Chair for Information Architecture

Lukas Treyer
treyer@arch.ethz.ch

New Methods in Urban Simulation
Zurich, Nov 21 2011

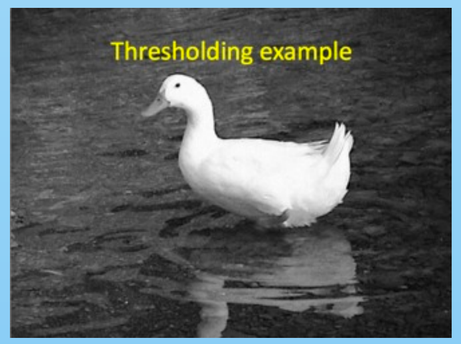
RENDERING / COMPUTER GRAPHICS

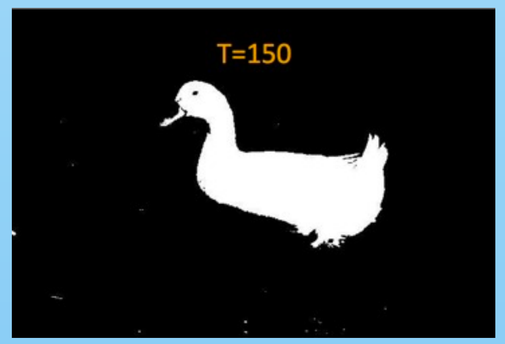
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

COMPUTER GRAPHICS

- Computer Vision
- Concepts in Computer Graphics
  - Rasterization
  - Raytracing / Physically Based Materials
  - Level of Detail
  - Linear Interpolation
- Photography
  - DOF
  - Image Design / Shift
- Light
  - 3 Point Light
  - Studio Lights
- Post Processing *(Photoshop/Composition)*

COMPUTER VISION





„Find Edges“ / Treshold
Source: Computer Graphics Laboratory ETH Zürich

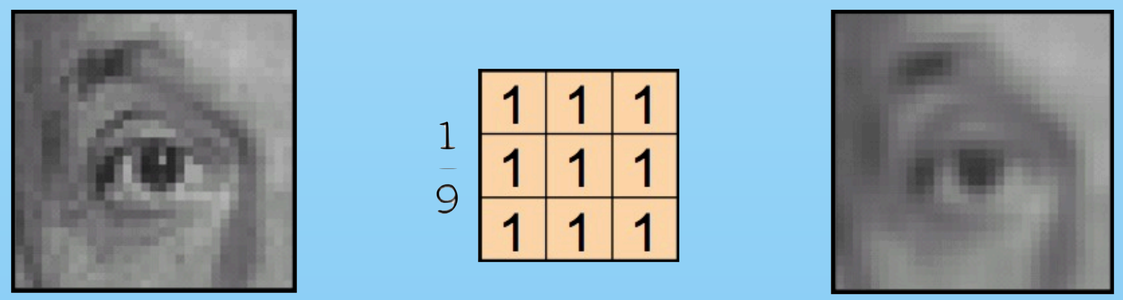
Image Processing

- Image Segmentation
  - *Which pixels form an object?*
  - *Grouping of related pixels*
- Correspondence (Find related/similar image regions)
  - *Lookup in database*
  - *Search in other images: optical flow, stereo etc.*

COMPUTER VISION

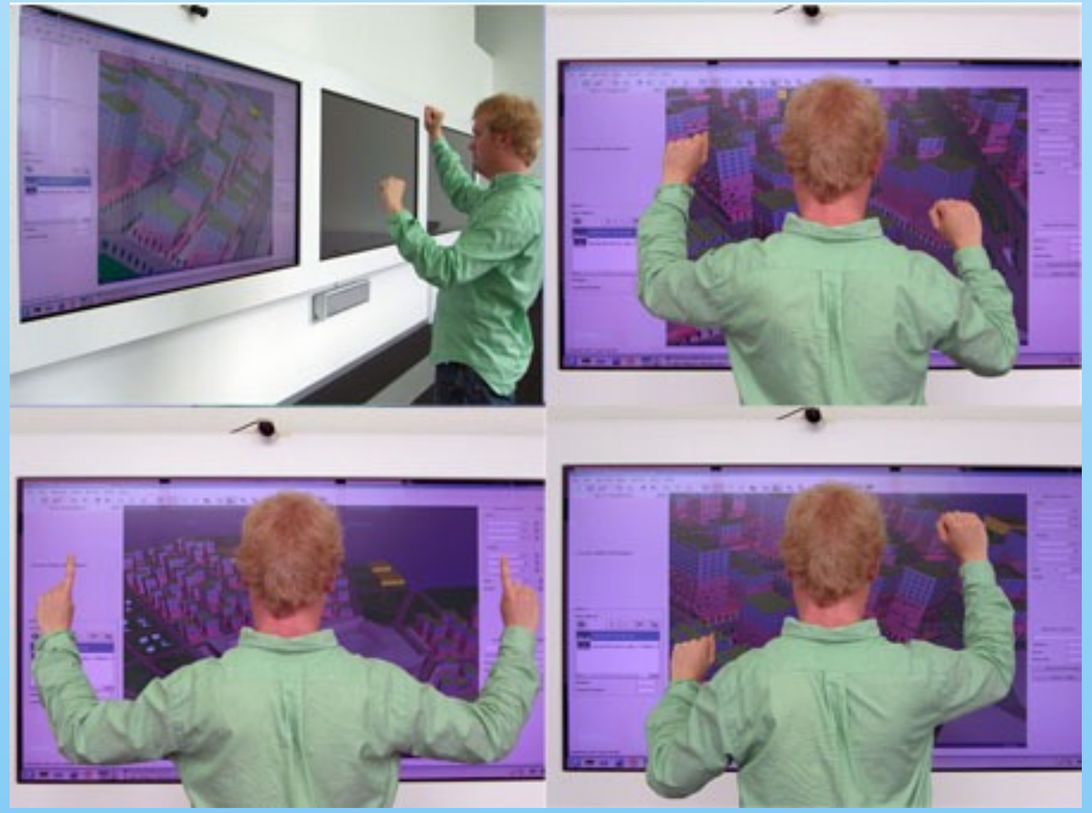
Image Processing

- Image Segmentation
- Image Filtering



Linear Filtering
Source: Computer Graphics Laboratory ETH Zürich

COMPUTER VISION



Stereoscopy
Source: Value Lab ETH Zürich

Image Processing

- Image Segmentation
- Image Filtering
- Stereoscopy / Kinect 3D Scan

COMPUTER VISION

Kinect
Source: http://www.xbox.com

Image Processing

- Image Segmentation
- Image Filtering
- Stereoscopy / Kinect 3D Scan

COMPUTER VISION

Image Processing

- Image Segmentation
- Image Filtering
- Stereoscopy / Kinect 3D Scan
- 3D Reconstruction
- Face / Pattern Detection ...

Stereoscopy
Source: Computer Graphics Laboratory ETH Zürich

COMPUTER VISION

Image Processing

- Image Segmentation
- Image Filtering
- Stereoscopy / Kinect 3D Scan
- 3D Reconstruction
- Face / Pattern Detection ...

Image Creation (Computer Graphics)

- Games (OpenGL/DirectX/Consoles)
- Rendering Engine
- Compositions (After Effects / Motion)

Stereoscopy
Source: Computer Graphics Laboratory ETH Zürich

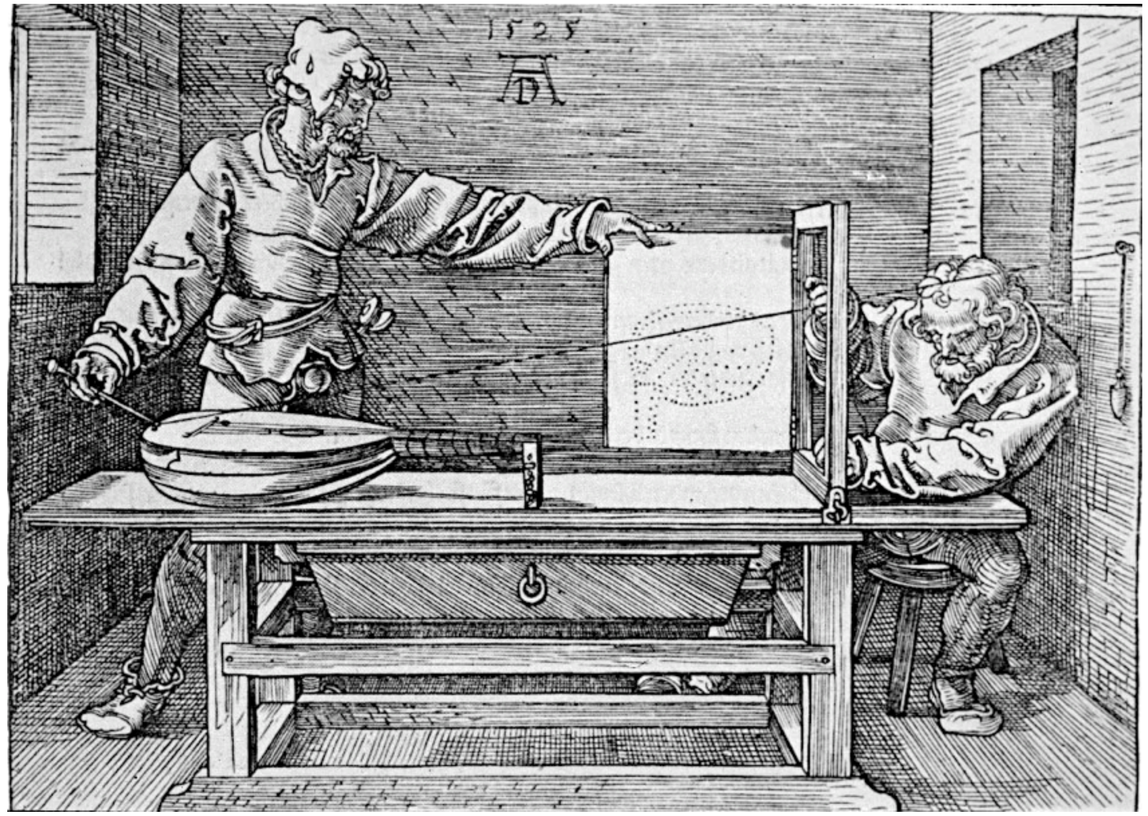
COMPUTER GRAPHICS

Computer Vision

Concepts in Computer Graphics

- Rasterization
- Raytracing / Physically Based Materials
- Level of Detail
- Linear Interpolation

CONCEPTS IN COMPUTER GRAPHICS

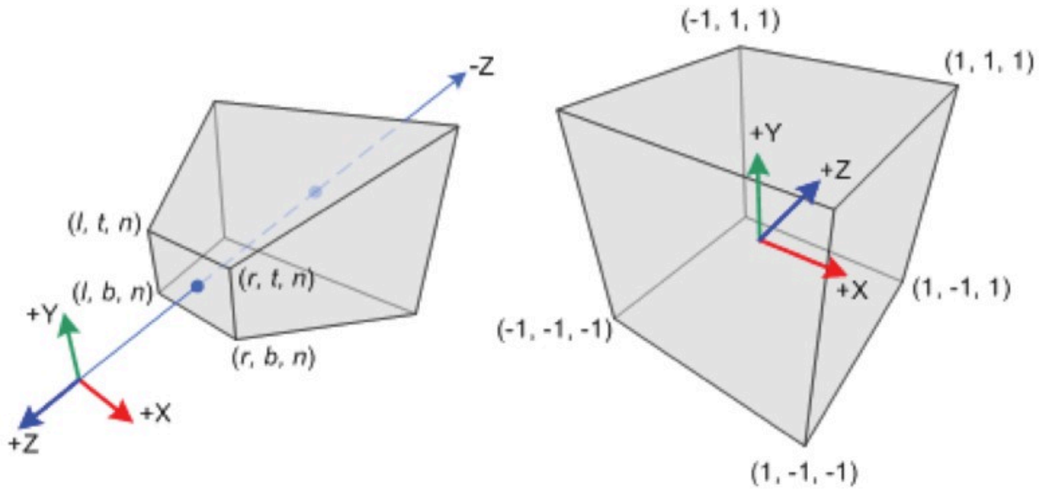


Albrecht Dürer, 1525: An early rasterization technique used to draw a perspectivly correct image
Source: http://de.wikipedia.org/wiki/Raytracing

Rasterization

- Projection of a series of (3D) objects/ shapes onto a 2D plane
- Base for all computer graphics creation

CONCEPTS IN COMPUTER GRAPHICS



Frustum and orthogonal view cube in OpenGL
Source: Computer Graphics Lab, ETH Zurich

Rasterization

- Projection of a series of (3D) objects/ shapes onto a 2D plane
- Base for all computer graphics creation
- Orthogonal or perspective projection
- basic algorithms implemented on graphic cards (OpenGL/DirectX)

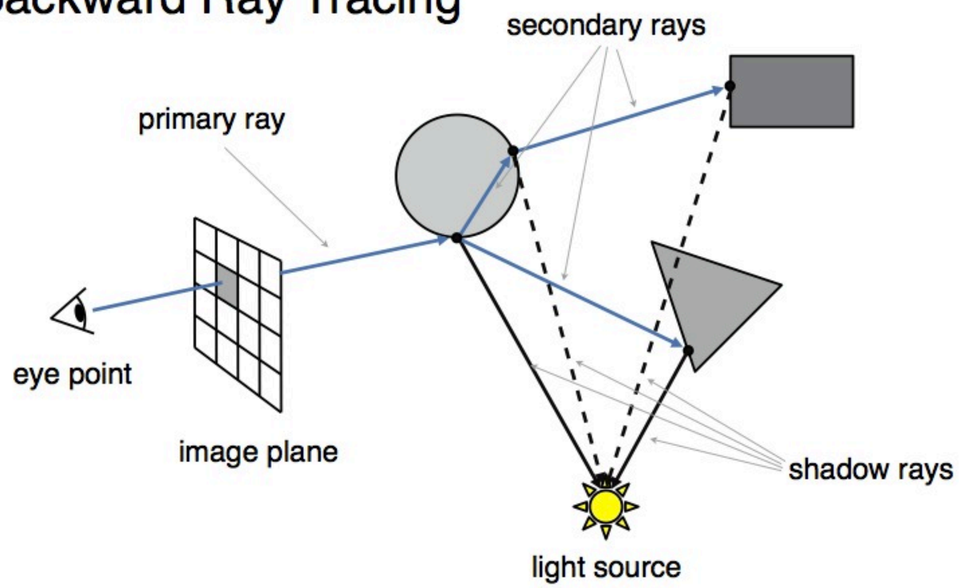
CONCEPTS IN COMPUTER GRAPHICS

Rasterization
Raytracing / Physically Based
Materials
Level of Detail
Linear Interpolation

CONCEPTS IN COMPUTER GRAPHICS

Raytracing / Physically Based

- Backward Ray Tracing

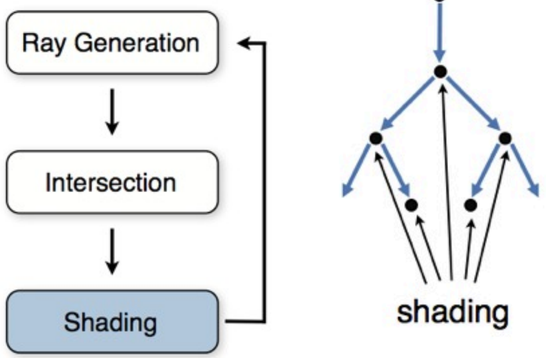


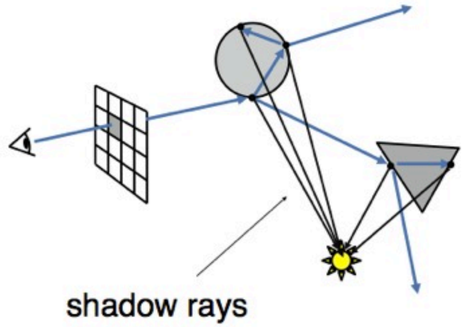
Backward Ray Tracing
Source: Computer Graphics Lab, ETH Zurich

CONCEPTS IN COMPUTER GRAPHICS

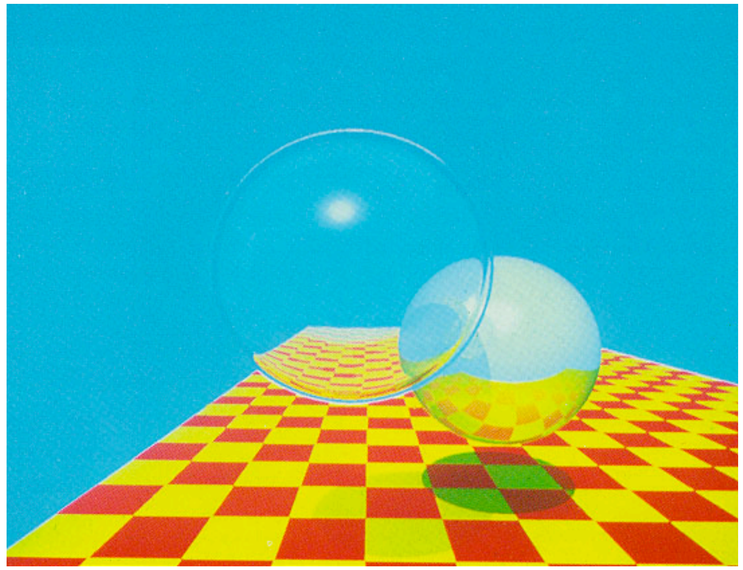
Raytracing / Physically Based

- Recursion tree



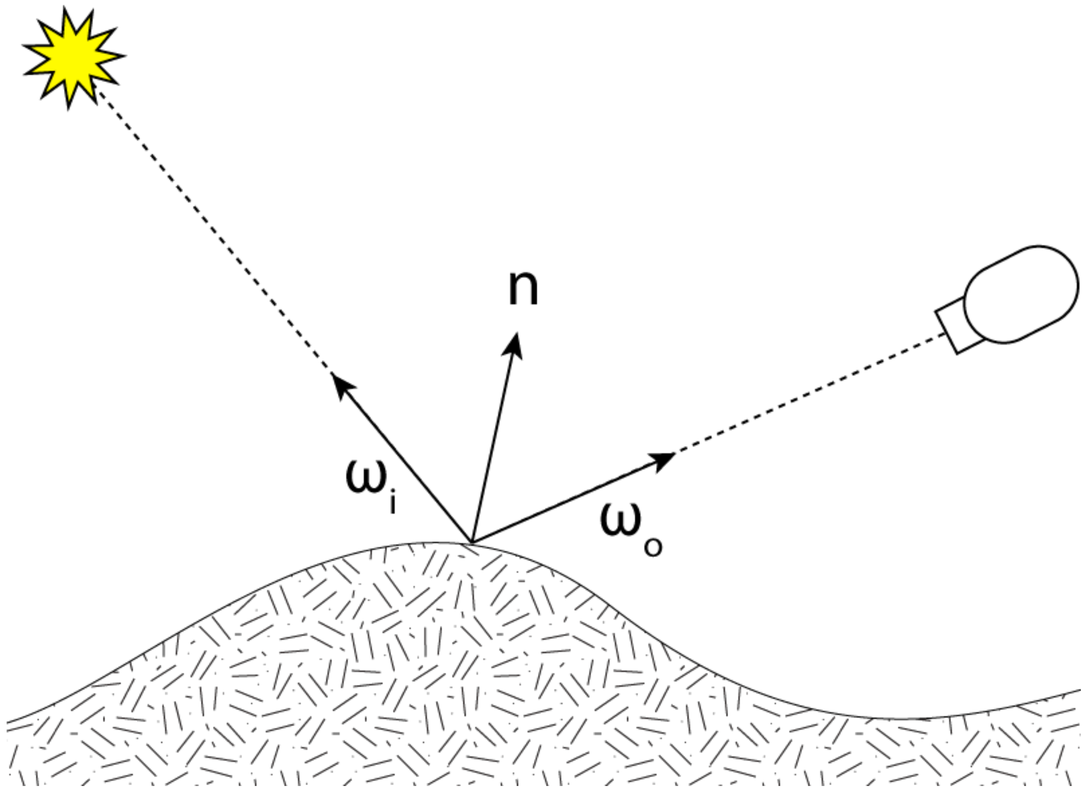


Backward Ray Tracing
Source: Computer Graphics Lab, ETH Zurich



Turner Whitted 1980
Source: http://de.wikipedia.org/wiki/Raytracing

CONCEPTS IN COMPUTER GRAPHICS



Bidirectional Reflectance Distribution Function, BRDF
Source: Wikipedia

Raytracing / Physically Based

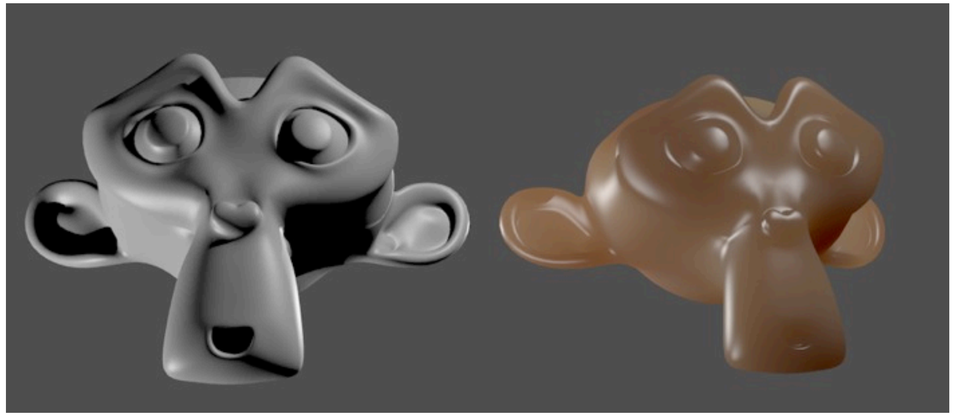
BRDF first defined by Fred Nicodemus around 1965:

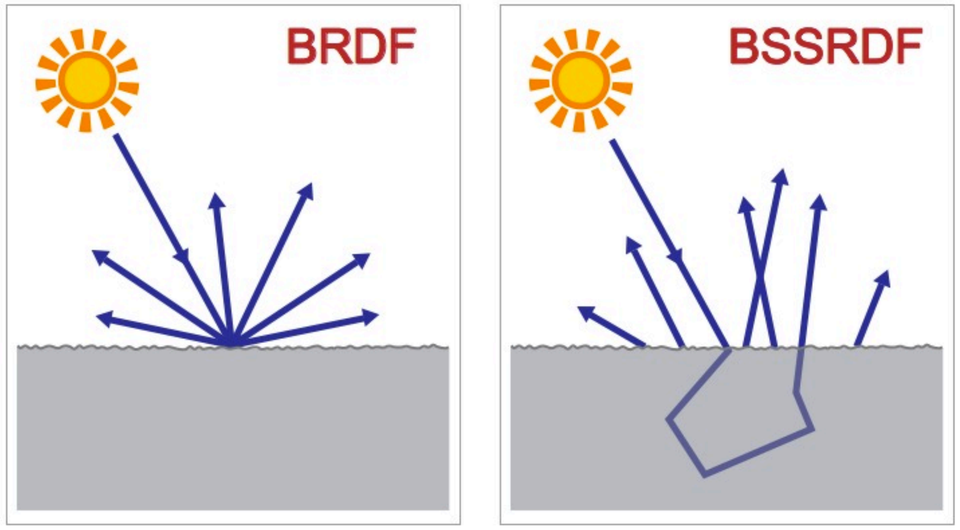
$$f_r(\omega_i, \omega_o) = \frac{dL_r(\omega_o)}{dE_i(\omega_i)} = \frac{dL_r(\omega_o)}{L_i(\omega_i)\cos\theta_i \, d\omega_i}$$

Physically based BRDFs have additional properties:

positivity: $f_r(\omega_i, \omega_o) \geq 0$
obeying Helmholtz reciprocity: fr(ωi,ωo) = fr(ωo,ωi).
conserving energy: $\forall \omega_i, \int_\Omega f_r(\omega_i, \omega_o) \cos\theta_o d\omega_o \leq 1$

CONCEPTS IN COMPUTER GRAPHICS





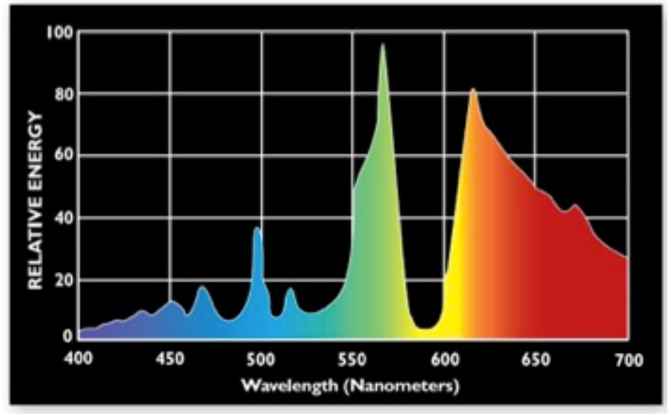
Bidirectional Surface Scattering Reflectance Distribution Function, BSSRDF
Source: Wikipedia

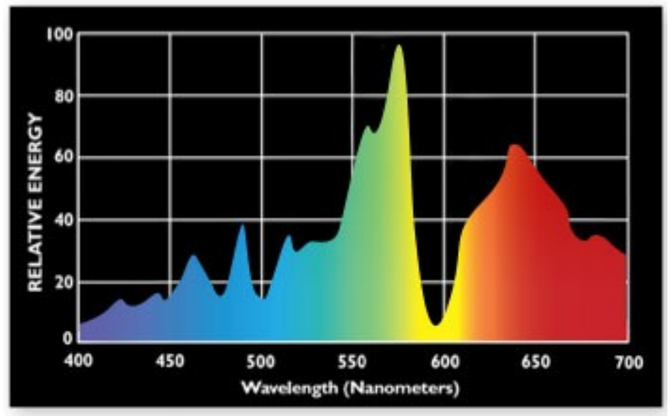
Raytracing / Physically Based

Physically based BRDFs have additional properties:

positivity: $f_r(\omega_i, \omega_o) \geq 0$
obeying Helmholtz reciprocity: fr(ωi,ωo) = fr(ωo,ωi).
conserving energy: $\forall \omega_i, \int_\Omega f_r(\omega_i, \omega_o) \cos\theta_o d\omega_o \leq 1$
BSSRDF

CONCEPTS IN COMPUTER GRAPHICS





Two different SPD‘s may appear as the same RGB value to the human eye!

Spectral Power Distribution, SPD
Source: Computer Graphics Laboratory, ETH Zurich

Raytracing / Physically Based

Physically based BRDFs have additional properties:

positivity: $f_r(\omega_i, \omega_o) \geq 0$
obeying Helmholtz reciprocity: fr(ωi,ωo) = fr(ωo,ωi).
conserving energy: $\forall \omega_i, \int_\Omega f_r(\omega_i, \omega_o) \cos\theta_o d\omega_o \leq 1$
BSSRDF
wavelength dependend: λ instead of RGB
→ $f_r(\lambda_i, \omega_i, \lambda_o, \omega_o)$

CONCEPTS IN COMPUTER GRAPHICS

Iridescence

Luminescence;
emission of light by a substance not resulting from heat
Source:Wikipedia

Raytracing / Physically Based

Physically based BRDFs have additional properties:

positivity: $f_r(\omega_i, \omega_o) \geq 0$
obeying Helmholtz reciprocity: fr(ωi,ωo) = fr(ωo,ωi).
conserving energy: $\forall \omega_i, \int_\Omega f_r(\omega_i, \omega_o) \cos\theta_o d\omega_o \leq 1$
BSSRDF
wavelength dependend: λ instead of RGB

→ $f_r(\lambda_i, \omega_i, \lambda_o, \omega_o)$

→ *account for effects such as iridescence or luminescence*

CONCEPTS IN COMPUTER GRAPHICS

Iridescence

Luminescence;
emission of light by a substance not resulting from heat
Source:Wikipedia

Raytracing / Physically Based

Physically based BRDFs have additional properties:

positivity: $f_r(\omega_i, \omega_o) \geq 0$
obeying Helmholtz reciprocity: fr(ωi,ωo) = fr(ωo,ωi).
conserving energy: $\forall \omega_i, \int_\Omega f_r(\omega_i, \omega_o) \cos\theta_o d\omega_o \leq 1$
BSSRDF
wavelength dependend: λ instead of RGB

→ $f_r(\lambda_i,\omega_i,\lambda_o,\omega_o)$

→ *account for effects such as iridescence or luminescence*

DEMO

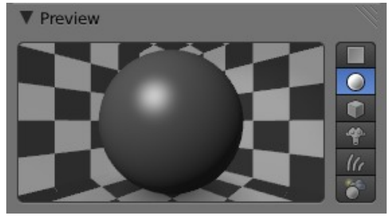
CONCEPTS IN COMPUTER GRAPHICS

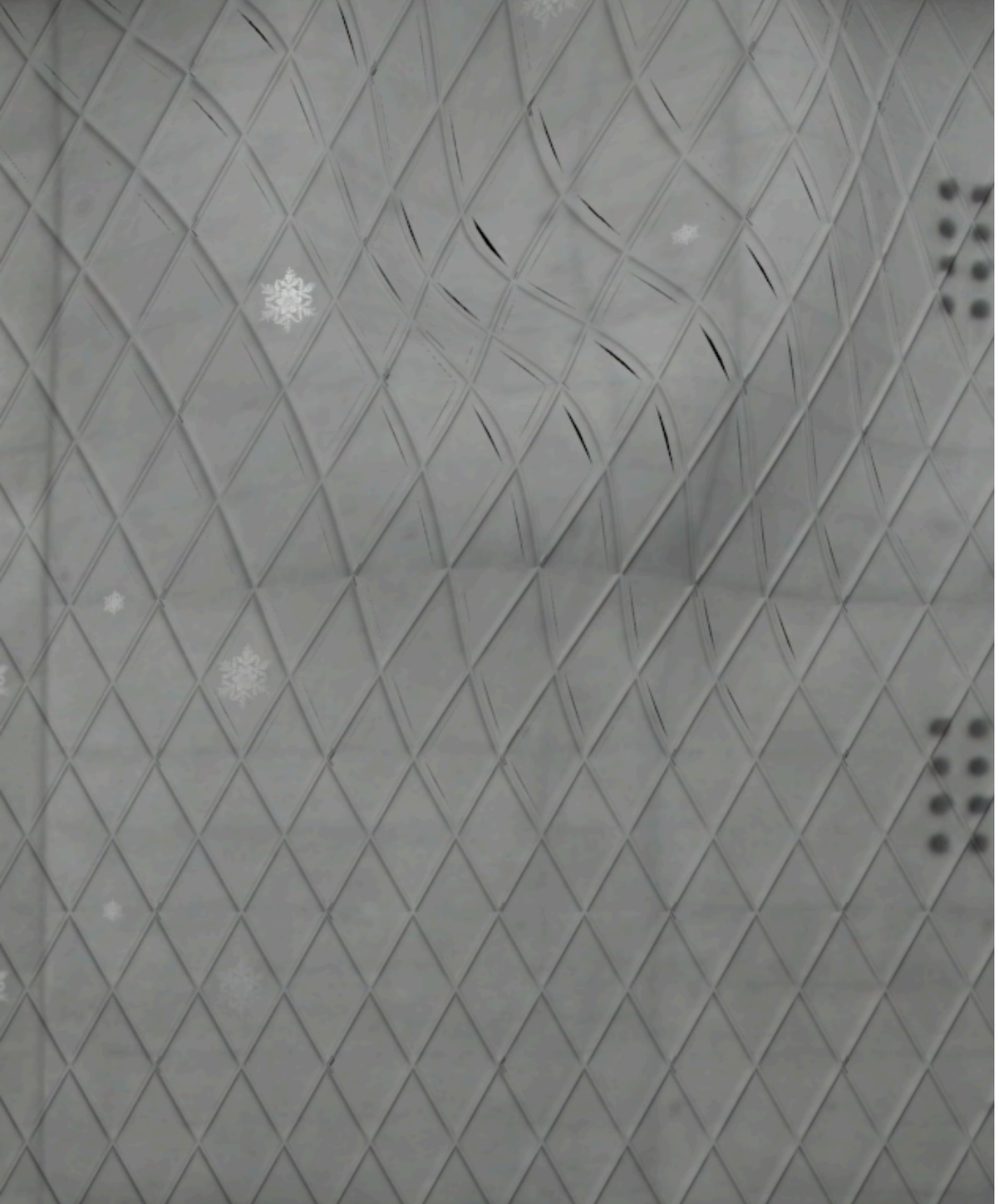
Rasterization
Raytracing / Physically Based
Materials
Level of Detail
Linear Interpolation

CONCEPTS IN COMPUTER GRAPHICS

Materials

- define how light is being reflected/ refracted (think of it as layers):
- diffuse color
- specularity

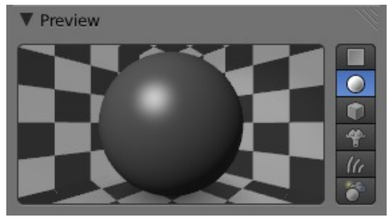




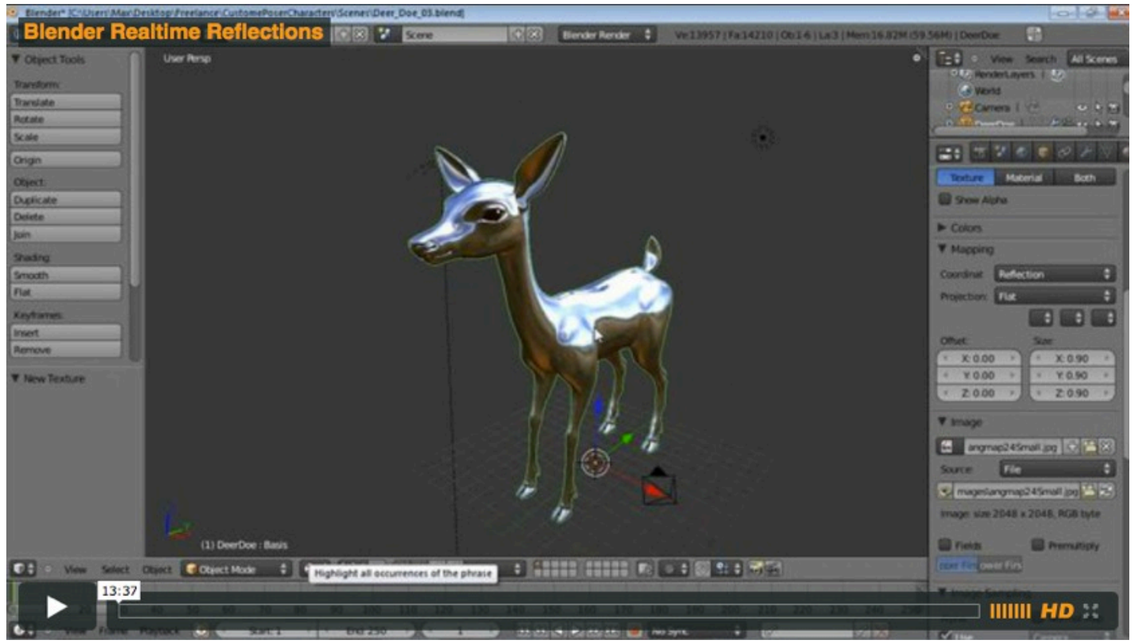
CONCEPTS IN COMPUTER GRAPHICS

Materials

- define how light is being reflected/ refracted (think of it as layers):
- diffuse color
- specularity



CONCEPTS IN COMPUTER GRAPHICS

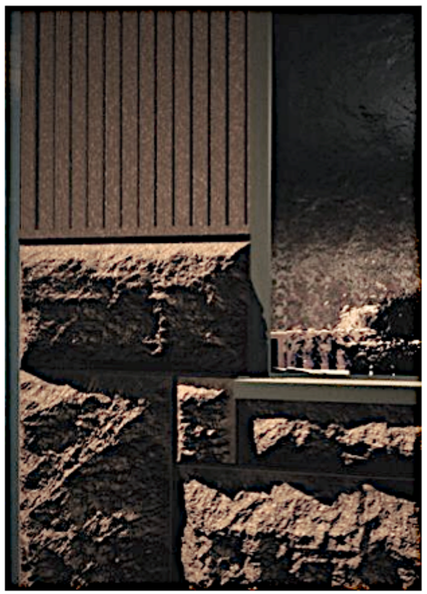


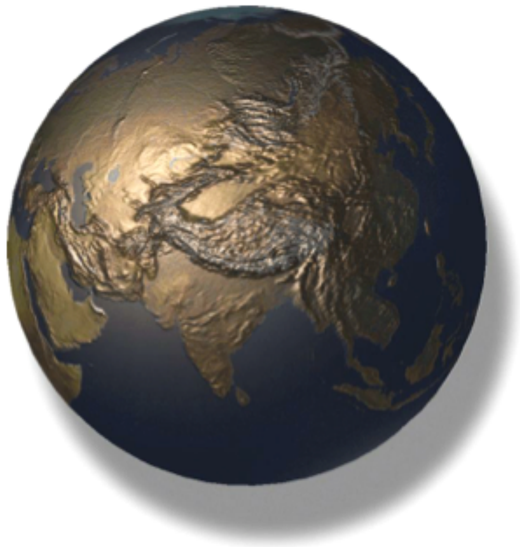
Realtime Reflections using an environmap
Source: McHammond.com

Materials

- define how light is being reflected/ refracted (think of it as layers):
- diffuse color
- specularity
- reflection
-

CONCEPTS IN COMPUTER GRAPHICS





Displacement and Bump Maps
Source: Computer Graphics Lab, ETH Zurich

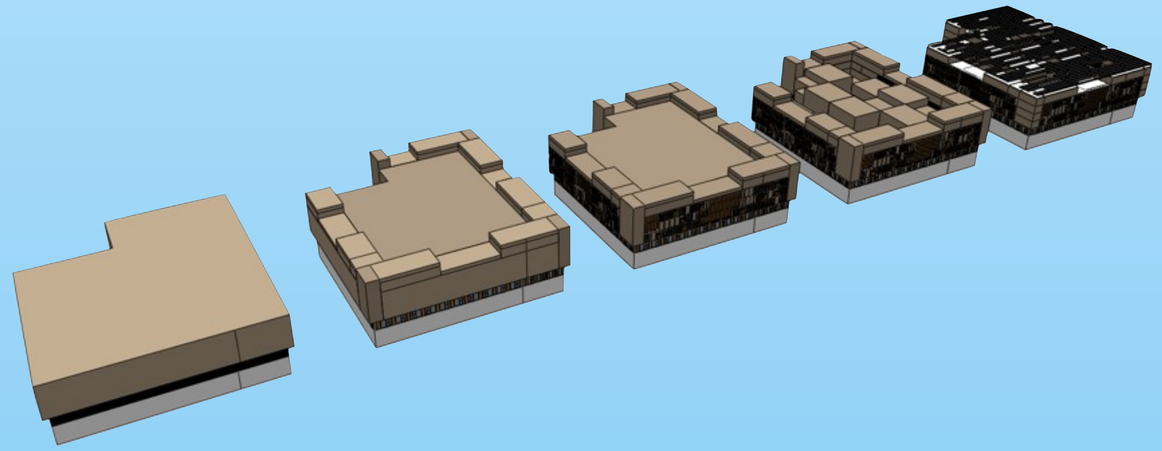
Materials

- define how light is being reflected/ refracted (think of it as layers):
- diffuse color
- specularity
- reflection
- displacement

- **Note:** bump maps have no bumps on shadow silhouette, no self-occlusion, no self-shadowing

CONCEPTS IN COMPUTER GRAPHICS

Rasterization
Raytracing / Physically Based
Materials
Level of Detail
Linear Interpolation



Visualizations for the Masdar Project
Source: Jan Halatsch

Vision for Masdar City
Source: Forster & Partners

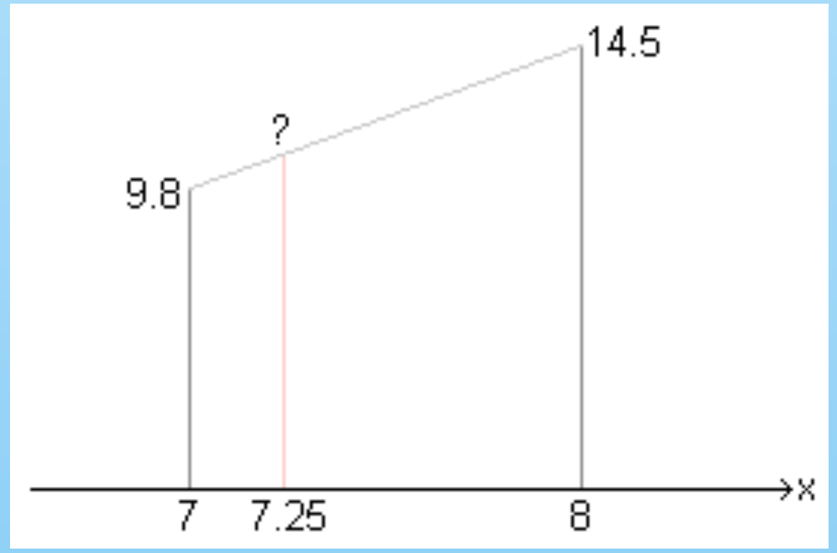
CONCEPTS IN COMPUTER GRAPHICS

Level of Detail, LoD

- by hand (modeling different versions)
- in procedural code
- camera distance
- low LoD when modeling, high LoD when rendering *(RenderMan; Pixar‘s render engine)*
- ***DEMO*** *(Reyn‘s Blog)*

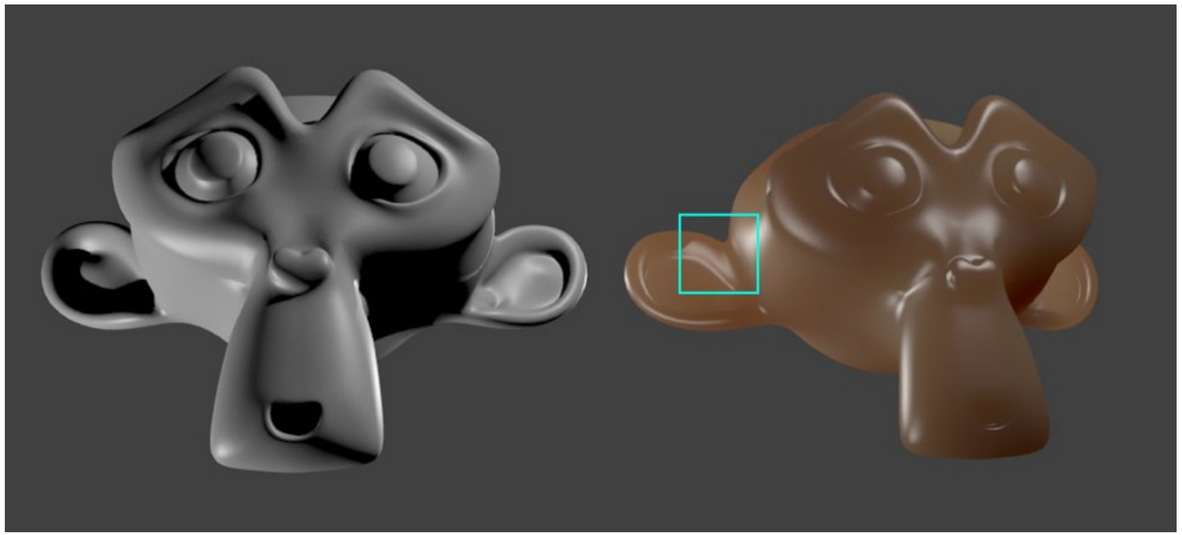
CONCEPTS IN COMPUTER GRAPHICS

Rasterization
Raytracing / Physically Based
Materials
Level of Detail
Linear Interpolation



Linear Interpolation
Source: 3dcenter.org

CONCEPTS IN COMPUTER GRAPHICS

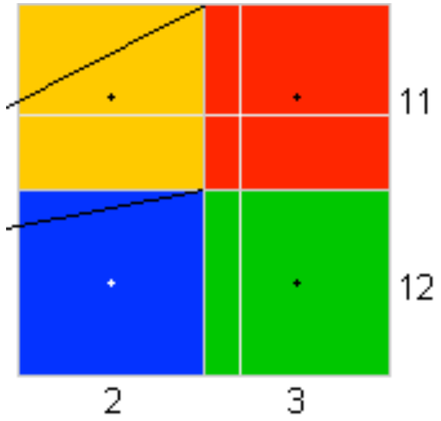


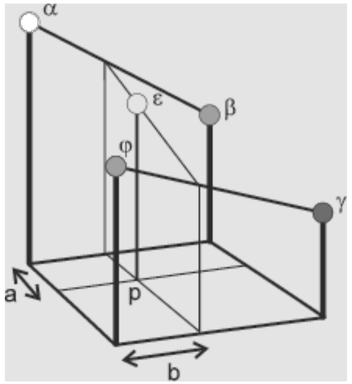
Linear Interpolation

Point Sampling (without linear interpolation)

Linear Interpolation

- scale an image

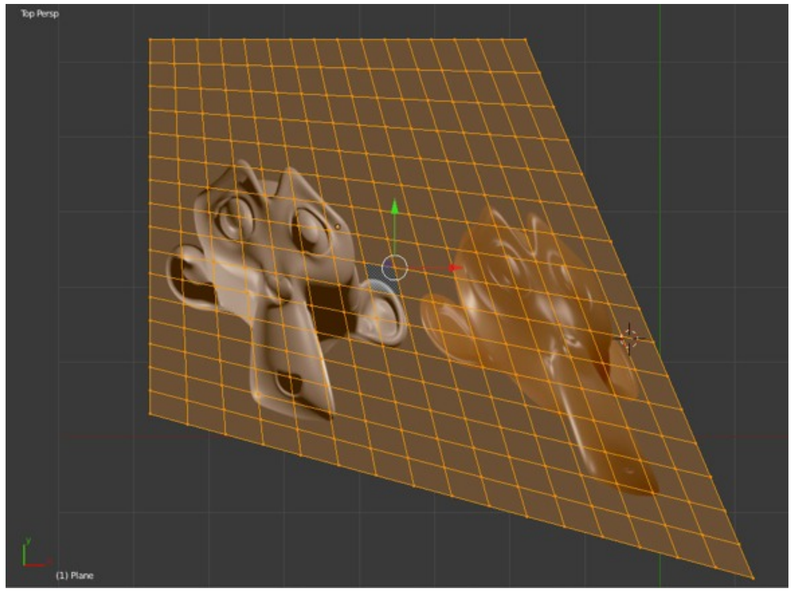




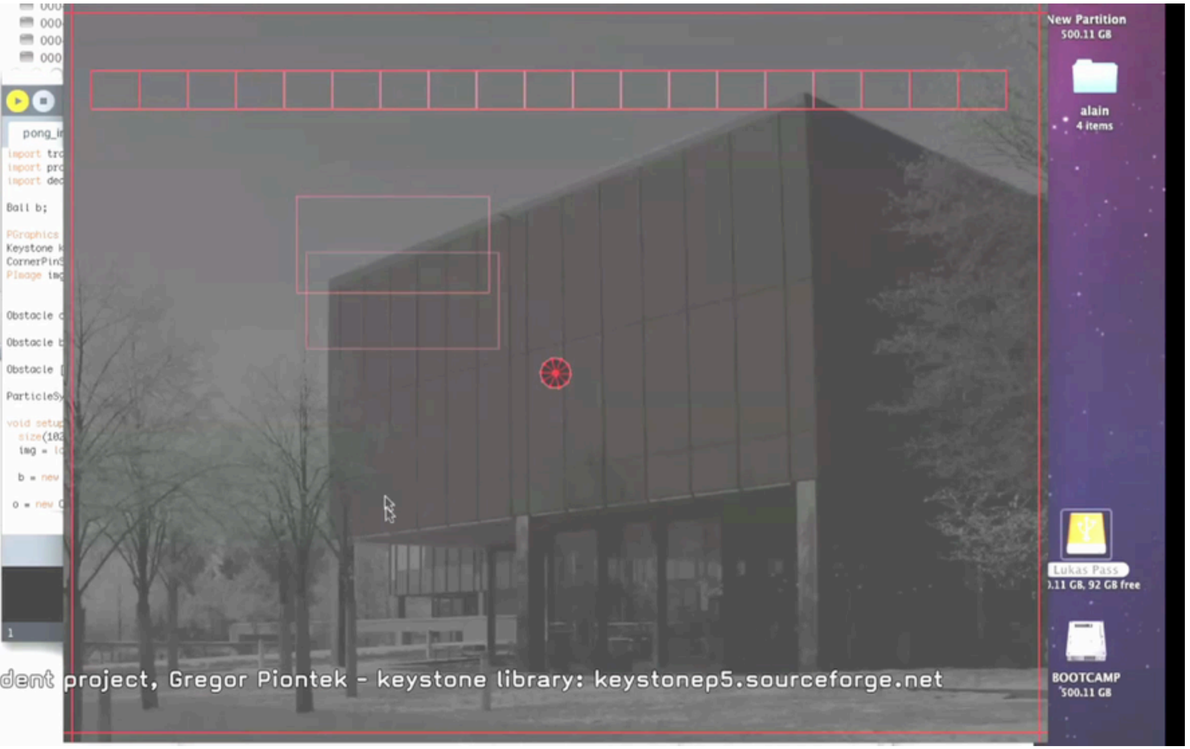
CONCEPTS IN COMPUTER GRAPHICS

Linear Interpolation

- warp an image / texture

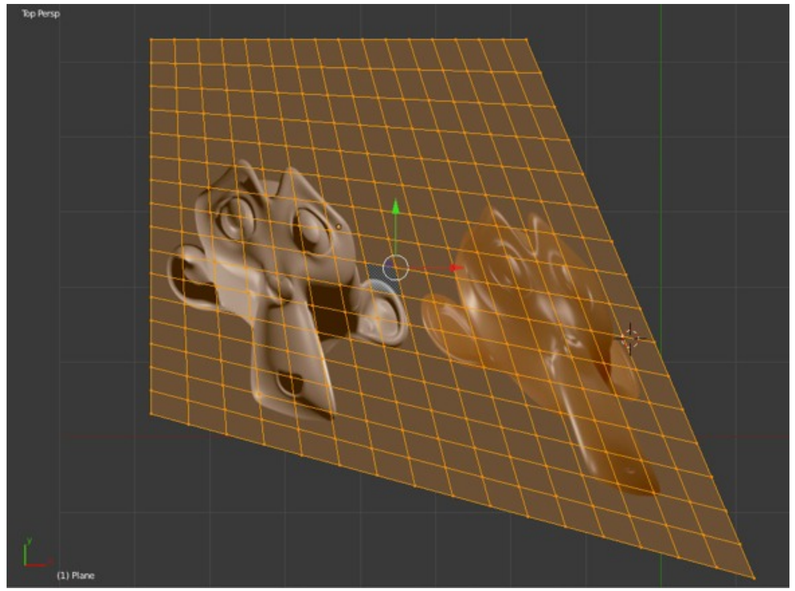


CONCEPTS IN COMPUTER GRAPHICS

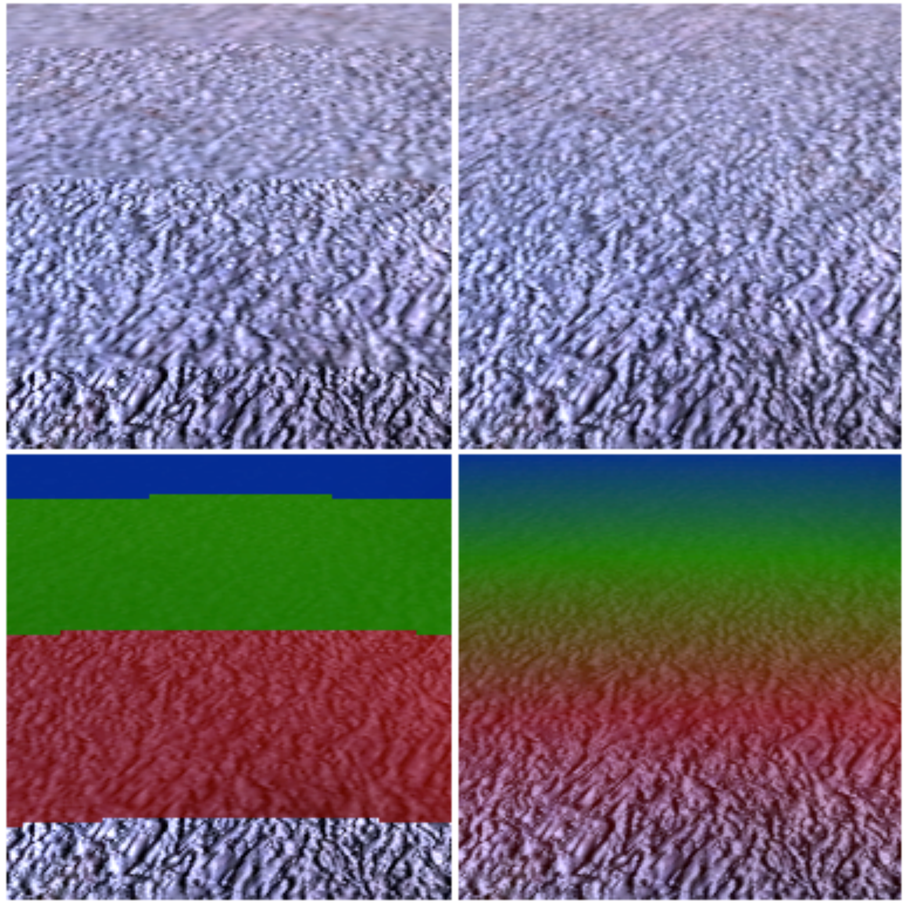


Linear Interpolation

- warp an image / texture



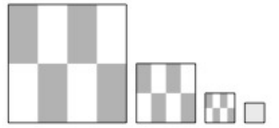
bilinear trilinear



CONCEPTS IN COMPUTER GRAPHICS

Linear Interpolation

- scale a texture



MIP Mapping
(same texture in different sizes)

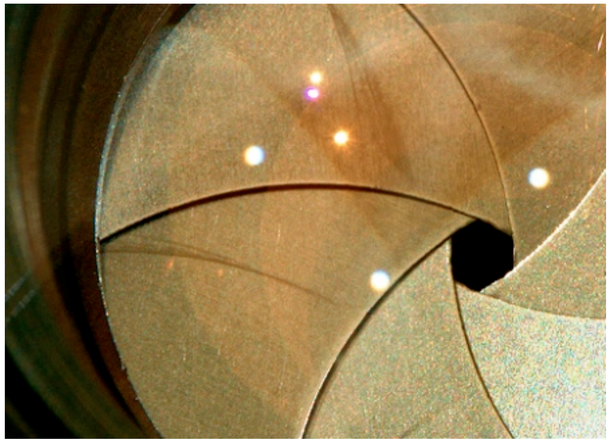
COMPUTER GRAPHICS

- Computer Vision
- Concepts in Computer Graphics
  - Rasterization
  - Raytracing / Physically Based Materials
  - Level of Detail
  - Linear Interpolation
- **Photography**
  - DOF
  - Image Design / Shift

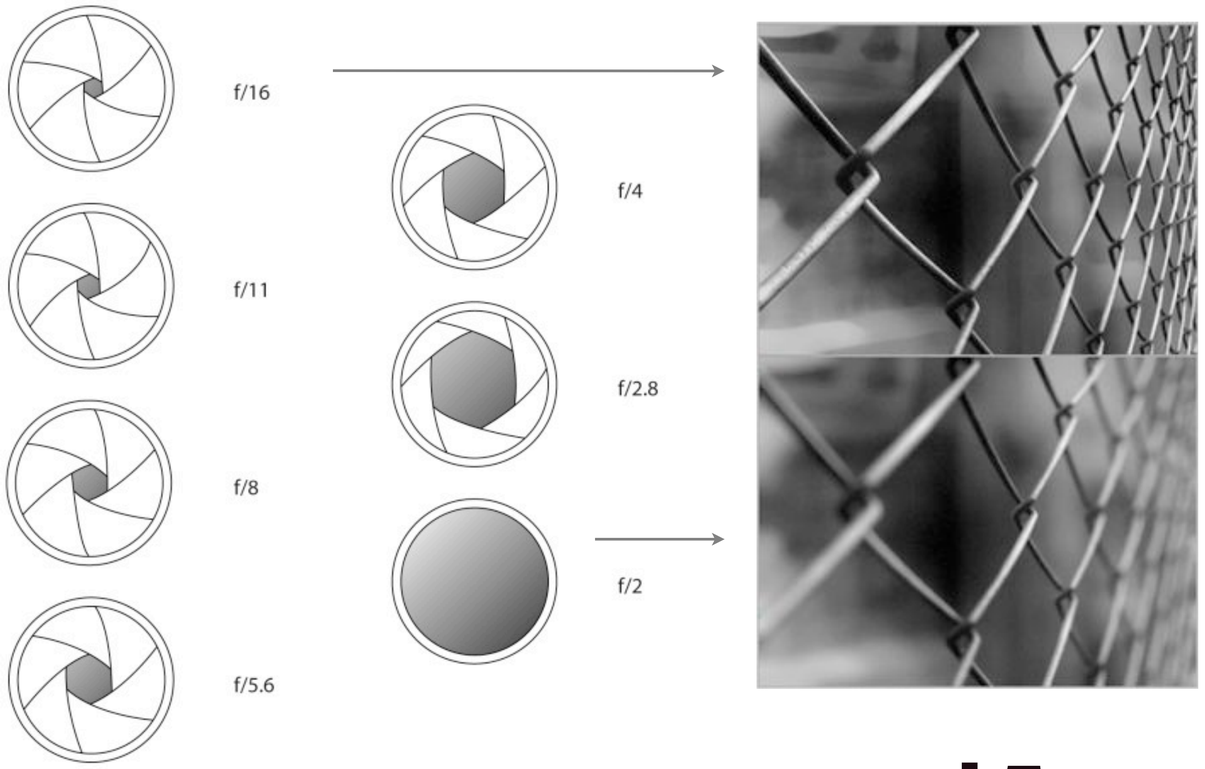
COMPUTER GRAPHICS

Photography

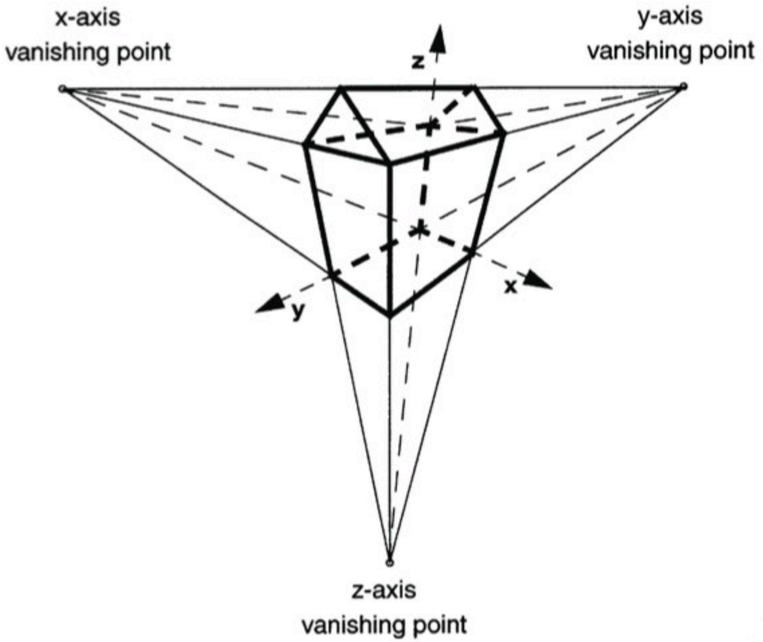
DOF - Depth of Field



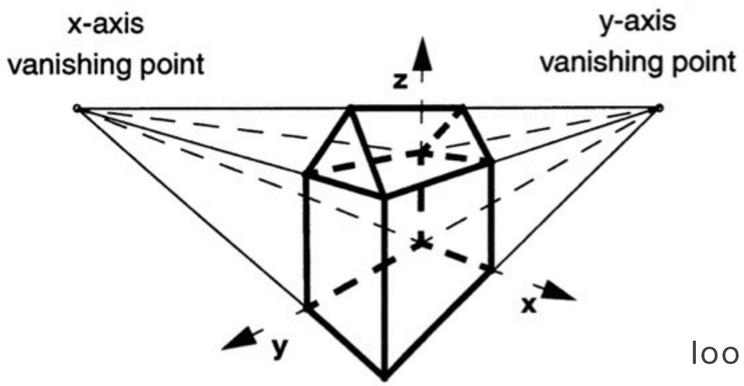
Aperture



COMPUTER GRAPHICS



real world



looking more solid

3 and 2 vanishing points
Source: Computer Graphics Laboratory, ETH Zurich

Photography

DOF - Depth of Field

Image Design / Shift

- choose a nice angle (up to artist)
- tradition in architectural photography to fake 2 point perspective with a shift lens

COMPUTER GRAPHICS

Computer Vision
Concepts in Computer Graphics
- Rasterization
- Raytracing / Physically Based Materials
- Level of Detail
- Linear Interpolation

Photography
- DOF
- Image Design / Shift

Light
- 3 Point Light
- Studio Lights

Post Processing *(Photoshop/Composition)*

COMPUTER GRAPHICS

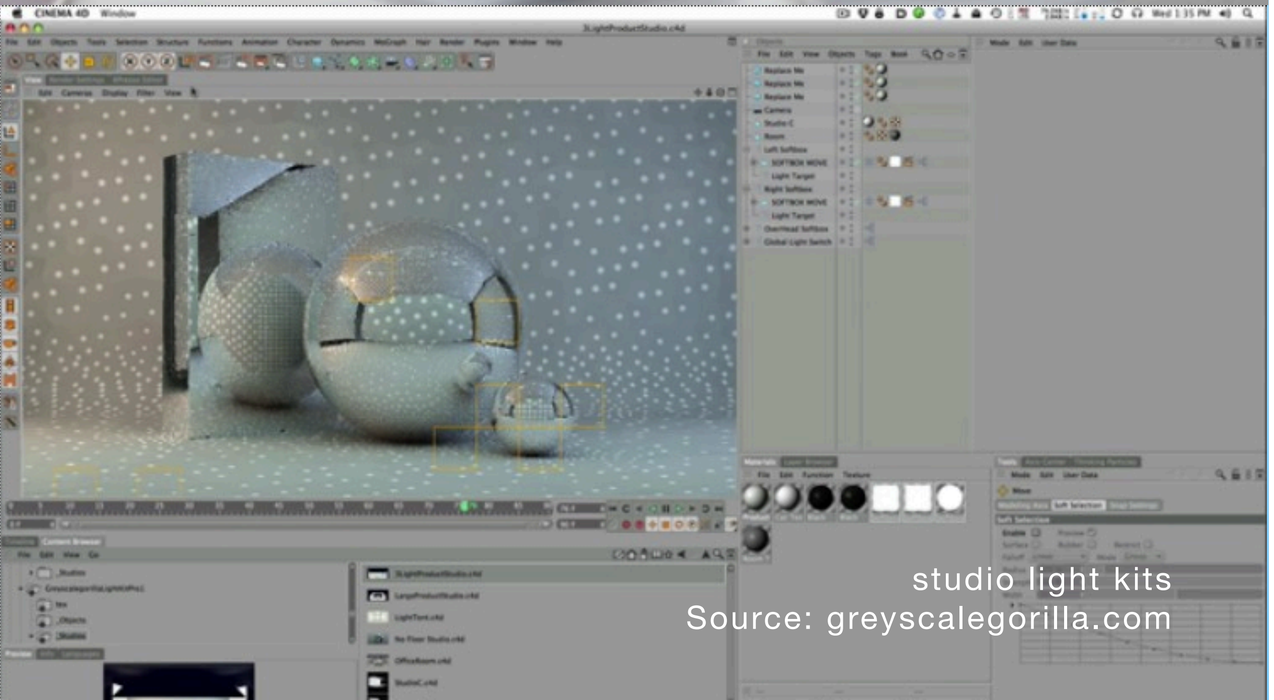
3 point lighting
Source: Wikipedia

Light

3 Point Light

- key-, fill-, backlight
- industry standard
- for use with low power machine, point lights are very time efficient

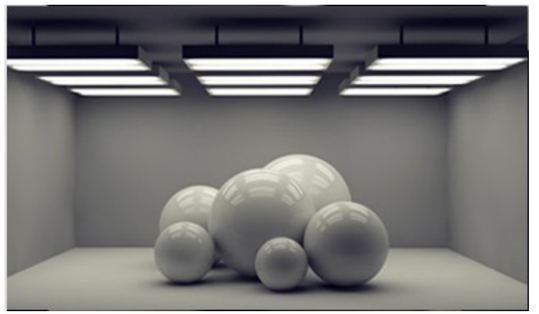
COMPUTER GRAPHICS



studio light kits
Source: greyscalegorilla.com

Light

3 Point Light
Studio Lights



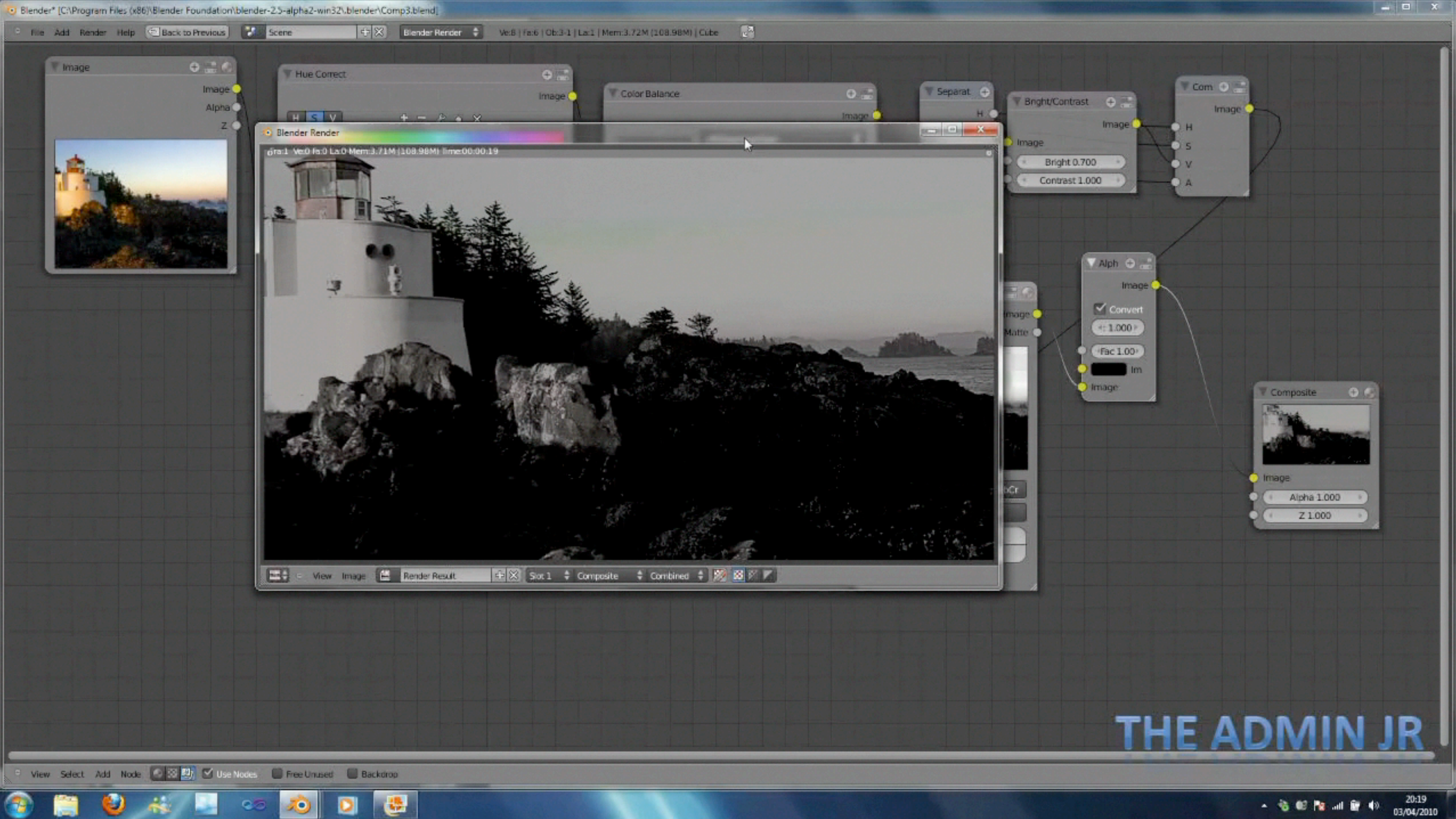
COMPUTER GRAPHICS

- Computer Vision
- Concepts in Computer Graphics
  - Rasterization
  - Raytracing / Physically Based Materials
  - Level of Detail
  - Linear Interpolation
- Photography
  - DOF
  - Image Design / Shift
- Light
  - 3 Point Light
  - Studio Lights
- **Post Processing** ***(Photoshop/Composition)***

COMPUTER GRAPHICS

Post Processing ***(Photoshop/Composition)***

- non destructive methods
- filters/adjustments as a program being applied to every frame


Blender* [C:\Program Files (x86)\Blender Foundation\blender-2.5-alpha2-win32\.blender\Comp3.blend]
File Add Render Help Back to Previous Scene Blender Render
Image
Image
Alpha
Z
Hue Correct
Image
H S V
Color Balance
Image
Separat
H
Bright/Contrast
Image
Image
Bright 0.700
Contrast 1.000
Com
Image
H
S
V
A
Blender Render
Time:00:00.19
Image
Matte
Alph
Image
Convert
1.000
Fac 1.00
Im
Image
Composite
Image
Alpha 1.000
Z 1.000
View Image Render Result Slot 1 Composite Combined
View Select Add Node Use Nodes Free Unused Backdrop
THE ADMIN JR
20:19
03/04/2010

Source: greyscalegorilla.com

THANK YOU!

Chair for Information Architecture