# Deep Learning

Jin Xie <jin.xie@uky.edu>

Materials mostly by Andrew Ng

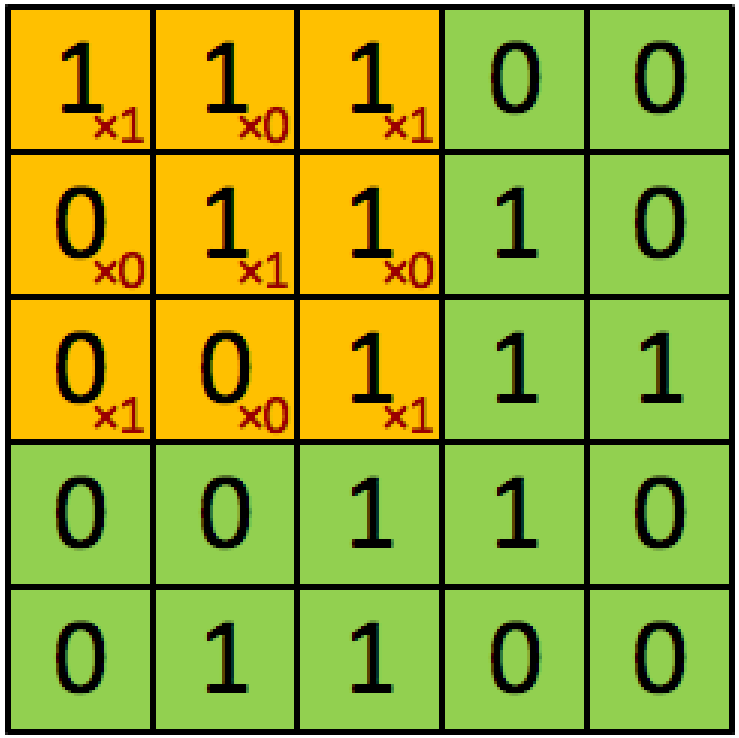
March 21, 2018

# Outline

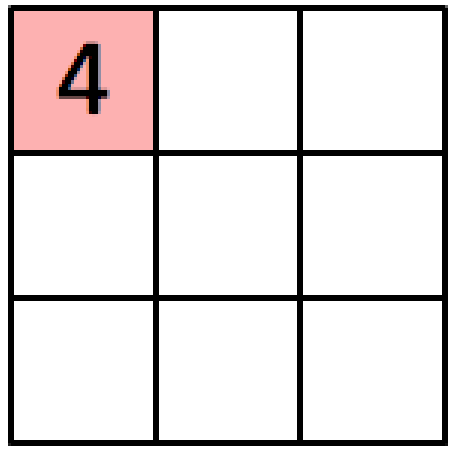
# Outline

# What is Convolution?






Image




Convolved Feature

# Filter Matrix: Blur



| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

(a) Filter

(b) Convolved Image

Figure: Blur

# Filter Matrix: Sharpen



| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | -1 | 0 | 0 |
| 0 | -1 | 5 | -1 | 0 |
| 0 | 0 | -1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

(a) Filter

(b) Convolved Image

Figure: Sharpen

# Filter Matrix: Edge Enhance



| | | | | |
|---|---|---|---|---|
| | 0 | 0 | 0 | |
| | -1 | 1 | 0 | |
| | 0 | 0 | 0 | |
| | | | | |

(a) Filter

(b) Convolved Image

Figure: Edge Enhance

# Filter Matrix: Edge Detect



| | | | | |
|---|---|---|---|---|
| | 0 | 1 | 0 | |
| | 1 | -4 | 1 | |
| | 0 | 1 | 0 | |
| | | | | |

(a) Filter

(b) Convolved Image

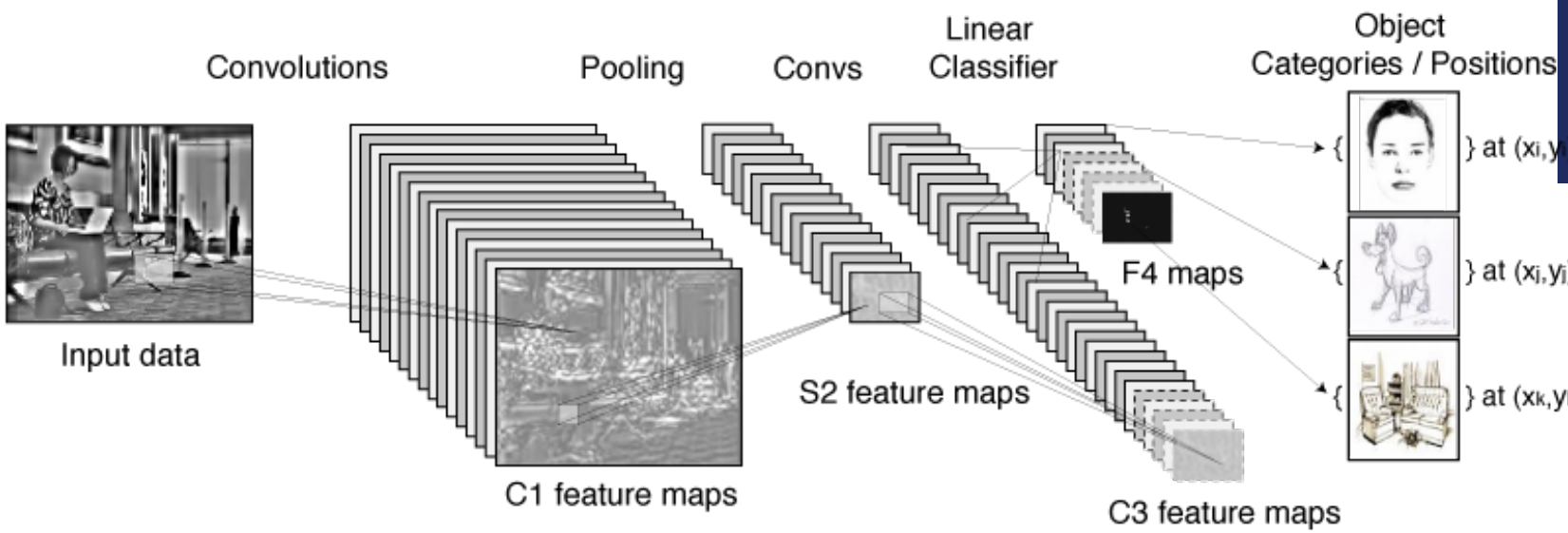
Figure: Edge Detect

# Feature Map or Convolved Image



# Each filter result in a feature map

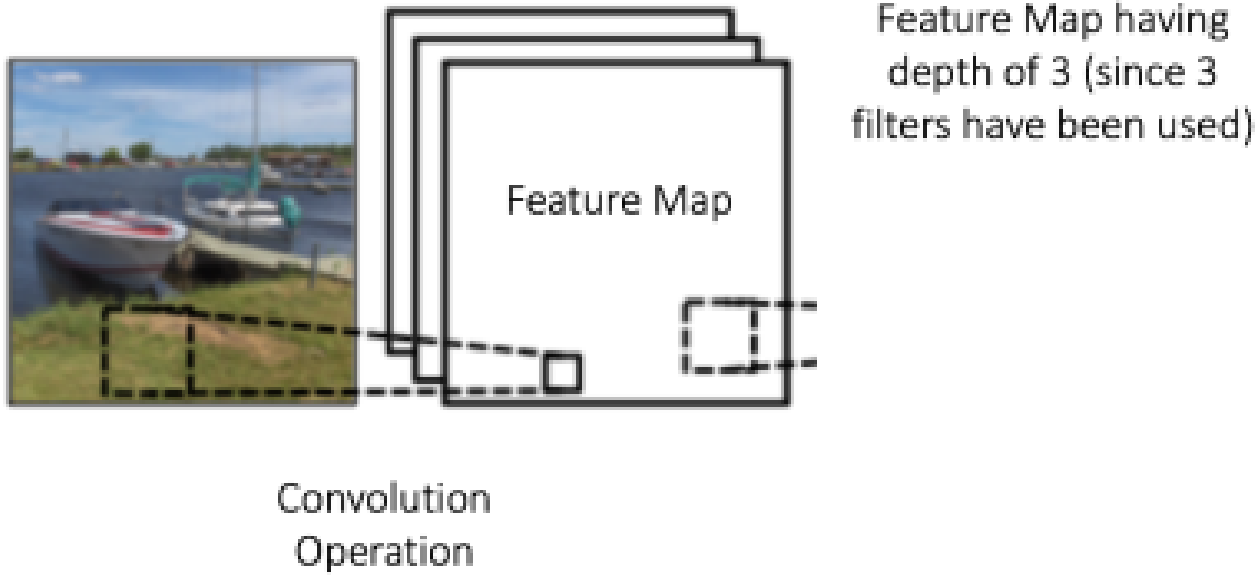




# Outline

# Parameter

**Depth**: Depth corresponds to the number of filters we use for the convolution operation.

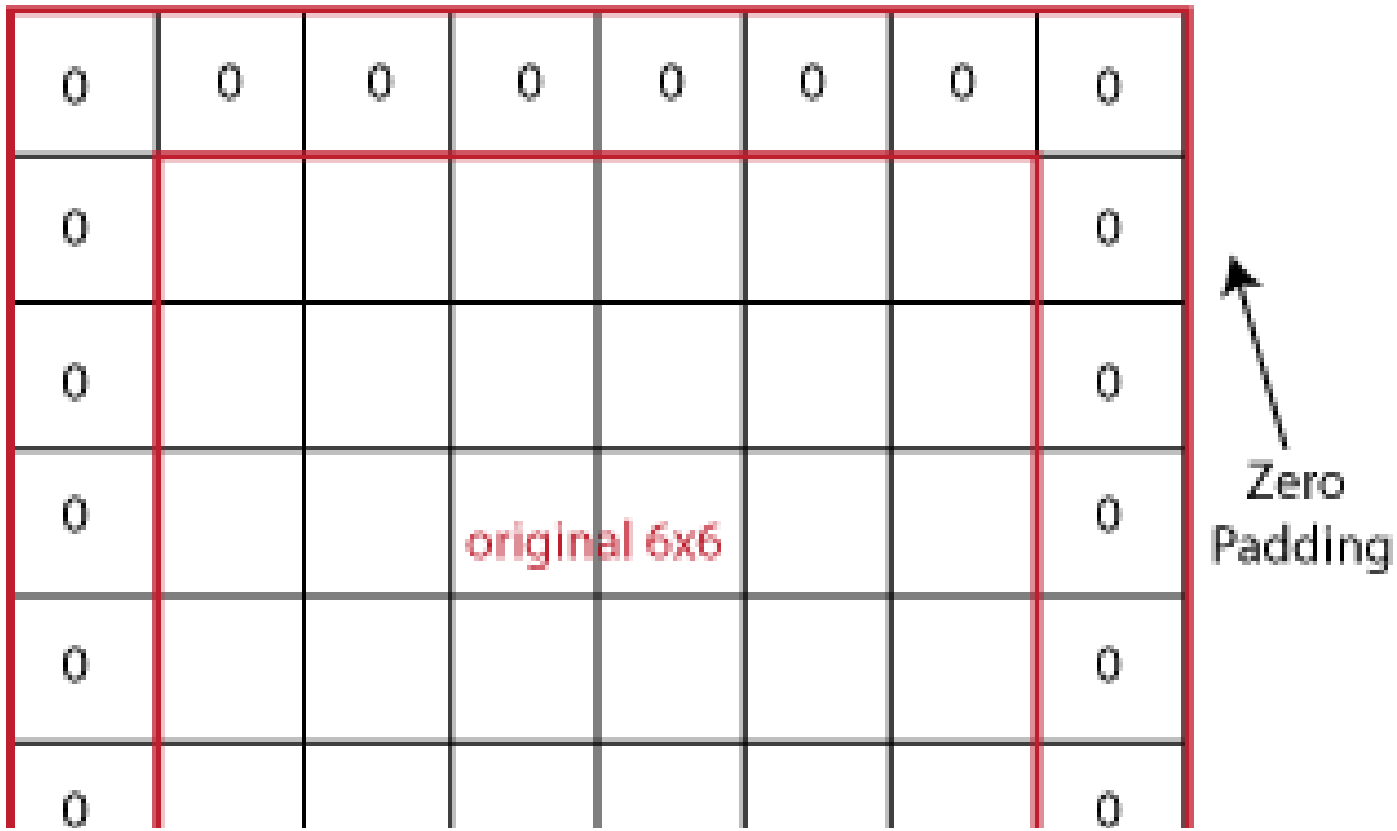


# Parameter

**Stride**: Stride is the number of pixels by which we slide our filter matrix over the input matrix. When the stride is 2, then the filters jump 2 pixels at a time as we slide them around.

# Parameter

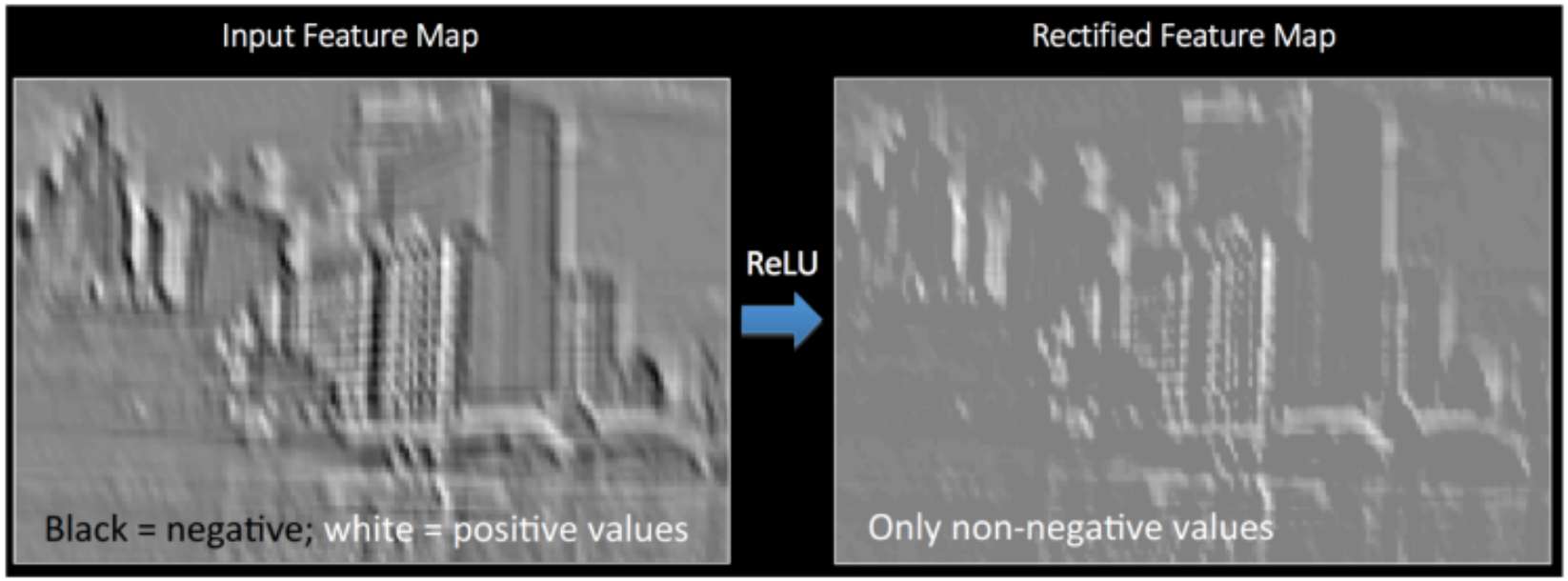
**Zero-padding**: Sometimes, it is convenient to pad the input matrix with zeros around the border, so that we can apply the filter to bordering elements of our input image matrix. There are also other types of padding methods.



# Outline

# ReLU

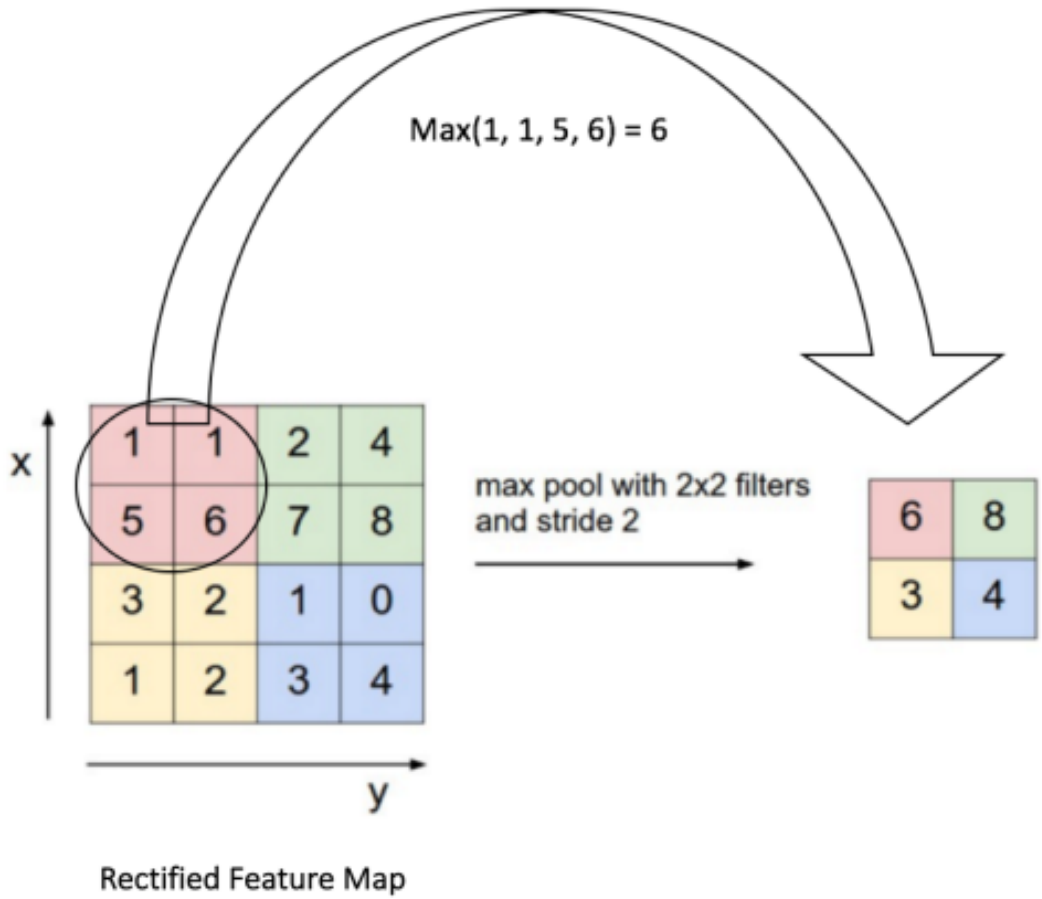
The purpose of ReLU is to introduce non-linearity in our ConvNet, since most of the real-world data we would want our ConvNet to learn would be non-linear (Convolution is a linear operation – element wise matrix multiplication and addition, so we account for non-linearity by introducing a non-linear function like ReLU).



# Outline

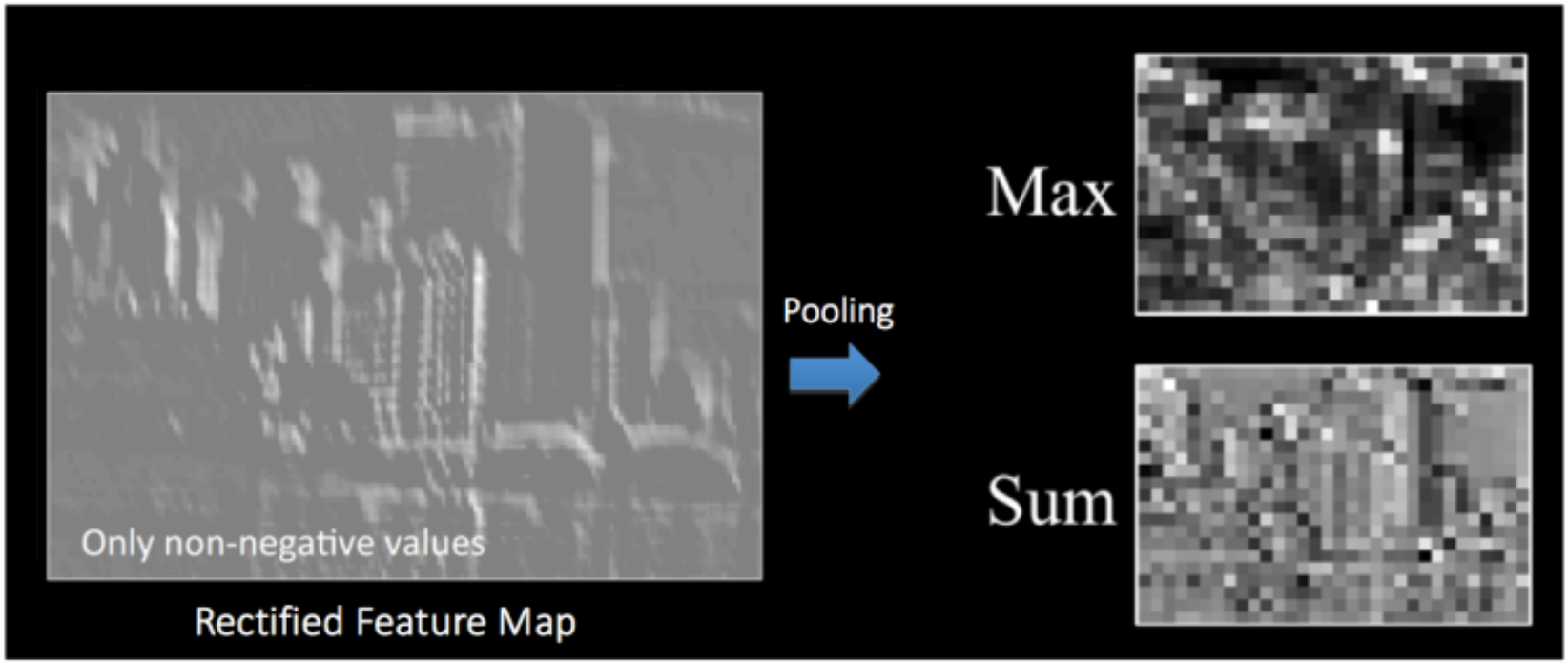
# Pooling Layer

Spatial Pooling (also called subsampling or downsampling) reduces the dimensionality of each feature map but retains the most important information. Spatial Pooling can be of different types: Max, Average, Sum etc.



# Pooling Layer





# Outline

# Add Fully Connected Layer



In the end, we flatten all the arrays to 1-dimension vectors and use them for training.