



BEHLUL

Behlul is an Efficient High Level Useful Library

Deep Learning from Scratch in C++

Mustafa Erdogan

03709173

Design is important

- Object oriented nodes-layers
- Only 1 class design (XOR)
- Multiple Layers (MNIST)

Current layers

- Convolution Layer
- ReLU Layer
- Max Pool Layer
- Dense Layer
- Softmax Layer
- Cross Entropy Layer

Data

- Driver Images 480x640
- The 10 classes to predict are:
- c0: safe driving
- c1: texting - right
- c2: talking on the phone - right
- c3: texting - left
- c4: talking on the phone - left
- c5: operating the radio
- c6: drinking
- c7: reaching behind
- c8: hair and makeup
- c9: talking to passenger



Preprocess

- cropped 80px left-right to make square
- resize to 28x28

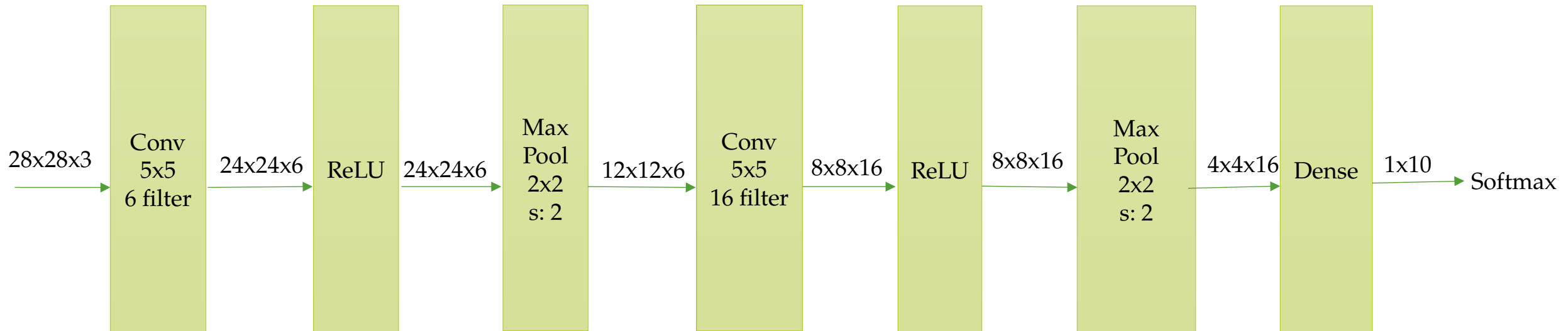
Split

- ~80% train data
- ~10% validation data
- ~10% test data

Model

- Conv_Layer 1: input: 28x28x3 filter: 5x5x3 #filters: 6 stride: 1 output: 24x24x6
- ReLU 1: output: 24x24x6
- Max_Pool 1: input: 24x24x6 filter: 2x2x6 stride: 2 output: 12x12x6
- Conv_Layer 2: input: 12x12x6 filter: 5x5x6 #filters: 16 stride: 1 output: 8x8x16
- ReLU 2: output: 8x8x16
- Max_Pool 2: input: 8x8x16 filter: 2x2x16 stride: 2 output: 4x4x16
- Dense_Layer: input: 4x4x16 output: 1x10
- Softmax
- Cross_Entropy

Model visual



Results

- 5 epoch: ~30 mins.
- Training accuracy: 0.78
- Validation accuracy: 0.75
- Test accuracy: 0.72

What's next?

- Deterministic initialization
- Xavier initialization
- Dropout
- More epochs
- More complicated model