

IN4085 Pattern Recognition

Final Project

“Classification of Handwritten Digits in Different Scenarios”

Tung Phan, ttphan, 4004868

Kevin van Nes, kjmvannes, 4020871

January 13, 2015

1 Introduction

This report concludes the final assignment of the TI4085 Pattern Recognition course. For this assignment, two classifiers had to be constructed for the classification of handwritten digits. One classifier was made for the scenario in which a lot of training data is available, whilst the other classifier was made for the scenario in which very few training data is available.

This report contains the details about the different phases in constructing the classification system, as well as the design choices that were made and why these were made. Furthermore, the system has been ran against a set of benchmark data, about which more information can also be found in this report. After successfully having created the system, it was ran against a so-called ‘live test’, which is a test in which real, self-written data was used to test the system against. Finally, some recommendations will be given to the company that will receive the system.

2 Preprocessing

The first phase we went through during this project was the phase in which we preprocessed the data. This was a necessary step, because the data (i.e. the handwritten digits) differed tremendously amongst each other, even within the same class. An example of this would be that a ‘4’ can be written in different ways or that digits would be sheared in various directions. To solve these inequalities, a few steps were taken to normalize the data, as to try to make the data as similar as possible, both inter-class and intra-class.

2.1 Boxing and resizing

The first step that was taken to preprocess the data was to normalize all the data objects in terms of their size. In order to do this, the PRTools functions ‘im_box’ and ‘im_resize’ were used. We decided to resize the images to 32x32 pixels, which later turned out to give better results than images of 64x64 pixels and images of 16x16 pixels. This is probably related to the fact that we use pixels as our features, so that smaller/larger images would have too few or too many features, respectively.

2.2 Binary image operations

2.3 Shearing for normalizing diagonally orientated digits

3 Feature Extraction

Hier PCA

No clue

4 Classifiers

Hier korte sectie intro

4.1 Scenario 1

Scenario 1 classifiers en percentage dat uit eigen tests kwam. (gemiddelde van 5/10 tests?)

4.2 Scenario 2

Scenario 2 classifiers en percentage dat uit eigen tests kwam. (gemiddelde van 5/10 tests?)

5 Benchmarking

Korte intro

5.1 Classifier Scenario 1

Benchmark resultaten + vergelijking eigen tests (evt. conclusies trekken uit verschillen/overeenkomsten)

5.2 Classifier Scenario 2

Benchmark resultaten + vergelijking eigen tests (evt. conclusies trekken uit verschillen/overeenkomsten)

6 Live Test

Korte intro

6.1 Segmentation

Hoe hebben we gesegmenteerd?

6.2 Preprocessing and Classification

Vertellen dat preprocessing en classification op dezelfde manier gebeurd is. Zowel Classifier 1 als 2 als trainer gebruiken, resultaten vergelijken met andere classification tests.

7 Recommendations

Recommendations voor bonusssspunt

8 Conclusion

Korte conclusie