LMP 1210H: Basic Principles of Machine Learning in Biomedical Research

Bo Wang
AI Lead Scientist, PMCC, UHN
CIFAR AI Chair, Vector Institute
Assistant Professor, University of Toronto

Object recognition

- Object recognition is the task of identifying which object category is present in an image.
- It's challenging because objects can differ widely in position, size, shape, appearance, etc., and we have to deal with occlusions, lighting changes, etc.
- Why we care about it
  - Direct applications to image search
  - Closely related to object detection, the task of locating all instances of an object in an image
    - E.g., a self-driving car detecting pedestrians or stop signs
- For the past 6 years, all of the best object recognizers have been various kinds of conv nets.

Datasets

- In order to train and evaluate a machine learning system, we need to collect a dataset. The design of the dataset can have major implications.
- Some questions to consider:
    - Which categories to include?
    - Where should the images come from?
    - How many images to collect?
    - How to normalize (preprocess) the images?

Image Classification

- Conv nets are just one of many possible approaches to image classification. However, they have been by far the most successful for the last 8 years.
- Biggest image classification "advances" of the last two decades
  - Datasets have gotten much larger (because of digital cameras and the Internet)
  - Computers got much faster
    - Graphics processing units (GPUs) turned out to be really good at training big neural nets; they're generally about 30 times faster than CPUs.
  - As a result, we could fit bigger and bigger neural nets.

MNIST Dataset

- MNIST dataset of handwritten digits
  - **Categories:** 10 digit classes
  - **Source:** Scans of handwritten zip codes from envelopes
  - **Size:** 60,000 training images and 10,000 test images, grayscale, of size 28×28
  - **Normalization:** centered within in the image, scaled to a consistent size
    - The assumption is that the digit recognizer would be part of a larger pipeline that segments and normalizes images.
- In 1998, Yann LeCun and colleagues built a conv net called LeNet which was able to classify digits with 98.9% test accuracy.
  - It was good enough to be used in a system for automatically reading numbers on checks.

ImageNet

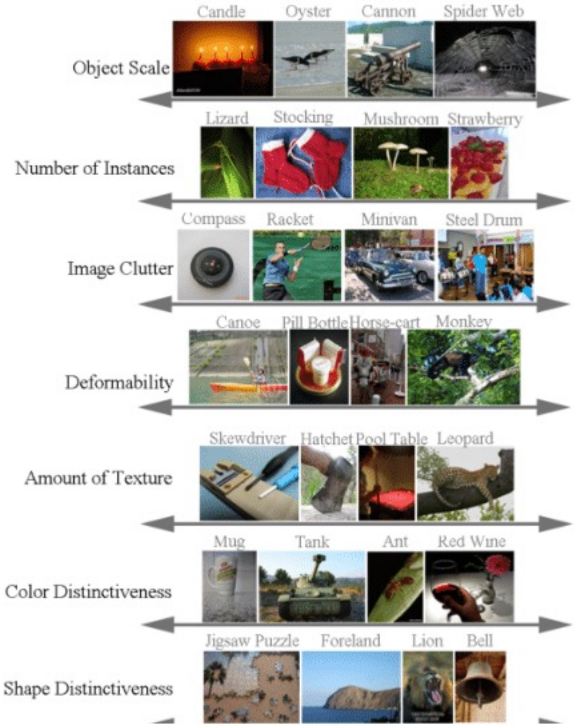
ImageNet is the modern object recognition benchmark dataset. It was introduced in 2009, and has led to amazing progress in object recognition since then.

ImageNet

- Used for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), an annual benchmark competition for object recognition algorithms
- Design decisions
  - **Categories:** Taken from a lexical database called WordNet
    - WordNet consists of "synsets", or sets of synonymous words
    - They tried to use as many of these as possible; almost 22,000 as of 2010
    - Of these, they chose the 1000 most common for the ILSVRC
    - The categories are really specific, e.g. hundreds of kinds of dogs
  - **Size:** 1.2 million full-sized images for the ILSVRC
  - **Source:** Results from image search engines, hand-labeled by Mechanical Turkers
    - Labeling such specific categories was challenging; annotators had to be given the WordNet hierarchy, Wikipedia, etc.
  - **Normalization:** none, although the contestants are free to do preprocessing

ImageNet

Images and object categories vary on a lot of dimensions



Russakovsky et al.

ImageNet

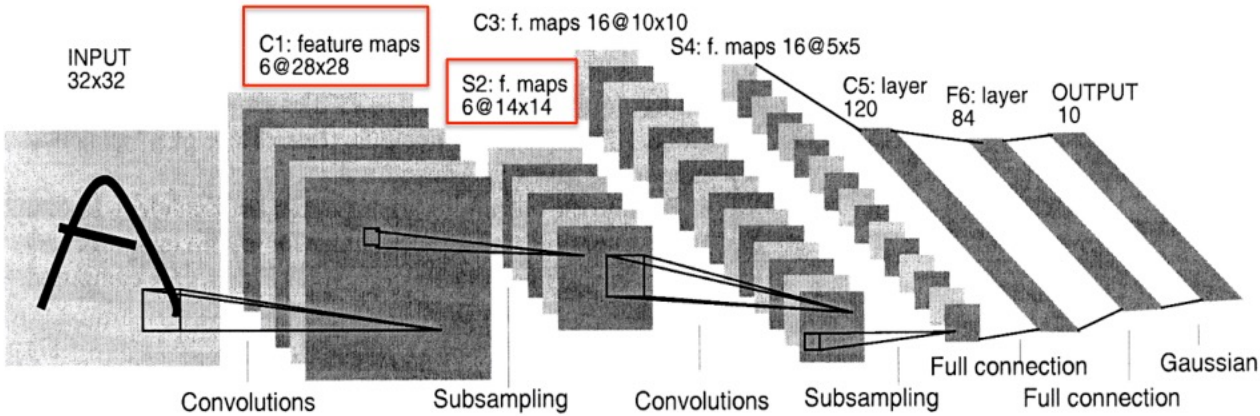
Size on disk:

MNIST
60 MB

ImageNet
50 GB

Case Study: LeNet

Here's the LeNet architecture, which was applied to handwritten digit recognition on MNIST in 1998:



Size of a Conv Net

Sizes of layers in LeNet:

Layer	Type	# units	# connections	# weights
C1	convolution	4704	117,600	150
S2	pooling	1176	4704	0
C3	convolution	1600	240,000	2400
S4	pooling	400	1600	0
F5	fully connected	120	48,000	48,000
F6	fully connected	84	10,080	10,080
output	fully connected	10	840	840

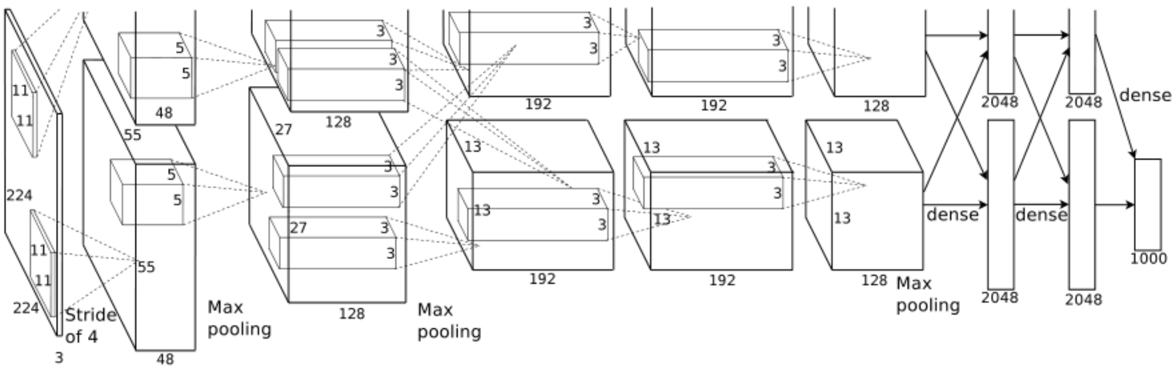
Conclusions?

Size of a Conv Net

- Rules of thumb:
  - Most of the units and connections are in the convolution layers.
  - Most of the weights are in the fully connected layers.
- If you try to make layers larger, you'll run up against various resource limitations (i.e. computation time, memory)
- Conv nets have gotten a LOT larger since 1998!

Case Study: AlexNet

- AlexNet, 2012. 8 weight layers. 16.4% top-5 error (i.e. the network gets 5 tries to guess the right category).



(Krizhevsky et al., 2012)

- They used lots of tricks we've covered in this course (ReLU units, weight decay, data augmentation, SGD with momentum, dropout)
- AlexNet's stunning performance on the ILSVRC is what set off the deep learning boom of the last 6 years.

Size of a Conv Net

	LeNet (1989)	LeNet (1998)	AlexNet (2012)
classification task	digits	digits	objects
categories	10	10	1,000
image size	16×16	28×28	$256 \times 256 \times 3$
training examples	7,291	60,000	1.2 million
units	1,256	8,084	658,000
parameters	9,760	60,000	60 million
connections	65,000	344,000	652 million

GoogLeNet

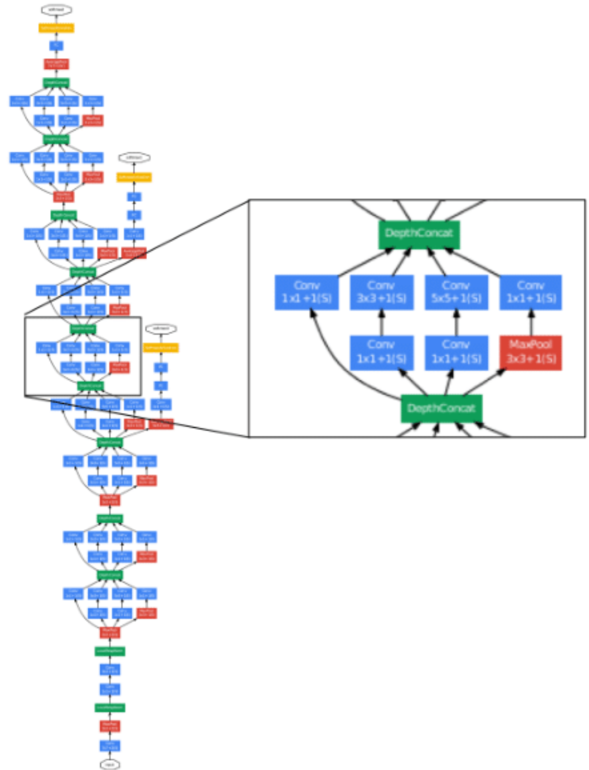
GoogLeNet, 2014.

22 weight layers

Fully convolutional (no fully connected layers)

Convolutions are broken down into a bunch of smaller convolutions

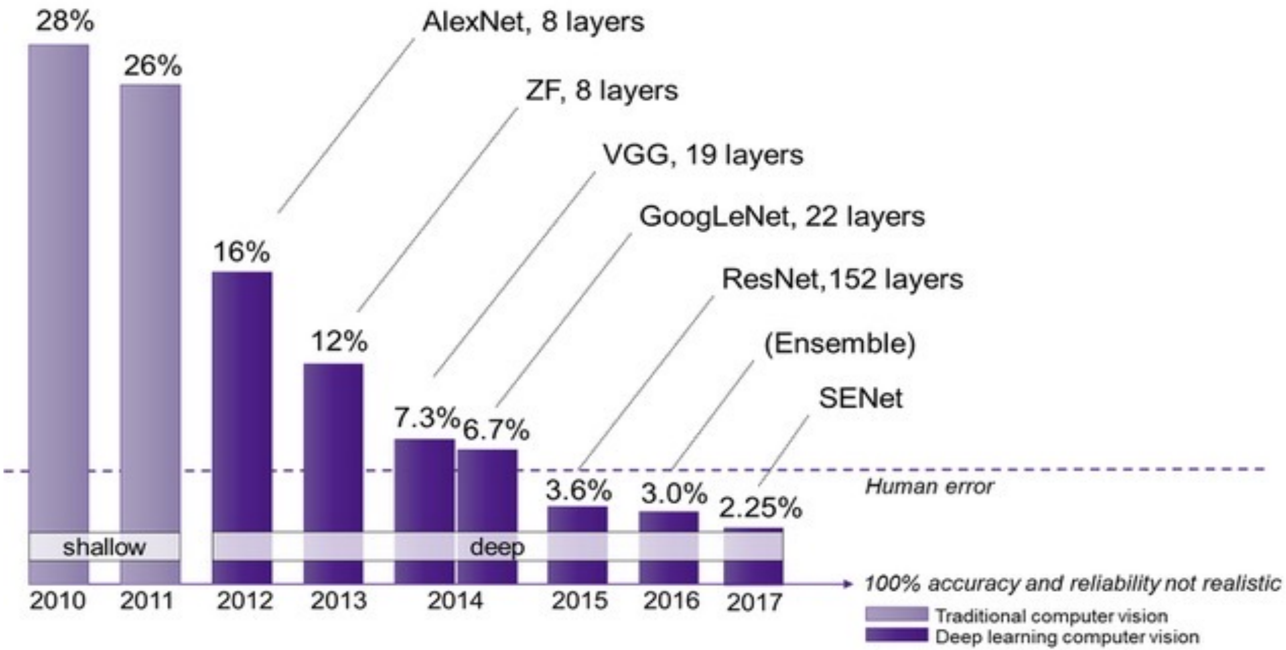
6.6% test error on ImageNet



GoogLeNet

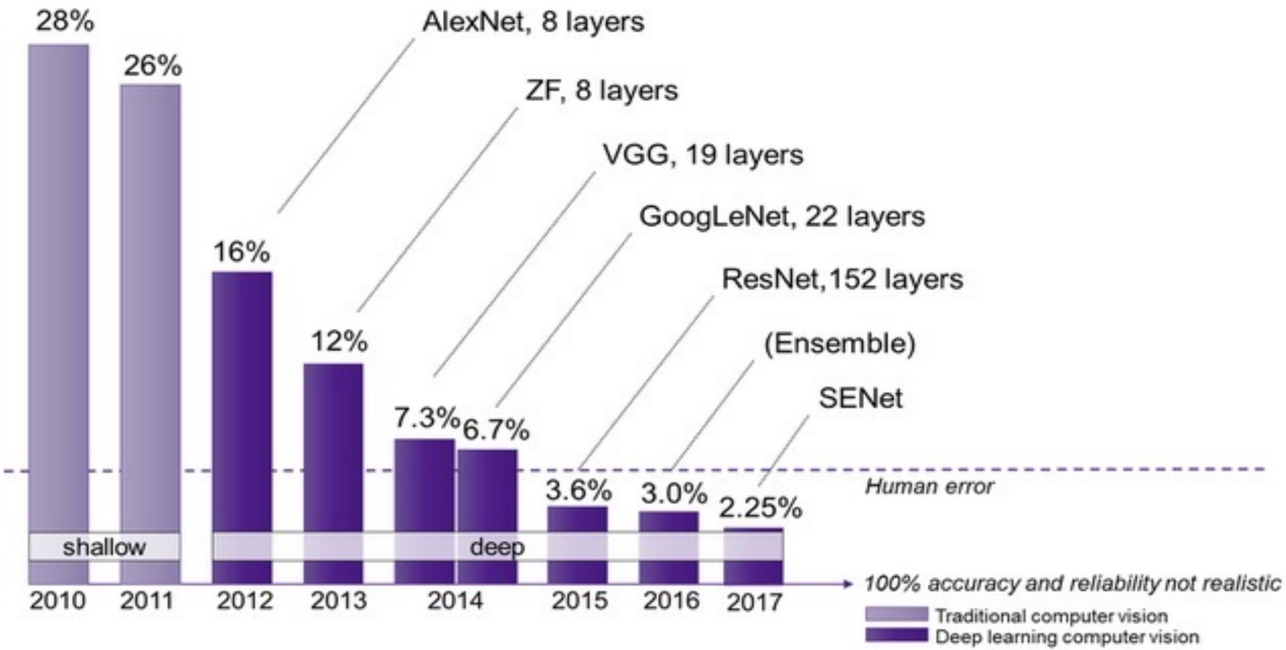
- They were really aggressive about cutting the number of parameters.
  - Motivation: train the network on a large cluster, run it on a cell phone
    - Memory at test time is the big constraint.
    - Having lots of units is OK, since the activations only need to be stored at training time (for backpropagation).
    - Parameters need to be stored both at training and test time, so these are the memory bottleneck.
  - How they did it
    - No fully connected layers (remember, these have most of the weights)
    - Break down convolutions into multiple smaller convolutions (since this requires fewer parameters total)
  - GoogLeNet has "only" 2 million parameters, compared with 60 million for AlexNet
  - This turned out to improve generalization as well. (Overfitting can still be a problem, even with over a million images!)

Classification --- A brief history of ImageNet Competition



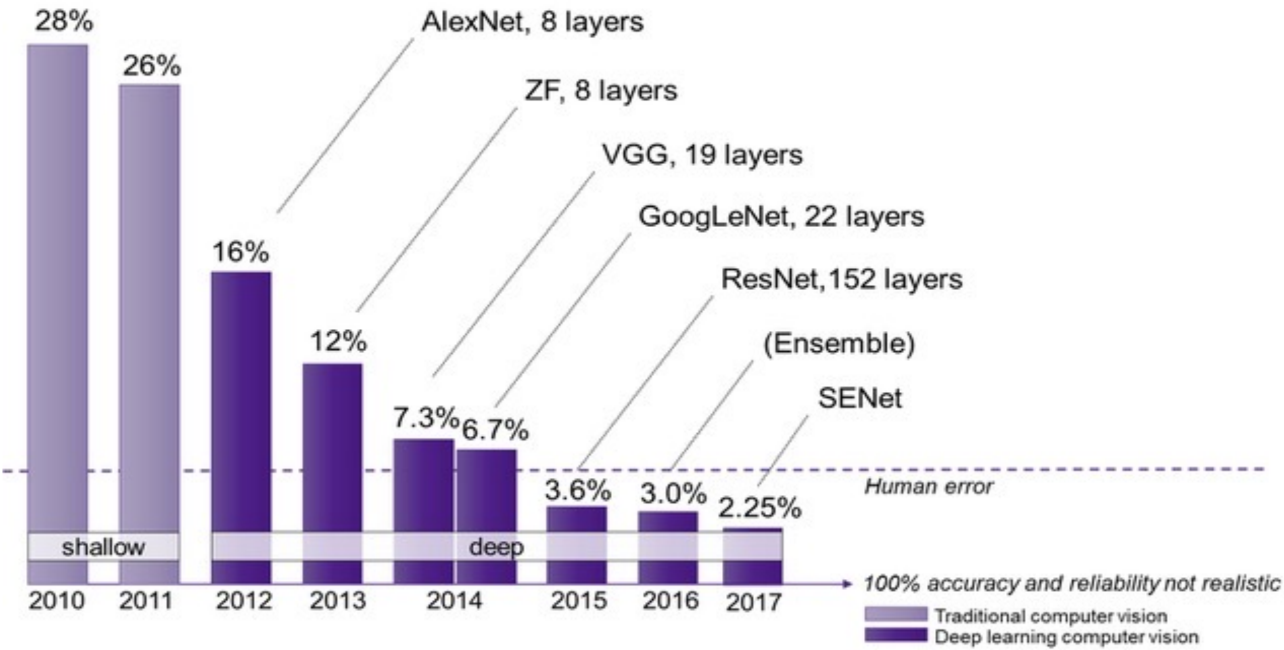
Observation 1: Deep models (mostly CNNs) dominate the task of object recognition.

Classification --- A brief history of ImageNet Competition



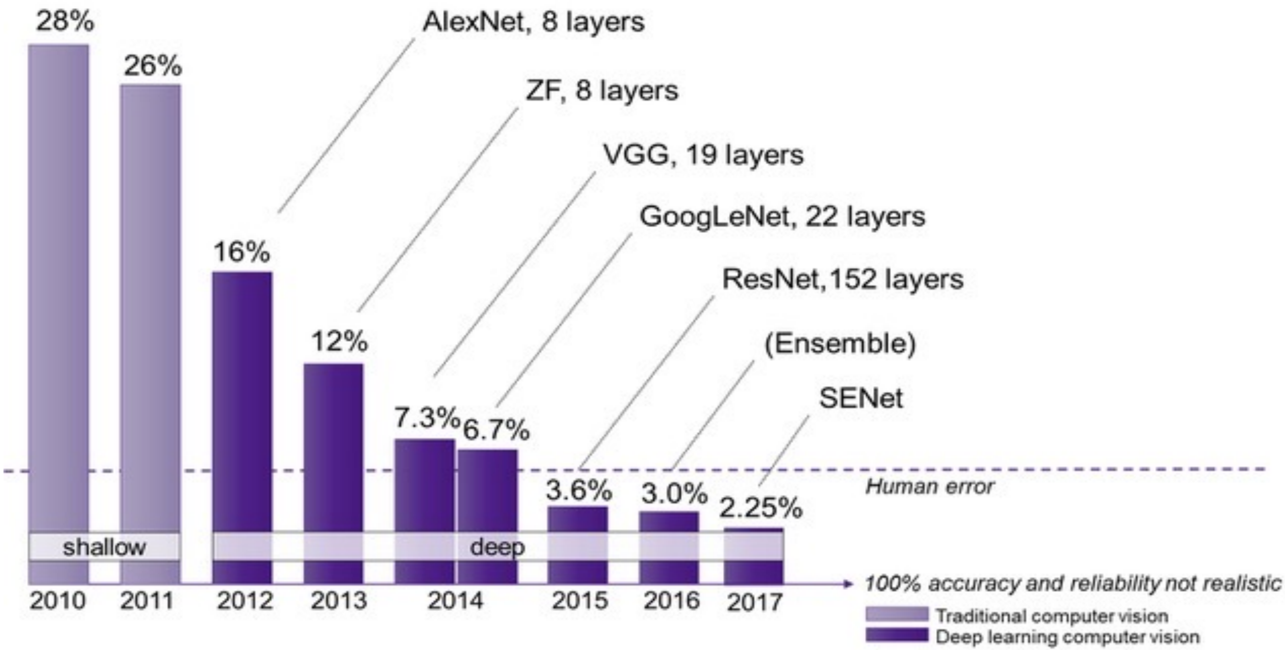
Observation 2: The accuracy by CNNs improves every year and even outperforms human performances.

Classification --- A brief history of ImageNet Competition



Observation 3: The STOA CNNs are getting deeper and deeper.

Classification --- A brief history of ImageNet Competition



Observation 4: The performance of object recognition is so good that they stopped the competition nowadays.

Beyond Classification

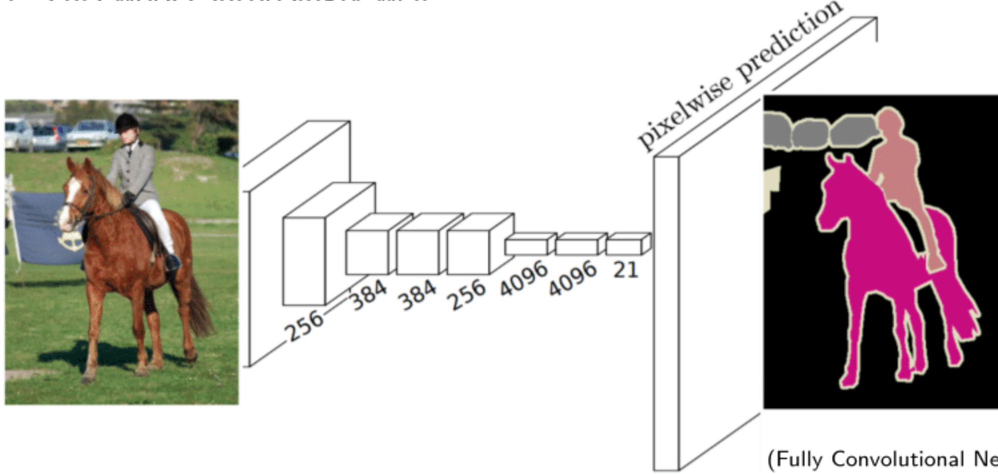
- The classification nets map the entire input image to a pre-defined class categories.
- But there are more than just class labels in an image.
  - where is the foreground object? how many? what is in the background?

{ Horse, Person, Car, }

(PASCAL VOC 2012)

Semantic Segmentation

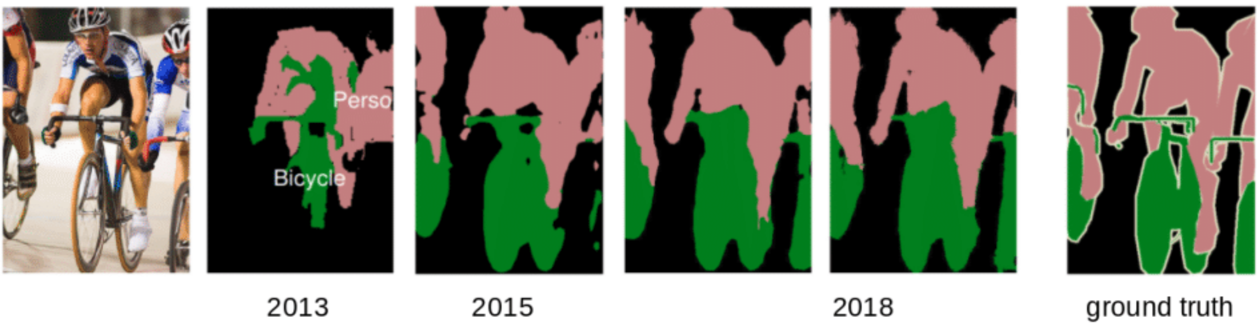
- Semantic segmentation, a natural extention of classification, focuses on making dense classification of class labels for every pixel.
- It is an important step towards complete scene understanding in compter vision.
  - Semantic segmentation is a stepping stone for many of the high-level vision tasks, such as object detection, Visual Question Answering (VQA).
- A naive approach is to adapt the existing object classification conv nets for each pixel. This works surprisingly well



(Fully Convolutional Networks, 2015)

Semantic Segmentation

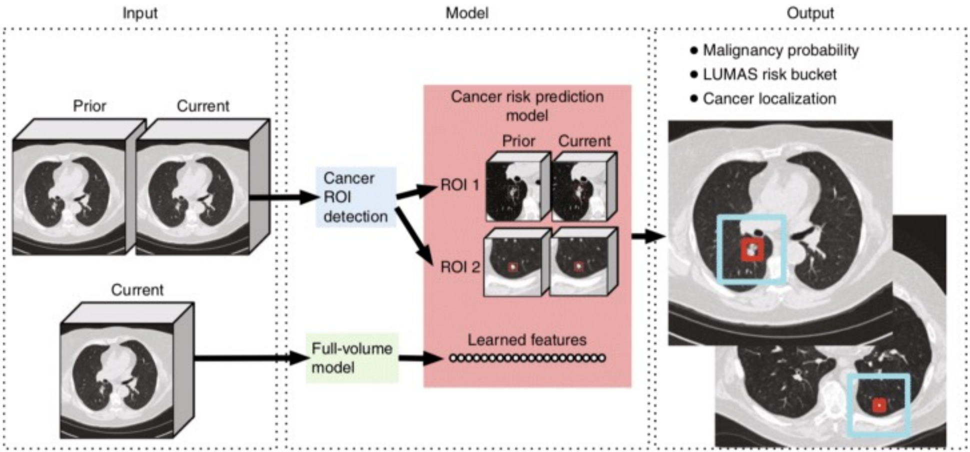
- After the success of CNN classifiers, segmentation models quickly moved away from hand-craft features and pipelines but instead use CNN as the main structure.
- Pre-trained ImageNet classification network serves as a building block for all the state-of-the-art CNN-based segmentation models.



from left to wright (Li, et. al., (CSI), CVPR, 2013; Long, et. al., (FCN), CVPR 2015; Chen et. al., (DeepLab), PAMI 2018)

CNN on HealthCare

- CNN has also been widely used in processing medical images.
- Pre-trained ImageNet classification network serves as a building block for most of medical image segmentation/classification models.

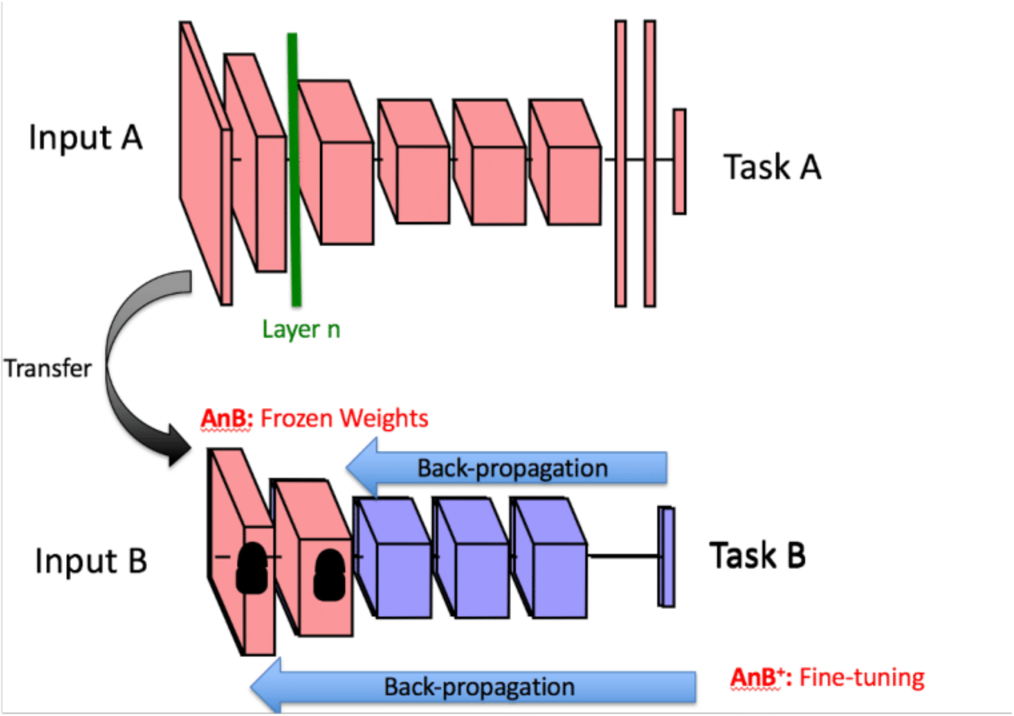


from: google AI: End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. Nature medicine, 2019.

Supervised Pre-training and Transfer Learning

- In practice, we will rarely train an image classifier from scratch.
  - It is unlikely we will have millions of cleanly labeled images for our specific datasets.
- If the dataset is a computer vision task, it is common to fine-tune a pre-trained conv net on ImageNet or OpenImage.
- Just like semantic segmentation tasks, we will fix most of the weights in the pre-trained network. Only the weights in the last layer will be randomly initialized and learnt on the current dataset/task.

Transfer Learning : Fine-Tuning



Supervised Pre-training and Transfer Learning

- When to fine-tune?
  - How many training examples we have in the new dataset/task?
    - Fewer new examples: more weights from the pre-trained networks are fixed.
  - How similar is the new dataset to our pre-training dataset? Microspy images v.s. natural images:
    - more fine-tuning is needed for dissimilar datasets.
  - Learning rate for the fine-tuning stage is often much lower than the learning rate used for training from scratch.

Closing Thoughts

1. Convolutional Layers are the building blocks of computer vision applications (e.g., object recognition/detection/segmentation)!
2. Convolutional Layers use tied weights and obtain translation invariance.
3. CNNs (or its variants) are the go-to methods for many image-related applications.
4. CNNs can be also interpreted as a ***feature/representation learning*** method for images.
5. Transfer learning by fine-tuning is often used in training CNNs.