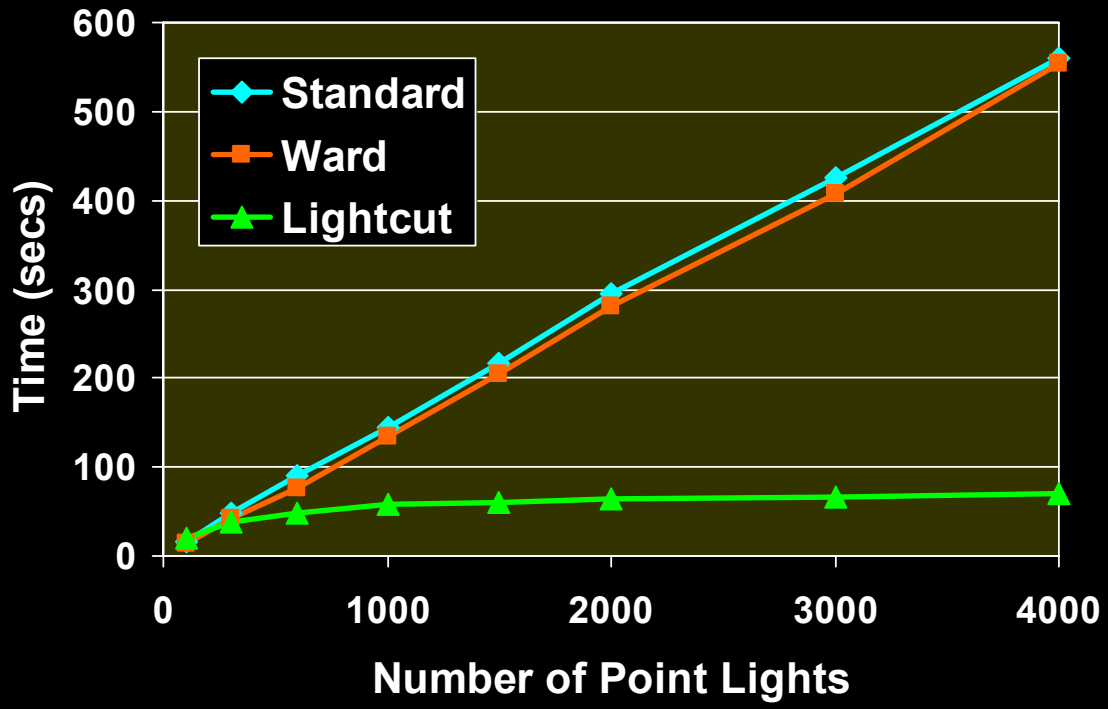


Tableau Scene

Complex Lighting

- Simulate complex illumination using point lights
  - Area lights
  - HDR environment maps
  - Sun & sky light
  - Indirect illumination
- Unifies illumination
  - Enables tradeoffs between components

Area lights + Sun/sky + Indirect

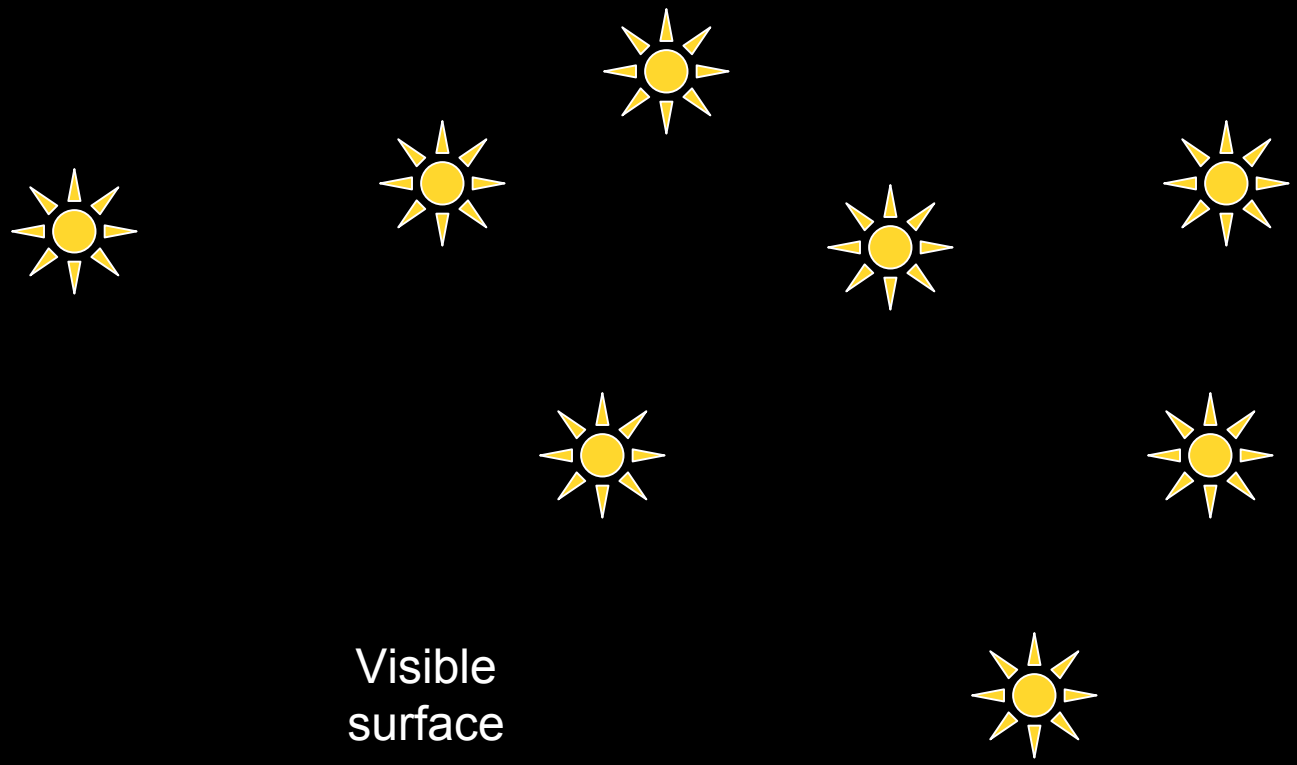
Related Work

- Hierarchical techniques
  - Hierarchical radiosity [eg, Hanrahan et al. 91, Smits et al. 94]
  - Light hierarchy [Paquette et al. 98]
- Many lights
  - [eg, Teller & Hanrahan 93, Ward 94, Shirley et al. 96, Fernandez et al. 2002, Wald et al. 2003]
- Illumination coherence
  - [eg, Kok & Jensen 92, Ward 92, Scheel et al. 2002, Krivanek et al. 2005]
- Env map illumination
  - [Debevec 98, Agarwal et al. 2003, Kollig & Keller 2003, Ostromoukhov et al. 2004]
- Instant Radiosity
  - [Keller 97, Wald et al. 2002]

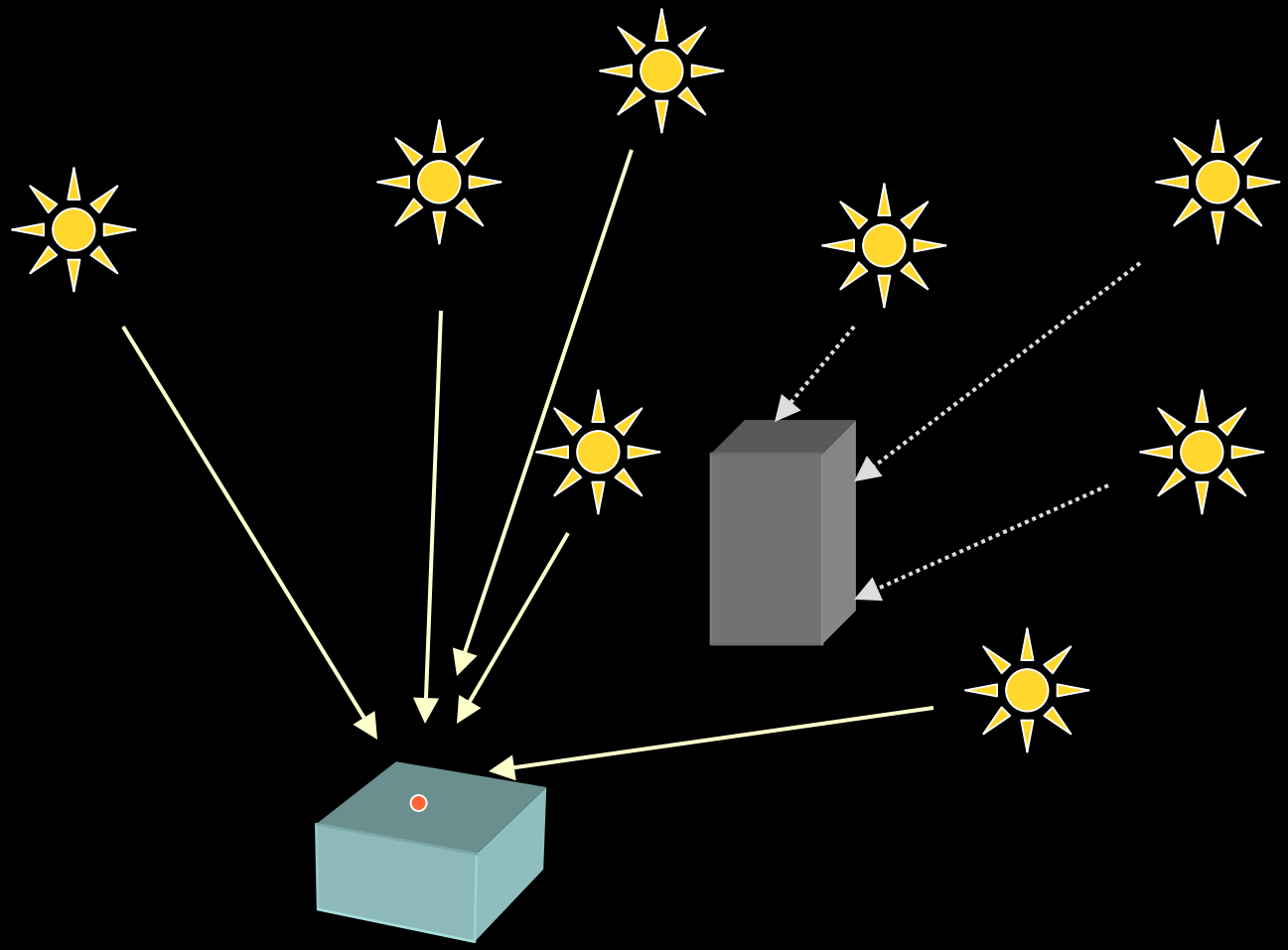
Talk Overview

- Lightcuts
  - Scalable accurate solution for complex illumination
- Reconstruction cuts
  - Builds on lightcuts
  - Use smart interpolation to further reduce cost

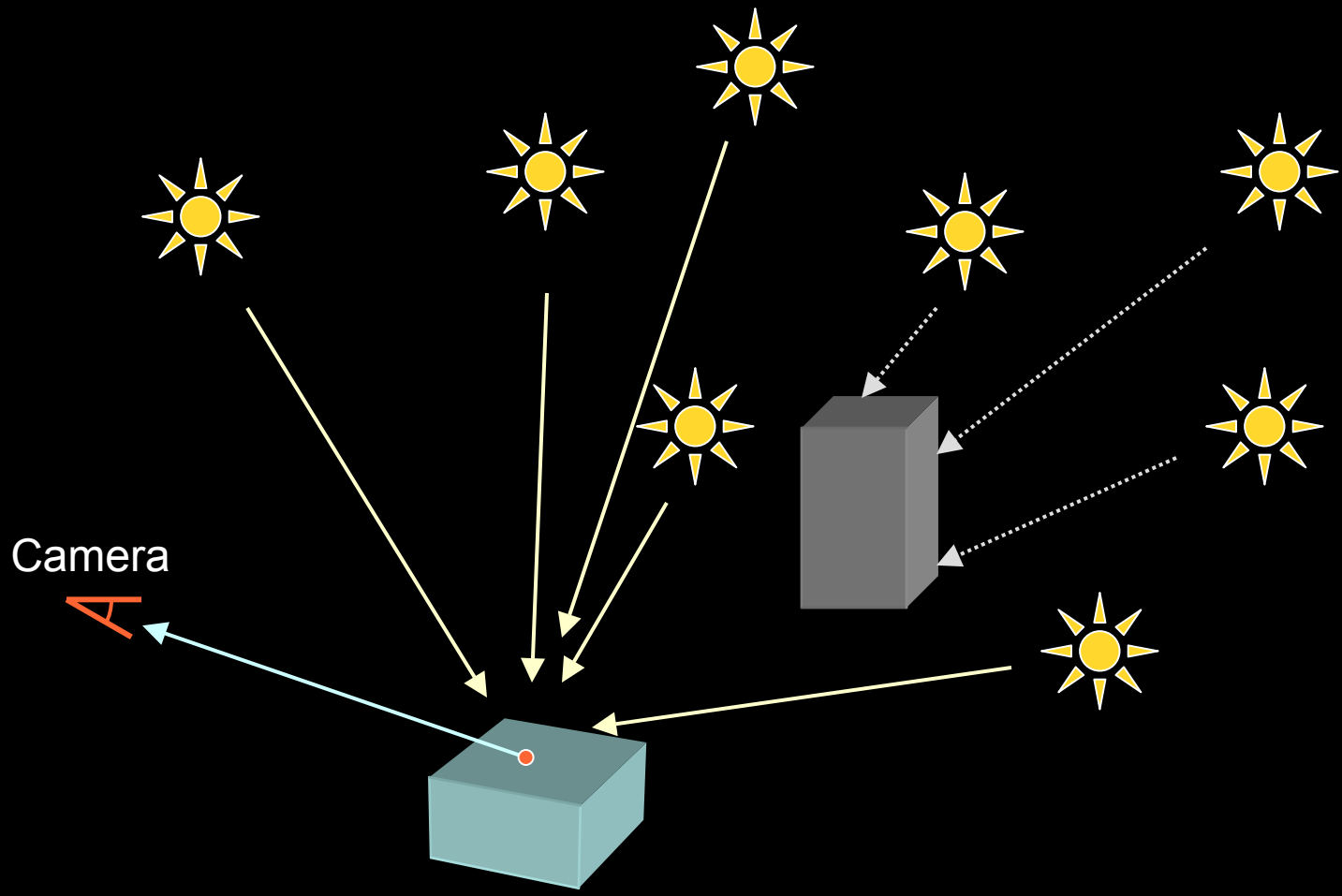
Lightcuts Problem



Lightcuts Problem

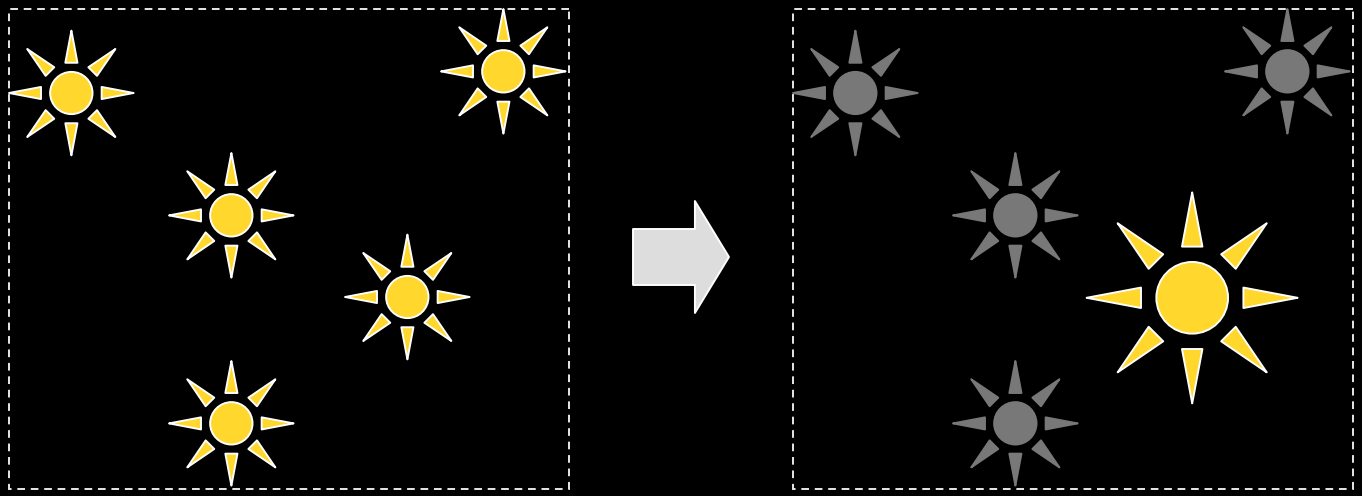


Lightcuts Problem



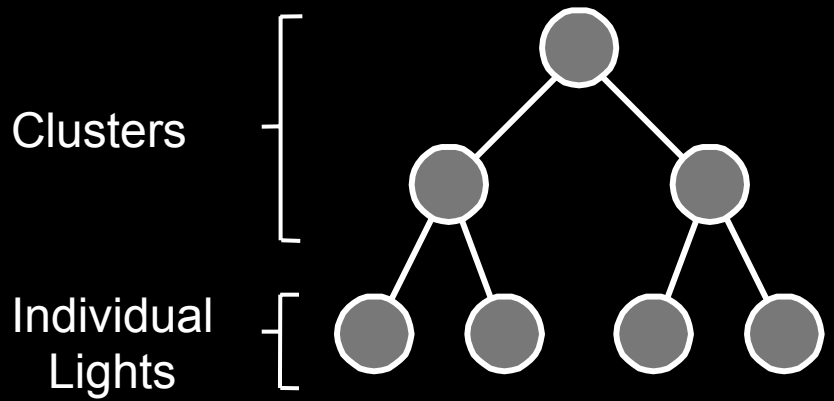
Key Concepts

- Light Cluster
  - Approximate many lights by a single brighter light (the representative light)



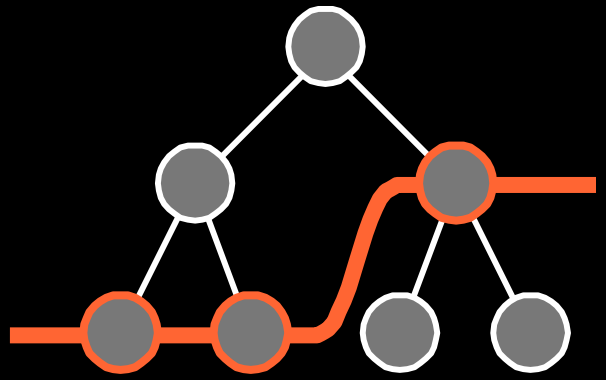
Key Concepts

- Light Cluster
- Light Tree
  - Binary tree of lights and clusters

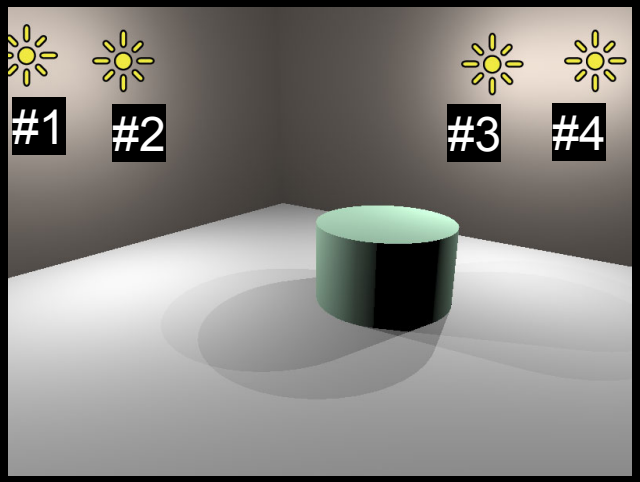


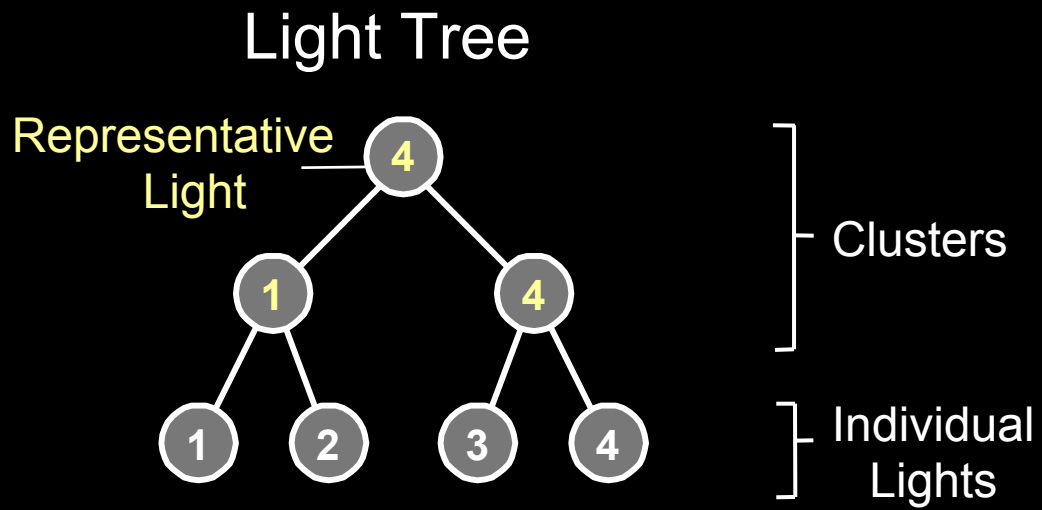
Key Concepts

- Light Cluster
- Light Tree
- A Cut
  - A set of nodes that partitions the lights into clusters



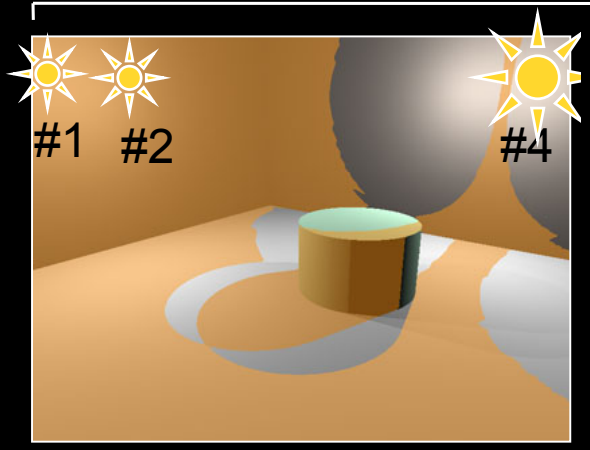
Simple Example

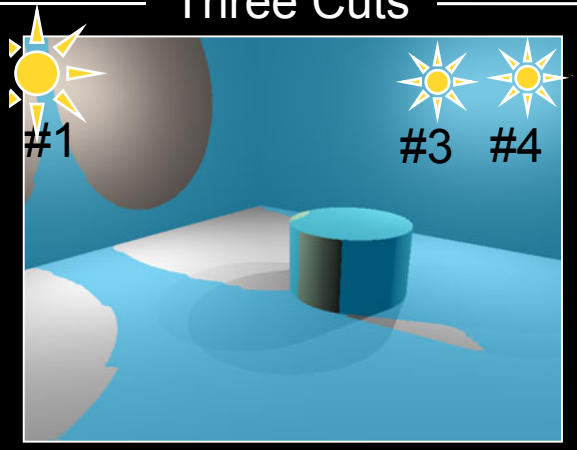


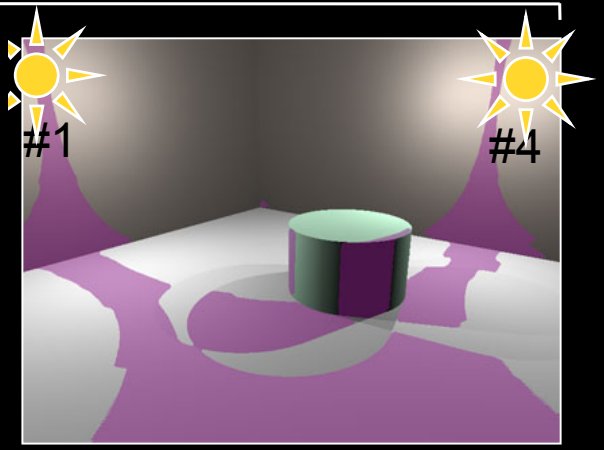


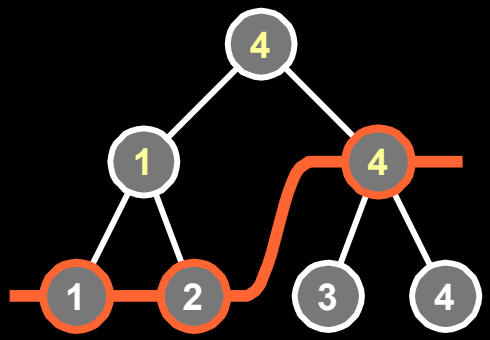
Three Example Cuts

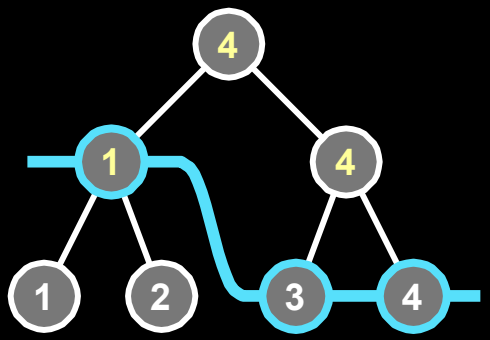
Three Cuts

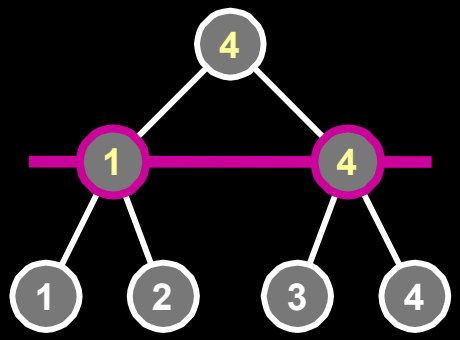






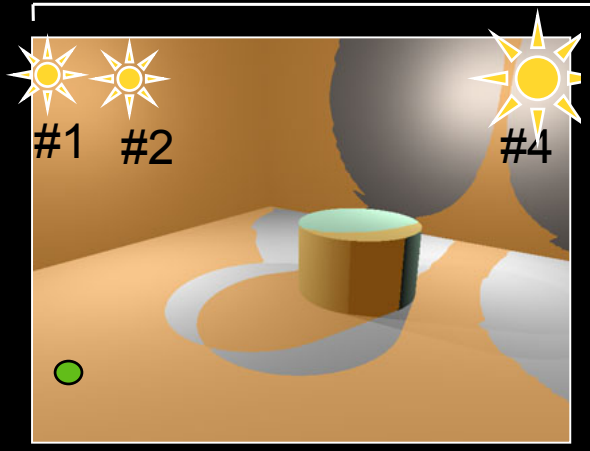


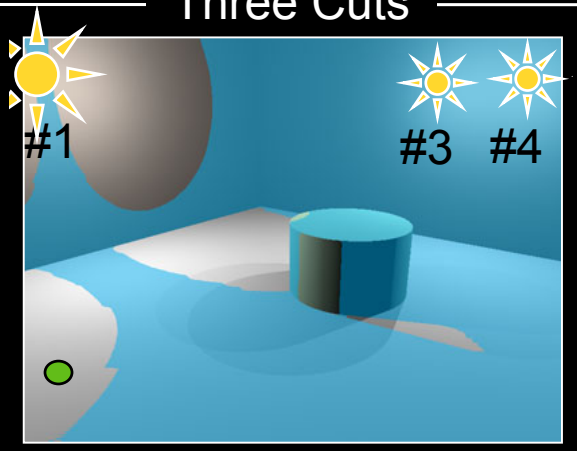


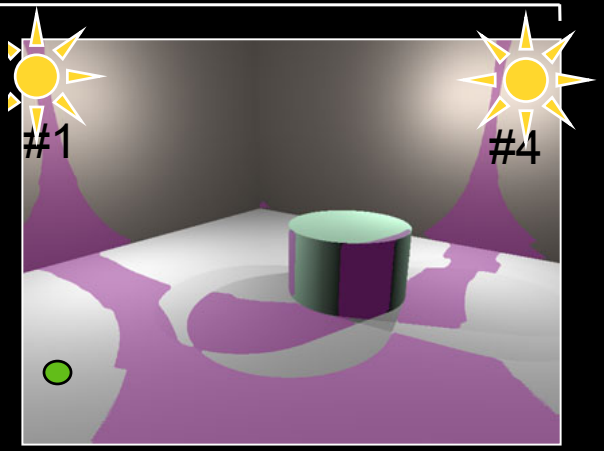


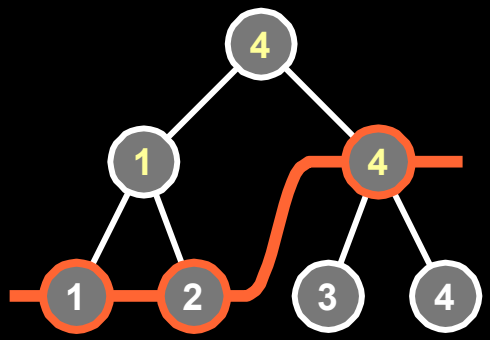
Three Example Cuts

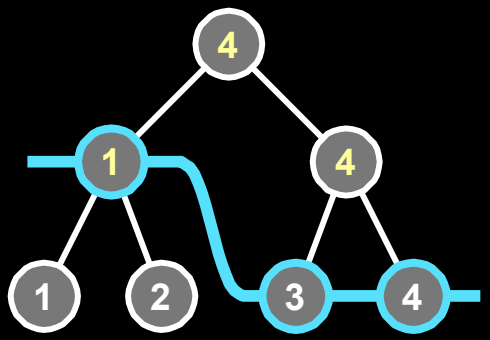
Three Cuts

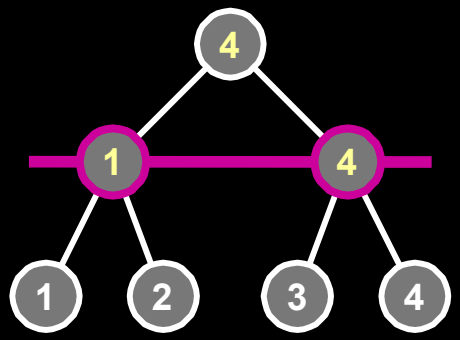








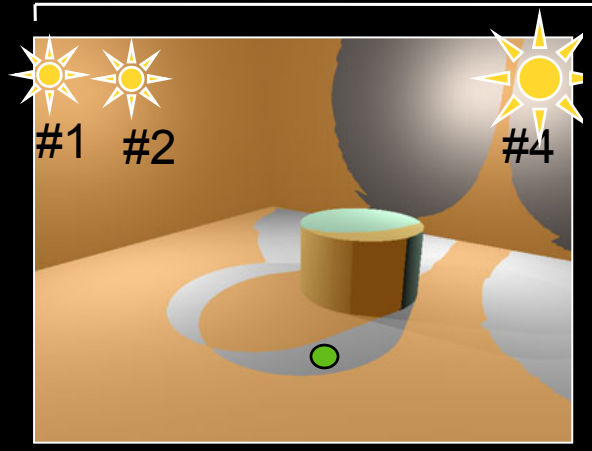


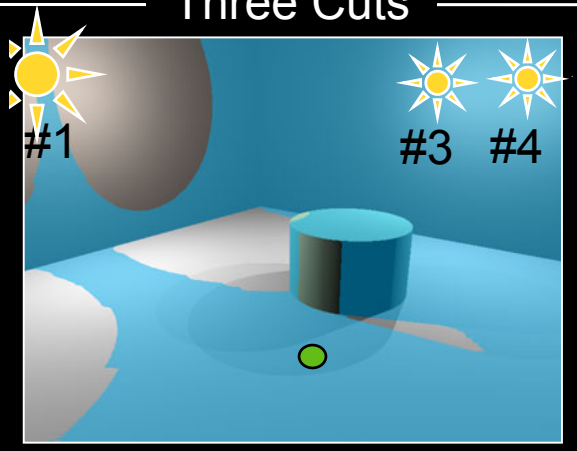


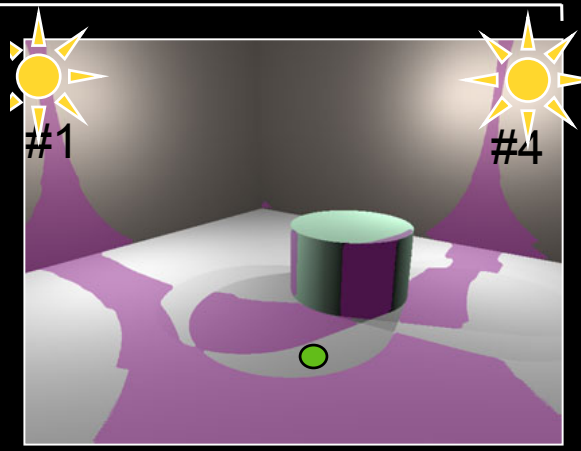
Good **Bad** **Bad**

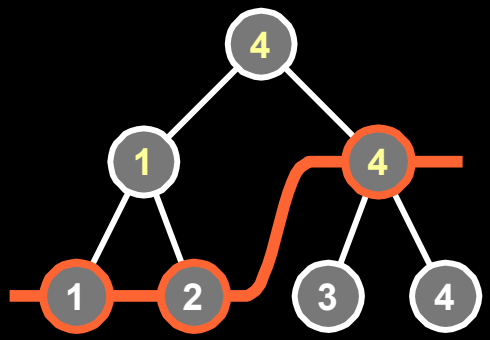
Three Example Cuts

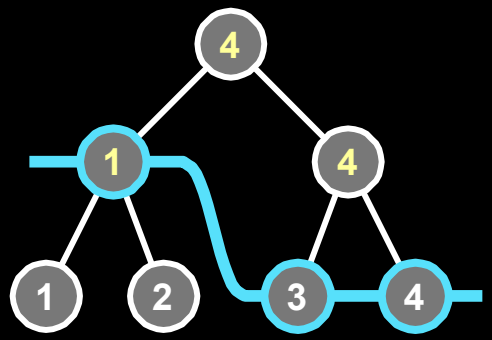
Three Cuts

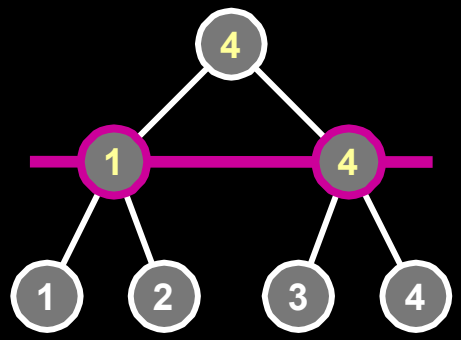








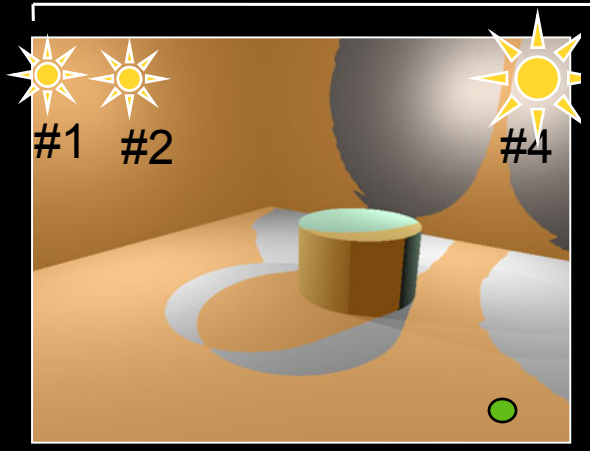


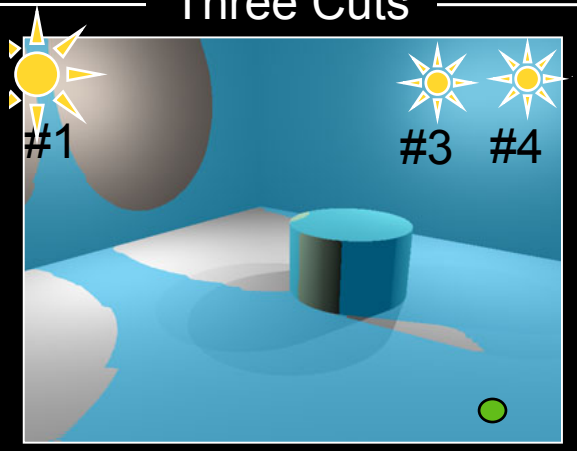


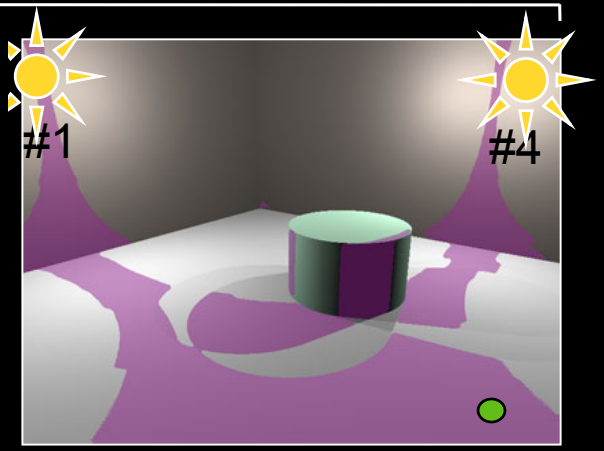
Bad **Good** **Bad**

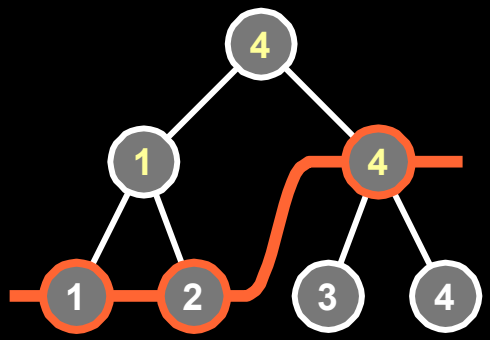
Three Example Cuts

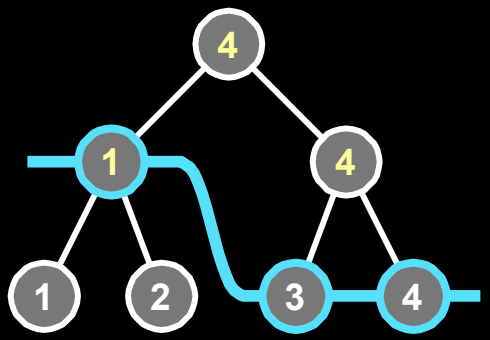
Three Cuts

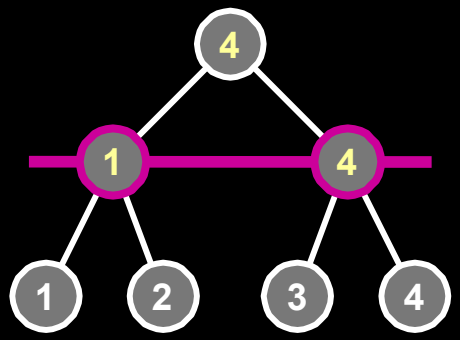












Good

Good

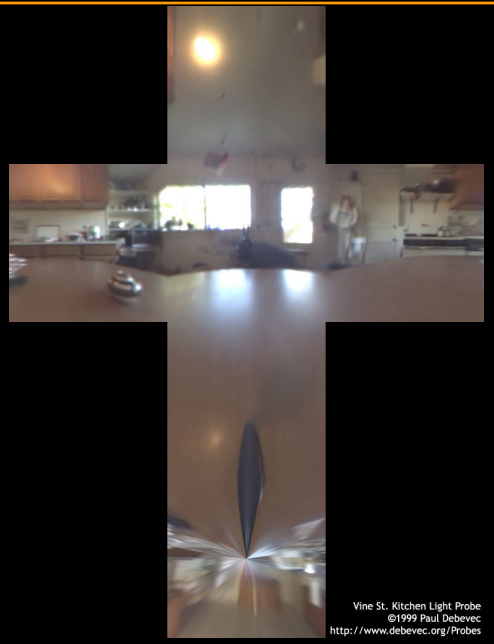
Good

Algorithm Overview

- Pre-process
  - Convert illumination to point lights
  - Build light tree
- For each eye ray
  - Choose a cut to approximate the illumination

Convert Illumination

- HDR environment map
  - Apply captured light to scene
  - Convert to directional point lights using [Agarwal et al. 2003]



- Indirect Illumination
  - Convert indirect to direct illumination using Instant Radiosity [Keller 97]
    - Caveats: no caustics, clamping, etc.
  - More lights = more indirect detail

Algorithm Overview

- Pre-process
  - Convert illumination to point lights
  - Build light tree
- For each eye ray
  - Choose a cut to approximate the local illumination
    - Cost vs. accuracy
    - Avoid visible transition artifacts

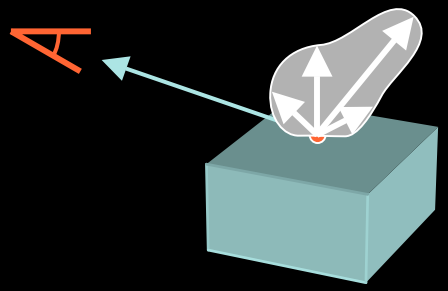
Perceptual Metric

- Weber's Law
  - Contrast visibility threshold is fixed percentage of signal
  - Used 2% in our results
- Ensure each cluster's error < visibility threshold
  - Transitions will not be visible
  - Used to select cut

Illumination Equation

$$\text{result} = \sum_{\text{lights}} M_i \; G_i \; V_i \; I_i$$

- M_i: Material term
- G_i: Geometric term
- V_i: Visibility term
- I_i: Light intensity



Currently support diffuse, phong, and Ward

Illumination Equation

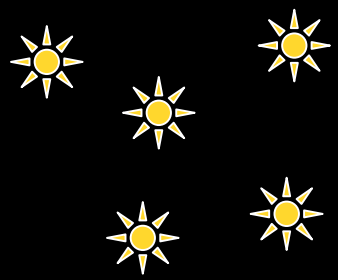
$$\text{result} = \sum_{\text{lights}} M_i \; G_i \; V_i \; I_i$$

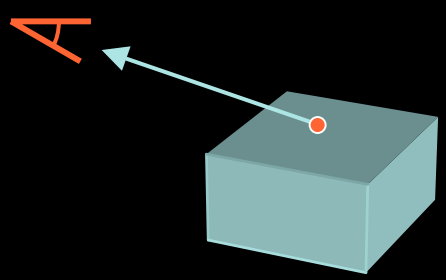
Material term

Geometric term

Visibility term

Light intensity





Illumination Equation

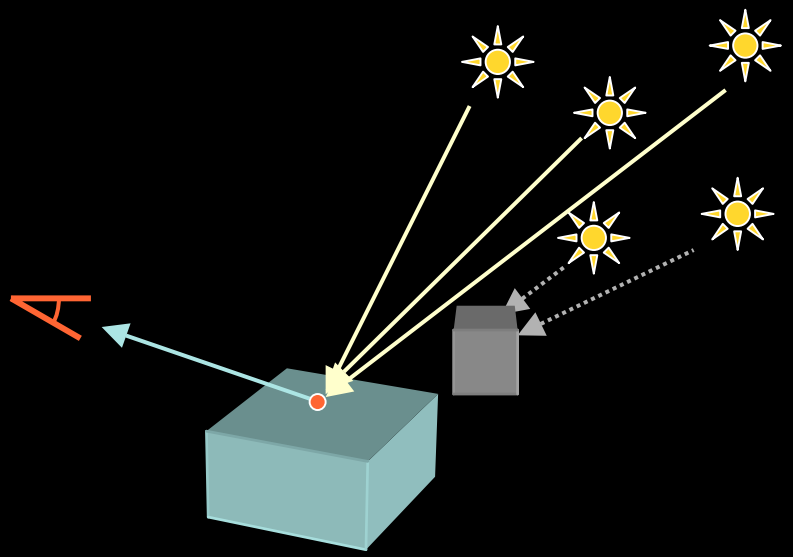
$$\text{result} = \sum_{\text{lights}} M_i \; G_i \; V_i \; I_i$$

Material term

Geometric term

Visibility term

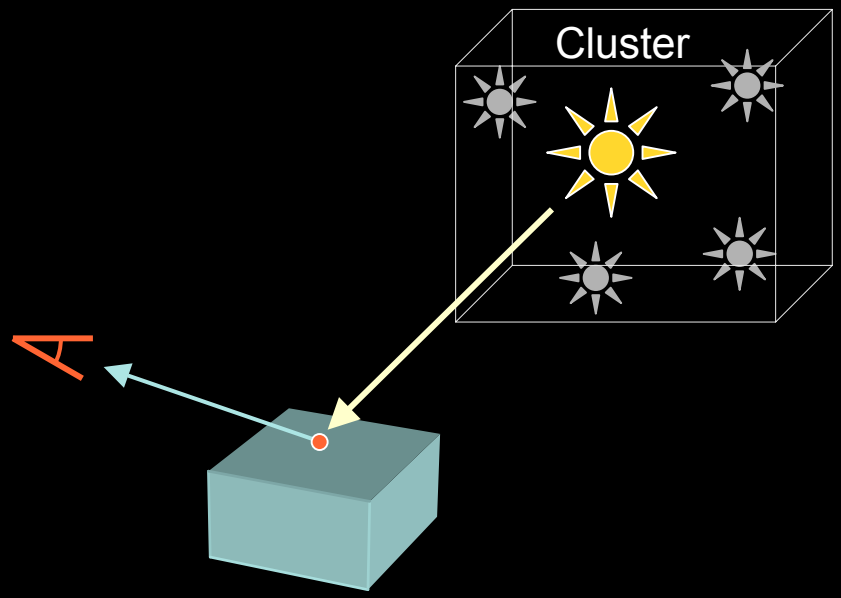
Light intensity



Cluster Approximation

$$\text{result} \approx M_j \; G_j \; V_j \sum_{\text{lights}} I_i$$

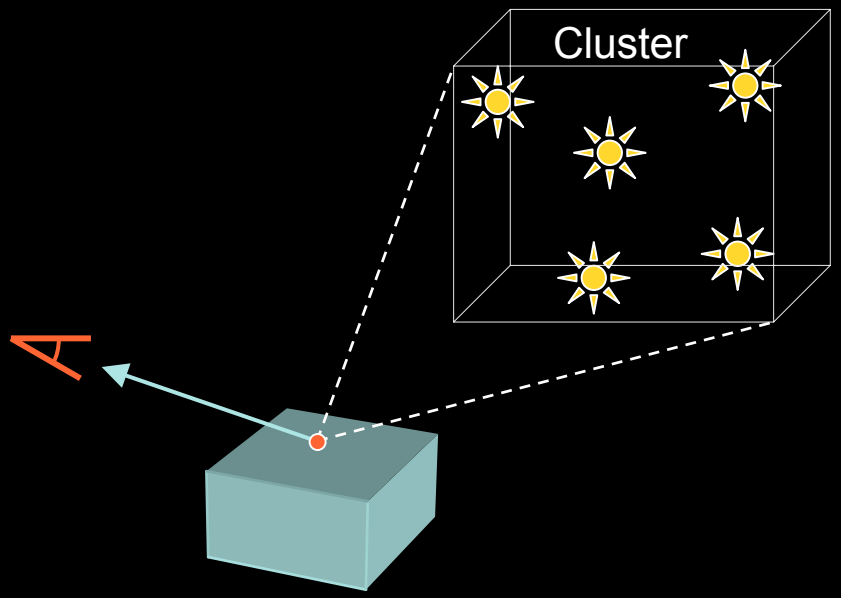
j is the representative light



Cluster Error Bound

$$\text{error} \leq M_{ub}\, G_{ub}\, V_{ub} \sum_{\text{lights}} I_i$$

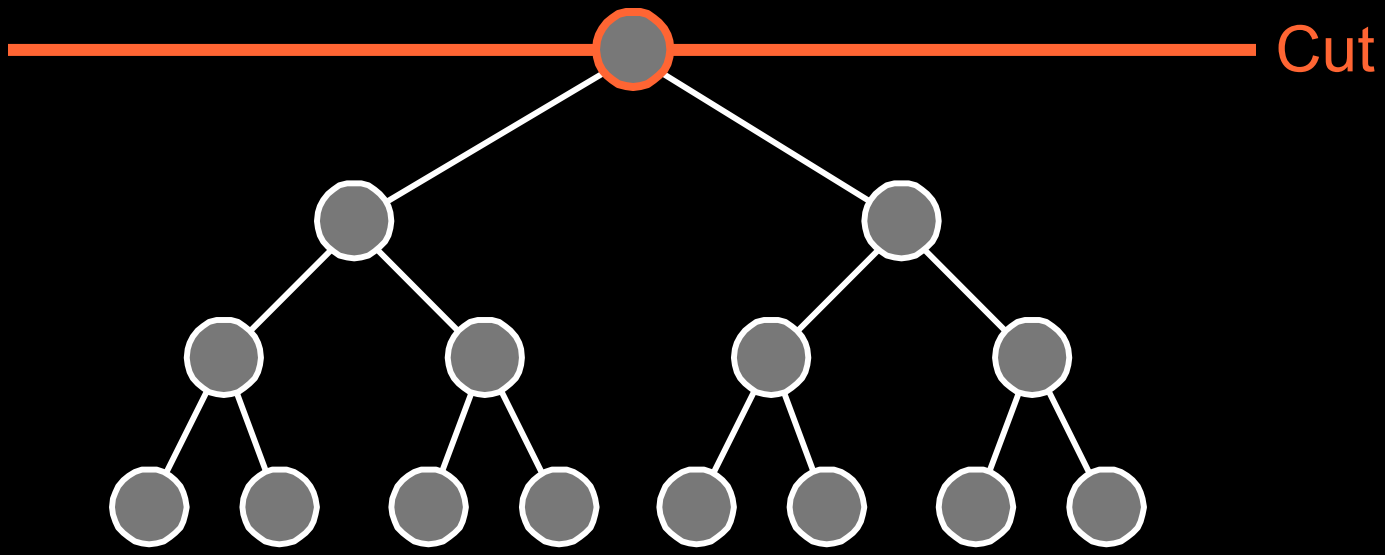
- Bound each term
  - Visibility <= 1 (trivial)
  - Intensity is known
  - Bound material and geometric terms using cluster bounding volume



ub == upper bound

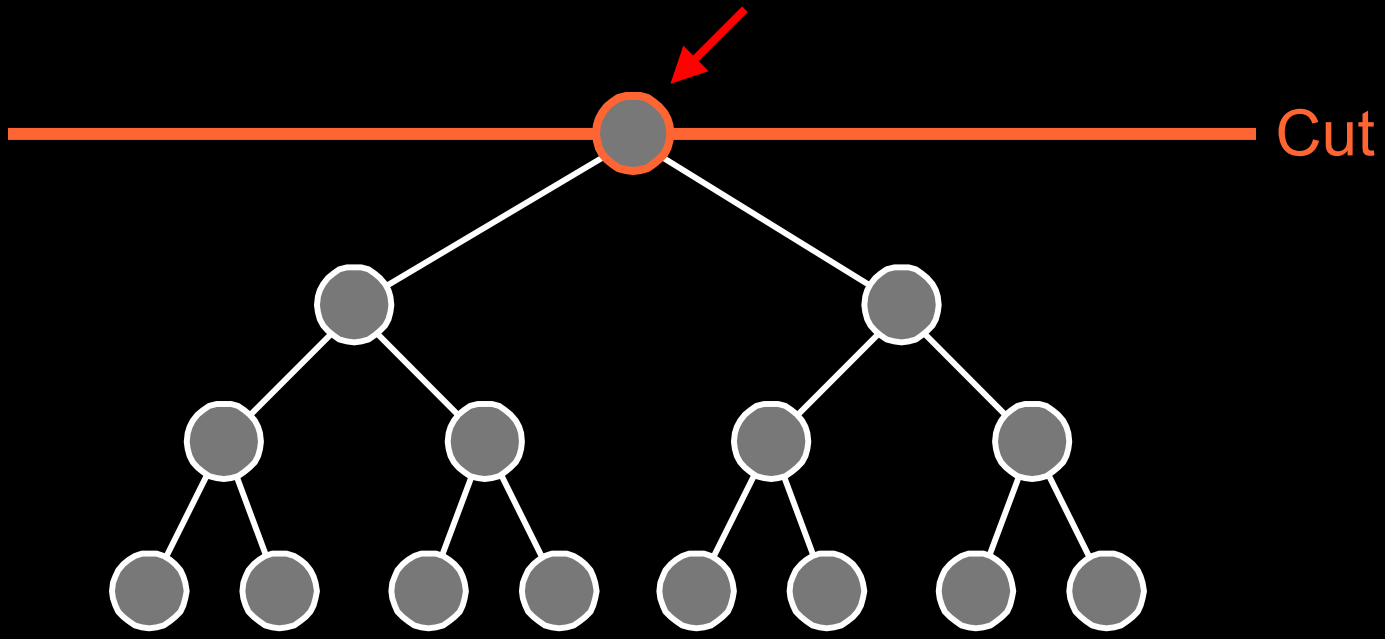
Cut Selection Algorithm

- Start with coarse cut (eg, root node)



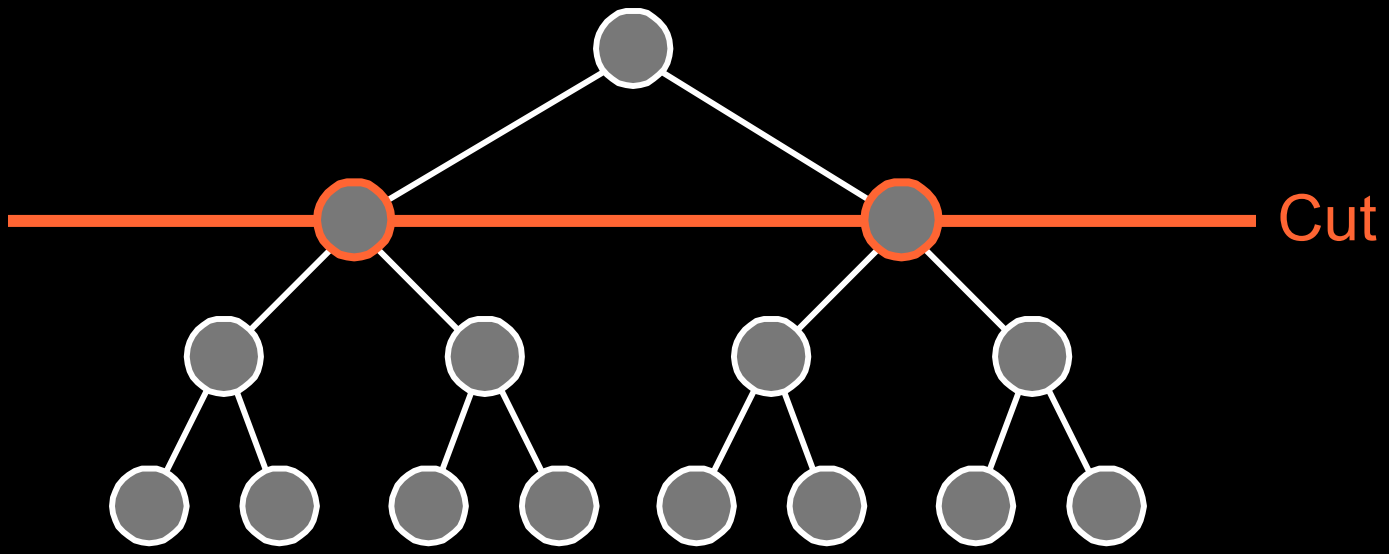
Cut Selection Algorithm

- Select cluster with largest error bound

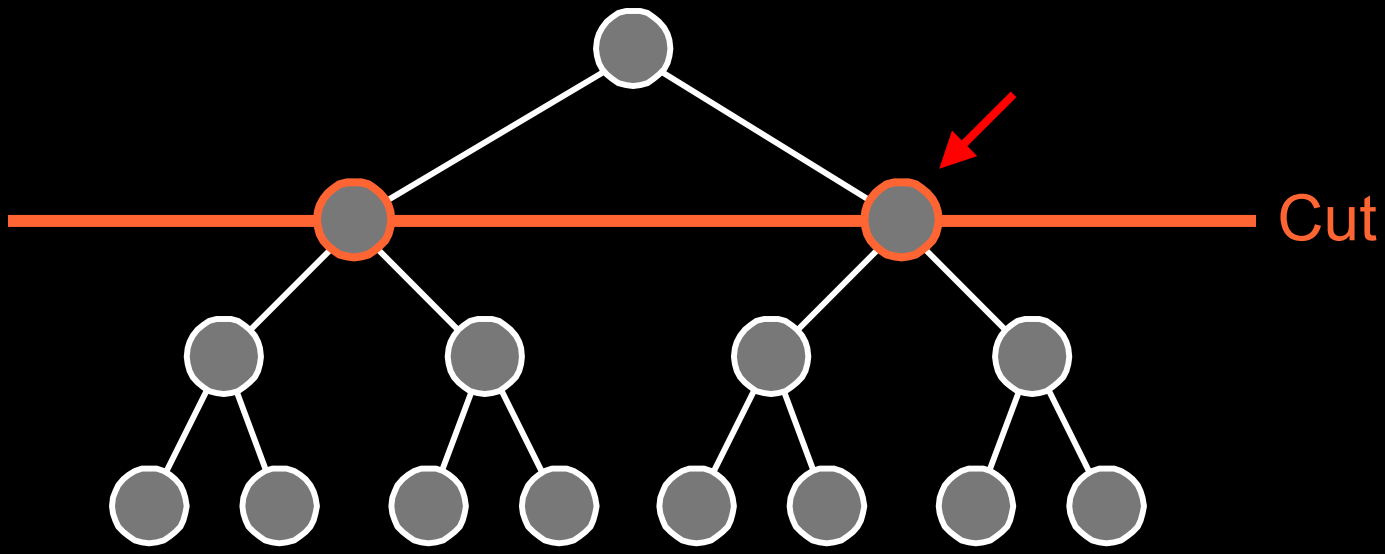


Cut Selection Algorithm

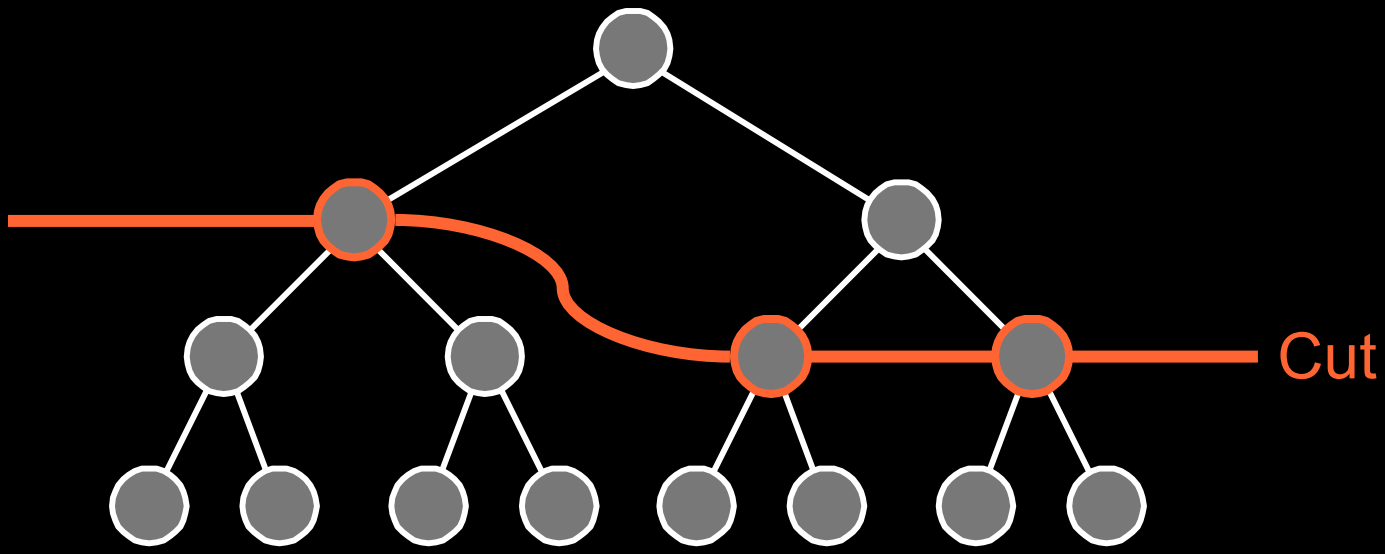
- Refine if error bound > 2% of total



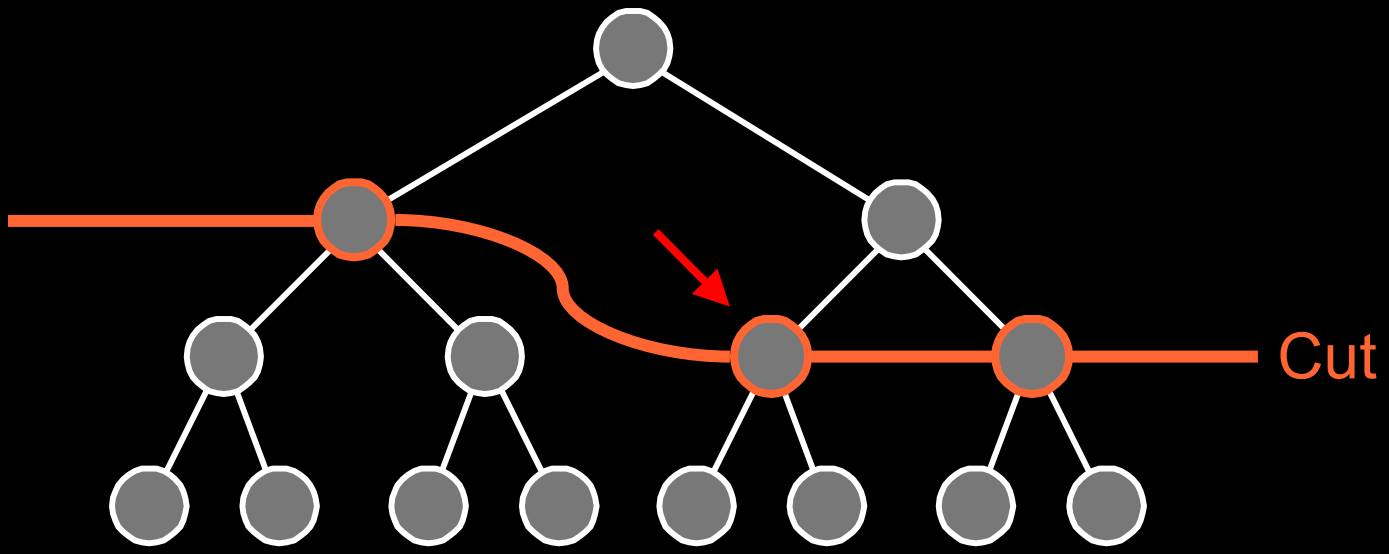
Cut Selection Algorithm



Cut Selection Algorithm

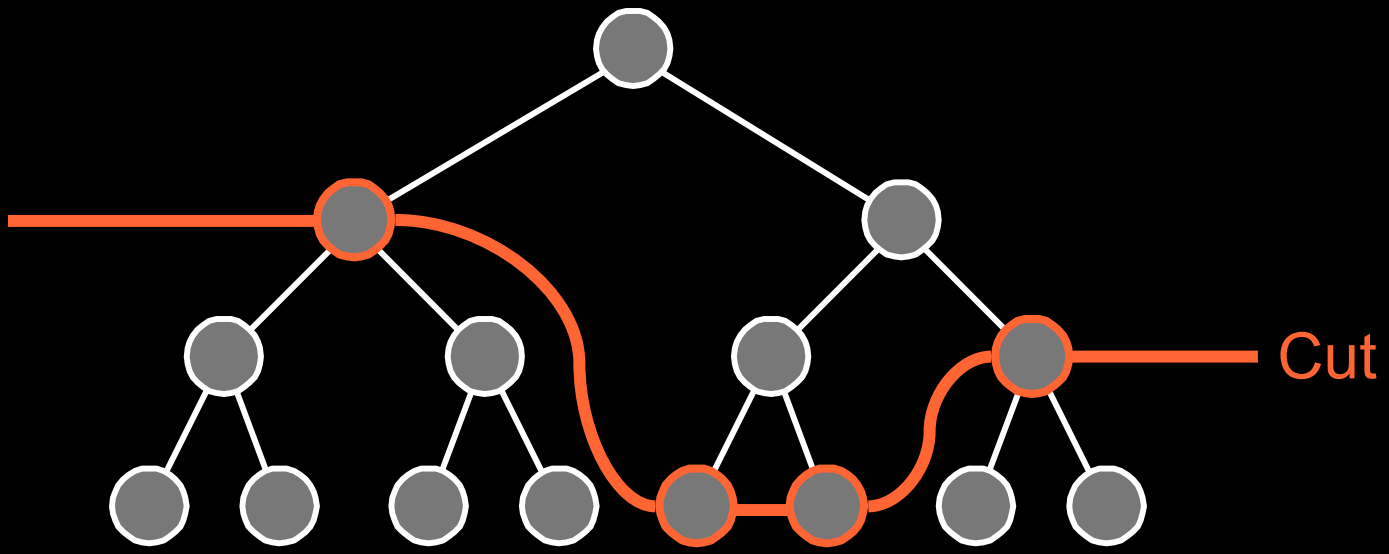


Cut Selection Algorithm



Cut Selection Algorithm

- Repeat until cut obeys 2% threshold



Kitchen, 388K polygons, 4608 lights (72 area sources)

Kitchen, 388K polygons, 4608 lights (72 area sources)

Combined Illumination

Lightcuts 128s

4 608 Lights
(Area lights only)

Lightcuts 290s

59 672 Lights
(Area + Sun/sky + Indirect)

Combined Illumination

Lightcuts 128s

4 608 Lights
(Area lights only)

Avg. 259 shadow rays / pixel

Lightcuts 290s

59 672 Lights
(Area + Sun/sky + Indirect)

Avg. 478 shadow rays / pixel
(only 54 to area lights)

Lightcuts Recap

- Unified illumination handling
- Scalable solution for many lights
  - Locally adaptive representation (the cut)
- Analytic cluster error bounds
  - Most important lights always sampled
- Perceptual visibility metric

Lightcuts implementation sketch, Petree Hall C, ~4:30pm

Talk Overview

- Lightcuts
  - Scalable accurate solution for complex illumination
- Reconstruction cuts
  - Builds on lightcuts
  - Use smart interpolation to further reduce cost

Reconstruction Cuts

- Subdivide image into blocks
  - Generate samples at corners
- Within blocks
  - Interpolate smooth illumination
  - Use shadow rays when needed to preserve features
    - Shadow boundaries, glossy highlights, etc.
- Anti-aliasing
  - (5-50 samples per pixel)

Image Subdivision

- Divide into max block size (4x4 blocks)

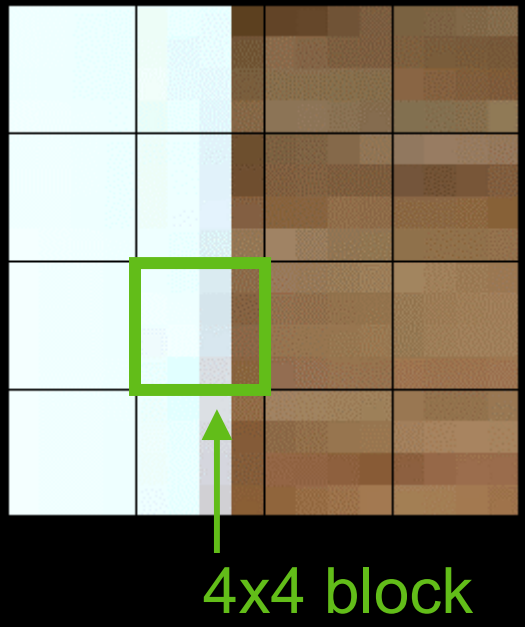


Image Subdivision

- Divide into max block size (4x4 blocks)
- Trace multiple eye rays per pixel
- Subdivide blocks if needed
  - Based on material, surface normal, and local shadowing configuration

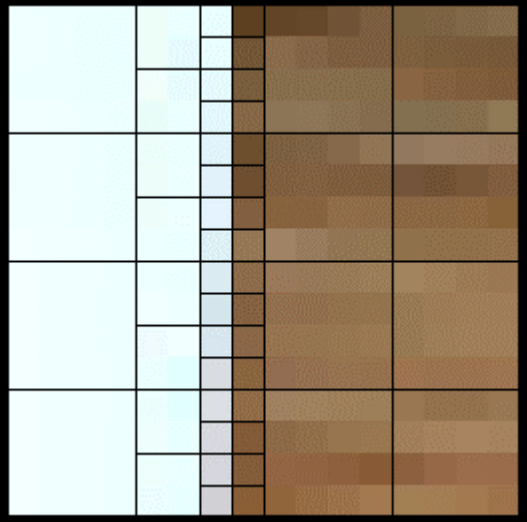
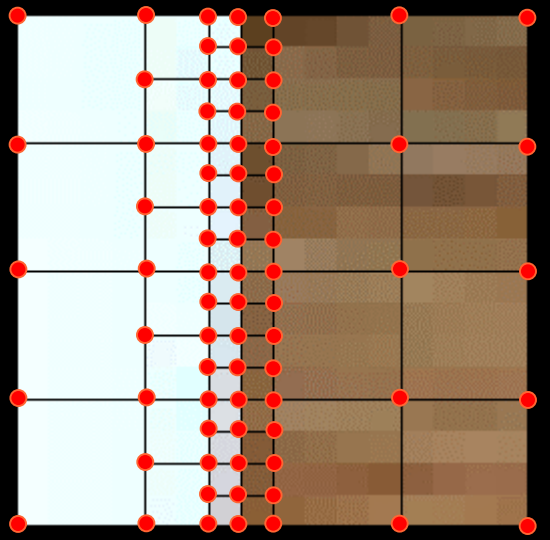


Image Subdivision

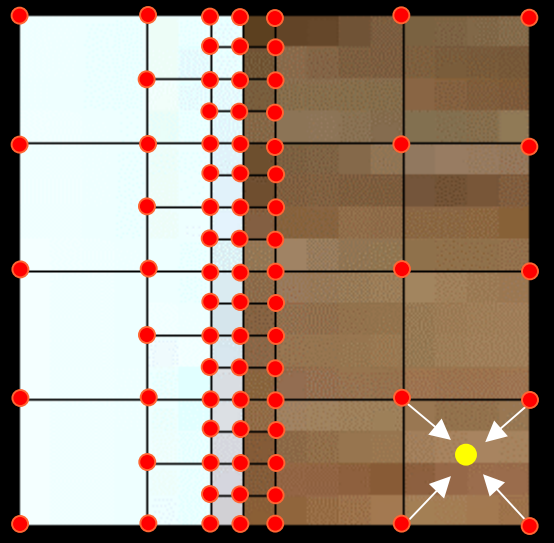
- Divide into max block size (4x4 blocks)
- Trace multiple eye rays per pixel
- Subdivide blocks if needed
  - Based on material, surface normal, and local shadowing configuration
- Compute samples at corners



Samples

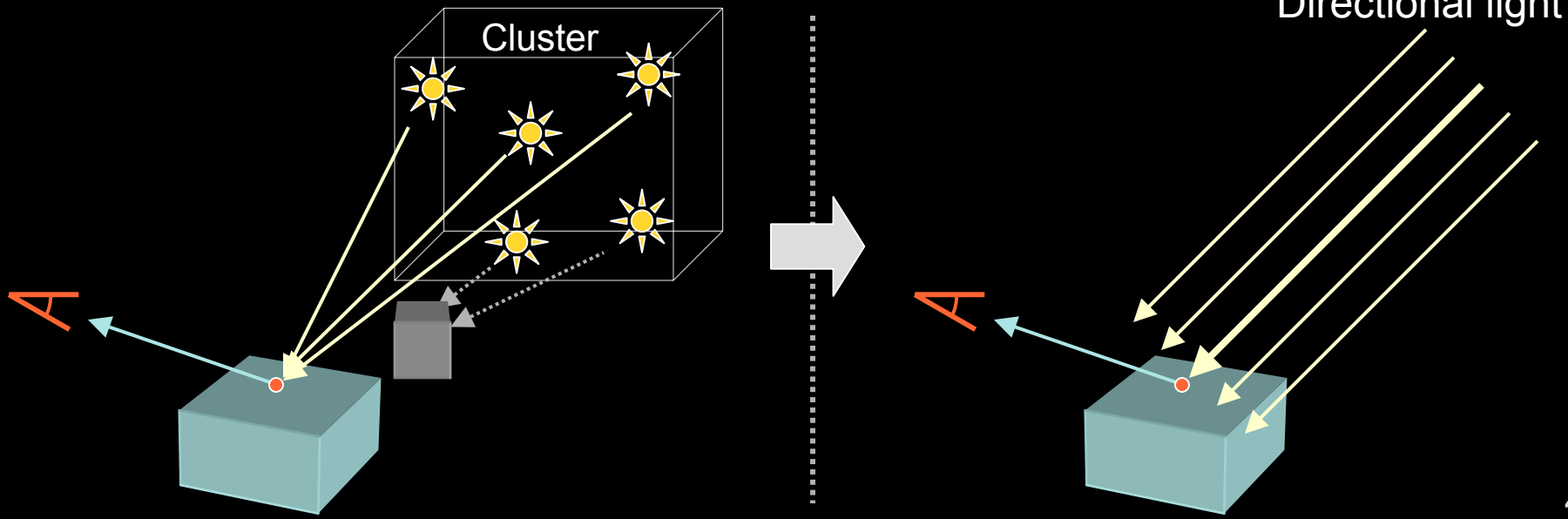
Image Subdivision

- Divide into max block size (4x4 blocks)
- Trace multiple eye rays per pixel
- Subdivide blocks if needed
  - Based on material, surface normal, and local shadowing configuration
- Compute samples at corners
- Shade eye rays using reconstruction cuts



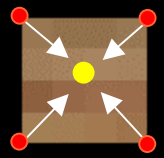
Sample Construction

- Compute a lightcut at each sample
- For each node on or above the cut
  - Create impostor light (directional light)
  - Reproduce cluster's effect at sample



Reconstruction Cut

- Top-down traversal of light tree
  - Comparing impostors from nearby samples



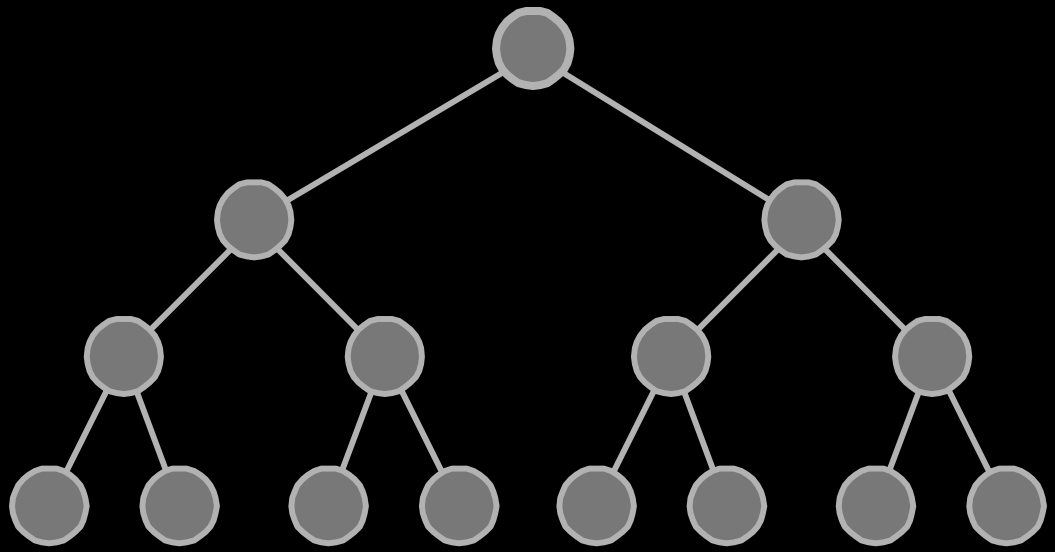
Not visited

Recurse

Occluded

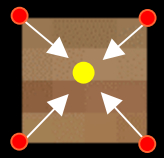
Interpolate

Shadow ray



Reconstruction Cut

- Recurse if samples differ significantly



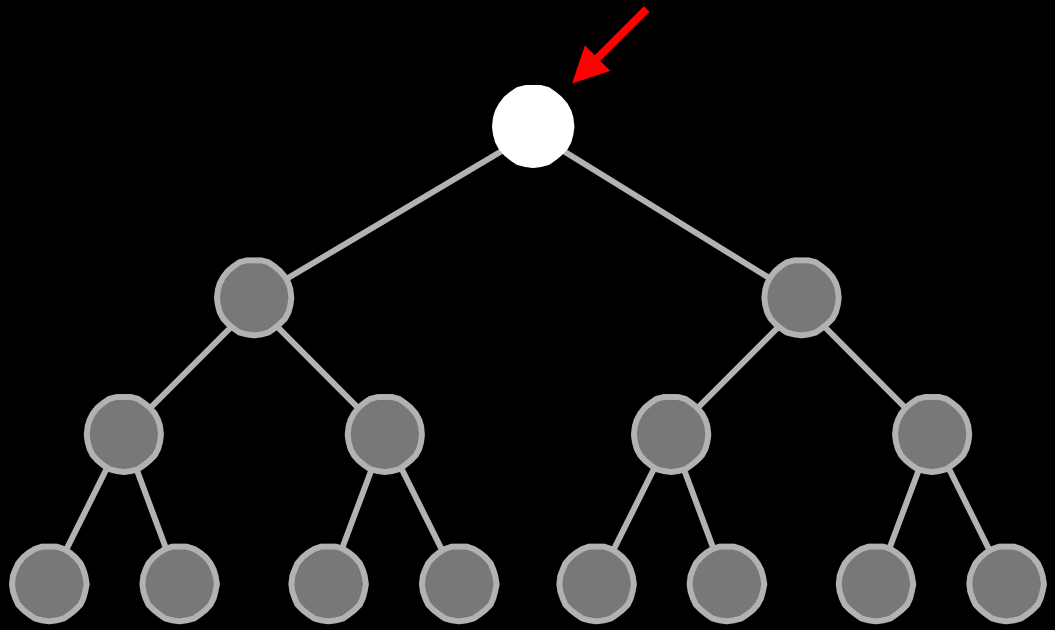
Not visited

Recurse

Occluded

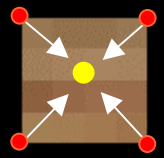
Interpolate

Shadow ray



Reconstruction Cut

- Discard if cluster occluded at all samples



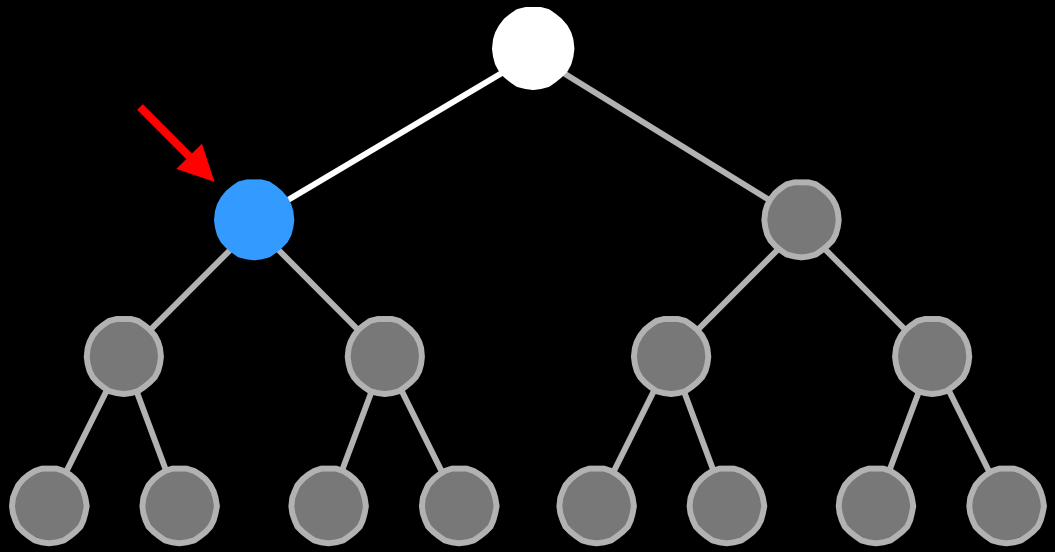
Not visited

Recurse

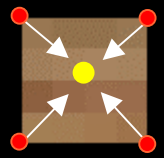
Occluded

Interpolate

Shadow ray



Reconstruction Cut



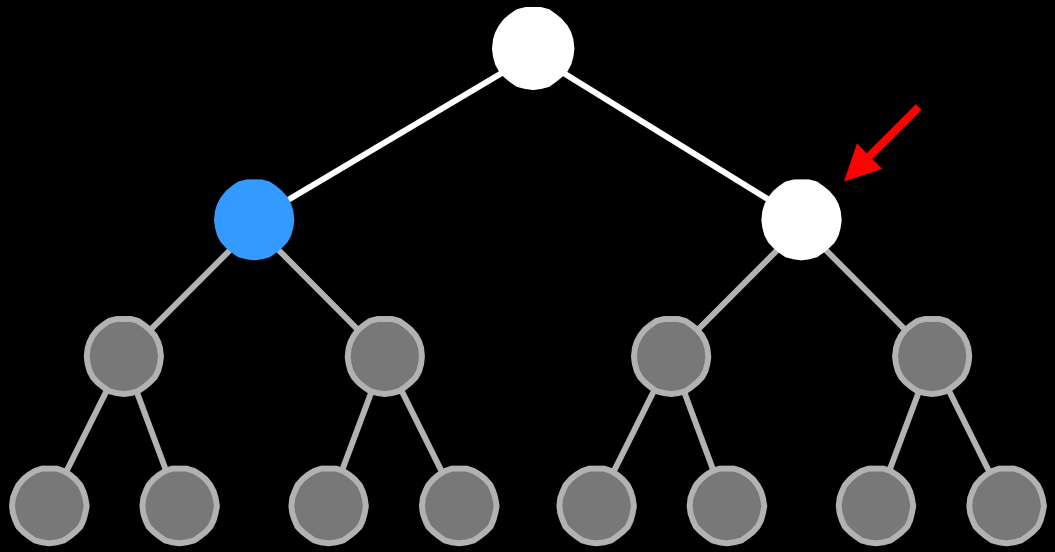
Not visited

Recurse

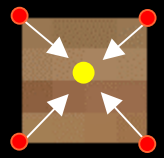
Occluded

Interpolate

Shadow ray



Reconstruction Cut



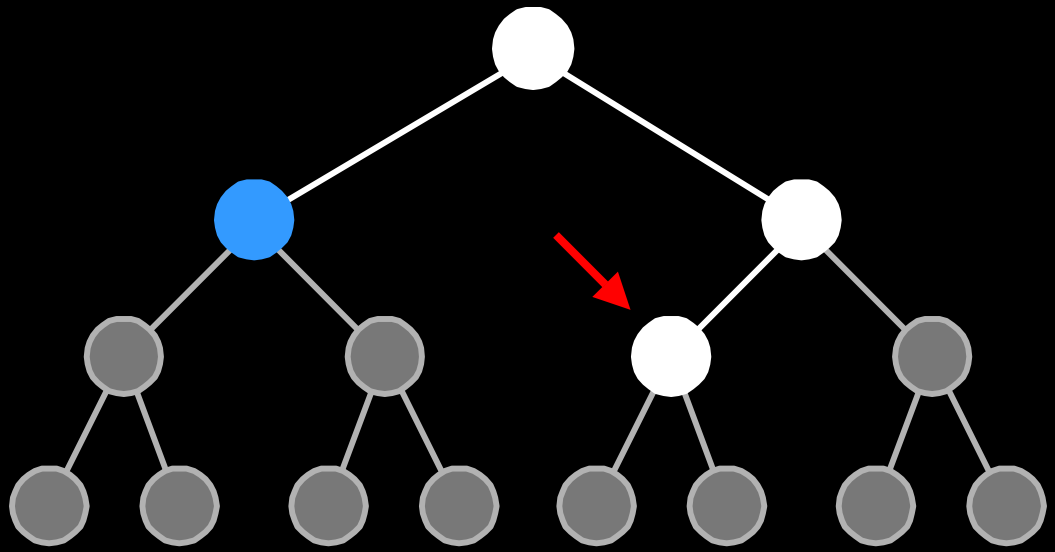
Not visited

Recurse

Occluded

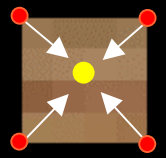
Interpolate

Shadow ray



Reconstruction Cut

- Interpolate if sample impostors are similar



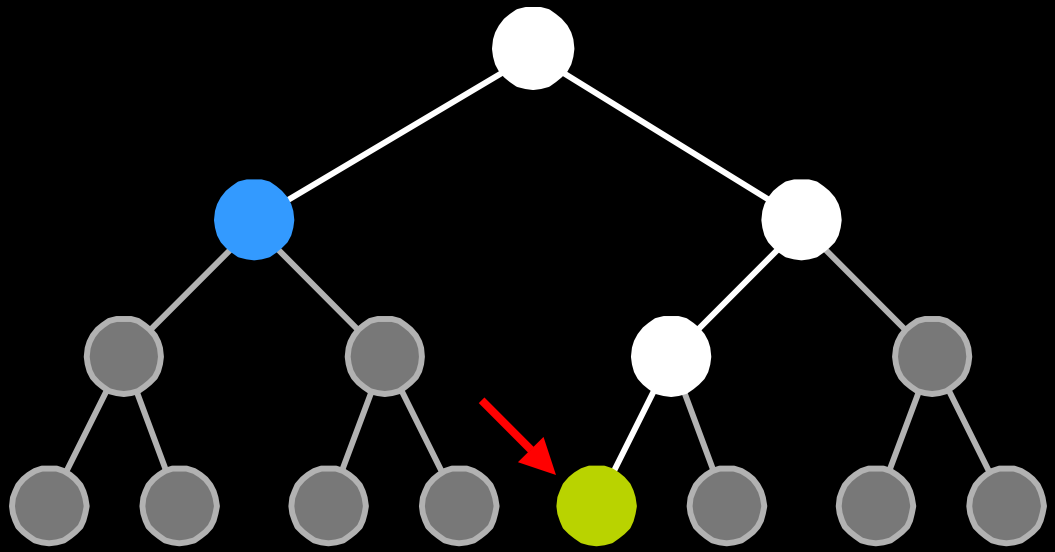
Not visited

Recurse

Occluded

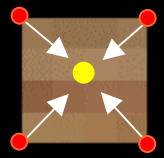
Interpolate

Shadow ray



Reconstruction Cut

- If cluster contribution small enough, shoot shadow ray to representative light
  - Lightcut-style evaluation



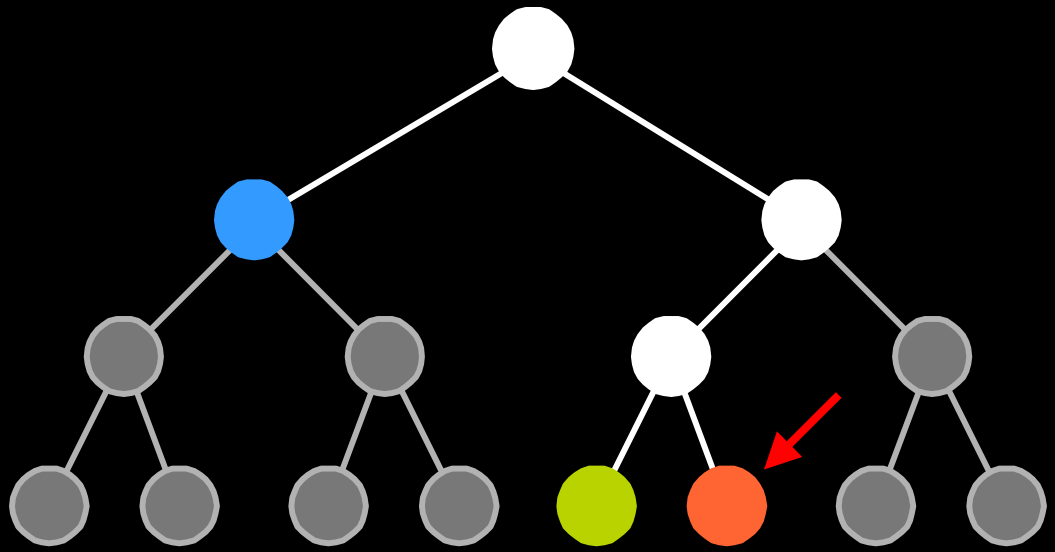
Not visited

Recurse

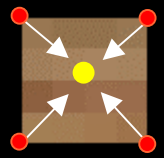
Occluded

Interpolate

Shadow ray



Reconstruction Cut



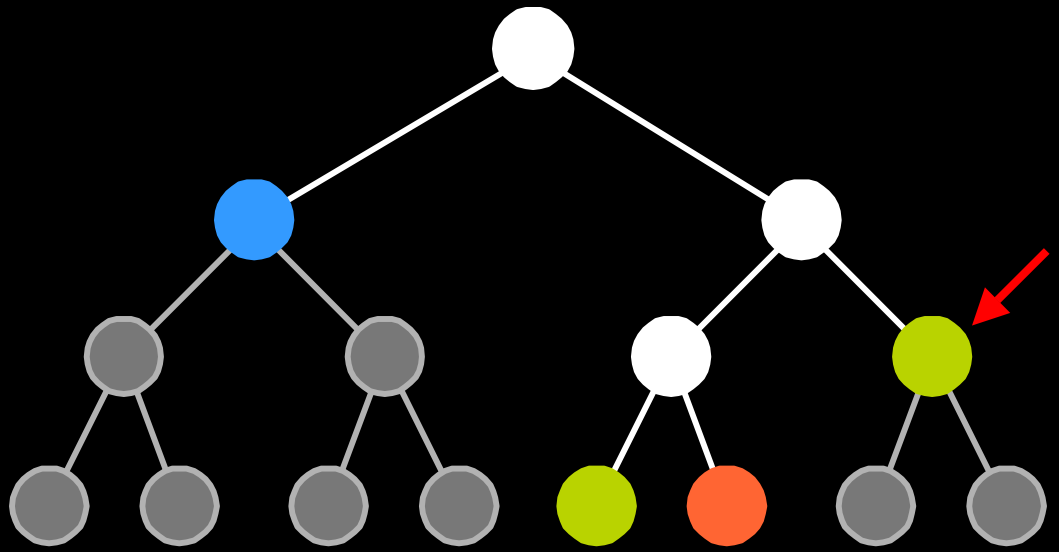
Not visited

Recurse

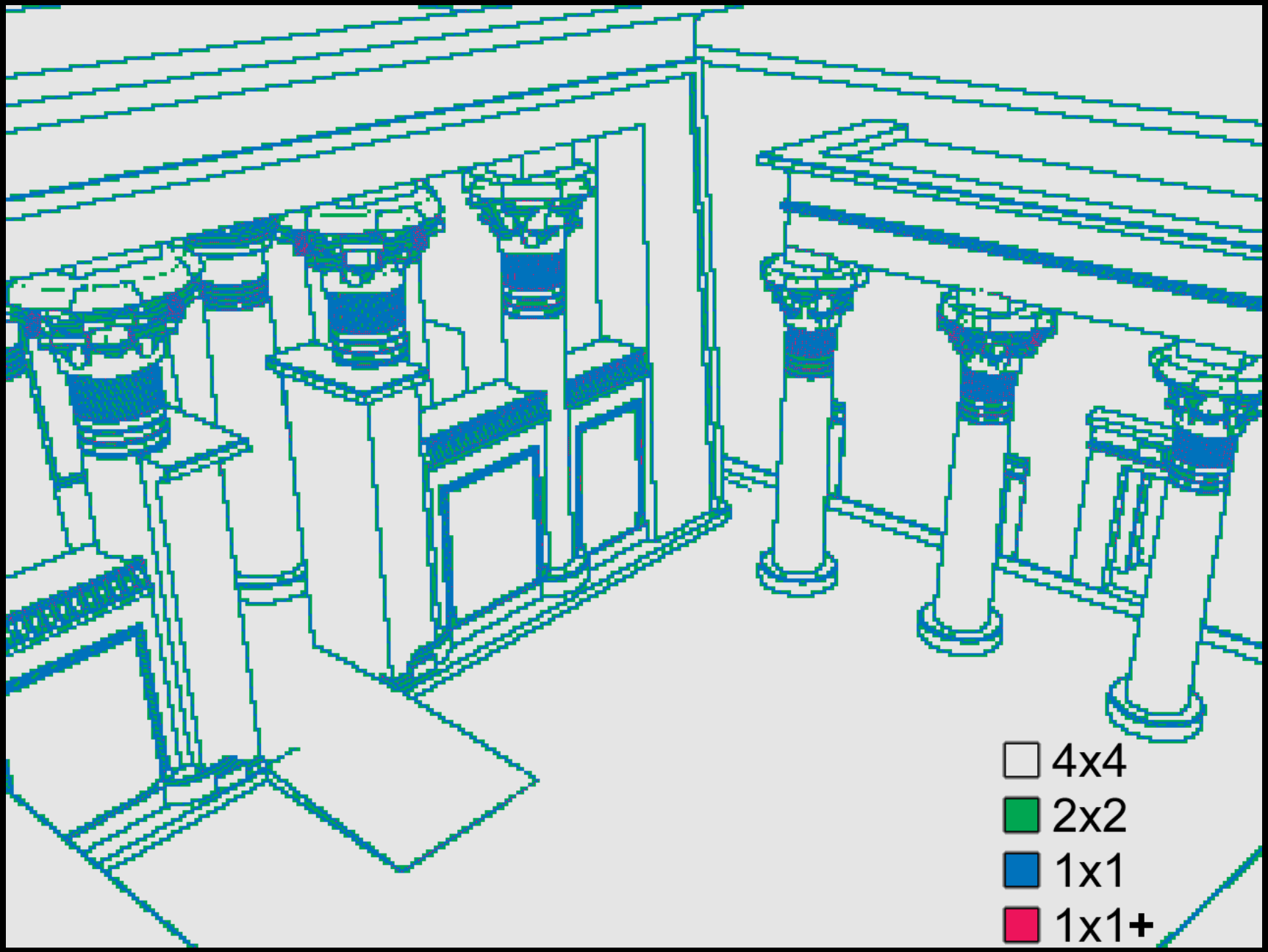
Occluded

Interpolate

Shadow ray



Temple, 2.1M polygons, 505064 lights, (Sun/sky+Indirect)



Temple, reconstruction cut block size

Result Statistics

- Temple model (2.1M polys, 505064 lights)

Cut type	Avg. shadow rays per cut
Lightcut	373
Reconstruction cut	9.4

Image algorithm	Avg. eye rays per pixel	Image time
Lightcuts only	1	225s
Combined (anti-aliased)	5.5	189s

Grand Central, 1.46M polygons, 143464 lights, (Area+Sun/sky+Indirect)

Avg. shadow rays per eye ray 46 (**0.03%**)

Tableau, 630K polygons, 13 000 lights, (EnvMap+Indirect)

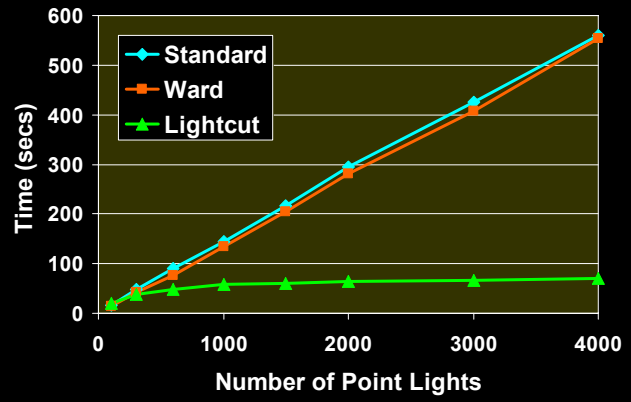
Avg. shadow rays per eye ray 17 (**0.13%**)

Bigscreen, 628K polygons, 639528 lights, (Area+Indirect)

Avg. shadow rays per eye ray 17 (**0.003%**)

Conclusions

- Lightcuts
  - Scalable, unified framework for complex illumination
  - Analytic cluster error bounds & perceptual visibility metric



- Reconstruction cuts
  - Exploits coherence
  - High-resolution, anti-aliased images

Future Work

- Visibility bounds
- More light types
  - Spot lights etc.
- More BRDF types
  - Need cheap tight bounds
- Other illumination types
  - Eg, caustics

Acknowledgements

- National Science Foundation grant ACI-0205438
- Intel corporation for support and equipment
- The modelers
  - Kitchen: Jeremiah Fairbanks
  - Bigscreen: Will Stokes
  - Grand Central: Moreno Piccolotto, Yasemin Kologlu, Anne Briggs, Dana Gettman
  - Temple: Veronica Sundstedt, Patrick Ledda, and the graphics group at University of Bristol
  - Stanford and Georgia Tech for Buddha and Horse geometry

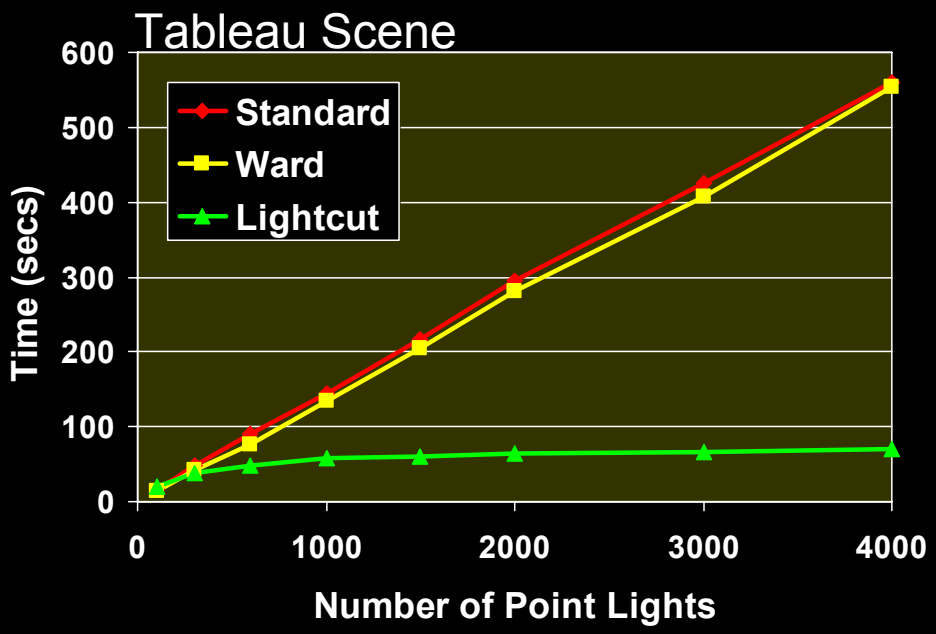
The End

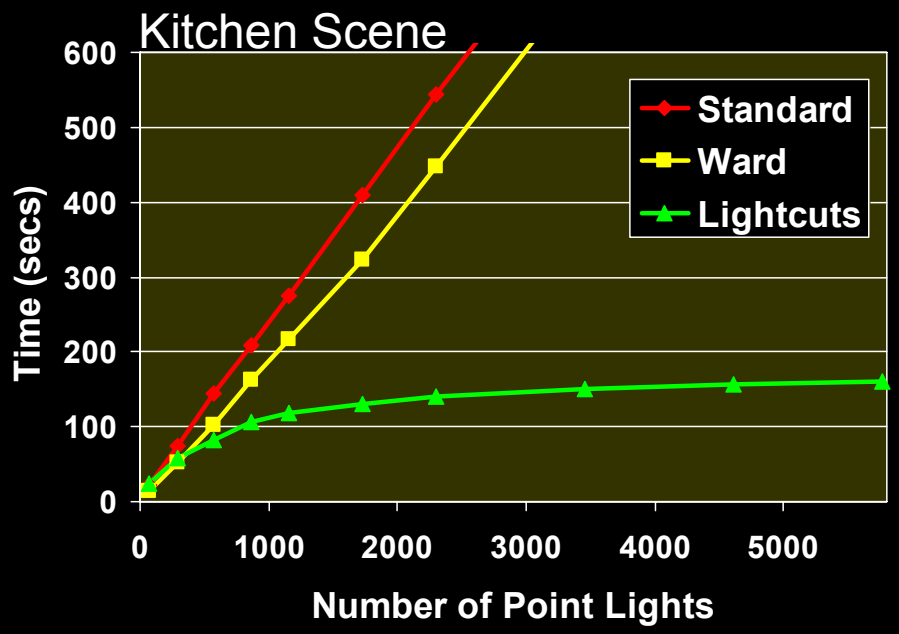
- Questions?

Lightcuts implementation sketch, Petree Hall C, ~4:30pm

Scalable

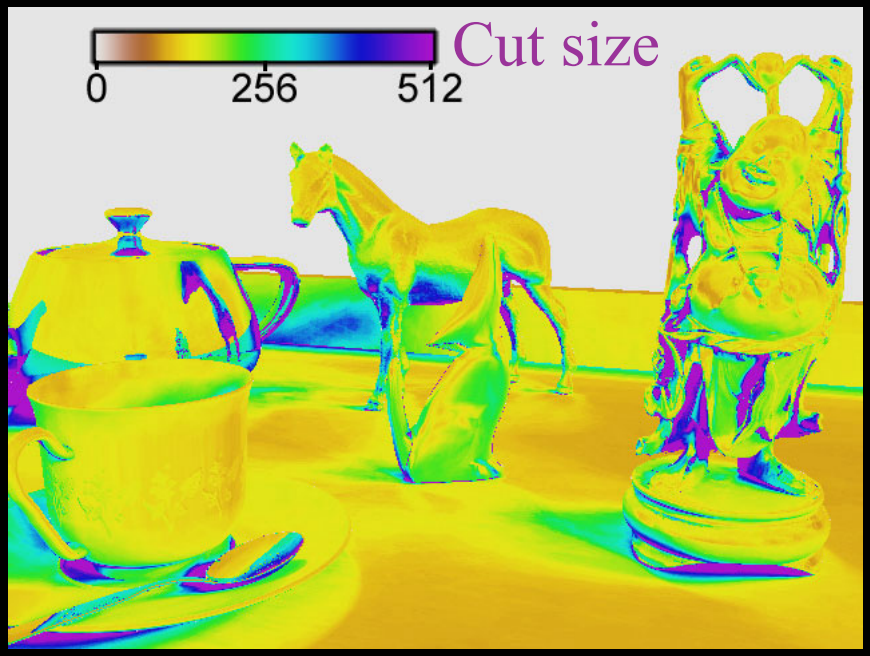
- Scalable solution for many point lights
  - Thousands to millions
  - Sub-linear cost

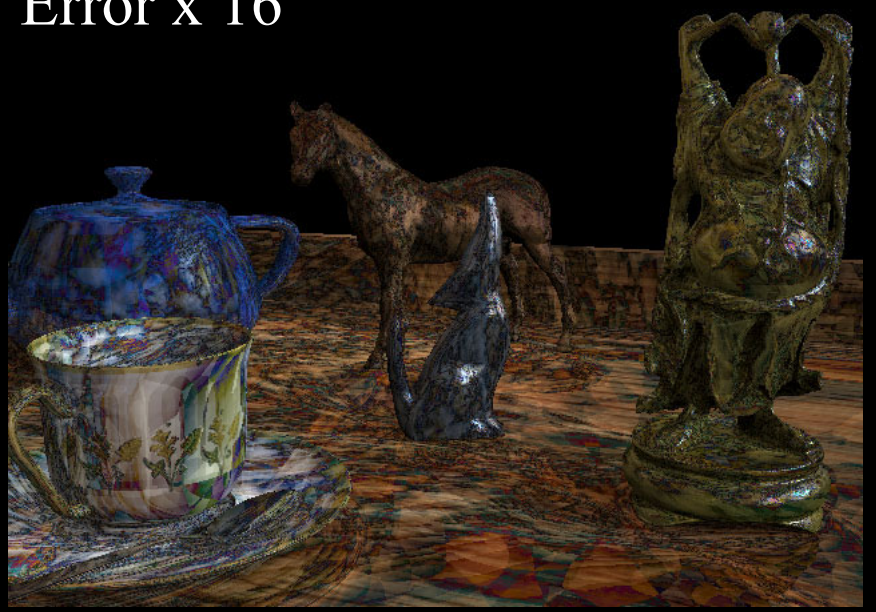




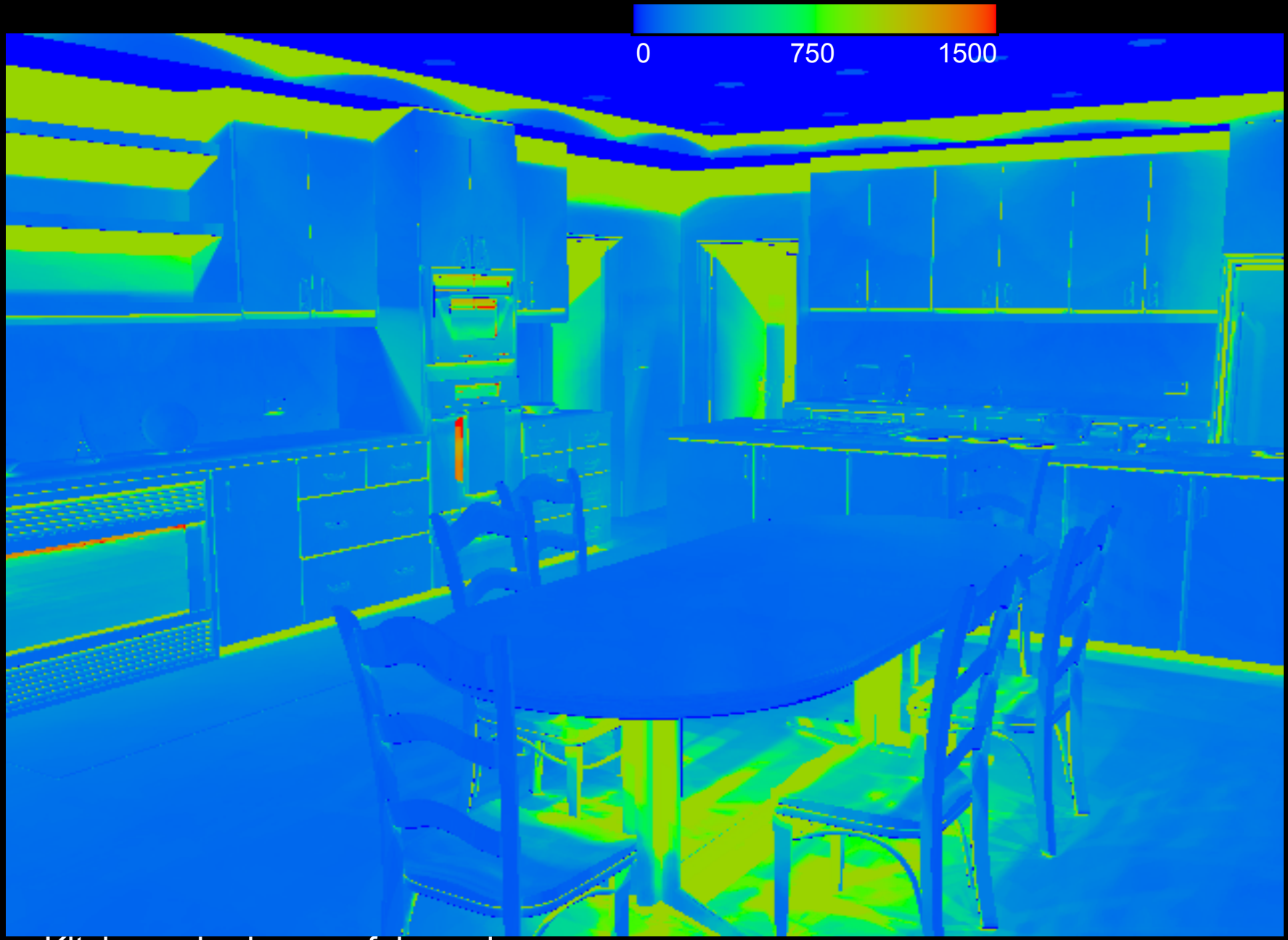

Lightcuts


Reference


Cut size
0
256
512


Error x 16

Kitchen, 388K polygons, 59,672 Lights



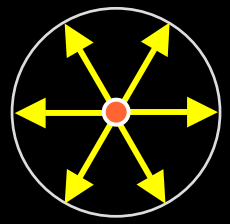
Kitchen, shadow ray false color

Tableau, shadow ray false color

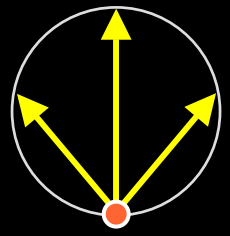
Kitchen with sample locations marked

Types of Point Lights

- Omni
  - Spherical lights



- Oriented
  - Area lights, indirect lights



- Directional
  - HDR env maps, sun&sky