### Image Based Lighting

dr. Francesco Banterle

francesco.banterle@isti.cnr.it

#### Image Based Lighting: why?

- Image Based Lighting (IBL):
  - To (re)light synthetic objects with real-world lighting

#### Image Based Lighting: why?

- IBL is very important:
  - advertisement: cars, forniture, etc.
  - visual effects: CGI, live motion, etc.
  - augmented reality
  - cultural heritage

- The input of IBL is real-world lighting
  - HDR imaging is the key
  - A HDR photograph captures a limited portion of light coming from the point of capture



- Solution:
  - To capture HDR panoramic image 360x180
  - These images are typically called either environment map or lightprobe



- How capturing spherical (360x180) images?
- Single shot panorama cameras:
  - SpheronVR: 50Mpix and 24 f-stops
  - iSTAR 360: 50Mpix and 27 f-stops
  - Roundshot: 160Mpix
- These cameras may be expensive...



- to capture a mirror sphere (e.g. xmas ball)
- to capture a panoramic image from multiple directions and exposure times. This requires post-processing; e.g. image stitching:
  - PTGui:
    - http://www.ptgui.com
  - Hugin (open source):
    - <u>http://hugin.sourceforge.net/download/</u>















#### IBL: Longitude Latitude Mapping



$$\mathbf{D} = \begin{bmatrix} \sin\theta\cos\phi\\ \cos\theta\\ \sin\theta\sin\phi \end{bmatrix} \qquad \begin{array}{l} \theta = \pi \left(\frac{1}{2}\left(1 - \frac{j}{\text{height}}\right) - \frac{1}{2}\right)\\ \phi = \frac{\pi 2i}{\text{width}} \end{array}$$

#### IBL: Longitude Latitude Mapping

- Advantages:
  - easy mapping to understand/implement
- Disadvantages:
  - not equal-area —>pixels cover different areas on the sphere
  - squeezed at the poles —> to take this into account

### IBL: angular mapping



$$\mathbf{D} = \begin{bmatrix} \cos \phi \sin \theta \\ \sin \phi \sin \theta \\ -\cos \theta \end{bmatrix}$$

$$\phi = \arctan(1 - 2y, 2x - 1)$$
  
$$\theta = \pi \sqrt{(2x - 1)^2 + (2y - 1)^2}$$

### IBL: angular mapping

- Advantages:
  - avoiding undersampling at edges
- Disadvantages:
  - not equal-area —>pixels cover different areas on the sphere
  - a bit more complicated

#### IBL: cube mapping



$$\mathbf{D} = \frac{1}{\sqrt{1 + (2x - 1)^2 + (2y - 1)^2}} \begin{bmatrix} 2x - 1\\ 2y - 1\\ 1 \end{bmatrix}$$

$$x \in \left[\frac{1}{3}, \frac{2}{3}\right] \land y \in \left[\frac{1}{2}, \frac{3}{4}\right].$$

#### IBL: cube mapping



$$\mathbf{D} = \frac{1}{\sqrt{1 + (2x - 1)^2 + (2y - 1)^2}} \begin{bmatrix} 2x - 1\\ 2y - 1\\ 1 \end{bmatrix}$$

$$x \in \left[\frac{1}{3}, \frac{2}{3}\right] \land y \in \left[\frac{1}{2}, \frac{3}{4}\right].$$

### IBL: cube mapping

- Advantages:
  - hardware support on the GPU
- Disadvantages:
  - not equal-area (pixels are bigger at edges) —>pixels cover different areas on the sphere

... and now?



 $L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\Omega^+} L_i(\mathbf{x}, \vec{\omega}_i) f_r(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) |\vec{n} \cdot \vec{\omega}_i| d\vec{\omega}_i$ 

 $L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\Omega^+} L_i(\vec{\omega}_i) f_r(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) |\vec{n} \cdot \vec{\omega}_i| d\vec{\omega}_i$ 

### $L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\Omega^+} L_i(\vec{\omega}_i) f_r(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) |\vec{n} \cdot \vec{\omega}_i| d\vec{\omega}_i$



- How to solve this integral?
  - Creating light sources from the environment map
  - Sampling the environment map

### Light sources generation

- Direction light sources are extracted from the environment map.
  - Properties: direction, and HDR color
- The integral is converted into:

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}_o) + \sum_{j=1}^N L_i^j f_r(\vec{\omega}_i^j, \vec{\omega}_o) |\vec{n} \cdot \vec{\omega}_i^j|$$

 Number of light sources is a parameter: more lights more time. Few lights —> bias (integral not converged)

### Light sources generation: uniform sampling



Subdivide the panorama in regular regions

#### Light source generation: uniform sampling

			•	•	•										
-	-	-		-	-	-	-								-
•	•	•	•	•	•	•	•	•			•	•	•	•	•
•	•	•		•	•	•	•	•			•	•	•	•	•
•	•	•	•	•	•	•	•	•			•	•	•	•	•
		•	•	•	•	•									
			•				•					•			
			•	•											
:	:	:	•	•	:	:	:	:	:	•	:	:	:	:	:
:	:	:	:	•	:	:	:	•	:	:	:	:	:	:	:
•••••••••••••••••••••••••••••••••••••••	:	•	•	•	:	:			•		•	•	•	:	
•••••••••••••••••••••••••••••••••••••••	•••••••••••••••••••••••••••••••••••••••	•	•	••••••	••••••	••••••			••••••		•	••••••			•••••••••••••••••••••••••••••••••••••••
•••••••••••••••••••••••••••••••••••••••	•••••••••••••••••••••••••••••••••••••••	•	•••••••••••••••••••••••••••••••••••••••	•••••	••••••	•••••		•••••••••••••••••••••••••••••••••••••••	•••••	•••••••••••••••••••••••••••••••••••••••	•	••••••	•••••••••••••••••••••••••••••••••••••••	•••••••••••••••••••••••••••••••••••••••	•••••••••••••••••••••••••••••••••••••••
•••••••••••••••••••••••••••••••••••••••	•••••••••••••••••••••••••••••••••••••••	•	•	••••••	••••••	•••••	•••••	•••••••••••••••••••••••••••••••••••••••	••••••	•••••••••••••••••••••••••••••••••••••••	•	•••••	•	•	••••••
•••••••••••••••••••••••••••••••••••••••	•••••••••••••••••••••••••••••••••••••••	•	•	•	•	•	•	• • •	•	•••••••••••••••••••••••••••••••••••••••	•	•	•	•	•
•••••••••••••••••••••••••••••••••••••••	•	•	•	••••••	•••••••••••••••••••••••••••••••••••••••	•	•	•	• • • • •	••••••	•	•	•	•••••••••••••••••••••••••••••••••••••••	•

Extracted light sources

















		•	•	•											
•	•	•	•	•	•	•	•	•						•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
•	•	•		•	•	•	•	•			•	•	•	•	•
•	•	•	•	•	•	•	•	•			•	•	•	•	•
		•	•	•	•	•	•	•				•	•		•
		•		•			•	•			•				
			•	•											
•		•		•	•	•			•	•	•	•	•	•	•
	_														
	•	•	•	•	•	•	•			•	•	•	•	•	•
•		:	:	:	:	•	1	:	1	:	:	:	:	•	•
•		-	-	:	•	•	•	•	•	:	•	•	•	•	•
•		•	•	•	•	•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•	•••••	•	•	•	•	•

Extracted light sources



Extracted light sources





## Sampling the Environment Map

• Solving:

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}_o) + \int_{\Omega^+} L_i(\vec{\omega}_i) f_r(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) |\vec{n} \cdot \vec{\omega}_i| d\vec{\omega}_i$$

 with monte-carlo methods; generating samples according to a probability distribution:

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}_o) + \frac{1}{N} \sum_{j=1}^N \frac{L_i(\vec{\omega}_i^{x_j}) f_r(\vec{\omega}_i^{x_j}, \vec{\omega}_o) |\vec{n} \cdot \vec{\omega}_i^{x_j}|}{p(x_j)}$$

• Few samples —> noise

### Sampling the Environment Map



### Sampling the Environment Map



# how to insert virtual objects?









![](_page_53_Picture_1.jpeg)

Images are courtesy of Karsch

![](_page_54_Picture_1.jpeg)

![](_page_54_Picture_2.jpeg)

Synthetic Objects + Support Geometry

Support Geometry Only

![](_page_55_Picture_1.jpeg)

Shadows to insert into the photograph (table)

![](_page_56_Picture_1.jpeg)

![](_page_56_Picture_2.jpeg)

![](_page_57_Picture_1.jpeg)

![](_page_58_Picture_1.jpeg)

![](_page_59_Picture_1.jpeg)

there is more...

### nobody expects... the Spanish Inquisition

- A single environment can capture only distant light sources
- Nearby light sources are not modeled as local light but as distant ones
- To increase realism there is the need to model them properly

![](_page_63_Picture_1.jpeg)

![](_page_64_Picture_1.jpeg)

![](_page_65_Picture_1.jpeg)

![](_page_66_Picture_1.jpeg)

![](_page_67_Picture_1.jpeg)

#### Questions?