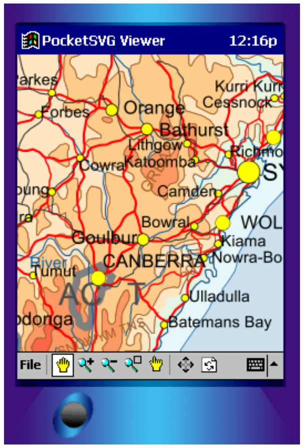
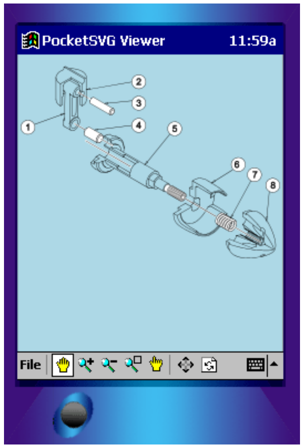
IMGD 4000: Graphics Trends in Games

by Emmanuel Agu

Professor Background

- Dr. Emmanuel Agu (professor, “Emmanuel”)
- Research areas
  - Computer Graphics (GPU rendering, mobile graphics, etc)
  - wireless networking and mobile computing
- Advise MQPs, MS and PhD theses



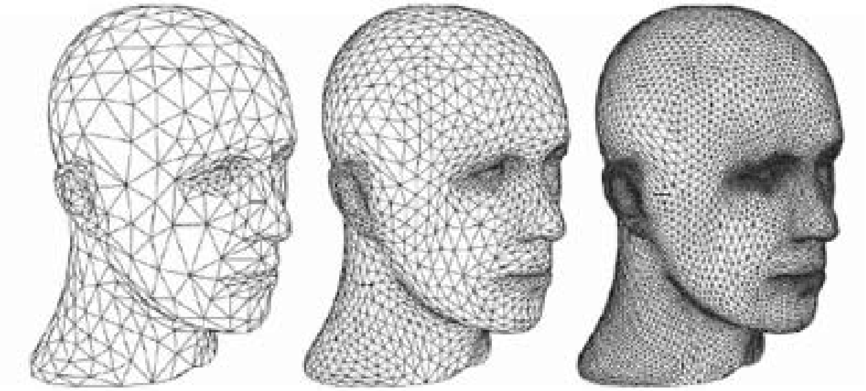


Graphics Trends for Games

1. Real Time LoD Management
2. Capture rendering data
3. Pre-computation to speed up run-time
4. Ray tracing
5. Real Time Global Illumination
6. Hardware-accelerated physics engines

Trend 1: Real-Time LoD Management

- Geometry shader unit, can generate new vertices, primitives from original set
- Tesselation and simplification algorithms on GPU
- Real-time change LoD in game



[Zorin and Schröder 2000]

Trend 2: Capture Rendering Data

- Old way: use equations to model:
  - Object geometry, lighting (Phong), animation, etc
- New way: capture parameters from real world
- Example: motion in most sports games (e.g. NBA 2K live) is captured.
  - How? Put sensors on actors
  - Actors play game
  - Capture their motion into database
  - Player motion plays back

  database entries

Courtesy: Madden NFL game

How is capture done?

- **Capture:**
  - Digitize real object geometry and attributes
  - Use cameras, computer vision techniques to capture rendering data
  - Place data in database, many people can re-use
- **Question:** What is computer vision?

Geometry Capture: 3D Scanning

- **Capturing geometry trend:** Projects on precise 3D scanning (Stanford, IBM,etc) produce very large polygonal models
- Some models too large to be loaded by most machines

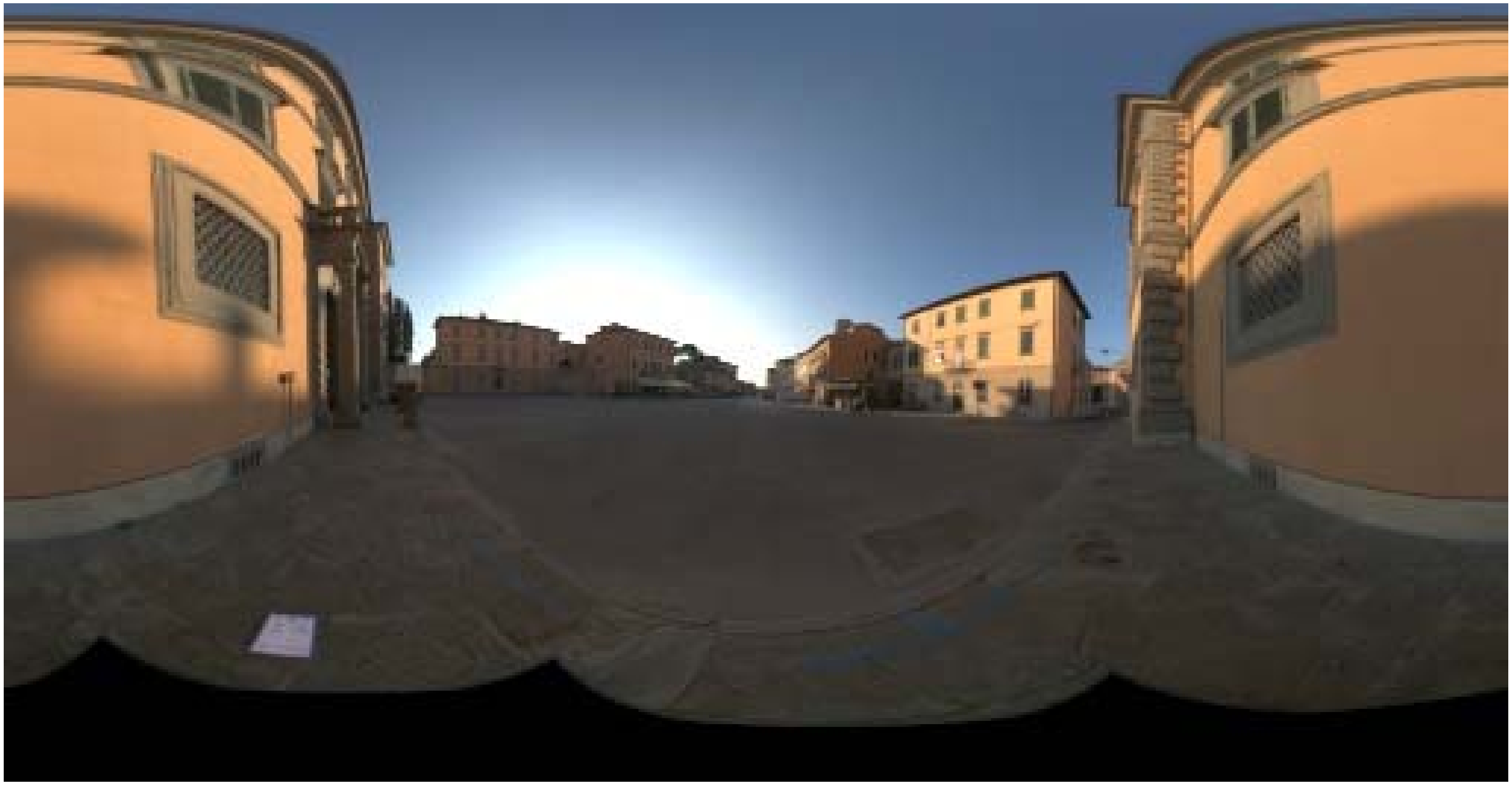
Courtesy: Stanford Michael Angelo 3D scanning project

Model: David

Largest dataset Size: 2 billion polygons, 7000 color images!!

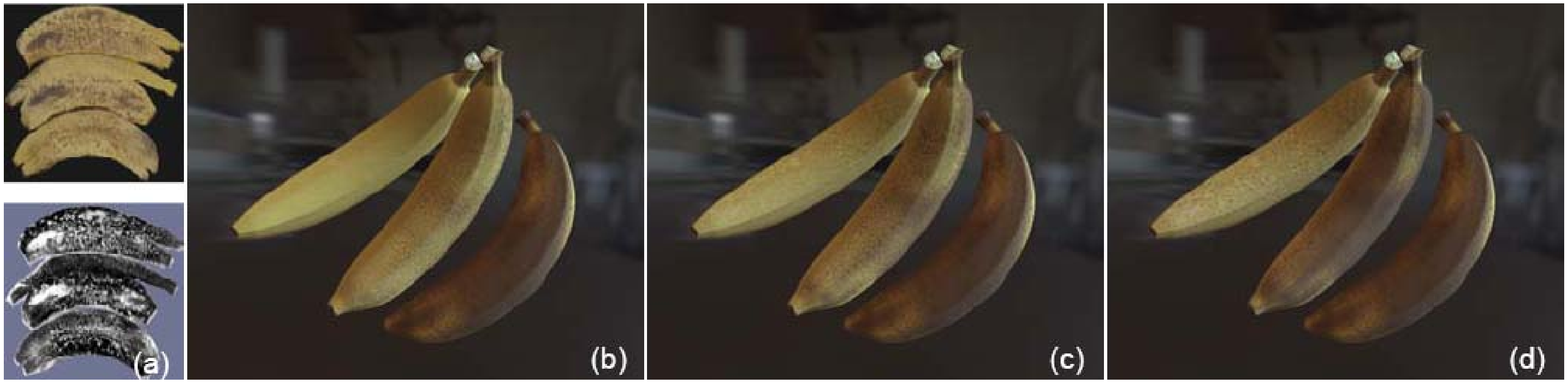
Light Probes: Capturing light

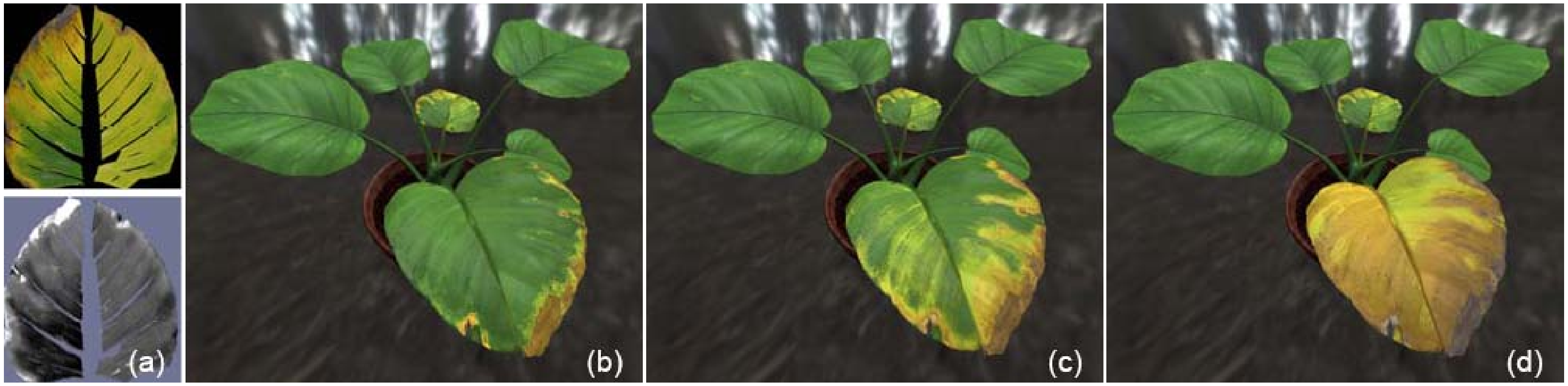
Amazing graphics, High Dynamic Range?



Capture Material Reflectance (BRDF)

- BRDF: How different materials reflect light
- Examples: cloth, wood, velvet, etc
- Time varying?: how reflectance changes over time
- Examples: weathering, ripening fruits, rust, etc

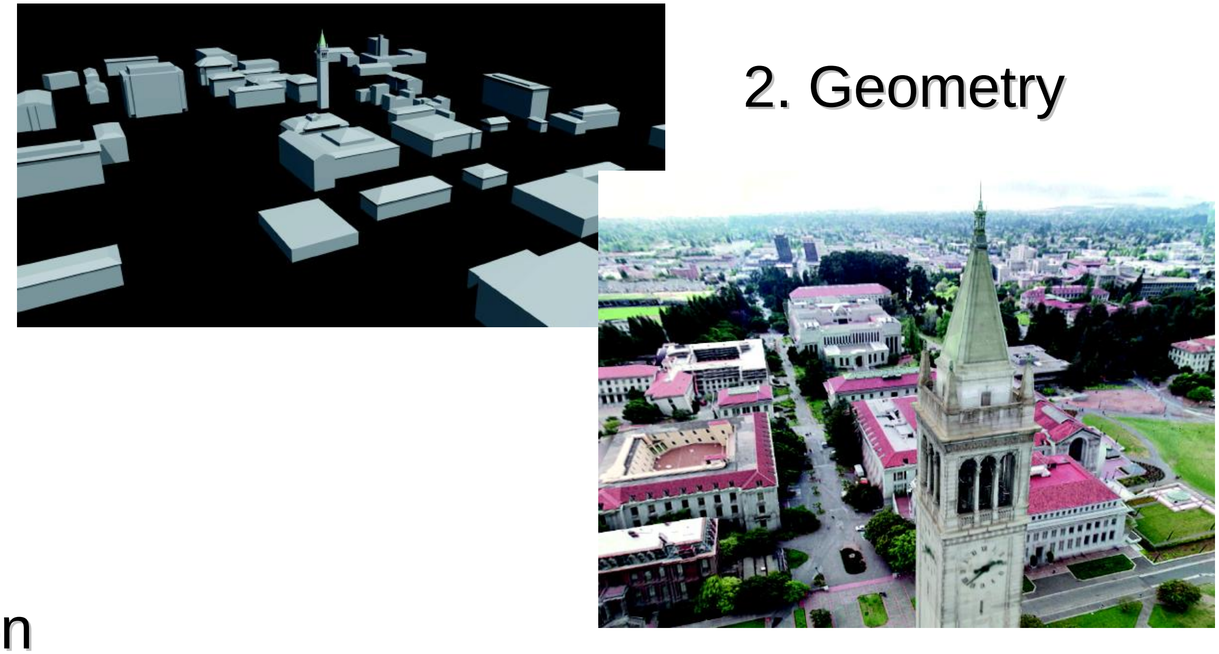




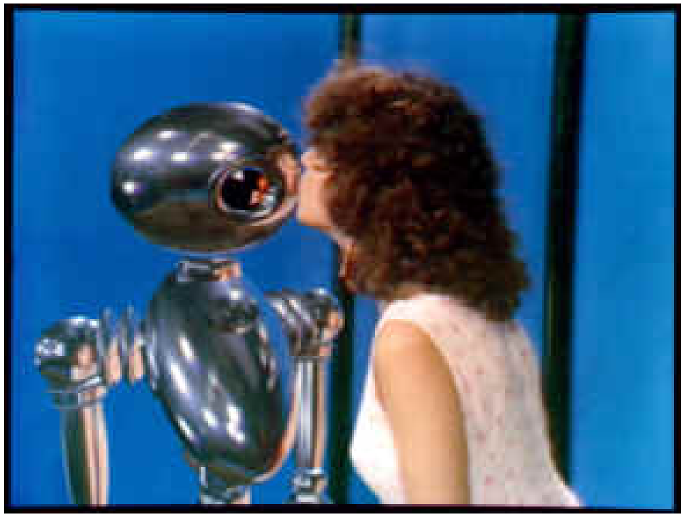
Exactly What Can We Capture?

1. Appearance (volume, scattering, transparency, translucency, etc)

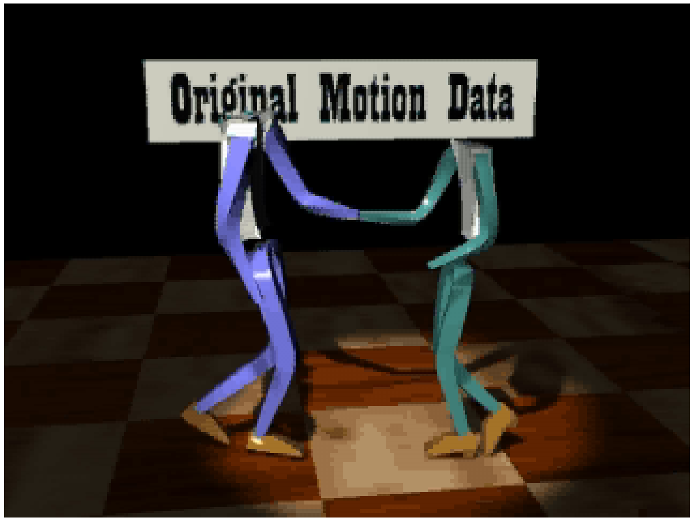
2. Geometry



3. Reflectance & Illumination



4. Motion



Why effort to capture?

- **Big question:** If we can capture real world parameters, is this really **computer graphics?**

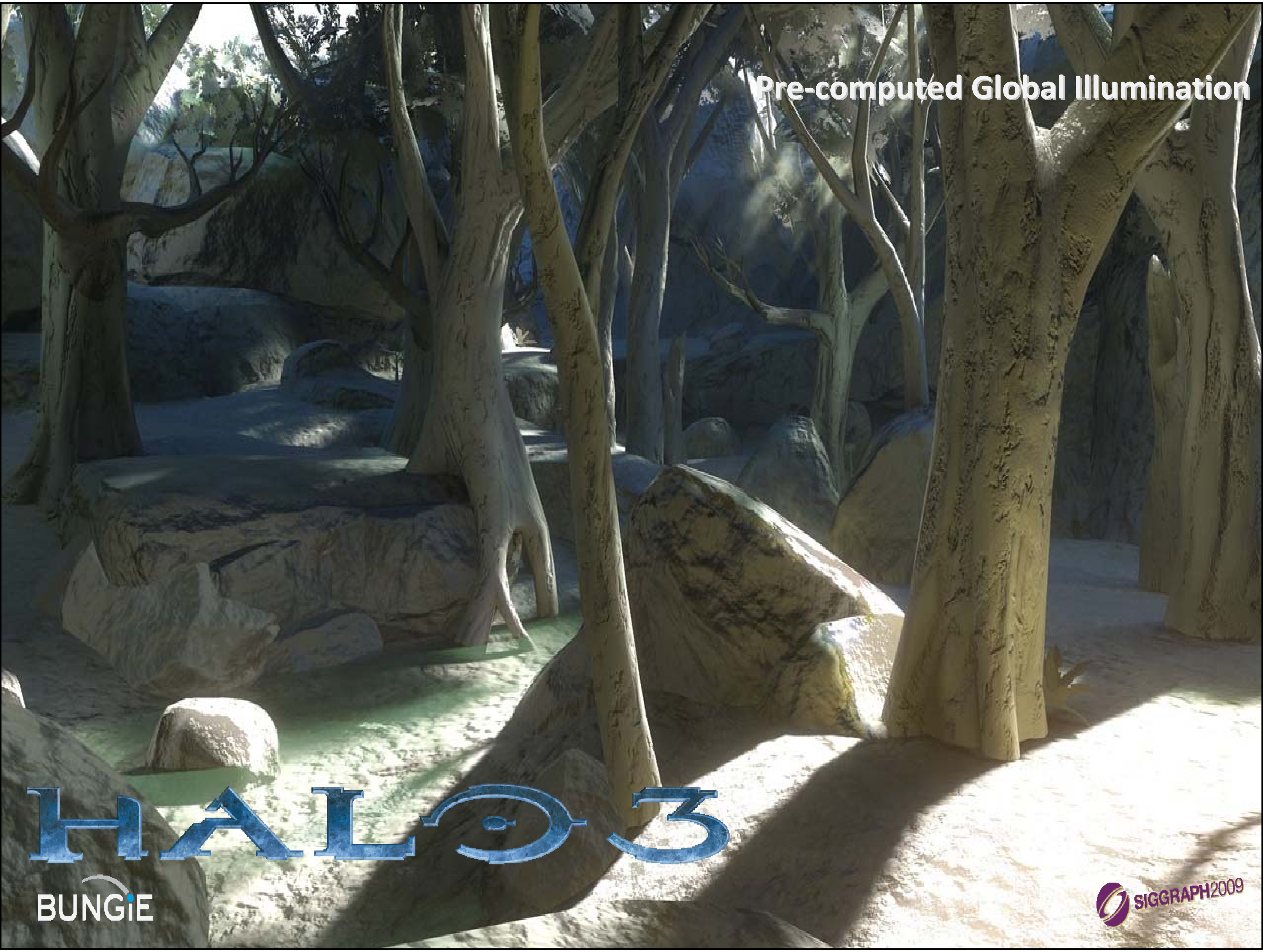
Trend 3: Pre-computation to speed up run-time

- Pre-computer lighting:
- Pre-computed Occlusion
- Pre-computed Radiance Transfer
  - Use spherical harmonics

Pre-Computing Lighting

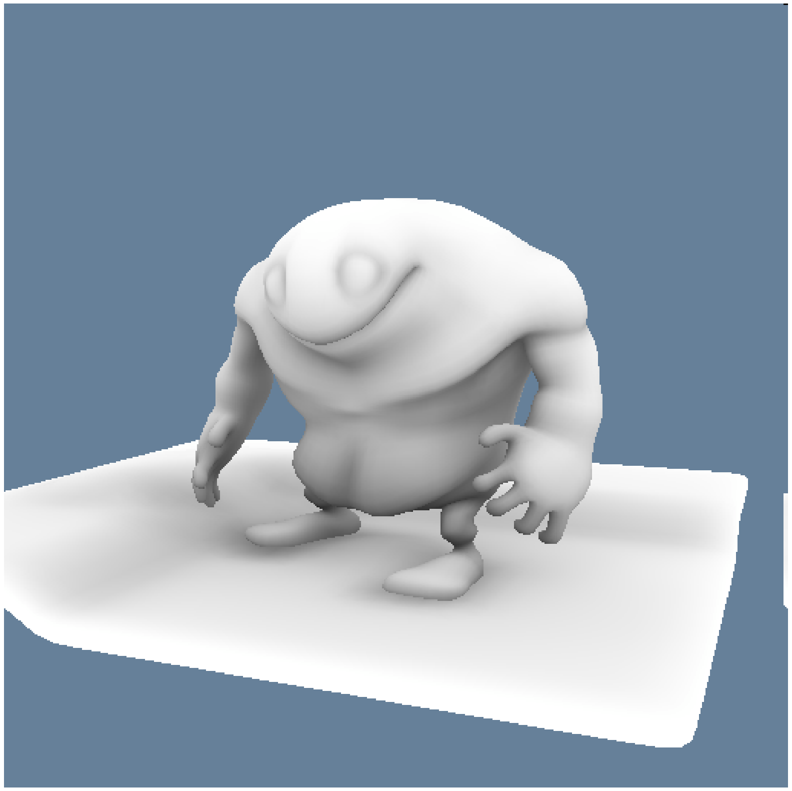
- Few years ago, pre-computing lighting was state-of the art
- Use GPU to pre-compute approximate lighting solutions
- Speeds up run-time
- What did it look like?

Pre-computed Global Illumination



Other Ideas

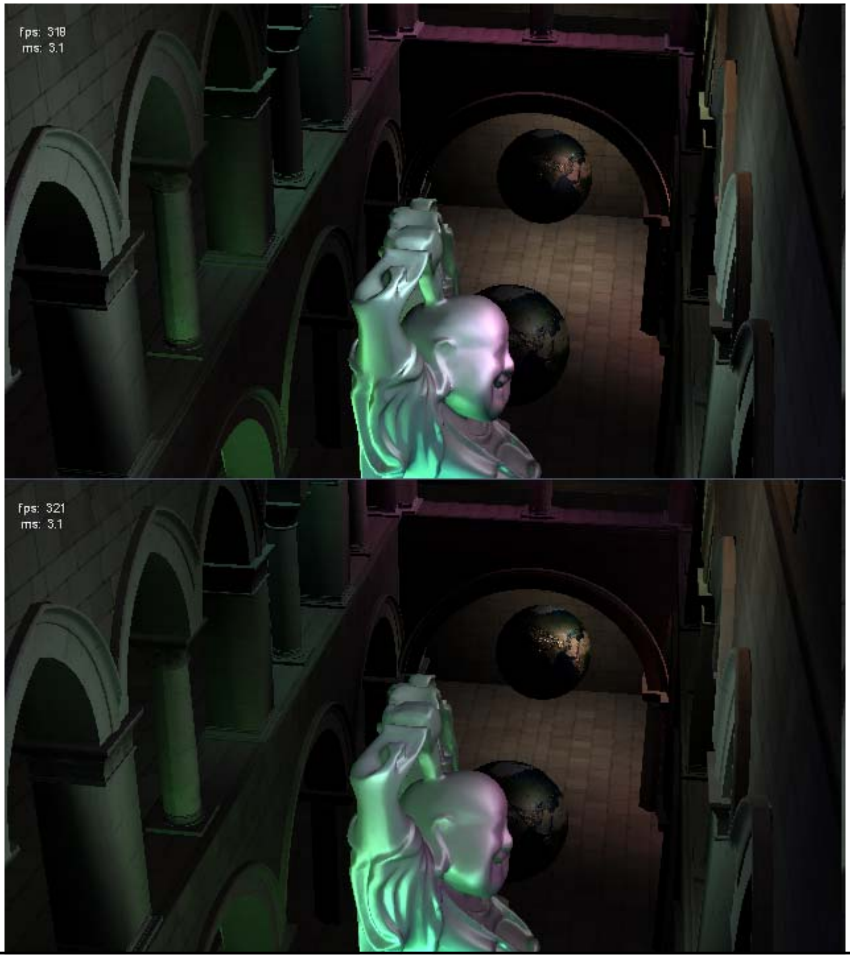
- Precompute lighting
  - If scene and light sources are static
- Precompute occlusion
  - Example: precomputed ambient occlusion



Courtesy Nvidia SDK 10

PRT

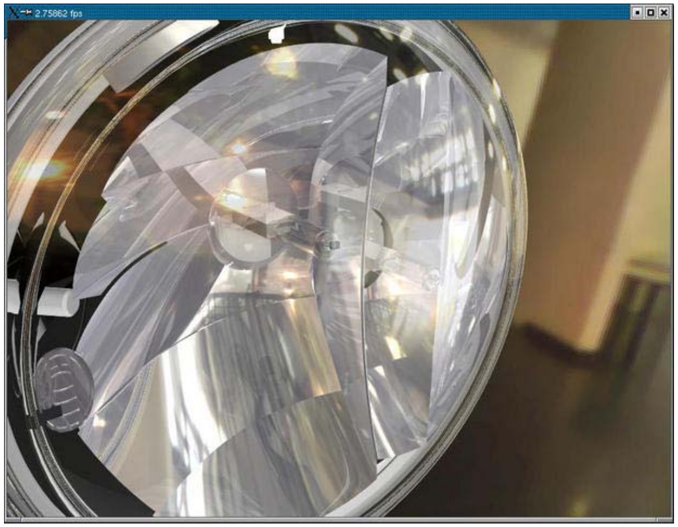
- Precomputed Radiance Transfer (PRT)
- Factorize and precompute light and material into Spherical Harmonics
- Light reflection is dot product at run time (Fast)

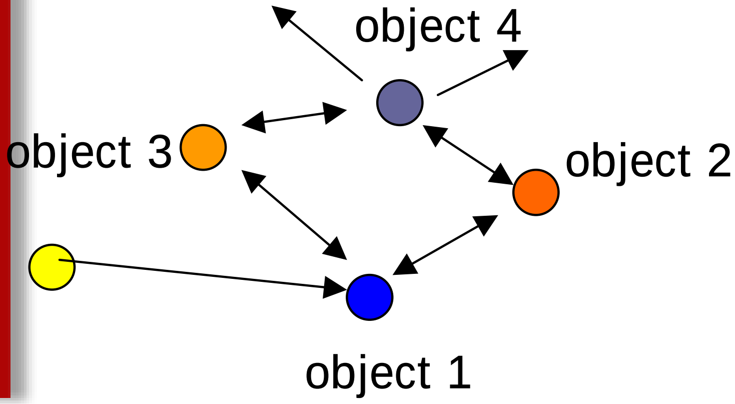


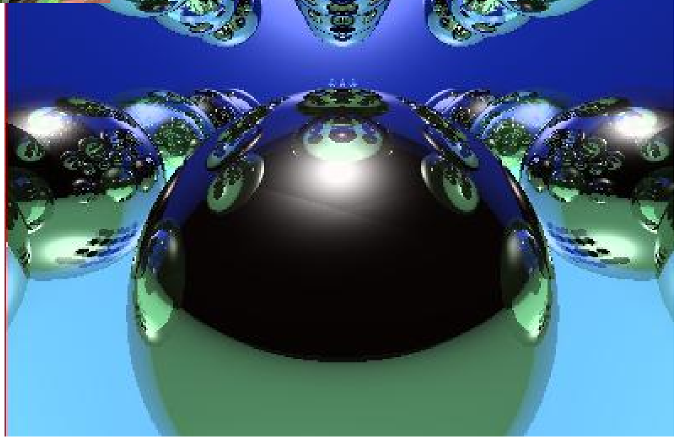
Courtesy Sponza Atrium
by Marko Dabrovic:

Trend 4: Raytracing in Games

- OpenGL models direct lighting, approximates GI
- Raytracing: A global illumination rendering
  - Reflection / refraction / shadow







Ray Tracing in Games

- Nvidia Optix ray tracer
- Needs high end Nvidia graphics card
- SDK is available on their website
- http://developer.nvidia.com/object/optix-home.html

Trend 5: Real time Global Illumination

- What's the difference?
  - Pre-compute means lookup at run-time
  - Approximate representations (e.g Spherical Harmonics)
  - Fast, but not always accurate
- Real Time Global Illumination: now state-of-the art
  - Calculate GI equations at run-time
  - Sometimes use GPU, hardware

Real Time Global Illumination

- Ray tracing enables global illumination
- Instead of billboards, imposters, images use physically-based appearance models
- Shadows
- Ambient Occlusion
- Reflections
- Transmittance
- Refractions
- Caustics
- Global subsurface scattering
- What does it look like?

Real-time Lighting in Games

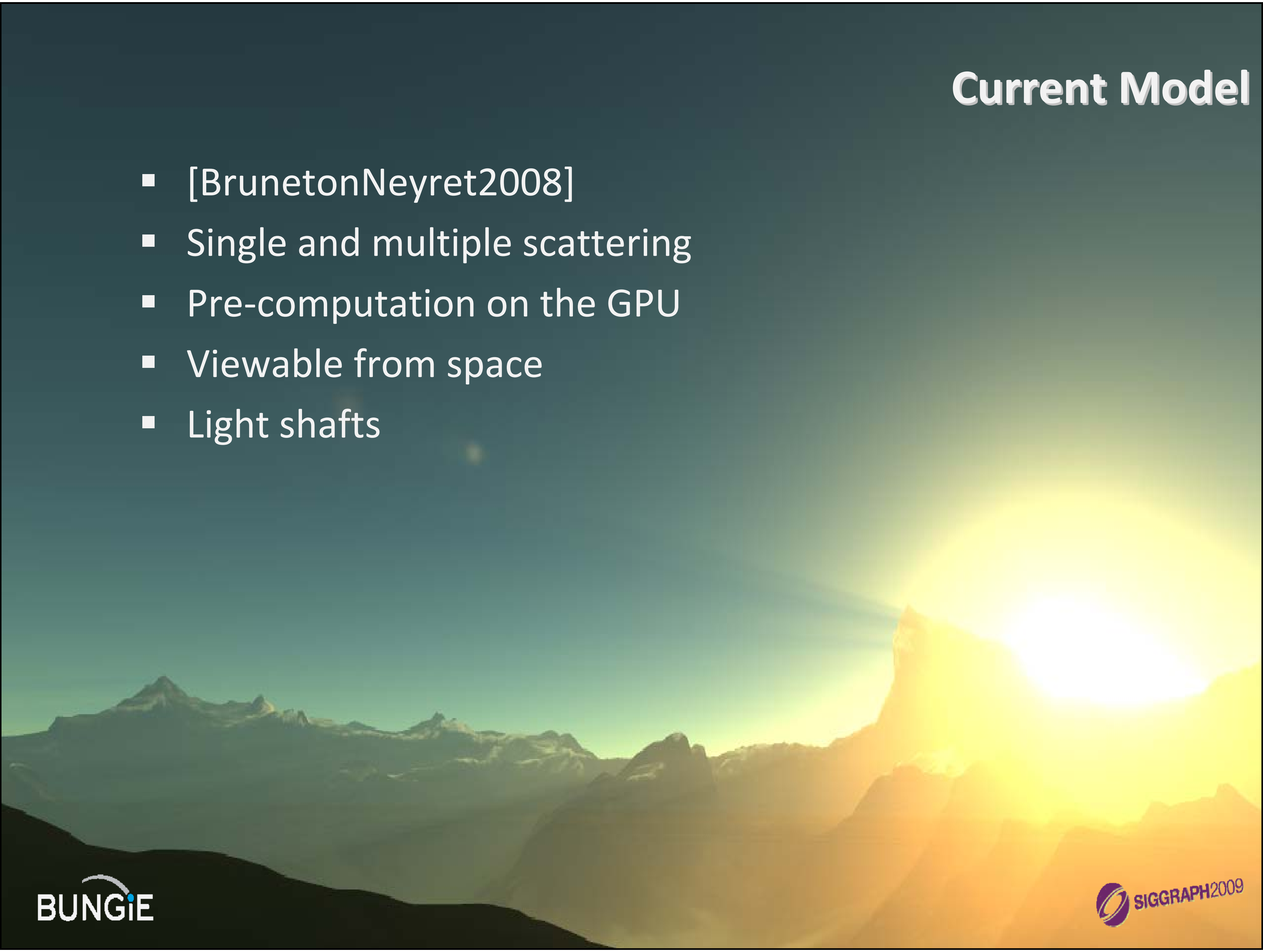
CRYENGINE 3
SIGGRAPH2009

Sky and Atmosphere: Previous Model

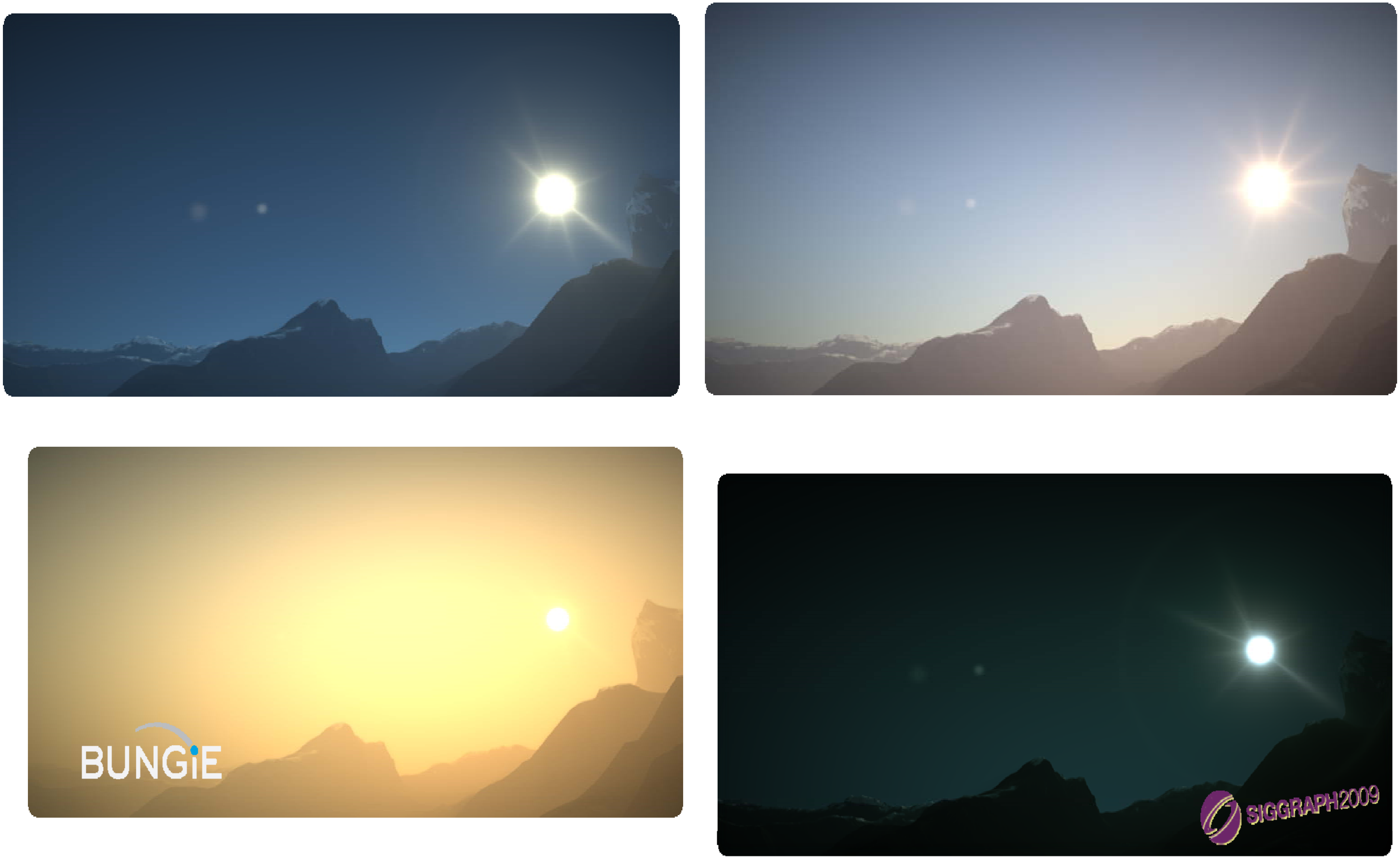
- Used in Halo 3
- [PSS99][PreethamHoffman03]
- Offline pre-computed sky texture
- Real-time scattering
- Single scattering only
- Viewable from ground only

Current Model

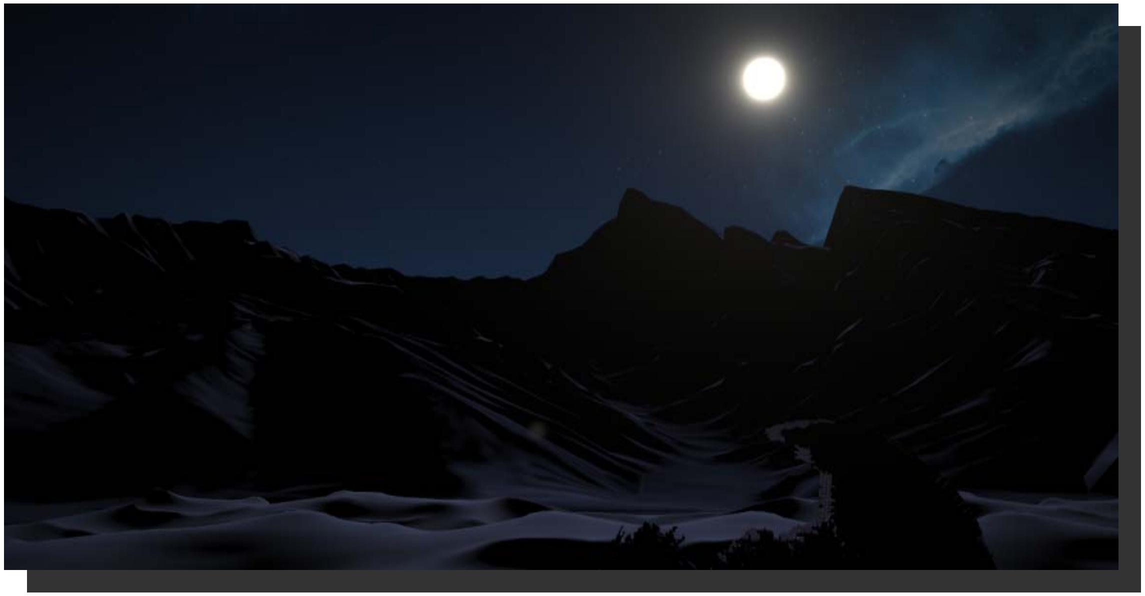
- [BrunetonNeyret2008]
- Single and multiple scattering
- Pre-computation on the GPU
- Viewable from space
- Light shafts



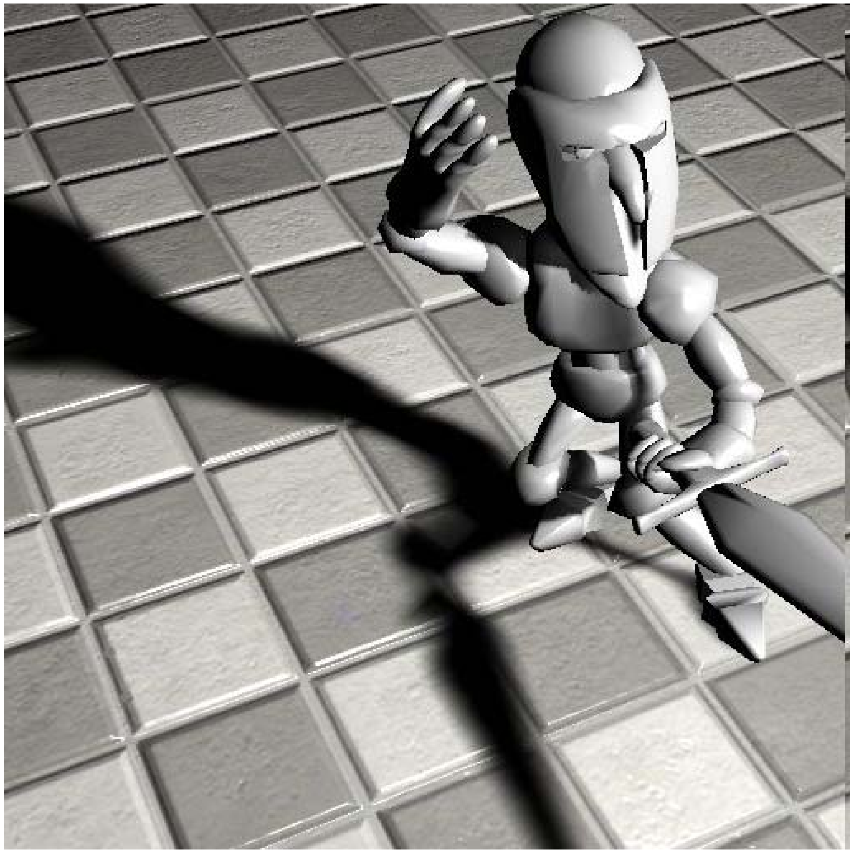
Different Atmospheres



Time Of Day



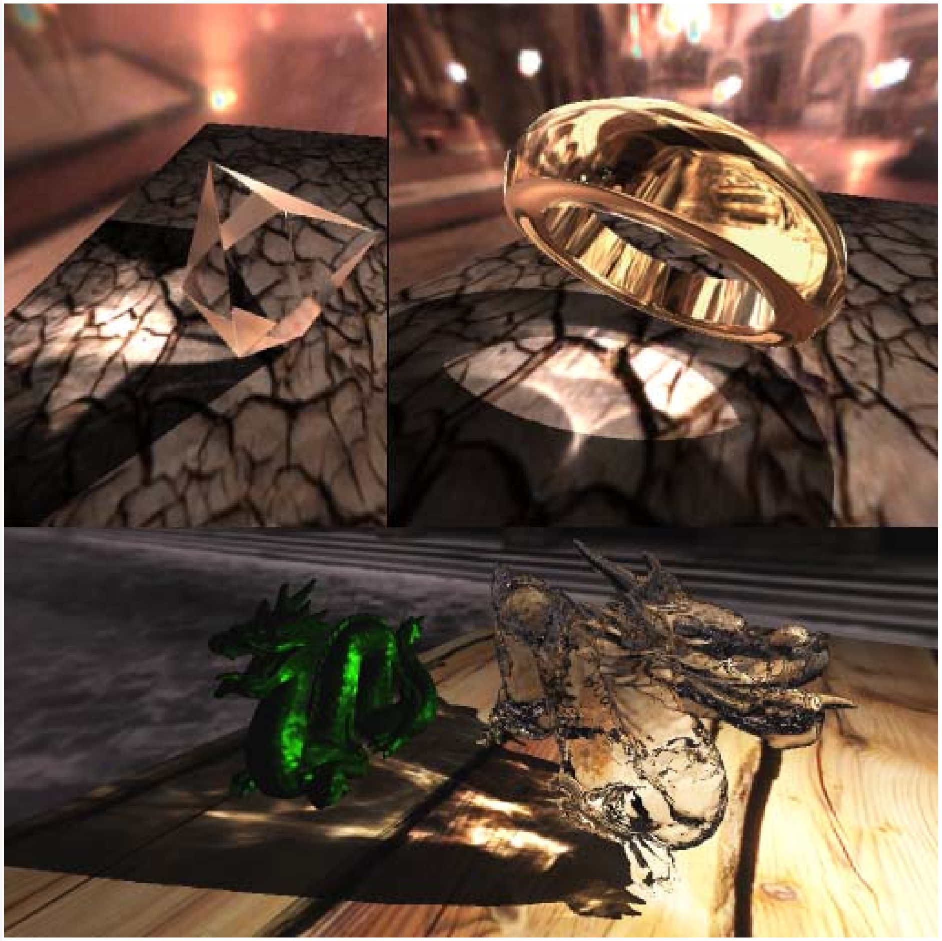
Shadows

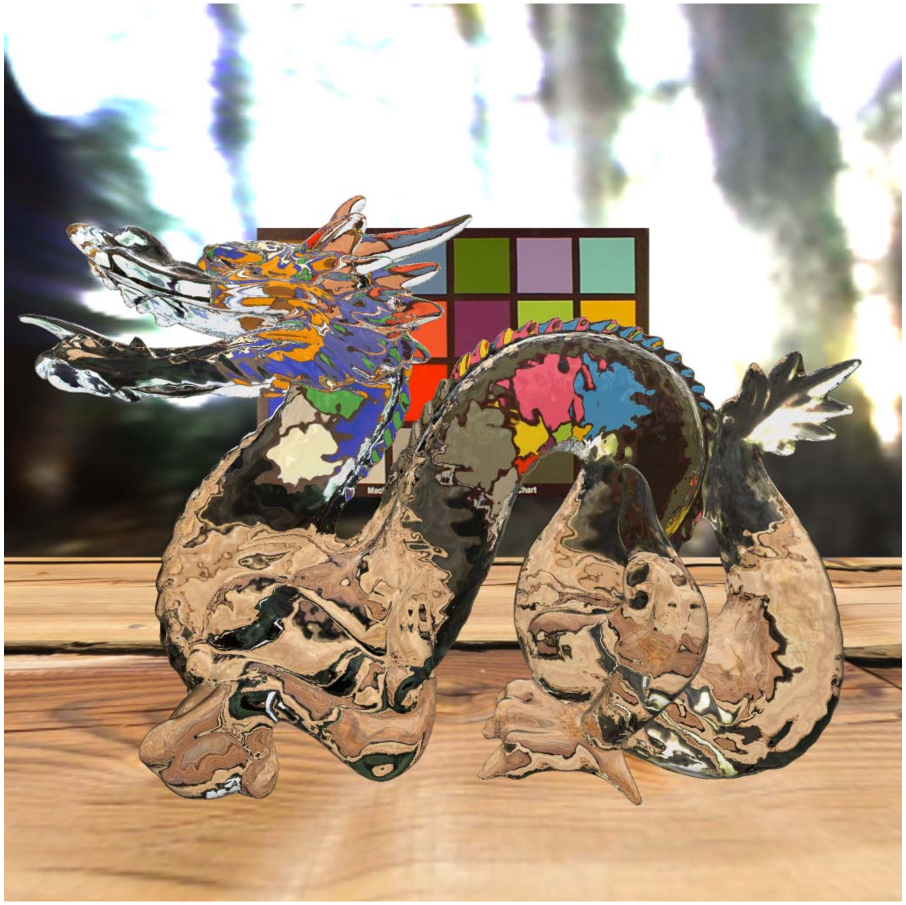


Variance shadow mapping
Courtesy Nvidia SDK 10

Courtesy Hellgate:London,
flagship studios inc

Caustics and Refraction





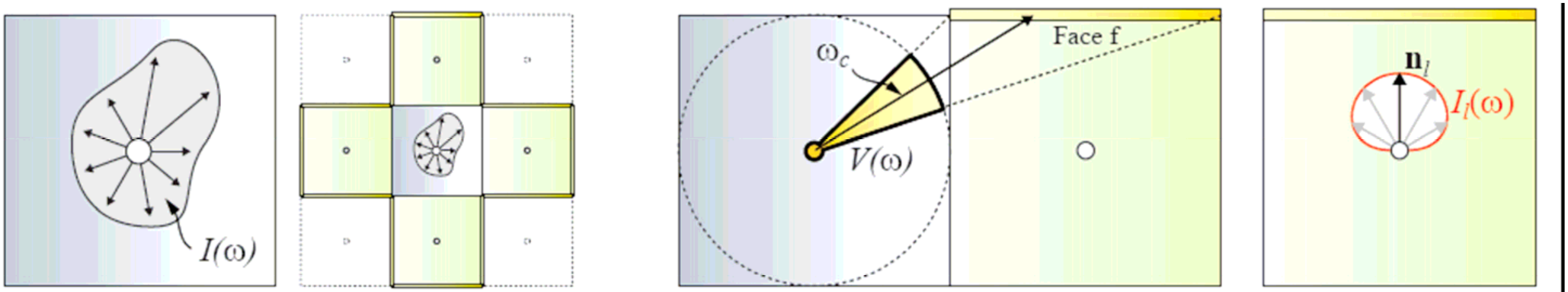
Courtesy Chris Wyman, Univ Iowa

CryEngine 3:GI with Light Propagation Volumes

- Goal: Real-time simulation of massive and indirect physically-based lighting

Core Idea

- **Main idea:** represent light bouncing as a set of secondary light sources: Virtual Point Lights (VPL)
- Project VPL into adjacent cells
- Reproject: figure out point light sources with same effect



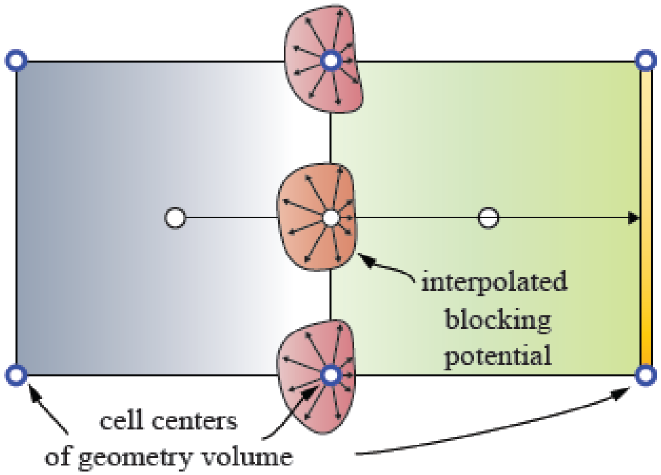
source cell

propagation along axial directions

source cell

destination cell

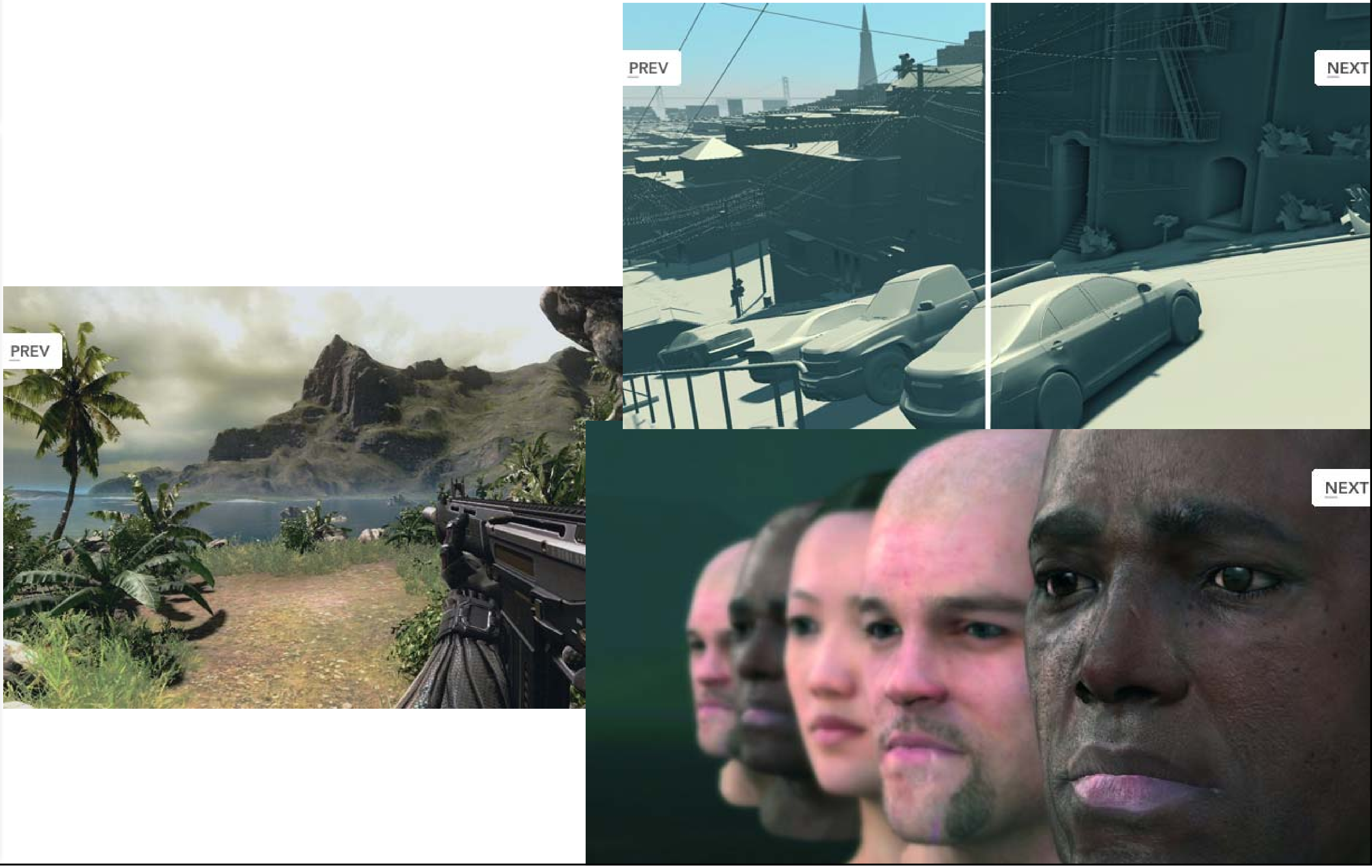
reprojection of the flux into a point light



interpolated blocking potential
cell centers of geometry volume

Crytek Crisis Engine Screenshots

- State of the art engine (graphics)



Trend 6: Physics Engines on GPU

- Nvidia Physx engine
- SDK: developer.nvidia.com/object/physx_features.html
  - Complex rigid body object physics system
  - Advanced character control
  - Ray-cast and articulated vehicle dynamics
  - Multi-threaded/Multi-platform/PPU Enabled
  - Volumetric fluid creation and simulation
  - Cloth and clothing authoring and playback
  - Soft Bodies
  - Volumetric Force Field Simulation
  - Vegetation

References

- Akenine Moller et al, Real-Time Rendering, 3rd edition
- Advances in Real-Time Rendering in 3D graphics and games, SIGGRAPH course notes 2009
- Anton Kaplanyan and Carsten Dachbacher, Cascaded light propagation volumes for real-time indirect illumination, in Proc. Si3D 2010
- Hao Chen and Natalya Tatarchuk, Lighting Research at Bungie, Advances in Real-Time Rendering in 3D Graphics and Games SIGGRAPH 2009 Course notes