

COL781

# Computer Graphics

# **What is computer graphics?**

# Games



# Animated films



# Visual effects



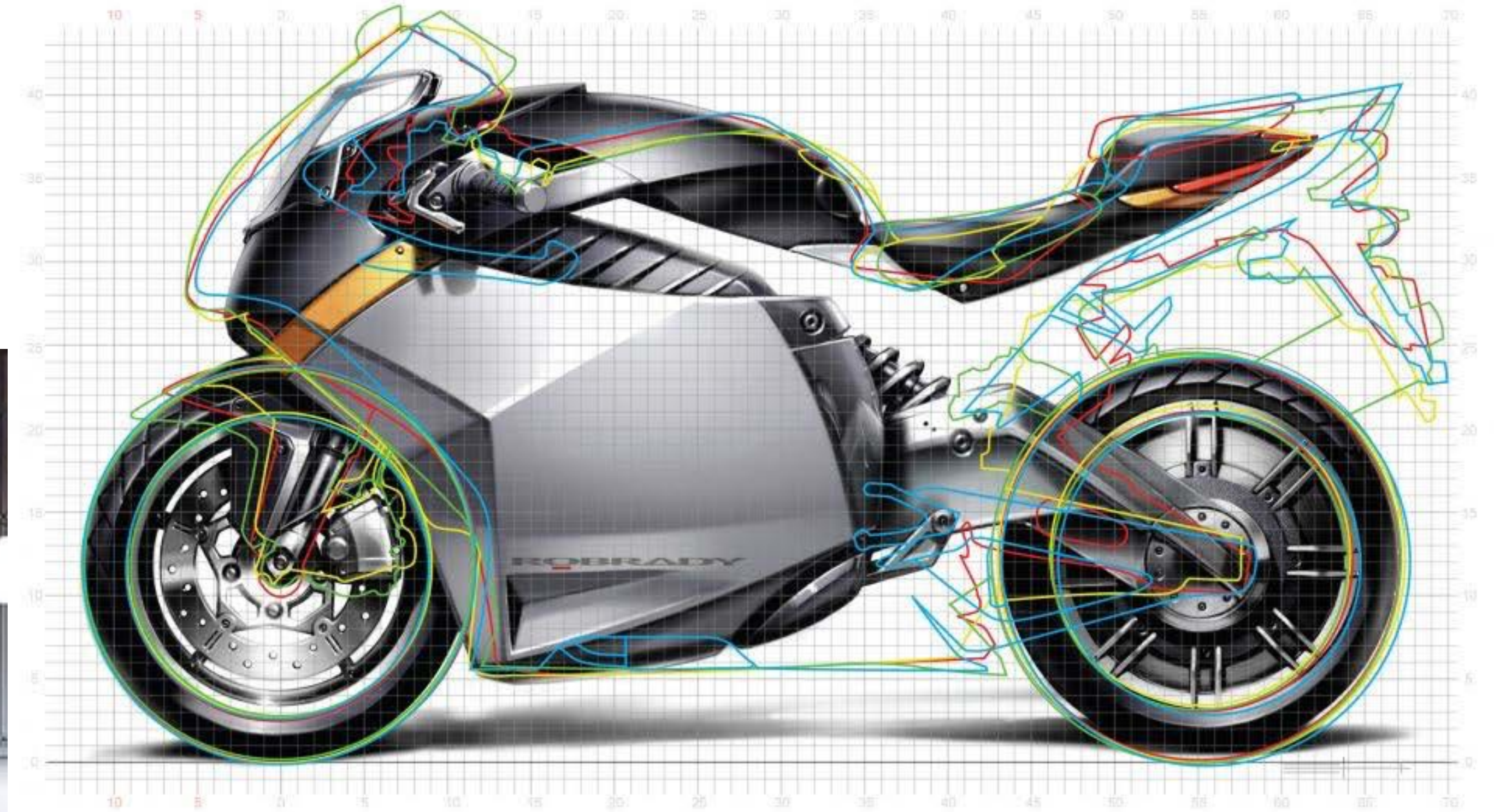
# What is computer graphics?

Computational methods to create and manipulate **visual information**.

# Art and design



# Industrial design

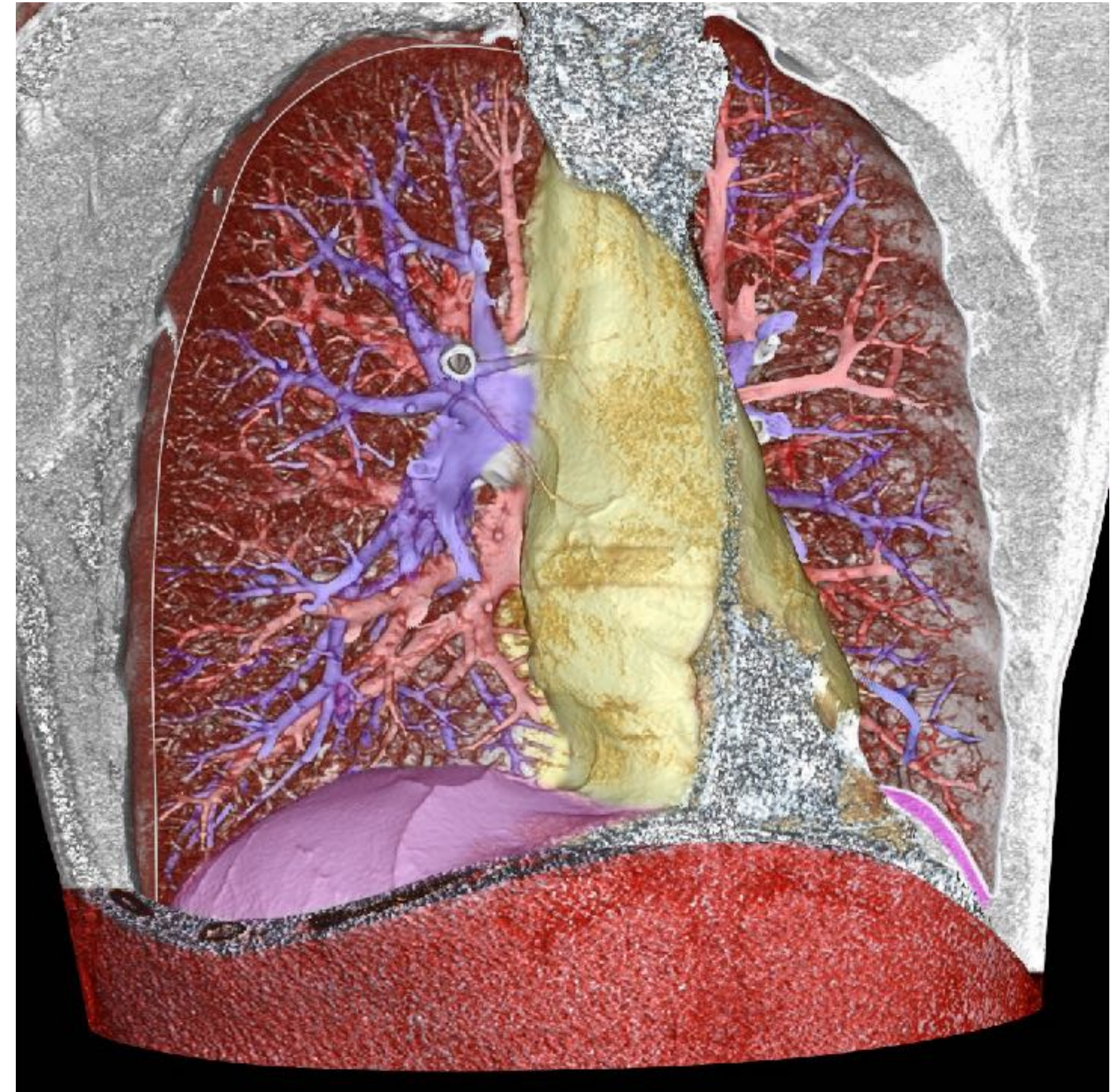
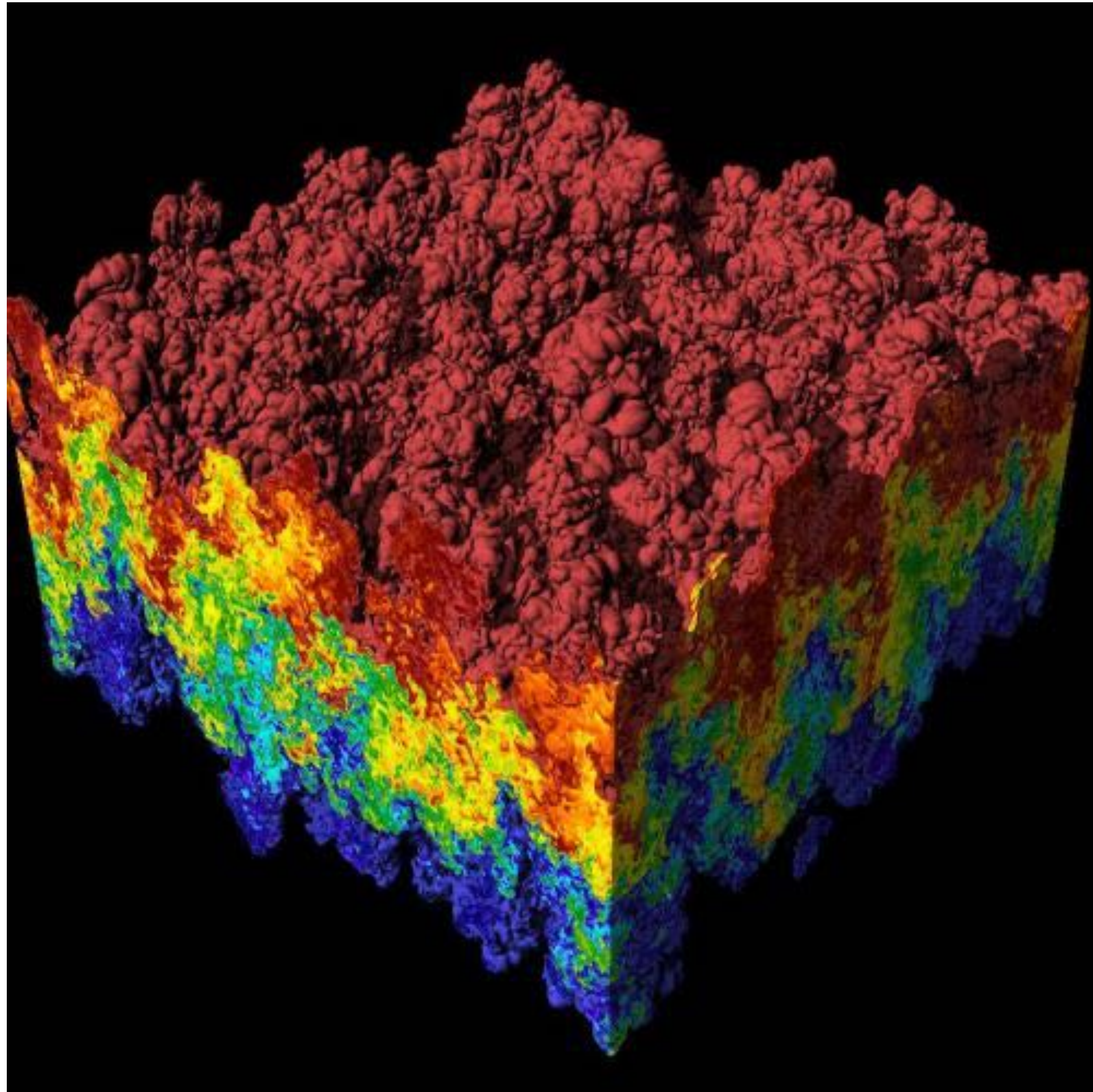




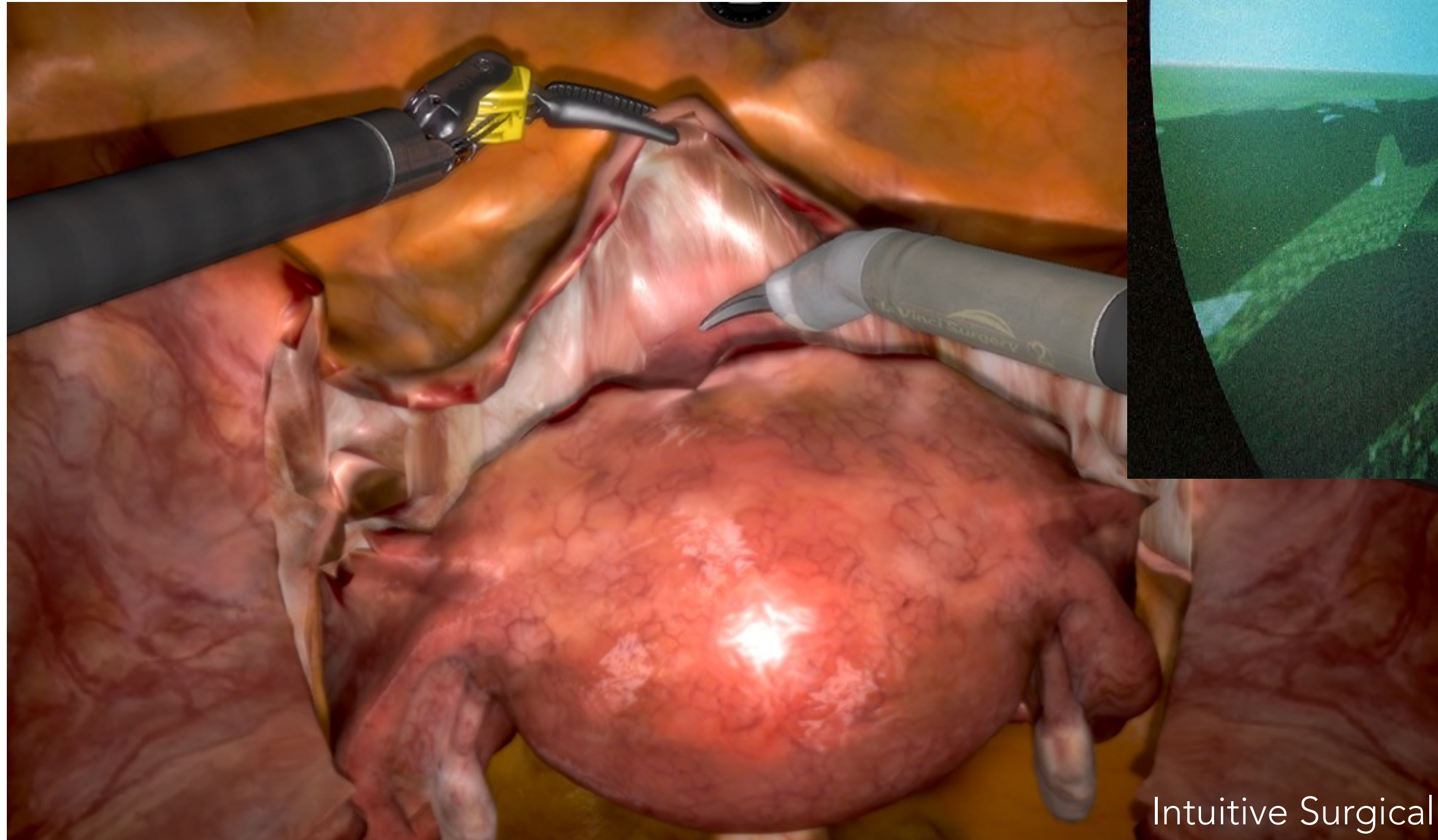
# Architecture



# Scientific and medical visualization



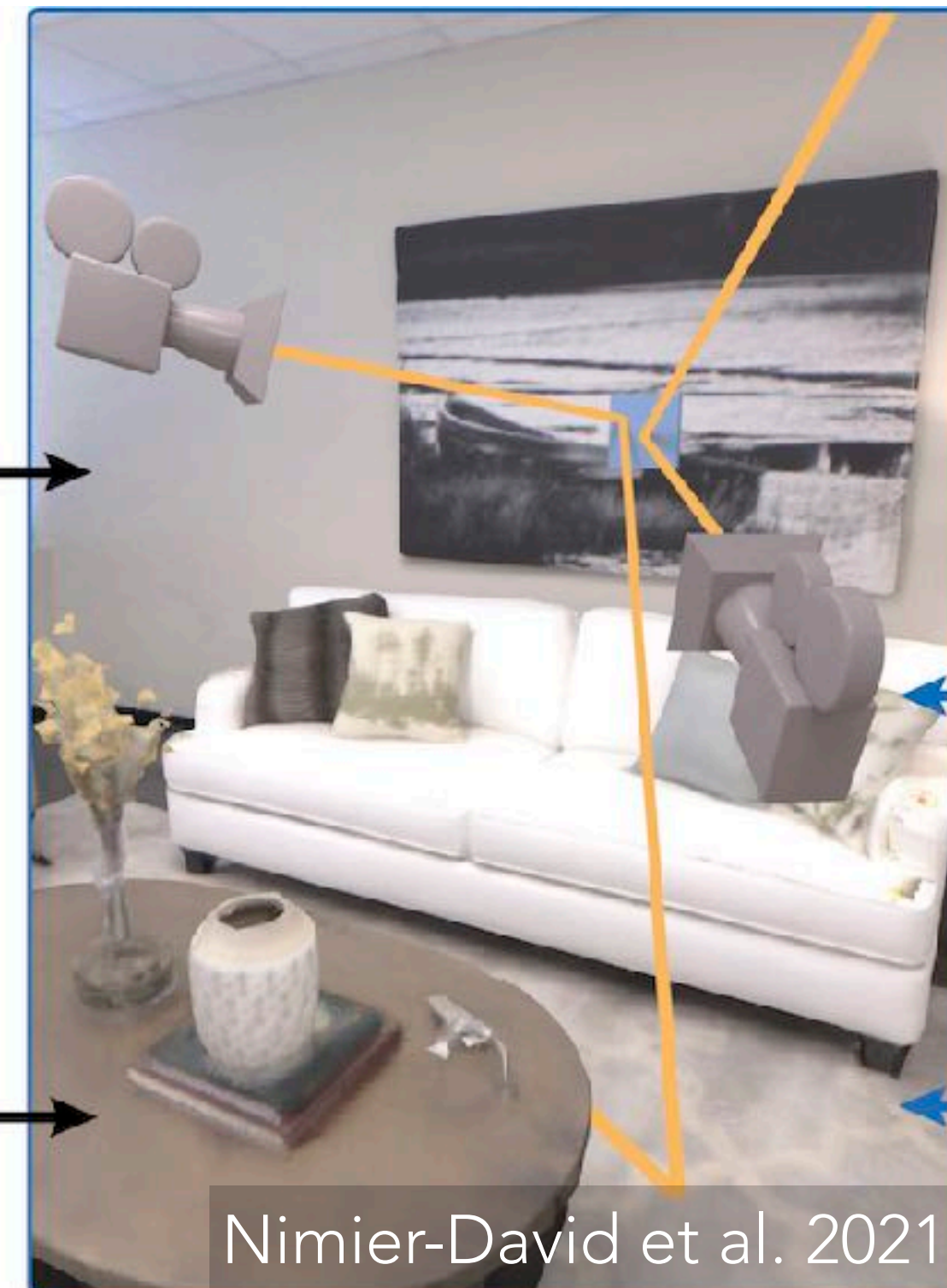
# Education and training



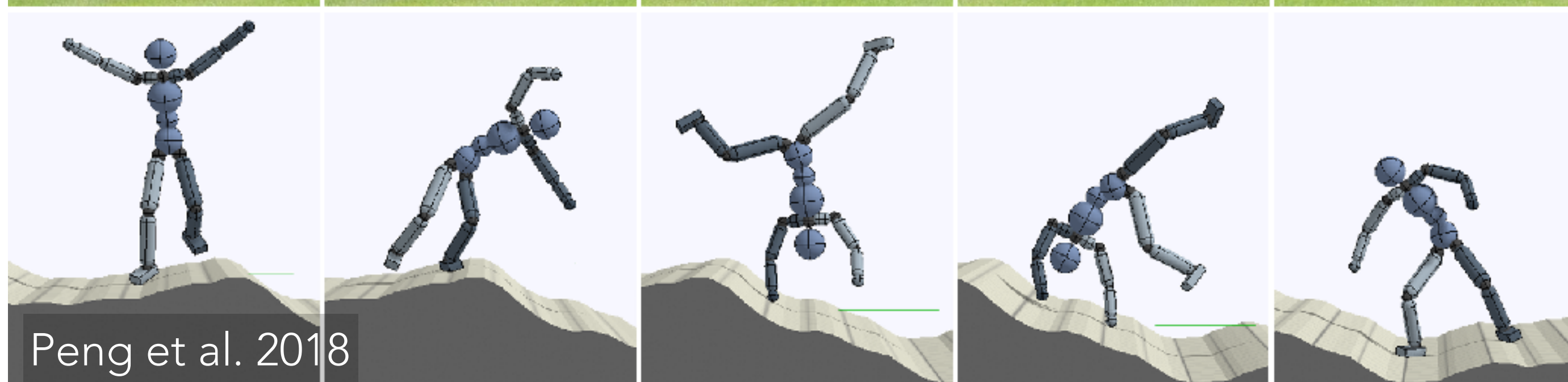
# Inverse graphics



Reconstructed geometry



Nimier-David et al. 2021



Peng et al. 2018

# Virtual and augmented reality



Oculus Quest 2



Microsoft HoloLens concept art



**SIGGRAPH 2023**  
**LOS ANGELES+ 6-10 AUG**

**TECHNICAL PAPERS TRAILER**

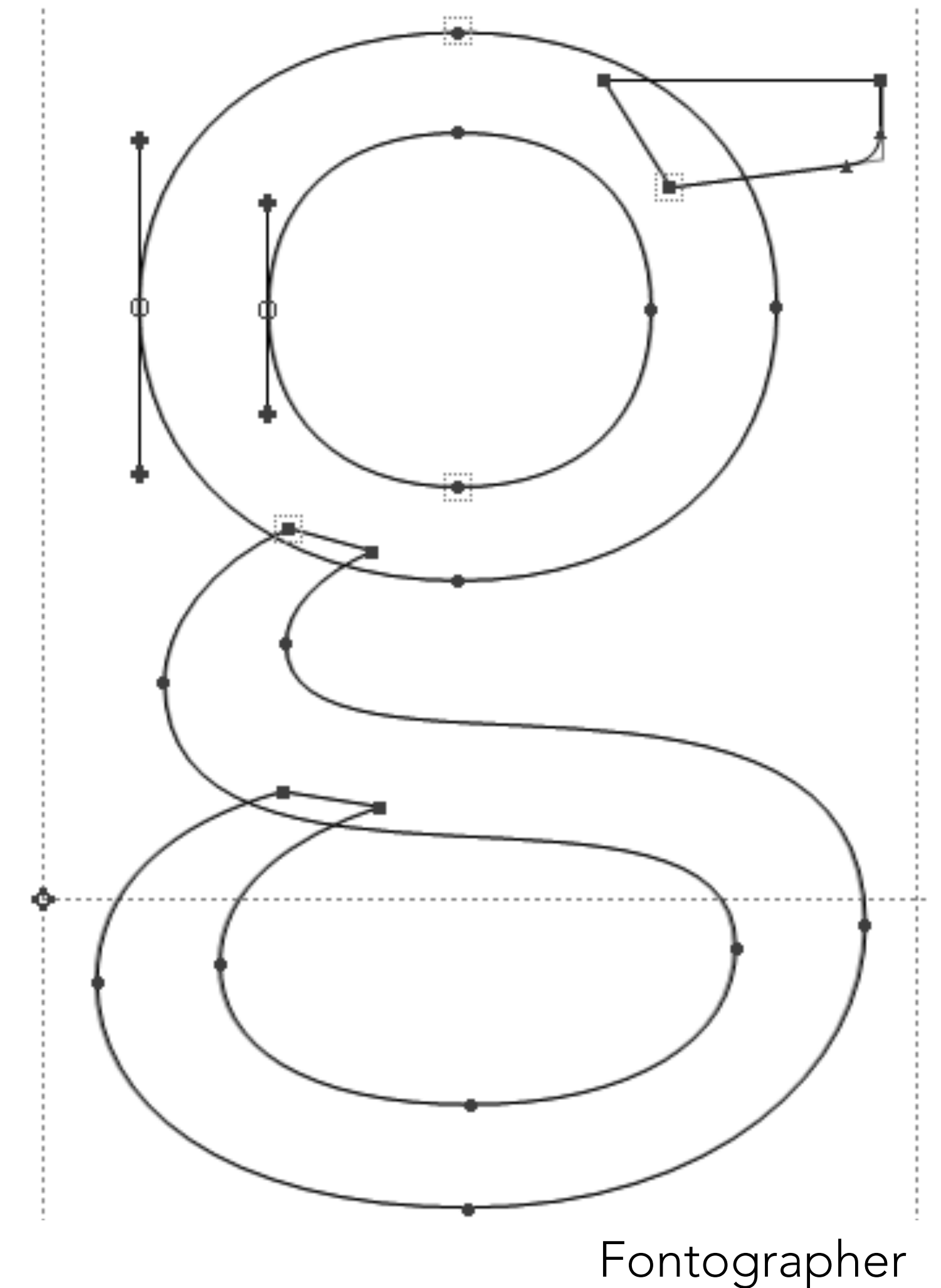
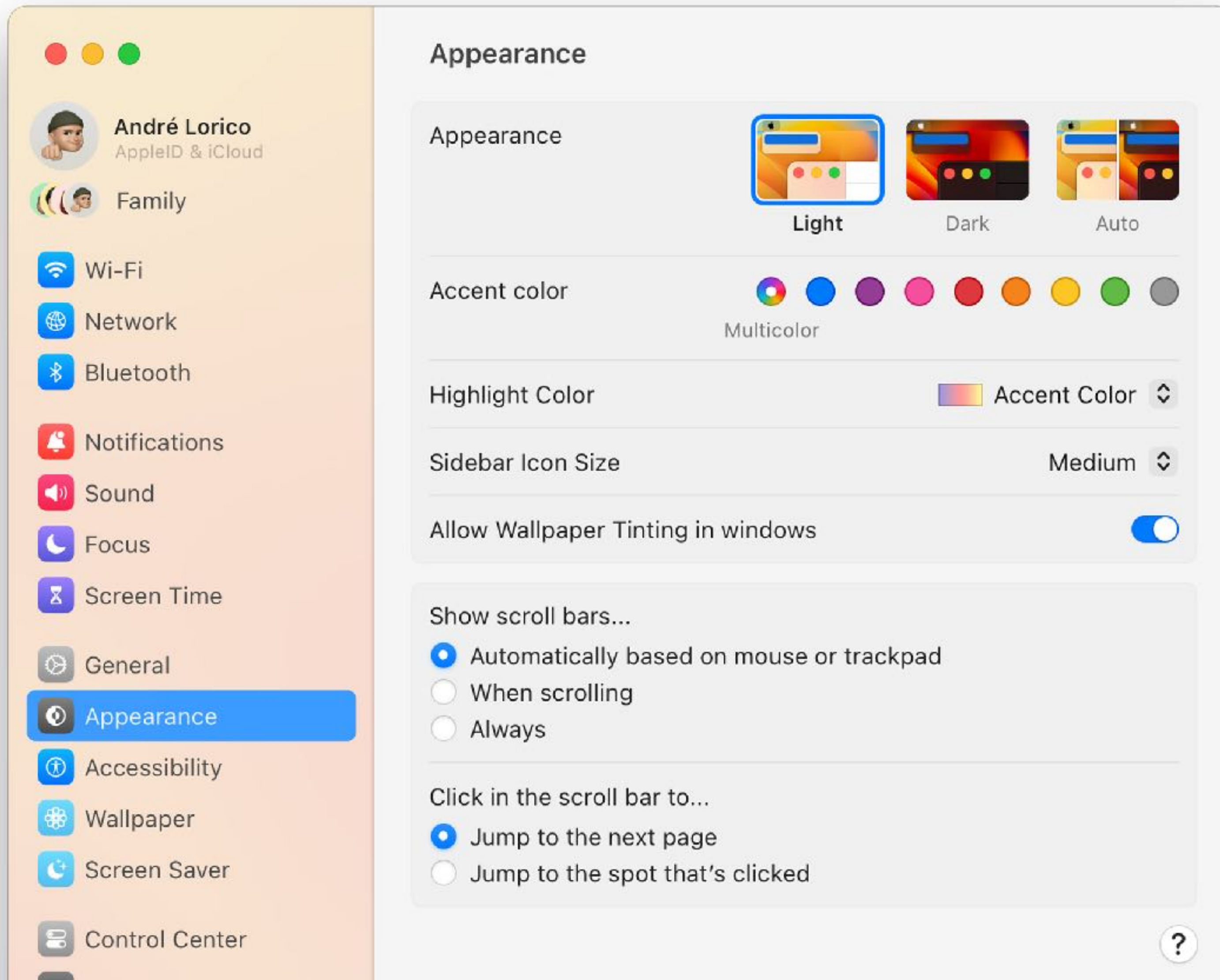
**THE PREMIER CONFERENCE & EXHIBITION ON  
COMPUTER GRAPHICS & INTERACTIVE TECHNIQUES**



Actually, computer graphics is **omnipresent** in how all of us interact with computers today!

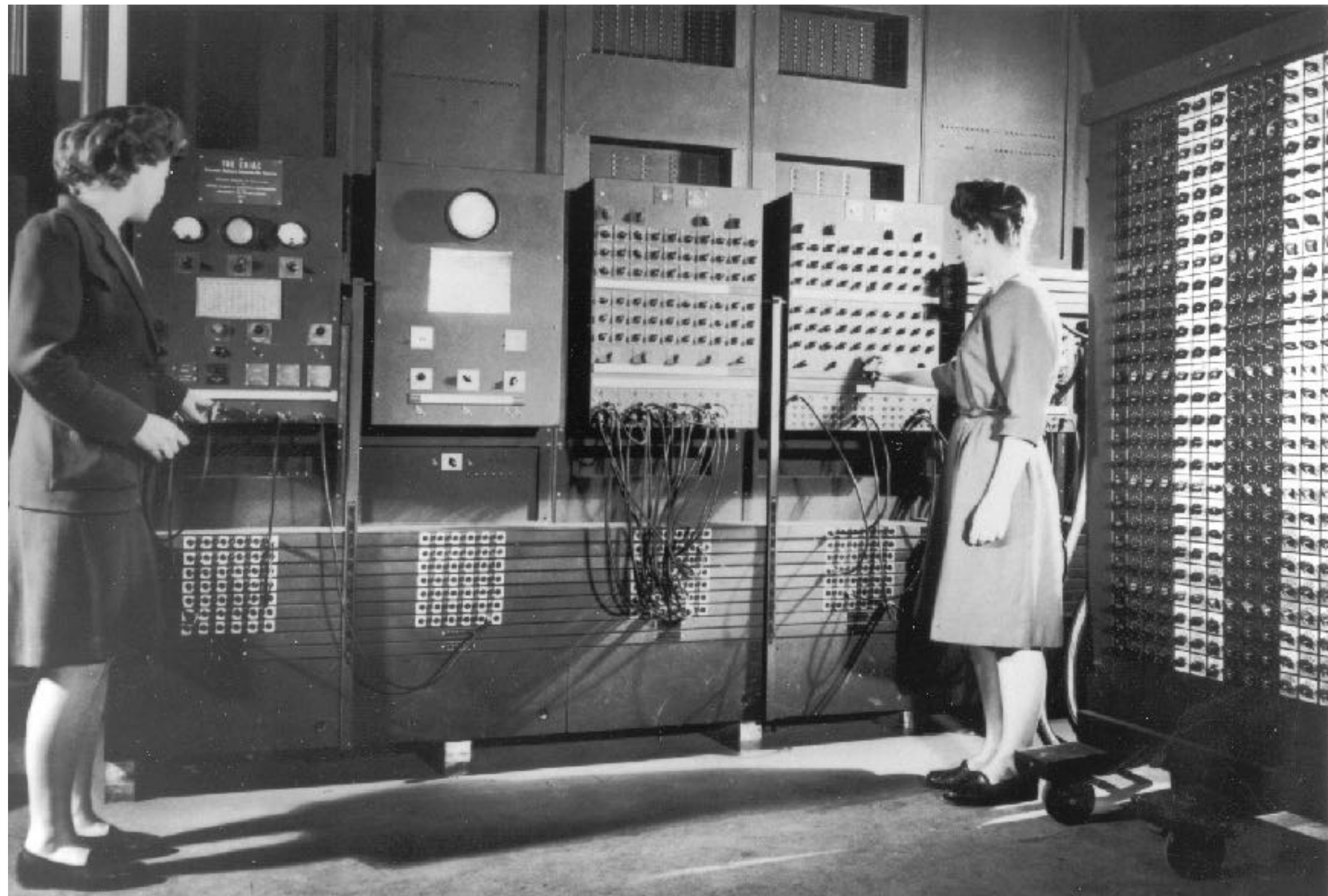
...Wait, how?

# Graphical user interfaces, typography

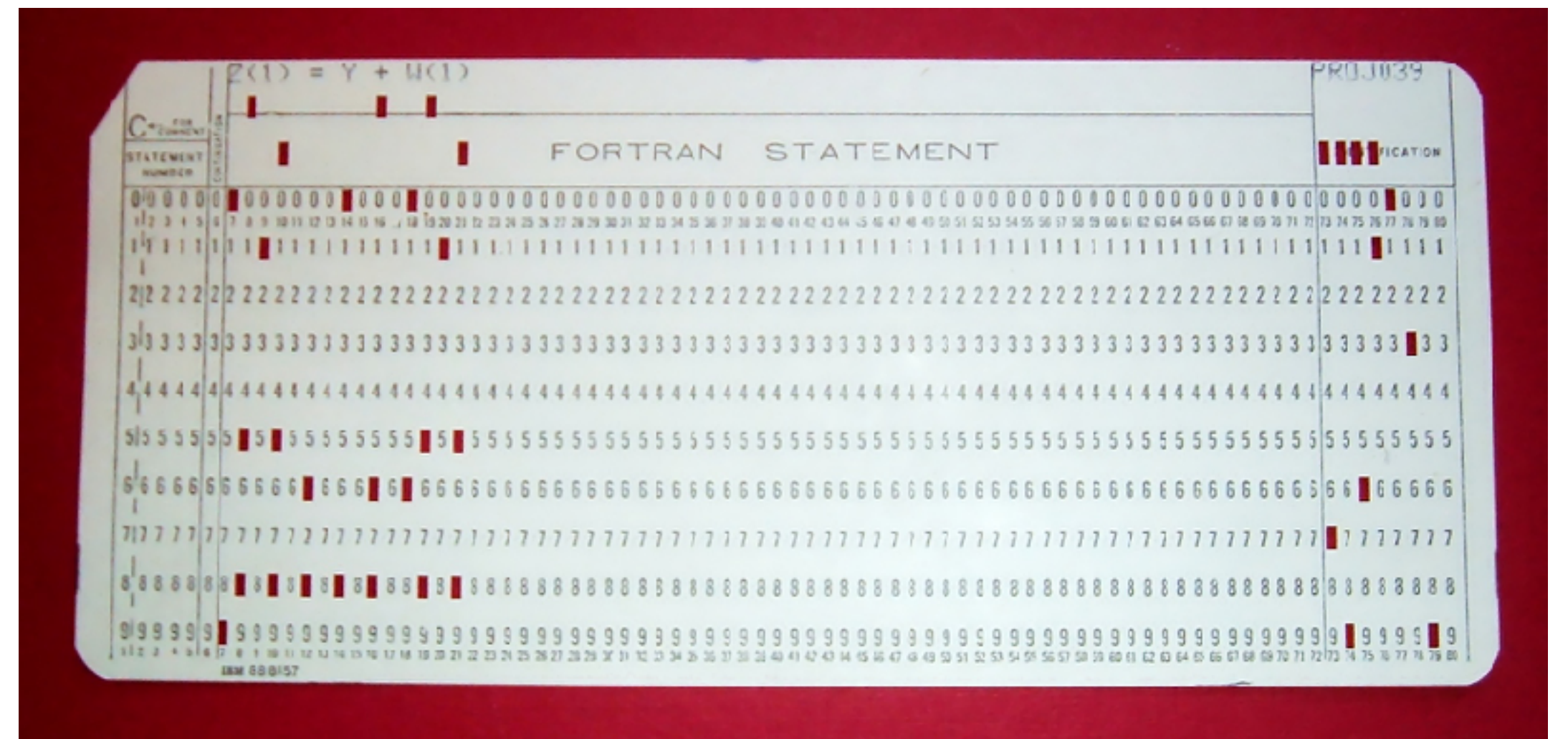




# Computing without graphics



ENIAC (1945)



Punched card from a Fortran program

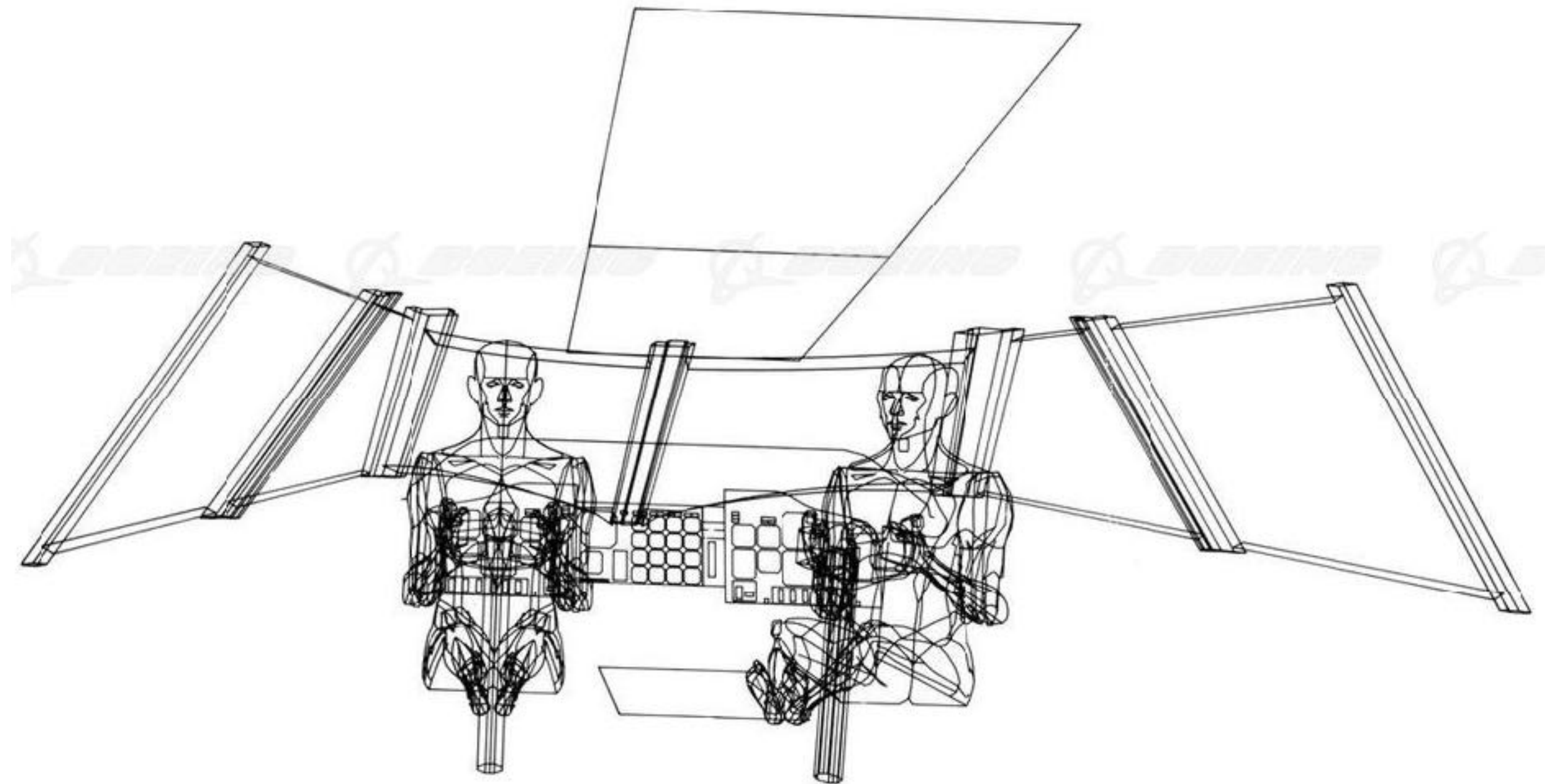
# Computing without graphics

```
[root@localhost ~]# ping -q fa.wikipedia.org
PING text.pmtpa.wikimedia.org (208.80.152.2) 56(84) bytes of data.
^C
--- text.pmtpa.wikimedia.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.585/0.526/0.544/0.280 ms
[root@localhost ~]# pwd
/root
[root@localhost ~]# cd /var
[root@localhost var]# ls -la
total 72
drwxr-xr-x. 18 root root 4096 Jul 30 22:43 .
drwxr-xr-x. 23 root root 4096 Sep 14 20:42 ..
drwxr-xr-x.  2 root root 4096 May 14 00:15 account
drwxr-xr-x. 11 root root 4096 Jul 31 22:26 cache
drwxr-xr-x.  3 root root 4096 May 18 16:03 db
drwxr-xr-x.  3 root root 4096 May 18 16:03 empty
drwxr-xr-x.  2 root root 4096 May 18 16:03 games
drwxrwx--T.  2 root gdm  4096 Jun  2 18:39 gdm
drwxr-xr-x. 38 root root 4096 May 18 16:03 lib
drwxr-xr-x.  2 root root 4096 May 18 16:03 local
lrwxrwxrwx.  1 root root    11 May 14 00:12 lock -> ../run/lock
drwxr-xr-x. 14 root root 4096 Sep 14 20:42 log
lrwxrwxrwx.  1 root root    10 Jul 30 22:43 mail -> spool/mail
drwxr-xr-x.  2 root root 4096 May 18 16:03 nis
drwxr-xr-x.  2 root root 4096 May 18 16:03 opt
drwxr-xr-x.  2 root root 4096 May 18 16:03 preserve
drwxr-xr-x.  2 root root 4096 Jul  1 22:11 report
lrwxrwxrwx.  1 root root     6 May 14 00:12 run -> ../run
drwxr-xr-x. 14 root root 4096 May 18 16:03 spool
drwxrwxrwt.  4 root root 4096 Sep 12 23:50 tmp
drwxr-xr-x.  2 root root 4096 May 18 16:03 yp
[root@localhost var]# yum search wiki
Loaded plugins: langpacks, presto, refresh-packagekit, remove-with-leaves
rpmfusion-free-updates | 2.7 kB | 00:00
rpmfusion-free-updates/primary_db | 206 kB | 00:04
rpmfusion-nonfree-updates | 2.7 kB | 00:00
updates/metalink | 5.9 kB | 00:00
updates | 4.7 kB | 00:00
updates/primary_db 73% [=====] | 62 kB/s | 2.6 MB | 00:15 ETA
```

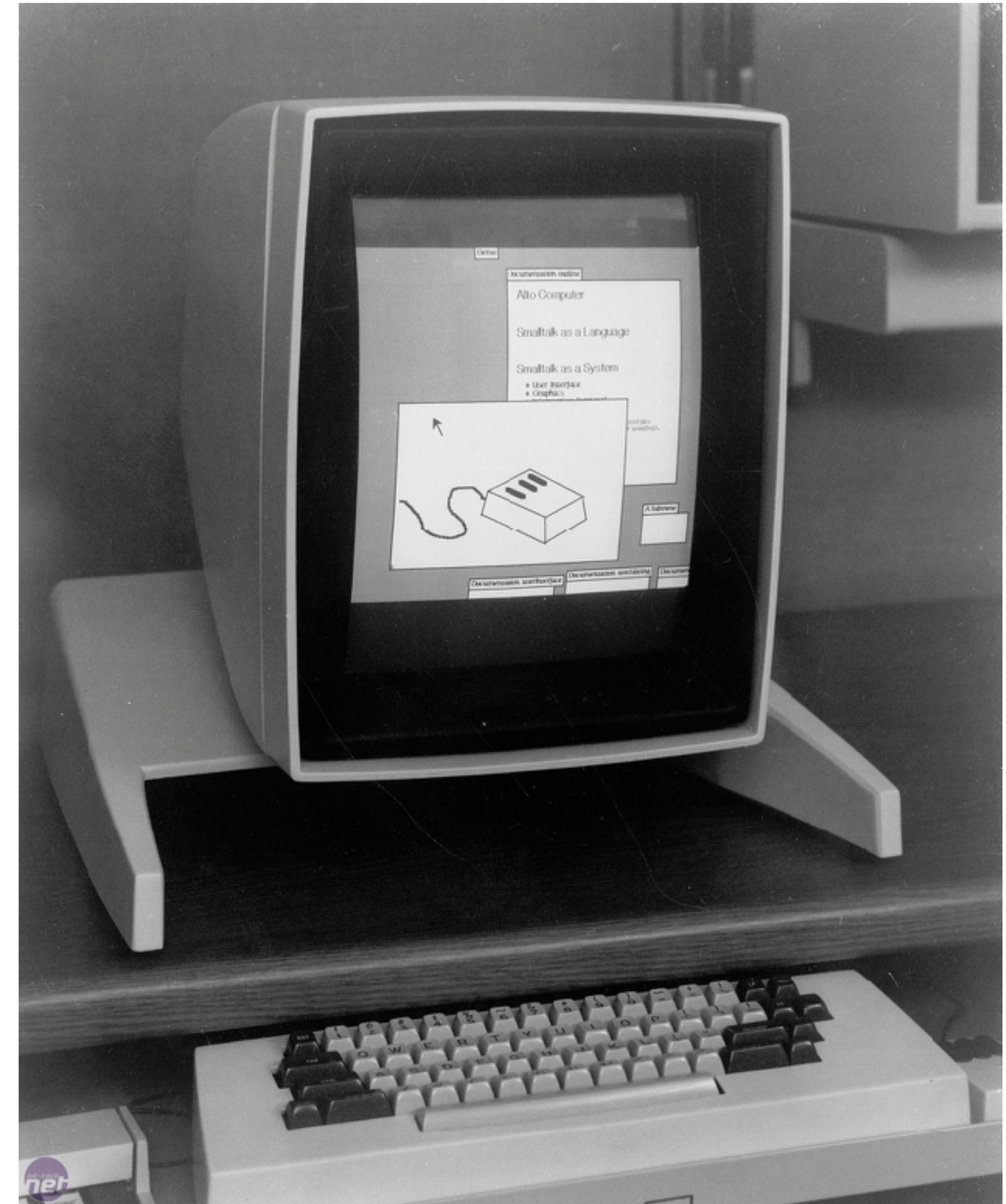
# Ivan Sutherland's "Sketchpad" (1963)



# Then...



William Fetter's Boeing Man  
Pilots a Jet (1964)



Xerox Alto (1973),  
the first PC with a GUI and a mouse

# ...and now



The Verge



Digital Trends

## **Puzzle:**

When Sutherland did his demo in the 1960s, black-and-white TVs were commonplace, and colour TVs were just becoming popular.

Why couldn't someone just connect a computer to a TV and show full images instead of just some lines?

Even a very low  $256 \times 240$  resolution  $\approx 60\text{k}$  pixels

Full colour with 3 bytes per pixel  $\approx 180\text{ kB}$  per image

At 25 frames per second  $\approx 36\text{ Mbps}$ !

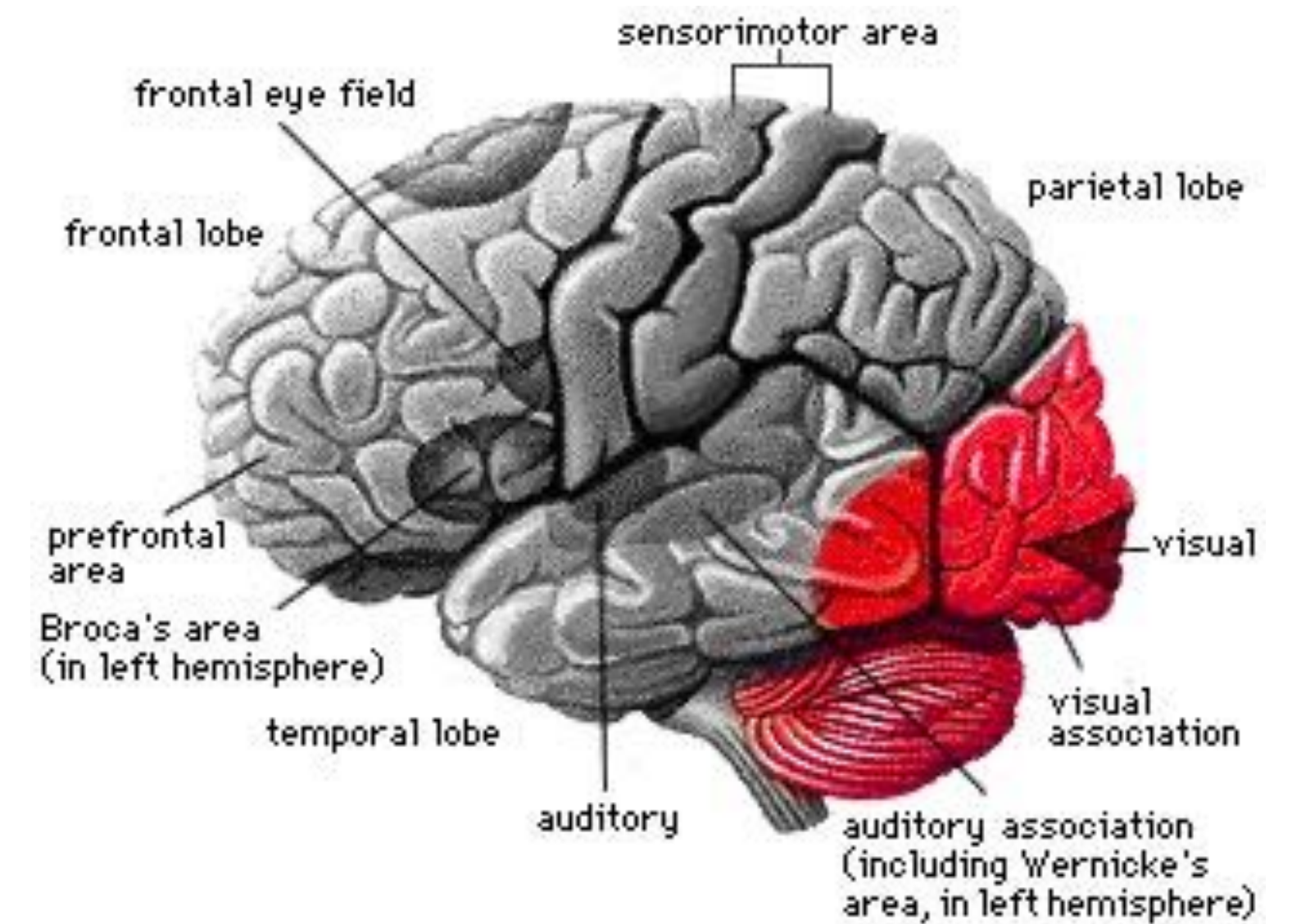
Graphics is a very high-bandwidth application!



Graphics is a very high-bandwidth application!

**In other words:** The eye is the highest-bandwidth input port to the brain.

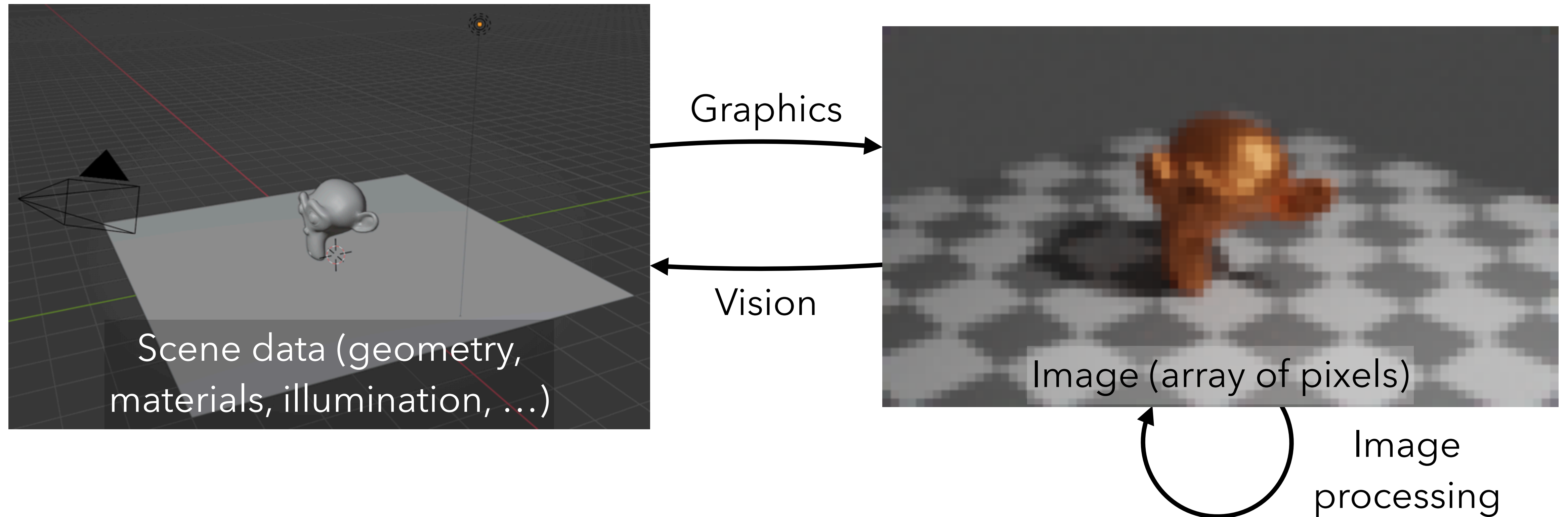
30% of the brain is dedicated to visual processing!





# Graphics vs. image processing vs. computer vision

Aren't all these fields about processing visual information?



# What this course is about

## **Scientific and mathematical foundations of graphics**

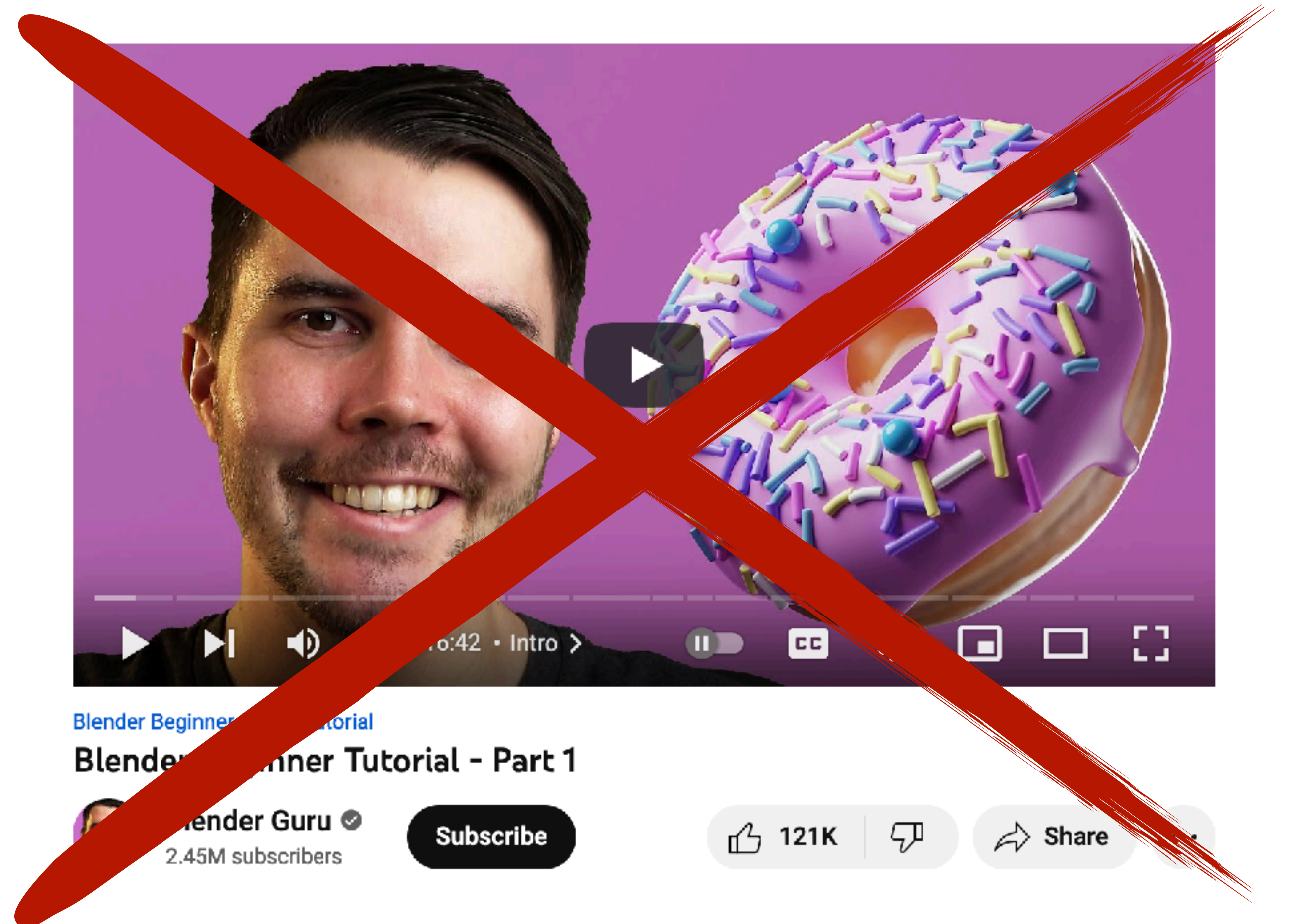
- Physics of light and colour, materials, dynamics for animation, ...
- Mathematics of curves and surfaces, perspective projection, sampling, ...

## **Representations, algorithms, and systems**

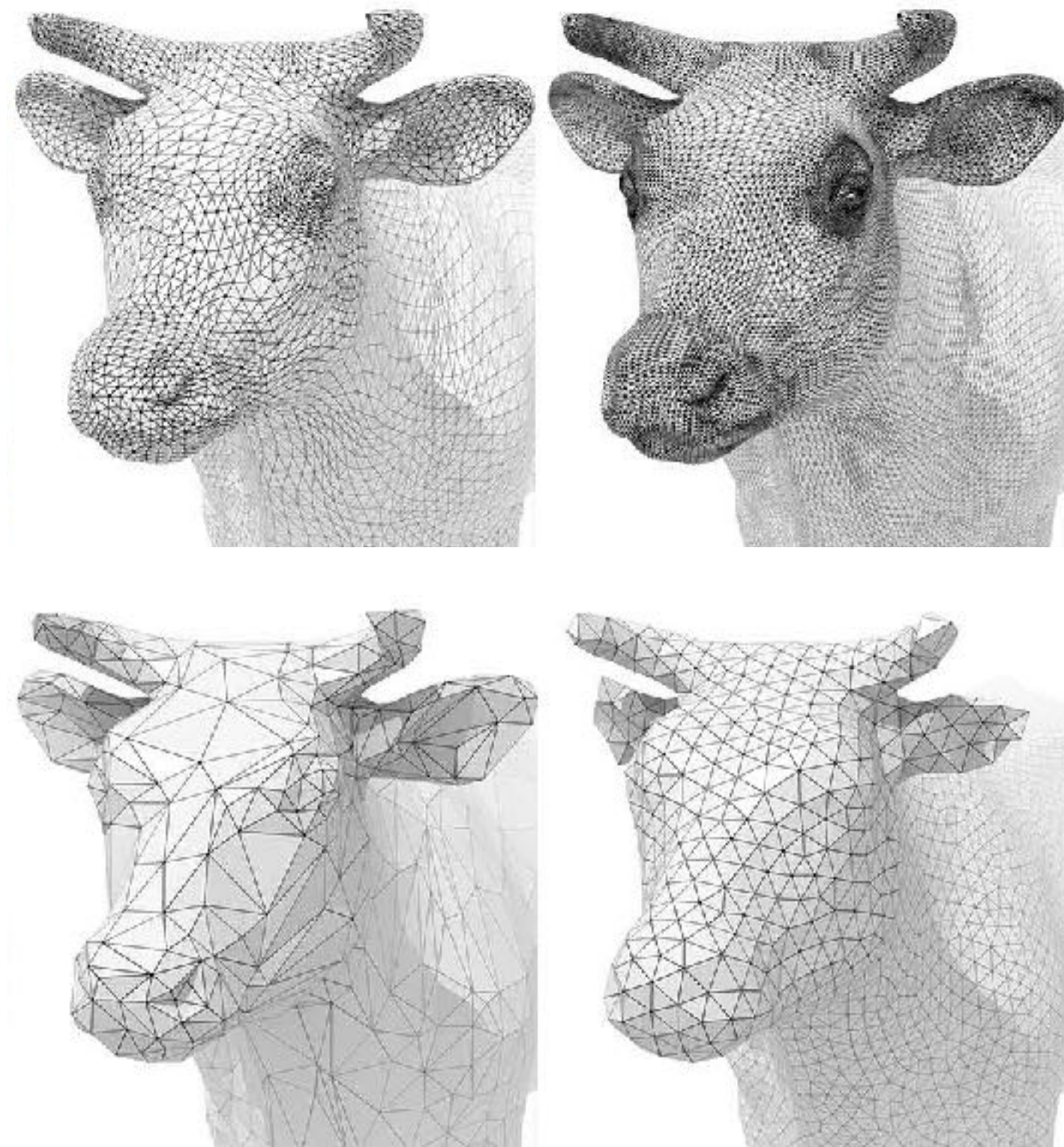
- Modelling geometry, images, transformations, ...
- Mesh subdivision, ray tracing, time integration, ...
- GPUs, hardware rendering pipeline, ...

# What this course is *not* about

- How to use 3D software
- How to be a graphics artist
- How to make a game



# Course content



Modelling



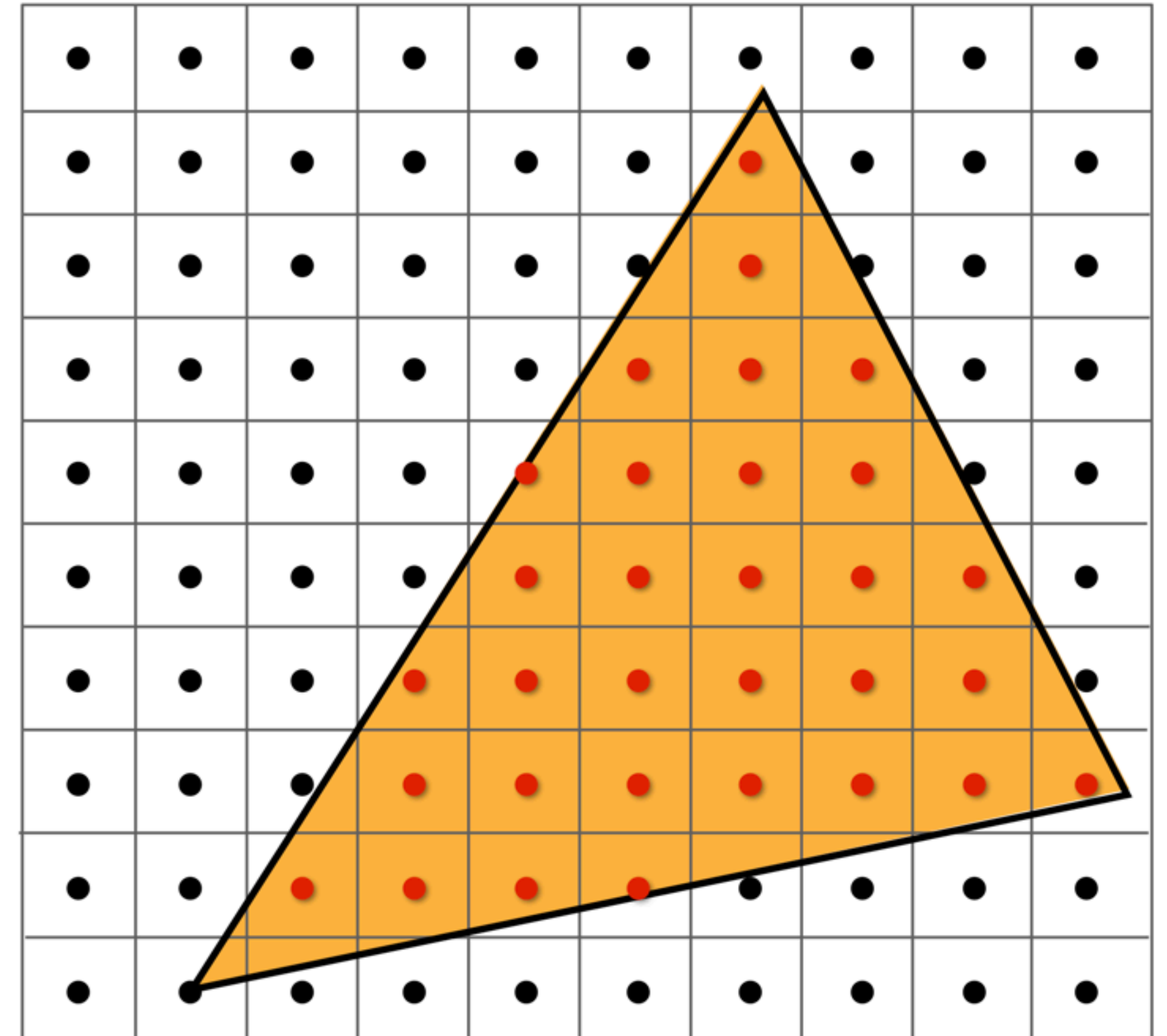
Rendering



Animation

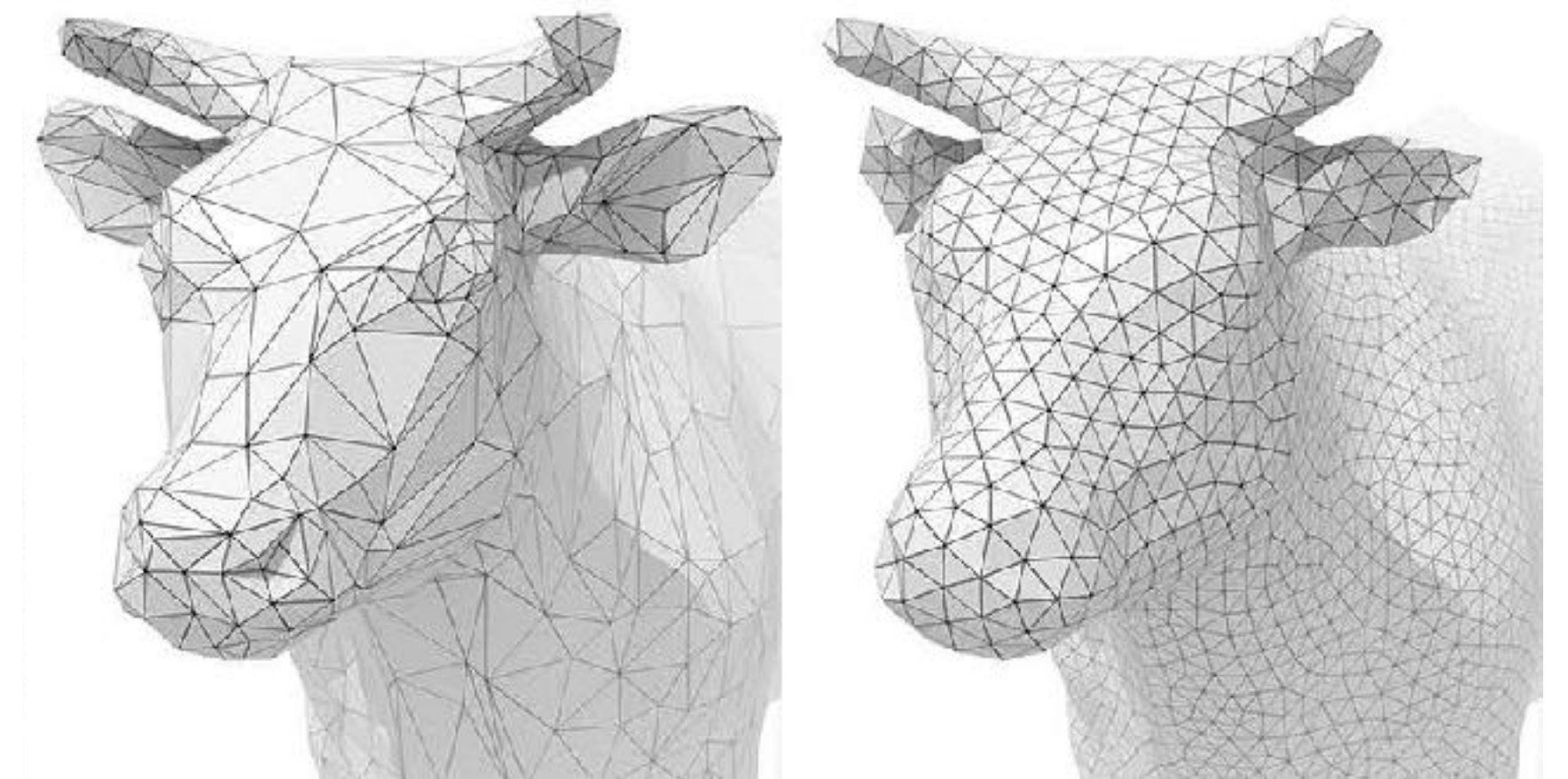
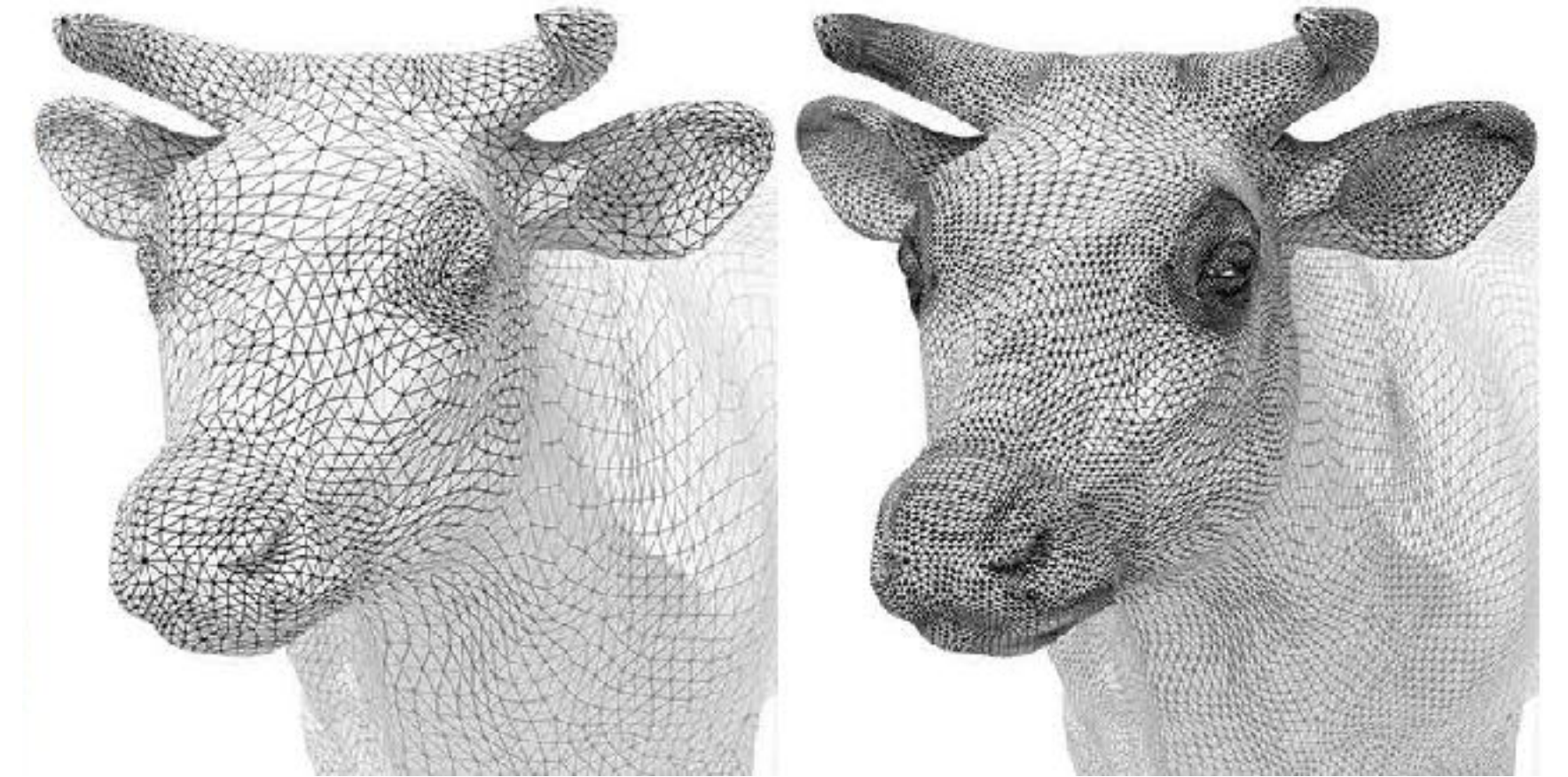
# Foundations

- Rasterizing shapes to images
- Sampling and aliasing
- Geometric transformations in 2D and 3D
- Perspective projection
- Visibility of surfaces
- Texture mapping



# Modelling

- Explicit and implicit representations
- Polygon meshes
- Bézier curves and surfaces
- Subdivision surfaces
- Mesh processing
- Spatial queries and acceleration structures



# Rendering

- Radiometry and colour
- Reflectance models
- The rendering equation
- Ray tracing and path tracing
- Variance reduction
- Real-time rendering



# Animation

- Hierarchical modelling
- Character animation and skinning
- Physics-based animation
- Time integration
- Collision handling
- PDEs for continuum physics





# Assignments

4 assignments:

- Foundations (rasterization)
- Modelling (mesh editing)
- Rendering (path tracing)
- Animation (mocap & simulation)

# Assignment policies

To be done in C++, some starter code provided

Groups of 1-2

Try to produce some **creative, artistic output** as well! I may show it off to the rest of the class 😊

**Late policy:** 4 free late days over whole semester, then 25% penalty per extra late day

# Assignment policies

**Academic honesty:** No copying of code or specific implementation details.

I cannot (and don't want to) prohibit you from discussing concepts, algorithms, and methods. But it must not be at such a detailed level that your code ends up being the same!

If two groups have "highly similar" code, **both** will receive the **exact same** penalty. It doesn't matter who copied from who.

- Submitted code **will** be checked for similarity with all submissions from this year **and from last year**

# Other logistics

**Course webpage:** <https://www.cse.iitd.ac.in/~narain/>, Ctrl+F 781 :)

**Office hours:** by appointment (please email me)

**Announcements:** on Moodle only

**Questions:** on Moodle Q&A forum only! Please **do not ask by email**

**Textbooks:** No required text, but the following are recommended:

- Hughes et al., *Computer Graphics: Principles and Practice*, 3rd Ed.
- Marschner and Shirley, *Fundamentals of Computer Graphics*, 4th or 5th Ed.

# Evaluation

- Assignments: 40% (4 × 10%)
- Exams:
  - Minor: 20%
  - Major: 35%
- Participation (in-class questions, Moodle Q&A): 5%

# Participation

I want the course to be as interactive as possible!

During the lecture:

- Ask questions at any time
- Post your answer to the in-class puzzle (bring a device to class!)

After each lecture, I will make a Q&A thread with the slides on Moodle

- Ask questions
- **Answer others' questions** if you know the answer
- Share links to other explanations of the topic