Course Introduction

Computer Graphics
Instructor: Sungkil Lee

Course Overview

Contacts

- **Office hour**
  - Wednesday 10:30-11:30, at my office (27328)
  - During the office hour, I will stay at my office as far as possible.

Teaching Assistants (TAs)

- **Section 41**
  - Hyojin Jung (정효진)

- **cglab.skku@gmail.com**
  - Send an email to this address to contact any of the TAs
  - Make sure to write your section in your email title.
  - Examples
    - [cg41] Something to tell …

Languages

- **English section**
  - Basically, most of the lecture will be given in English.
  - But, when it is considered too complex or hard to explain, Korean can be also used for Korean students.

- **Korean section**
  - Lectures are given in Korean.
  - But, the materials are shared with English lectures.

Course Summary

- **Implication of CG**
  - Computer graphics is a fundamental tool for creating and manipulating visual media including games, animation, virtual reality, and web, and is also a crucial component for science and engineering software.

- **What to cover**
  - This course covers basic theory and practical techniques of computer graphics for digital media.

- **Particulars in this course**
  - This course particularly deals with modern-style shader programming for its implementation.

What you will learn in this course

- **Algorithms for creating realistic images**
- **Having fun improving your C++ programming skills**
  - CG is one of the most appropriate topics for object-oriented C++ programming
  - You will also learn how to use third-party libraries
- **GPU programming**
  - The concepts of OpenGL programming
  - This course is a *very unique class that covers modern-style OpenGL* which utilizes the power of modern GPU.
  - The basic knowledge of GPU programming can be easily extended to mobile graphics (e.g., OpenGL ES) and general-purpose GPU programming (e.g., CUDA, OpenCL).

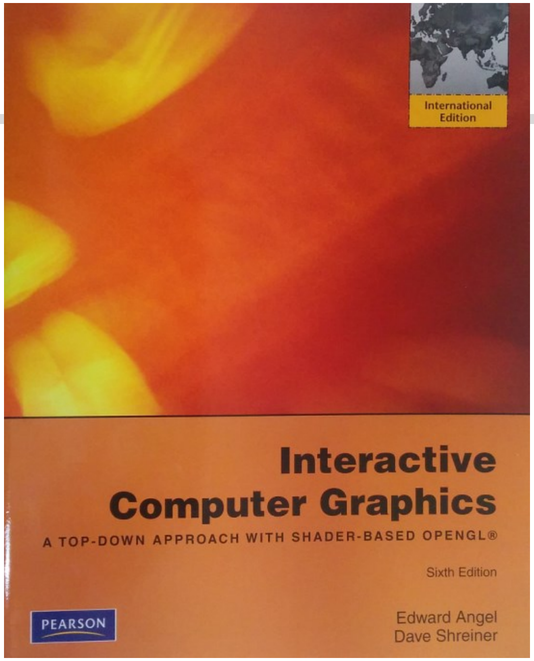
Prerequisites

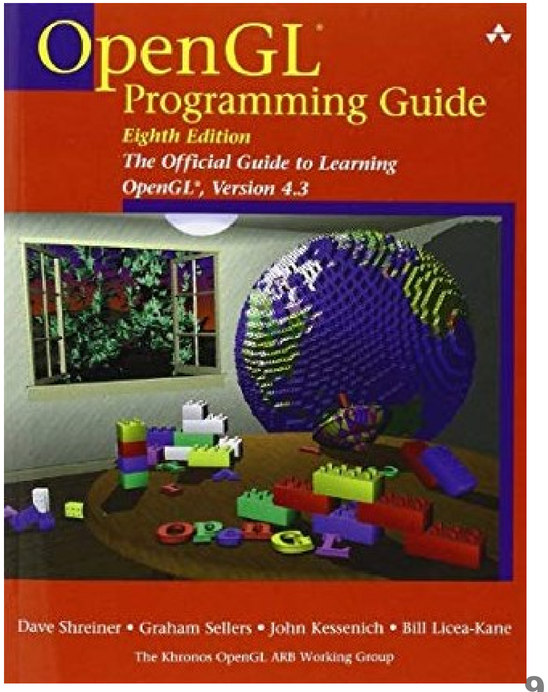
- **Data structures, Algorithms**
  - The core of CG can be effective data structures and algorithms for computing realistic imagery, which can be also parallelized.
  - If you did not learn data structures or algorithms, I recommend taking the course after having them first.
- **C++**
  - The concept of object-oriented programming
  - The concept of event-driven programming
  - Still one of the most powerful languages for high-performance computing
- **Linear Algebra**
  - The basics of vector and matrix manipulation
  - Mostly high-school algebra

Textbook and References

- **Textbook**
  - Interactive Computer Graphics: A Top-Down Approach with Shader-Based OpenGL
  - Edward Angel and Dave Shreiner
  - 6th Edition, 2011

- **References**
  - OpenGL Programming Guide: The Official Guide to Learning OpenGL, Versions 4.3 or later} (aka Red Book). Dave Shreiner, 2013.
  - http://www.opengl.org/


International Edition
Interactive Computer Graphics
A TOP-DOWN APPROACH WITH SHADER-BASED OPENGL®
Sixth Edition
PEARSON
Edward Angel
Dave Shreiner


OpenGL
Programming Guide
Eighth Edition
The Official Guide to Learning OpenGL®, Version 4.3
Dave Shreiner • Graham Sellers • John Kessenich • Bill Licea-Kane
The Khronos OpenGL ARB Working Group

Course Webpage

- **Course web page: http://cg.skku.edu/course/cg/**

ID: cg
password: skku

  - All course materials will be announced and uploaded on the web.
  - Various resources concerning this course will be also available.
  - Check the web page regularly.
  - Only assignment submission will use i-campus.

Grading Policy

- **General grading rule**
  - In general, hard-working students will get a good grade.
  - *Programming assignments* are very important throughout the course.
  - Your attendance has a non-trivial effect on your grade.

- **Organization**
  - Attendance and *attitude*: 10 %
  - Assignments: 40%
  - Mid-term exam: 20%
  - Final exam: 30%

Grading Policy

- **Rules on attendance**
  - When you are absent in the class more than **5 times** (i.e., from 6 times), you will fail to pass this course; i.e., you get F grade.
  - Absence with any reasons will not be considered presence.
    - For example, even a serious job interview will just be an absence.
    - Do not bring a document that proves you were absent.
  - However, the following cases will be regarded as an exception:
    - Department/college activities
    - Military services (e.g., 예비군훈련)
    - Your family passed away; go there immediately and let me know later.
  - Two late attendances are equivalent to one absence.
  - One absence will be -1 point in your total score (100 pts).

Desired Attitudes for this Course

- **This course is ~~not~~ an easy-going one.**
  - You will learn a lot of unique stuffs, unavailable from other courses.
  - Participate the course actively.

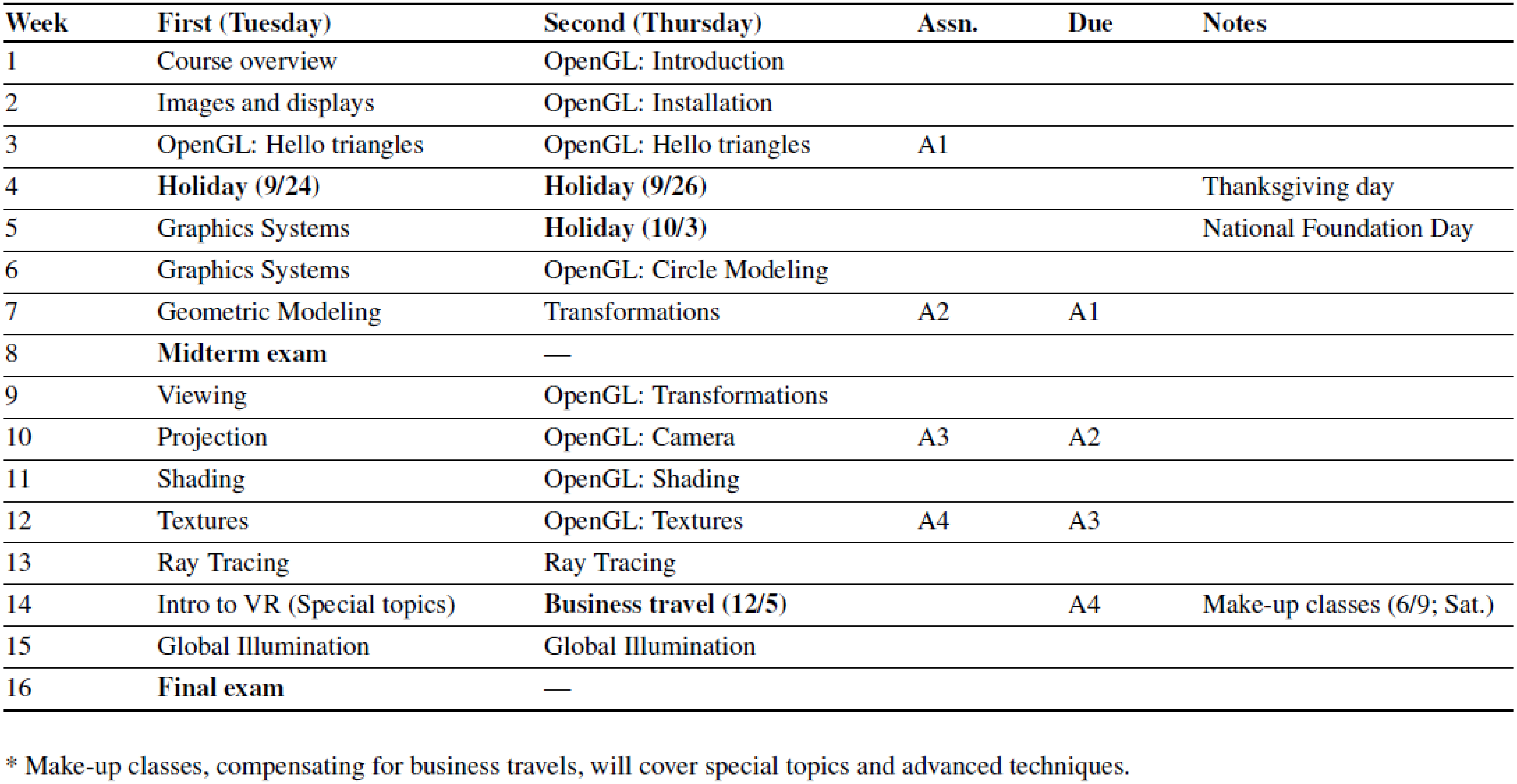
- **Basic etiquettes**
  - Attend in time: when you are late, it might interrupt lecture.

Desired Attitudes for this Course

- **No cheating!**
  - Many of the assignments are available from the last year ones.
  - For many years, I have found a lot of cases.
  - All of the assignments are intended to improve your programming skills. Hence, do it on your own. It will significantly raise your value.
  - If cheating is found in any cases, you will not pass this course.
    - Remember that you immediately get F grade for the cheating.

Schedule

- **The course will basically follow the schedule below:**



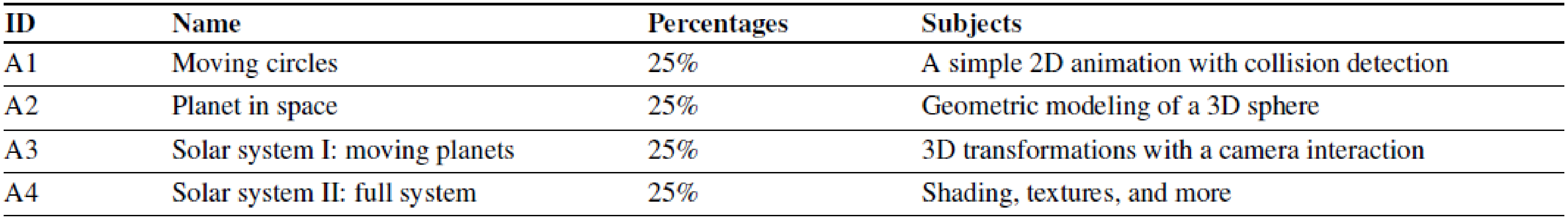
Week	First (Tuesday)	Second (Thursday)	Assn.	Due	Notes
1	Course overview	OpenGL: Introduction			
2	Images and displays	OpenGL: Installation			
3	OpenGL: Hello triangles	OpenGL: Hello triangles	A1		
4	**Holiday (9/24)**	**Holiday (9/26)**			Thanksgiving day
5	Graphics Systems	**Holiday (10/3)**			National Foundation Day
6	Graphics Systems	OpenGL: Circle Modeling			
7	Geometric Modeling	Transformations	A2	A1	
8	**Midterm exam**	—			
9	Viewing	OpenGL: Transformations			
10	Projection	OpenGL: Camera	A3	A2	
11	Shading	OpenGL: Shading			
12	Textures	OpenGL: Textures	A4	A3	
13	Ray Tracing	Ray Tracing			
14	Intro to VR (Special topics)	**Business travel (12/5)**		A4	Make-up classes (6/9; Sat.)
15	Global Illumination	Global Illumination			
16	**Final exam**	—			

* Make-up classes, compensating for business travels, will cover special topics and advanced techniques.

Programming Assignments

- **Four assignments in total will be given in the course.**
  - They are designed for step-by-step improvements, leading from geometric modeling to a more complex 3D animation.
  - When you follow the schedule step by step, they will be in an acceptable level of difficulty.
  - A submission due for each is usually given 2-3 weeks in most cases.
  - You may need to fully spend at least three to four days for each assignment.

Programming Assignments: Subjects



ID	Name	Percentages	Subjects
A1	Moving circles	25%	A simple 2D animation with collision detection
A2	Planet in space	25%	Geometric modeling of a 3D sphere
A3	Solar system I: moving planets	25%	3D transformations with a camera interaction
A4	Solar system II: full system	25%	Shading, textures, and more

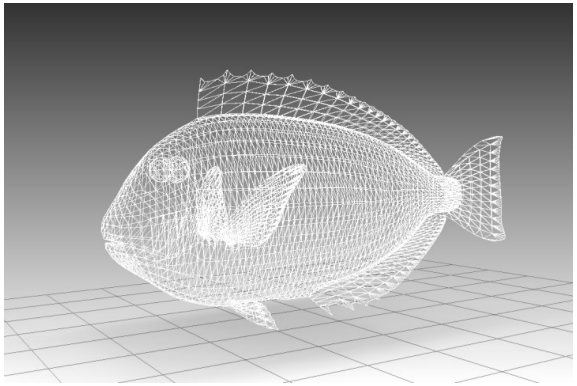
Any questions?

Computer Graphics Overview

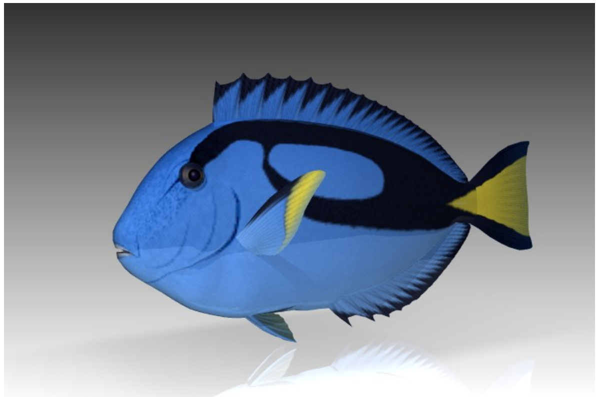
Big Picture of Computer Graphics (CG)

- **Computer graphics deals with:**
  - all aspects of creating images with a computer: hardware, software, and applications.

- **Three primary research areas in CG**



Modeling

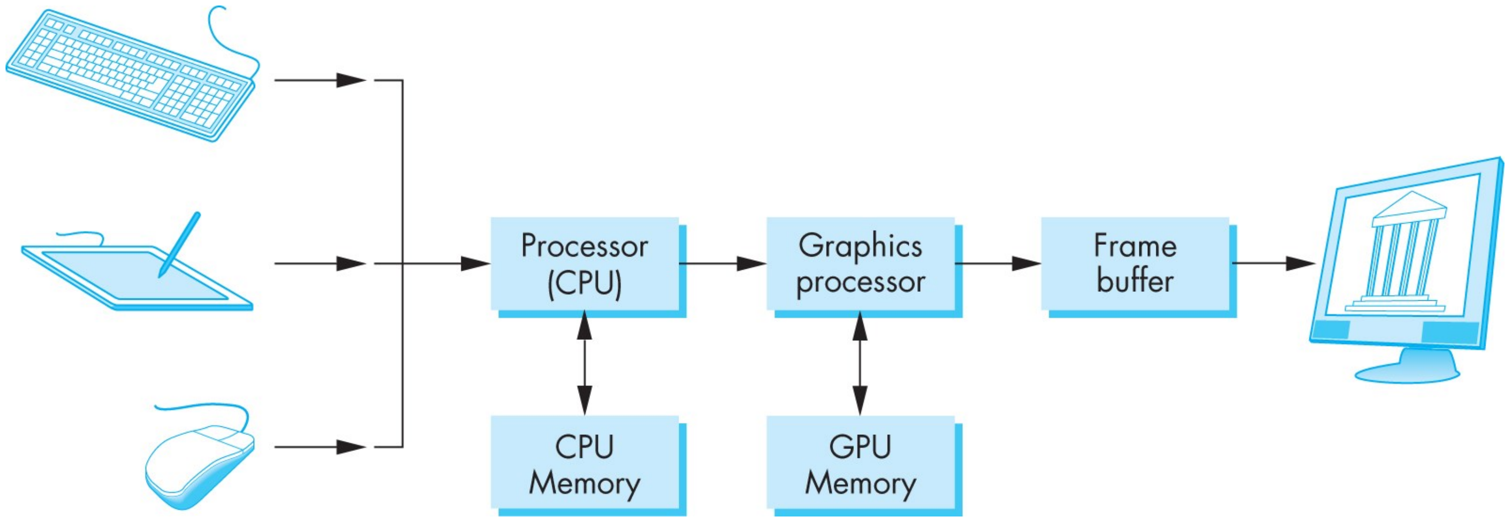


Rendering

Animation

Interactive Graphics System

- **Basic system for interactive graphics**
  - Recent mobile/embedded systems often include touch-sensitive display for both input and output.



Input devices

Output device

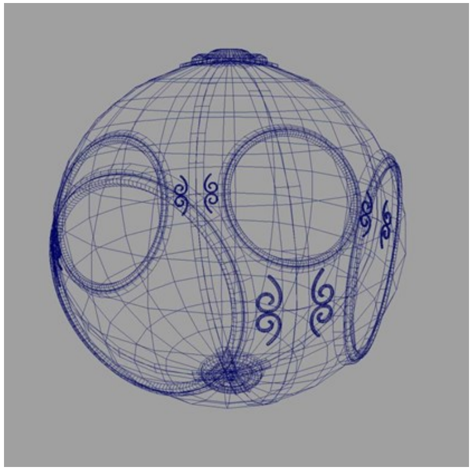
Brief History

- **1950s:**
  - Computer graphics goes back to the earliest days of computing
  - Strip chart recorder, pen plotters

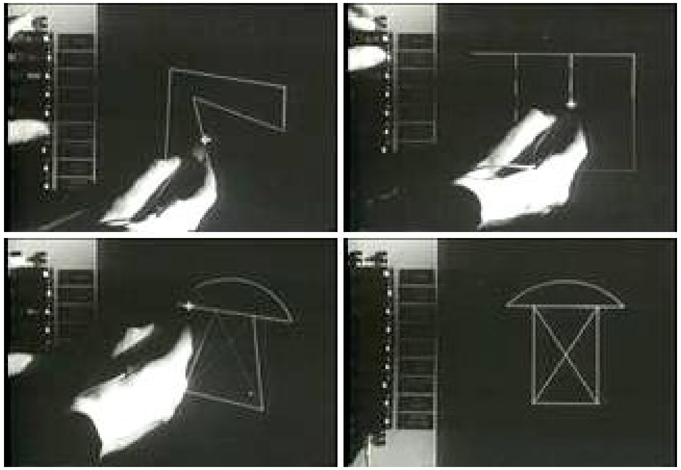
Strip chart recorder and pen plotters
(HP 7035B, Robotron K6418)

Brief History

- **1960s:**
  - Raster graphics with wireframe display
    - Early predecessor of the modern raster graphics
  - Sketchpad (Turing-awarded, 1988)
    - Software written by Ivan Sutherland (Ph.D. thesis at MIT).
    - The early concept of display loop, still common in computer graphics
    - Computer display new images by light pen movements



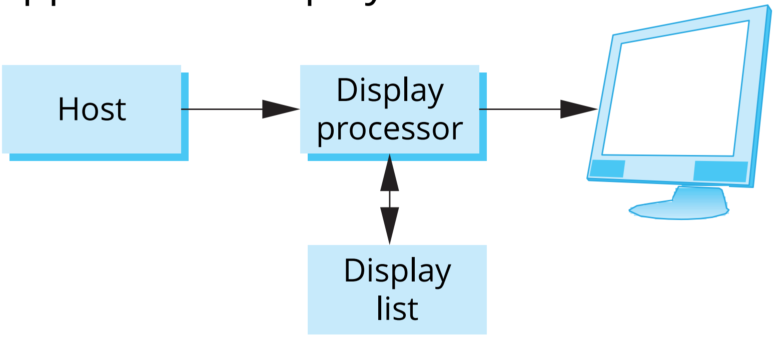
Wireframe display



Sketchpad (Sutherland)

Brief History

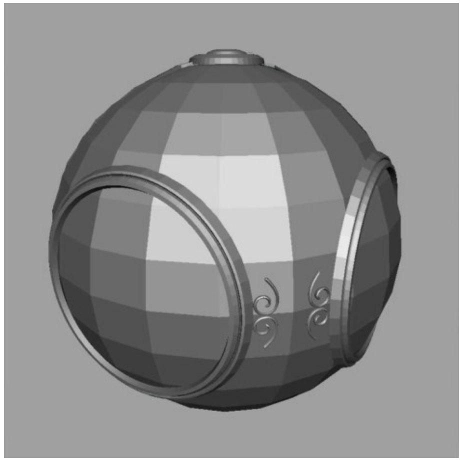
- **1960s:**
  - Direct view storage tube (DVST), created by Tektronix
    - Did not require constant refresh
    - Similar to CRT but with highly persistent phosphor
    - Opened door to use of computer graphics to CAD community
  - Display processors
    - Rather than have the host computer try to refresh display use a special purpose computer called a *display processor* units (DPUs)
    - Host compiles display list and sends to DPU.
    - Early OpenGL supports this display list mode.



Display-processor architecture

Brief History

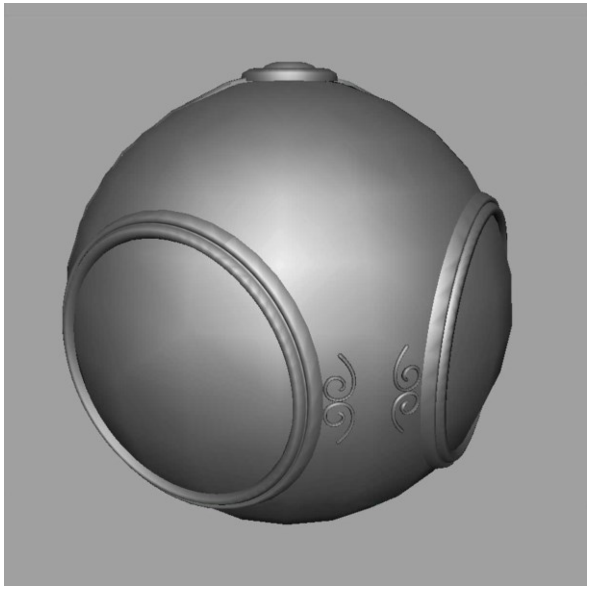
- **1970s:**
  - Two graphics standard committees formed by International Federation of Information Processing Societies (IFIPS; 1973)
    - GKS: Europeal effort (becomes ISO 2D standard)
    - Core: North Americal effort (3D but fails to become ISO Standard)
  - Raster graphics
    - Allows us to go from lines and wire frame images to filled polygons
    - Image produced as an array of picture elements (pixels) in the frame buffer



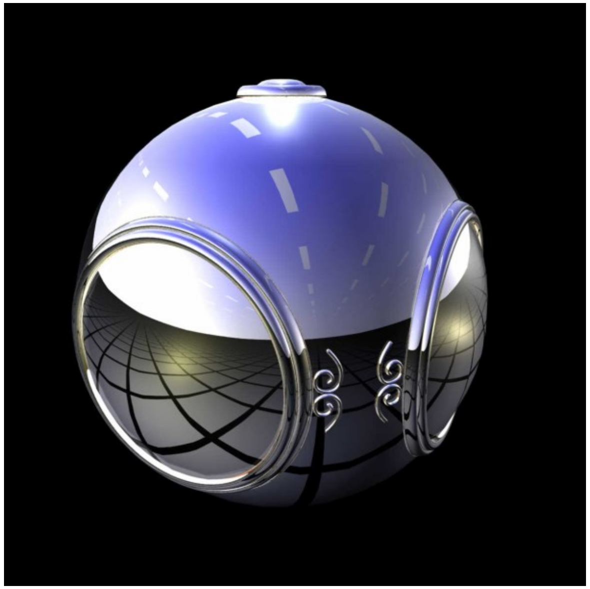
Early raster display

Brief History

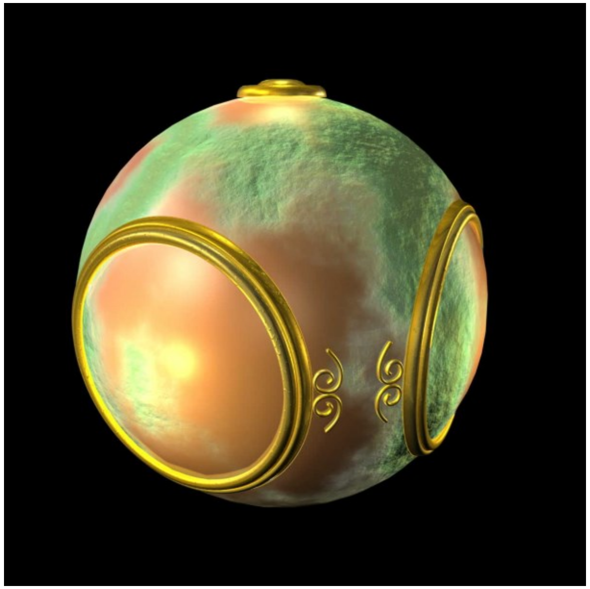
- **1980s:**
  - Realism comes to computer graphics



Smooth shading



Environment mapping



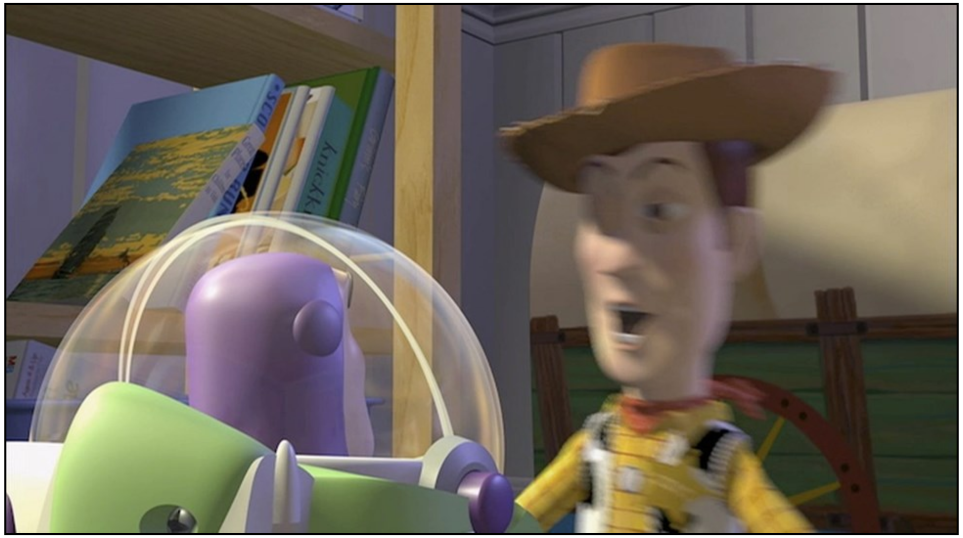
Bump mapping

Brief History

- **1980s:**
  - Hardware geometry engine by Silicon Graphics Inc. (SGI)
    - VLSI implementation of graphics pipeline (1982)
    - Iris and Iris GL: Operating system and Graphics API of SGI
  - Industry-based standards
    - CAD Community: PHIGS (Programmers Hierarchical Graphics System)
    - Pixar: RenderMan API (and REYES architecture)
  - Networked graphics: X Windows system (by DEC/MIT)
    - Client-server architecture with graphics

Brief History

- **1990s:**
  - Iris GL later became OpenGL (1992)
    - an easy-to-use platform-independent rendering API (no windowing support)
    - Close enough to hardware to get excellent performance
  - New hardware capabilities
    - Texture mapping, blending, accumulation, stencil buffers
  - *Toy Story* : the first completely computer-generated feature-length movie



Toy story, Pixar (1995)

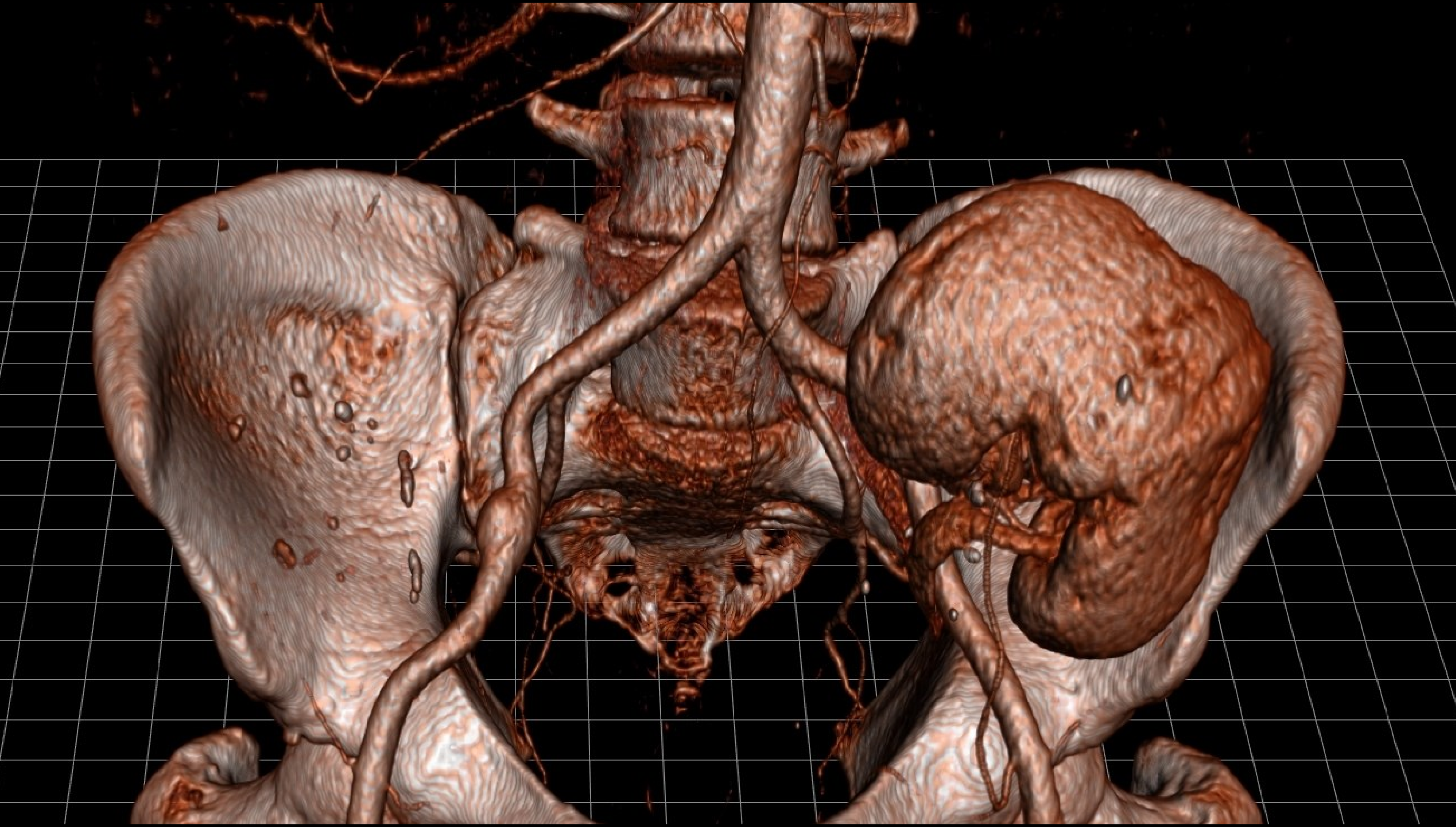
Brief History

- **2000s-now:**
  - Graphics cards for PCs dominate market
    - NVIDIA, ATI (now AMD), 3DLabs
  - Game boxes and game players determine direction of market
    - Playstation, Wii, XBOX (+Kinect), ...
  - Computer graphics software in movie industry
    - 3DS Max, Maya, Lightwave
  - Programmable rendering pipelines: GPU named by NVIDIA
  - Display technology
    - LCD, PDP (Plasma display panel), LED, OLED, ...

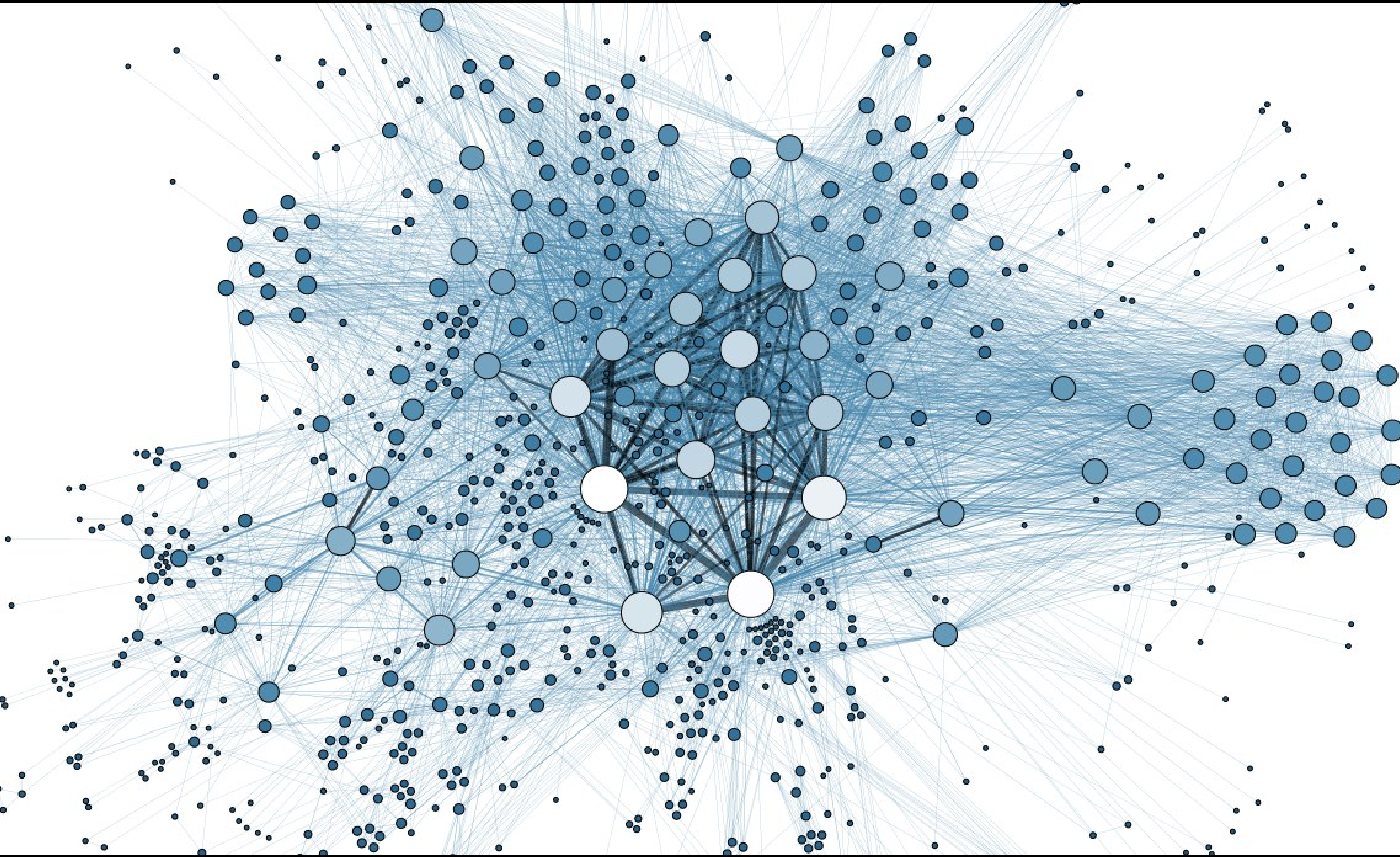
Applications:
What can we do with CG?

Entertainment - Film Production (Toy Story, Pixar 1995)
First CG-Generated Full-Length Movie

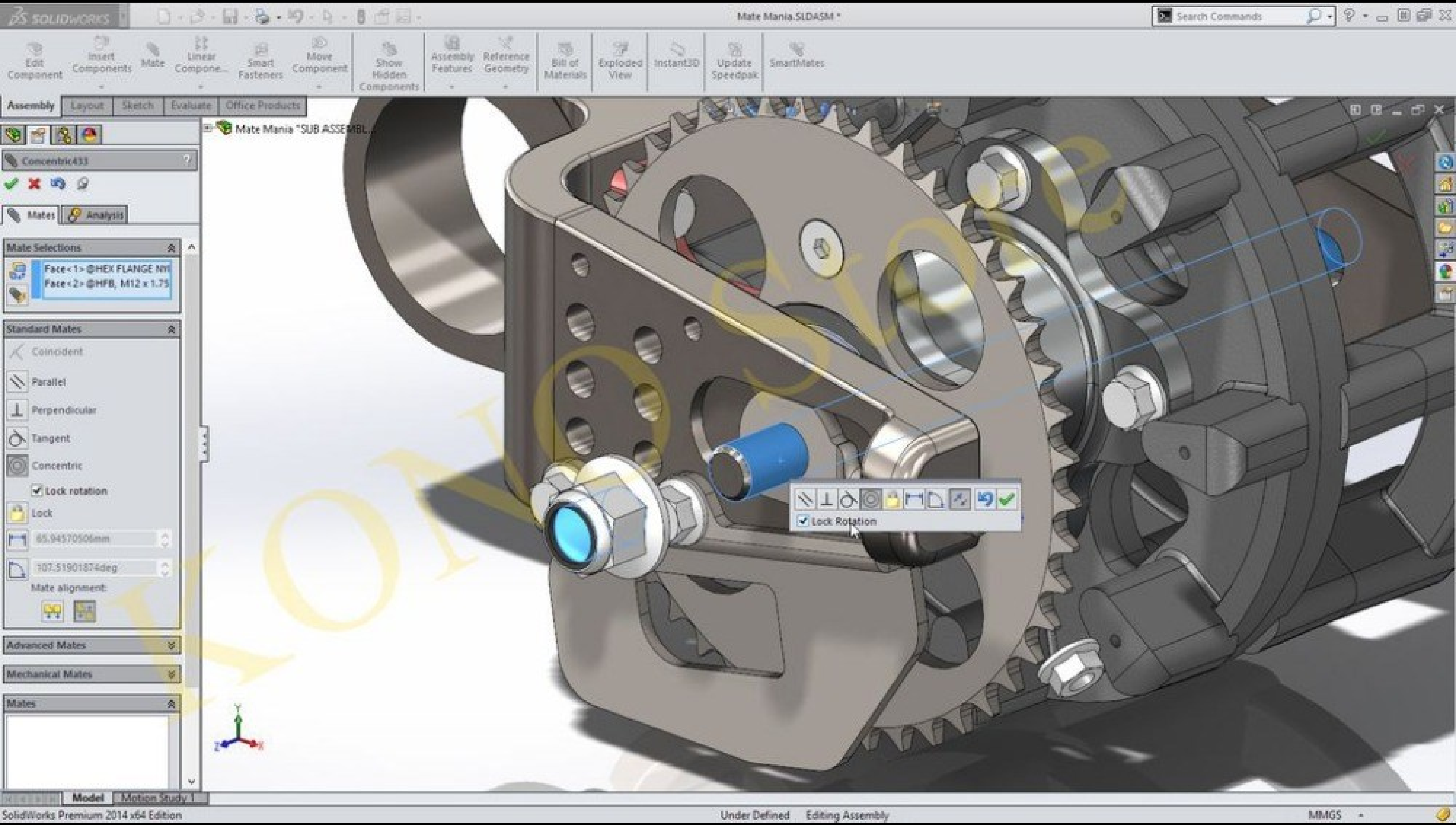
Entertainment - Games (Unreal Engine RTX Demo, Epic Games)



Science and Engineering - Medical Visualization



Science and Engineering - Data Visualization



Mate Selections:
Face<1>@HEX FLANGE NYl
Face<2>@HFB, M12 x 1.75

☑ Lock rotation

65.94570506mm

107.51901874deg

SolidWorks Premium 2014 x64 Edition   Under Defined   Editing Assembly   MMGS

Science and Engineering - CAD (Virtual prototyping)

SEAT MUST BE IN FORWARD 7 INCHES OF TRAVEL DURING TAKEOFF AND LANDING
SEAT MUST BE IN FORWARD 7 INCHES OF TRAVEL DURING TAKEOFF AND LANDING

Training and Simulation - Virtual Reality (Flight Simulation)

Training and Entertainment - Virtual Reality (Oculus VR)

Any further questions?