VU Rendering SS 2013
186.101

Thomas Auzinger

Károly Zsolnai

Institute of Computer Graphics and Algorithms (E186)
Vienna University of Technology

http://www.cg.tuwien.ac.at/staff/ThomasAuzinger.html

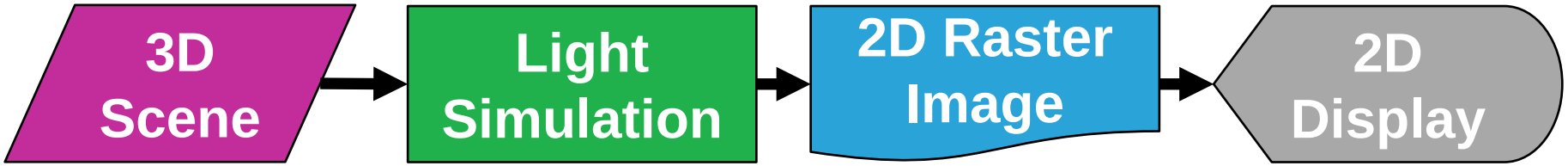
http://www.cg.tuwien.ac.at/staff/KarolyZsolnai.html

VU Rendering SS 2013

Unit 06 – Tone Mapping

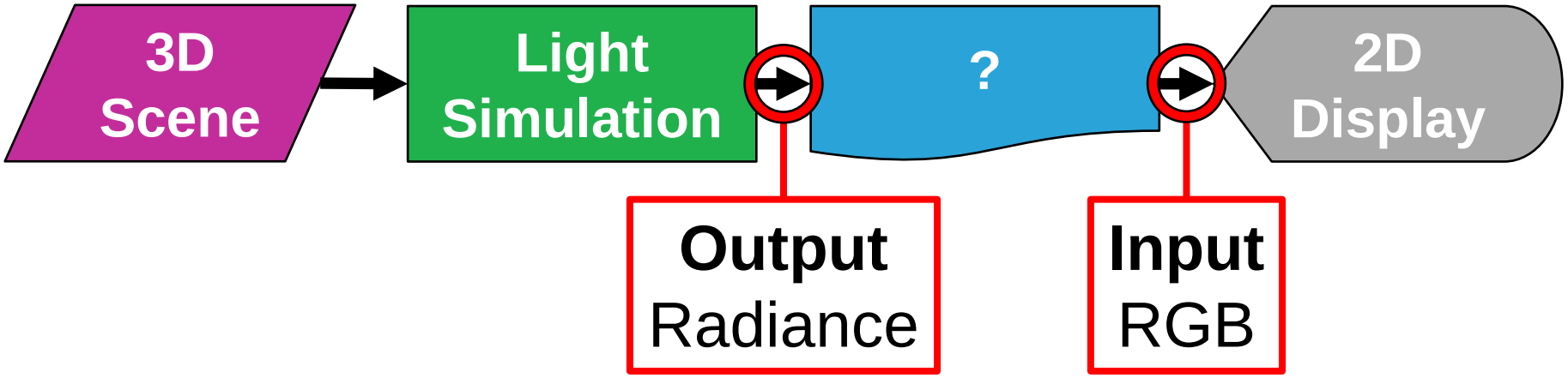
Scope

Rendering pipeline



Scope

Rendering pipeline



Dynamic Range

Dynamic Range	Device/Setting
Real world	
800,000:1	Surface illuminated sun/moon
100:1	Diffuse white/black surface
80,000,000:1	Expected real world dynamic range

Dynamic Range

Dynamic Range	Device/Setting
Real world	
800,000:1	Surface illuminated sun/moon
100:1	Diffuse white/black surface
80,000,000:1	Expected real world dynamic range
Human vision	
100:1	Photoreceptors
10:1	Pupil size
100,000:1	Neural adaptation
100,000,000:1	Dynamic range of human vision

Dynamic Range

Dynamic Range	Device/Setting
Real world	
800,000:1	Surface illuminated sun/moon
100:1	Diffuse white/black surface
80,000,000:1	Expected real world dynamic range
Human vision	
100:1	Photoreceptors
10:1	Pupil size
100,000:1	Neural adaptation
100,000,000:1	Dynamic range of human vision
Technology	
1,000:1	Display
256:1	8 bit image file

Tone Mapping

- Output of our light simulation:
  real world dynamic range or
  High dynamic range (HDR)

- Input of conventional display devices:
  Low dynamic range (LDR)

- Need to apply range compression. Referred to as
  - **Tone mapping**
  - **Tone reproduction**

Tone Mapping

1. **Range compression**
   Convert HDR luminance to LDR luminance
   (Content of this lecture)

2. **Color space conversion**
   Convert LDR luminance to desired color space
   (RGB, XYZ, CMYK, ...)
   (Covered in the color lecture 186.826)

Examples

Bright light source with no ambient light

(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

Examples

Short exposure

(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

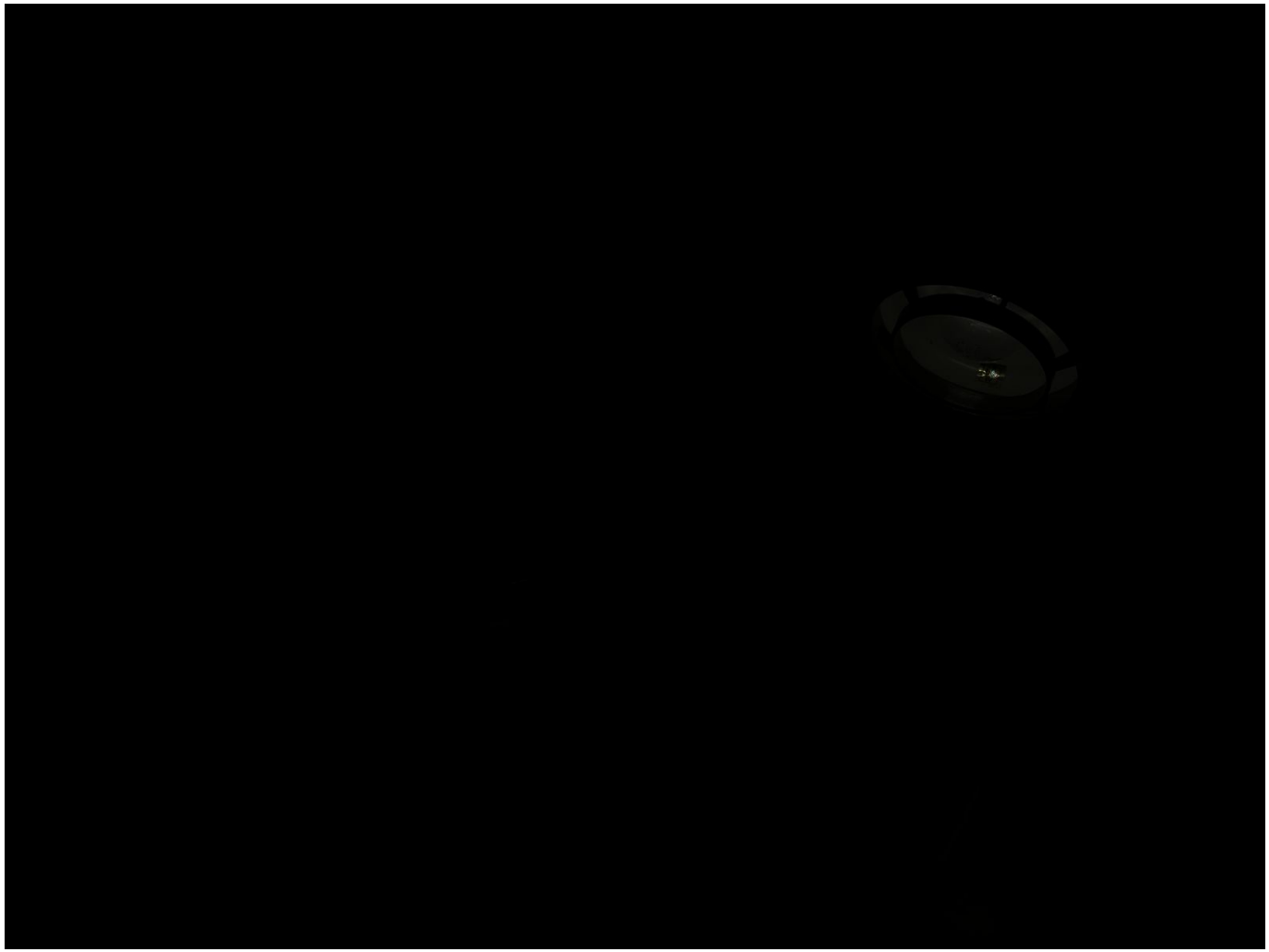
Examples

Long exposure

(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

Naive Tone Mapper

Division by maximum



(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

Naive Tone Mapper

Clamp

(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

Naive Tone Mapper

Exponential mapping

(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

Working Tone Mapper

Reinhard tone mapper

(from http://dcgi.felk.cvut.cz/home/cadikm/tmo/)

Misused Tone Mapping

Read http://www.cs.bris.ac.uk/~reinhard/tm_comp/

Misused Tone Mapping

Read http://www.cs.bris.ac.uk/~reinhard/tm_comp/

Tone Mapper Taxonomy

- **Global**
  Mapping function is uniform on the whole image.

- **Local**
  Mapping function depends on the values of the pixels in the neighborhood of the currently mapped pixel.

Tone Mapper Taxonomy

- **Global**

  Mapping function is uniform on the whole image.

  + Fast

  - Loss of detail

- **Local**

  Mapping function depends on the values of the pixels in the neighborhood of the currently mapped pixel.

  - Slow

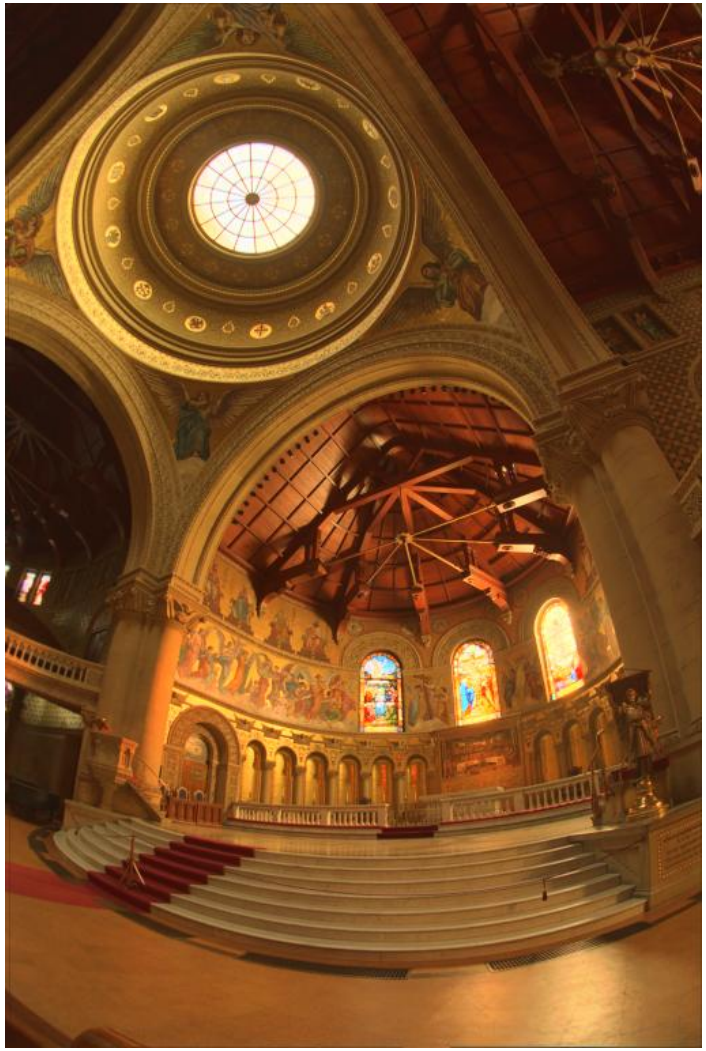
  + Local contrast enhancement

Reinhard Tone Mapper

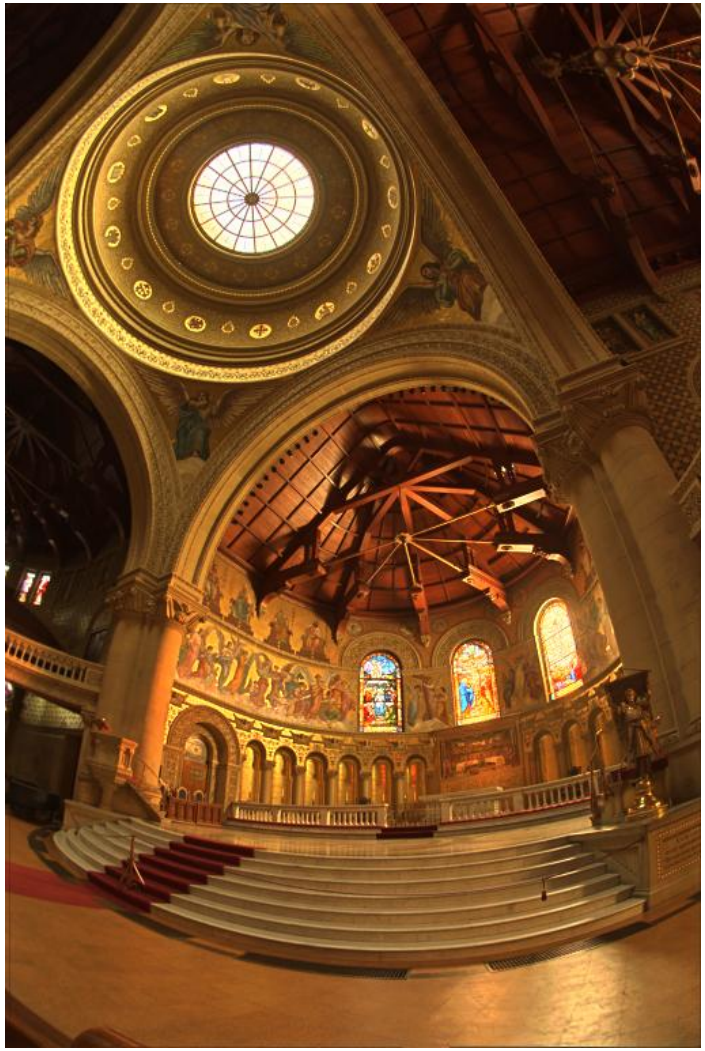
- Developed by Reinhard et al.

  Reinhard et al., *Photographic tone reproduction for digital images*, Siggraph 2002
- Widely used
- Global and local variant

Reinhard Tone Mapper



Global



Local

(from http://cybertron.cg.tu-berlin.de/pdci08/tonemapping/tone_mapping.html)

Reinhard Tone Mapper

Processing steps (global version)

- Compute log average (luminance space)

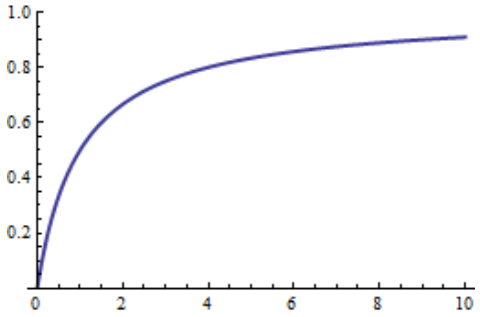
$$L_{\text{avg}} = \exp\left(\frac{1}{N}\sum_{x,y}\log(L_{\text{input}}(x,y))\right) = \sqrt[N]{\prod_{x,y} L_{\text{input}}(x,y)}$$

- Map average value to *middle-gray* given by a

$$L_{\text{scaled}}(x,y) = \frac{a}{L_{\text{avg}}} L_{\text{input}}(x,y)$$

- Compress high luminances

$$L_{\text{output}}(x,y) = \frac{L_{\text{scaled}}(x,y)}{1 + L_{\text{scaled}}(x,y)}$$



Reinhard Tone Mapper

Processing steps (local version)

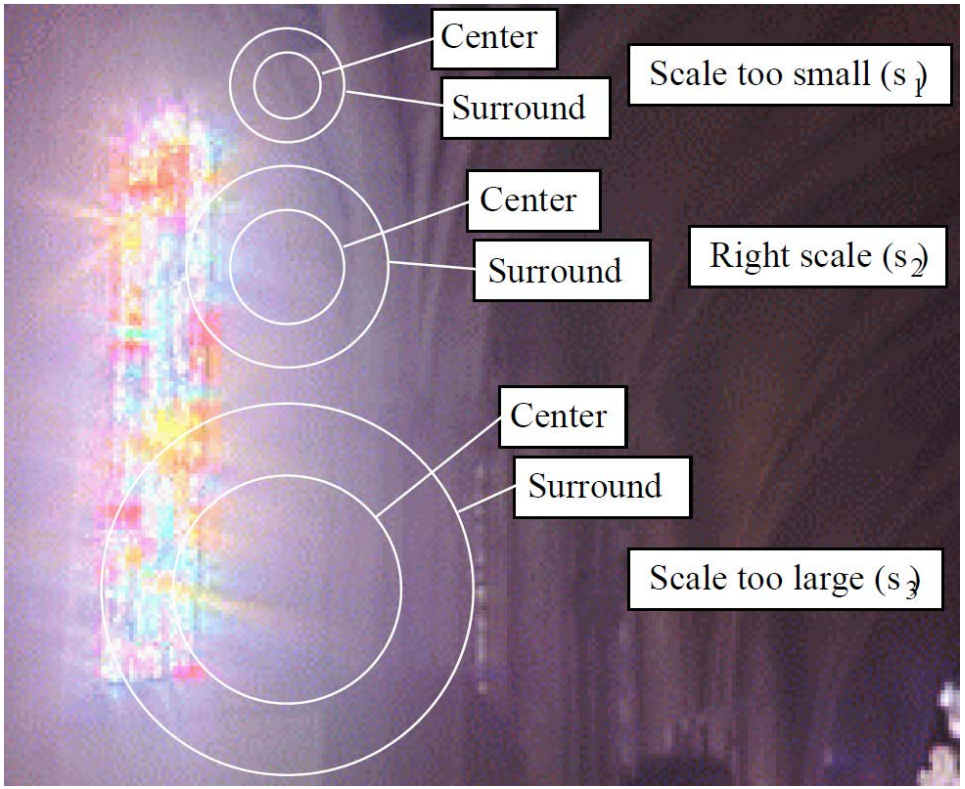
- Compute log average
- Map average value to *middle-gray*
- Compress luminances by spatially varying function

$$L_{\text{output}}(x,y) = \frac{L_{\text{scaled}}(x,y)}{1 + V(x,y,s(x,y))}$$

  - Local average $V(x,y,s)$ with scale s (think Gaussian smoothing).
  - Local scale $s(x,y)$ where $L_{\text{scaled}}(x,y)$ does not change.

Reinhard Tone Mapper

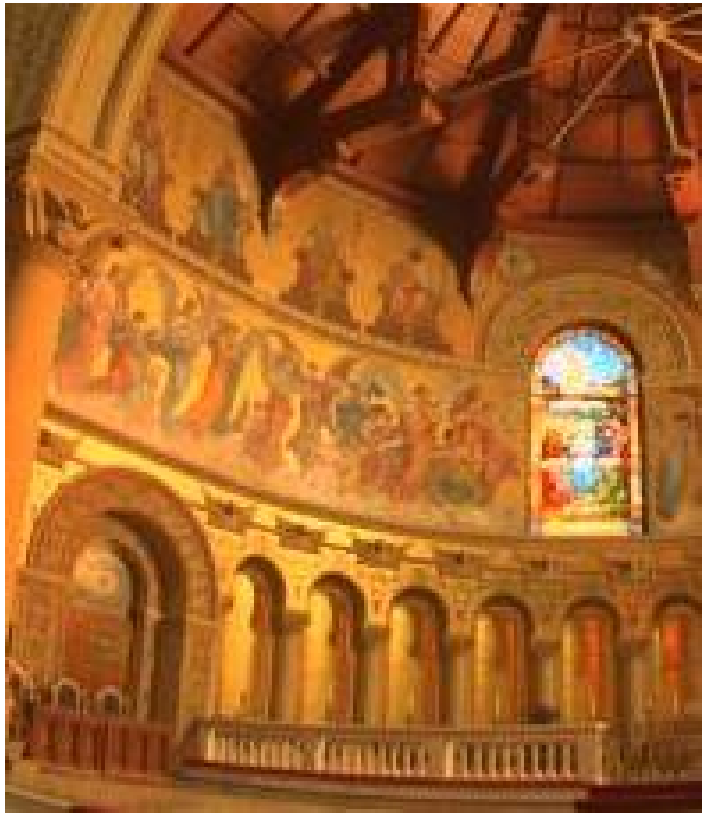
Scale estimation at circle center points

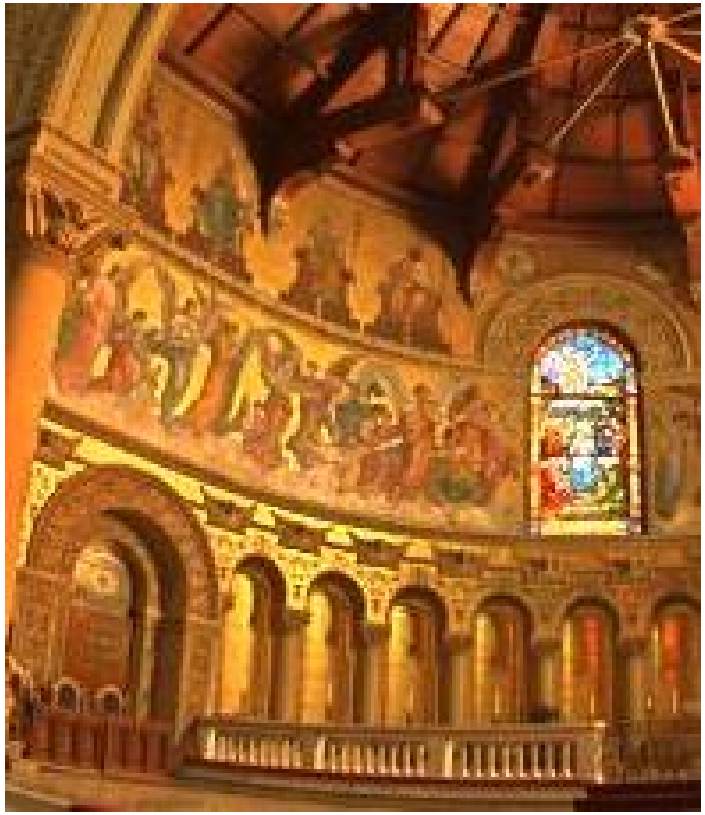


(from **Reinhard et al. 2002**)

- **Center** area should be **smooth**
- **Surround** area should **not** be **smooth**

Reinhard Tone Mapper





Global

Local

(from http://cybertron.cg.tu-berlin.de/pdci08/tonemapping/tone_mapping.html)

Other Tone Mappers

Wide variety using

- bilateral filters

  F. Durand and J. Dorsey, *Fast bilateral filtering for the display of HDR images*, Siggraph 2002

- gradient processing

  Fattal et al., *Gradient domain HDR compression*, Siggraph 2002

- etc... (wikipedia.de lists over 20)

Literature

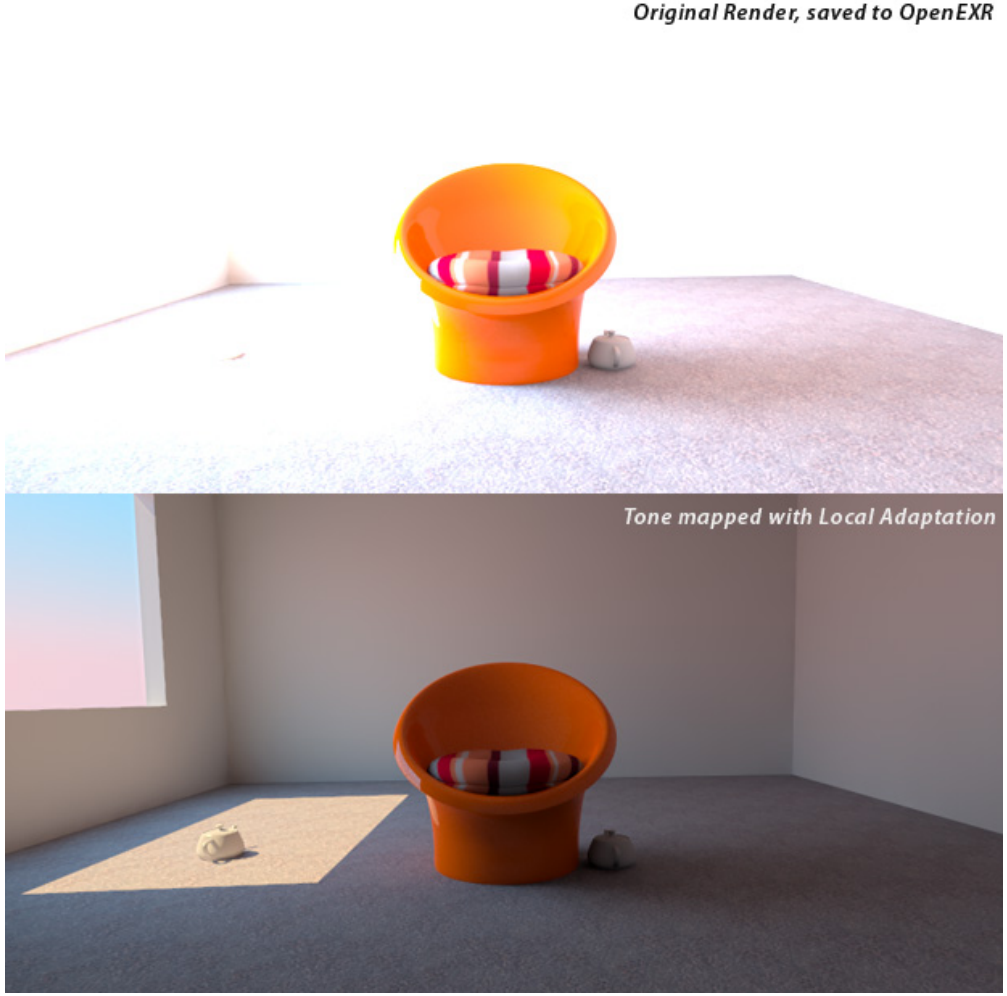
Textbooks

- Reinhard et al., *High Dynamic Range Imaging*, 2. ed., 2010

Papers

- Reinhard et al., *Photographic tone reproduction for digital images*, Siggraph 2002
- F. Durand and J. Dorsey, *Fast bilateral filtering for the display of HDR images*, Siggraph 2002
- Fattal et al., *Gradient domain HDR compression*, Siggraph 2002

Questions?



(from http://www.artstorm.net/journal/2009/07/color-management-linear-workflow/)