Multi-script Writer Identification using Dissimilarity

Diego Bertolini[1], Luiz S. Oliveira[2], Robert Sabourin[3]

[1]*Federal University of Technology - Parana (UTFPR), Campo Mourão, PR, Brazil*
[2]*Federal University of Parana (UFPR, DInf), Curitiba, PR, Brazil*
[3]*Ecole de Technologie Superieure (ETS), Montreal, QC, Canada*

***Abstract*—Multi-script writer identification consists in identifying a person of a given text written in one script from the samples of the same person written in another script. The rationale behind this is that the writing style of an individual remains constant across different scripts. While this hypothesis may hold, recent results on a multi-script writer identification competition show that classical writer-dependent classifiers fail in this task. In this work we investigate the efficacy of a writer-independent classifier based on dissimilarity for multi-script writer identification. The classifiers were trained using two different texture descriptors (LBP and LPQ). Our experiments on 475 writers of the QUWI dataset, which is composed of Arabic and English samples, show that the proposed strategy surpasses the results published in the literature by a large margin, achieving error rates similar to single-script writer identification systems.**

***Index Terms*—Writer Identification; Texture; Dissimilarity;**

I. Introduction

Writer identification is the task of determining the author of a sample handwriting from a set of writers. This problem has attracted a great deal of attention due to the large number of potential applications, such as personalized handwriting recognition, automatic forensic document examination, writer retrieval, and classification of ancient manuscripts.

In this context, a large number of scripts have been considered in the literature including Latin [1], [2], [3], [4], Arabic [5], Chinese [6], Japanese [7], Bengali [8], and Oriya [9]. All those works have in common the fact that they deal with single-script handwritten texts. More recently, the research on writer identification has been extended to a multi-script environment by studying and validating the idea of recognizing an individual of a given text written in one script from the samples of the same individual written in another script. The main hypothesis is that the writing style of and individual remains constant across different scripts. Djeddi et al. [10] presented encouraging results on a database composed of 126 writers with four samples per writer (two in English and two in greek). In ICDAR 2015, a competition on multi-script writer identification was proposed on the QUWI database [11], which contains 1,017 writers with four samples per writer (two in Arabic and two in English).

In this competition, 300 writers from QUWI dataset were used for two different tasks [12]. In the first one, the training data contained only Arabic samples while the testing set was composed of English samples. The second task, training and testing were composed of English and Arabic samples, respectively. Surprisingly, the results reported by the organizers were quite poor and not comparable to the ones presented in [10], despite of the different scripts. The competitors enrolled in this challenge have used different kinds of features and classifiers. The common characteristic was that all systems used a writer-dependent (WD) strategy, i.e., a classifier trained to recognize k classes, where k is the number of writers. Besides being used in most biometric systems, the performance of the WD approach drops as the number of writer increases and with the limited number of references per writer.

An alternative to the WD approach is the Writer-independent (WI), which tries to alleviate the aforementioned difficulties by converting the k-class pattern recognition problem into a 2-class problem through a dichotomy transformation. Therefore, it uses a single 2-class classifier to match each input questioned piece of handwriting to one or more references. Another advantage of this approach is that even a writer that did not contribute for the training set can be identified by the system. This is possible because the WI model is trained to predict whether the questioned handwritten and the references were written by the same person or not.

In this work we argue that one may take advantage of this characteristic of the WI approach for multi-script writer identification. The rationale behind is that if the WI model is built to predict whether or not two pieces of handwriting were written by the same person, the model should be able to do that independently of the script that has been used. Of course, this hypothesis holds if the representation is similar for different scripts. A representation that looks similar for different scripts is the texture, which has been successfully applied to writer identification on different western-script datasets, such as IAM [13], BFL [2], and Firemaker [14].

To validate these ideas we perform a series of experiments on an extended version of the QUWI database used for ICDAR 2015 competition, which contains 475 writers, using two different textural descriptors, the Local Binary Pattern and Local Phase Quantization. Our experimental results show that the proposed approach surpasses the results reported in the 2015 ICDAR competition by a large margin, achieving a performance comparable to single-script writer identification systems.

II. The Dissimilarity Framework

The main attractive of the dissimilarity approach is the possibility of reducing any insurmountable pattern recognition problem to a 2-class problem. It works by extracting the feature vectors from both questioned and reference samples and then computing what we call the dissimilarity feature vectors. In ideal conditions, it is expected that if both samples come from the same writer (genuine), then all the components of such a vector should be close to 0, otherwise, the components should be far from 0.

Given a queried handwritten document and a reference handwritten document, the goal consists in determining whether or not the two documents were produced by the same writer. Let V and Q be two vectors in the feature space, labeled l_V and l_Q respectively. Let Z be the dissimilarity feature vector resulting from the dichotomy transformation $Z = |V - Q|$, where $|\cdot|$ is the absolute value. This dissimilarity feature vector has the same dimensionality as V and Q.

In the dissimilarity space, there are two classes that are independent of the number of writers: the within class $(+)$ and the between class $(-)$. The dissimilarity vector Z is assigned the label l_Z,

$$l_Z = \begin{cases} + & \text{if } l_V = l_Q, \\ - & \text{otherwise} \end{cases} \quad (1)$$

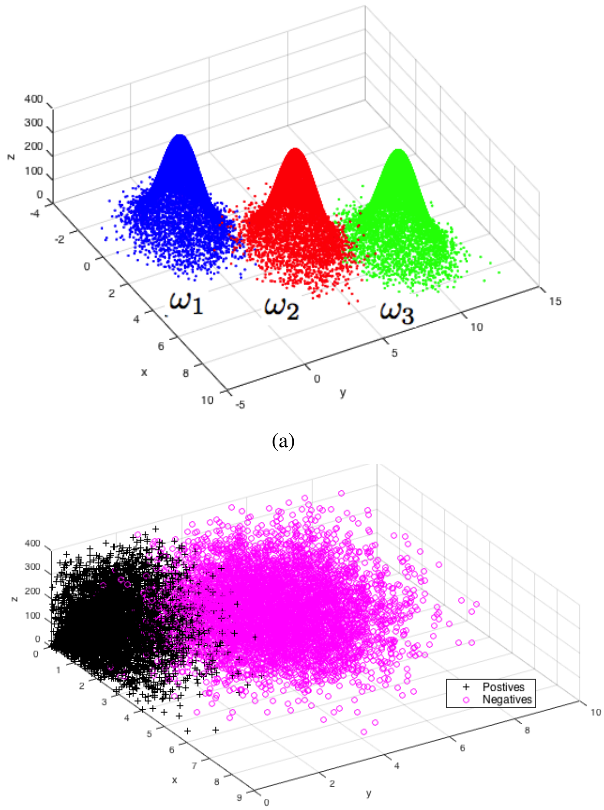
Figure 1 exemplifies such transformation, where (a) shows the data in the feature space from three different writers $\{\omega_1, \omega_2, \omega_3\}$ and (b) depicts dissimilarity feature vectors, which are the results of the dichotomy transformation between the features of each pair of samples to form vectors.

One may observe from Figure 1 that the dichotomy transformation impacts the geometry of the distribution. In the feature space, multiple boundaries are necessary to discriminate the three classes. In the dissimilarity space, on the other hand, only one boundary is necessary, since the problem is reduced to a 2-class classification problem.

The number of samples in the dissimilarity space is larger, because these samples are made up of every pair of feature vectors. We can also see in Figure 1 that, if both samples come from the same writer (genuine), then all the components of such a vector should be close to 0, otherwise they come from different writers (a forgery), in which case the components should be far from 0. This is true under favorable conditions. However, as in any other feature representation, the dissimilarity feature vector can be affected by intra-writer variability. This variability could generate values that are far from zero, even when the dissimilarity between the samples produced by the same writer is measured.

As mentioned earlier, one advantage of this approach is that even writers whose specimens were not used for training can be identified by the system. This characteristic is quite attractive, since it obviates the need to train a new model every time a new writer is introduced. In our experiments, we emphasize this feature by using disjoint sets of writers for training and testing.

The dissimilarity framework requires the classifiers to discriminate between genuine (positive) and forgeries (negative). To generate the positive samples to train the SVM classifier, we computed the dissimilarity vectors among the R genuine



(a)

(b)

Figure 1. Dichotomy transformation: (a) samples in the feature space (b) samples in the dissimilarity space where (+) stands for the vectors associated to the within class and (-) stands for the vectors associated to the between class.

samples (references) of each writer which resulted in $\binom{R}{2}$ different combinations. The same number of negative samples is generated by computing the dissimilarity between one reference of one writer against one reference of other writers picked at random. Following the findings of our recent study [2], the best results were found using 9 references per writer.

III. Database

In the last few years different databases devoted for writer identification have been published in the literature [13], [15], [16]. However few of them can be used in the context of multi-script writer identification. In this work we have adopted the QUWI database [11], which contains 4,068 handwritten text images from 1,017 different writers. Part of this dataset (300 writers) was used in the ICDAR 2015 Competition on Multi-script Writer Identification and Gender Classification [12].

In order to acquire the handwritten samples, volunteers were instructed to produce four pages of handwriting as follows: The first one contains approximately six handwritten lines in the Arabic language of free-text. The second page contains an Arabic text of three paragraphs to be copied by all the writers. Similarly, the third page contains about six handwritten lines in English of free-text. The first and the third pages can be used for text-independent writer identification tasks, whereas the second and fourth page can to be used for text-dependent writer identification tasks. Figure 2 shows some samples of the Arabic and English handwriting extracted from QUWI database.

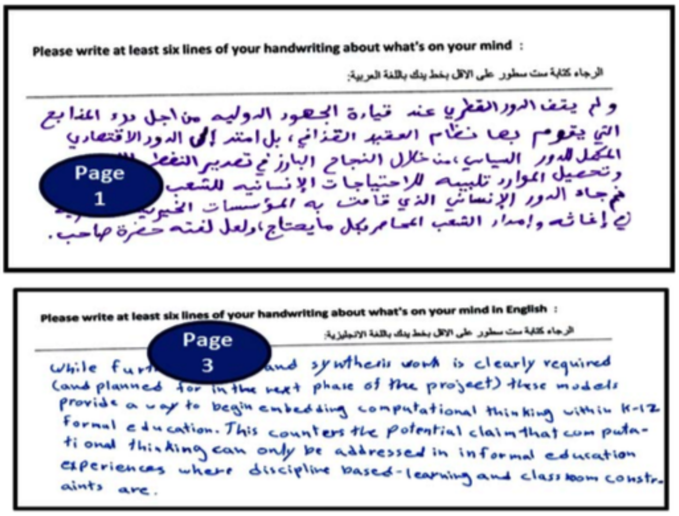


Figure 2. Arabic (page 1) and English (page 3) handwriting samples from the QUWI database (extracted from [12])

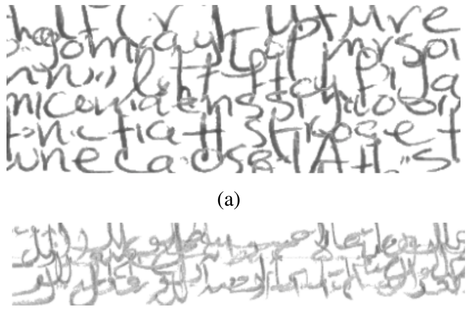
In our experiments we have used four letters of the 475 writers that have been made available to us[1].

IV. Representation

In order to generate the texture, the document is binarized and scanned top-down and left-right to detect all the connected components of the image. Small components, such as periods, commas, strokes, and noise, are discarded. The bounding box of the remaining components is then used to extract the original components of the gray level image. The components in gray levels are then aligned with the new image using the center of mass of the bounding box. This algorithm, described in details in [2], compacts the handwriting generating texture images. Then, the texture is segmented into nine 256×256 blocks. Figure 3 shows two examples of the handwriting texture produced from English and Arabic handwritings for the same writer.

After creating the textures, the next step consists in dealing with representation. The literature presents a considerable number of texture descriptors that have been successfully applied on different domains of problems, such as, Grey-Level Co-occurrence Matrix (GLCM) [17], Gabor Filters [18], Threshold Adjacency Statistics [19], Local Binary Patterns (LBP) [20] (and their variants, e.g., CLPB [21] and RLPB [22]), and Local Phase Quantization (LPQ) [23]. In our previous experiments on writer identification, the best results were always achieved using LBP and LPQ, therefore, in this study we have used only these two descriptors. To make this paper self-contained, in the next two subsection we present some details about LBP and LPQ.

[1]The authors would like to thank QUWI team for making the data available.



(a)

(b)

Figure 3. Example of the texture produced from (a) English and (b) Arabic handwritings for the same writer.

A. Local Binary Patterns

The LBP labels the pixels of an image by thresholding a 3×3 neighborhood of each pixel with the center value. Then, considering the results as a binary number and the 256-bin histogram of the LBP labels computed over a region, they used this LBP as a texture descriptor. The LPB operator $\text{LBP}_{P,R}$ produces 2^P different binary patterns that can be formed by the P pixels in the neighbor set on a circle of radius R. However, certain bins contain more information than others, and so, it is possible to use only a subset of the 2^P LBPs. Those fundamental patterns are known as uniform patterns. A LBP is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice-versa when the binary string is considered circular. For example, 00000000, 001110000 and 11100001 are uniform patterns. It is observed that uniform patterns account for nearly 90% of all patterns in the (8,1) neighborhood and for about 70% of all patterns in the (16, 2) neighborhood in texture images [20], [24].

Accumulating the patterns that have more than two transitions into a single bin yields an LBP operator, denoted $\text{LBP}^{u2}_{P,R}$, with fewer than 2^P bins. For example, the number of labels for a neighborhood of 8 pixels is 256 for the standard LBP but 59 for LBP^{u2}. Then, a histogram of the frequency of the different labels produced by the LBP operator can be built [20]. In this work, the best results were achieved through the traditional configuration ($\text{LBP}^{u2}_{8,2}$), which generates a feature vector of 59 components.

B. Local Phase Quantization

The LPQ [23] is based on quantized phase information of the Discrete Fourier Transform (DFT). It uses the local phase information extracted using the 2-D DFT or, more precisely, a Short-Term Fourier Transform (STFT) computed over a rectangular $M\times M$ neighborhood N_x at each pixel position x of the image $f(x)$ defined by

$$F(u,x) = \sum\nolimits_{y\in N_x} f(x-y)e^{-j2\pi u^T y} = w_u^T f_x \qquad (2)$$

where w_u is the basis vector of the 2-D DFT at frequency u, and f_x is another vector containing all M^2 image samples from N_x.

The STFT can be implemented using a 2-D convolutions $f(x)e^{-2\pi j u^T x}$ for all u. In LPQ only four complex coefficients are considered, corresponding to 2-D frequencies $u_1 = [a, 0]^T$, $u_2 = [0, a]^T$, $u_3 = [a, a]^T$, and $u_4 = [a, -a]^T$, where a is a scalar frequency below the first zero crossing of the DFT $H(u)$. $H(u)$ is DFT of the point spread function of the blur, and u is a vector of coordinates $[u, v]^T$. More details about the LPQ formal definition can be found in [23], where Ojansivu e Heikkila introduced all mathematical formalism. At the end, we will have an 8-position resulting vector G_x for each pixel in the original image. These vectors G_x are quantized using a simple scalar quantizer (Eq. 3, and 4), where g_j is the *j*th component of G_x [23].

$$q_j = \begin{cases} 1, & \text{if } g_j \geqslant 0 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

$$b = \sum_{j=1}^{8} q_j 2^{j-1}. \tag{4}$$

The quantized coefficients are represented as integer values between 0-255 using binary coding (Eq. 4). These binary codes will be generated and accumulated in a 256-bin histogram, similar to the LBP method. The accumulated values in the histogram will be used as the LPQ 256-dimensional feature vector.

V. Experimental Results

In this work we adopted the experiment protocol proposed in [12], which consists in two distinct tasks: i) Writer Identification using Arabic samples for training and English samples for testing and ii) Writer identification using English samples for training and Arabic samples for testing. In addition, we added two single-script experiments better assess the stability of the proposed methods: iii) Writer identification using Arabic for training and testing and iv) Writer identification using English for training and testing.

Regarding the database, the organizers [12] state that the training data was composed of 300 Arabic and 300 English handwritten samples. For validation and testing, other 100 Arabic and English samples were made available, respectively. Unfortunately, from [12] it is not possible to conclude which samples (free text of copied) were used for training and testing, hence we were not able to use the same setup. In our experiments, we used 475 samples for training and validation (one sample per writer) and other 475 for testing (one sample per writer). Still in this section, we describe the different scenarios we have elaborated to use free and copied texts.

As the base classifier, we make use of Support Vector Machines (SVMs) with Gaussian kernel. The free parameters of the system and for SVM training were chosen using 5-fold cross validation. Various kernels were tried out, and the best results were reached using a Gaussian kernel. Parameters C and γ were determined through a grid search. Note that normalization was performed by linearly scaling each attribute to the range [-1,+1]. The Equal Error Rate (EER), which is computed in the testing set, was used for evaluation purposes

$$EER = \frac{FP + FN}{TP + TN + FP + FN} \tag{5}$$

where FP, FN, TP, and TN stand for False Positive, False Negative, True Positive, and True Negative, respectively.

The identification problem consists in identifying writer I among all the writers enrolled in the system. Given an input feature vector x from a texture image S, we determine the identity $I_c, c \in 1, 2, \ldots, N$, where N is the number of writers enrolled in the system. Hence, $S \in I_c$, if $\max_c\{D_{model}(x, R_c)\}$, where D_{model} is the dissimilarity model trained to return an estimation of posterior probability, which indicates that S and the reference R_c belong to the same writer. In this work, nine texture images per author were used as references (R = 9) to generate positive and negative samples, and nine texture images (S = 9) were used for identification. In this way, both reference and questioned documents were compacted and nine pieces of texture were extracted. The nine fragments are classified independently generating a partial decision and then a final decision is computed by combine all partial decisions. Different fusion rules were tried out but the Sum Rule produced the best results

We first present the results of single-script experiments so that we can assess the discriminant power of the dissimilarity approach on a less complex problem. Since QUWI dataset contains free (text-independent) and copied text (text-dependent), we have considered two different scenarios. Besides the intrinsic difference between free and copied texts, in this database the free texts are considerable smaller. In the first scenario, we have used copied text for training and free text letters for testing. Then, the second scenario, the letters were inverted, i.e, free text for training and copied for testing. The results of these experiments using classifiers trained with LBP and LPQ are reported in Table I.

Table I

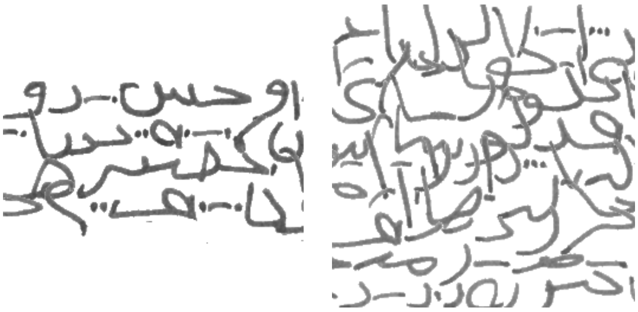
Error rates (%) on single-script using free and copied texts for training and testing and vice-versa.

Script	Scenario 1		Scenario 2	
	LPB	LPQ	LBP	LPQ
Arabic	25.7	19.4	6.1	3.6
English	31.2	20.3	1.7	1.1

To the best of our knowledge, the best single-script results on this dataset are 87.6% (EER = 12.4%) and 82.7% (EER = 17.3%) for Arabic and English scripts, respectively [25]. In our experiments, the results achieved in the second scenario surpassed by a fair margin the results reported in [25].

The discrepancy in terms of performance between both scenarios is related to the different amount of text available for copy and free text. Since the samples of free texts are composed of few lines, we could not create dense textures as we did on copied texts, which are composed of three paragraphs. Figure 4 shows the texture created from free and copied texts.

Training the dissimilarity model with less dense textures imposes no problem since we can generate a large number of



(a) (b)

Figure 4. Examples of texture created from (a) free text and (b) copied text

samples for training. However, the number of texture samples is limited to nine for the identification process, therefore, it is important to count on dense texture images to achieve a good identification rate. The results reported in Table I corroborate that.

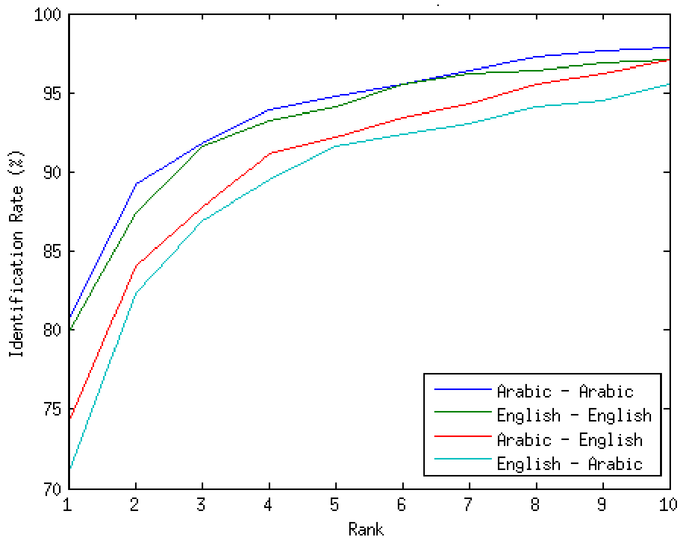
Regarding the multi-script identification, we replicate the same experiments but using different scripts for training and testing. Table II shows that, similarly to the previous experiments, the best results were achieved using free text for training and copied for testing. Another interesting finding is that we may reach almost the same results on the English script using either English or Arabic scripts for training. The opposite does not hold, though. Regarding the experiment with English for training and Arabic for testing, the lowest ERR is 5.5% using the classifier trained with LPQ, which is about two percentage points less than the single-script experiment using Arabic for training and testing.

Table II
ERROR RATES (%) ON MULTI-SCRIPT USING FREE AND COPIED TEXTS FOR TRAINING AND TESTING AND VICE-VERSA.

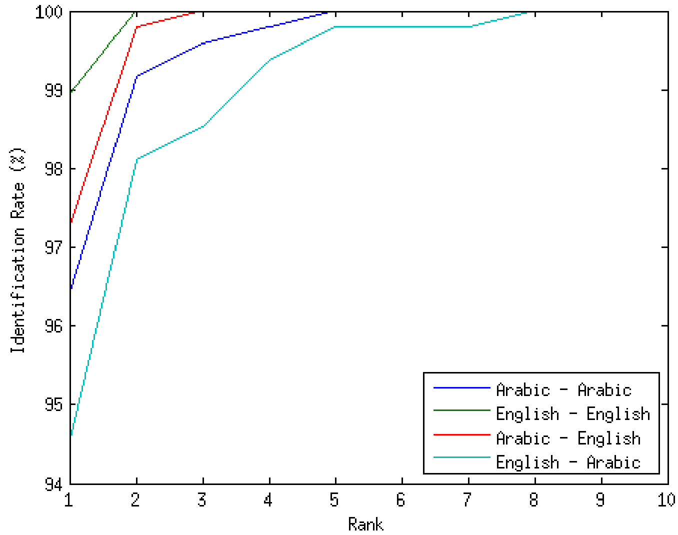
Script		Scenario 1		Scenario 2	
Training	Testing	LPB	LPQ	LBP	LPQ
Arabic	English	22.1	25.9	1.3	2.8
English	Arabic	38.3	29.1	9.1	5.5

Figure 5 shows the Cumulative Match Characteristic (CMC) curves of the classifier trained with LPQ for scenarios I and II. The CMC curves show the probability of identification against the 1:N candidate list size returned. The faster the CMC curve approaches 1, which indicates that the user always appears on a particular size of candidate list, the better the matching algorithm.

Table III presents the best two results reported in [12]. The Nuremberg system [26] relies on the computation of Zernike moments at the contours of the handwriting while the CVC method uses a variant of the LPB with PCA for dimensionality reduction. As we have stated before, we were not able to use same database setup employed in the competition so a straightforward comparison is not fair. However, based on the different scenarios we covered in this study it is clear that the dissimilarity approach surpasses by a large margin the results



(a)



(b)

Figure 5. The CMC curves for the classifiers trained with LPQ (a) scenario I and (b) scenario II.

of the competition for both tasks I (training with Arabic and testing with English) and II (training with English and testing with Arabic).

Table III
ERROR RATES (%) REPORTED IN [12]

System	Task I	Task II
Nuremberg	45	71
CVC	79	86

To verify whether the power of the proposed strategy lies on the features or on the dissimilarity model, we have trained a writer-dependent classifier (multi-class SVM) using the LPQ features. This SVM uses a Gaussian kernel and similarly to the previous experiments, the parameters C and γ were optimized by means of a grid search with hold-out validation, using the training set to train SVM parameters and the validation set to evaluate the performance.

Using this multi-class SVM our smallest EER on tasks I and II were around 70%. In other words, our results using this model are no better than those reported in Table III. One may argue that the features we have used or even those employed in [26] do not have the discriminant power necessary to achieve a good performance on writer identification. However, these features have been successfully used for single-script writer identification reaching state-of-the-art performance.

The results we have presented in this study show that succeeding in multi-script identification is rather a matter of classification strategy than feature definition. This corroborate our hypothesis that the dissimilarity approach is a suitable strategy to deal with multi-script writer identification.

VI. Conclusion

In the beginning of this work we have argued that we could benefit from the WI approach for multi-script writer identification. Our hypothesis was if a WI approach is built to predict whether or not two pieces of the handwriting were written by the same person, the model should be able to do that independently of the script that has been used. As we have stated before, this should hold if the representation is similar for both scripts.

To validate such an hypothesis, we have performed a set of experiments on 475 writers of the QUWI database, which contains copied and free texts of the same writer in Arabic and English scripts. Two different scenarios were set up in our experimental protocol. In the first scenario, we have used copied text for training and free text letters for testing. Then, the second scenario, the letters were inverted, i.e, free text for training and copied for testing. In both cases, the results corroborate our initial hypothesis. The error rates were considerable smaller than those reported in the literature and the results found on the second scenario were comparable to single-script writer identification systems.

As future works we plan to investigate other types of representation to build the dissimilarity feature vectors such as those that can be extracted from convolutional neural networks.

References

[1] A. Schlapbach and H. Bunke, "A writer identification and verification system using hmm based recognizers," *Pattern Analysis and Applications*, vol. 10, pp. 33–43, 2007.

[2] D. Bertolini, L. S. Oliveira, E. Justino, and R. Sabourin, "Texture-based descriptors for writer identification and verification," *Expert Systems With Applications*, vol. 40, no. 6, pp. 2069–2080, 2013.

[3] A. Bensefia, T. Paquet, and L. Heutte, "A writer identification and verification system," *Pattern Recognition Letters*, vol. 26, no. 13, pp. 2080–2092, 2005.

[4] I. Siddiqi and N. Vincent, "Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features," *Pattern Recognition*, vol. 43, pp. 3853–3865, 2010.

[5] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 701–717, 2007.

[6] X. Wang and X. Ding, "n effective writer verification algorithm using negative samples," in *International Workshop on Frontiers in Handwriting Recognition,*, 2004, pp. 509–514.

[7] H. Kameya, S. Mori, and R. Oka, "A segmentation-free biometric writer verification method based on continuous dynamic programming," *Pattern Recognition Letters*, vol. 27, no. 6, pp. 567–577, 2006.

[8] S. Chanda, K. Franke, U. Pal, and T. Wakabayashi, "Text independent writer identification for bengali script," in *20th International Conference on Pattern Recognition*, 2010, pp. 2005–2008.

[9] S. Chanda, K. Franke, and U. Pal, "Text independent writer identification for oriya script," in *10th IAPR International Workshop on Document Analysis Systems*, 2012, pp. 369–373.

[10] C. Djeddi, I. Siddiqi, L. Souici-Meslati, and A. Ennaji, "Text-independent writer recognition using multi-script handwritten texts," *Pattern Recognition Letters*, vol. 34, pp. 1196–1202, 2013.

[11] S. Al-Maadeed, W. Ayouby, A. Hassaine, and J. Aljaam, "Quwi: An arabic and english handwriting dataset for offline writer identification," in *13th International Conference on Frontiers in Handwriting Recognition*, 2012, pp. 746–751.

[12] C. Djeddi, S. A. Maadeed, A. Gattal, I. Siddiqi, L. Souici-Meslati, and H. E. Abed, "Icdar2015 competition on multi-script writer identification and gender classification using quwi database," in *13th International Conference on Document Analysis and Recognition*, 2015, pp. 1191–1195.

[13] U. V. Marti and H. Bunke, "The IAM-database: an english sentence database for offline handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, no. 1, pp. 39–46, 2002.

[14] D. Bertolini, L. S. Oliveira, E. Justino, and R. Sabourin, "Assessing textural features for writer identification on different writing styles and forgeries," in *International Conference on Pattern Recognition*, 2014, pp. 268–272.

[15] C. Freitas, L. Oliveira, R. Sabourin, and F. Bortolozzi, "Brazilian forensic letter database," in *11th International Workshop on Frontiers on Handwriting Recognition*, Montreal, Canada, 2008.

[16] L. Schomaker and L. Vuurpijl, "Forensic writer identification: A benchmark data set and a comparison of two systems," Nijmegen, Tech. Rep., February 2000.

[17] R. M. Haralick, K. S. Shanmugan, and I. Dunstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.

[18] D. Gabor, "Theory of communications," *Journal of Institution of Electrical Engineers*, pp. 429–457, 1946.

[19] N. A. Hamilton, R. S. Pantelic, K. Hanson, and R. D. Teasdale, "Fast automated cell phenotype image classification," *BMC Bioinformatics*, vol. 8, 2007. [Online]. Available: http://www.biomedcentral.com/1471-2105/8/110

[20] T. Ojala, M. Pietikäinen, and D. Harwook, "Comparative studyof texture measures with classification based on feature distributions," *Pattern Recognition*, vol. 29, pp. 51–59, 1996.

[21] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions on Image Processing*, vol. 19, pp. 1657–1663, 2010.

[22] J. Chen, V. Kellokumpu, G. Zhao, and M. Pietikäinen, "RLBP: Robust Local Binary Pattern," in *Proceedings of the British machine vision conference*, 2013, pp. 1–12.

[23] V. Ojansivu and J. Heikkila, "Blur insensitive texture classification using local phase quantization." in *Proc. Image and Signal Processing (ICISP 2008)*, 2008, pp. 236–243.

[24] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, 2002.

[25] S. Al-Maadeed, A. Hassaine, A. Bouridane, and M. A. Tahir, "Novel geometric features for off-line writer identification," *Pattern Analysis and Applications*, pp. 1–10, 2014.

[26] V. Christiein, D. Bernecker, and E. Angelopoulou, "Writer identi cation using vlad encoded contour-zernike moments," in *13th International Conference on Document Analysis and Recognition*, 2015, pp. 906–910.