Convolutions and Transfer Learning for Computer Vision

Brian Plancher
Barnard College, Columbia University
Harvard John A. Paulson School of Engineering and Applied Sciences
brianplancher.com

Quick Disclaimer:

Today will be **both** **too fast**
and **too slow**!

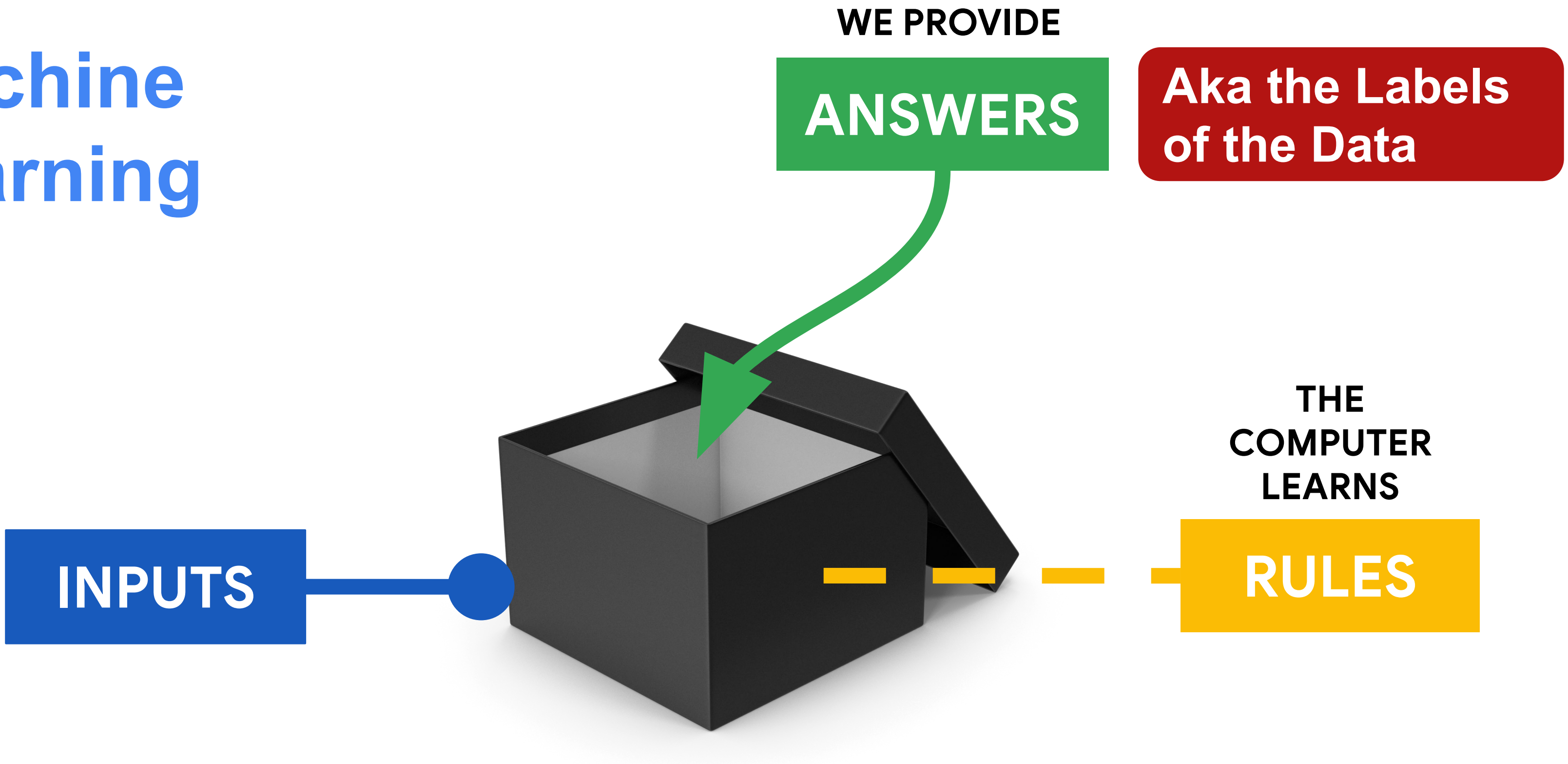
Camera feed

```
Starting inferencing in 2 seconds...
Taking photo...
Predictions (DSP: 9 ms., Classification: 322 ms., Anomaly: 0 ms.):
    car: 0.07812
    truck: 0.92188
```

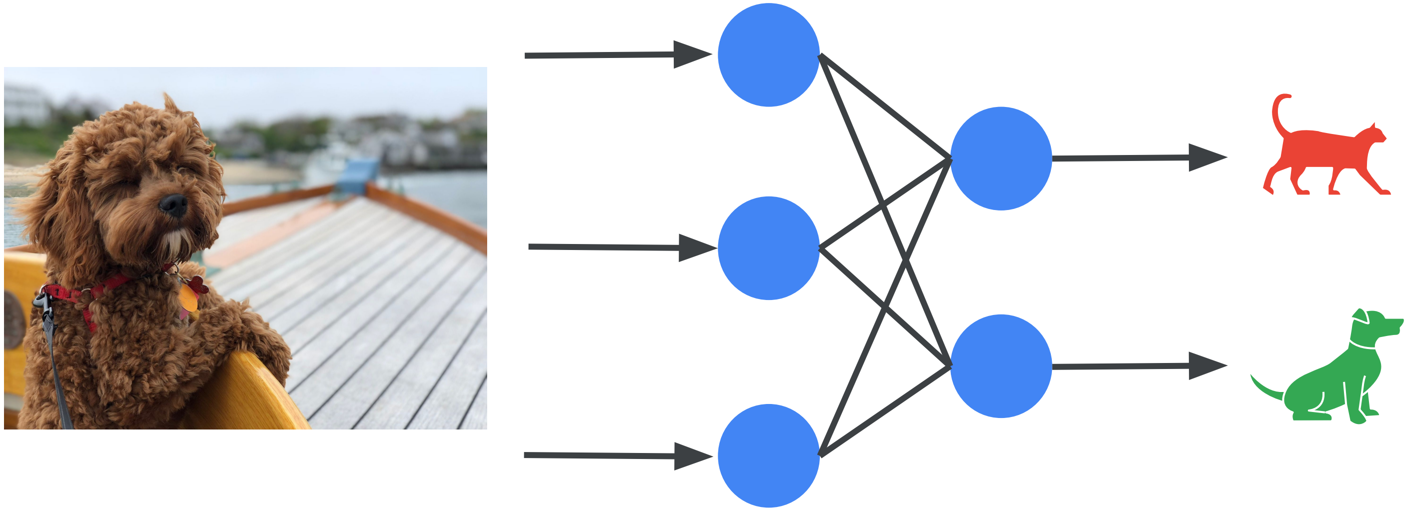
By the end of today: **Hands-on Computer Vision** (Object Classification)

We will explore the **science** behind computer vision and **collect data** and **train** our own custom model to recognize objects using **Edge Impulse**

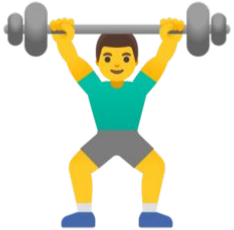
Machine Learning

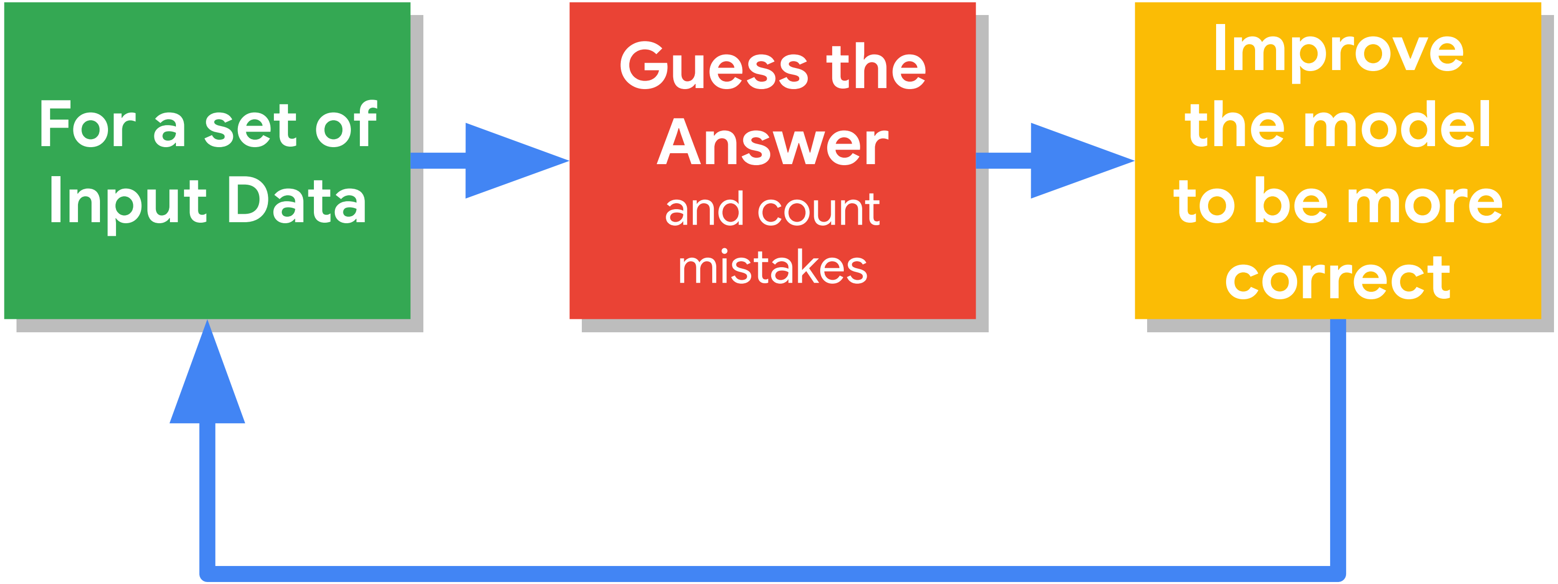


Machine Learning with **neural networks**?

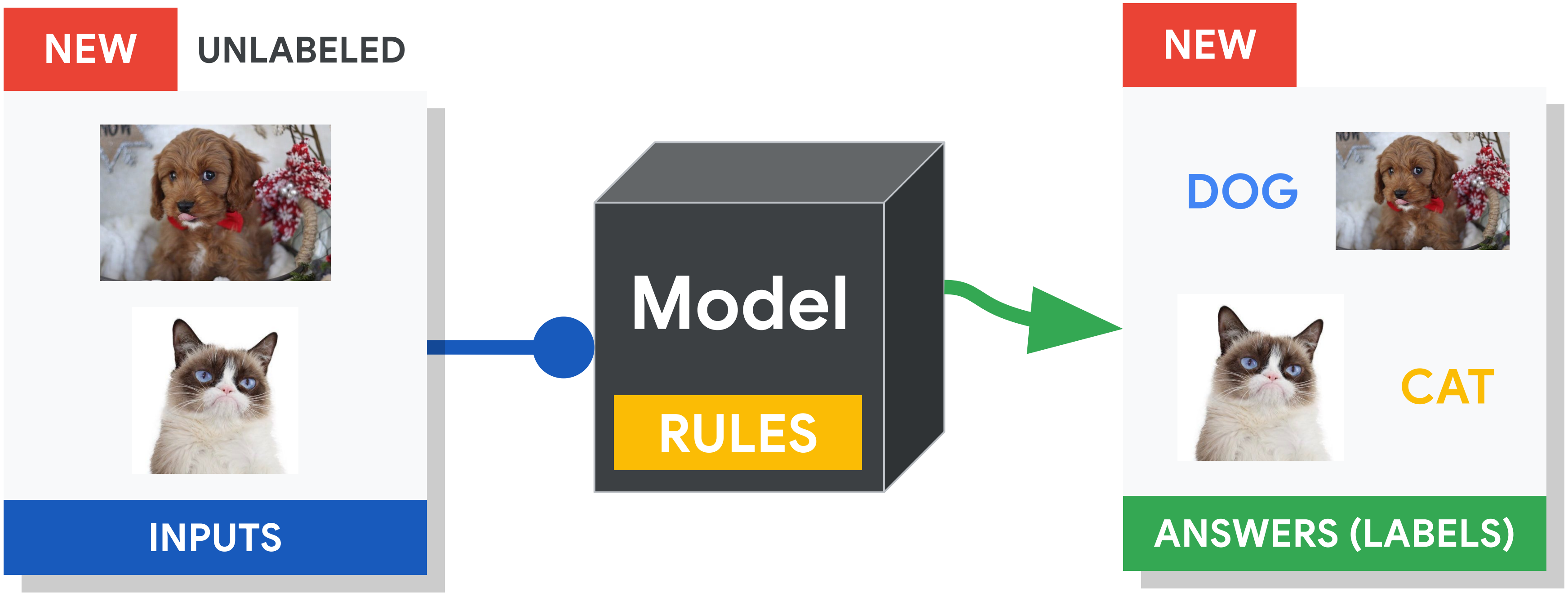


Training the machine





After it's **learned** use it for **inference**:

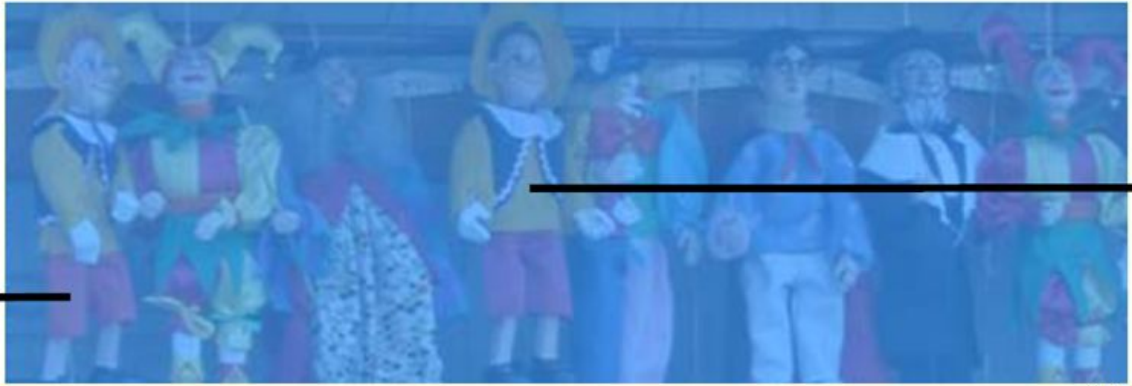


Computer Vision is Hard

Computer Vision is Hard

What color are the pants and the shirt?

?



?

Slide Credit: Hamilton Chong

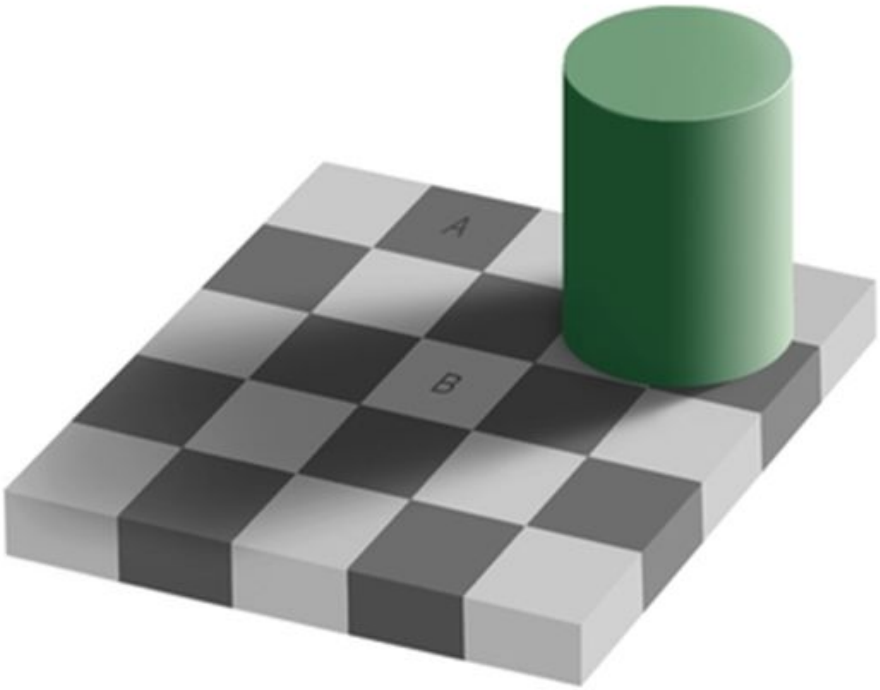
Computer Vision is Hard

Slide Credit: Hamilton Chong

Computer Vision is Hard

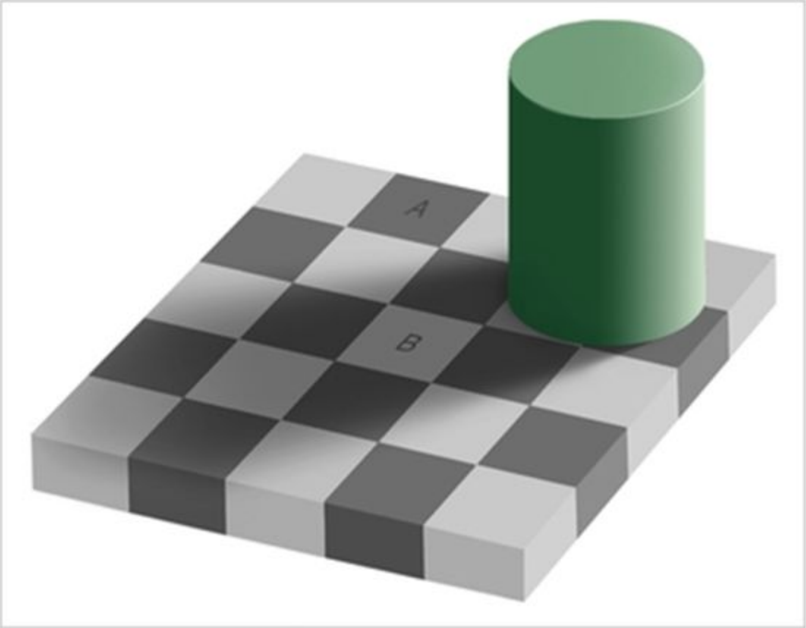
Slide Credit: Hamilton Chong

Computer Vision is Hard

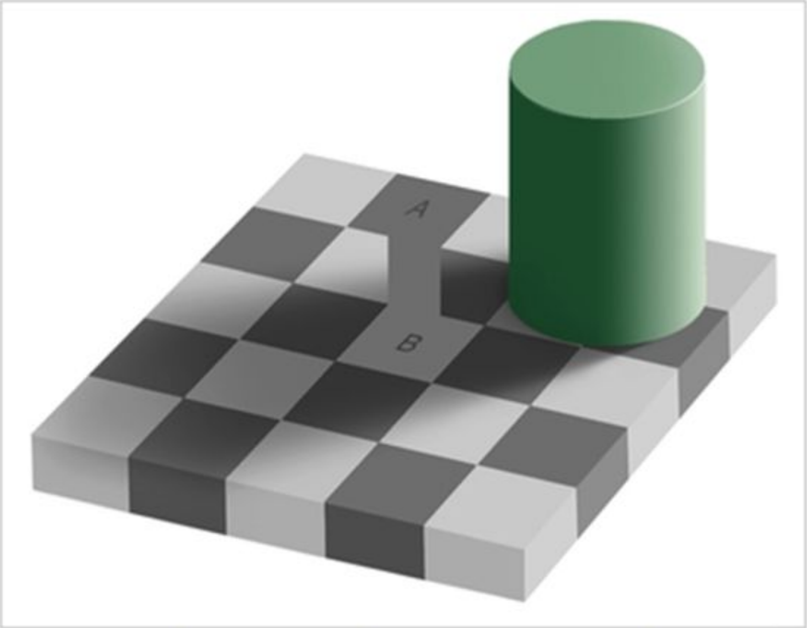


Is square A or B darker in color?

Computer Vision is Hard

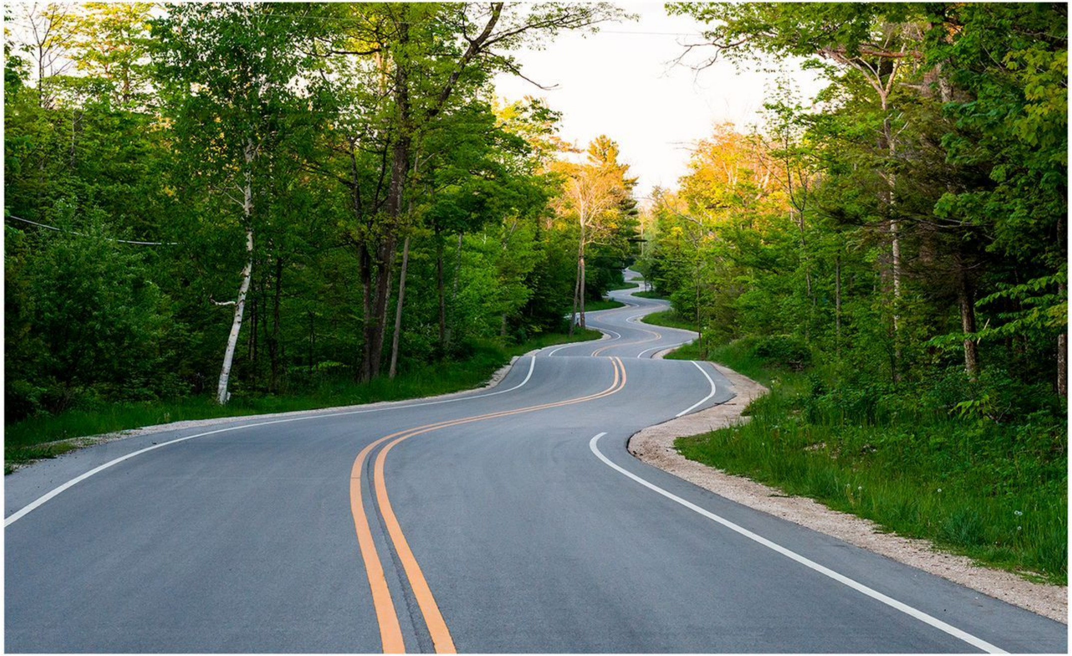


Areas of the image A and B are the same color



A rectangle of the same color has been drawn connecting the two areas of the image

What **Features** of the image might be important for self driving cars?

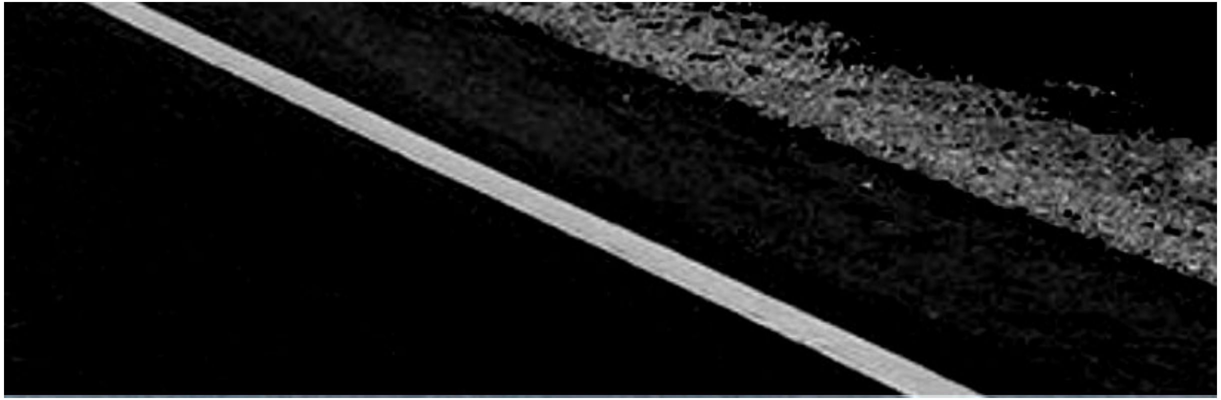


What **Features** of the image might be important for self driving cars?

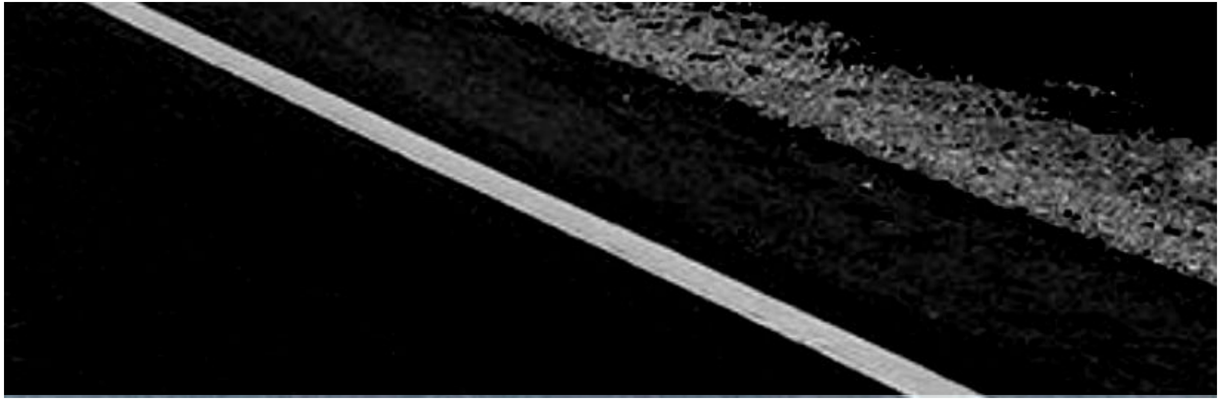
Maybe straight lines to see the lanes of the road?

How might we find these features?

How might we find these features?

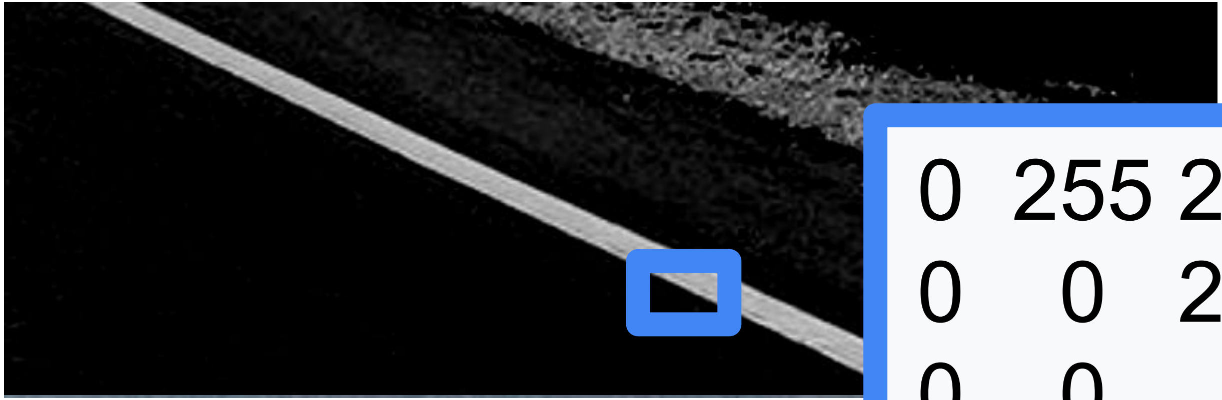


How might we find these features?



Black: 0
White: 255

How might we find these features?

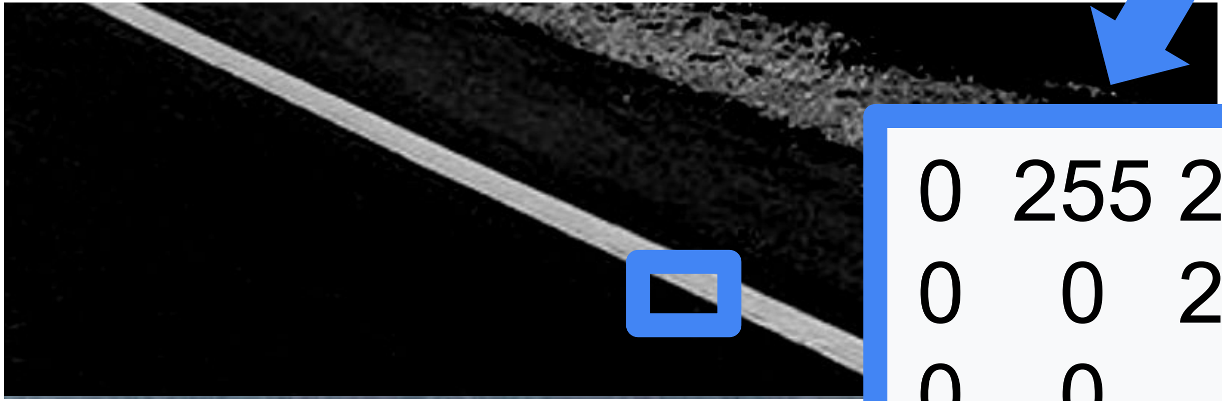


0	255	255	255	255
0	0	255	200	255
0	0	0	255	255
0	0	0	0	255
0	0	0	0	0

Black: 0
White: 255

How might we find these features?

Look for a Big Change!



0	255	255	255	255
0	0	255	200	255
0	0	0	255	255
0	0	0	0	255
0	0	0	0	0

Black: 0
White: 255

How might we find these features?
Convolutions

How might we find these features?
Convolutions

Original Image

0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255

How might we find these features?
Convolutions

Original Image

0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255

Filter

-1	0	1
-1	0	1
-1	0	1

How might we find these features?
Convolutions

Original Image

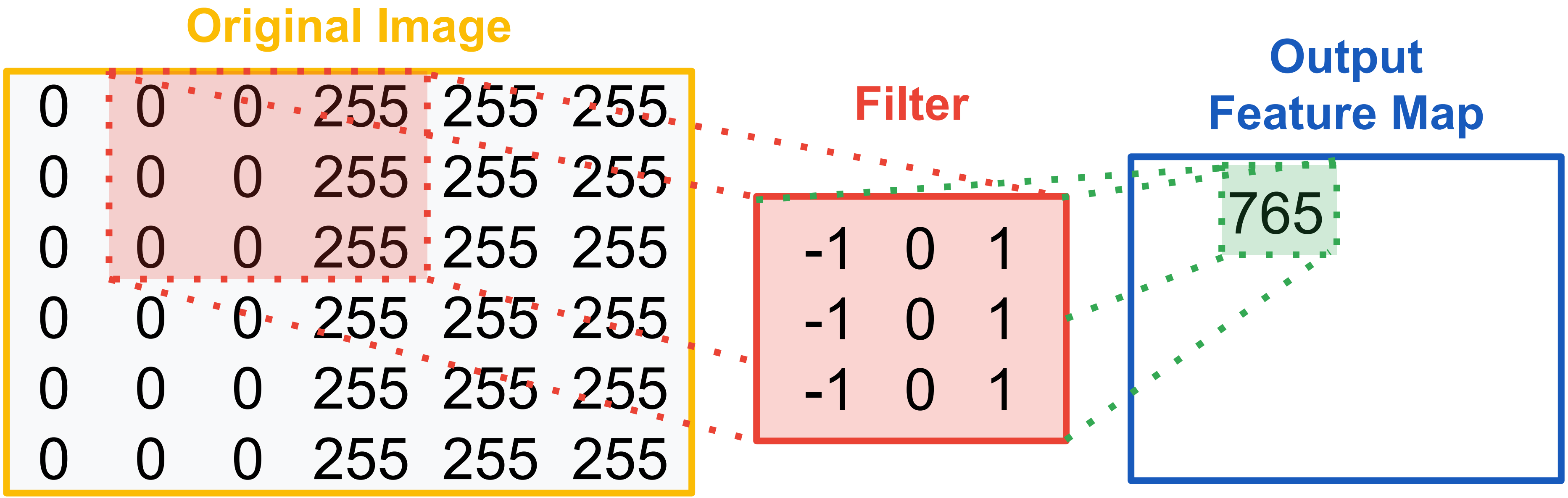
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255

Filter

-1	0	1
-1	0	1
-1	0	1

How might we find these features?

Convolutions



How might we find these features?
Convolutions

Original Image

0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255
0	0	0	255	255	255

Filter

-1	0	1
-1	0	1
-1	0	1

Output Feature Map

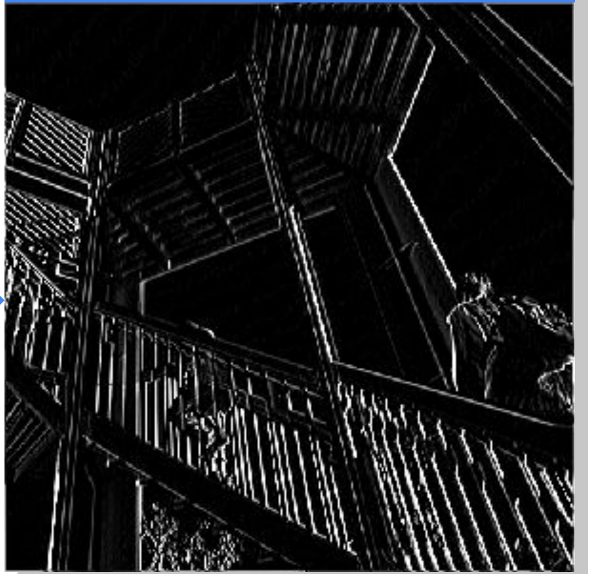
0	765	765	0
0	765	765	0
0	765	765	0
0	765	765	0

Colab Link

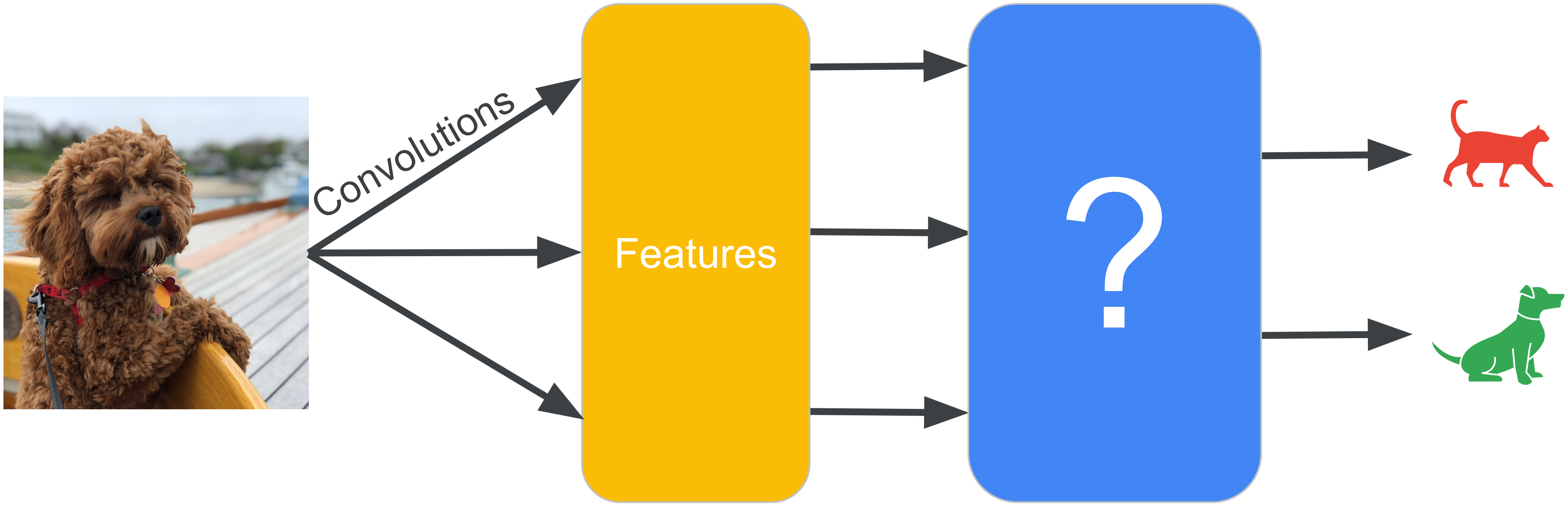
How might we find these features?
Convolutions

-1	0	1
-2	0	2
-1	0	1

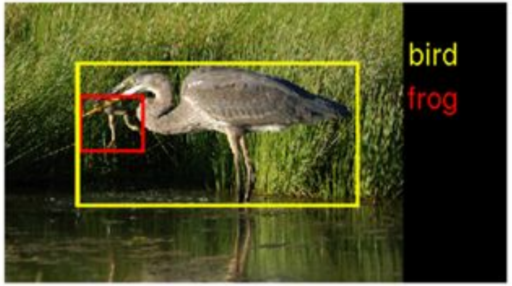
Features

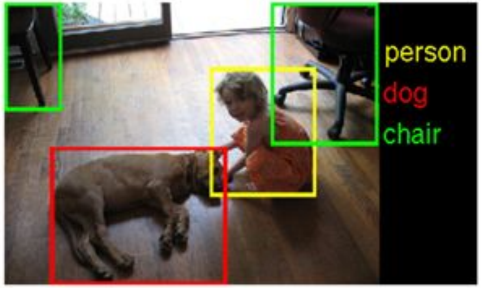


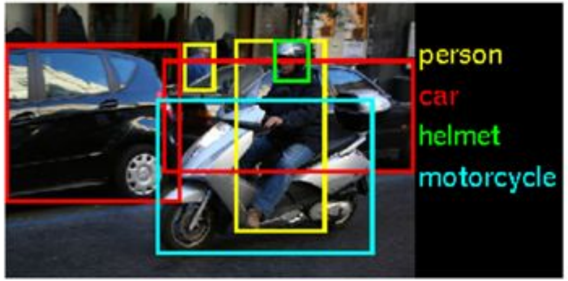
How might we combine these features to **classify an object**?



The ImageNet Challenge and the birth of CNNs







The ImageNet Challenge provided 1.2 million examples of 1,000 **labeled** items and challenged algorithms to learn from the data and then was tested on another 100,000 images

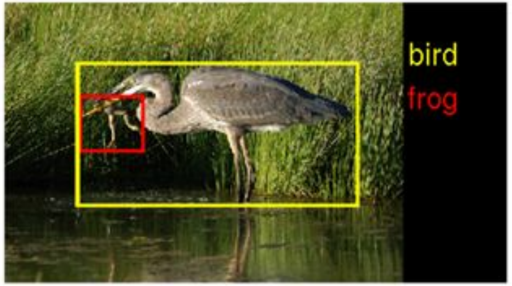
The ImageNet Challenge and the birth of CNNs

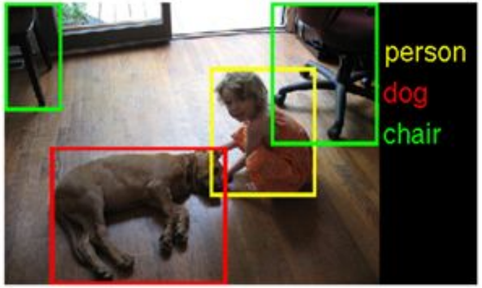
Traditional Machine Learning Flow

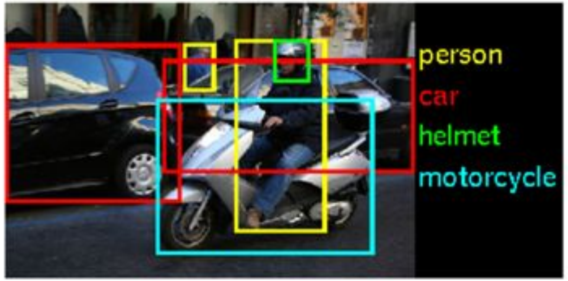
Regression, Clustering, etc.

Vertical Lines, Horizontal Lines, Changes in Color, Changes in Focus, etc.

The ImageNet Challenge and the birth of CNNs



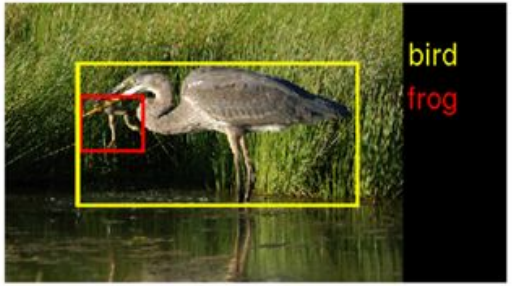


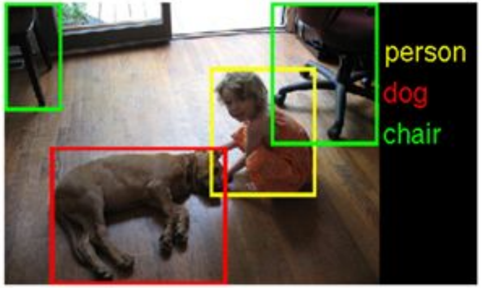


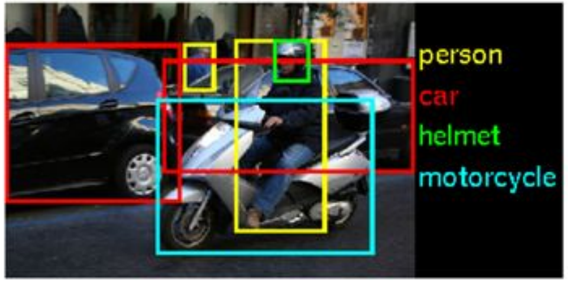
In 2010 teams had
75-50% error

In 2011 teams had
75-25% error

The ImageNet Challenge and the birth of CNNs

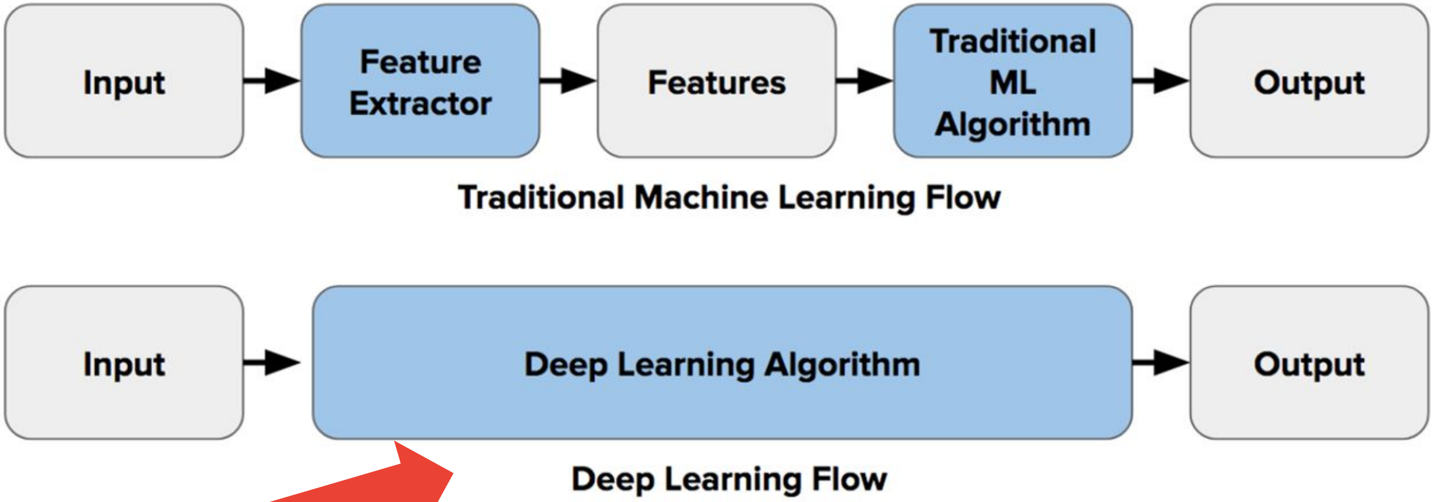






In 2012 still no team had less than 25% error barrier except **AlexNet at 15%**

The ImageNet Challenge and the birth of CNNs

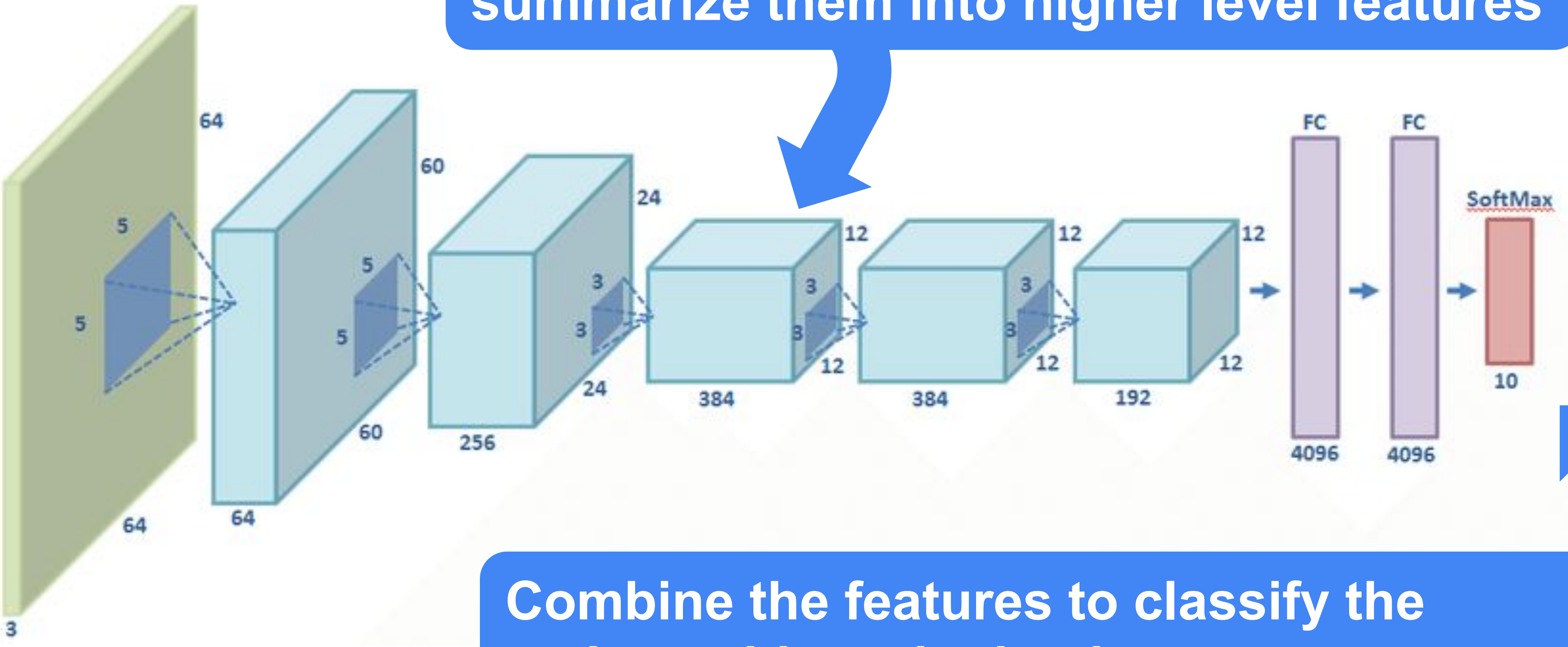


Let the computer figure out its own features and how to combine them!

[AlexNet Paper]

AlexNet

Use convolutions to find features and the summarize them into higher level features



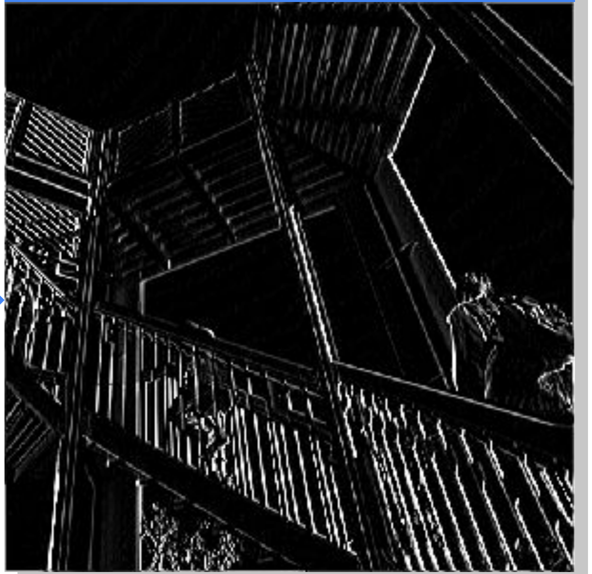
Combine the features to classify the various objects in the dataset

[Colab Link]

How might we find these features?
Convolutions

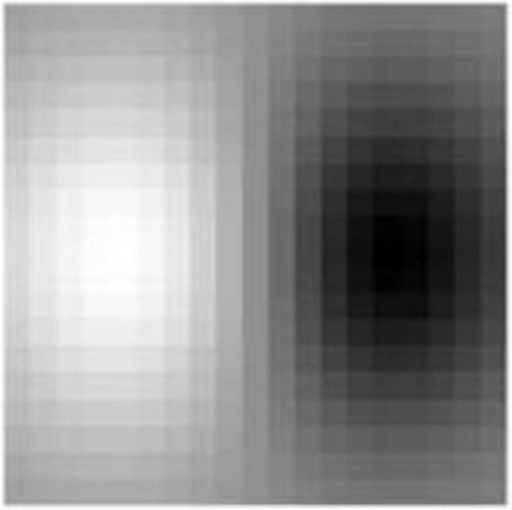
-1	0	1
-2	0	2
-1	0	1

Features

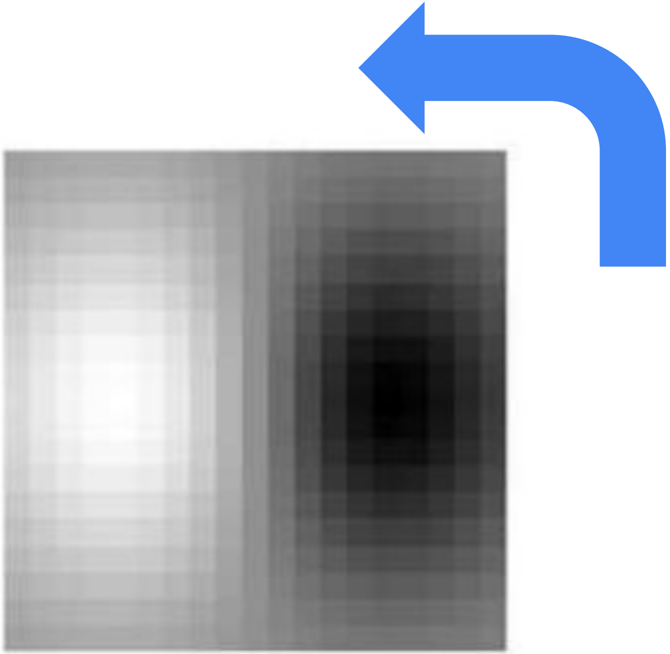


How might we find these features?
Convolutions

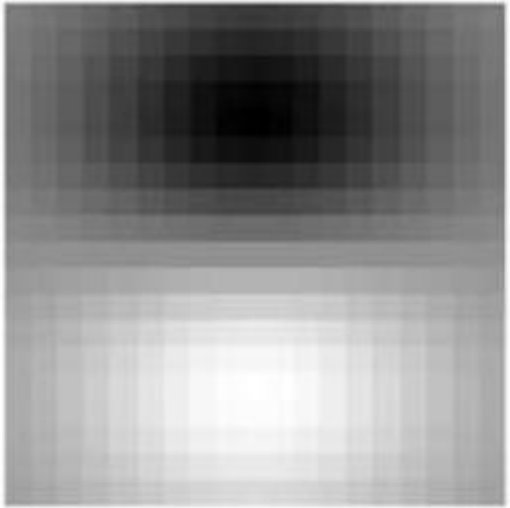
-1	0	1
-2	0	2
-1	0	1



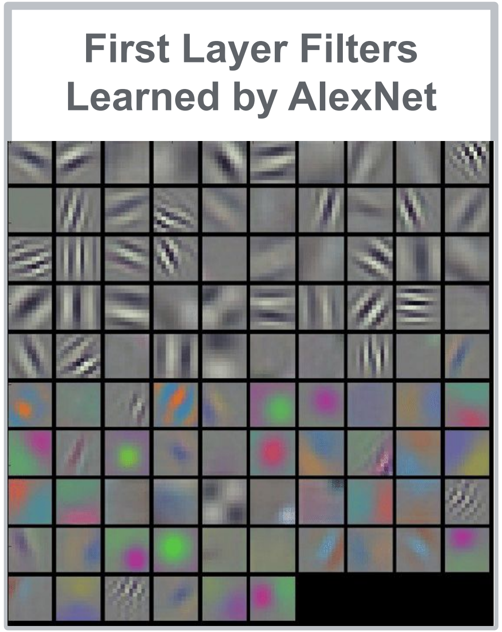
How might we find these features?
Convolutions

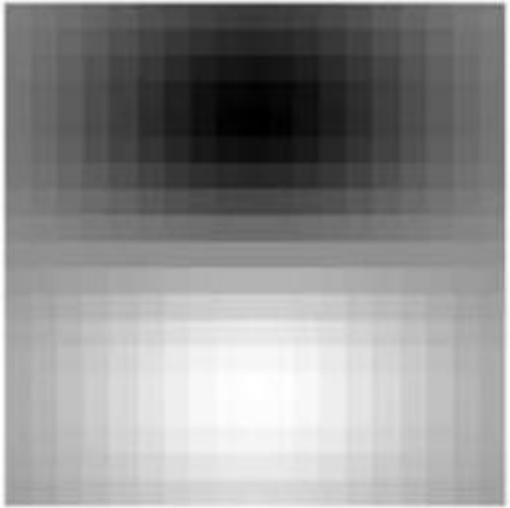


How might we find these features?
Convolutions

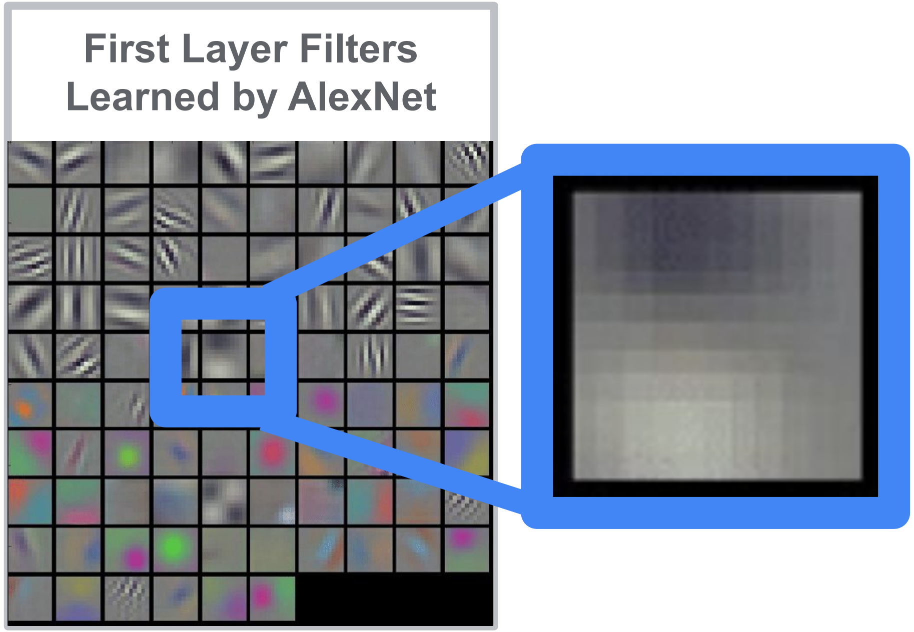


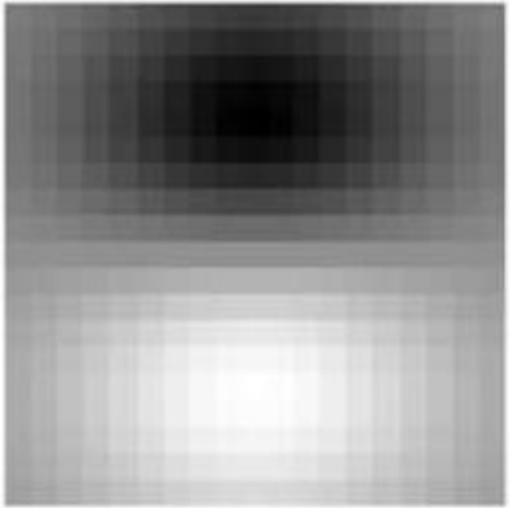
How might we find these features?
Convolutions





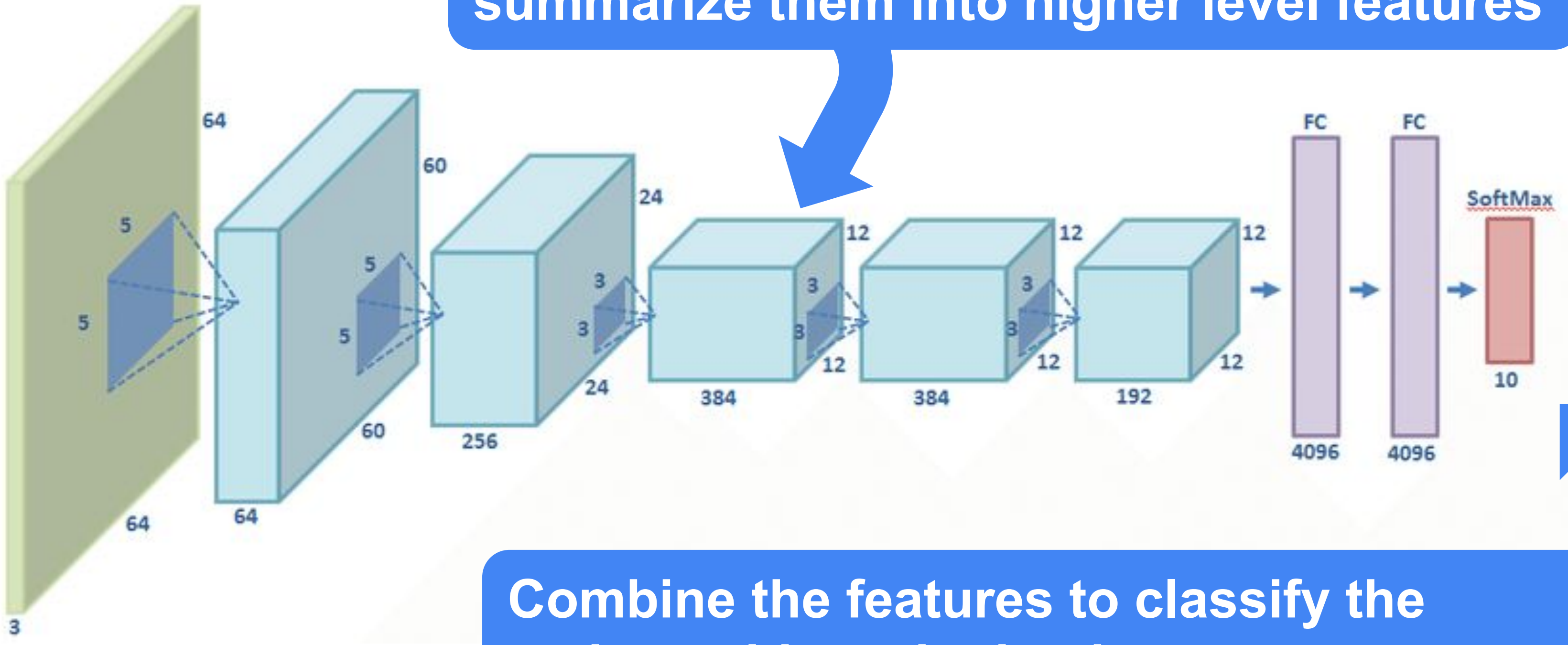
How might we find these features?
Convolutions





[AlexNet Paper]

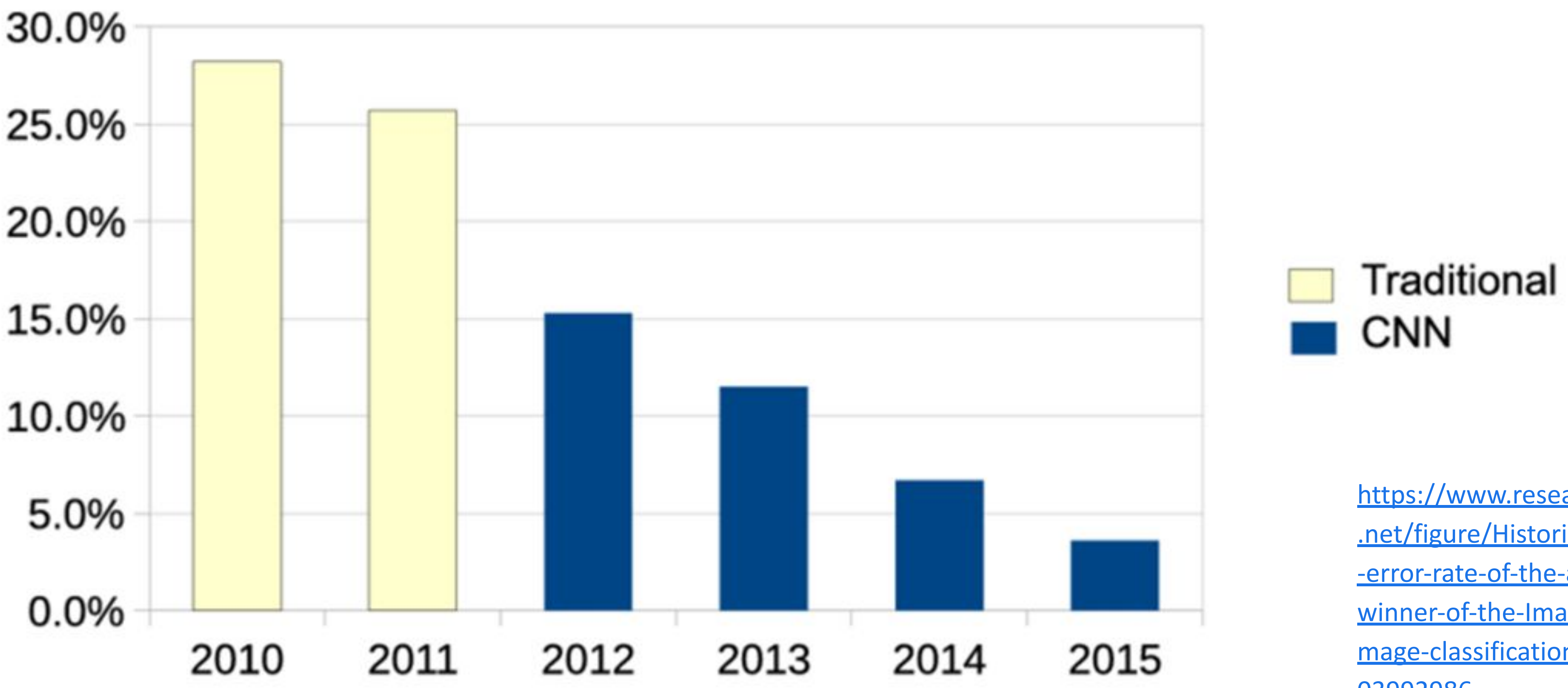
AlexNet



Use convolutions to find features and the summarize them into higher level features

Combine the features to classify the various objects in the dataset

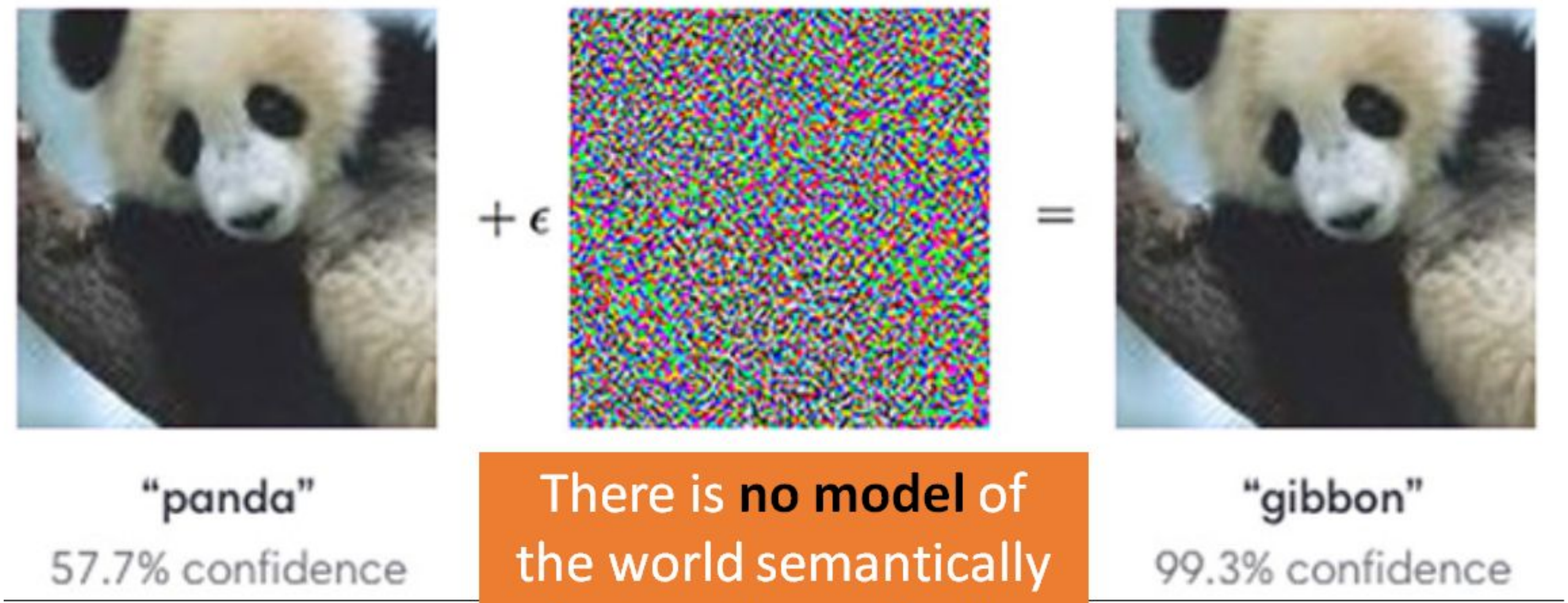
The ImageNet Challenge and the birth of CNNs



https://www.researchgate.net/figure/Historical-top5-error-rate-of-the-annual-winner-of-the-ImageNet-image-classification_fig7_303992986

A word of caution...

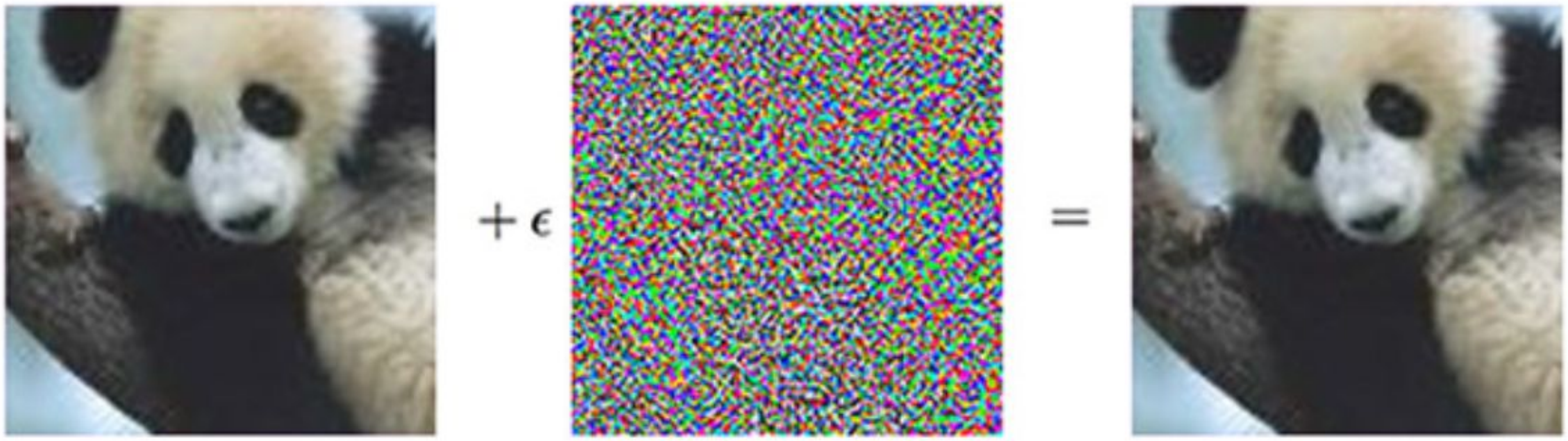
Ackerman "Hacking the Brain With Adversarial Images"



There is **no model** of the world semantically just mathematically https://www.vox.com/future-perfect/2019/4/8/18297410/ai-tesla-self-driving-cars-adversarial-machine-learning

A word of caution...

Ackerman “Hacking the Brain With Adversarial Images”

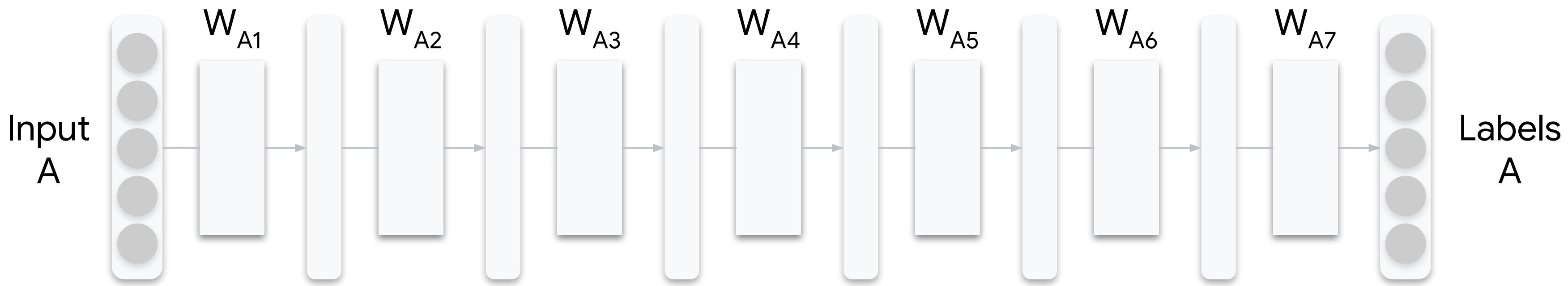


“panda”
57.7% confidence

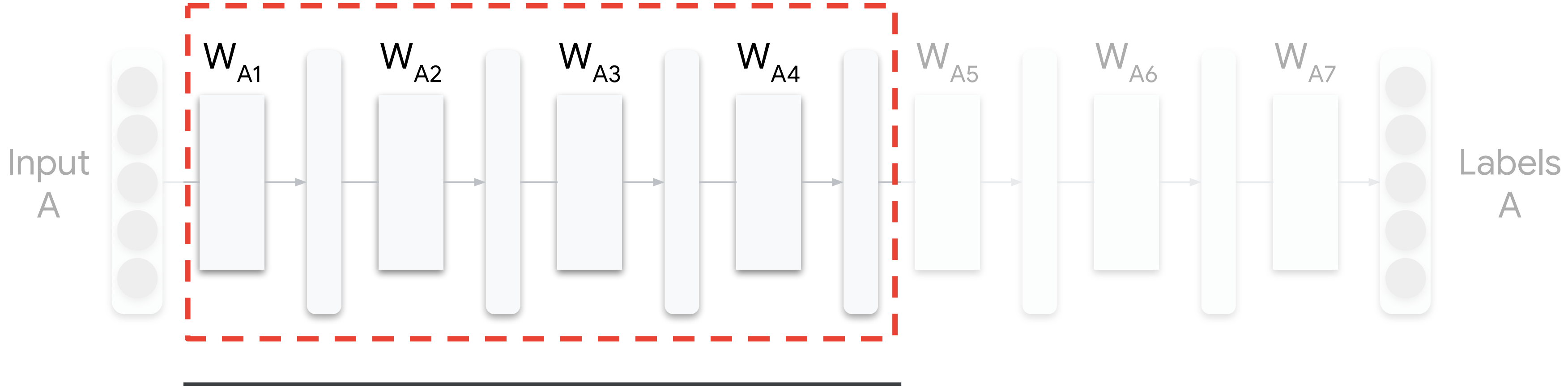
There is **no model** of the world semantically just mathematically

“gibbon”
99.3% confidence

Transfer Learning: Saving time and computational resources

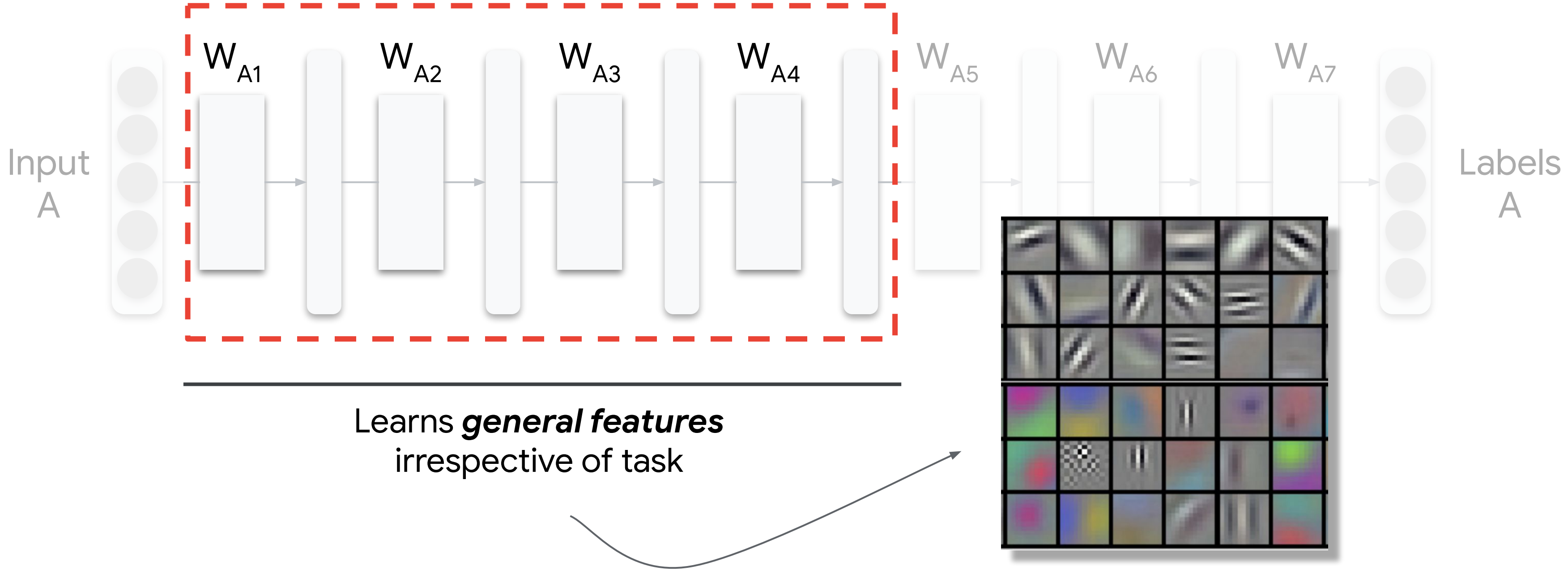


Transfer Learning: Saving time and computational resources

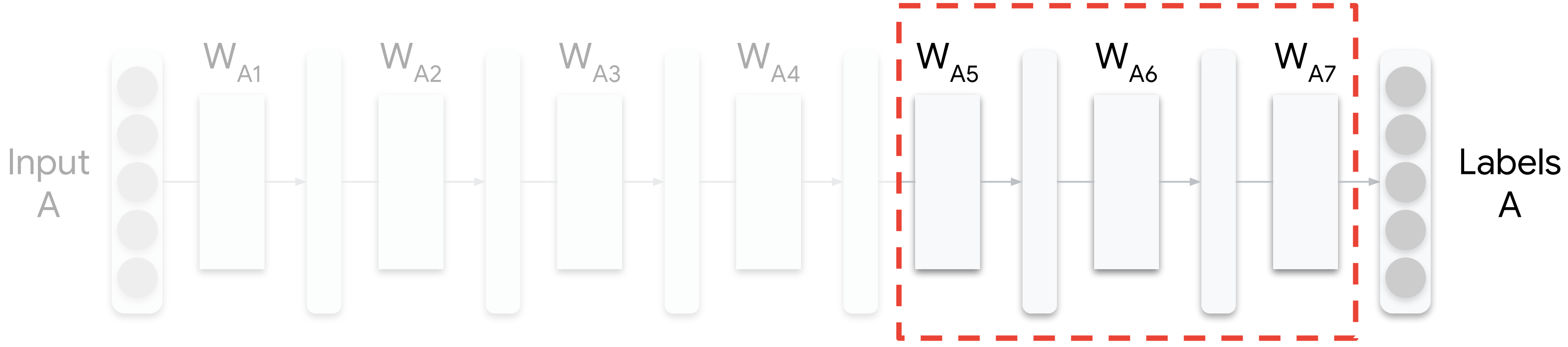


Learns ***general features***
irrespective of task

Transfer Learning: Saving time and computational resources

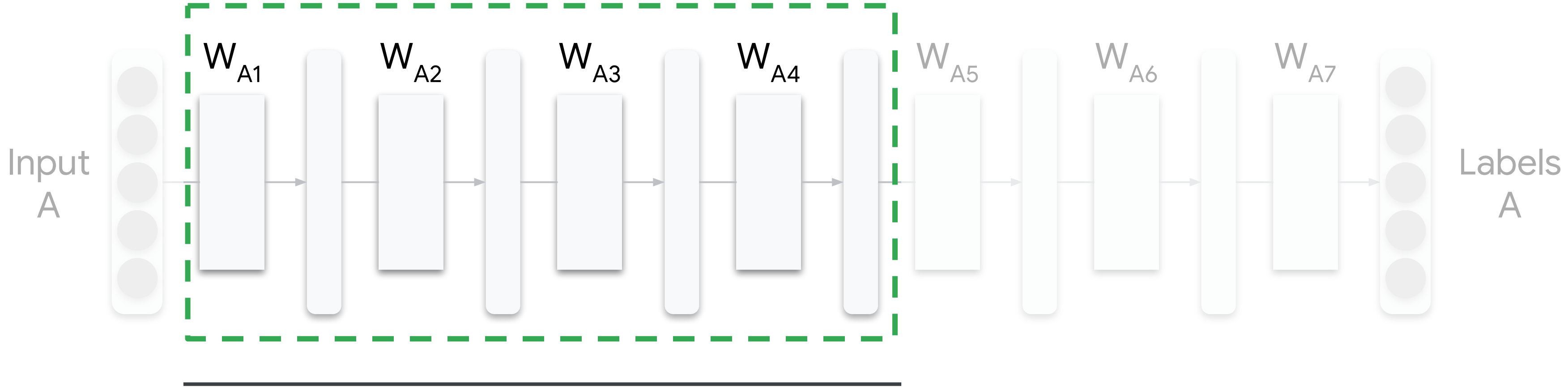


Transfer Learning: Saving time and computational resources



Task-specific features

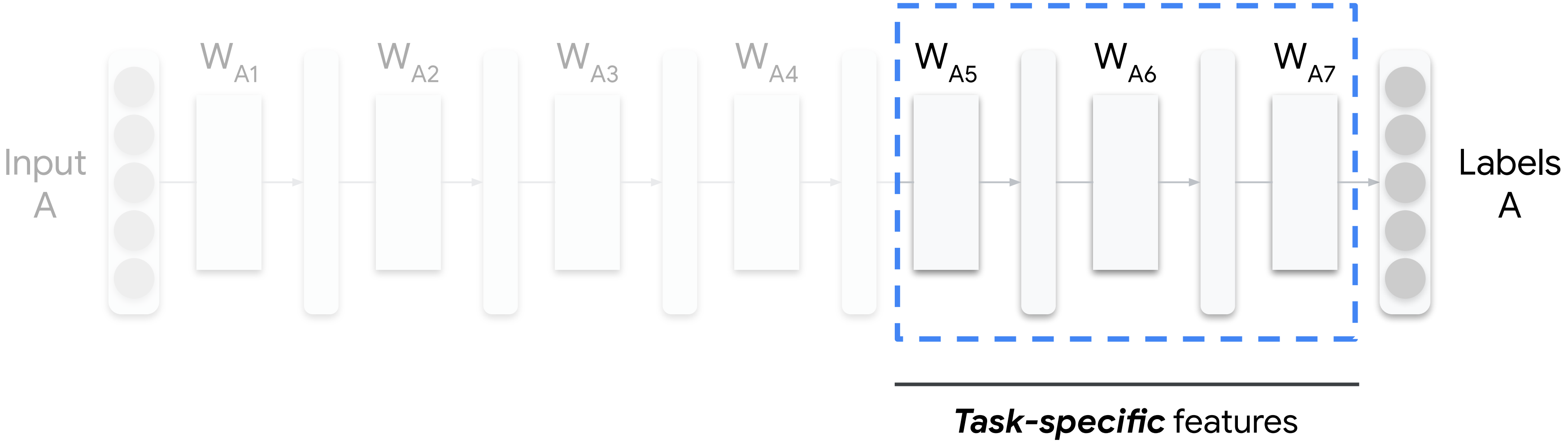
Transfer Learning: Saving time and computational resources



Learns ***general features*** irrespective of task

Reuse (freeze general feature extraction)

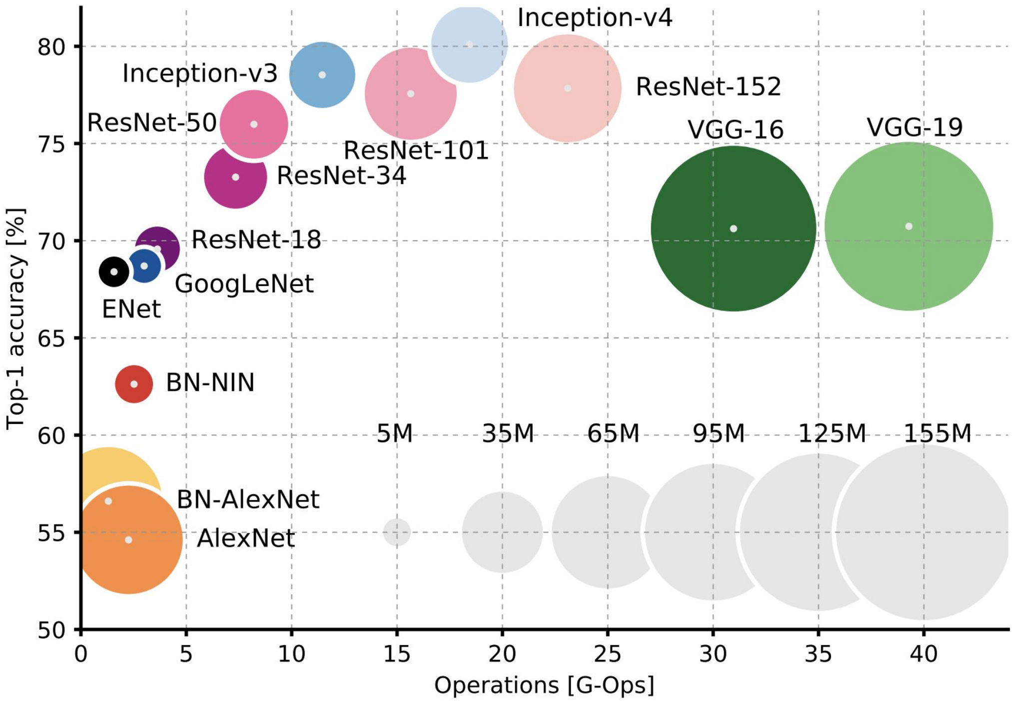
Transfer Learning: Saving time and computational resources



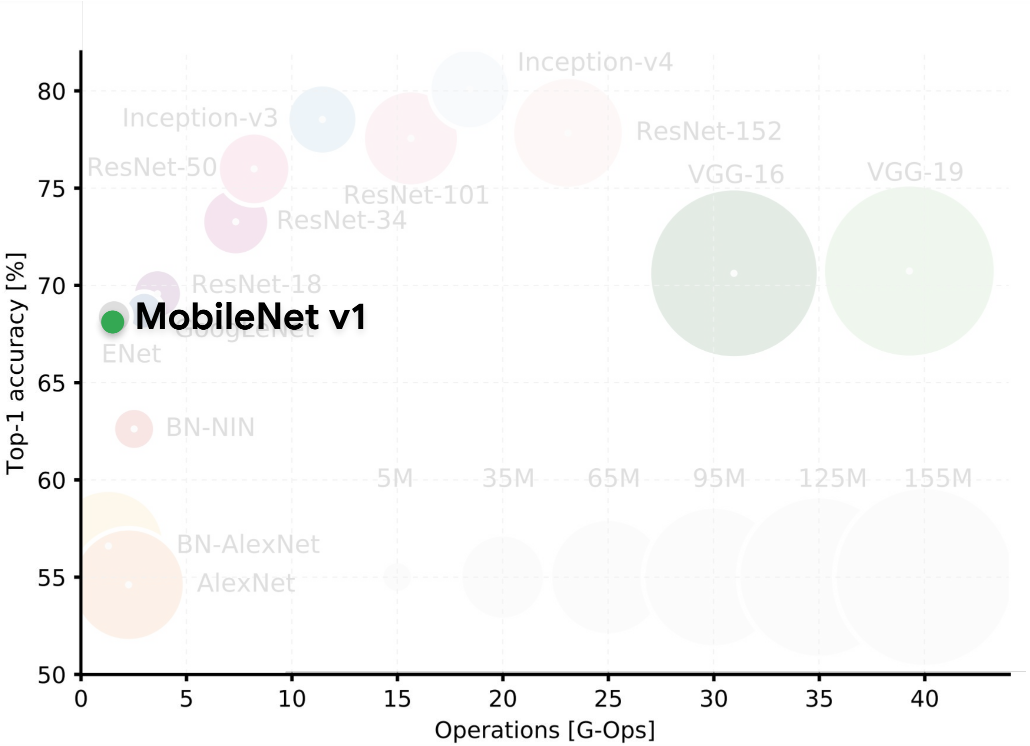
Train only last few layers

So **what model** should we transfer from?

Model **Evolution**



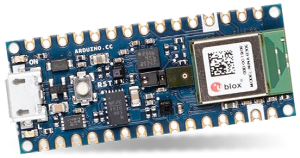
Model **Evolution**



MobileNet v1

Model	Size	Top-1 Accuracy
MobileNet v1	***16 MB***	0.713

Fine for mobile phones with GB of RAM, but 64X microcontroller RAM



Our board only has **256KB** of RAM (memory)

Further **Optimizations**

Multiply-Accumulates

α	**Image Size**	**MACs** (millions)	**Params** (millions)	**Top-1 Accuracy**
1	224	569	4.24	**70.7**
1	128	186	4.14	64.1
0.75	224	317	2.59	68.4
0.75	128	104	2.59	61.8
0.5	224	150	1.34	64.0
0.5	128	49	1.34	56.2
0.25	224	41	0.47	50.6
0.25	128	14	0.47	41.2

Further **Optimizations**

Multiply-Accumulates

α	Image Size	MACs (millions)	Params (millions)	Top-1 Accuracy
1	224	569	4.24	**70.7**
1	128	186	4.14	64.1
0.75	224	317	2.59	**68.4**
0.75	128	104	2.59	61.8
0.5	224	150	1.34	64.0
0.5	128	49	1.34	56.2
0.25	224	41	0.47	50.6
0.25	128	14	0.47	41.2

Further **Optimizations**

Multiply-Accumulates

α	**Image Size**	**MACs** (millions)	**Params** (millions)	**Top-1 Accuracy**
1	224	569	**4.24**	**70.7**
1	128	186	4.14	64.1
0.75	224	317	**2.59**	**68.4**
0.75	128	104	2.59	61.8
0.5	224	150	1.34	64.0
0.5	128	49	1.34	56.2
0.25	224	41	0.47	50.6
0.25	128	14	0.47	41.2

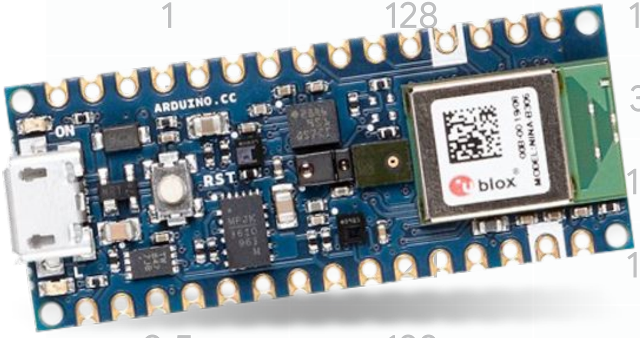
Further **Optimizations**

Multiply-Accumulates

α	**Image Size**	**MACs** (millions)	**Params** (millions)	**Top-1 Accuracy**
1	**224**	**569**	**4.24**	**70.7**
1	**128**	**186**	**4.14**	**64.1**
0.75	224	317	2.59	68.4
0.75	128	104	2.59	61.8
0.5	224	150	1.34	64.0
0.5	128	49	1.34	56.2
0.25	224	41	0.47	50.6
0.25	128	14	0.47	41.2

Further **Optimizations**

α	Image Size	MACs (millions)	Params (millions)	Top-1 Accuracy
1	224	569	4.24	70.7
1	128	186	[illegible]	[illegible]
[illegible]	[illegible]	317	[illegible]	[illegible]
[illegible]	[illegible]	104	[illegible]	[illegible]
[illegible]	[illegible]	150	[illegible]	[illegible]
0.5	128	49	1.34	56.2
0.25	224	41	0.47	50.6
0.25	128	14	0.47	41.2

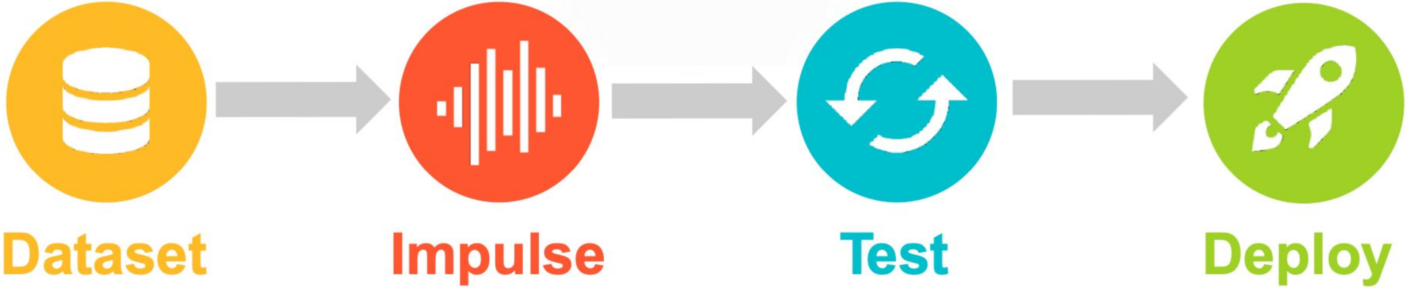


We will need to **both** reduce alpha and the image size!

The TinyML Workflow

Camera feed

```
Starting inferencing in 2 seconds...
Taking photo...
Predictions (DSP: 9 ms., Classification:
    car: 0.07812
    truck: 0.92188
```



Latam Regional Workshop on SciTinyML: Scientific Use of Machine Learning on Low-Power Devices

11-15 July 2022
Online

Further information:
https://tinyMLedu.org/SciTinyML
edu@tinyML.org

Convolutions and Transfer Learning for Computer Vision

Brian Plancher
Barnard College, Columbia University
Harvard John A. Paulson School of Engineering and Applied Sciences
brianplancher.com