Session 1: Introduction to AI, Neural Networ Development Tools

Session 1: Introduction to A Networks, and Developme

- **Course Structure and Expectations:** Overview of the course syllabus, objectives, and asse
- **Development Environment Setup:** Installing Python, PyTorch, and essential libraries.
- **Introduction to Modern Coding AI Tools:** Overview of tools like Cursor to enhance coding
- **Introduction to Frontend and Backend Concepts:** Brief discussion to prepare for integrati
- **Fundamentals of AI and Neural Networks:** Introduction to core AI concepts, types of neura

Course Structure and Expectations

Introduction:

- You can find the Course Structure **here** (https://s3.us-east-1.amazonaws.com/theschoolof have 20 sessions covering AI fundamentals to advanced generative models, with focus on for LLM training.
- Learning Objectives:
  - Understand core AI concepts and neural network architectures
  - Gain practical skills in implementing AI models
  - Explore cutting-edge topics in generative AI and large language models.
- Course Philosophy:
  - Hands-on approach: learn by doing
  - Focus on both theoretical foundations and practical applications
  - Staying current with the rapidly evolving AI landscape
- What to Expect:
  - Weekly sessions combining lectures, demonstrations, and hands-on exercises
  - Gradual progression from basics to advanced topics
  - Culmination in a capstone project showcasing acquired skills
- Setting Expectations:
  - The learning curve can be steep, but persistence pays off
  - Importance of practice and experimentation outside of class
  - Collaborative learning: engage with peers and instructors

Development Environment Setup

Introduction:

- You need to have a laptop (with admin access) to work on this course. You can survive only frustrating as you go deeper into the course due to the different kinds of files and persistent
- We recommend you use Visual Studio or **Cursor** as your development **IDE**, but you're free
- Please make sure that you know what different Python Environments are, and ensure that y and other essential libraries (which will be introduced as the course progresses)
- You need to create a **Google Colab** account, which you'll use to train most of the earlier ne
- You also need to create an **AWS account**, which we will use later to train deeper neural ne for the assignments that need it. Please make your AWS account today if you don't have or
- You also need to create a **HuggingFace account**, which you'll use to host your application

Introduction to Modern Coding AI Tools (aka Vibe Codi

Introduction:

- **Let's start with a hands-on example**! Let's build a simple Chrome Plugin.
- **Efficiency and Speed**: Tools like Cursor.ai and Claude.dev can generate, debug, and optir time.
  - rather too quickly, hence we'd need to be really good at Git!
- **Enhanced Collaboration**: These tools act as virtual coding partners, providing suggestions logic in real-time
- **Adaptability Across Languages**: They support multiple programming languages, helping seamlessly.
  - making this course possible!
- **Contextual Understanding**: These AI tools offer deep code context analysis, enhancing th they produce
- **Error Reduction**: Automated debugging and testing features reduce human errors and imp
- **Accessibility for All Skill Levels**: Both beginners and experts can leverage AI assistance fluency.
- **Vibe-Coding**: This is an AI-related code; we will NOT be vibe-coding AI. Everything else wi learn more as we proceed.

Course Structure and Expectations

In this course, we will dive into creating complete applications that integrate both frontend and
aims to give you a solid understanding of these two core concepts and how we will utilize mod
applications.

Frontend: The User Interface (UI)

The front end of an application refers to everything the user interacts with directly—essentially
includes the layout, buttons, menus, input fields, and any other elements that users can see ar
explore:

- **HTML, CSS, JavaScript**: The foundational web technologies for structuring (HTML), styling
  (JavaScript) to the user interface.
- **React.js**: A powerful frontend library for building dynamic and responsive web applications.
  shared and explained in advance.

We will start with a simple HTML/JS/CSS, progressively building more complex frontends that
and AI models.

Backend: The Logic and Data Handling

The **backend** refers to the server side of an application where the core logic, databases, and A
processing, and communication between the front end and the server. Key backend concepts
include:

- **Python**: For the backend, we'll primarily use Python because of its popularity in AI and mac
  a wide variety of frameworks like **Flask** and **FastAPI** that make building backend services s

    - **Flask**: A lightweight web framework used to build simple and scalable backends. It's per
      that the frontend can consume.
    - **FastAPI**: Similar to Flask but optimized for high performance and async operations, whi
      applications.
- **APIs**: Application Programming Interfaces (APIs) will be a core part of our backend system
  as APIs that can be consumed by frontend clients or other services. REST APIs (using Flas

Integrating Frontend and Backend

As we progress, we will move from isolated frontend or backend projects to fully integrated ap
will unfold:

1. **Backend APIs (Python)**: We'll learn how to create flexible and scalable backend services
   your models as REST APIs that can be called from the frontend.
2. **Hosting and Deployment:** Lastly, we will cover how to host and deploy these applications.
   like AWS for deploying both the frontend and backend.

Frameworks and Tools in Focus

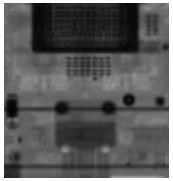
- **Frontend:** React.js, HTML, CSS, JavaScript
- **Backend:** Python, Flask, FastAPI, REST APIs
- **Deployment:** AWS

Fundamentals of AI and Neural Netv

Receptive Fields or Attention Spans (THE CORE OF TS

The **main** logic behind getting AI where it is today was to solve the core problem of rec

Let's look at these 2 modalities below:

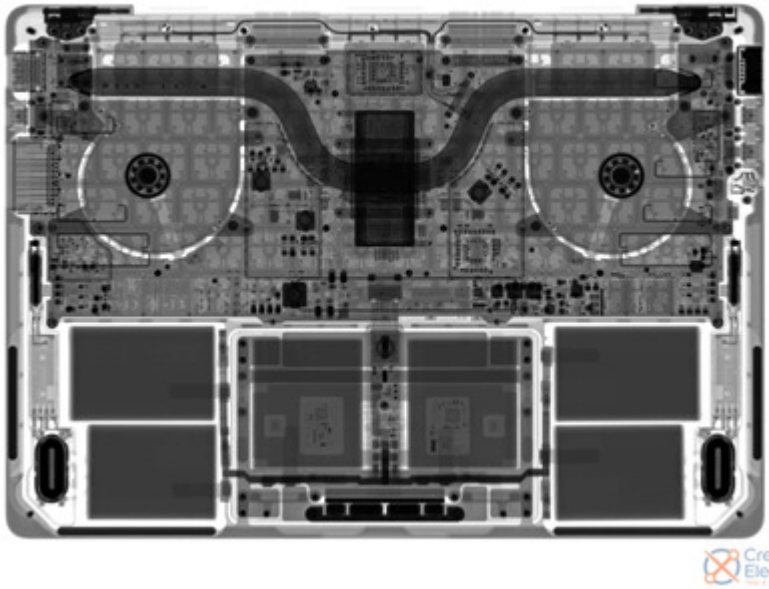


do it. How can you now believe her?

I hope you'll not be able to identify what this image is all about or what this

You need more context in both cases. In the case of the images, it is called the "Receptive Fi
the "Attention Span".

Let's look at the complete information.



We all saw her kill that man;
that captured it. We even fou
But she said she didn't **do it**

The moment you see the complete picture, you can make a lot mor

But we’re getting off ourselves here! I’m supposed to cover this in sections 4-6. Today we’l
things. Not only do we need to have a “complete” picture, but we should also be able to b

building blocks.

For text, it’s simple: the words. But for images, I have to convince yo

Just like we have many languages and characters in to make words, images also have

Do you wanna see?

Example 1 (https://canvas.instructure.com/courses/12597696/files/30810
(https://canvas.instructure.com/courses/12597696/files/308105515/download?dow
(https://canvas.instructure.com/courses/12597696/files/308105408?
(https://canvas.instructure.com/courses/12597696/files/308105408/downloa

And there are MANY MANY different kinds of building blocks. CNNs use a certain kinc

You see since our language is made of words and we need to use a lot of them to process w
understanding that “words” are the building blocks for NL

For images it’s not so obvious. It’s not ONLY the colors that are required to “see” something
possible colors.

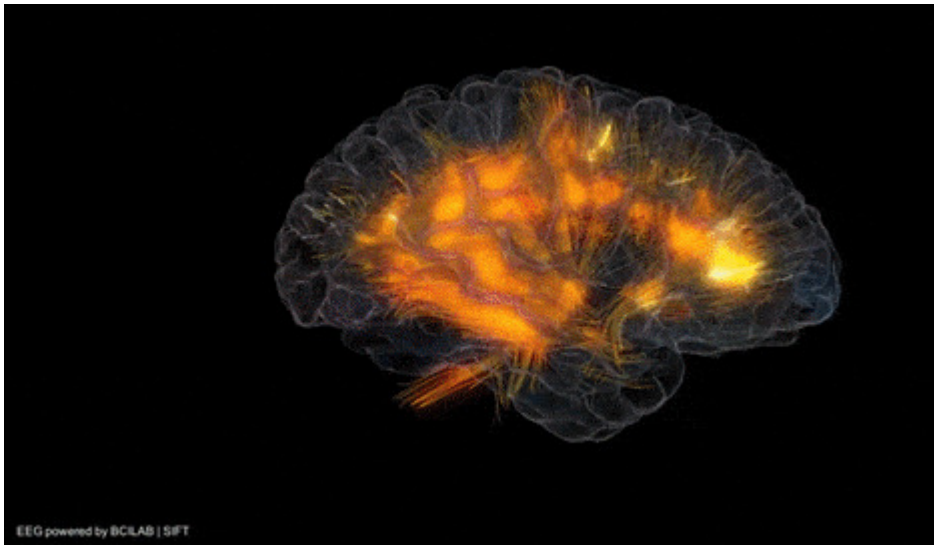
These edges and gradients are combined to make textures and patterns. These textures and
of objects. And these parts of objects are then combined to make objects! These objects a

This is very similar to how characters are used to make words. Words are then used to make sen
a sentence do they have the right meaning). Then sentences are combined to make a paragrap
whole stories or contexts!

Today our focus is to be able to break things down.

We need to break a song into building elements and so or

Building the Intuition: Data Representation



*An ultra-dense connected network of flowing information **inwards!***

What do we observe in the image above?

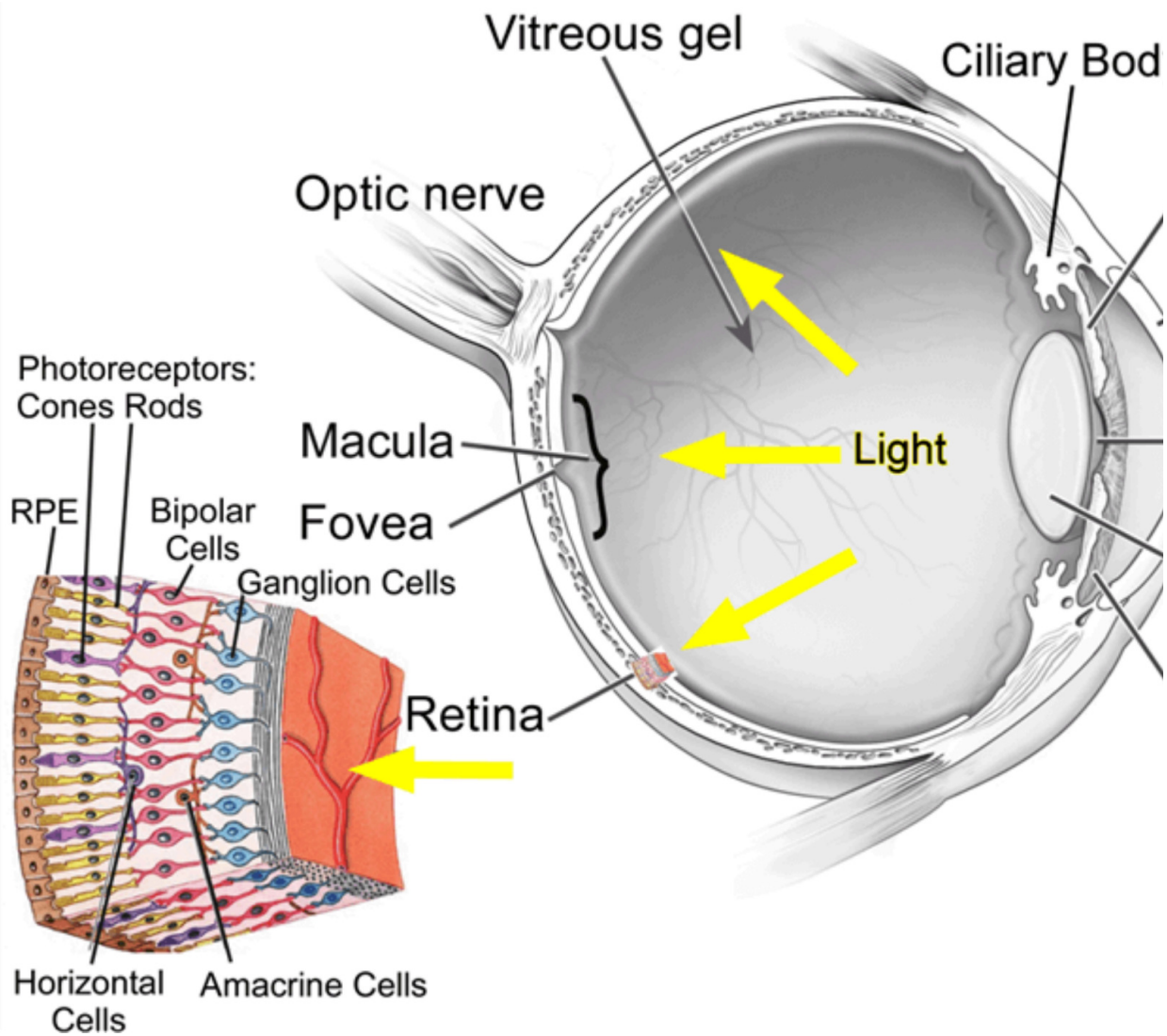
In this course, we will focus on vision and text (with a bit of audio as well). Let us see how visi

Human Eye

Human Eye Made on MidJourney!

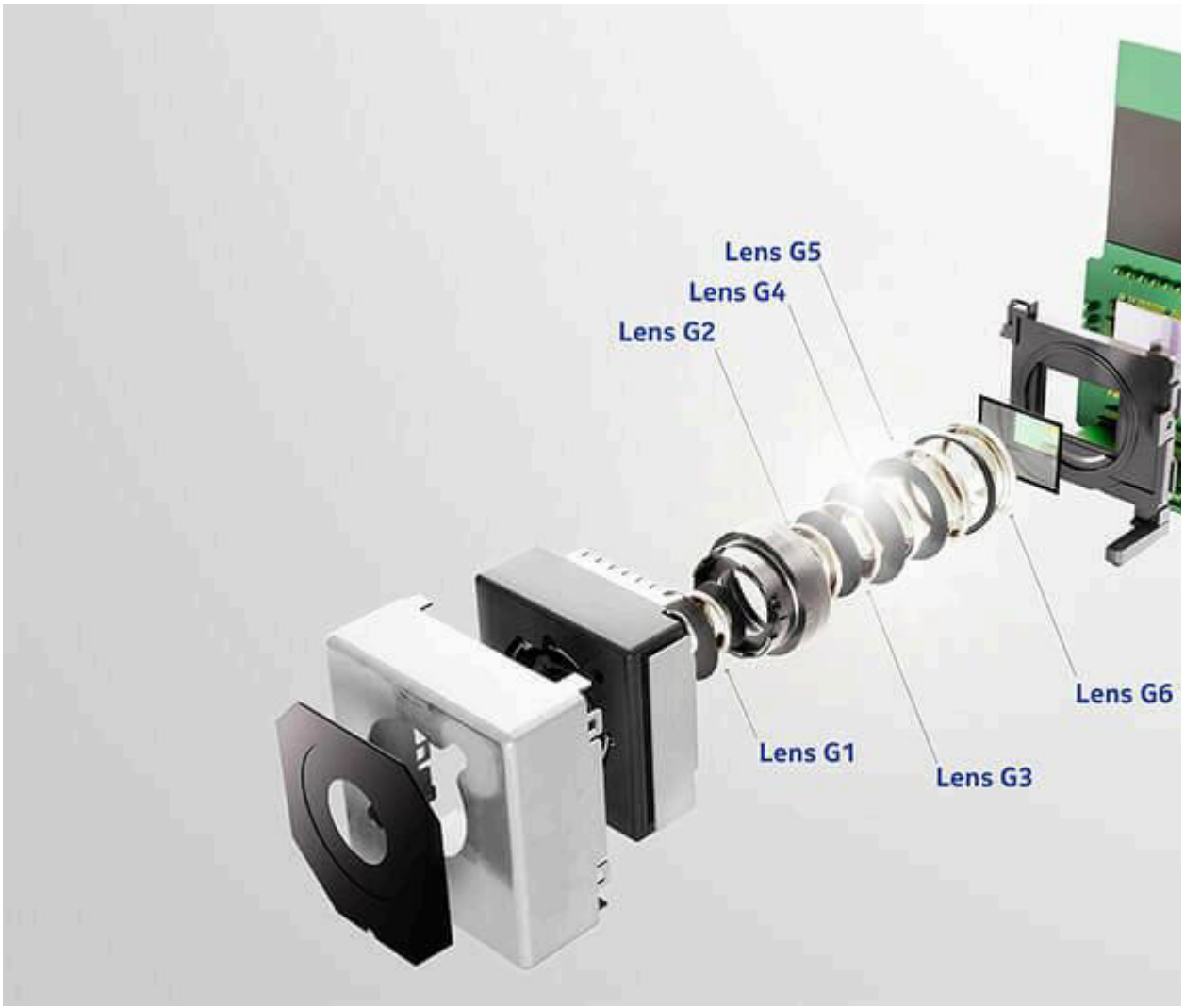
The human eye is the result of billions of years of evolution **(read this amazing post (https://x.com/VisualCap/status/1829927402174075020)** , and what our eyes and

Those beautiful orange structures are actually muscles that pull the lens to help us focus,
Camera lens. Light falls on the photo sensors in the center (just as it would



Source (https://www.intechopen.com/chapters/26714)

Similar to our eyes, humans have invented the whole sensing system, and th



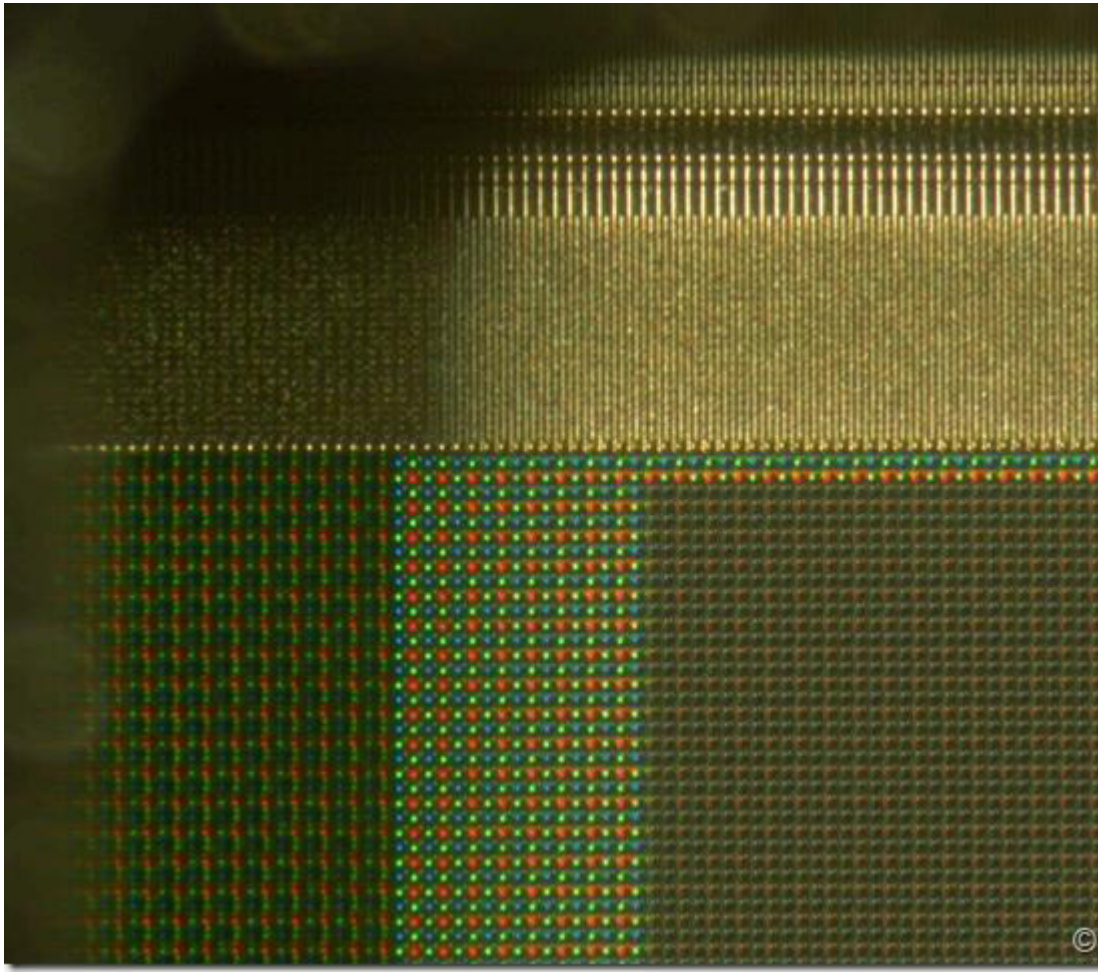
A Camera lens and sensor system

If we use an electron microscope and look at our retina, this is wh

Rods and Cones under electron microscope

Rods: Cones ratio is 20:1. Rods see BnW and Cones see cc

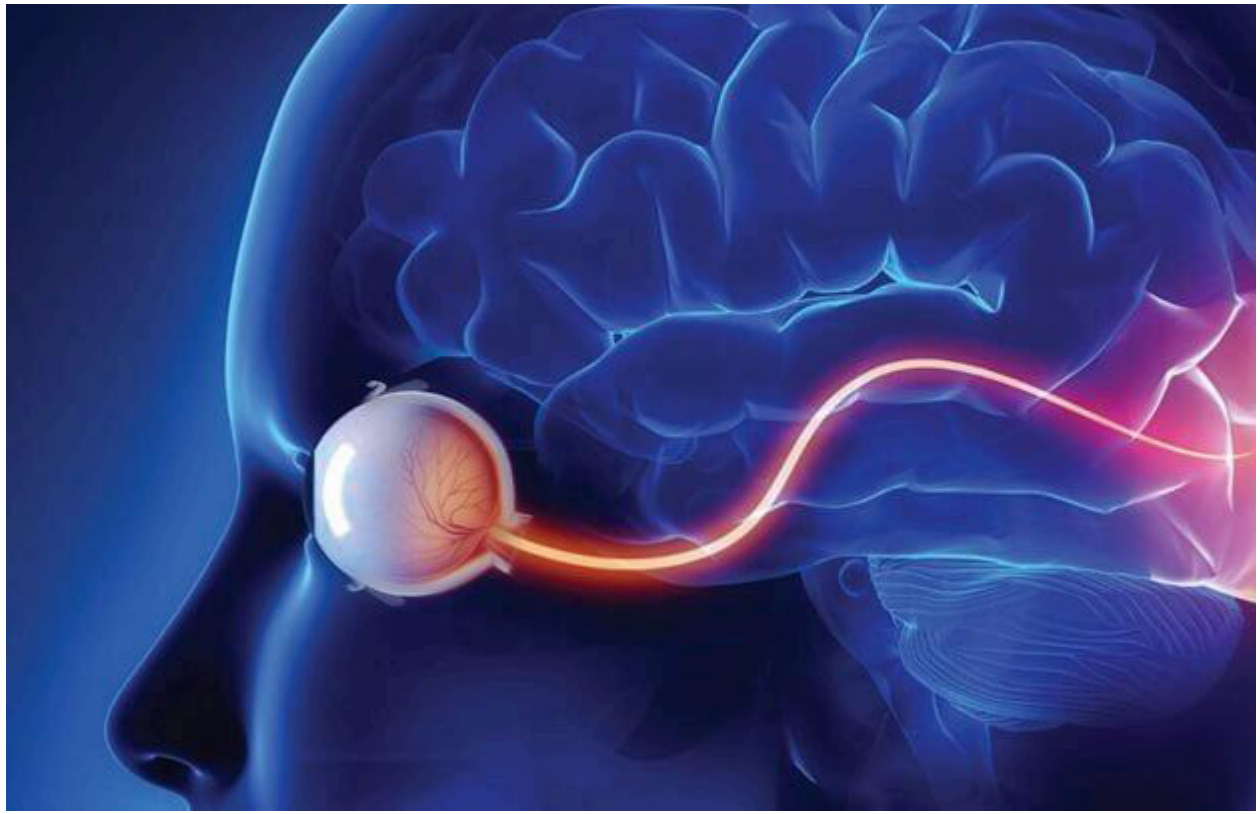
This is what a camera sensor looks like:



Camera CMOS sensor under a microscope

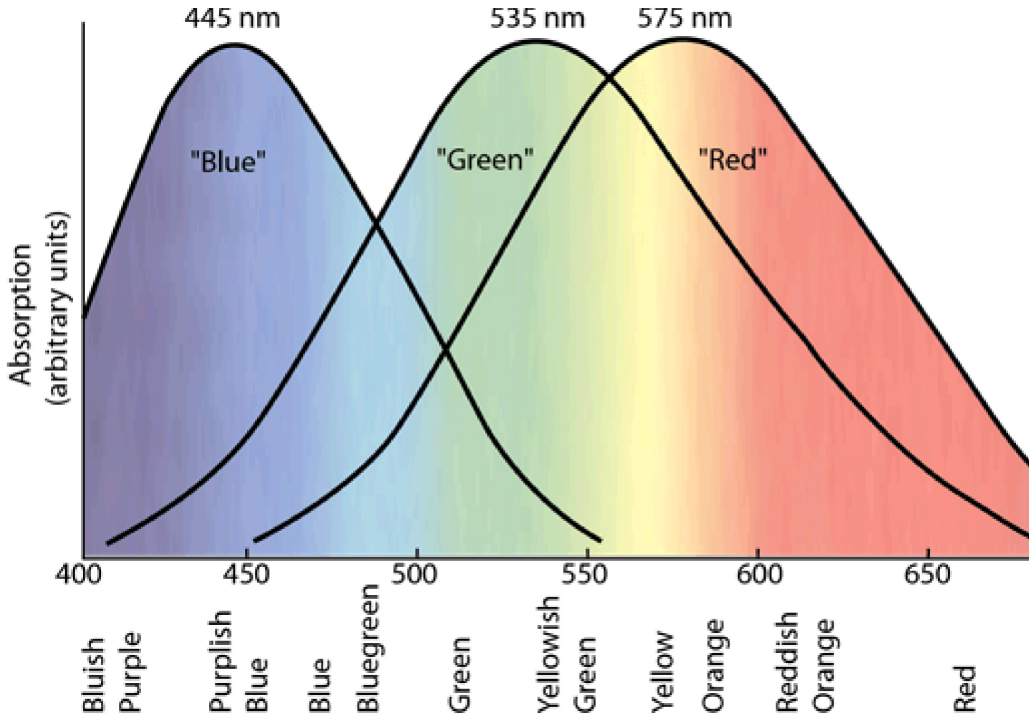
And just like a camera, the human eye also has **three types (https://askabio cones#:~:text=We%20have%20three%20types%20of,blue%2C%20green%2C%20and%20red)** of

need a lot of light to work, and hence so many rods. Our optic nerve is connected to our visu
place:



The visual path from the retina to the visual cortex

One thing that we should know is, that saying that our cones see red, green, and blue, is mi
colors, and many around them, as can be seen in this ima



Source (http://hyperphysics.phy-astr.gsu.edu/hbase/vision/colcon.h

Understanding this concept is super critical for us! You can download E

(https://canvas.instructure.com/courses/12597696/files/308107835?v

(https://canvas.instructure.com/courses/12597696/files/308107835/download

Channels or Breaking it down!

Any color can be made through a combination of different kinds of "primary color" combinatio
use CMYK, CMY, YIQ, LAB, HSV, HSL, HVC, Munsell, YUV, YCbCr, HSI, CIE
(https://canvas.instructure.com/courses/12597696/files/308108256?
(https://canvas.instructure.com/courses/12597696/files/308108256/download

Representing same image in different color models.

What I want you to focus on, is the fact that there are numerous ways of represent

And you must realize this.

On a side note:

I want you to realize the fact that nearly the same **word sound** can be made in 293 different v
be represented in any of the 19500 languages and dialects, just in **India** **(https://www.thehir**
is-home-to-more-than-19500-mother-tongues/article24305725.

Again the emphasis here is to understand the fact, that a "complex" concept could be re

Just like different units of measurement:

For distance

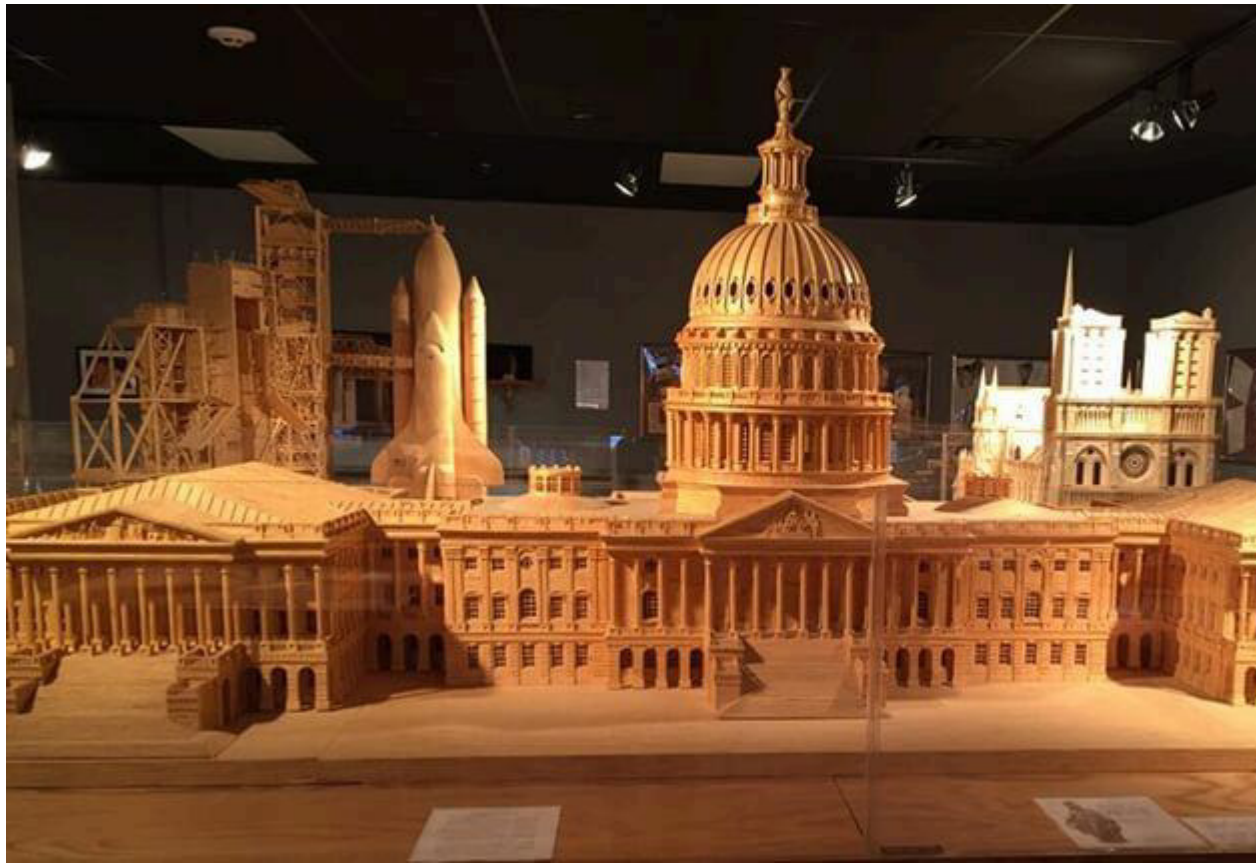
Kilometer (km), Meter (m), Centimeter (cm), Millimeter (mm), Mile (mi), Yard (yd), Foot (ft), Inc
Unit (AU), Light-year (ly)

For Weight

Kilogram (kg), Gram (g), Milligram (mg), Microgram (μg), Pound (lb), Ounce (oz), Carat (ct), unit (amu), Moles (M)

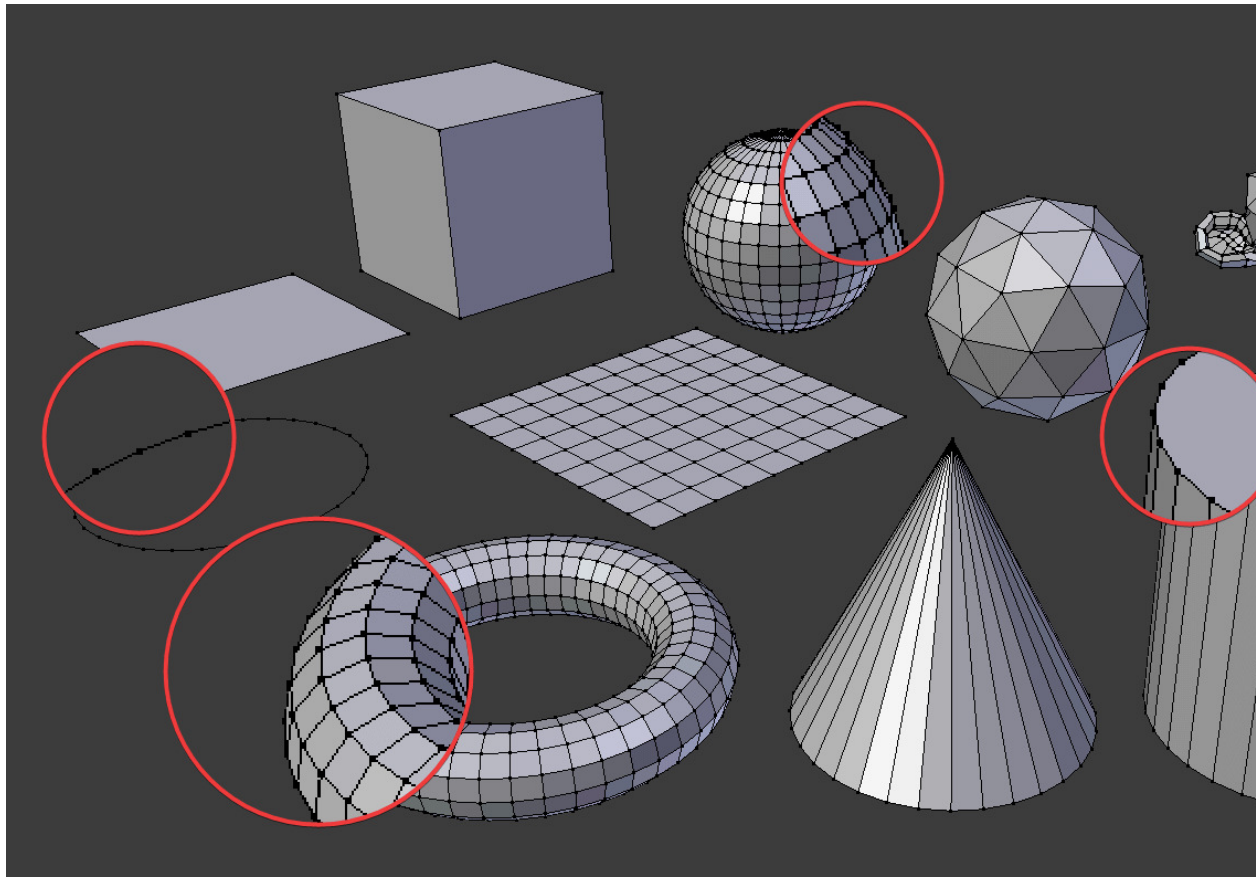
Expressing the same concept in these different units does not change the underlying value, use them is different.

Now I want you to guess what is this



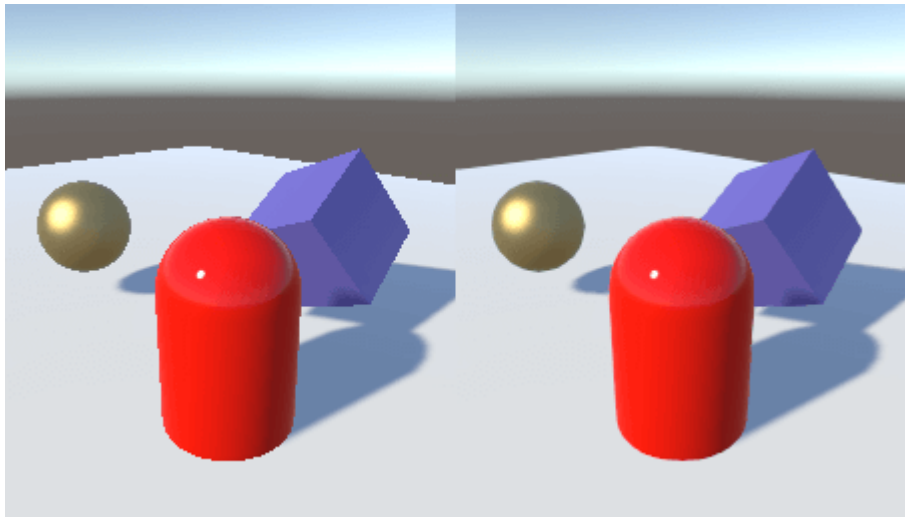
Source (https://www.matchstickmarvels.com/the-models/)

In 3D graphics, we actually never see curved edges. Ultimately, it is broken do



Everything in graphics in creating using a line.

We have to use a lot of tricks to fake smoothness



LeftL: Original image. Right: Antialiased image

So we have seen a few simpler "primary" representations (like colors, edges, and characters concepts).

And there is an ocean of different kinds of representations between these primary a

For instance

Edges & Gradients, Textures, and Patterns, Part of Objects and

Characters, Words, Sentences, Paragraphs, Book

Air vibration, Phoneme, Sound of a word, Sound of a full sen

Etc.

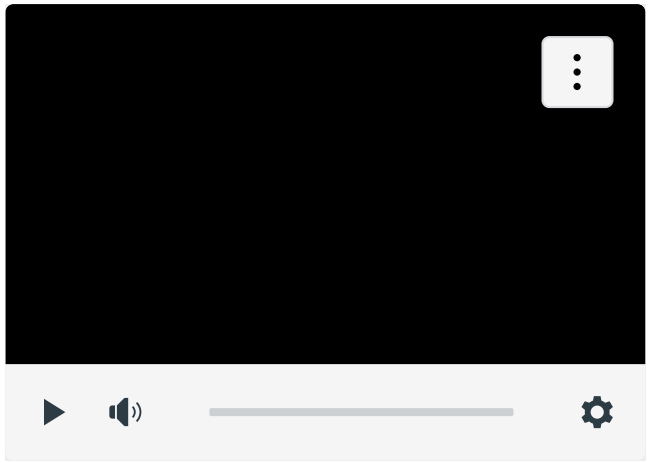
When I segregated these concepts above in a bucket like "Sound of words" or "Textures and F
something that we know as a "Block" of layers. Every layer would have 100s of thousands
responsible only for one thing. For instance:

At some **higher-level blocks**, a few of the channels could be vocals

, synth, and drums.

When we think in terms of channels, we can "split" the song being played above into its indivi
by piano, guitar, bass, drums, etc. When everything comes together

Let's listen to this:



This was generated by **MusicLM** (https://google-research.github.io/seanet/musiclm

The main soundtrack of an arcade game. It is fast-paced and upbeat, with a catchy electric
easy to remember, but with unexpected sounds, like cymbal crashes

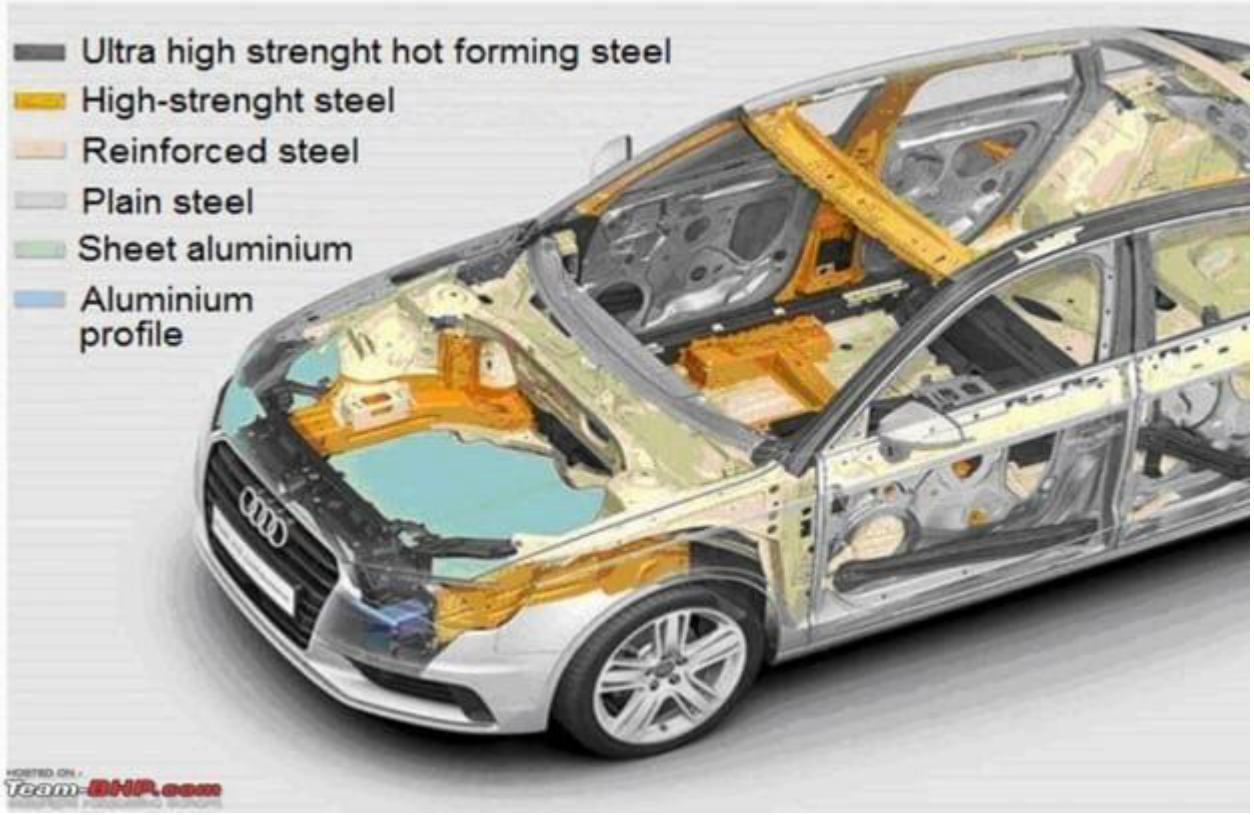
In the image below, one of the "lowest-level blocks" could be the alphabet. What is the maxim
block (assuming the size of the characters does not matte

Typography Art

In the image below, a somewhat mid-level block could be represented b

Ingredients for Rice Pulao

Based on the concept, these channels (individual concepts/features) would be anything



Car segmented based on its material.

and component level here:

All the components of the car

In both cases, we're using different kinds of channels, but I want you to be convinced that in t
"basic" than in the latter image. The later image had more "complex

What did we learn till now?

Anything complex can be made from simpler stuff!

Well, that is what AI is all about. How do we break down the problem into the simplest possib
trillions of ways to represent anything that we may want

Origin or The Story of Two Missing Cats

CAT 1

Image created on MidJourney.

CAT 2

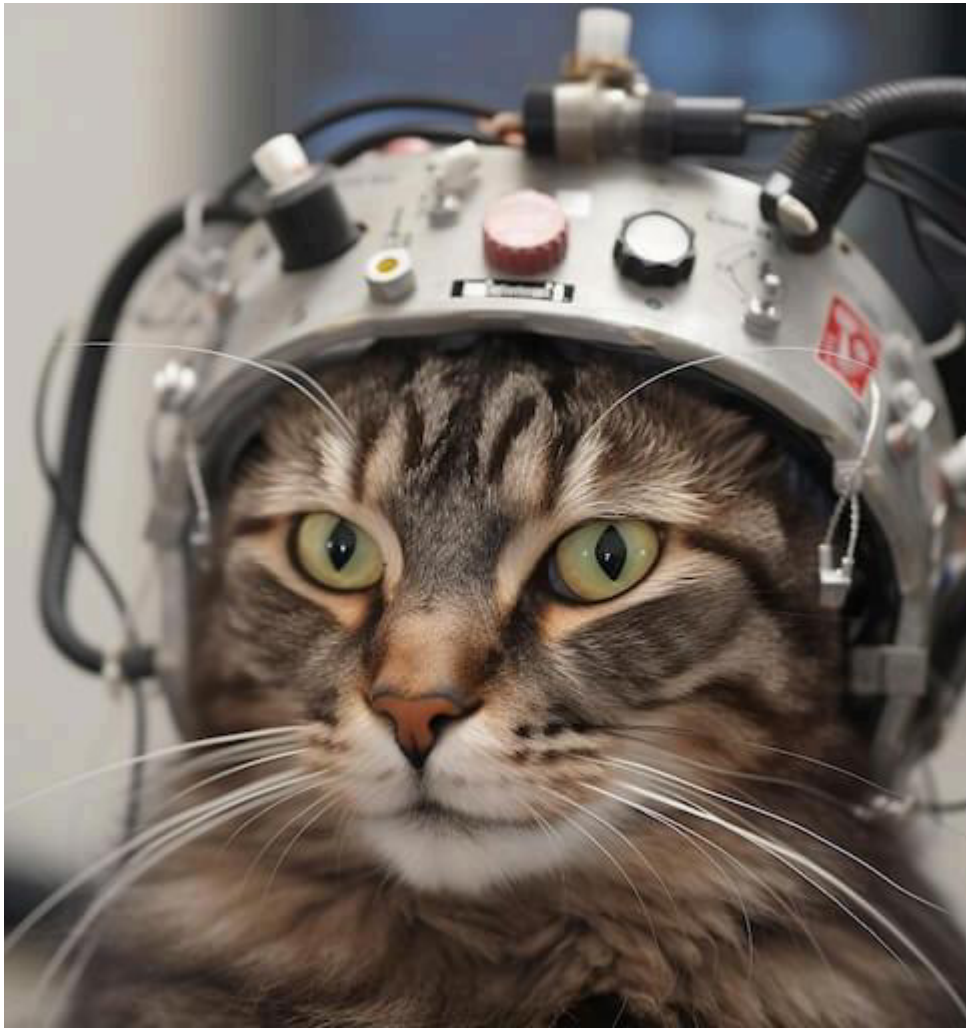
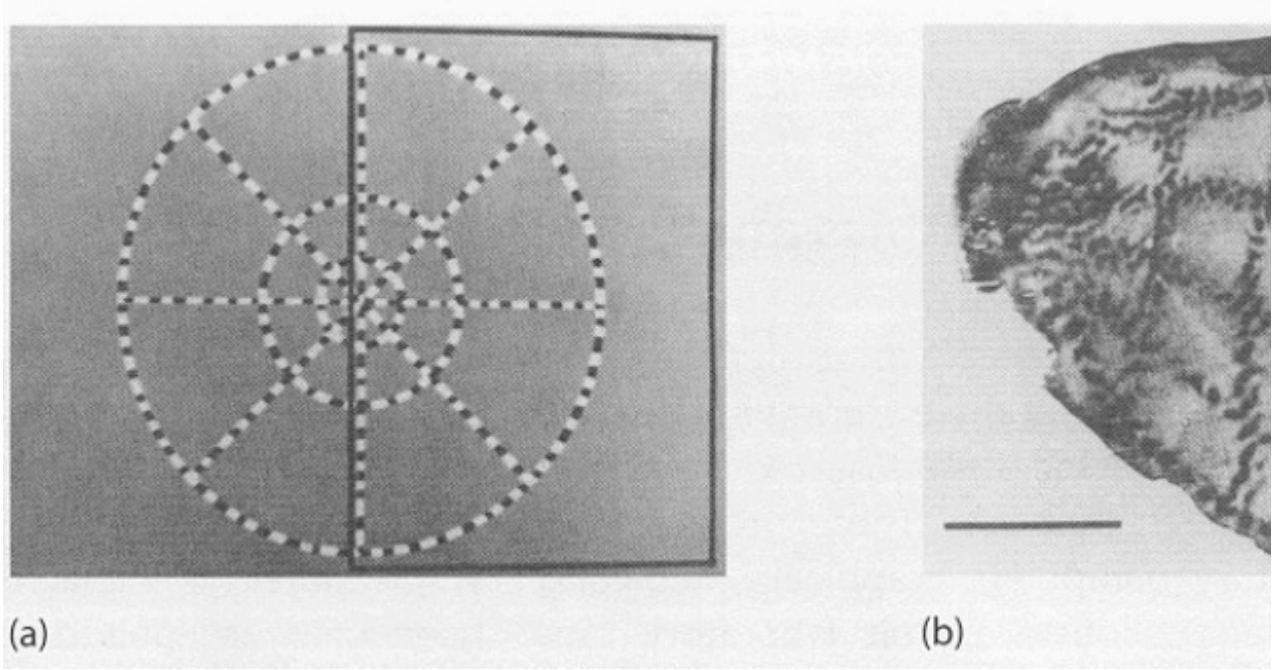


Image created on MidJourney.

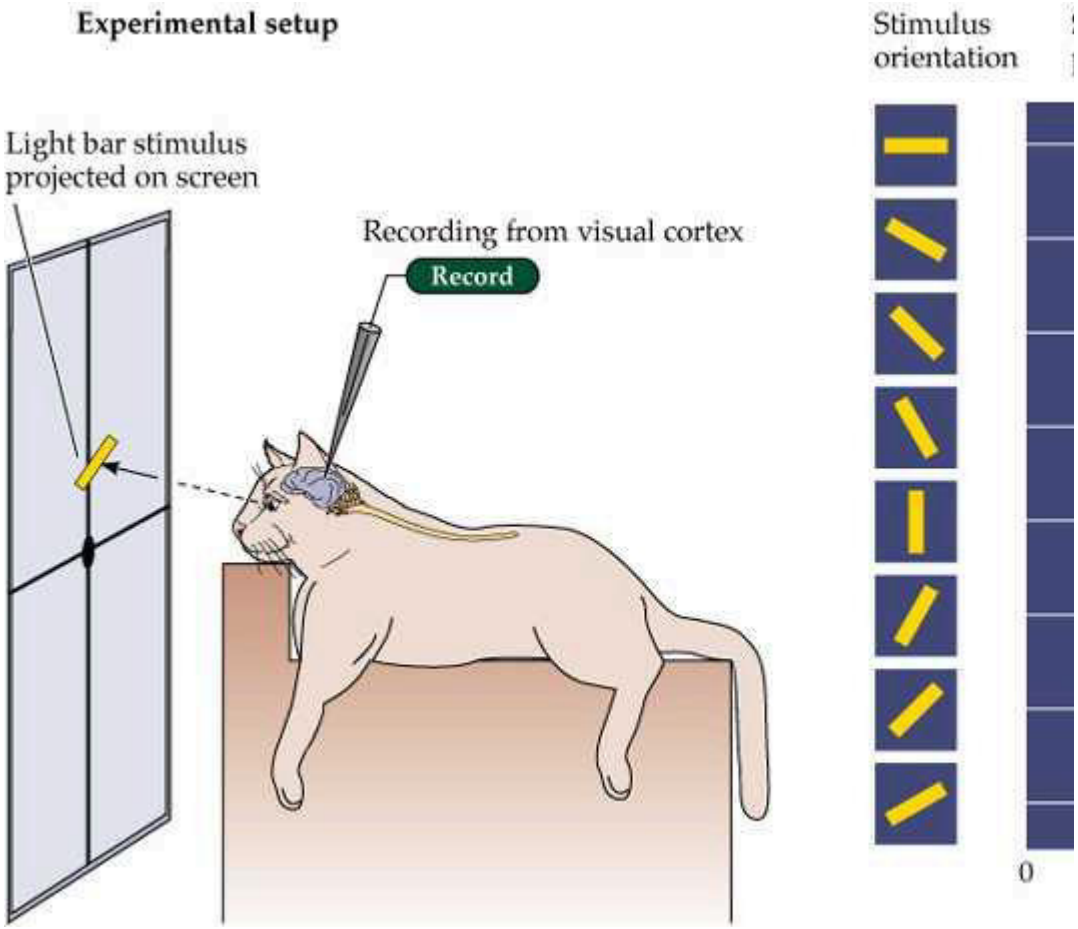
CAT 1: Experimental Results



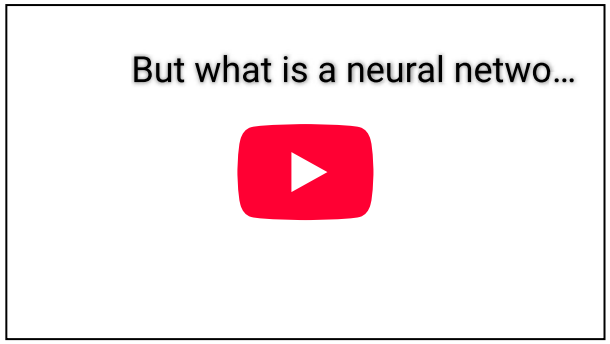
If you are interested in learning more, read more about retinotopic maps and fM

When we see the image on the left, it gets "printed" on our brains, literally, as shov

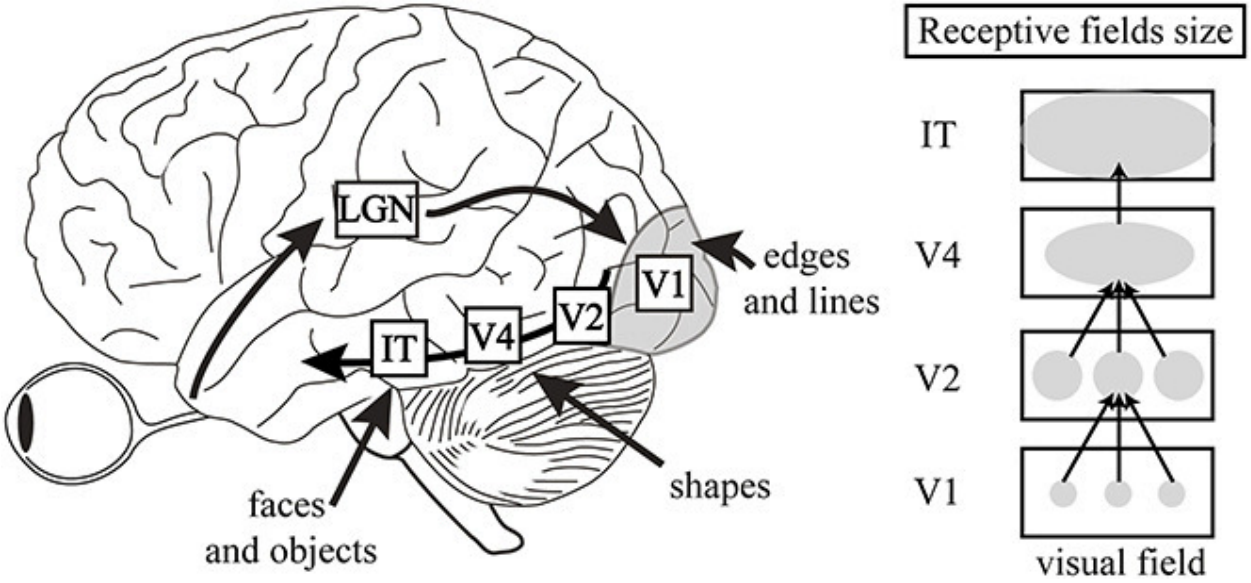
CAT 2: Experimental Results



Would highly recommend you to look at this video and subscribe to



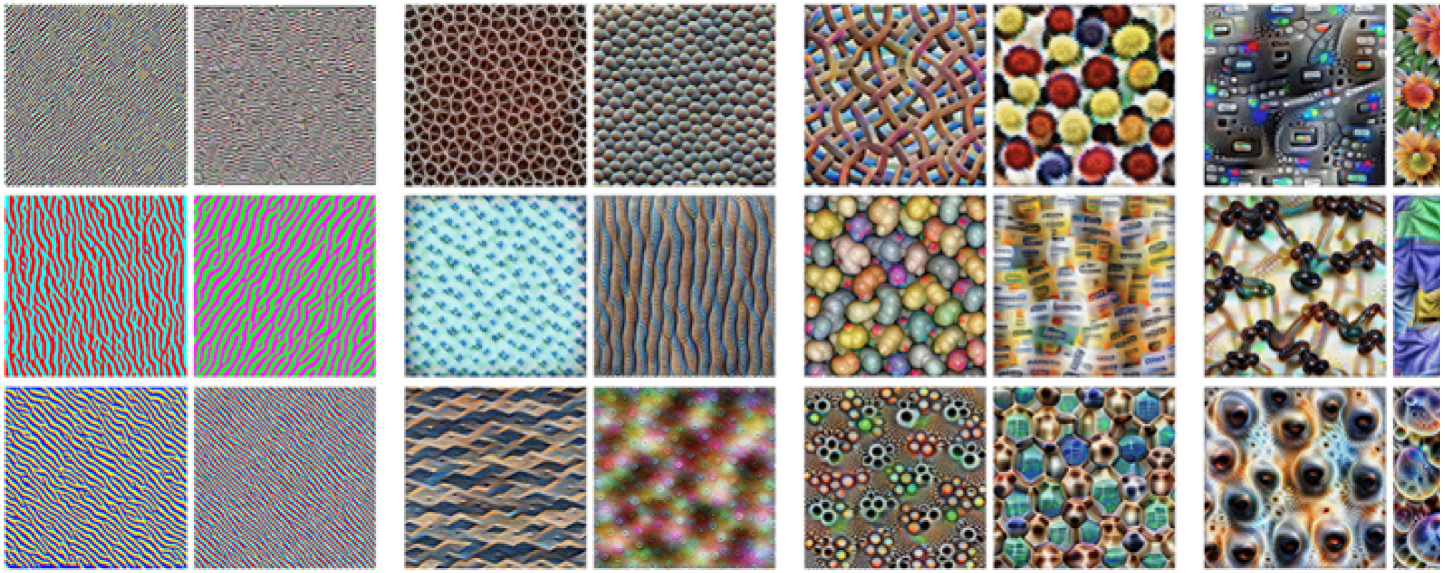
In our own visual cortex, we start with simple shapes (edges and gradients), combine
(patterns/textures/part of objects), and finally get to see (the objects



If you are interested in the above topic, please read more on this ***link*** ***(https://www.researchgate.net/figure/Left-A-typical-hierarch***

starts-at_fig1_267872860) .

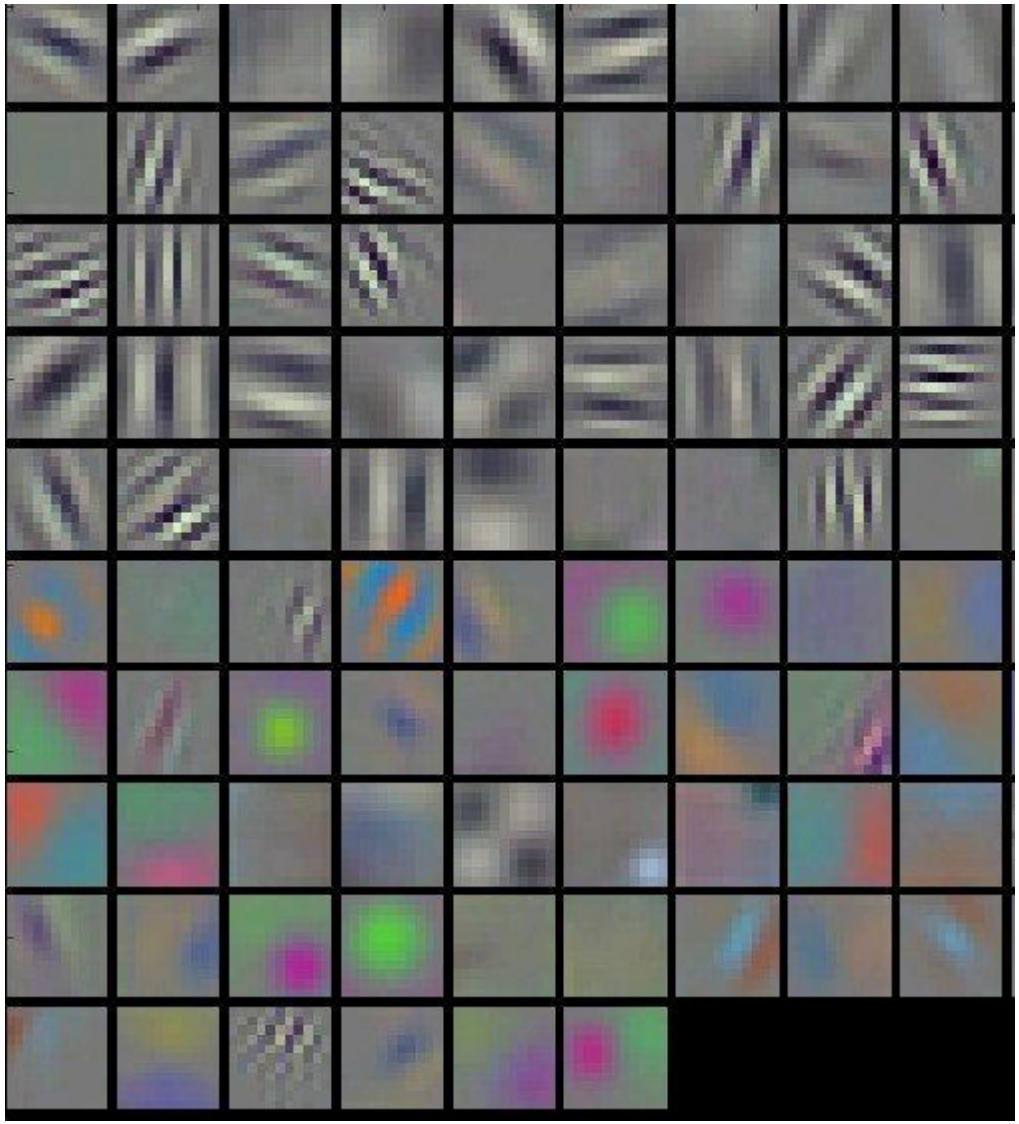
Something dramatically similar happens in the deep neural networks! And being artificial, we
happens inside them!



A Kernel is a feature extractor, and these feature extractors are what is learned in a DNN. Above you see an "extrapolation" of the f

that feature.

We can even query the very first "block" of the DNN and see individual features



Above what you see is what a kernel or feature extractor or learned matrix exacts. Since this kernel was 11x11 in size it can extract big image, we'll see something similar to the image above.

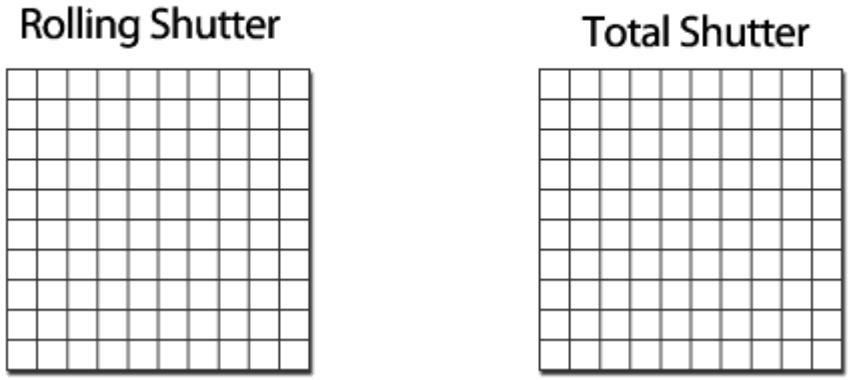
Let's look at this time-lapse to appreciate how simple strokes can make something really bea
while our brain-eye system will spend less than 100ms

Again, the concept we are stying to learn here is that *complex structures or things can be buil*
that we are communicating right now using just 26 alphab

Core Concepts

We need to understand a few other concepts before we jump to Convolutions. One such c
difference between a Rolling Shutter Camera and a Global/Total Sh

The reason behind these effects is how the pixels are actually

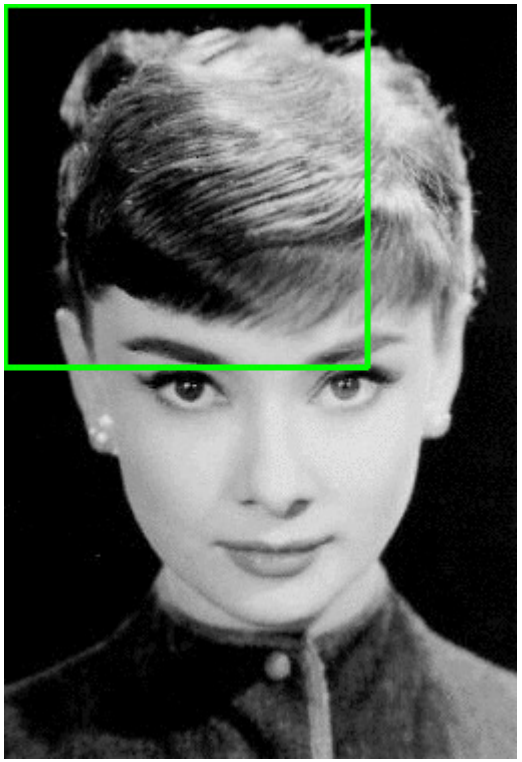


Rolling vs Global Shutter

Of course, Global Shutter cameras always make sense, but they are complex, expensive, a
phones use Rolling Shutter cameras, but now the speed of capture has increased a lot, hen
Shutters are digital in nature, and Global Shutters are mecha

DNNs would use a rolling shutter kind of concept, but our brains use global sh

A slightly extended concept was used during the Jurrasic era by computer vision engin



Sliding window

To actually implement a global shutter kind of processing in DNNs, we would need to change
processing is based on "clock ticks" and we can process a small amount of data

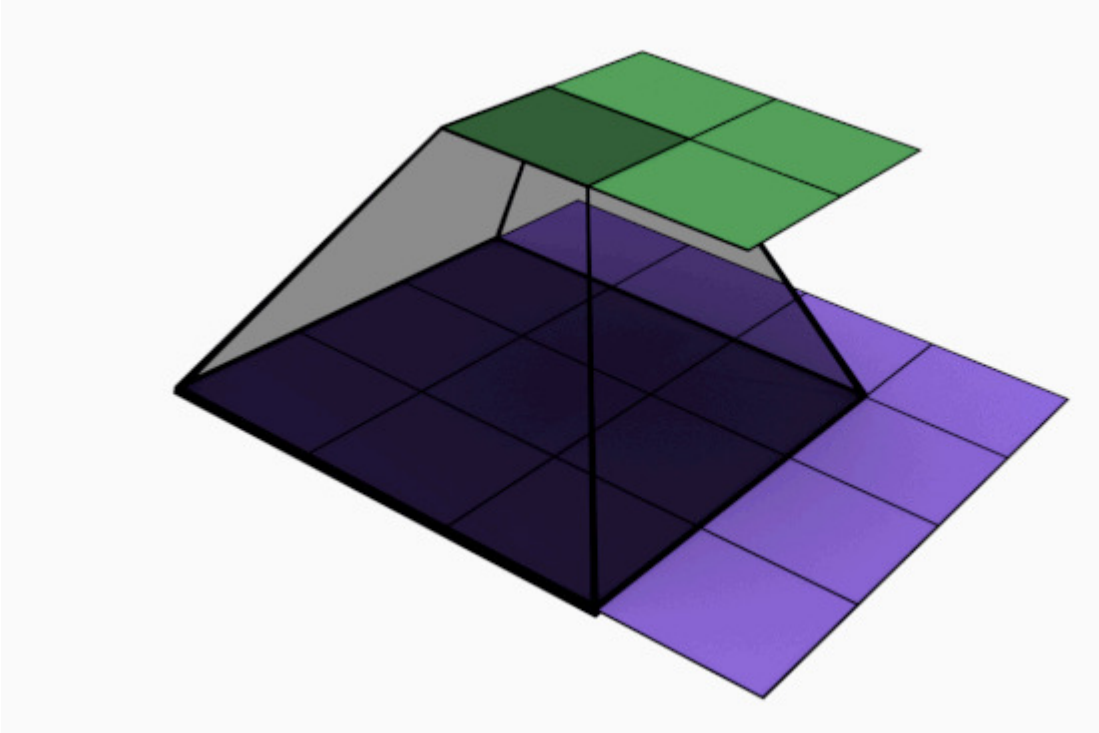
In both of these examples, let us focus on the "area in context". In the images above we eith
block of data that we are processing or "reading".

Now "reading" here basically means, that we are storing it for some kin

The pixels that are going to be stored will be worked upon by some algorithm, which ultima
would be multiplied/divided/added/etc to them.

These numbers are called kernels.

Convolutions

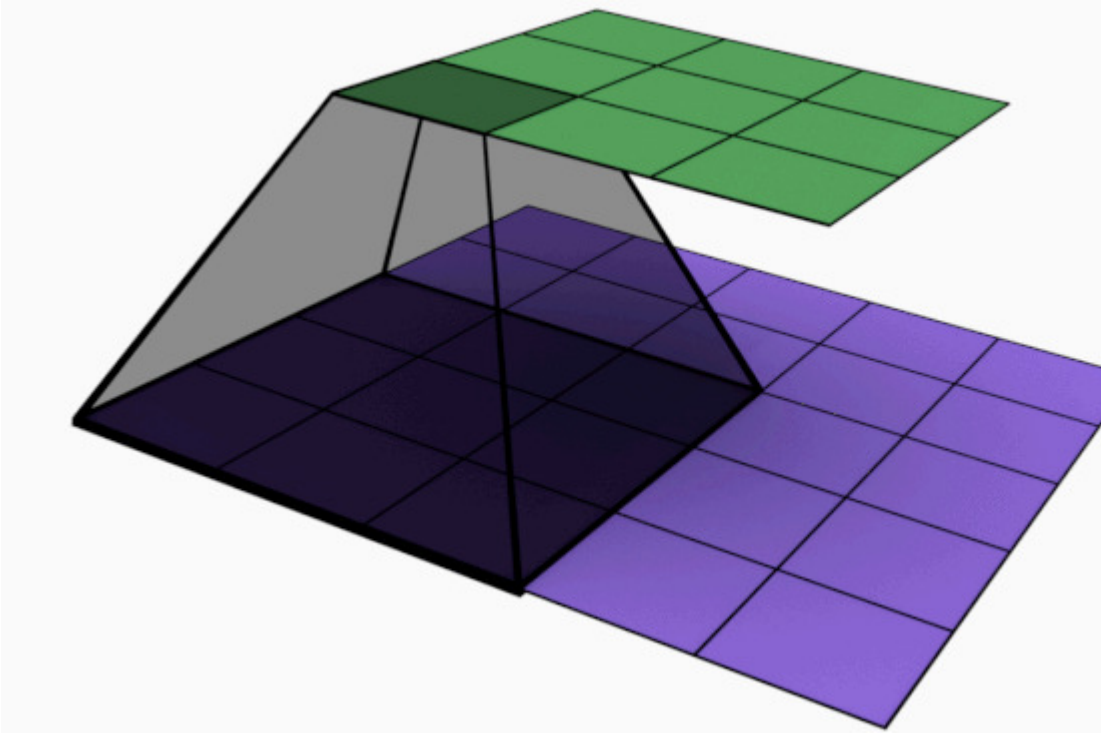


A 3x3 kernel on a 4x4 image.

Here we are "reading" 3x3 numbers on a 4x4 image. The moment we read these 3x3 pixels
numbers (identified by a DNN). These "other 3x3 numbers" are cal

Let's check this quickly (https://codepen.io/wallat/embed/yLyr

So, there exist simple 3x3 matrix numbers that can easily identify bas



A 3x3 kernel on a 5x5 image.

So if we convolve a 3x3 kernel on a 5x5 image, the output we would create will have a resoluti

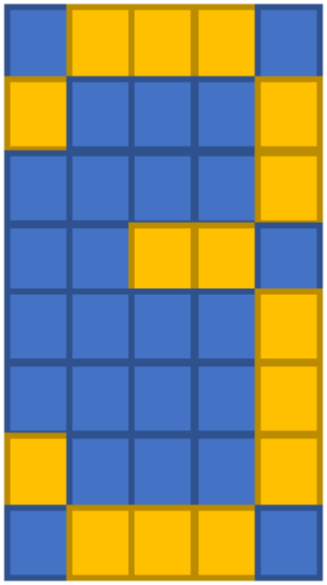
- we are not going outside of the image (by adding imaginary numbers). This by the way is c
  black/white (0/1/255) numbers and then allow our kernel to slide out of the image (why wou
- we are not using a stride of more than 1, i.e. we will cover each 3x3 section immediately aft

In the images above, our kernel skips/jumps only 1 pixel. If we were to jump/skip 2 pixels, ther

Fully Connected Layers

Convolutions are fairly new compared to Fully Connected Layers, and quickly took over, the
because FC layers suffered from a serious issue (which transform

Let's look at this image:



Digit 3: represented by 5x8 pixels

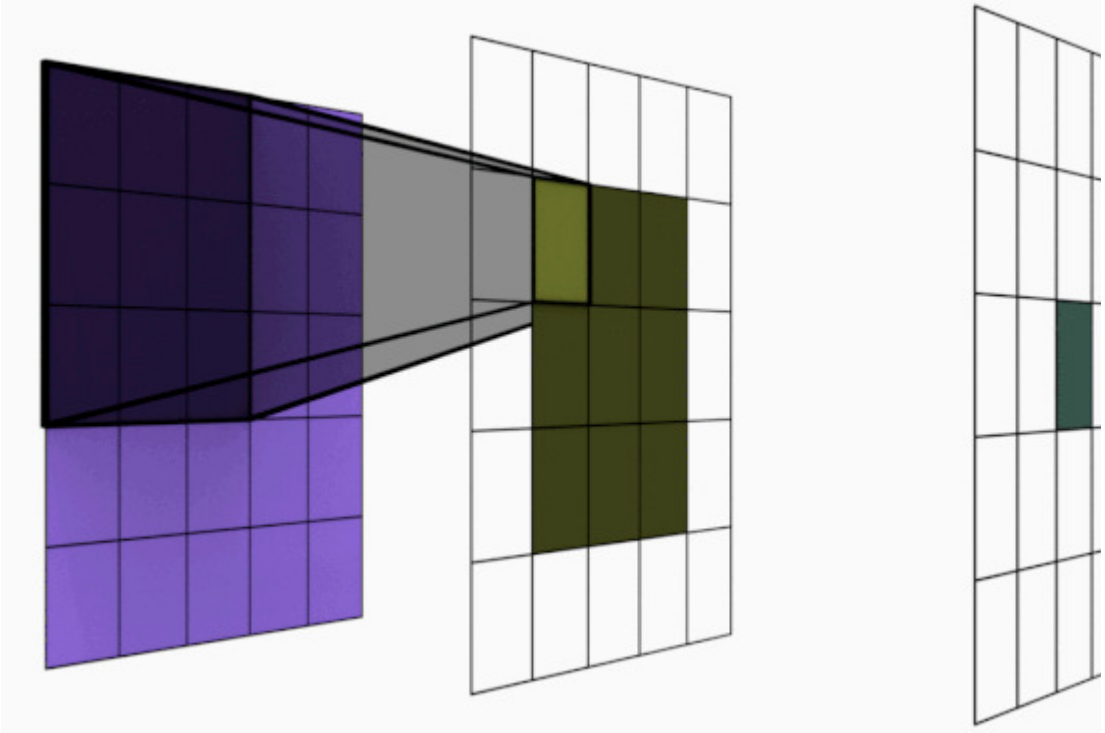
We can unroll this image and make something like this:

Same Digit 3: represented by 40x1 pixels.

More about these in the next session.

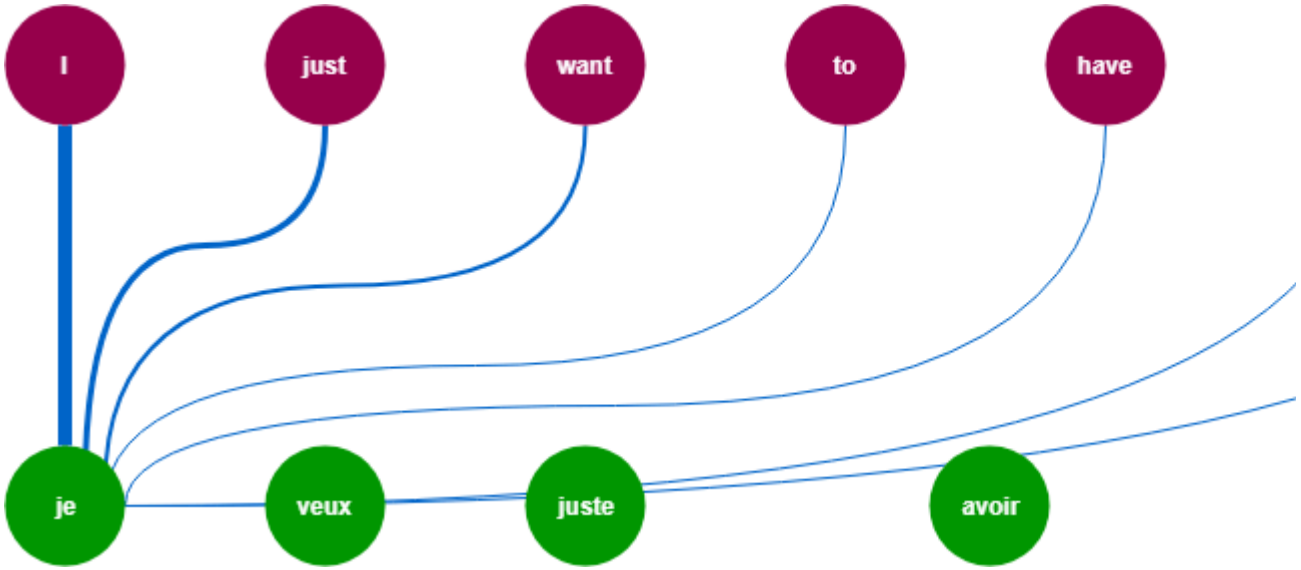
Receptive Fields and Attention

The most important image in ERA for Vision:



Receptive Fields

The most important image in ERA for NLP/Audio:



Let's check the ***source*** *(https://www.linkedin.com/pulse/explanation-attention-based-encoder-decoder-deep-*

Important Note on Vision vs NLP.

In the case of language, we have broken down concepts into words. Each word means some
example) we know what it means. In fact, we (sort of) have a fixed exhaustive dictionary of w
GPT4 out of!

In the world of 3D graphics, we have done the same thing. We have these edges that make
make primitives like spheres, cubes, etc., and complex sha

But we are (not all) graphics designers and we don't have the intuitions to break down th

But for a moment let's imagine that we indeed had characters and words like concepts in visi
directly use them to define the image/world (just like in NLP). And now you immediately start
for Vision and NLP would meet!

The convolutions we saw above would help us create these vision words (edges, gradients,
Layers (Transformers) would help us connect them with language in a

The Assignment

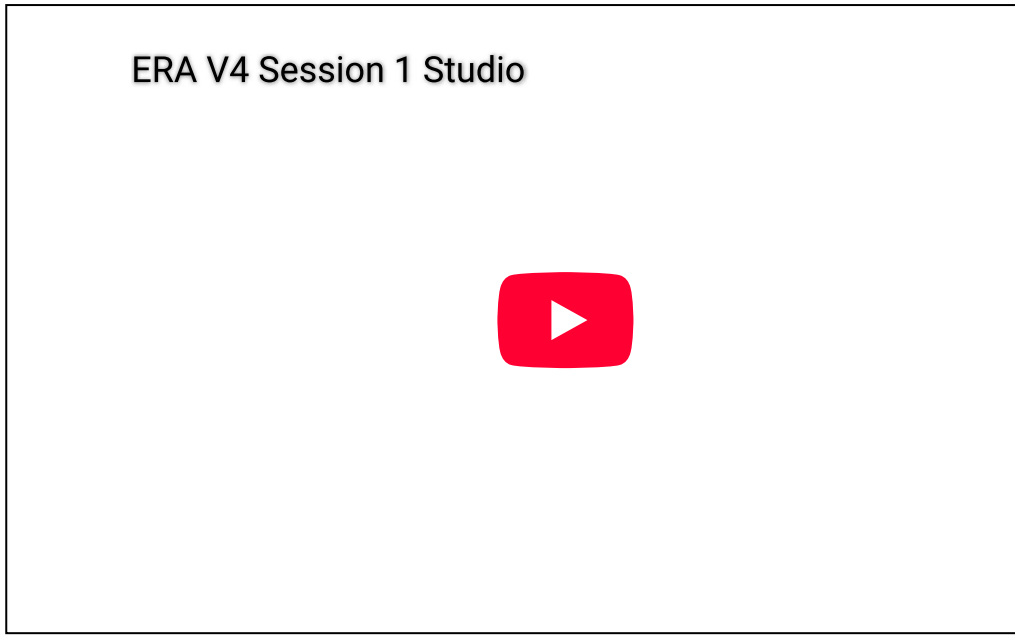
1. Your quiz questions are on a separate quiz listed as "S1 Quiz".
2. Make Something!
    1. We saw 2 examples of using Cusor to make Chrome Extensions! Be creative, make son
    2. Here are some examples **(you're not supposed to use these; these examples are to**
        1. Find and list the latest movies on Netflix
        2. Find the next Cricket Match or F1 Grand Prix
        3. Change the font on any website to anything that you like!
        4. Make a clipboard that stores all of your "copy" texts on your browser
        5. A small web browser-based game!

3. Do not spend more than 1 hour doing this; this may be too addictive!
4. Follow the steps. It is better to start from scratch, rather than to fix big. Next time, improv

3. Once done,
    1. upload a short video of it working on YouTube and share the link (to capture the video, y
    2. Zip your folder and share the code.
4. Did you share it on LinkedIn? If yes, please share the link and get 200 pts extra! This is opt
5. 200 + 200 Pts
6. 0.99 Weeks

Videos

Studio (Please prefer GMeet Recording below)



Google Meet (**Chat (https://canvas.instructure.com/courses/12597696/files/30**
(https://canvas.instructure.com/courses/12597696/files/308331885/download?downl
(https://canvas.instructure.com/courses/12597696/files/308333901?v
(https://canvas.instructure.com/courses/12597696/files/308333901/download

ERA V4 Session 1 GMeet