Pattern Recognition

Writer Identification

Ayman Shawky
Sec: 1, BN: 11
aymanazzam63@gmail.com

Ayman Abdelnaby
Sec: 1, BN: 13
aymania08@gmail.com

Ayman Adel
Sec: 1, BN: 12
ayman.elakwah@gmail.com

Khaled Nassar
Sec: 1, BN: 17
khaled.nassar99@eng-st.cu.edu.eg

January 19, 2021

Contents

1 Introduction

Writer identification problem is the problem of assigning a known writer to an unknown text script from a group of possible writers. This is useful in many areas of interest such as forensics, security, historical documents analysis and many more. Writer identification has seen a plethora of research and attention from researchers in recent years, however, it still remains a difficult problem to solve due to variations in writing style and conditions, and the problems arising from variations in image quality due to document degradation and other incidental factors.

2 Proposed Method

The pipeline, shown in Figure 1, shows the main modules that the data goes through before making a decision: Preprocessing, Feature Extraction and Classification.
First, each image is processed to retrieve the handwritten paragraph only to reduce the overall computational complexity and remove other irrelevant information from the image so that only the handwritten part affects the decision making process.
In feature extraction module, the implementation was based on the methodology in [1] as their feature vector was based on Local Binary Pattern (LBP) which is used in many computer vision problems due to its robustness against noise. [1] introduced their own variation of LBP, Sparse Radial Sampling Local Binary Pattern (SRS-LBP), which allows for sampling of the circular patterns of traditional LBP at a lower computational cost. In addition, the comparison operator of the traditional LBP is changed to a threshold that is derived statistically from each image. Then the feature vector is formed as the histogram of each SRS-LBP at different radii, concatenated together after removing the 0 pattern which corresponds to foreground and background only patterns, thus providing no information. Then the features are normalized using L2 normalization before applying a Nearest Neighbour classification to produce the resulting label.

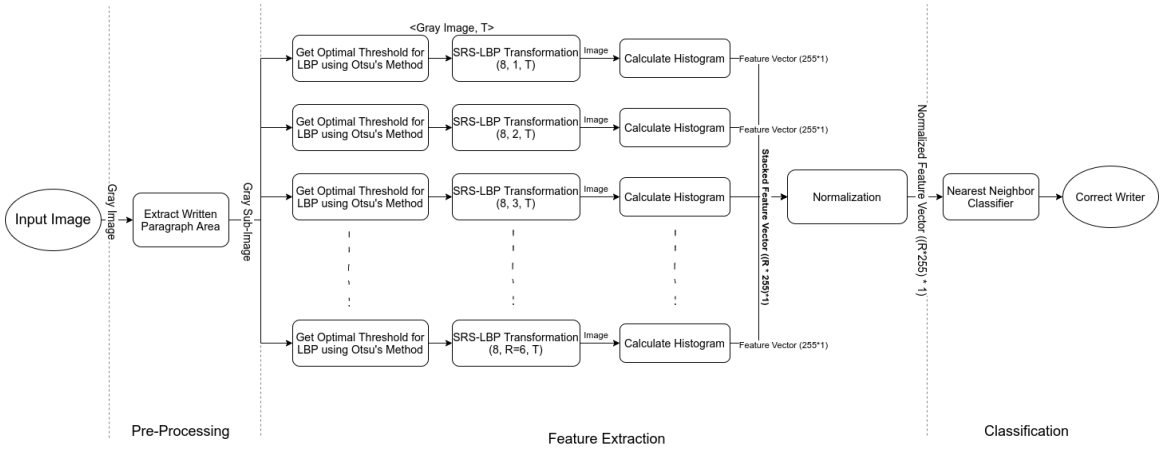

Figure 1: Pipeline

2.1 Preprocessor

The preprocessor module consists of 3 small modules. the first module to remove the unneeded parts, the second module to crop the image on the paragraph part only and the third module to get better accuracy. For example if the input image as Figure 2, after the first module the output will be as Figure 3 and after the second module the output will be as Figure 4.

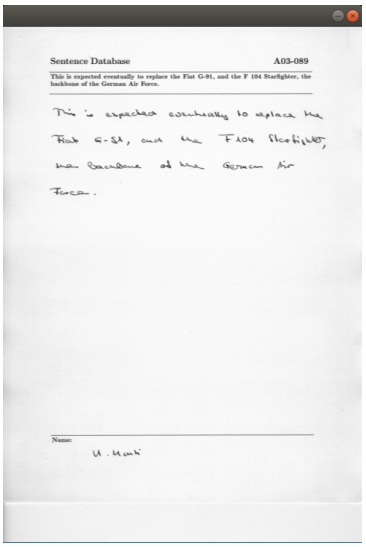

Figure 2: Input Image

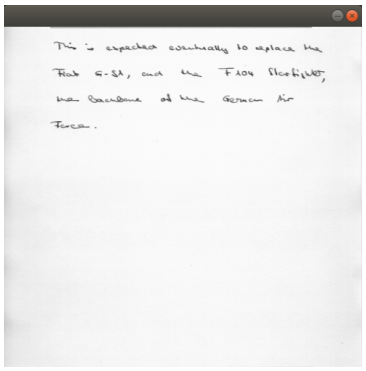

Figure 3: Image between the two lines

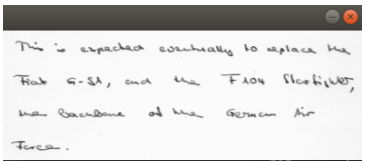

Figure 4: Paragraph Image

The pipeline for the first module as Figure 5, we make binarization then get the contours and finally we crop the part between the first line above the center of image and the first line below the center of image. The pipeline for the second module as Figure 6, we remove the noise using a Gaussian filter then get the binary image and contours as the previous module but this time will be on smaller part of image and finally we crop the bounding box around the contours.

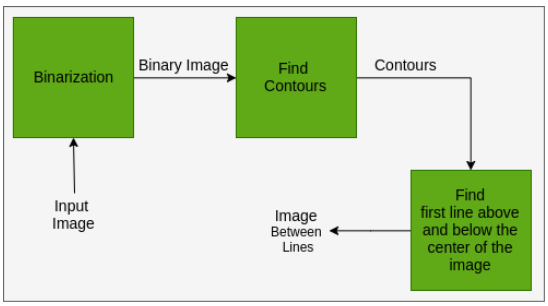

Figure 5: The First Module

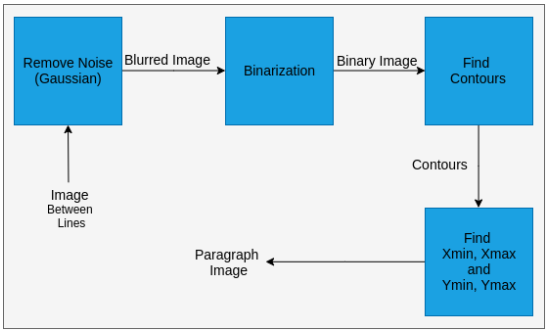

Figure 6: The Second Module

2.2 Feature Extraction

This module is responsible for extracting discriminating features from the preprocessed input and forward it to the classifier to train and classify test images properly. Formally, the input is a gray image for the written text paragraph and the output should be a feature vector.

We chose to follow a texture based approach which can take the whole paragraph as an input and extract its features. One of the most common text descriptors is the Local Binary Pattern, the authors in [1] proposed a variation of it which is the Sparse Radial Sampling LBP. We followed their method and it proved to obtain better features.

The extraction pipeline is divided into two steps. First, LBP-transform maps each pixel in the input image to another integer representing the relation between it and the adjacent pixels.
Then, pooling of LBP into a histogram representation of an image.

2.2.1 SRS-LBP transformation

LBP encapsulates the local geometry at each pixel by encoding binarized differences with pixels of its local neighbourhood. We can define LBP of a pixel as follows:

$$LBP_{P,R,t} = \sum_{p=0}^{P-1} s_t(g_p - g_c) * 2^p \tag{1}$$

where g_c is the gray level of the central pixel, g_p is the $p-th$ neighboring pixel in a circle with radius R and s_t is a binarization function with a parameter t. So, it can be defined as:

$$s(x) = \begin{cases} 1, & x >= t \\ 0, & x < t \end{cases} \tag{2}$$

Normal LBP considers t equal to 0, while the proposed method cares for adjacent pixels that differs from the central pixel with a certain threshold.
This threshold is obtained using Otsu's method on the absolute difference images for each neighbor. A large image with size (P * size of original image) is formed and Otsu's method is applied on it to obtain this threshold. This effectively separates the significant differences from insignificant ones.

The LBP transform is applied to the input image several times with different radii and P = 8 neighbors as shown in Figure 7, and the output images is forward to the pooling module.

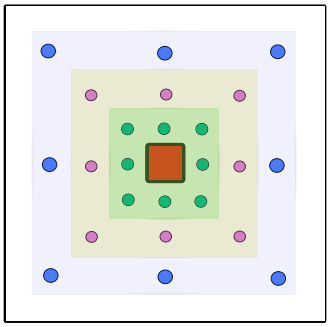

Figure 7: Neighboring Pixels (P = 8, R=3)

2.2.2 SRS-LBP Pooling

Multiple LBP images with different radii are fed to this module and it should output a normalized feature vector representing the sample image.

A histogram is computed for each of the SRS-LBP images. The zero component is discarded as it represents foreground or background patterns. Then all these histograms are stacked together into one feature vector of size ((R * 255) * 1), then the Hellinger Kernel is applied on this vector for normalization.
This normalized feature vector is then passed to the classifier for training or classification.

2.3 Classifier

We used nearest neighbour classifier. The data is composed of only three writers. Each one of them has only 2 pages. Based on our approach that each page represents a pattern. The number of training patterns is small. Hence, nearest neighbour classifier is an acceptable choice. The 2 patterns of each writer define everything about him/her. Thus, we are not afraid of outliers (i.e. due to the small number of training patterns, we have no other choice than to assume the data is correct). The test pattern is classified the same as the nearest pattern.

2.4 Performance Analysis

We analyzed the performance of our algorithm based on IAM dataset. We worked with complete form text only. After excluding the writers who have less than two pages, there were 301 writers. We then loop until we reach the required number of test cases. Each iteration, three different random writers are chosen from the set. For each writer, two random pages are chosen for training and the others are used for testing. Then, we go through the normal pipeline. After classification, we count the number of correct predictions. We also calculate the average execution time of preprocessing, feature extraction, training and classification. Misclassified cases are printed out so that we can tune the hyperparameters based on these cases. There is also an option to show the related images of the misclassified case once it is discovered. Finally, we calculate the accuracy percentage by dividing the number of correct predictions by the total number of test cases.

3 Previous Trials

Some of the previous trials that lead to the current pipeline are shown in Table 1
Note that not all previous trials were recorded, but those that weren't still offered some much needed information nonetheless.

LBP: We tried different LBP techniques SRS, the original LBP implemented by us (Normal), the original LBP implemented by skimage (Original).

Histogram Normalization: The normalization technique used after constructing the histogram (L1 - L2 - None).

PCA: PCA Explained variance (None means PCA was not used).

Features Normalization: The normalization technique used after PCA module (L1 - L2 - None).

Classifier: The type of classifier used.

#test cases: Number of test cases.

Seed: The seed value used by the random generator that is used to sample writers from the dataset.

Table 1: Previous Trials

Accuracy	LBP	Histogram norm	PCA	Features norm	Calssifier	# test cases	seed
75	SRS (1→3)	L1	0.99	L2	Nearest Neighbour	100	0
76	Original	L1	0.99	L2	Nearest Neighbour	100	
86	Original	L2	0.99999	None	SVM	100	20
87	Original	L1	0.99	None	Nearest Neighbour		
88	Original	L2	0.99999	None	SVM	100	20
90.6	Normal (3)	L1	0.99999	None	SVM	100	0
91	Original	L2	None	None	SVM	100	20
91.5	Normal (1→5)	L1	0.99999	None	SVM	100	0
91.5	SRS	L1	0.99999	None	SVM	100	0
91.5	Original	None	0.99999	None	SVM	100	0
92.8	Original	L2	None	None	SVM	1000	20
93	SRS (1→10)	L2	None	None	SVM	100	20
93.4	Original	L1	0.99999	None	SVM	100	0
94	SRS (1→7)	L2	None	None	SVM	100	20
95	SRS	L2	None	None	SVM	100	20
95.1	Original	L2	0.99999	None	SVM	100	10
95.3	Original	L2	0.99999	None	SVM	100	0
95.3	Original	L2	0.99999	L1	SVM	100	10
98.1	Original	L2	0.99999	None	SVM	100	0
98.1	Original	L2	None	None	SVM	100	0
99.1	SRS	L2	None	None	SVM	100	0
99.4	SRS (1→6)	None	None	Hellinger	Nearest Neighbour	2001	0
99.85	SRS (1→6)	None	None	Hellinger	Minimum Distance	2011	30
99.95	SRS (1→6)	None	None	Hellinger	Nearest Neighbour	2011	30
100	SRS (1→6)	None	None	Hellinger	Nearest Neighbour	1000	20

It can be proven that using the Hellinger kernel offered a huge improvement in performance. This may not be visible from the table but that is due to using a lower number of test cases in the trials where it wasn't used while using a large number of tests in the ones where it was.

4 Future Work

Although a high accuracy was reached in some of the runs/trials, but there will always be some room to improve.
For example, incorporating the extracted text in the algorithm to give some context to the features would be better than classifying according to the features without any prior knowledge. In other words, knowing that a certain feature (or more) corresponds to a certain letter could offer information that is not available in the current system and help avoid misclassifications.

References

[1] A. Nicolaou, A. D. Bagdanov, M. Liwicki, and D. Karatzas, "Sparse radial sampling lbp for writer identification," ICDAR, 2015