Convolutional Neural Nets

Oliver Schulte

Simon Fraser University

Introduction to Deep Learning

Further Notes with Demos

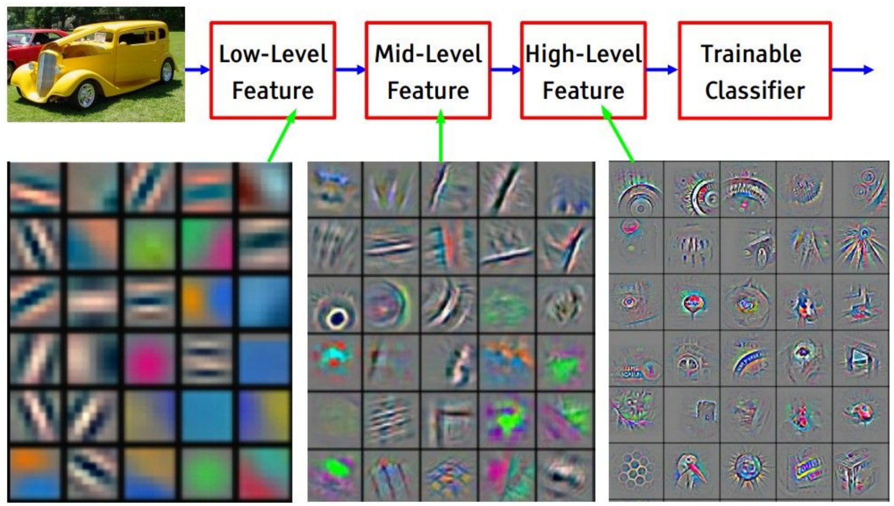
Overview

- CNNs are appropriate for data with a grid structure
  - In 1D: a sequence
- In 2D: a grid (e.g. image with pixels)
- Two key ideas:
  1. Slide fixed-size window over image/sequence/sentence = feature map
  2. 1 hidden node represents window activation at 1 position ➔can repeat sliding window idea to obtain higher-level features
- Assumes maximum-length input

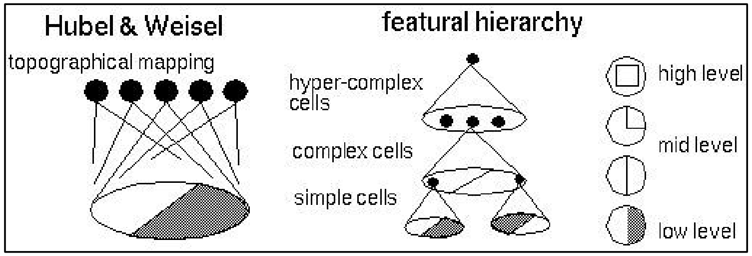
Hierarchical Features

Preview

[From recent Yann LeCun slides]



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]



Filters

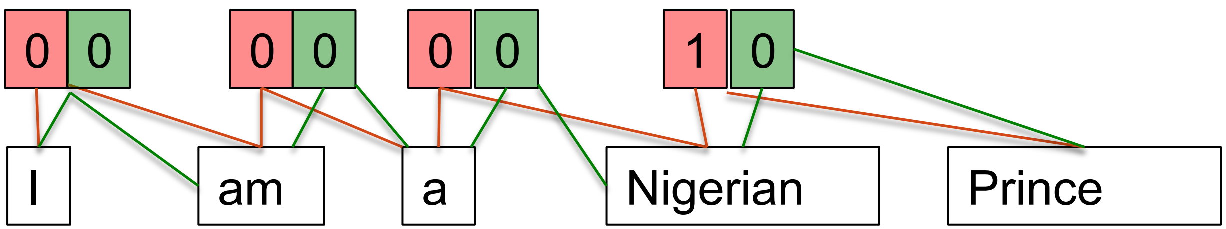
1D Example

- Want to classify a sentence as spam-like or not.
- Assume all sentences have at most 5 words e.g.
  - "I am a Nigerian Prince"
  - "You can get rich quick"

I	am	a	Nigerian	Prince

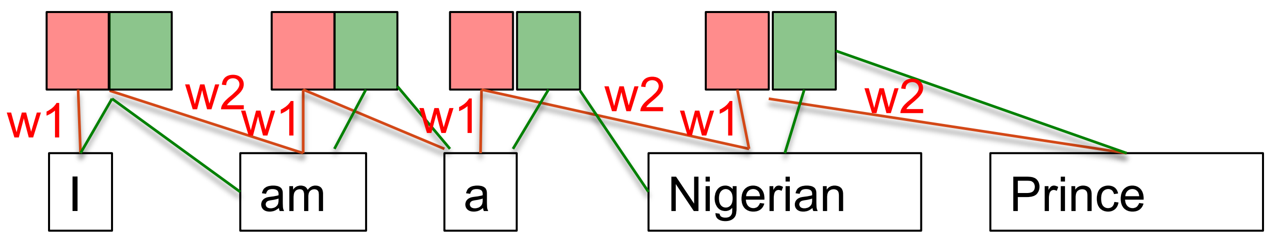
Window Size 2

- Boolean Feature/Filter 1: "Nigerian Prince"
- Boolean Feature/Filter 2: "get rich"
- 2 Filters x 4 window positions = 8 Features



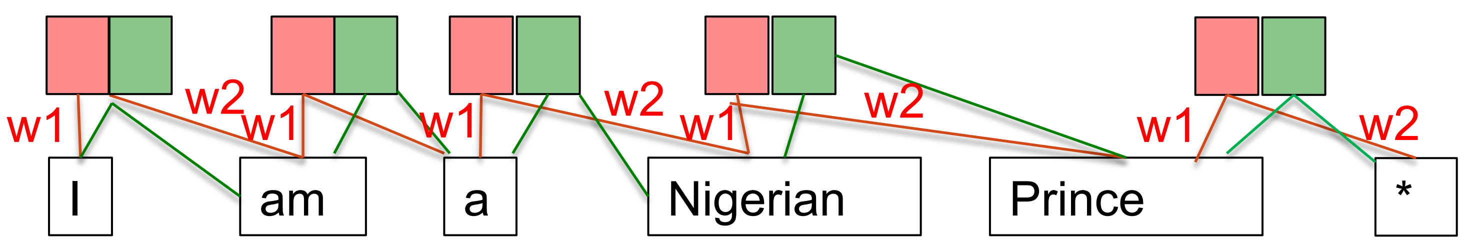
Trainable Filters

- As with basis functions, train weights to learn filters rather than hardcode them
- Since all red nodes compute the same feature, they use the same weights
  - an example of *parameter tying*
- Similarly we have 2 more weights w3 and w4 for the green filter (not shown)



Padding

- How can we apply a binary filter to the last word?
- Answer 1: You can't (see above).
- Answer 2: We can **pad** the sentence with a * or 0 input



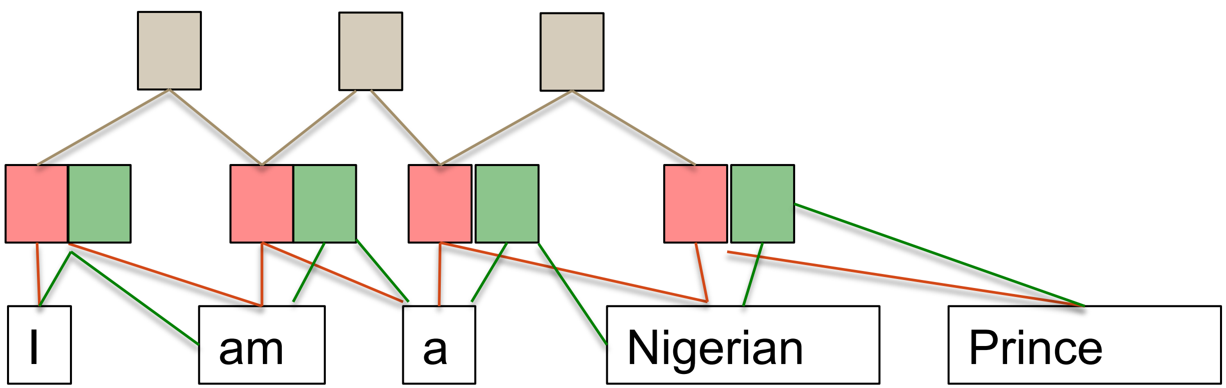
2D Version

- Think images with pixels
  - More reasonable to assume fixed maximum size
- Filter = small window size
- Slide over different positions
- May have to add 0s at boundary point ("zero-padding")
- cnn2d-example.pdf

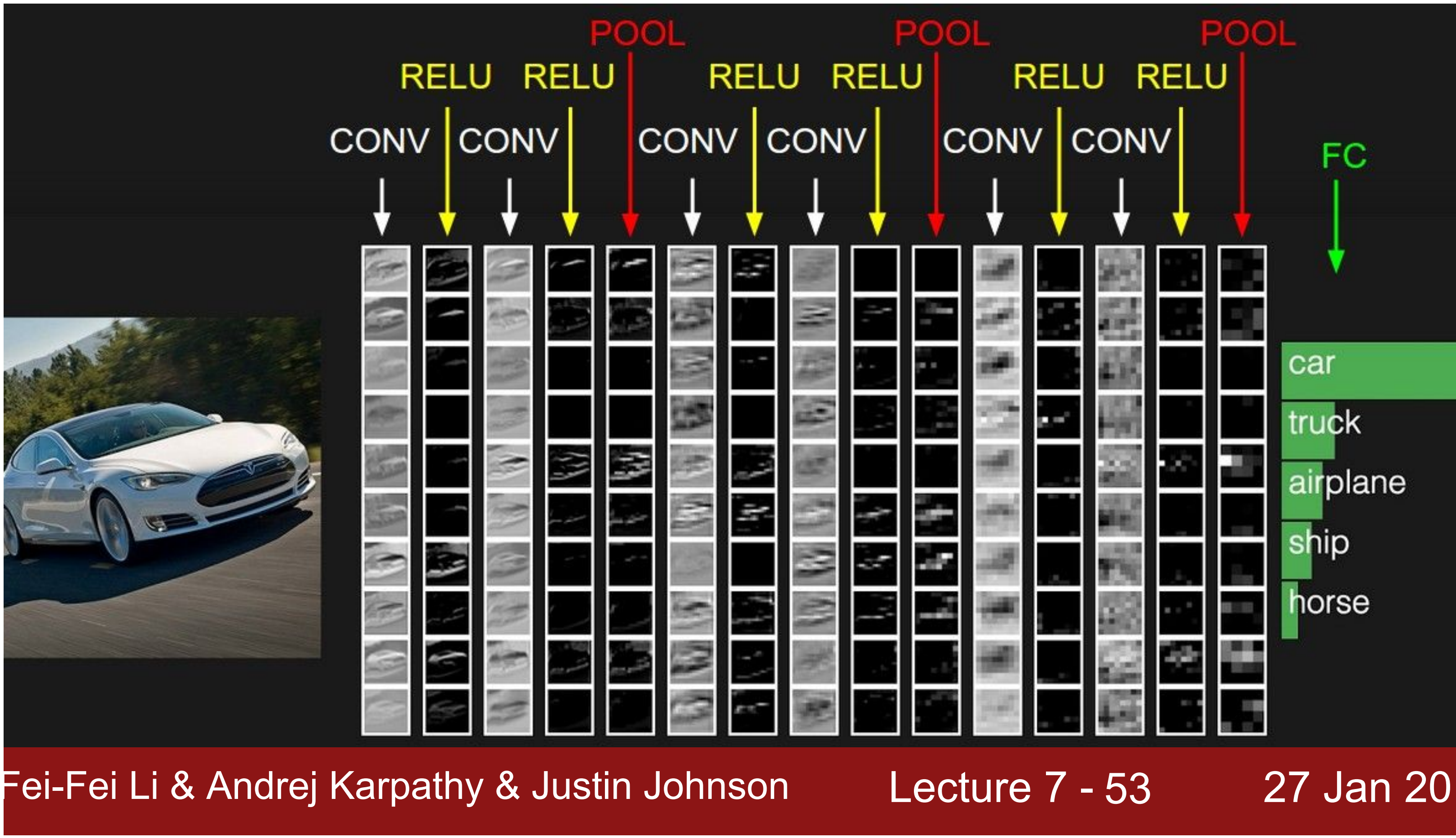
Deep CNN

Hierarchical Filters

- Neat Insight: Can use the same sliding window idea on the features in the first hidden layer.
- And the second, the third, etc
- Each convolutional layer generates higher-level features that cover a larger part of the input



Example Architecture



Convolutional Layer

- Stride: how far to move the filter horizontally/vertically.
  - Common values: 1,2
- What if filter goes over the edge?
  - Zero-padding: add "imaginary" 0 pixels (see Stanford)
  - Textbook/Tensorflow:
    - Same = zero-padding
    - Valid = stop when boundary is reached
  - Biases: can add a fixed constant to each filter value

Convolution

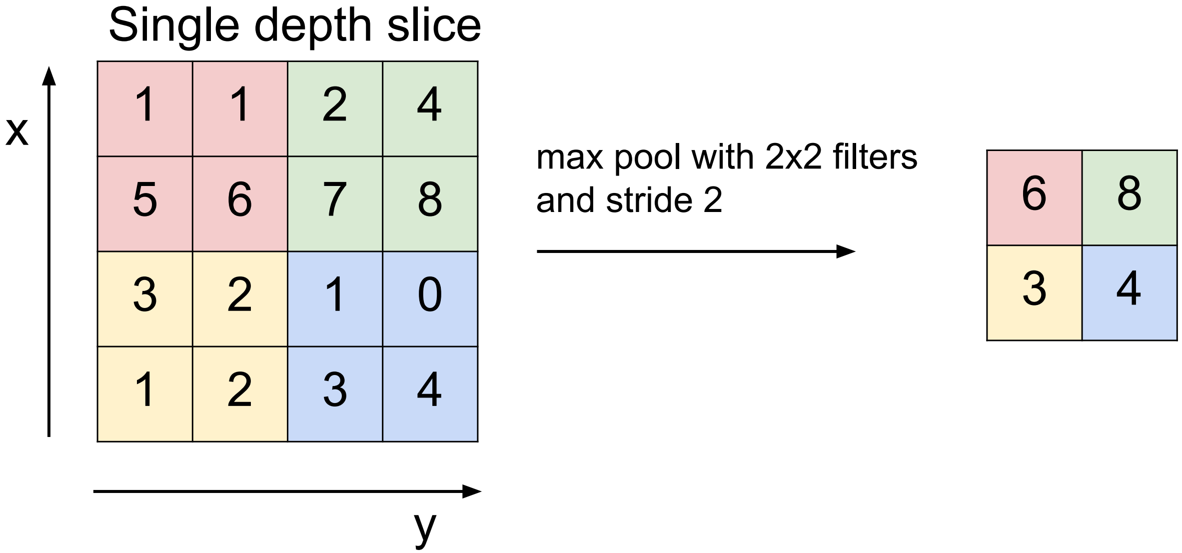
- Convolution of two vectors = sum element-wise products. E.g.
  - (1,2,3) * (4,5,6) = 4+10+18=32
- Convolution of two matrices A*B = sum element-wise products.
- Convolution computation:

1. Fix grid patch *p* and filter *f* of the same shape.
2. Output(p,f) = 0
3. For each channel c = 1,…,C
   Output(p,f) += grid(p,c) * f
   End for
4. Output(p,f) += f.bias

- Produces one number for each patch-filter combination
- CNN Demo and Homework Exercise. Visualization.

Alternating Layers

- Typical alternative types of layers:
  - convolutional (sliding window)
  - standard feed-forward to combine extracted features (relu)
  - pooling: extract fixed feature from window, e.g. max



CNN and Adversarial Examples

- "Deep Neural Nets are Easily Fooled"
  - Video
  - Paper

Conclusion

- Convolutional Neural Networks are widely used in computer vision
- Assume maximum input size
- Given fixed maximum, can set up neural network where hidden node computes learned feature of small fixed window

➢ The neural net encodes knowledge of 2D topology

- Apply another feature map to hidden layer ➔ hierarchical feature learning