UNIK4690 Project

Akhsarbek Gozoev - akhsarbg
Sadegh Hoseinpoor - sadeghh
Key Long Wong - keylw

May 18, 2018

Project description

The purpose of the software is to recognise text from any surface with uneven lighting. Hence this falls under the "Optical character recognition" (OCR) problem

As OCRs are still a challenging task even for companiese like Google, ref. reader to Googles OCR translator application on smartphones, drawbacks such as; difficulty finding all the text on the photo becasue of lighting, noise etc., therefore we will have to limit our software significantly.

Initial limitations

- English alphabet + numbers [0-9]
- Homogeneous background
- Computer printed text
-

Project components.
The group have come to the conclusion that the OCR software has 3 main components to it.

1. *Finding text on a photo and returning the text within a bounding box*
2. *Do preprocessing on the segmented text; rotation, greylevel normalization, symbol segmentation*
3. *Classification of the symbols*

Additionally there is one more very important component for this OCR software to work, ***labled data***. Even though one might not need to code for this part, a good poolof labeld data is needed to be able to classify symbols. More on this under description for this part.
(4. *Gathering labeled data to train a classification algorithm*)

Project INIT.
As we want to test the prof of concept first we simplyfied the SW to just be:
Recognise numbers [0-9] from a binary img, with computer printed numbers on homogenous background. Contaioning one horizontal line of numbers

Second step
Assume sequence(n number of lines) of numbers. not horizontal lines. on homogeneous background.

Report

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

Week 2 26.04.18

- Charcter Segmentation - Projection Histograms - OpenCV
  - By projection 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.

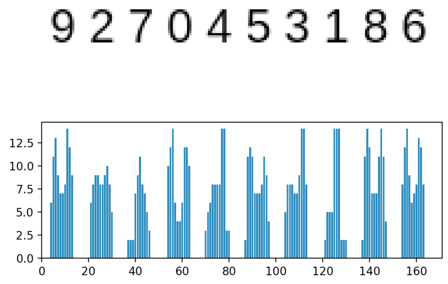
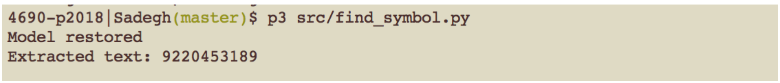


Figure 1: [0-9] segmented with projection histogram

- 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
    - ∗ Optimazation 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]

Week 4 17.05.18

- ——
  - ——
  - ——
- ———