Intro to Neural Nets

Session 4: Working with Image Data (CNNs)

Session Agenda

Convolutional Neural Networks (CNNs)

- What CNNs try to accomplish
- What is a convolution?
  - Padding, strides, filters
- What is pooling?
  - Max, min, avg pooling.

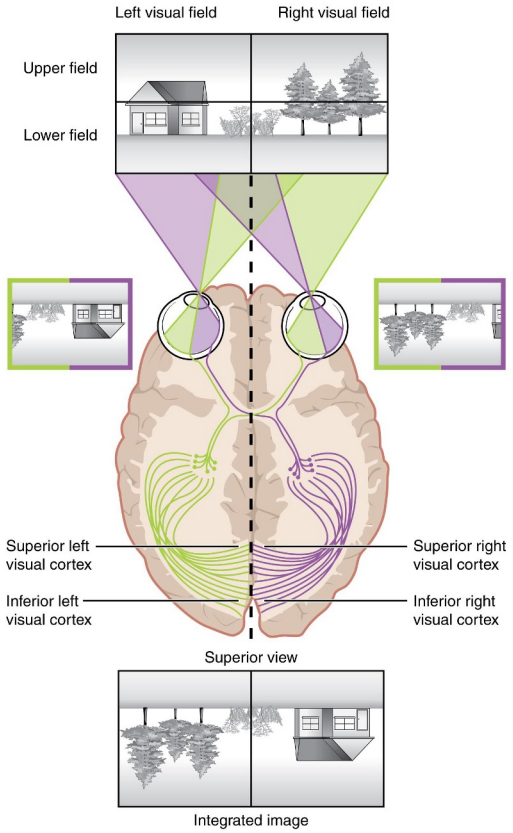
Other Stuff

- CNN specific techniques to avoid overfitting (data augmentation).
- Extracting feature representations from your trained model.
- Adapting pre-trained models (transfer learning).

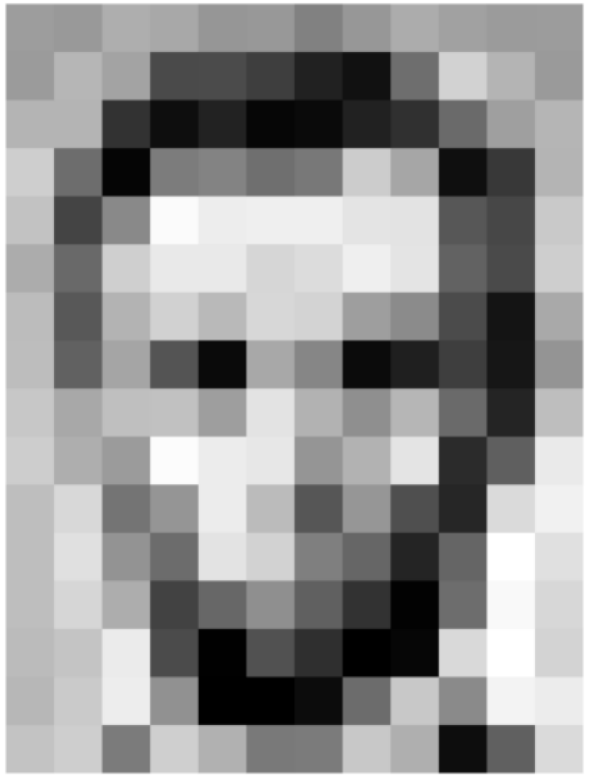
Inspiration for Convnets

Our Visual System

- Human eye is basically a 576-megapixel video camera.
- For comparison, the Pixel 6 camera is 50-megapixels.
- The human field of vision is not a square; something like a video camera that records individual image frames comprised of 24,000 x 24,000 pixels.



Images are Numbers



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

What the computer sees

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

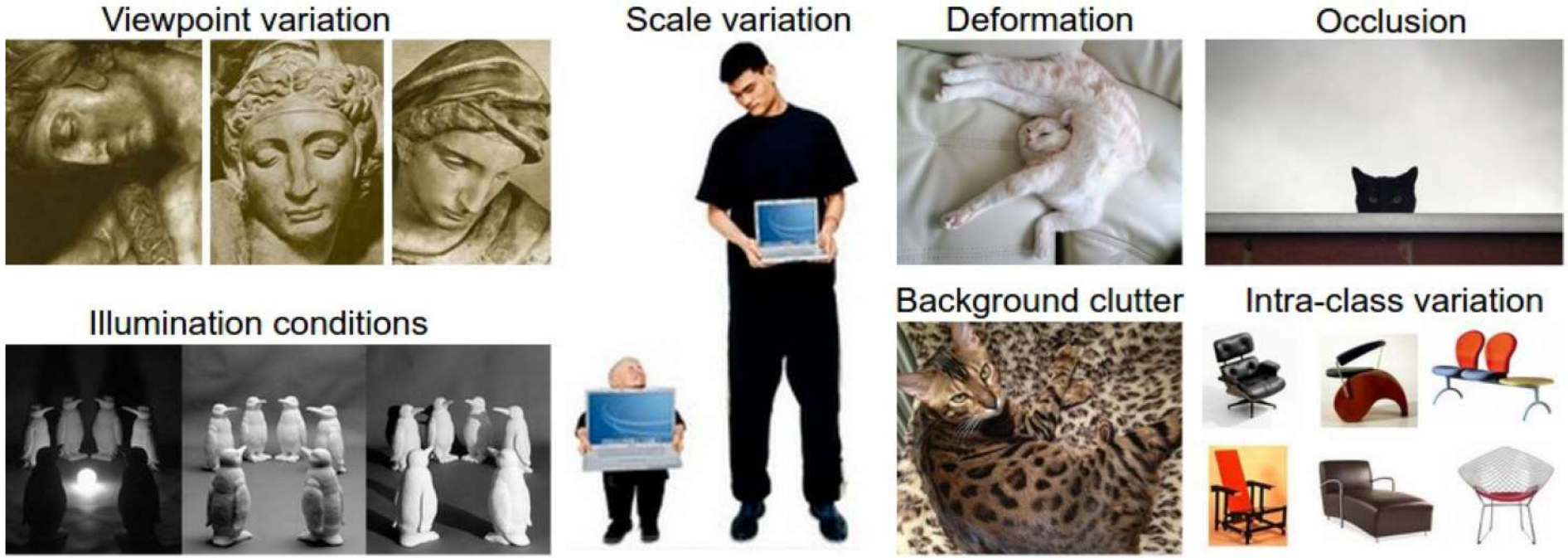
An image is just a matrix of numbers [0,255]! i.e., 1080x1080x3 for an RGB image

Manual Feature Extraction

Problems?

Manual Feature Extraction

Domain knowledge → Define features → Detect features to classify

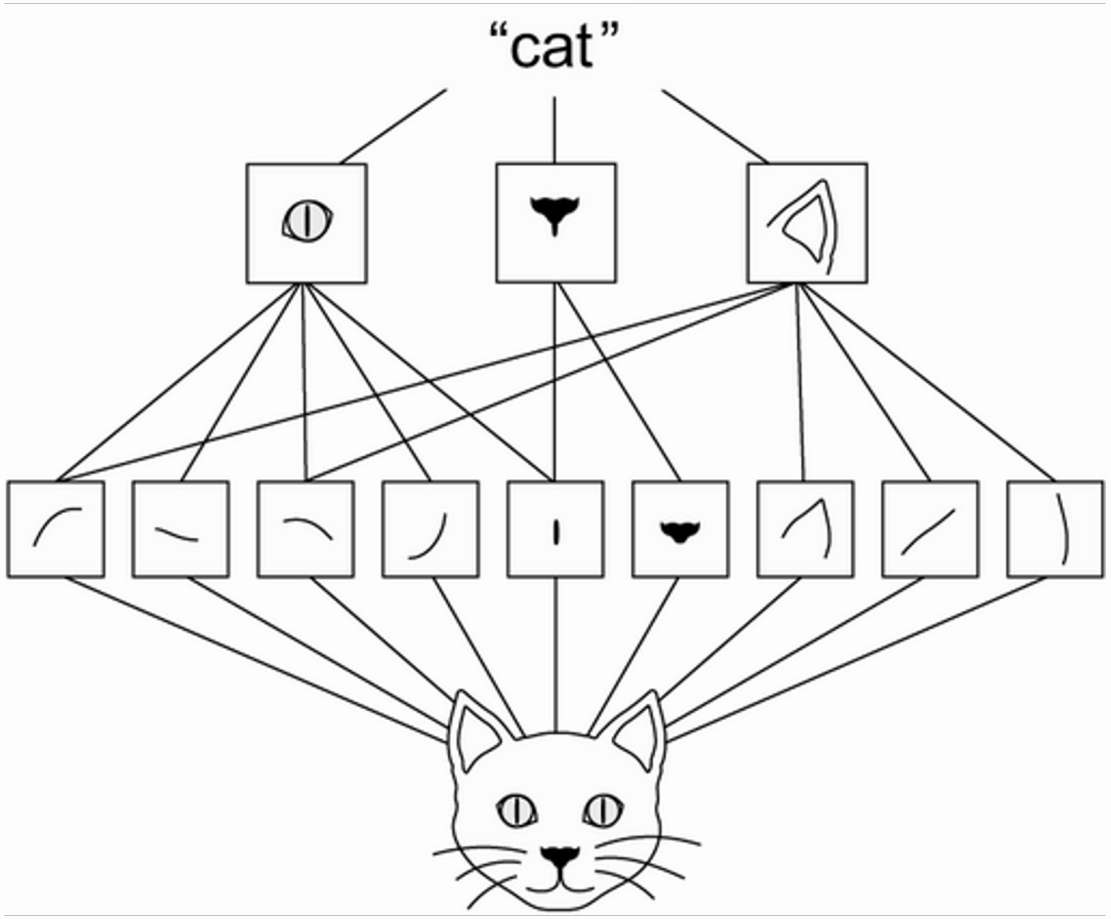


We Don't Detect Features 'Manually' Like That

How Does Your Visual System Work?

- We think that your brain processes individual visual receptors in groups, identifies combinations of inputs in proximity to one another that imply something like an edge (edge detection), combines that with color and so on. These low-level features are then processed together to arrive at higher level objects (e.g., a nose, a mouth, an eye).
- Those higher-level features are then processed together to yield a face (perhaps someone we know or do not know). Hence why you might have a hard time recognizing someone who has a new haircut, or who is wearing a facemask!

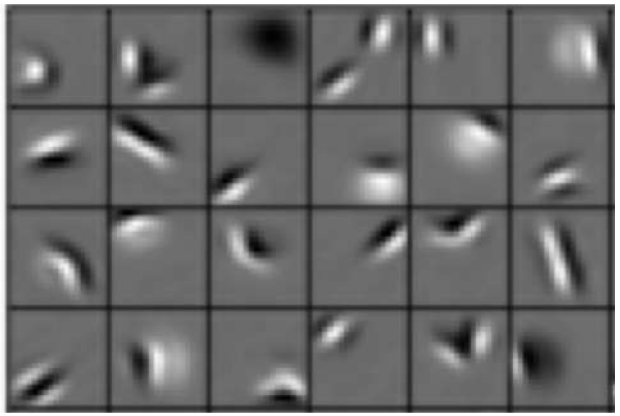
Feature Detection / Aggregation



Learning Feature Representations

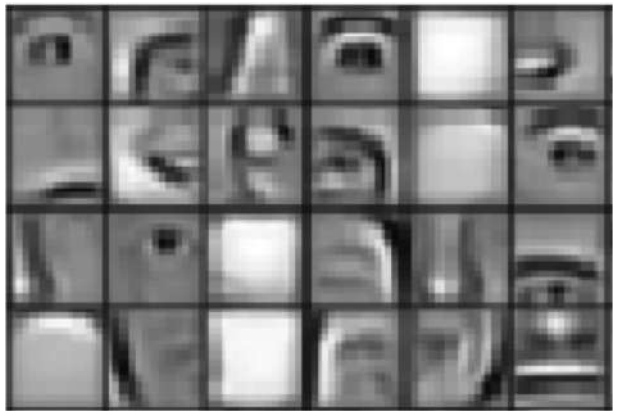
Can we learn a **hierarchy of features** directly from the data instead of hand engineering?

Low level features



Edges, dark spots

Mid level features

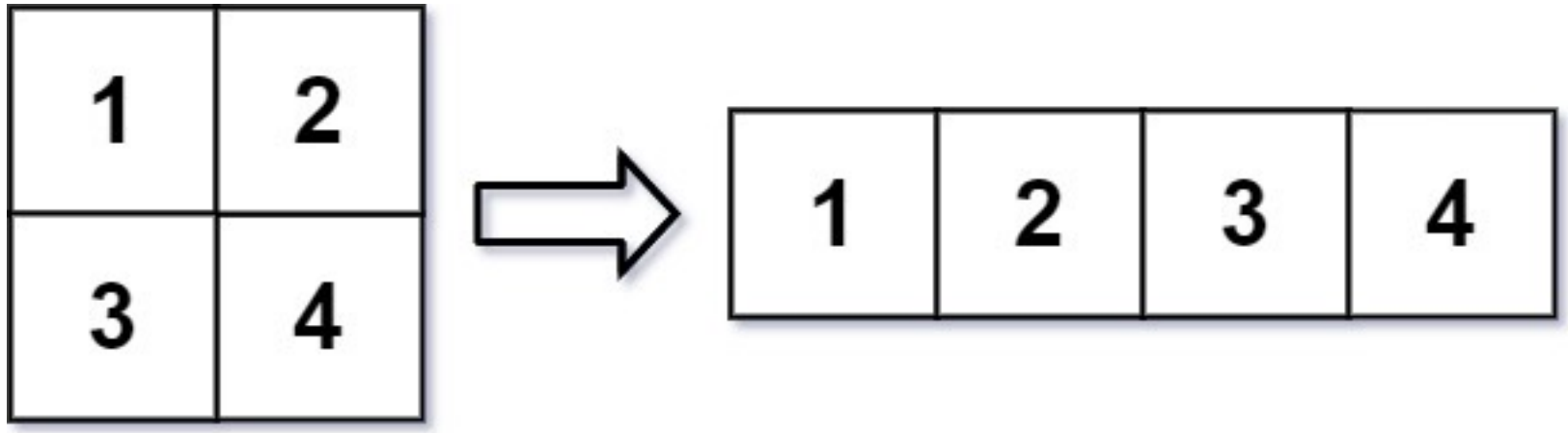


Eyes, ears,nose

High level features

Facial structure

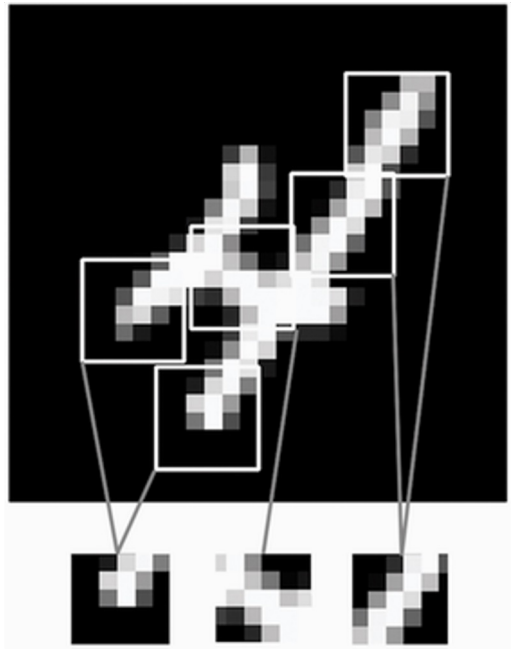
Why are Fully Connected Dense Layers Not Good for This?



We Need to Preserve Spatial Arrangements

Hone-in On Sub-sections of the Image

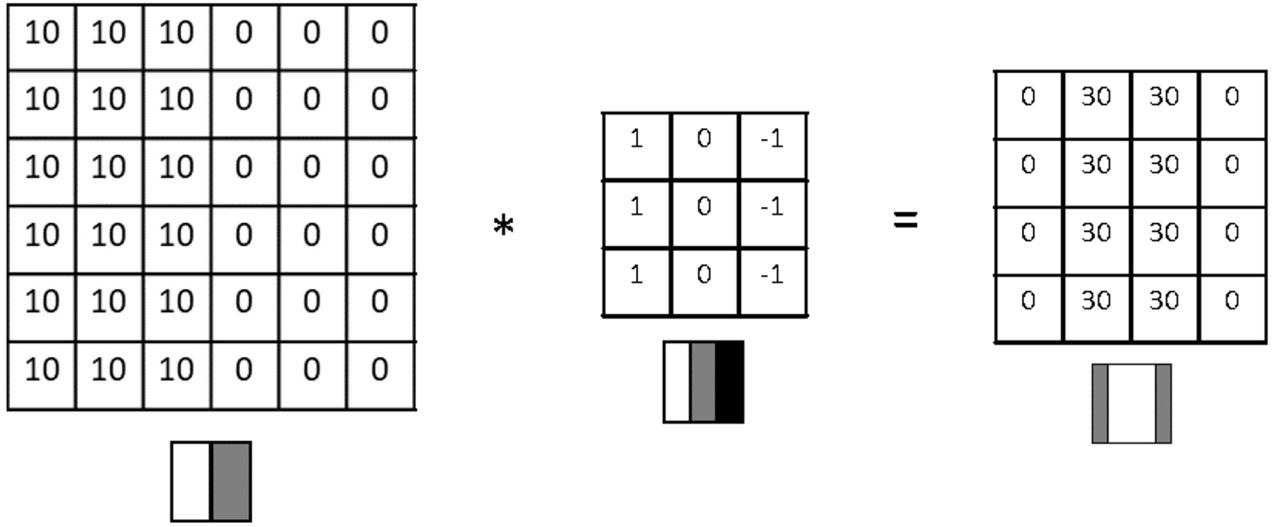
- So, if we have a 28x28 image, we might separately consider 3x3 pixel subsection of that image. Each subsection (they can be overlapping) is represented by its own node in the first hidden layer.
- That local input matrix (subfield) is considered in tandem with a ‘filter’ a matrix of weights. A filter might be something like



The Convolution Operation

Consider in Matrix Representation

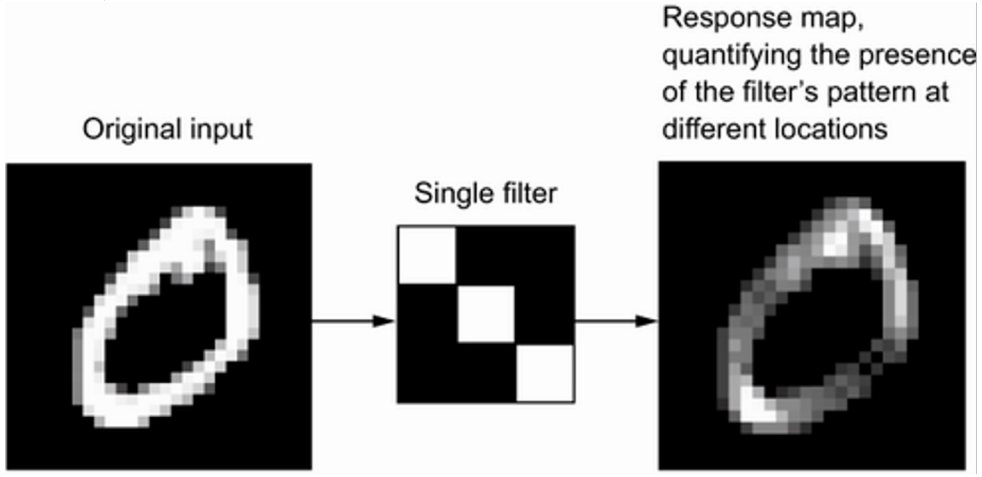
- We have the raw image data (a.k.a. input feature map), the filter, and the result of passing our filter over our image (a.k.a. output feature map). We will have one output feature map for a given image, per filter (each filter is intended to detect a different type of feature).
- The filter elements are just weights for the Conv layer; we learn the filter values as part of the backpropagation process. So, the CNN will figure out what features to look for to predict the label (probably what a baby does when its first board and first learning how to process visual information).



The Convolution Operation

Consider in Matrix Representation

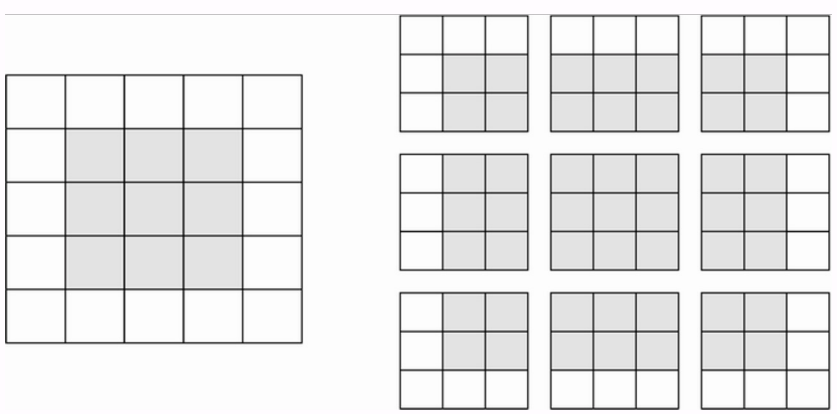
- We have the raw image data (a.k.a. input feature map), the filter, and the result of passing our filter over our image (a.k.a. output feature map). We will have one output feature map for a given image, per filter (each filter is intended to detect a different type of feature).
- The filter elements are just weights for the Conv layer; we learn the filter values as part of the backpropagation process. So, the CNN will figure out what features to look for to predict the label (probably what a baby does when its first board and first learning how to process visual information).

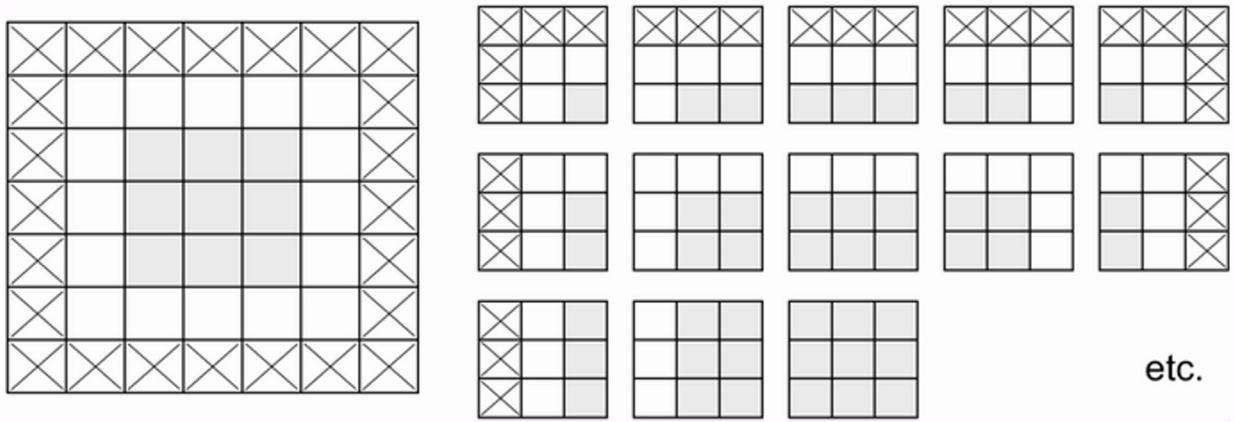


Padding

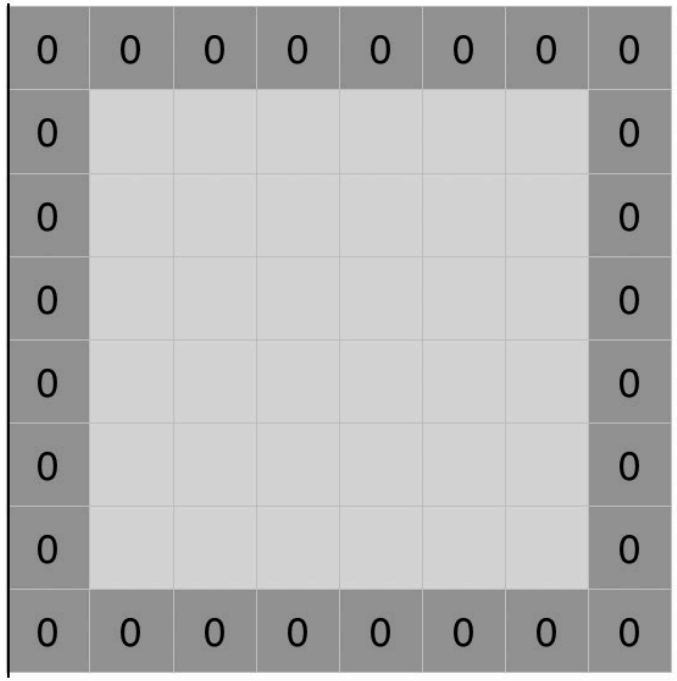
Padding

- To prevent the transformation from down-sampling (reducing the size of the matrix during convolution to output), we can pad the edges of the image with 0's.

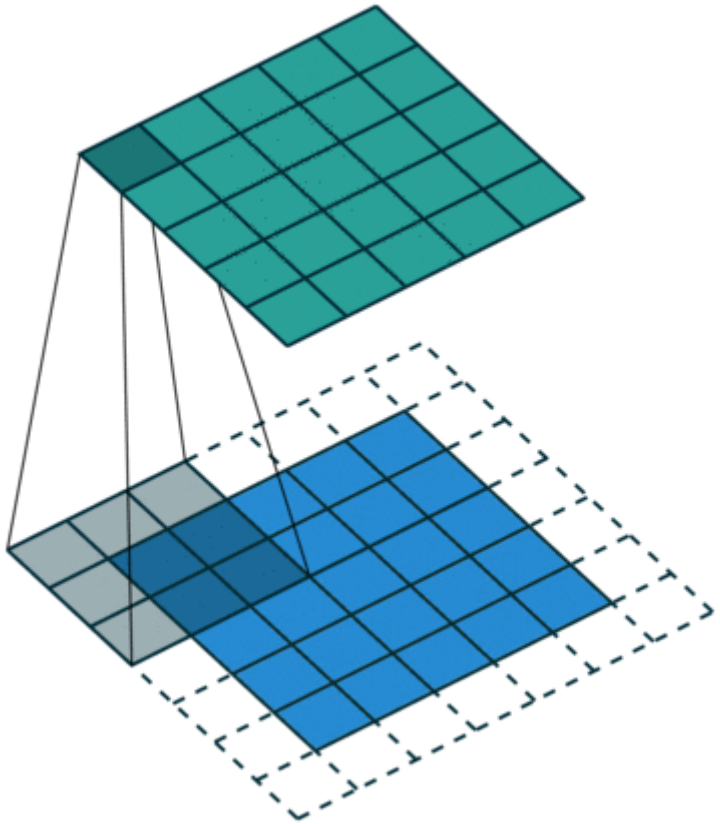




Padding Technique



Zero-padding added to image



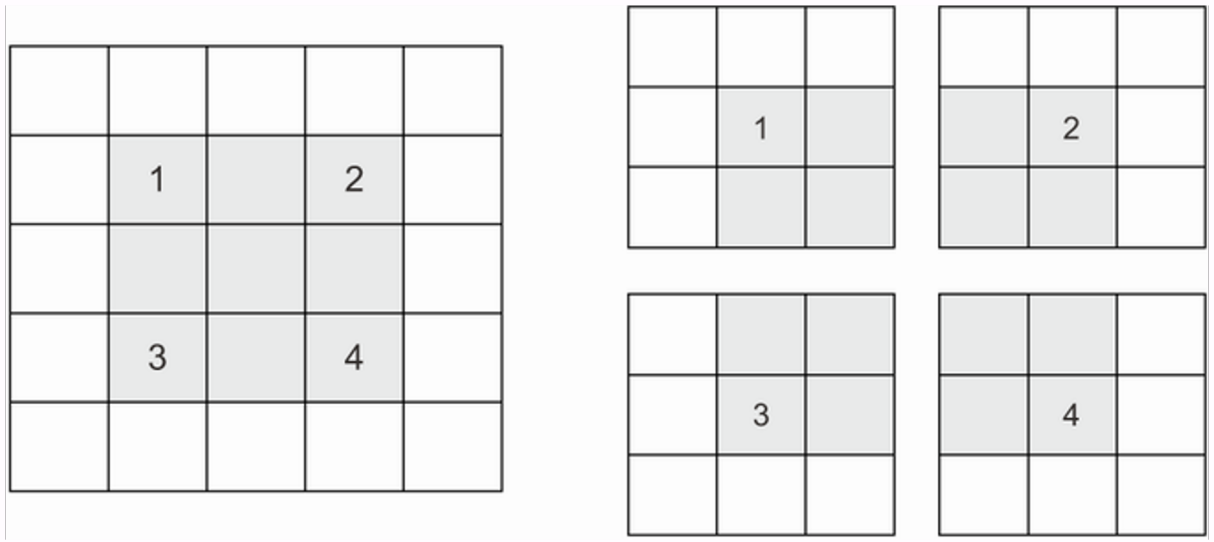
Add appropriate padding to keep the dimension after filtering

$n \times n$ input image with $f \times f$ filters
add $p = (f-1)/2$ padding keep the same dimension of input image

Strides

Strides

- Often, we will pass the filter over every pixel cell, but we don't have to; we might pass over every other cell. This is what strides refers to (skipping).



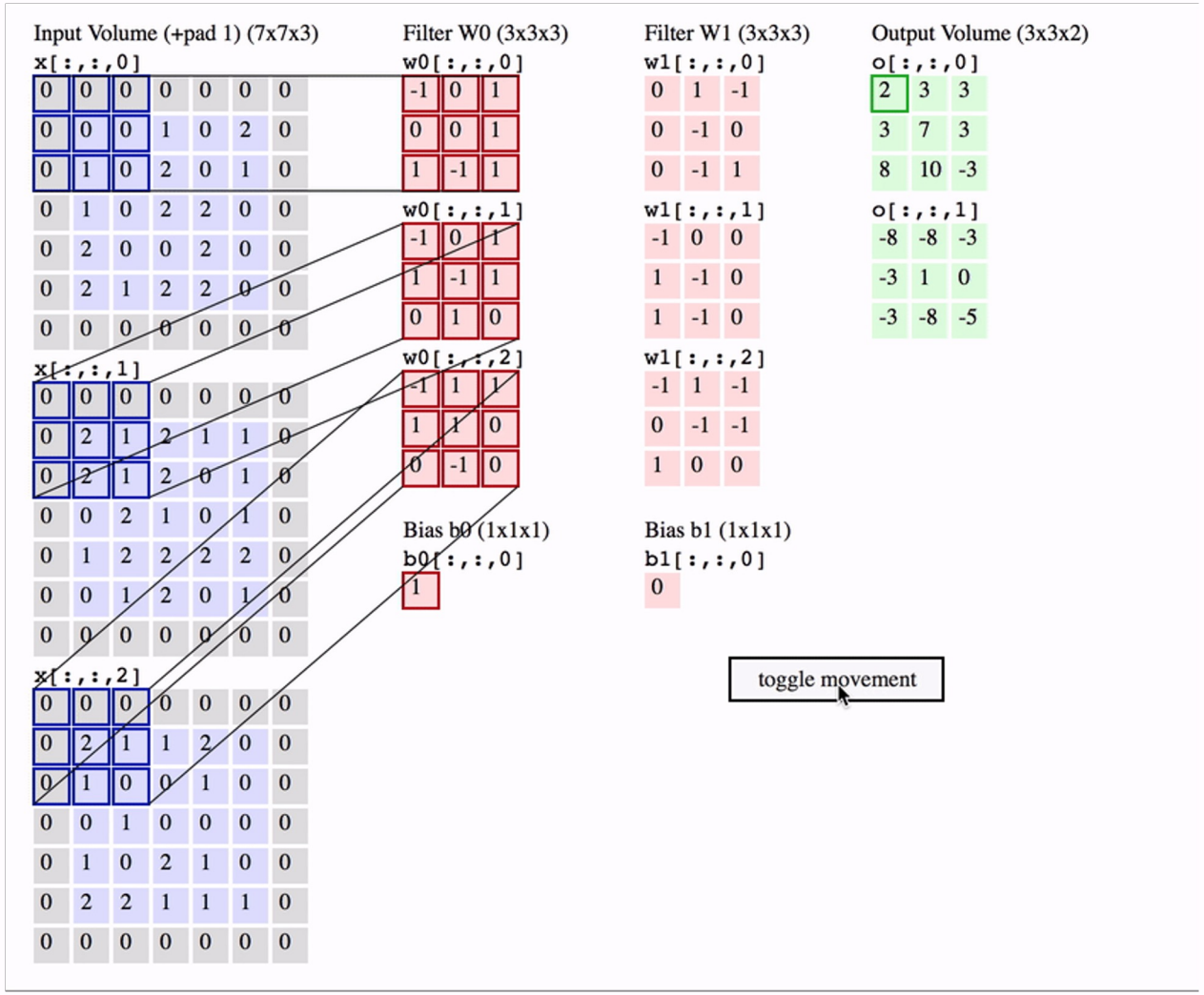
Padding and Strides

Padding

- To prevent the transformation from down-sampling (reducing the size of the matrix during convolution to output), we can pad the edges of the image with 0's.

Strides

- Often, we will pass the filter over every pixel cell, but we don't have to; we might pass over every other cell.
- This is what strides refers to (skipping).



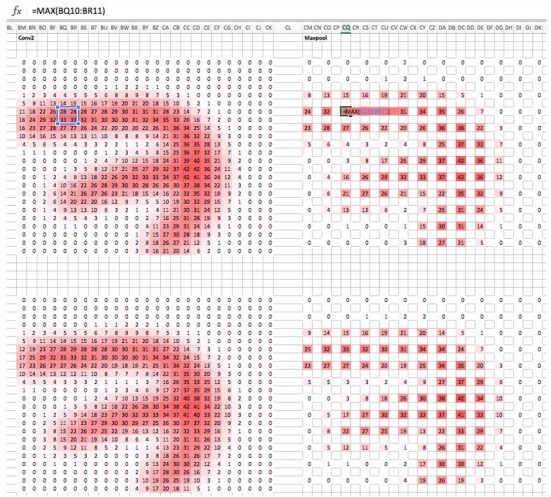
What is Pooling?

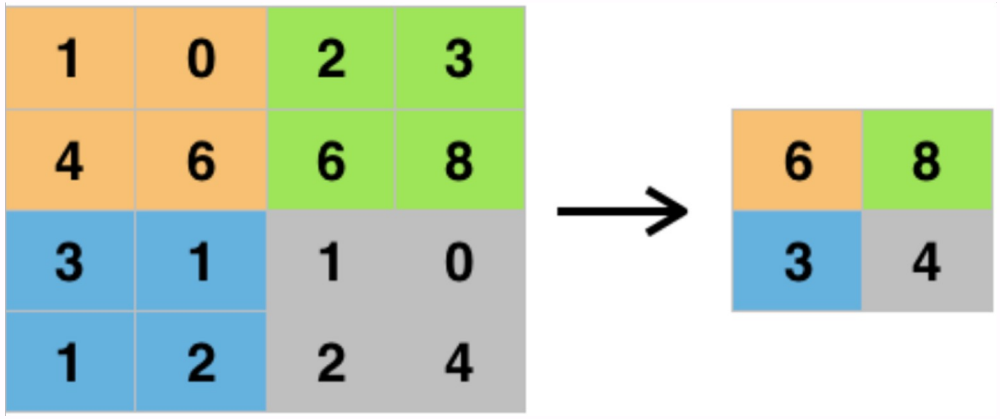
Down-sampling Detected Features

- The idea is to compress the resulting data down into a coarser representation, to reduce model complexity, and to also force attention toward a broader section of the original image (helps reduce overfitting).

Forcing Attention to Larger Blocks of the Original Image

- Because we typically use stride = pool width, the pooling output is aggregating over segments of the input.





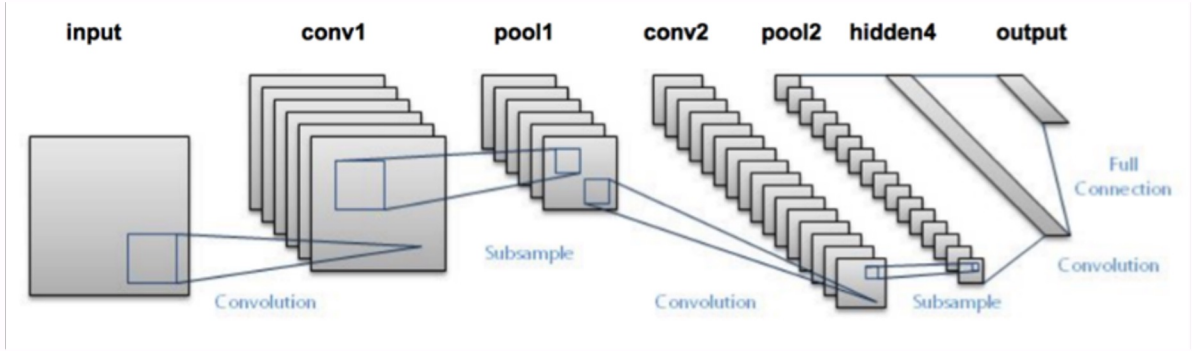
Basic Image Labeling Topology

Convolution Layers Apply (Multiple) Filters

- Each filter is a matrix of trainable weights.
- As you move through the network, the number of features typically rises exponentially.
- More filters as you move along means it allows more permutations / combinations

Progressively Smaller Filter Maps

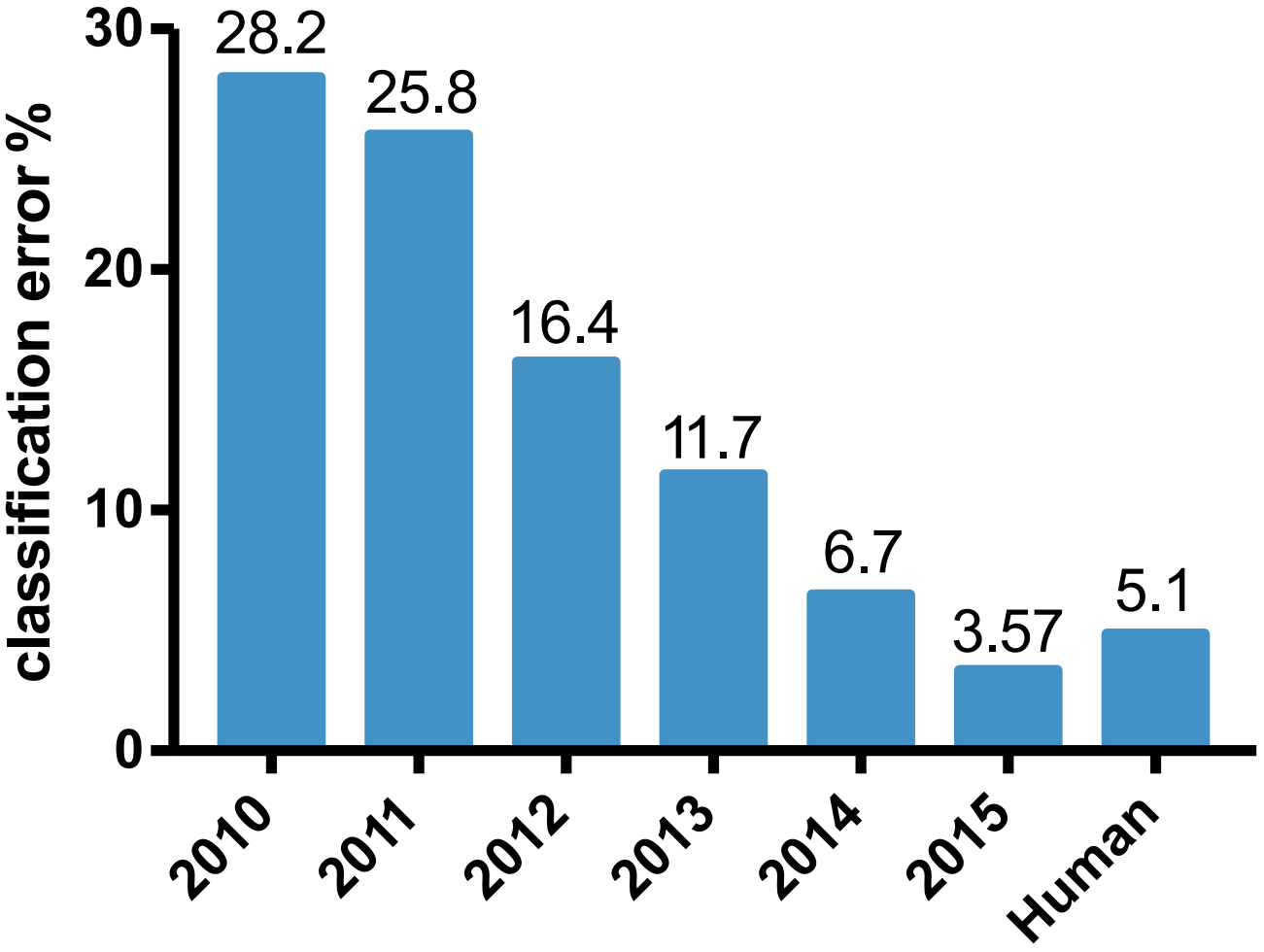
- Smaller filter map arises from the pooling steps, which means that each element of the final map distills features (high level features, derived from low level features, derived from raw pixels) derived from a larger segment of the original picture.



ImageNet Challenge

Classification task: "Top 5 error among 1000 categories": rate at which the model does not output correct label in top 5 predictions

ImageNet Challenge: Classification Task



2012: AlexNet. First CNN to win.
- 8 layers, 61 million parameters

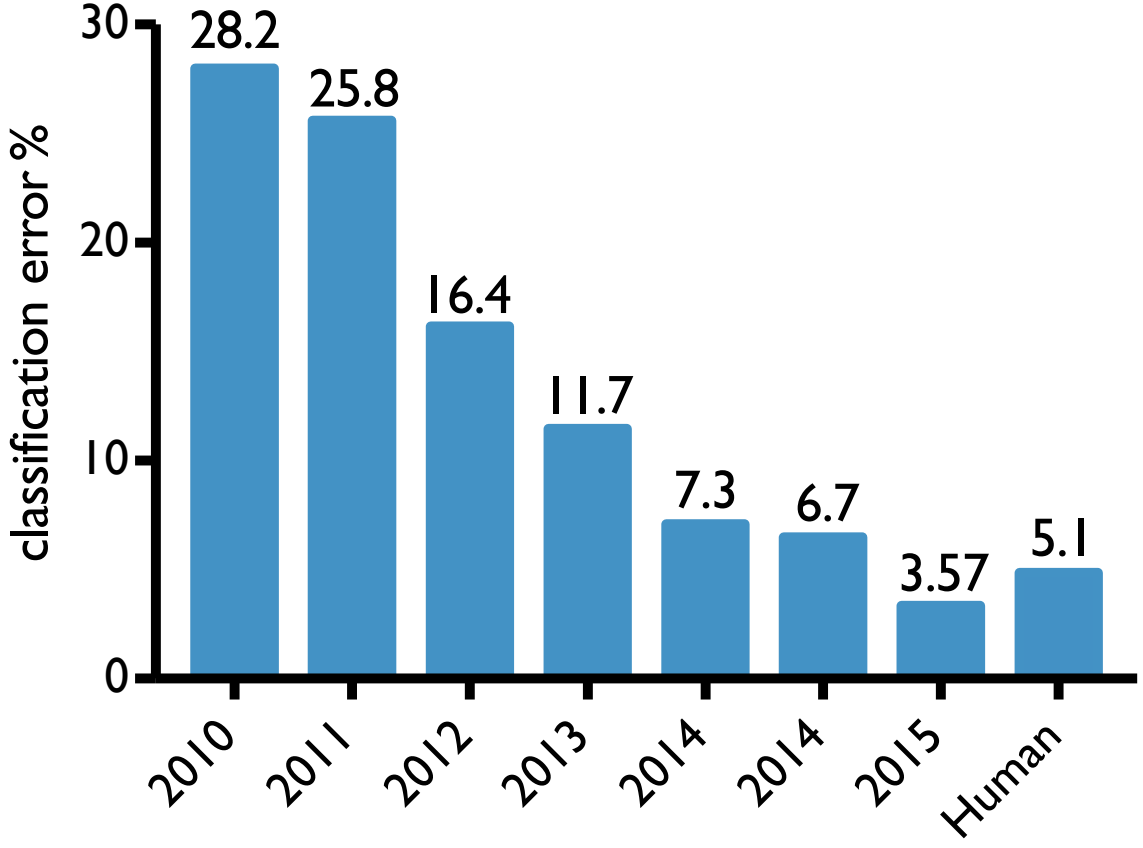
2013: ZFNet
- 8 layers, more filters

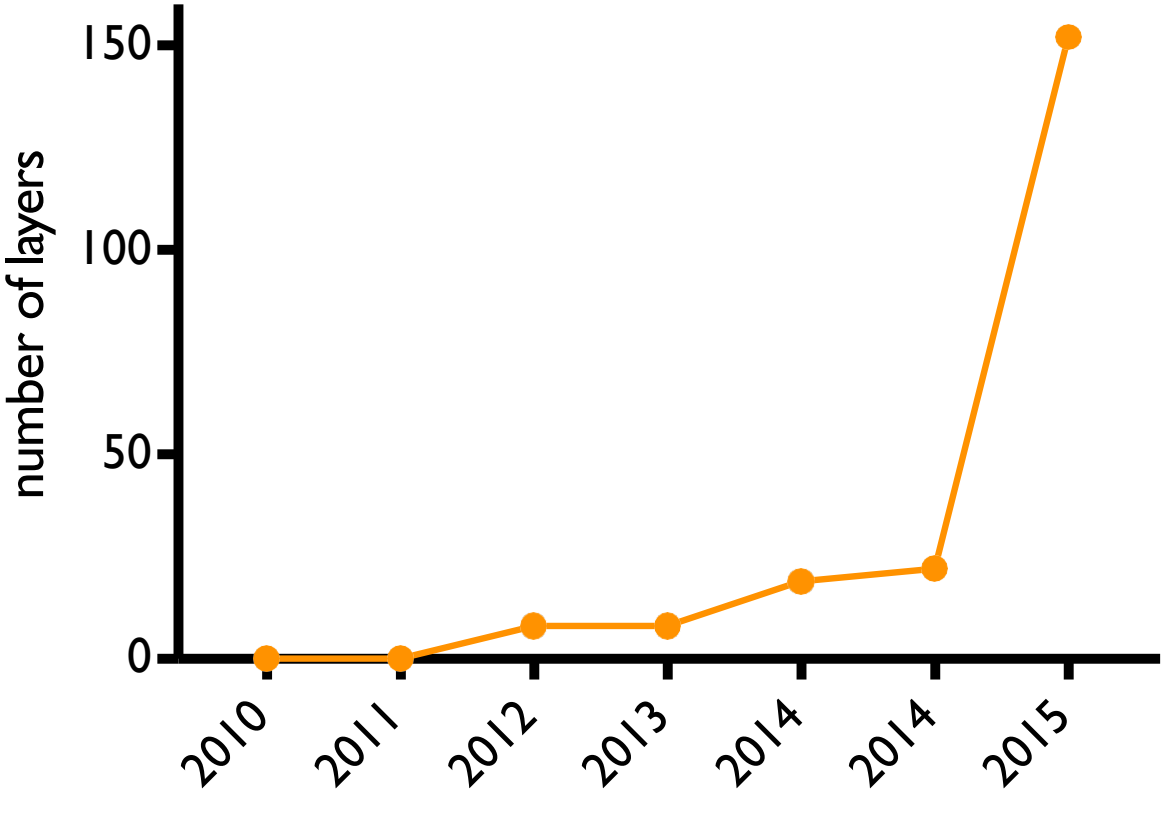
2014: VGG
- 19 layers

2014: GoogLeNet
- "Inception" modules
- 22 layers, 5million parameters

2015: ResNet
- 152 layers

ImageNet Challenge: Classification Task





Computer Vision

6 big tasks

1. *Image and video classification*
2. Object detection
3. Semantic segmentation
4. Motion detection
5. Scene text recognition
6. Visual Q&A

Smartphones

- SenseID mobile face unlock solution
- Emoticon recognition

Smart cars

- License plate recognition
- Gesture recognition
- Road condition detection

Entertainment

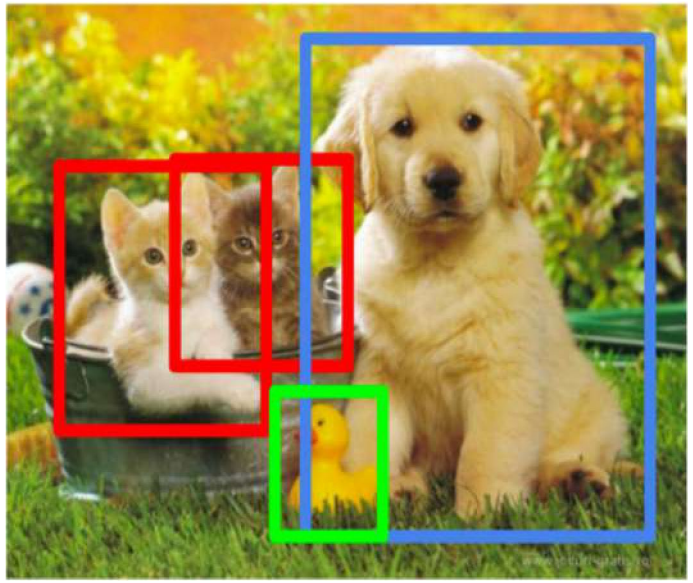
Smart Health

- Food testing
- CT analysis

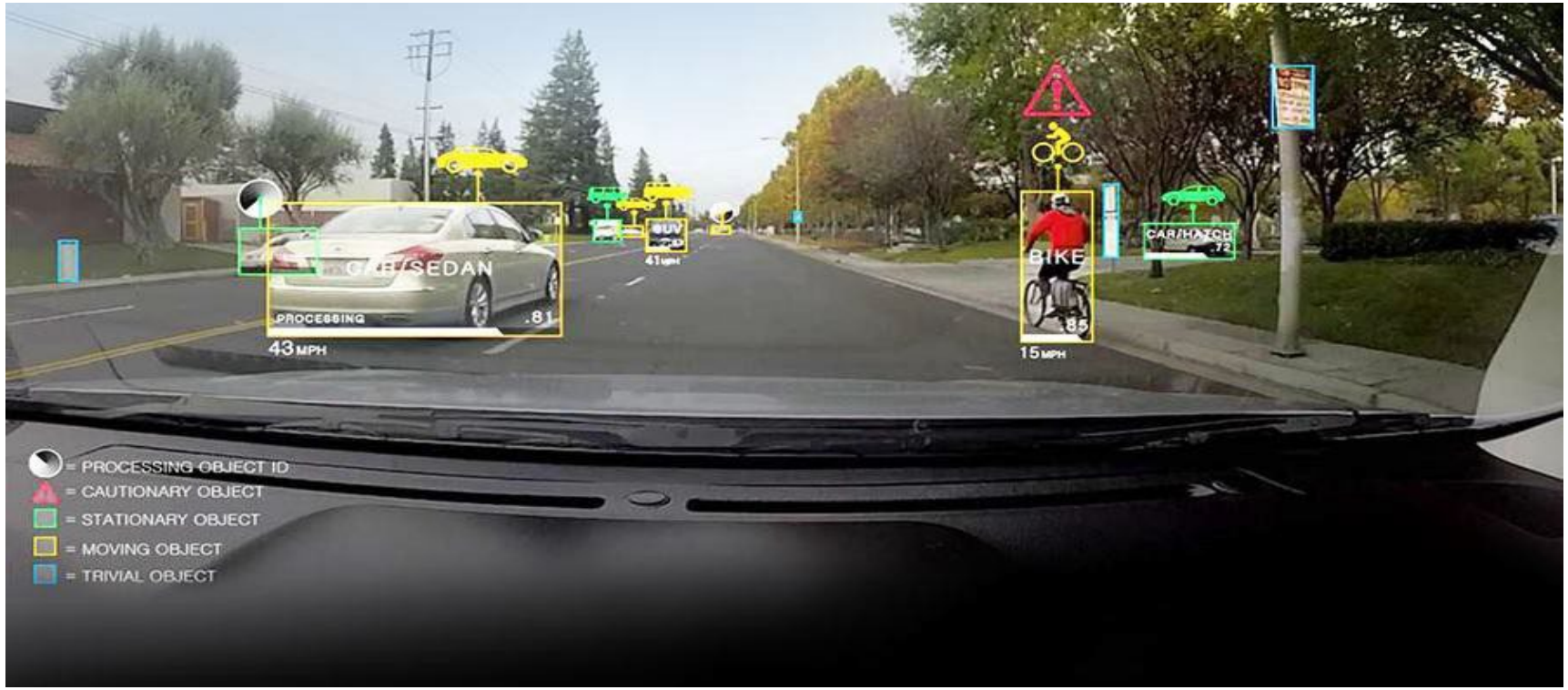
Enterprise business

- Security system

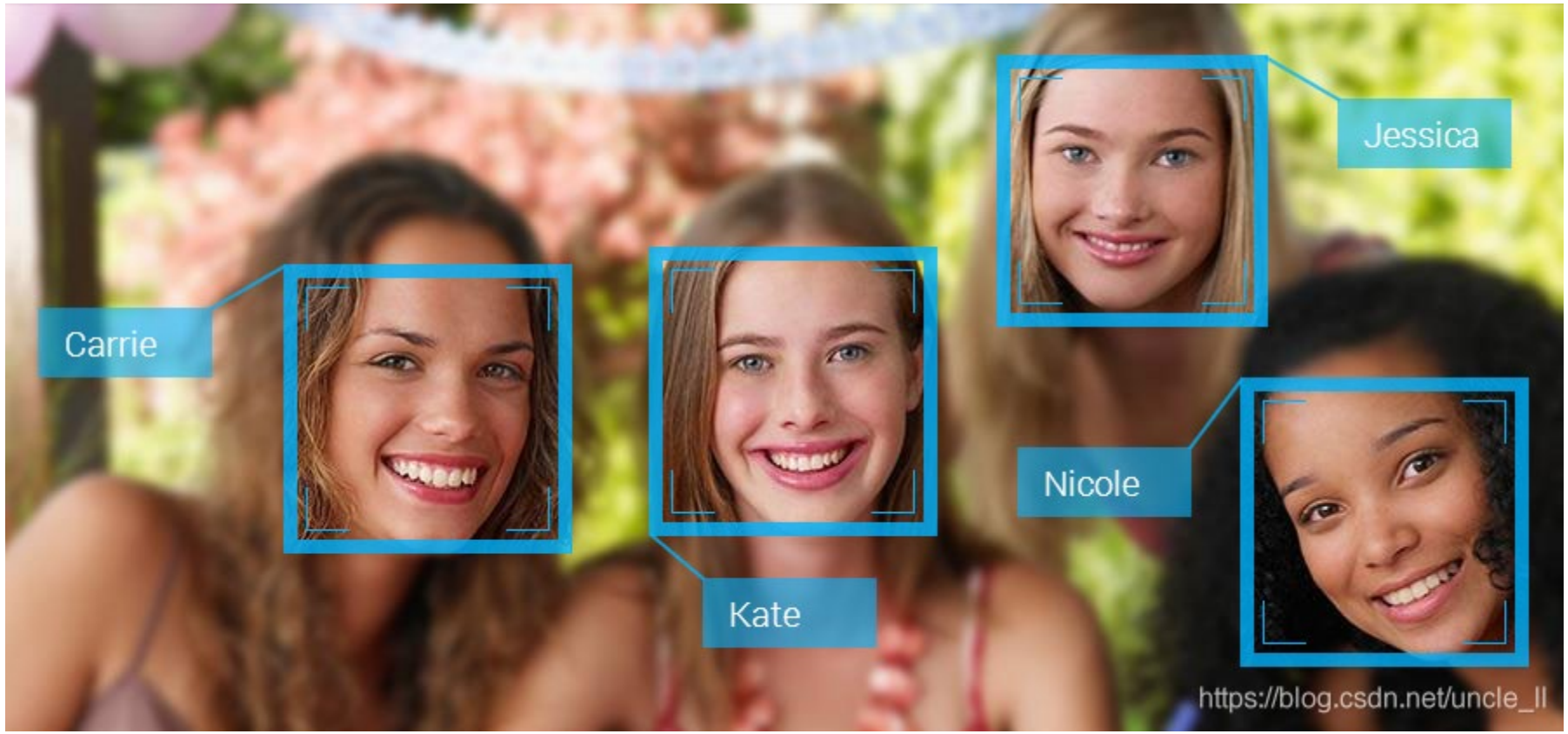
Task 2. Object Detection: The Foundation of Unmanned Vehicles



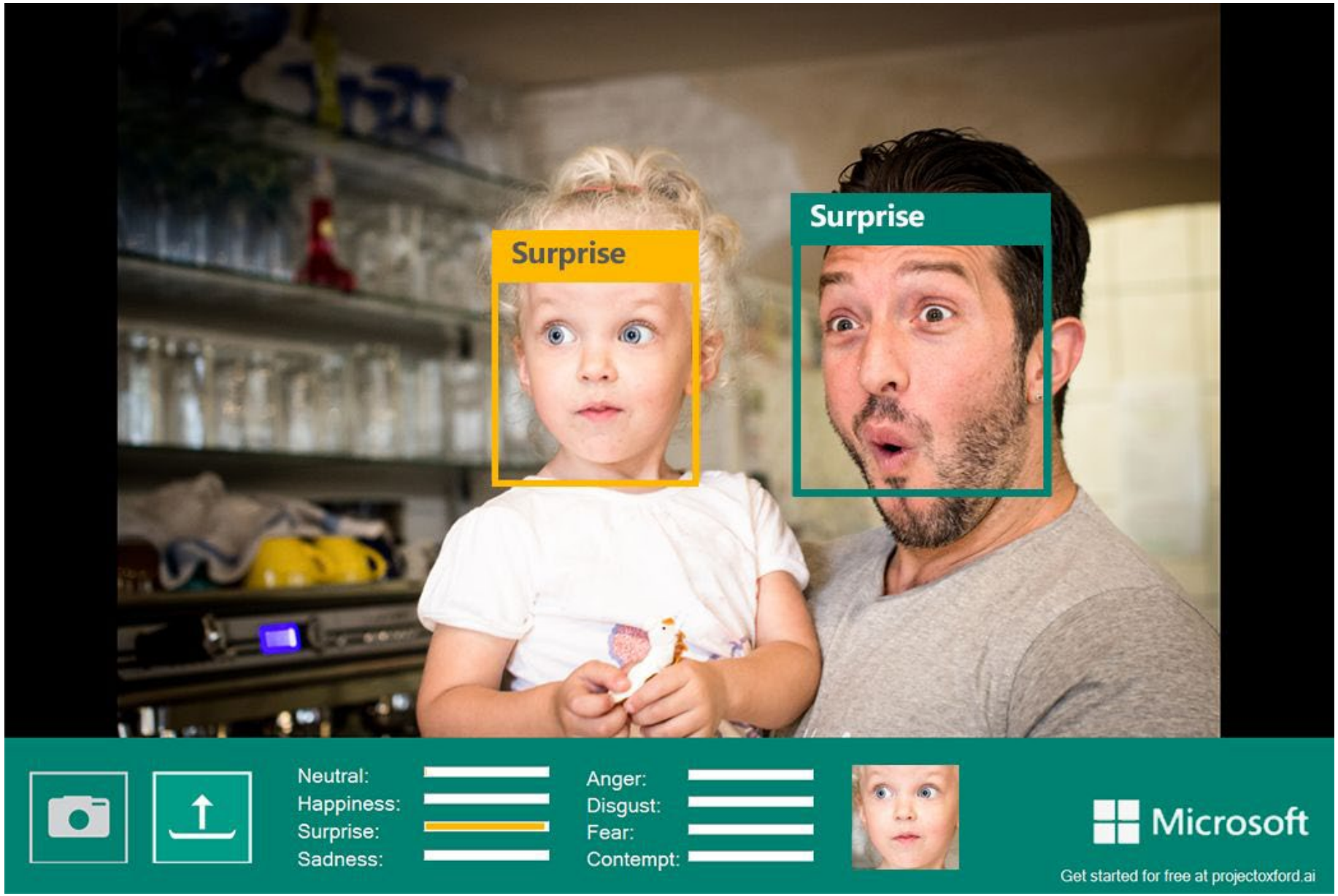
CAT, DOG, DUCK



Face detection: Phone face unlock

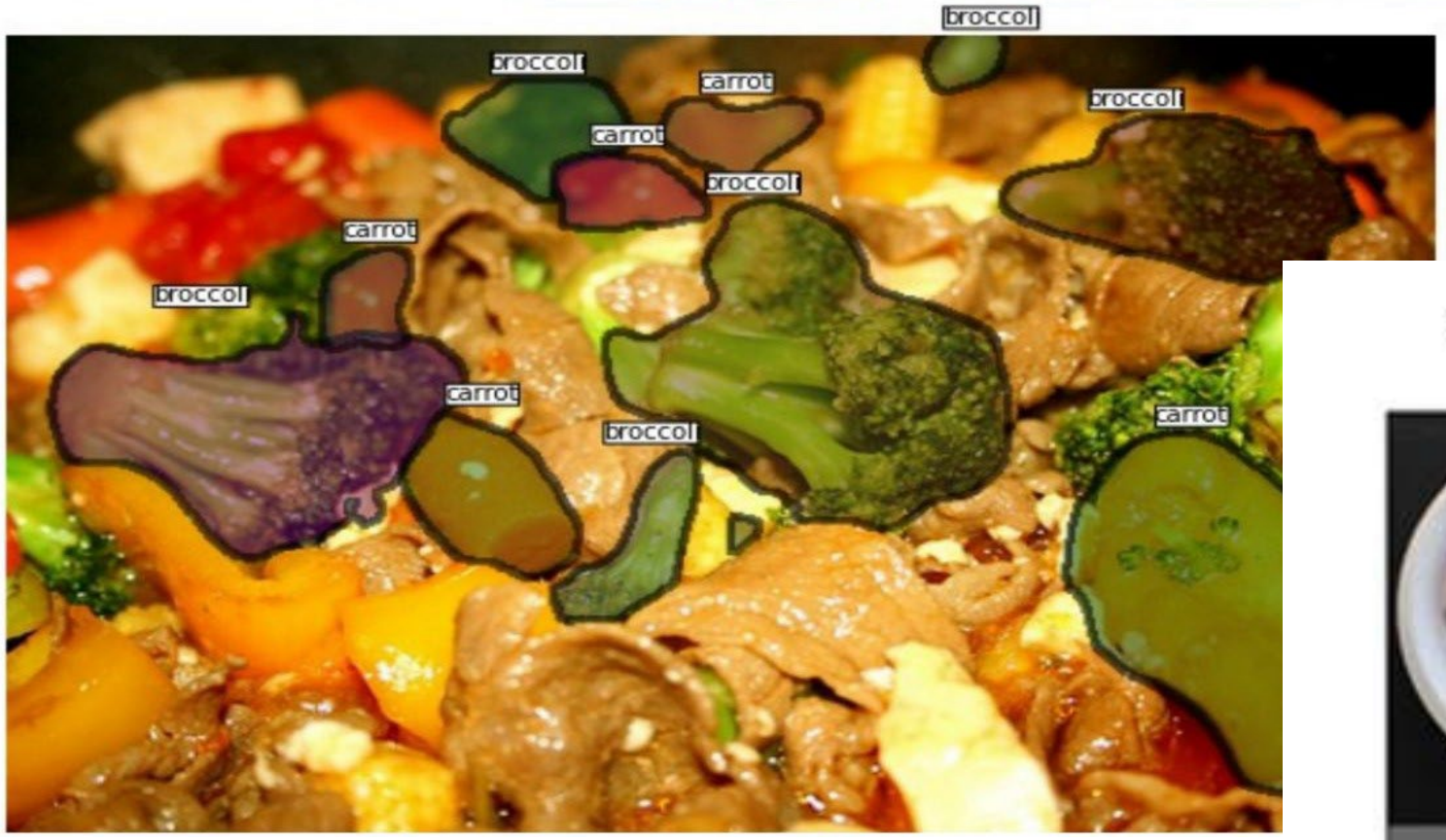


Face detection + emotion classification



Food semantic segmentation + nutrient content analysis

Food semantic segmentation + nutrient content analysis



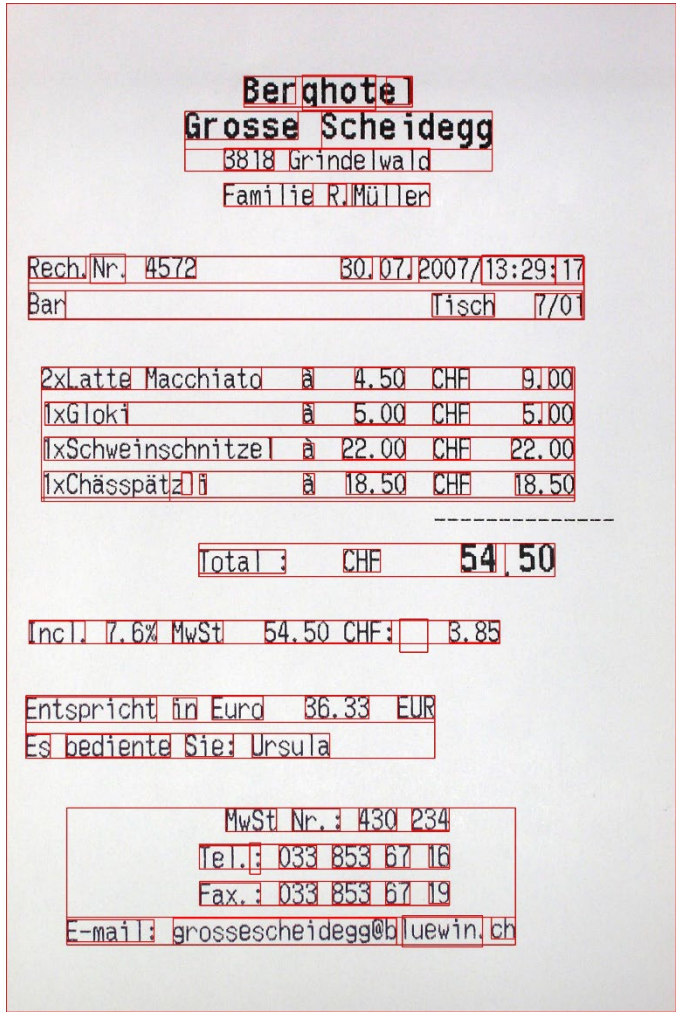
Segmentation:

Portion & Nutrition Estimation:

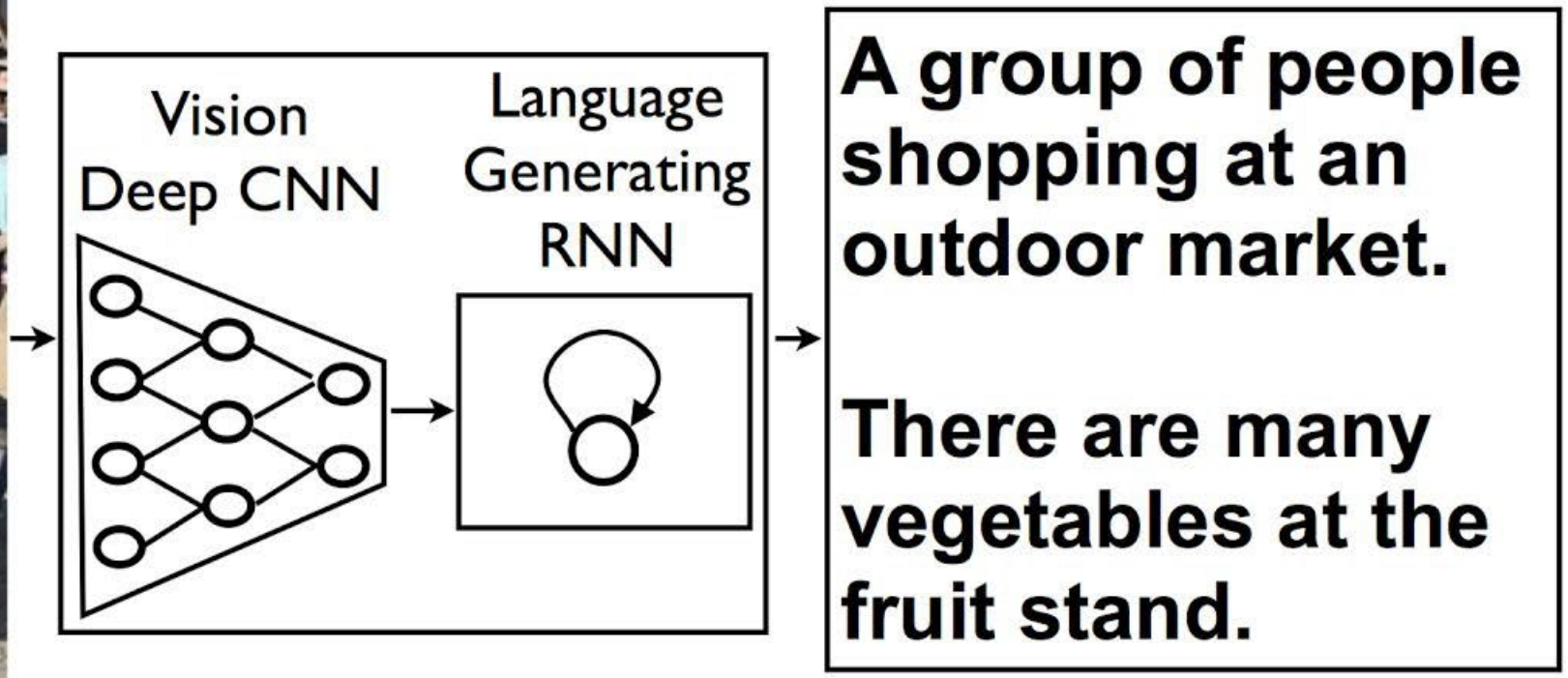
Sandwich: 146g
French Fries: 85g
Strawberries: 114g

Motion Detection

Scene text recognition (Document Parsing)



Task 6. Machine Vision + Natural Language Processing
Generate representations based on images



http://arxiv.org/abs/1411.4555 "Show and Tell: A Neural Image Caption Generator"

Visual Question and Answer (VQA) challenge

What vegetable is on the plate?
Neural Net: broccoli
Ground Truth: broccoli

What color are the shoes on the person's feet ?
Neural Net: brown
Ground Truth: brown

How many school busses are there?
Neural Net: 2
Ground Truth: 2

What sport is this?
Neural Net: baseball
Ground Truth: baseball

What is on top of the refrigerator?
Neural Net: magnets
Ground Truth: cereal

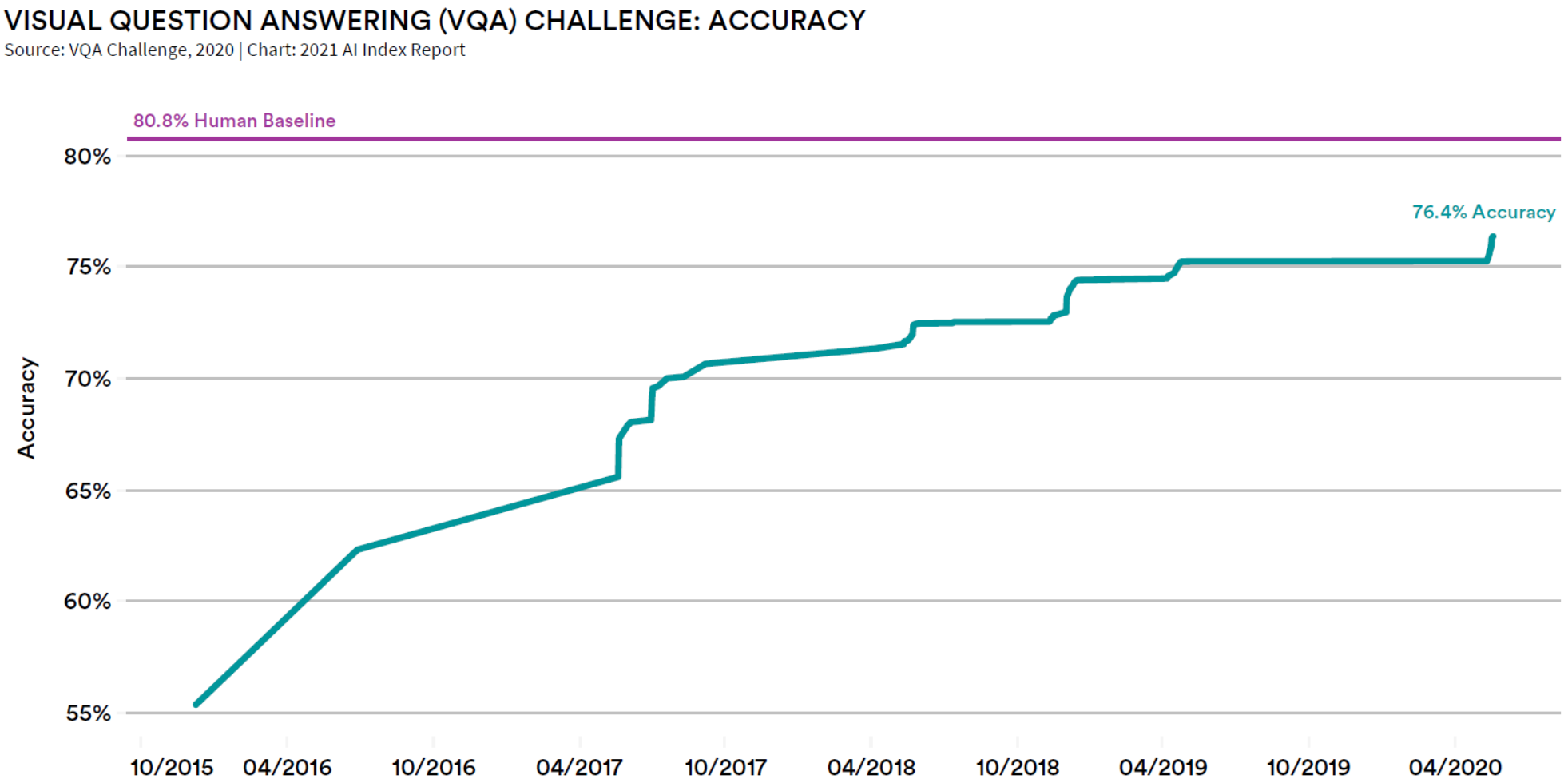
What uniform is she wearing?
Neural Net: shorts
Ground Truth: girl scout

What is the table number?
Neural Net: 4
Ground Truth:40

What are people sitting under in the back?
Neural Net: bench
Ground Truth: tent

https://avisingh599.github.io/deeplearning/visual-qa/

Visual Question and Answer (VQA) challenge



Deep Learning for Computer Vision: Summary

Foundations

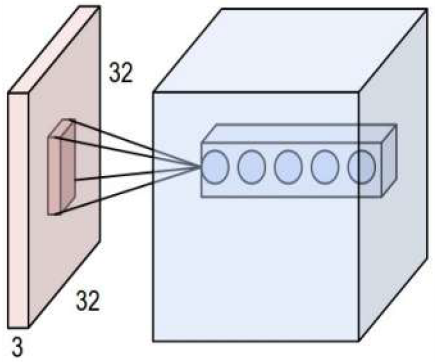
- What is computer vision?
- Representing images
- Convolutions for feature extraction

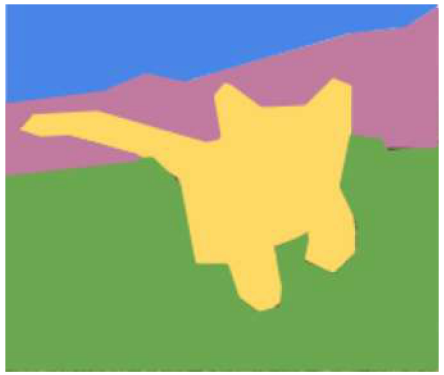
CNNs

- CNN architecture
- Application to classification

Applications

- Object detection
- Object segmentation
- Motion detection
- Scene text recognition
- Visual Q&A





Questions?