

48

Illumination environments: the Basis Functions way

- Lighting environment:
a *continuous* function $f : \Omega \rightarrow \mathbb{R}$
- Where $f(\hat{v})$ = amount of light coming from direction \hat{v}
- Store f through basis functions

set of all unit vectors (i.e. surface of the unit sphere)

or \mathbb{R}^3 if RGB colored light

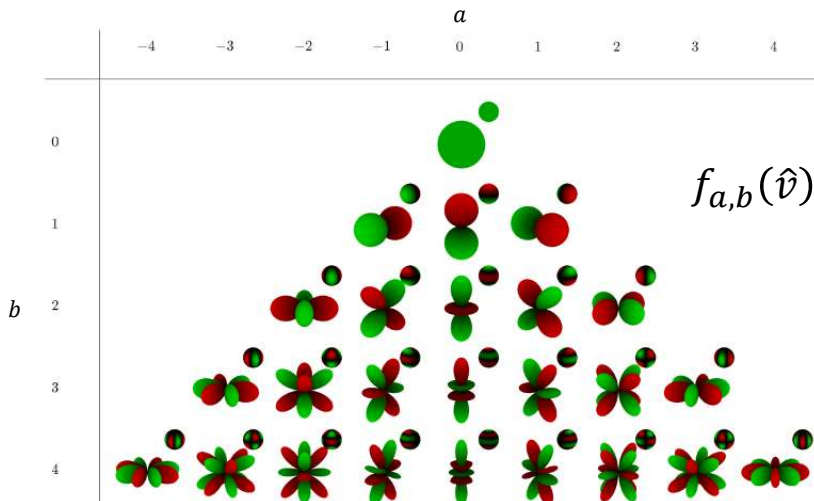
fixed spherical "basis" functions (always the same ones)

$$f(\hat{v}) = a_{0,0} \cdot f_{0,0}(\hat{v}) + a_{1,-1} \cdot f_{1,-1}(\hat{v}) + a_{1,0} \cdot f_{1,0}(\hat{v}) + a_{1,+1} \cdot f_{1,+1}(\hat{v}) + \dots$$

a few scalar values to be stored, in order to model f

49

Illumination environments: with basis functions



50

Illumination environments: with basis functions

- Spherical Harmonics (SPH) in brief:
 - store Illumination Env as a small number (1,4,9,16...) of scalar **weights** of as many fixed **spherical basis functions**.
- Pros:
 - very compact
 - models continuous function well: smooth environments
 - allows for efficient computation of the Lighting equation
- Cons:
 - continuous functions *ONLY*
 - Bad for hi-freq details, e.g. no hard lights
 - not much variations (unless very many coefficient used)
- Often good for background lights

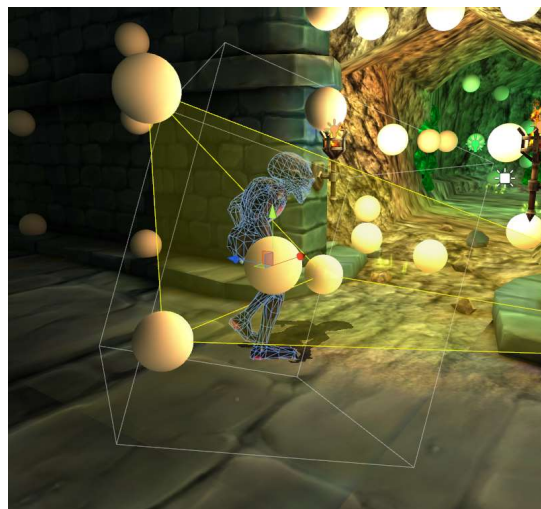
51

Light probes (position-dependent lighting env)

- A light probe == a (precomputed) lighting env to be used near a given (xyz) position of the scene
- Light Probe lighting:
 - preprocessing: disseminate the scene with light probes
 - Store them as... low res environment maps
 - ...or, with SPH (standard solution)
 - at rendering time, for a object currently in pos (xyz), use an interpolation of near-by "light probes"
 - note: two (or more) SPH function can be interpolated! (easy: just interpolate the weights)
- Widely used !

52

Light probes (position-dependent lighting env)



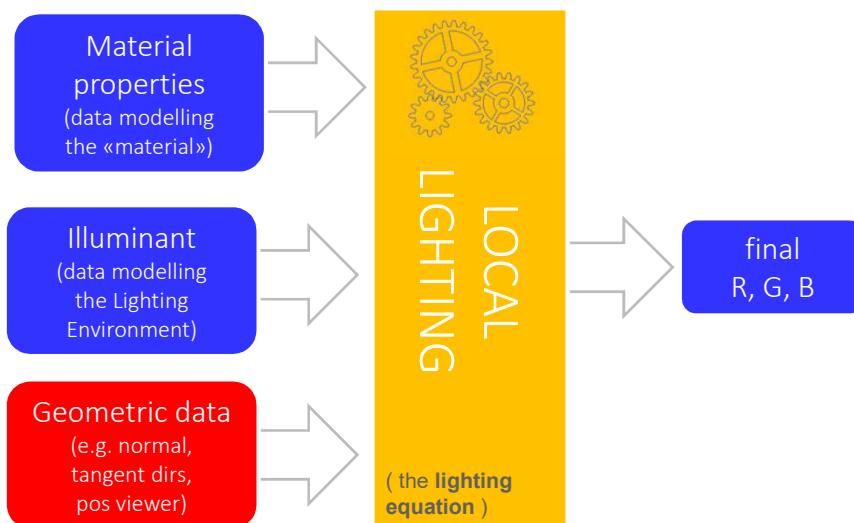
53

Light probes (position-dependent lighting env)



54

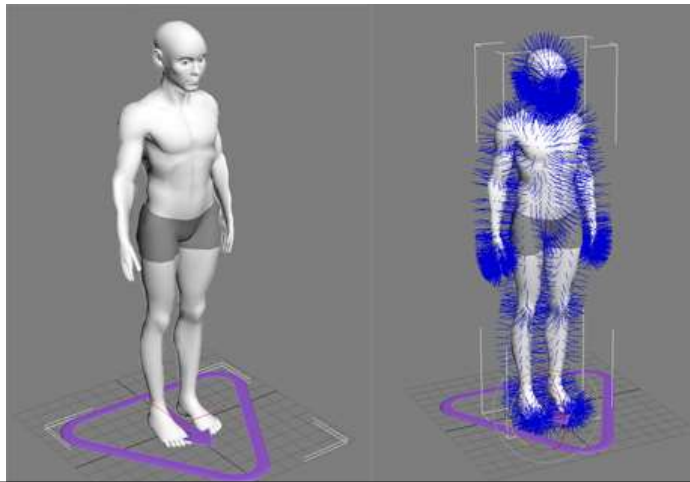
Local lighting in brief



55

Reminder: normals

- Per vertex attribute of meshes, or stored in normal maps



56

Reminder: (per vertex) Tangent directions



57