CS771A : Assignment 2

Group Name:
E-bot

Name	Roll Number
Aditya Kushwaha	200051
Aman Pal	200102
Amit Kumar	200109
Amit Madhesiya	200111
Kaushal Kishor Shukla	200497
Sameer Khan	200853

Submitted to:
Prof Purshottam Kar

1 Task- 1

1.1 Identifying Foreground and Background pixels

In the given dataset of the images we have to differentiate between the background and the character inside the images. As mentioned in the problem statement background pixels has a light shade whereas the obfuscating lines has a darker shade and the boarders of the characters are even darker.

To differentiate the character and the background of the image on the basis of the brightness, we used HSV color format. To know the average values of the background pixels, we took last column of the image as it mostly contains background pixels. Now we have the avg value of brightness(v) of the background pixels. We used it to create a mask on character outlines. We selected the pixels which have brightness value less than 0.9*v. We are taking 0.9 times v to make sure that all background segments will be excluded.

We are considering the fillings of character outlines also as background. Now, we applied the mask on the image, which gives us only character outlines and obfuscating lines.

1.2 Dealing with obfuscating lines

Removing the obfuscating lines becomes easy using a image processing technique called **Erosion**. We simply converted the masked image into gray scale. Then using a small threshold value converted it to binary images. Then applied the erosion technique using a kernel size of (3X3). Larger kernel size might erase character outline itself.

1.3 Segmenting image into pieces

As mentioned in the problem statement we are only interested in the last character of the image. First we took the last section (350X500) of the whole image. Than we trimmed the image into pieces by looking for vertical columns of pixels that contain very few non-background pixels.

1.4 Learning ML model

1.4.1 Model used

The model used is a logistic regression model, a linear classification algorithm. Here we have to classes ’EVEN’ and ’ODD’. We tried a bunch of models like KNN, SVM, Neural Networks etc. all models performed perfectly while the prediction (accuracy/ parity match score = 1.0). We choose logistic regression because it is simple, fast and taking very less space to store the trained model.

1.4.2 Pre-processing of the data

We have data as images of size 500X100 pixels. We already trimmed the images such that it only contains last digit (minimum blank space) of the hexadecimal number. The final size of the segmented image comes out 80X80, i.e. we have 6400 features for a ML model.

Now, the dimensions of the our dataset is 3 (2000X80X80, here we are using binary image which has only one channel). But most of ML models works on datasets with less than or equal to 2 dimensions. So we flatten the each image into linear array. Now, our dataset is size of 2000X6400, 2 dimensions.

1.4.3 Training algorithm and hyper-parameters

We used default training algorithm and hyper-parameters from the sk-learn library for the logistic regression model.

1.4.4 validation procedure

We performed a train-test split of the data into training and testing sets, 30% of the data was used for testing, while the remaining 70% was used for training. We also shuffled the data before splitting as it is a common practice to ensure randomness in the data split.