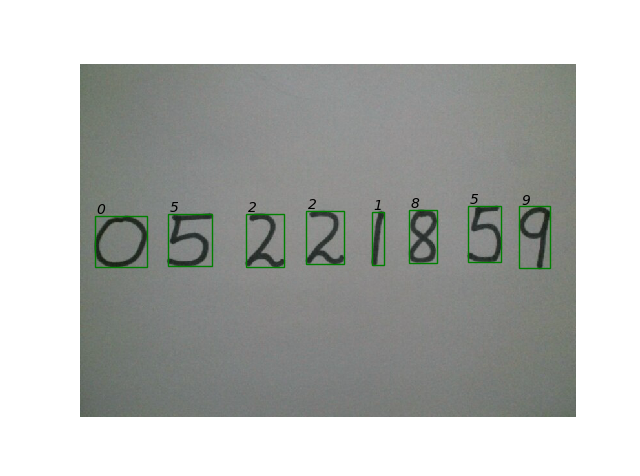
z

Digit recognizer

using knn classification

Project 3 | Datamining | 20/12/2017

**Problem:**

you have one image contains set of handwritten digits need to extract every digit from the image and get the right number of the digit

**solution:**

our solution is divided into three main stages the first one is about the preprocessing in the input image and how to extract every digit in it and the second stage is about building the machine learning model (KNN) and train it to be ready for using the third stage is about testing our model to get the result of our solution.

1. **Preprcessing :**

* Read image using opencv
* Converting it to gray scale
* Then applying global threshold to get digit in black and background in white
* Negative the image to make it the same as data set
* Using skimage.morphology to get ride of noise objects which are small connected pixels
* Using skimage.label to find digits in the image and label it with unique number
* Loop over image to cut the digit from the image and appending then to list
* For every extracted digit find the HOG descriptor with length(36) which will used for measuring the distance

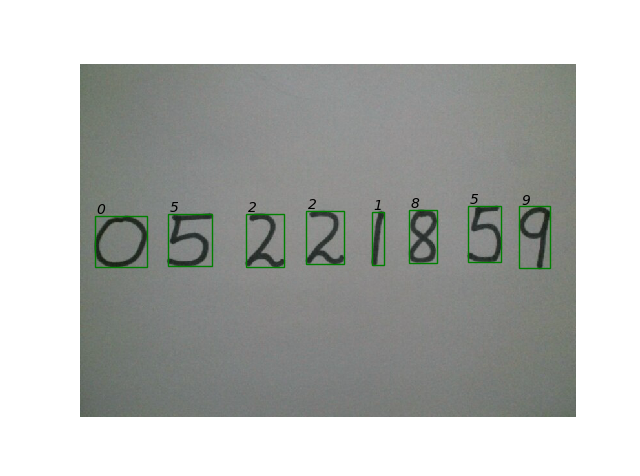
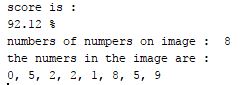
**2-training :**

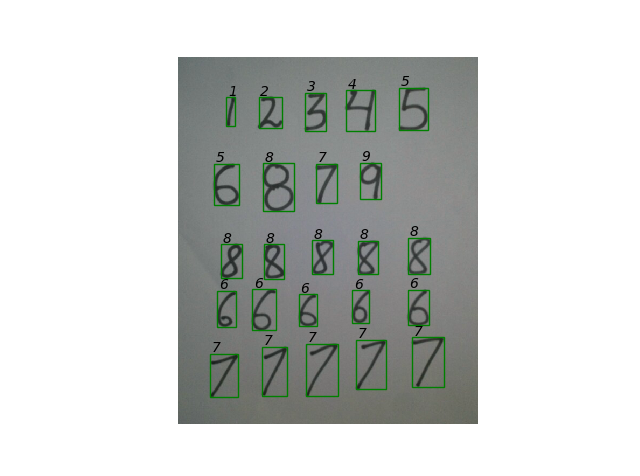
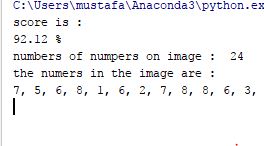
* Load our data set (“mnist-original”)
* Using skilearn to split the data set into training part and testing part
* Lope over every image in the train part to find the HOG descriptor with length (36)
* Use the skilearn.KNeighborsClassifier() to find to build our model and training it with the splited train part
* Save the model on the disk for future use in testing and trying

**3-testing our model :**

* Now load the model form the disk to test the model and find it’s accuracy (92.78%)
* Using new images to test the model

After challenging work the results was great





---------------------------------------------------------------------------------

**Dataset (mnist-original):**

We are not going to create a new database, but we will use the popular “**MNIST database of handwritten digits”**

The data files “mnist-original.mat” contain two columns

**1-data** (784\*700000)

which have 7000 gray-scale images of hand-drawn digits, from zero through nine.

Each image is 28 pixels in height and 28 pixels in width, for a total of 784 pixels in total. Each pixel has a single pixel-value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255, inclusive.

**2-label** (1\*70000)

Have the right class for every image in the first column

**The actual samples for each digit was** :

| **Digits** | **Number of samples** |
| --- | --- |
| 0 | 6903 |
| 1 | 7877 |
| 2 | 6990 |
| 3 | 7141 |
| 4 | 6824 |
| 5 | 6313 |
| 6 | 6876 |
| 7 | 7293 |
| 8 | 6825 |
| 9 | 6958 |

**Problem with data set:**

We first was using dataset from skimage.digit but images in this dataset was small (8,8) it causes problem while calculating HOG descriptor so we turned to use “mnist-original” dataset which have images with size (28,28)

It give us free when calculating HOG descriptor with length of 36