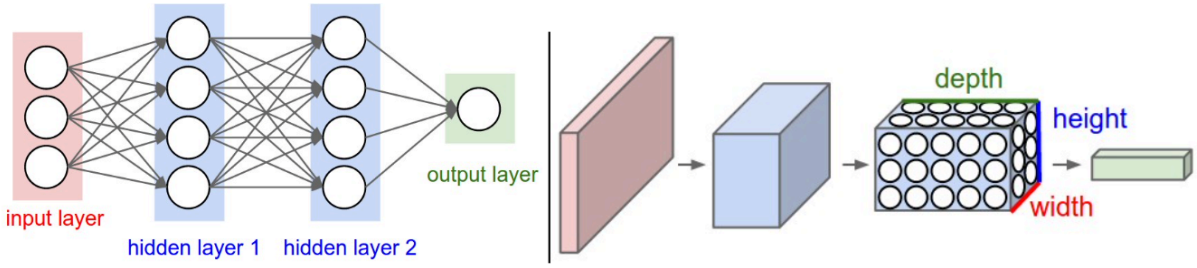
CNNs cont'd

Michael Wollowski

This presentation is a select summary of the following document:
https://cs231n.github.io/convolutional-networks/

Conceptual Differences between FFN and CNN



- On left a FFN
- On right, CNN which transforms a 3D input into a 3D output.
- The depth/height/width block gives a sense of the type of processing

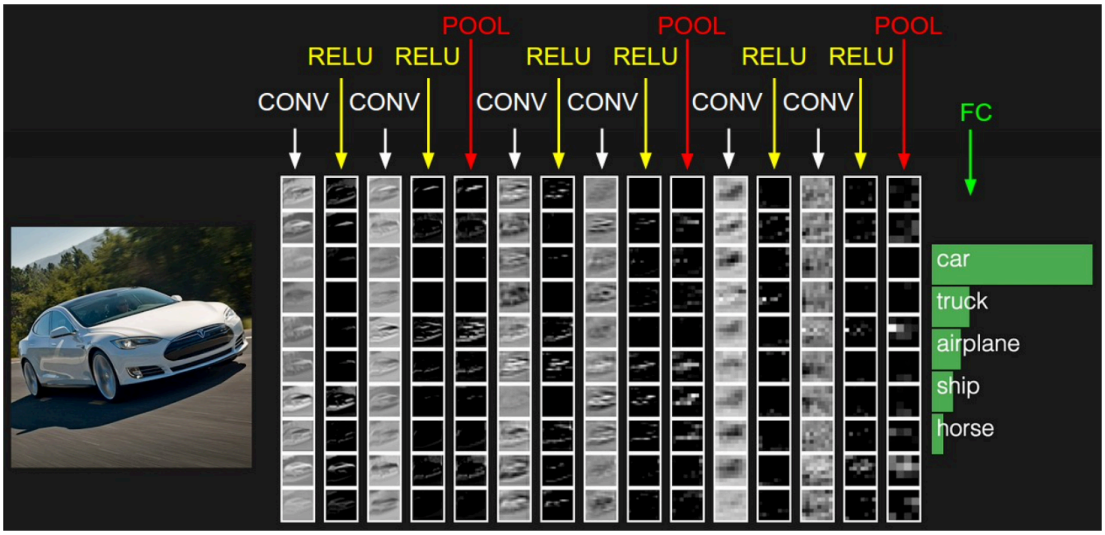
ConvNet

- ConvNet is a simple CNN for CIFAR-10 classification.
- CIFAR-10 contains 60,000 32x32 color images in 10 different classes.
- The classes represent airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks.
- There are 6,000 images of each class

ConvNet

- Input: [32x32x3] raw pixels in RGB
- 12 Filters: 3x3 filter size, 1-bit stride, 1-bit padding
- After convolution: [32x32x12]
- RELU layer
- After RELU, still [32x32x12]
- POOLing: 2x2 receptive field, 2-bit stride
- After pooling: [16x16x12].
- Fully-connected layer: computes the class scores
- [1x1x10]

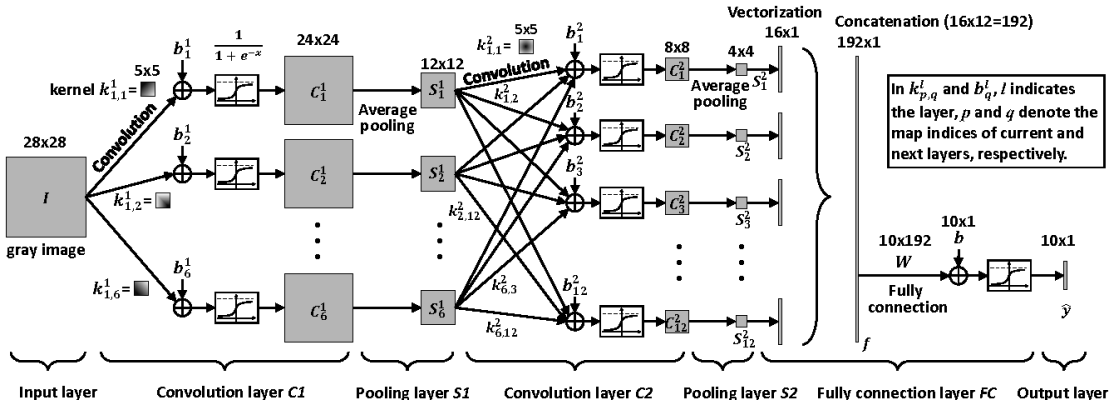
Sample Activations of a ConvNext



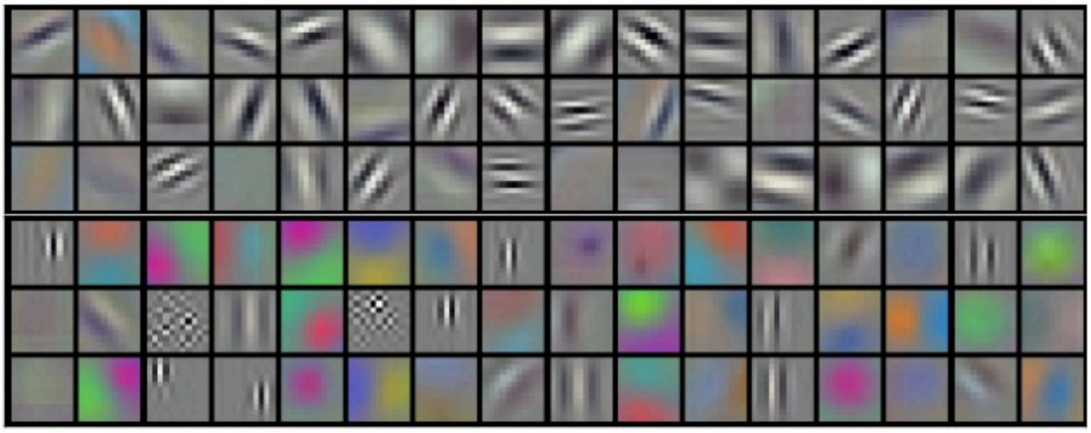
Specifying a CNN

- Hyper-parameters that an engineer needs to specify:
  - Number of Convolutional layers.
  - Depth, i.e. the number of filters. This may be different for different convolutional layers
  - Filter size, common values are 3x3 and 5x5
  - Stride of the filter application, typically it is 1 bit
  - Amount of padding, zero or one or two.
  - Number and placement of pooling layers
  - Size of pooling window, typically this is 2x2
  - Stride for pooling, typically, this is 2 bits
  - Pooling function: typically max

Location of learning: Weights and filter values



Filters learned by Krizhevsky et al.



Pattern Recognition

- What did we learn?

X1	X2	*h1*	*h2*	*y1*
0	0	0	0	0
0	1	1	0	1
1	0	1	0	1
1	1	2	1	0

Hierarchical Nature of Pattern Recognition