Log

Week 1 19.04.18

- Feedback on project proposal
- Overview of project
  - simplification
  - binary image → numbers → straight text → Classify
- init; github - atom
- first test of charcter Segmentation

- Charcter Segmentation - Projection Histograms - OpenCV
  - By projecting the histogram of the binary image on the Y-axis, we can find where the sentences/lines of text appears. Following, a projection histogram on the X-axis can discover where the charecters appear.

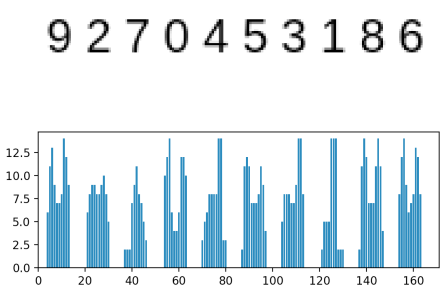

Figure 1: [0-9] segmented with projection histogram

- Classification - Perceptron neural network - TensorFlow
  - MNIST dataset - Datasett consisting of several thousand handwritten labeled numbers
    - Numbers ranging from [0-9]
    - Images are 28x28pixels
  - Hyperparameter tuneing
    - Activation function
    - Number of hidden layers
    - Nodes in hidden layers
    - Cost function
    - Optimization function
    - Learning rate
  - Theoretic accuracy of the network with 2 hidden layers 98%
    - Measured accuracy 97%

```
4690-p2018|Sadegh(master)$ p3 src/find_symbol.py
Model restored
Extracted text: 9220453189
```

Figure 2: First output with classification. input see Figure 1

Week 3 03.05.18

- Rotation of text
  - Hough transform
  - *cv2.minAreaRect()*
- How to distinguish between upside-down, and verticle vs horisontal text segments
  - Classifiy in all 4 rotations, and choose the classification with highest avrage confidence
- Classification - Perceptron neural network - Error
  - Error rate too high, test-set accuracy 97%, validation set accuracy $< 50\%$
  - CNN - TensorFlow Estimator API
    - Challenging documantation; load/save models
  - Dataset - FNIST - Group contribution
    - Dataset including several fonts
    - English alphabet, and numbers [0-9]

- —
  - —
  - —
- ——