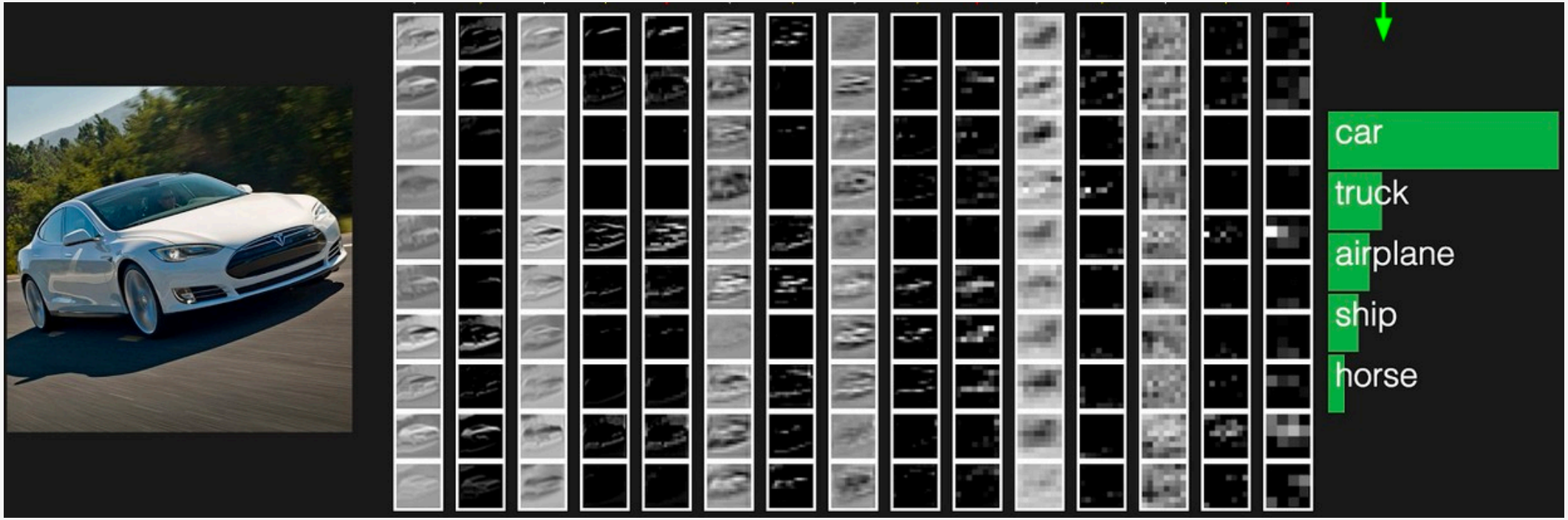
Lecture 19:
Convolutional Neural Networks

CNNs and Object Detection

- Convolutional neural networks because the go-to for object detection in 2010s (LeCun/Hinton)
- Extremely good at classifying data based on structured grid (images, text, signals)

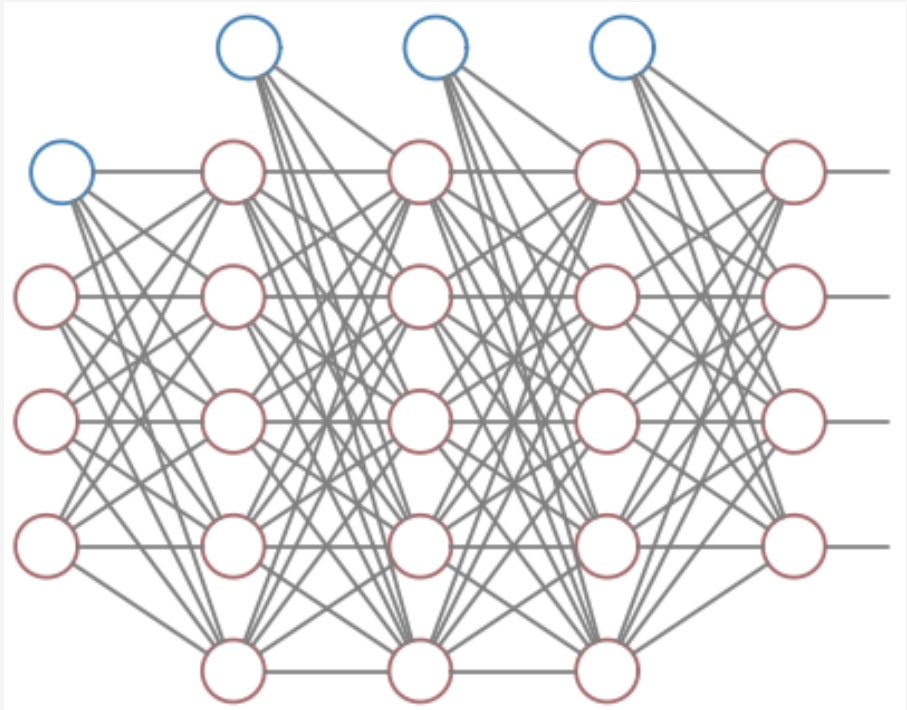


Convolutional Neural Network

- CNNs are still neural networks
- Organized in clever ways for structured data

Input Neurons

Pixels



Truck

Car

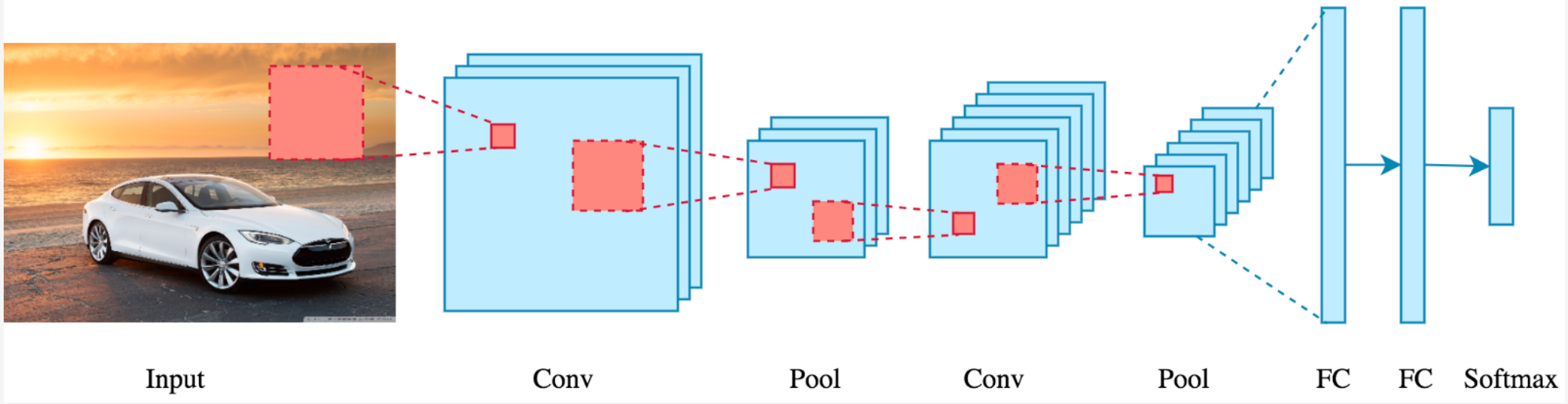
Motorcycle

Pedestrian

Output Neurons

Convolutional Neural Network

- CNNs are still neural networks
- Organized in clever ways for structured data
- CNNs automatically extract features from image and then use FFNN to classify image



Convolutional Filters

- Basic building block of a CNN is the **Convolution Layer**
- A convolution is a mathematical operation for merging two sets of information

Binary Image

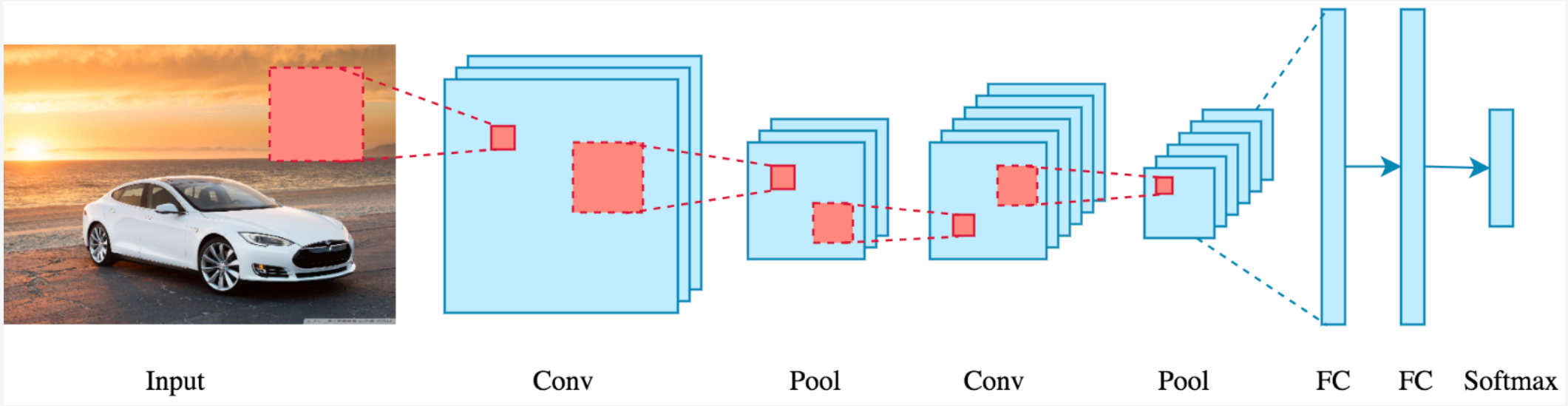
1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

1	0	1
0	1	0
1	0	1

convolutional filter

Convolutional Filters

- Basic building block of a CNN is the **Convolution Layer**
- A convolution is a mathematical operation for merging two sets of information



Convolutional Filters

1	0	1
0	1	0
1	0	1

- Basic building block of a CNN is the **Convolution Layer**
- Think of applying filter as a standard dot product

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

Convolutional Filters

1	0	1
0	1	0
1	0	1

- Basic building block of a CNN is the **Convolution Layer**
- Think of applying filter as a standard dot product

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

4		

Convolutional Filters

1	0	1
0	1	0
1	0	1

- Basic building block of a CNN is the **Convolution Layer**
- Think of applying filter as a standard dot product

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

4		

Convolutional Filters

1	0	1
0	1	0
1	0	1

- Basic building block of a CNN is the **Convolution Layer**
- Think of applying filter as a standard dot product

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

4		

Convolutional Filters

1	0	1
0	1	0
1	0	1

- What other kinds of filters might be useful for extracting features?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Convolutional Filters

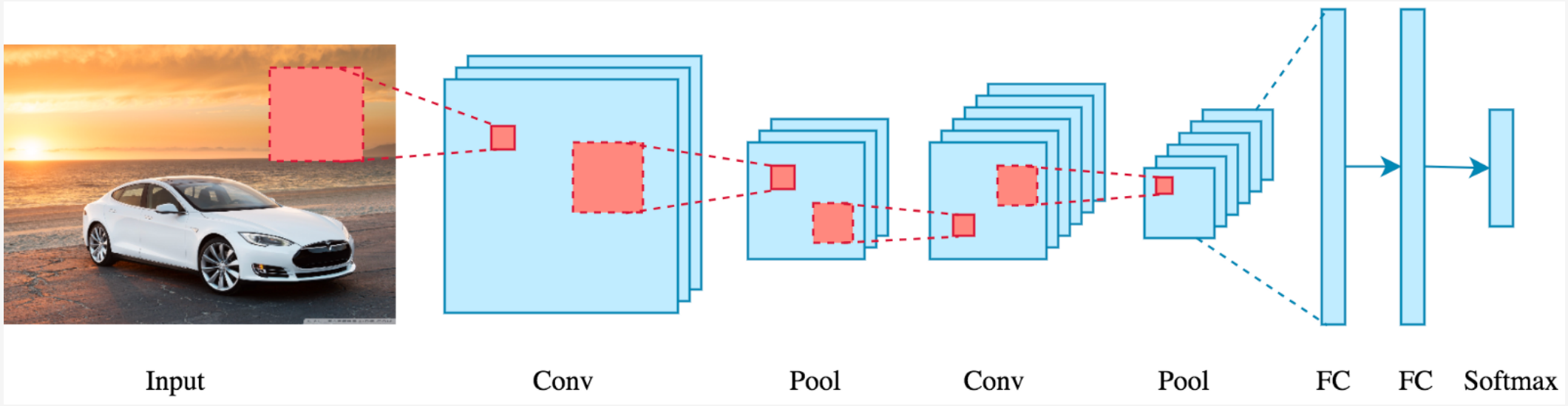
1	0	1
0	1	0
1	0	1

- What other kinds of filters might be useful for extracting features?
- **Idea!** – Use standard NN tricks to **LEARN** the best form of these filters

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

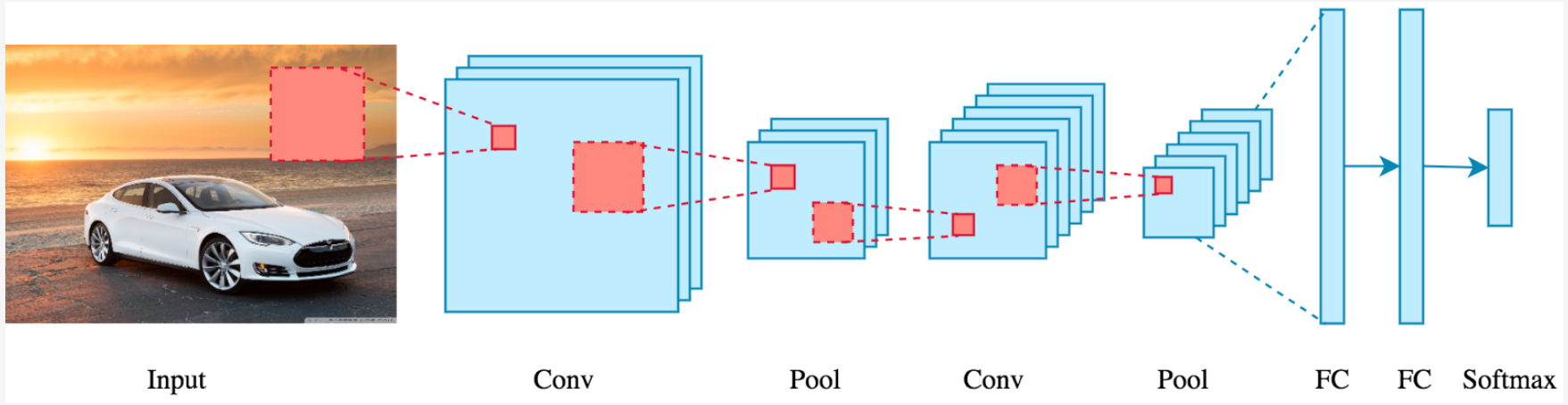
Convolutional Layer Details

- Typically, apply activation function to the neurons in convolutional layer (ReLU most popular)
- This seemingly simple idea from computer vision actually has a ton of nice properties for NNs



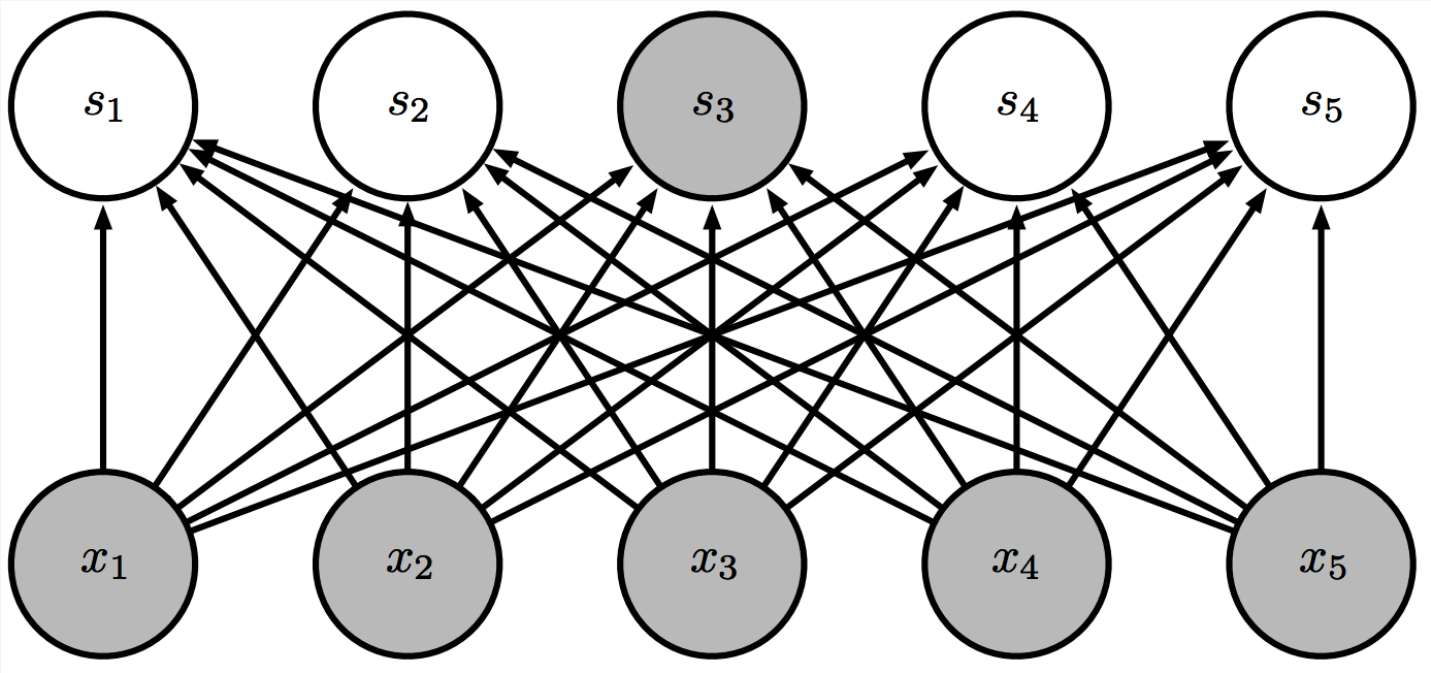
Convolutional Layer Details

- Typically, apply activation function to the neurons in convolutional layer (ReLU most popular)
- This seemingly simple idea from computer vision actually has a ton of nice properties for NNs
  - Convolutional Filters are **SPARSE**
  - Convolutional Filters utilize **WEIGHT SHARING**
  - Convolutional Filters are **TRANSLATION INVARIANT**



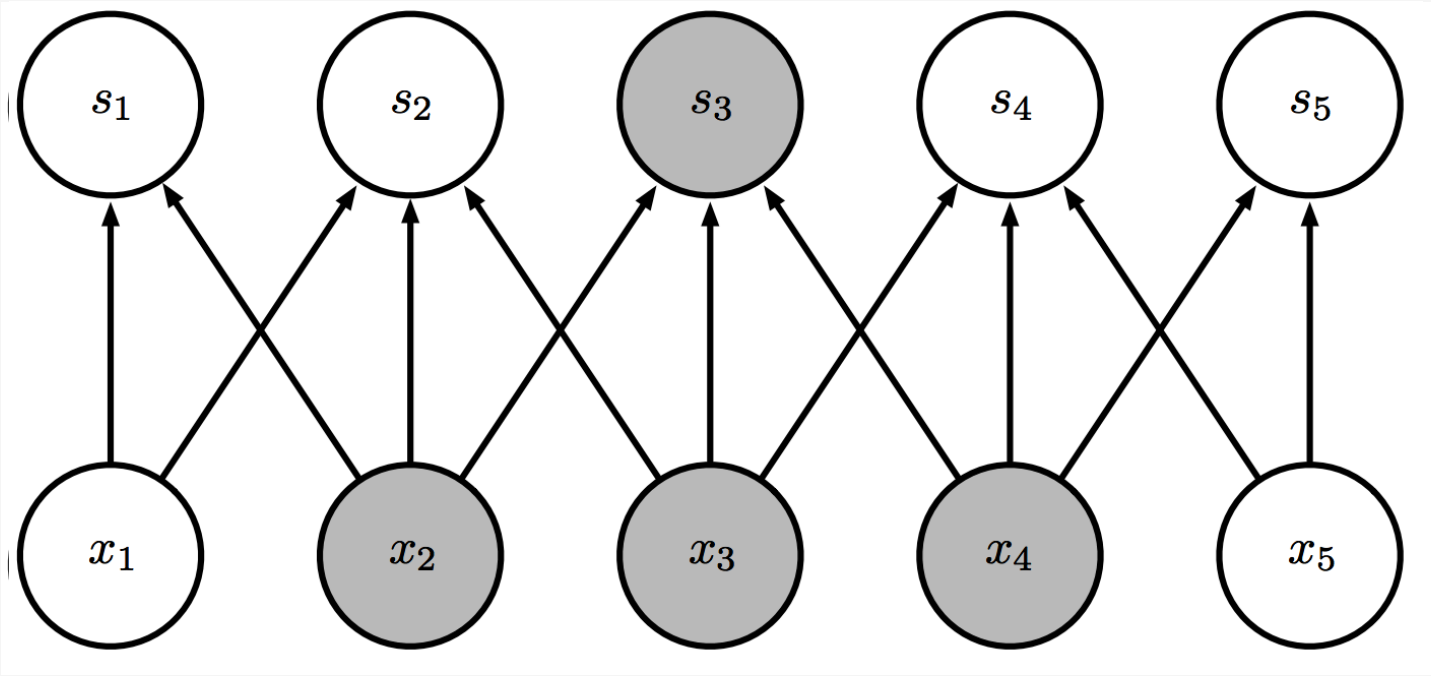
Convolutional Layer Details

- Typically, apply activation function to the neurons in convolutional layer (ReLU most popular)
- This seemingly simple idea from computer vision actually has a ton of nice properties for NNs
  - Convolutional Filters are **SPARSE**



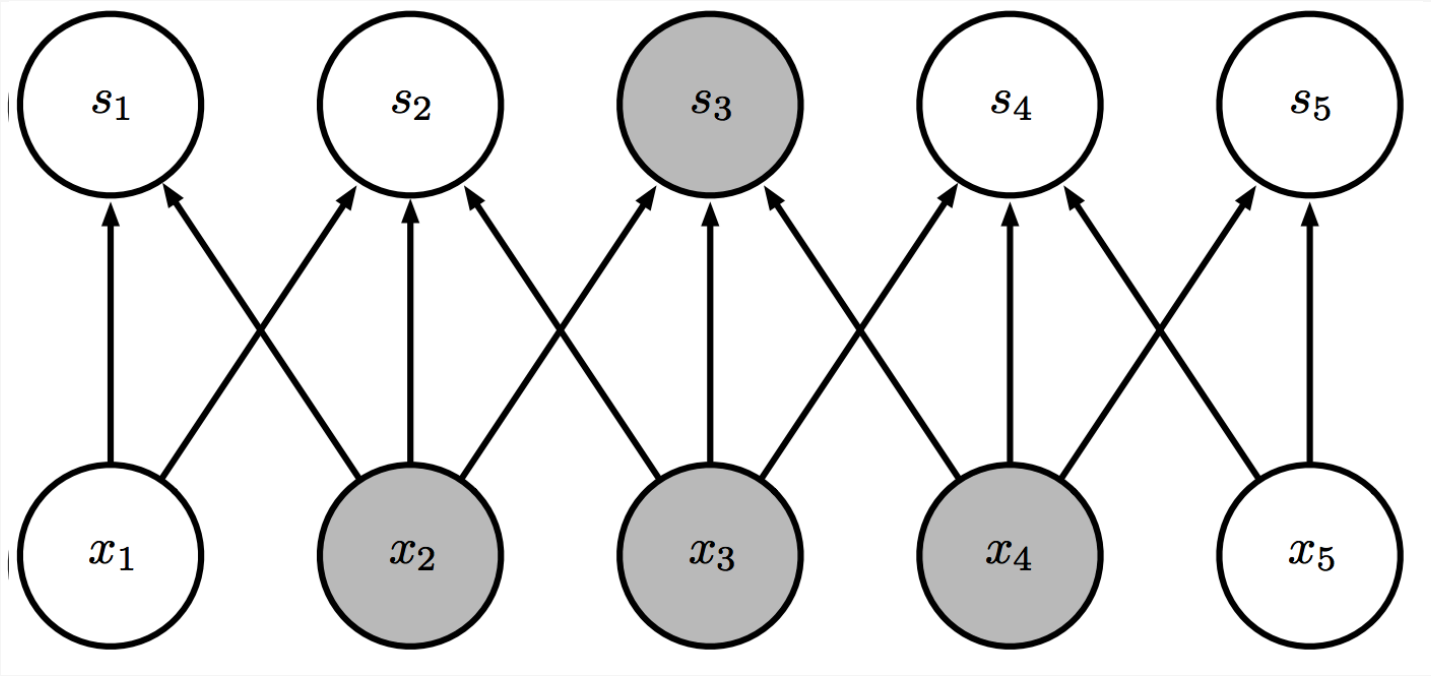
Convolutional Layer Details

- Typically, apply activation function to the neurons in convolutional layer (ReLU most popular)
- This seemingly simple idea from computer vision actually has a ton of nice properties for NNs
  - Convolutional Filters are **SPARSE**



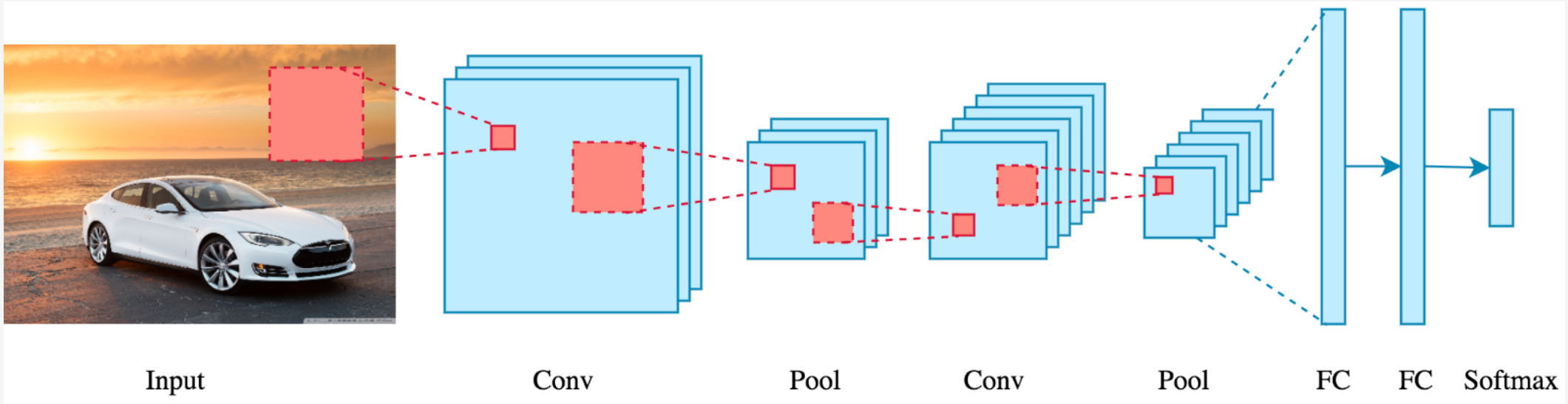
Convolutional Layer Details

- Typically, apply activation function to the neurons in convolutional layer (ReLU most popular)
- This seemingly simple idea from computer vision actually has a ton of nice properties for NNs
  - Convolutional Filters utilize **WEIGHT SHARING**



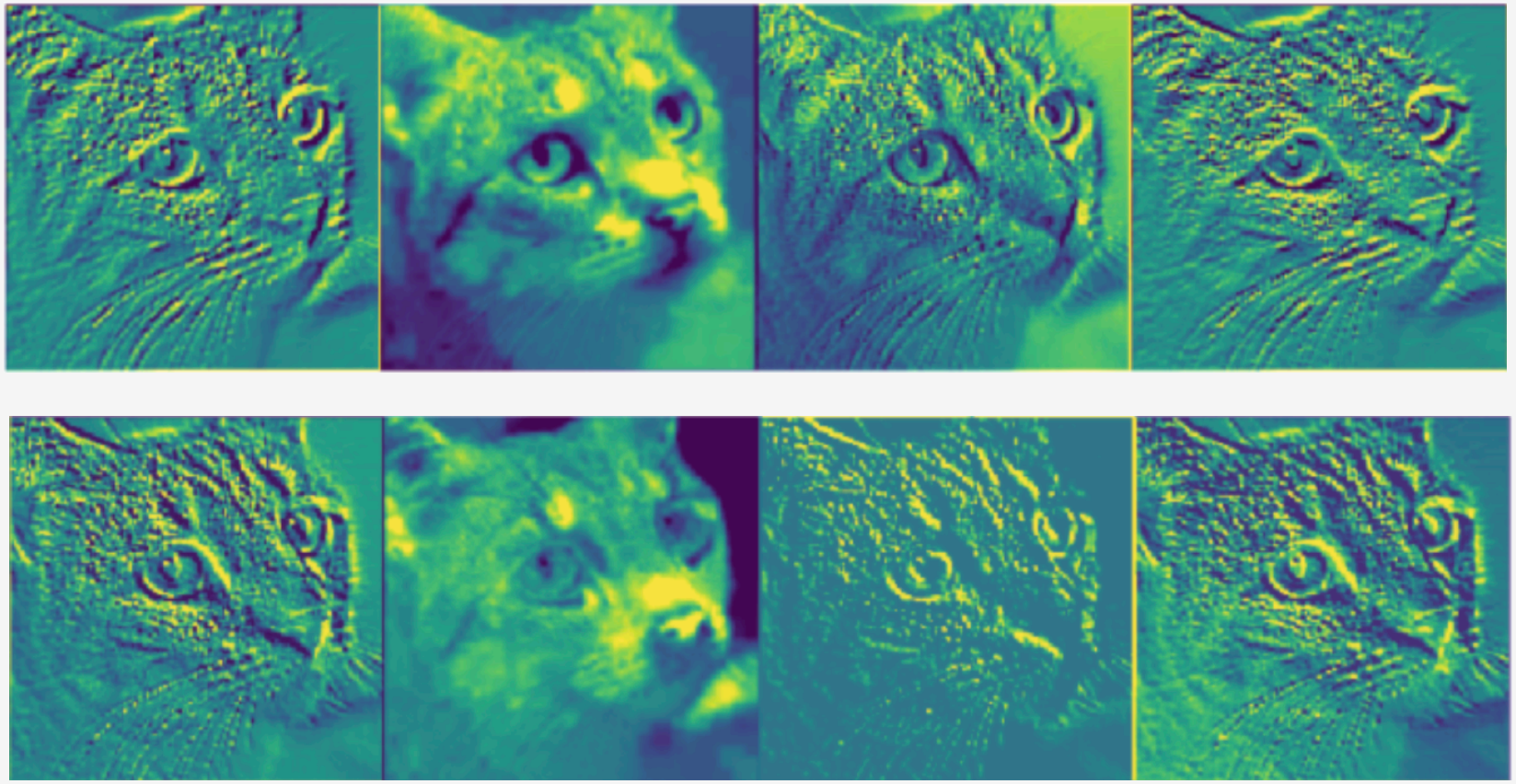
Convolutional Layer Details

- Typically, apply activation function to the neurons in convolutional layer (ReLU most popular)
- This seemingly simple idea from computer vision actually has a ton of nice properties for NNs
  - Convolutional Filters are **TRANSLATION INVARIANT**



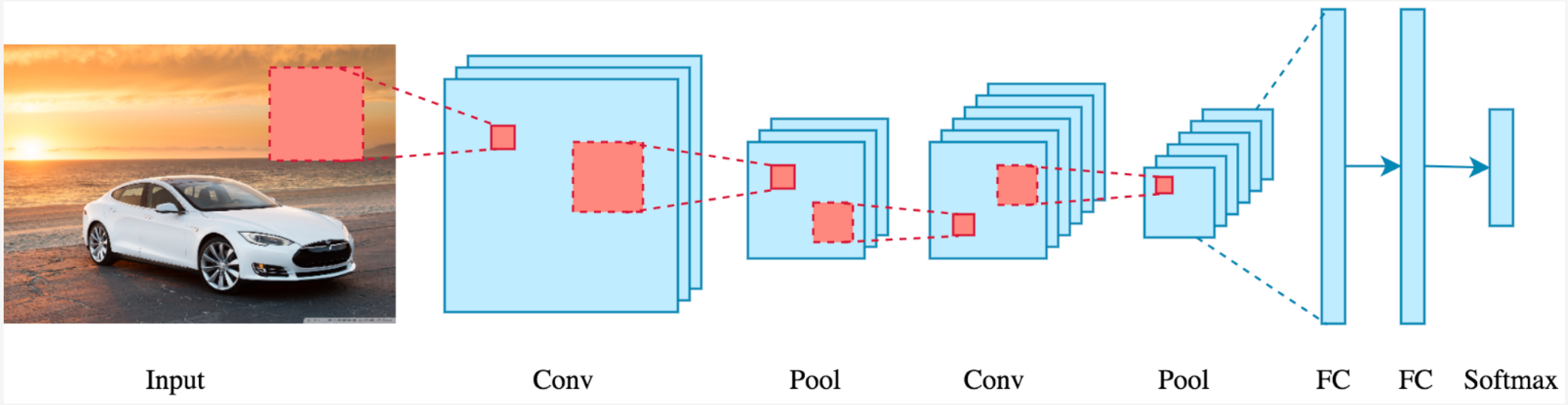
Low-Level Convolutional Features

- In first convolutional layer, can learn features like this



Pooling Layers

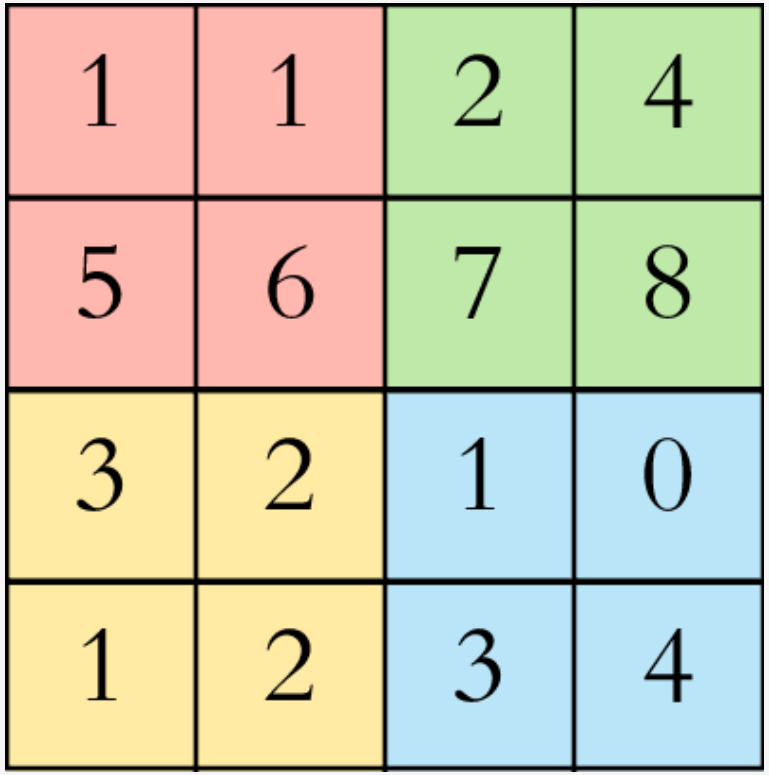
- The next layer after a Conv layer is typically a **Pooling Layer**
- Pooling layers act to:
  - Downsample the image (controls complexity)
  - Grow downstream receptive fields (allows local features to interact)

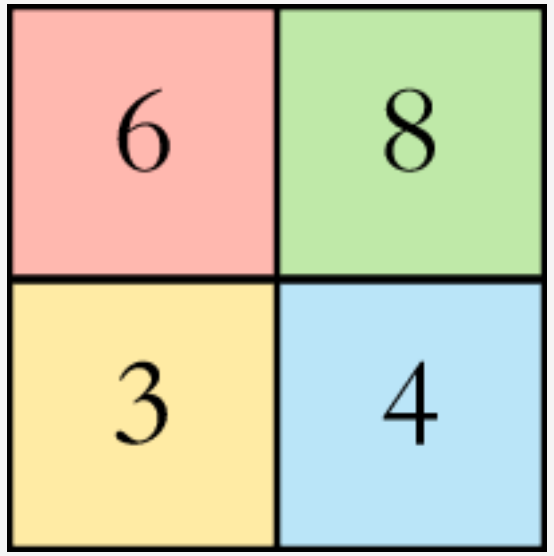


Pooling Layers

- The most common type of pooling layer is called a **Max Pool**

Conv Layer

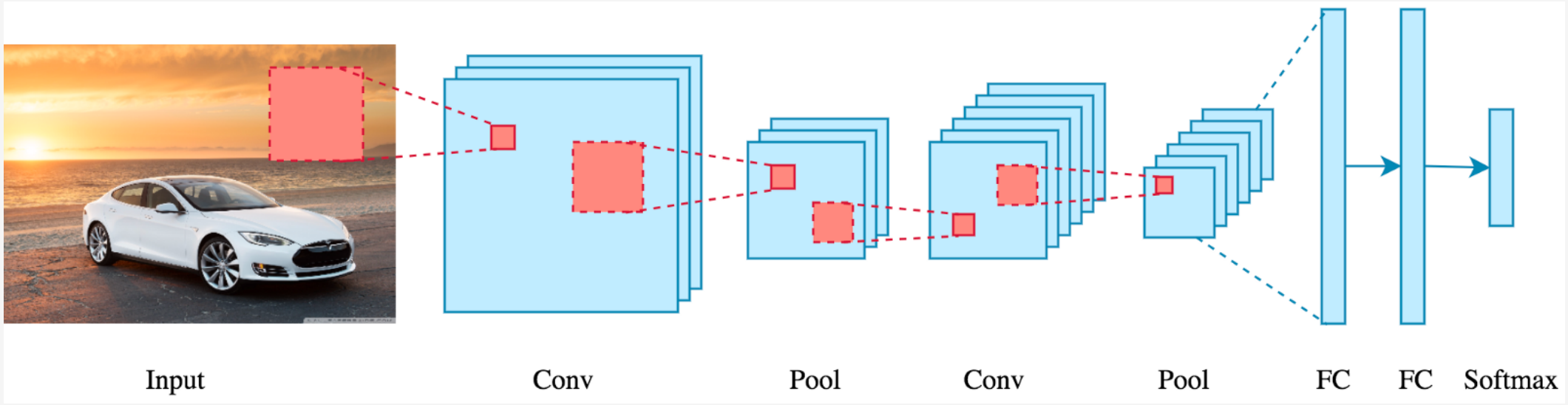




Pool Layer

Learning a Feature Heirarchy

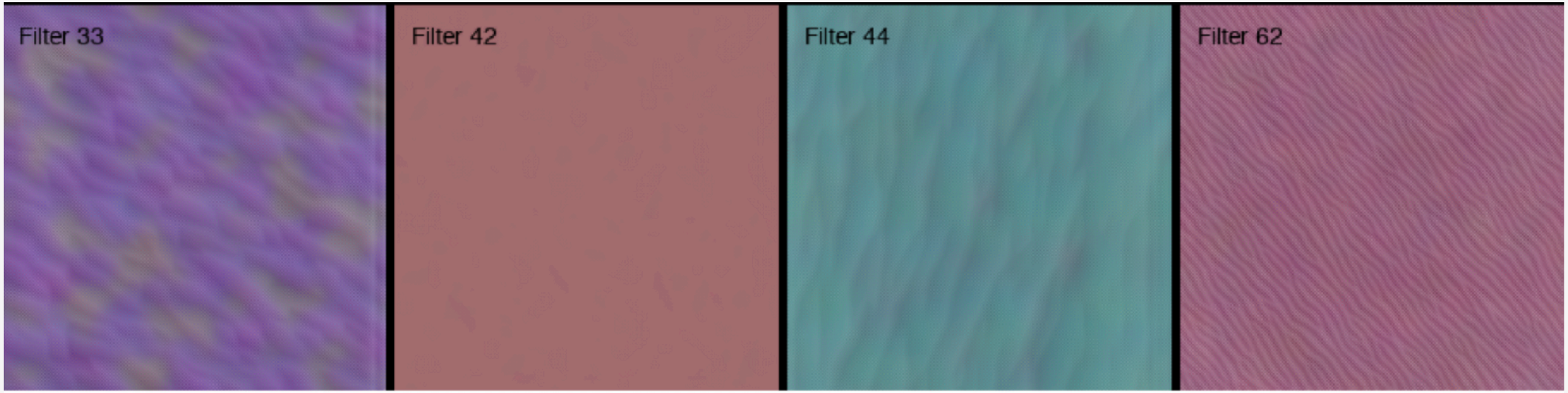
- Next we do another Conv layer followed by another Pool layer
- What does this do for us?
  - Downsample the image (controls complexity)
  - Grow downstream receptive fields (allows local features to interact)



Learning a Feature Hierarchy

- Next we do another Conv layer followed by another Pool layer
- What does this do for us?
- Allows local features to interact in similar way. Eventually learn more complex features

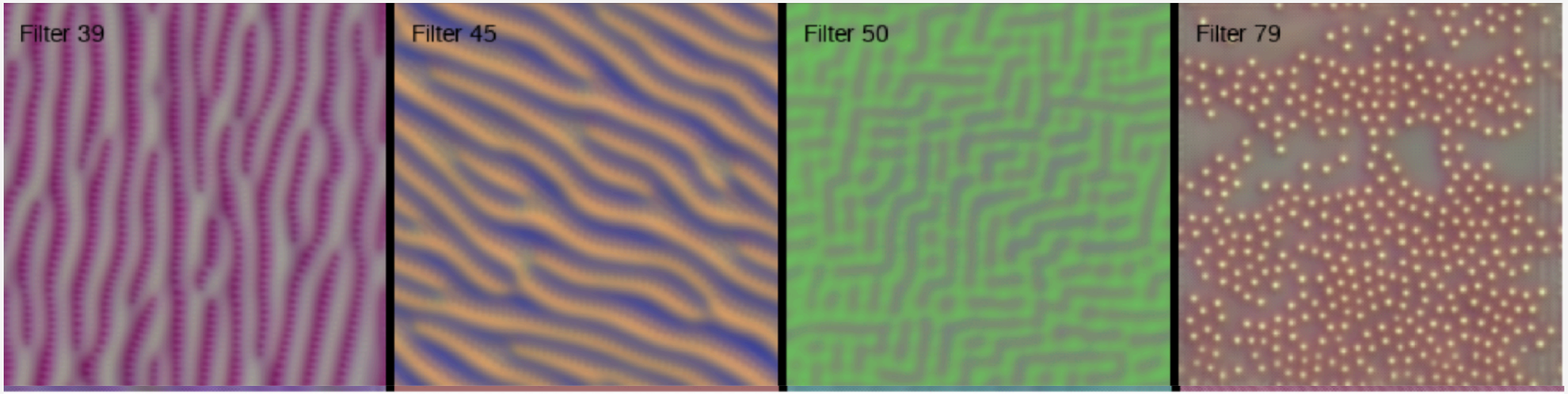
features learned in 1st Conv Layer



Learning a Feature Hierarchy

- Next we do another Conv layer followed by another Pool layer
- What does this do for us?
- Allows local features to interact in similar way. Eventually learn more complex features

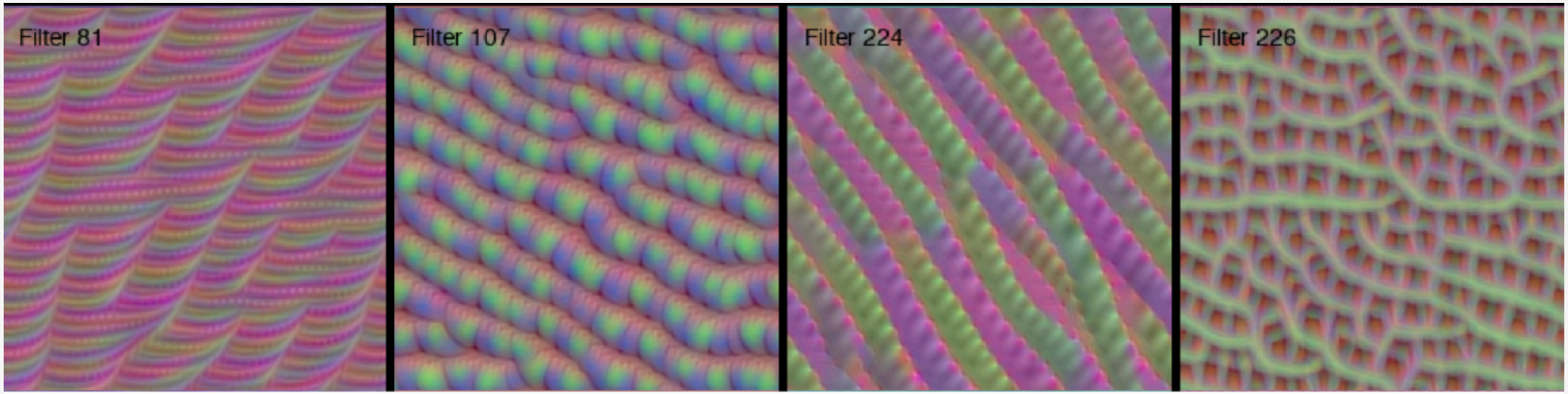
features learned in 2nd Conv Layer



Learning a Feature Hierarchy

- Next we do another Conv layer followed by another Pool layer
- What does this do for us?
- Allows local features to interact in similar way. Eventually learn more complex features

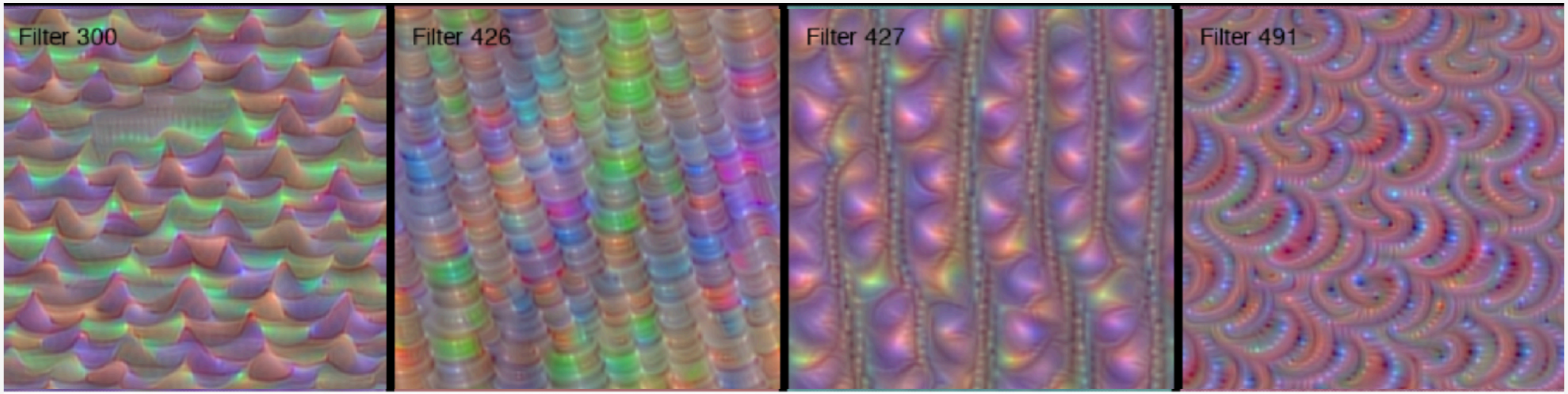
features learned in 3rd Conv Layer



Learning a Feature Hierarchy

- Next we do another Conv layer followed by another Pool layer
- What does this do for us?
- Allows local features to interact in similar way. Eventually learn more complex features

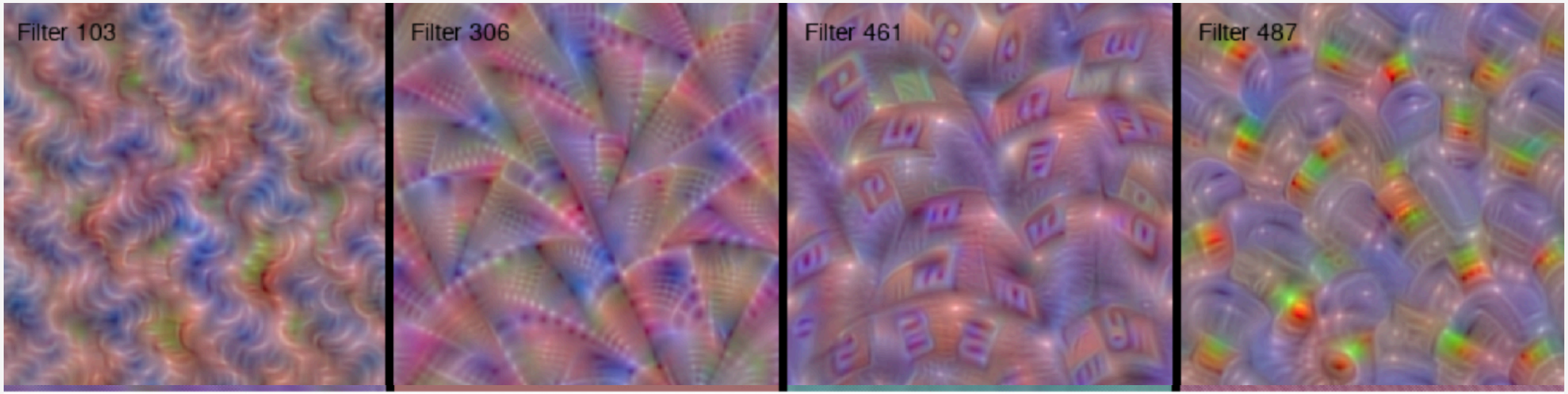
features learned in 4th Conv Layer



Learning a Feature Hierarchy

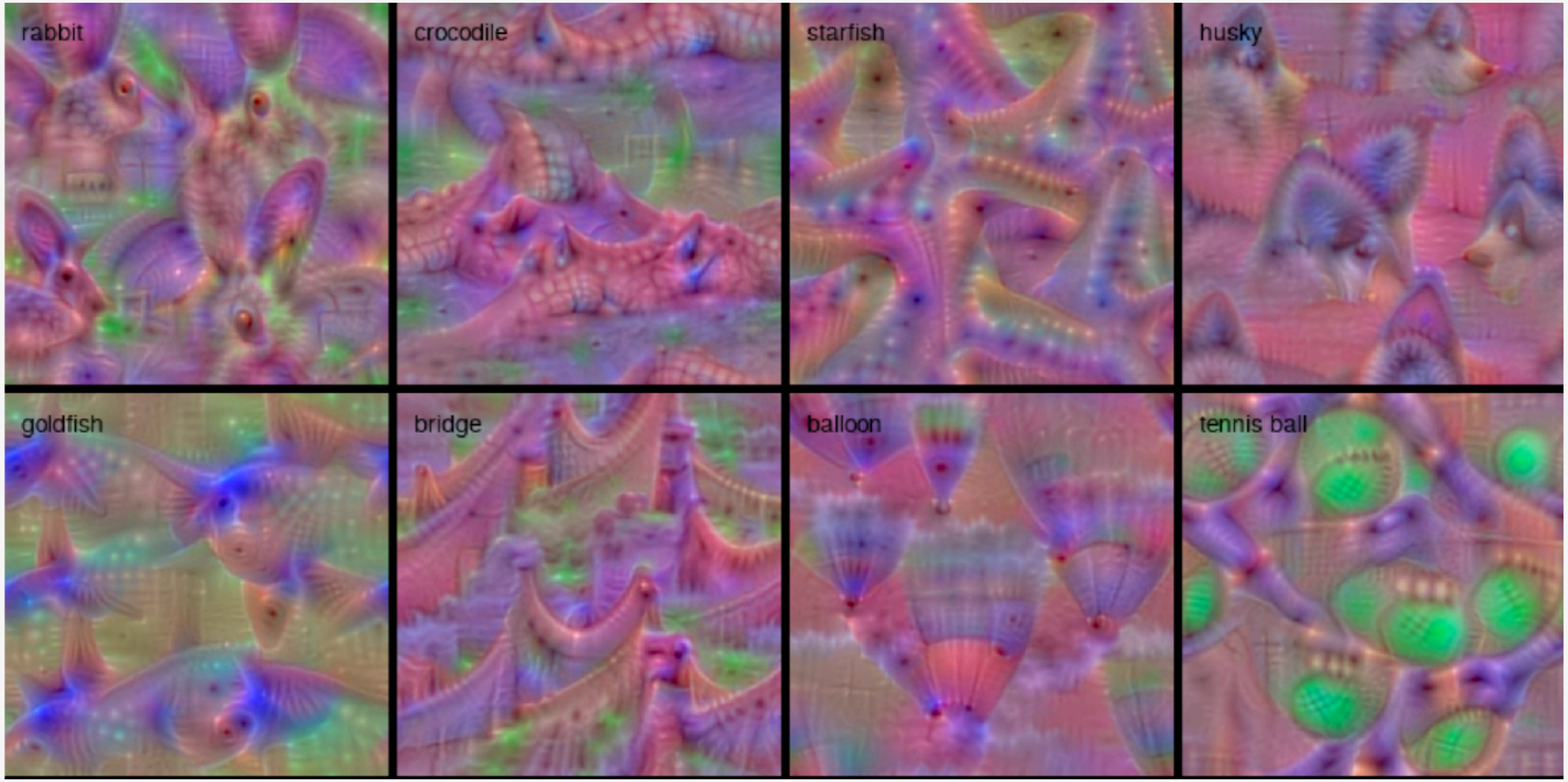
- Next we do another Conv layer followed by another Pool layer
- What does this do for us?
- Allows local features to interact in similar way. Eventually learn more complex features

features learned in 5th Conv Layer



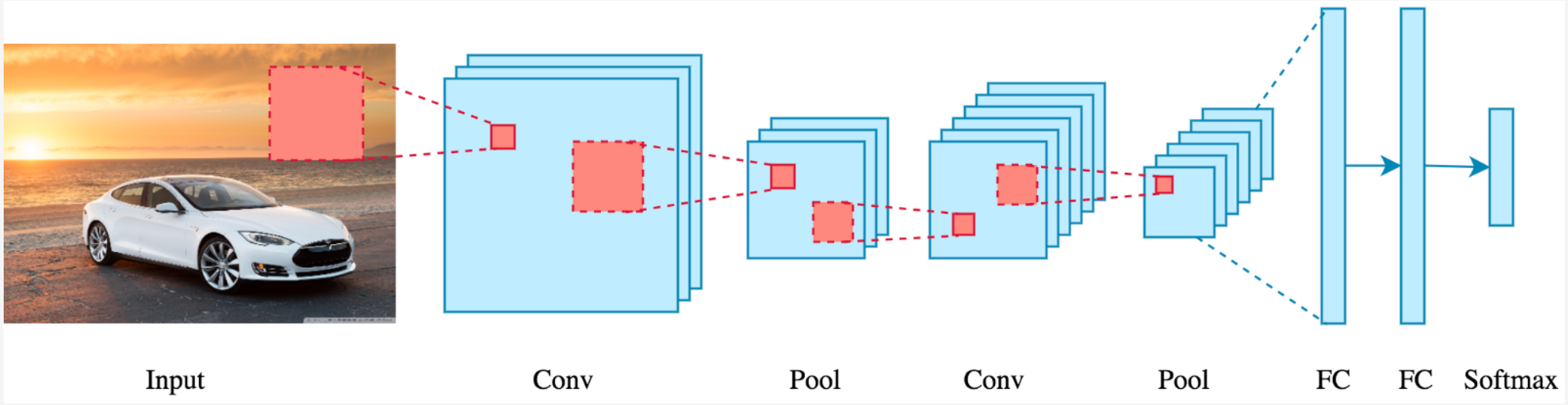
Learning a Feature Hierarchy

- What does the network see when we've learned these high-level features?

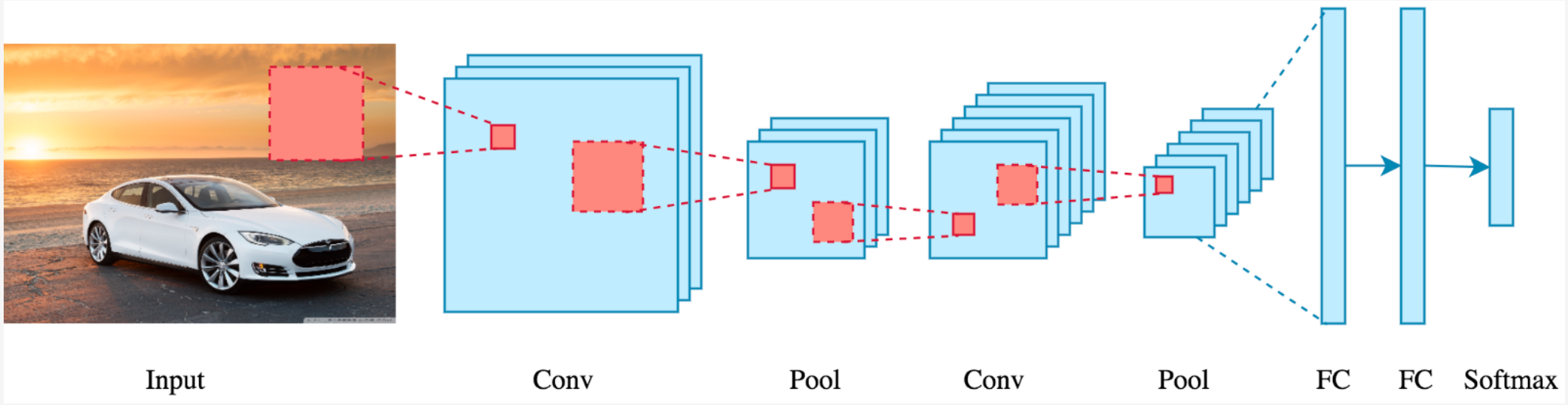


Making Predictions

- OK, we've created these (allegedly) wonderful features. How do we make predictions?
- The usual way. Feed result of last pooling layer into standard fully connected layers
- Then to a sigmoid (or something) output layer with neurons for each class (object)

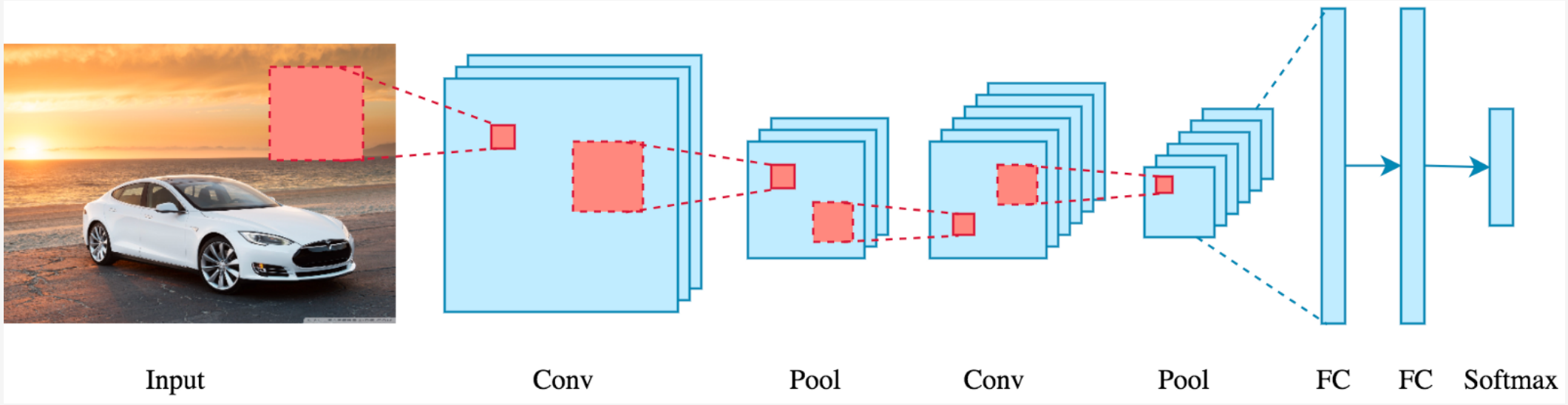


How Do We Train This Monstrosity?



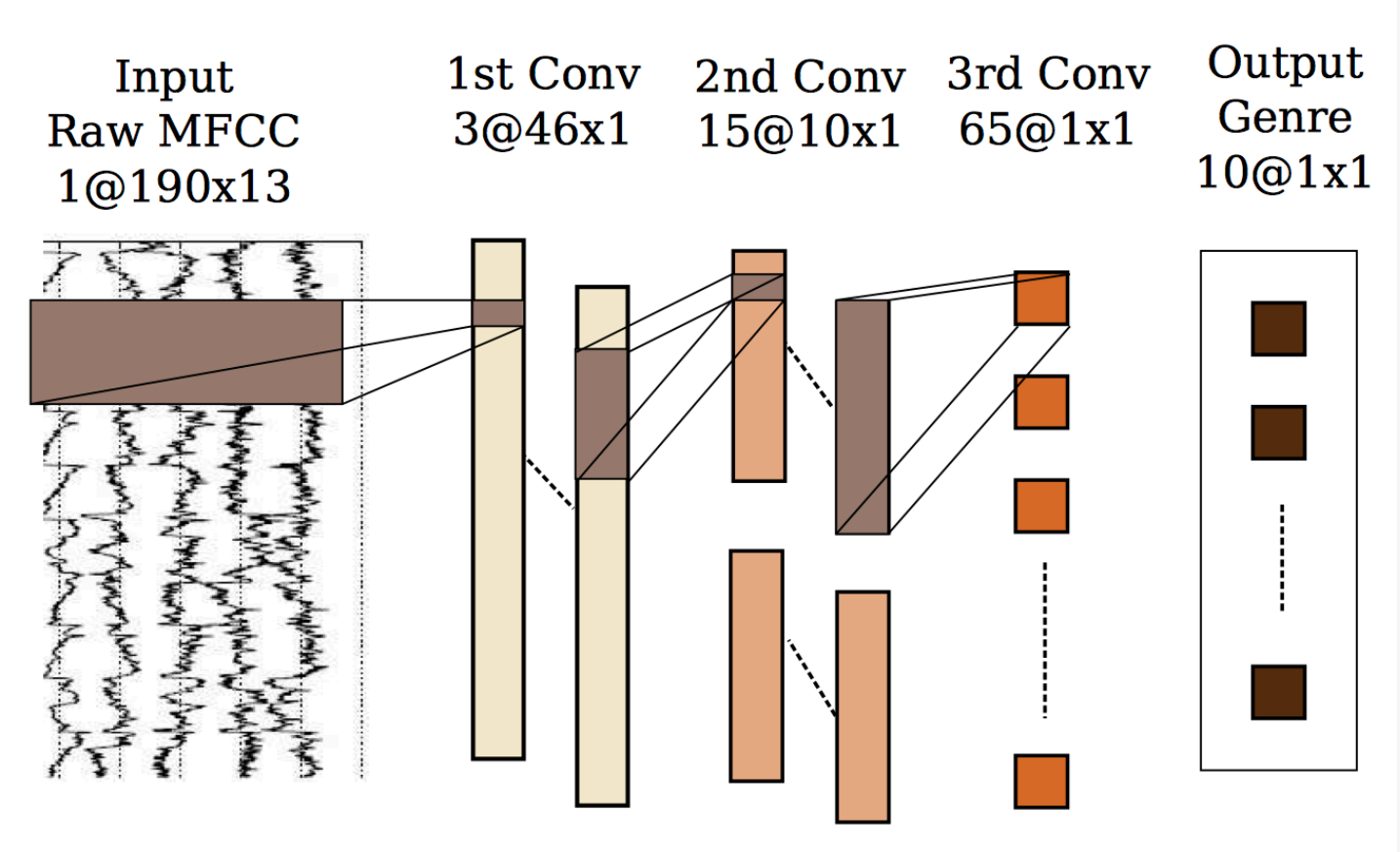
How Do We Train This Monstrosity?

- You probably guessed it. SGD with Back Propagation!
- Details/Data Structures a little different, but essentially the same ideas



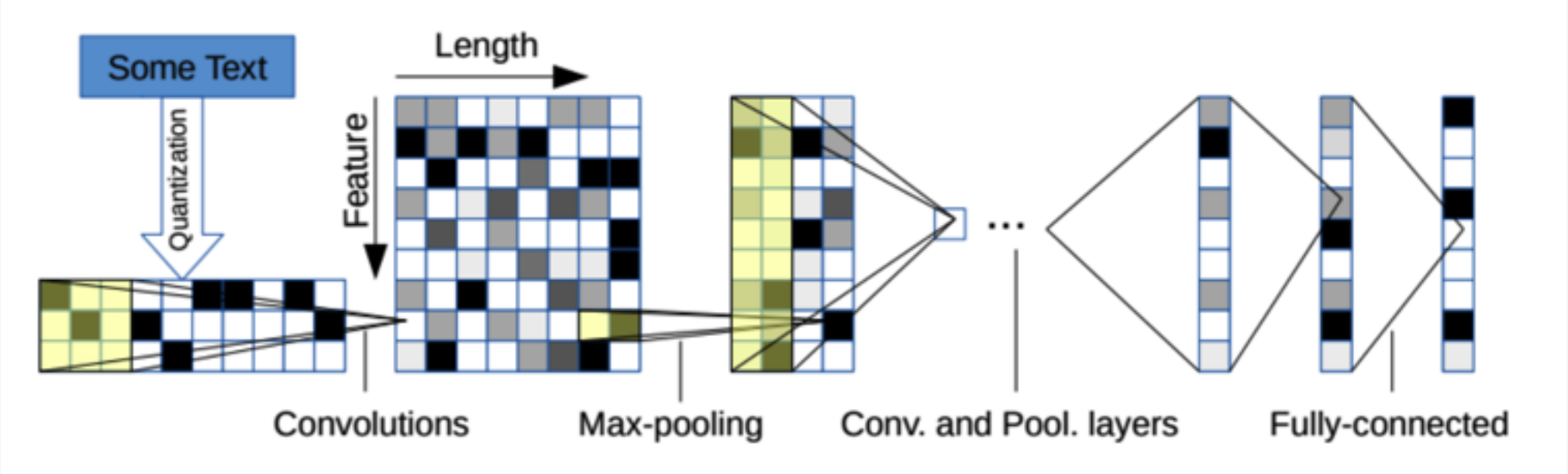
CNNs for 1D Signals

- Can apply CNNs to audio signals. Learn to detect music genre.



CNNs for Natural Language Processing

- Can apply CNNs to text to do things like sentence classification.



Acknowledgements

Many images in this lecture were taken from:

- **A. Dertat**. *Applied Machine Learning Part 4: Convolutional Neural Networks*
- **Goodfellow**, et. al. *Deep Learning*
- Zhang, et al. *Character-Level Convolutional Networks for Text Classification*
- Li, et al. *Automatic Music Pattern Feature Extraction Using Convolutional Neural Network*