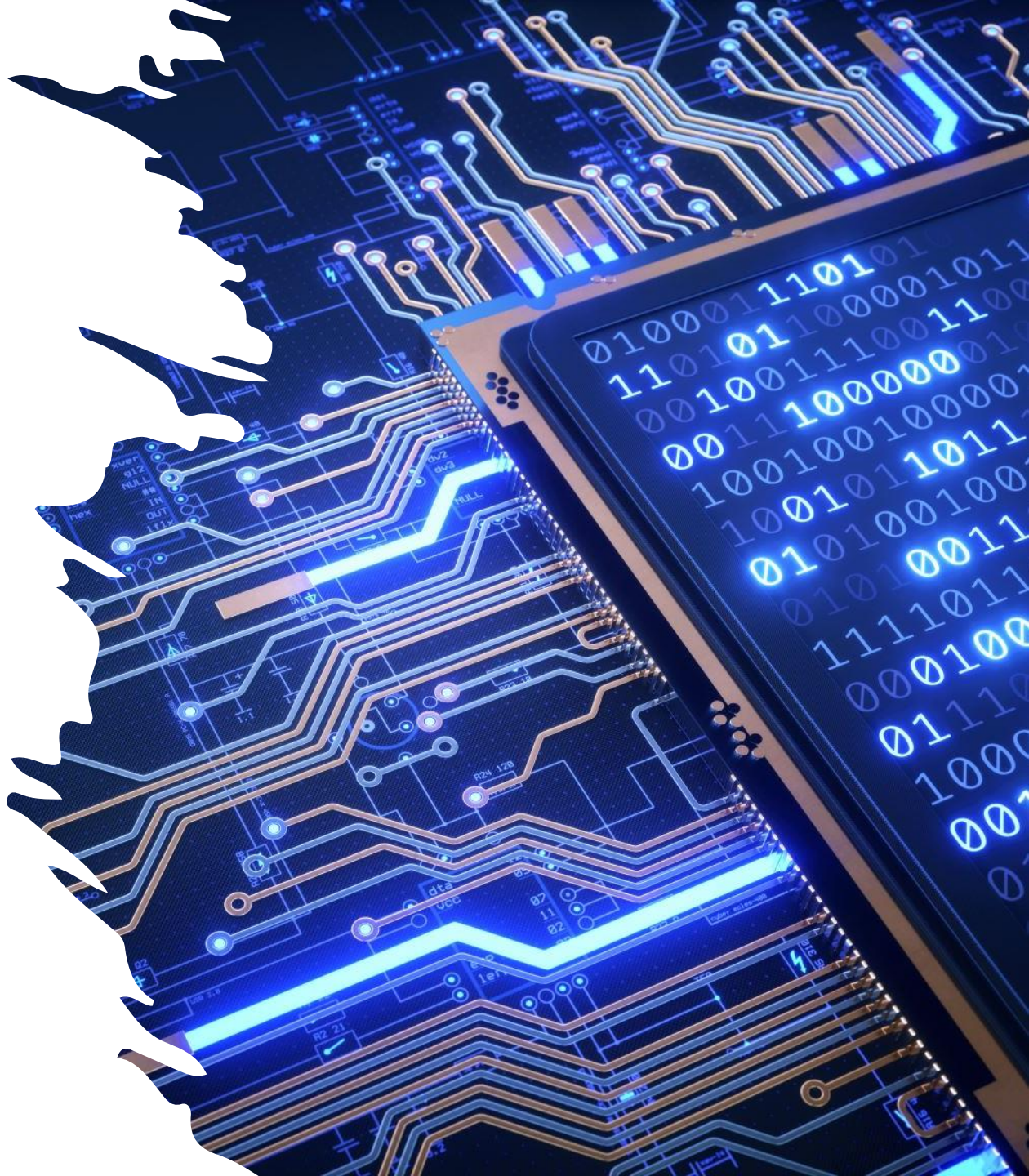
# Computer Vision Beyond Image Classification

RYAN T. WHITE, PH.D.

FLORIDA INSTITUTE OF TECHNOLOGY



# Background

Grew up in Logan, WV

Super into math!

the whole city →

Moved to Florida in 2009
for grad school in math

# Wheelchair-user since 2009

2008

- B.S. Mathematics
  Chadron State College

2015

- Ph.D. Applied Mathematics
  Florida Tech

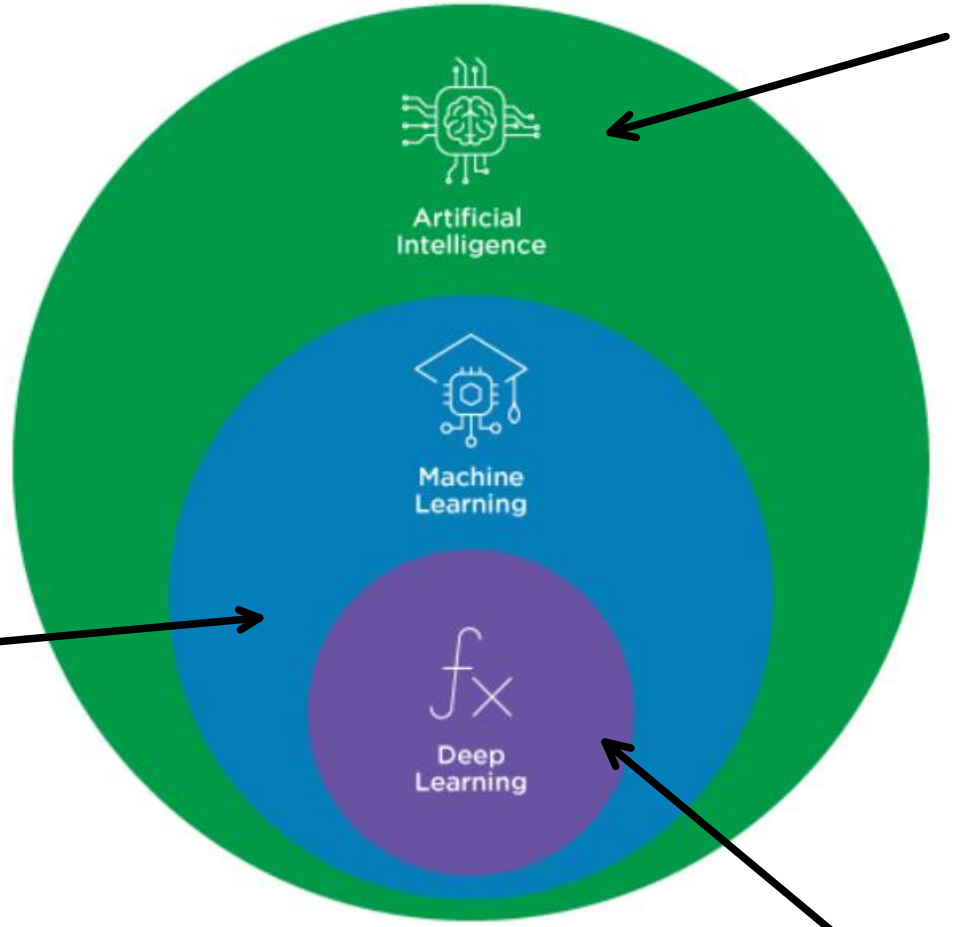
2019

- Assistant Professor
  Florida Tech

2020

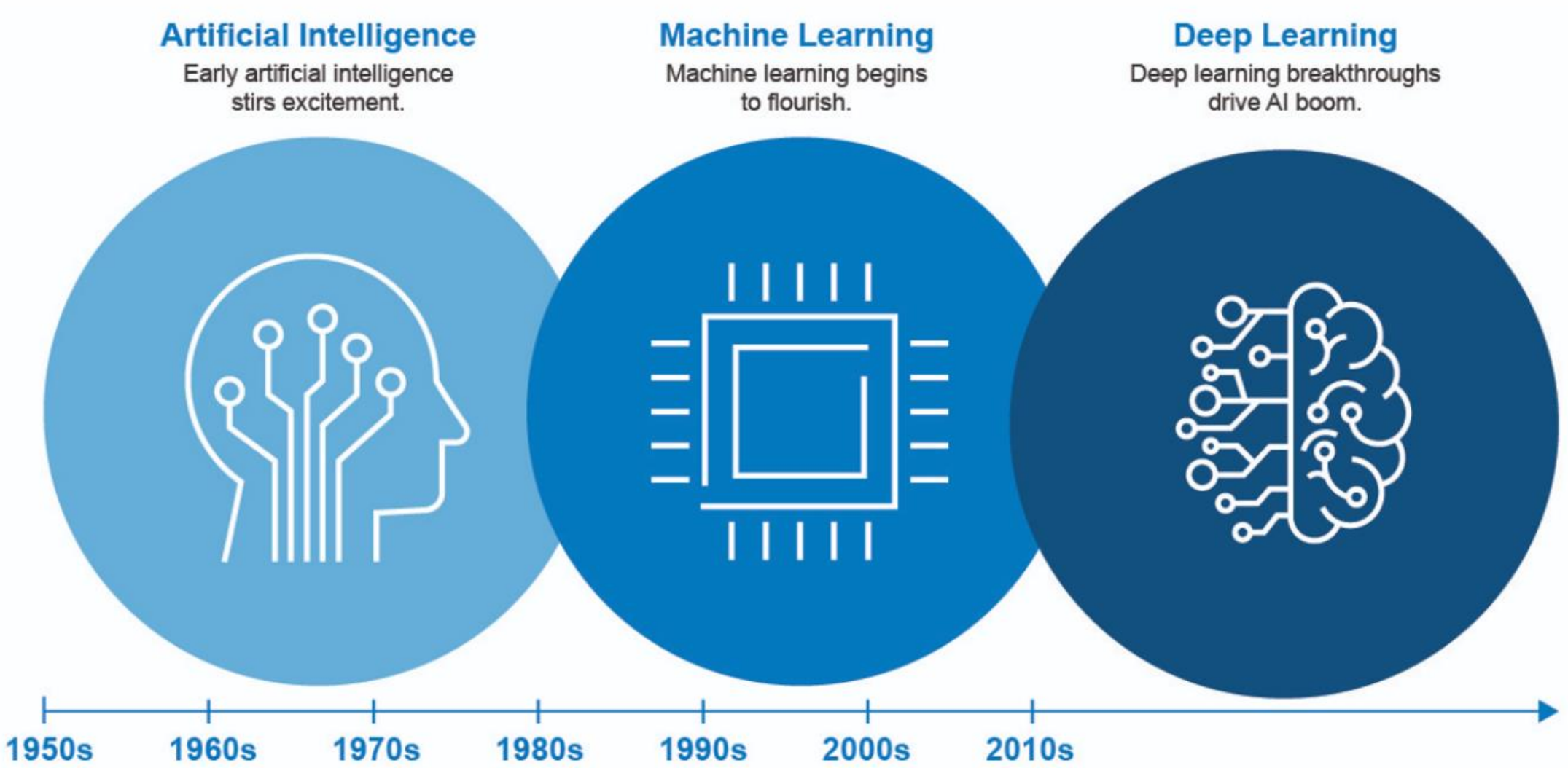
- Senior Advisor on Data Sciences
  Engage-AI

Machines **thinking** and **learning**




Algorithms that **learn** from data

A class of many-layered (**deep**) neural networks that **learn** from data


Artificial Intelligence
Early artificial intelligence stirs excitement.
Machine Learning
Machine learning begins to flourish.
Deep Learning
Deep learning breakthroughs drive AI boom.
1950s
1960s
1970s
1980s
1990s
2000s
2010s

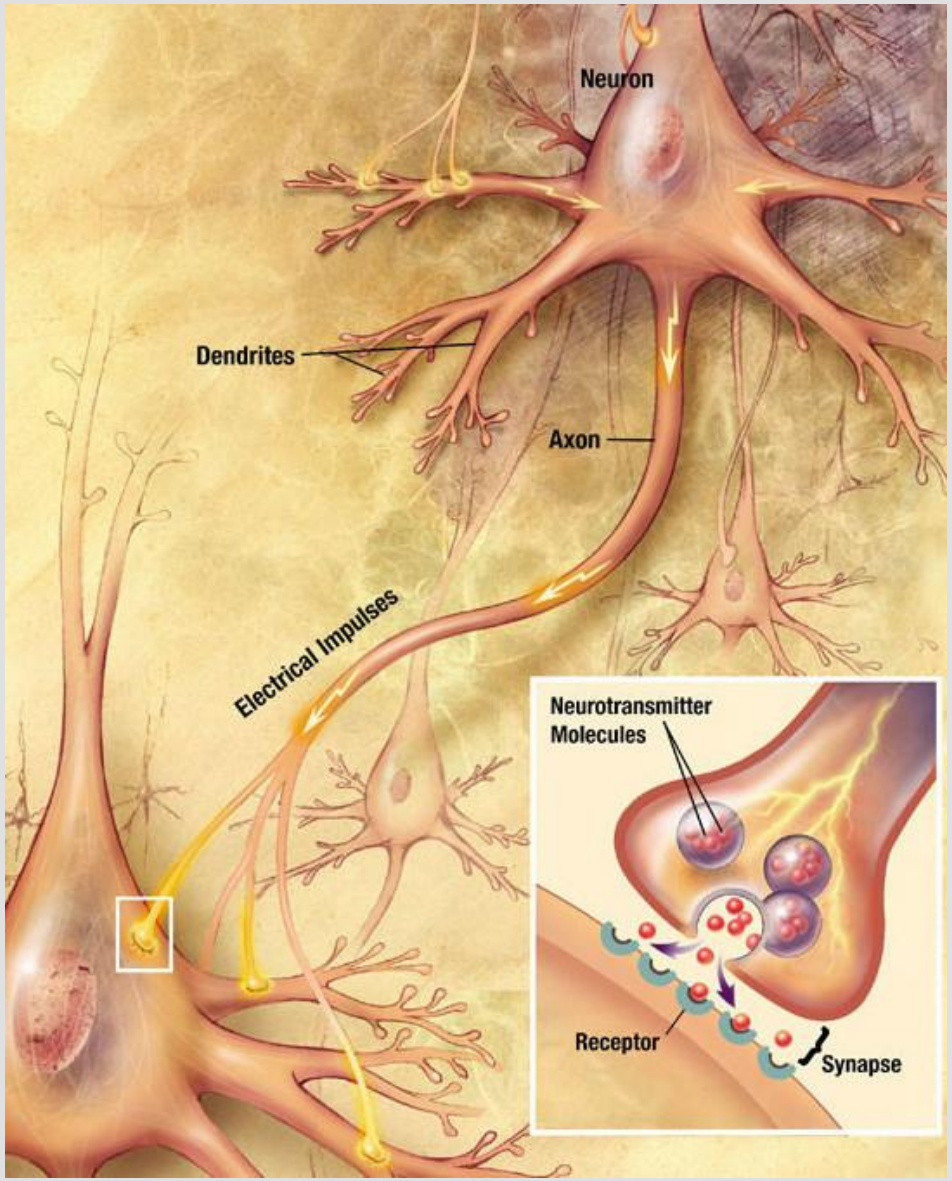
# Neurons

- Electrical impulses are received by neurons
- Neurons can:

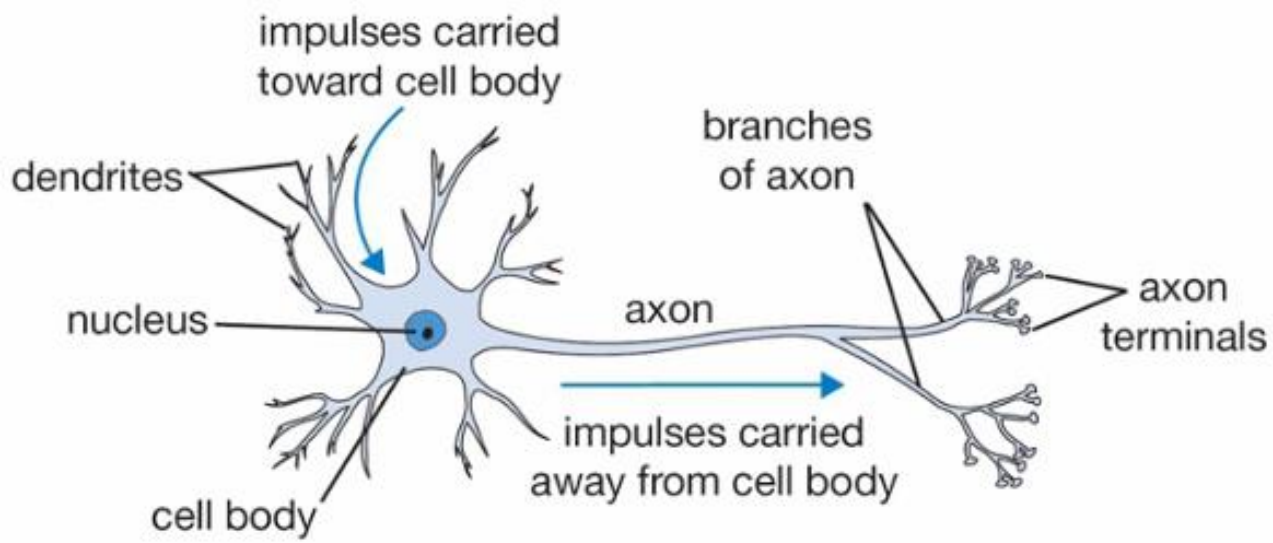
  excite (firing rate ↑)

  **or**

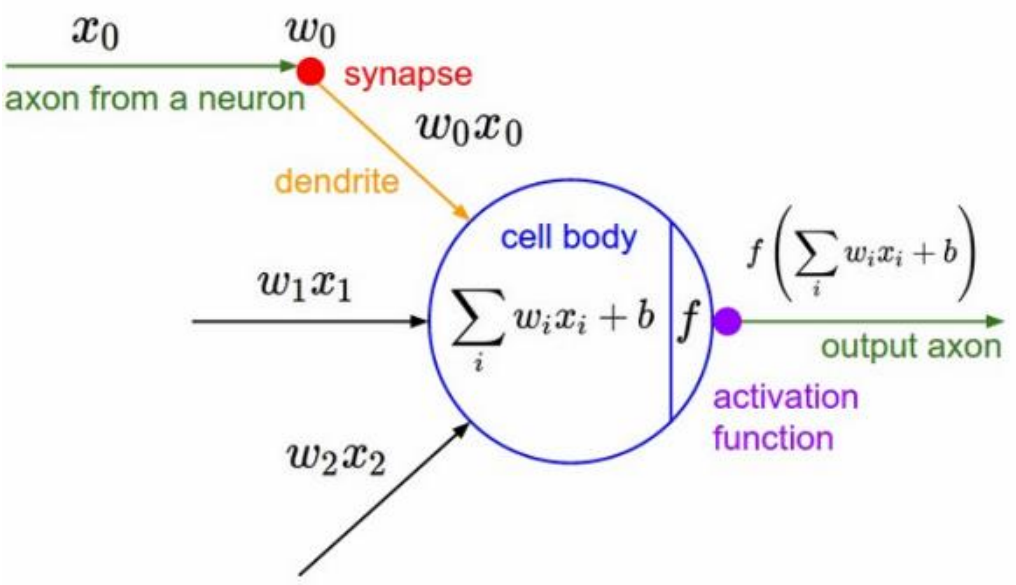
  inhibit (firing rate ↓)
- Signal passes to other neurons
- Signal is sent through the nervous system



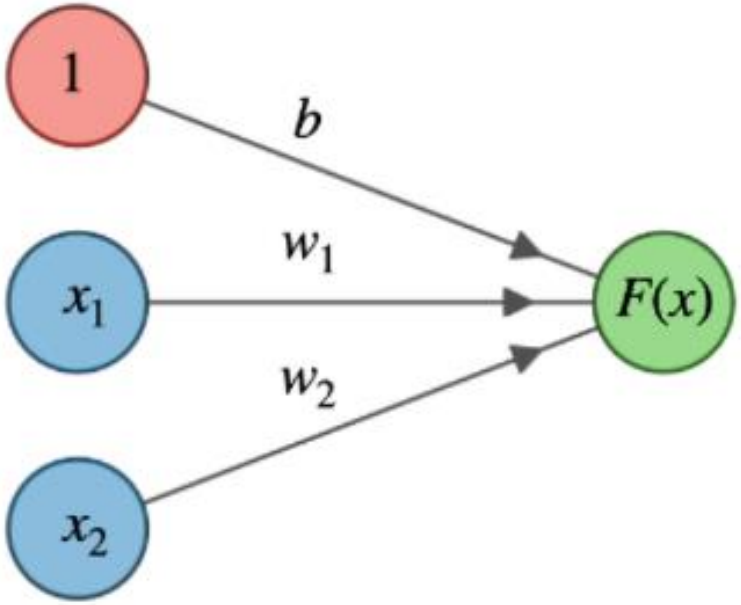
# Biological Neuron




# Artificial Neuron



# Linear Regression

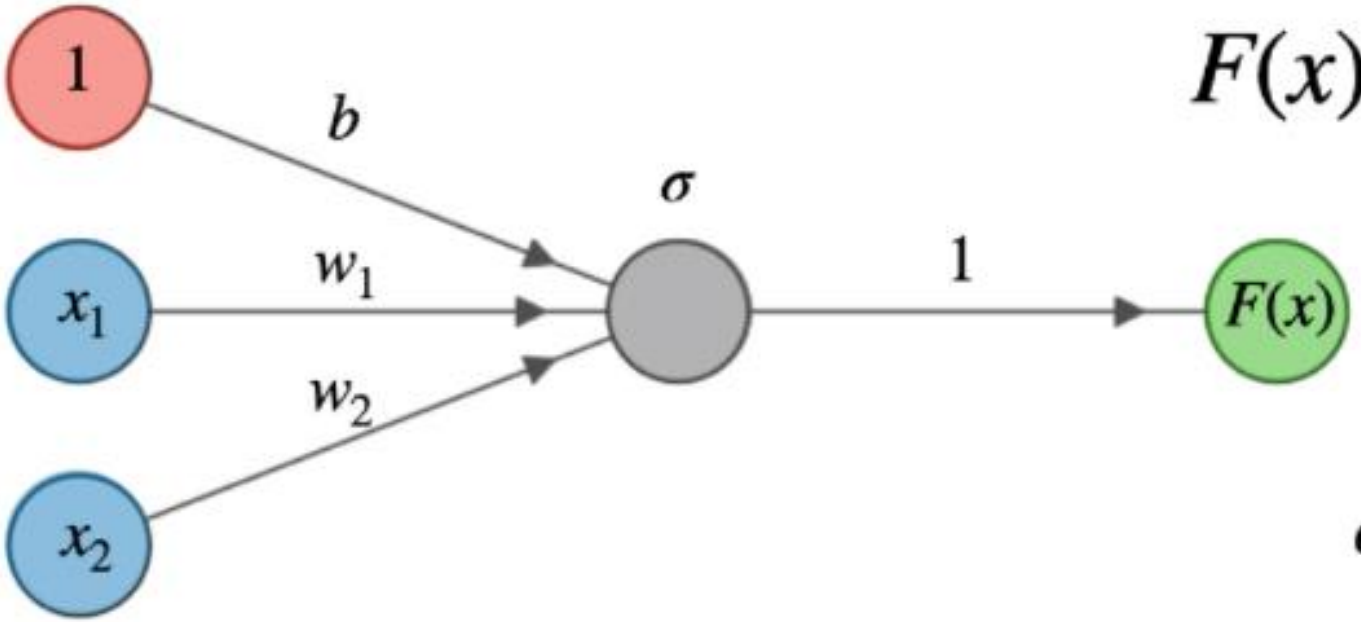



Input Layer $\in \mathbb{R}^3$ Output Layer $\in \mathbb{R}^1$

$$F(x) = w^T x + b$$

$$F(x) = w_1 \cdot x_1 + w_2 \cdot x_2 + 1 \cdot b$$

# Logistic Regression




$$F(x) = \sigma(w^T x + b)$$

$$\sigma(h) = \frac{1}{1 + e^{-h}}$$

Input Layer $\in \mathbb{R}^3$ Hidden Layer $\in \mathbb{R}^1$ Output Layer $\in \mathbb{R}^1$

# Breast Cancer Classification

- Dataset: 10 features of breast cells, including
  - Radius
  - Texture
  - Perimeter
  - Area
  - Smoothness
  - …
  - Labels (malignant or benign)

- Logistic Regression is 97% accurate at **classifying** cancer cells

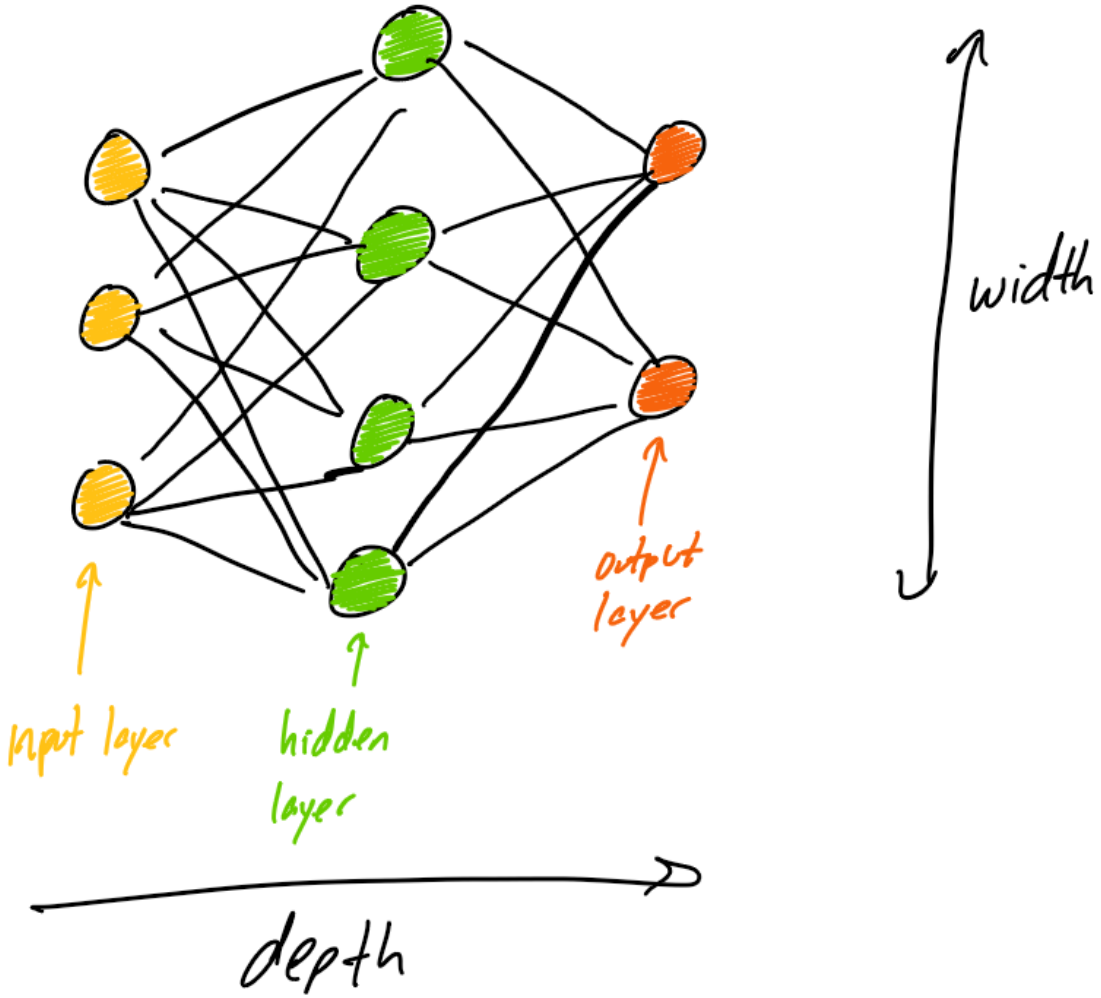
Training Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.99 | 0.97 | 0.98 | 168 |
| 1 | 0.98 | 0.99 | 0.99 | 258 |
| accuracy | | | 0.98 | 426 |
| macro avg | 0.98 | 0.98 | 0.98 | 426 |
| weighted avg | 0.98 | 0.98 | 0.98 | 426 |

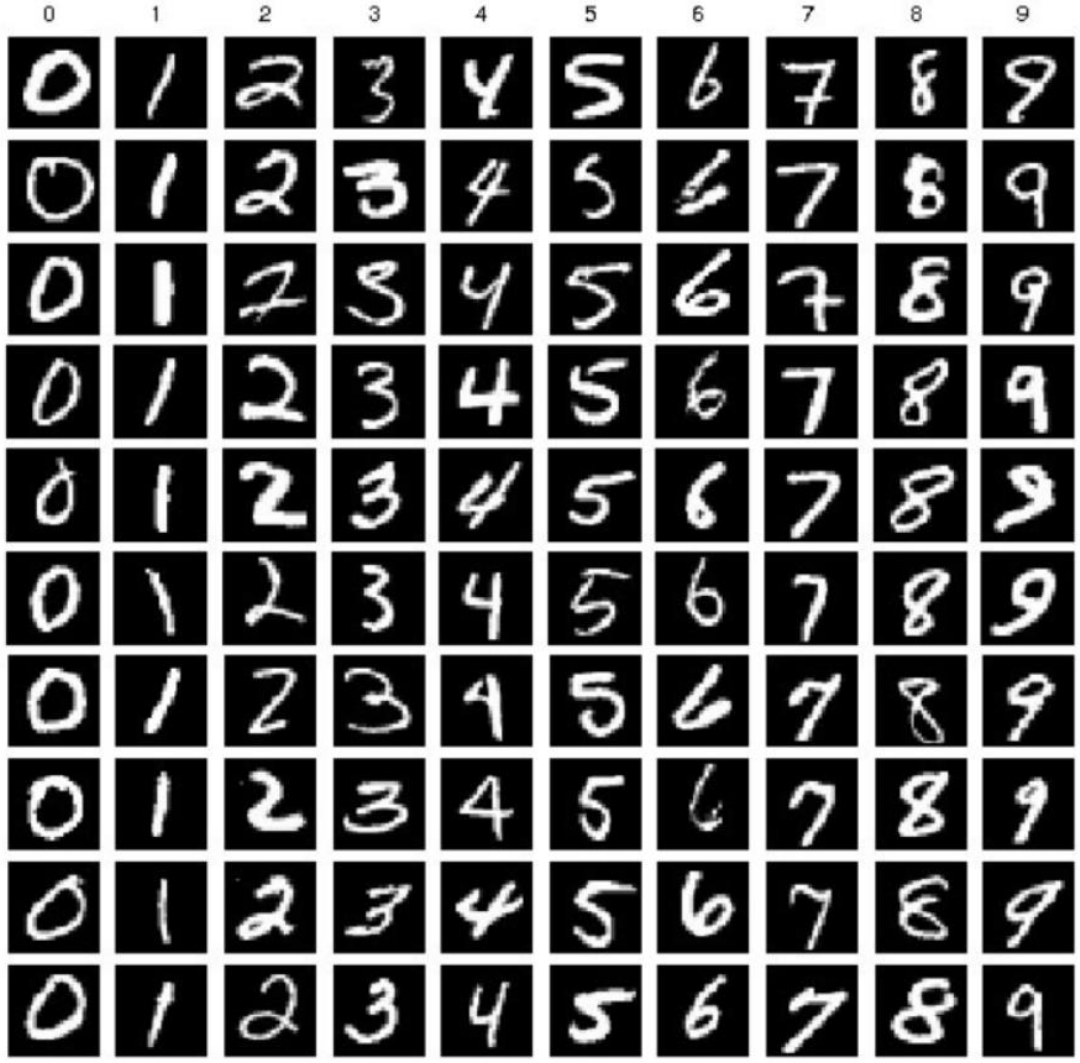
Testing Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.95 | 0.93 | 0.94 | 44 |
| 1 | 0.97 | 0.98 | 0.97 | 99 |
| accuracy | | | 0.97 | 143 |
| macro avg | 0.96 | 0.96 | 0.96 | 143 |
| weighted avg | 0.96 | 0.97 | 0.96 | 143 |

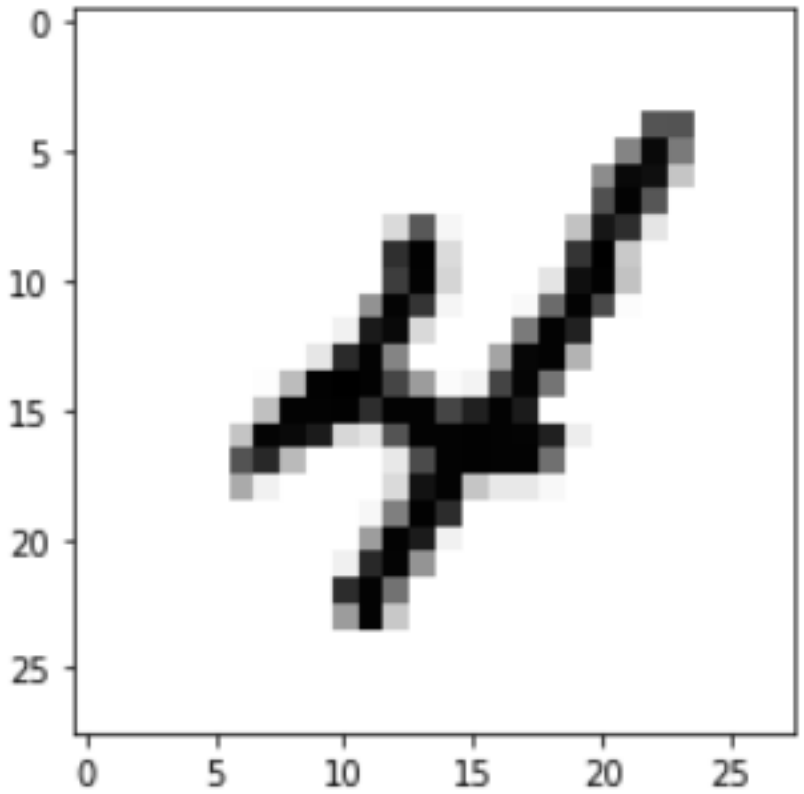
# Neural Networks



# Handwritten Digit Classification



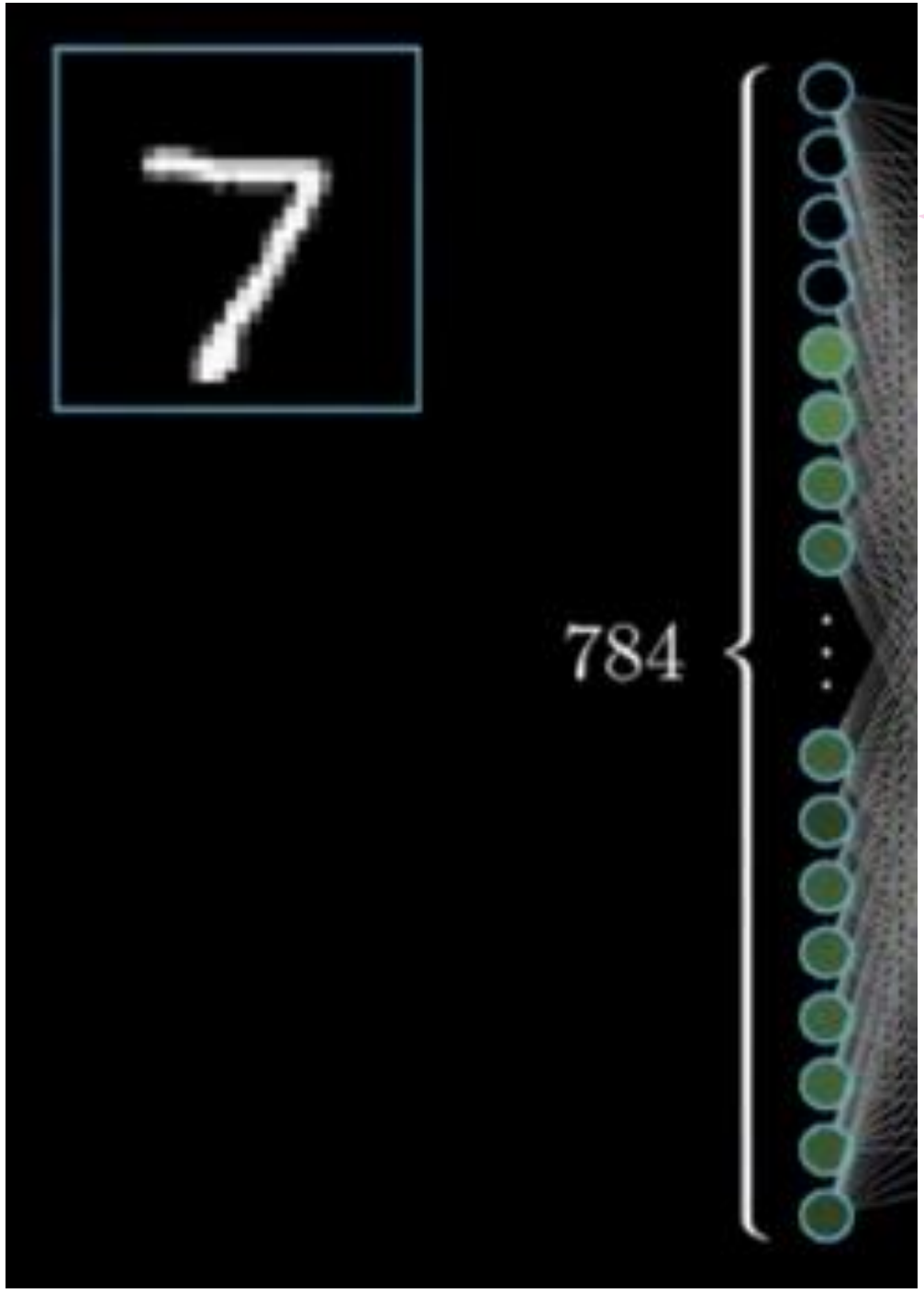
# Computer Vision




What you see

```
[[  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 189 190   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 143 247 153   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 136 247 242  86   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 192 252 187   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0  62 185  18   0   0   0   0  89 236 217  47   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0 216 253  60   0   0   0   0 212 255  81   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0 206 252  68   0   0   0  48 242 253  89   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0 131 251 212  21   0   0  11 167 252 197   5   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0  29 232 247  63   0   0   0 153 252 226   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0  45 219 252 143   0   0   0 116 249 252 103   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   4  96 253 255 253 200 122   7  25 201 250 158   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0  92 252 252 253 217 252 252 200 227 252 231   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0  87 251 247 231  65  48 189 252 252 253 252 251 227  35   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0 190 221  98   0   0   0  42 196 252 253 252 252 162   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0 111  29   0   0   0   0  62 239 252  86  42  42  14   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0  15 148 253 218   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0 121 252 231  28   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0  31 221 251 129   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0 218 252 160   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0 122 252  82   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0]]
```

What a computer sees





# A Neural Network

- 784 inputs
  - one for each pixel
- 784+1 weights
- 10 neurons
  - one for each digit

# Logistic Regression's Performance

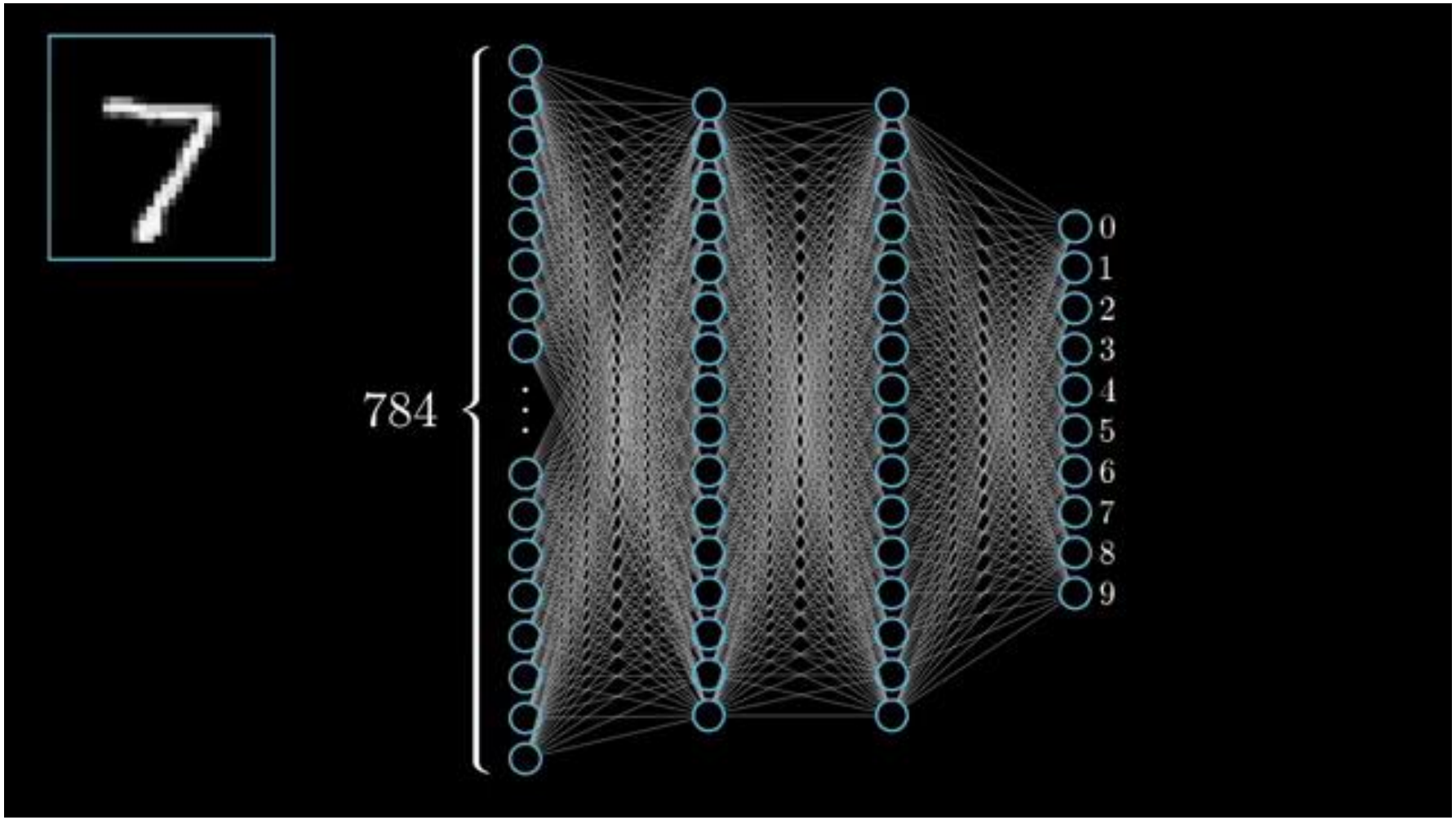
Training Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.97 | 0.98 | 0.98 | 5923 |
| 1 | 0.96 | 0.98 | 0.97 | 6742 |
| 2 | 0.94 | 0.92 | 0.93 | 5958 |
| 3 | 0.92 | 0.91 | 0.92 | 6131 |
| 4 | 0.94 | 0.95 | 0.94 | 5842 |
| 5 | 0.91 | 0.91 | 0.91 | 5421 |
| 6 | 0.96 | 0.97 | 0.96 | 5918 |
| 7 | 0.95 | 0.95 | 0.95 | 6265 |
| 8 | 0.91 | 0.90 | 0.91 | 5851 |
| 9 | 0.92 | 0.92 | 0.92 | 5949 |
| accuracy | | | 0.94 | 60000 |
| macro avg | 0.94 | 0.94 | 0.94 | 60000 |
| weighted avg | 0.94 | 0.94 | 0.94 | 60000 |

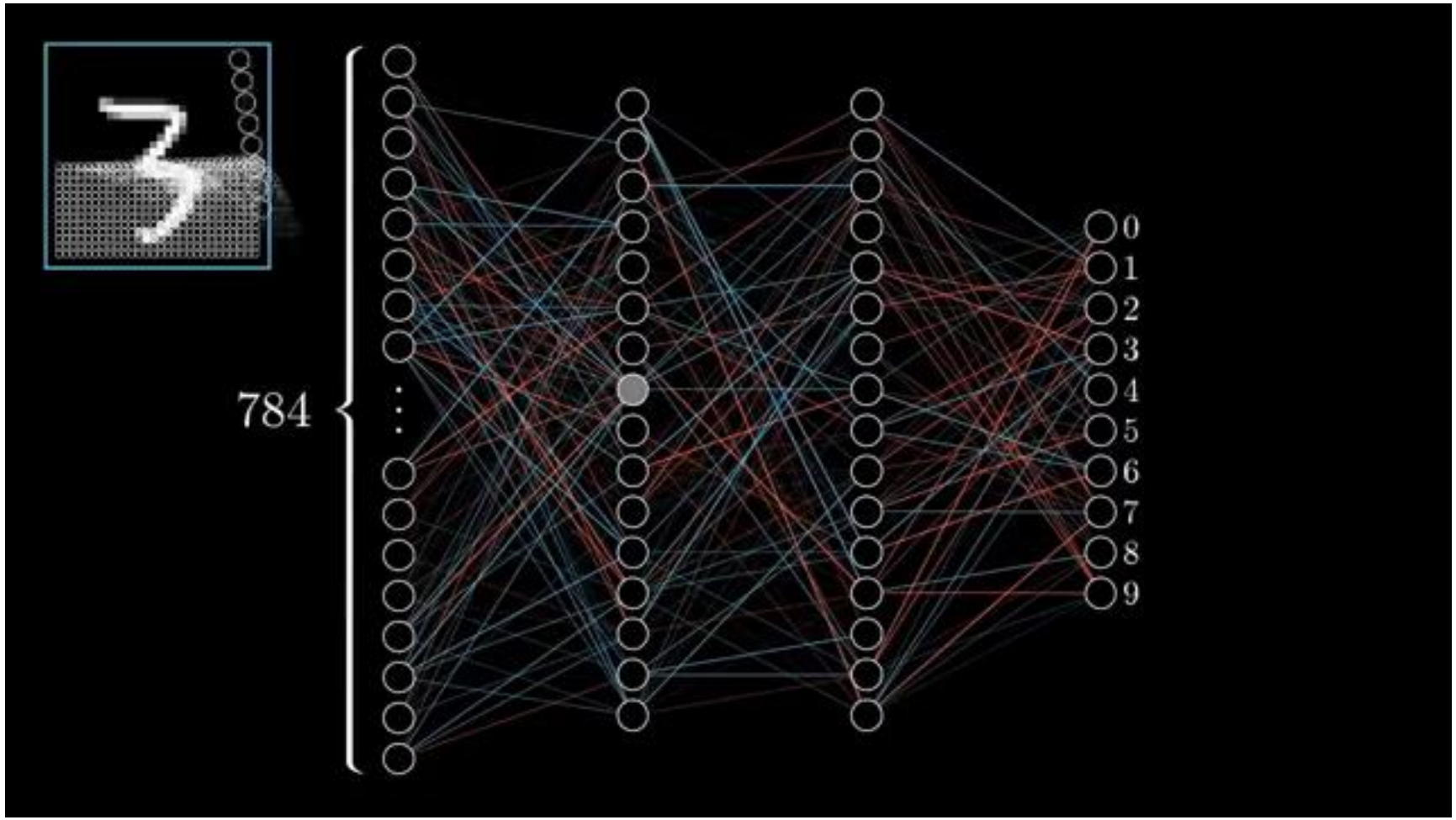
Testing Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.95 | 0.97 | 0.96 | 980 |
| 1 | 0.96 | 0.98 | 0.97 | 1135 |
| 2 | 0.93 | 0.90 | 0.91 | 1032 |
| 3 | 0.90 | 0.92 | 0.91 | 1010 |
| 4 | 0.94 | 0.94 | 0.94 | 982 |
| 5 | 0.90 | 0.87 | 0.88 | 892 |
| 6 | 0.94 | 0.95 | 0.95 | 958 |
| 7 | 0.93 | 0.92 | 0.93 | 1028 |
| 8 | 0.88 | 0.88 | 0.88 | 974 |
| 9 | 0.91 | 0.92 | 0.91 | 1009 |
| accuracy | | | 0.93 | 10000 |
| macro avg | 0.92 | 0.92 | 0.92 | 10000 |

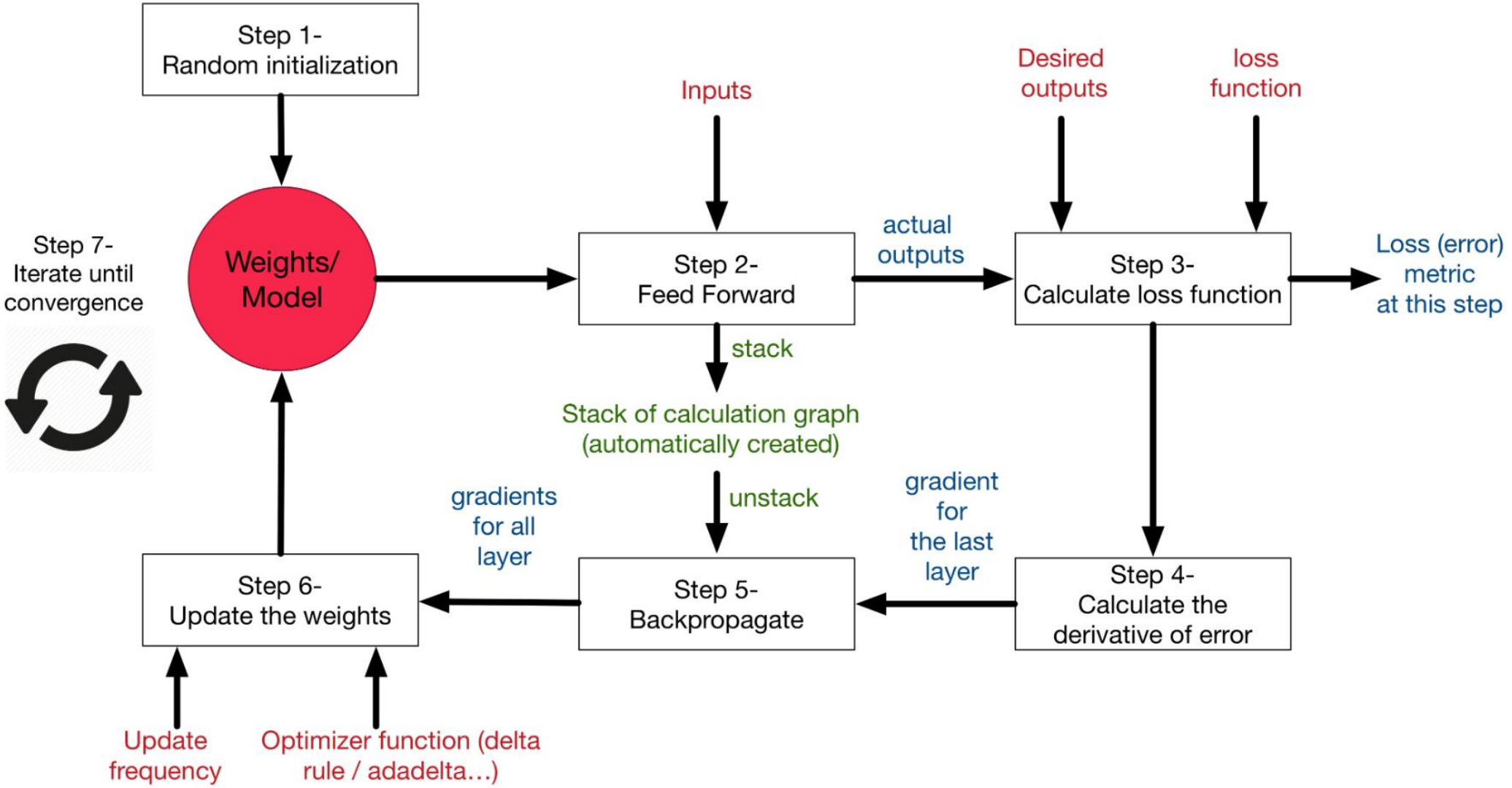
# Feeding Data Forward



# Cost (or Loss)



# Training Neural Networks

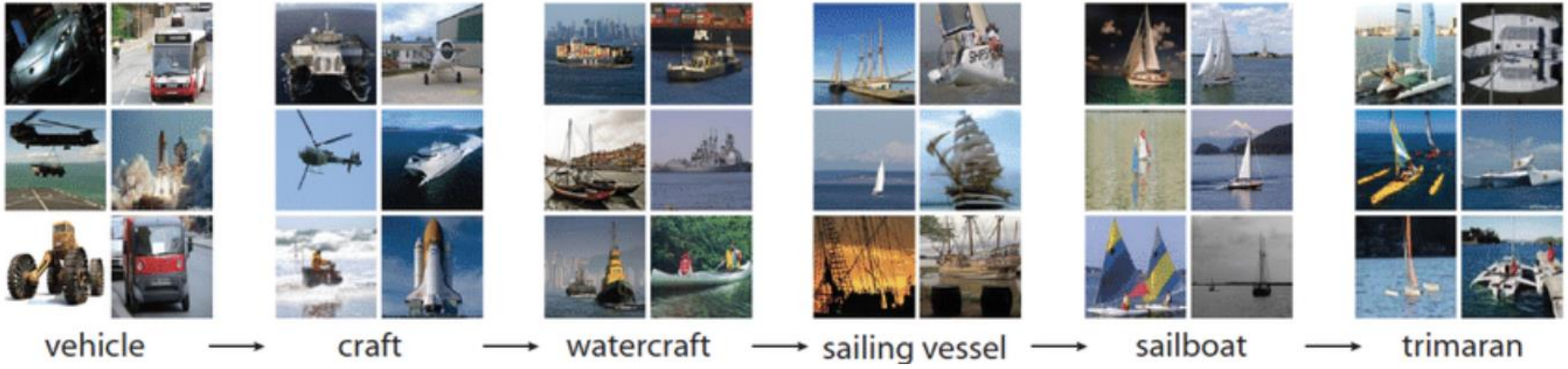


# IMAGENET Large Scale Visual Recognition Challenge

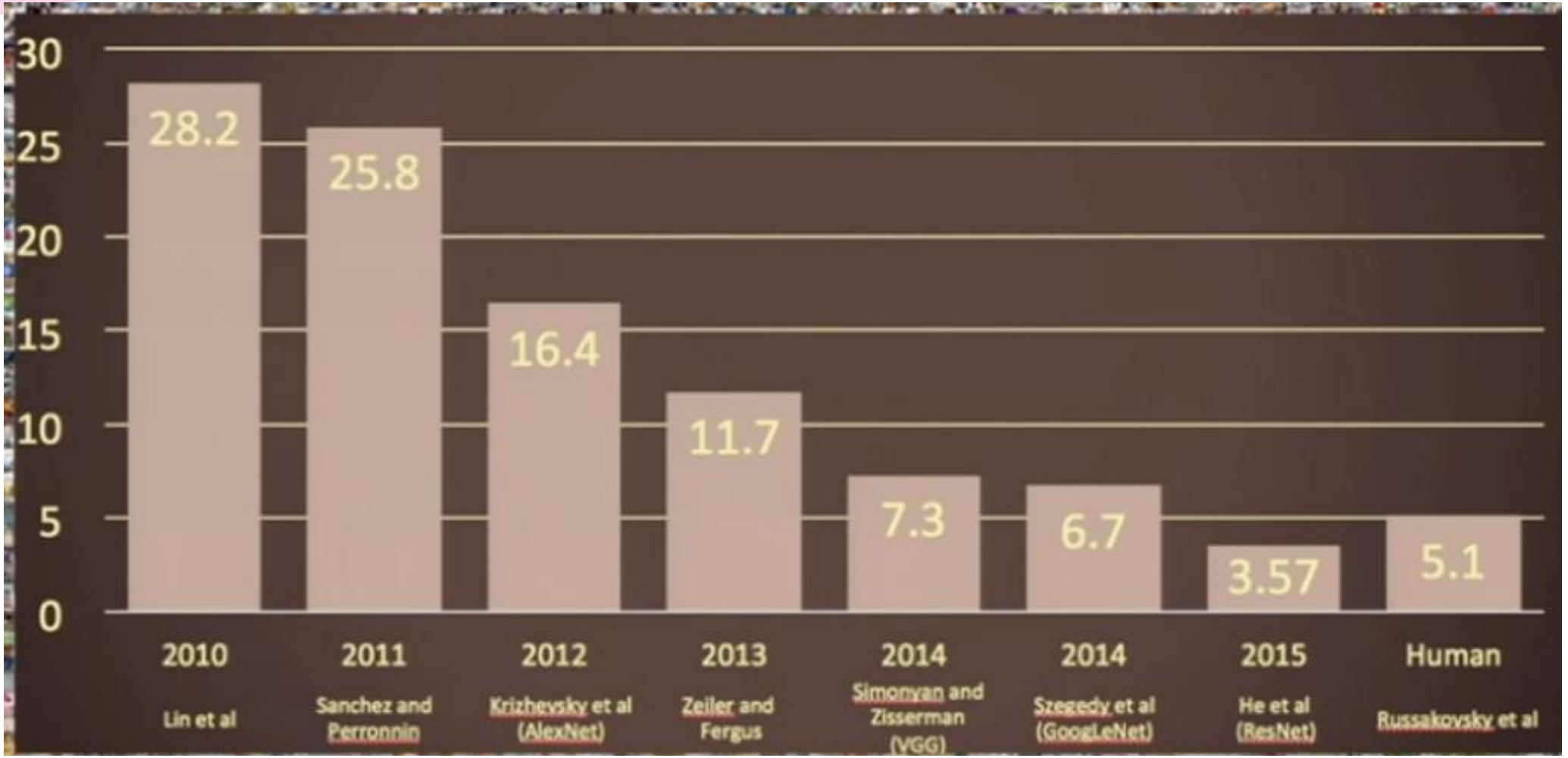
The Image Classification Challenge:

1,000 object classes

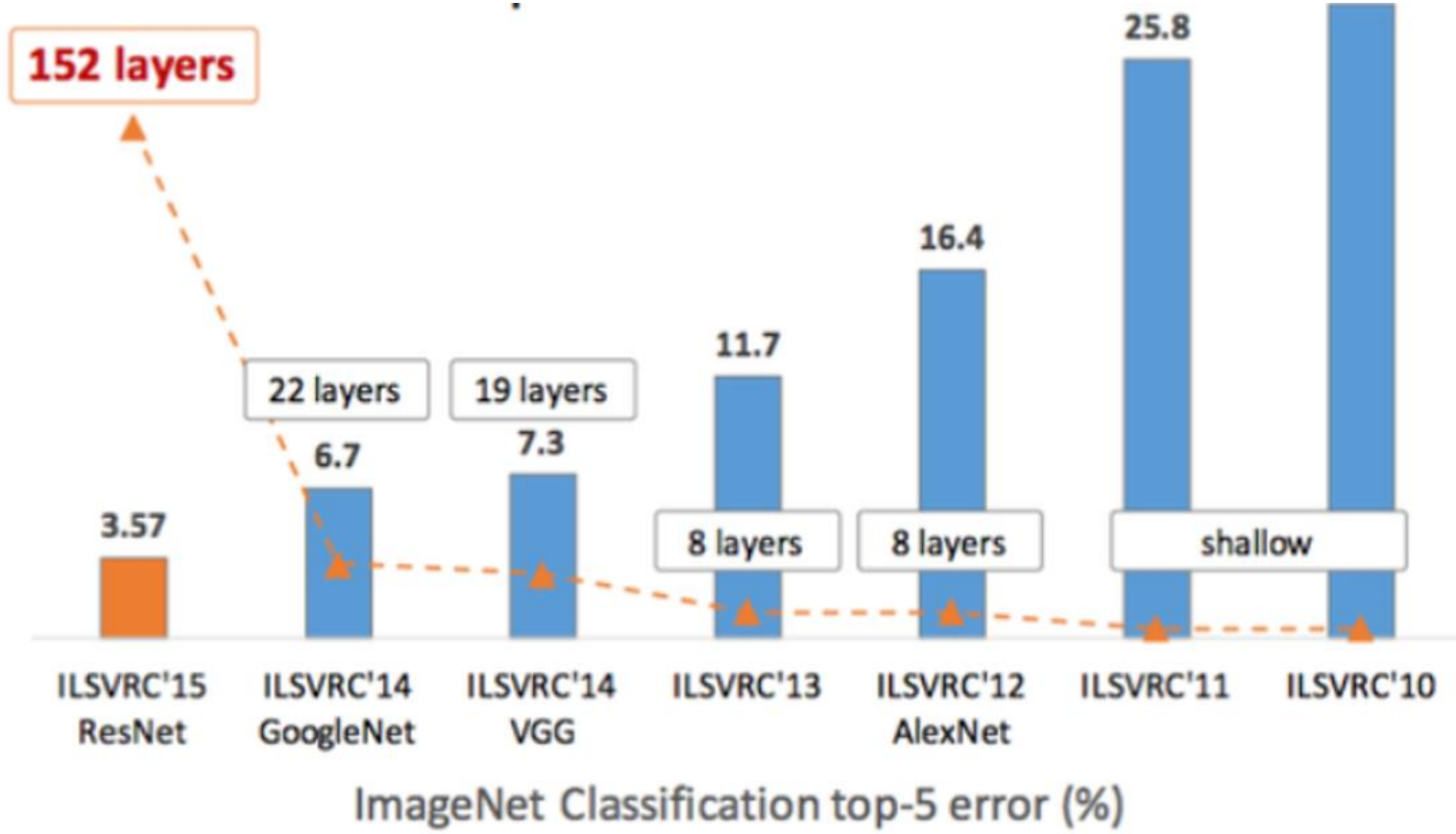
1,431,167 images

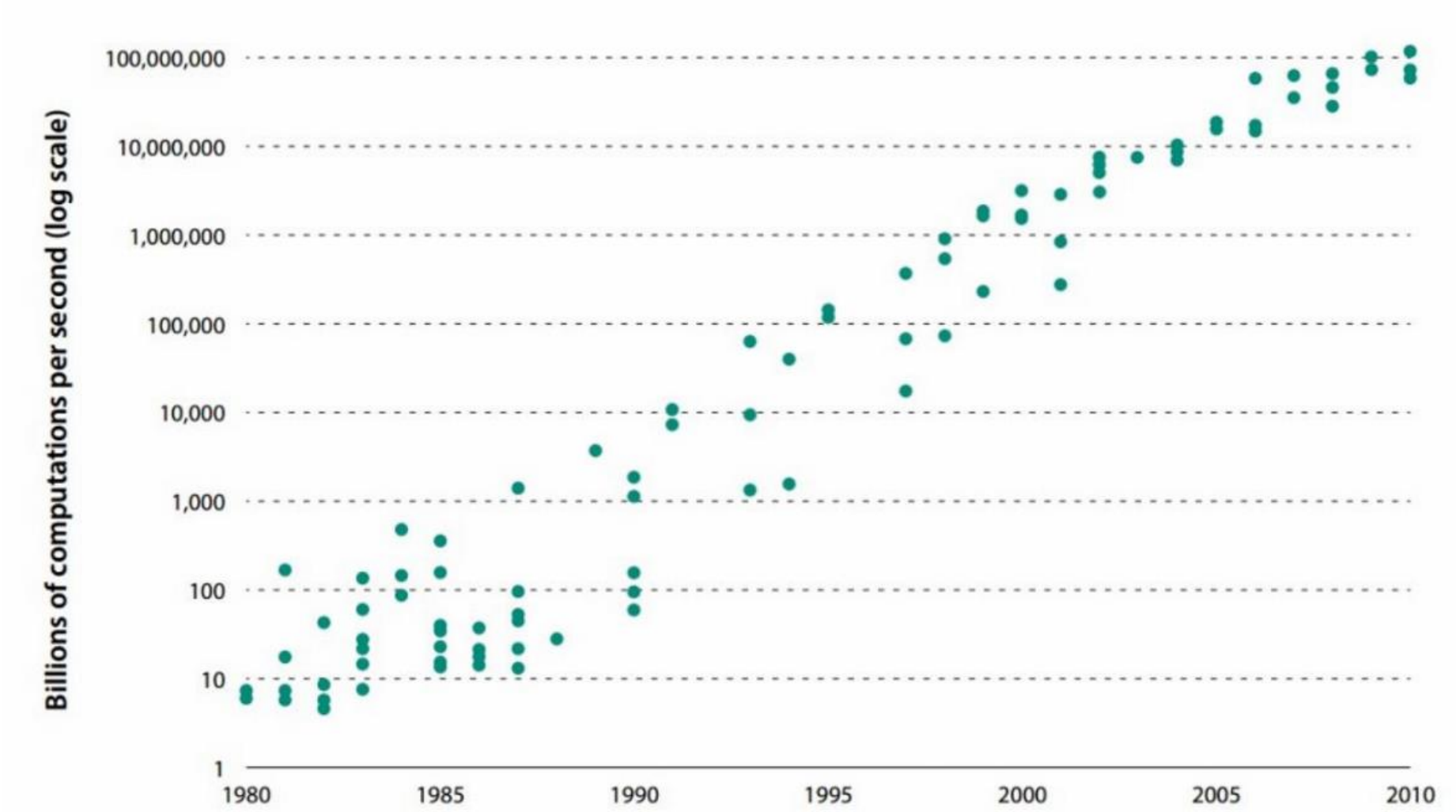
# ImageNet


30
25
20
15
10
5
0
28.2
25.8
16.4
11.7
7.3
6.7
3.57
5.1
2010
Lin et al
2011
Sanchez and Perronnin
2012
Krizhevsky et al (AlexNet)
2013
Zeiler and Fergus
2014
Simonyan and Zisserman (VGG)
2014
Szegedy et al (GoogLeNet)
2015
He et al (ResNet)
Human
Russakovsky et al

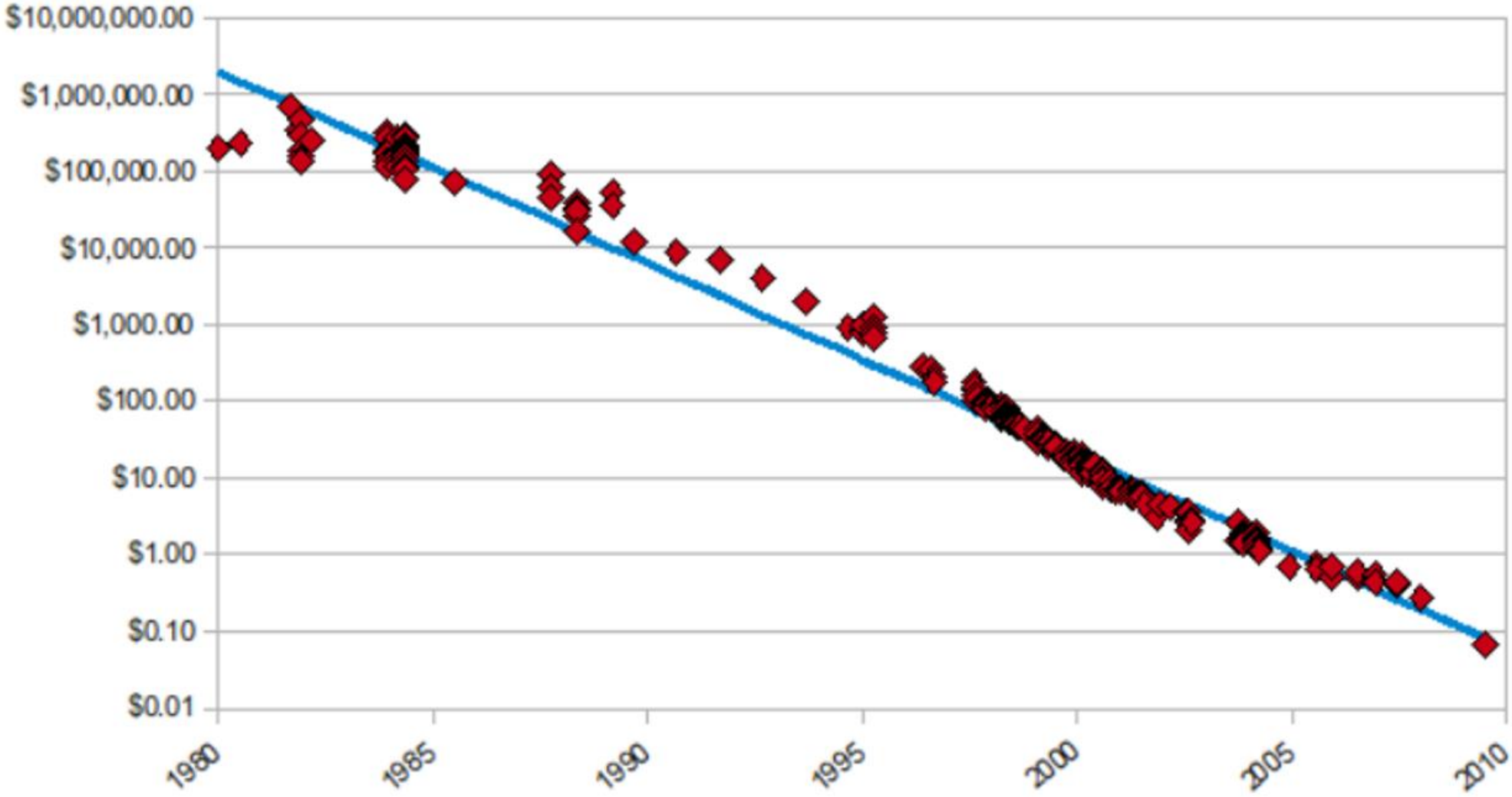
# **Deep** Learning



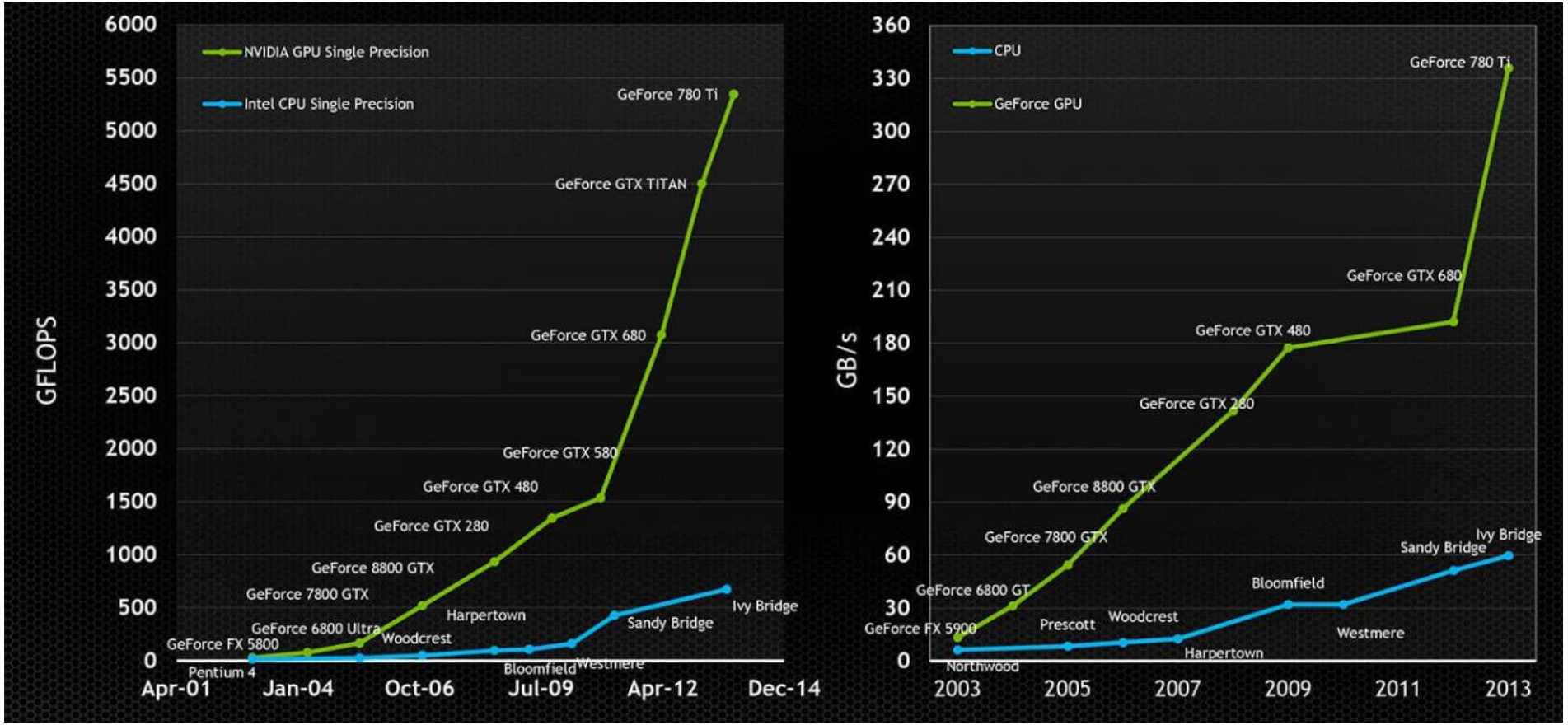
# $1 of Computing Power



# $ per GB of Storage

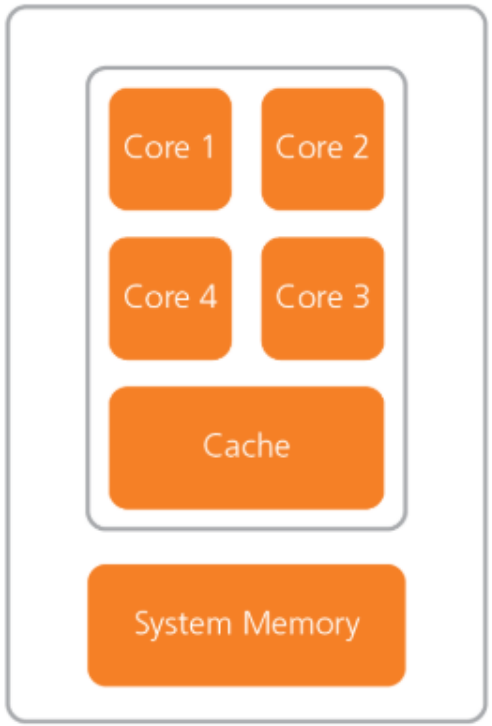


# The Rise of GPU Computing

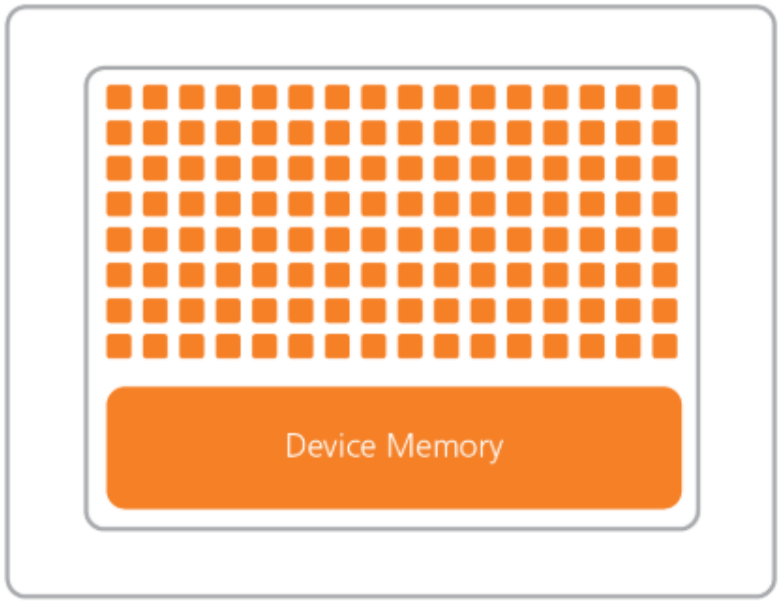


# Why GPUs?

CPU (Multiple Cores)

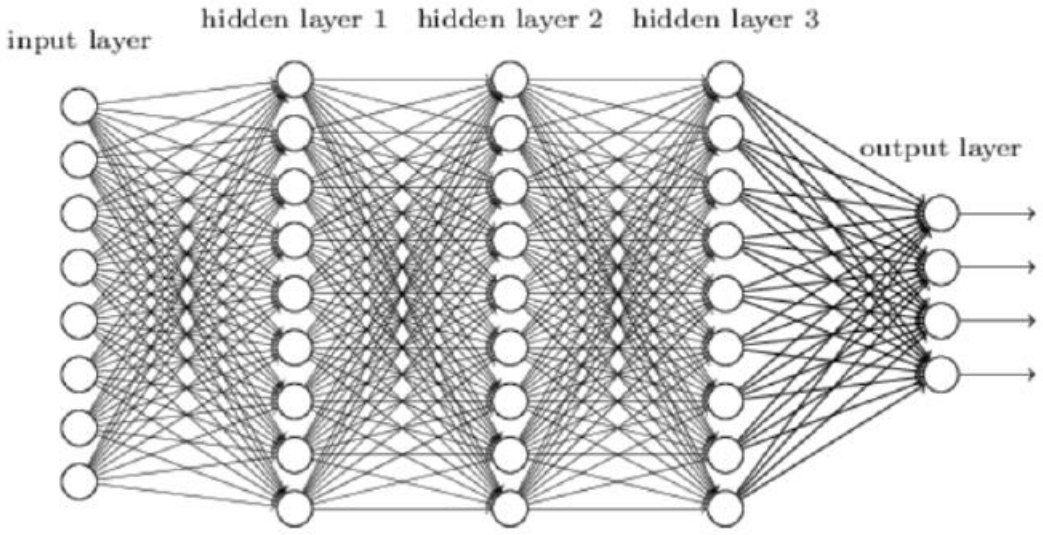


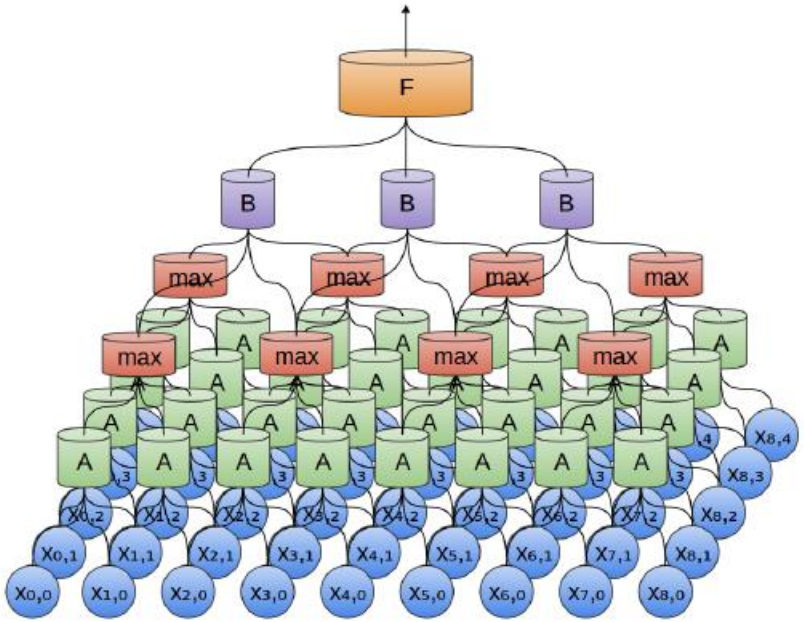

GPU (Hundreds of Cores)




NVIDIA
CUDA

# Innovative Neural Architectures



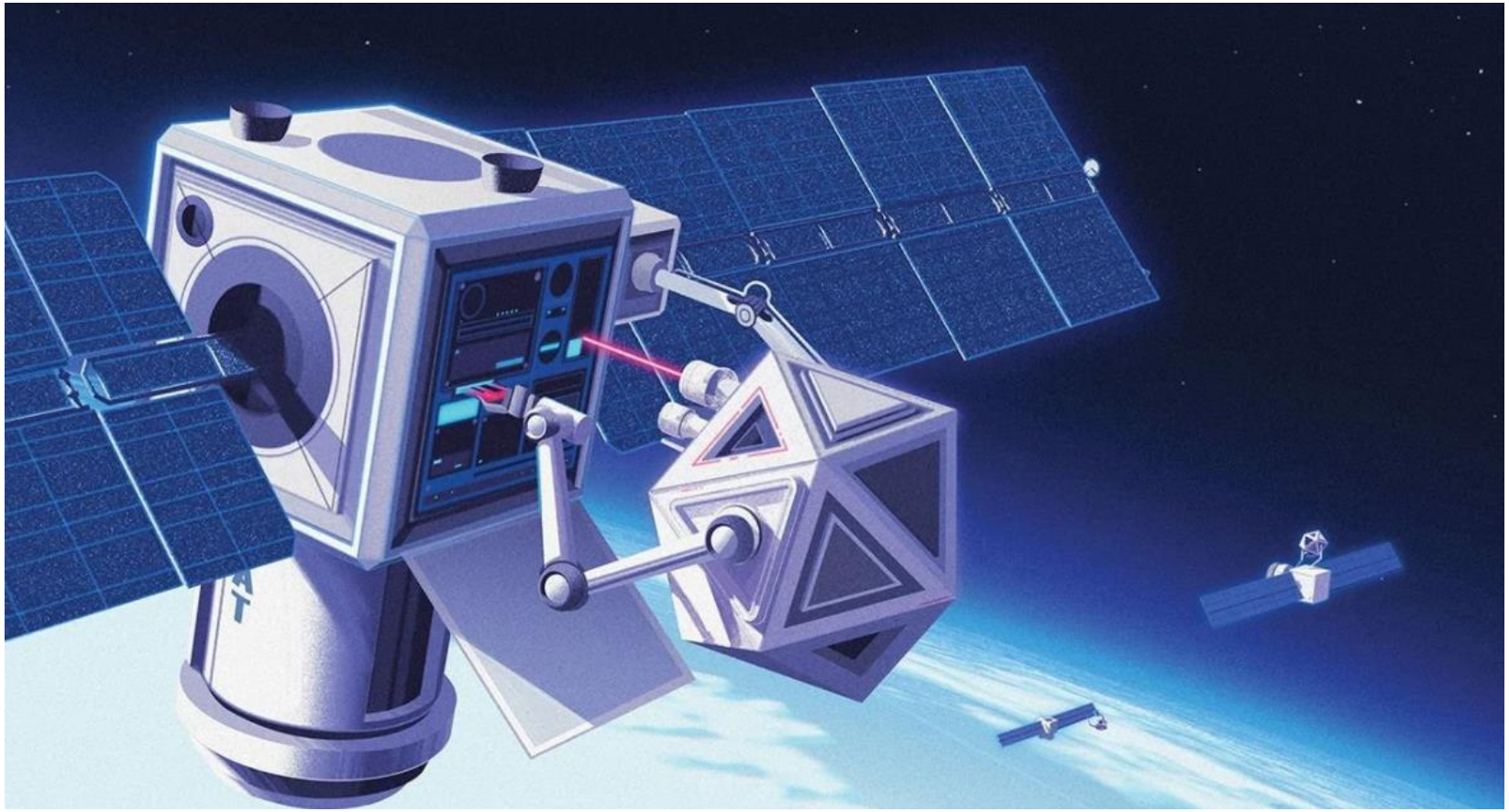



# Dysfunctional and Obsolete Satellites

*There is no other area of human activity where we build something that's worth a half-billion dollars or a billion dollars, and never look at it again, never fix it, and never upgrade it*

--Gordon Roesler (DARPA)

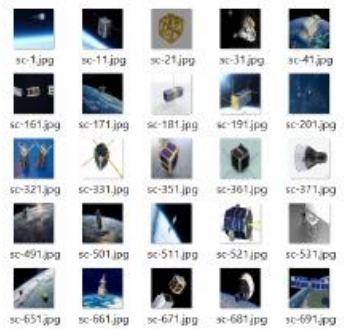
# Current Project: On-Orbit Repair



Supported by U.S. Space Force through the Air Force Research Laboratory

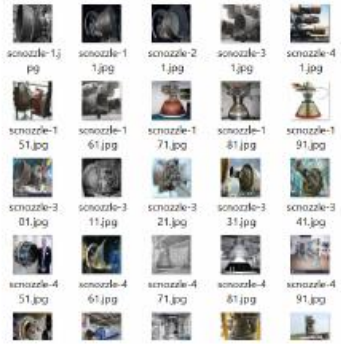
# Image Classification

(a) Spacecraft



(b) Meteoroid

(c) Nozzle



(d) Fuel Tank

(e) Solar Panel

# VGG-Net

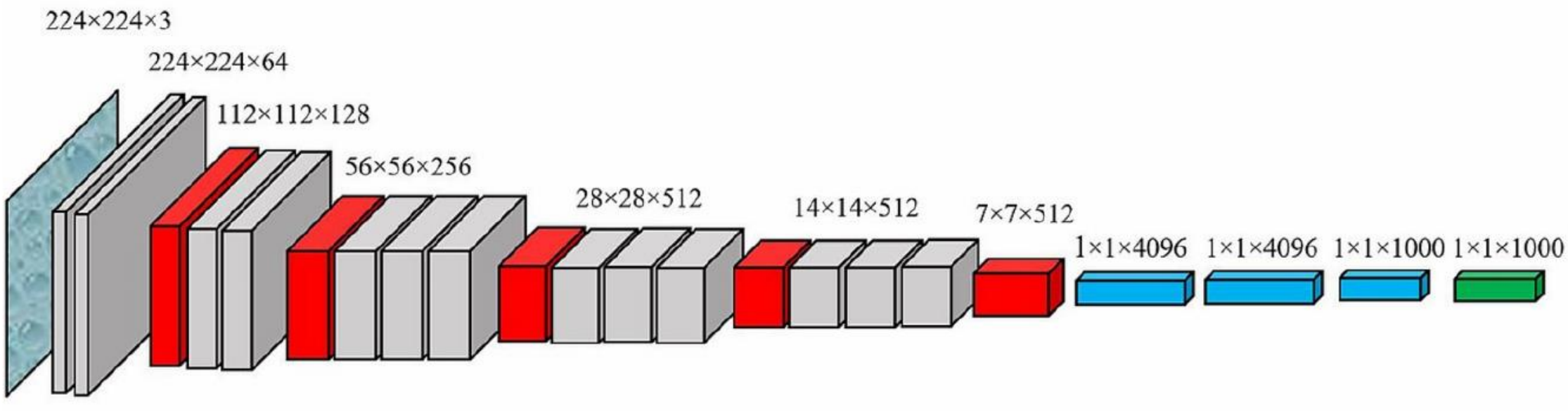
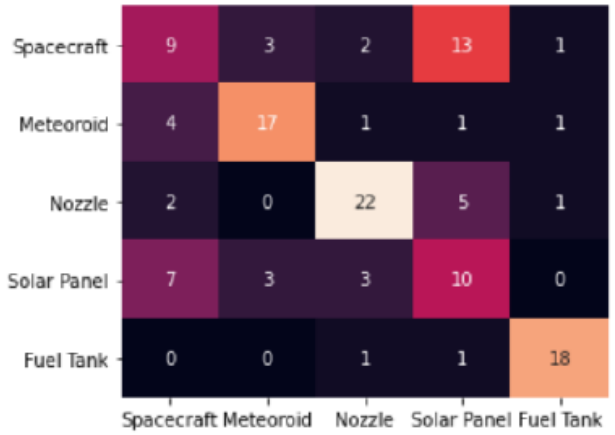



Figure: The VGG-16 Architecture with convolutions, max-pooling, fully-connected, and softmax layers

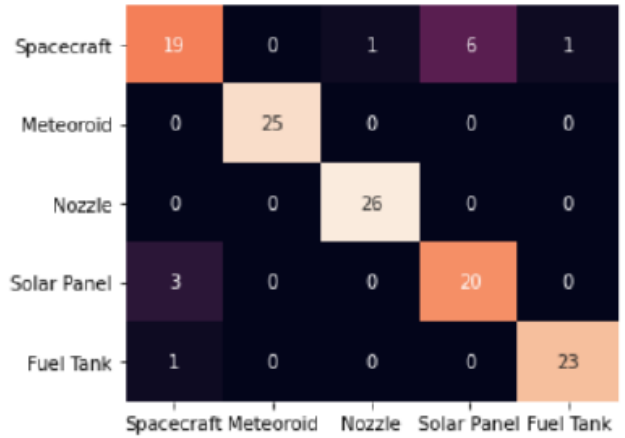
138m weights!

# Performance



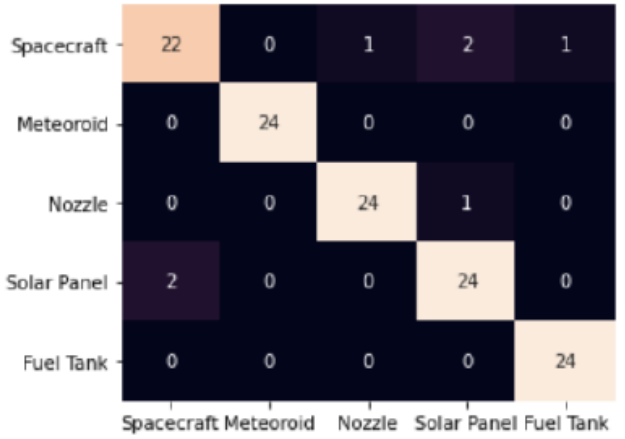

(a) NASNet Large

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Spacecraft | 0.41 | 0.32 | 0.36 | 28 |
| Meteoroid | 0.74 | 0.71 | 0.72 | 24 |
| Nozzle | 0.76 | 0.73 | 0.75 | 30 |
| Solar Panel | 0.33 | 0.43 | 0.38 | 23 |
| Fuel Tank | 0.86 | 0.90 | 0.88 | 20 |
| accuracy | | | 0.61 | 125 |
| macro avg | 0.62 | 0.62 | 0.62 | 125 |
| weighted avg | 0.61 | 0.61 | 0.61 | 125 |




(b) VGG-16

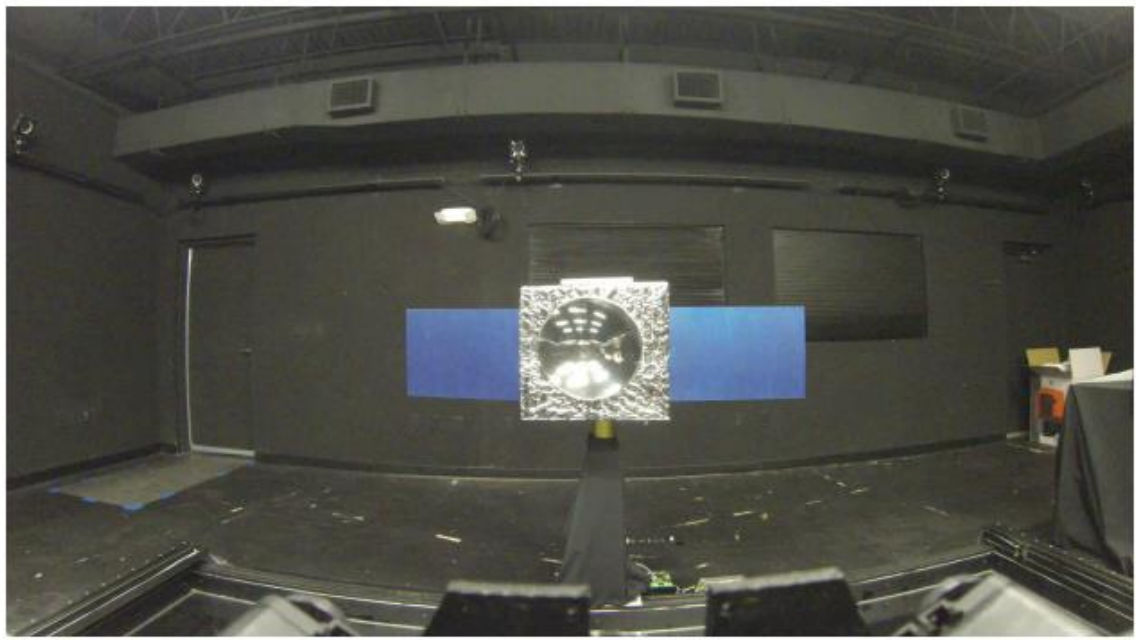
| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Spacecraft | 0.83 | 0.70 | 0.76 | 27 |
| Meteoroid | 1.00 | 1.00 | 1.00 | 25 |
| Nozzle | 0.96 | 1.00 | 0.98 | 26 |
| Solar Panel | 0.77 | 0.87 | 0.82 | 23 |
| Fuel Tank | 0.96 | 0.96 | 0.96 | 24 |
| accuracy | | | 0.90 | 125 |
| macro avg | 0.90 | 0.91 | 0.90 | 125 |
| weighted avg | 0.90 | 0.90 | 0.90 | 125 |



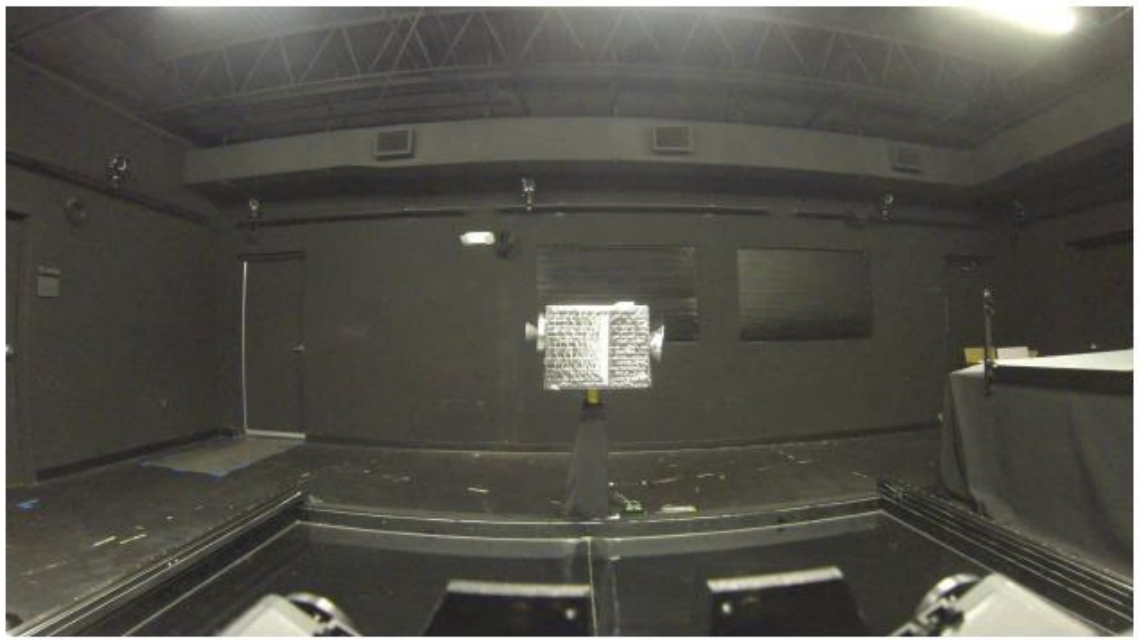

(c) VGG-19

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Spacecraft | 0.92 | 0.85 | 0.88 | 26 |
| Meteoroid | 1.00 | 1.00 | 1.00 | 24 |
| Nozzle | 0.96 | 0.96 | 0.96 | 25 |
| Solar Panel | 0.89 | 0.92 | 0.91 | 26 |
| Fuel Tank | 0.96 | 1.00 | 0.98 | 24 |
| accuracy | | | 0.94 | 125 |
| macro avg | 0.95 | 0.95 | 0.95 | 125 |
| weighted avg | 0.94 | 0.94 | 0.94 | 125 |

1. **Trim** the videos and **extract** still frames
2. Shape frames to $224 \times 224 \times 3$ and standardize in each color channel
3. Select some frames with only **1 class displayed** (experiment: solar or nozzle)



(a) **only** solar panel visible



(b) **only** nozzle visible

# A Test...

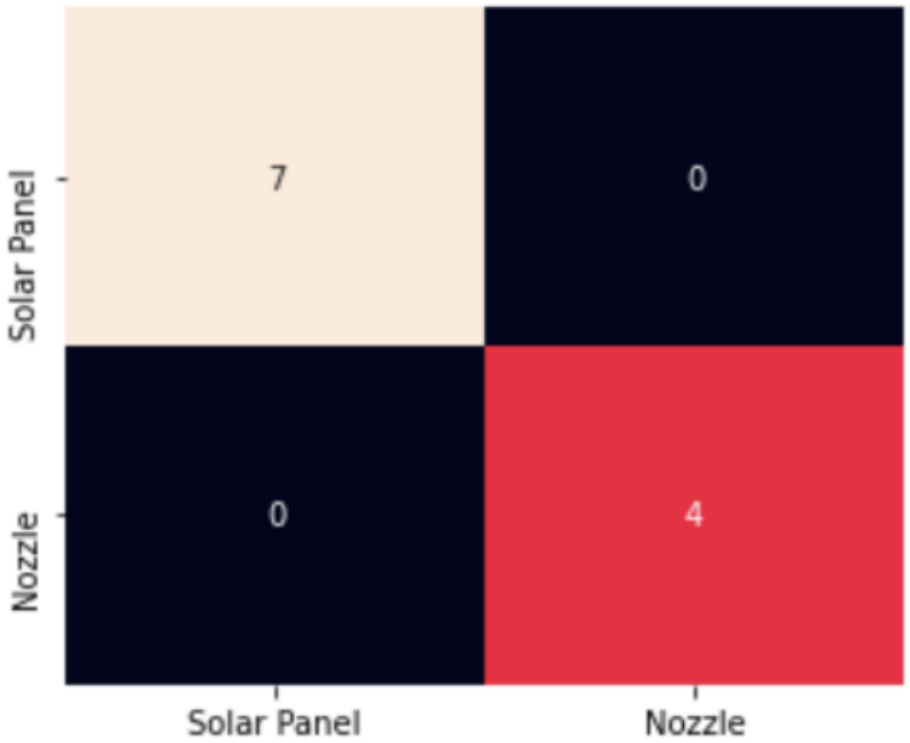



Figure: A VGG-19 experiment on a small set of data is perfect

# A Harder Problem…

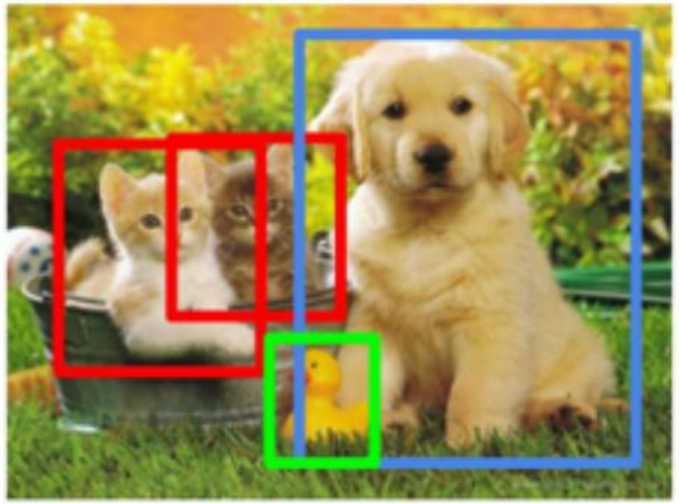
**Classification**

CAT

**Classification + Localization**

CAT

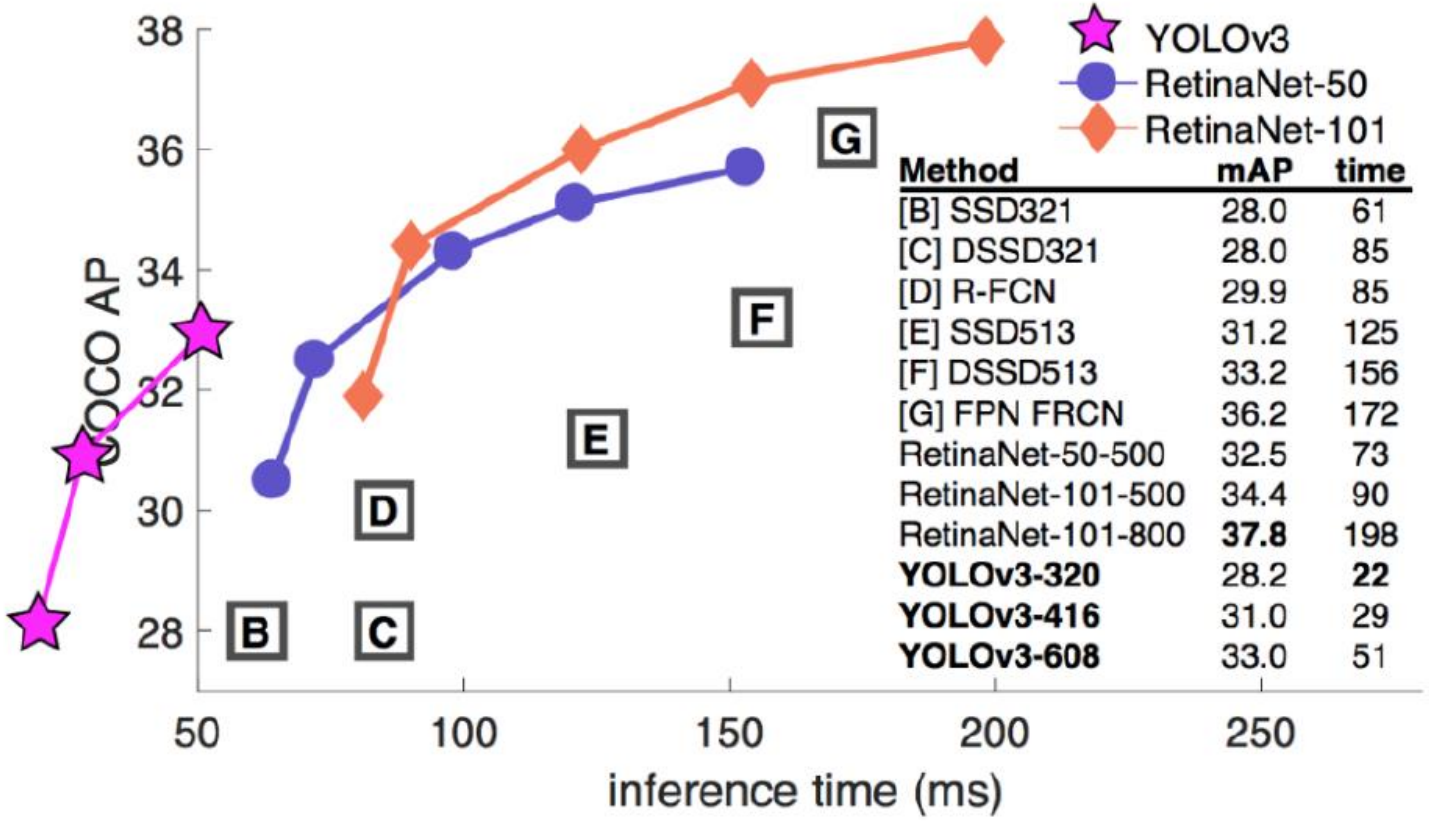
**Object Detection**



CAT, DOG, DUCK

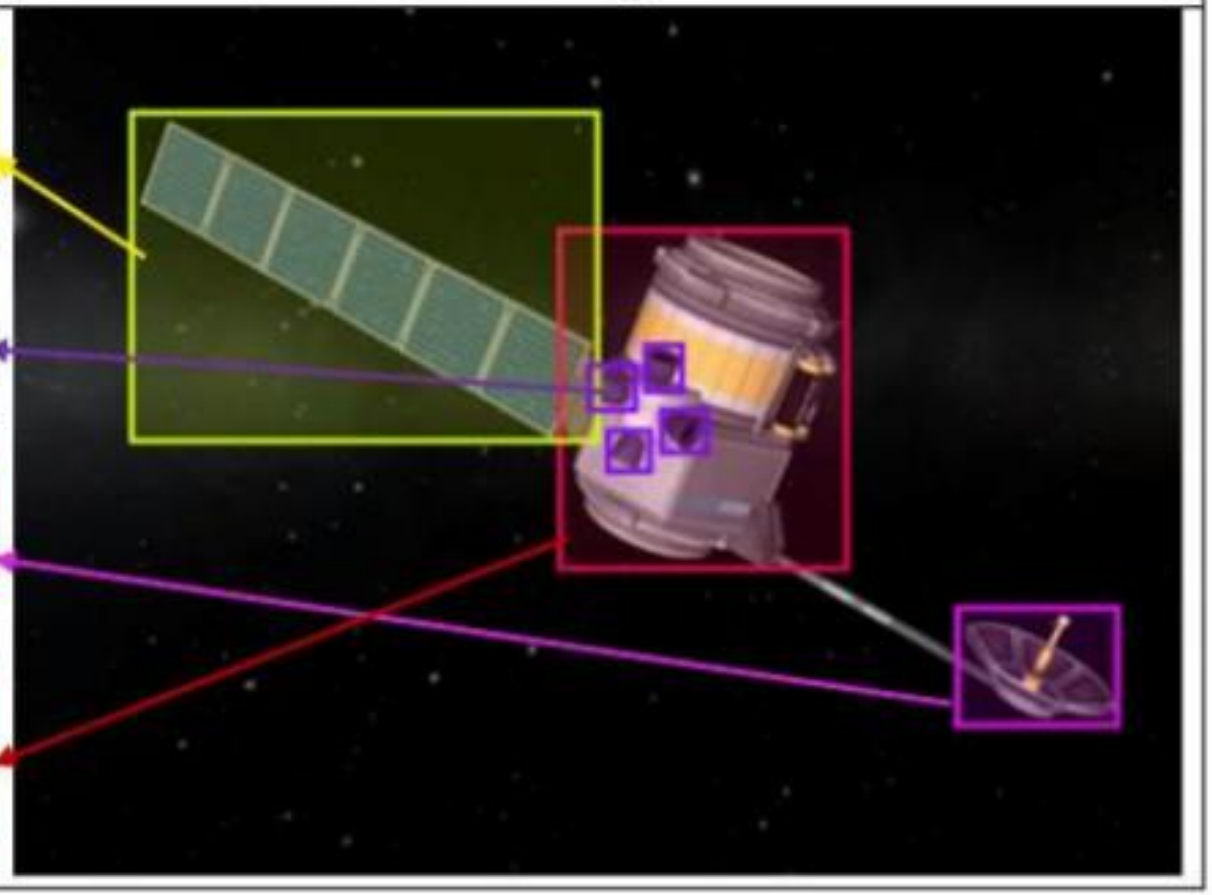
# Object Detection Algorithms

- You Only Look Once (YOLO) is the **fastest**
- YOLO is ideal with **weak** on-board computers

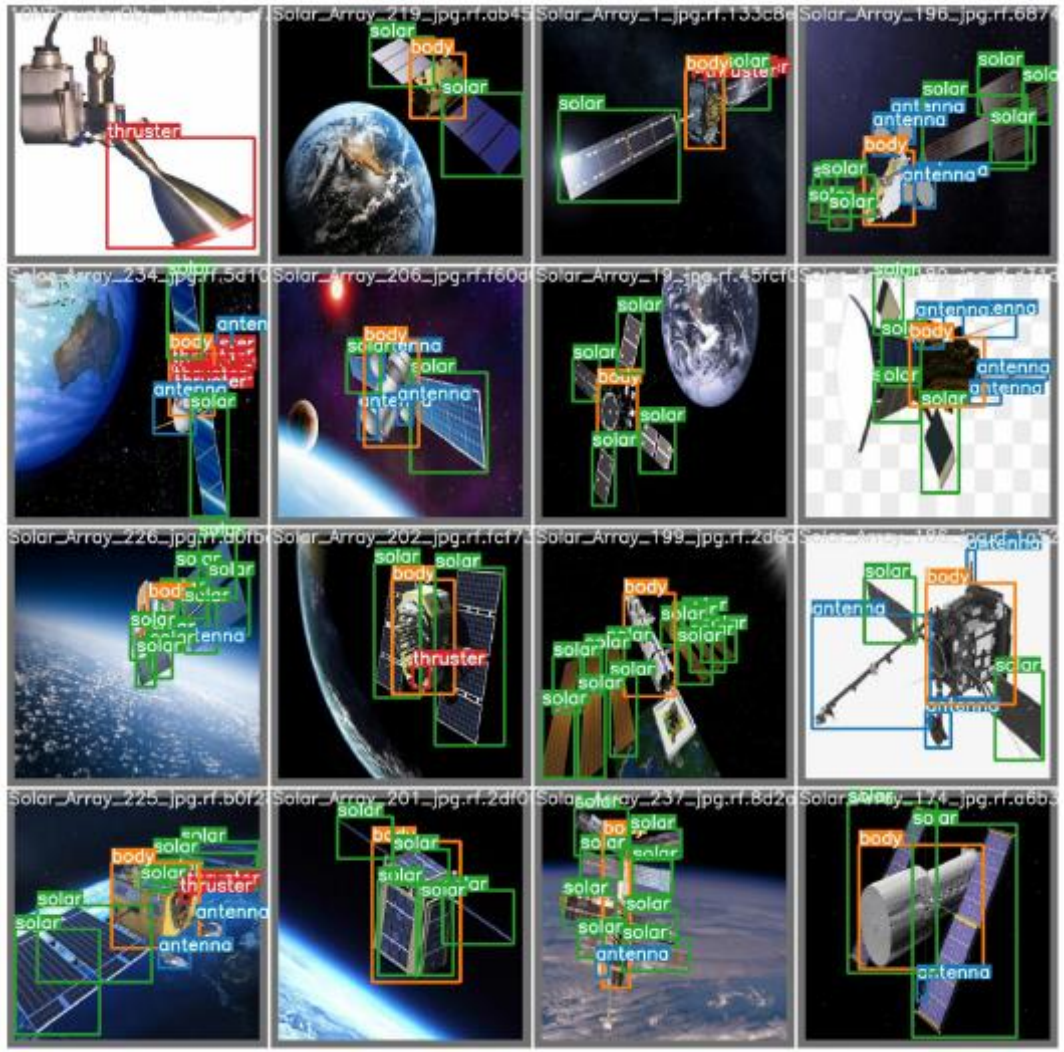



| Method | mAP | time |
|---|---|---|
| [B] SSD321 | 28.0 | 61 |
| [C] DSSD321 | 28.0 | 85 |
| [D] R-FCN | 29.9 | 85 |
| [E] SSD513 | 31.2 | 125 |
| [F] DSSD513 | 33.2 | 156 |
| [G] FPN FRCN | 36.2 | 172 |
| RetinaNet-50-500 | 32.5 | 73 |
| RetinaNet-101-500 | 34.4 | 90 |
| RetinaNet-101-800 | **37.8** | 198 |
| **YOLOv3-320** | 28.2 | **22** |
| **YOLOv3-416** | 31.0 | 29 |
| **YOLOv3-608** | 33.0 | 51 |

# Data for Object Detection

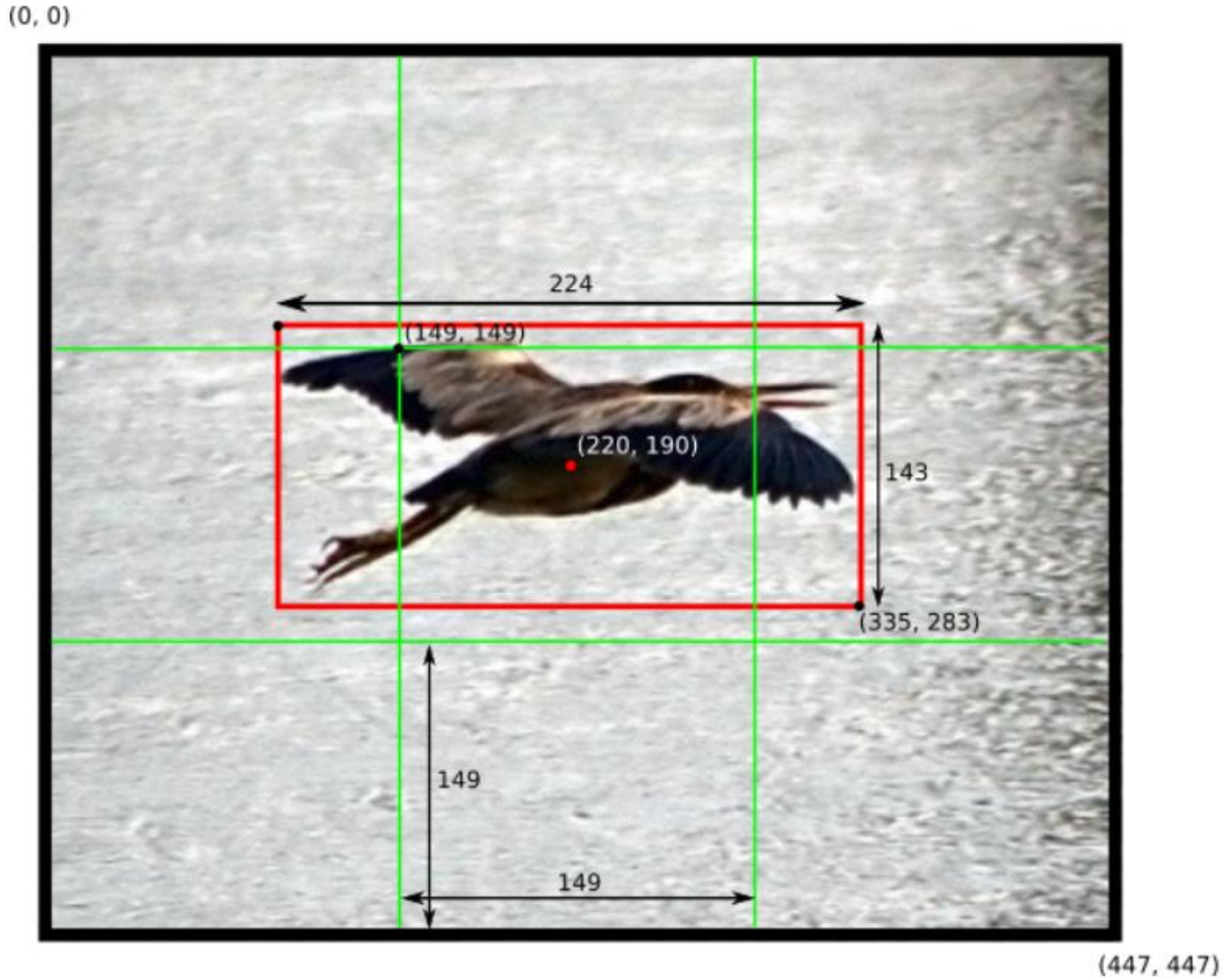
| Class | #Annotations | Example |
|---|---|---|
| Solar Array | 1204 |  |
| Thrusters | 737 | |
| Antennas | 692 | |
| Satellite Bodies | 416 | |

# Our Dataset



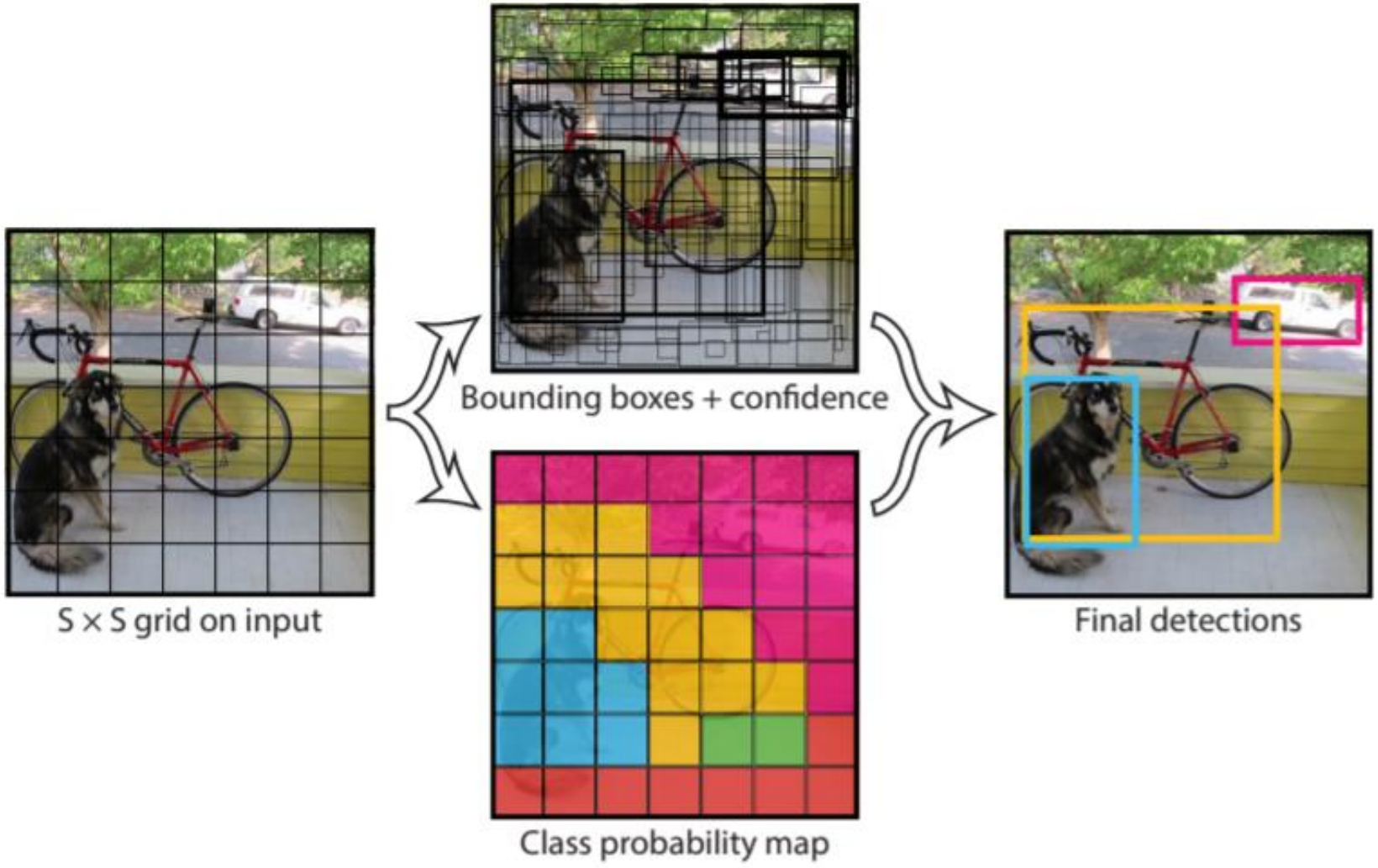
# Bounding Boxes

- Is there an object in the box?
- Where is the center (x,y)?
- How wide is it?
- How high is it?
- What is the object?

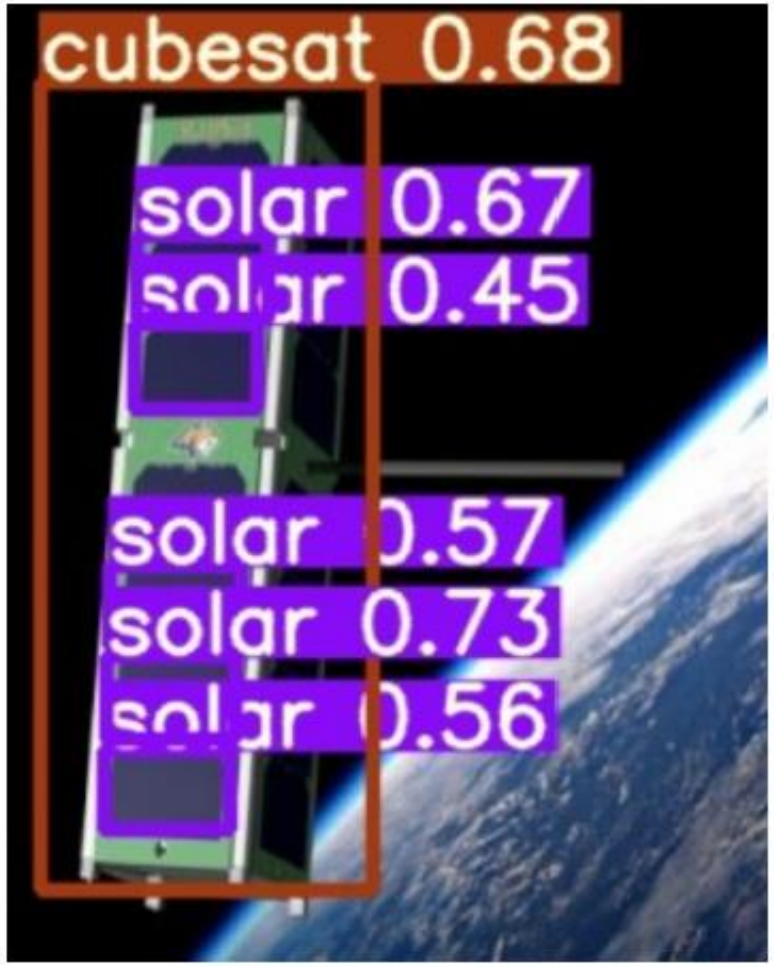



$$y = (1, 220, 190, 224, 143, 0, 1, 0, 0)$$

# How YOLO Works



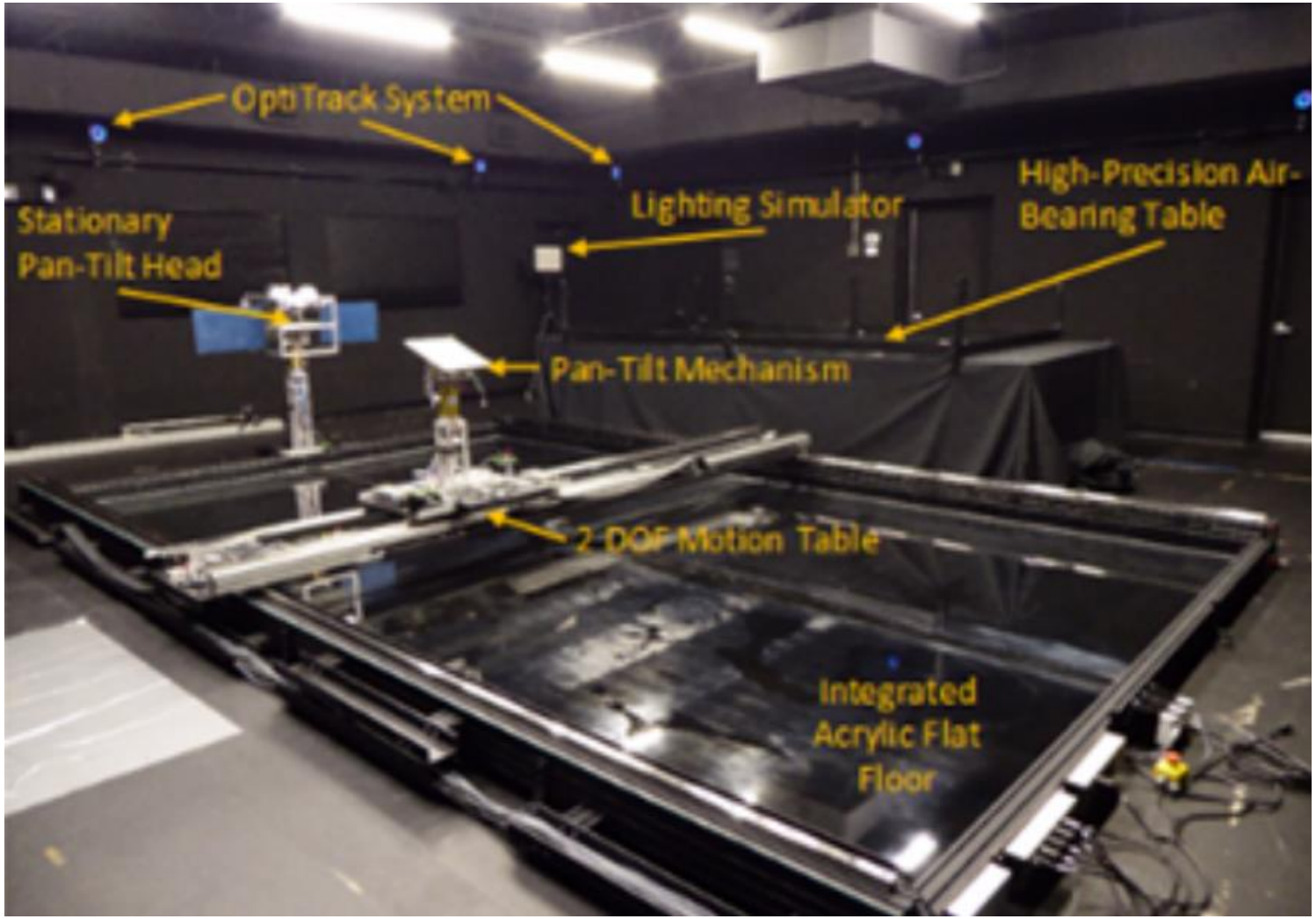
# Testing Examples



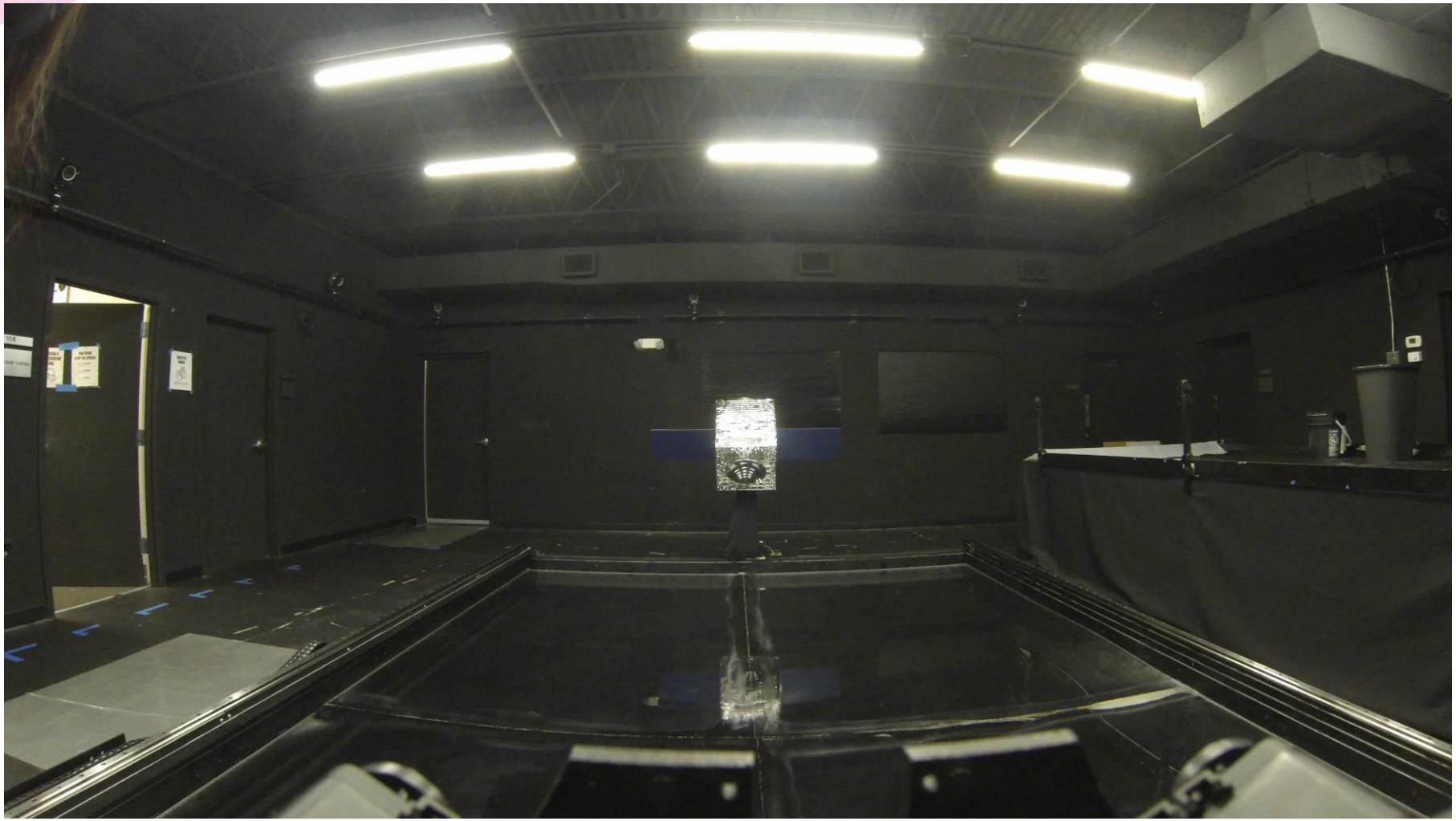

# Unconventional Data


body 0.57
solar 0.84

# ORION Lab at Florida Tech



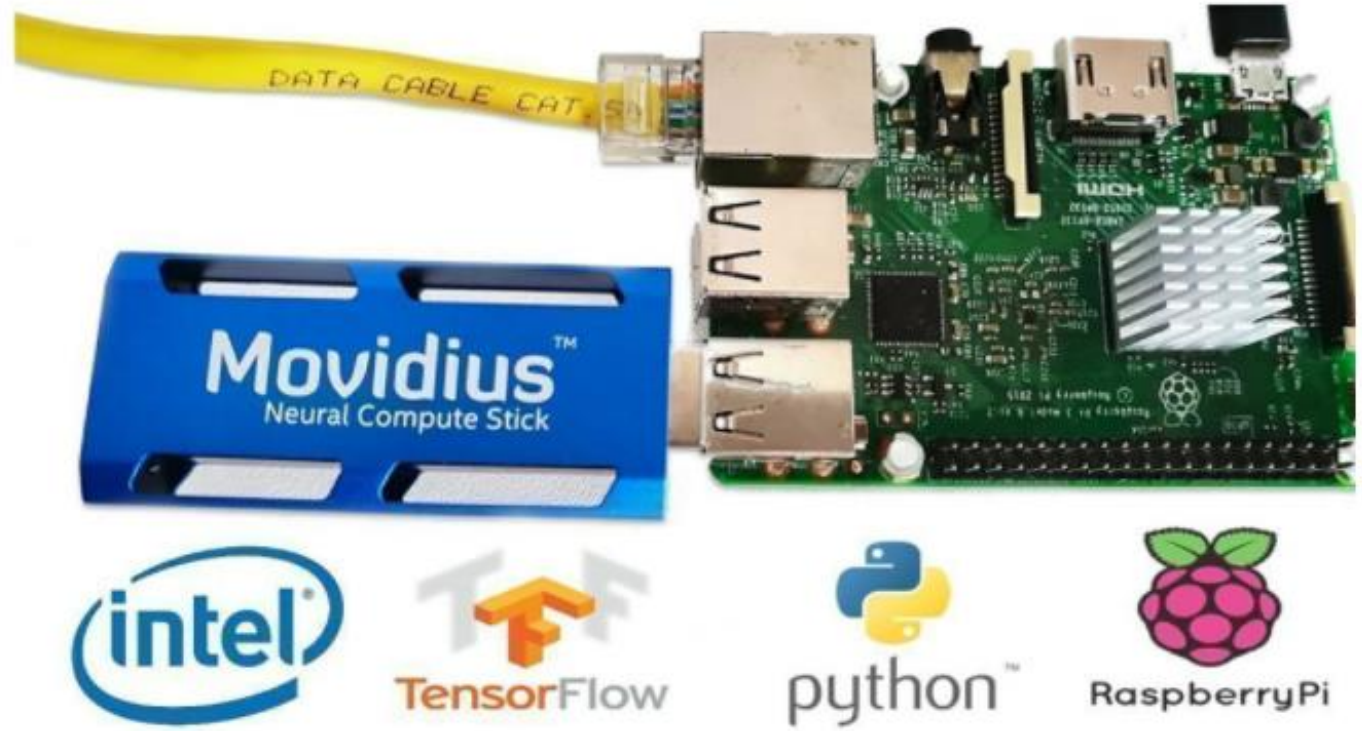
# Lab Capabilities



# Tests in the Lab

# Future Plans

More testing with on-board hardware



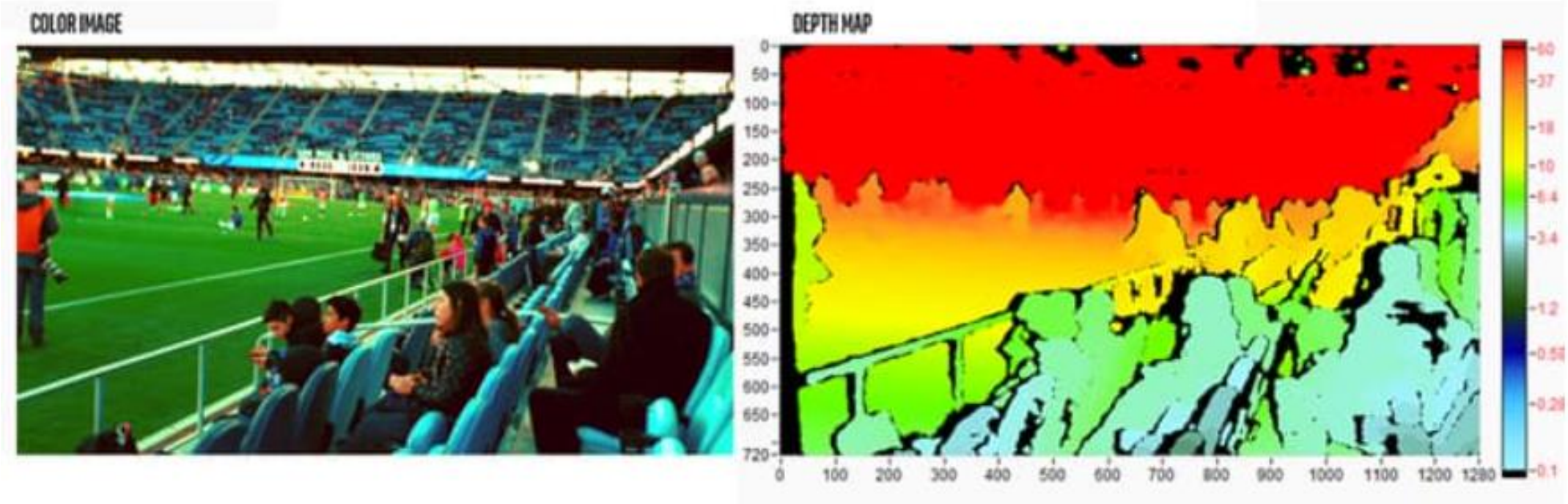
# Future Plans

Strap the Computer/Camera to a Drone

Flightpath Planning, Guidance, Navigation

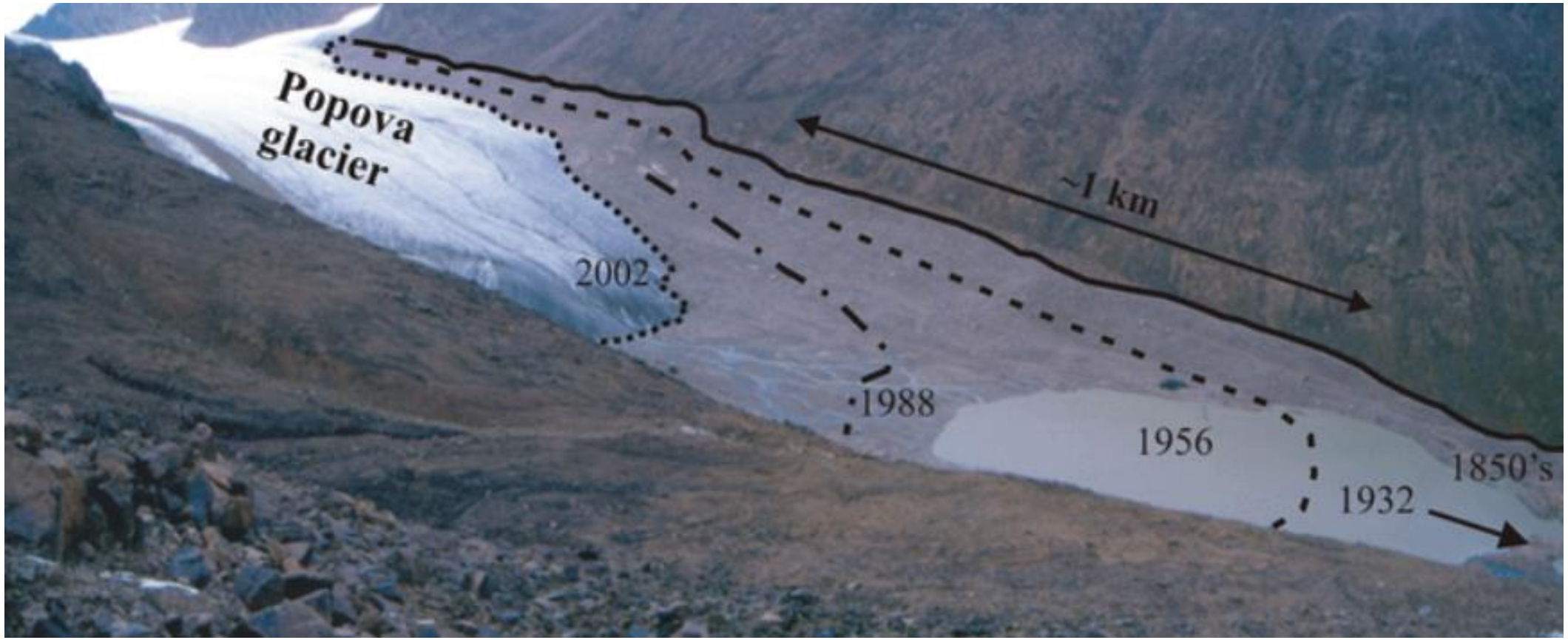
# Future Plans

## Exploit Depth Data

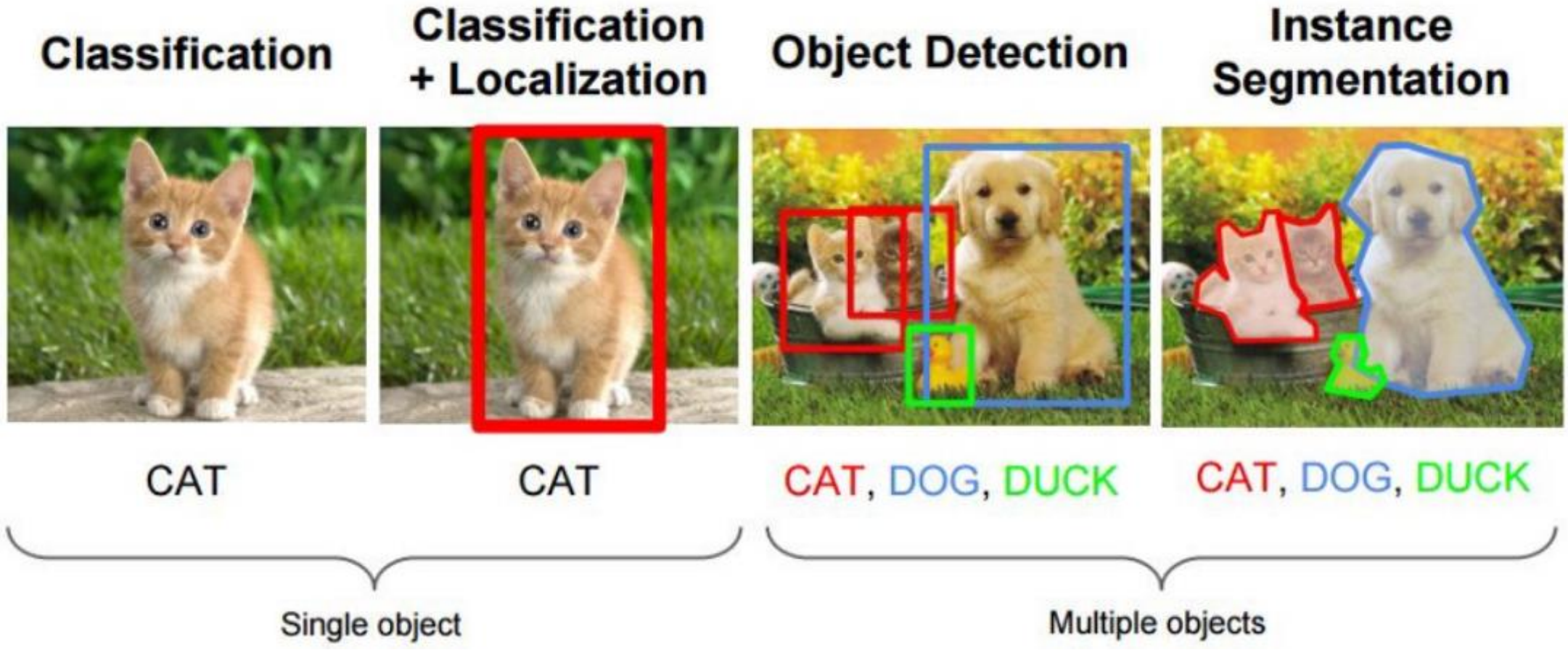


# O’Higgins Glacier Ice Flow

# Glacier Recession



# An Even Harder Problem

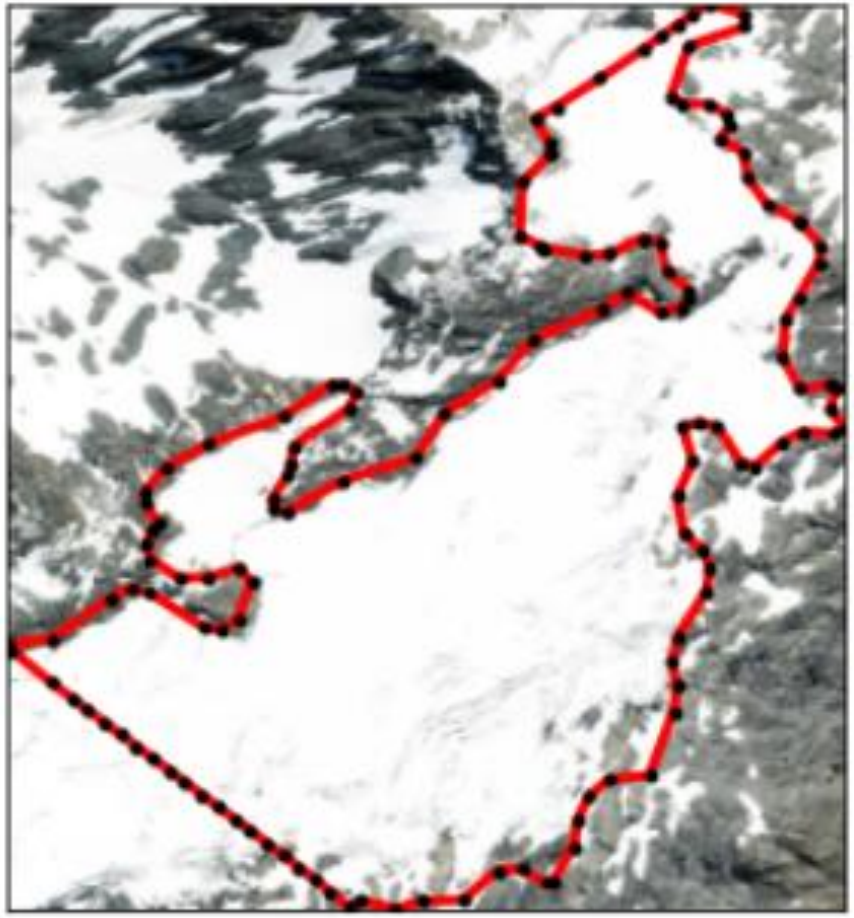


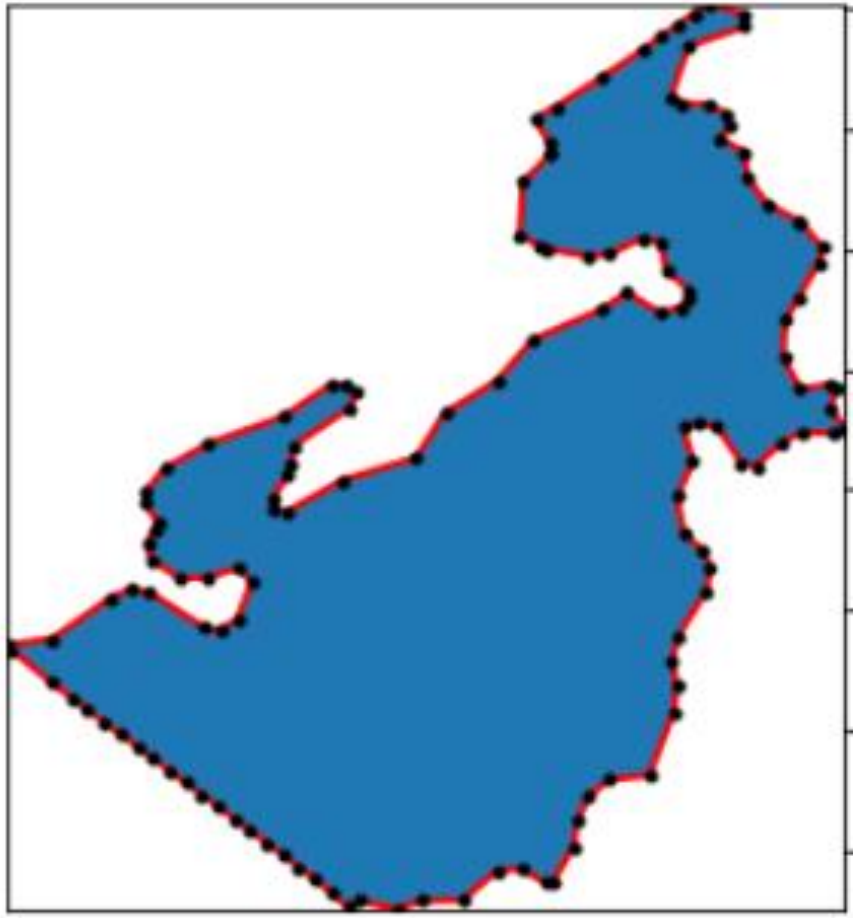
# Questions

- Which glaciers are changing over time?
- How are they changing?
- How quickly are they changing?
- With 10,000+ glaciers, can humans do it manually?

Current work here supported by the National Science Foundation

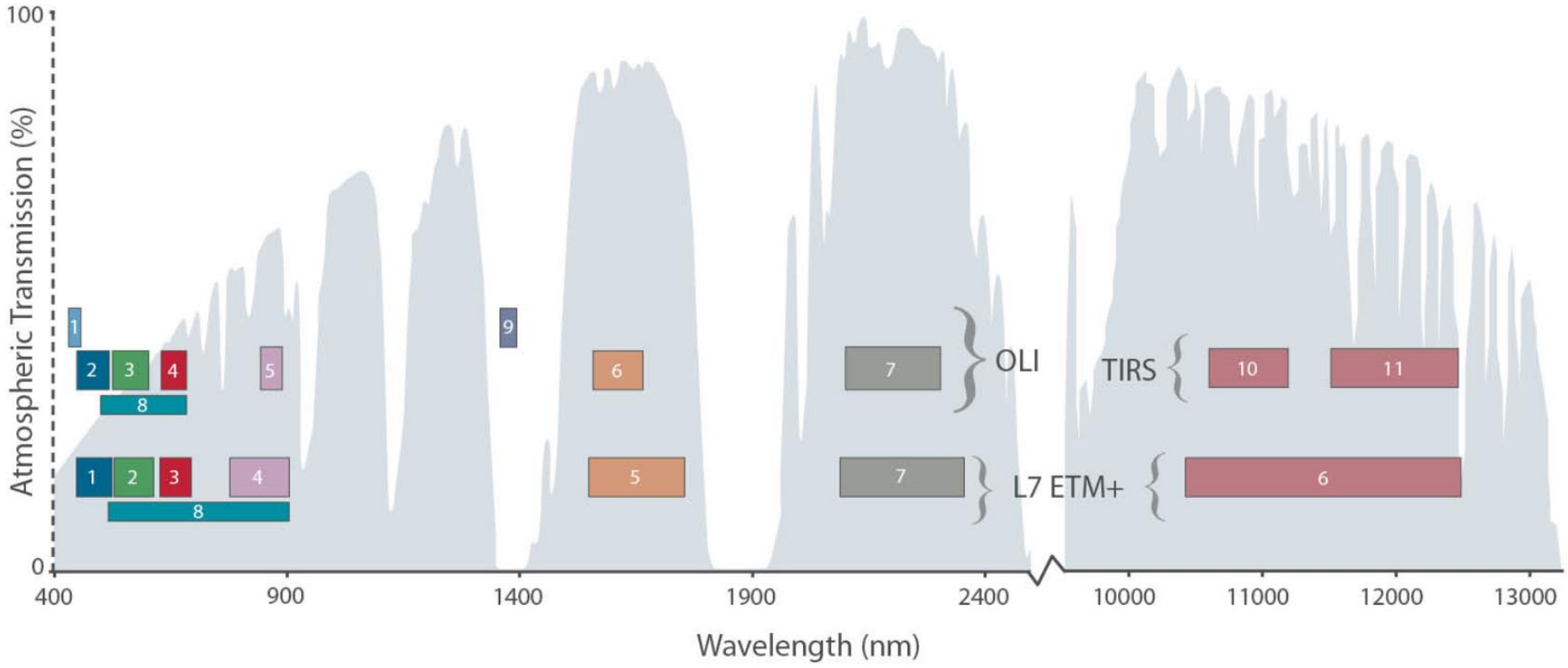
# Current Project: Measuring Glaciers



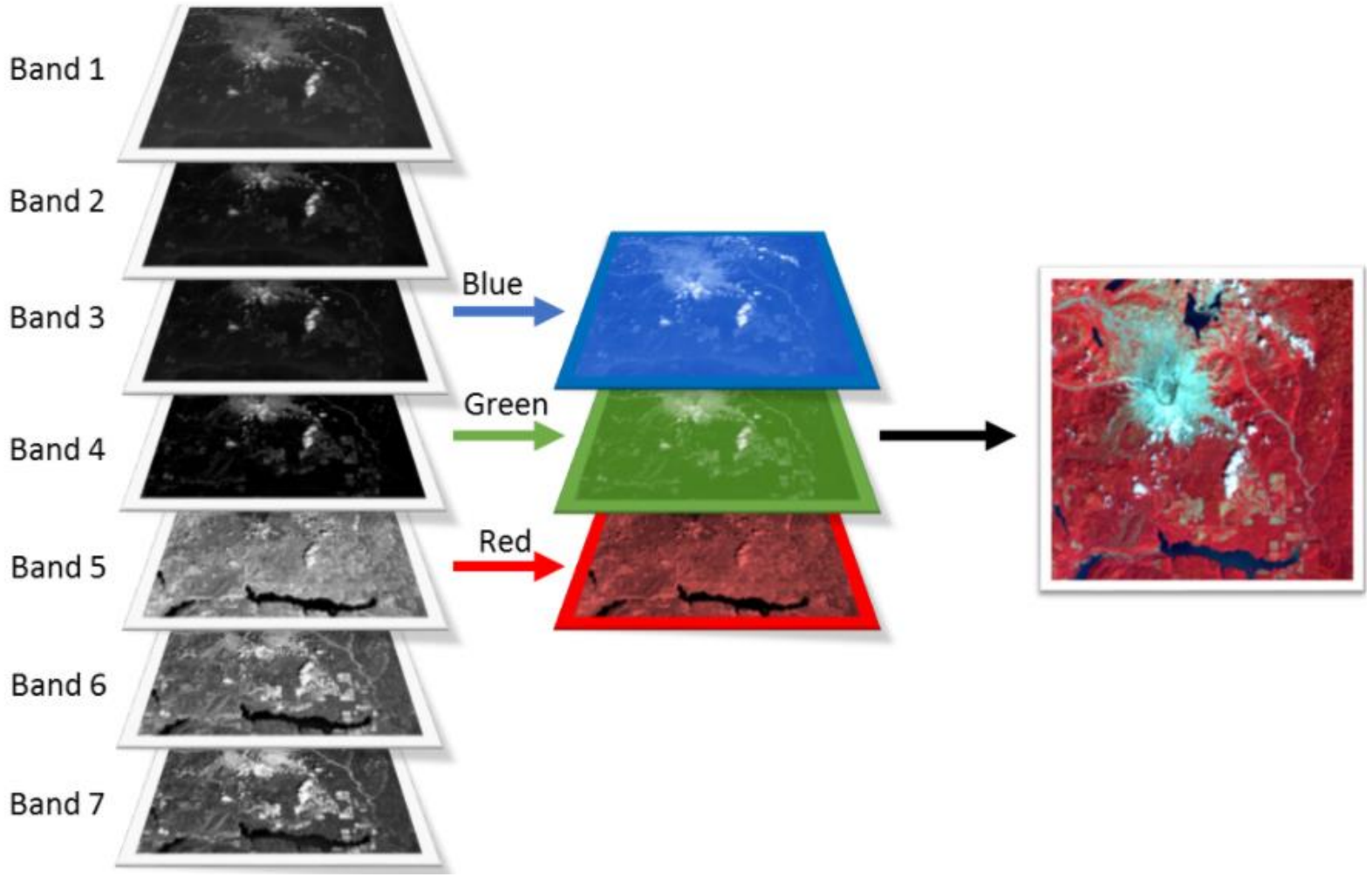


# Data Source: Landsat

# LANDSAT Satellite Imagery

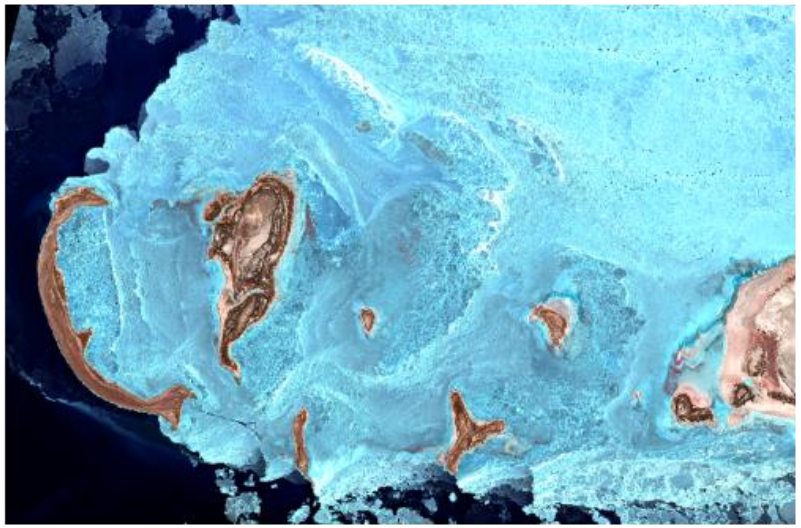


# 7-channel “Pictures”

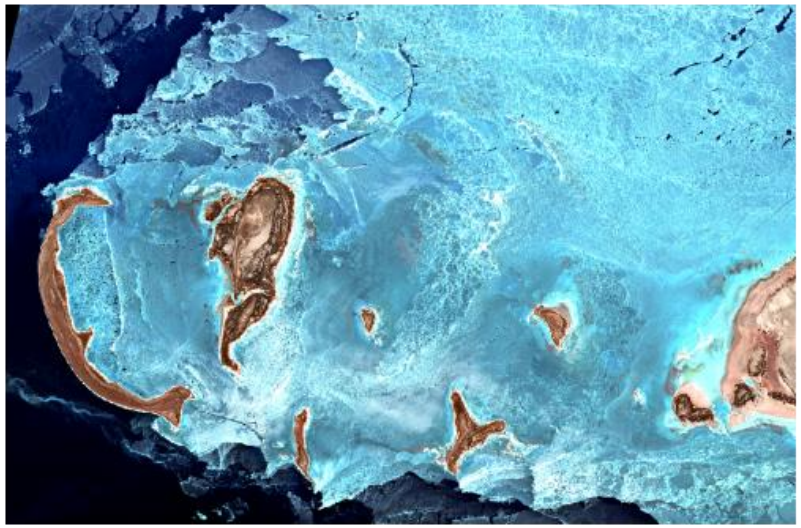


# Time-Lapse Satellite Images

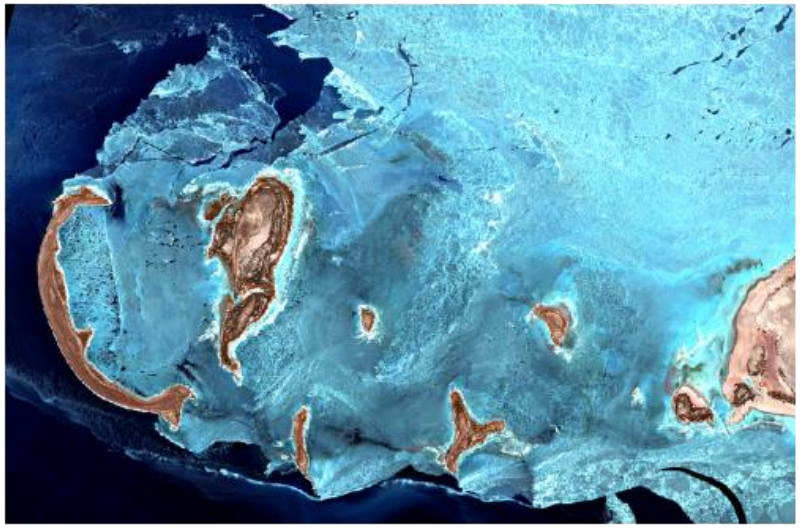
2020-12-11



2020-12-26

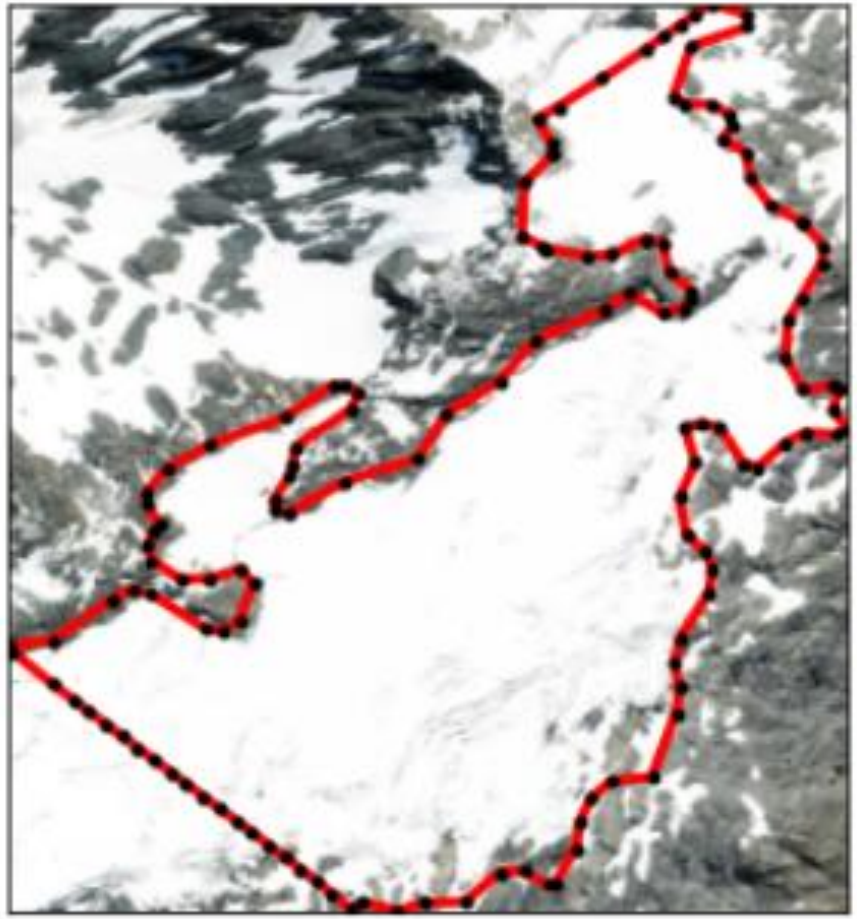


2021-01-05

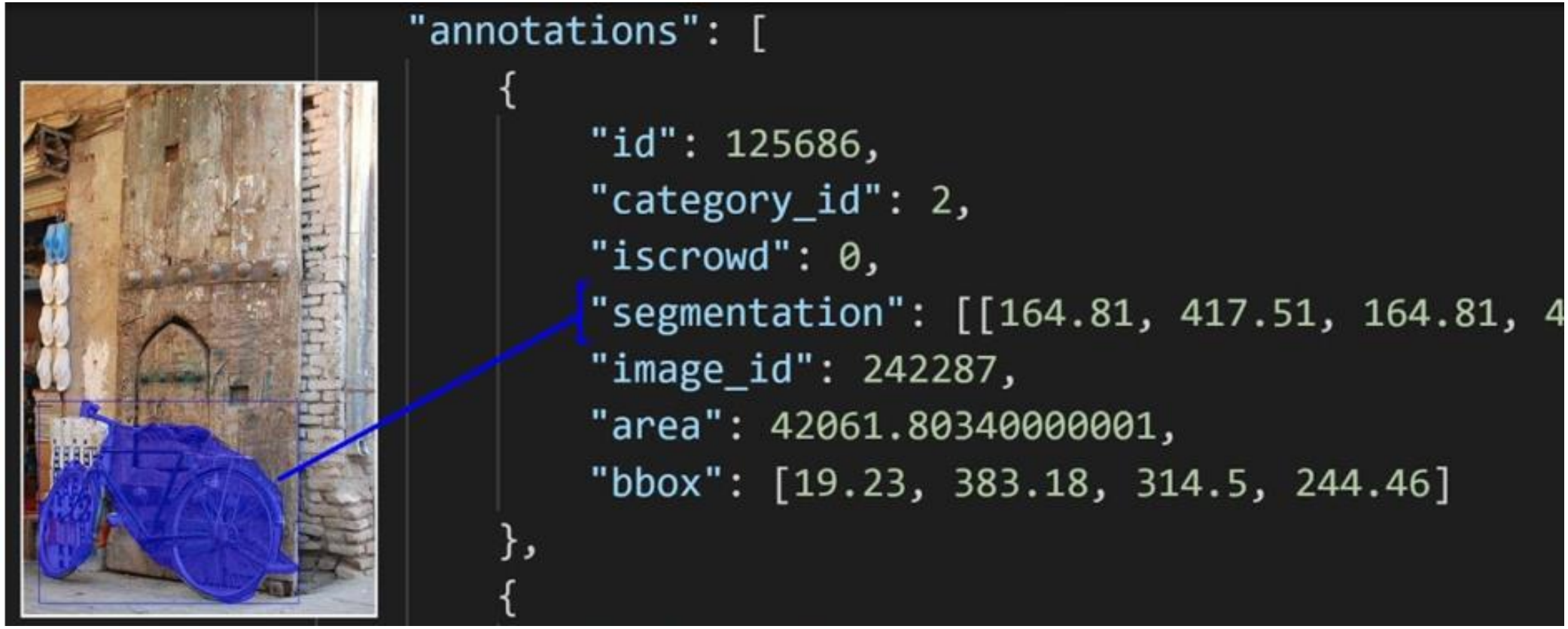


2021-01-25

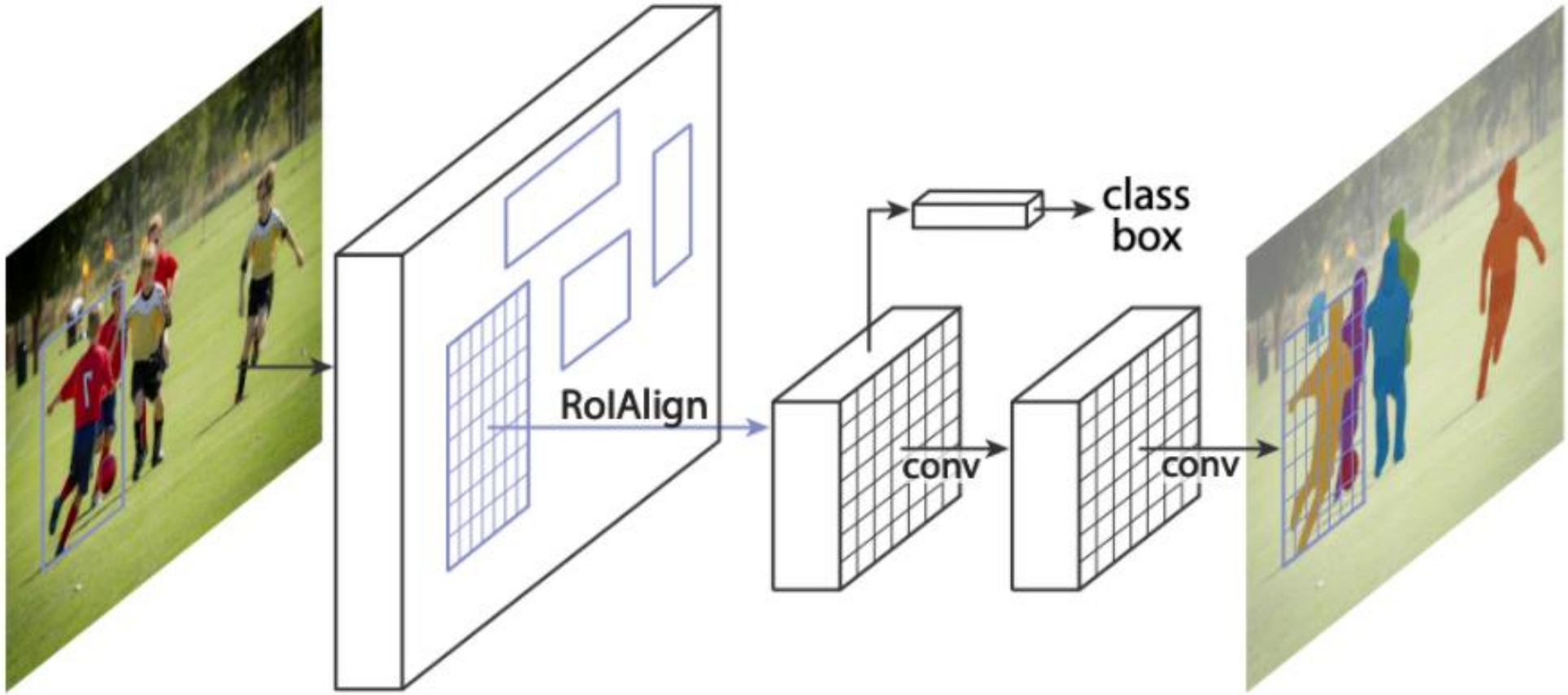
# Predicting Polygons (Masks)



# Data for Image Segmentation



# Mask R-CNN



# Questions?


Contact

Ryan T. White, Ph.D.
rwhite@fit.edu
www.ryantwhite.com
https://www.linkedin.com/in/ryantwhite5/