

# GLOBAL ILLUMINATION IN GAMES

**Jaroslav Křivánek**

Charles University in Prague





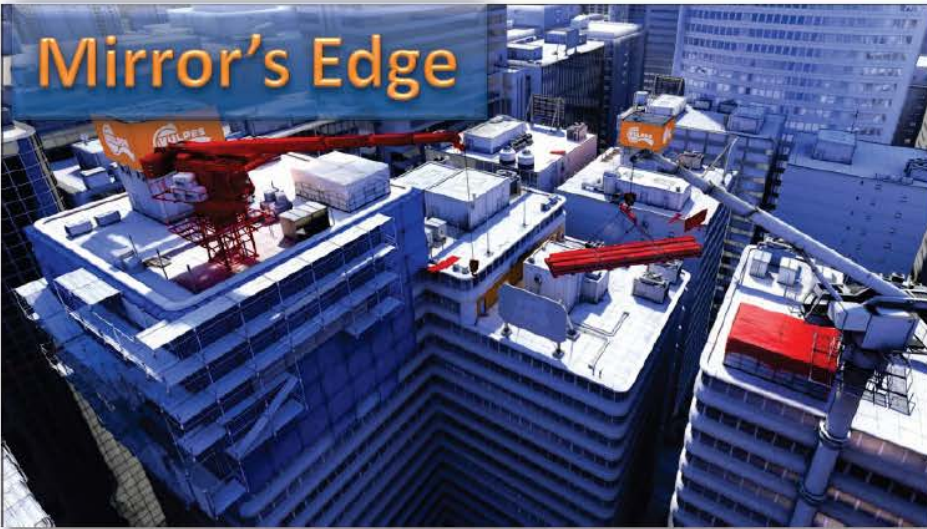
# Diffuse Global Illumination in Crysis 2™

# Diffuse Global Illumination in Crysis 2™



# Global illumination in games

Mirror's Edge



Halo 3



RAGE



Danger Planet



# Global illumination – Color bleeding



---

# **Pre-computed indirect lighting (baking, lightmaps)**

---

# Pre-computed indirect lighting (baking, lightmaps)

- Slides only
  - [http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-06-david\\_larsson-slides.pdf](http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-06-david_larsson-slides.pdf)
- Annotated slides
  - [http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-06-david\\_larsson-notes.pdf](http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-06-david_larsson-notes.pdf)
- **NOTE: The above slides have been used in the lecture and the material therein will be part of the exam!**

---

# **Details on radiosity normal mapping**

---

**(part of precomputed indirect lighting)**



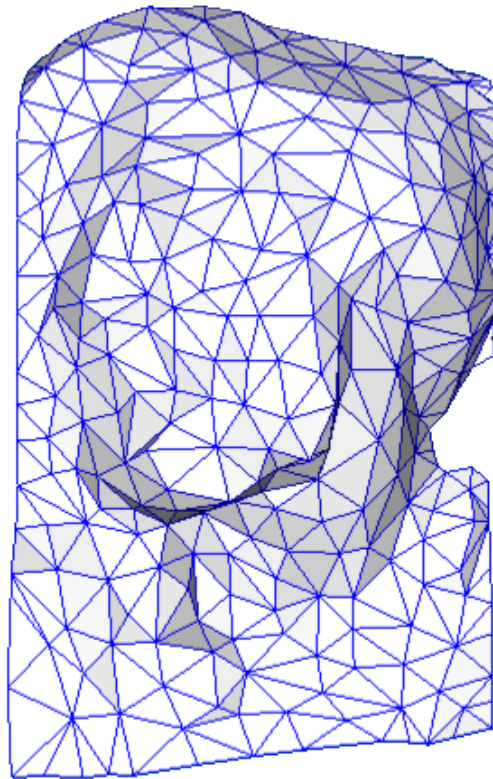
# Radiosity Normal Mapping

- Combine light mapping and normal maps
- Popularized by **Valve** for **Half-Life 2**
- See full slides by Habel & Wimmer: Efficient Irradiance Normal Mapping
  - <https://www.cg.tuwien.ac.at/research/publications/2010/Habel-2010-EIN/>

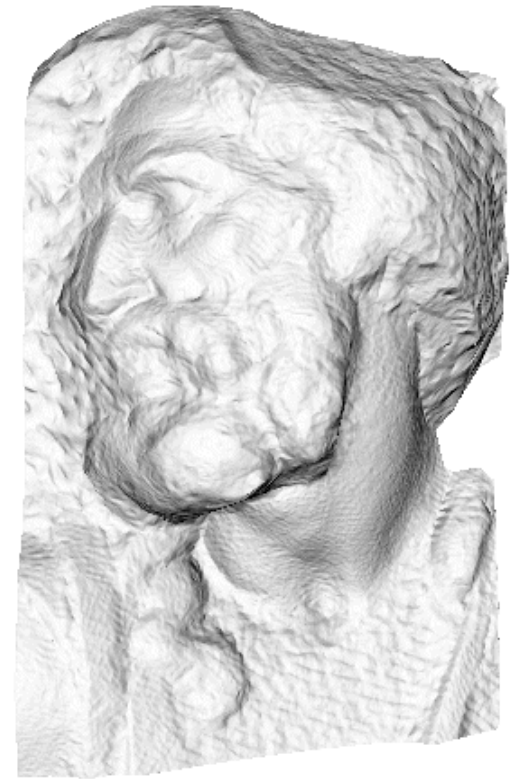
# Normal maps – Adding details to surfaces



original mesh  
4M triangles

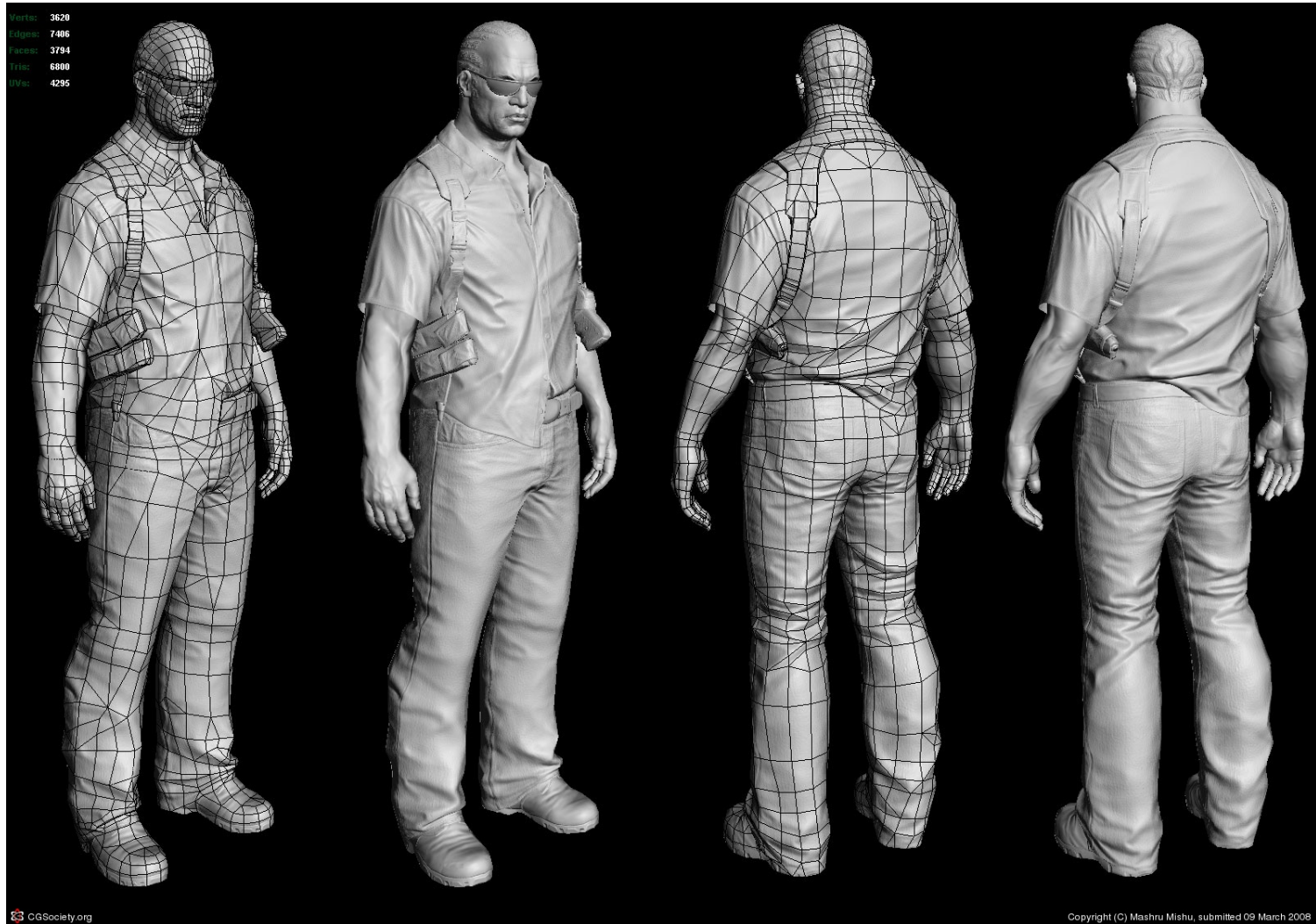


simplified mesh  
500 triangles

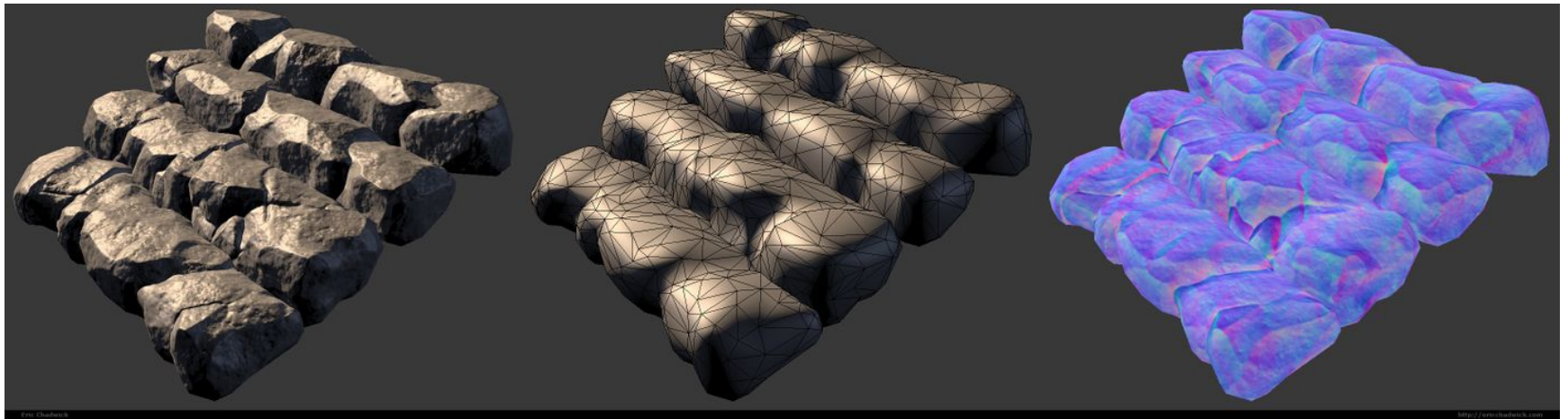


simplified mesh  
and normal mapping  
500 triangles

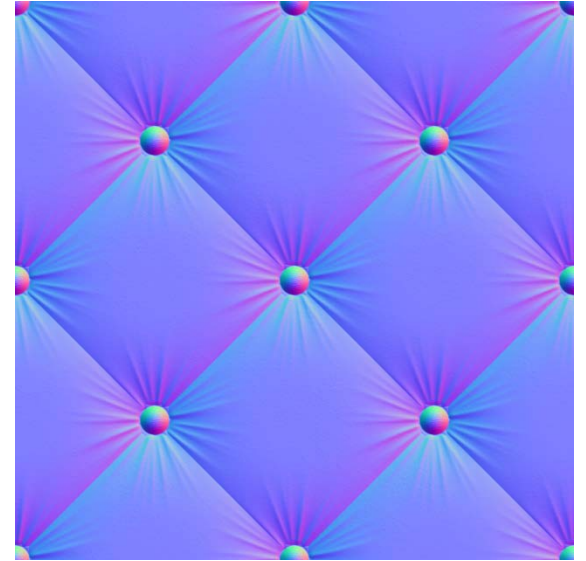
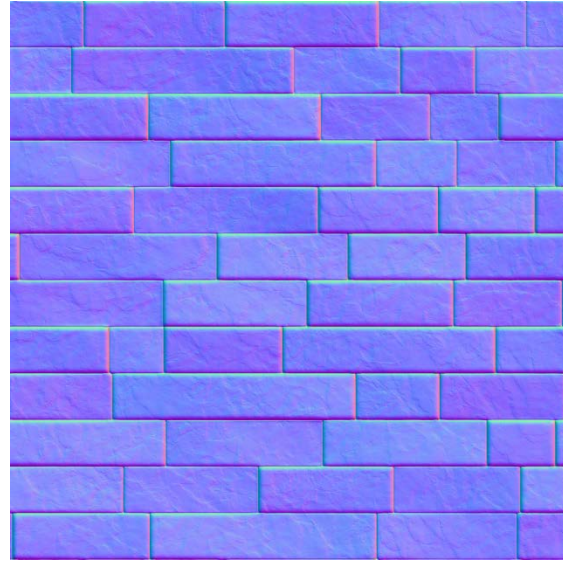
# Normal maps – Adding details to surfaces



# Normal maps – Adding details to surfaces



# Normal map examples



# Motivation for radiosity normal mapping



Light map only

# Motivation for radiosity normal mapping



Irradiance normal mapped

# Motivation for radiosity normal mapping



Irradiance normal mapped no texture



# Idea

- **Regular light map**

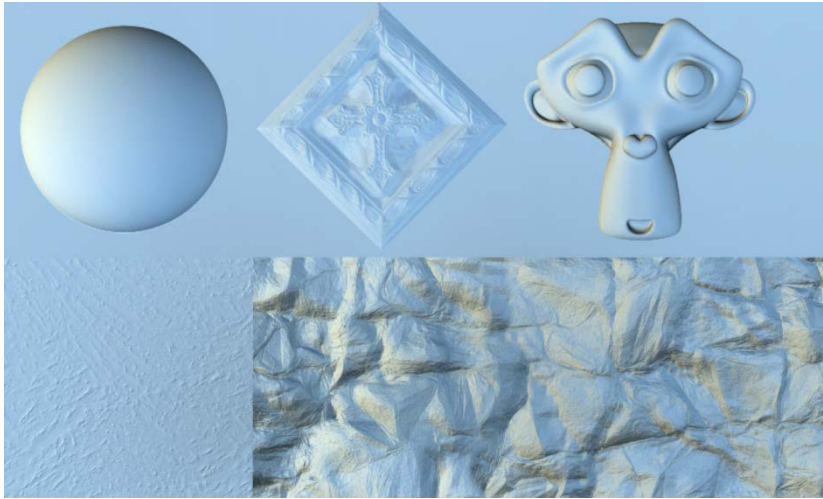
- One RGB triplet per texel

- **Radiosity light map**

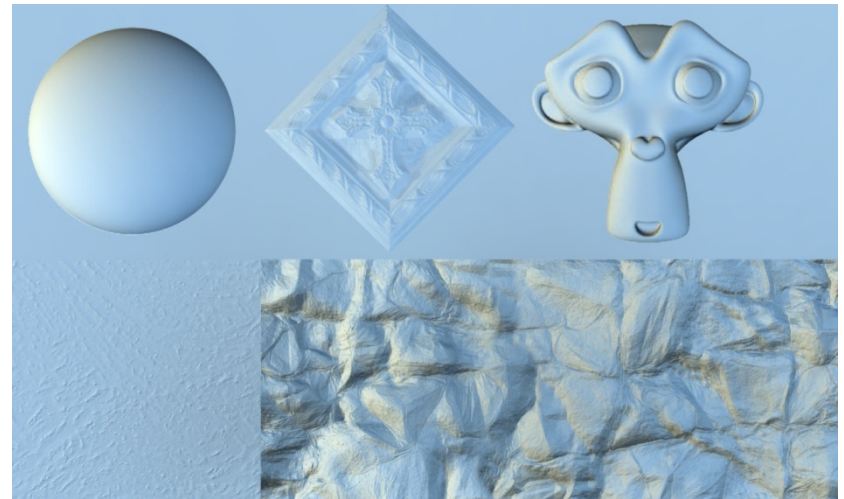
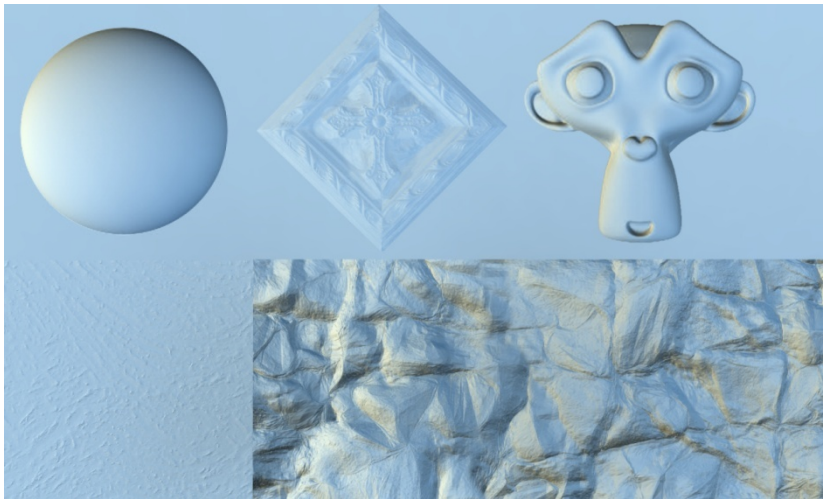
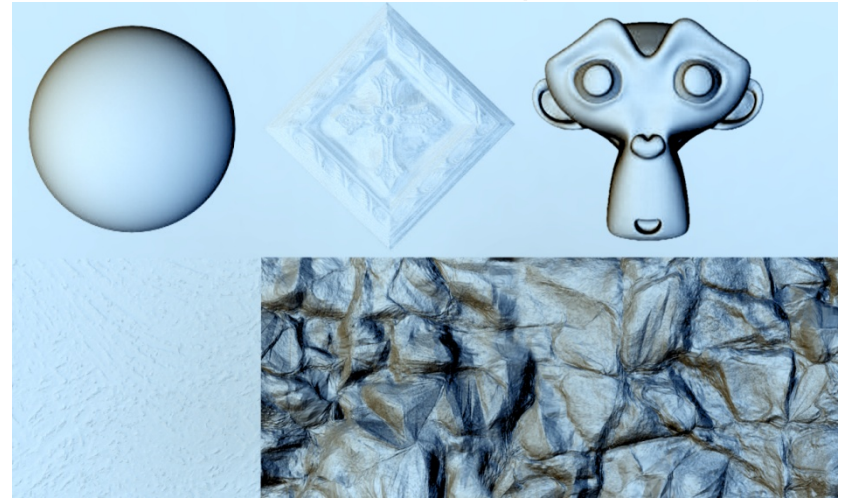
- Directional distribution of incident light for each texel
  - That is, essentially an irradiance EM for each texel
- Question: how to represent it efficiently
  - Irradiance EM: 9 SH coefficients (i.e. RGB triplets) per texel
  - Half-life 2 basis: 3 coefficients per texel
  - H-basis: 6 coefficients per texel

# Bases Comparison: 3/6/9 Coefficients

Ground truth



Half-Life 2 basis (3 coeffs)



$H_6$  (6 coefficients)

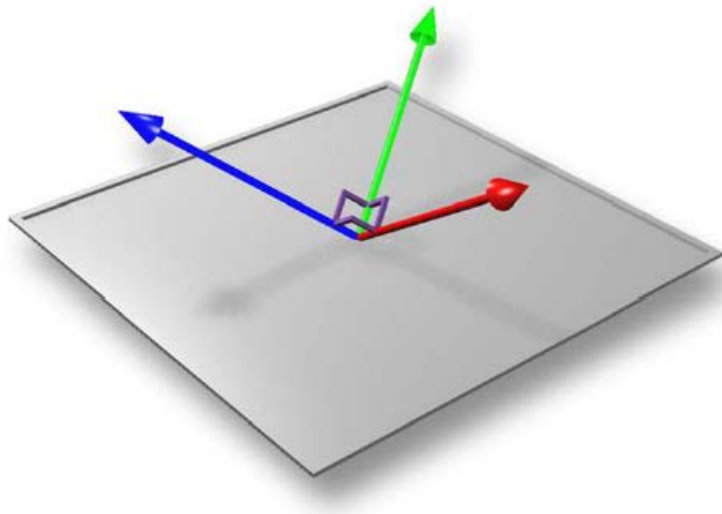
SH 3 bands (9 coefficients)

# Bases Comparison: 3/6/9 Coefficients

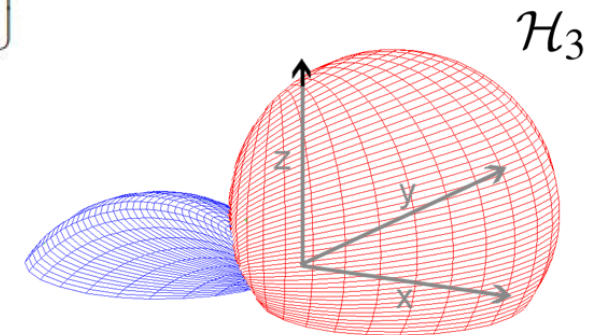
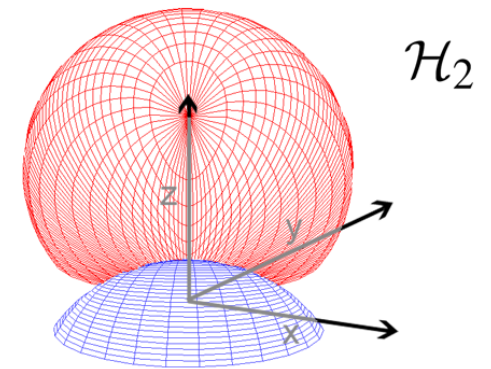
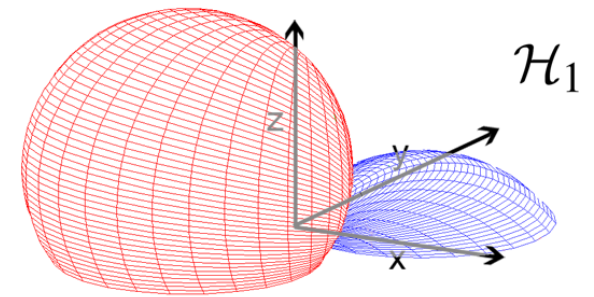
- SH irradiance environment maps (9 coeffs)
  - Extremely close to reference but costly and bulky
- Half-Life basis
  - Efficient and light, but a bad approximation of reference
- H6 is a good compromise

# Half-Life2 basis

- Cosine lobes around three orthogonal directions



$$\left\{ -\frac{1}{\sqrt{6}}, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{3}} \right\}$$
$$\left\{ -\frac{1}{\sqrt{6}}, -\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{3}} \right\}$$
$$\left\{ \sqrt{\frac{2}{3}}, 0, \frac{1}{\sqrt{3}} \right\}$$



# Radiosity Normal Mapping with Half-Life 2 basis

## ■ Precomputation

- ❑ Traditionally, when computing light map values using a radiosity preprocessor, a single color value is calculated
- ❑ In Radiosity Normal Mapping, we transform our basis into tangent space and compute light values for each vector.

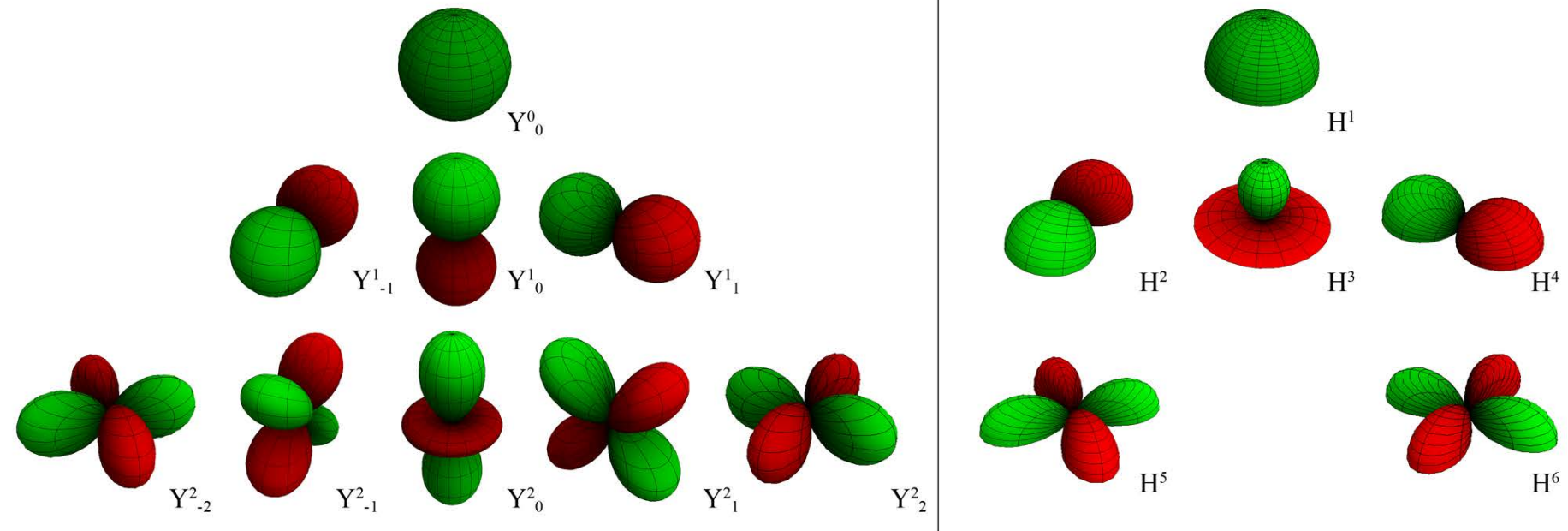
## ■ At the pixel level (Half-Life 2 basis)

- ❑ Transform the normal from a normal map into our basis
- ❑ Sample three light map colors, and blend between them based the transformed vector

```
lightmapColor[0] * dot( bumpBasis[0], normal ) +  
lightmapColor[1] * dot( bumpBasis[1], normal ) +  
lightmapColor[2] * dot( bumpBasis[2], normal )
```

# Radiosity Normal Mapping

- H-basis (Habel and Wimmer)
  - Spherical harmonics adapted to the hemisphere
  - <https://www.cg.tuwien.ac.at/research/publications/2010/Habel-2010-EIN/>



Spherical harmonics  
(used e.g. in Ramamoorthi 2001)

H6-basis  
(proposed by Habel)

---

# **Real-time dynamic GI: Point-based Global Illumination**

---

# Point-based global illumination

- Slides only

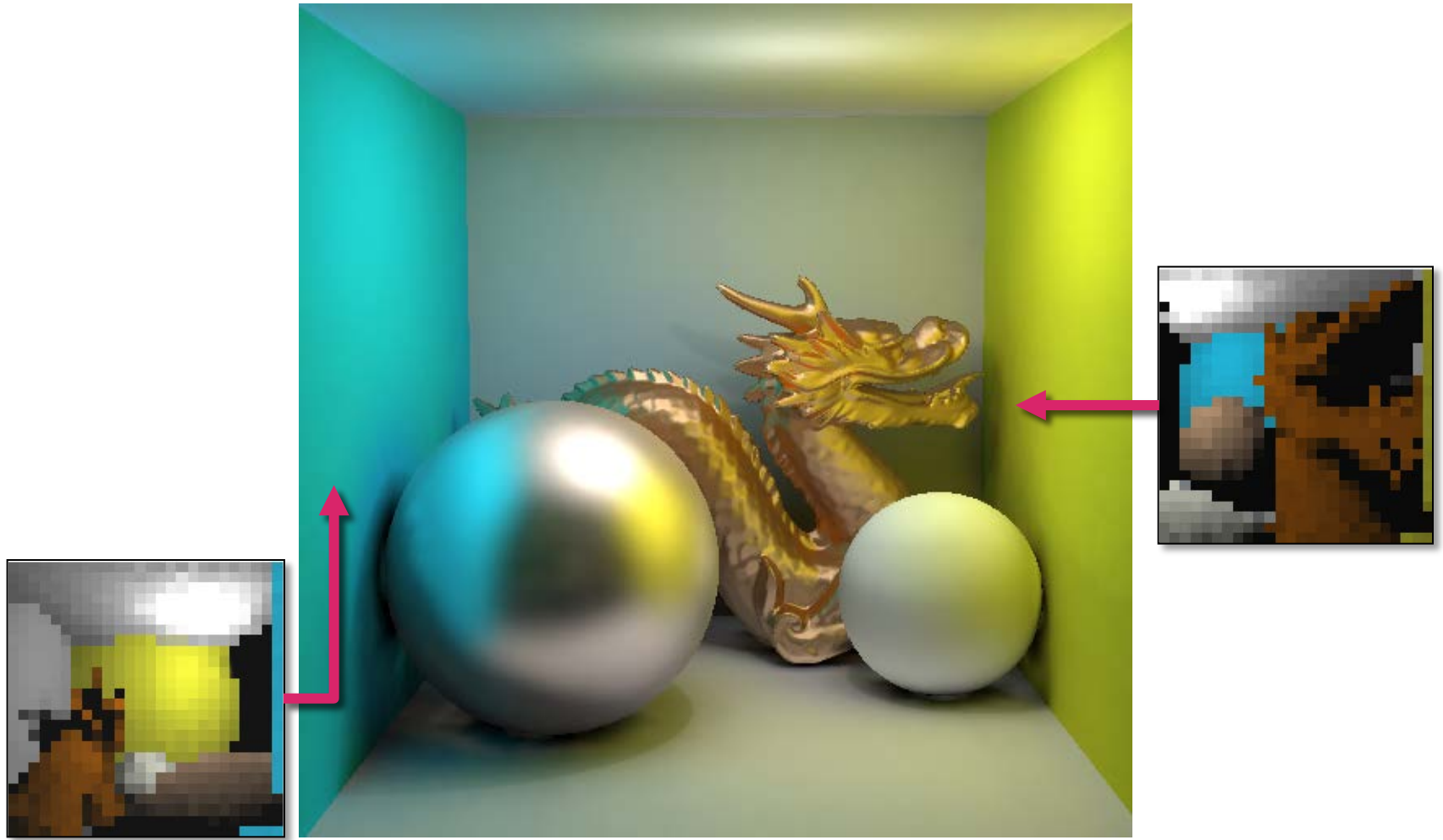
- [http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-05-michael\\_bunnell-slides.pdf](http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-05-michael_bunnell-slides.pdf)

- Annotated slides

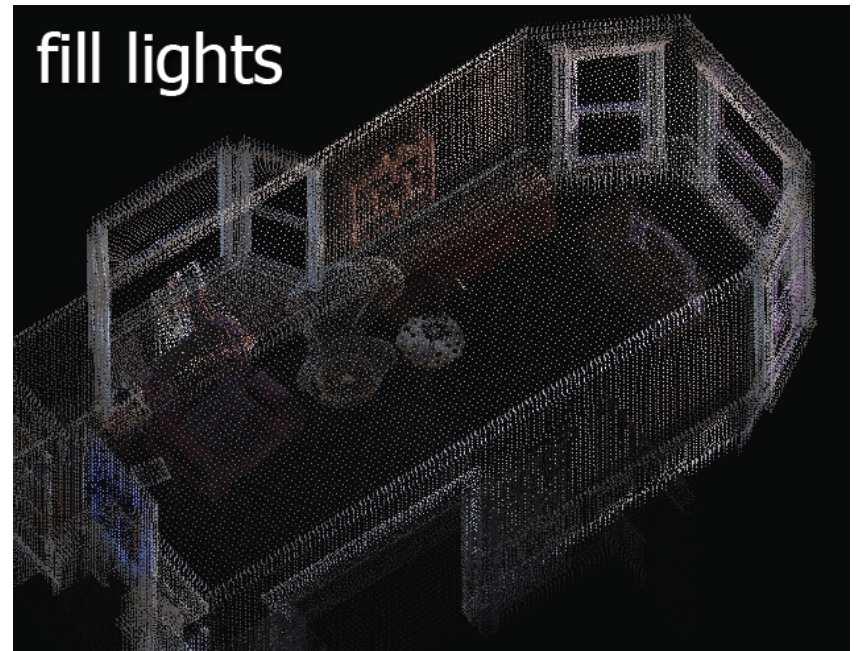
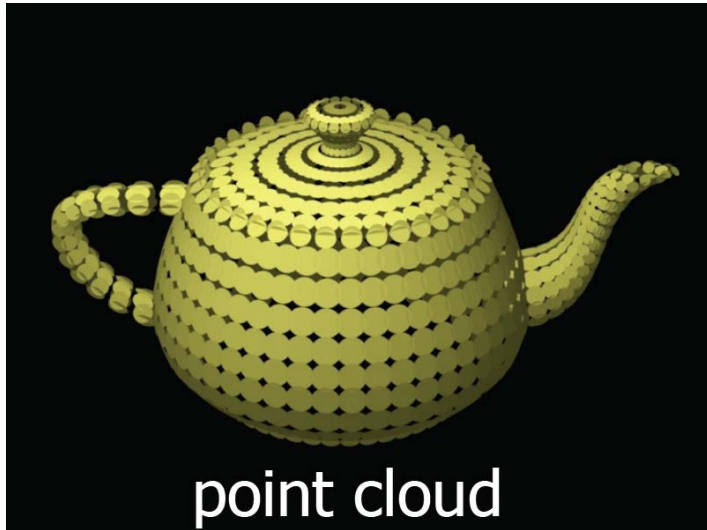
- [http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-05-michael\\_bunnell-notes.pdf](http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-05-michael_bunnell-notes.pdf)



# Point-based global illumination

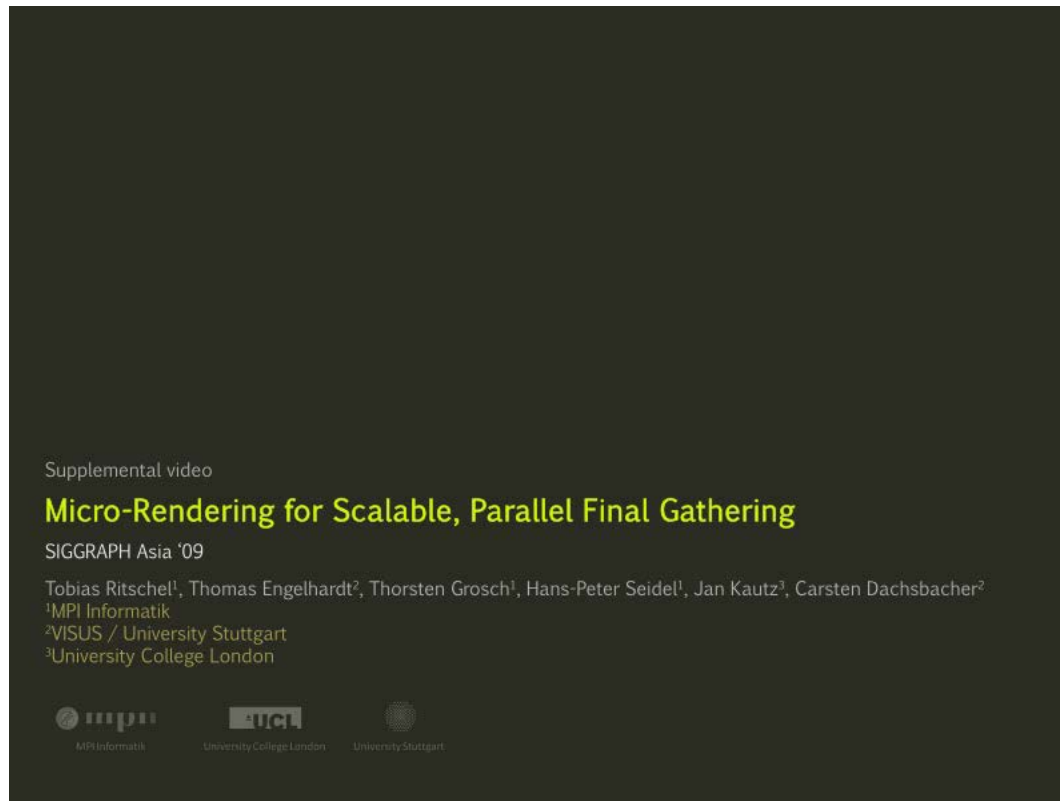


# Point-based global illumination



# Point-based global illumination

- Play micro-rendering video (<https://www.youtube.com/watch?v=Z9u8EdFbmiI>)



# Point-based global illumination

## Literature

### ■ Original idea

- M. Bunnell, “Dynamic ambient occlusion and indirect lighting”, GPU Gems 2
- See also: [http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-05-michael\\_bunnell-notes.pdf](http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-05-michael_bunnell-notes.pdf)

### ■ Application in movie production

- P. Christensen, “Point-based approximate color bleeding”, Pixar tech memo #08-01

### ■ Real-time implementation (CUDA)

- T. Ritschel et al, “Micro-rendering for scalable, parallel final gathering”, SIGGRAPH Asia 2009

---

# **Real-time dynamic GI: Light-propagation volumes**

---

- ☐ [Lighting the Environment](#)
  - + [Lighting Quick Start Guide](#)
  - + [Types of Lights](#)
    - [Shadow Casting](#)
  - + [Light Mobility](#)
  - + [Lightmass Global Illumination](#)
    - [Reflection Environment](#)
    - [Ambient Occlusion](#)
    - [Light Shafts](#)
    - [Light Functions](#)
    - [Ambient Cubemaps](#)
    - [Distance Field Ambient Occlusion](#)
    - [IES Light Profiles](#)
    - [Indirect Lighting Cache](#)
    - [Lit Translucency](#)
    - [Ray Traced Distance Field Soft](#)
    - [Shadows](#)
    - [Light Propagation Volumes](#)
    - [Bump Mapping w/o Tangent Space](#)
- + [Materials](#)
- + [Post Process Effects](#)
- + [Cascade Particle Systems](#)
- + [SpeedTree](#)
  - [Media Framework](#)
- + [Graphics Programming](#)
- + [User Interface \(UI\)](#)
- + [Persona: Skeletal Mesh Animation](#)
- + [Sequencer Editor](#)
- + [Audio and Sound](#)
- + [AI and Behavior Trees](#)
- + [Paper 2D](#)
- + [Online & Networking](#)
- + [Physics Simulation](#)
- + [Landscape Outdoor Terrain](#)
- + [Matinee and Cinematics](#)
- Sequencer
  - + [Packaging and Cooking Games](#)
  - + [Advanced Features](#)

English ▾

+ Lighting the Environment

 SHARE: [f](#) [t](#) [r](#) [in](#) [G+](#)

# Light Propagation Volumes

Advanced Unreal Engine 4.9
*Light Propagation Volumes are a feature in development and not ready for production.*

### On this page:

- [Enabling Light Propagation Volumes](#)
- [Basic Scene Setup](#)
- [Light Propagation Volume Settings](#)
  - [Adjusting the Look and Performance](#)
- [Directional Light Settings](#)
- [Viewing Global Illumination](#)
- [Visualizing Light Propagation Volumes](#)
- [GI Replace Material Switch](#)
- [Other Notes](#)



# Light propagation volumes

- Anton Kaplanyan's slides form the 2010 course "Global illumination across industries"
  - [http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-07-anton\\_kaplanyan.pptx](http://cgg.mff.cuni.cz/~jaroslav/gicourse2010/giai2010-07-anton_kaplanyan.pptx)

# THANK YOU!

## Questions?



Computer  
Graphics  
Charles  
University

[\[cgg.mff.cuni.cz/~jaroslav\]](http://cgg.mff.cuni.cz/~jaroslav)