

Lecture 1:

Course Introduction + Review of Throughput HW Architecture

**Visual Computing Systems
Stanford CS348K, Spring 2025**

Hello from the course staff

Your instructor (me)



Prof. Kayvon

Your CAs



Purvi



Vishnu

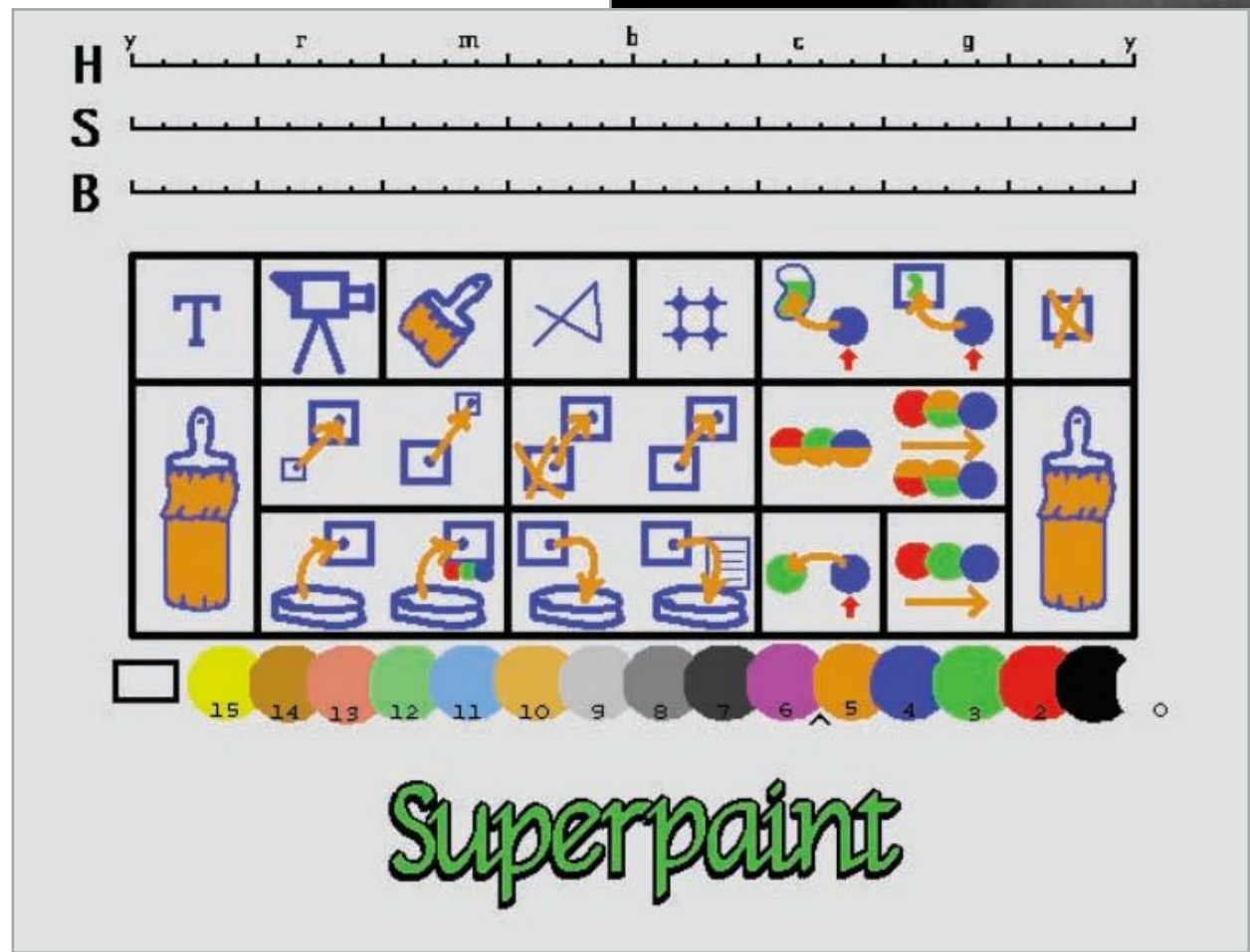
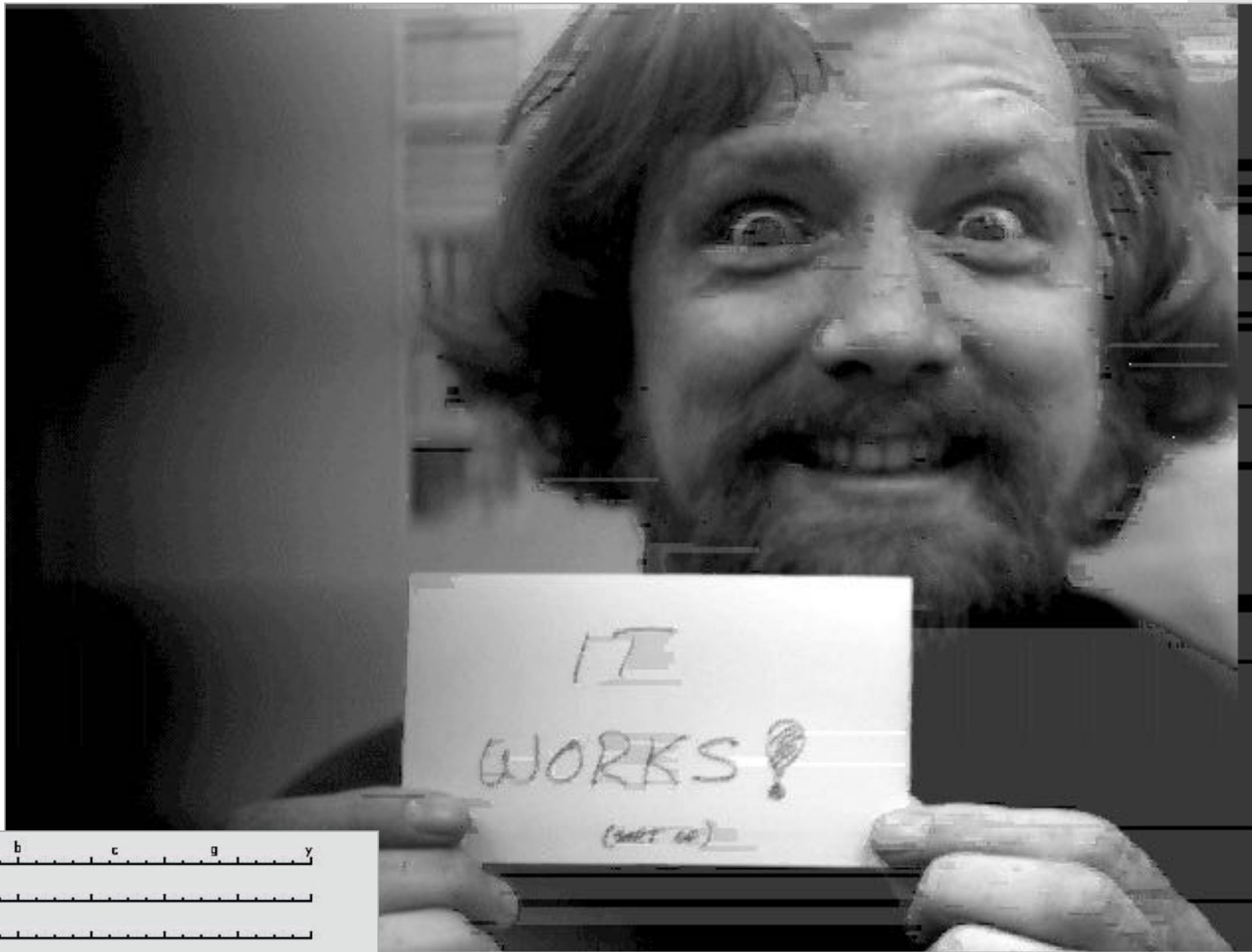
**Visual computing applications have always demanded
some of the world's most advanced computing systems**



Ivan Sutherland's Sketchpad on MIT TX-2 (1962)

The frame buffer

Shoup's SuperPaint (PARC 1972-73)

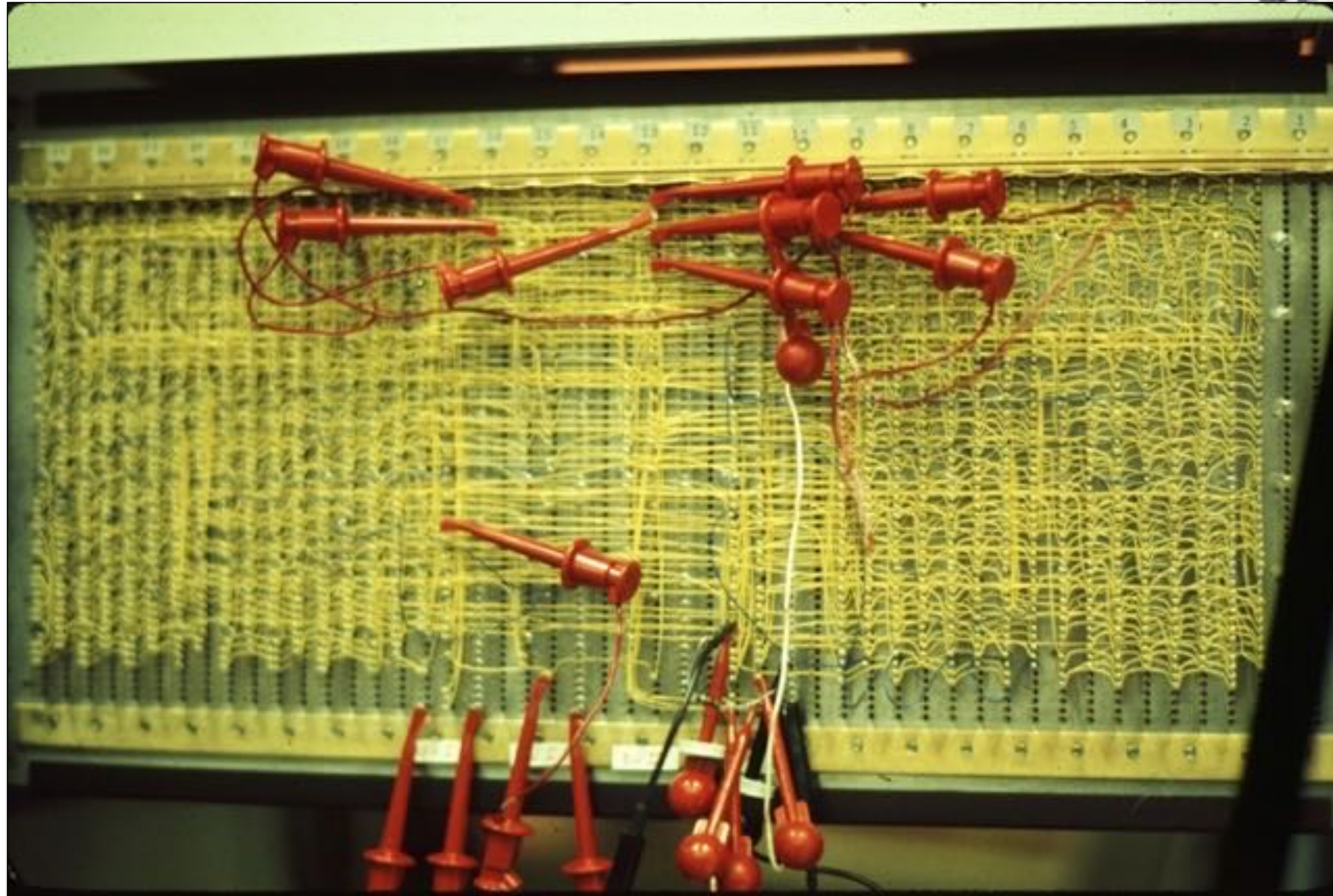


16 2K shift registers (640 x 486 x 8 bits)



The frame buffer

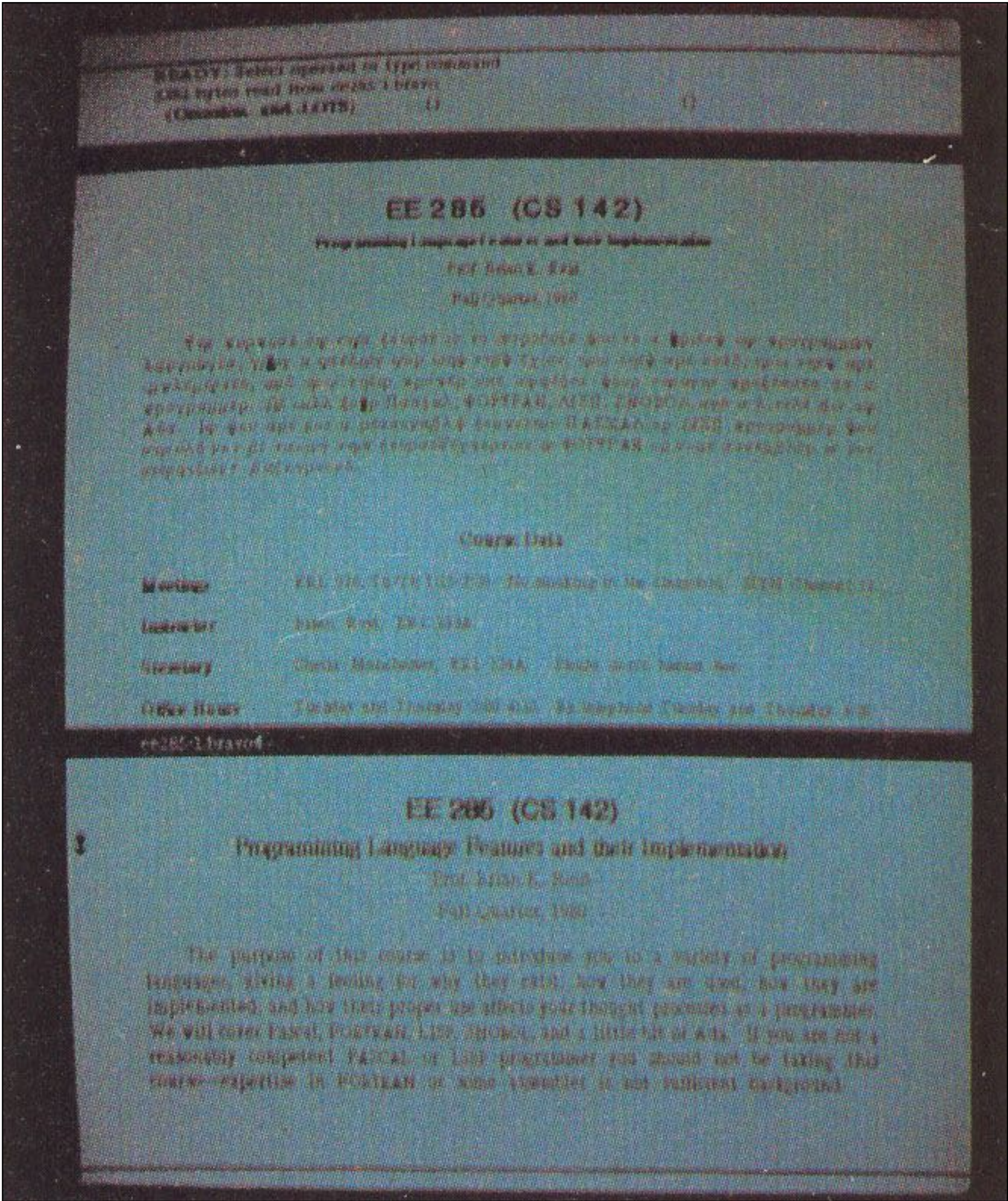
Shoup's SuperPaint (PARC 1972-73)



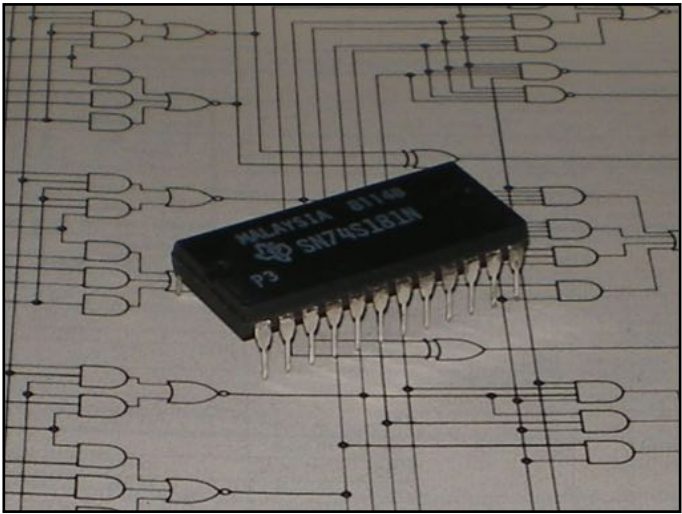
16 2K shift registers (640 x 486 x 8 bits)



Xerox Alto (1973)



Bravo (WYSIWYG)



TI 74181 ALU

Goal: render everything you've ever seen

“Road to Pt. Reyes”
LucasFilm (1983)



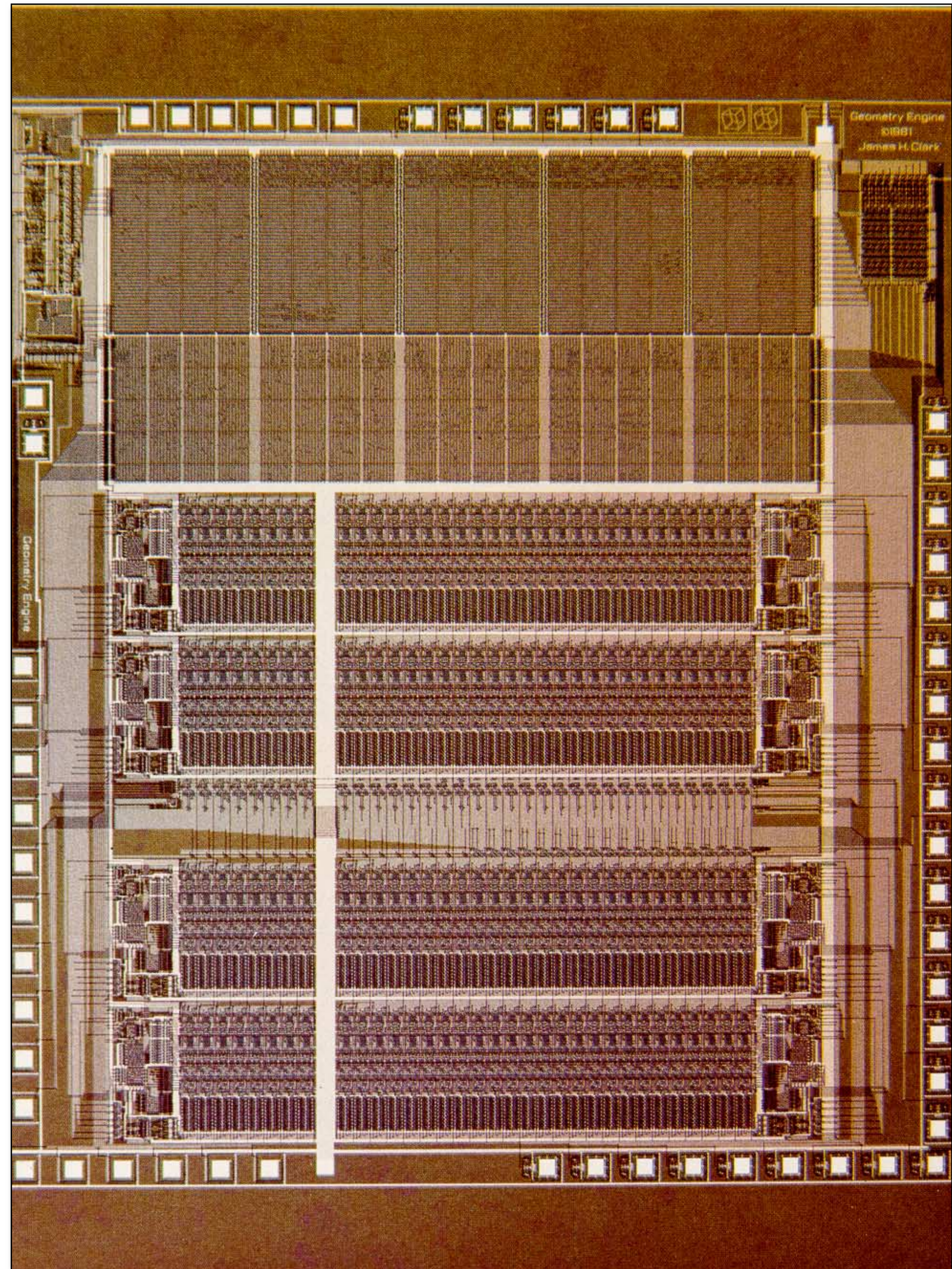
Pixar's Toy Story (1995)



“We take an average of three hours to draw a single frame on the fastest computer money can buy.”
- Steve Jobs

Clark's geometry engine (1982)

ASIC for geometric transforms
used in real-time graphics



NVIDIA Titan RTX 4090 GPU



~ 80 TFLOPs fp32 *

About the performance of the world's top supercomputer in 2004 **

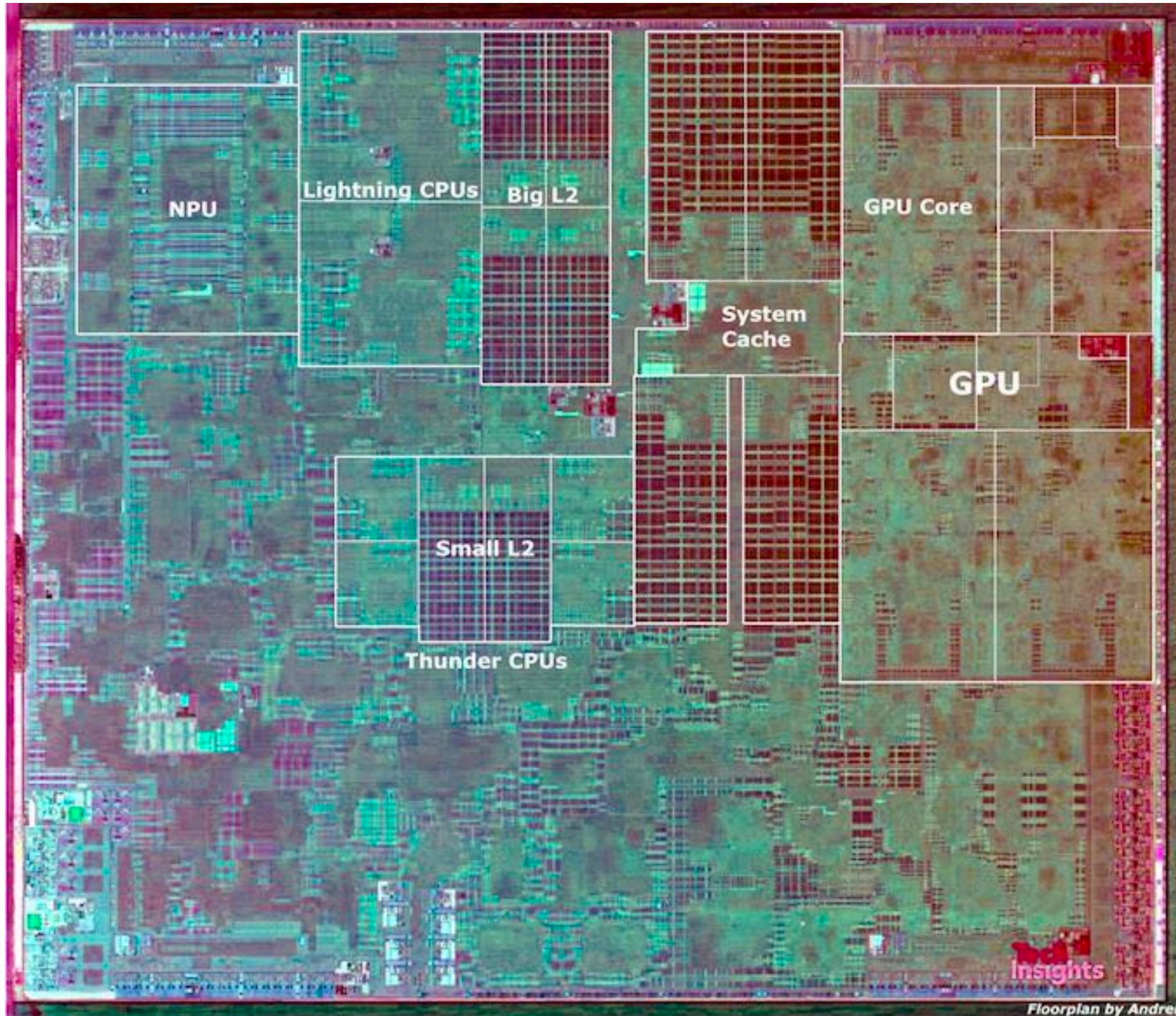
*** doesn't count texture filtering ops, ray tracing ops, and 1300 TFLOPS of DNN compute**

**** not apples-to-apples since BlueGene/L is double precision flops**





Modern smartphones utilize multiple processing units to quickly generate high-quality images

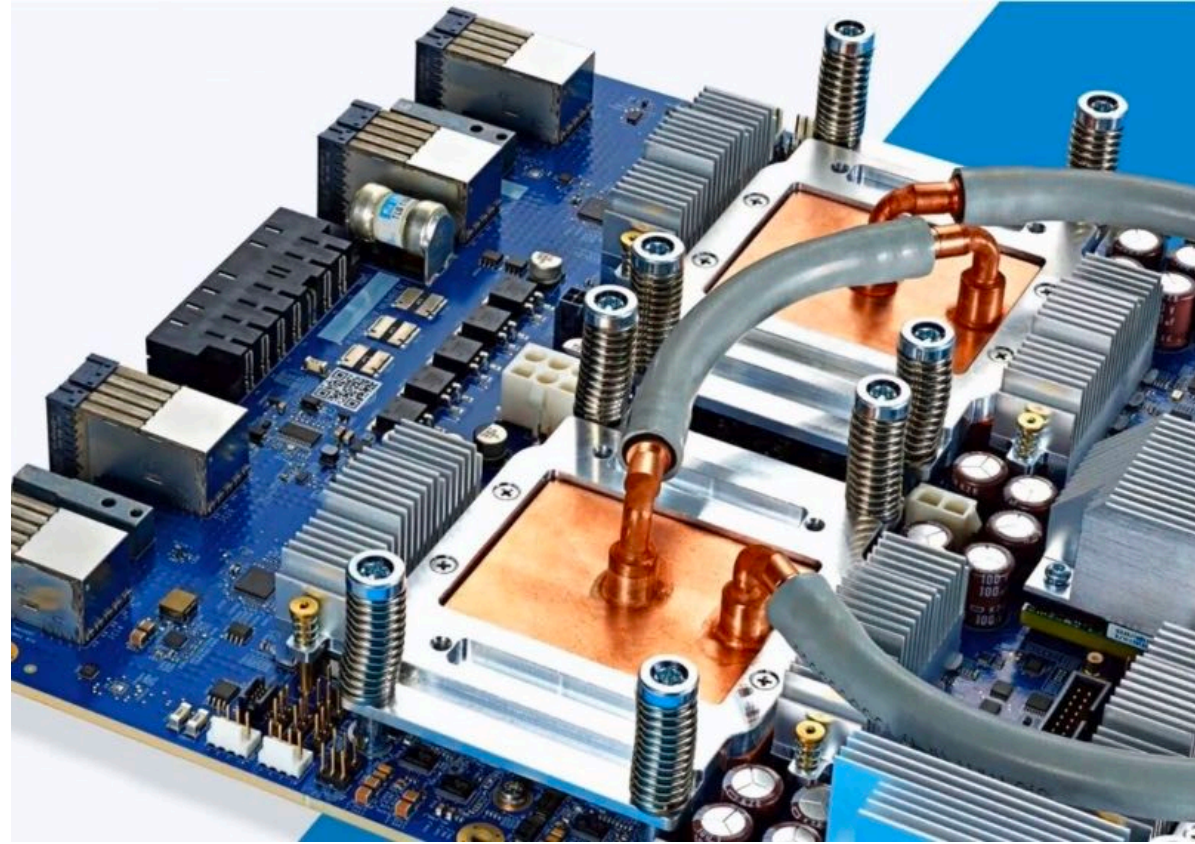


Apple A13 Bionic

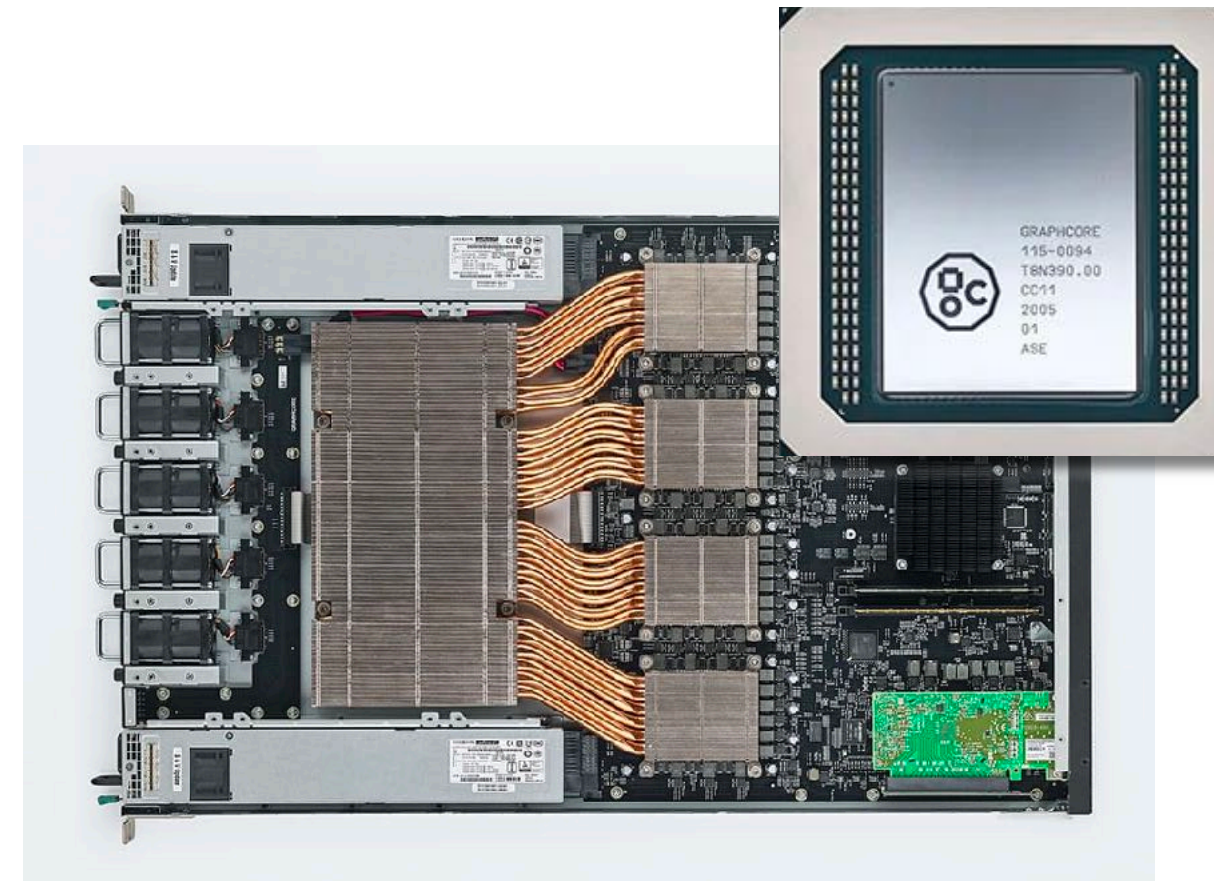


Multi-core CPU (heterogeneous cores)
Multi-core GPU
Neural accelerator
Sensor processing accelerator
Video compression/decompression HW
Etc...

Hardware acceleration of DNN inference/training



Google TPU3



GraphCore IPU



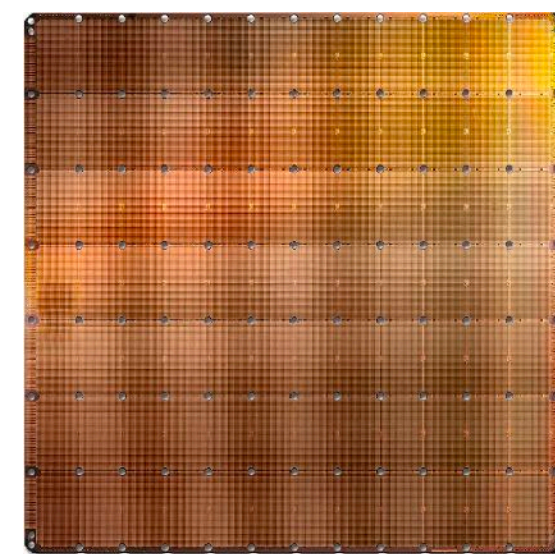
Apple Neural Engine



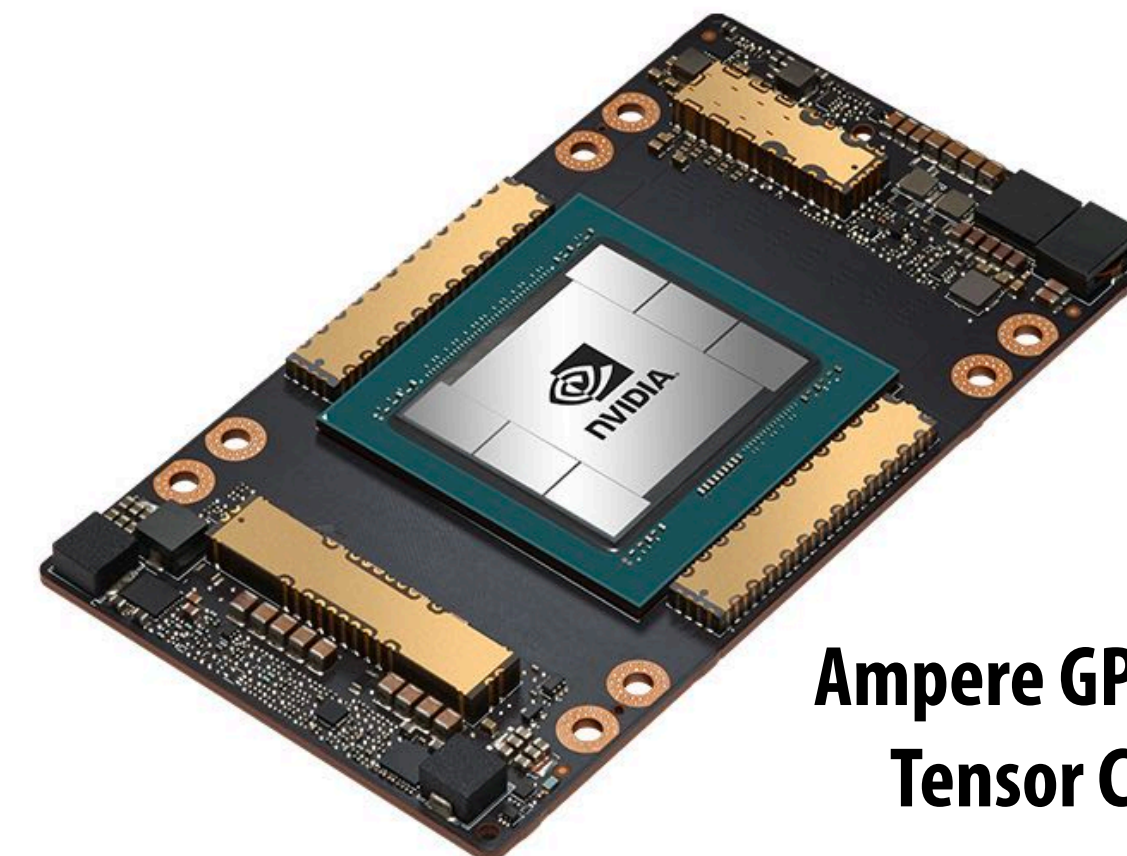
**Intel Deep Learning
Inference Accelerator**



**SambaNova
Cardinal SN10**



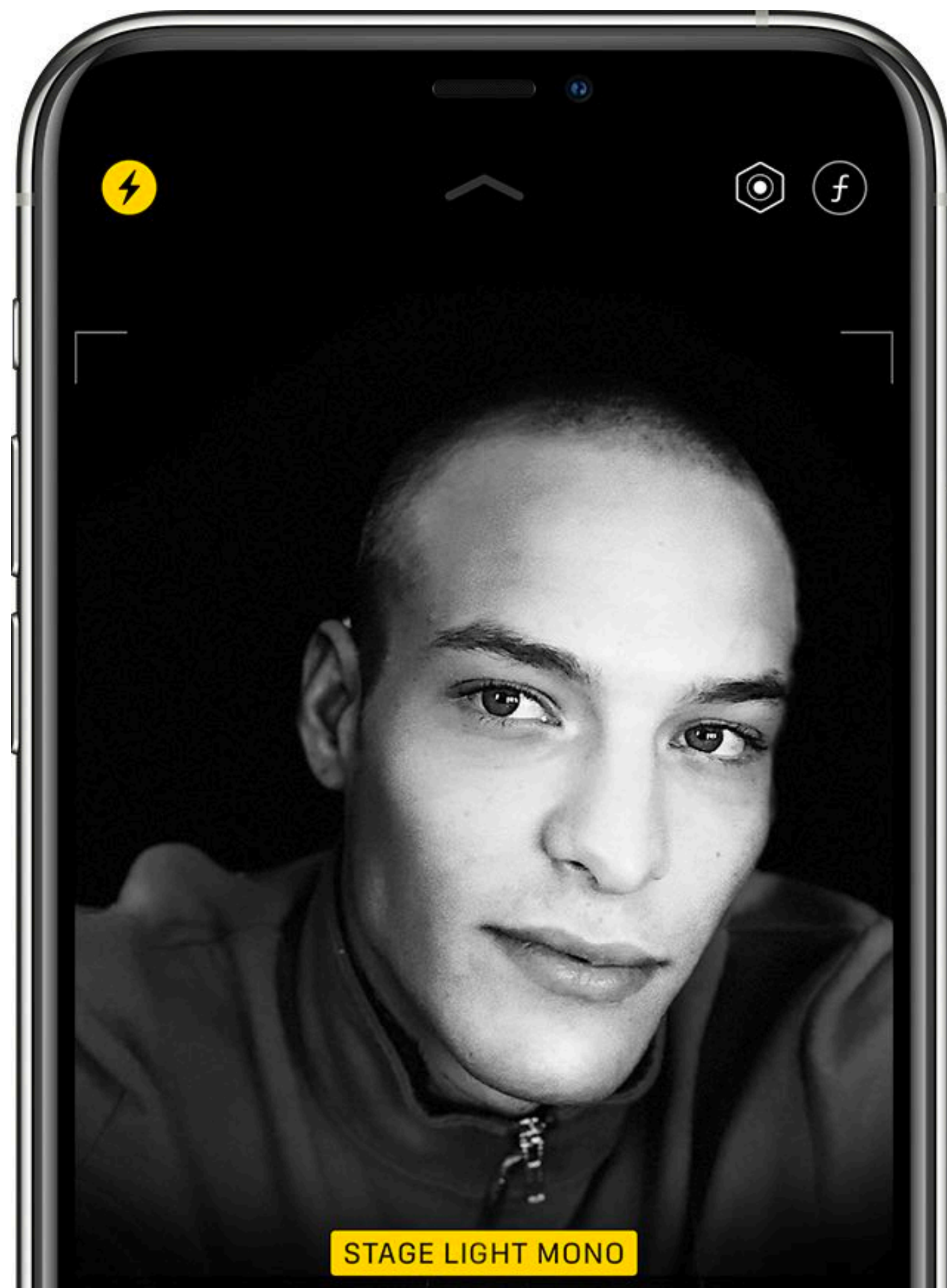
Cerebras Wafer Scale Engine



**Ampere GPU with
Tensor Cores**

Digital photography: major driver of compute capability of modern smartphones

Portrait mode
(simulate effects of large aperture DSLR lens)



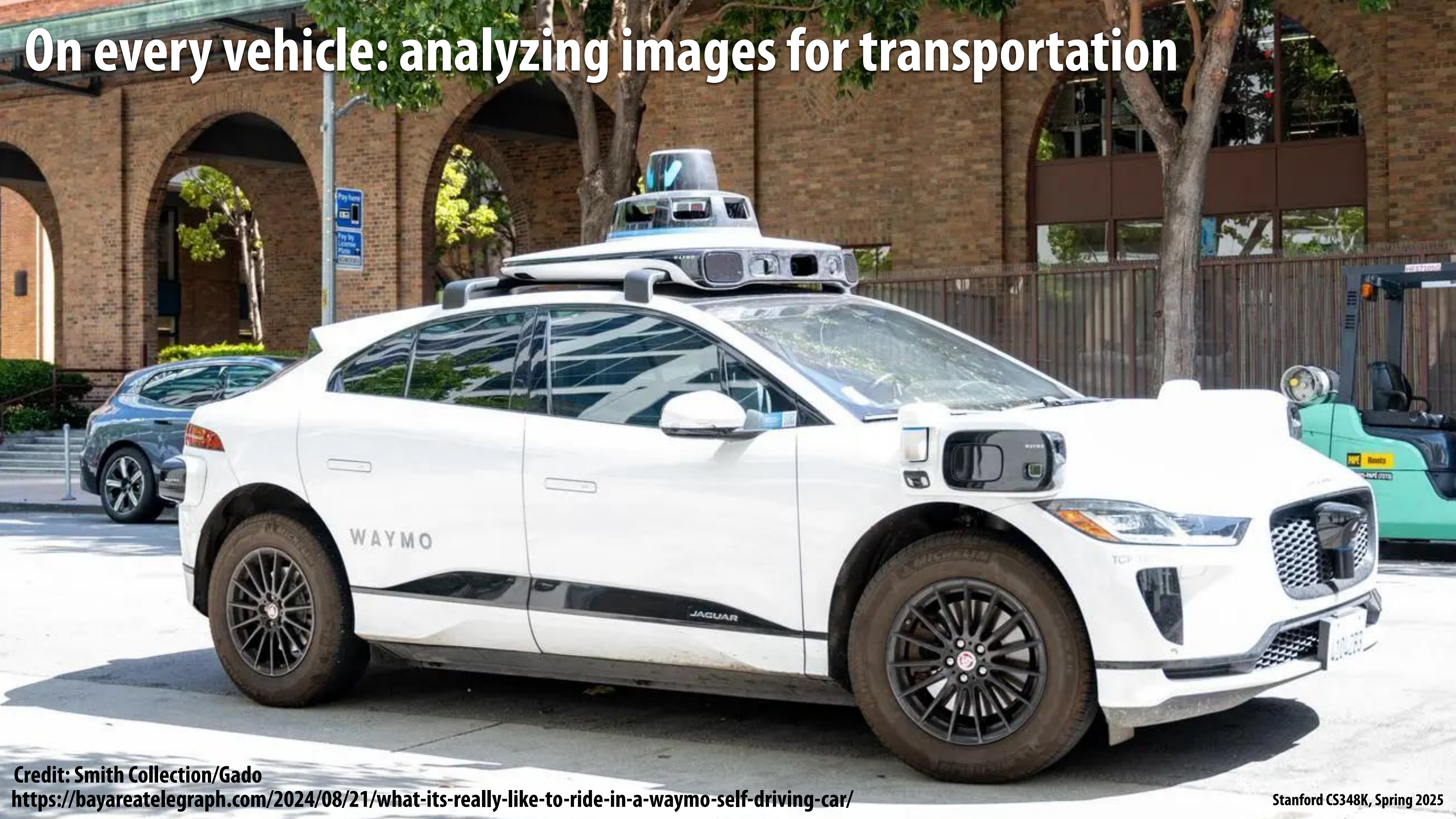
High dynamic range (HDR) photography



Apple Vision Pro (2024)

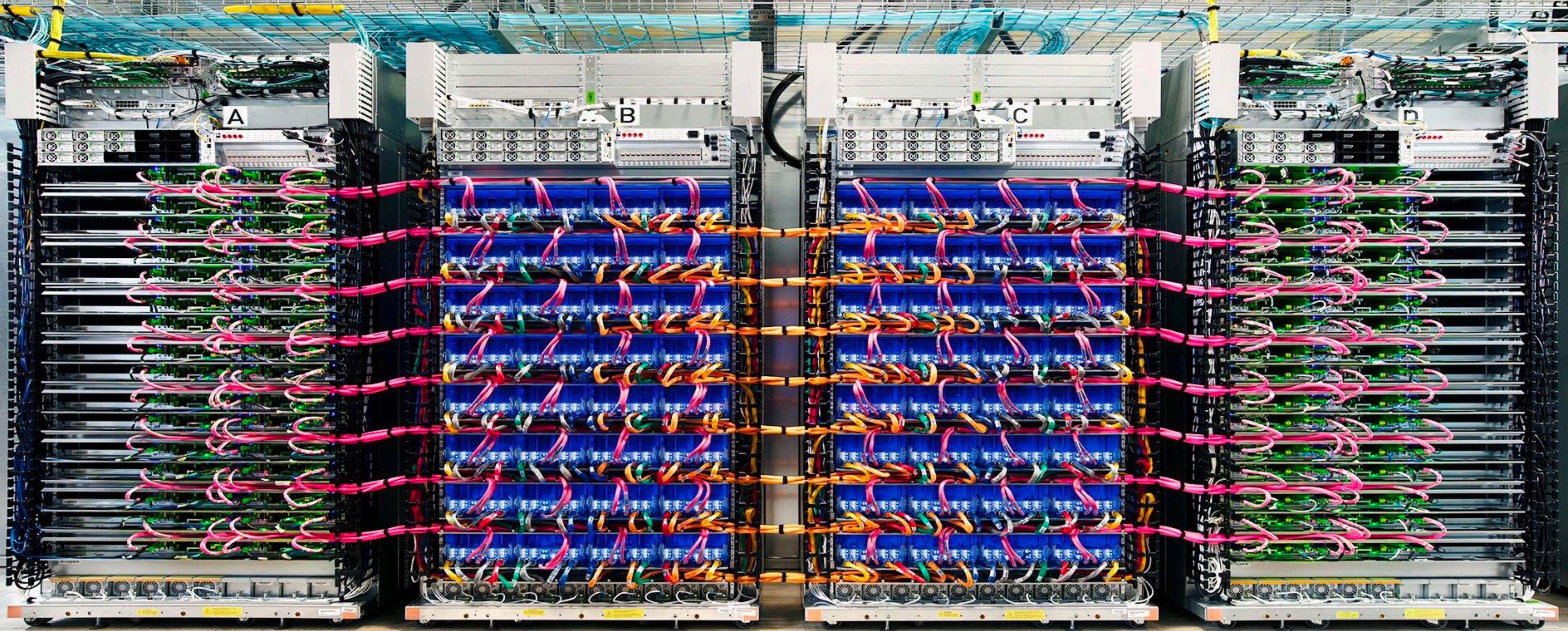
**~11.4M visible pixels per panel
(28 Mpixel display)**





On every vehicle: analyzing images for transportation

Datacenter-scale applications



Google TPU pods

Image Credit: TechInsights Inc.

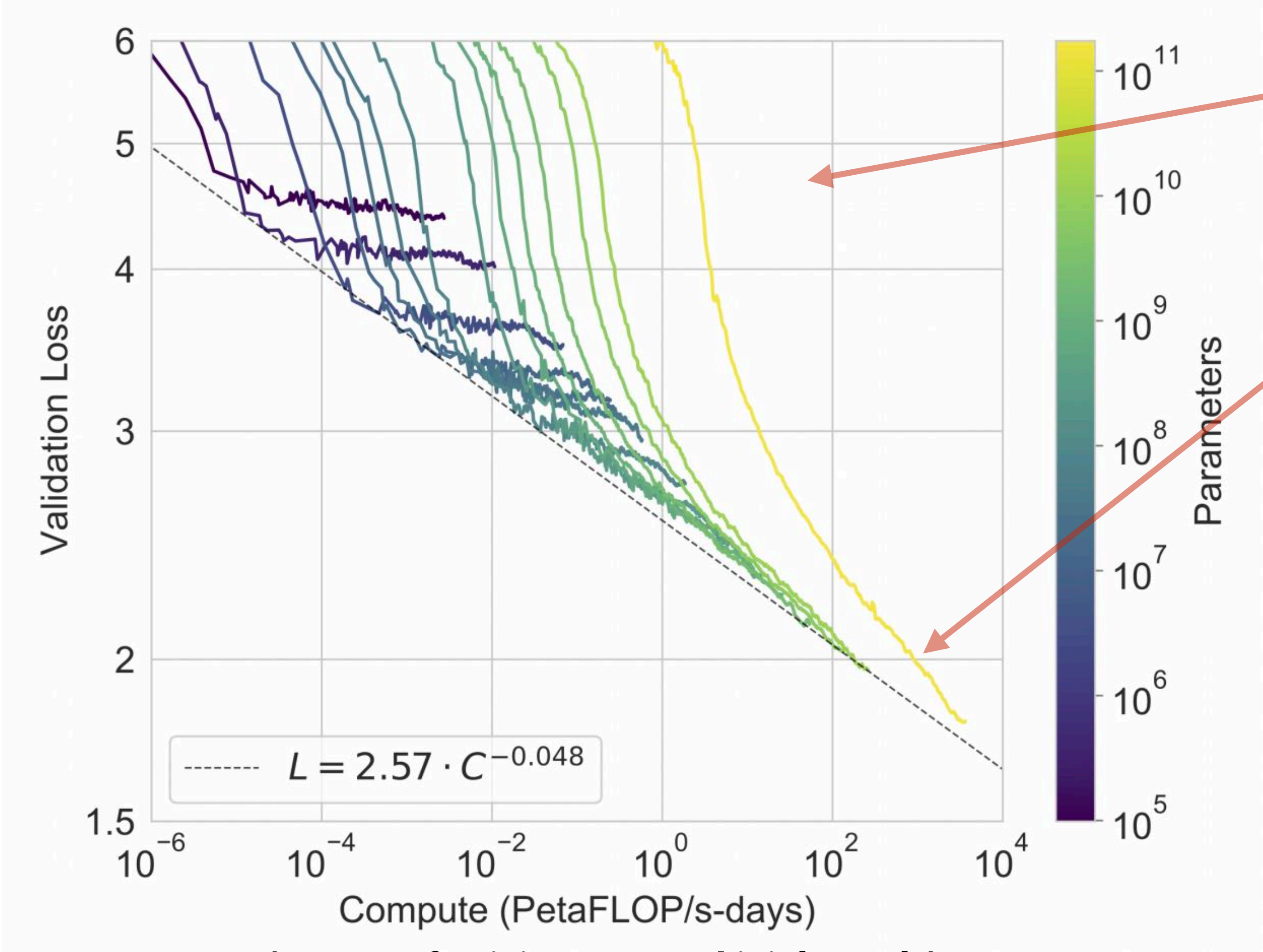
NVIDIA B200 DGX



36 PFLOPs (Tensor Core BF16 precision)
600 TFLOPS (FP32 math)

Scaling up (for training big models)

Example: GPT-3 language model



(Amount of training — note this is log scale)

Very big models +
More training
=
Better accuracy

Power law effect:
exponentially more compute to take
constant step in accuracy

Training image/video/text foundation models



Video generated by OpenAI's Sora.

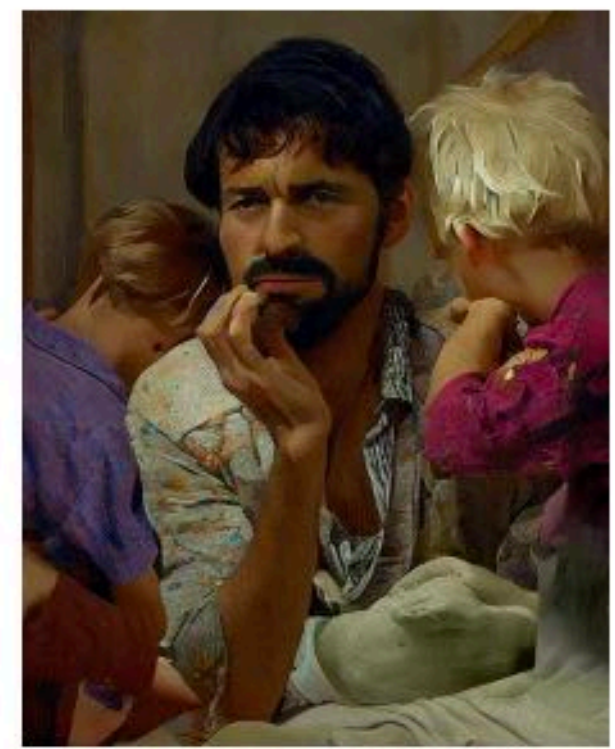
AI generated visual content

[ControlNet 2023]

Input (Canny Edge)



Default



Automatic Prompt

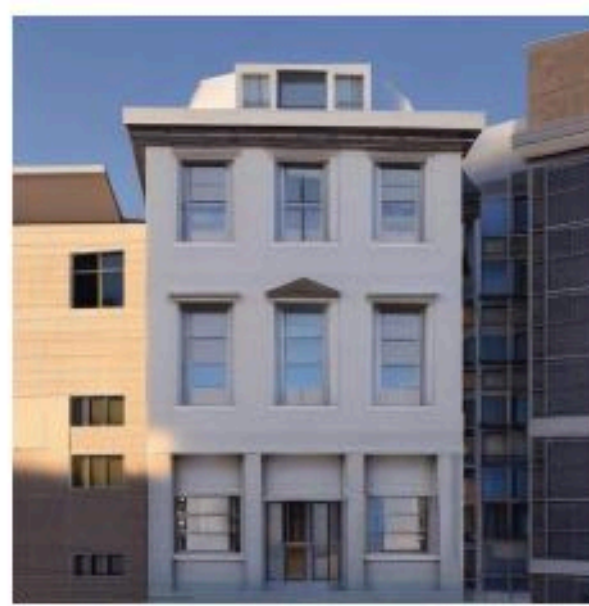


“a man with beard sitting with two children”

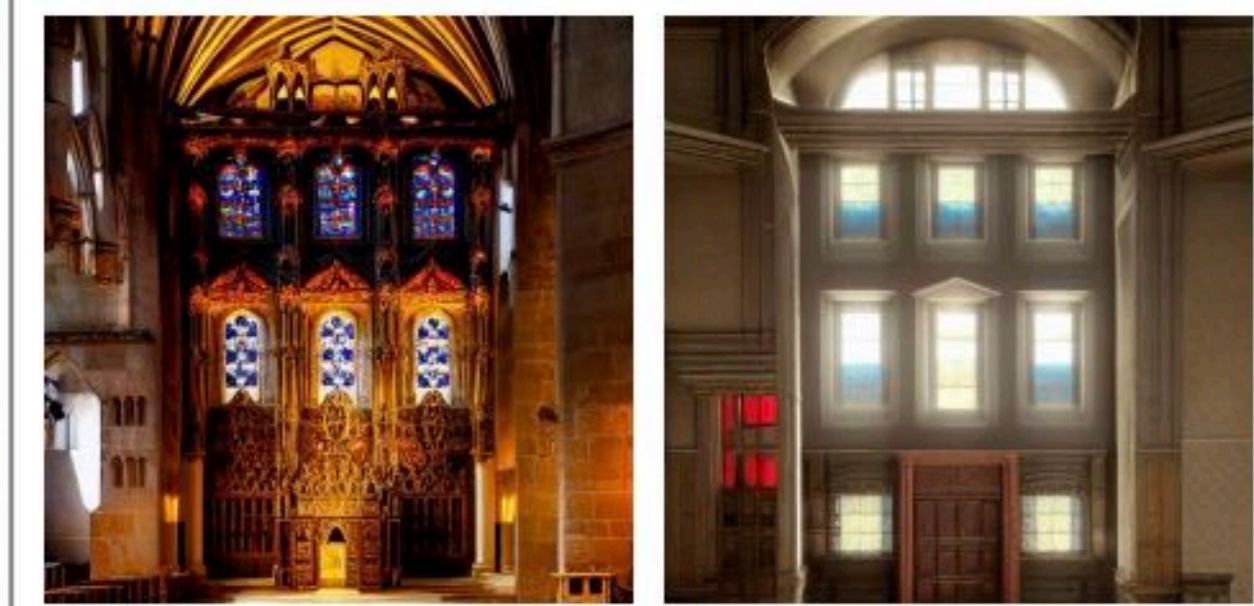
User Prompt



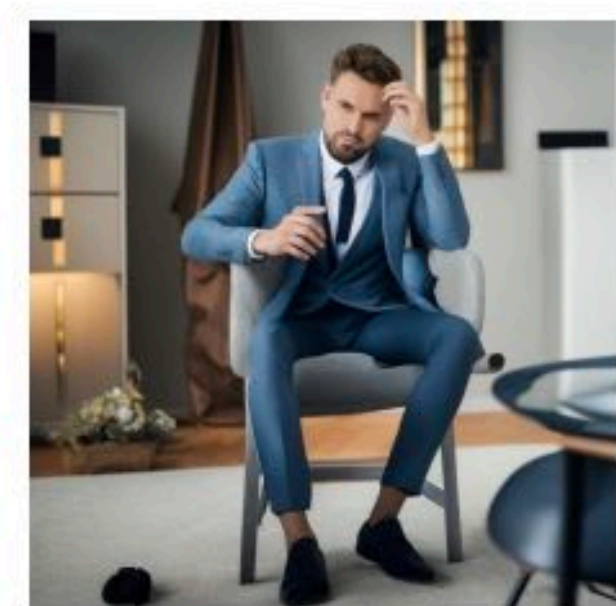
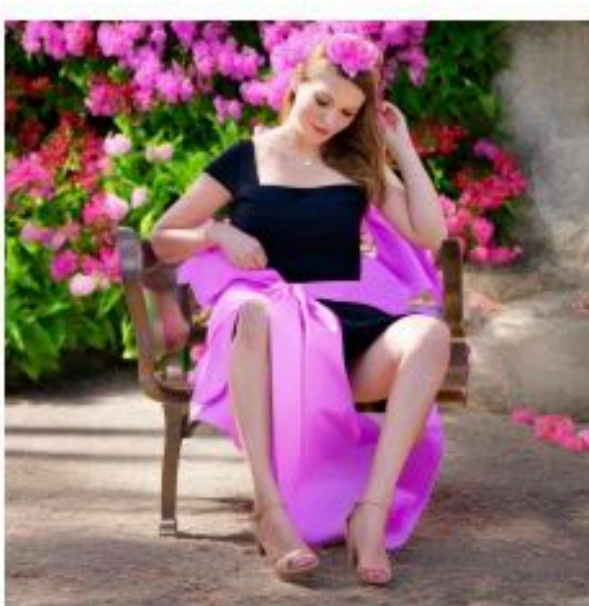
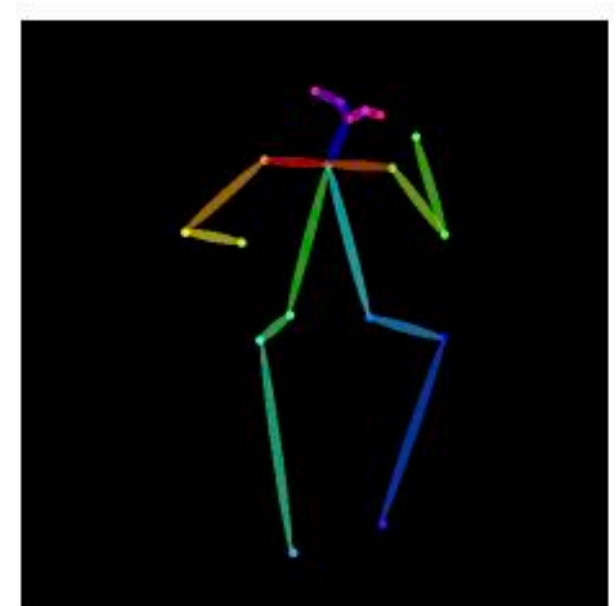
“mother and two boys in a room, masterpiece, artwork”



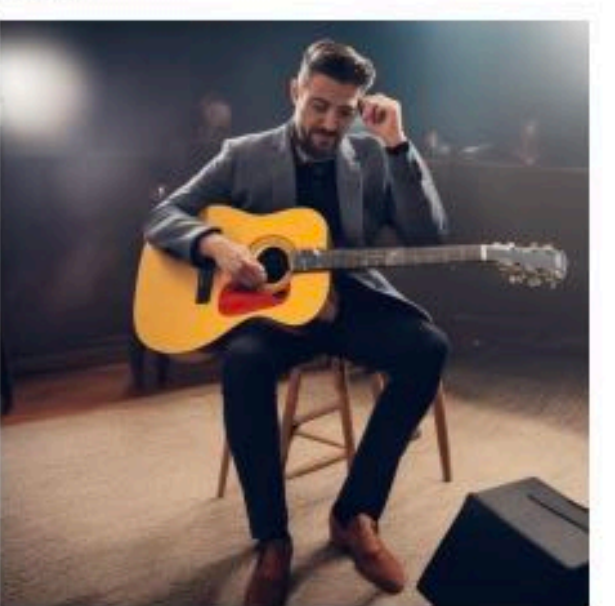
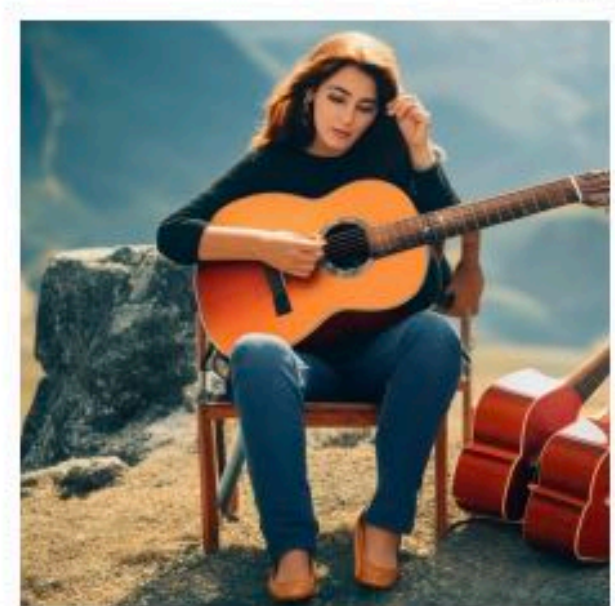
“a building in a city street”



“inside a gorgeous 19th century church”



astronaut



“music”

Youtube

Transcode, stream, analyze...



#LuisFonsi #Despacito #Imposible
Luis Fonsi - Despacito ft. Daddy Yankee

6,703,305,990 views • Jan 12, 2017

36M 4.4M SHARE SAVE ...



Google VPU transcoding HW

What is this course about?

Accelerator hardware architecture?

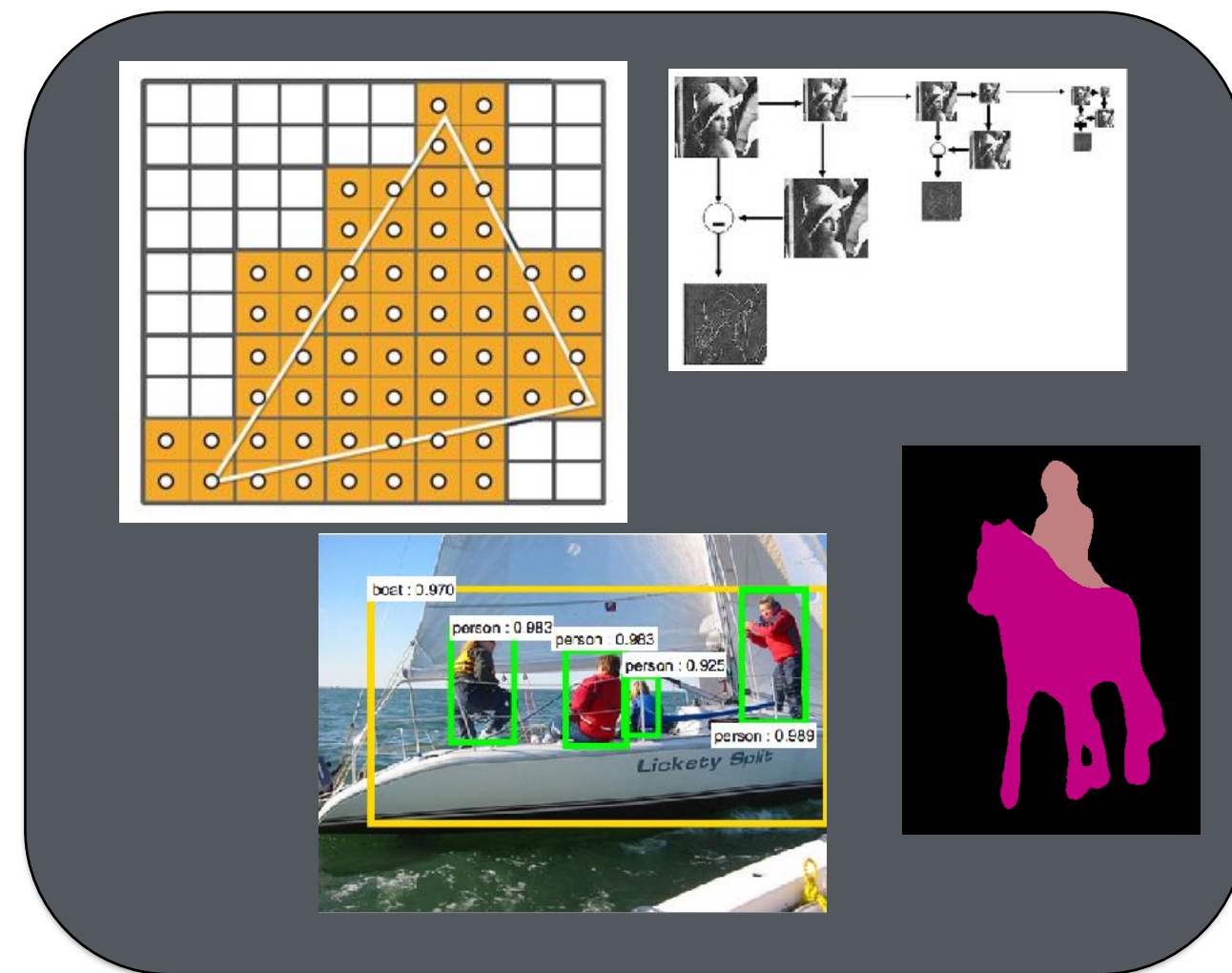
Graphics/vision/digital photography algorithms?

Programming systems?

What we will be learning about

Visual Computing Workloads

Algorithms for image/video processing,
DNN evaluation, generative AI, etc.



**If you don't understand key workload characteristics,
how can you design a “good” system?**

What we will be learning about

Modern Hardware Organization

High-throughput hardware designs
(parallel, heterogeneous, and specialized)
fundamental constraints like area and power

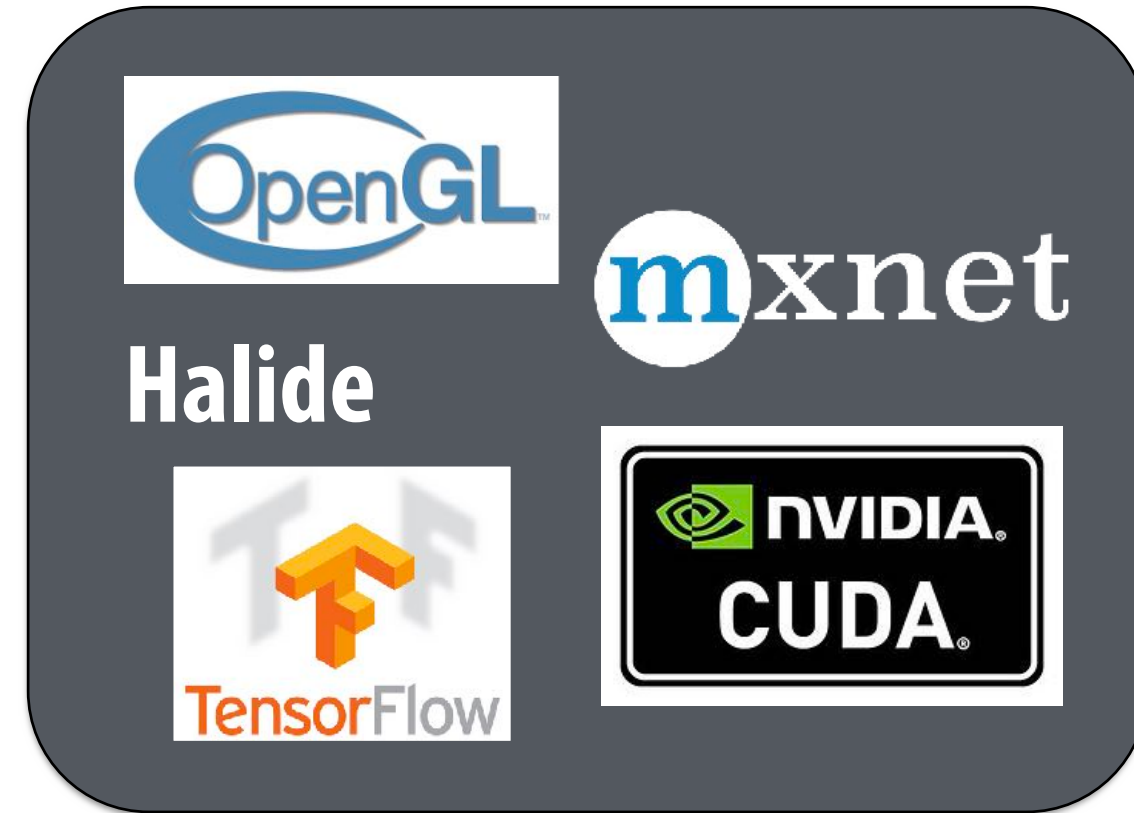


If you don't understand key constraints of modern hardware, how can you design algorithms that are well suited to run on it efficiently?

What we will be learning about

Programming Model Design

Choice of programming abstractions,
level of abstraction issues,
domain-specific vs. general purpose, etc.



Good programming abstractions enable productive development of applications, while also providing system implementors flexibility to explore highly efficient implementations

This course is about architecting efficient, scalable systems...

It is about the process of understanding the **fundamental structure of problems in the visual computing domain, and then leveraging that understanding to...**

To design more efficient and more robust algorithms

To build the most efficient hardware to run these algorithms

To design programming systems to make developing new applications simpler, more productive, and highly performant

2025 course topics

The digital camera photo processing pipeline in modern smartphones

What a modern smartphone camera does

Programming abstractions for scheduling image processing code onto parallel hardware

Making generative AI (for images, videos, animation, and more) usable

The problem of controlling the output of these models

And how we obtain good training data for them

Use of differential rendering for 3D reconstruction/capture

The role of differentiable programming in systems like NeRF, Gaussian Splatting, etc..

The design of a modern differentiable programming language for the GPU

Developing AI Agents for 3D environments

Making LLM-based agents and computer game bots

Training agents in simulation, and the high-performance simulation systems needed to do this

How do we evaluate these agents?

Generative AI for interactive worlds (“world models”)

Will future world simulators be more like game engines? Or more like learned models?

**ALWAYS SUBJECT
TO CHANGE!**

Logistics and Expectations

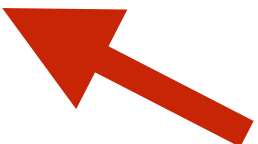
Logistics

- **Course web site:**

- **<http://cs348k.stanford.edu>**
- **My goal is to post lecture slides the night before class**

- **All announcements will go out via Ed Discussion**

My expectations of you

- **50% participation (25% in class + 25% written responses)**
 - There will be ~1 assigned technical paper reading per class
 - You will submit a response to each reading by 9am on class days via Gradescope
 - We will do an impromptu in class question during most class days as well
 - We will start most classes with a 30-45 minute discussion of the reading 
- **50% self-selected term project**
 - I suggest you start thinking about projects now
 - Proposals will be due in week 4 (we can meet 1-on-1 now about ideas, even if they are covered in later lecture topics)
 - Teams of up to 3

Implications:

Attendance is required

Auditing is not permitted

Reading response template

Reminder: We will concatenate all responses and give everyone in the class a PDF of all responses. If you wish your answers to be anonymous to the class, please leave your name off your PDF.

Part 1: Top N ($N < 3$) takeaways from discussions in the last class. Note: this part of the response is unrelated to the current reading, but should pertain to the discussion of the prior reading in class (or just discussion in the class in general):

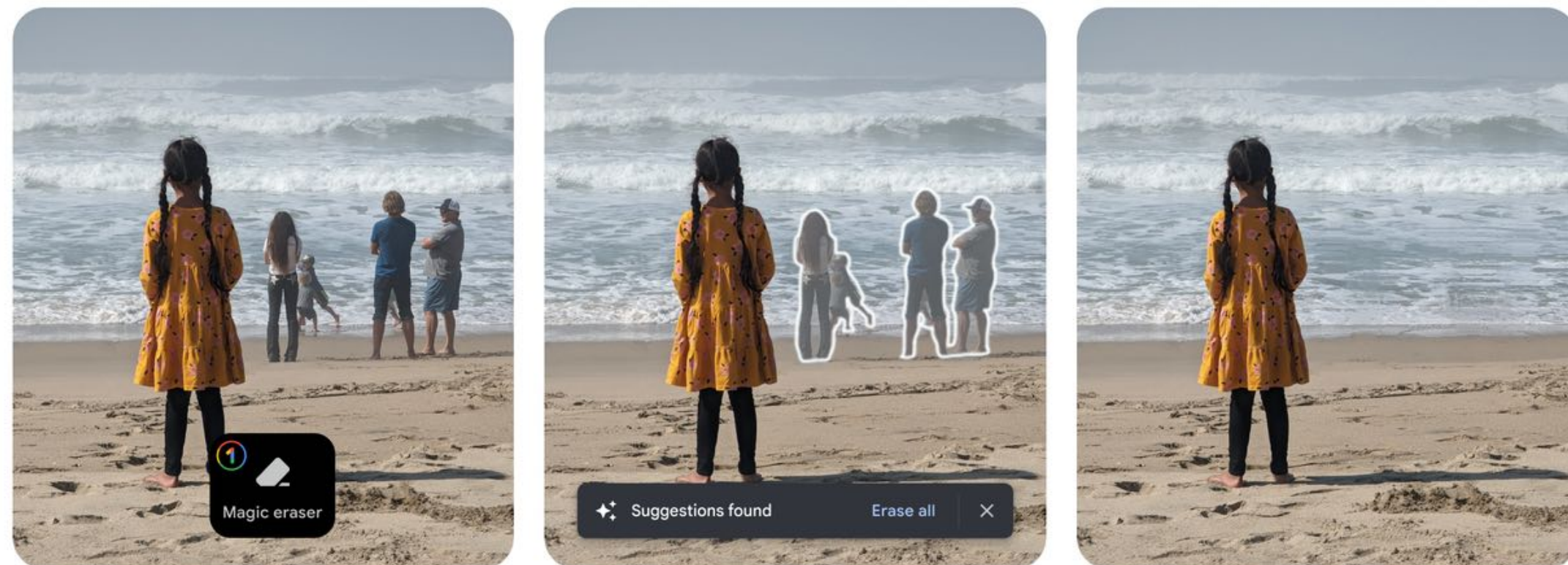
- What was the most surprising/interesting thing you learned?
- Is there anything you feel passionate about (agreed with, disagreed with?) that you want to react to?
- What was your big takeaway in general?

Part 2: Answers/reactions to instructor's specific prompts for this reading. (Please see course website for prompts).

Part 3: [Optional] Do you have unanswered questions you would like to have specifically addressed. (We also encourage you to just post these questions on Ed immediately so anyone can answer!)

Activity: let's design two systems

System 1: OpenAI is getting into the smartphone camera business. You were just hired as the lead architect.



Magic Eraser Feature

Systems architects begin with explicitly stating goals, non-goals, and assumptions

- **“Given these inputs, we wish to generate these outputs...”**
- **“We are working under the following constraints”**
 - **Example: the output images should have these properties**
 - **Example: the algorithm should have certain performance...**
 - **Should run in real time**
 - **Should be widely parallelizable, so it can run efficiently on a multi-core GPU**
 - **Example: the user experience must have these properties**
 - **Should not require user intervention to get “good” output**

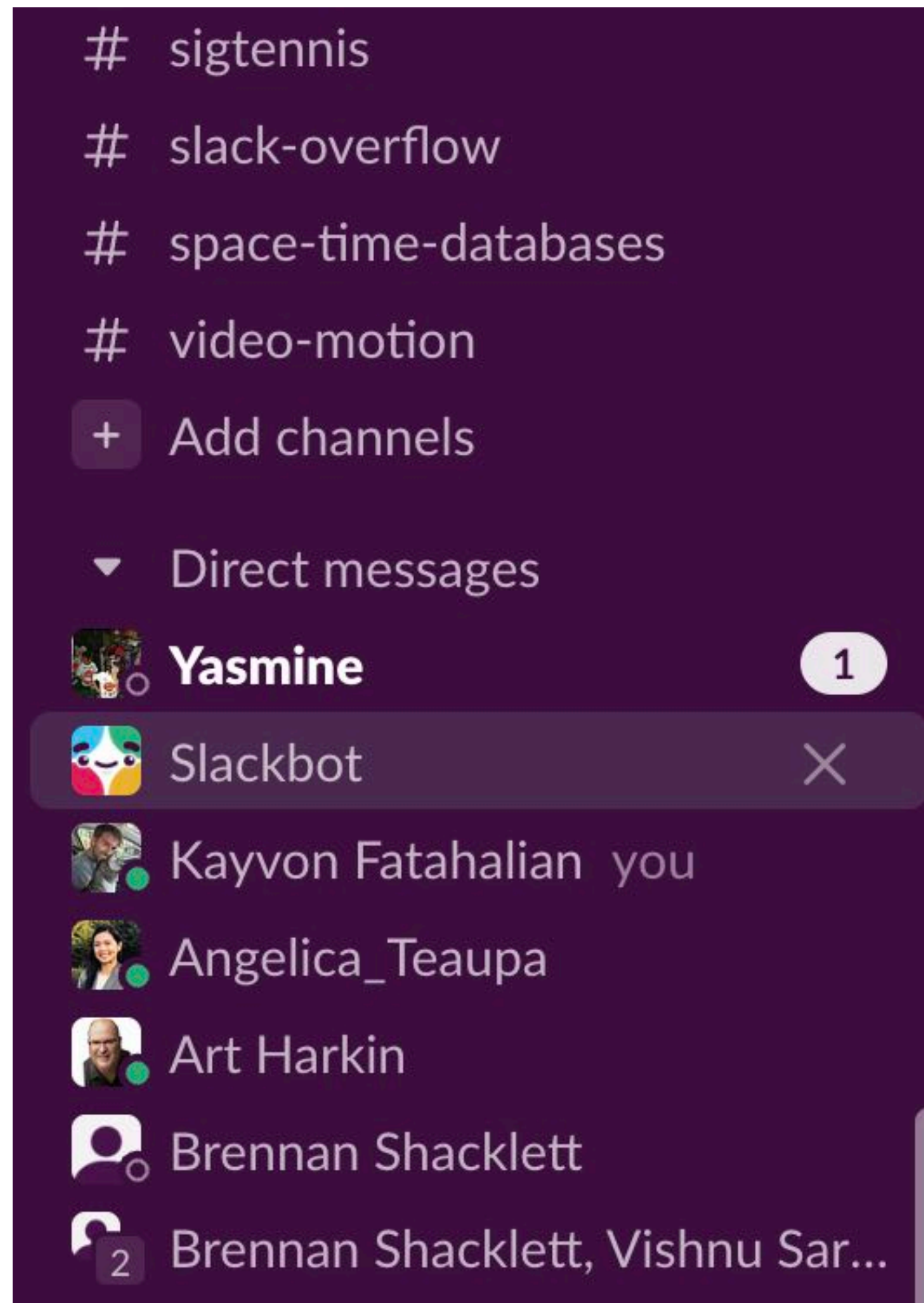
Why is establishing goals and constraints so important?

- **It defines the requirements of a "good" solution**
- **It provides a framework for assessing/evaluating the quality of the designer's decisions**
- **And for communicating with others: It provides context that leads to more generalizable knowledge. Since readers likely do not have the same goals and constraints as a paper's authors, understanding the author's goals and constraints helps readers understand which design decisions are applicable to their own problems**

Discussion

- **What are your image quality / feature list goals?**
- **What are your performance goals? Why?**
- **What are your user experience goals?**

System 2: Kayvon wants to have an office accessible to the world



Can we solve the case of a remote person interrupting me in my office for a quick conversation (in a socially acceptable way)?



EXIT

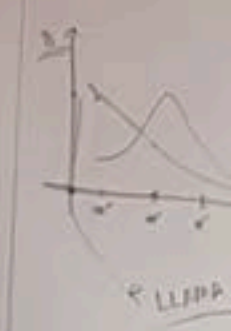
What's your UST Halloween costume?

CLIPPY

(you can make up a story, costume, what they are doing, etc.)

msb group dresses in mascot (brand, mascot, name, logo, etc.)

Promote!





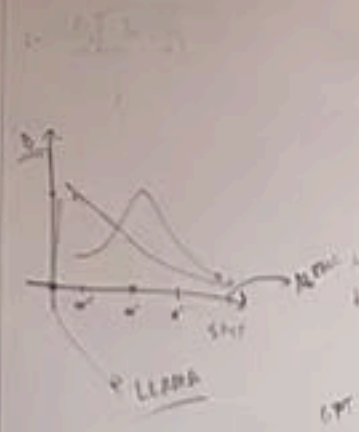


EXIT

What's your ULST halloween costume?

CLIPPY
(you can make this a group costume where you are all Clippy, etc.)

msb group dresses in michael (ground, stacy, jana) less shoes
Pizza Roll





EXIT

What's your ULST halloween costume?

Clippy

(if u can make this a scary costume, please do ahead, etc.)

msb group dresses / a michael (found a little, young) 1916 shoes
Penny Rd.

Systems architects begin with explicitly stating goals, non-goals, and assumptions

- **What are the goals of the system?**
- **What are non-goals?**
- **What are the key constraints?**

Tonight's reading

- **“What Makes a Graphics System Beautiful,” (2019), a blog post by me about thinking about goals and constraints.**
 - **The ideas in this post are how I want to you think about the systems we discuss in this course**
- **“Burst Photography for High Dynamic Range and Low-light Imaging on Mobile Cameras” (2016)**
 - **How a key feature in the Google Pixel phone camera works**
 - **Tonight read the front part of the paper for goals/constraints/assumptions.**
 - **We'll finish up the technical details of the paper after next lecture**

Welcome to CS348K!

- See website for tonight's reading