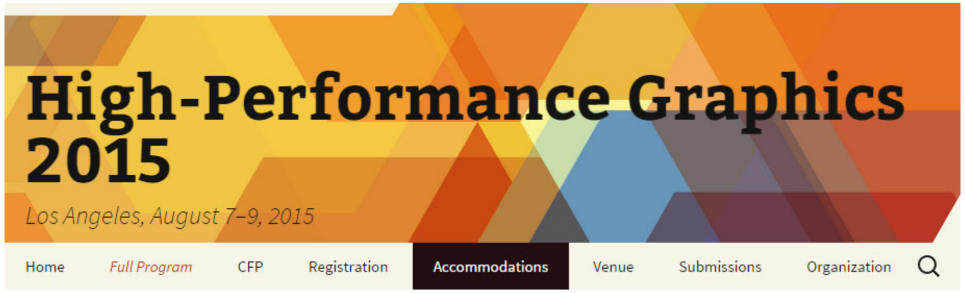
CS380: Computer Graphics
Introduction

Sung-Eui Yoon
(윤성의)

Course URL:
http://sgvr.kaist.ac.kr/~sungeui/CG

About the Instructor

- **Notable recognitions**
  - **Co-chairs at ACM Symp. on Interactive 3D Graphics and Games**
  - **Test-of-time award at High Performance Graphics**



High-Performance Graphics 2015

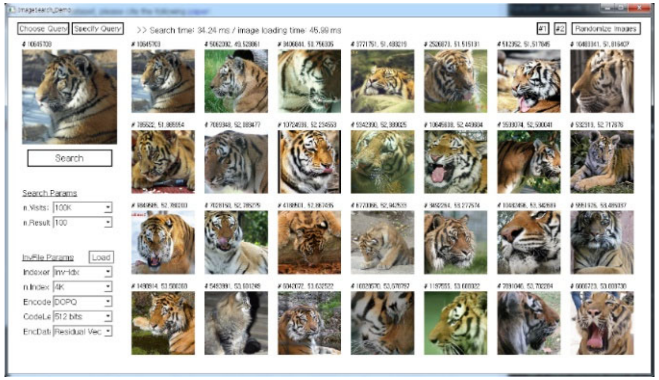
Los Angeles, August 7–9, 2015

  - **Interns/post.doc/collaborations at Disney, Adobe, AMD, Pixar**
  - **Produced two professors on rendering (GIST) and related topics**

Research: Intelligent Ray Tracing, Image Search, Motion Planning

- Designing ***scalable and intelligent graphics and geometric algorithms*** to efficiently handle massive models

Photo-realistic rendering



>> Search time: 34.24 ms / image loading time: 45.99 ms

Search Params

n.Visits: 100K

n.Result: 100

InvFile Params

Indexer: Inv-Idx

n.Index: 4K

Encode: DOPQ

CodeLe: 512 bits

EncDat: Residual Vec

Image search

Motion planning

Paper and video: https://sgvr.kaist.ac.kr/category/papers/paper-international/

YouTube videos: http://www.youtube.com/user/sglabkaist

Course Information of CS380

Instructor: **Sung-eui Yoon**

Email: **sungeui@kaist.edu**

Office: **3432 at CS building**

Office hours: **~~Right after class time~~ (or by appt.)**

KLMS discussion page:

- **Use this one for sharing Q&A with other students, instead of personal communication (e.g., email) to TAs**

KLMS: homework submissions

Course webpage: http://sglab.kaist.ac.kr/~sungeui/CG/

Class Time

- Date: every MW
  - Time: 2:30pm ~ 3:45pm
  - Youtube video + zoom sessions due to too many students
- 4 credit course
  - OpenGL courses or some talks will be given by TAs

TAs

- TA email address: cs380ta@gmail.com
  - Use KLMS board first

- JaeYoon Kim (김재윤), KyuBeom Han (한규범), SuHyeon Ha (하수현), Jumin Lee (이주민), YoungJu Na (나영주)
  - Office: E3-1, 3443호

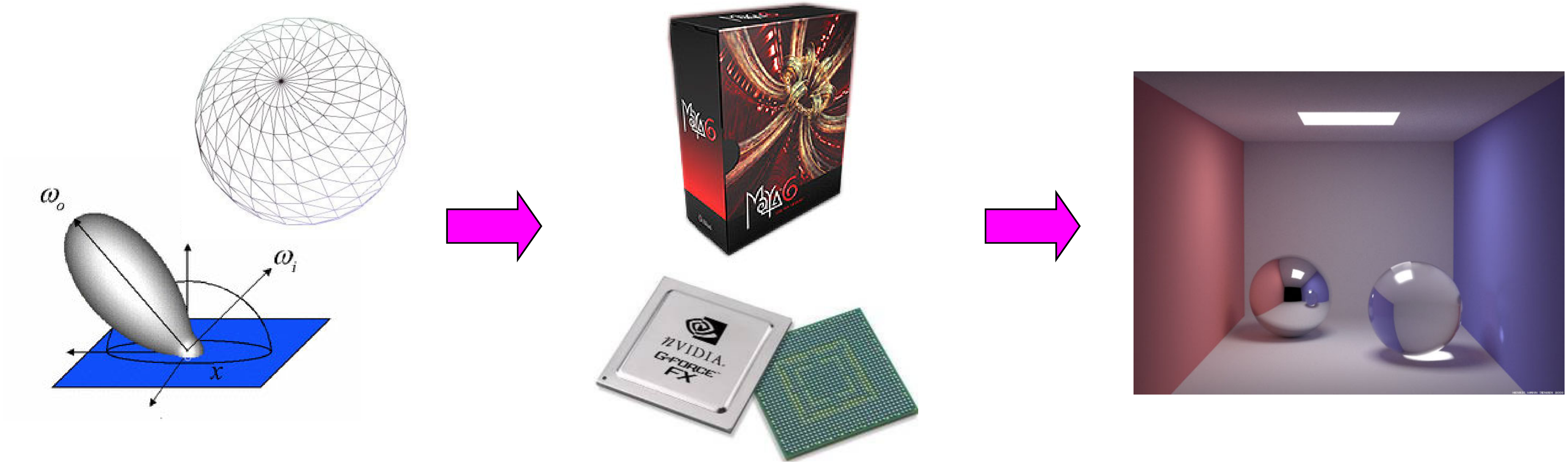
- TaeYeon Kim (김태연)
  - Office: E-1, 3446호

Prerequisites

- Basic knowledge of linear algebra
  - E.g., matrix multiplication and inversion
- Some level of programming skill
  - Require you to know or self-study C-like language (e.g., C and C++)
- If you are unsure, consult the instructor at the end of this class
  - You can check the programming assignments of the prior homepage

Overview

- We will discuss various parts of computer graphics



Modelling

Simulation & Rendering

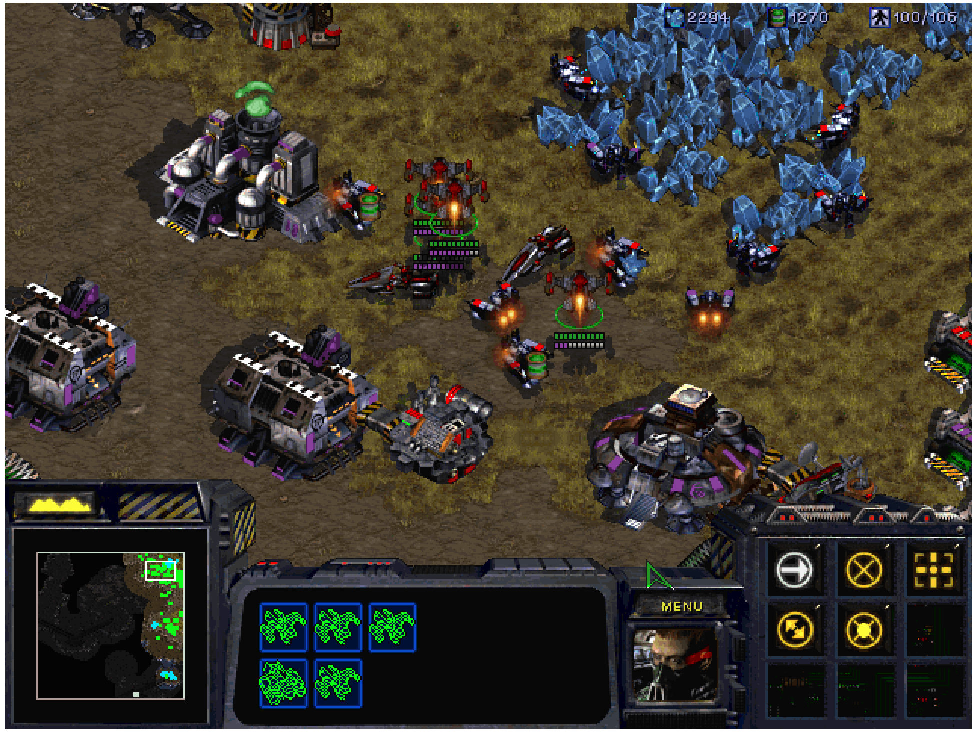
Image

Computer vision inverts the process
Image processing deals with images

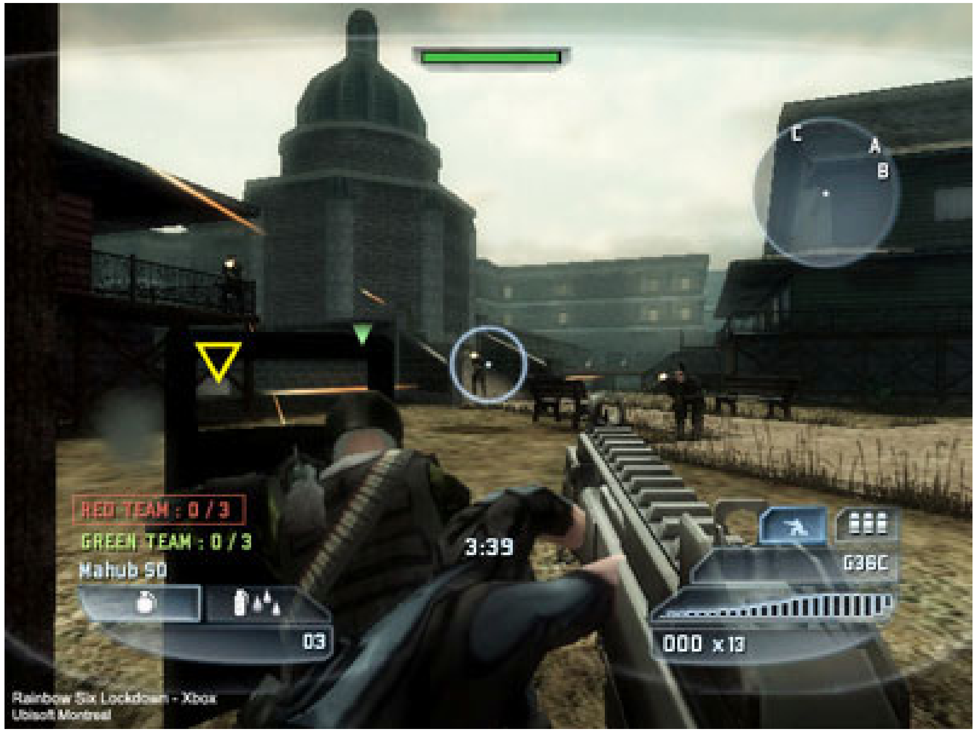
Application of Computer Graphics

- Games
- Movies and film special effects
- Product design and analysis
- Medical applications
- Scientific visualization

Games



2D game

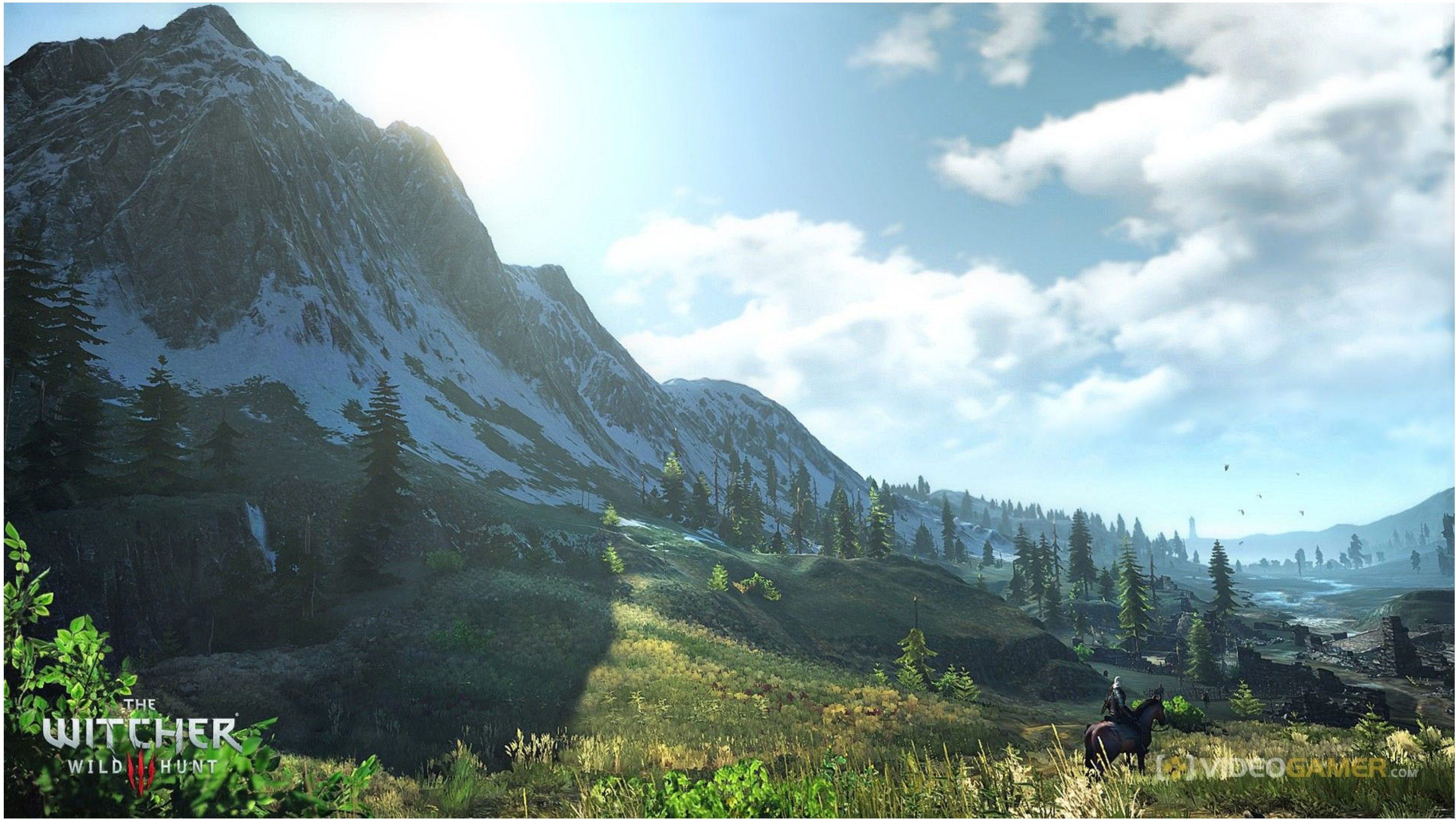


Rainbow Six Lockdown - Xbox
Ubisoft Montreal

3D shooting game

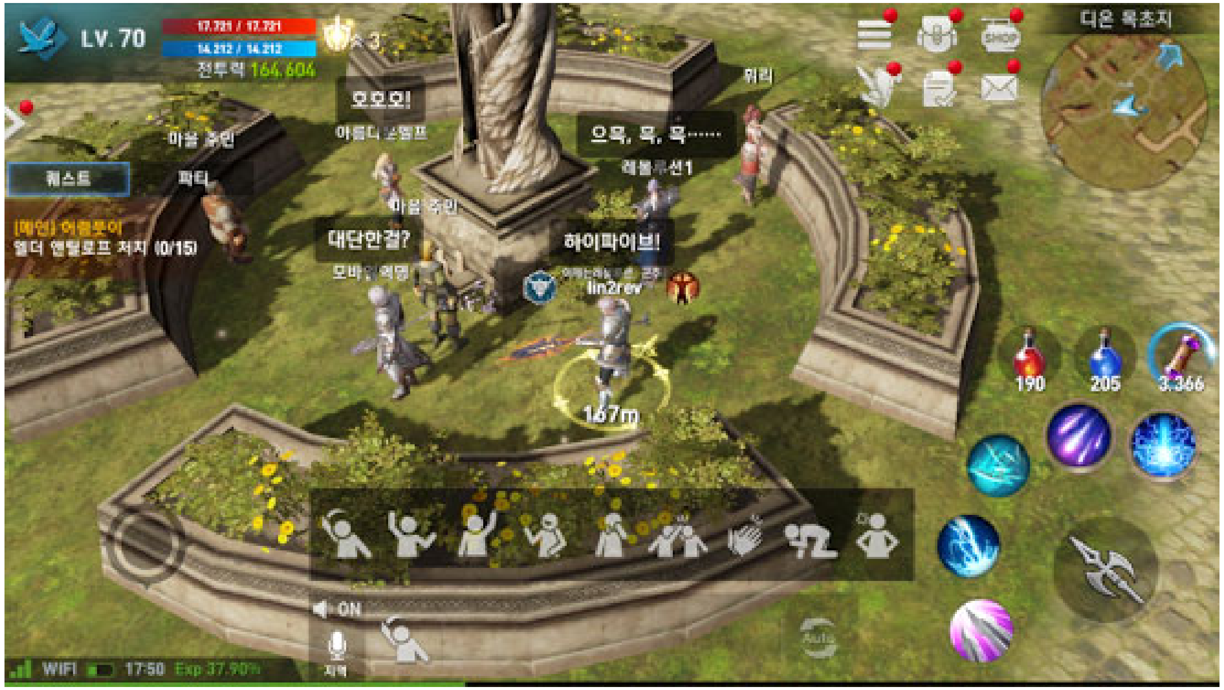
Large-Scale Open World w/ High Quality Rendering

- Witcher 3
  - Used its own engine

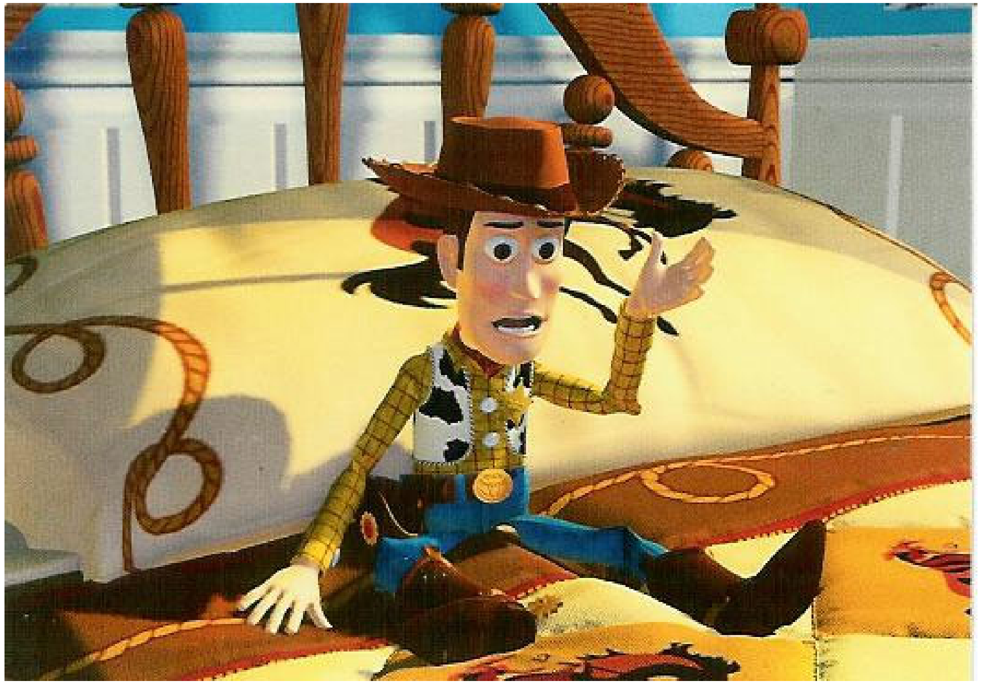


High Quality Mobile Games

- Big game industry at Korea
- Lineage 2 – Revolution
  - Based on Unreal engine



Movies and Film Special Effects



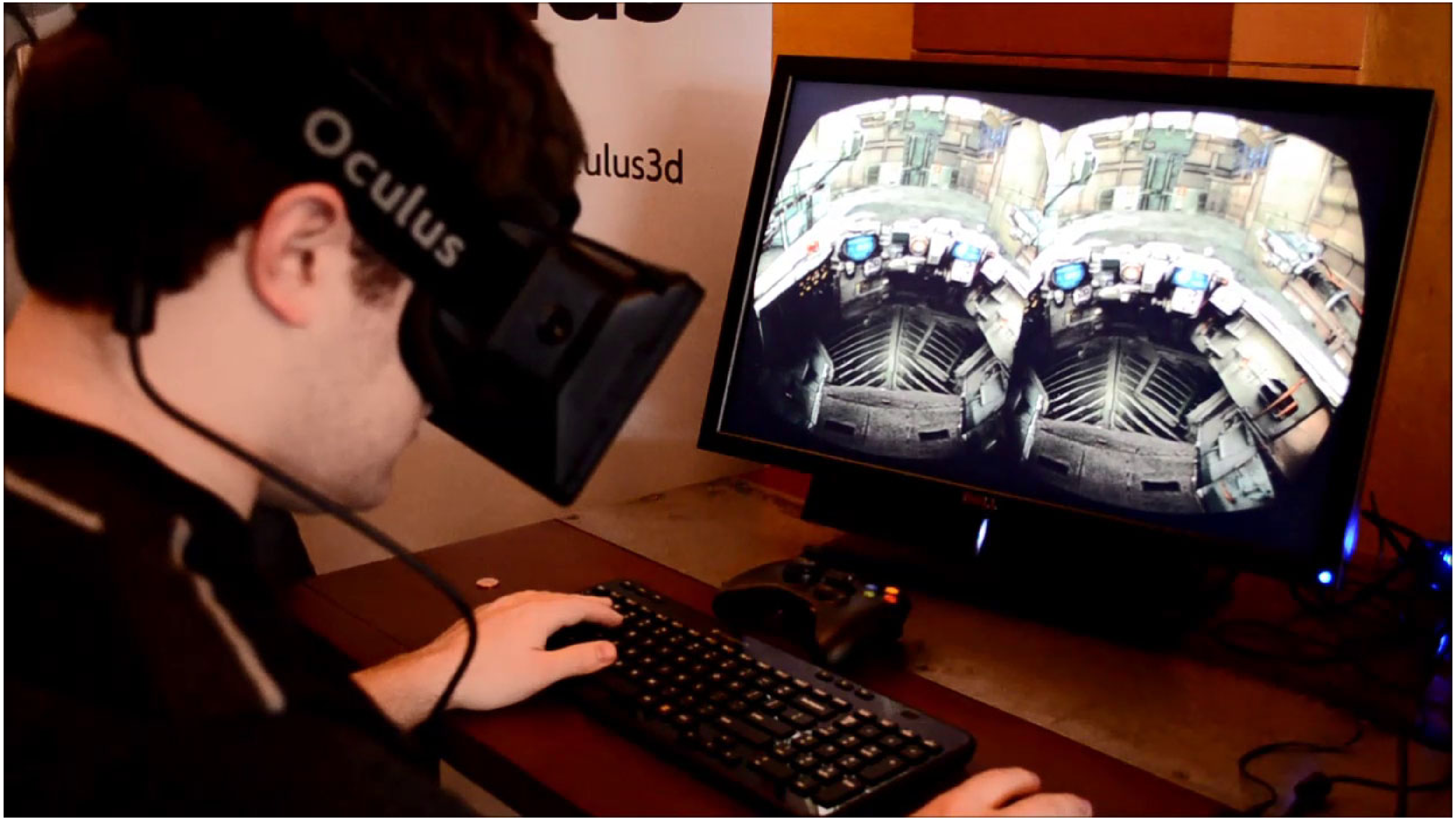
Toy story

Matrix

3D Movies

Avatar

Head-Mounted Display (HMD) for VR



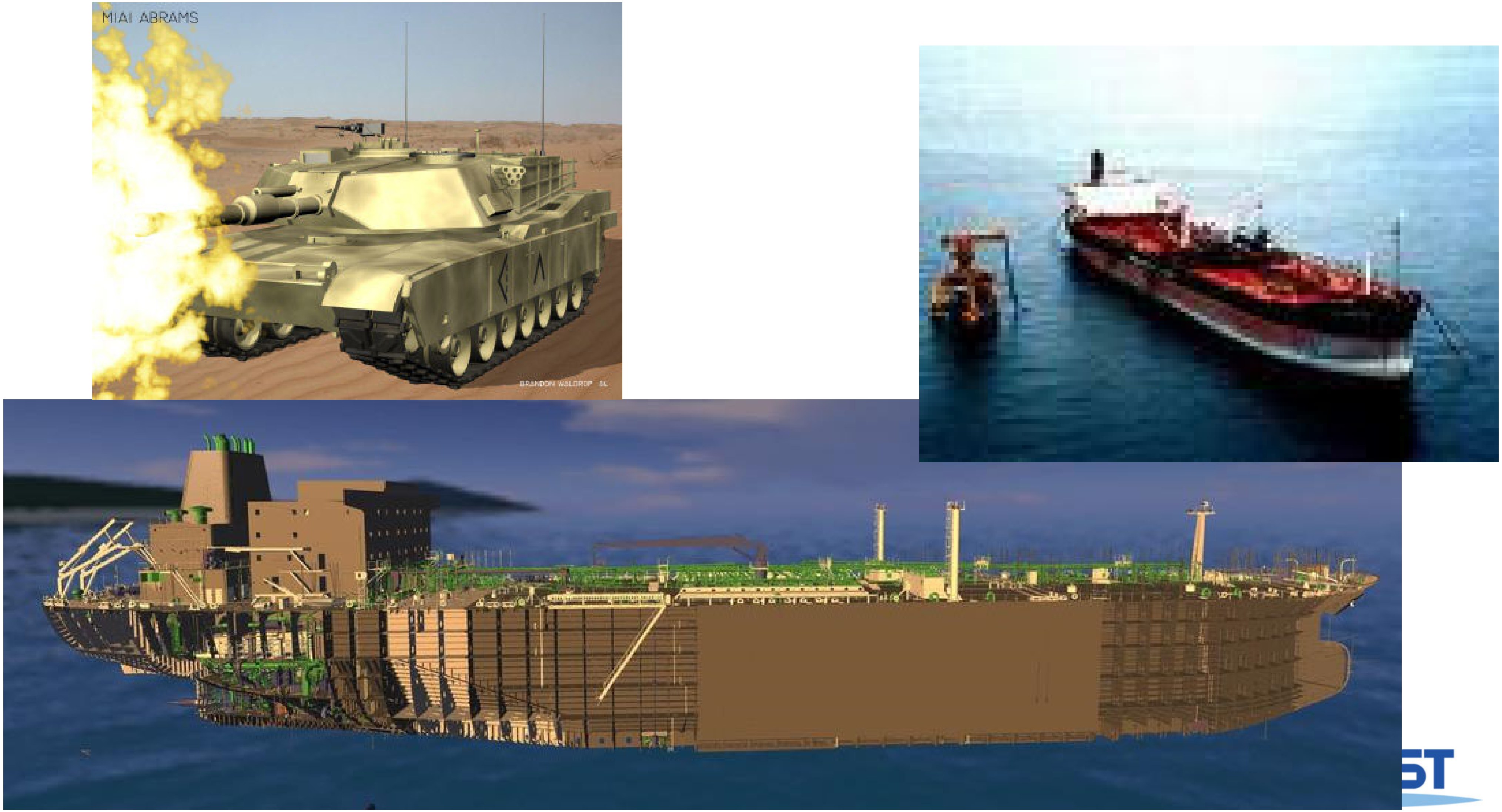
Spatial Computing supported by Apple

Ack: Techopedia

Interesting App. For Augmented Reality (AR)

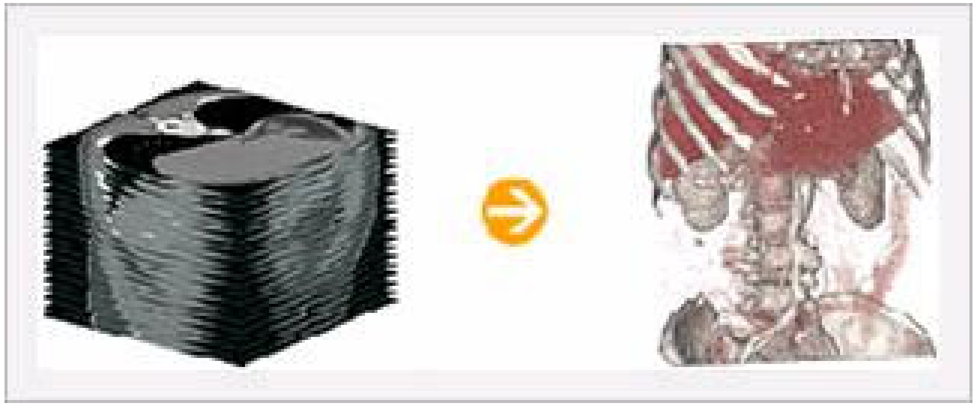
Product Design and Analysis

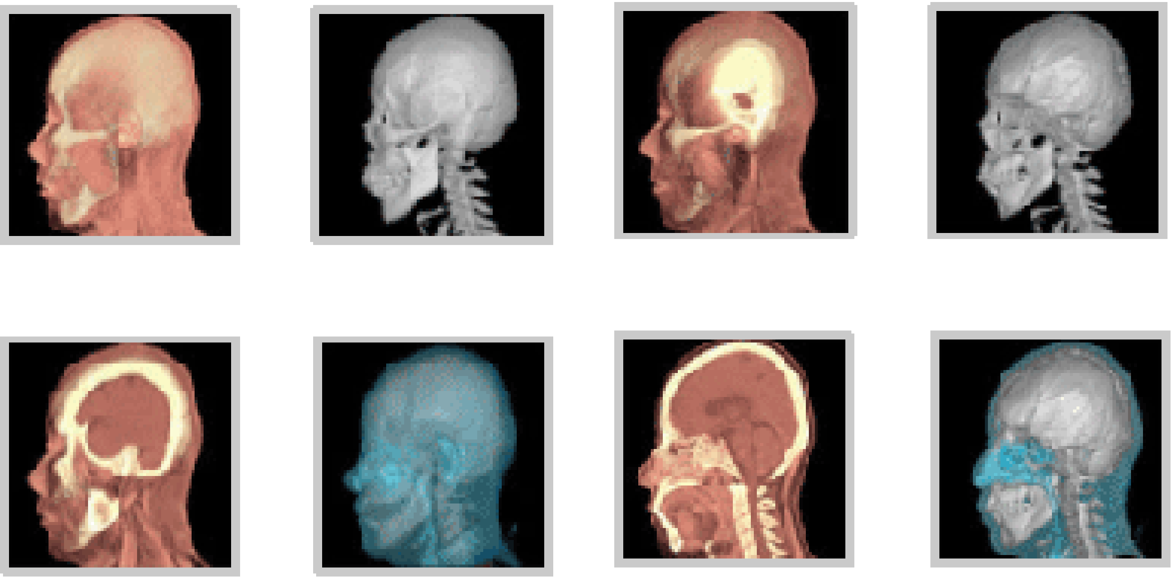
- **Computer-aided design (CAD)**



Medical Applications

- **Visualizing data of CT, MRI, etc**

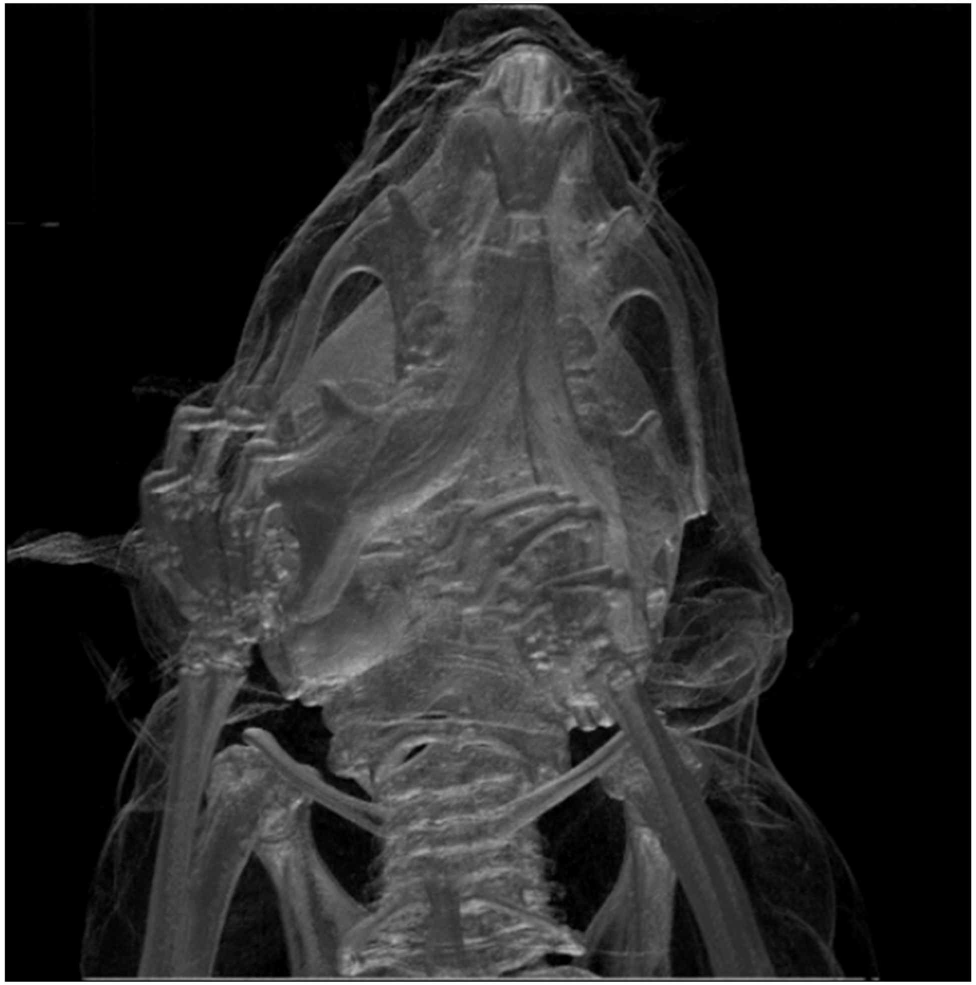




Rapidia homepage

Medical Applications

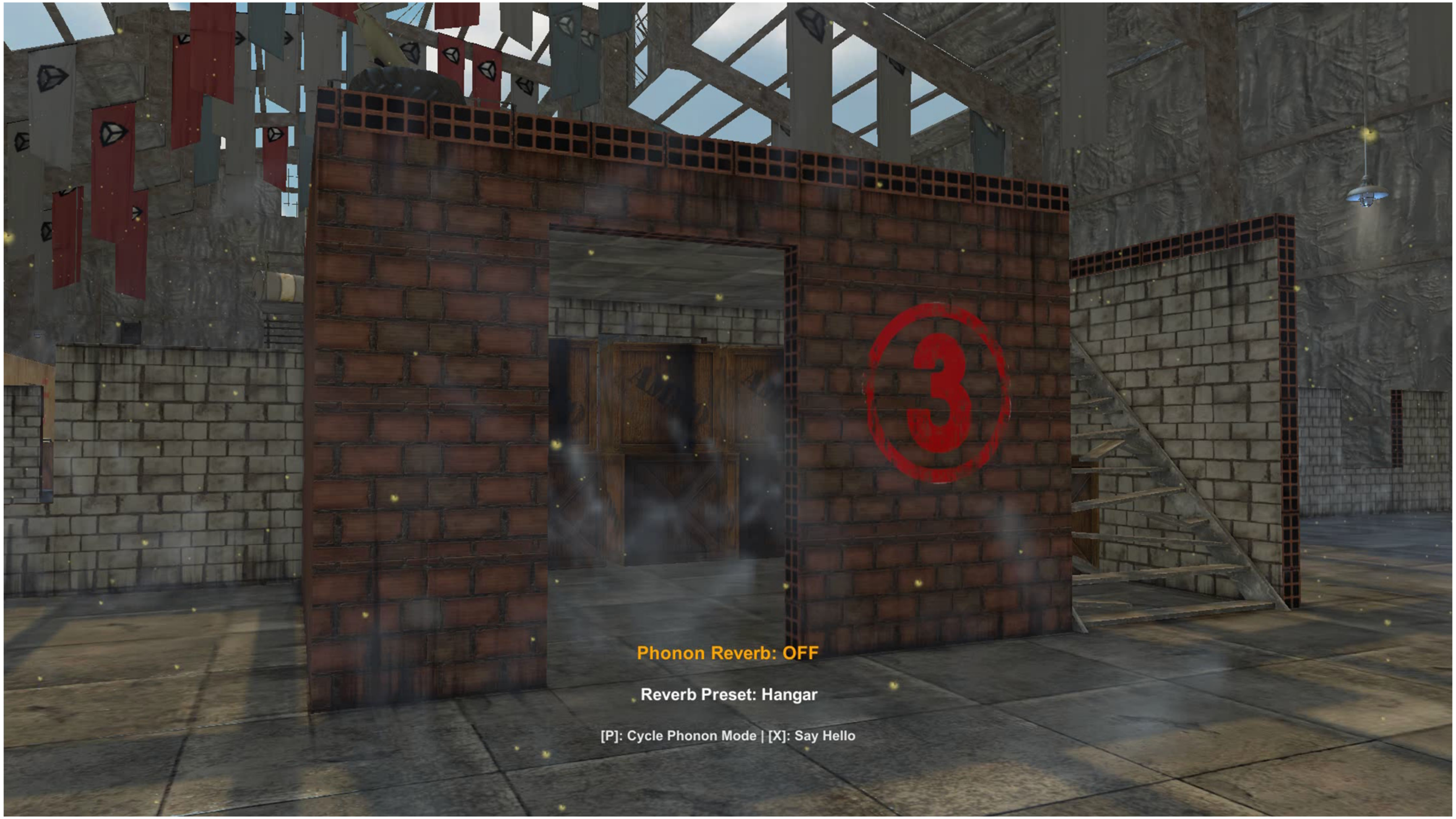
- **Visualizing data of CT, MRI, etc**



Wikipedia

Mouse skull (CT)

Sound Rendering



Phonon Reverb: OFF

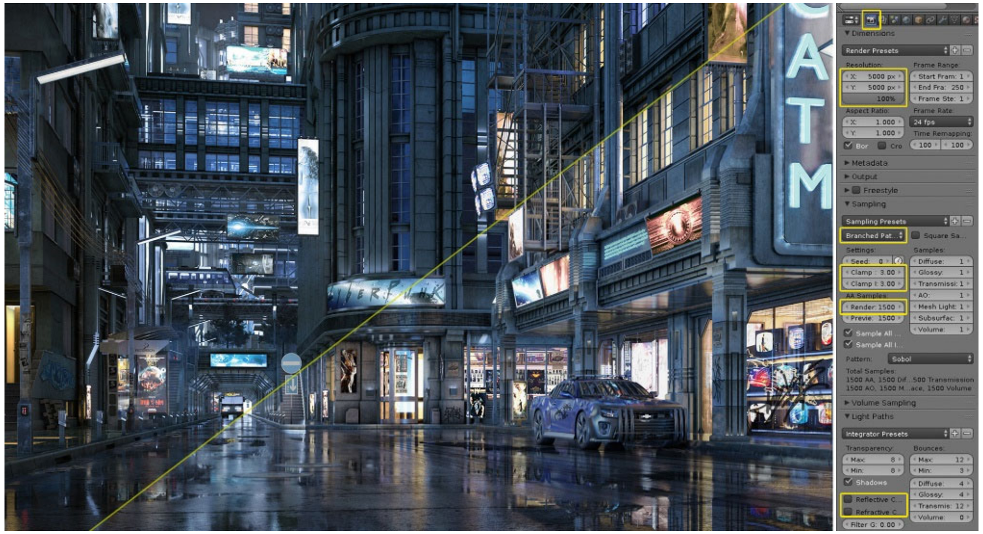
Reverb Preset: Hangar

[P]: Cycle Phonon Mode | [X]: Say Hello

Realistic Data Generation for Deep Learning

- **Apply CG techniques for generating realistic data for deep learning, which require lots of data**

GPU



```
▼ Dimensions
Render Presets
Resolution:          Frame Range:
X:  5000 px          Start Fram: 1
Y:  5000 px          End Fra:  250
    100%             Frame Ste: 1
Aspect Ratio:        Frame Rate:
X:  1.000            24 fps
Y:  1.000            Time Remapping:
☑ Bor  □ Cro         100   100
► Metadata
► Output
► □ Freestyle
▼ Sampling
Sampling Presets
Branched Pat...      □ Square Sa...
Settings:            Samples:
Seed:    0           Diffuse:      1
Clamp :  3.00        Glossy:       1
Clamp I: 3.00        Transmissi:   1
AA Samples:          AO:           1
Render: 1500         Mesh Light:   1
Previe:  1500        Subsurfac:    1
☑ Sample All ...     Volume:       1
☑ Sample All I...
Pattern:   Sobol
Total Samples:
1500 AA, 1500 Dif...500 Transmission
1500 AO, 1500 M...ace, 1500 Volume
► Volume Sampling
▼ Light Paths
Integrator Presets
Transparency:        Bounces:
Max:   8             Max:         12
Min:   8             Min:          3
☑ Shadows            Diffuse:      4
□ Reflective C...    Glossy:       4
□ Refractive C...    Transmis:    12
Filter G: 0.00       Volume:       0
```

Realistic modeling, rendering & simulation

Training data for learning

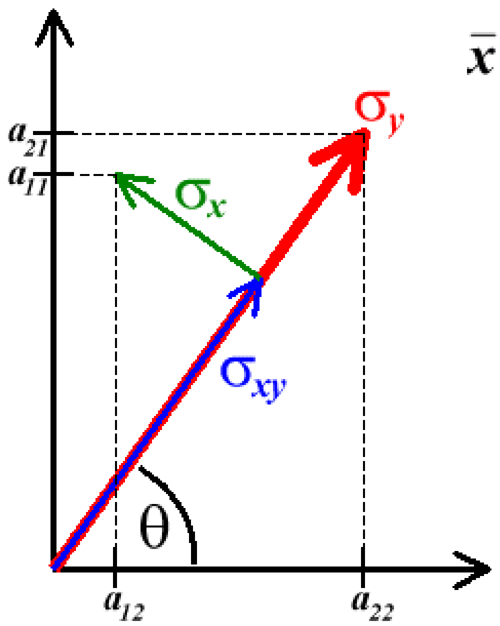
Topics

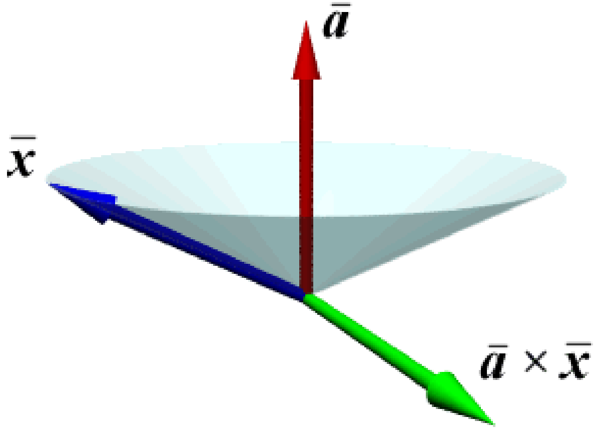
- Mathematical tools
- 3D models and interaction
- Hidden surface removal
- Rasterization
- Lighting and shading
- Shadows
- Texture mapping
- Ray tracing
- Global illumination
- Curves and surfaces
- Simplification and levels of detail
- Collision detection
- Graphics hardware, etc
- Some recent technology: AI based approaches Nerf, etc.

Mathematical Tools

- Homogeneous coordinates
- Vectors
- Planes
- Frames
- Transformations

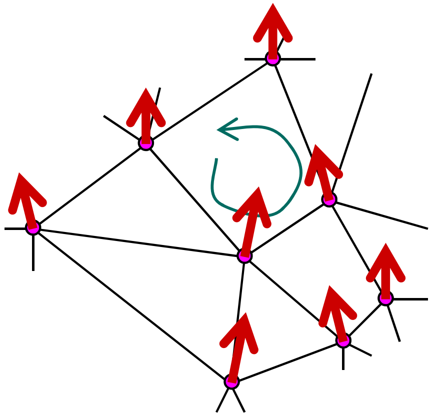
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

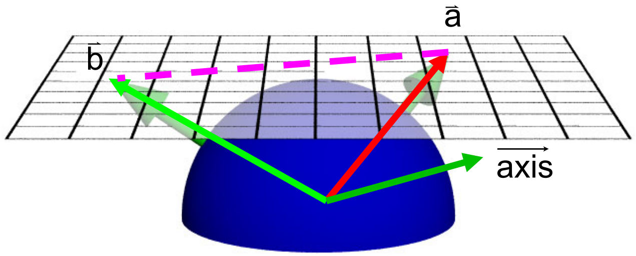




3D Models and Interaction

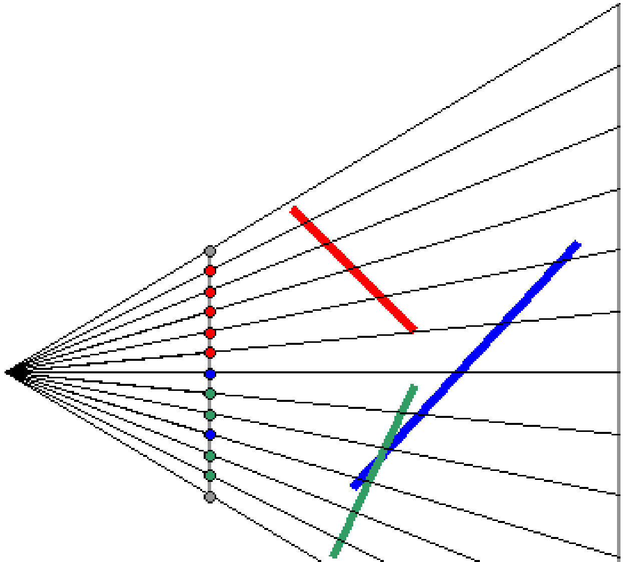
- Loading and view models
- Picking and selection
- Modeling a trackball
- Virtual reality (VR) is all about interaction

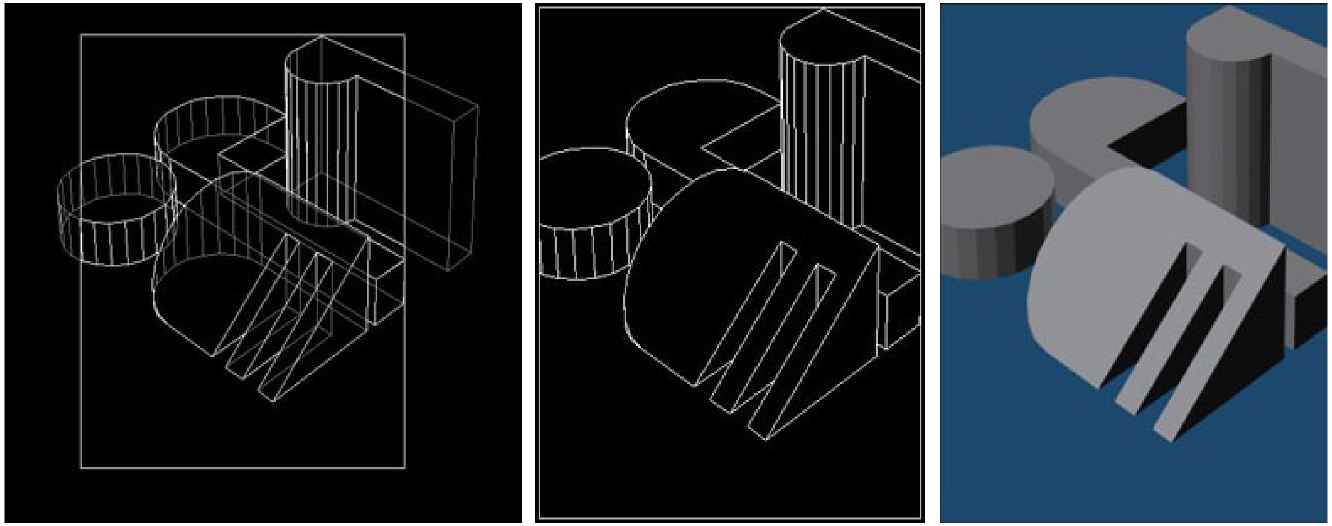




Hidden Surface Removal

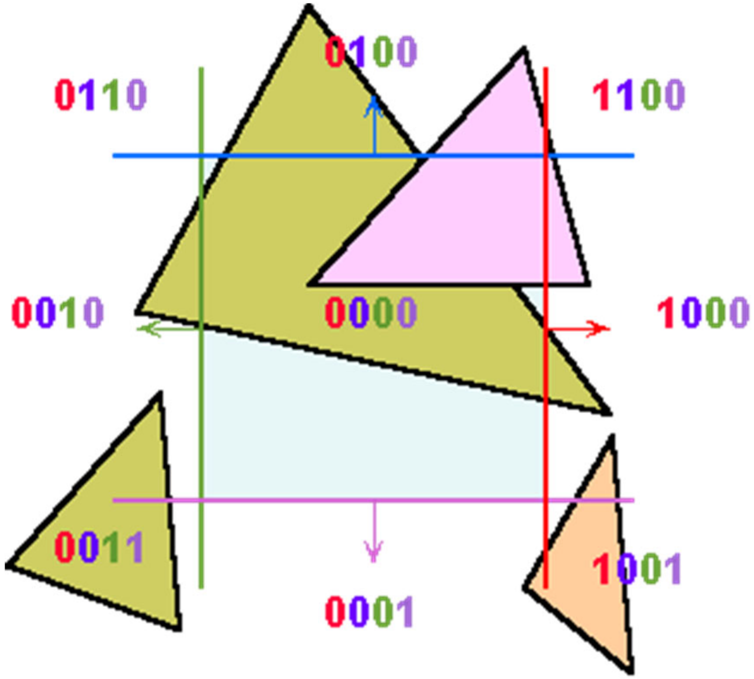
- Classic problem
- BSP trees
- Ray casting
- Depth buffering

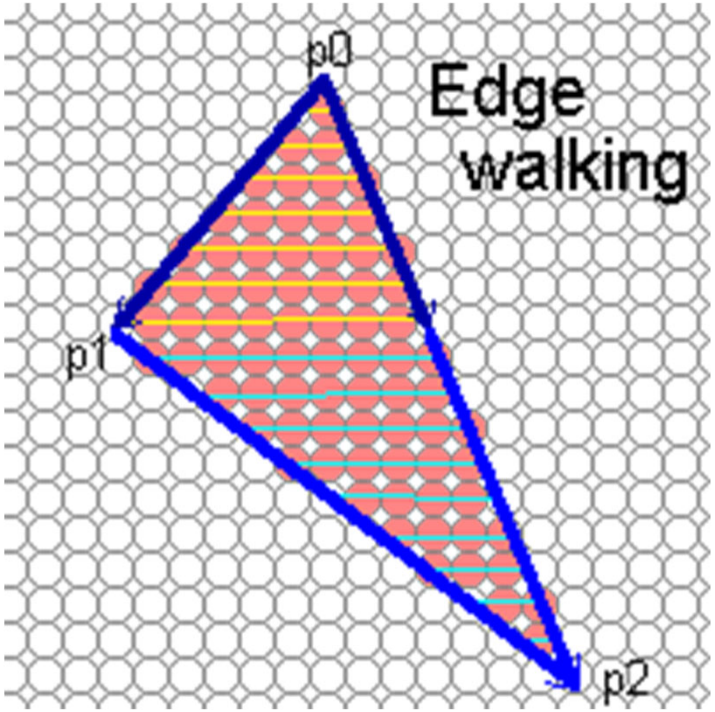




Rasterization

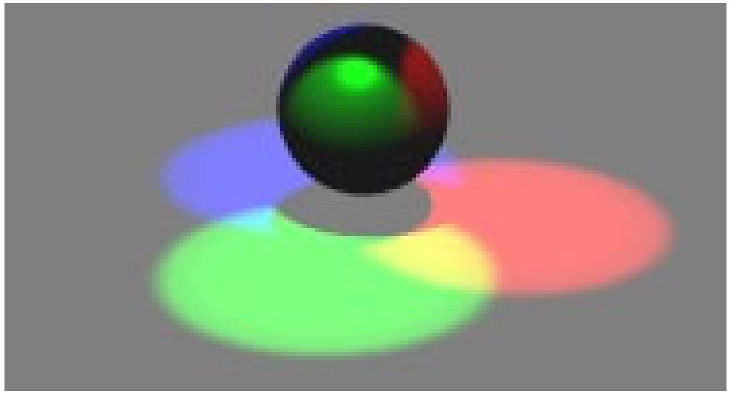
- Clipping
- Scan conversion



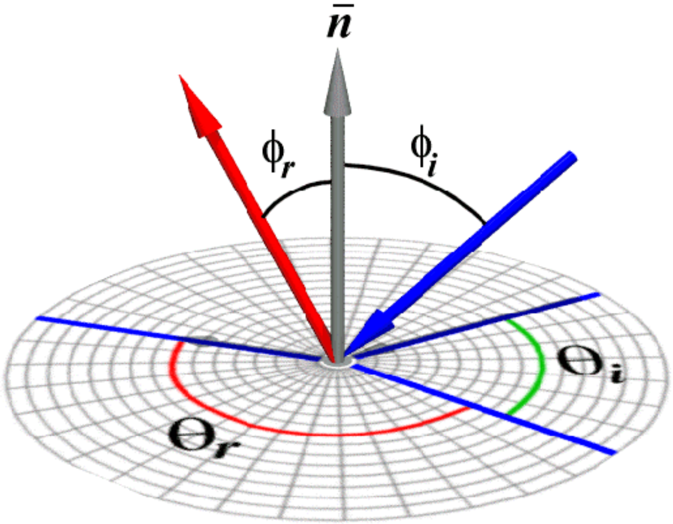


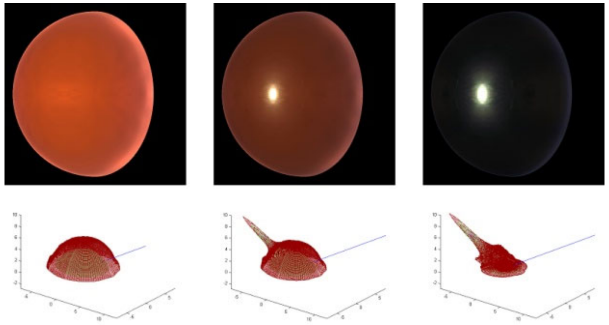
Lighting and Shading

- Flat, gouraud, and phong shading
- Empirical and physically-based illumination models
- BRDFs



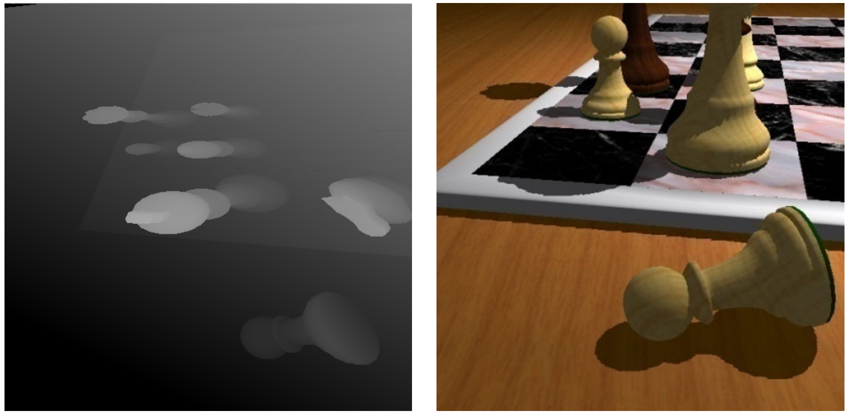
$\rho(\theta_r, \phi_r, \theta_i, \phi_i)$



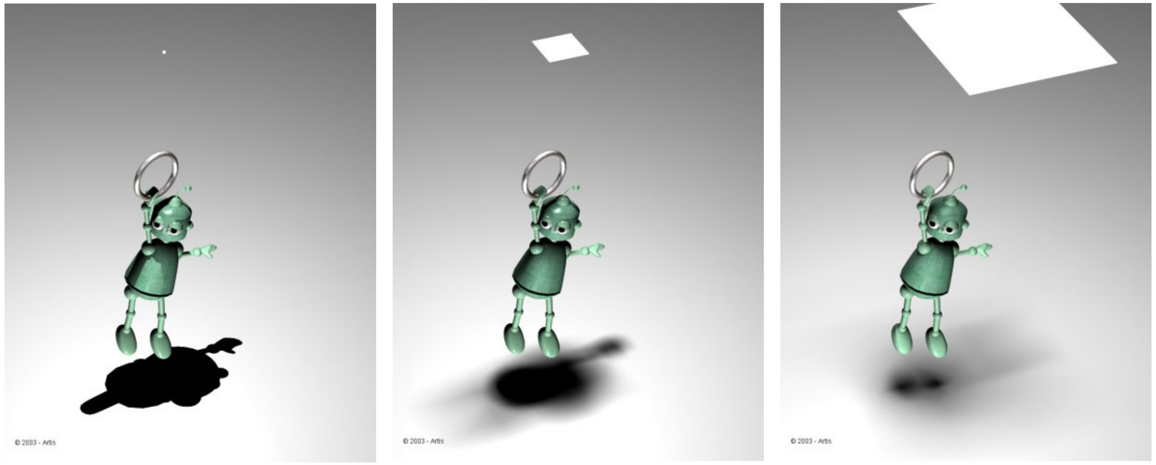


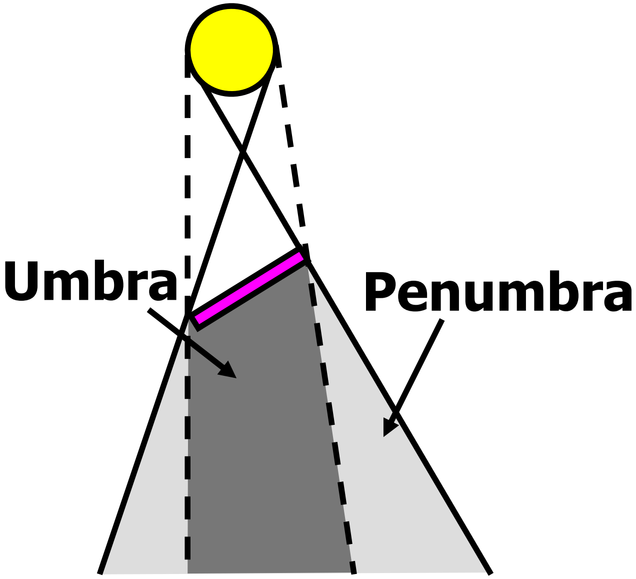
Shadows

- Shadow volumes
- Shadow maps



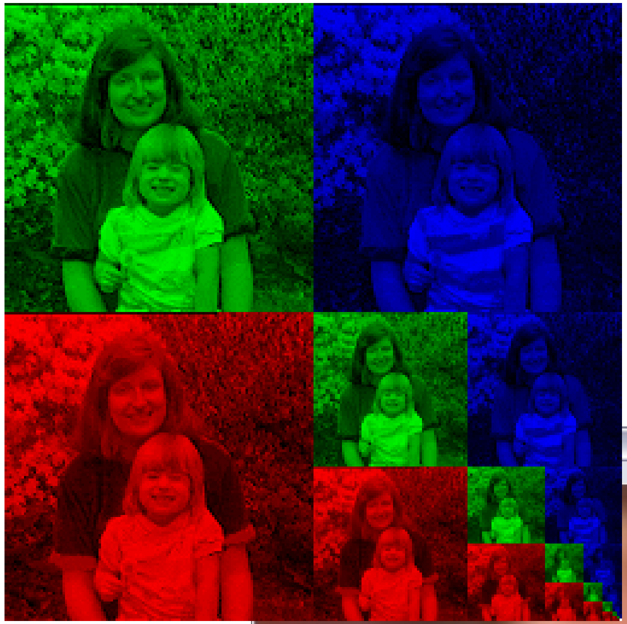
Images courtesy of Stamminger and Drettakis 02

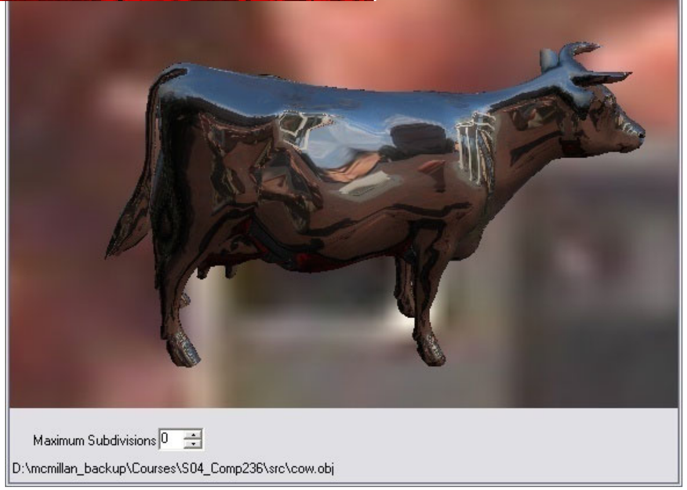




Texture Mapping

- Surface parameterization
- Mipmaps and filtering
- Reflection and environment mapping



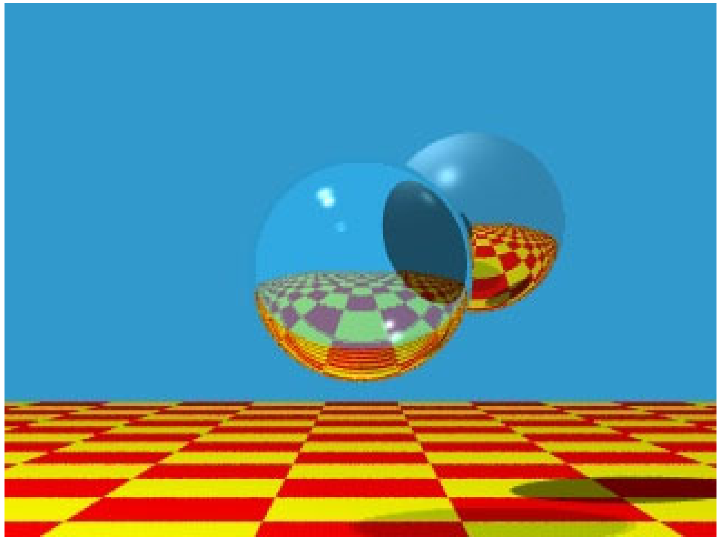


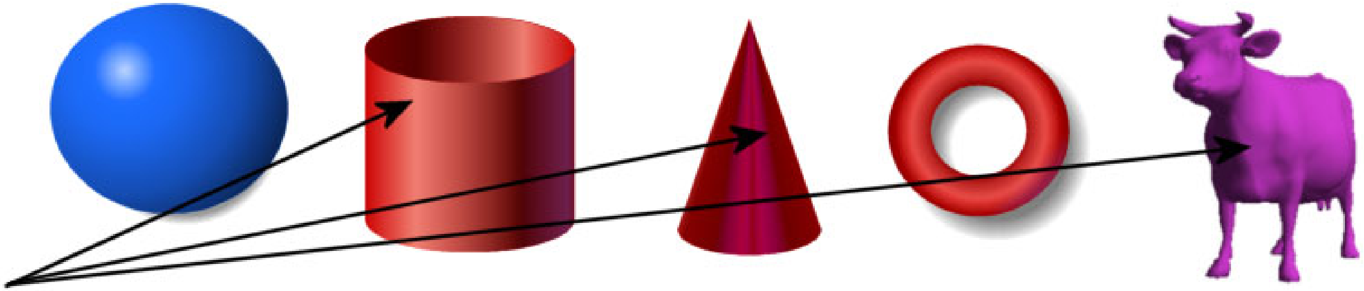
Maximum Subdivisions 0

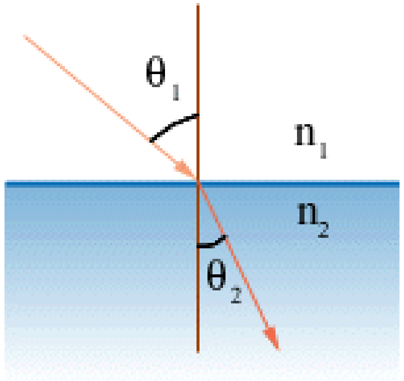
D:\mcmillan_backup\Courses\S04_Comp236\src\cow.obj

Ray Tracing

- Object intersection
- Reflection and refraction
- Depth-of-field, motion blur, glossy reflections, soft shadows

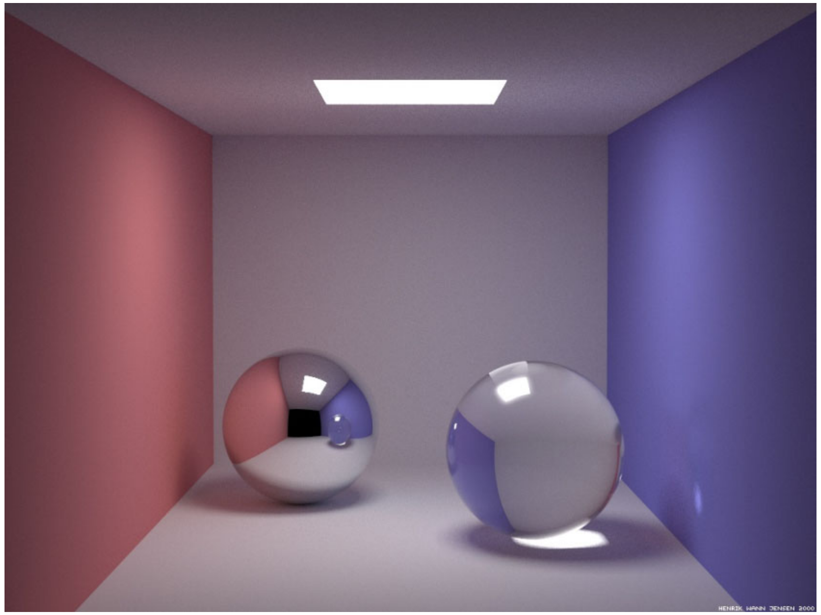


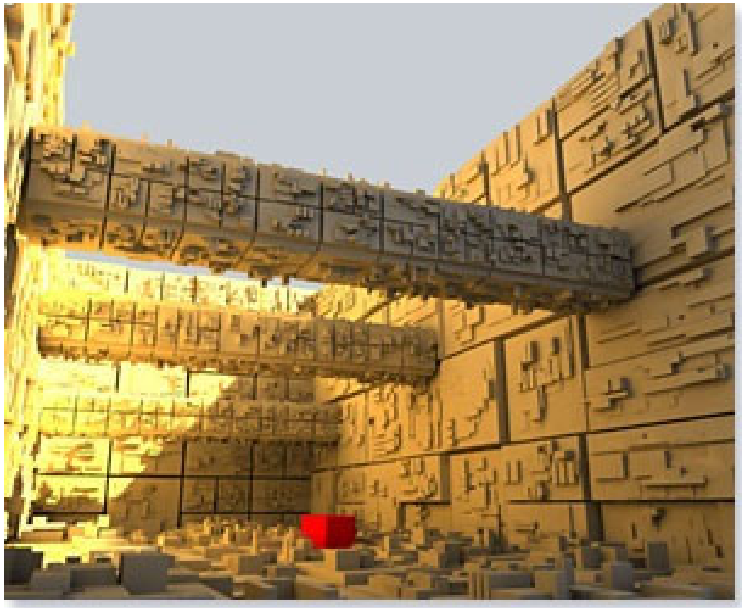




Global Illumination

- Rendering equation
- Path tracing, photon mapping, radiosity



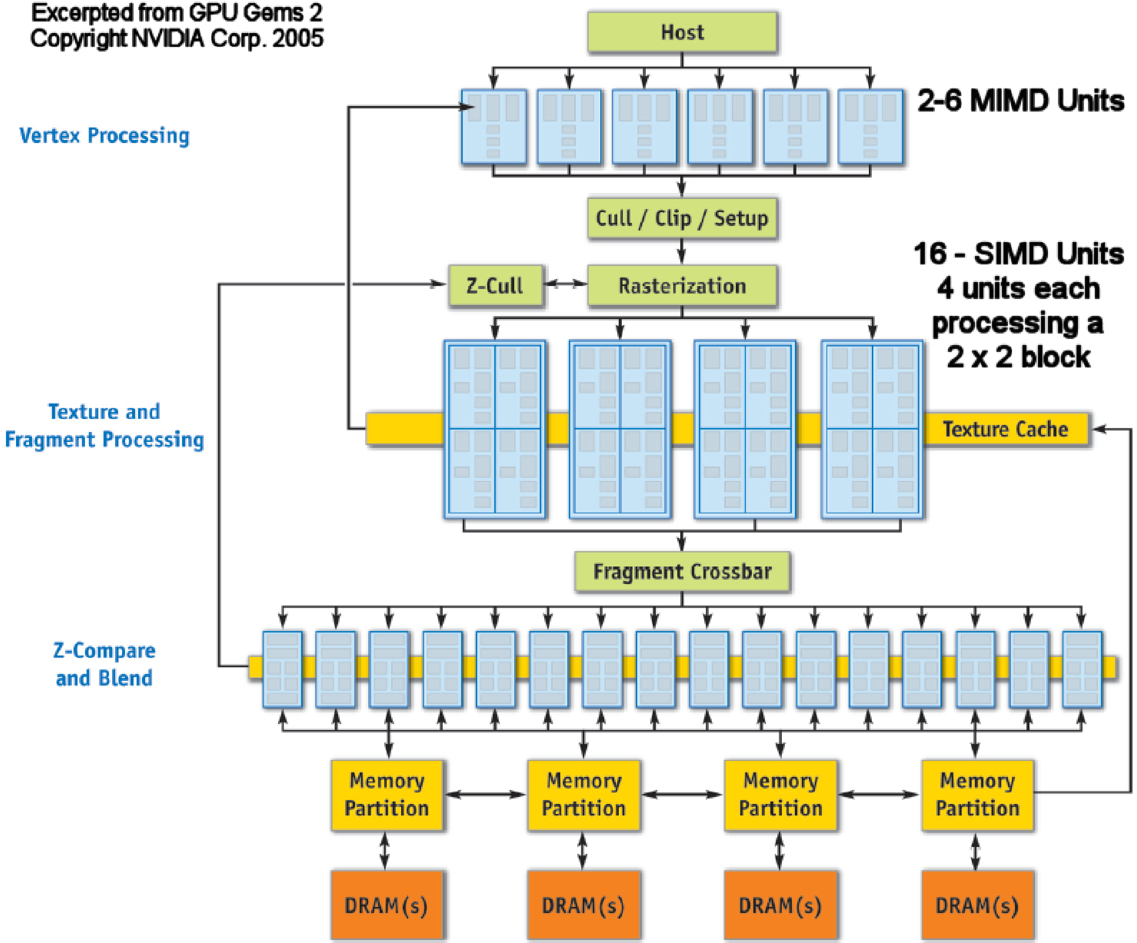


Images courtesy of Caligari (www.caligari.com)

Graphics Hardware

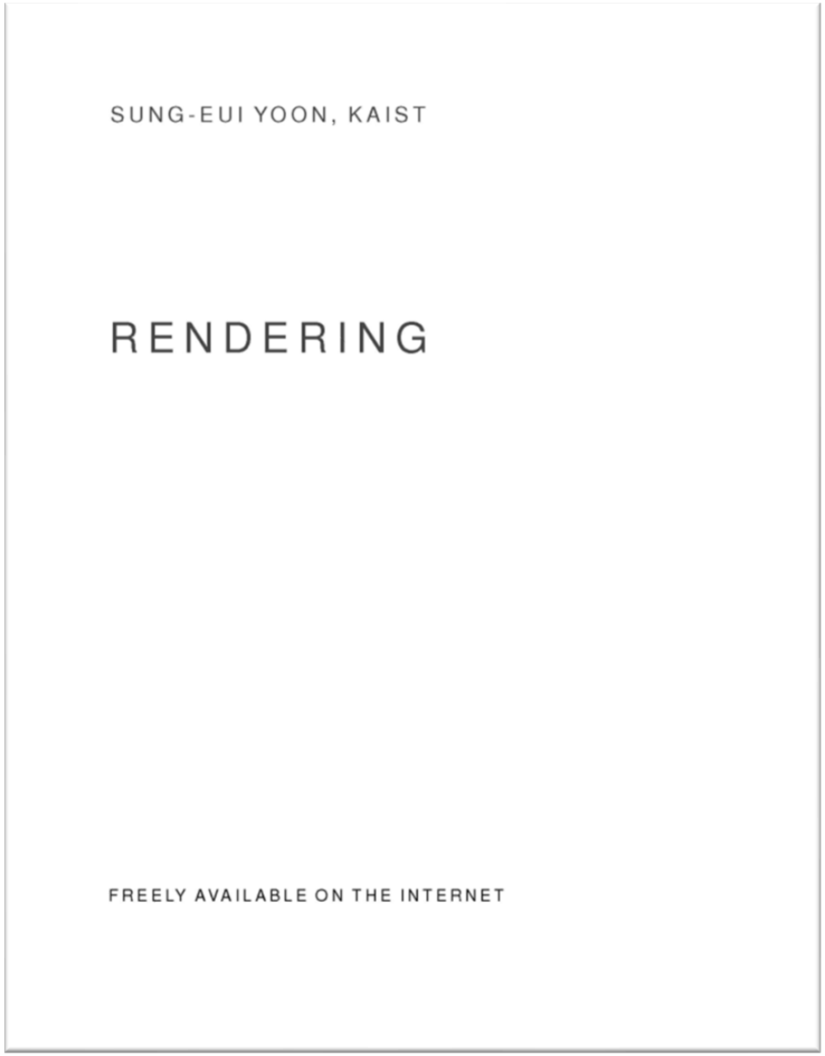
- History
- Architecture
- Shading languages
- Future

Excerpted from GPU Gems 2
Copyright NVIDIA Corp. 2005



Textbook

- Rendering
  - Sung-euí Yoon
  - 1st Edition, 2018
  - Freely available



SUNG-EUI YOON, KAIST

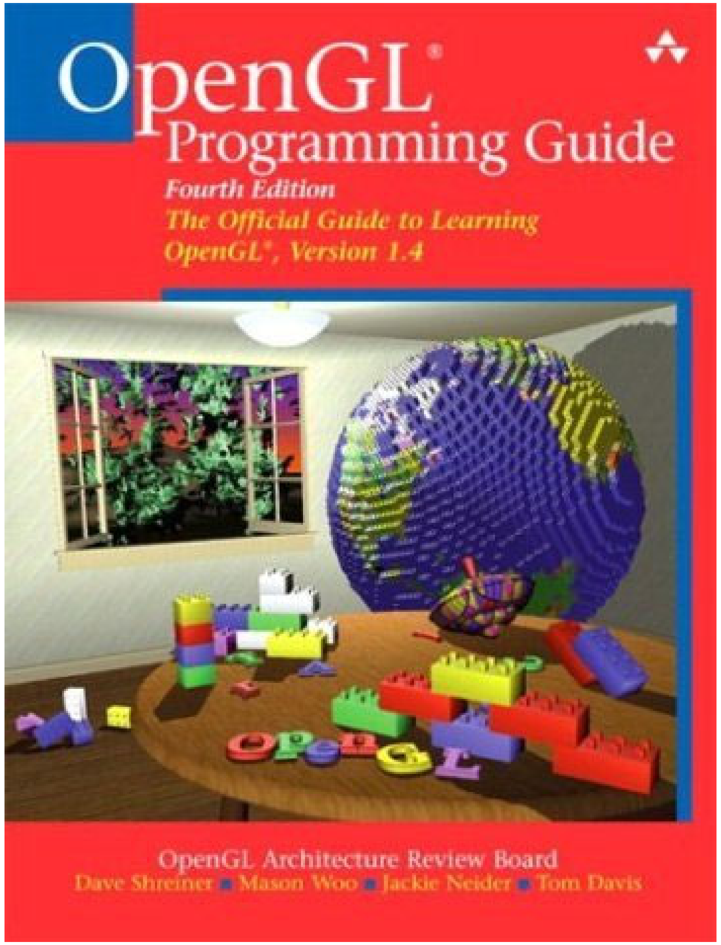
RENDERING

FREELY AVAILABLE ON THE INTERNET

https://sgvr.kaist.ac.kr/~sungeui/render/

Reference – OpenGL

- **OpenGL Programming Guide**
  - **Addison-Wesley Professional**
  - **Ver 4.3 is ordered to KAIST library**
- **Version 1.1 is available at internet and the course webpage**
- **Reference book is also available**

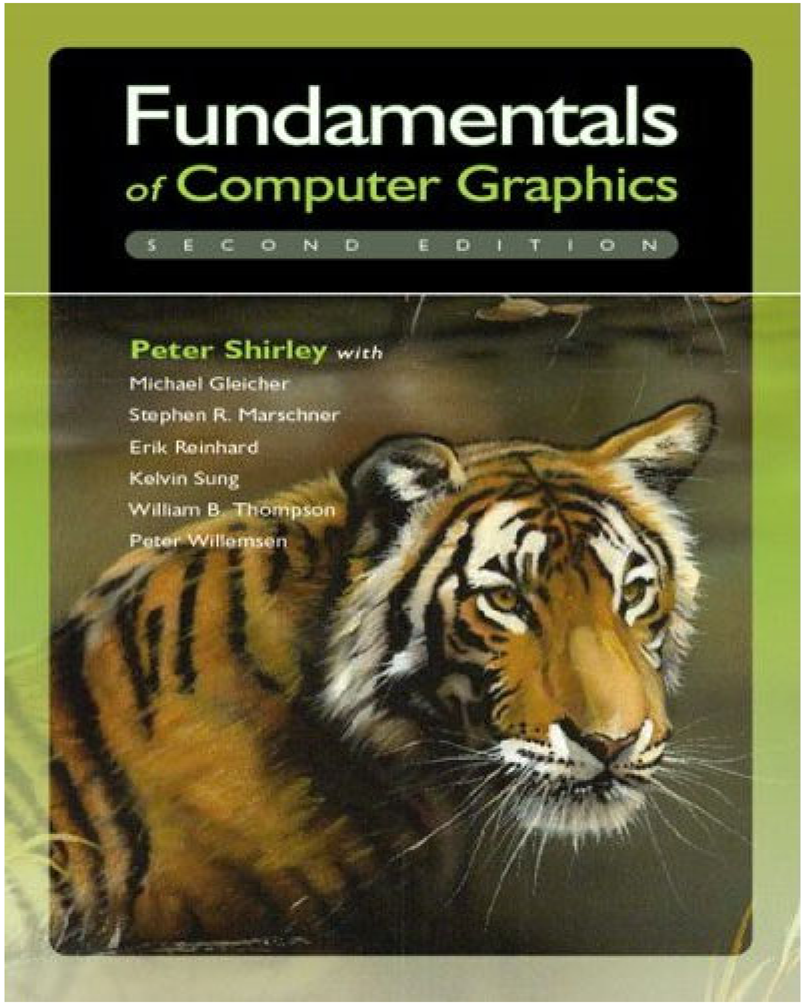

OpenGL®
Programming Guide
Fourth Edition
The Official Guide to Learning OpenGL®, Version 1.4
OpenGL Architecture Review Board
Dave Shreiner ▪ Mason Woo ▪ Jackie Neider ▪ Tom Davis

http://www.glprogramming.com/blue

Reference

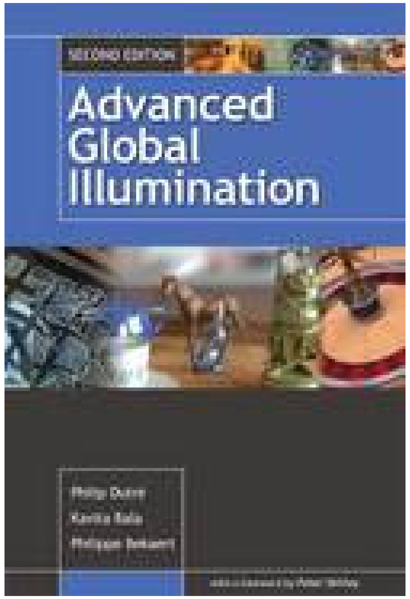
- **Fundamentals of Computer Graphics**
  - **Peter Shirley et al.**
  - **AK Peters**

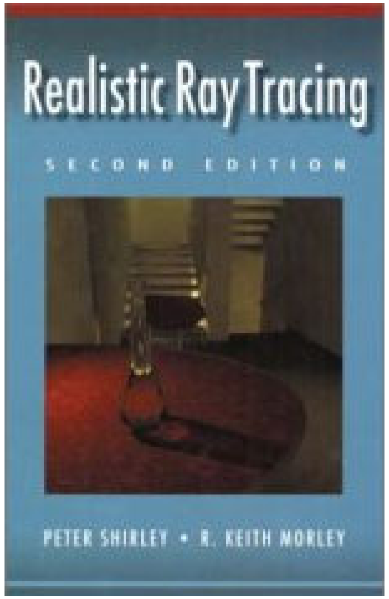
  - **Available at the KAIST library**

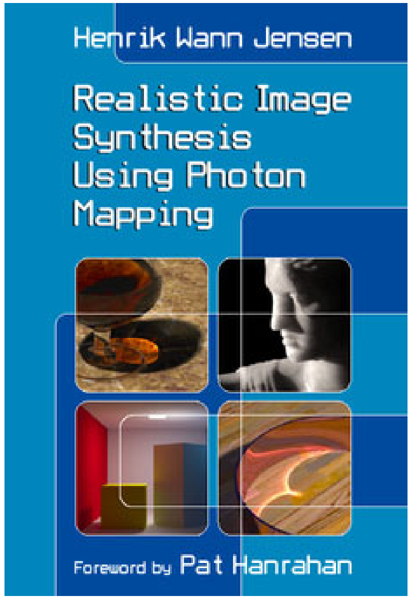


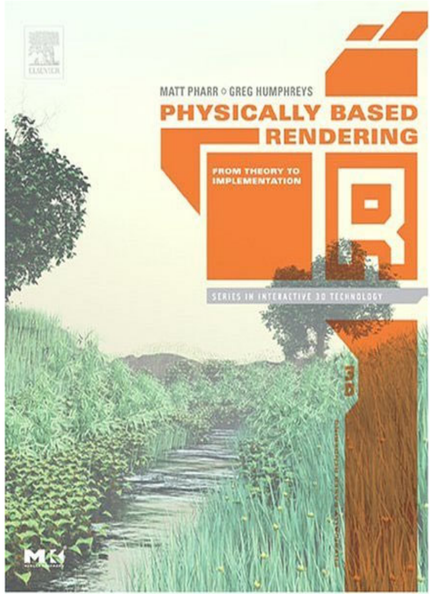
Resource on Physically-based Rendering

- **Reference**
  - **Physically based renderig, Matt Pharr et al.**
  - **Advanced Global Illumination, Philip Dutre et al. 2nd edition**
  - **Realistic Image Synthesis Using Photon Mapping, Henrik Jensen**
  - **Realistic Ray Tracing, 2nd edition, Peter Shirley et al.**








MATT PHARR ◦ GREG HUMPHREYS
PHYSICALLY BASED RENDERING
FROM THEORY TO IMPLEMENTATION

Other Reference

- Technical papers
  - Graphics-related conference (SIGGRAPH, CVPR/ECCV, etc)
  - http://kesen.huang.googlepages.com/

- Course homepages
- Google or Google scholar

COMPUTER GRAPHICS
PROCEEDINGS
acm Transactions on Graphics
Proceedings of ACM SIGGRAPH 2002

Program Assignments (PAs) and Quiz

- 5 or 6 PAs
  - Viewing and manipulating 3D models with OpenGL
  - Rasterization and clipping
  - Texture mapping and lighting
  - Raytracing
  - Trying out some recent implementations provided by paper authors
  - Their difficulty is growing!
  - Require you to know or self-study C/C++

- Quiz
  - We will frequently have quiz sessions, which also serve as attendance check

Homework for Each Class

- Go over the next lecture slides before the class
  - Just 10 min ~ 20 min for this should be okay

- Two video summary submission every week starting even from this week
  - Submit two before the next Mon. class
  - Preparation for paper presentation

- Question submissions two times during the whole semester

Student Lecture and Paper Presentation

- Related to your interest (student lecture) and research activity (paper presentation), which is useful for your long-term career
  - Edu 4.0 course asking students' participation
  - Things are changing rapidly due to chatgpt, etc.
- Make a team of 1 ~ 2 persons; 2 is better!
  - Two presentations per team
- Identify a lecture topic and a recent paper present during the semester
  - Lecture topic list will be available

Tentative Grading Policy

- **Mid-term/Final-term: 30%**
  Attendance, quiz and assignments: 30%
  Presentations: 40%

- **Late policy**
  - **No score for late submissions**
  - **Submit your work before the deadline!**

Class Attendance Rule

- Late two times → count as one absence
- Every two absences →lower your grade (e.g., A- → B+)

- To check attendance, I'll call your names or take pictures

- If you are in situations where you should be late, notify earlier w/ proper certificate or official documents

Honor Code and Etiquette

- Collaboration encouraged, but *assignments must be your own work*
- Cite any other's work if you use their codes
  - If you copy someone else's codes, you will get F
  - We will use a code copy checking tool to find any copy
- Classroom etiquette
  - Help you and your peer to focus on the class
  - Turn off cell phones
  - Arrive to the class on time
  - Avoid private conversations
  - Be attentive in class

Official Language in Class

- English
  - I’ll give lectures in English
  - I may explain again in Korean if materials are unclear to you
  - You are also recommended to use English, but not required

Next Time...

- Screen & world space
- Basic OpenGL usage

About You

- Name

- What is your major?
- Previous graphics experience

- Any questions?

- Online submission within today
  - https://forms.gle/aHT8abgjaYAsV2wDA
  - You can also find the link at the course homepage