CS5489
Lecture 11.1: Neural Networks and Deep Learning Part V: High-Level Vision Applications

Kede Ma

City University of Hong Kong (Dongguan)

Slide template by courtesy of Benjamin M. Marlin

Outline

Image Classification

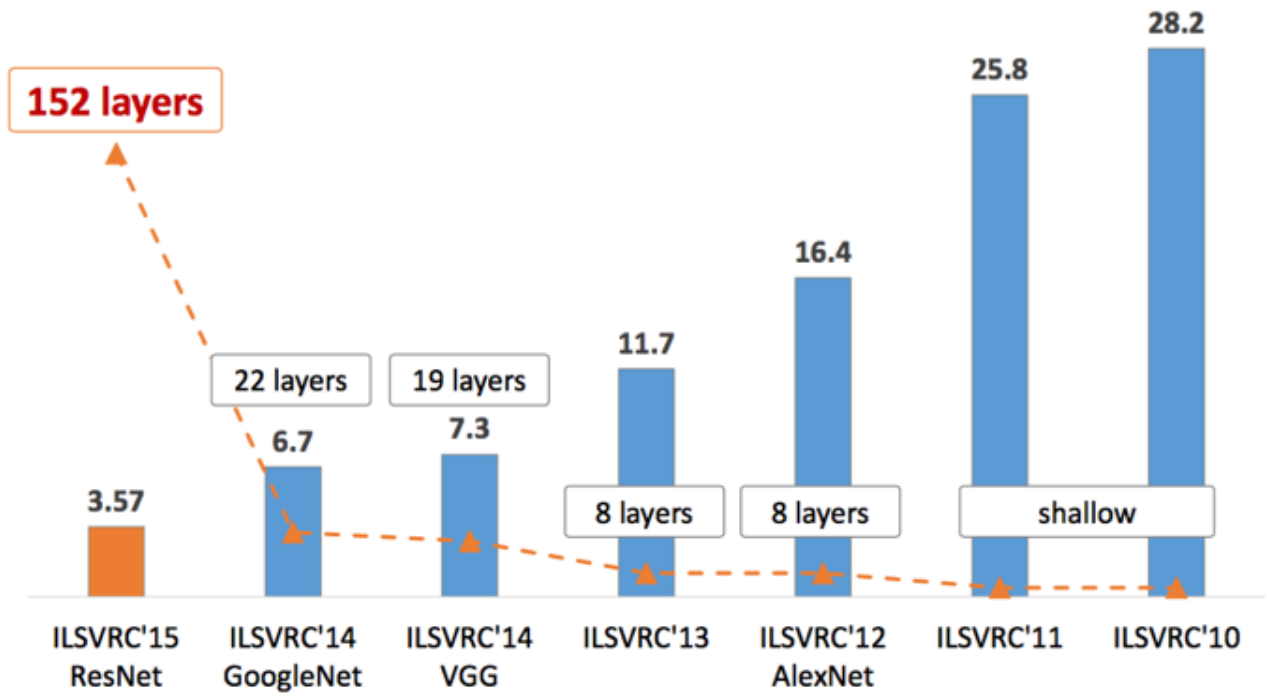
- Goal: Given a photographic image, predict the object class (a.k.a., object recognition)
  - Typically only one main object is present
  - If enough classes are considered, then it's a generic high-level vision task

ImageNet

- ImageNet Large Scale Visual Recognition Challenge (ILSVRC)
  - 1,000 image classes
  - 1.2 million images
  - Human performance: 5.1%?

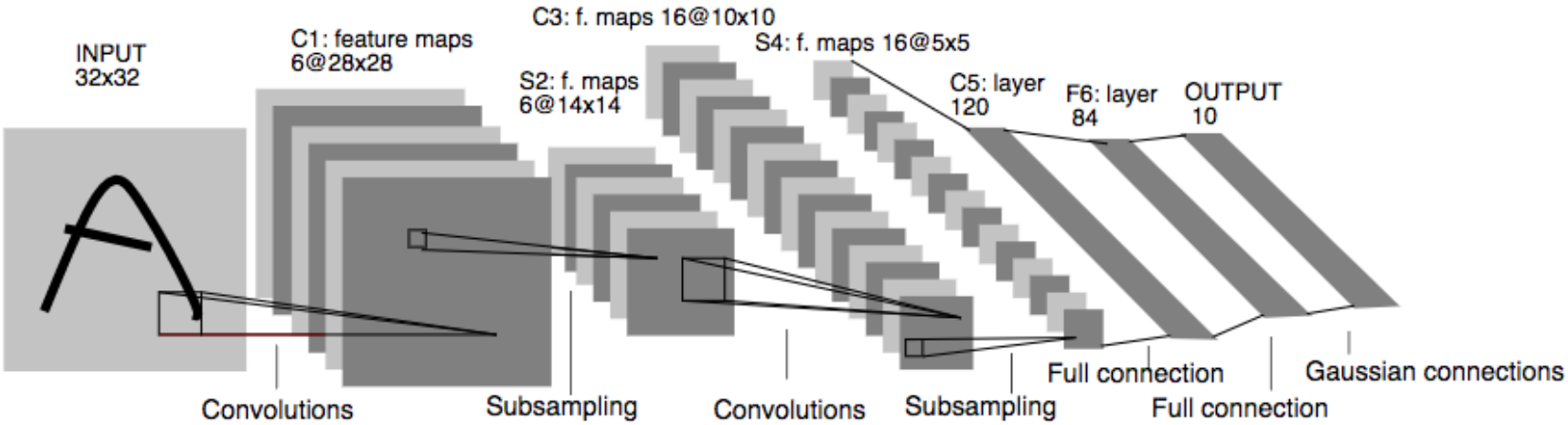
Performance of Deep Learning

- The introduction of ILSVRC coincided with the emergence of deep learning
- Top-5 error rate decreased as deeper NNs were developed
  - Not just deeper, but also the architecture design was smarter



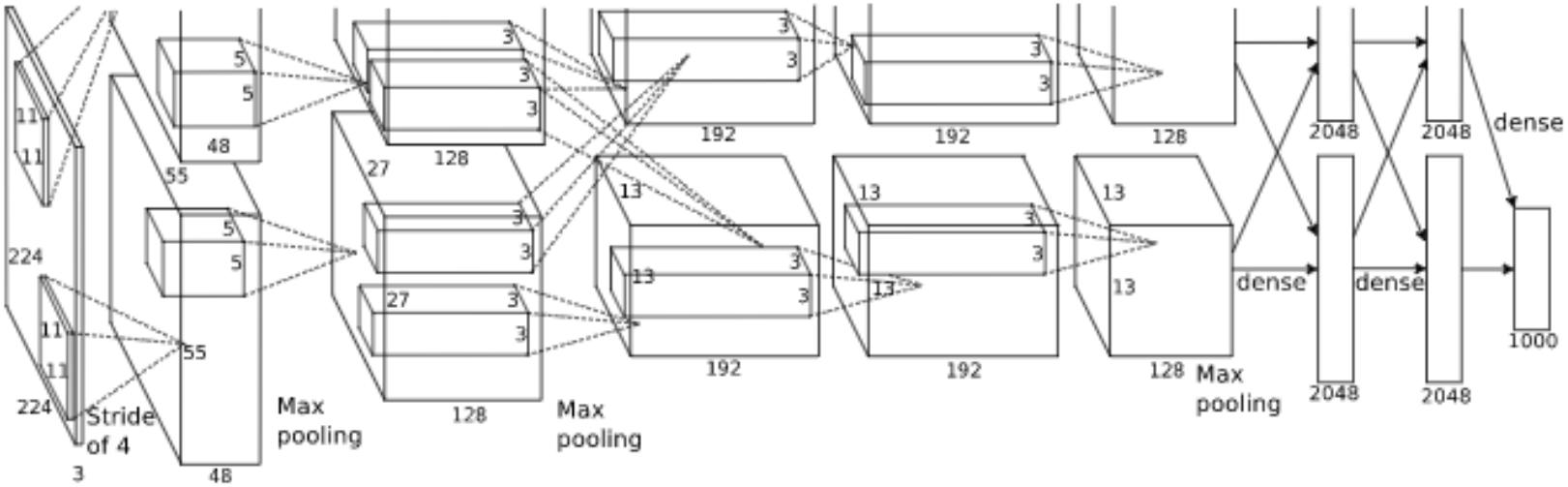
LeNet-5 (1998)

- The standard CNN architecture
  - 7 layers
  - Convolutions & pooling, final fully-connected layer
- Designed for hand-written digit recongition (MNIST)



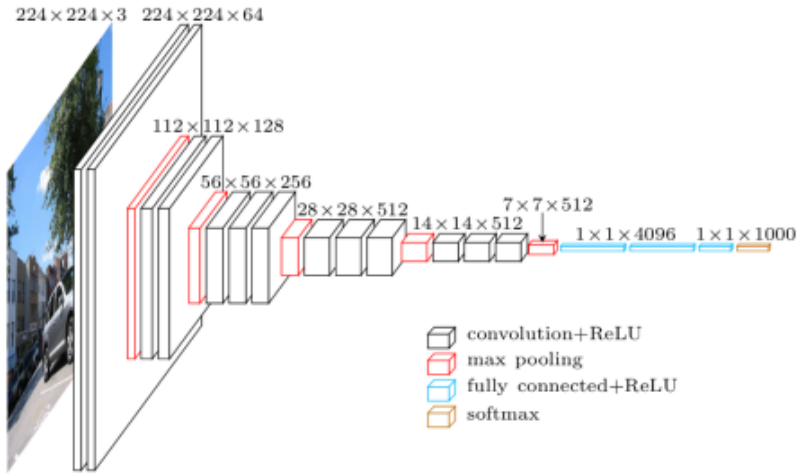
AlexNet

- Similar architecture to LeNet, but deeper (14 layers)
  - 11×11, 5×5, 3×3 convolutions
- Tricks used: ReLU, local response normalization, dropout, max pooling, data augmentation, SGD momentum
- One of the first networks trained on GPUs
  - It is split in two pipelines because it was trained on 2 GPUs simultaneously



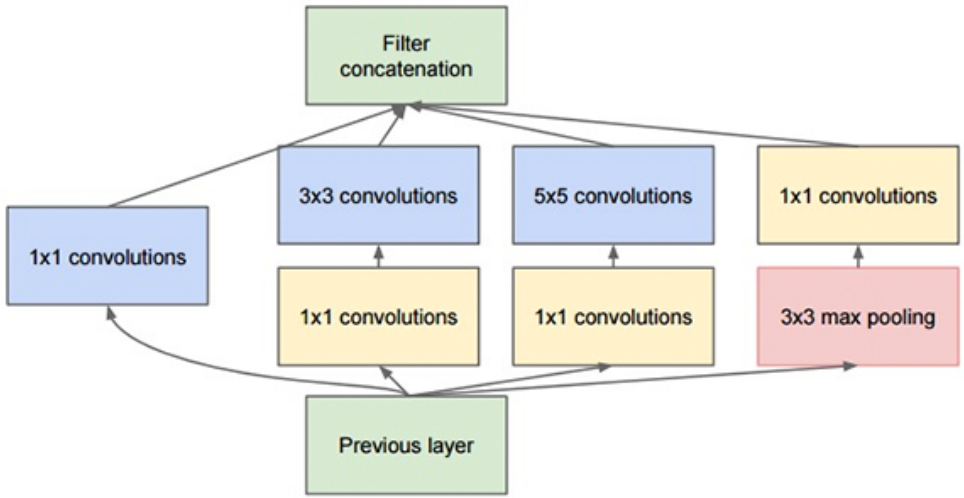
VGGNet

- Same style as LeNet and AlexNet
- Design choices:
  - Only use 3×3 convolution filters (less parameters)
  - Stack several convolution layers, followed by pooling
  - Number of feature channels doubles after each stage
    - More higher-level features
    - VGG features are very effective in modeling aspects of human perception!



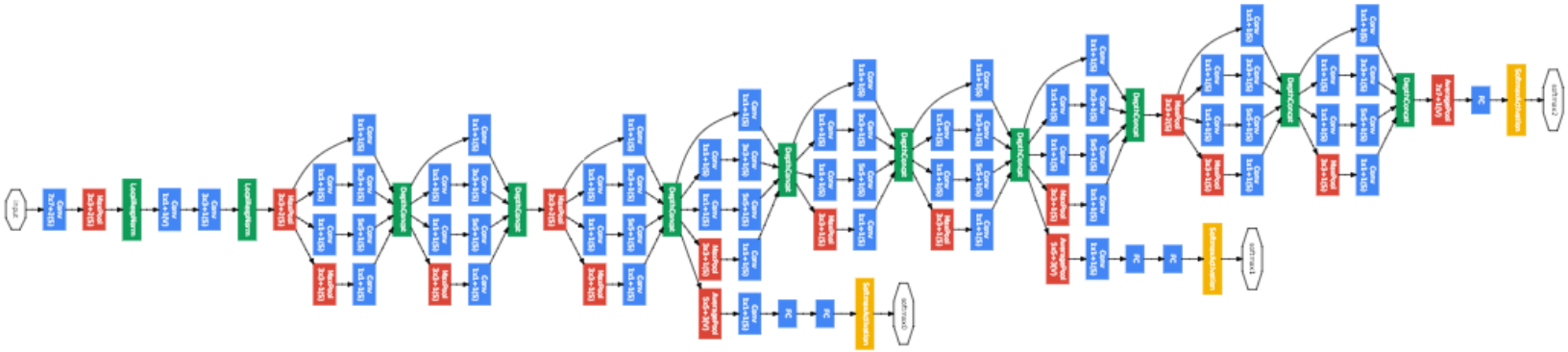
Inception Module

- "Network-in-Network" architecture
  - Several convolution filters in parallel
    - Extract features at different scales (1×1, 3×3, 5×5)
    - Pool features (3×3 max)
  - Features are concatenated and passed to next block



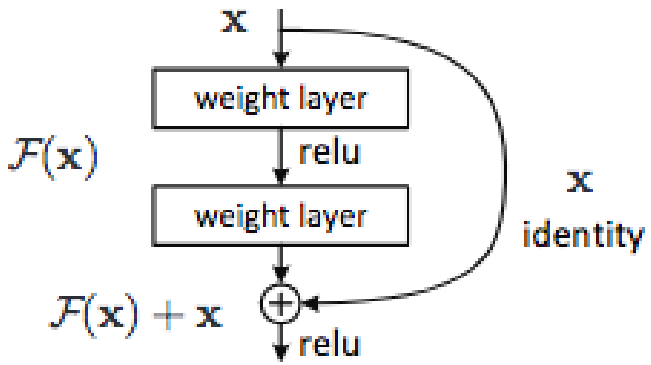
InceptionNet (V1)

- 9 inception modules, 22 layers
  - 50 convolution blocks
- Auxiliary classification tasks
  - Using features in the middle of the network to perform classification
  - Strengthen supervisory signals to the middle and earlier layers



Residual Learning

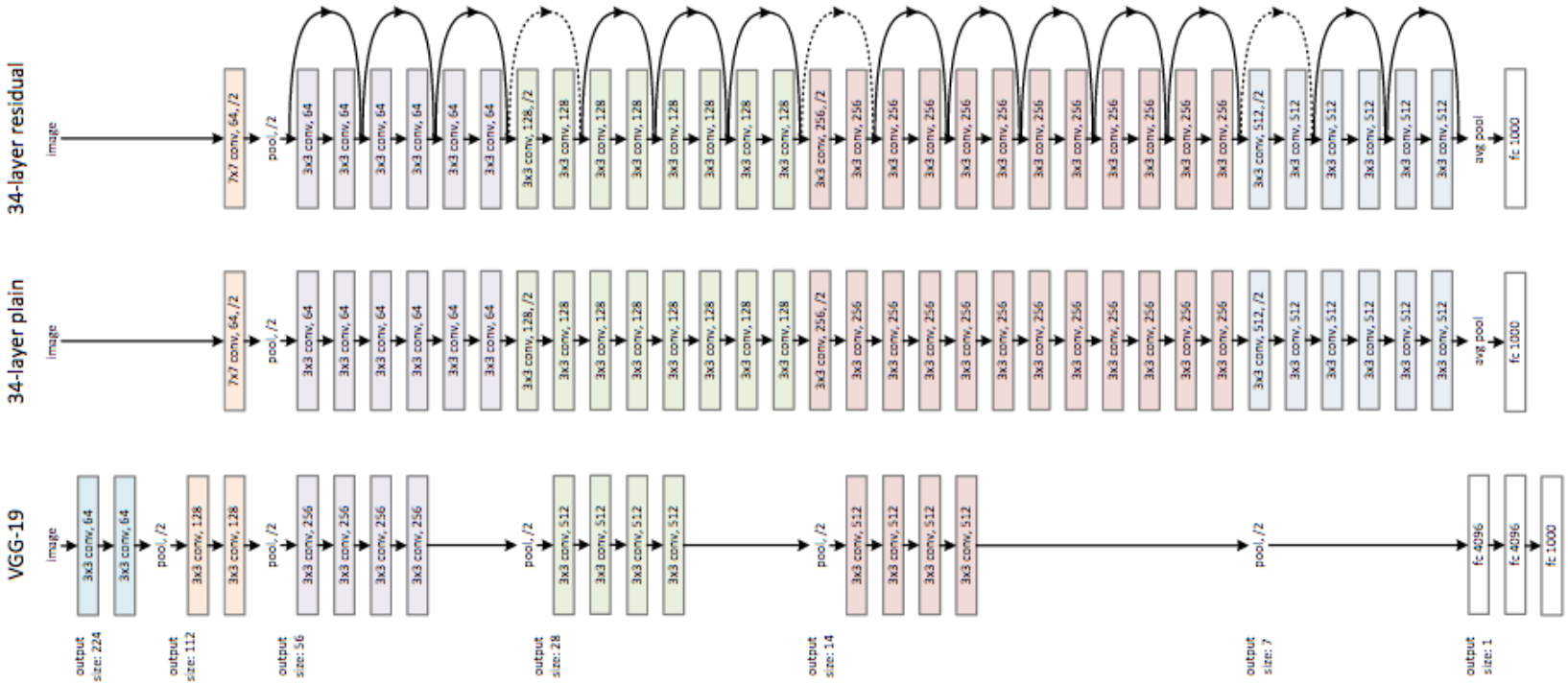
- The network is learning a function (image to class)
  - Build the function block-by-block
  - Each block learns a residual, which is added to the previous block
  - Keep all the previous information and make small changes with the residual
  - Novel interpretation: ResNets behave like ensembles of relatively shallow networks



A. Veit, M. Wilber, and S. Belongie, "Residual Networks Behave Like Ensembles of Relatively Shallow Networks," *arXiv preprint*, 2016.

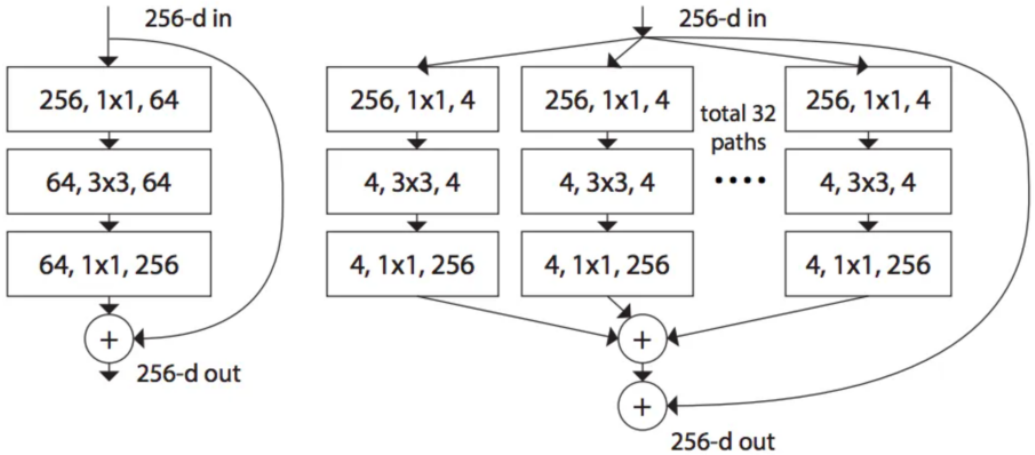
Residual Network (ResNet)

- 34 layers, 50 layers, 100 layers, 1,000 layers
  - 3×3 filters
  - Residual connection occurs every two layers



ResNeXt

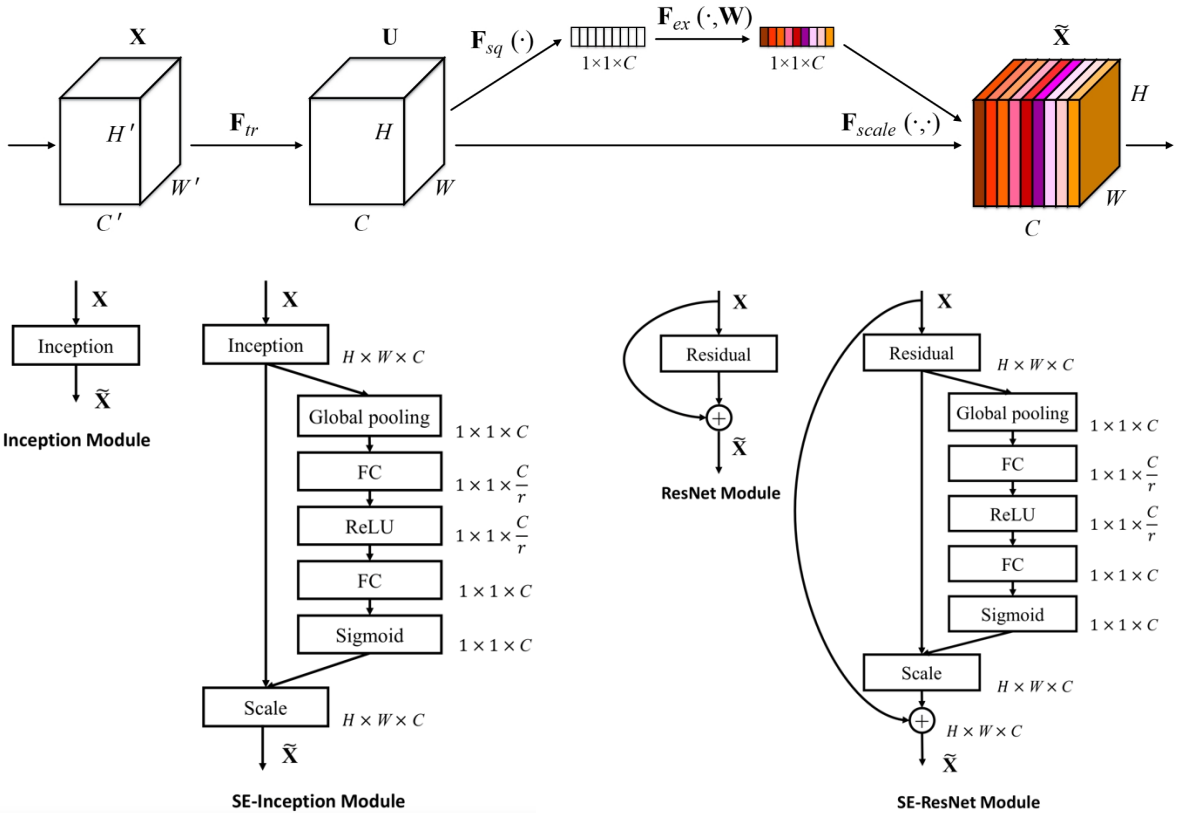
- ResNeXt combines the split-transform-aggregate strategy in the Inception network and the residual learning in ResNet
  - The number of paths inside the ResNeXt block is defined as **cardinality**
  - All the paths contain the same topology
  - Instead of having high depth and width, having high cardinality helps in decreasing validation error



Left: A block of ResNet. **Right**: A block of ResNeXt with cardinality = 32. A layer is show as (# in channels, filter size, # out channels)

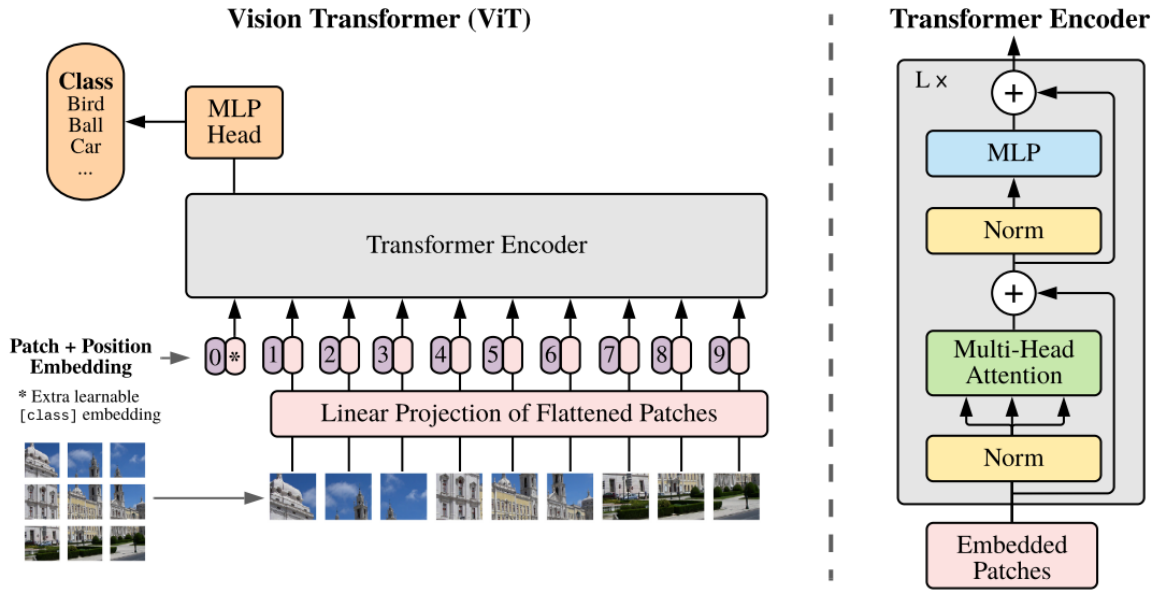
Squeeze-and-Excitation Networks (SENets)

- A block for CNNs that improves channel interdependencies
  - Add parameters to each channel of a convolutional block so that the network adaptively adjusts the weighting of each feature map



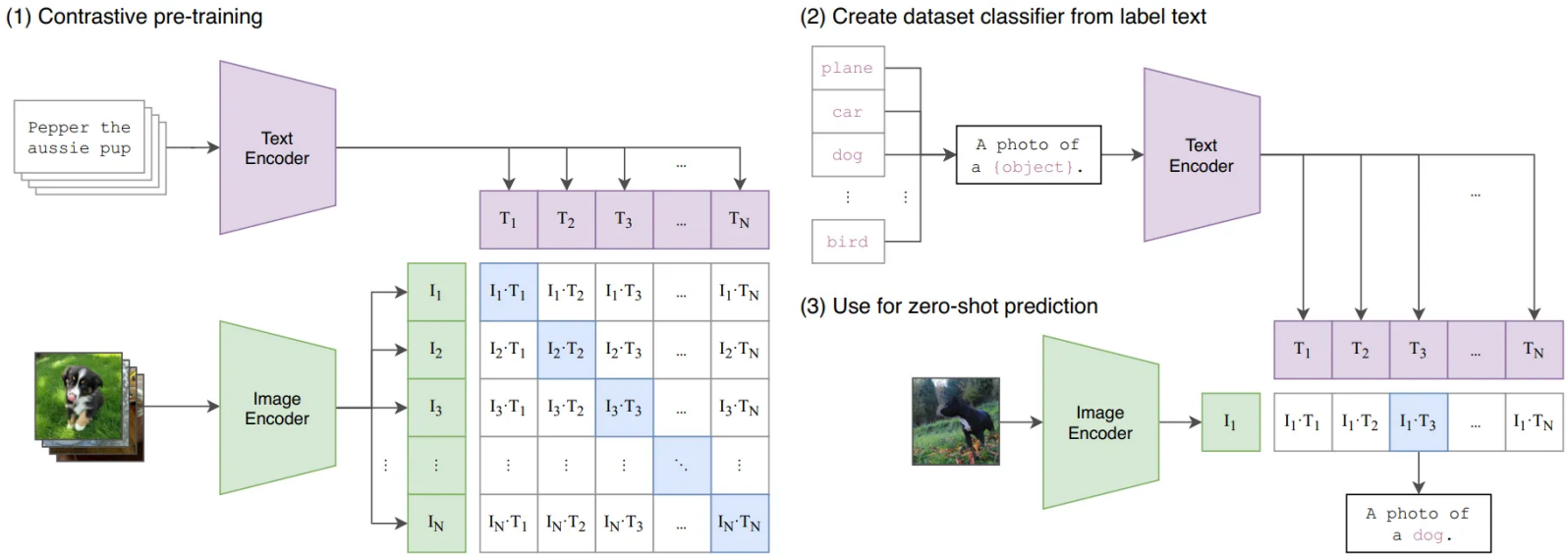
Vision Transformers (ViTs)

- Vision Transformers (ViTs) emerge as a competitive alternative to CNNs that are currently state-of-the-art in computer vision and widely used for various image recognition tasks
- Three major processing elements in transformer encoder: Layer norm, multi-head attention, multi-layer perception (MLP)



Contrastive Language-Image Pretraining (CLIP)

- CLIP: An open-source, multi-modal, zero-shot model
  - Image encoders: ViT or ResNet architectures
  - Text encoders: Transformers
  - These encoders are trained to maximize the similarity of a dataset of 400 million (image, text) pairs



Error vs. Date

Error vs. # of Parameters

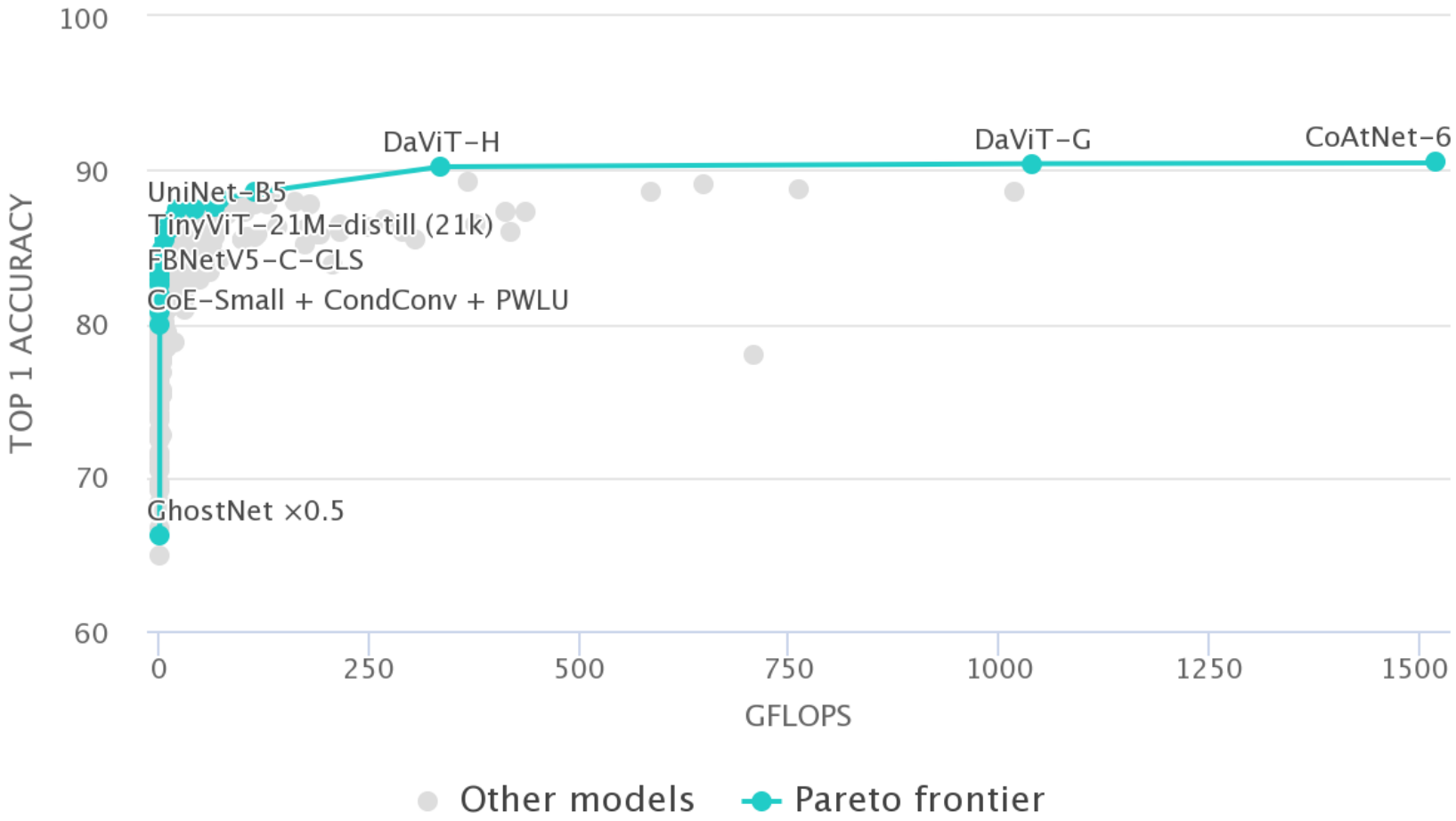


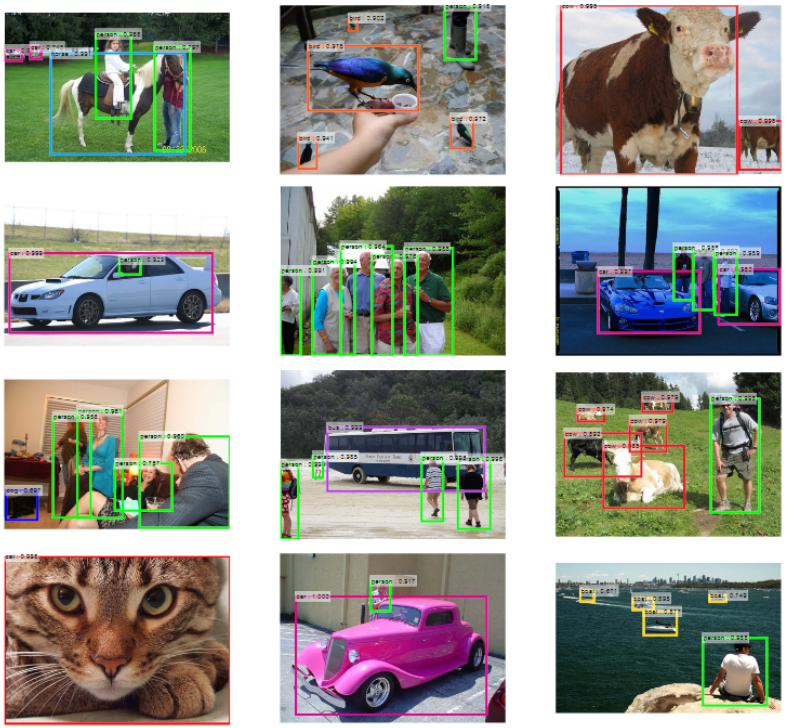
Image Classifiers in PyTorch

- PyTorch includes several pre-trained image classifiers
  - VGG16, InceptionV3, ResNet50, ResNeXt50, ViT, CLIP
  - Already trained on ImageNet
- Can use these networks to
  - Perform image classification (for 1000 classes only)
  - Extract image features for our own task
  - Transfer learning - modify a pre-trained network for our own task

Outline

Object Detection

- Goal: Identify and locate objects within an image or video
  - Not only involves recognizing the object categories within an image but also accurately determining their locations
  - Typically represented as bounding boxes



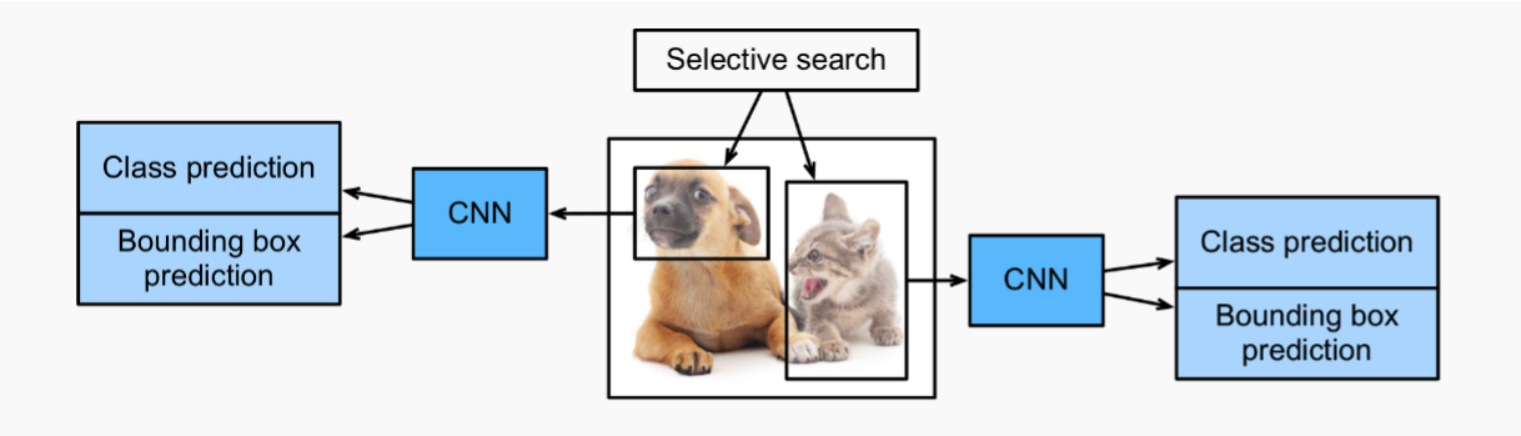
MS COCO

- Microsoft common objects in context (MS COCO)
  - Over a million images
  - Encompassing 80 different object categories

(a) Iconic object images (b) Iconic scene images (c) Non-iconic images

Region-based CNN (R-CNN)

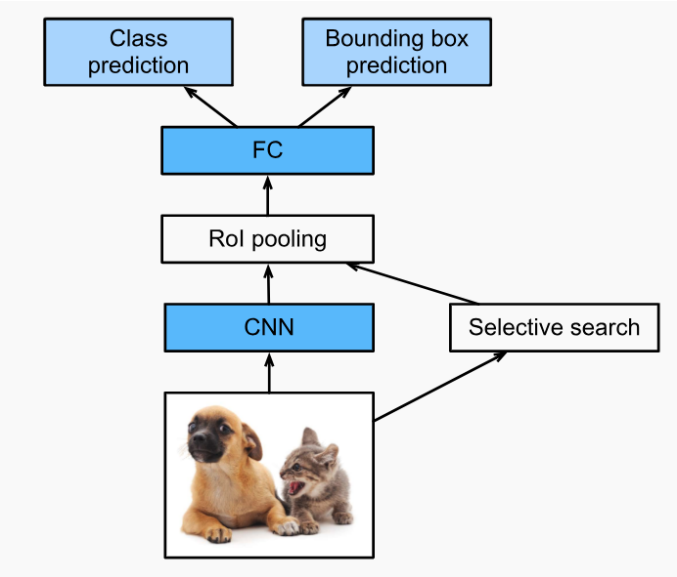
- Four steps: region proposal ($\sim$ 2k) $\rightarrow$ feature extraction $\rightarrow$ classification $\rightarrow$ bounding box regression



- Limitations:
  - Bad candidate region proposals: selective search algorithm
  - Time-consuming: $2,000$ forward propagations needed per image

Fast R-CNN

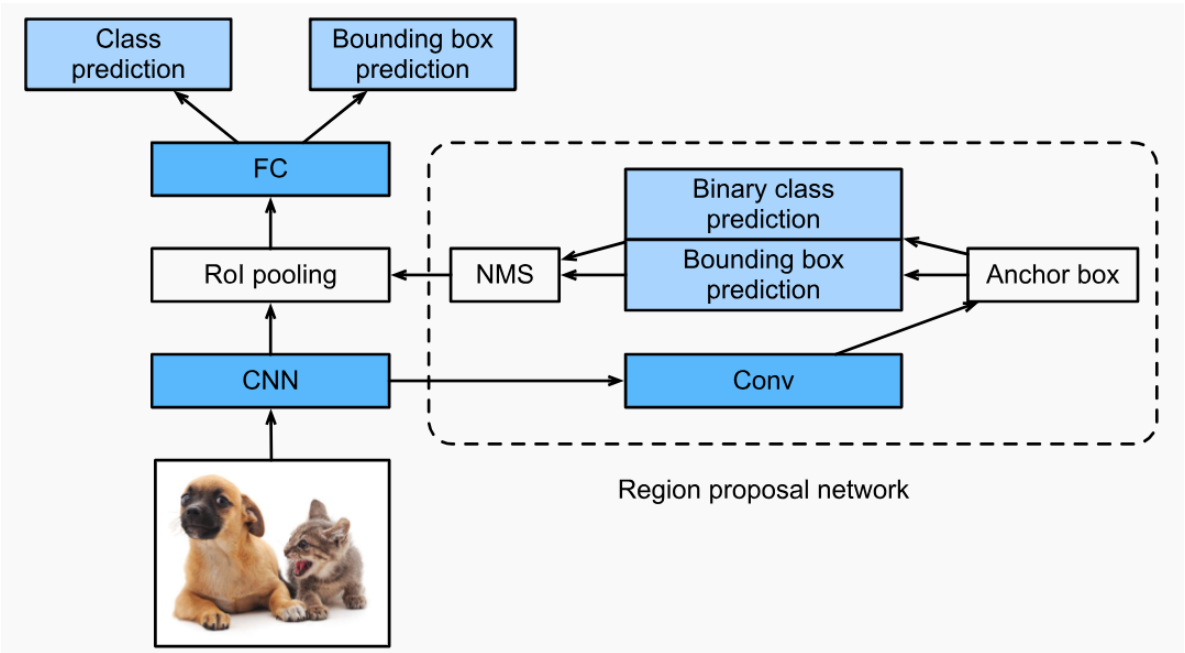
- Four steps: input and convolutional feature extraction → region proposal → region of interest (RoI) pooling → classification and bounding box regression



- Limitations:
  - Fixed RoI sizes: reduce accuracy for objects of different scales or aspect ratios
  - Non-learnable region proposal step

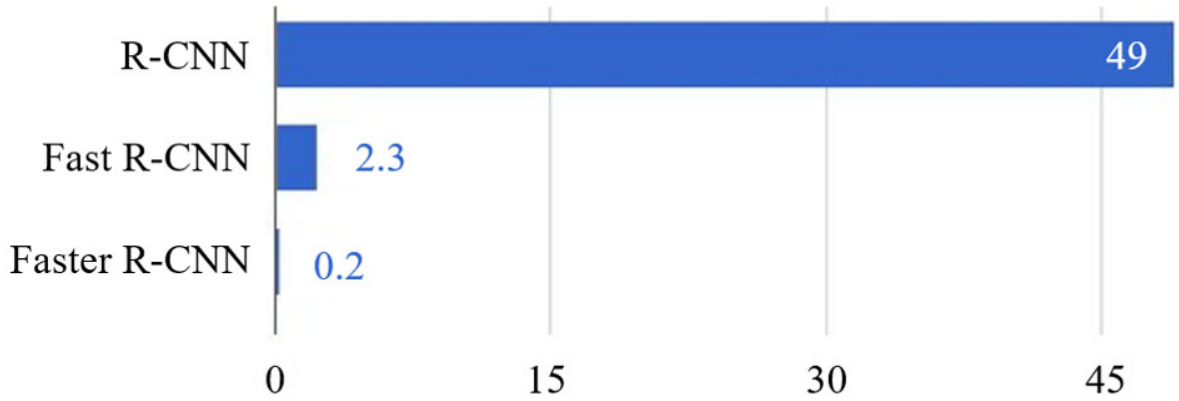
Faster R-CNN

- Four steps: input and convolutional feature extraction $\rightarrow$ region proposal network $\rightarrow$ RoI pooling $\rightarrow$ classification and bounding box regression



Comparison

- Improvement:
  - Fast R-CNN: extract feature with only one forward propagation by using RoI Pooling
  - Faster R-CNN: using learnable RPN instead of the traditional selective search method



Object Detection: Lots of Variables

- One-stage, two-stage, and Transformer series are three main frameworks in object detection:
  - One-stage focus on speed and may sacrifice some accuracy
  - Two-stage has high accuracy, but the speed is relatively slow
  - Transformer series is based on Vision Transformers
- Which framework to choose depends on the application and performance requirements

One-Stage

- YOLOv1
- YOLOv2
- ...
- YOLOv8

Two-Stage

- RCNN
- Fast RCNN
- ...
- Faster RCNN

Transformer Series

- DETR
- DN-DETR
- ...
- Focus-DETR

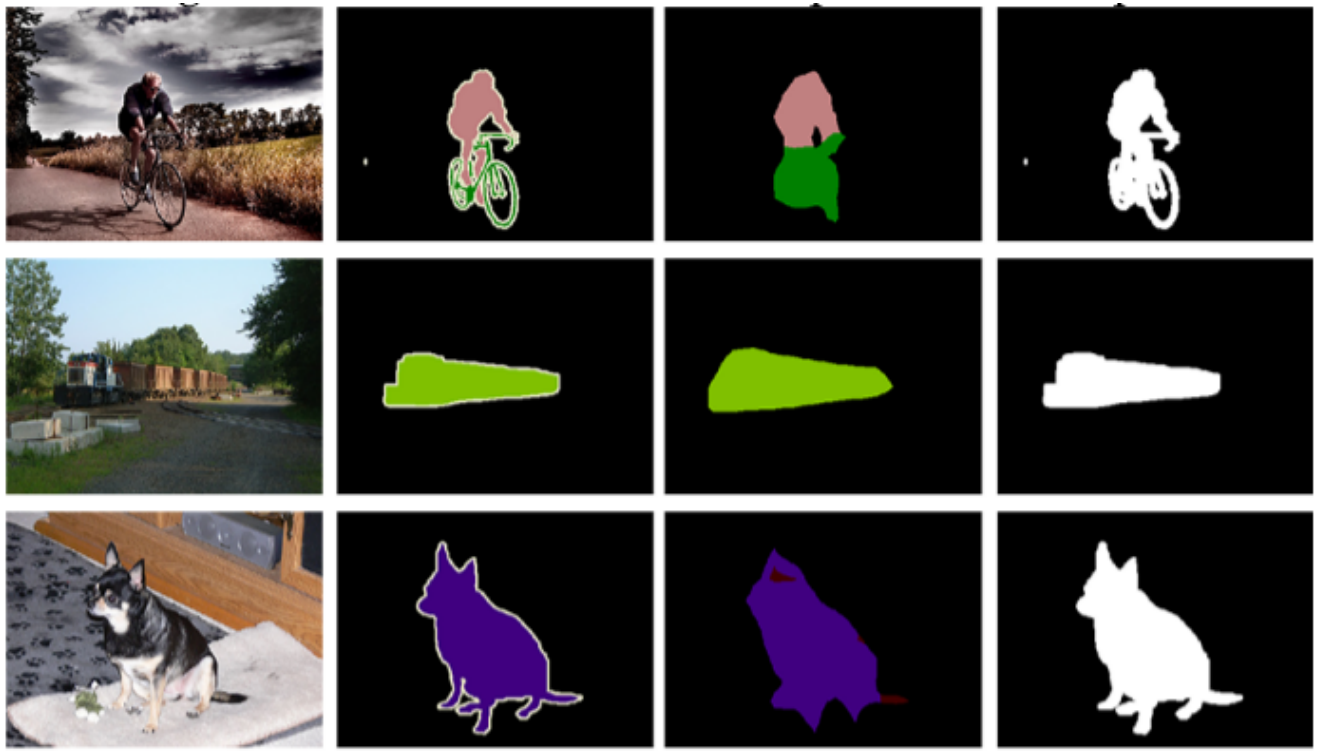
Outline

Semantic Segmentation

- Goal: Assign a semantic class label to each pixel in an image

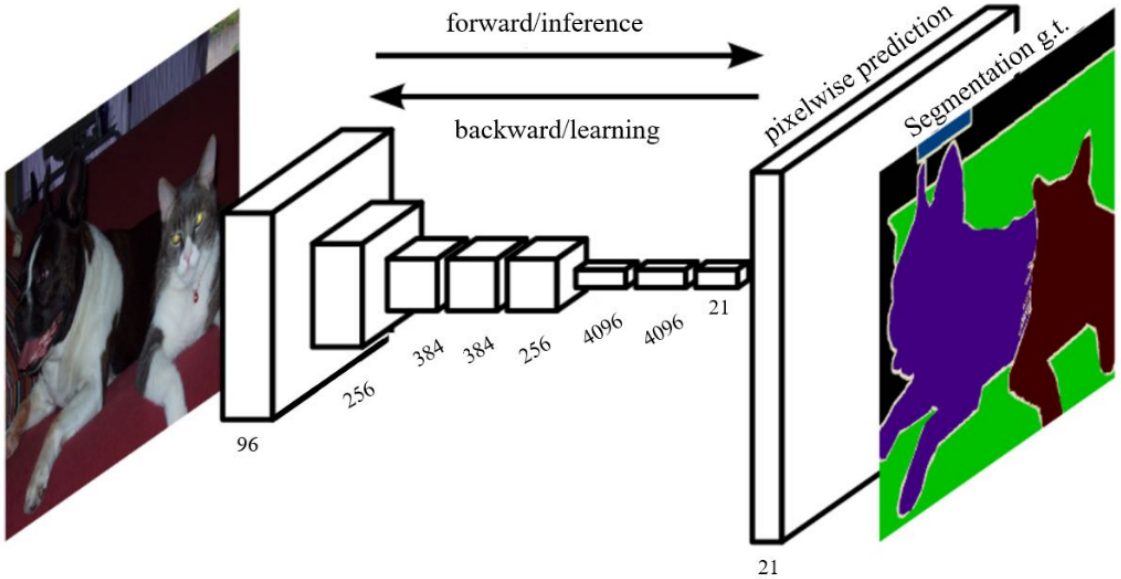
PASCAL VOC

- PASCAL visual object classes (PASCAL VOC)
  - 20 object classes
  - Images: 9.9k (VOC2007) & 23k (VOC2012)
  - Objects: 24.6k (VOC2007) & 54.9k (VOC2012)



Fully Convolutional Networks (FCNs)

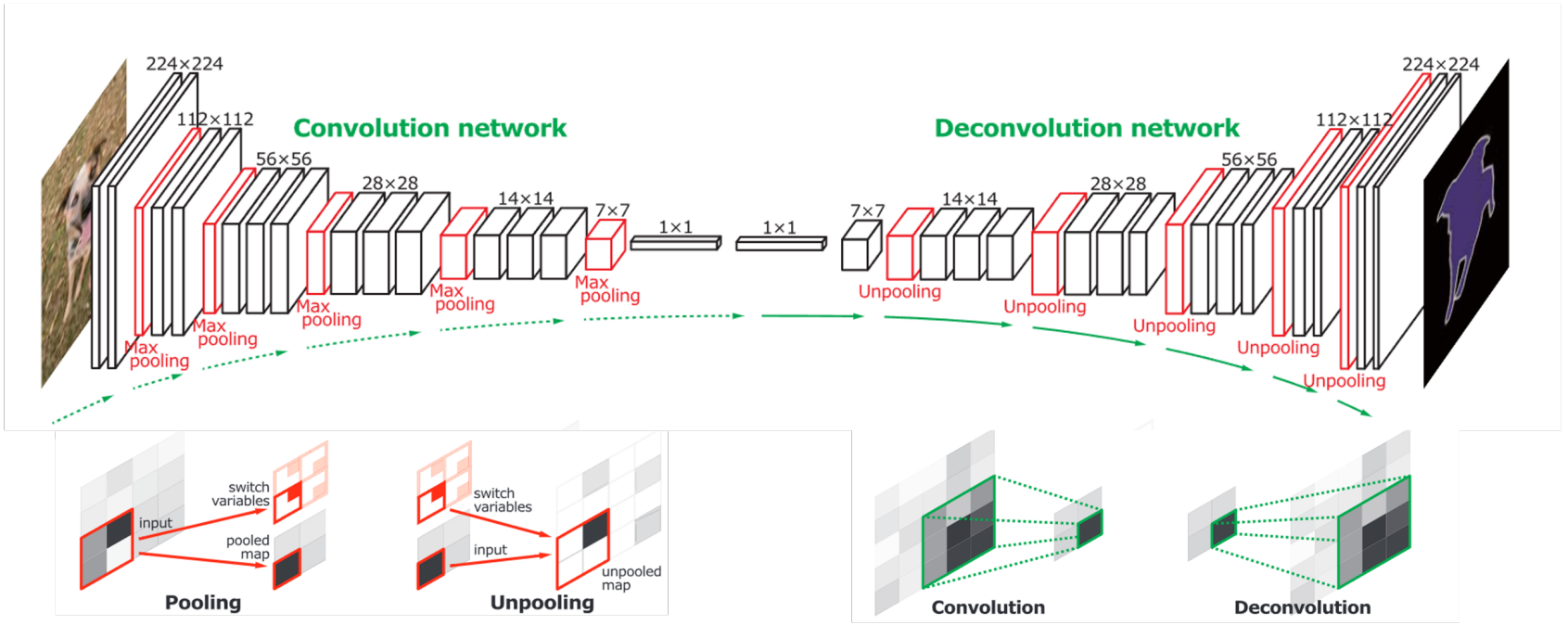
- Fully convolutional networks (FCN) is a framework for image semantic segmentation. The core idea:
  - A fully convolutional network without fully connected layers, capable of adapting to inputs of arbitrary sizes
  - A skip architecture that combines results from different depth layers while ensuring both robustness and precision



- Limitations: Directly upsample the compact features

DeConv

- DeConv proposes a deconvolution network with unpooling operation to predict the segmentation map
  - Convolution network: downsample the feature representations
  - Deconvolution network: upsample the compact feature maps and refine the dense predictions



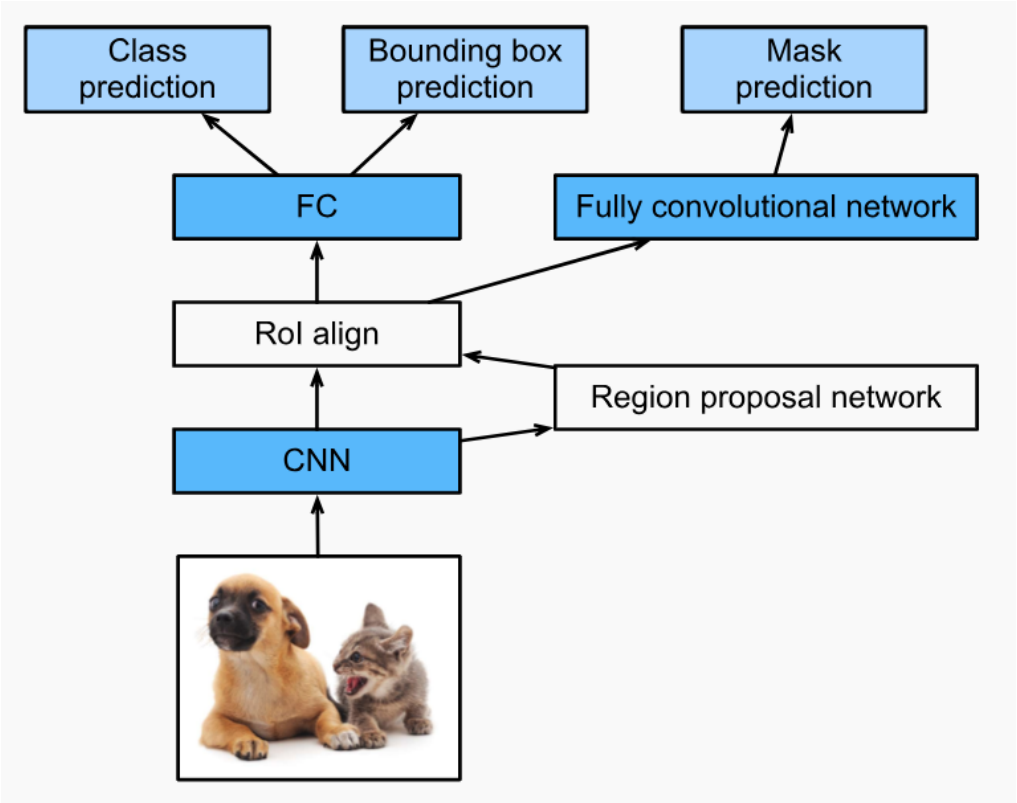
Outline

Instance Segmentation

- Goal: Detect objects in an image and label each pixel at the same time
- Compared with object detection, it outputs a mask instead of a bounding box
- Compared with semantic segmentation, it distinguishes between different instances in the same category

Mask R-CNN

- Four steps: input and convolutional feature extraction → region proposal network (RPN) → RoI align → classification and bounding box regression



Outline

Conventional Model Comparison Methodology

- Pre-select a number of images from the space of all possible natural images (*i.e.*, natural image manifold) to form the test set
- Collect the human label for each image in the test set to identify its ground-truth category
- Rank the competing classifiers according to their goodness of fit (*e.g.*, accuracy) on the test set

Limitations

- The test sets are small, fixed, and extensively reused
- More fundamentally, the underlying philosophy is to **prove** a classifier to be correct, which is impossible to achieve

MAximum Discrepancy (MAD) Competition

- MAximum Discrepancy (MAD) attempts to **falsify** two classifiers by maximizing their prediction discrepancy
- A classifier that is harder to be falsified in MAD is considered better

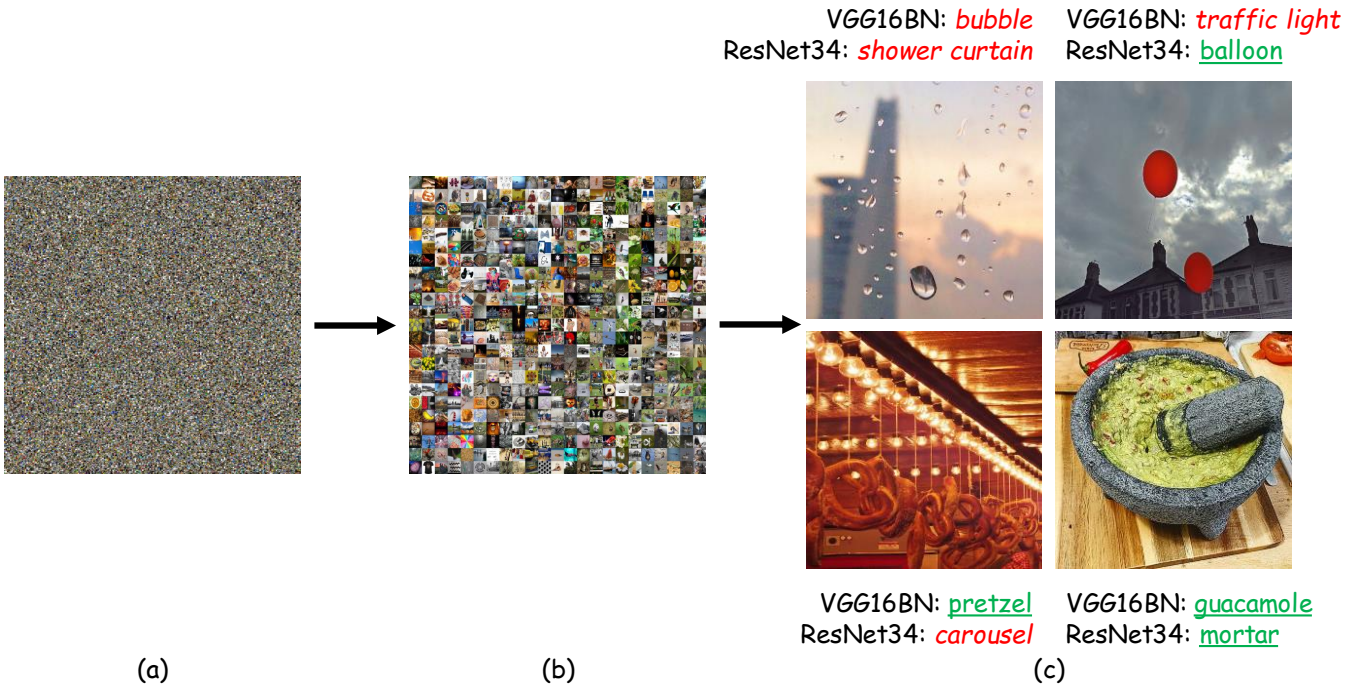


Figure: green underlined and *red italic* texts indicate correct and incorrect predictions

Quantify the Discrepancy

- We can not use the zero-one loss, why?

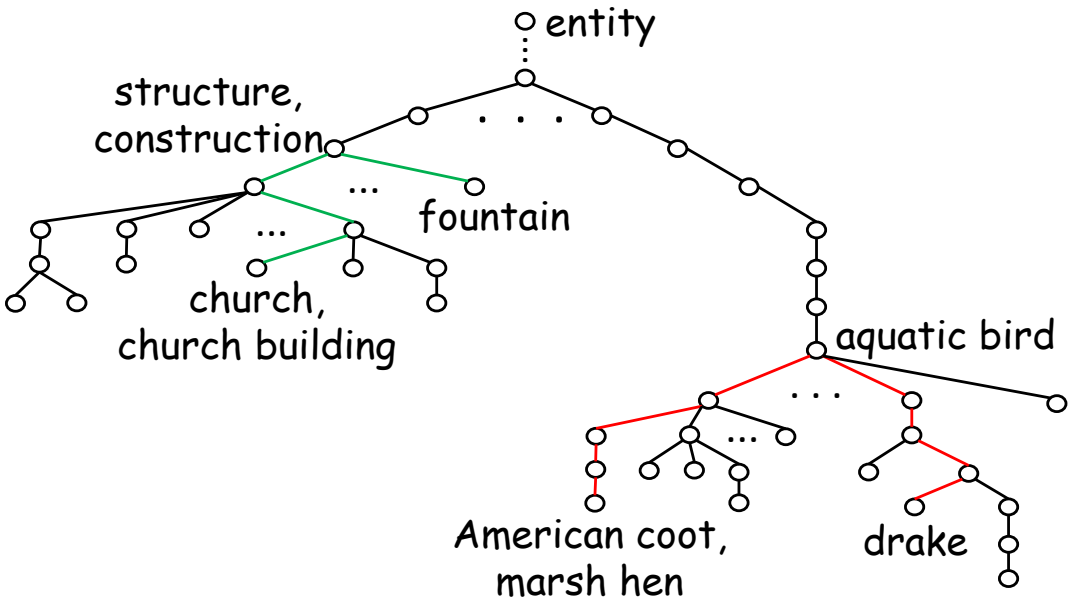


Figure: In the sub-tree of WordNet, we highlight the shortest paths from "fountain" to "church" and from "drake" to "American coot" in green and red, respectively. The semantic distance between the two aquatic birds is much shorter than that between the two constructions of completely different functionalities.