

Intelligent Products

% Environment recognition by using audio samples

Assignment:
Intelligent Products DG234

Assignor:
dr.ir. E. I. Barakova

Student:
Bram Rutten
s140795
Industrial Design,
B3.3 (premaster)

Index

Introduction	4
Digit recognition	5
Results	6
Environment recognition with audio samples	7
Results	8
Reflection	9

Introduction

During the assignment Intelligent Products I worked on two projects. The first project is about recognizing hand drawn digits. I worked together with Lennard Bunk and Dimi Wiertz.

The second project is about recognizing environments by analyzing audio samples.

This second project is connected to my main project this semester. The main project is about a product that executes different actions based on the environment you are in.

Digit recognition

We used a range of numbers from 0 to 9. The drawn numbers were converted to a row vector by placing them on a 9 by 6 grid matrix. For example, for a number 3 it looks like this (figure 1).

The corresponding vector will look like this:

```
0 5 3 2 5 3 0 0 4 3 2 3 3 3 0 %0
0 1 2 6 0 0 0 0 2 2 2 1 1 1 0 %1
0 0 4 4 5 0 0 0 3 2 1 2 2 3 0 %2
0 3 4 5 5 2 0 0 2 4 3 2 2 5 1 %3
0 1 5 3 3 0 0 0 1 1 1 2 4 2 1 %4
0 4 4 3 5 0 0 0 0 4 1 4 1 3 3 %5
0 4 4 3 4 0 0 0 0 3 2 4 2 4 0 %6
0 0 2 4 4 0 0 0 3 1 1 3 1 1 0 %7
0 7 4 4 7 0 0 4 2 3 4 3 2 4 0 %8
0 4 3 7 0 0 0 1 3 2 3 1 1 3 0 %9
```

These are vectors for the numbers 0 to 9. For training the program more vectors are required. We used four in total (figure 2).

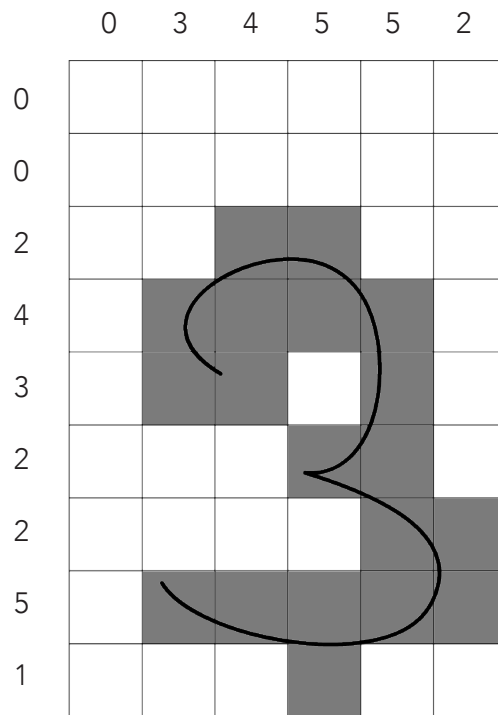


figure 1

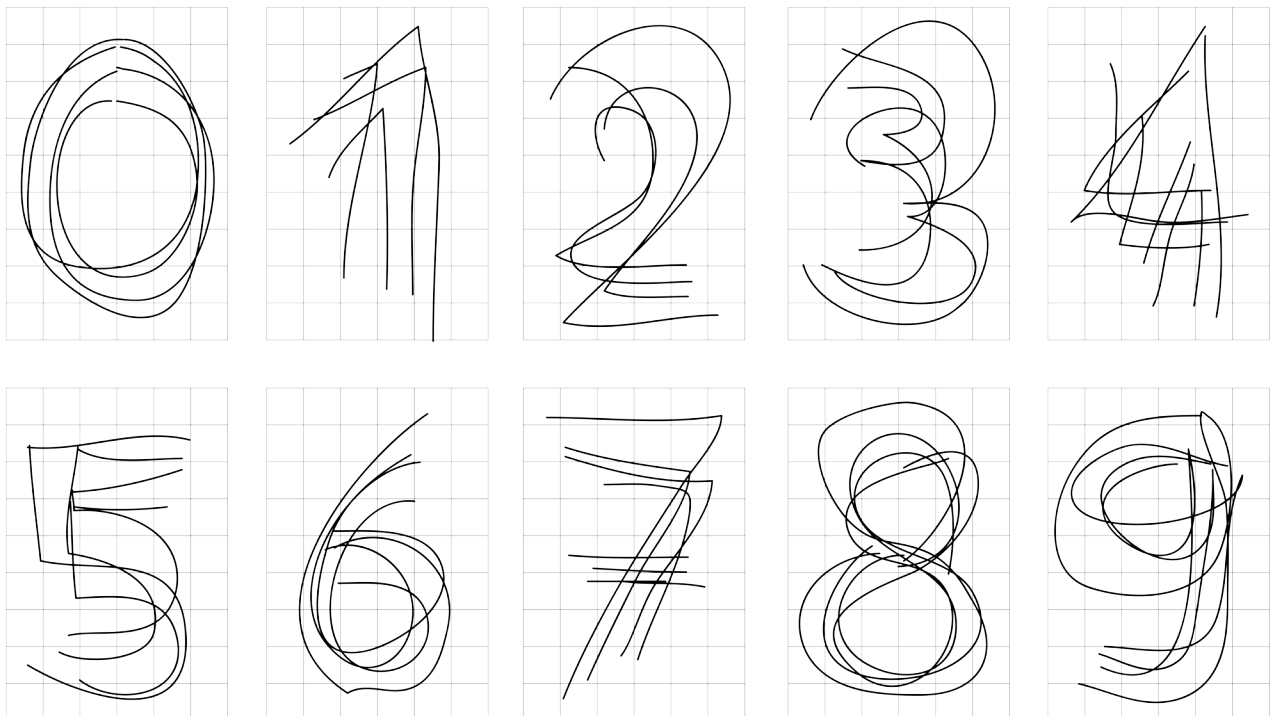


figure 2

Results

When you run the program, you can see that the vectors of some numbers look alike (figure 3). Number 1, 4 and 7 look the same according to the program. Number 5, 6 and 9 look similar as well. Number 2 and 8 are the only vectors that differ from the rest. To prevent this from happening the matrices should be more detailed. Numbers that are drawn to one edge of the canvas result in vectors that are similar. Almost all the vectors start with one or two zeros, because the left columns of the matrices are empty.

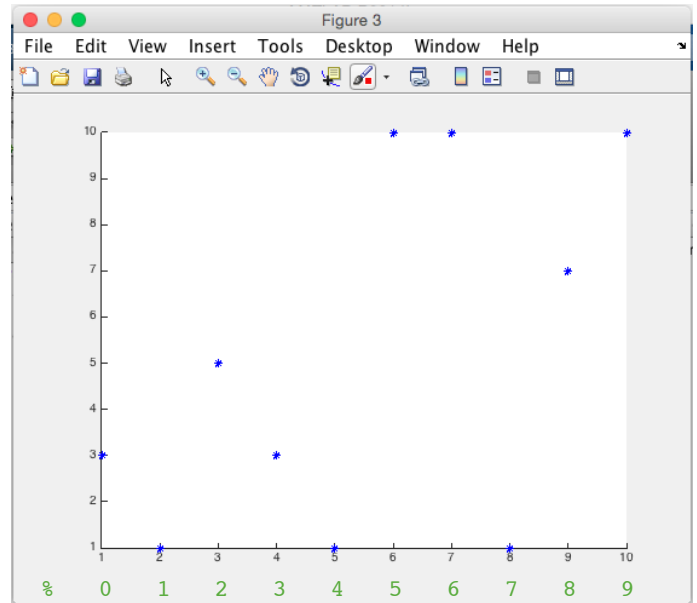


figure 3

Environment recognition with audio samples

The recordings are made on the street. One recording on the bike, one on a crossing before a red traffic light and one on the same crossing walking through a green light.

I used the wavread-function to convert audio to usable data:

```
x = wavread('20141217_104449green.wav')
```

Unfortunately the bike-recording got corrupted somehow. Matlab refused to convert this file. The converted data for the traffic light samples look like this:

```
0.0525
 0.0282
 0.0085
-0.0026
-0.0174
-0.0196
-0.0037
-0.0036
-0.0203
-0.0240
-0.0175
-0.0132
-0.0102
-0.0079
 0.0003
 0.0162
```

Of course when you use this data in the neural gas algorithm, the algorithm will recognize each number as one vector, because there is one in a row. To prevent this from happening I tried to create horizontal vectors. Like this:

```
-0.0742  -0.0715  -0.0687  -0.0644  -0.0553  -0.0493  -0.0475  -0.0562  -0.0718
```

```
Columns 902404 through 902412
```

```
-0.0724  -0.0631  -0.0624  -0.0623  -0.0585  -0.0562  -0.0511  -0.0436  -0.0360
```

```
Columns 902413 through 902421
```

```
-0.0285  -0.0256  -0.0224  -0.0196  -0.0276  -0.0274  -0.0063  0.0042  -0.0038
```

```
Columns 902422 through 902430
```

Matlab refused to run the vectors in this way unfortunately. It was a very instructive struggle after all. I learned a lot from this great website: <http://www.cyclismo.org/tutorial/matlab/vector.html>

Results

The plot shows difference between the left and right side. This could mean that there is a difference between the audio samples. In the TestSet I used the data from the green traffic light for the first hundred rows and for the second hundred rows I used data from the red traffic light. Some of the points are cluttered, which could indicate corresponding pieces of data in the TrainingSet.

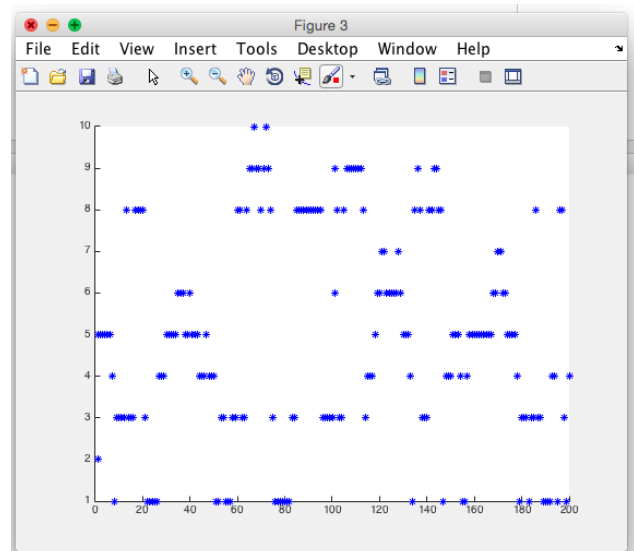


figure 4

Reflection

This assignment is different from the other two I did this semester. As a premaster I entered this assignment without any programming experience. Except for a little bit of exercise during the sdl-weeks, in Arduino, I am completely new with this, so the lectures were quite a shock. The drawn number recognition was a fun way to get acquainted with Matlab. Looking back I would have liked more explanation about the code behind the Neural Gas algorithm.

Bit by bit I unraveled the code of the Neural Gas algorithm, with the help of the reference pages from Matlab. I looked up almost every term and slowly I started to understand small parts of the algorithm. It took me ages to find out how to convert column vectors into row vectors and still I did not succeed completely. But I do have a understanding of what you can do with these algorithms and how they work.