HDR and Image-Based Lighting

© Alyosha Efros

CS180: Comp. Vision & Computational Photography
Alexei Efros, UC Berkeley, Fall 2023

…with a lot of slides stolen from Paul Debevec

Why HDR?

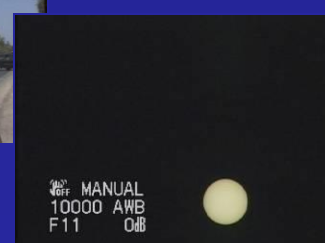
Problem: Dynamic Range

1

1500

25,000

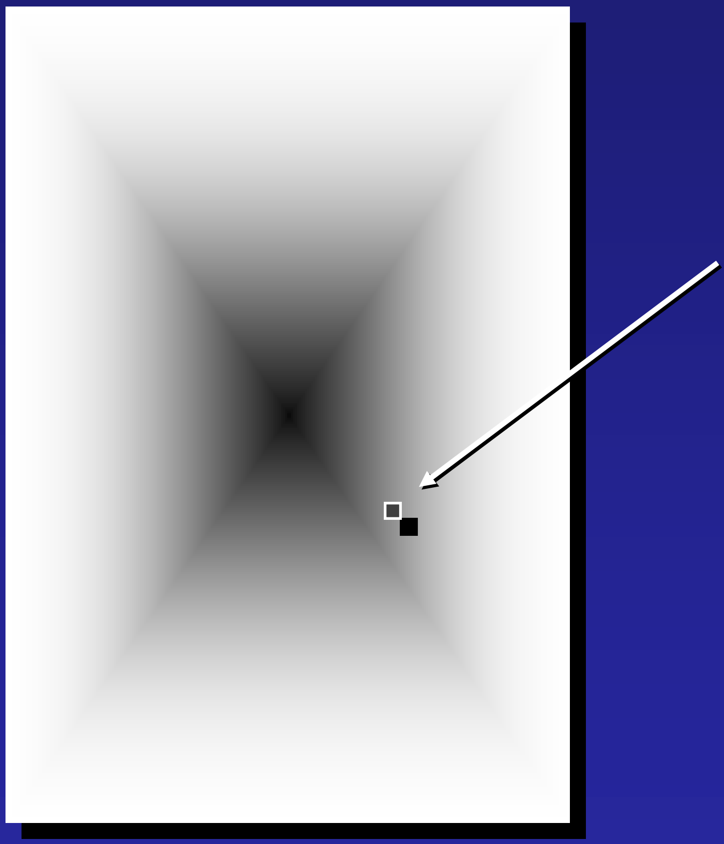
400,000



2,000,000,000

The real world is high dynamic range.

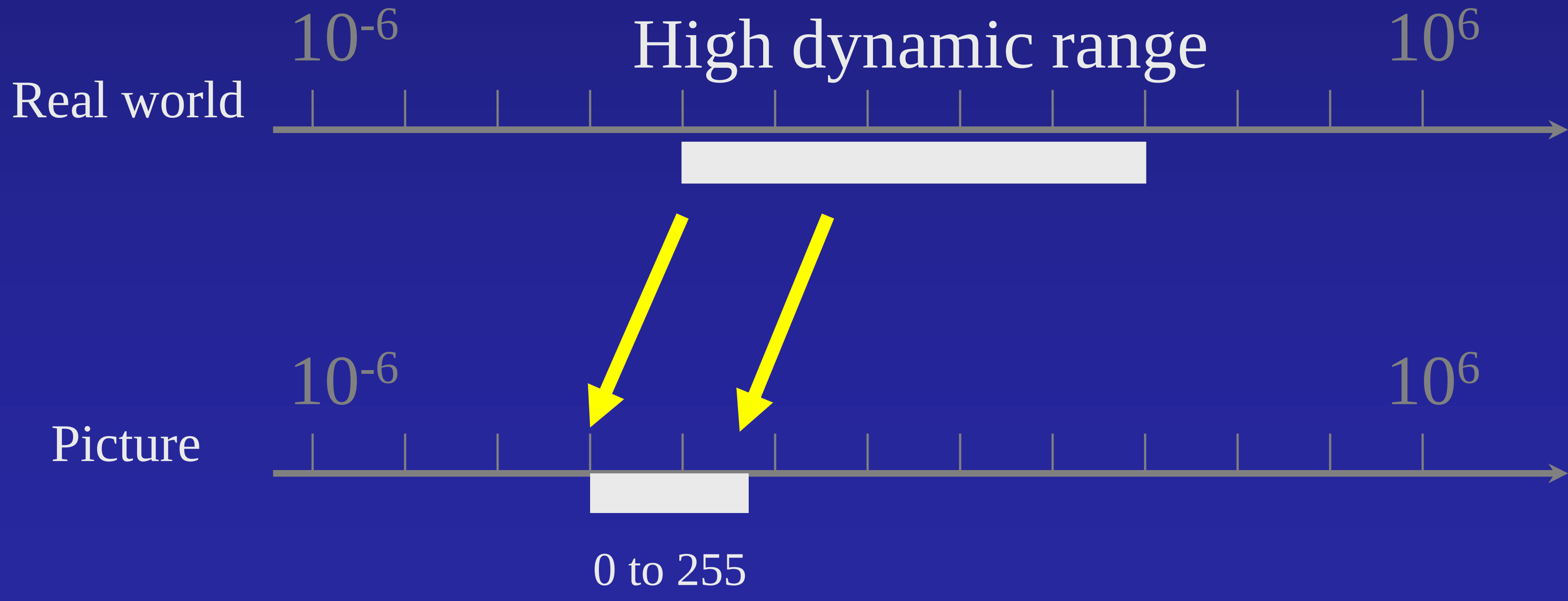
Image



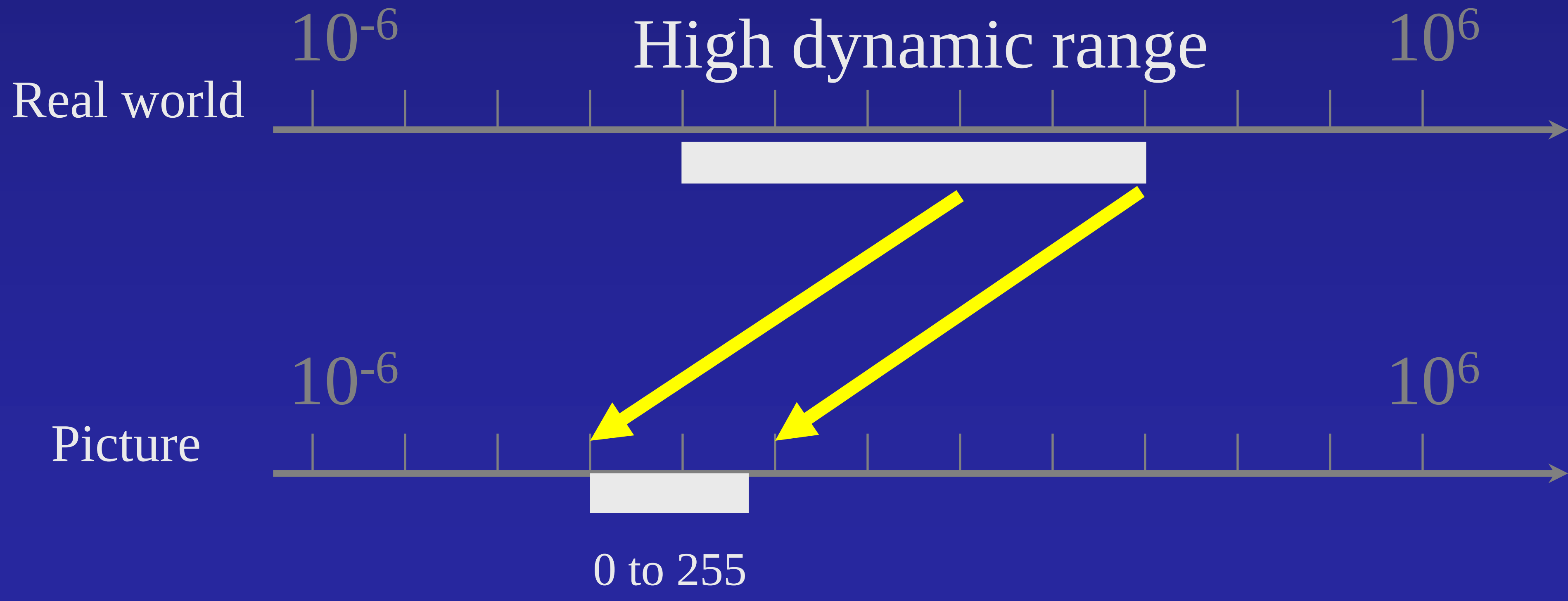
pixel (312, 284) = **42**

42 photos?

Long Exposure

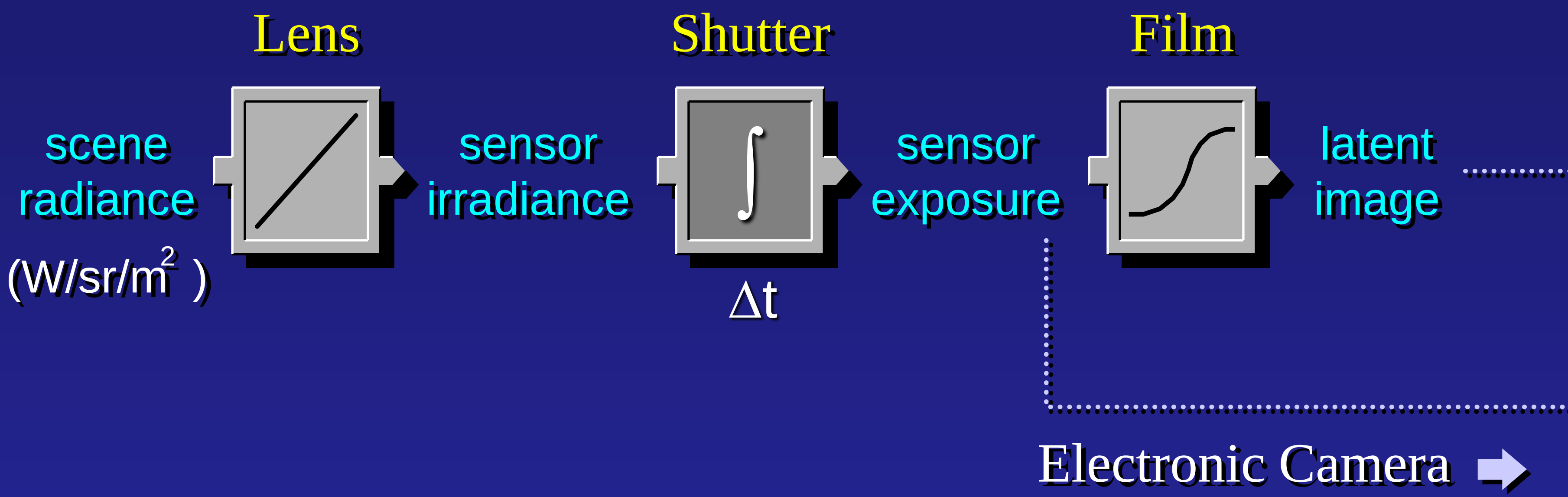


Short Exposure

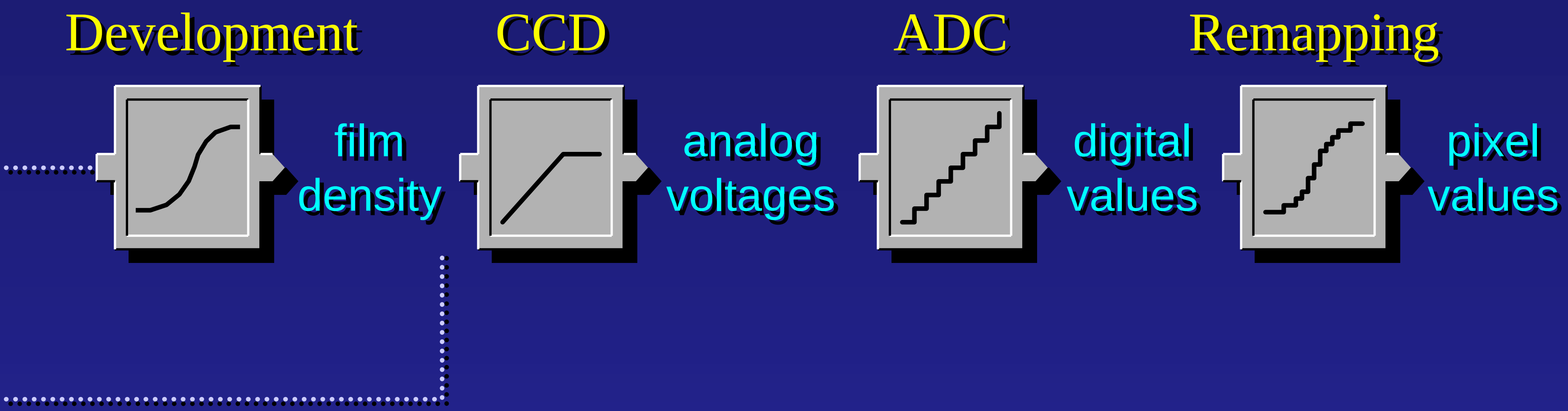


Camera Calibration

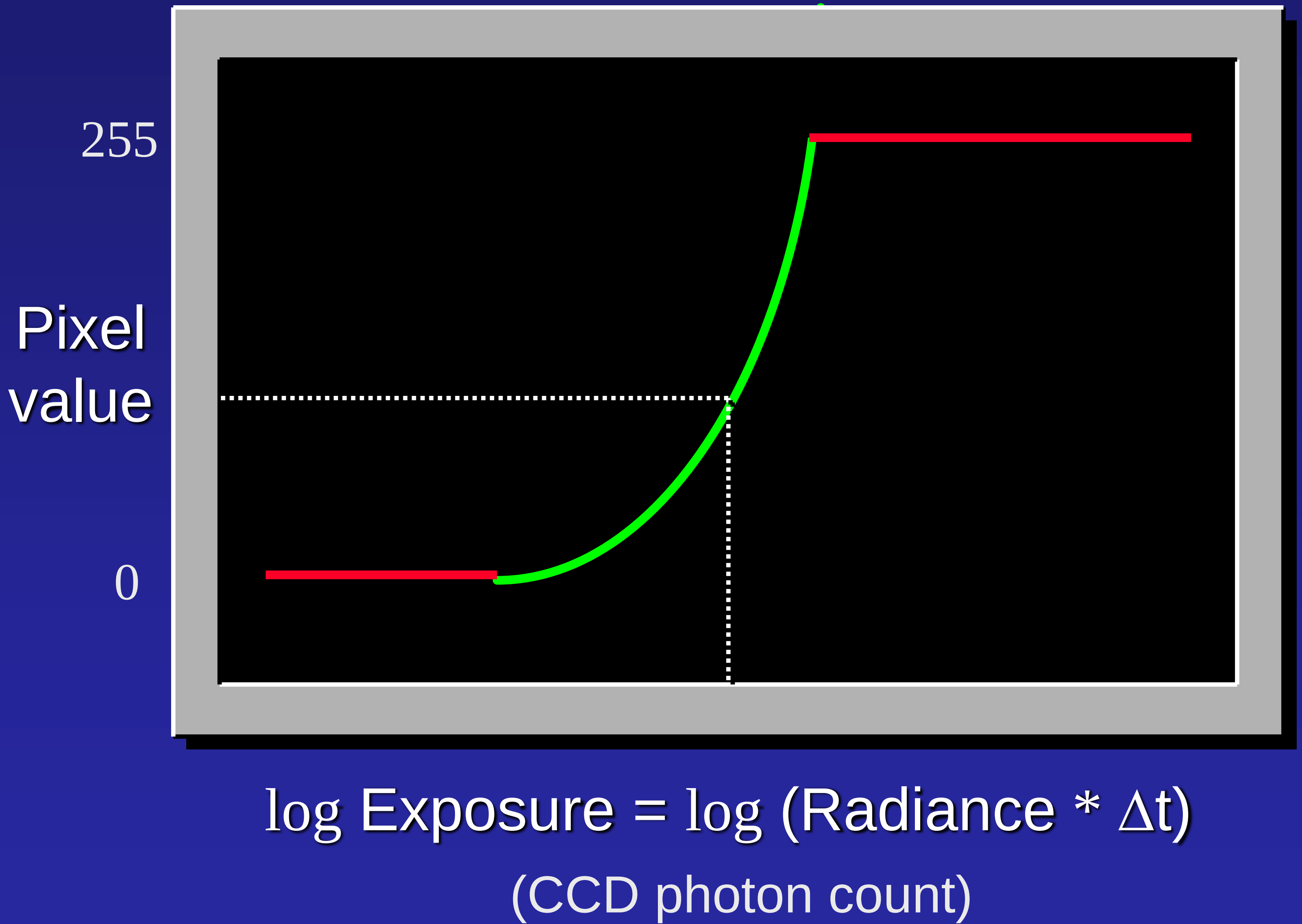
- Geometric
  - How pixel **coordinates** relate to **directions** in the world
- Photometric
  - How pixel **values** relate to **radiance** amounts in the world



The Image Acquisition Pipeline


Development
CCD
ADC
Remapping
film density
analog voltages
digital values
pixel values

Imaging system response function



Varying Exposure

Camera is not a photometer!

- Limited dynamic range
  - ⇒ Perhaps use multiple exposures?
- Unknown, nonlinear response
  - ⇒ Not possible to convert pixel values to radiance
- Solution:
  - – Recover response curve from multiple exposures, then reconstruct the ***radiance map***

Recovering High Dynamic Range Radiance Maps from Photographs

Paul Debevec
Jitendra Malik

Computer Science Division
University of California at Berkeley

August 1997

Ways to vary exposure

Ways to vary exposure

- Shutter Speed (*)
- F/stop (aperture, iris)

- Neutral Density (ND) Filters

Shutter Speed

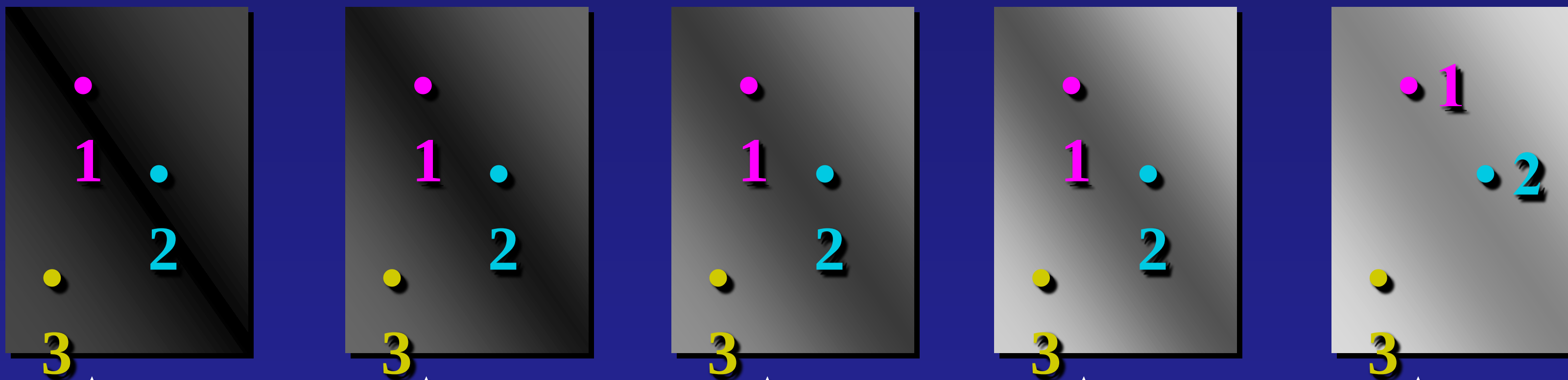
- **Ranges: Canon D30: 30 to 1/4,000 sec.**
- **Sony VX2000: ¼ to 1/10,000 sec.**
- **Pros:**
- **Directly varies the exposure**
- **Usually accurate and repeatable**
- **Issues:**
- **Noise in long exposures**

Shutter Speed

- **Note: shutter times usually obey a power series – each "stop" is a factor of 2**
- **¼, 1/8, 1/15, 1/30, 1/60, 1/125, 1/250, 1/500, 1/1000 sec**
- **Usually really is:**
- **¼, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256, 1/512, 1/1024 sec**

The Algorithm

Image series



Δt = 1/64 sec | **Δt = 1/16 sec** | **Δt = 1/4 sec** | **Δt = 1 sec** | **Δt = 4 sec**

Pixel Value Z = f(Exposure)

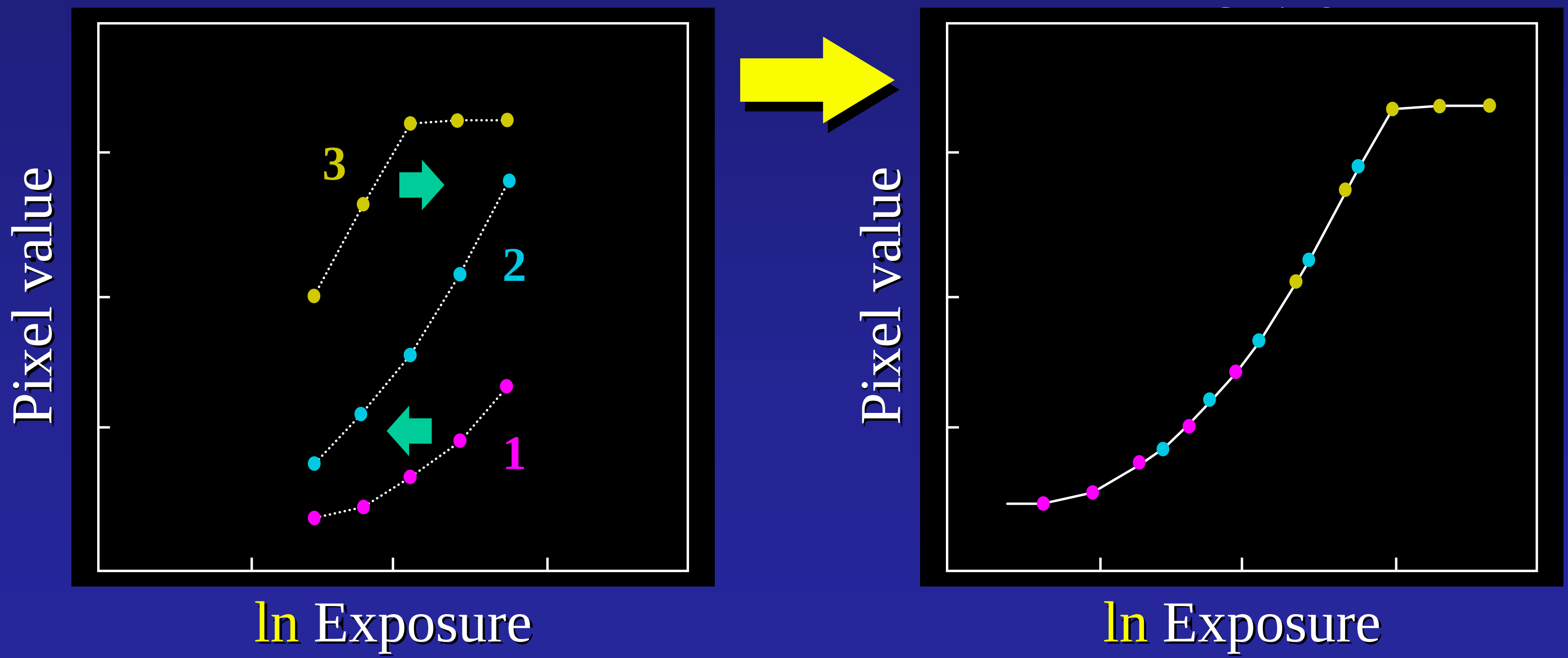
Exposure = Radiance ´ Δt

log Exposure = log Radiance + log Δt

Response Curve

Assuming unit radiance for each pixel

After adjusting radiances to obtain a smooth response



The Math

- Let *g(z)* be the *discrete* inverse response function
- For each pixel site *i* in each image *j*, want:

$$\ln Radiance_i + \ln \Delta t_j = g(Z_{ij})$$

- Solve the overdetermined linear system:

$$\sum_{i=1}^{N} \sum_{j=1}^{P} \left[\ln Radiance_i + \ln \Delta t_j - g(Z_{ij})\right]^2 + \lambda \sum_{z=Z_{min}}^{Z_{max}} g''(z)^2$$

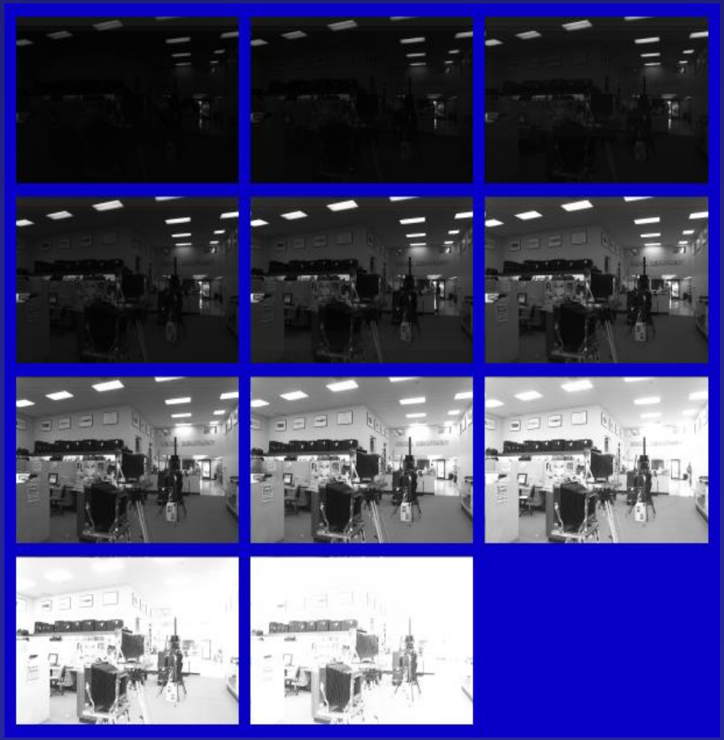
fitting term — smoothness term

Matlab Code

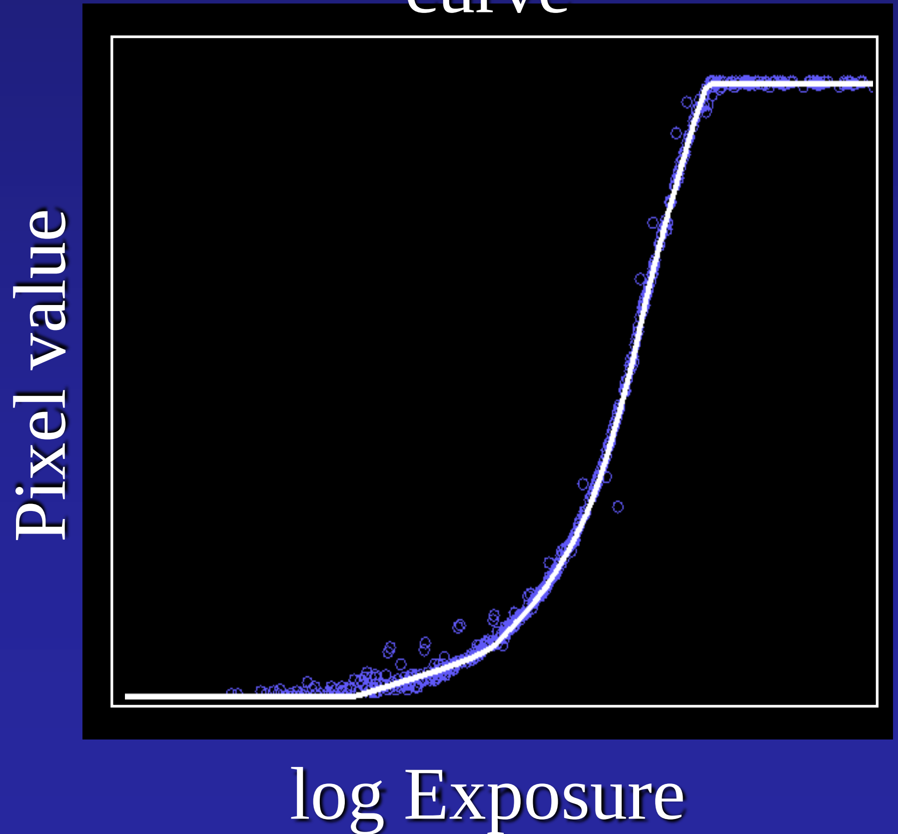
```
function [g,lE]=gsolve(Z,B,l,w)

n = 256;
A = zeros(size(Z,1)*size(Z,2)+n+1,n+size(Z,1));
b = zeros(size(A,1),1);

k = 1;                     %% Include the data-fitting equations
for i=1:size(Z,1)
  for j=1:size(Z,2)
    wij = w(Z(i,j)+1);
    A(k,Z(i,j)+1) = wij; A(k,n+i) = -wij; b(k,1) = wij * B(i,j);
    k=k+1;
  end
end

A(k,129) = 1;              %% Fix the curve by setting its middle value to
k=k+1;

for i=1:n-2                %% Include the smoothness equations
  A(k,i)=l*w(i+1); A(k,i+1)=-2*l*w(i+1); A(k,i+2)=l*w(i+1);
  k=k+1;
end

x = A\b;                   %% Solve the system using SVD

g = x(1:n);
lE = x(n+1:size(x,1));
```

Results: Digital Camera

Kodak DCS460
1/30 to 30 sec



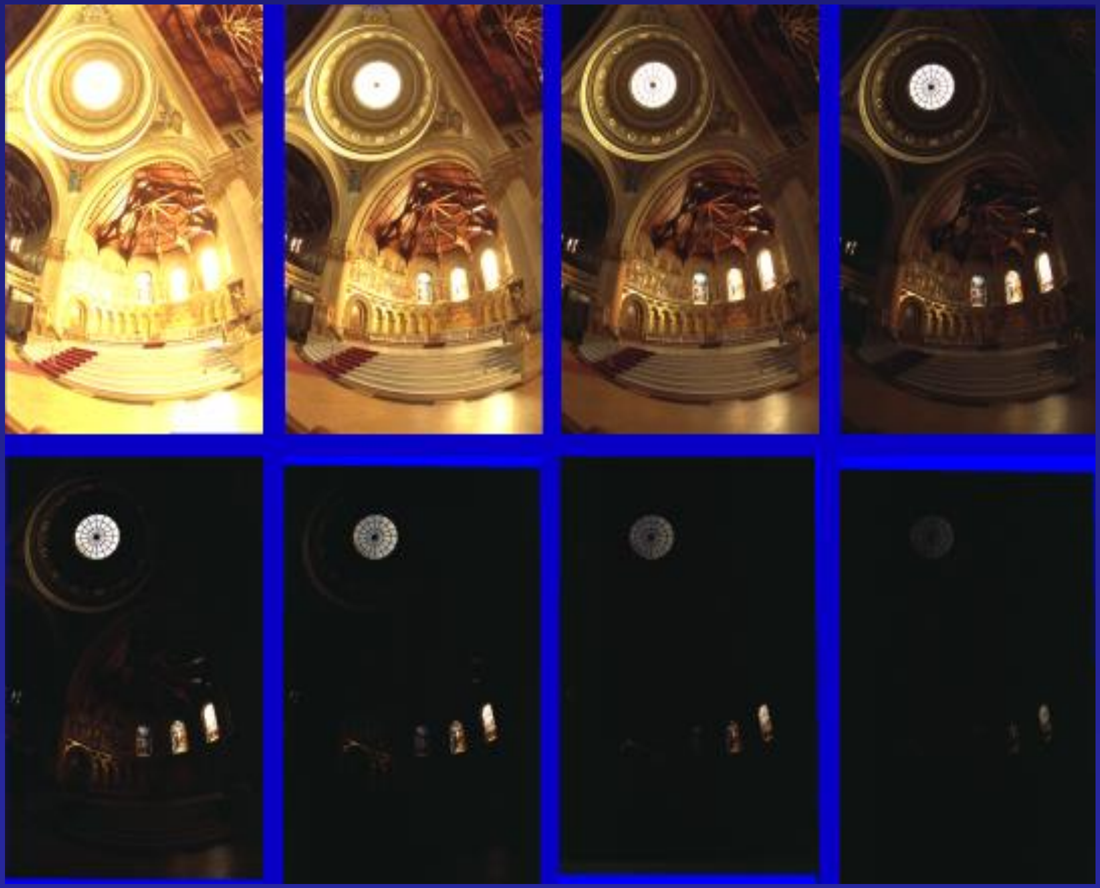
Recovered response curve



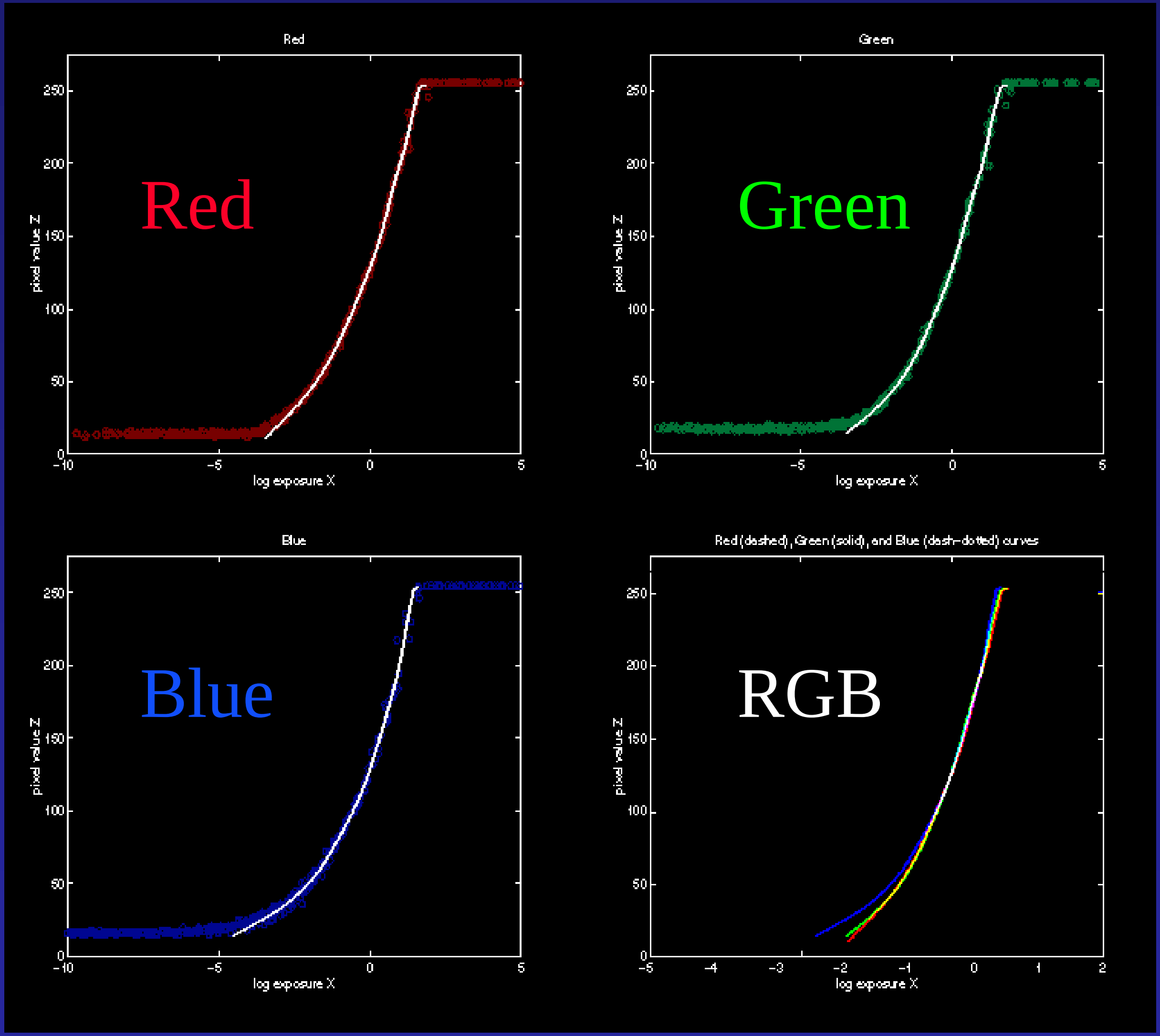
Reconstructed radiance map

Results: Color Film

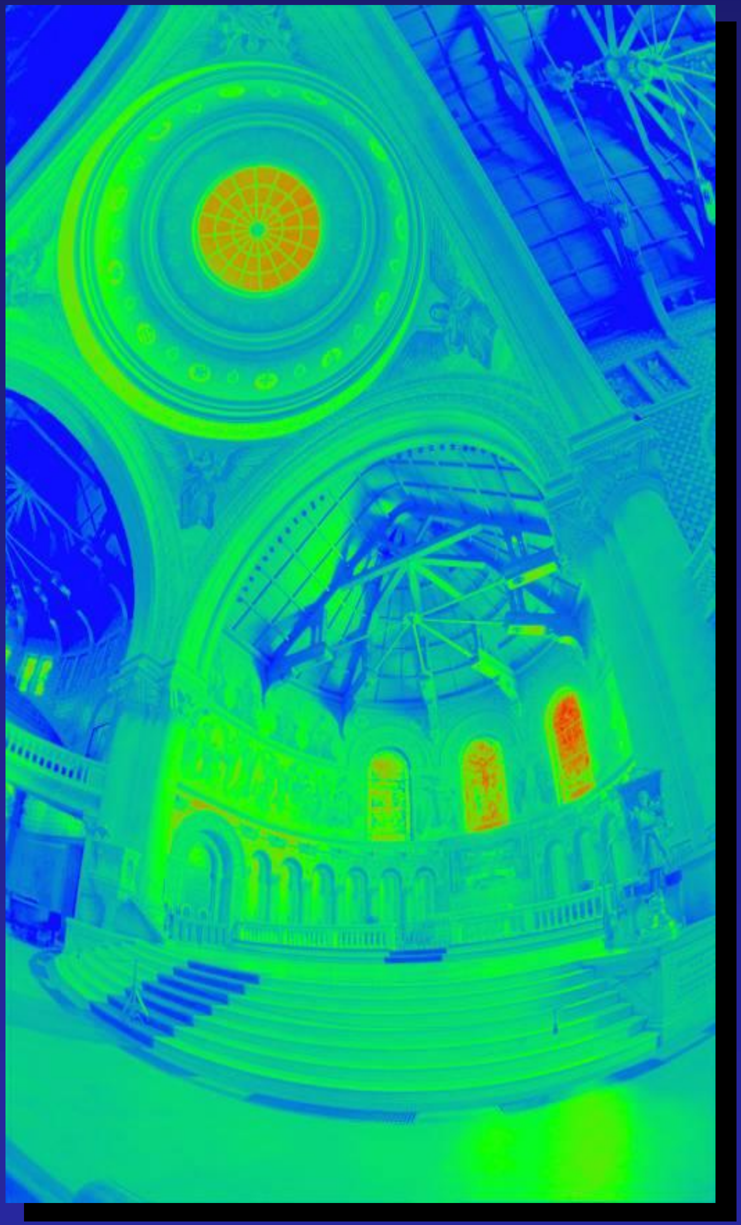
- Kodak Gold ASA 100, PhotoCD



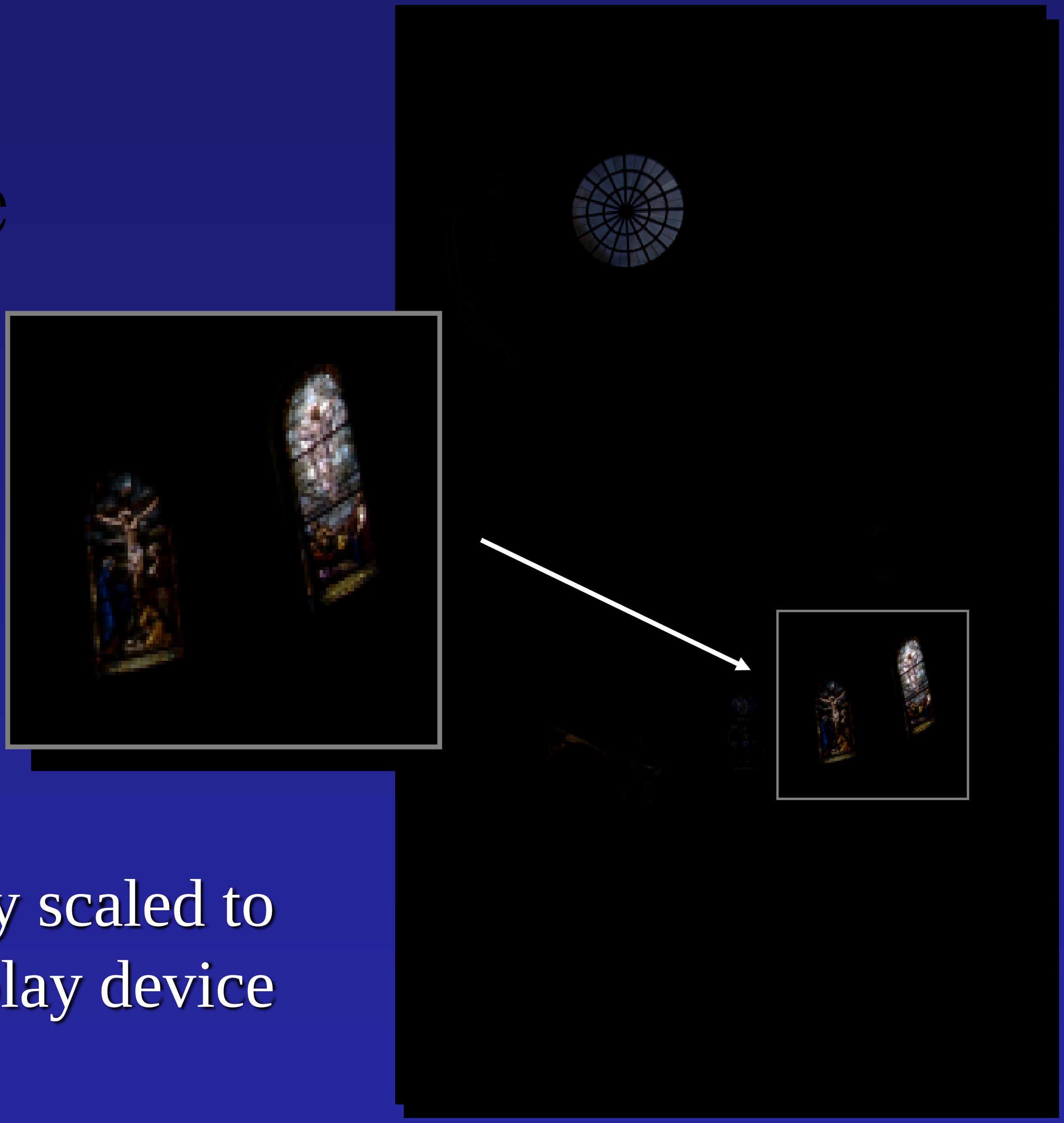
Recovered Response Curves



The Radiance Map

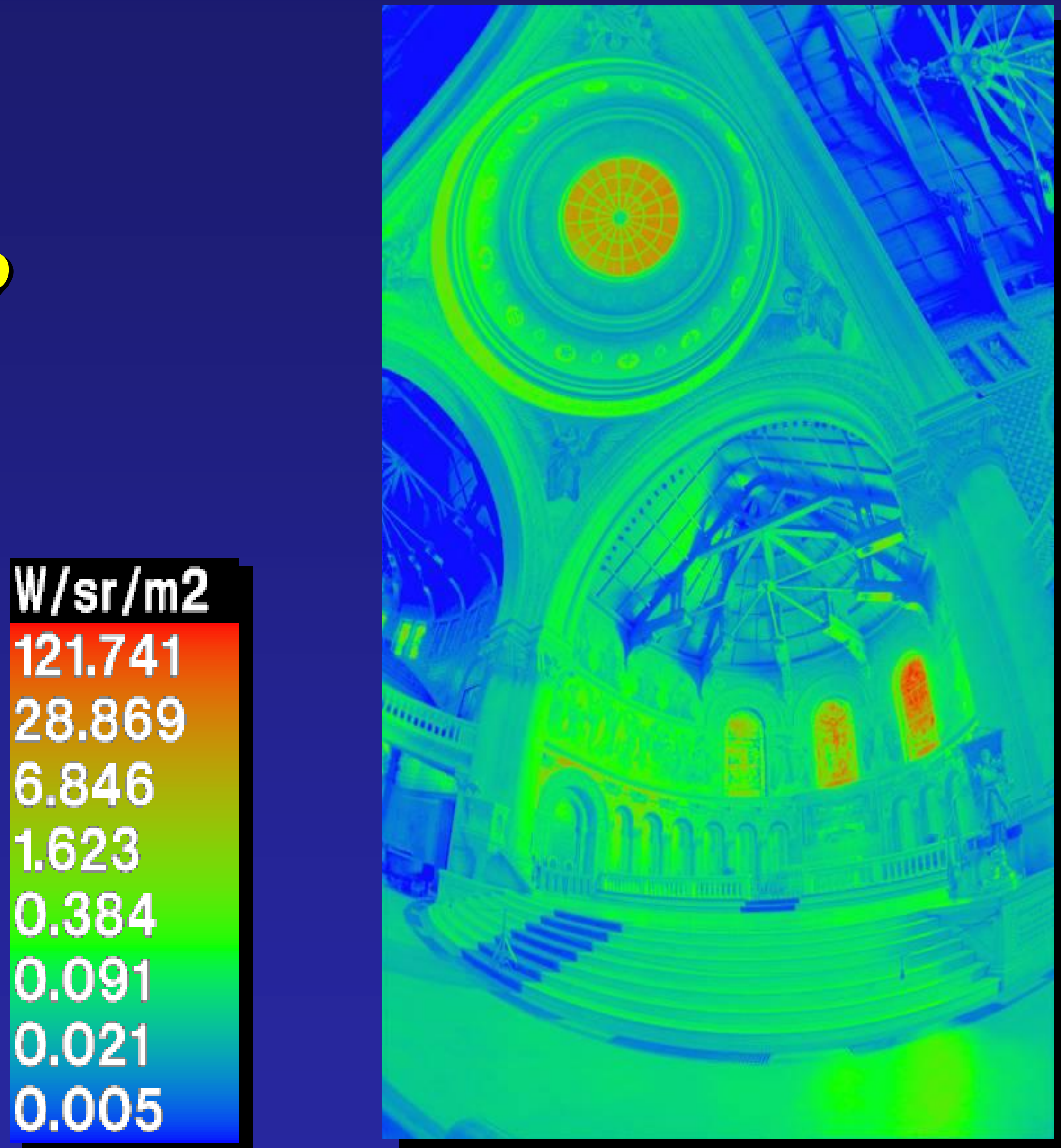


The Radiance Map



Linearly scaled to display device

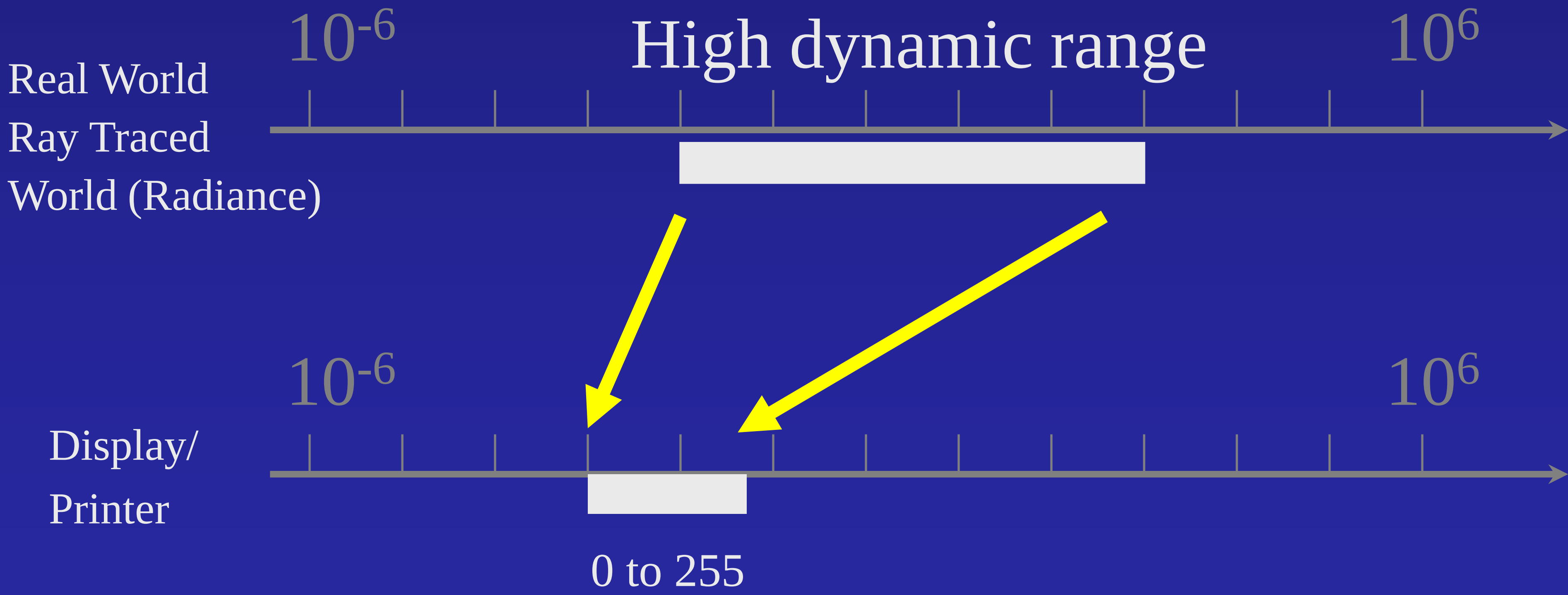
Now What?



Tone Mapping

- How can we do this?

  Linear scaling?, thresholding? Suggestions?

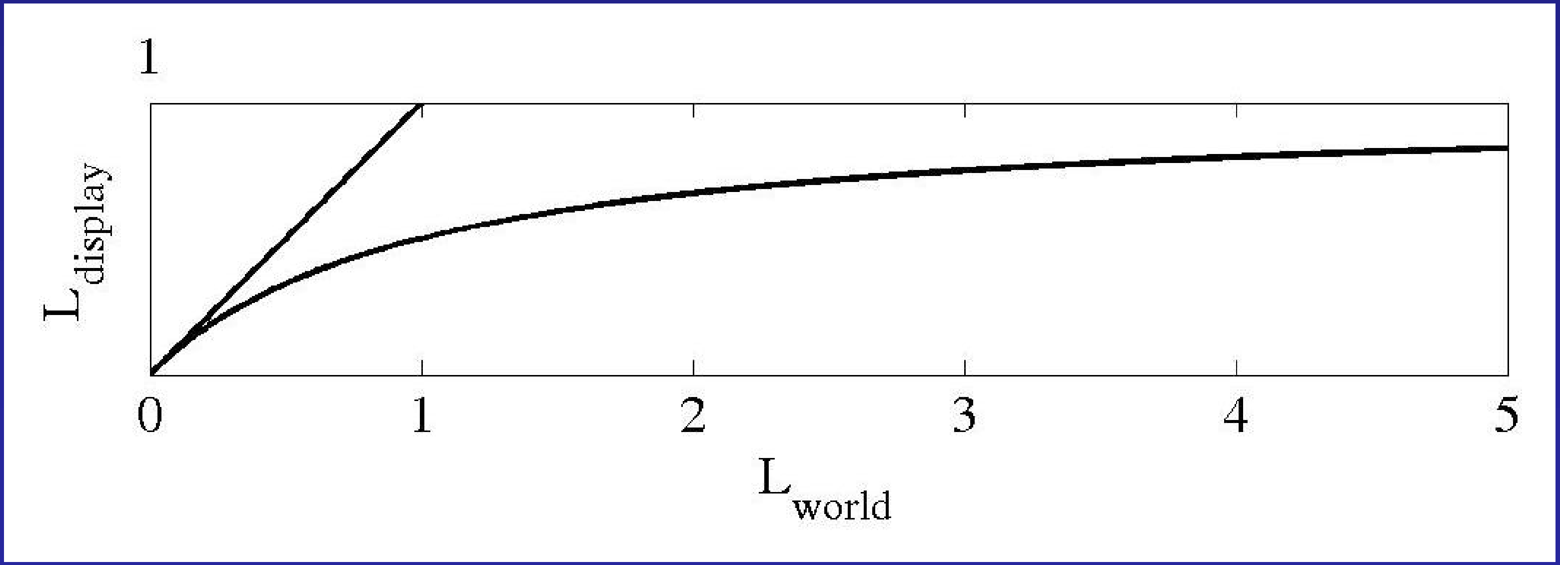


Simple Global Operator

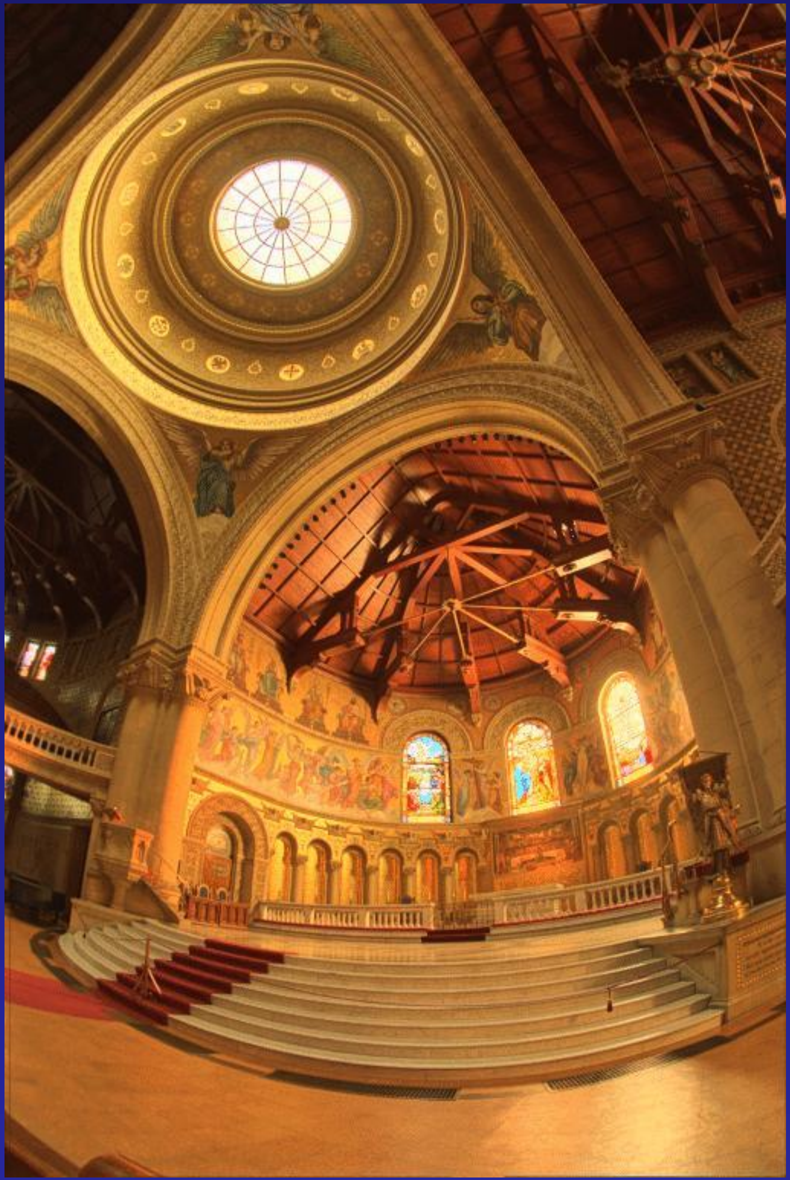
- Compression curve needs to
  - Bring everything within range
  - Leave dark areas alone
- In other words
  - Asymptote at 255
  - Derivative of 1 at 0

Global Operator (Reinhart et al)

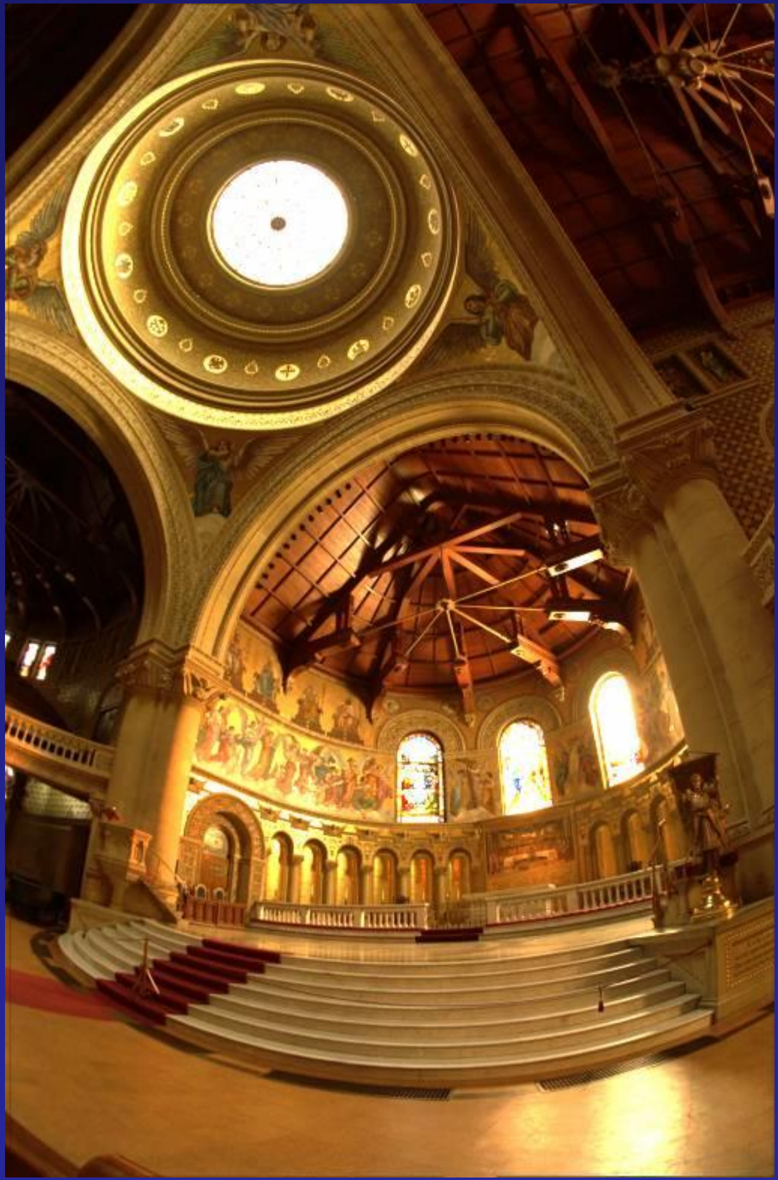
$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$



Global Operator Results

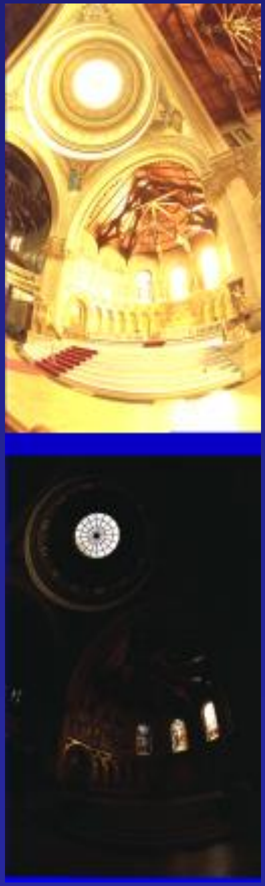


Reinhart Operator

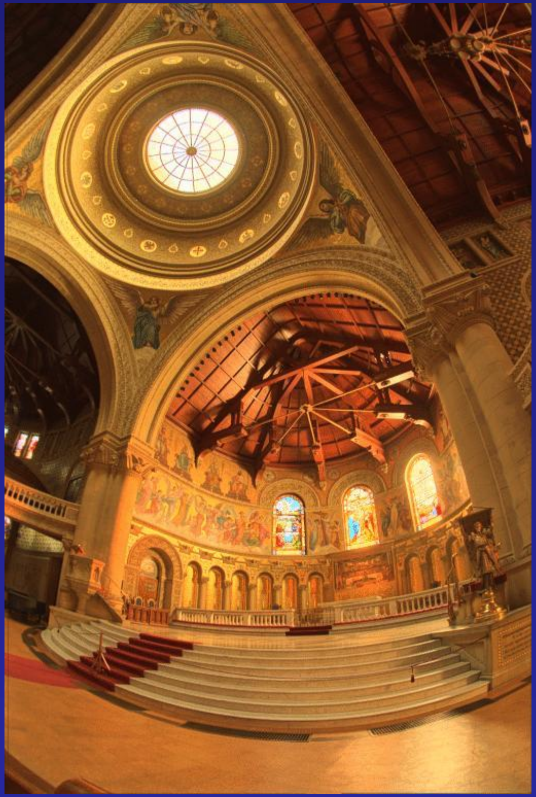


Darkest 0.1% scaled to display device

What do *we* see?



Vs.



What does the eye sees?

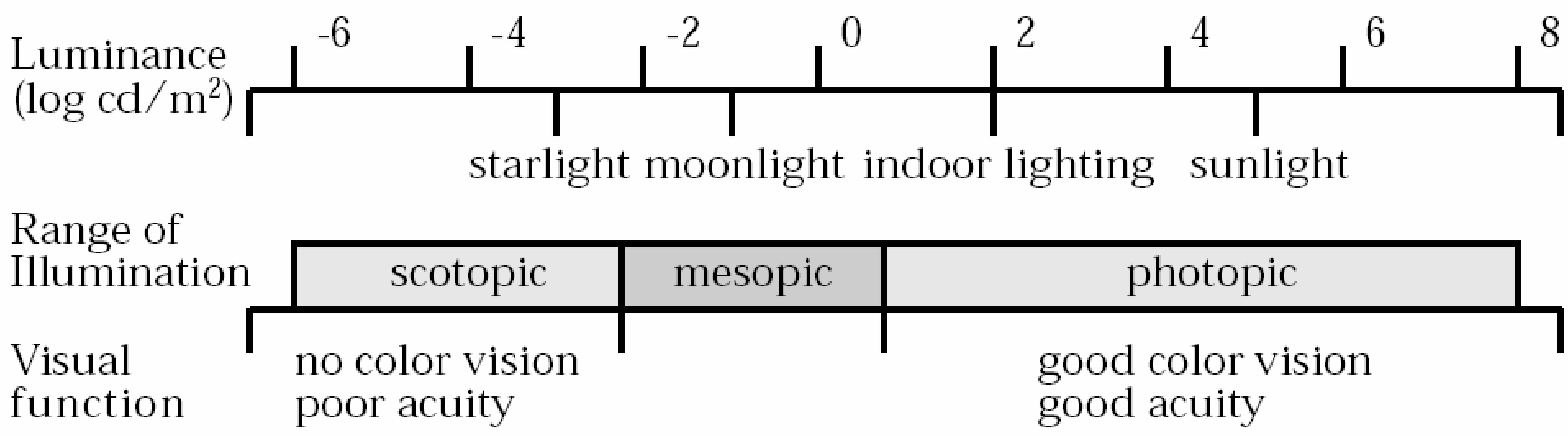
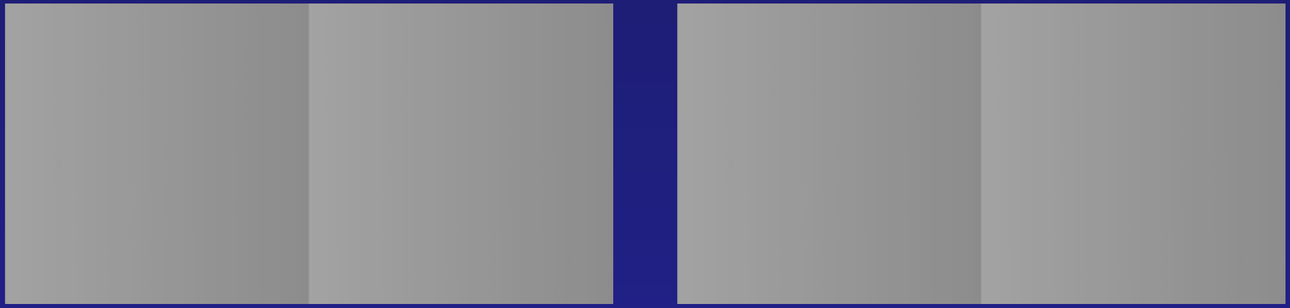


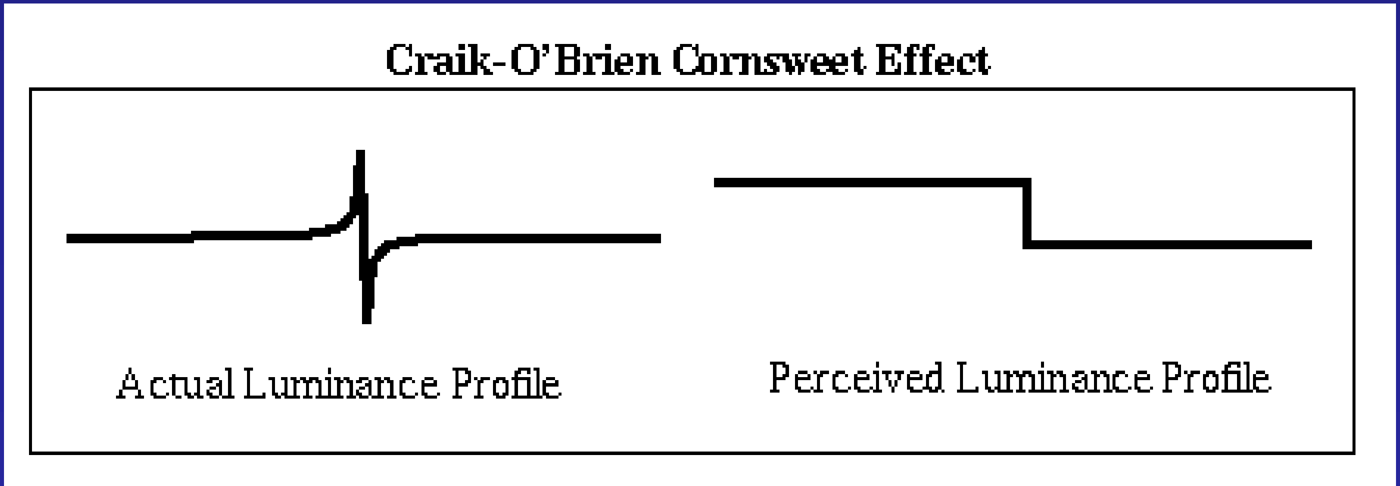
Figure 1: The range of luminances in the natural environment and associated visual parameters. After Hood (1986).

The eye has a huge dynamic range

Do we see a true radiance map?

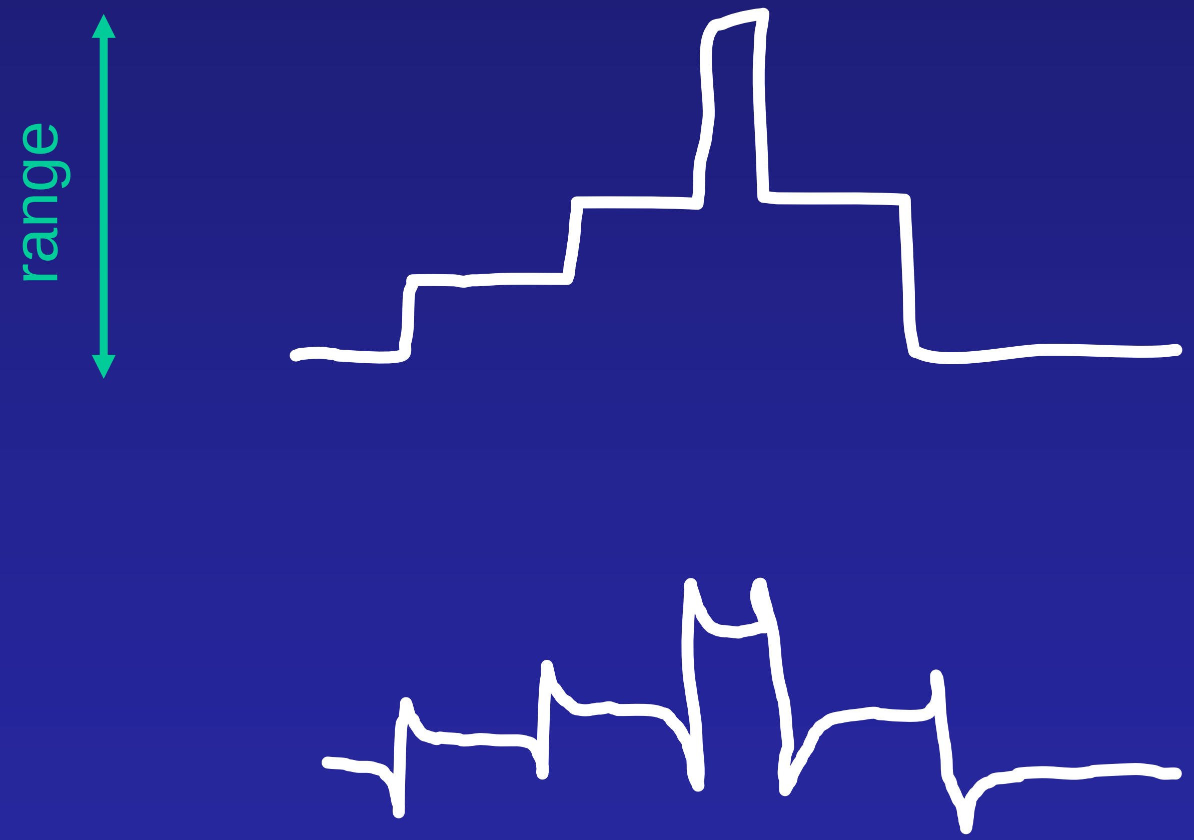
Metamores





Can we use this for range compression?

Compressing Dynamic Range



Inserting Synthetic Objects

Why does this look so bad?

- Wrong camera orientation
- Wrong lighting
- No shadows

Solutions

Wrong Camera Orientation

- Estimate correct camera orientation and renender object
  - Requires camera calibration to do it right

Lighting & Shadows

- Estimate (eyeball) all the light sources in the scene and simulate it in your virtual rendering

But what happens if lighting is complex?

- Extended light sources, mutual illumination, etc.

Environment Maps

+

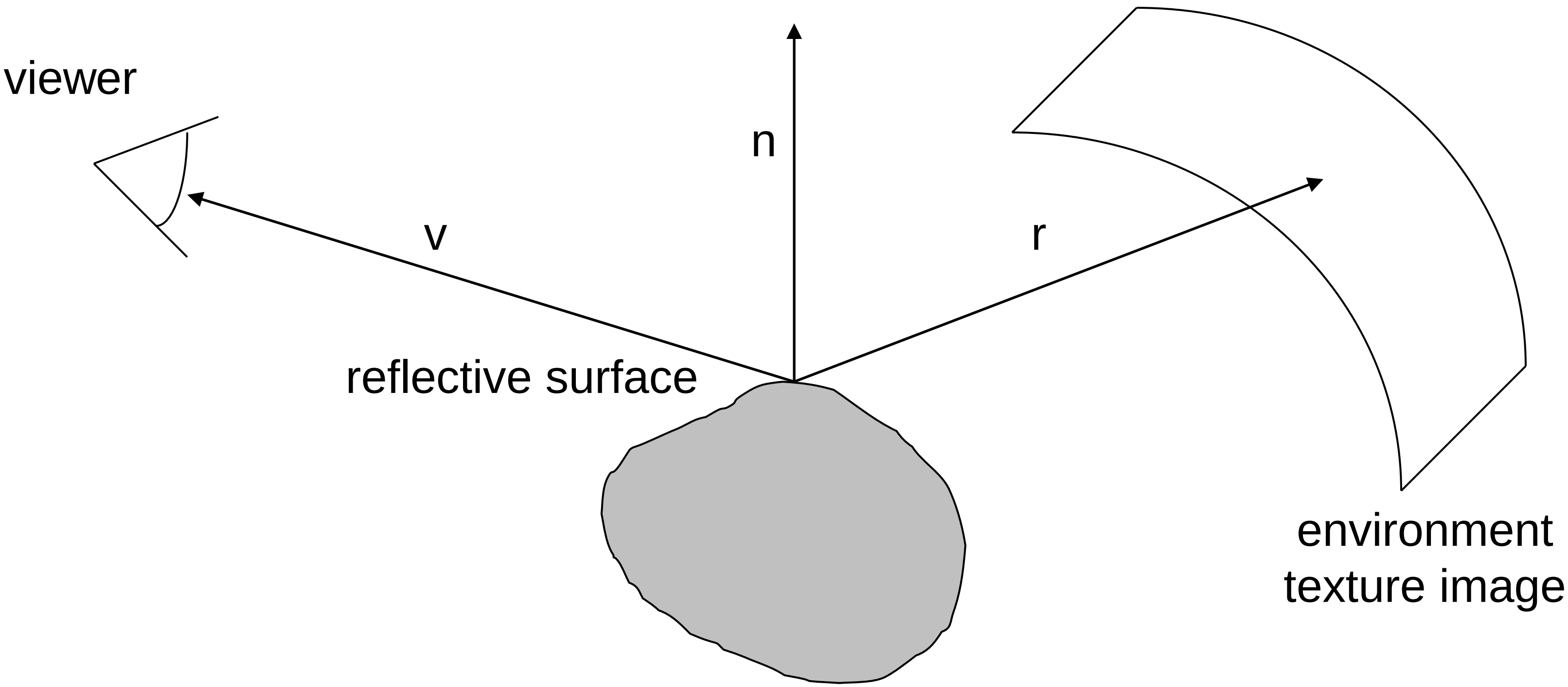
Simple solution for shiny objects

- Models complex lighting as a panoramic image
- i.e. amount of radiance coming in from each direction
- A plenoptic function!!!

Environment Mapping

projector function converts reflection vector (x, y, z) to texture image (u, v)

Reflected ray: $\boldsymbol{r}=2(\boldsymbol{n}\cdot\boldsymbol{v})\boldsymbol{n}-\boldsymbol{v}$



Texture is transferred in the direction of the reflected ray from the environment map onto the object

What is in the map?

Environment Maps

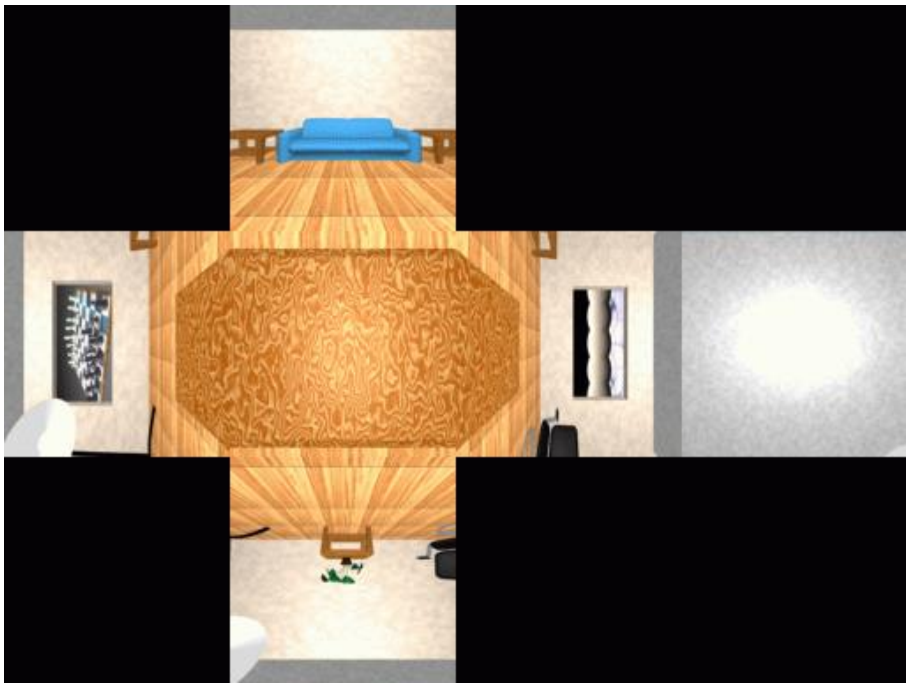
The environment map may take various forms:

- Cubic mapping
- Spherical mapping
- other

Describes the shape of the surface on which the map "resides"

Determines how the map is generated and how it is indexed

Cubic Map Example



Cubic Mapping

The map resides on the surfaces of a cube around the object

- Typically, align the faces of the cube with the coordinate axes

To generate the map:

- For each face of the cube, render the world from the center of the object with the cube face as the image plane
  - Rendering can be arbitrarily complex (it's off-line)

To use the map:

- Index the R ray into the correct cube face
- Compute texture coordinates

Spherical Map Example

Sphere Mapping

Map lives on a sphere

To generate the map:

- Render a spherical panorama from the designed center point

To use the map:

- Use the orientation of the R ray to index directly into the sphere

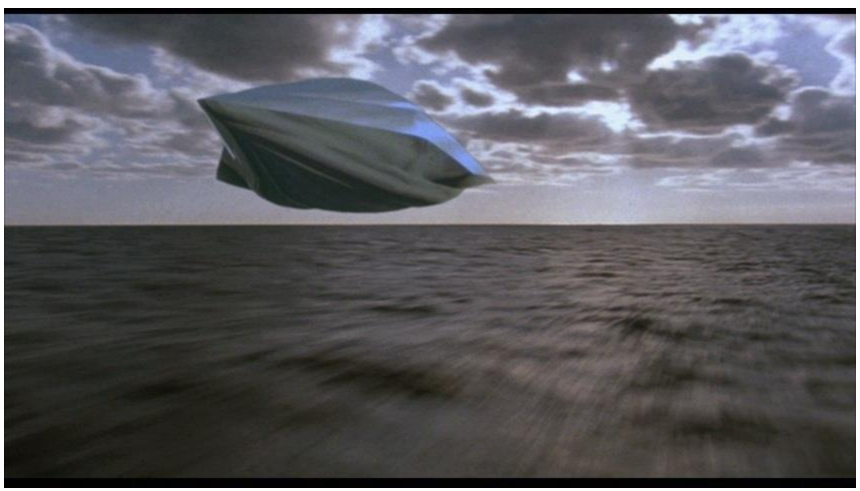
What approximations are made?

The map should contain a view of the world with the point of interest on the object as the Center of Projection

- We can’t store a separate map for each point, so one map is used with the COP at the center of the object
- Introduces distortions in the reflection, but we usually don’t notice
- Distortions are minimized for a small object in a large room

The object will not reflect itself!

What about real scenes?



From *Flight of the Navigator*

What about real scenes?

from Terminator 2

Real environment maps

We can use photographs to capture environment maps

- The first use of panoramic mosaics

How do we deal with light sources? Sun, lights, etc?

- They are much much brighter than the rest of the enviarnment

User High Dynamic Range photography, of course!

Several ways to acquire environment maps:

- Stitching HDR mosaics
- Fisheye lens
- Mirrored Balls

Scanning Panoramic Cameras

Pros:

- *very* high res (10K x 7K+)
- Full sphere in one scan – no stitching
- Good dynamic range, some are HDR

Issues:

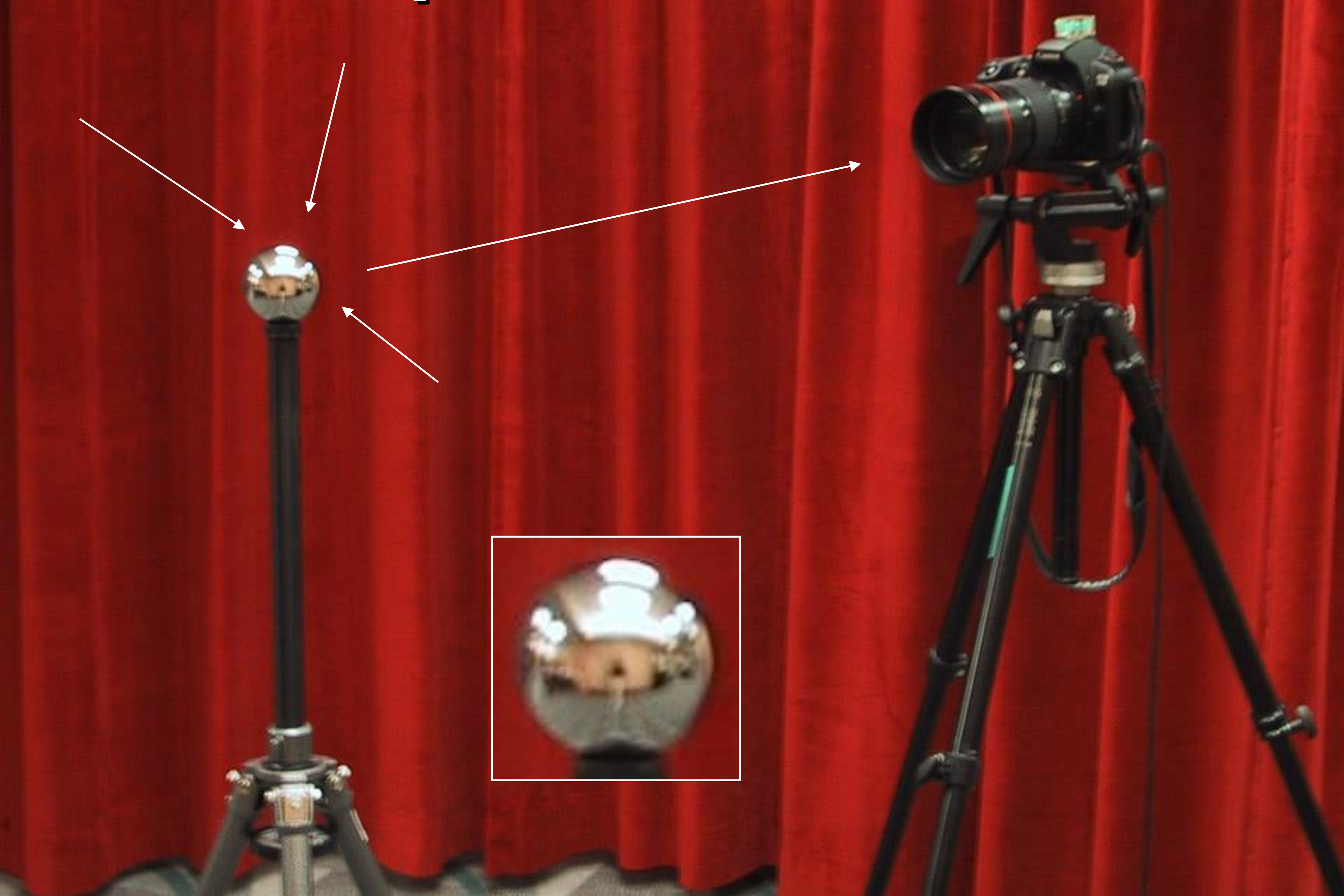
- More expensive
- Scans take a while

Companies: Panoscan, Sphereon

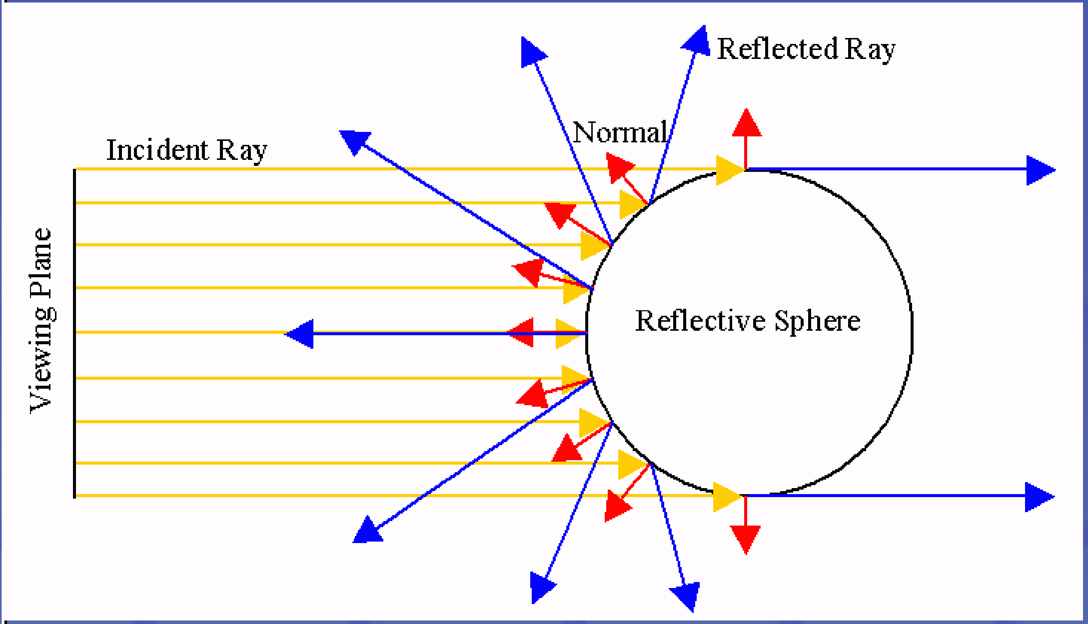
See also www.kaidan.com

Fisheye Images

Mirrored Sphere

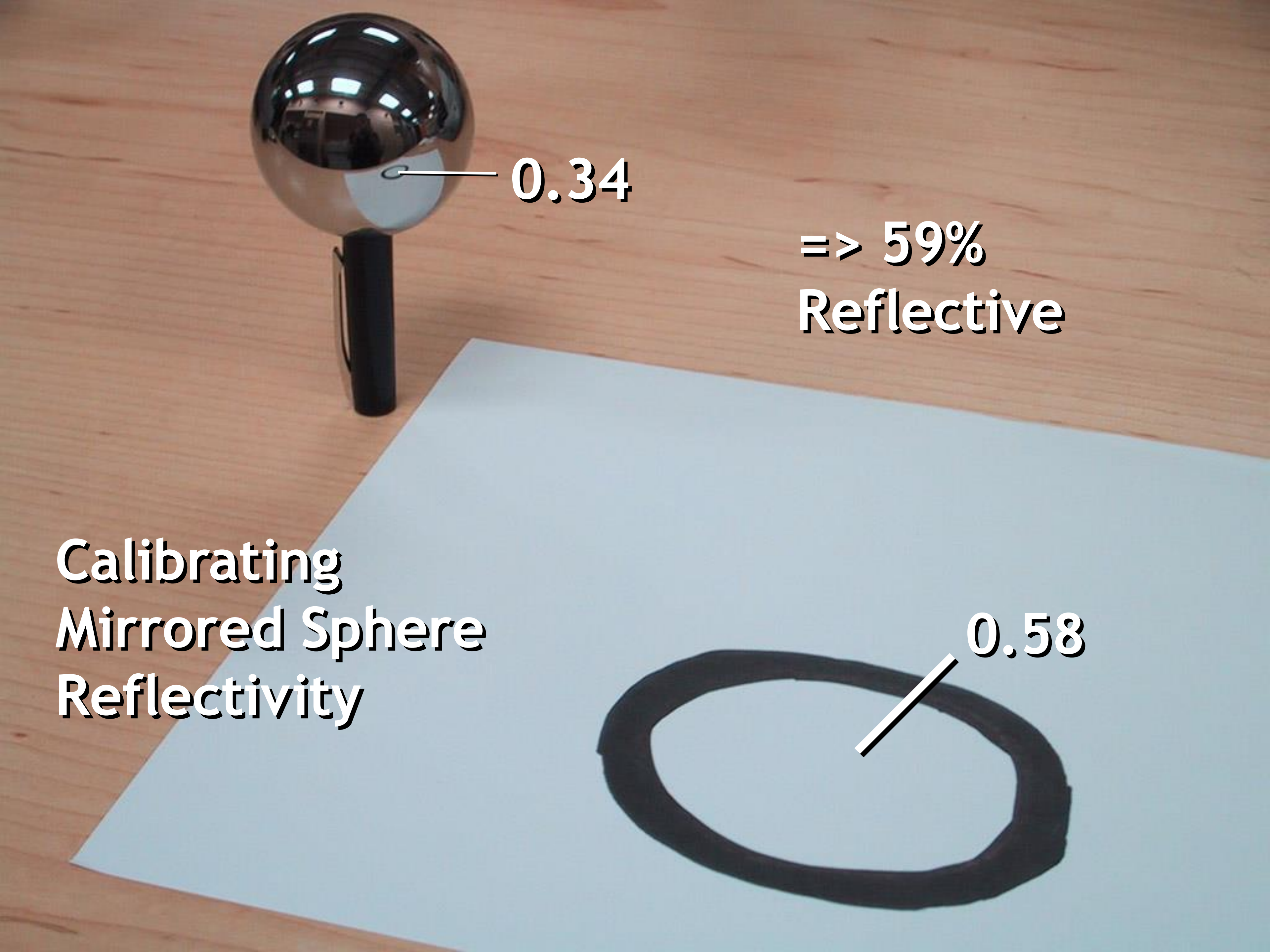



Canon
Canon
REMOTE SWITCH


Reflected Ray
Normal
Incident Ray
Viewing Plane
Reflective Sphere

Sources of Mirrored Balls

- 2-inch chrome balls ~ $20 ea.
  - McMaster-Carr Supply Company www.mcmaster.com
- 6-12 inch large gazing balls
  - Baker's Lawn Ornaments www.bakerslawnorn.com
- Hollow Spheres, 2in – 4in
  - Dube Juggling Equipment www.dube.com
- FAQ on www.debevec.org/HDRShop/


0.34
=> 59%
Reflective
Calibrating
Mirrored Sphere
Reflectivity
0.58

Real-World HDR Lighting Environments

Funston Beach

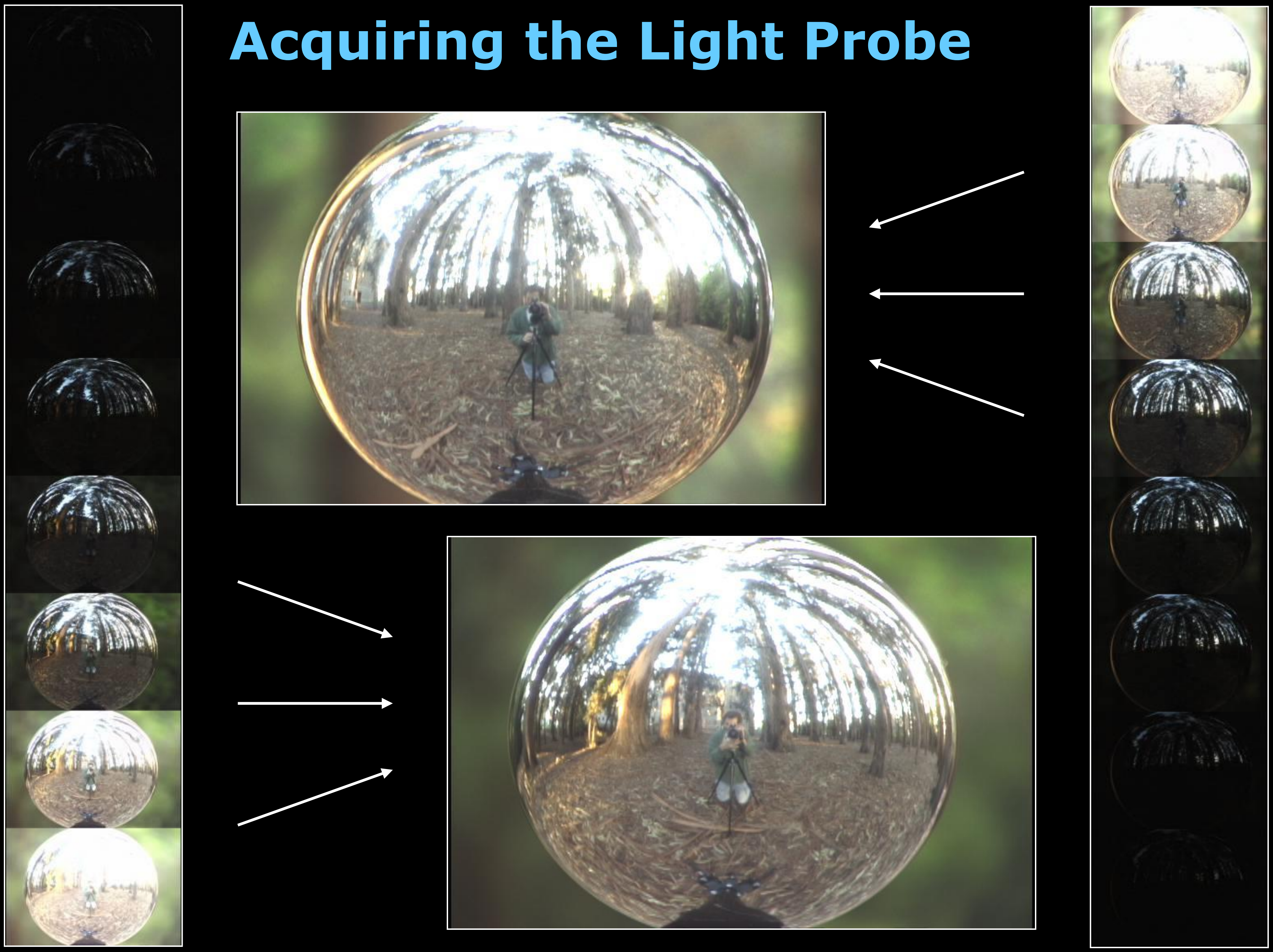
Eucalyptus Grove

Uffizi Gallery

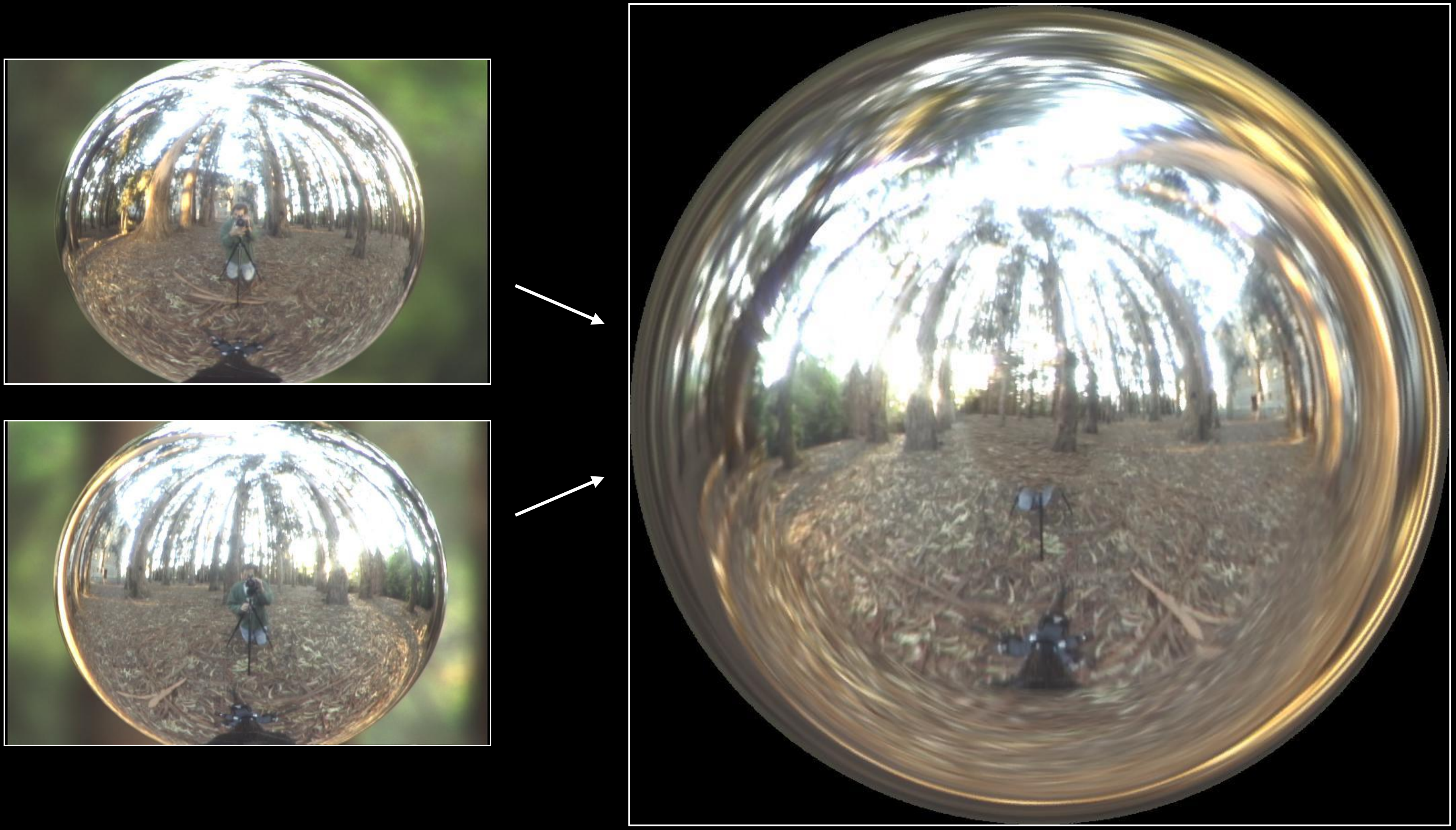
Grace Cathedral

Lighting Environments from the Light Probe Image Gallery:
http://www.debevec.org/Probes/

Acquiring the Light Probe



Assembling the Light Probe



Not just shiny…

SIGGRAPH2004

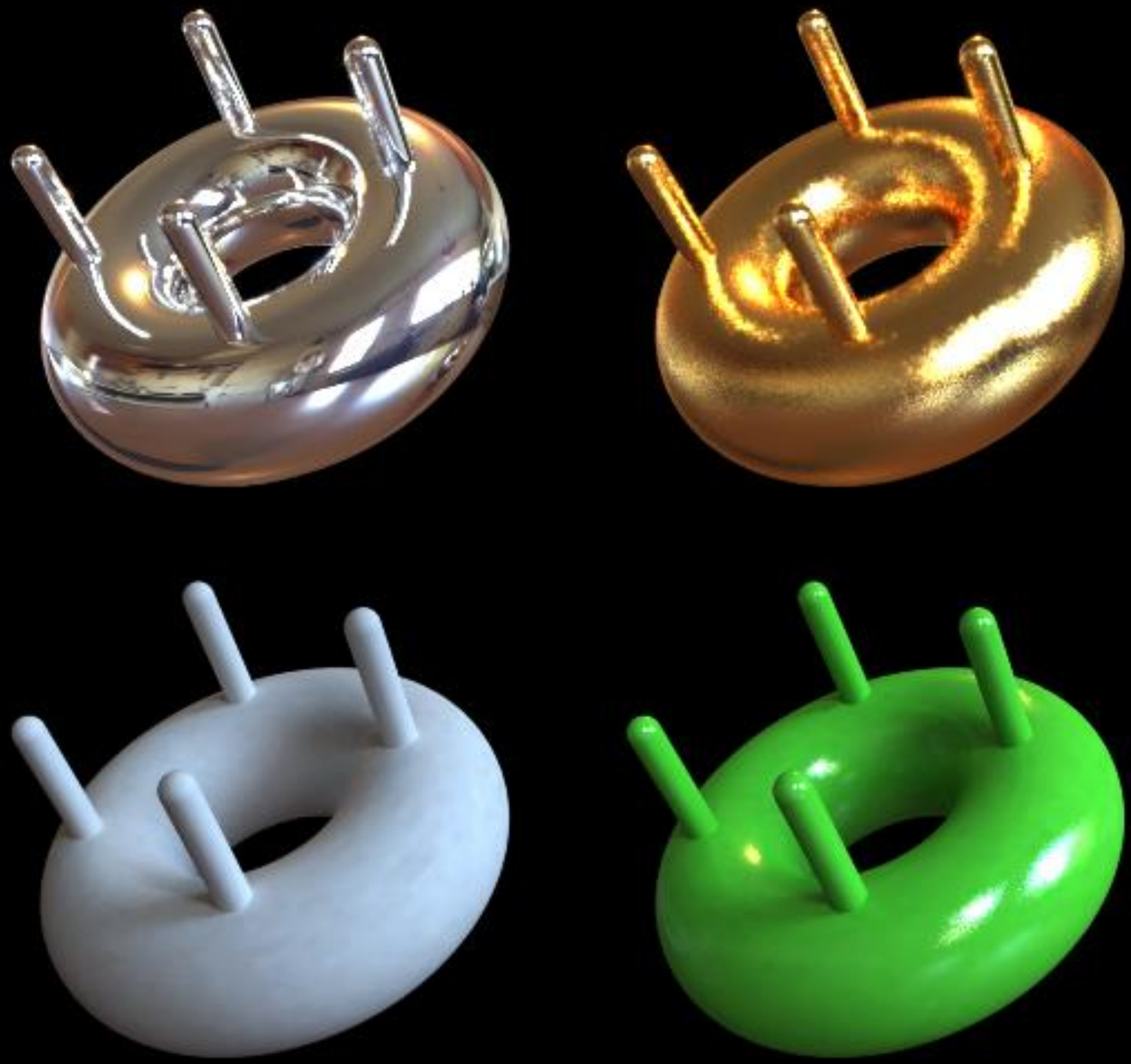
We have captured a true radiance map

We can treat it as an extended (e.g spherical) light source

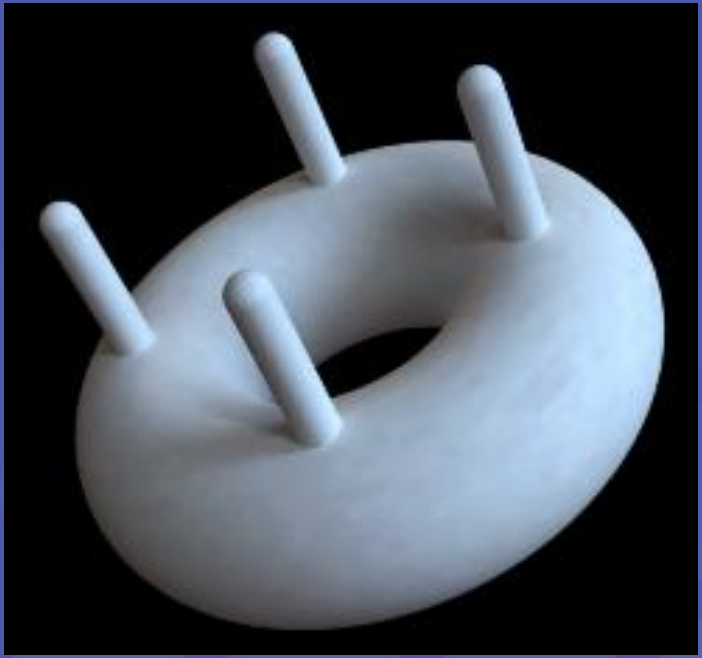
Can use Global Illumination to simulate light transport in the scene

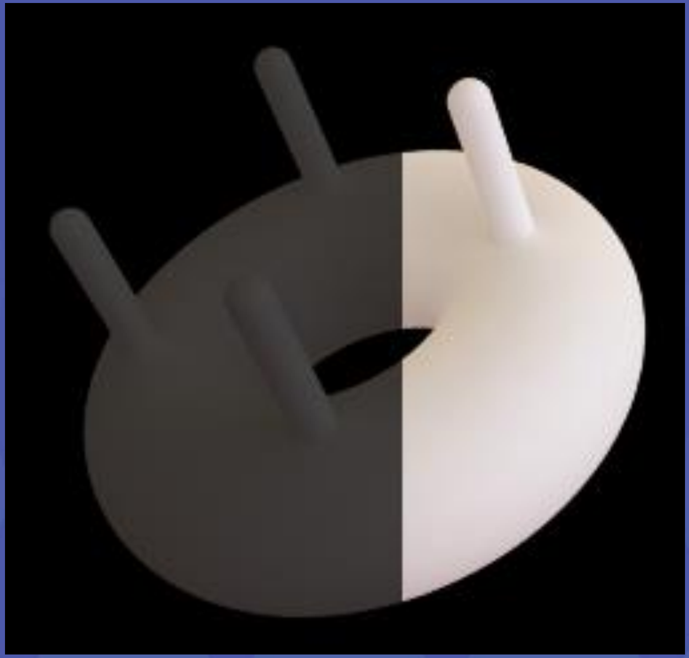
- So, all objects (not just shiny) can be lighted
- What’s the limitation?

Illumination Results



Comparison: Radiance map versus single image



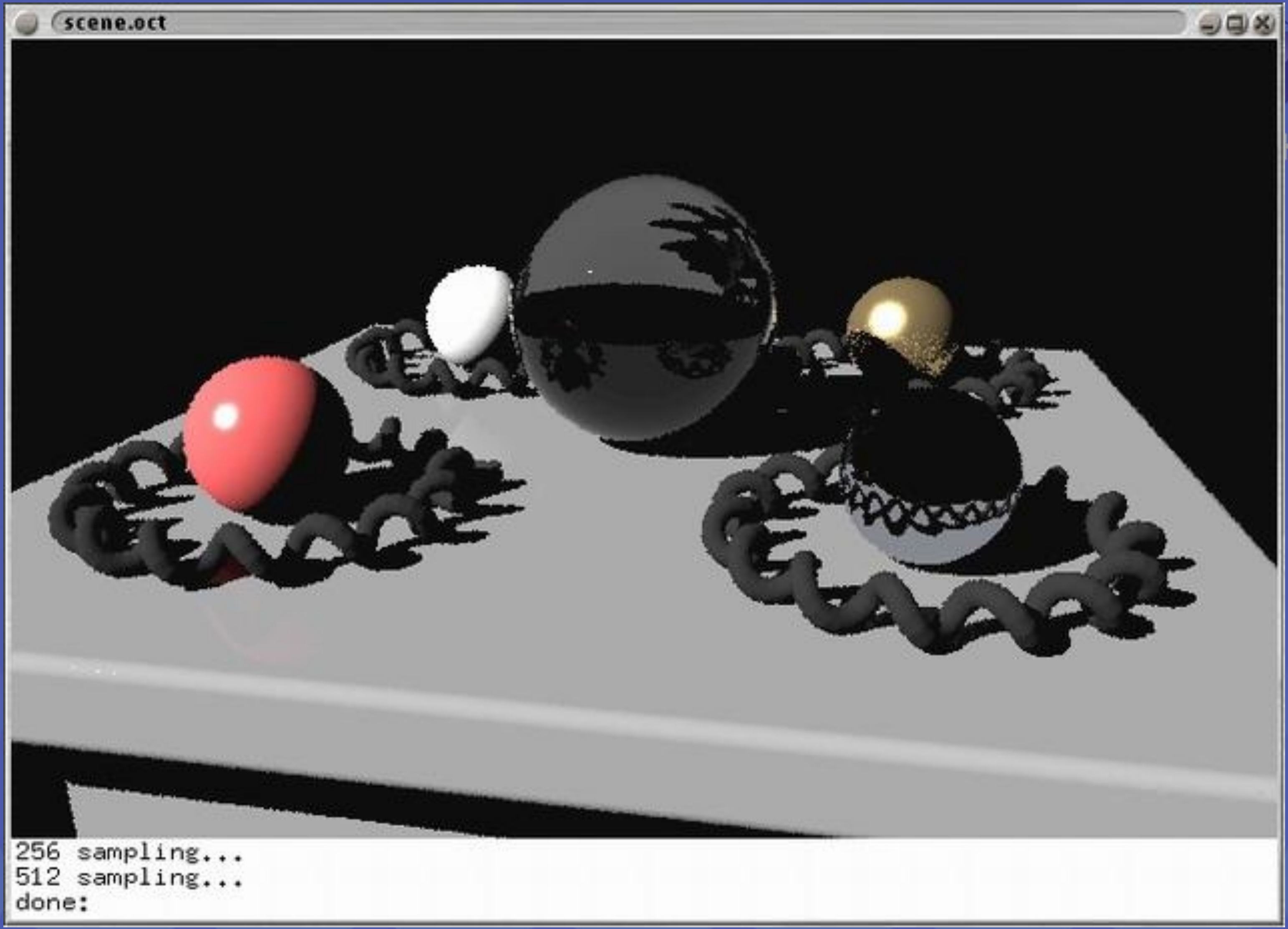


Putting it all together

Synthetic Objects

+

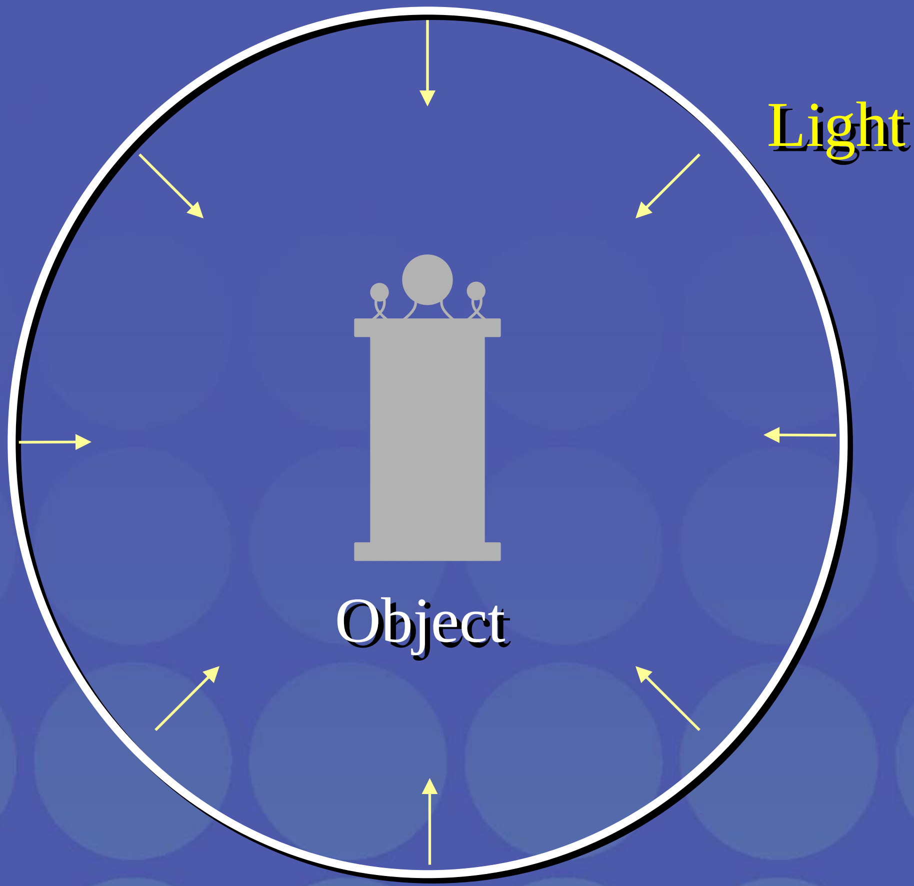
Real light!



CG Objects Illuminated by a Traditional CG Light Source

Illuminating Objects using Measurements of Real Light

SIGGRAPH2004



Environment assigned "glow" material property in Greg Ward's RADIANCE system.

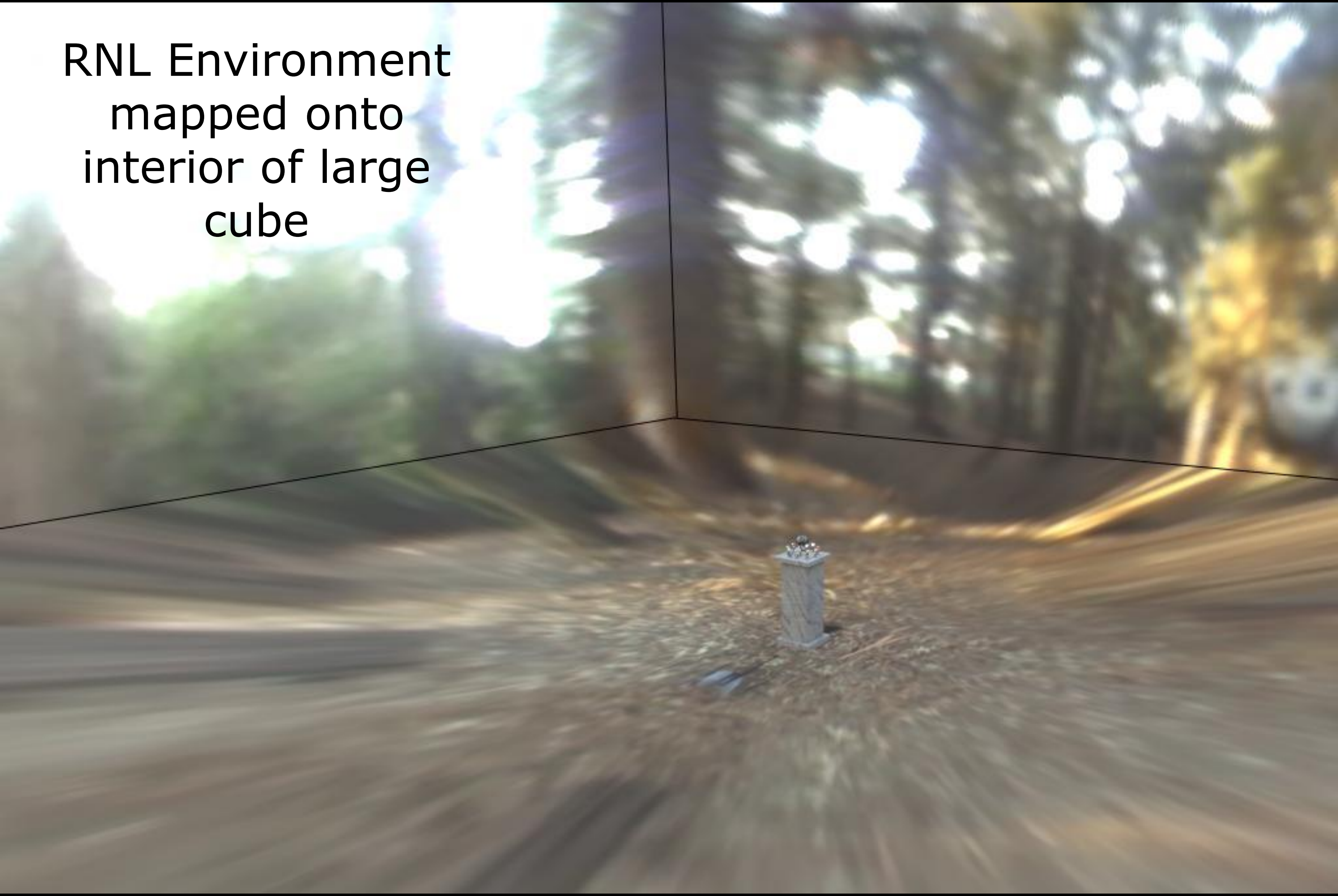
http://radsite.lbl.gov/radiance/

Paul Debevec. A Tutorial on Image-Based Lighting. IEEE Computer Graphics and Applications, Jan/Feb 2002.

Rendering with Natural Light

SIGGRAPH 98 Electronic Theater

RNL Environment mapped onto interior of large cube



SIGGRAPH2004

MOVIE!

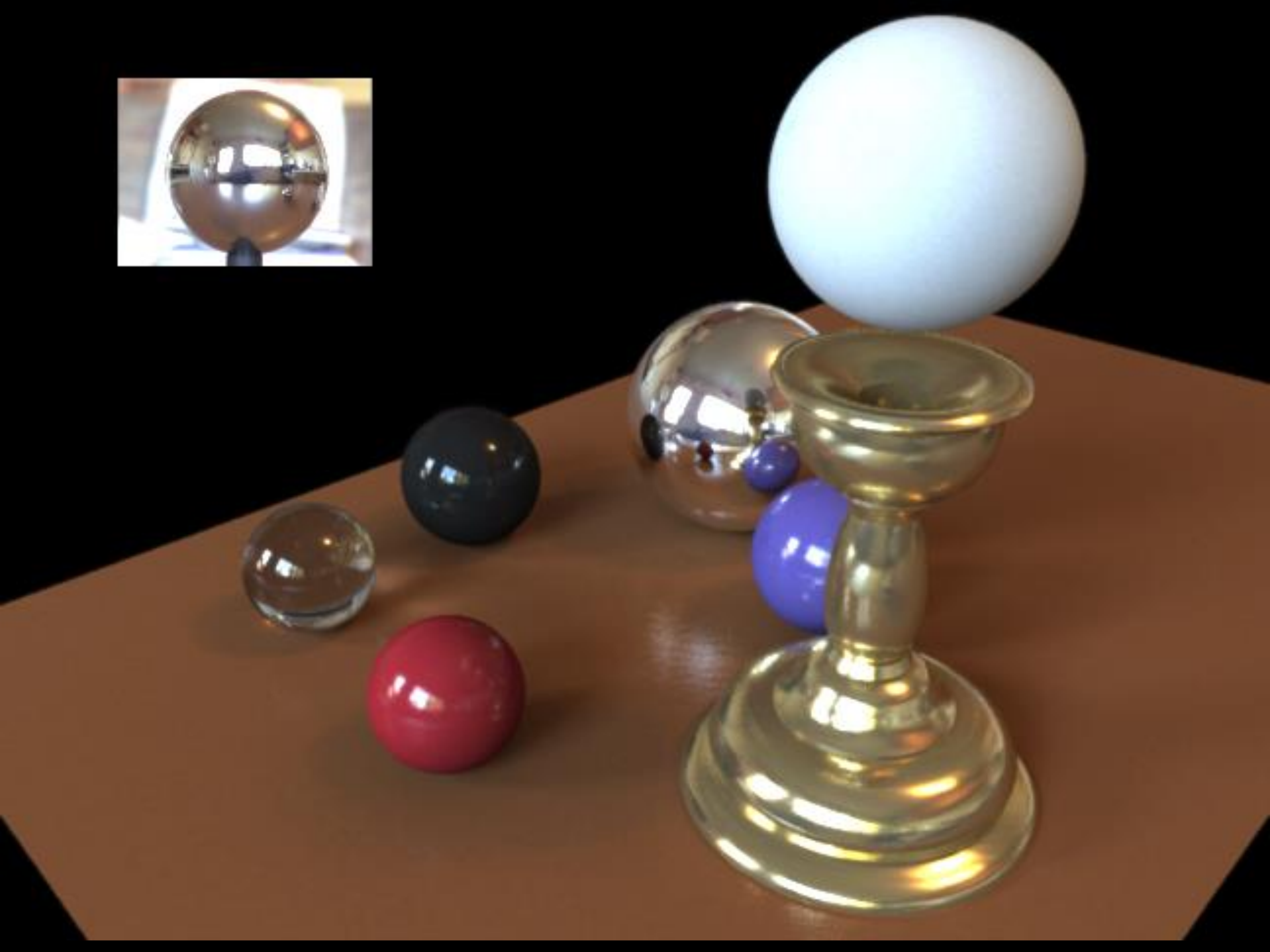
https://www.youtube.com/watch?v=F8Z3ubriTiY&ab_channel=PaulDebevec

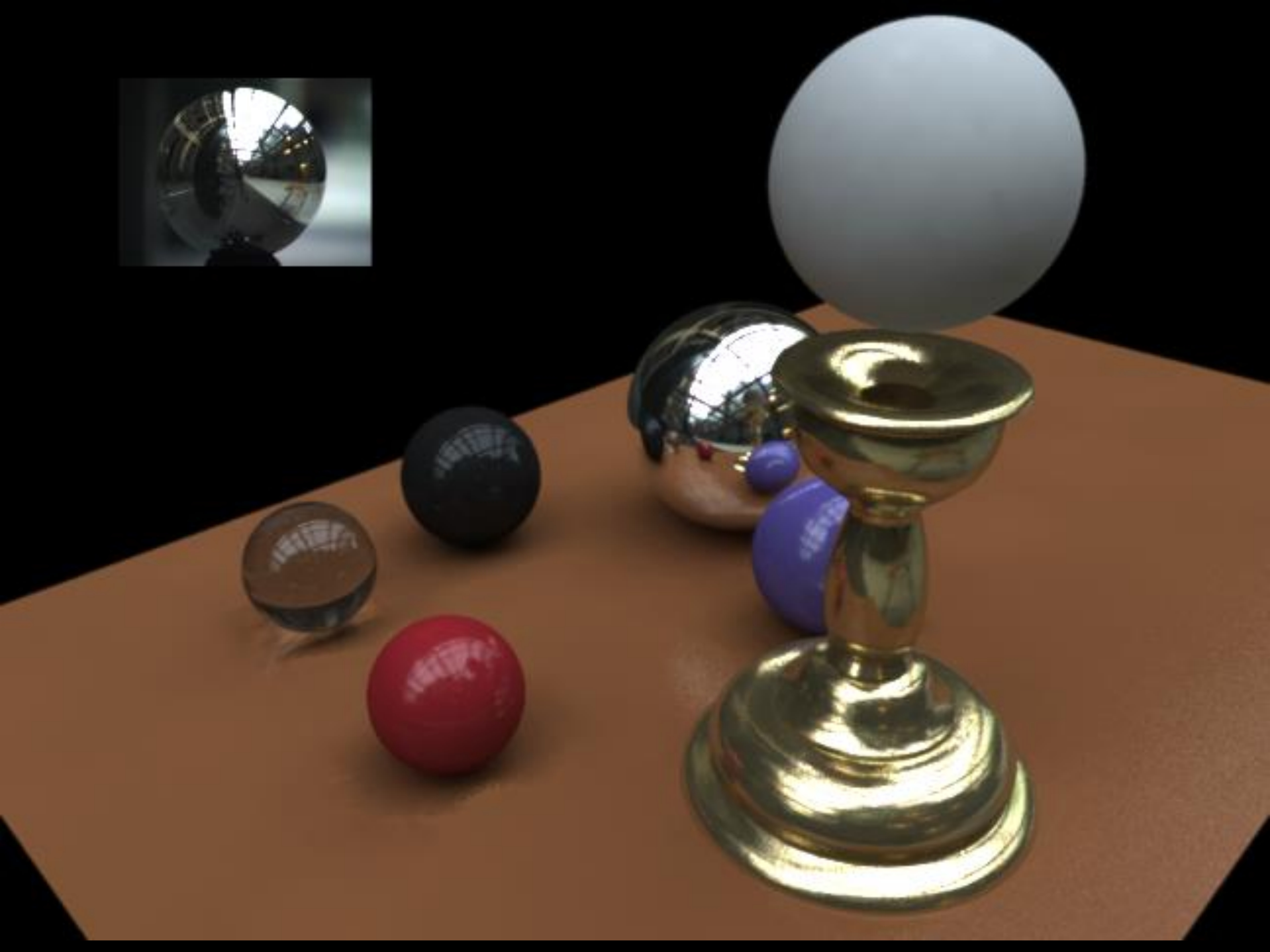
We can now illuminate
synthetic objects with real light.

How do we add synthetic objects to a
real scene?

It's not that hard!

http://www.nickbertke.com/



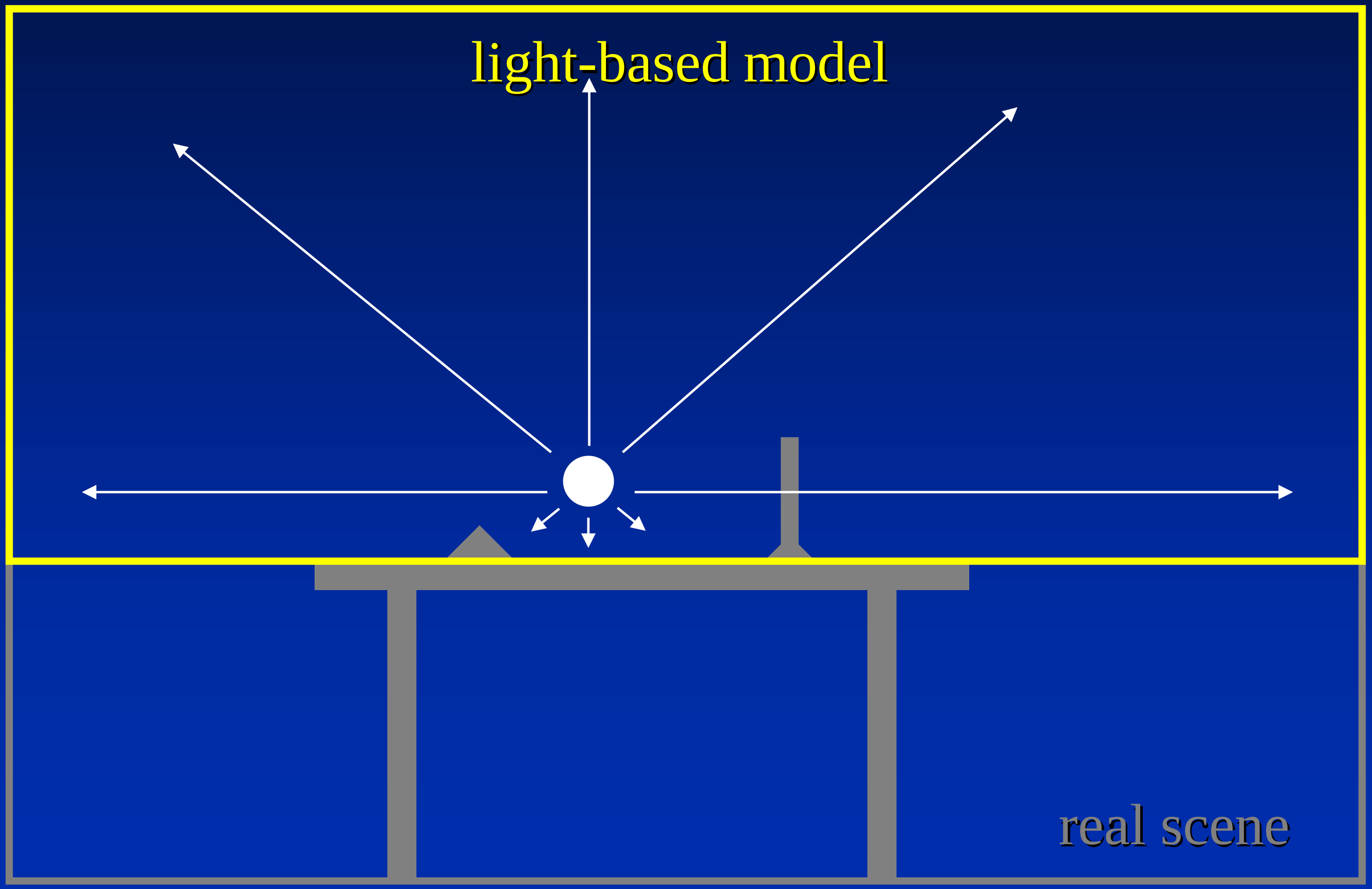


Real Scene Example

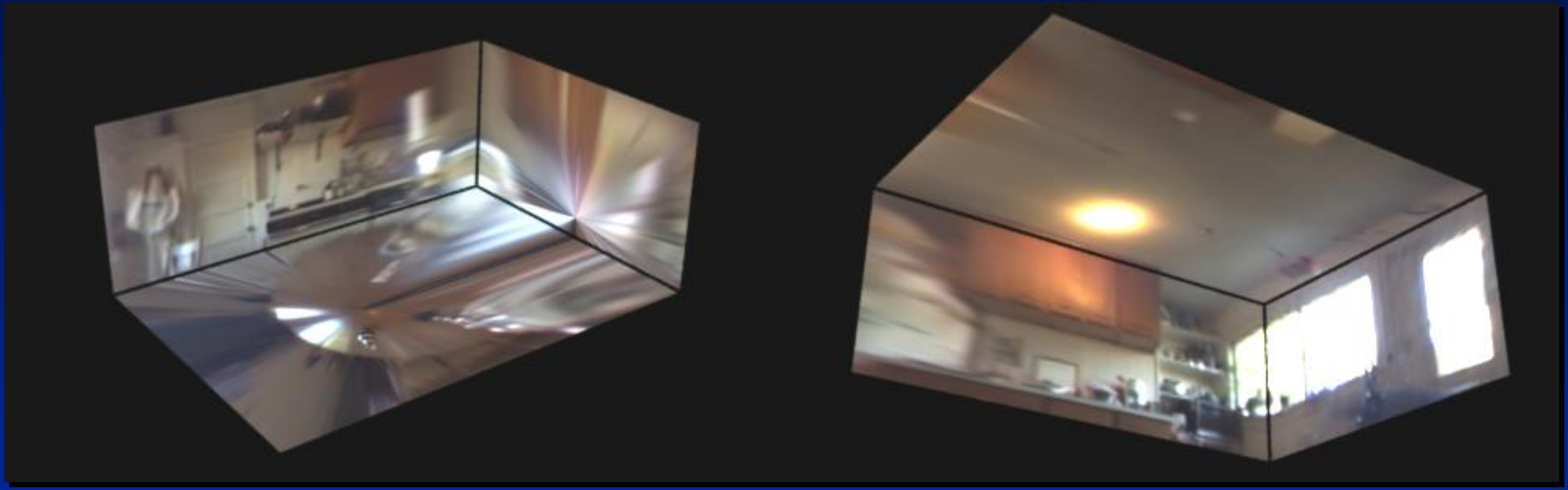
Goal: place synthetic objects on table

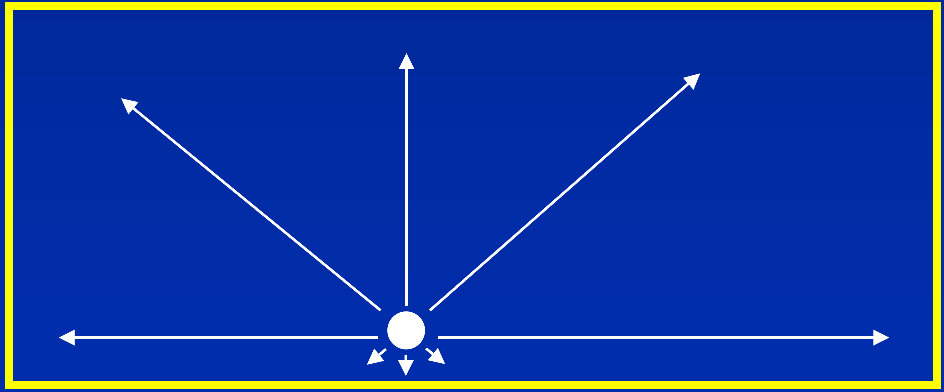
Light Probe / Calibration Grid

Modeling the Scene

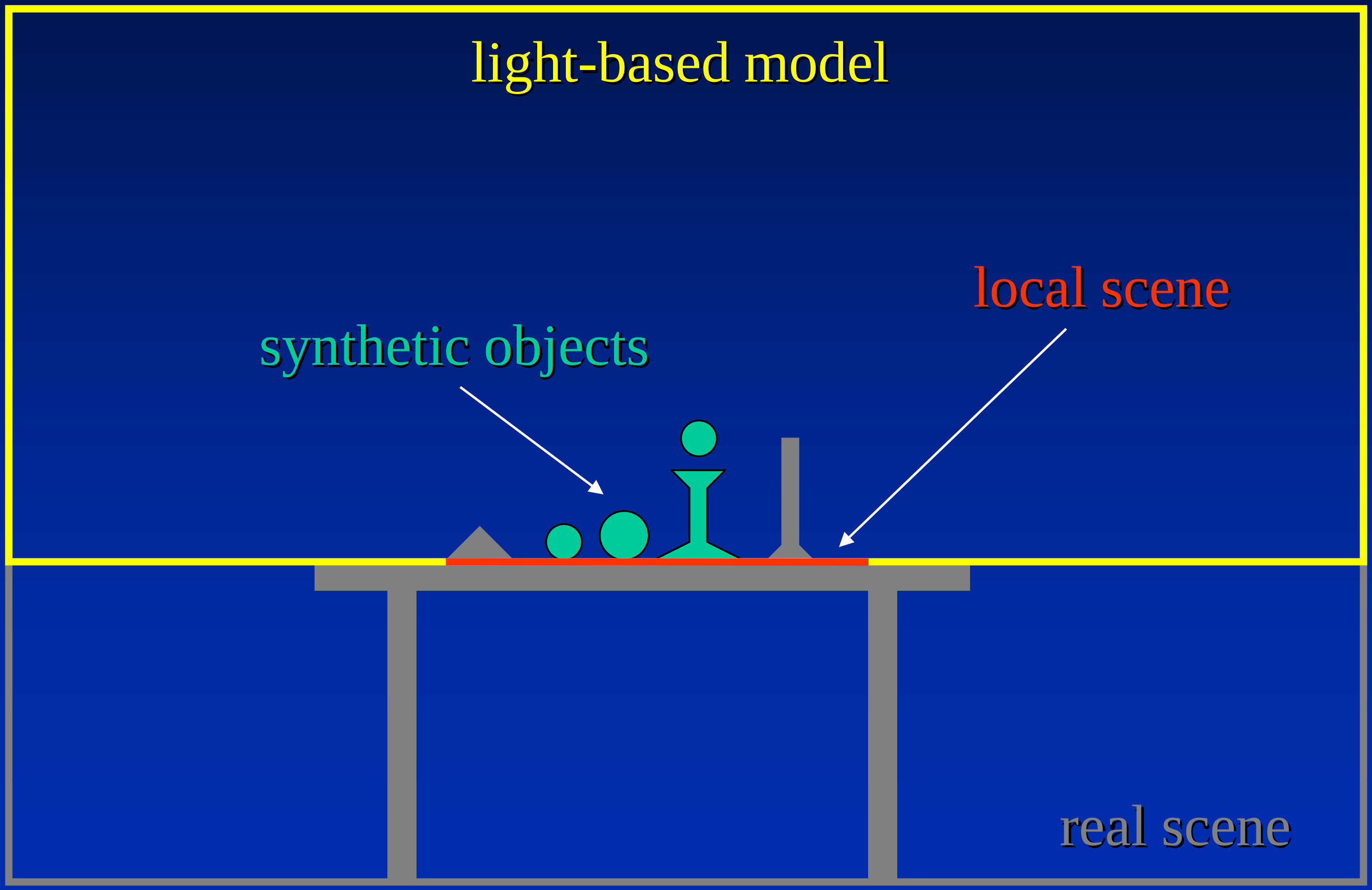


The *Light-Based* Room Model

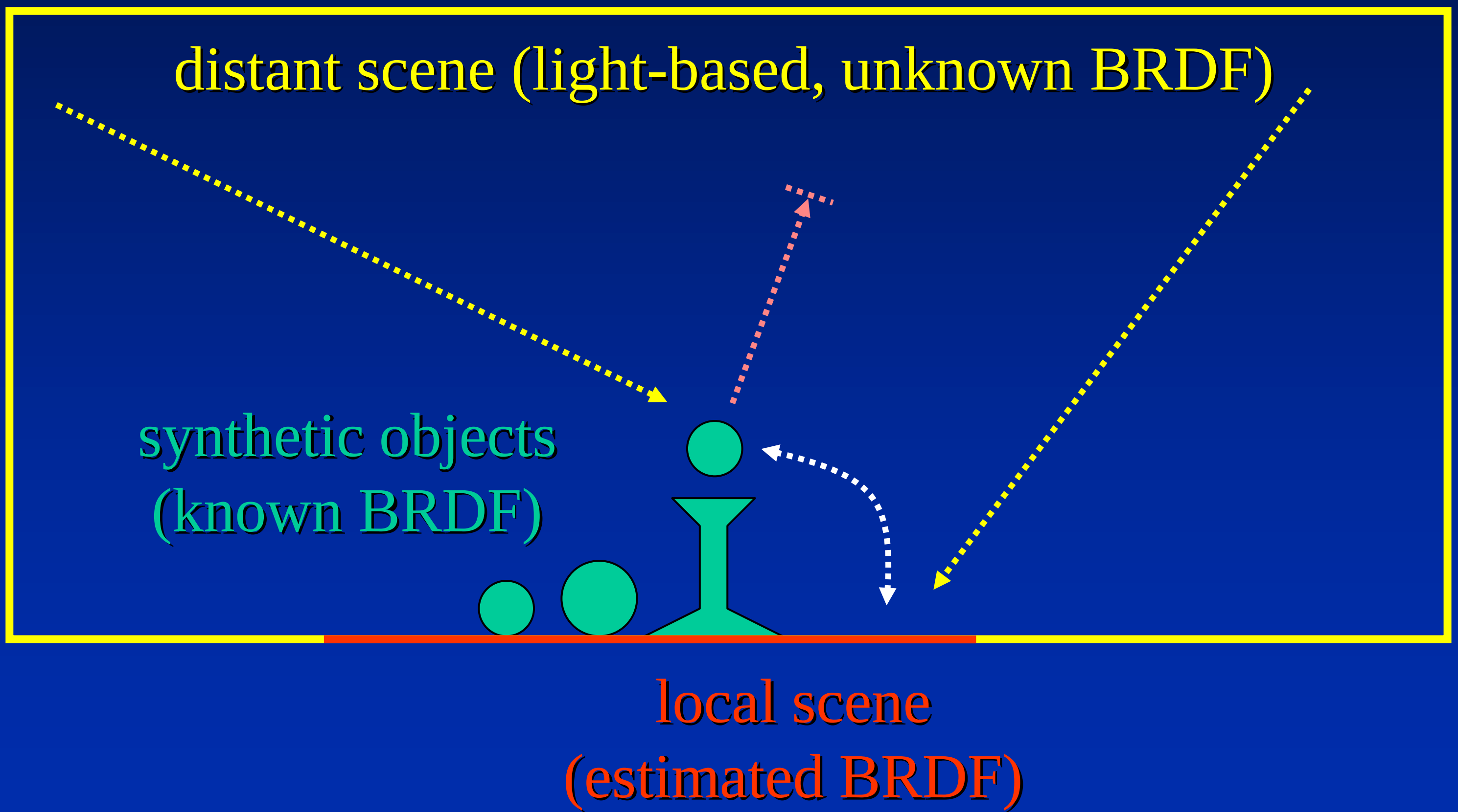




Modeling the Scene



The Lighting Computation



Rendering into the Scene

Background Plate

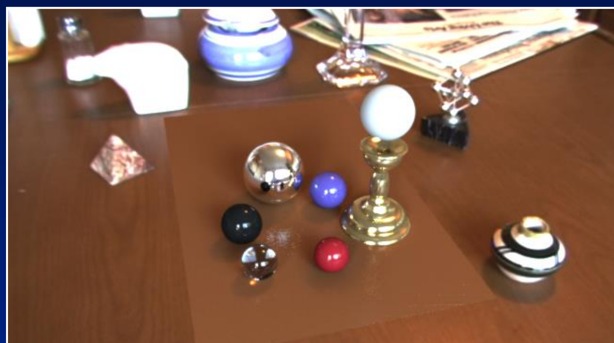
Rendering into the Scene

Objects and Local Scene matched to Scene

Differential Rendering

Local scene w/o objects, illuminated by model

Differential Rendering (2)
Difference in local scene



-

=