

Computer Graphics

Contents

- Computer graphics: what and why
- Many scenes, two rendering paradigms and one image
- Course organization

Computer graphics: what and why

Motivation

"Inspired by nature, incorporate science and art with technology to create virtual environments that exist or never could have existed."

"Computer graphics is science and art of communicating visually via computer display and its interaction devices" *

"A collaboration between art and technology" **

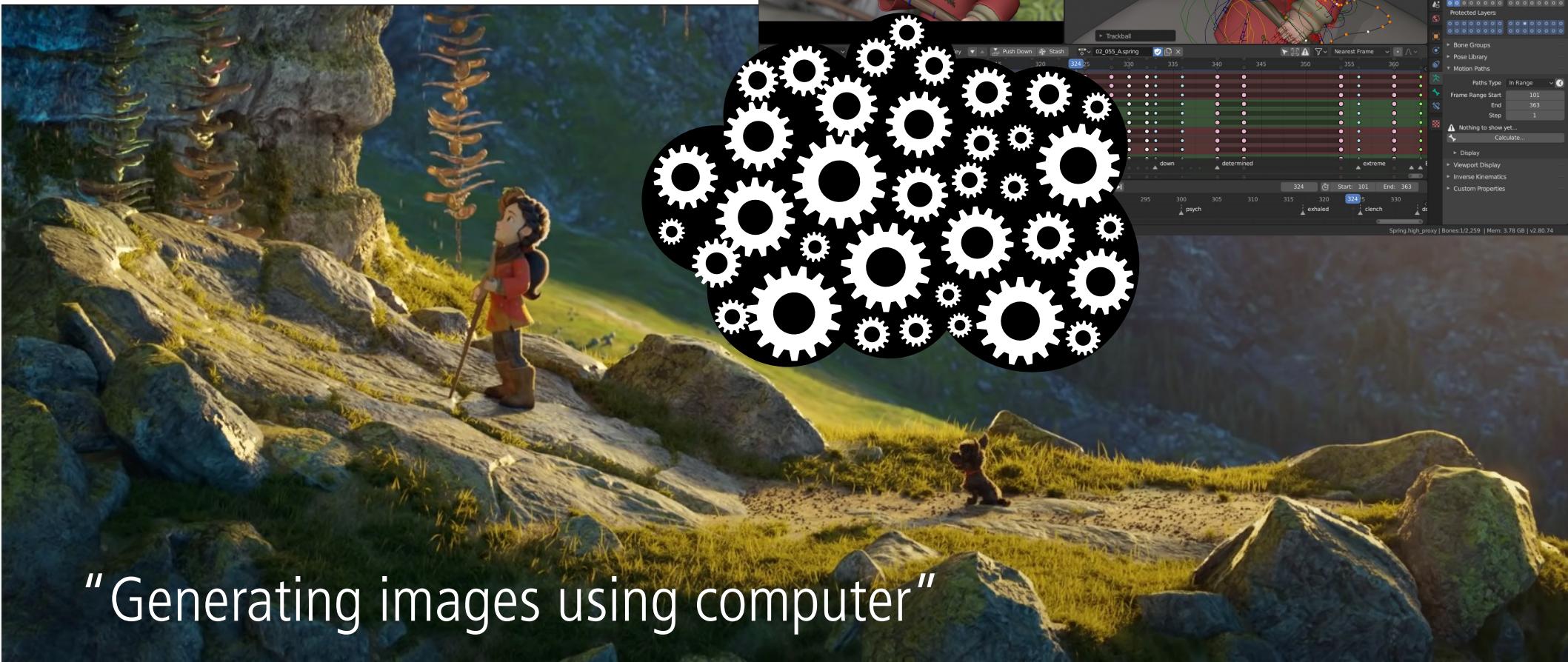
To visualize

To express

* Book: principles and practices

** Pixar

Computer graphics

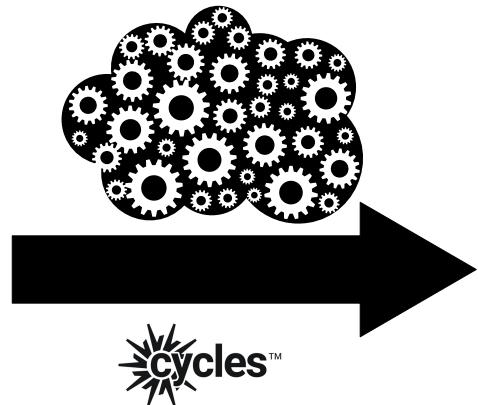
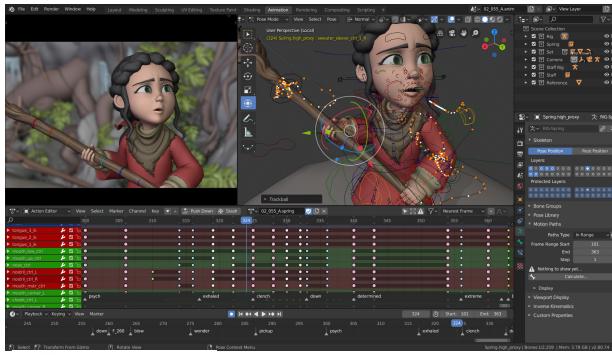


"Generating images using computer"

Computer graphics

Generating images:

- **3D scene** – what will be present in image
- **Rendering** – how image is generated from 3D scene
- **Image display** – how image is displayed



Computer graphics and you

- You like **physics** and would like to see its practical applications in generating amazing imagery and effects.
- You like **mathematics**: computer graphics is applied mathematics. Enough said.
- You like **programming**: computer graphics is exciting application that employs complex architectures for modeling and rendering and in return gives very gratifying results.
- You like **art** and **design**: Computer graphics is not only about tools which serve for simulating and rendering 3D scenes - it is also how we use those tools to create something that exists or never existed
- You like **animated films** or **VFX**: yes there is a lot of computer graphics there combined with other disciplines to support stories to remember
- You like **computer vision** or image processing: graphics is about creating and manipulating images
- You like **human-computer interaction**: computer graphics enables visual interfaces and interaction
- You like **computer games**: amazing application of computer graphics combined with different disciplines
- You like **visualization**: biology? Chemistry? Geology? Astronomy? Computer graphics is there for you!

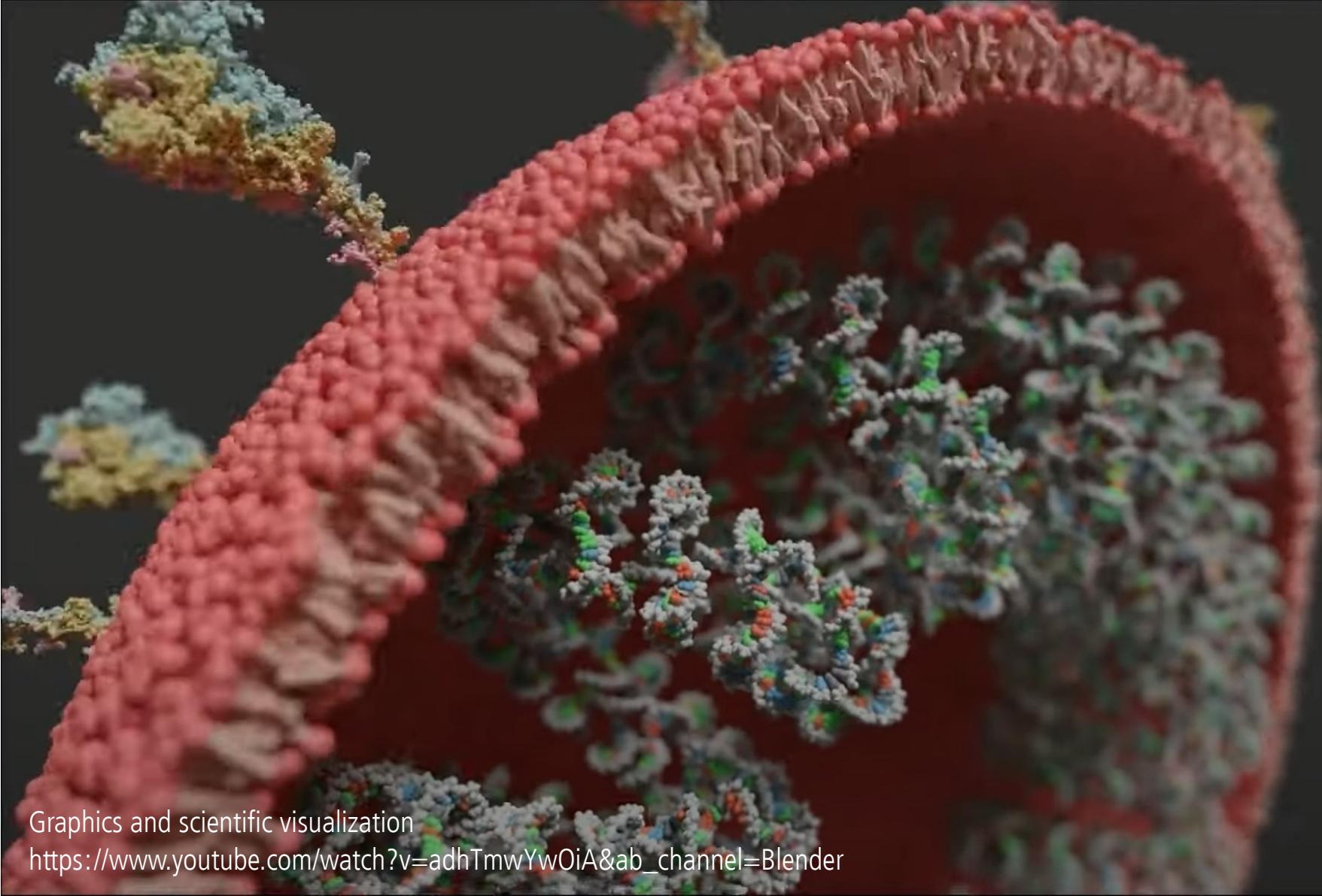


Graphics and computer games

<https://www.rockstargames.com/reddeadredemption2/>

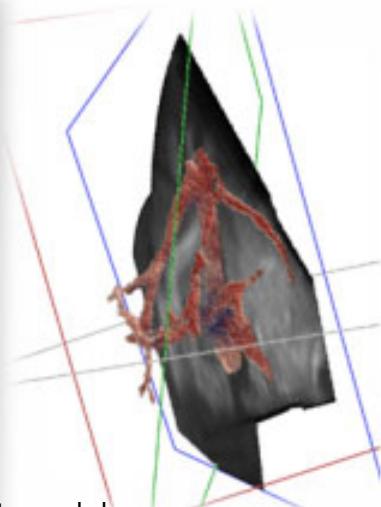
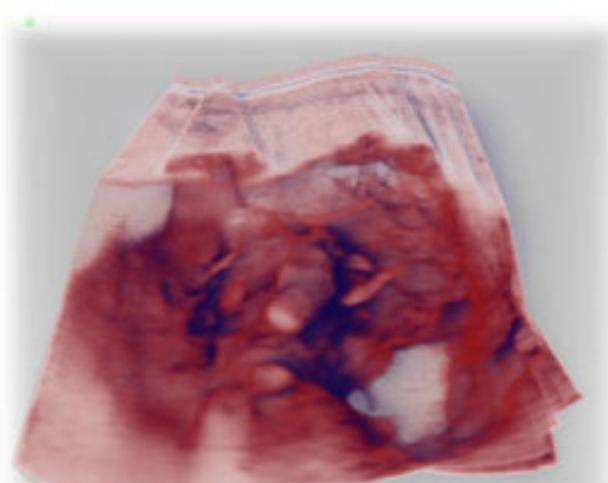
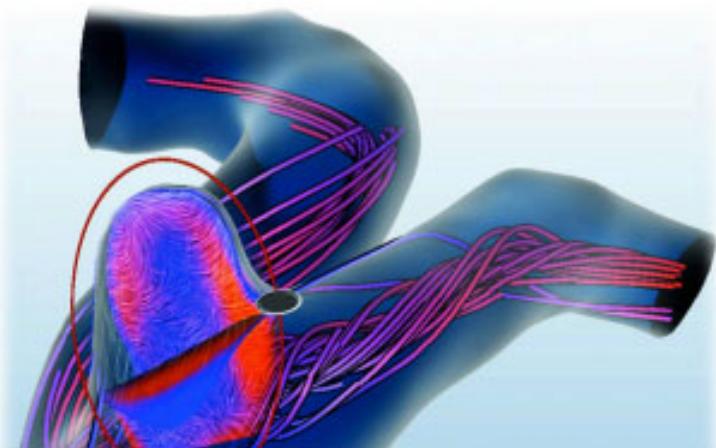
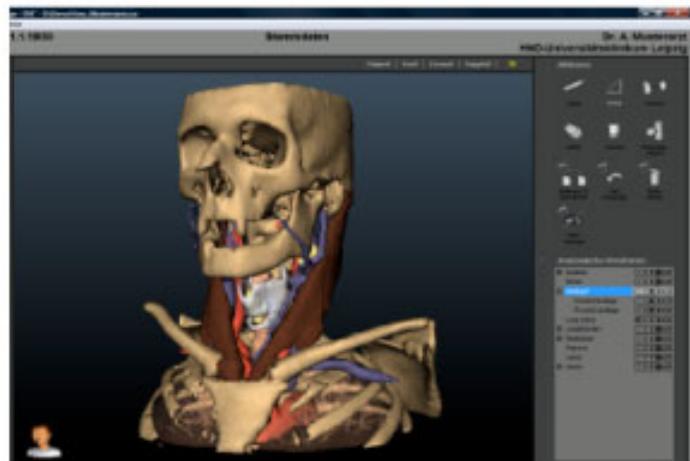
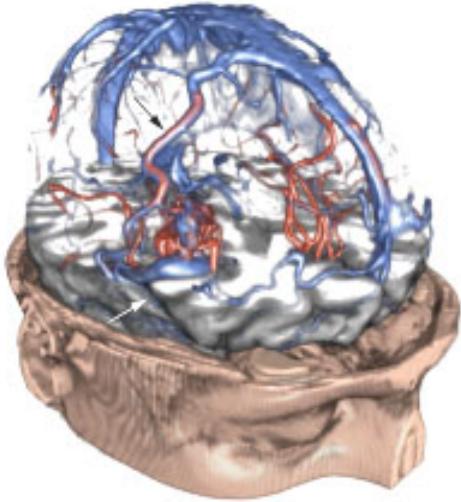


Graphics and animated film
<https://www.pixar.com/soul>



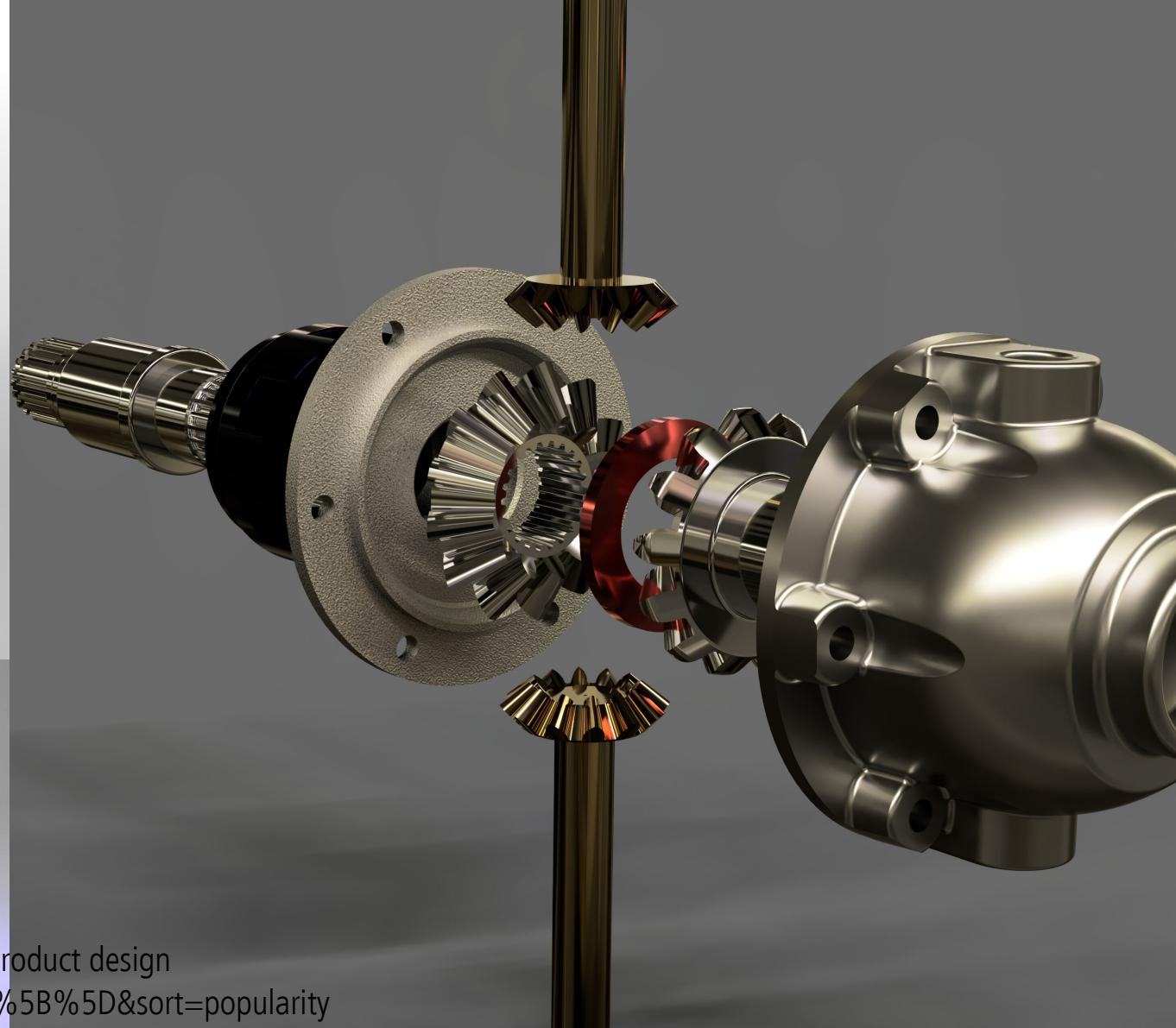
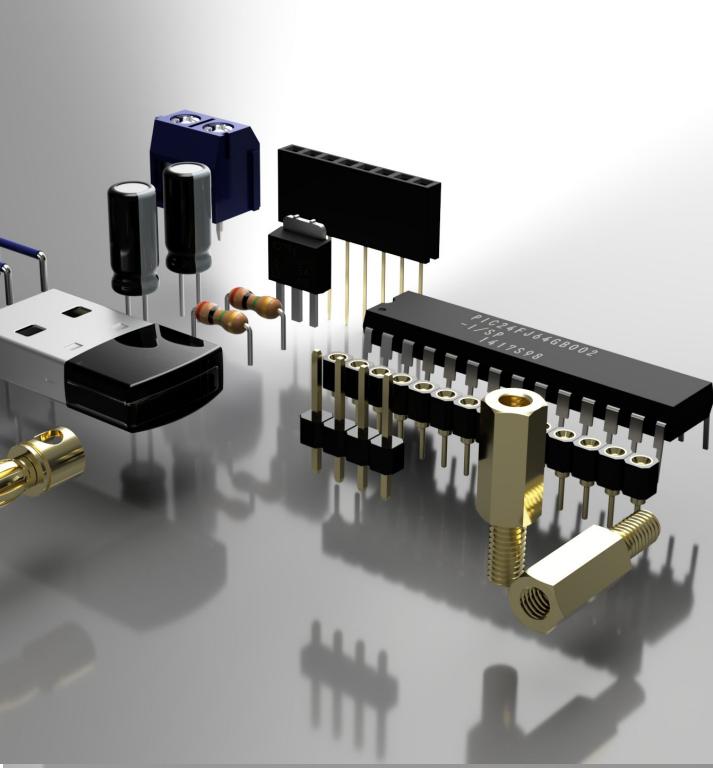
Graphics and scientific visualization

https://www.youtube.com/watch?v=adhTmwYwOjA&ab_channel=Blender

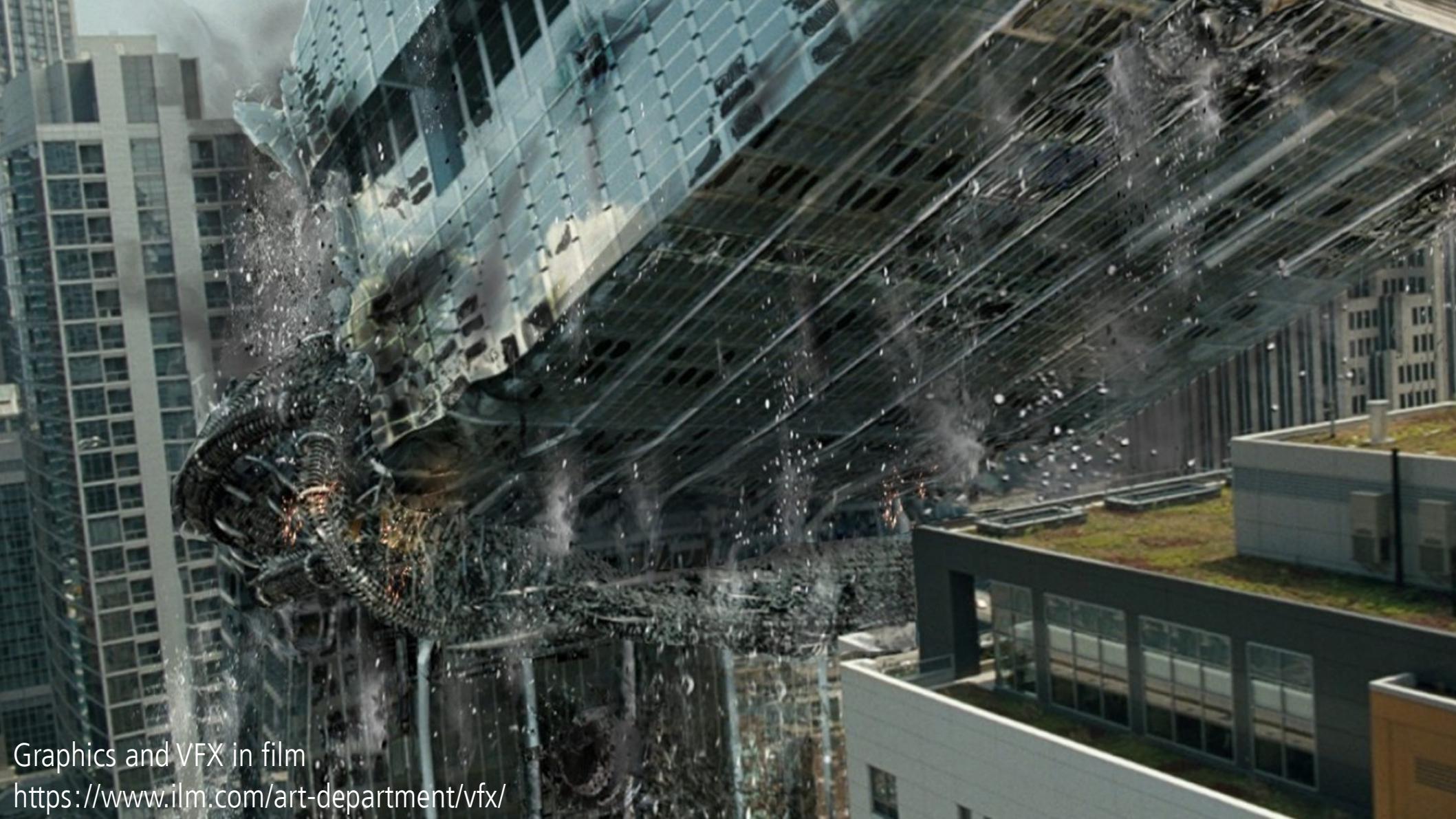


Graphics and medicine

<https://www.siggraph.org/news/eurographics-celebrates-computer-graphics-in-medicine/>



Graphics and CAD, manufacturing, engineering, product design
<https://gallery.autodesk.com/projects/all#filters=%5B%5D&sort=popularity>



Graphics and VFX in film

<https://www.ilm.com/art-department/vfx/>



Graphics and simulation for VFX in film

https://www.youtube.com/watch?v=lS--1gRjfRk&ab_channel=Rebelway



Graphics and motion capture for VFX in film

<https://www.fxguide.com/fxfeatured/weta-digital-s-remarkable-face-pipeline-alita-battle-angel/>



Graphics and arhitecture

<https://www.blenderguru.com/articles/20-jaw-dropping-architectural-renders>



Graphics and interior design; ergonomic design

<https://www.blenderguru.com/articles/20-jaw-dropping-architectural-renders>

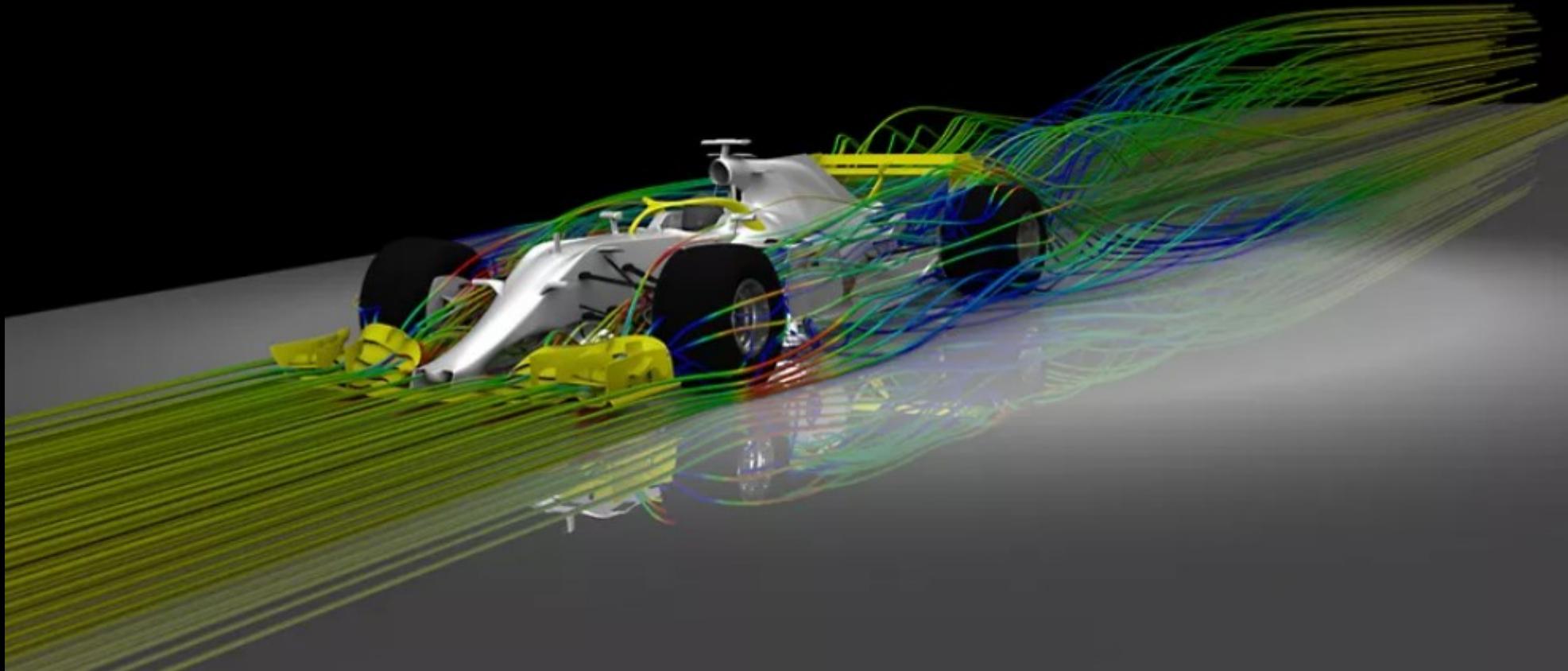


graphics and illumination planning
http://graphics.cs.aueb.gr/graphics/research_lightingopt.html



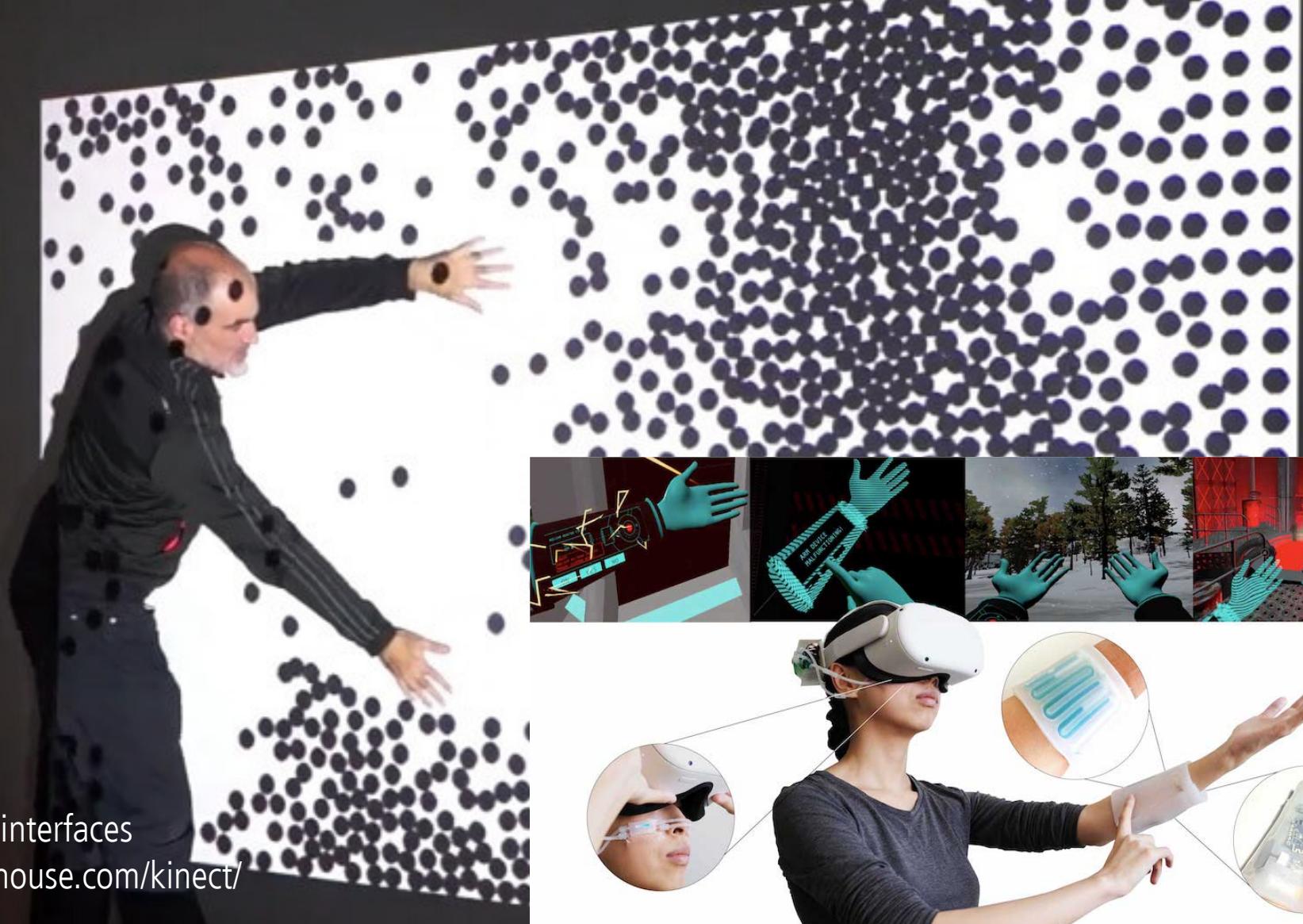


Graphics and new media art
<https://refikanadol.com/>



Graphics and predictive simulations

<https://www.ansys.com/company-information/the-ansys-story>



Graphics and user interfaces
<https://parametrichouse.com/kinect/>
<http://plopes.org/>

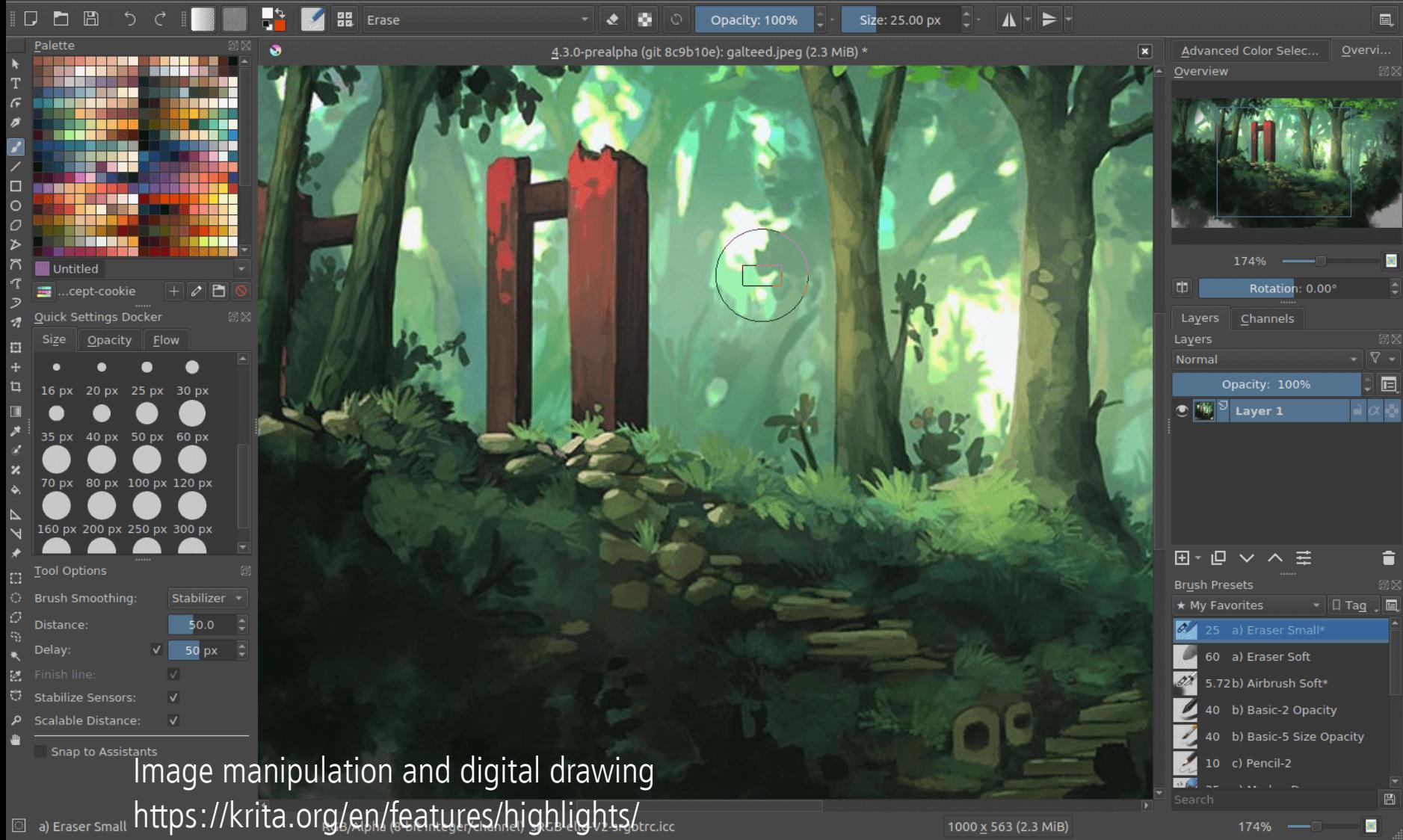
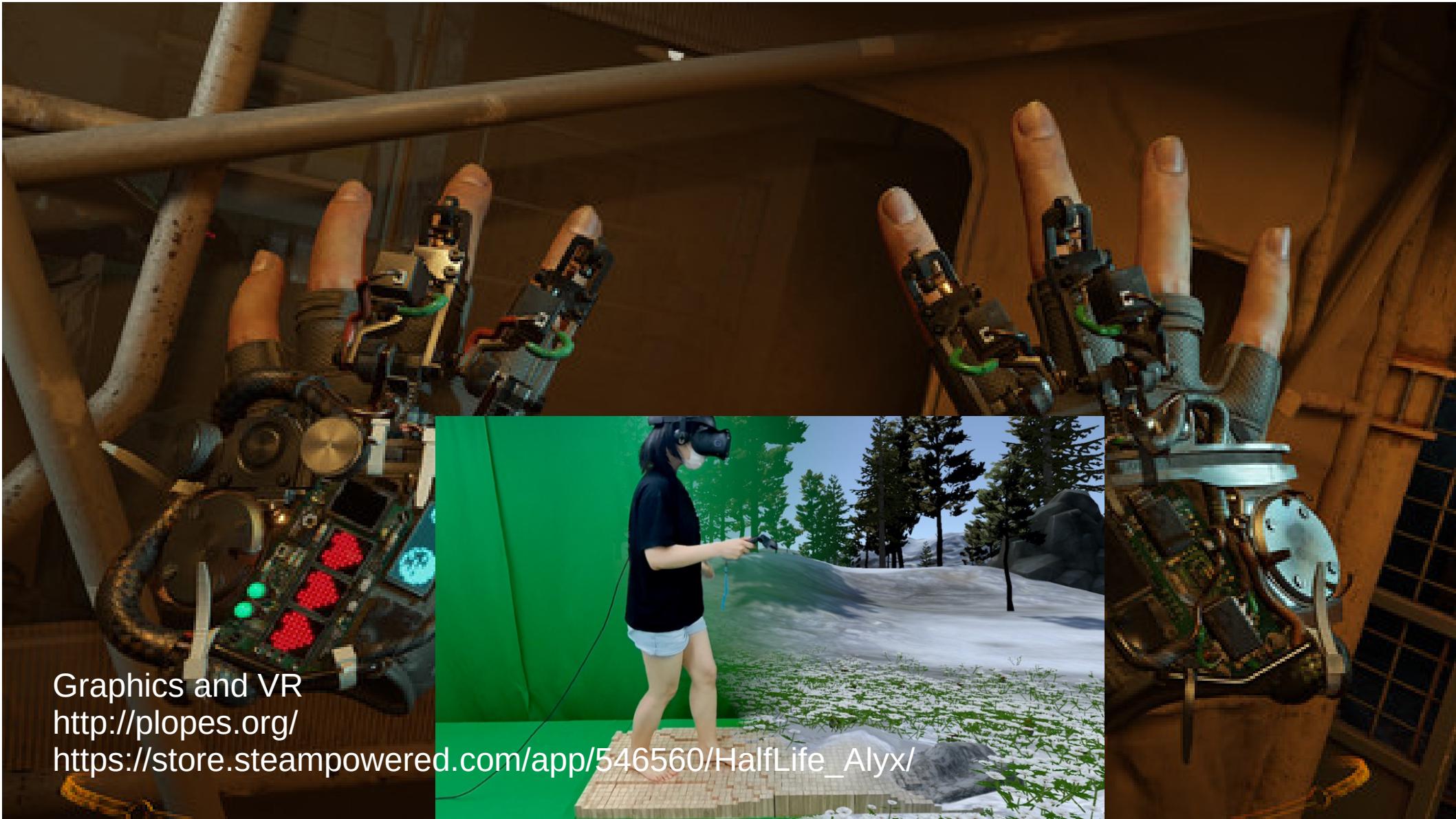


Image manipulation and digital drawing

<https://krita.org/en/features/highlights/>

1000 x 563 (2.3 MiB)



Graphics and VR
<http://plopes.org/>
https://store.steampowered.com/app/546560/HalfLife_Alyx/

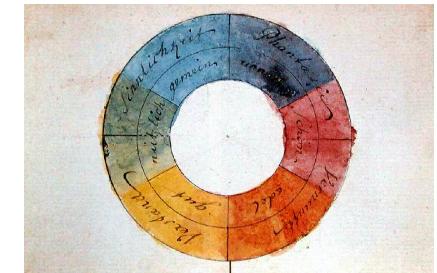
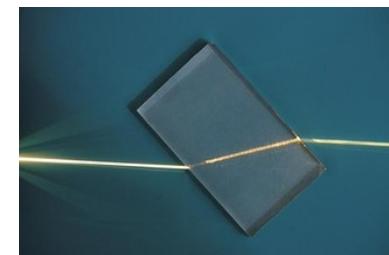
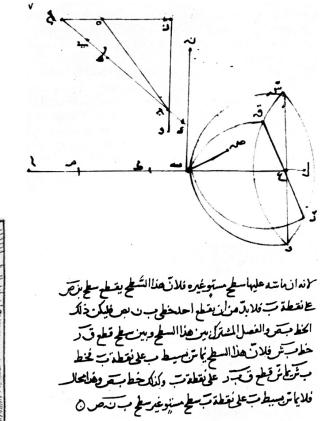
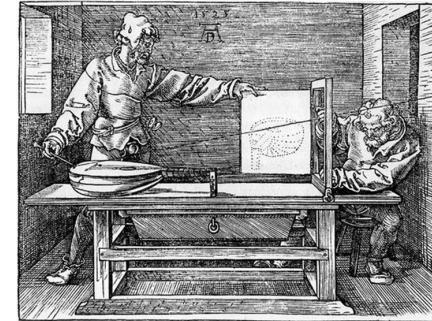


Computer graphics and AR

<https://deltareality.com/portfolio/vaillant-ar-facility-tour/>

Bit of history

- Geometric optics (Alhazen (Ibn al-Haytham), "the father of Optics", 1010)
- Development of perspective projection in drawing: perspective machine (e.g., Albrecht Dürer, 1525)
- Development in optics (Physics, e.g. Young, 1807)
- Johann Wolfgang von Goethe and Theory of Colors, 1810
- Geometry, linear algebra, statistics

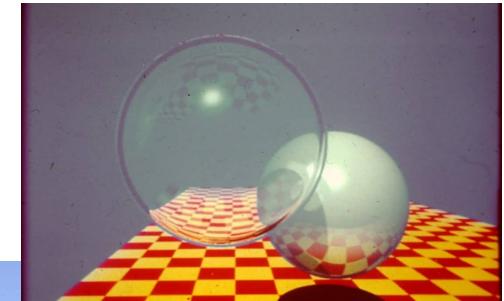


Bit of history

- Computer science and hardware (IBM 2250, 1964)
- VFX (ILM, 1975)
- Ray-tracing (J. Turner Whitted, 1979)
- Animated films (Pixar, 1986)
- Games (Naughty dog, 2013)



INDUSTRIAL
LIGHT & MAGIC



Computer graphics today



<https://www.siggraph.org/>



<https://research.nvidia.com/research-area/real-time-rendering>



<https://www.research.autodesk.com/blog/inside-autodesk-research-explore-our-research-teams/#industry-futures>

Many scenes, two renderers and one image

Glimpse into image generation

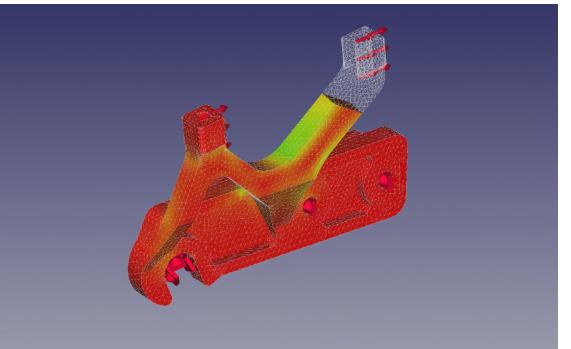
Analogy: taking a photograph

- World surrounding photographer → **3D scene**
- Light interaction with world and image formation → **rendering**
- Result: **Image**



3D scene creation

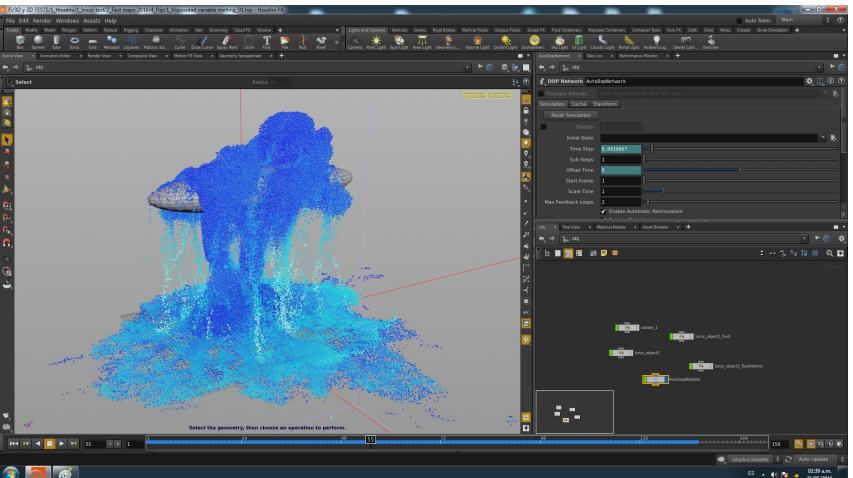
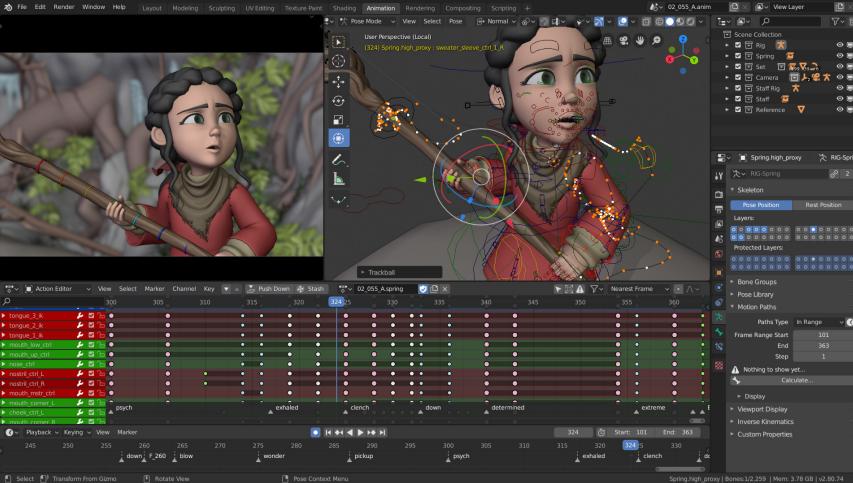
- Modeling: artists, engineers, designers, etc.
 - CAD modeling, sculpting
 - Animation
 - Simulation
- Acquisition from real world
 - Scanning



<https://www.freecadweb.org/>

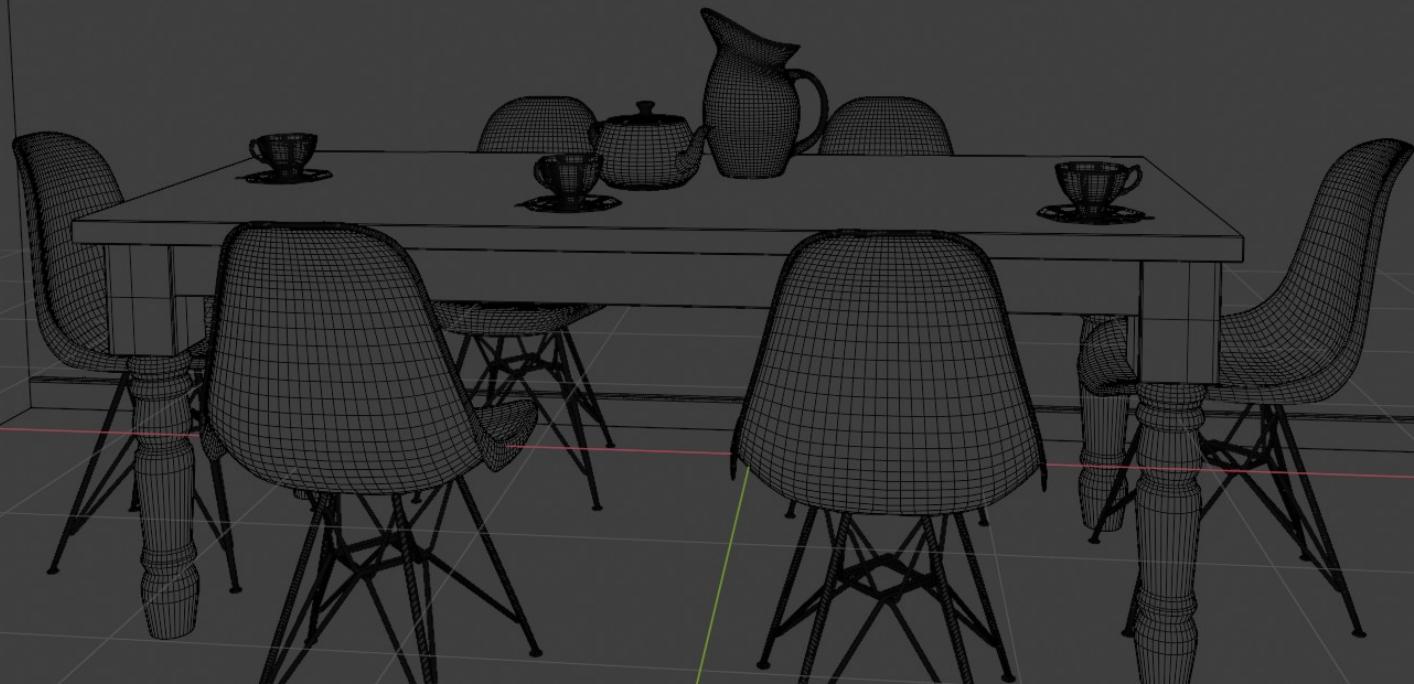


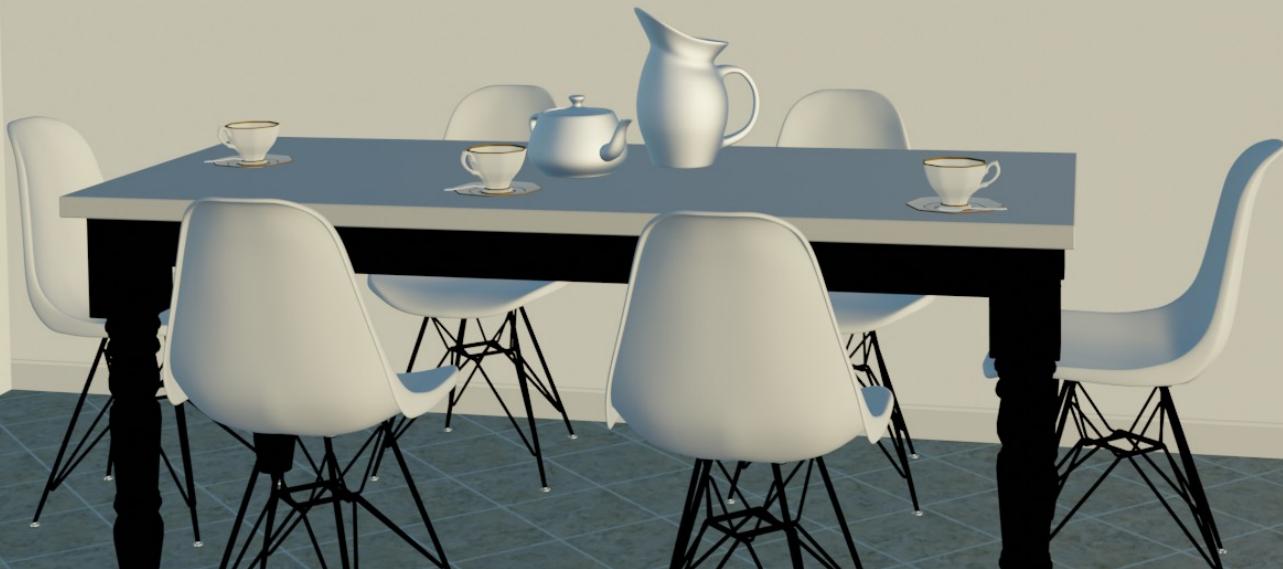
<https://alicevision.org/>



<https://e7p.artstation.com/projects/XbPLD>

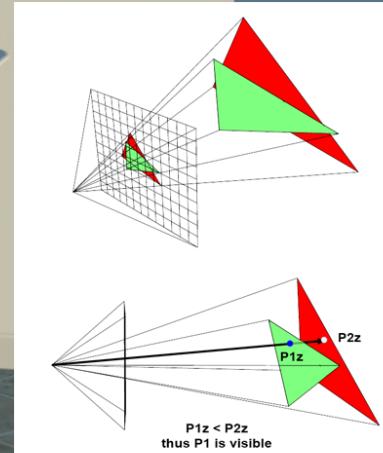
3D scene





A dining room scene featuring a long, rectangular table set for tea. The table is covered with a white cloth and holds a white teapot, a small white pitcher, and four blue and white teacups on saucers. Six chairs with black frames and curved backrests are arranged around the table. The room is dimly lit, with light streaming in from a window on the right side, which has white horizontal blinds. The blinds cast long, thin shadows across the wall and onto the table. A single framed photograph of autumn leaves hangs on the wall above the table.

Rendering: rasterization-based rendering



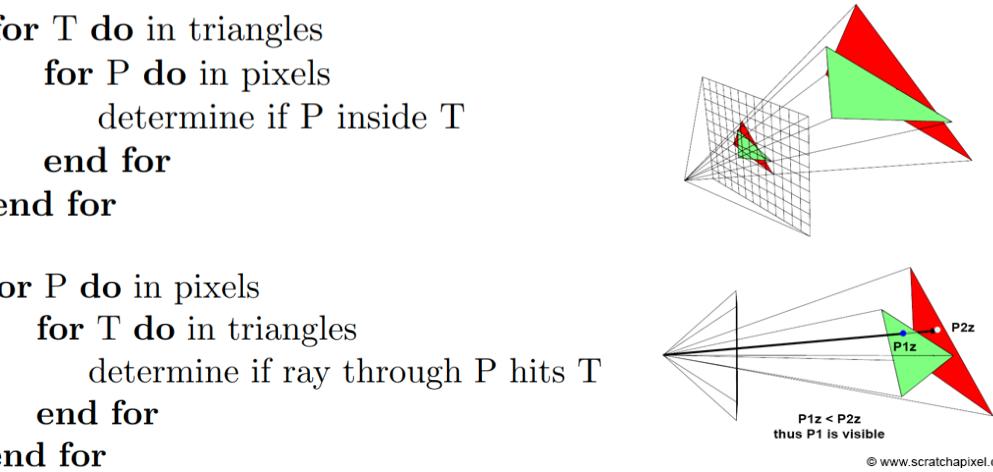
Rendering

- Rasterization-based rendering
- Ray-tracing based rendering



```
for T do in triangles  
  for P do in pixels  
    determine if P inside T  
  end for  
end for
```

```
for P do in pixels  
  for T do in triangles  
    determine if ray through P hits T  
  end for  
end for
```



Rendering

- Rasterization-based rendering: interactivity
- Ray-tracing based rendering: fidelity



Image



Different exposures of the same render: https://docs.blender.org/manual/en/latest/render/color_management.html

Landscape of computer graphics

- <https://www.realtimerendering.com/portal.html>



Real-Time Rendering Portal

Last changed: September 28, 2022

This page is devoted to sites and tools we use on a continuing basis. They're personal picks, and reflect our own biases.

1. [Ke-Sen Huang's conference pages](#) has links for papers from all the major computer graphics conferences and workshops. The pages by Tim Rowley are not available directly, but [this archive](#) contains them.
2. [Advances in Real-Time Rendering in 3D Graphics and Games](#), [Introduction to Real-Time Ray Tracing](#), [Open Problems in Real-Time Rendering](#), [An Overview of Next-Generation Graphics APIs](#), and [Stylized Rendering in Games](#) SIGGRAPH course materials are hosted on our site.
3. [SIGGRAPH 2021 links](#), compiled by Stephen Hill. Also see link pages for [SIGGRAPH 2020 links](#) [SIGGRAPH 2019 links](#) (and [SIGGRAPH 2019 ray tracing links](#)), [SIGGRAPH 2018](#), [SIGGRAPH 2017](#), [SIGGRAPH 2016](#), [SIGGRAPH 2015](#), [SIGGRAPH 2014](#), [SIGGRAPH 2013](#), [SIGGRAPH 2012](#) and [SIGGRAPH 2011](#).
4. [Game Developers Conference 2019 links](#), also [2018](#), [2017](#), [2016](#). There's none for 2015, but before then Javier "Jare" Arevalo collected [GDC 2014](#), [2013](#), and [2012](#) presentations. Also see the [GDC Vault](#).
5. [Graphics Programming weekly](#) - Jendrik Illner summarizes graphics blog articles. Think of it as your one-stop blog. He also has a nice [searchable collection](#).
6. [NVIDIA](#) (and [NVIDIA Research](#)), and [AMD](#) (plus [GPUOpen](#)) graphics developer sites - demos, code samples, white papers, etc. Other worthwhile code samples at [Humus-3D](#).
7. Min Chen's list of [Computer Graphics Forum](#) State-of-the-Art (STAR), survey, and review papers since 2010.
8. [The Journal of Computer Graphics Techniques](#) - open access (free to all) and many articles include code samples.
9. [Journal of Graphics Tools](#) (JGT) code repository.
10. [Graphics Gems Repository](#) - contains the source code for many graphics algorithms. Search the contents by [category](#), by [author](#), or by [book](#).
11. [Developer sites and mailing lists](#): [GameDev.net](#) is active, as is [OpenGL.org](#), [Ogre Forums](#), [GD Algorithms](#) archives dying out but searchable ([subscribe](#)), and [FlipCode](#) (old, closed, but some good things in the archives).
12. [Game company publication pages](#): alphabetically, and a few quite dated, but here goes. [Frostbite](#), [Guerrilla Games](#), [Unreal Engine](#), [Unity](#), [Ready At Dawn](#), [Tri-Ace](#), and [Activision](#).
13. [Film company publication pages](#): [Disney](#) ([Hyperion renderer specific](#)) and [Pixar](#).
14. [Commercial research lab pages](#): [Microsoft Research Asia](#), [Microsoft Research U.S.](#), and [Cesium](#) (GIS).
15. The [Level Up Report](#) by Mark DeLoura is a free weekly that provides pointers to all sorts of developments and resources for learning through games, coding, and making.
16. [Level 80](#) has a constant stream of information for game artists and content creators.
17. [Free \(and good\) books online](#)

Course organization

Course note

- This course is not about art, design, game-development, film, visualization for engineering and science domains, etc.
- This course gives foundations for creating imagery for arbitrary discipline
- Computer graphics is a tool to create beautiful imagery - but tool alone is not enough to create those images!
 - If you are interested in applying graphics to specific domain area (game, film, sci-vis, etc.) I encourage you to obtain the required domain knowledge as well!

About lecturer

- My background
- My interests
- My experiences

About you

- What is computer graphics for you?

Course contents

- Lectures
- Project
- Exam

Lectures plan

10 lectures; 3 x 45min

- Lecture 1: introduction (this) and rendering overview
- Lecture 2: 3D scene – transforms and shapes
- Lecture 3: 3D scene – materials: scattering and texture
- Lecture 4: 3D scene - lights and cameras
- Lecture 5: Rendering - rendering introduction and raytracing
- Lecture 6: Rendering - rasterization
- Lecture 7: Images – image + project
- Lecture 8: More on 3D scene + project
- Lecture 9: More on rendering + project
- Lecture 10: More on image + project

Lectures

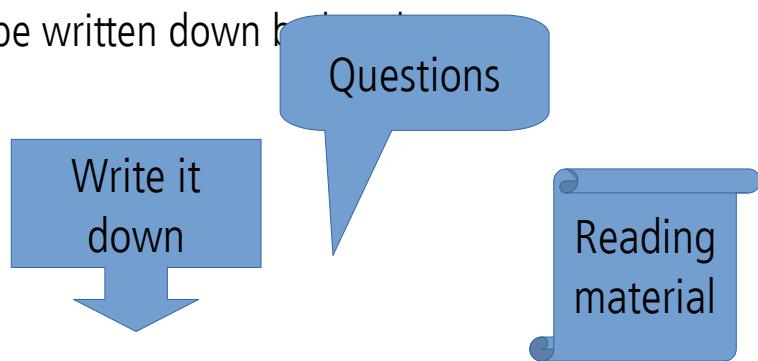
- The point of lectures is to give the structure and to give highlights on foundations
 - They are starting point, a map, which is for you to explore.
 - Consultations are always possible!
- During lectures write down important points ask if anything is unclear
 - Feel free to interrupt! If something is not clear for you it is a high possibility it is also not clear for someone else – it is good to repeat!

Map of computer graphics

- Point of lectures is to give structure and it is up to you to fill it.

Note for lectures

- All information will be on slides
- Important knowledge will be written on board
- For each lecture, Moodle questions we be available for learning for exam
 - Important elements from lecture
- Note for slides
 - Slides were intended for both lectures and as reading material. Therefore, some slides contain a lot of text which is intended for student to read at home. Those slides will have special icon.
 - Important elements will be highlighted and noted that they should be written down by the student.
 - Your interaction is crucial for best learning experience



Tools for learning and experimenting

- Blender is our friend: <https://www.blender.org/>



Projects

- During lectures we will cover wide range of methods and ideas conceptually
- During project work, you will have time to dive deeper into technical aspects and implementations of what is the most interesting to you.
- Projects: a time dedicated to experience real-life development (and research).
 - Projects can be started even after this lecture: decide on topic and start investigating!.
 - Projects should be time for you to research and work on your own - consultations are always possible!
 - Projects are made to be fun and engaging: choose what you like!

Projects

- TODO: shortly describe projects + add images of what results might look like
 - Low level of abstraction: coding and focus on rendering
 - High level of abstraction: modeling, animation and interaction in DCC or game engine
- <https://github.com/lorentzo/IntroductionToComputerGraphics/wiki/Projects>

Course outcomes

- Understand fundamental concepts and theory: those are immortal
- Typically mathematics, algorithms and methods used in computer graphics
- Understanding how foundations are supporting technology and which technology exists for you to create
- Map of the computer graphics

Special outcomes

- “Seeing world with different eyes”
 - Example: shadows – with and without
- Foundations for expression and visualization

Course web-site and materials

- Course web-site
 - <https://github.com/lorentzo/IntroductionToComputerGraphics>
 - All materials are available in advance with most recent updates
 - TIP: read materials before lecture – it helps for following the lecture

Additional learning material

- <https://www.realtimerendering.com/>
- <https://www.scratchapixel.com/>
- <https://learnopengl.com/>
- <https://raytracing.github.io/>
- <https://pbrt.org/>

Course grading

- Project
 - Technical contribution and documentation
- Exam
 - 30min, theory, multiple choices type of exam