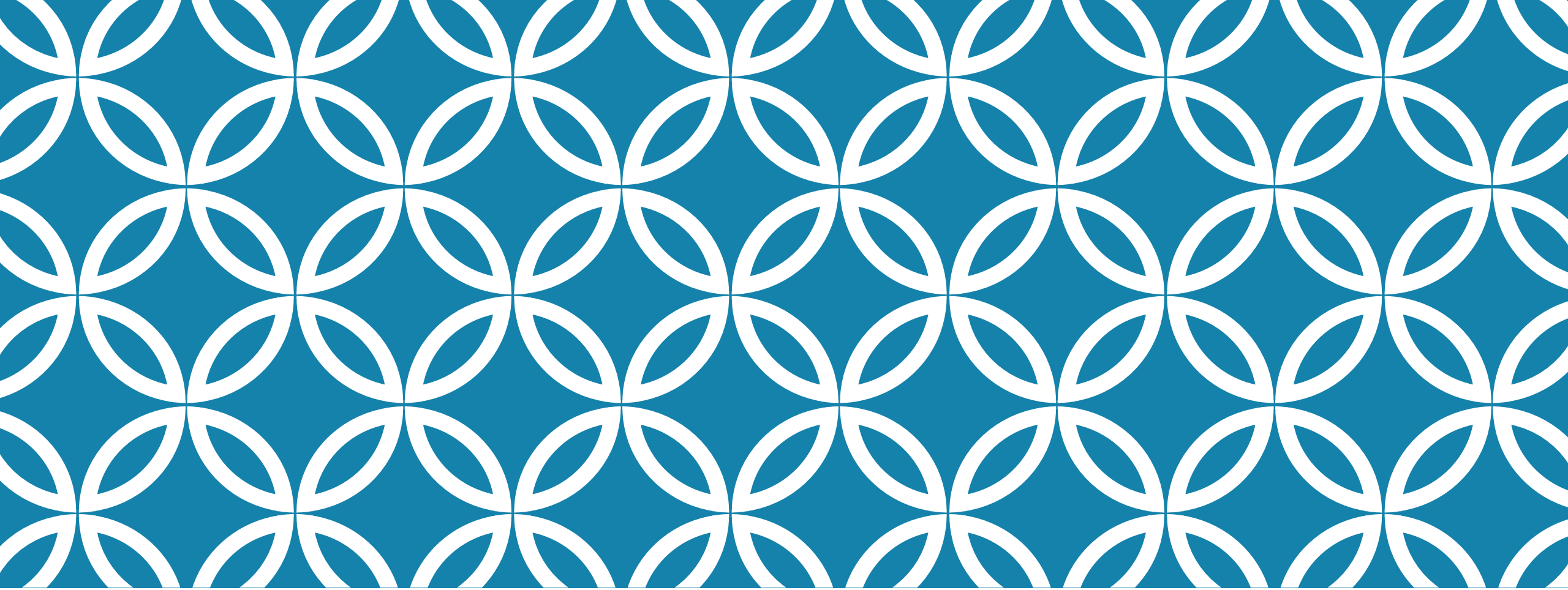


COMPUTER VISION TRANSFER LEARNING

Grzegorz Beringer

Based on materials by
Adam Brzeski

Gradient PG, 15/11/2018

AGENDA

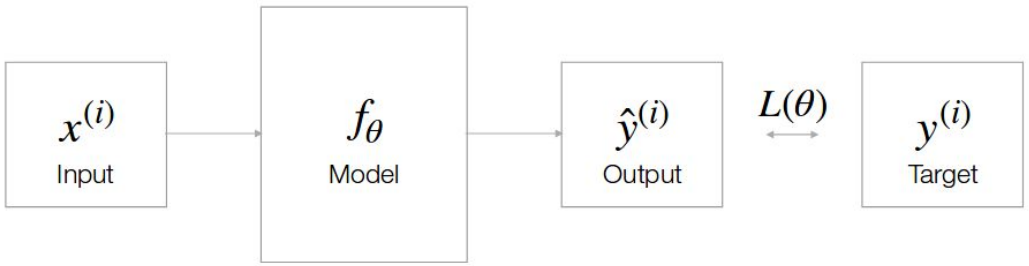
0. Quick recap
1. Deep Learning
2. Computer Vision:
    a. Convolutional Neural Network (CNN)
    b. Applications and Examples
3. Transfer Learning

QUICK RECAP

Supervised Learning

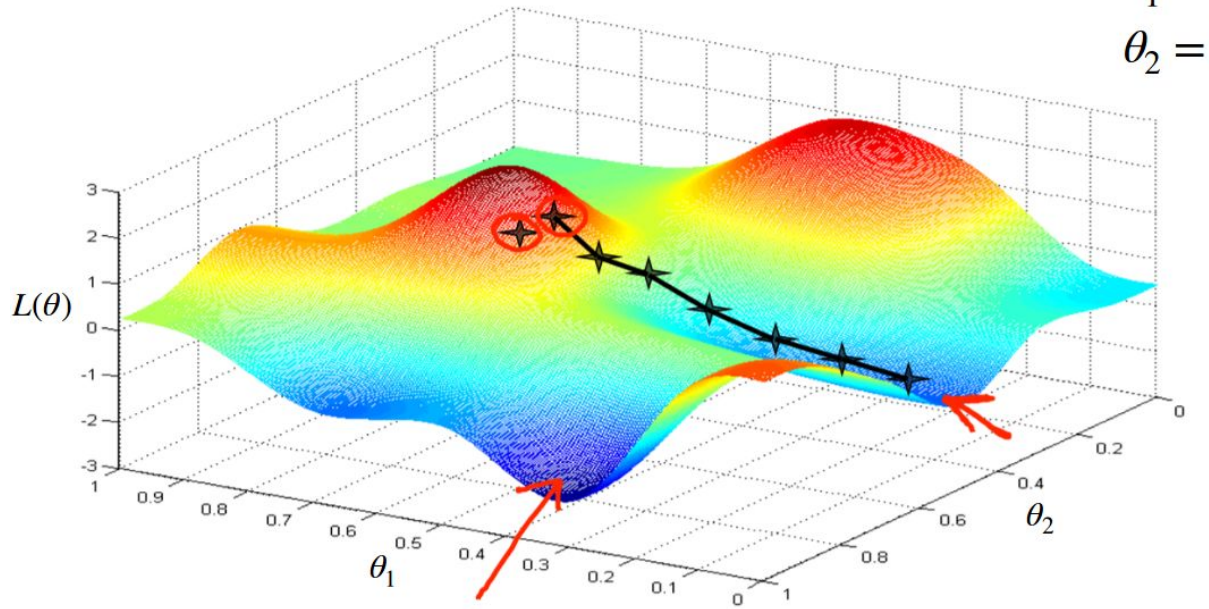
X → Model → Y

$$\hat{y}^{(i)} = f_\theta(x^{(i)})$$



$\theta_1 = a$

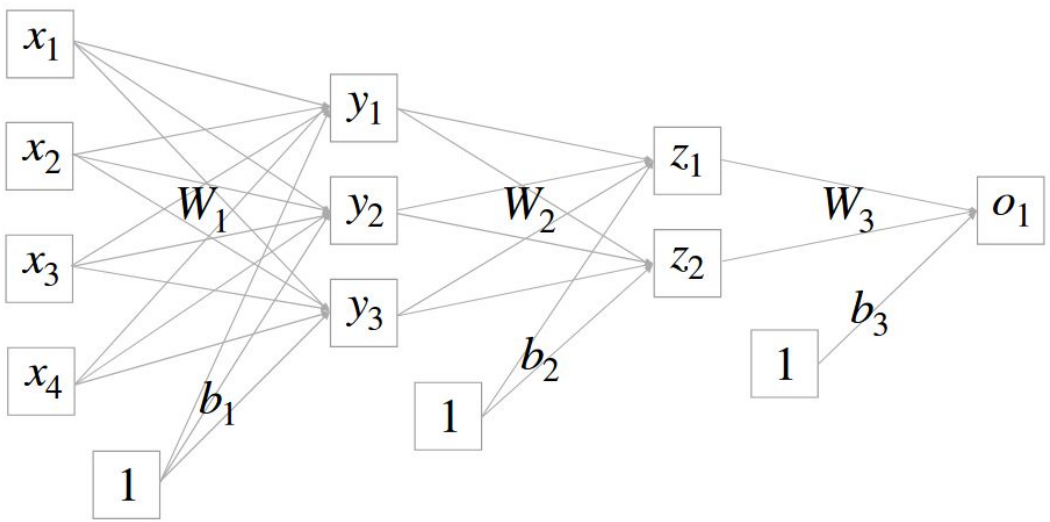
$\theta_2 = b$



Gradient to the rescue!

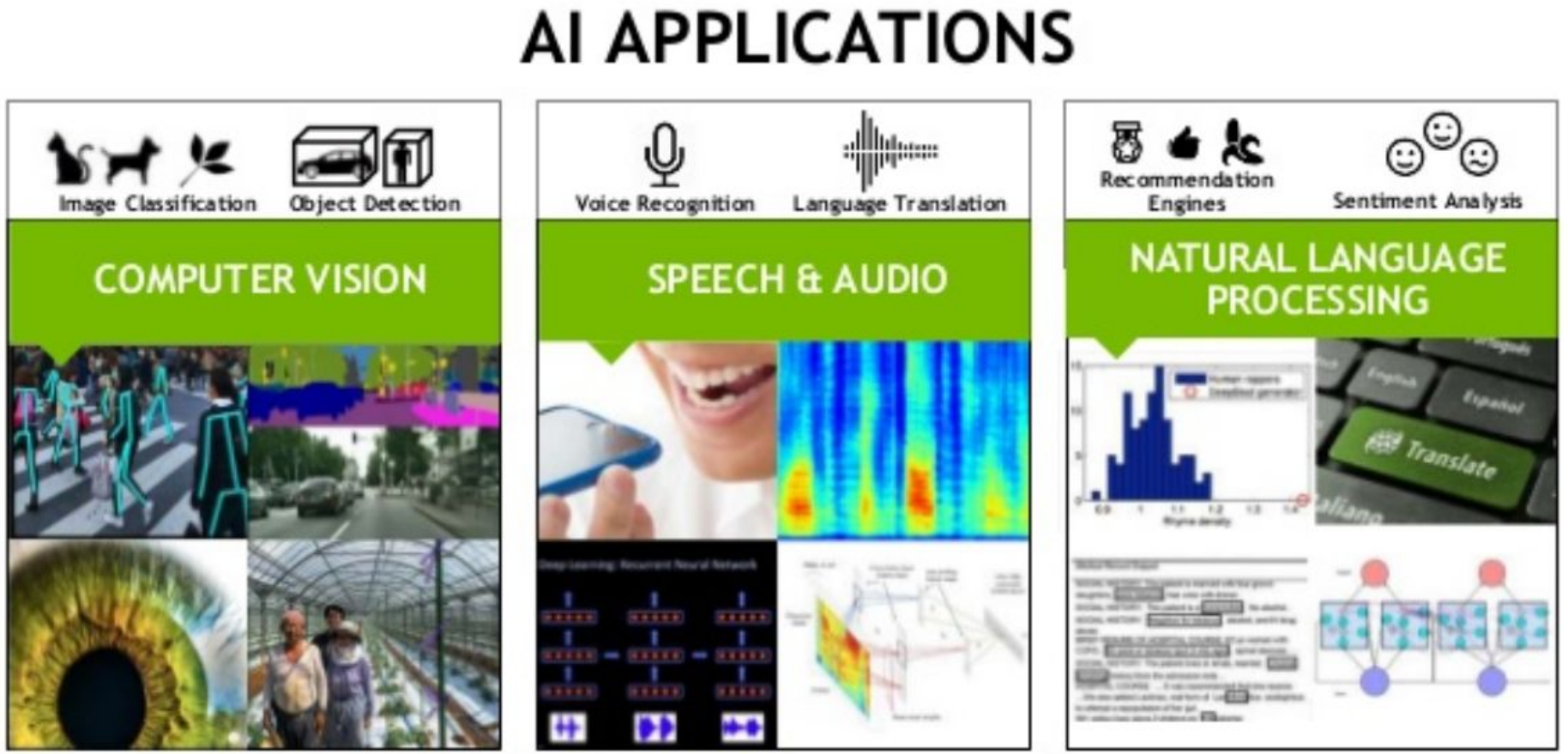
$$\theta^{(i+1)} = \theta^{(i)} - \alpha \cdot \nabla L(\theta)$$

Input Hidden 1 Hidden 2 Output



DEEP LEARNING

DEEP LEARNING



Source: https://www.slideshare.net/NVIDIA/deep-learning-workflows-training-and-inference

COMPUTER VISION

IMAGE CLASSIFICATION

Assign correct label set to an input image:

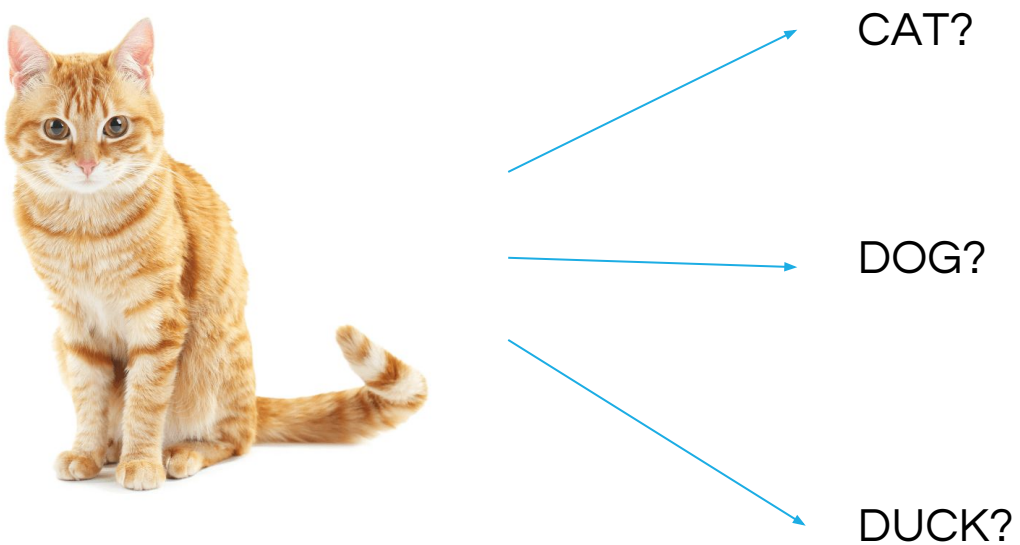
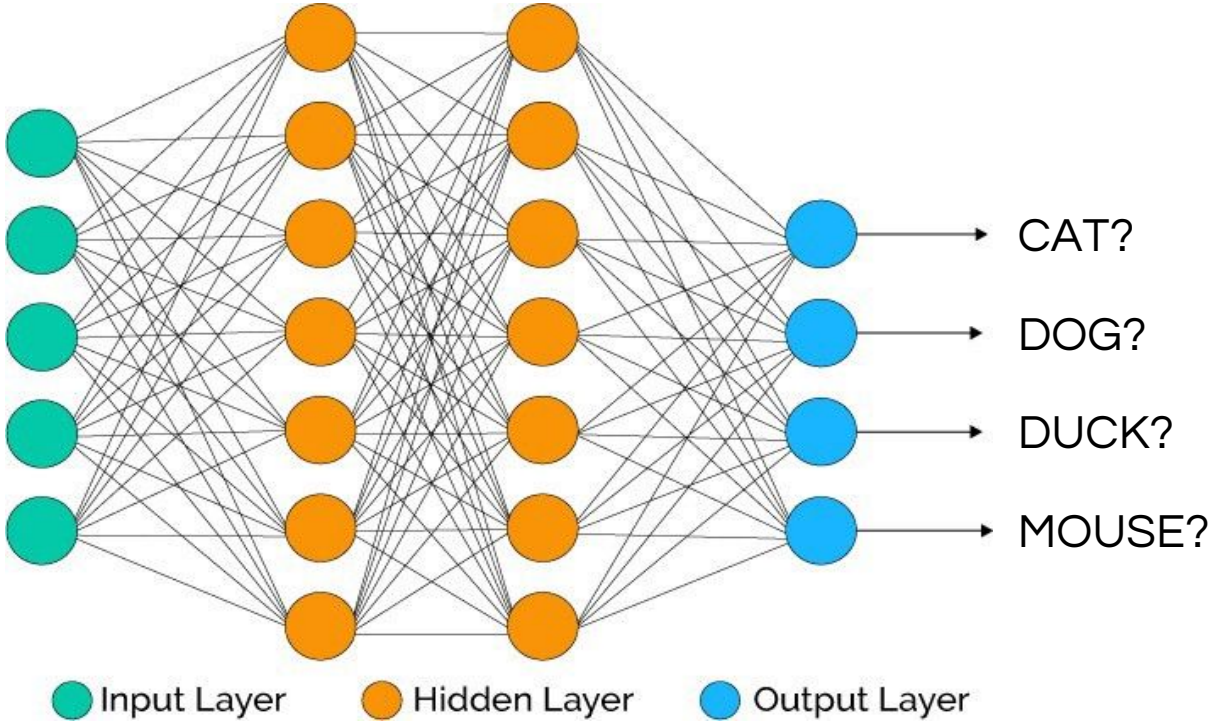


IMAGE CLASSIFICATION

We could use Fully-Connected Neural Networks (FCNN) from last time...



NEURAL NETWORKS FOR CV

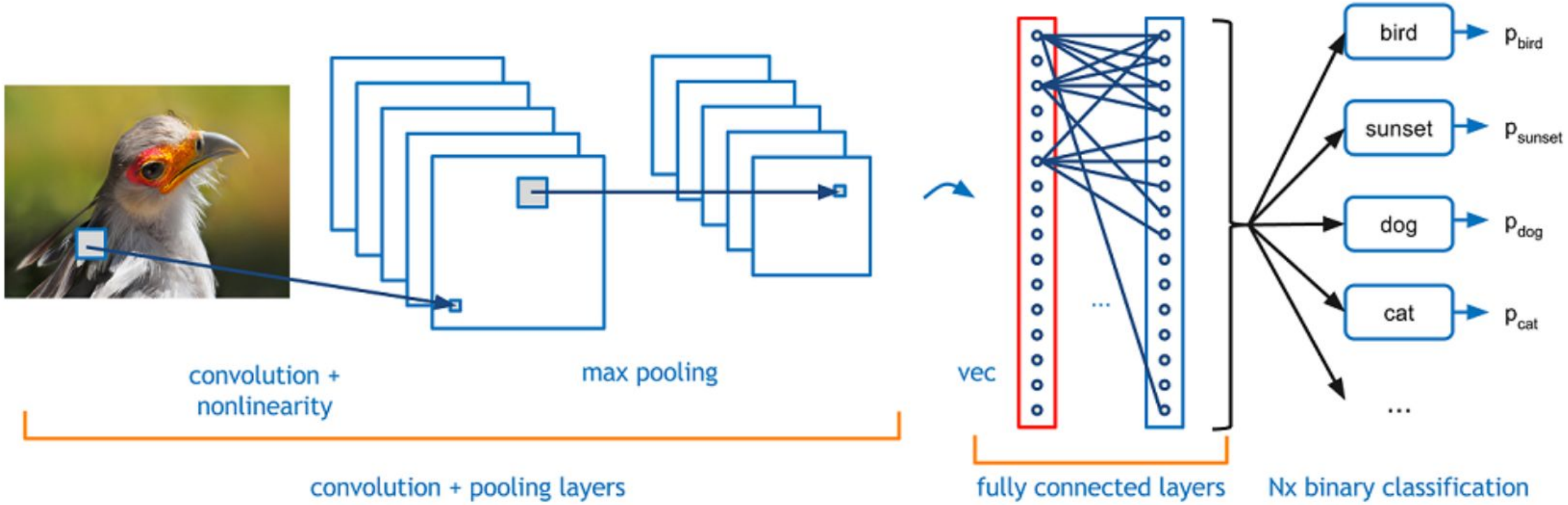
… but they probably won’t work well. Why is that?

- Waaaaaaay too much parameters…
- Not good with spatial data…

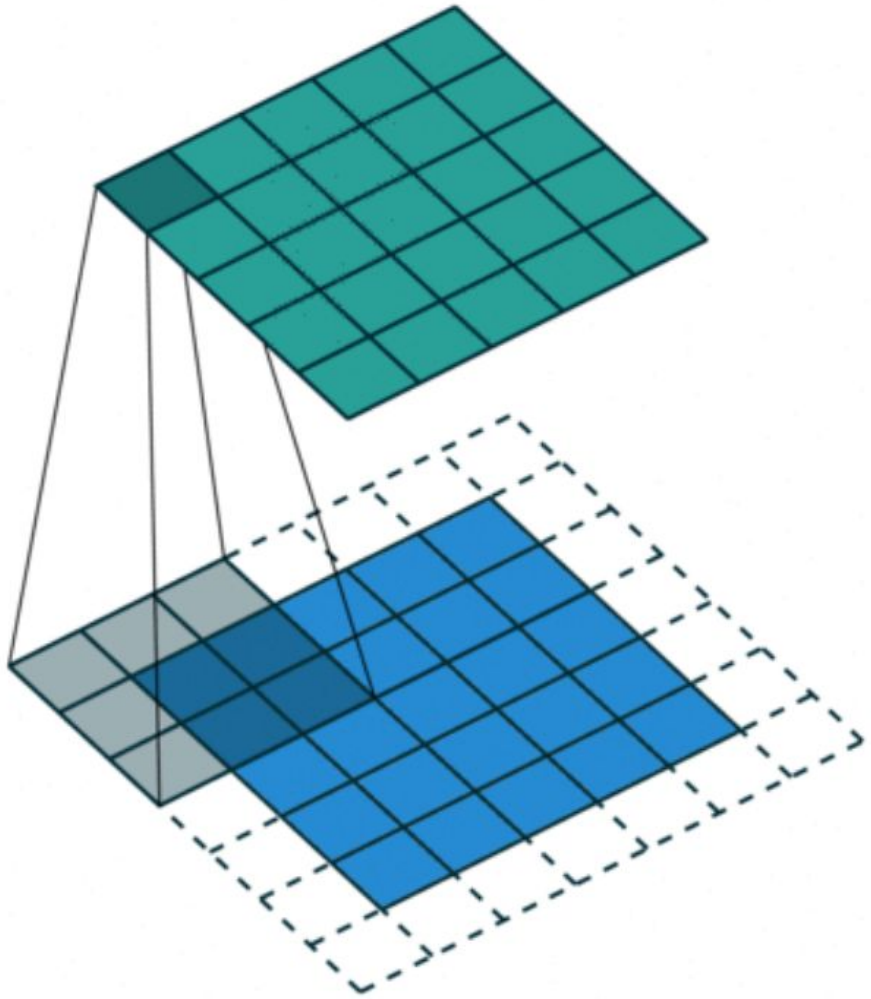
CONVOLUTIONAL NEURAL NETWORKS

*Adapted from last year's slides on Computer Vision by **Jakub Powierza***

CNN ARCHITECTURE

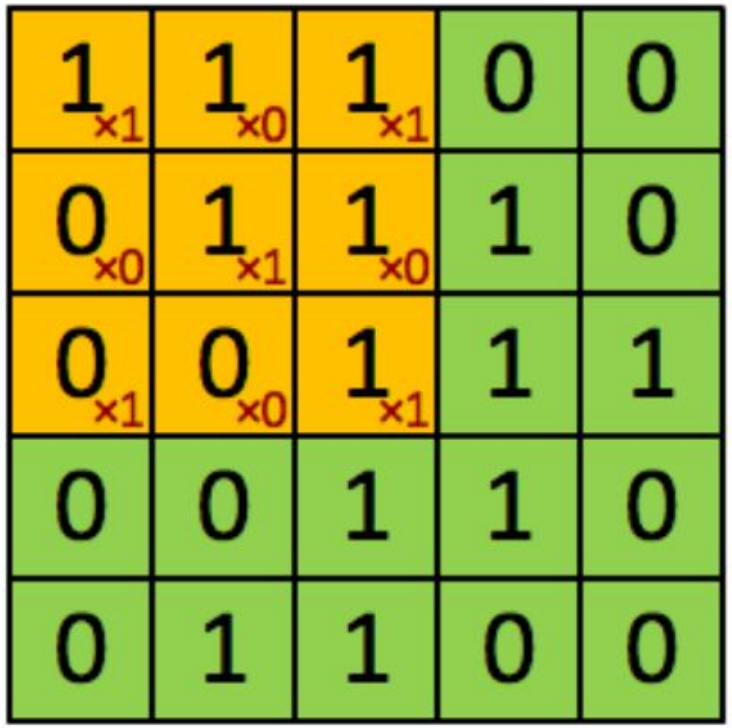


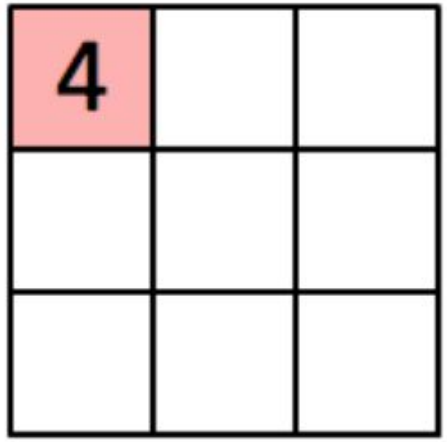
CONVOLUTION



- Creates "**feature maps**",
- Apply **filters** on the image,
- Move such filter over the image and calculate **feature**,
- Follow the **stride** (how many fields it should "jump"),
- Is defined by **kernel size** (filter size),
- Can use **padding** for bigger receptive field.

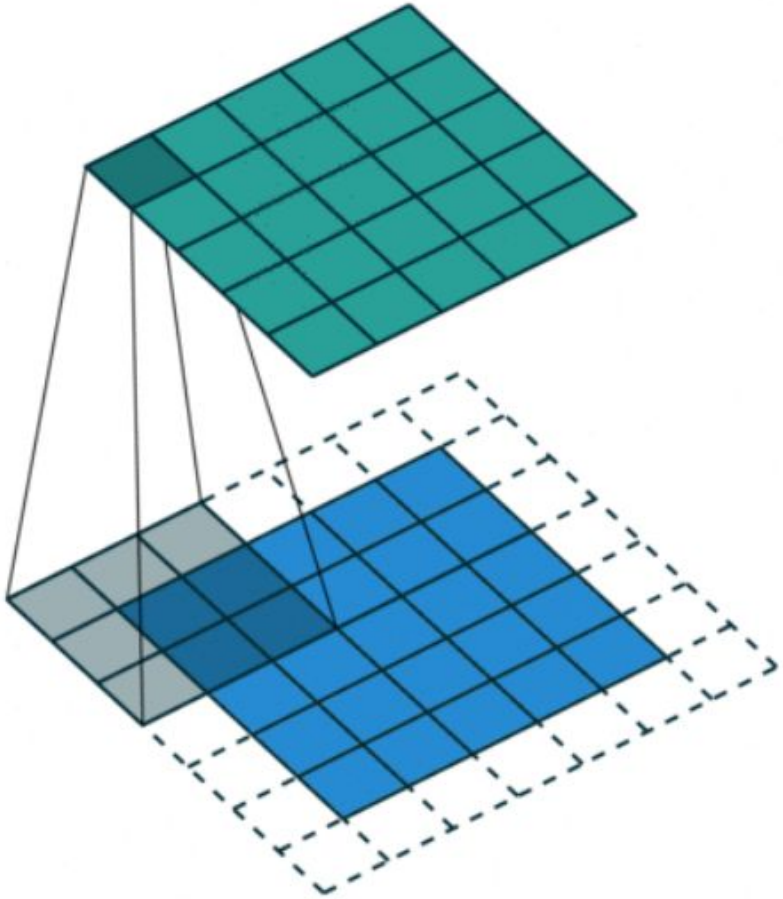
CONVOLUTION



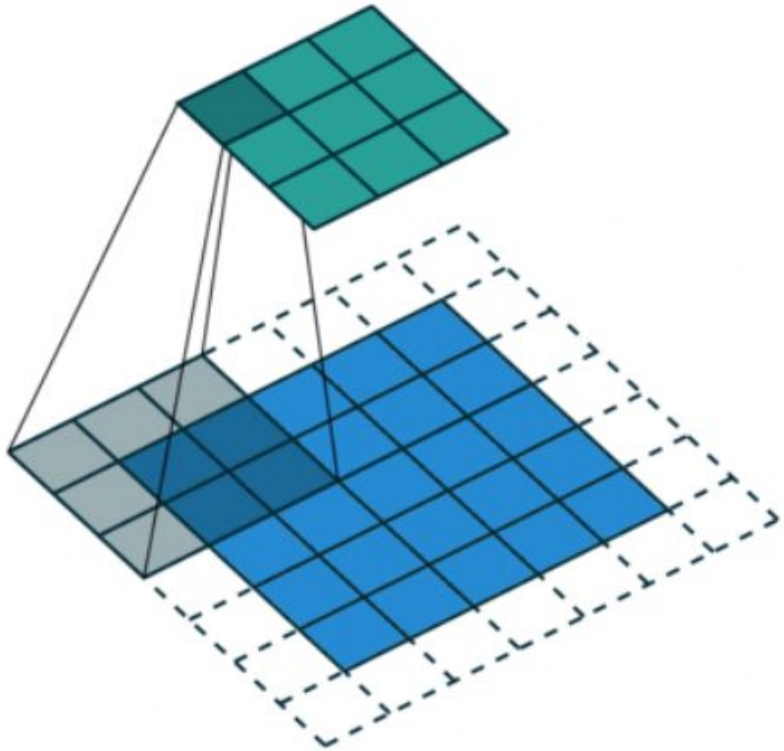


STRIDE

$Stride = 1$



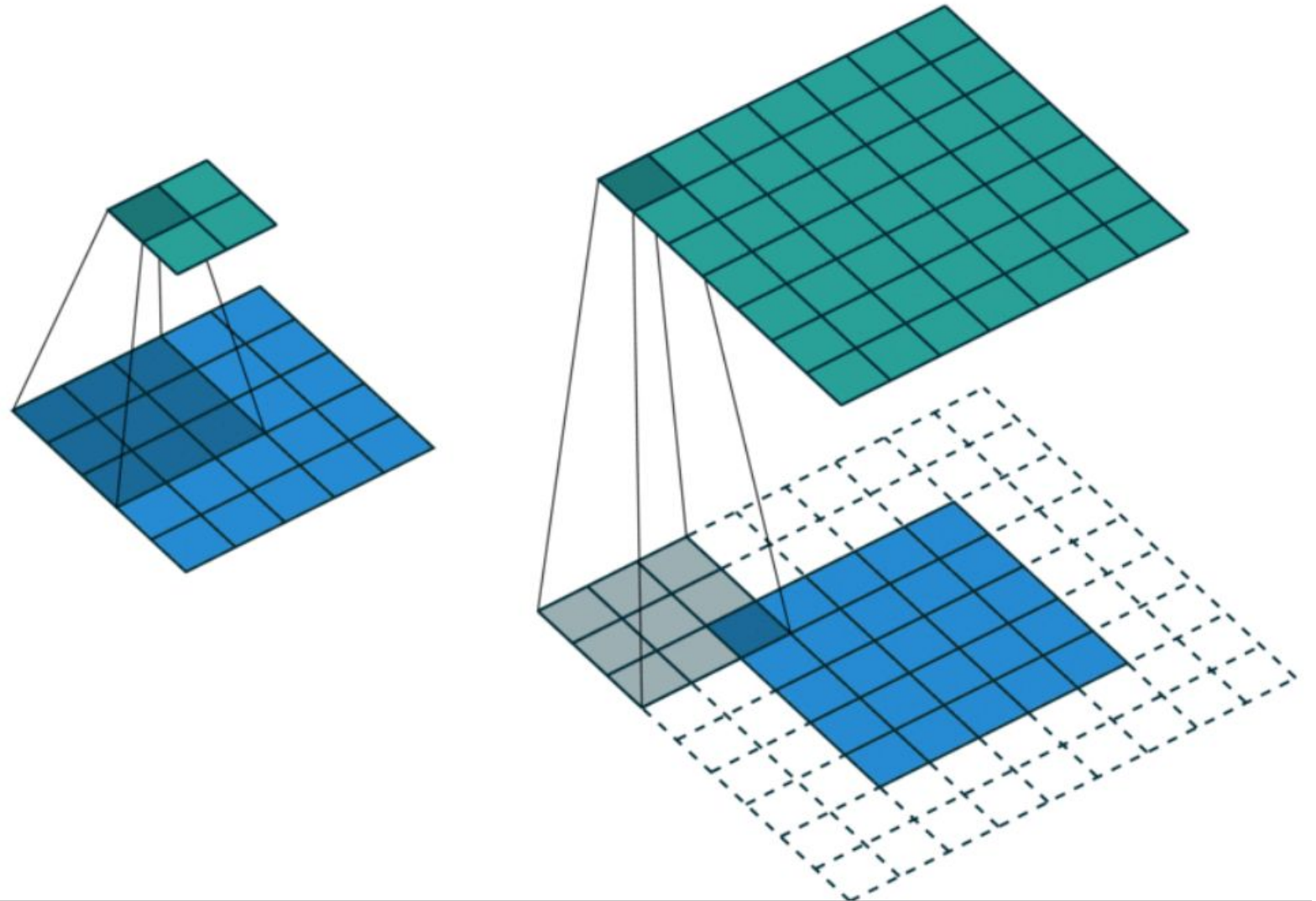
$Stride = 2$



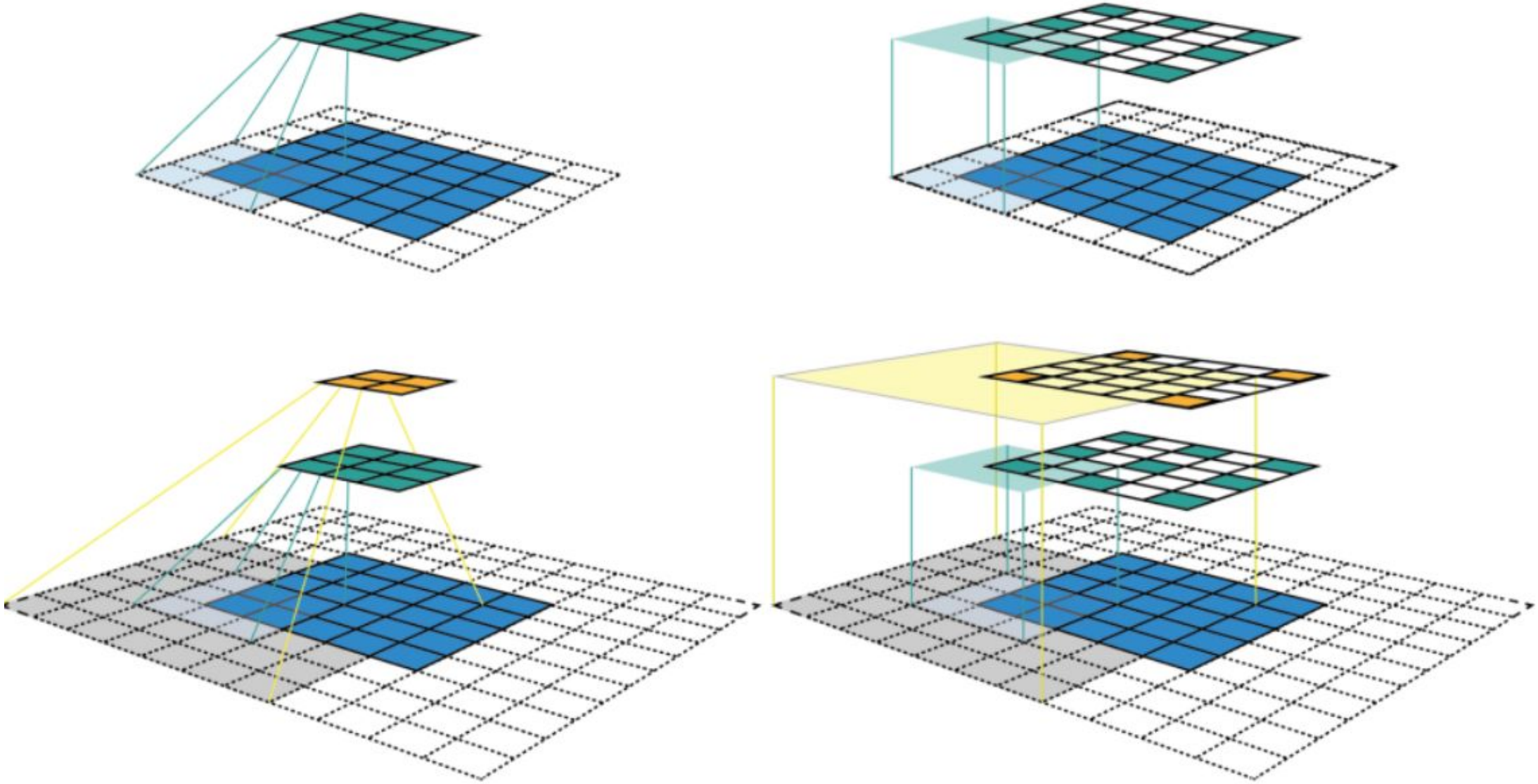
PADDING

$Padding = 0$

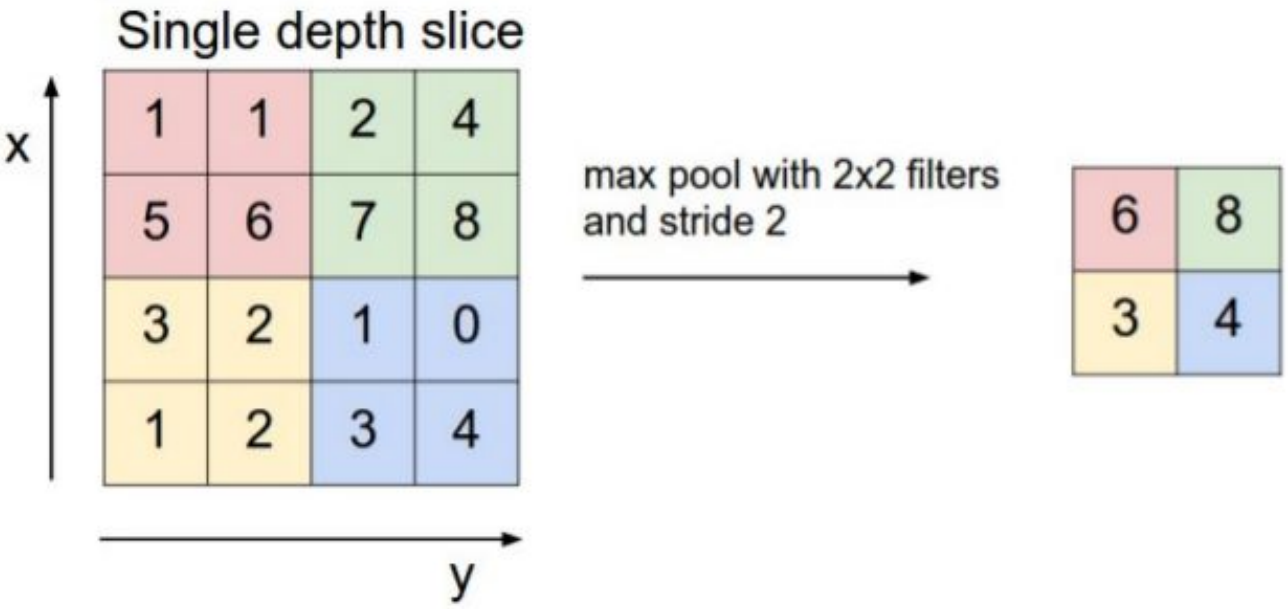
$Padding = 2$



RECEPTIVE FIELD

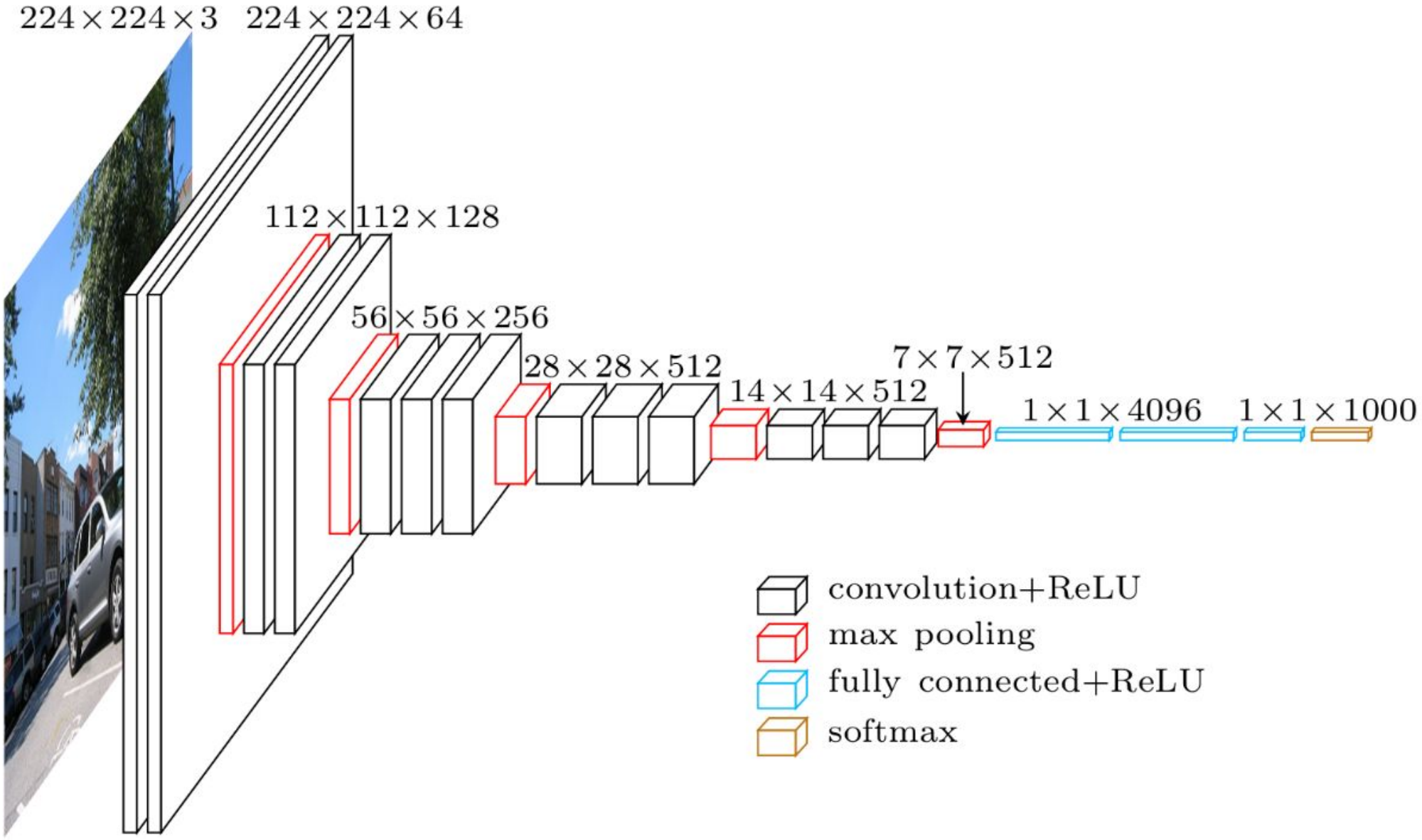


POOLING



- Pooling **reduces** spatial space,
- **Reduces** amount of parameters,
- Reduces overfitting,
- A simple **routing** (during back propagation),
- Most common: MaxPooling,
- Also: AvgPooling, ...

SAMPLE ARCHITECTURE: VGG



COMPUTER VISION APPLICATIONS

IMAGE CLASSIFICATION

Assign correct label set to an input image:

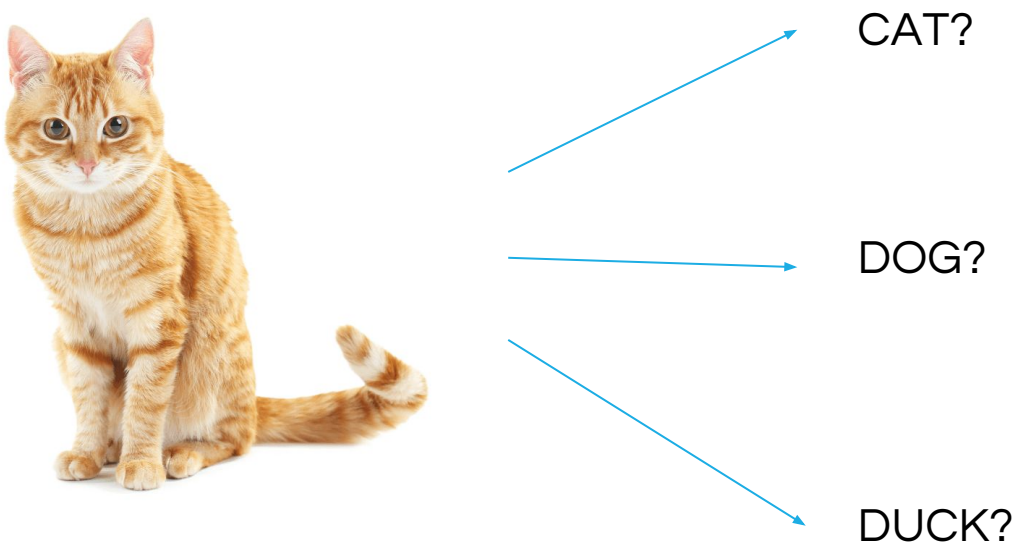


IMAGE CLASSIFICATION

High accuracy:

- Inception, Xception
- ResNet, ResNext
- DenseNet
- NasNet
- SENet

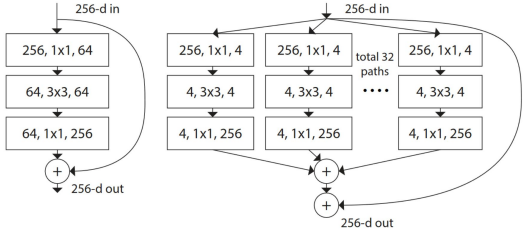
Fast:

- MobileNet
- ShuffleNet

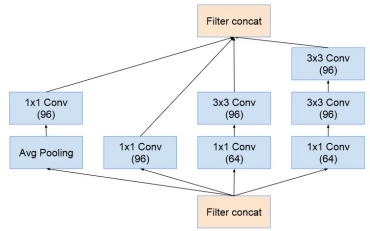
	224 × 224		320 × 320 / 299 × 299	
	top-1 err.	top-5 err.	top-1 err.	top-5 err.
ResNet-152 [10]	23.0	6.7	21.3	5.5
ResNet-200 [11]	21.7	5.8	20.1	4.8
Inception-v3 [44]	-	-	21.2	5.6
Inception-v4 [42]	-	-	20.0	5.0
Inception-ResNet-v2 [42]	-	-	19.9	4.9
ResNeXt-101 (64 × 4d) [47]	20.4	5.3	19.1	4.4
DenseNet-264 [14]	22.15	6.12	-	-
Attention-92 [46]	-	-	19.5	4.8
Very Deep PolyNet [51] †	-	-	18.71	4.25
PyramidNet-200 [8]	20.1	5.4	19.2	4.7
DPN-131 [5]	19.93	5.12	18.55	4.16
SENet-154	**18.68**	**4.47**	**17.28**	**3.79**
NASNet-A (6@4032) [55] †	-	-	17.3‡	3.8‡
SENet-154 (post-challenge)	-	-	**16.88‡**	**3.58‡**

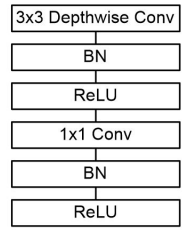
State-of-the-art CNNs results on ImageNet validation set.

Source: https://arxiv.org/pdf/1709.01507.pdf

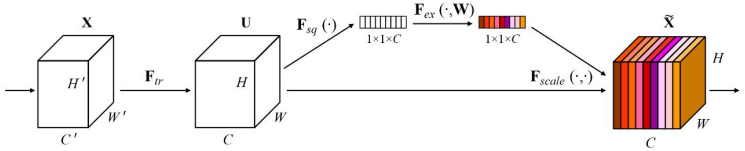


ResNet block **ResNext block**





Inception module **MobileNet block**



Squeeze-and-Excitation block

PROBLEM

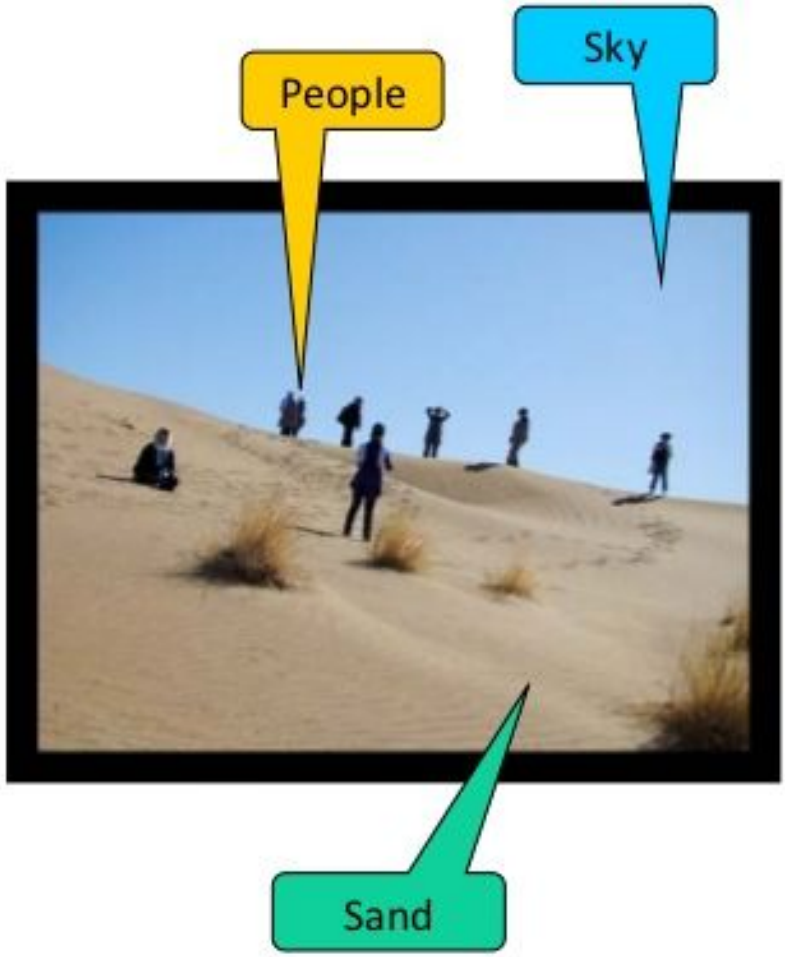
What if we need infer real values from images instead of predicting a class?

REGRESSION

- Can be solved just like multilabel classification
- Use proper final activation function e.g. sigmoid for <0,1> outputs or no activation
- Change loss function

PROBLEM

What if images could belong to multiple classes at once



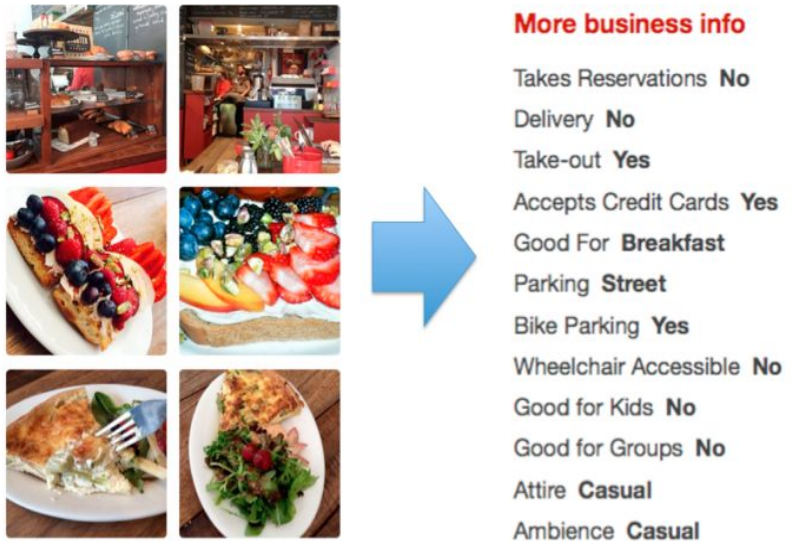
MULTILABEL CLASSIFICATION

- Can be solved just like multiclass classification
- Change final activation function: Softmax → Sigmoid
- Change loss function
- Watch out! Class balancing and data splits become a challenge!

Image source:
https://www.slideshare.net/dloiacono/ahmadi

PROBLEM

Train a model for predicting variables when instead of single images with labels we are given bags of images, such that the label refers to some image or images in the bag.



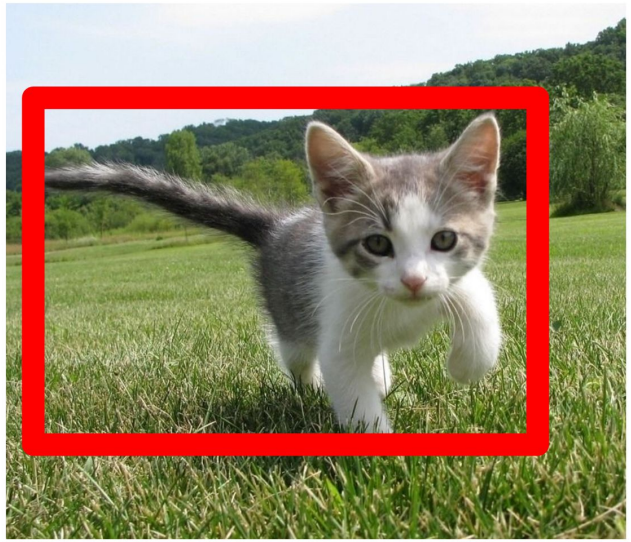
MULTIPLE INSTANCE LEARNING

- If we can predict variables on bag-level, we can treat entire bags as images or aggregate image features
- See https://www.kaggle.com/c/yelp-restaurant-photo-classification as an example
- If we must predict variable on image level, the problem is more difficult. We'll come back to this later on.

Image source:
https://engineeringblog.yelp.com/2016/05/yelp-kaggle-photo-challenge-interview-2.html

PROBLEM

Classify image and predict localization of an object



CAT

Image source:
http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture11.pdf

OBJECT LOCALIZATION

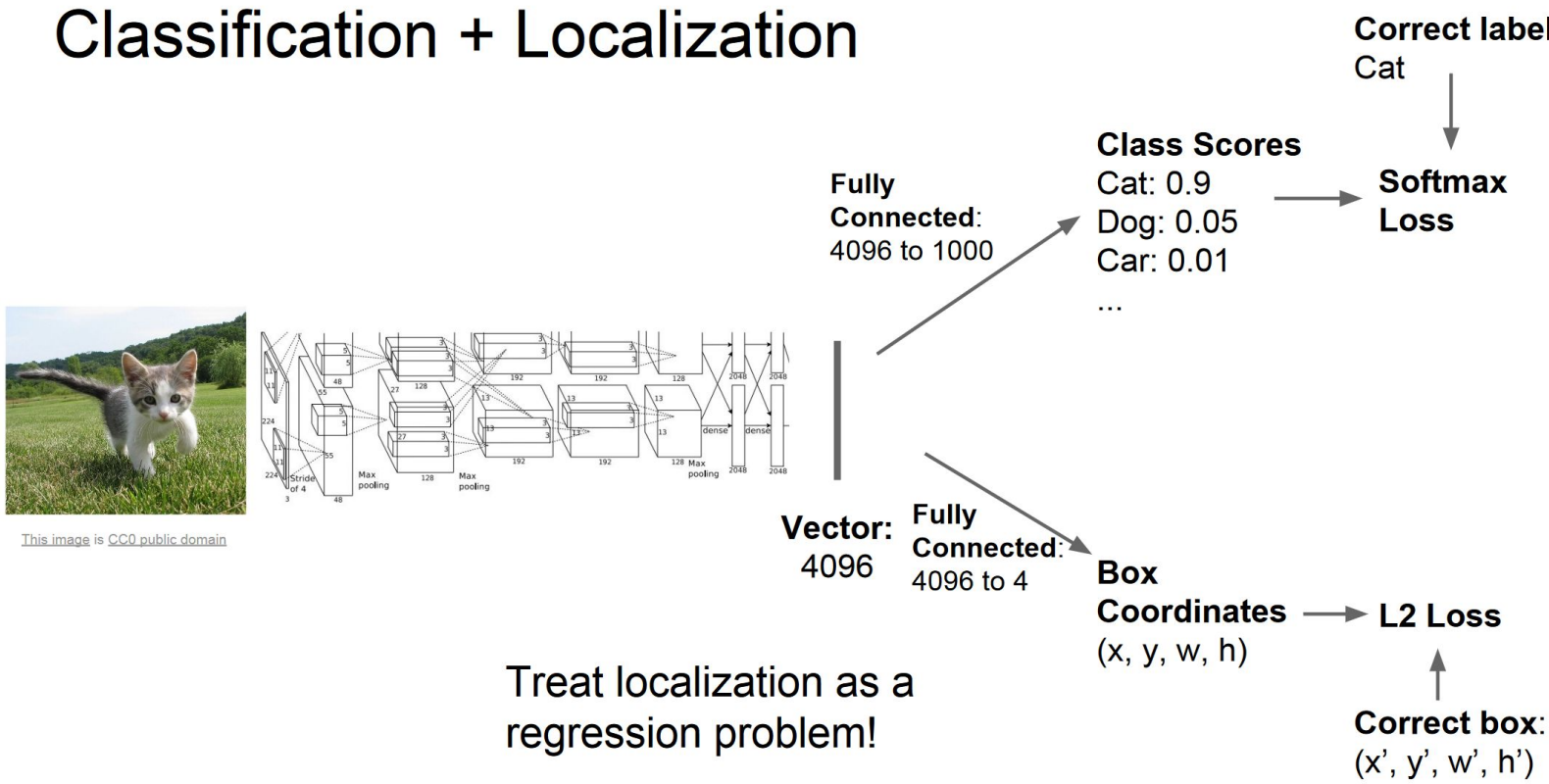


Image source:
http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture11.pdf

PROBLEM

Detect multiple objects of the same kind with potential occlusion

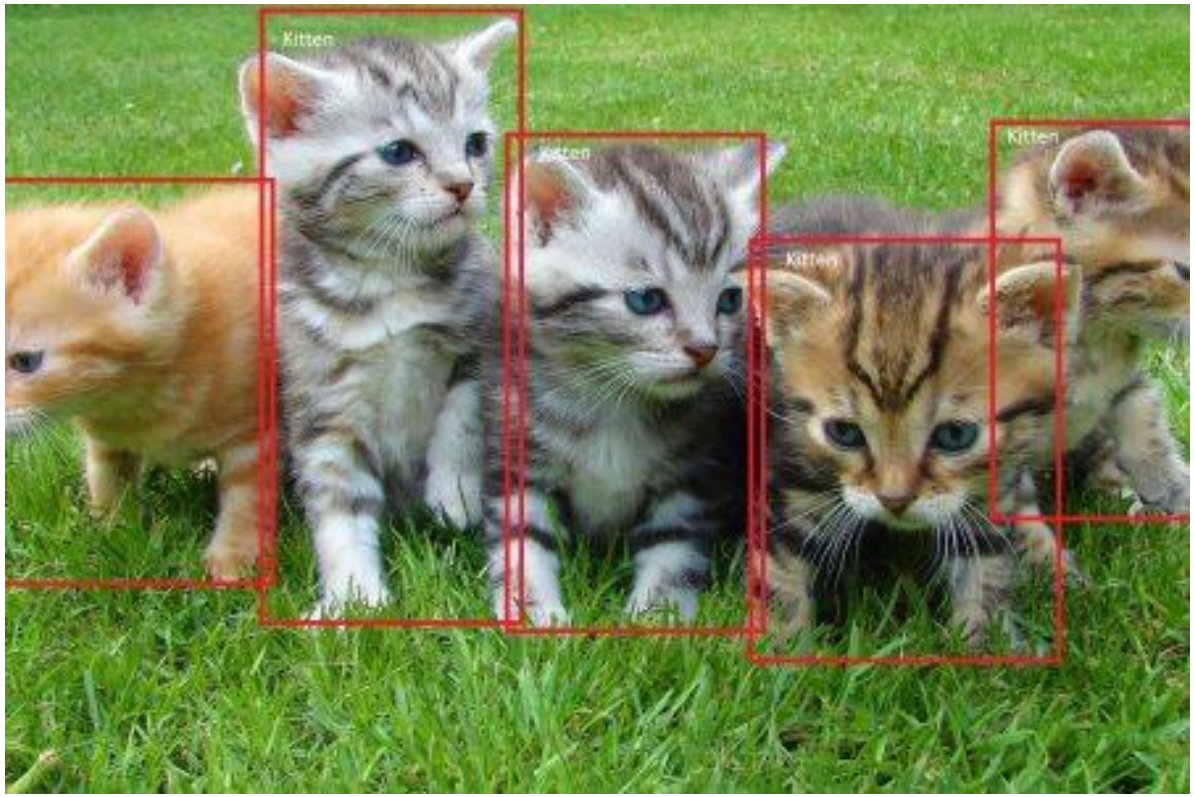
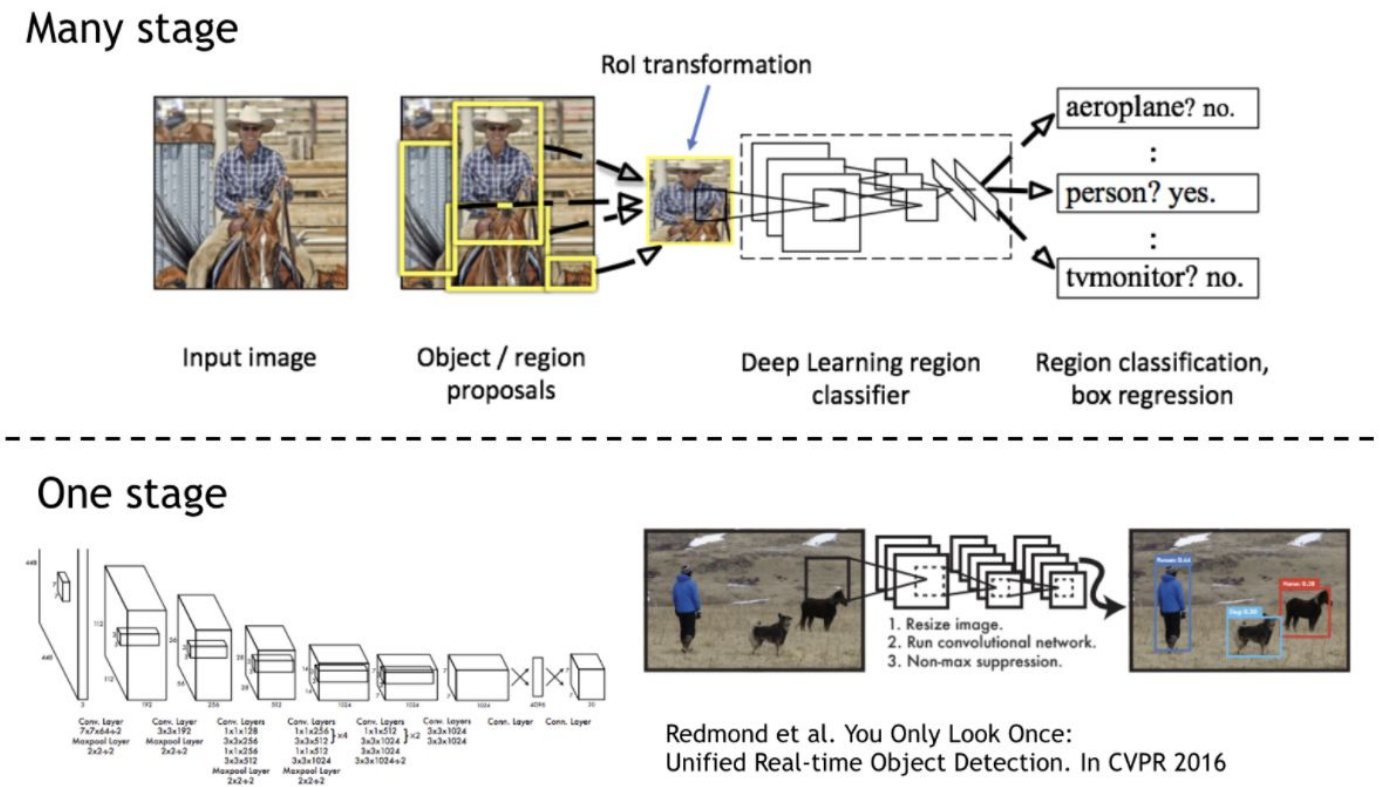


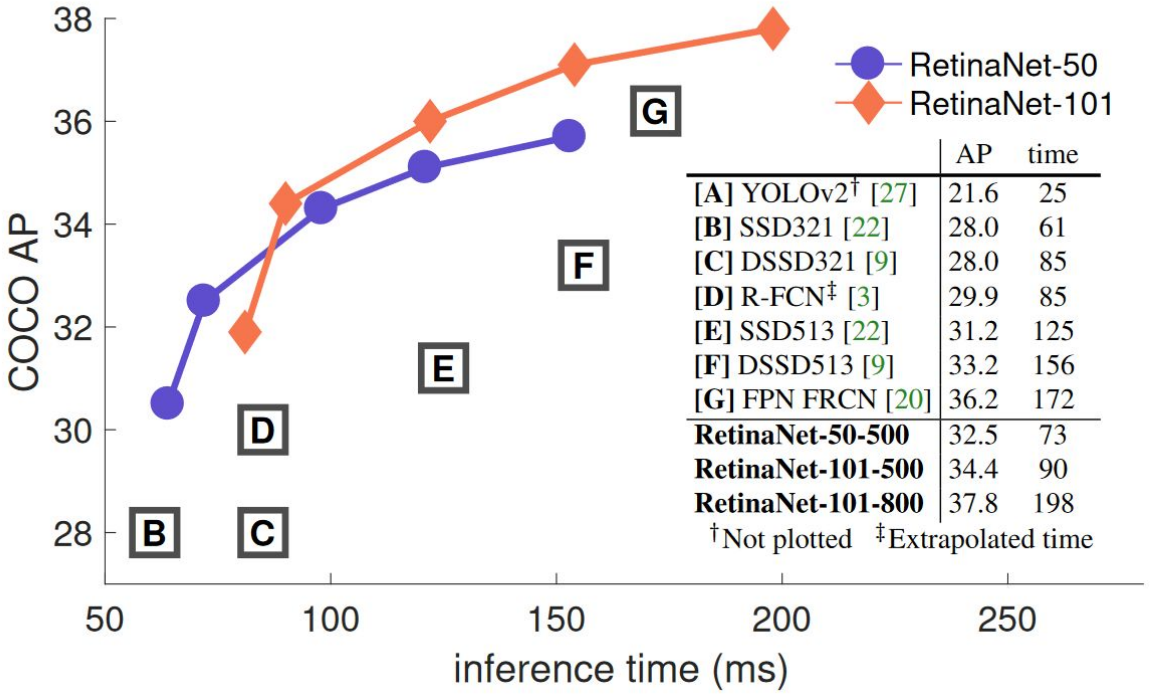
Image source:
https://www.microsoft.com/developerblog/2017/04/10/end-end-object-detection-box/

OBJECT DETECTION

- Accurate, 2-stage detectors:
  - Faster-RCNN
  - … and its numerous extensions
- Fast and also accurate 1-stage detectors:
  - YOLO
  - SSD
  - RetinaNet

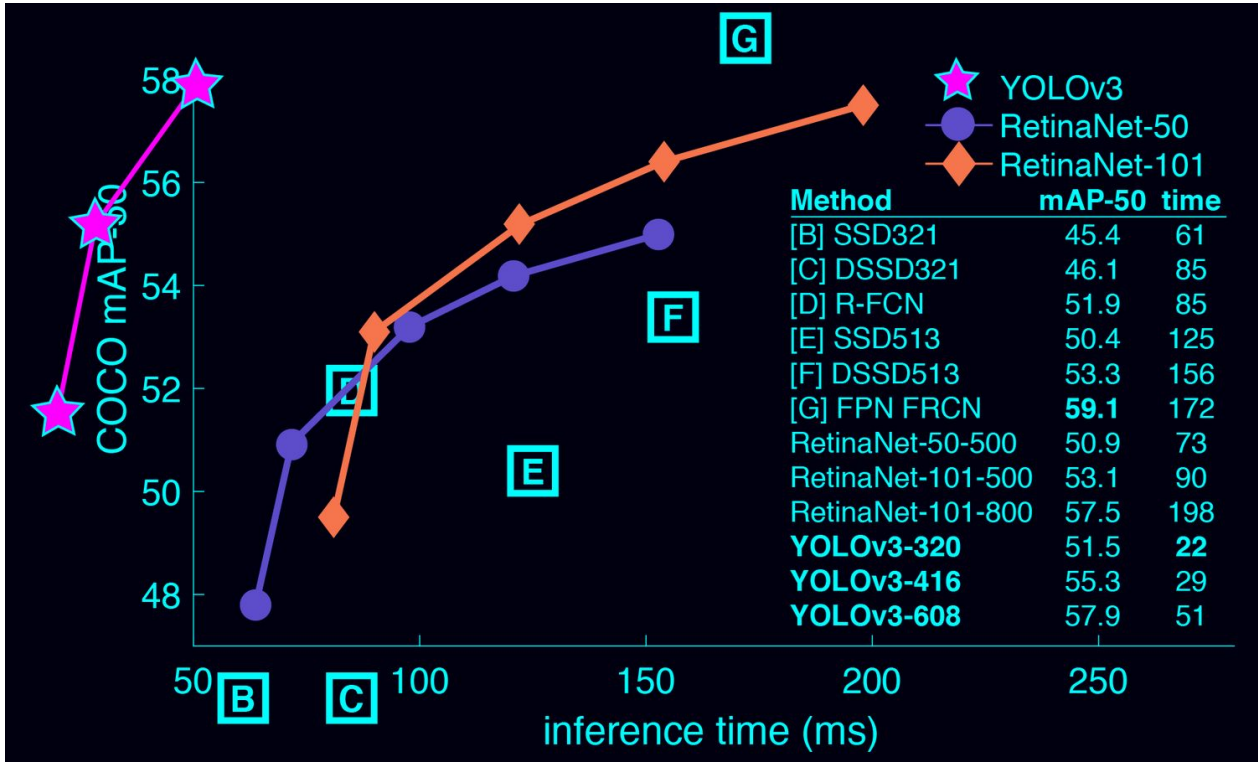


OBJECT DETECTION



	AP	time
[A] YOLOv2† [27]	21.6	25
[B] SSD321 [22]	28.0	61
[C] DSSD321 [9]	28.0	85
[D] R-FCN‡ [3]	29.9	85
[E] SSD513 [22]	31.2	125
[F] DSSD513 [9]	33.2	156
[G] FPN FRCN [20]	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	37.8	198

†Not plotted ‡Extrapolated time

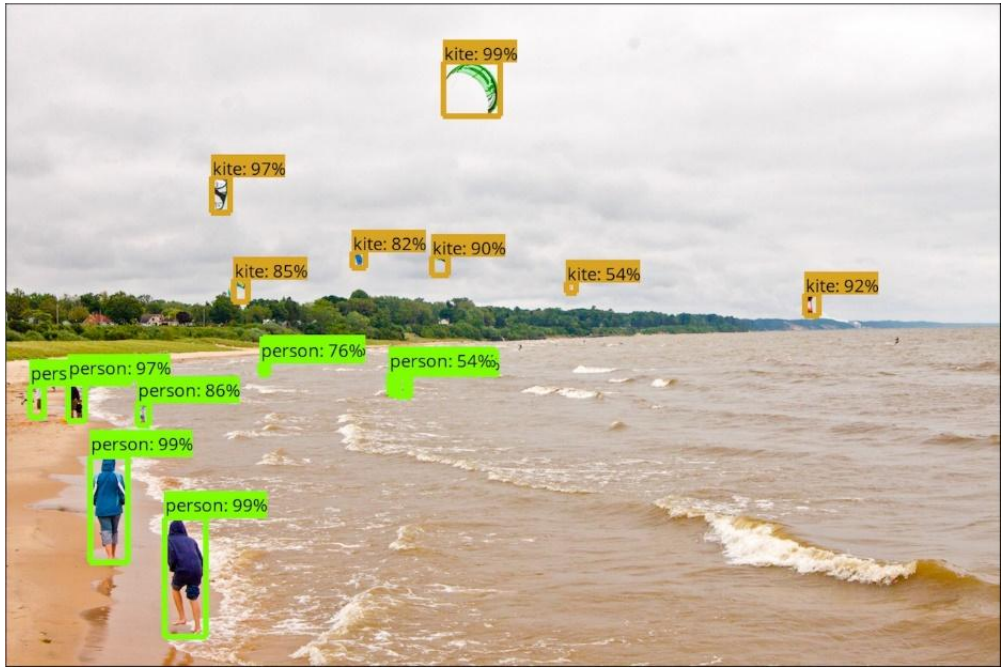


Method	mAP-50	time
[B] SSD321	45.4	61
[C] DSSD321	46.1	85
[D] R-FCN	51.9	85
[E] SSD513	50.4	125
[F] DSSD513	53.3	156
[G] FPN FRCN	**59.1**	172
RetinaNet-50-500	50.9	73
RetinaNet-101-500	53.1	90
RetinaNet-101-800	57.5	198
YOLOv3-320	51.5	**22**
YOLOv3-416	55.3	29
YOLOv3-608	57.9	51

OBJECT DETECTION

Great open-source code:

- TensorFlow Object Detection API
  https://github.com/tensorflow/models/tree/master/research/object_detection
  - Faster R-CNN, SSD, Mask R-CNN
- Facebook Detectron (for Caffe2)
  https://github.com/facebookresearch/Detectron
  - RetinaNet, Faster R-CNN, Mask R-CNN
- YOLO v3
  https://pjreddie.com/darknet/yolo/
  - Yolo v1, v2, v3



PROBLEM

Generate text description for an image

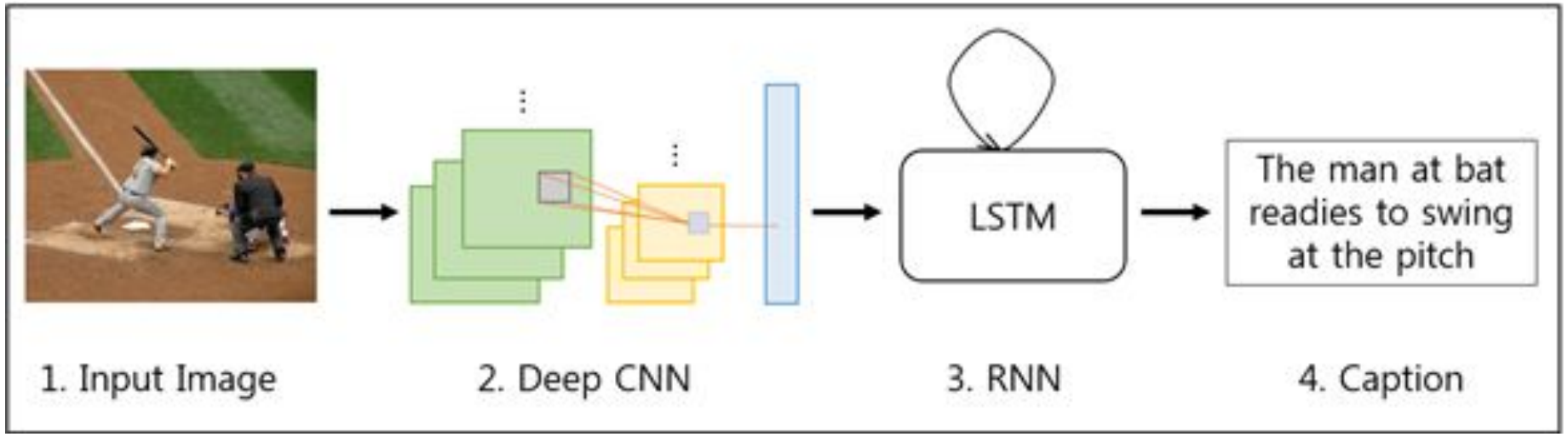
"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

IMAGE CAPTIONING

Sequence prediction: CNN + LSTM



PROBLEM

Read text from an image

Image source:
http://teaching.paganstudio.com/digitalfoundations/?p=171

OPTICAL CHARACTER RECOGNITION

- Sequence prediction again: CNN + LSTM
- Attention models again
- Example: Google's Attention-OCR

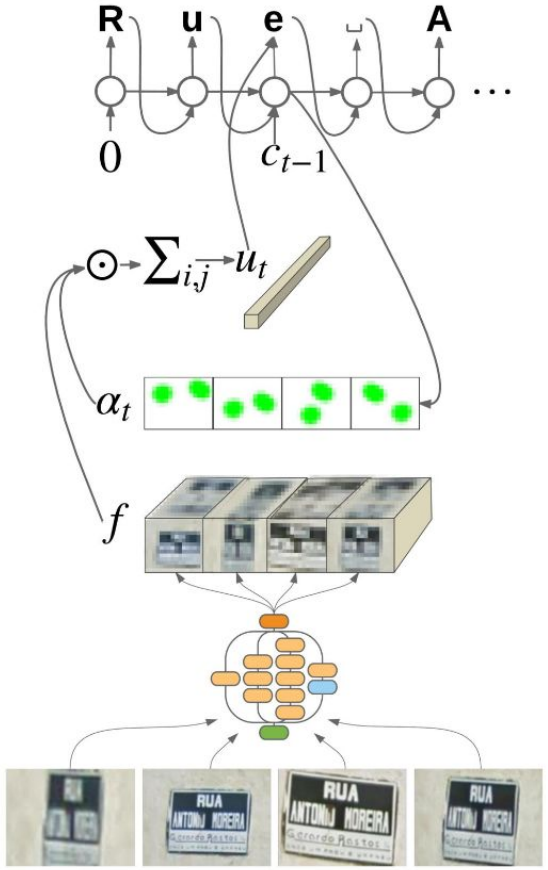


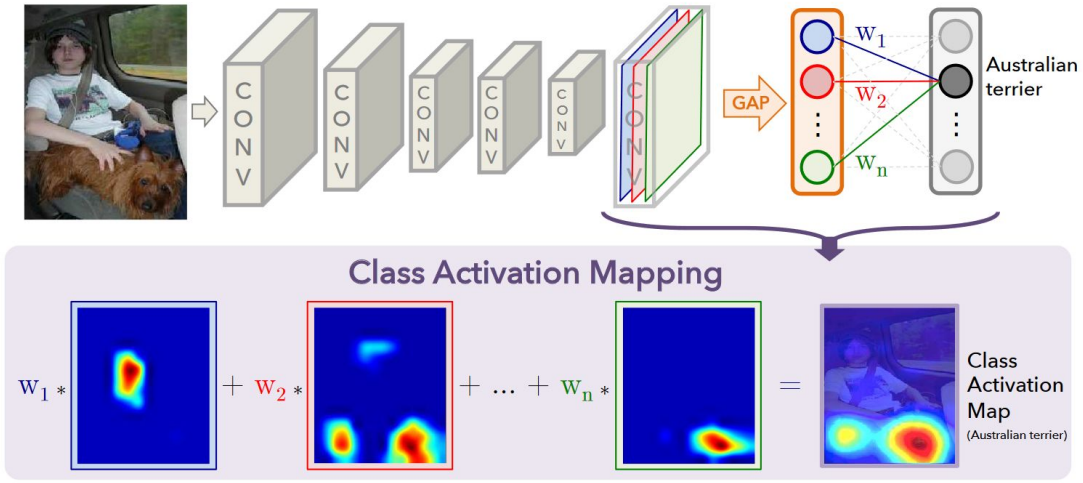
Image source:
https://arxiv.org/pdf/1704.03549.pdf

PROBLEM

What if we need to locate object but there are no locations in the training data?

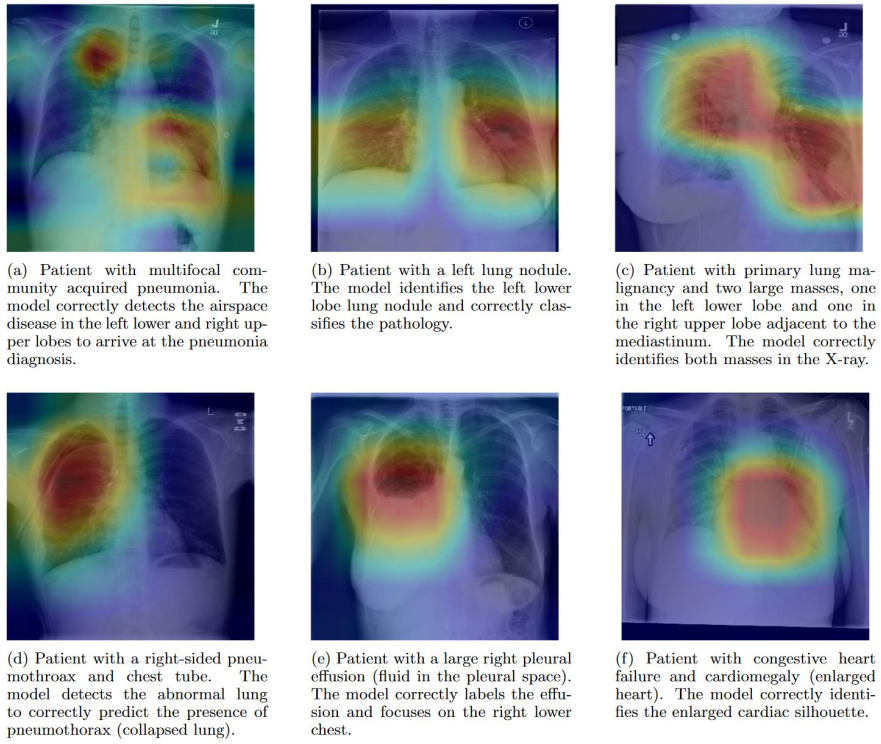
LOCATION = ???!!!

WEAKLY-SUPERVISED OBJECT LOCALIZATION



Class Activation Maps

Source: http://cnnlocalization.csail.mit.edu/Zhou_Learning_Deep_Features_CVPR_2016_paper.pdf



(a) Patient with multifocal community acquired pneumonia. The model correctly detects the airspace disease in the left lower and right upper lobes to arrive at the pneumonia diagnosis.

(b) Patient with a left lung nodule. The model identifies the left lower lobe lung nodule and correctly classifies the pathology.

(c) Patient with primary lung malignancy and two large masses, one in the left lower lobe and one in the right upper lobe adjacent to the mediastinum. The model correctly identifies both masses in the X-ray.

(d) Patient with a right-sided pneumothroax and chest tube. The model detects the abnormal lung to correctly predict the presence of pneumothorax (collapsed lung).

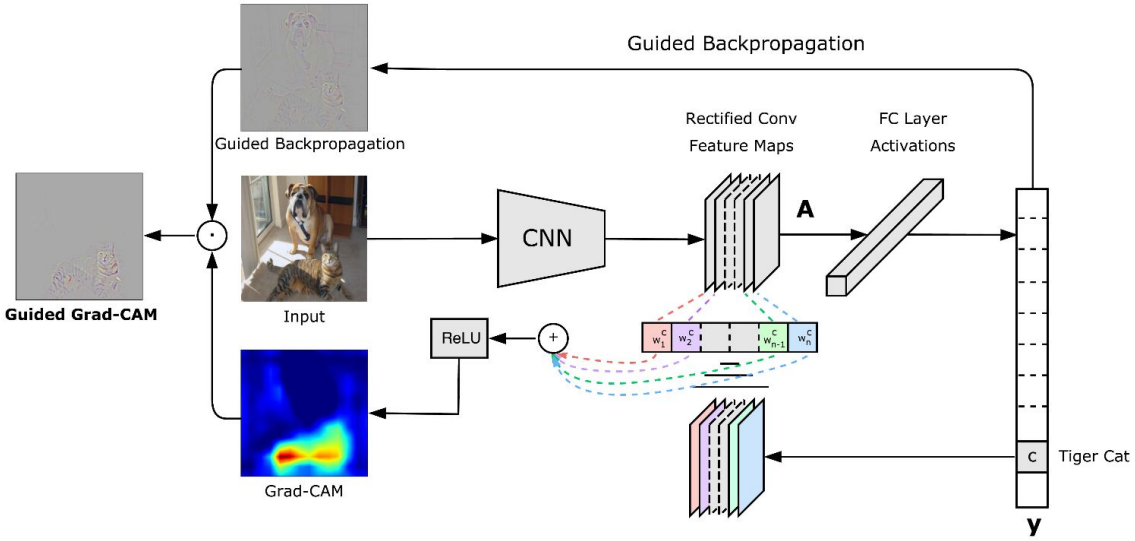
(e) Patient with a large right pleural effusion (fluid in the pleural space). The model correctly labels the effusion and focuses on the right lower chest.

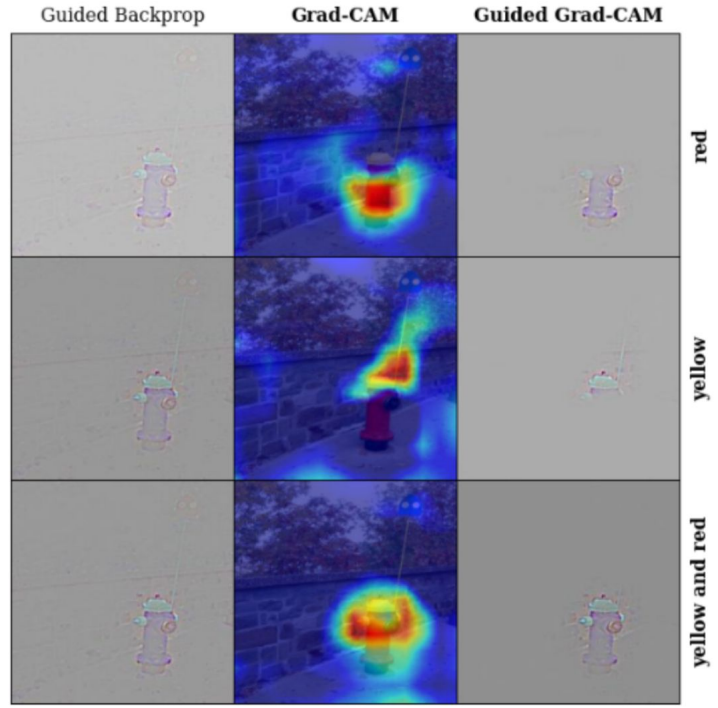
(f) Patient with congestive heart failure and cardiomegaly (enlarged heart). The model correctly identifies the enlarged cardiac silhouette.

CheXNet: Pneumonia Detection on Chest X-Rays

Source: https://arxiv.org/pdf/1711.05225.pdf

WEAKLY-SUPERVISED OBJECT LOCALIZATION





Grad-CAM and guided grad-CAM activation and saliency maps

Source: https://arxiv.org/pdf/1610.02391v1.pdf

PROBLEM

Partition an image into meaningful regions

Image source:
https://vision.in.tum.de/research/image_segmentation

SEMANTIC SEGMENTATION

Fully Convolutional Networks

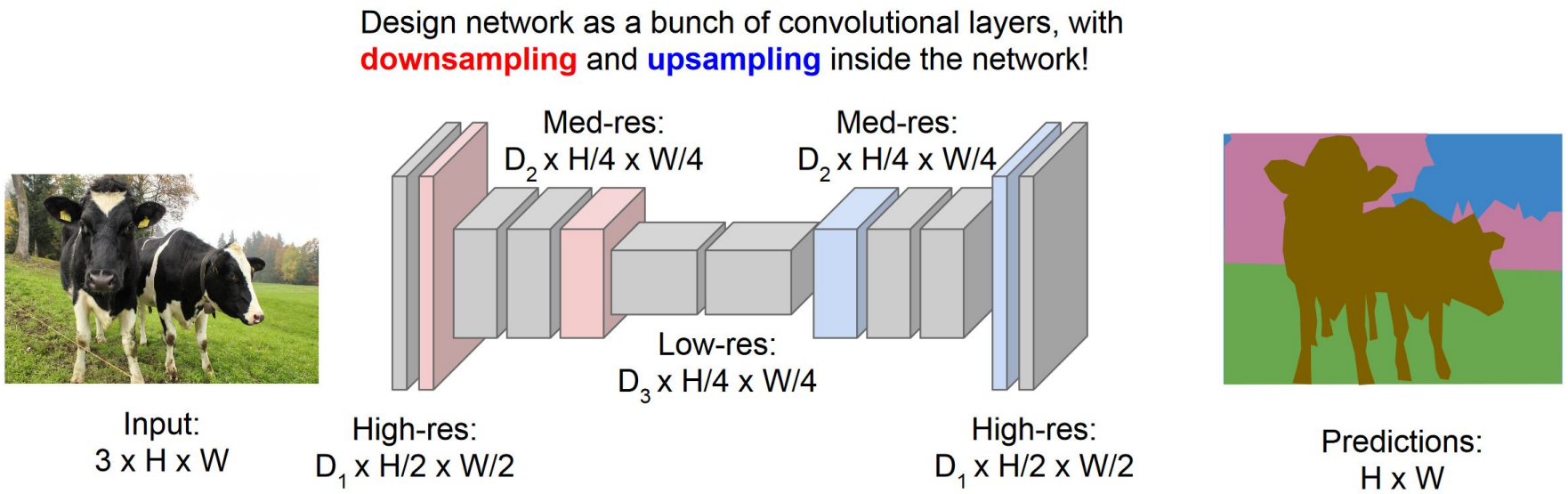


Image source:
http://cs231n.stanford.edu/slides/2018/cs231n_2018_lecture11.pdf

SEMANTIC SEGMENTATION

U-Net networks

- Commonly used architecture
- Performs well on low-size datasets
- Multiple applications in medic image processing research
- Unfortunately, quite slow

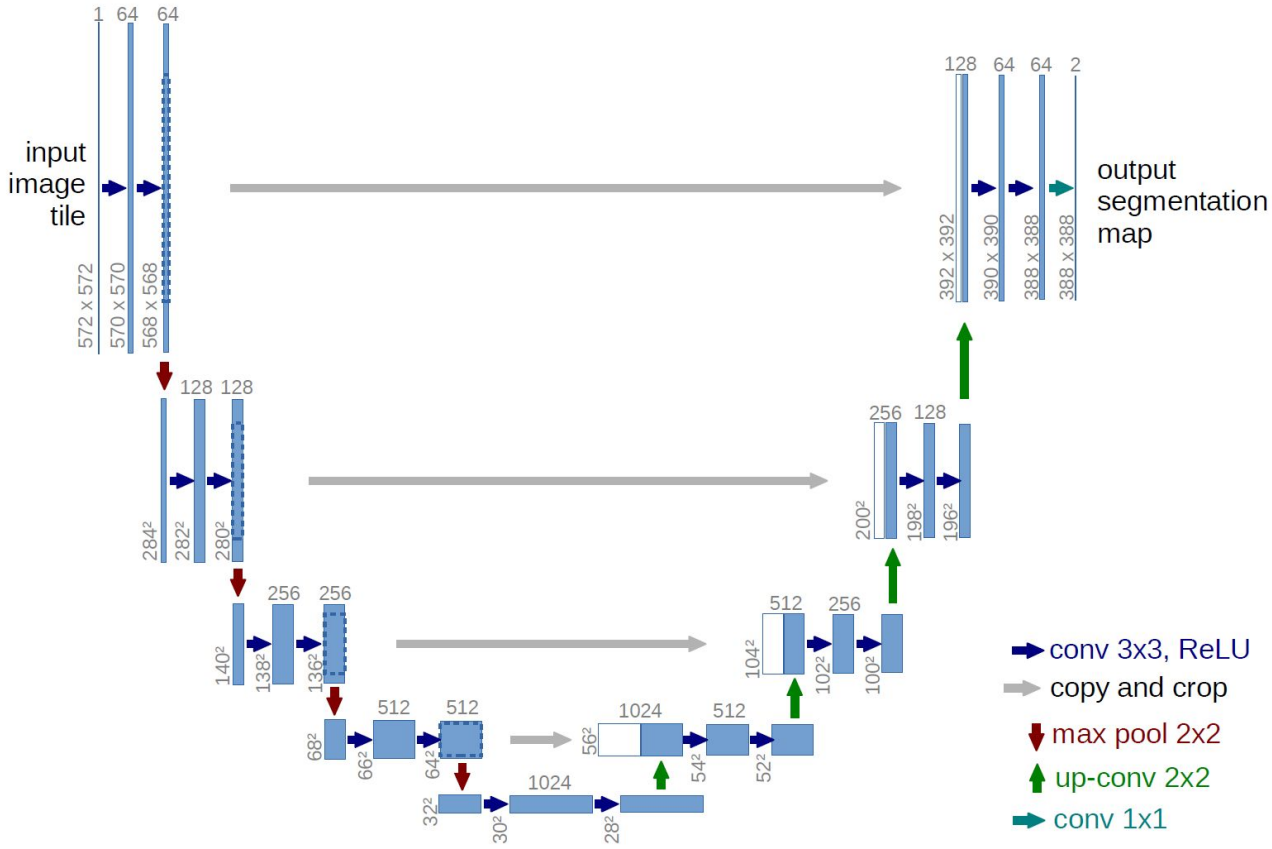


Image source:
https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/

PROBLEM

Find similar images for given image query

IMAGE RETRIEVAL / VISUAL SEARCH

Example: Pinterest Visual Search

- AlexNet and VGG bottlenecks
- „salient color signatures” (segmentation, color clustering)
- Object detection using text information

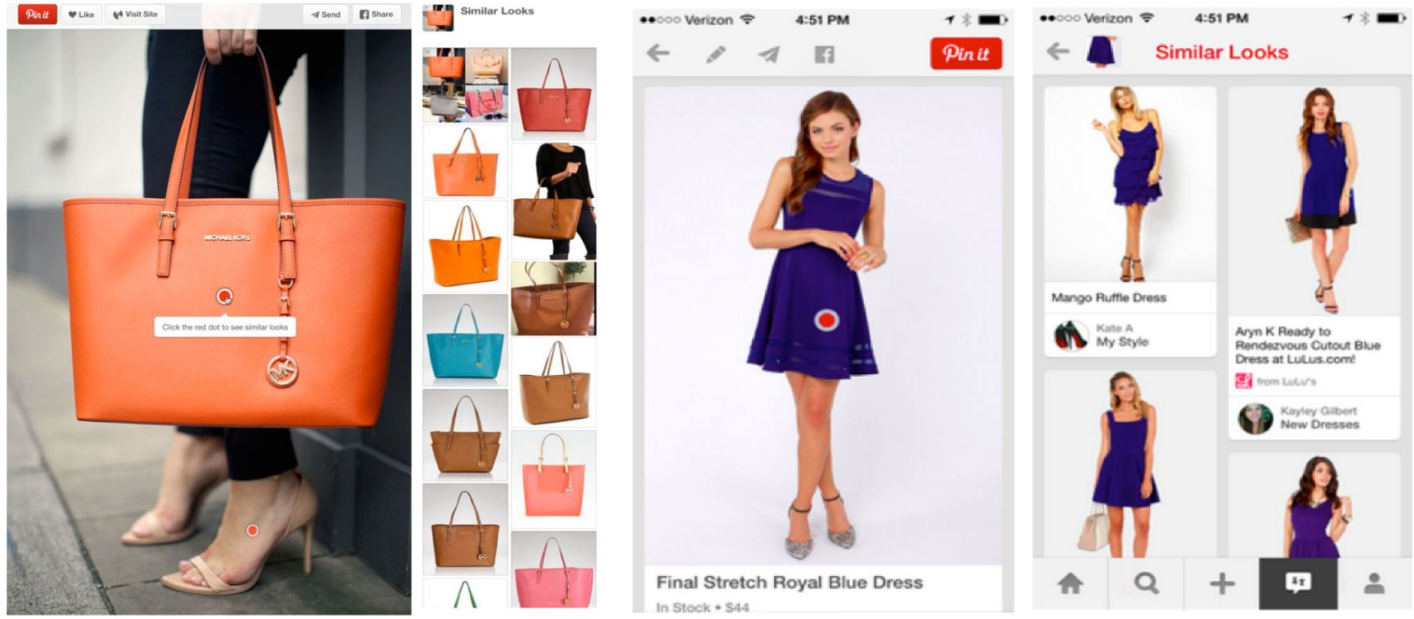


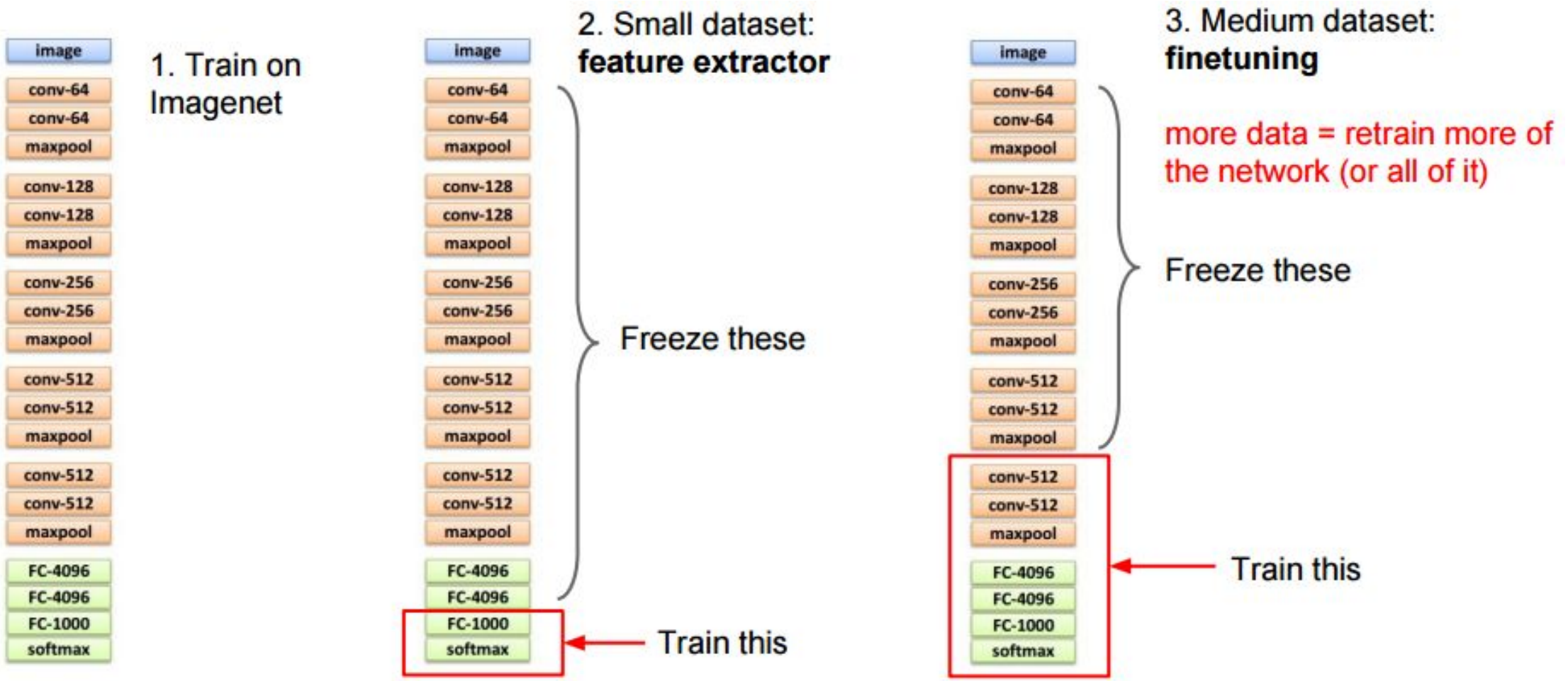
Image source:
https://arxiv.org/pdf/1505.07647.pdf

THERE IS MUCH MORE…

- Face recognition
- Generative models
- 3D reconstruction
- Pose estimation
- Odometry
- Object tracking
- …

TRANSFER LEARNING HANDS-ON

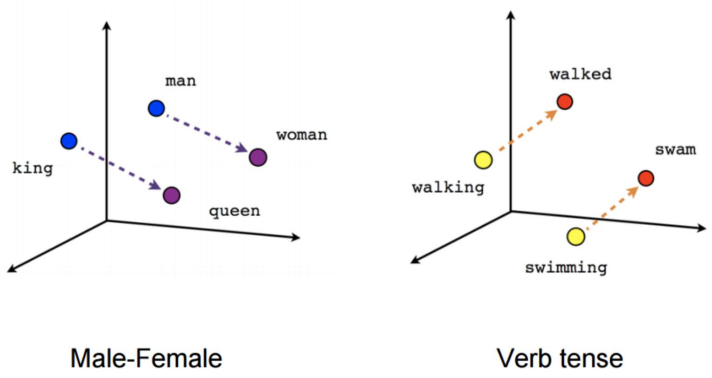
TRANSFER LEARNING



Source: http://cs231n.github.io/

TRANSFER LEARNING

Not only Computer Vision...



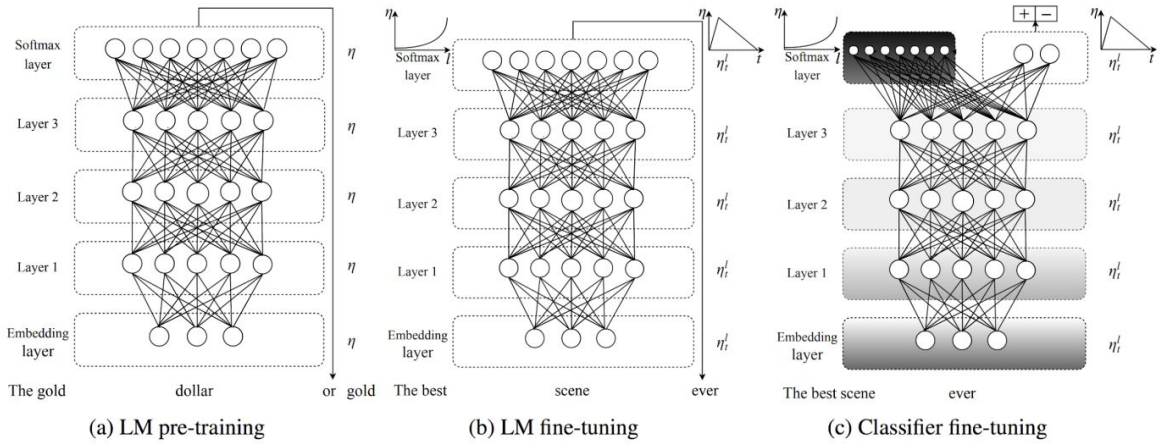
Male-Female Verb tense

Source: word2vec paper

OpenAI Retro Contest

April 5 to June 5, 2018

Source: OpenAI Retro Contest Results



(a) LM pre-training (b) LM fine-tuning (c) Classifier fine-tuning

Source: Universal Language Model Fine-tuning for Text Classification

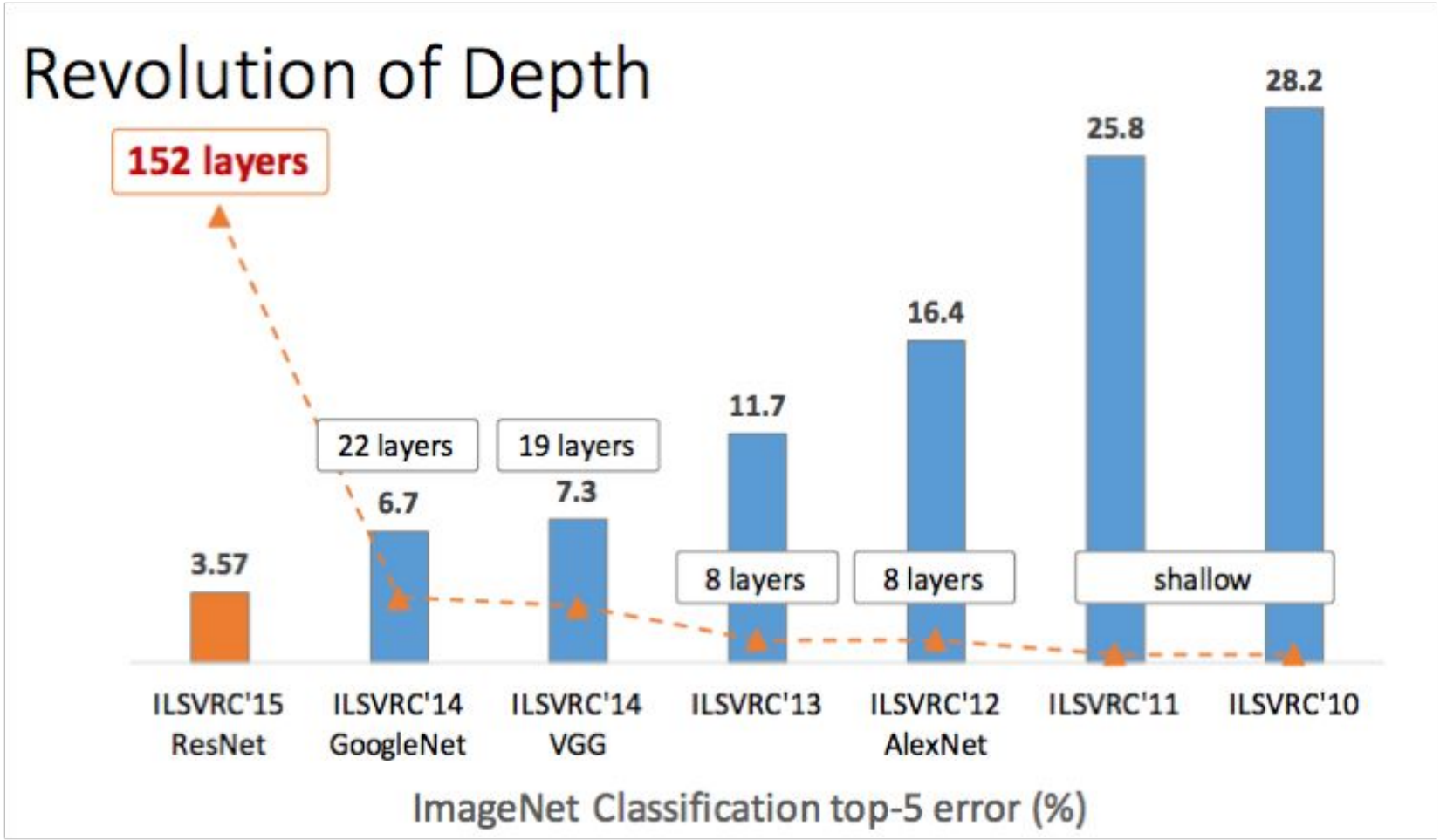
HANDS-ON

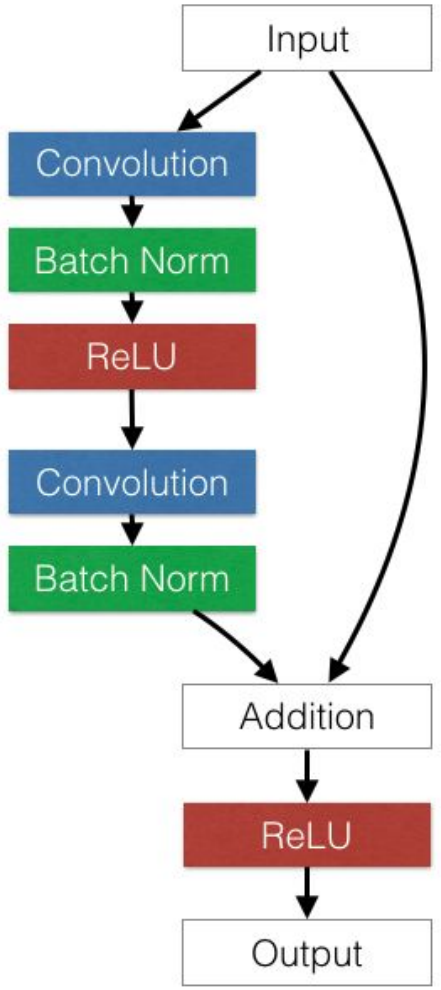
ImageNet

Transfer Learning

MIT Indoor 67

ResNet-152





Source: ResNet paper

TRANSFER LEARNING in KERAS

```
1  from keras.applications import VGG16
2
3  vgg_conv = VGG16(weights='imagenet',
4                   include_top=False,
5                   input_shape=(224, 224, 3))
```

Transfer learning is super easy in Keras, if you use pretrained models available in `keras.applications`.

ResNet-152 isn't there, so we will use another way to load pretrained model.

LET'S START THE HANDS-ON!

Repository:
https://github.com/gberinger/resnet-finetune-demo

Original code:
https://github.com/cta-ai/resnet-finetune-demo

Tutorial describing the code in detail (in Polish):
http://www.cta.ai/en/publications/02

READING MATERIALS

What to do next?

- **Stanford CS231n** - Convolutional Neural Networks for Visual Recognition:
  - Website - lecture notes etc.
  - YT lectures with Andrey Karpathy
- **Deep Learning Specialization** on Coursera with **Andrew Ng**
  - Specialization Info
  - Course 4: Convolutional Neural Networks
- Welch Labs - Learning to See
- A curated list of Awesome Computer Vision Resources
- Read links from this presentation if you're interested in particular application!

Thank you!