Kurdistan Journal of Applied Research (KJAR) | Print-ISSN: 2411-7684 – Electronic-ISSN: 2411-7706 | kjar.spu.edu.iq
Volume 2 | Issue 3 | August 2017 | DOI: 10.24017/science.2017.3.64

Writer Identification on Multi-Script Handwritten Using Optimum Features

Ahmed Abdullah Ahmed
Computer Science Dept.
Kurdistan Technical Institute
Sulaimani Heights, Sulaimani, Iraq
ahmed.abdullah@kti.edu.krd

Harith Raad Hasan
Sulaimani polytechnic university,
Kurdistan Technical Institute
Sulaimani, Iraq
harith.hasan@spu.edu.iq

Fariaa Abdalmajeed Hameed
Technical College of Informatics
Sulaimani polytechnic university,
Sulaimani, Iraq
fariaa.hameed@spu.edu.iq

Omar Ismael Al-Sanjary
Faculty of Information Science
and Engineering
Management and Science University,
omar_ismael@msu.edu.my

Abstract: ***Recognizing the writer of a text that has been handwritten is a very intriguing research problem in the field of document analysis and recognition. This study tables an automatic way of recognizing the writer from handwritten samples. Even though much has been done in previous researches that have presented other various methods, it is still clear that the field has a room for improvement. This particular method uses Optimum Features based writer characterization. Here, each of the samples written is grouped according to their set of features that are acquired from a computed codebook. This proposed codebook is different from the others which segment the samples into graphemes by fragmenting a certain part of the writing known as ending strokes. The proposed technique is employed to a locate and extract the handwriting fragments from ending zone and then grouped the similar fragments to generate a new cluster known as ending cluster. The cluster that comes in handy in the process of coming up with the ending codebook through picking out the center of the same fragment group. The process is finalized by evaluating the proposed method on four datasets of the various languages. This method being proposed had an impressive 97.12% identification rate which is rates the best result on the ICFHR dataset.***

Keywords: Off-line text-independent writer identification, feature extraction, codebook, fragments.

I. INTRODUCTION

The objective of offline writer identification is to determine the specific writer who has written a certain article from a group of known writers. The identification is mainly based on matching the writing preference of the individual that can be picked out form his/her handwriting. Writer identification has become a very important field in document analysis and recognition in the past twenty years. This can be attributed to the fact that there are some very diverse applications in forensics, biometrics as well as paleography.

There are two main families of approaches for writer identification that have been proposed by the various researchers and practitioners. The first one is global approaches [1, 2, 3, 4] which mainly looks at the general look and feeling of writing. The second one is local approaches [5, 6, 7, 8] which uses local feature that have been acquired from characters, graphemes or even sub graphemes of the writing. The third one is combining global and local features [9, 10] which is common for bettering the writer identification performance.

Recent research that has been carried out on writer identification have mainly concentrated on the extraction of redundant patterns in writing which are commonly known as codebook [5, 6, 7]. In all these researches that have been conducted, the ink trace is divided into graphemes, sub-graphemes or K adjacent segments, which are then grouped into clusters that represent the frequent writing shapes constituting the codebook. The codebook can be attributed to a specific writer where specific writing patterns are grouped together and attributed to each writer [6, 11]. The can also be grouped as universal if the redundant patterns have been acquired from a global dataset under study [12] or even from an independent dataset [7, 9, 10]. It is the general conclusion that the universal codebooks tend to depict better results in comparison to the writer based codebooks [6, 11].

Out of the commonly known codebook base methods, the universal codebooks specific to graphemes/ fraglets are acquired from [10, 12]. The graphemes are acquired from a process of segmentation founded on the evaluation of the upper contour minima. The graphemes of writing which are being researched are then compared to the codebook patterns in order to count the number of occurrences of each of the codebook entries in certain writing. The writer is then characterized by the probability distributions of the codebook patterns in certain writing. This process has been criticized due to its dependence on the segmentation process and if any chance the segmentation process breaks down, then there is an impact on the process that follow including the features extraction, clustering and identification. Also, the process has the disadvantage of containing the semantic data of the script being studied.

The criticism above has been looked at by Siddiqi et al. in [9]. He proposed that the codebooks be generated from the stroke fragments or the sub graphemes. The fragments can be acquired by studying a reduced scale of observation and grouping the text into a large number of small sub-images (windows) of size n×n pixels. Clustering of the patterns is then done to come up with a set of representative patterns constituting the codebook. Interestingly, the codebook generated from this method

has some very simple writing shapes and its effectiveness is equivalent of the Bulacu et al [10]. Similar to the latter however, this method is also dependent on an independent dataset that is used in acquisition of the codebook.

This paper is organized in such a way that we first look at the datasets; the other section discusses the details of the proposed method as section three. Section four will then present the experimental results and lastly section five contains the conclusions.

II. DATA SETS

In this study, we will look at four different data sets namely 1) **GRDS** data set [13], (2) **IAM** data set [14], (3) Kurdish Handwritten Data set (**KURD**) and (4) **ICFHR** Handwritten Database [15]. Following is a brief description of all the datasets named above.

A. GRDS Data set

The GDRS [13] is a new dataset generated by a research group in the Computational Intelligence Laboratory at the National Center for Scientific Research "Demokritos", Greece. It has 26 independent writers who have all been acquired from eight different sample texts written the four different languages that is English, German, Greek and French. All languages have each two texts. The dataset has been employed in the ICDAR 2011 Writer Identification Contest [13].

B. IAM Data set

The IAM data set [14] has form with hand written English texts of dissimilar content by 650 different writers. One page is credited to 350 writers, two pages with 300 writers and at last four pages containing 125 writers. It is the most common dataset for not only evaluation of writer identification but also handwriting recognition as well as other related tasks. In our study, we make use of 600 documents by 300 different writers.

C. KURD Data set

This dataset was developed at the research laboratory at Kurdistan Technical Institute – Iraq. The reason as to why it was developed was that there was need to evaluate writer recognition systems in reference to the Kurdish text. There were a total of 1076 people of varied ages as well as educational backgrounds who contributed to 4 pages each. There were allowed to write any content and they were not also restricted to the type if color they used nor the writing instrument. The pages were then scanned at 300 dpi and then stored in the tiff format. This study has made use of 800 document by 200 various writers.

D. ICFHR Data set

This is an Arabic writer identification data base that is in use in the current study. This is a very essential data due to the fact that it contains various samples of unrestricted Arabic handwritten documents that have been scanned using 256 grey levels equivalent to 600 dpi. The texts have been volunteered by 206 writers who are all varied in nationality, educational background, age and gender. Each writer was requested to write 3 paragraphs in Arabic. Once this was done, the data set was organized to evaluate the work done by the participants in the competition [15]. The first two paragraphs were used for training while was used for the testing. The first two paragraphs were then taken for certain writers to test the ability of the system to tell the unknown writers.

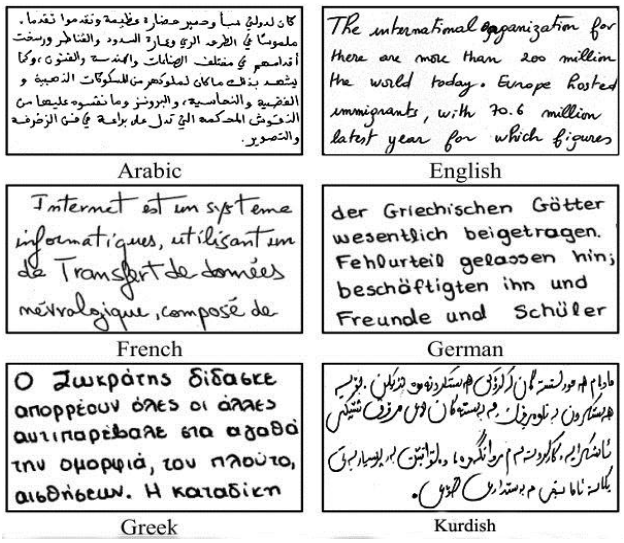

Figure 1: Samples from the IAM, GRDS, KURD and ICFHR data sets.

Some of the images scanned in Arabic, English, French, German, Greek and Kurdish that were acquired from the data set have been shown in Figure. The Summary Statistics of the four data sets is illustrated in Table 1.

Table 1 Summarizes the statistics of the four data sets.

	ICFHR	KURD	GRDS	IAM
Arabic	550samples	-	-	-
English	-	-	52samples	600samples
Kurdish	-	800samples	-	-
German	-	-	52samples	-
Greek	-	-	52samples	-
French	-	-	52samples	-

III. METHODOLOGY

This section discusses the techniques that have been utilized by proposed a new method to describe the writer of a given writing samples by using new features known as Optimum Features. The proposed method has already gone through the pre-processing stage (binarization process, connected components detection, and removal of punctuation marks). The primary process of this technique is divided into five steps with the aim of representing the writer's distinctive way, starting with contour detection to extract significant information from the handwriting. This distinguishes the writer through his writing style with the help of Moore's algorithm. Curve fragmentation is employed to a locate and extract the

handwriting from Ending zone as small fragments, and the fragments are normalized in order to have equalized coordinates of the fragment points to the origin (0, 0). The similar fragments of Ending strokes are grouped together to create the Ending cluster. This cluster lead to develop a new codebook called Ending Codebook by choosing the center of every similar fragments group. The entire phases/steps of writer representation are further discussed in the next sub-sections.

A. Contour Detection

The pre-processing stage on the grayscale of the handwritten images began with binarization process through Otsu's algorithm [16], and applying connected components detection to detect the writing with the help of 8-connectivity method, after which punctuation marks are removed. Following the above pre-processing stage, the proposed technique used contour detection for the extraction of information from the writing shape that provides a description of the writer characteristic. Contour detection, better known as border following is used on digital images for extracting writing boundary as features that are subsequently used in writer representation. In other words, the accurate contour extraction will generate accurate features that will contribute to the chances for accurate writer representation of the samples. In the current study, Moore's algorithm [17] is used to locate a black pixel after which it is referred to as the 'start' pixel. The algorithm necessitates going back to the white pixel every time a black pixel is hit and going around the current pixel in a clockwise direction, visiting every pixel in its Moor vicinity, until the black pixel is hit. It eradicates the moment where the start pixel is visited for the second time and the black pixels that have been walked over will serve the pattern contour. In this regard, the contour tracing step is significant as the extracted writing boundary will be revealed on the writer description. The proposed method obtained the significant code from the script's writing contours that includes the writing specification and enables the preservation of the writer-dependent variations between character shapes.

B. Curve Fragment Extraction

In the proposed method, the curves fragment taken from the contours of connected components are used, as the primary study focus entails the extraction of the fragments of the connected components that contains significant and effective information that assists in writer identification. The above is carried out instead of segmenting the writing in small windows to form a codebook in a method that has been extensively utilized in writer identification as this may not carry any semantic information [18]. Moreover, the primary aim behind the analysis of handwriting fragments is to examine the shapes that appear repeatedly in varying characters/graphemes in a specific writing sample. To this end, specific parts of different characters share common patterns that are drawn by the writer in the same way regardless of the written character. This is demonstrated by Figure 2, where different characters fragments written by the same writer are similar (e.g. Ending strokes). This can also be much noted in the samples of writing fragments as illustrated in Figure 3 where different characters with the same form was excerpted from Figure 2.

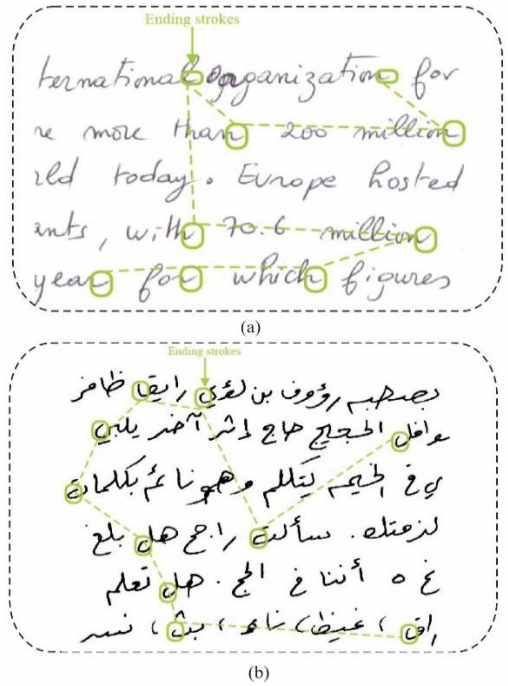

Figure 2 Similar patterns in different characters: (a) Sample from IAM Data set [14], (b) Sample from ICFHR Data set [15].

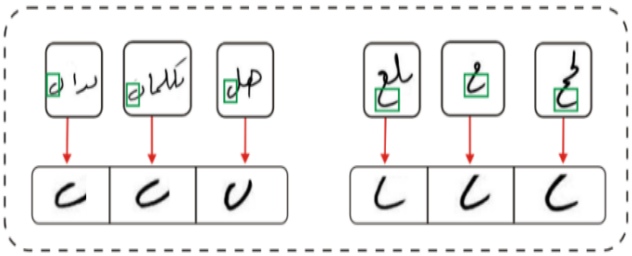

Figure 3 Excerpt Patterns of Ending stroke

The proposed method concentrates on the important writing areas that provides the most important information on the basis of the foundation of this study and the theoretical work contributed by the handwriting analysis experts which contends that ending-strokes is deemed to be the most significant handwriting parts as they can contribute in predicting personality traits. Figures 2, 3 support this theory of the handwriting experts [19,20] which serve the main concern of this study. The significant of the proposed method is to characterize the handwriting based on the Ending strokes not like previous studies which are employed all the handwriting text to represent the writer.

The current method extracted number of contour fragments from text image with a particular length or number of points (L_f) with specific overlapping gap which both have been empirically selected, based on series of experiments that have been applied with different length

of fragment and variant overlapping gap. The best length of fragment was 45 points with 15 points as overlapping gap. Furthermore, every contour fragment with (L_f) points can be employed as a code and the number of such codes can be manipulated by modifying the distances between the starting fragment points (overlapping gap). These distances can be fixed to a certain value in this study overlapping gap was 15 points. The curve fragment extraction variables of connected component "***you***" are presented in Figure 4.

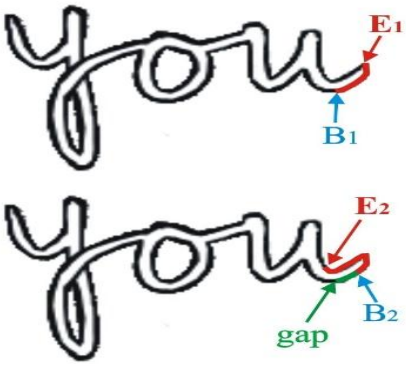

Figure 4 Curve fragment code extraction approach for two fragments where (L_f = 45 and *gap* = 15), B_1 and E_1 are the beginning point and the ending point of a segment with L_f points on the contour

Once all the fragments of the connected components have been extracted, the proposed method computes the average of the distances between all points of each fragment with the left or right side of the bounding box of the connected component depend on script whether left to right or right to left writing. If the fragment is located near to that area means this fragment in the Ending zone, and any fragment not belong to the Ending zone will be discarded.

Distance calculation of every fragment points in the connected component inside the bounding box, this process is very critical because it is determining the location of each fragment based on their location the proposed method decide if the fragment assign as an Ending stroke or discard the fragment. Moreover, the ending zone have been determined empirically based on number of experiments have been applied with different size of connected components such as 10, 20, 30, 40 and 50% from the actual size of the bounding box of the connected components based on these experiments the optimum size of the Ending zone was (η = 30%) of the connected components which is consider as area of interest, Figure 5 illustrates the Ending zone.

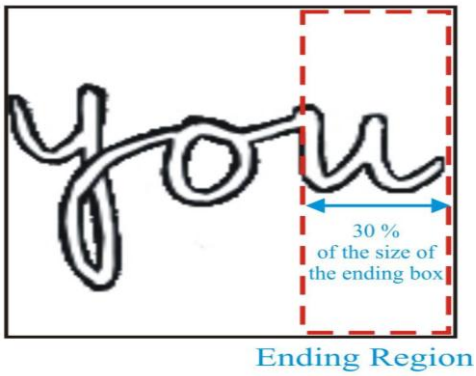

Figure 5 Determining Ending Region

After compute, the distance of the fragment points with the left or right side of the bounding box which is represent the border of the connected component, the current technique divides the fragment to Ending strokes and other strokes based on their location, Figure 6 illustrates the Ending strokes after removing unwanted strokes. As mentioned before 41 curve fragments have been extracted from the connected component "***you***", and now the proposed method reduced the number of extracted fragments to 30 % because it has been deleted unimportant fragments which are not located at the Ending region of a connected component. Based on the theory of the handwriting analysis experts which says all the fragments that are located in the Ending zone of the connected component is important to represent the writer.

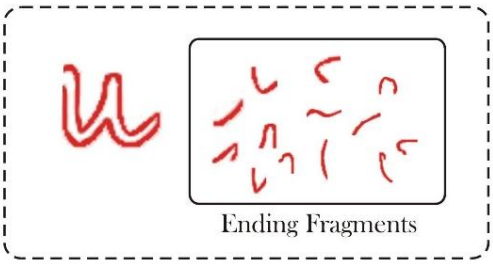

Figure 6 Contour and shapes of extracted fragments of the Ending zone when L_f = 45 and *gap* = 15 using curve fragment method

After the entire curve fragments of the ending zone have been extracted, the next step involves the normalization of the coordinates of every fragment to prepare them for the process of clustering.

C. Fragment Normalization

It is crucial to normalize the patterns coordinates to origin coordinate (0, 0) and the standard deviation of one prior to grouping the similar fragments into clusters. Normalization employs the following relations [21].

$$\vec{\mathbf{x}} \leftarrow (\vec{\boldsymbol{x}} - \boldsymbol{\mu_x})/\boldsymbol{\sigma_x}$$
$$\vec{\mathbf{y}} \leftarrow (\vec{\mathbf{y}} - \boldsymbol{\mu_y})/\boldsymbol{\sigma_y} \qquad (\mathbf{1})$$

In the above equations, $\vec{x}$ and $\vec{y}$ depict the collections of x and y coordinates of a contour fragment, μ_x and μ_y

depict the averages, while σ_x and σ_y depict the standard deviation of $\vec{x}$ and $\vec{y}$, respectively. In this respect, vectors that have the normalized fragments coordinates are all of the same length $2L_f$, and they represent codes that are used for training a codebook and for extracting feature vectors from samples of handwriting.

After the normalization of the curve fragments, the next step of the proposed method is the clustering of the normalized code fragments into different categories and this is carried out by conducting a comparison of the extracted fragments with the help of Euclidean distance to identify the similarity/differences between the two fragments.

D. Clustering of Fragments

To reiterate, in the prior section, every sample was represented as a curve fragments code set. In this section, the fragments are grouped according to their pairwise similarity. Moreover, the fragments within each cluster are different from those of other clusters, and every individual cluster depicts a set of segments. The current method requires the selection of a clustering algorithm that does not call for the determination of a priori number of clusters. Accordingly, the hierarchical clustering is used to group all the similar patterns in one cluster and for the calculation of the fragments distance, the Euclidian distance measures is employed to measure the distance among the fragments.

This method begins with each object as one class, and then the merging of objects into classes until the entire objects that are similar are merged in one cluster. This calls for the proposed model to provide a definition of a distance (or similarity) measure that enables the comparison between the two classes. The hierarchical clustering is presented in Figure 7. In the present study, such pattern is deemed as seven fragments which are grouped together all the similar fragments in one cluster based on similarity threshold. In this research, the threshold similarity to merge the clusters is 50% which is empirically determined.

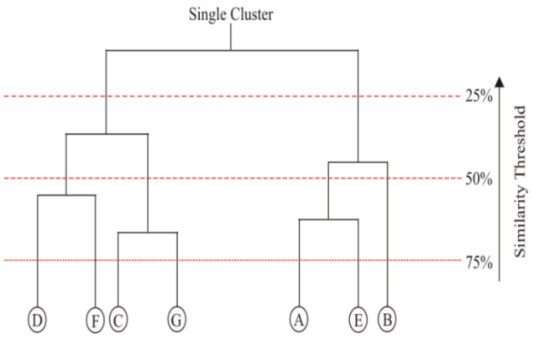

Figure 7 Hierarchical clustering

The distance found between the two classes is computed as the average distance between patterns features in various clusters. The present study used the average-link technique for clustering purpose, where the distance between two classes is referred to as the average of the distances between the entire objects within the two classes. The method is specifically presented in the following equation.

$$Dist(c_i, c_j) = \underset{x \in c_i, y \in c_j}{avg} Dist(x, y) \quad (2)$$

where, c_i and c_j be two classes. *Dist* defines the distance between c_i and c_j.

Furthermore, owing to the unknown number of classes for each writer, the study makes use of the distance criterion to depict the number of clusters, where for every writer, the proposed method produces one set of clusters from the Ending strokes. Figure 8 illustrates an example of clustering similar fragments. There are different numbers of clusters, with each cluster containing homogenous groups of similar segments which are distinct to each particular cluster. The clusters are separated by the black square as presented in Figure 8.

E. Codebook Generation

In this section, the generation of writer's codebook namely Ending codebook, is explained. Every cluster contains different number of fragments and each has seemingly homogeneous groups of similar patterns that are distinct to itself (different from other cluster elements). After grouping the similar fragments, the most similar member of each cluster is selected by calculating the distance among the fragments through Euclidian distance measures. The average distance of each fragment is then calculated to identify the center fragment in every cluster. Consequently, all the centers and their cardinalities are gathered to form the new codebook. In the generation stage, a new codebook will be produced known the Ending codebook as depicted in Figure 9.

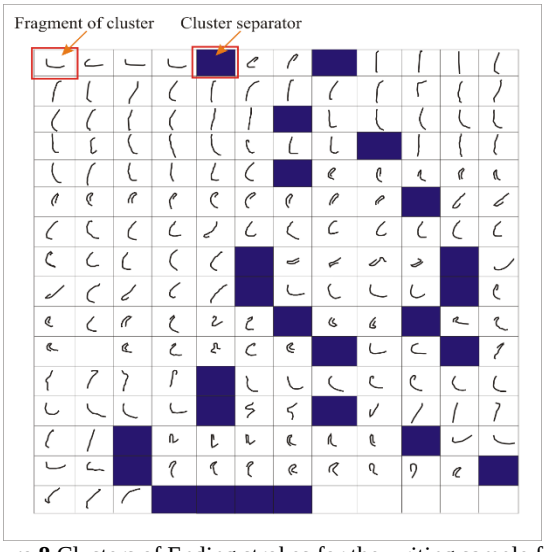

Figure 8 Clusters of Ending strokes for the writing sample from ICFHR Data set [15]

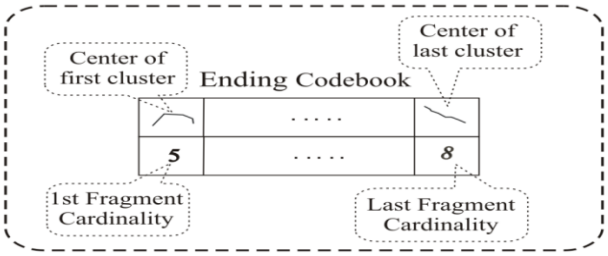

Figure 9 Ending Codebook generation

V. WRITER IDENTIFICATION

As discussed earlier, each document in the reference dataset is depicted by a set of patterns in the Ending codebook and the test document T is depicted by the codebook. In this context, the dissimilarity or similarity between the writing samples is identified by calculating the distance between their respective features and test document T is compared with trained document D. According to [9, 22], different distance measure can be utilized for the comparison of two distributions of sample χ^2 distance, Hamming distance, Bhattacharyya, Minkowski and Non-Intersection. Such studies reached to the conclusion that χ^2 distance showed optimum performance among the other distances. Thus, the result presented in the next sections is according to the χ^2 distance which is defined by the following equation;

$$\chi^2(p,q) = \sum_{i=1}^{\dim} \frac{(p_i - q_i)^2}{p_i + q_i} \quad \textbf{(3)}$$

In the equation (Equation 3), p and q denote the two similarities (histograms) for comparison, p_i denotes the segment i of the histogram and *dim* denotes the total number of histogram segments. The distance is calculated for the Ending codebook.

VI. EXPERIMENTAL RESULTS

In order to quantitatively evaluate the performance of the proposed scheme, a query document is compared with all the documents in the training document and computes the similarity index (distance) with each of these documents. The retrieved list is then sorted in the order of increasing distance from the query document. Ideally, the writer at rank 1 should match the writer of the query document. However, this study retrieves a longer list of probable writers up to a given rank N to increase the chances of finding the author of query document in the list (Top N). For each evaluation, this study reports the Top1, Top5 and Top10 identification rates. Top 1 means that the query document is matched with the first ranked sample in the sorted list. Similarly, Top10 means that the writer of the query document is within the top 10 writers retrieved by the system. This study evaluates the performance of ending codebook. Initial experiments were conducted on different data sets, 206 writers from the ICFHR dataset [15], 600 documents of 300 different writers from IAM data set [14], 208 documents of 26 different writers in four different languages (German, English, French, and Greek) from GRDS data set [13] and finally we have used 800 documents of 200 different writers from KURD Data set. Table 2 summarizes the results of ending codebook on different Data sets.

Table 2 Performance of The Proposed Method on Different Data Sets

Data Set	# Writers	Identification Rate
KURD	200	94.63 %
ICFHR **[15]**	206	97.12 %
IAM **[14]**	300	95.59 %
GRDS **[13]**	26	100.0 %

In addition to the above evaluations, the following results provides a performance comparison between the proposed writer identification method with the state- of-the- art methods found in the literature that have been implemented using the same ICFHR2012 database [15]. The benchmark was performed against four recent methods, which are: (1) Oriented Basic Image Feature Columns and the Delta encoding [23]; (2) Combination of edge-hinge and grapheme features proposed by [24] – their method was implemented in 2012 using the ICFHR dataset by Wayne Zhang [15]; (3) Oriented Basic Image Features (oBIFs) system according to local symmetry and orientation [15, 25]; (4) SVMs with a diffusion kernel by Yanir Serroussi via YT team [15, 26]. It is worth mentioned that the benchmarked methods 2, 3 and 4 were the top 3 performers published in ICFHR2012 Competition on Writer Identification - Challenge 2: Arabic Scripts held in 2012 [15].

It is observed from the Table 3 that the results clearly revealed the domination of the proposed method over the existing methods: The identification rate was markedly improved by about 2 to 4 percentage points

Table 3 Performance Comparisons of Writer Identification Methods

Authors	Year	Dataset	Writers	Performance
Wayne Zhang [15]	2012	ICFHR	206	95.3 %
Newell and Griffin [15]	2012	ICFHR	206	95.3 %
YT team [15]	2012	ICFHR	206	93.29%
Newell and Griffi [23]	2014	ICFHR	206	95.3 %
Proposed Method	**2017**	**ICFHR**	**206**	97.12 **%**

VII. CONCLUSIONS

This paper studied the offline text-independent writer identification problem using Ending codebook approach. The Ending fragments code extraction methods were introduced and their performances were examined. Results achieved through these approaches were better than the performances of the existing methods. This technique extracts the code from the contour detection of connected components and they have the advantage of

using the repeatedly appearing shapes which might be parts of different characters. Besides, this method extracts from particular area of writing which is ending strokes. Four databases, namely, GRDS, IAM, KURD, and ICFHR were used for testing the method.

VIII. REFERENCES

[1]. D. Chawki, L. Souici-Meslati, "A texture based approach for Arabic Writer Identification and Verification," In Machine and Web Intelligence (ICMWI), International Conference on IEEE, pp. 115-120, 2010.

[2]. U. Garain, T. Paquet, "Off-line Multi-Script Writer Identification using AR Coefficients, '' In Proc of the International Conference on Document Analysis and Recognition, Spain, pp. 991-995, 2009.

[3]. D. Chawki, I. Siddiqi, L. Souici-Meslati, A. Ennaji, ''Multi Script Writer Identification Optimized with Retrieval Mechanism,'' In Proc. of the International Conference on Frontiers Handwriting Recognition, Bari, Italy, pp. 507 – 512, 2012.

[4]. D. Chawki, L. Souici-Meslati, A. Ennaji, ''Writer Recognition on Arabic Handwritten Documents'', In Proc. of the International Conference on Image and Signal Processing, Agadir, Morocco, pp 493-501, June 2012.

[5]. L. Schomaker, M. Bulacu, "Automatic Writer Identification Using Connected-Component Contours and Edge-Based Features of Upper-Case Western Script," IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(6), pp. 787–798, 2004.

[6]. I. Siddiqi, N. Vincent, "Writer Identification in Handwritten Documents," In Proc. of the 9th International Conference on Document Analysis and Recognition, 1, pp. 108–112, 2007.

[7]. R. Jain, D. Doermann, "Offline Writer Identification using K-Adjacent Segments," In Proc. of the 11th International Conference on Document Analysis and Recognition (ICDAR'11) on IEEE, pp. 769-773, 2011.

[8]. A.A. Ahmed, G. Sulong, "Arabic Writer Identification: A Review of Literature," Journal of Theoretical & Applied Information Technology 69(3), 2014.

[9]. I. Siddiqi, N. Vincent, "Text independent writer recognition using redundant writing patterns with contour-based orientation and curvature features," In Pattern Recognition, 43(11), pp. 3853 – 3865, 2010.

[10]. M. Bulacu, L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), Special Issue - Biometrics: Progress and Directions, IEEE Computer Society, 29(4), pp. 701-717, 2007.

[11]. A. Bensefia, A. Nosary, T. Paquet, L. Heutte, "Writer identification by writer's invariants," In Proc. of the International Workshop on Frontiers in Handwriting Recognition, pp. 274-279, 2002.

[12]. A. Bensefia, T. Paquet, L. Heutte, "A writer identification and verification system," In Pattern Recognition Letters, 26(13), pp. 2080 – 2092, 2005.

[13]. G. Louloudis, N. Stamatopoulos, B. Gatos, "ICDAR 2011 - Writer Identification Contest," In Proc of the 11th International Conference on Document Analysis and Recognition, pp. 1475-1479, China, 2011.

[14]. U. Marti, H. Bunke, "The IAM-database: an English sentence database for offline handwriting recognition," In International Journal on Document Analysis and recognition, 5(1), pp. 1433–2825, 2002.

[15]. A. Hassaïne, S. Al-Maadeed, "ICFHR2012 competition on writer identification - Challenge 2: Arabic scripts," In Frontiers in Handwriting Recognition (ICFHR), International Conference on IEEE, pp. 835-840, 2012.

[16]. N. Otsu, "A threshold selection method from gray-level histograms," IEEE transactions on systems, man, and cybernetics 9(1), pp. 62-66, 1979.

[17]. P. Sharma, M. Diwakar, N. Lal, "Edge Detection using Moore Neighborhood," International Journal of Computer Applications, 61(3), pp. 26–30, 2013.

[18]. I. Siddiqi, N. Vincent, "Combining Global and Local Features for Writer Identification," in Proceedings of the 11. Int. Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 19–21, 2008.

[19]. E.C. Djamal, R. Darmawati, S.N. Ramdlan, "Application Image Processing to Predict Personality Based on Structure of Handwriting and Signature," in International Conference on Computer, Control, Informatics and Its Applications, pp. 163–168, 2013.

[20]. N. Mogharreban, S. Rahimi M. Sabharwal. "A combined crisp and fuzzy approach for handwriting analysis," Fuzzy Information, 2004. Processing NAFIPS'04. IEEE Annual Meeting, 1, pp. 351-356, 2004.

[21]. L. Schomaker, M. Bulacu, K. Franke, "Automatic writer identification using fragmented connected-component contours," Proc. - Int. Work. Front. Handwrit. Recognition, IWFHR, pp. 185–190, 2004.

[22]. M. Bulacu, L. Schomaker, A. Brink, "Text-independent writer identification and verification on offline arabic handwriting," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, 29(4), pp. 769–773, 2007.

[23]. A.J. Newell, L.D. Griffin, "Writer Identification Using Oriented Basic Image Features and the Delta Encoding," Pattern Recognition, 47(6), pp. 2255–2265, Jun. 2014.

[24]. L. V. D. Maaten, E. Postma, "Improving automatic writer identification," in Proc. of 17th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC), pp. 260–266, 2005.

[25]. A.J. Newell, L.D. Griffin, “Natural image character recognition using oriented basic image features,” in Proceedings - 2011 International Conference on Digital Image Computing: Techniques and Applications, DICTA, pp. 191–196, 2011.

[26]. H.J. Escalante, T. Solorio, M.M. Gómez, “Local Histograms of Character n-grams for Authorship Attribution,” Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 1, pp. 288–298, 2011.