15-494/694: Cognitive Robotics

Dave Touretzky

Lecture xx:
ImageNet and Transfer Learning

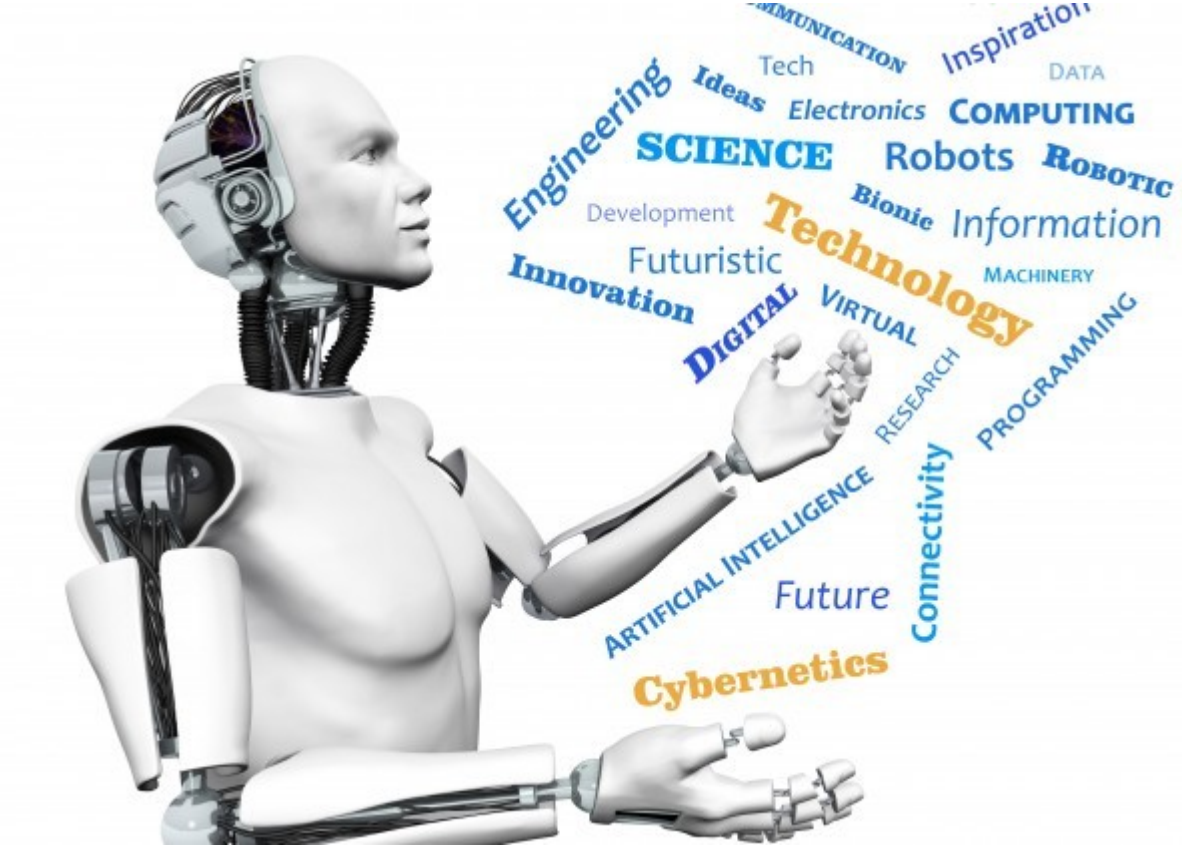


Image from http://www.futuristgerd.com/2015/09/10

Object Recognition Challenge

- Computer vision researchers use challenge events to measure progress in the state of the art.
- PASCAL VOC (Visual Object Classes) Challenge:
  - Ran from 2005 to 2012
  - 2005 version had 4 categories (bicycles, motorcycles, people, cars) and 1578 training images
  - 2012 version had 20 categories and 5717 training images

ImageNet

- Created by Fei-Fei Li at Stanford.
- See www.image-net.org
- 15 million labeled images, 22,000 categories
- ILSVRC: ImageNet Large Scale Visual Recognition Challenge: 2009-2017
  - 1000 categories, including 118 dog breeds
  - 1.2 million training images

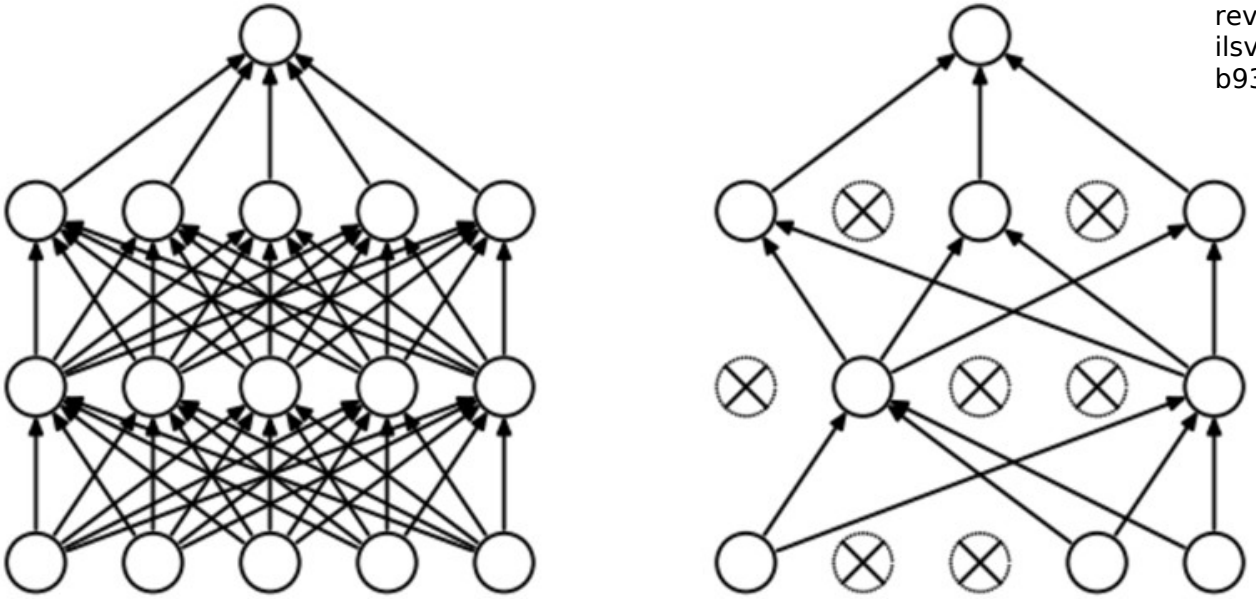
AlexNet

- The winners of the 2012 ILSVRC:
  - Alex Krizhevsky, Ilya Sutsker, and Geoffren Hinton
  - Deep convolutional neural net (DCNN) called AlexNet
  - Trained using two GPU boards
  - Introduced ReLU in place of tanh
  - Used “dropout” to reduce overfitting
  - Error rate of 15.3% was 10% better than the runner-up
  - Put deep neural nets on the map

Dropout in AlexNet

- For each training step, disable 50% of the neurons for both the forward and backward pass.
- Reduces overfitting.

Figure from https://medium.com/coinmonks/paper-review-of-alexnet-caffenet-winner-in-ilsvrc-2012-image-classification-b93598314160



(a) Standard Neural Net

(b) After applying dropout.

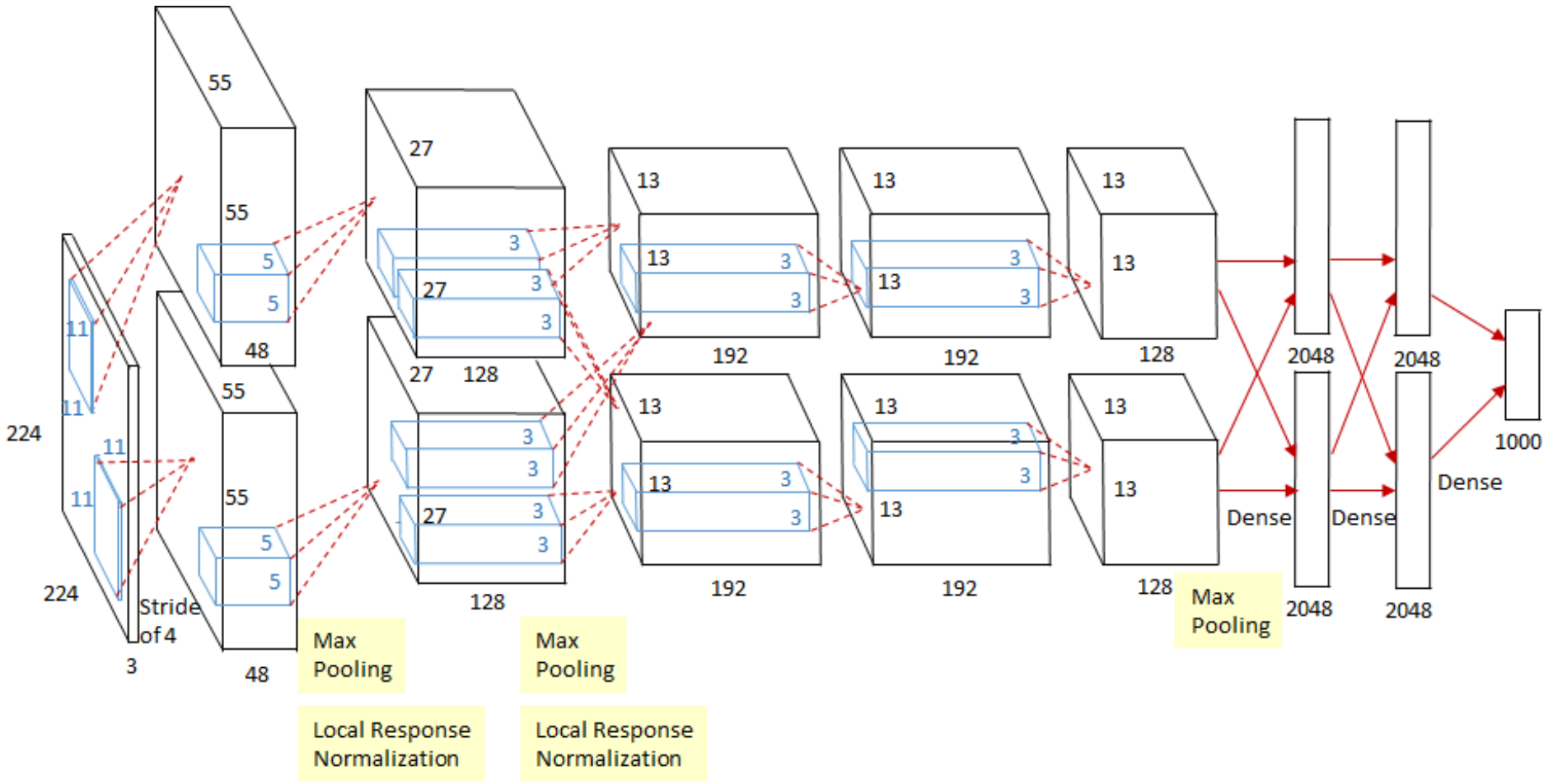
Data Augmentation in AlexNet

- Take random 224x224 crops of a 256x256 image, plus their horizontal reflections. Increases training set size by a factor of $32^2 \times 2 = 2048$.

- Add random factors to RGB values to simulate variations in lighting.

- These steps help the network generalize better.

AlexNet Architecture



All hidden layers were split in two and trained on different GPU boards due to GPU memory limitations.

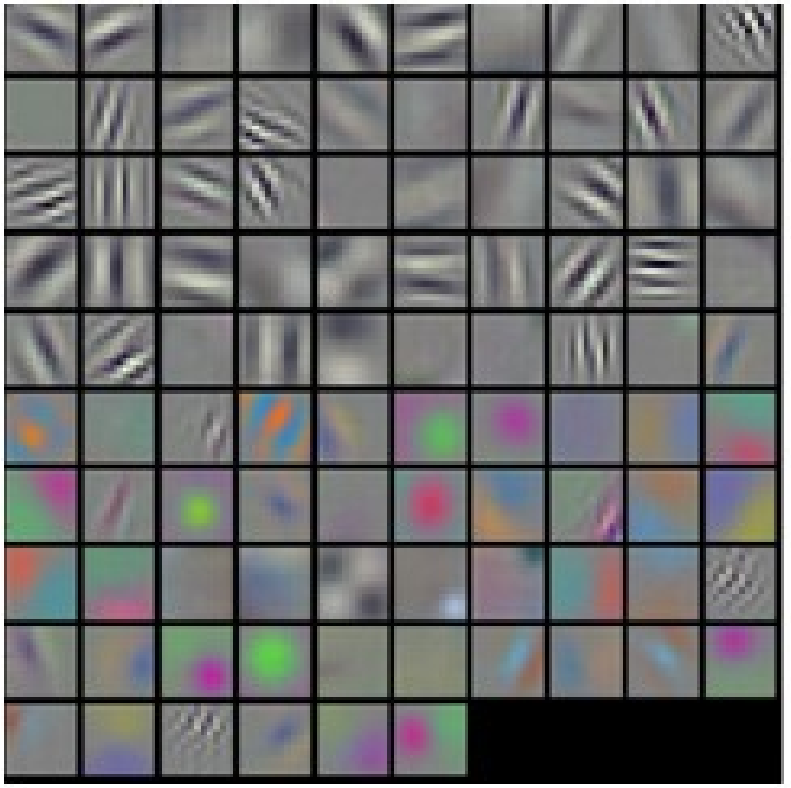
AlexNet Layer 1 Kernels

AlexNet's 96 11x11 layer 1 kernels.

First 48 trained on GPU 1 look for edges.

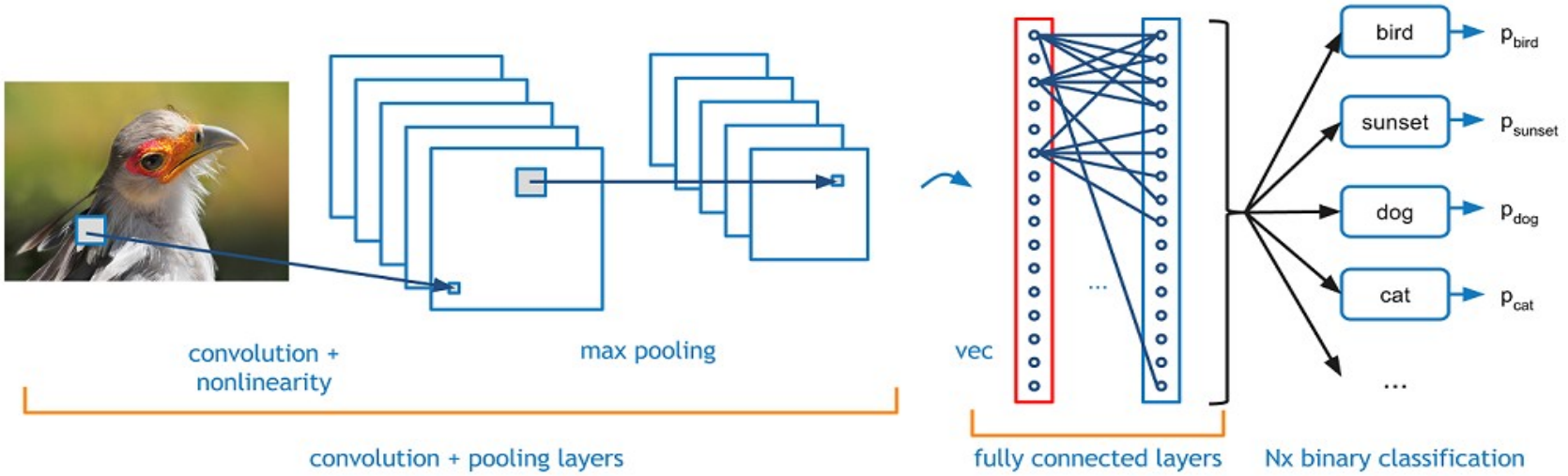
Second 48 trained on GPU 2 look for color.

This separation is a natural consequence of the normalization terms in the early layers.



Visualizations of filters

Generic Object Recognition CNN



https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

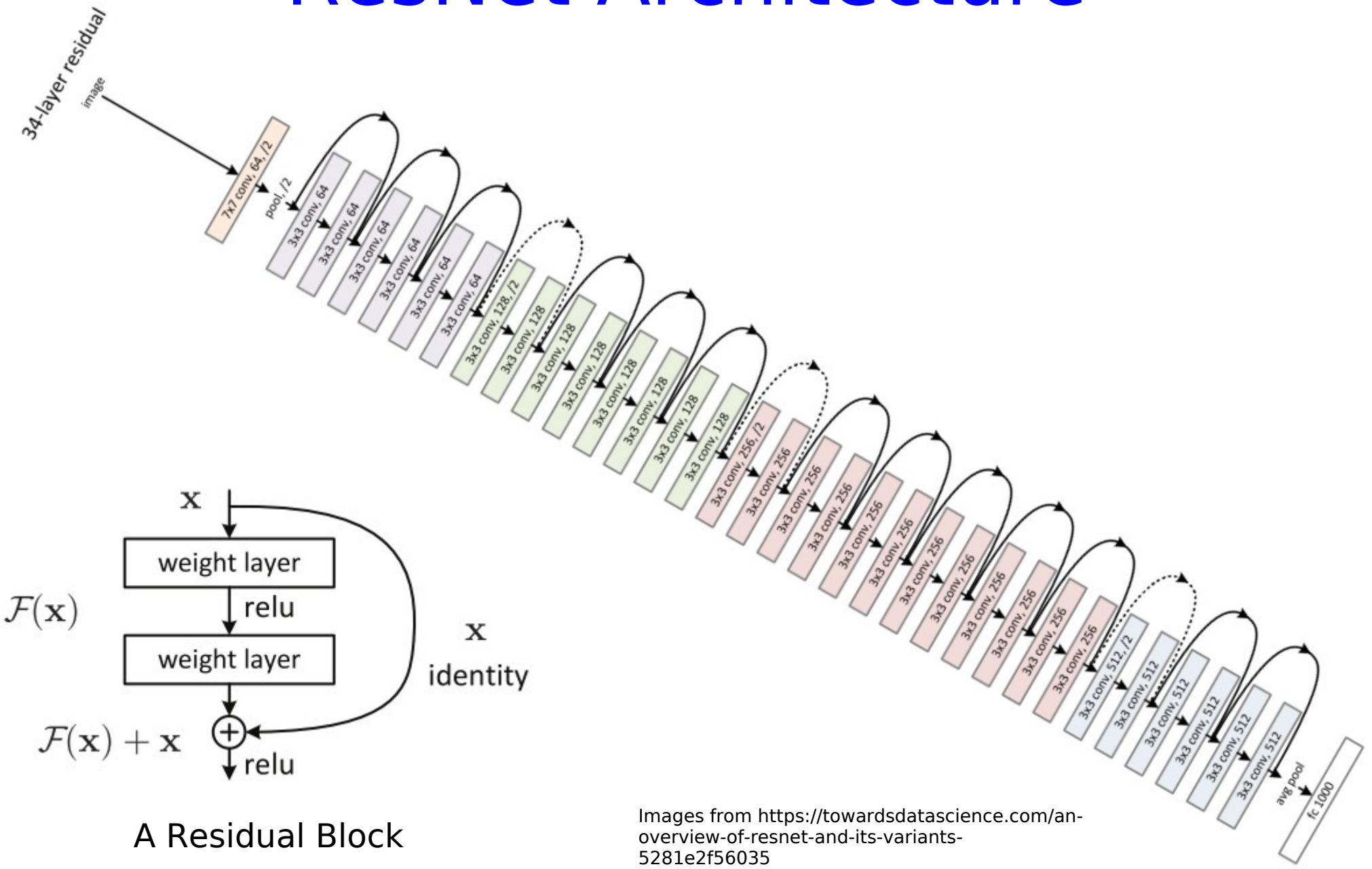
After AlexNet

- AlexNet had 8 layers: 5 convolutional and 3 fully connected.
- In 2015 Microsoft won the ILSVRC using a deep neural network with 100 layers.
- By the end of the ILSVRC in 2017, the best entrants were seeing accuracies of over 95% (error rate < 5%).

Residual Blocks

- Residual blocks were introduced in ResNet:
  - For really deep networks, it's hard for the error signal to propagate backwards through many layers.
  - Solution: add shortcut connections, e.g., from layer i to layer i+2, so that error can back-propagate more quickly.
  - A residual block contains hidden layers with a shortcut connection.

ResNet Architecture



A Residual Block

Images from https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035

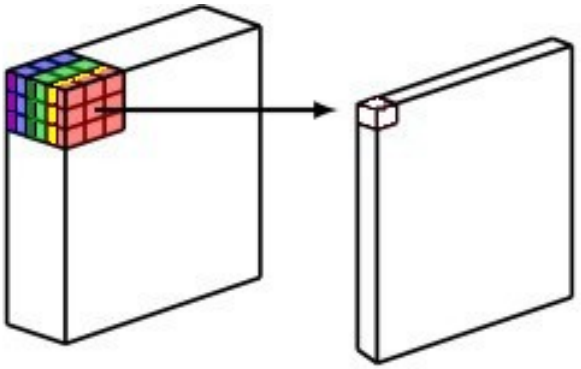
Mobile Implementations

- People want to implement computer vision on mobile phones. Networks must be reduced in size.

- Various architectures explore ways to reduce the size of the network and the number of multiply-add operations.
  - Separable convolutions
  - Bottlenecks

- Examples: MobileNet, SqueezeNet

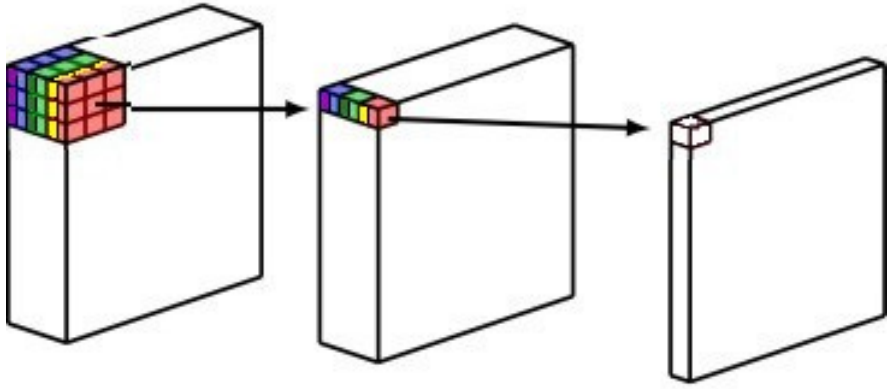
Separable Convolutions



(a) Conventional Convolutional Neural Network

3x3 kernel over 6 channels

3x3x6 = 54 weights



(b) Depthwise Separable Convolutional Neural Network

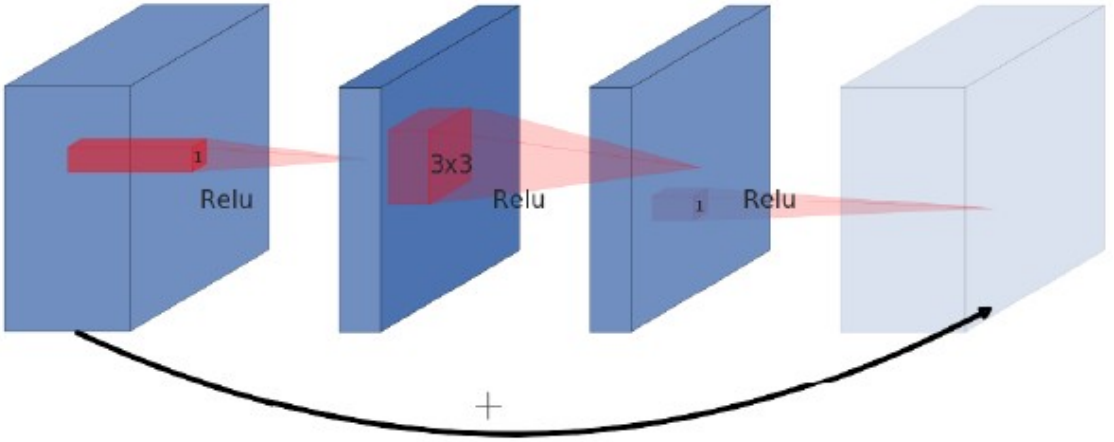
One 3x3 kernel applied to all 6 channels (depthwise convolution)

Linear weighted combination of the results (pointwise convolution)

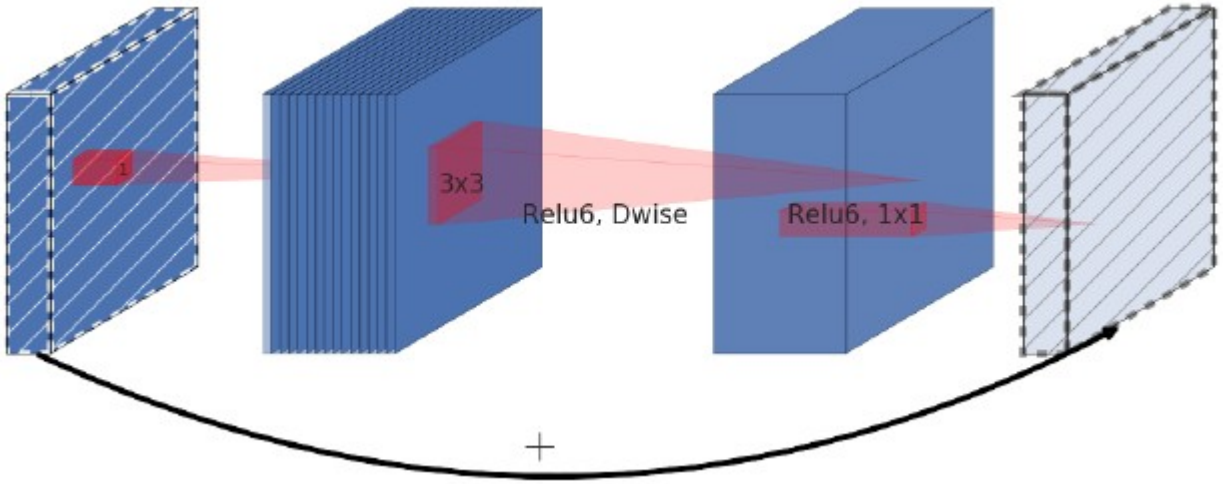
3x3 + 6 = 15 weights

Image modified from
https://www.sciencedirect.com/science/article/pii/S0168169918318696

Bottlenecks with Residuals



MobileNet: residual bottleneck



MobileNetV2: inverted residual bottleneck

Images from https://towardsdatascience.com/mobilenetv2-inverted-residuals-and-linear-bottlenecks-8a4362f4ffd5

PyTorch Vision Models

- PyTorch contains several pre-trained object recognition models, including AlexNet, ResNet, Inception, VGG, and MobileNetV2.
- Look in torchvision.models for a list.
- Models are trained on the ImageNet dataset.

MobileNetV2 on Cozmo

- See the course's demos folder.
- Uses pre-trained MobileNetV2 model from torchvision.models.
- Feeds a 224x224 Cozmo camera image into the network and reports the top 5 labels.

Transfer Learning

- How can we quickly train a visual classifier for a new object class?
- Use the last hidden layer of a pre-trained ImageNet classifier as a feature vector.
- Train a classifier on the new categories using just 1-2 layers of trainable weights, or just use k-nearest neighbor.
- This is how Teachable Machine works.

Teachable Machine

https://teachablemacine.withgoogle.com