|  |  |
| --- | --- |
| Cairo University  Faculty of Engineering  Computer Engineering Department | CMPN450  Fall 2018 |

**Pattern Recognition and Neural Networks**

**Project Report**

Team members:

Bahi Ali  
Berlnty Kerlos  
Gehad Mohsen  
Yasmine Alaa

1. **Project Pipeline:**

The pipeline starting with reading the images then preprocessing the image where noise and all typed and extra white spaces are removed, and the image is being divided into smaller blocks, then the feature extraction that applies the LBP(Local Binary Pattern) feature then the feature vector is sent to Model Selection that uses KNN, finally the output is the predicted class number from KNN.

1. **Preprocessing Module:**This module takes the image, apply Gaussian filter, convert it to grayscale image, invert the color then dilate for two iteration to make the lines border and easier to be detected, then we have two trials:

* Successful trial:

Firstly, search for contours in the image, it will find lines, sentences, blocks and even some noise; so, to detect only the three separating lines, we sort the contours according to their distance from the from the x-axis and check for this condition

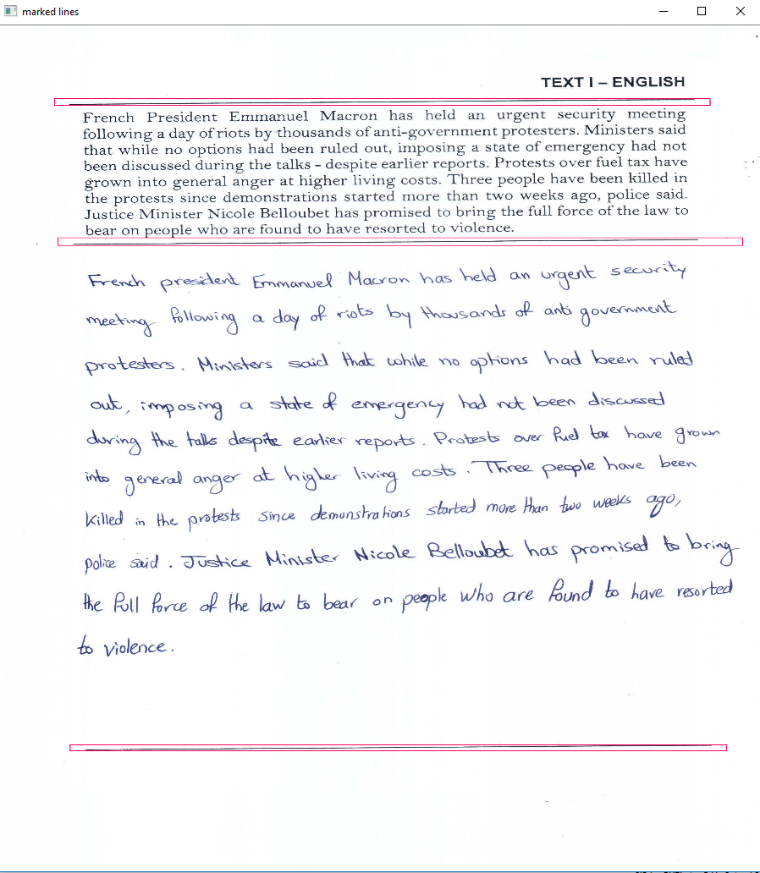
((contour\_height <= 0.02 \* image\_height) **and** (contour\_width > 0.6 \* image\_width)**and** (contour\_width>= 50\*contour\_height))

Then to remove noise like small components of some dark parts in the image we use other condition

((contour\_width >0.02\* image\_width) **and**

(contour\_height>0.02\* image\_height) **and**

(contour\_width > contour\_height))

Then crop the image according to new height and width  
the new height is the distance starting from the contour of the sentence following the second separating line and the contour of sentence preceding the third separating line.  
and of course other scenarios are handled like not being able to detect all the three lines.  
and also the new width of the image is the distance between the minimum x-axis point value of the sentences’ contours and the maximum (x-axis point value+ contour width). Then the cropped image is returned to be used by the next module.

* Another successful but eliminated trial:

Applying Hough lines after dilation the eliminating lines of width less than 40% of image width then searching for connected components  
that will extract the three separating lines clearly, BUT its **time costly**   
then cropping the image according to coordinates of the lines.

According to the paper we followed to implement our solution, it suggested a certain method for extracting a texture for the writer out of its handwriting by compressing the lines and words and reducing all white spaces between any connected components. We applied that part, but it resulted in low accuracy, so it was discarded later.

1. **Feature Extraction/Selection Module:**

* Successful trial:

We implemented LBP (Local Binary Pattern) feature that was explained in the document, we used LBP feature with 8 points to be considered as neighbors and radius =1

Then we use built in LBP with different methods and radius and no of points then gets the histogram of the result

* *First trial:*

Method=”default” , No\_Points=8 , radius=1

We used KNN classifier, k=1

Accuracy: 75%

Second trial:

Method=”default” , No\_Points=8 , radius=1

SVM Classifier with gamma=0.001

Accuracy=73%

* *Trials with KNN classifier:*

knn=3 points=8 rad=1 m='defualt' acc=81.81%

knn=3 points =8 rad=1 m='nri\_uniform' acc=88.6%

knn=3 points =8 rad=3 m='nri\_uniform' acc=86%

knn=3 points =16 rad=1 m='nri\_uniform' acc=87.5%

knn=3 points =8 rad=1 m='uniform' acc=87.5%

knn=3 points =16 rad=1 m='uniform' acc=87.3%

* *Last Trial:*

We make 1 iteration of erosion with 5x100 ones kernel

Then LBP with method=”uniform” radius=3 No\_Points=24

With KNN classifier ,k=1

Accuracy=99% on AIM

* unsuccessful trial:

We worked on the ‘A Set of Novel Features for Writer Identiﬁcation’ paper that introduced some features,

the group we use in this trial were extracted (out of each sentence) from the connected Components features which are the average distance between two successive bounding boxes of each connected component and the average distance of two consecutive words and the average , median , standard deviation of the width of connected component in each line , We also analyzed the closed loops occurring in handwritten text and calculate the average size of the blobs in each line and add it to the feature vector of the line

* *Results:*

We test this on dataset from IAM on three writers for each iteration with 100 iterations, but the accuracy was less than the accuracy of the final successful trial, but the time was less than the final successful trial

1. **Model Selection and Training Modules:**

We tried SVC, Linear SVC and KNN classifiers but KNN came with the best accuracy regarding to our features.

1. **Enhancements and Future work:**

First, we tried to crop image and detect the writing area using Hough lines which is ready made functions but its time costly regarding to our dataset images, then used the contours and some logic which reduced the consumed time greatly.

Try to combine the implemented but not used features with the LBP to get better accuracy.

1. **Workload distribution**

|  |  |
| --- | --- |
| Bahi Ali | BLP implementation  Fractal Features of ‘A Set of Novel Features for Writer Identiﬁcation’ paper |
| Berlnty Kerlos | Preprocessing (dividing image to texture blocks)  Cropping written part of image |
| Gehad Mohsen | Connected components features of ‘A Set of Novel Features for Writer Identiﬁcation’ paper |
| Yasmine Alaa | BLP implementation  Lower and upper contour feature of ‘A Set of Novel Features for Writer Identiﬁcation’ paper |