ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2020.0315

ARABIC HANDWRITTEN CHARACTERS RECOGNITION VIA MULTI-SCALE HOG FEATURES AND MULTI-LAYER DEEP RULE-BASED CLASSIFICATION

Soumia Djaghbellou[1], Zahid Akhtar[2], Abderraouf Bouziane[3] and Abdelouahab Attia[4]
[1,4]Department of Computer Science, Mohamed El Bachir El Ibrahimi University of Bordj Bou Arreridj, Algeria
[2]Department of Computer Science, University of Memphis, United State of America
[3]LMSE Laboratory, Mohamed El Bachir El Ibrahimi University Bordj Bou Arreridj, Algeria

Abstract

Optical character recognition systems for handwritten Arabic language still face challenges, owing to high level of ambiguity, complexity and tremendous variations in human writing styles. In this paper, we propose a new and effective Arabic handwritten characters recognition framework using multi-scale histogram oriented gradient (HOG) features and the deep rule-based classifier (DRB). In the feature extraction stage, the proposed framework combines multi-scale HOG features, and then the DRB is applied on comprehensive HOG features to obtain the final classification label/class. This study involves experimental analyses that were conducted on the publicly available cursive Arabic Handwritten Characters Database (AHCD) containing 16800 characters. Experimental results demonstrate the efficacy of the proposed recognition system compared to the existing state-of-the-art-systems.

Keywords:

Arabic Character Recognition, Writing, DRB Classifier, HOG, AHCD

1. INTRODUCTION

Optical character recognition (OCR) systems have attained considerable progress owing to its impressive accuracy, and have demonstrated promising prospects in the field of handwritten Arabic characters recognition [1]. The Arabic s has several different traits, thereby making it a unique language. Not only the dialects but also handwritings among the Arabic speakers and writers vary with respect to context and locations.

The Arabic script includes 28 alphabets. As illustrated in Table.1, each alphabet can assume four to two shapes depending on its position in a word like the beginning, middle, and end or isolated. Hence, the position-based variability and the different writing styles of Arabic alphabets pose great challenges to automated character identification. Researchers have investigated various approaches for Arabic OCR by employing different techniques of preprocessing, features extraction and classification [2], e.g., recognition using segmentation [3], raw pixel data [4] and simple deep sparse auto encoder [4].

In particular, Shatnawi and Abdallah [6] have proposed a model to recognize characters with real-world distortions in Arabic handwriting using a dataset containing 48 examples of each Arabic handwritten character (i.e., 28 letters and 10 digits) obtained from 48 different writers. The model in [6] achieved a recognition rate of 73.4%. Elzobi et al. [7] have employed the Gabor wavelet transform to extract the mean and standard deviation of image. Classification stage was carried out by employing a Support Vectors Machine (SVM) classifier. The IESK-arDB [8] and IFN/ENIT [9] datasets were used to evaluate the proposed approach. The authors reported an average 71% recognition rate. But, the proposed scheme [7] suffers from high memory as well as run time requirements. Sahlol and Suen [10] have investigated several pre-processing schemes with various features for Arabic handwritten character recognition. The presented system in [10] was trained and tested with artificial neural network (ANN) on ENPRMI dataset. The reported results indicate that the system was capable to identify 88% of the test set correctly. Maqqor et al. [11] designed a system for handwritten Arabic texts utilizing sliding window technique for feature extraction with multiple classifiers. The evaluation of their model was performed on text images of IFN/ENIT database that achieved a recognition rate of 76.54%. From all of the preceding observations, it is inferred that Automatic recognition of cursive Arabic handwritten characters has comparatively received less attention, and hence, it is taken up as the main subject of this study.

This paper presents a framework for cursive Arabic handwritten characters using multi-scales Histogram Oriented Gradient (HOG) descriptor for feature extraction and Deep rule based (DRB) classifier for classification. Empirical test results on the publicly available cursive Arabic Handwritten Characters Database (AHCD) containing 16800 characters are more promising and comparable than the existing methods in Arabic handwritten optical characters recognition.

The remainder of this article is organized as follow: section 2 provides a brief description of the main Arabic script characteristics. The architecture of the proposed system, with details of various processing steps such as feature extraction and classification, is presented in sections 3. Experimental dataset, results and analysis are discussed in section 4. In section 5, conclusions are outlined.

Table.1. Arabic alphabet shapes

No.	Name	Isolated	Connected		
			Beginning	Middle	End
1	Alif	ا	ا	ـا	ـا
2	Baa	ب	بـ	ـبـ	ـب
3	Taa	ت	تـ	ـتـ	ـت
4	Thaa	ث	ثـ	ـثـ	ـث
5	Jeem	ج	جـ	ـجـ	ـج
6	Haa	ح	حـ	ـحـ	ـح
7	Khaa	خ	خـ	ـخـ	ـخ
8	Daal	د	د	ـد	ـد
9	Thal	ذ	ذ	ـذ	ـذ
10	Raa	ر	ر	ـر	ـر
11	Zaa	ز	ز	ـز	ـز
12	Seen	س	سـ	ـسـ	ـس
13	Sheen	ش	شـ	ـشـ	ـش
14	Saad	ص	صـ	ـصـ	ـص

15	Dhad	ض	ضـ	ـضـ	ـض
16	Tta	ط	طـ	ـطـ	ـط
17	Dha	ظ	ظـ	ـظـ	ـظ
18	Ain	ع	عـ	ـعـ	ـع
19	Ghain	غ	غـ	ـغـ	ـغ
20	Faa	ف	فـ	ـفـ	ـف
21	Qaf	ق	قـ	ـقـ	ـق
22	Kaaf	ك	كـ	ـكـ	ـك
23	Lam	ل	لـ	ـلـ	ـل
24	Meem	م	مـ	ـمـ	ـم
25	Noon	ن	نـ	ـنـ	ـن
26	Heh	ه	هـ	ـهـ	ـه
27	Waaw	و	و	ـو	ـو
28	Yaa	ي	يـ	ـيـ	ـي

2. ARABIC WRITING CARACTERISTICS

It is estimated that Arabic language is used by 1.8 billion people. In fact, Arabic language is celebrated as a universal language on December 18 of each year [12]. Arabic writing is known for/by the beauty and the diversity of its styles, it is semi-cursive in both forms: printed and handwritten. This language is written/read from right to left. The Arabic writing alphabet has several different traits, thereby making it a unique language. For instance, Arabic alphabet consists of 28 basic letters with only 3 vowels (ا.ي.و). These letters change their shape according to their position in the word, which is also represented in Table.1. Moreover, the notion of an uppercase or lowercase letter does not exist in Arabic language, thus the writing is unicameral.

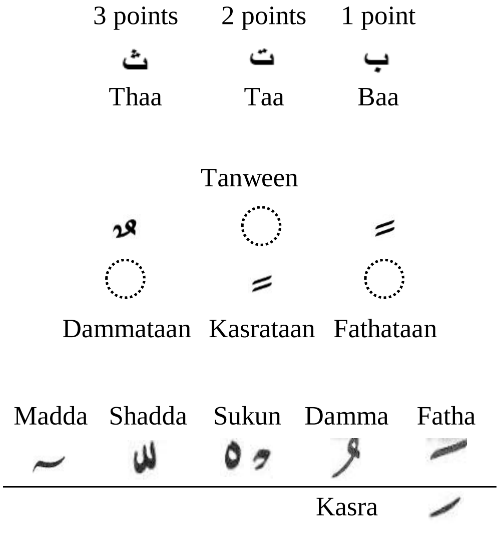

Fig.1. Arabic diacritic marks [12]

The Arabic writing is rich with diacritic marks (i.e., points/dots, fatha, damma, sukun, madda, kasra, shadda and tanween) that is represented in the Fig.1. It is worth noting that 15 letters of Arabic have one or more dots [13]. The Arabic characters can be joined either from right side, left side or from both sides. However, six of these characters (ا.د.ذ.ر.ز.و) are not connectable with their successors in a word, which cause a separation of the word into parts or sub-words [13] as show Fig.2. In Arabic writing, the same character or the same word can be written with different styles/ways and sizes by different writers or even by the same writer [14] as shown in Fig.3.

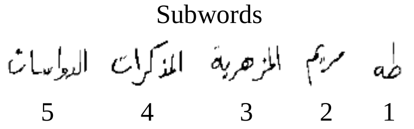

Fig.2. Words with different sub-words [13]

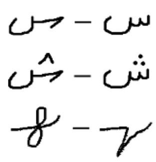

Fig.3. Characters written in different ways [14]

3. PROPOSED ARABIC HANDWRITTEN CHARACTERS RECOGNITION FRAMEWORK

Arabic language is used as a first language by 260 million people. Also, it is an official language for 26 countries. Automated Arabic language script (especially cursive handwriting) recognition is a challenging task, as different ethnic backgrounds and respective writers are influenced by their location and other languages if the country has more than one official language. Moreover, the complexity in Arabic character recognition is increased not only by the fact that Arabic characters can have more than one shape depending on their position (i.e., initial, middle, end or isolated) but also the writing strokes that tend to be different because it is, contrary to Latin, written from right to left side. This work aims at the recognition Arabic normal as well as cursive handwritten characters using multi-scale features and deep multi-layer rule-based classifier.

The keynote of the proposed framework is obtaining a discriminant features set along with an efficacious classifier that gives the probability of the image realism. The proposed method, as illustrated also in Fig.4, is composed of four main stages: character image acquisition (dataset acquisition), image pre-processing operations, multi-scale descriptive Histogram of Oriented Gradient (HOG) features extraction, and multi-layer deep rule based (DRB) classification. In the following section, these steps are described in detail.

3.1 IMAGE PRE-PROCESSING STEP

In order to facilitate the feature extraction process and increase the classification rate, image pre-processing is an important phase. It consists of reducing noise over the data and attempting to keep the significant information of the character form. The image pre-processing phase in work is made of three different operations. The first one is to convert the grayscale images into binary images by minimizing the inter-class variance of the white and black pixels. The next is to detect and delete all pixels that do not belong to the character shape, thereby representing noises. The next process is the normalization phase, which brings the character images to standard sizes and reduces all types of variations. The character normalization process in this study is composed of standardization of the size, correction of the inclination of the lines (skew correction), correction of the slant of the character (slant correction) and estimation of the baseline that aligns the character.

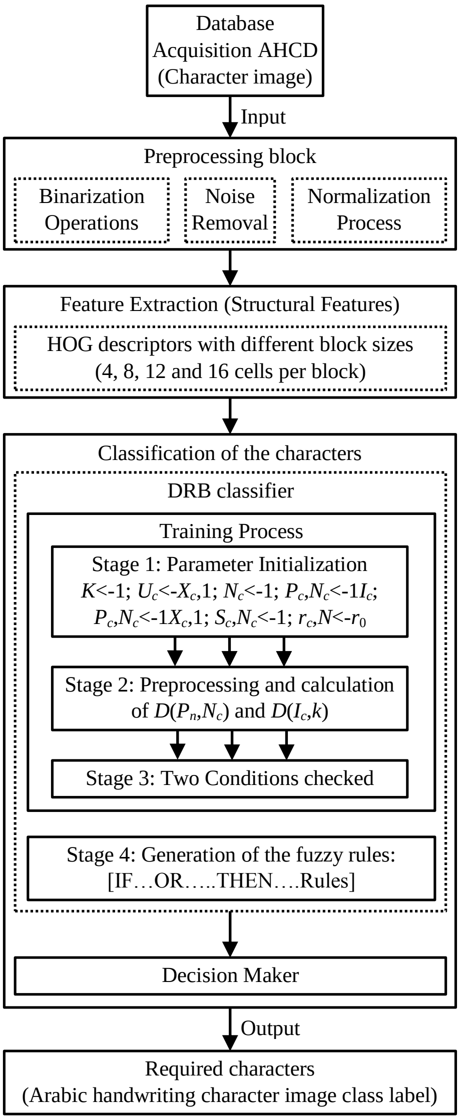

Fig.4. Architecture of proposed recognition system

3.2 FEATURE EXTRACTION

The quality and the relevance of the extracted features are very important for a good performance of a recognition system. In this study, we chose the Histogram of oriented gradient (HOG) descriptor, one of successful descriptors that is widely being used for object detection and recognition.

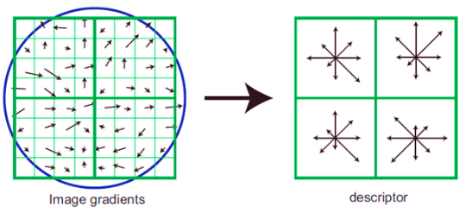

Fig.5. Histogram of gradients (HOG)

The HOG descriptor can be computed by decomposing the original image into smaller connected zones, called cells. For each block inside the cell, a 1-D histogram of edge orientations or gradient directions for the pixels is obtained as shown in Fig.5. The combination or concatenation of all these histograms represents the features [15].

To create the histogram of orientations, the possible orientations in boxes are divided in 0° to 180° degrees or 0°-360° degrees and the value of each box is the count of the number of gradients that fall in each box. Each gradient of the cell votes for the closest orientation by contributing to its orientation. After histograms creation, to account for changes in illumination and contrast, the gradient strengths must be locally normalized that requires grouping the cells together into larger, spatially connected blocks. The HOG descriptor is then applied on the concatenated vector of the components of the normalized cell histograms from all of the block regions. These blocks typically overlap, thus the same cell participates several times in the final descriptor [16], as illustrated in Fig.6. In this work, we propose to employ multiple cells with different scales/sizes and resolutions in order to capture eminent features. Specifically, we chose cells of 4, 8, 12 and 16 sizes to give more chances for the classifier to identify and recognize a maximum number of Arabic characters.

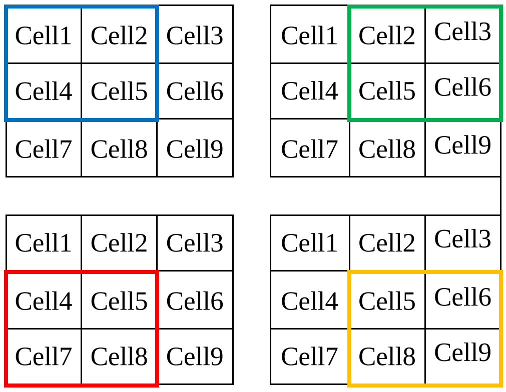

Fig.6. Grouping cells into units/blocks (an example of 4 cells per block)

3.3 CLASSIFICATION METHOD

After the extracting the discriminate and pertinent characteristics of each image, the different character classes and decision of success or rejection labels is obtained using multi-layer deep rule based (DRB) classifier.

DRB Training Architecture: The DRB classification scheme is based on main four sub-stages [5].

3.3.1 Initialization Stage:

In this stage, the classifier start with a vector normalization operation $X_{c,1}$ that is applied to the global feature vector of the first image $I_{c,1}$ of the class c, with an initialization of the different meta-parameters of the system:

$$k \leftarrow 1;\ U_c \leftarrow X_{c,1};\ N_c \leftarrow 1;\ P_{c,Nc} \leftarrow I_{c,1};\ P_{c,Nc} \leftarrow X_{c,1};\ S_{c,Nc} \leftarrow 1;\ r_{c,N} \leftarrow r_0,$$

where k is the current time instance, U_c is the global mean of all the observed data of the class c, Nc is the number of prototypes, $P_{c,Nc}$ prototypes of the class c, $S_{c,Nc}$ $n$br of images of the data cloud and r is the radius of the area of the data cloud.

3.3.2 Preparation Stage:

For all images of all the classes I_c, k, X_c, k is the normalization vector applied to its corresponding feature vectors, the global mean U_c is updated with the following instruction:

$$U_c \leftarrow \frac{k-1}{k} U_c + \frac{1}{k} \bar{x}_{c,k} \tag{1}$$

Then the density of all the prototypes $P_{c,Nc}$ existing and of the new image I_c, k denoted by: $D(P_{c,Nc})$ and $D(I_{c,k})$

3.3.3 Updating Stage:

From the U_c, $D(P_{c,Nc})$ and $D(I_{c,k})$ calculated previously, two conditions are identified:

IF $[D(I_{c,k}) > \max D(P_{c,Nc})]$ OR $[D(I_{c,k}) < \min D(P_{c,Nc})]$ THEN $(I_{c,k})$ is a new prototype and initializes a new data cloud. However, this condition is not verified $P_{c,n}$ the nearest prototype $I_{c,k}$ is calculated, where:

$$P_{c,n} = \arg\min\left(\left\|\bar{x}_{c,k} - p_{c,j}\right\|\right), \ j = 1, 2, ..., N_c \tag{2}$$

where $j = 1, 2, \ldots, N_c$

IF $[\|x_{c,k}\text{-}p_{c;n}\| < r_c, N_c]$ THEN $[I_{c,k}$ is assigned to $p_{c;n}]$ ELSE $[I_{c,k}$ is out of the area of $p_{c;n}]$ and new data cloud is added. The next image is grabbed at stage 2 (preparation stage) with the processing of all the training samples.

3.3.4 Fuzzy Rules Generation Stage:

In this last stage, the system generate one Anaya rule based on the identified prototypes

$$R: IF\left(I \sim P_{c,1}\right) OR...OR\left(I \sim P_{c,Nc}\right) THEN\left(class\, c\right) \tag{3}$$

where '~' denotes similarity, I, P and N_c represent a particular image with its corresponding vector feature vector x, prototype, and the number of prototypes of the c^{th} class, respectively.

In Table.2 we show the visualization of some examples of fuzzy rules that were built upon the prototypes given in Fig.7. These prototypes were identified after the training process for the following characters: "ا", "ج", "س", "م" and "هـ".

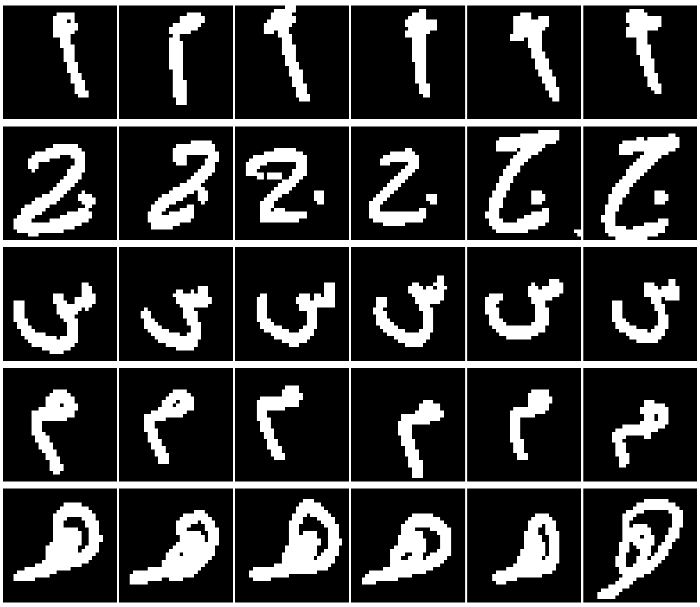

Fig.7. Identified prototypes

Table.2. Examples of fuzzy rules generated for some characters of the AHCD dataset

Fuzzy rules							
IF (I~	OR (I~	OR (I~	OR (I~	OR…	OR (I~	OR (I~	THEN (character Alif ا)
IF (I~	OR (I~	OR (I~	OR (I~	OR…	OR (I~	OR (I~	THEN (character Jeem "ج")
IF(I~	OR (I~	OR (I~	OR (I~	OR...	OR (I~	OR (I~	THEN (character seen "س")
IF (I~	OR (I~	OR (I~	OR (I~	OR...	OR (I~	OR (I~	THEN (character meem "م")
IF (I~	OR (I~	OR (I~	OR (I~	OR…	OR (I~	OR (I~	THEN (character Heh "هـ")

4. EXPERIMENTS AND RESULTS

In this section, we present the experimental analyses of the proposed Arabic handwritten characters recognition framework.

4.1 DATASET

The images from AHCD were utilized in this work to test the proposed method.

AHCD is a publicly available Arabic characters images database collected by Ahmed and Hazim [18] to cover all shapes of Arabic letters (28 letters). The AHCD database was collected from 60 writers such that each participant wrote each character (28 character) ten times on two forms. This database is composed of 16800 characters, partitioned into two sets:

- 80% training → 13440 characters to 480 images per class.
- 20% testing →3360 characters to 120 images per class.

4.2 DISCUSSION OF THE RESULTS AND COMPARATIVE STUDY

To evaluate the efficacy of the proposed Arabic character recognition system, the experiments were constructed and tested using MATLAB R2017a with the following hardware specifications: Windows 7, 64 bits Processor Intel (R) core (TM) i3-3214U and CPU @ 1.80GHz, RAM (4 GB).

The Table.3 illustrates the rate and the number of missed and correct classification on the testing dataset (3360 images), together with the total number and average of correct and miss-classification. It can be observed in Table.3 that handwritten Arabic characters are very hard to be recognized, as the miss-classification average of their classes is generally higher. The higher misclassification rates are usually caused by the similarity between characters in strokes, dots and structures, such as Taa ت and Thaa ث (because of the dots), jeem ج and haa ح khaa خ, Daal د and Thal ذ, Raa ر and Zaa ز, seen س and sheen ش, sad ص and dhad ض, Tta ط and dha ظ, Ainع and ghain غ, and faa ف and Qaf ق with a total mis-classification rate of 25, 38%. However, with the other characters, the proposed method achieved smaller average of 2.5%, 6.7%, 7.5%, 4.2%, 18.3%, for class 1, 2, 23, 24 and 26, which shows the recognition ability of proposed Arabic handwritten characters classification scheme.

Table.3. Number of correct and wrong recognition and the rate of correct-classification and misclassification

The Character	Class	Correct Classification	Number of correct Classification	Miss classification	Number of missed classification
Alif - ا-	1	97.5%	117	2.5%	3
Baa - ب-	2	93.3%	112	6.7%	8
Taa - ت-	3	66.6%	80	33.3%	40
Thaa -ث-	4	68.3%	82	31.7%	38
Jeem -ج-	5	75%	90	25%	30
Haa - ح-	6	69.2%	83	30.8%	37
Khaa -خ-	7	71.7%	86	28.3%	34
Daal - د-	8	75.8%	91	24.2%	29
Thal - ذ-	9	70.8%	85	29.2%	35
Raa - ر-	10	80%	96	20%	24
Zaa - ز-	11	72.5%	87	27.5%	33
Seen - س-	12	80.8%	97	19.2%	23
Sheen -ش-	13	77.5%	93	22.5%	27
Sad - ص-	14	75%	90	25%	30
Dhad -ض-	15	62.5%	75	37.5%	45
Tta - ط-	16	70.8%	85	29.2%	35
Dha - ظ-	17	70%	84	30%	36
Ain - ع-	18	67.5%	81	32.5%	39
Ghain -غ-	19	70.8%	85	29.2%	35
Faa - ف-	20	50%	60	50%	60
Qaf - ق-	21	55%	66	45%	54
Kaaf - ك-	22	80.8%	97	19.2%	23
Laam -ل-	23	92.5%	111	7.5%	9
Meem -م-	24	95.8%	115	4.2%	5
Noon - ن-	25	64.2%	77	35.8%	43
Heh - ه-	26	81.7%	98	18.3%	22
Waw -و-	27	74.2%	89	25.8%	31
Yaa -ي-	28	79.2%	95	20.8%	25
The total number of correct classification = 2507					
The total number of miss classification = 853					
Miss-classification rate = 25.38%					

The comparative results with prior method are reported in Table.4. In particular, Table.4 shows the results of recognition rates obtained by the proposed system, various recognition offline systems and different existing techniques. From Table.4, it is apparent that the performance of the proposed framework is competitive with the other prior systems. For example, Shatnawi and Abdallah [6] presented a model based on real distortions in Arabic handwriting for real handwritten character examples to recognize characters. They used these distortion models to synthesize handwritten examples that are more realistic. The dataset used in these experiments contains 48 examples of each Arabic handwritten character class (28 letters and10 digits) obtained from 48 different writers. They achieve a recognition rate of 73.4%, while our method attained 76.41%. Moreover, Elzobi et al. [7] have employed the Gabor wavelet transform to extract the mean and standard deviation of image with SVM classifier. Their method involved higher memory and run time. The method in [7] obtained an average 71% of recognition rate. Similarly, Sahlol and Suen [10] investigated a method for handwritten Arabic characters recognition based on pre-processing and artificial neural network (ANN) to finally achieve 88% accuracy. Maqqor et al. [11] have designed a recognition system based on sliding window technique for features extraction and a combination of the multiple HMMs classifiers to give a recognition accuracy of 76.54%.

From Table.3 and Table.4, it is obvious that the performance of the proposed system is very competitive and acceptable. Hence, we can state that the experimental results demonstrate the potency and validity of our proposed scheme.

Table.4. Recognition results of the proposed framework and existing systems

Authors	Databases	Models/ Techniques	Recognition Rate
Proposed system	Arabic Handwritten characters (AHCD)	Multi-scale HOG features and multi-layer DRB classification technique	74.61%
Shatnawi and Ab dallah [6]	Private dataset	A real distortion in Arabic handwriting using real handwritten character examples to recognize characters	73.4%

Elzobi et al. [7]	IESK-arDB [8] and IFN/ENIT [9]	The Gabor wavelet transform and SVM for classification	71%
Sahlol and Suen [10]	ENPRMI dataset	New preprocessing process with different features With artificial neural network (ANN)	88%
Maqqor et al. [11]	IFN/ENIT	A combination of Multiple Classifiers with Sliding window technique for feature extraction	76.54%

5. CONCLUSION

In this work, we have proposed an Arabic handwritten character recognition system using multi-scale Histogram of Oriented Gradient features and multi-layer Deep rule base classification scheme. Experimental results on publicly available AHCD dataset show that the proposed multi-scale and multi-resolution features with multi-layer Deep rule base classification scheme can outperform the existing state-of-the-art method. It is hoped that this work will open the door for applications of deep rule based classifier to the problems of Arabic script recognition. As a future work, we plan to devise novel attention based deep features as well as classification schemes that can also be useful for databases containing larger variations of Arabic handwritten characters and other types of data such as texts, words, etc.

REFERENCES

[1] Ahmed El Sawy, M. Loey and Hazem E.L. Bakry, “Arabic Handwritten Characters Recognition using Convolutional Neural Network”, *WSEAS Transactions on Computer Research*, Vol. 5, pp. 11-19, 2017.

[2] A. Lawgali, “Arabic Character Recognition: A Survey”, *International Journal of Signal Processing, Image Processing and Pattern Recognition*, Vol. 8, No 2, pp. 401-426, 2015.

[3] A. Cheung, M. Bennamoun and N.W. Bergmann, “An Arabic Optical Character Recognition System using Recognition-Based Segmentation”, *Pattern Recognition*, Vol. 34, No. 2, pp. 215-233, 2001.

[4] A. Goyal, K. Khandelwal and P. Keshri, “Optical Character Recognition for Handwritten Hindi”, CS229 Notes-Machine Learning, Stanford University, pp. 1-5, 2010.

[5] P.P. Angelov and G.U. Xiaowei, “Deep Rule-Based Classifier with Human-Level Performance and Characteristics”, *Information Sciences*, Vol. 463, pp. 196-213, 2018.

[6] M. Shatnawi and S. Abdallah, “Improving Handwritten Arabic Character Recognition by Modeling Human Handwriting Distortions”, *ACM Transactions on Asian and Low-Resource Language Information Processing*, Vol. 15, No. 1, pp. 1-12, 2015.

[7] M. Elzobi, A. Al Hamadi, Z. Al Aghbari, L. Dings and A. Saeed, “*Gabor Wavelet Recognition Approach for Off-Line Handwritten Arabic using Explicit Segmentation*”, Image Processing and Communications Challenges, Springer, pp. 245-254, 2014.

[8] M. Elzobi, Moftah, A. Al Hamadi and Z. Al Aghbari, “A Database for Handwritten Arabic and An Optimized Topological Segmentation Approach”, *International Journal on Document Analysis and Recognition*, Vol. 16, No. 3, pp. 295-308, 2013.

[9] M. Pechwitz, S.S. Maddouri, V. Margner, N. Ellouze and H. Amiri, “IFN/ENIT-Database of Handwritten Arabic 575 Words”, *Proceedings of International Symposium on Writing and Documents*, pp. 127-136, 2002.

[10] A. Sahlol and C. Suen, “A Novel Method for the Recognition of Isolated Handwritten Arabic Characters”, *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 1-13, 2014.

[11] A. Maqqor, A. Halli, K. Satori and H. Tairi, “Off-Line Recognition Handwriting Combination of Mutiple Classifiers”, *Proceedings of International Conference on Information Science and Technology*, pp. 1-12, 2014.

[12] Yasser M. Alginahi, “Arabic Character Segmentation: A Survey”, *International Journal on Document Analysis and Recognition*, Vol. 16, No. 2, pp. 105-126, 2013.

[13] K. Jumari and M.A. Ali, “A Comparative Evaluation of Selected Off-Line Arabic Handwritten Character Recognition Systems: A Survey”, *Jurnal Teknologi*, Vol. 36, No. 1, pp. 1-18, 2012.

[14] Ramzi A. Haraty, “Arabic Text Recognition”, *International Arab Journal of Information Technology*, Vol. 1, No. 2, pp. 156-163, 2004.

[15] S. Khorashadizadeh and A. Latif, “Arabic/Farsi Handwritten Digit Recognition using Histogram of Oriented Gradient and Chain Code Histogram”, *International Arab Journal of Information Technology*, Vol. 13, No. 4, pp. 1-13, 2016.

[16] Navneet Dalal and Bill Triggs, “Histograms of Oriented Gradients for Human Detection”, *Proceedings of International Conference on Computer Vision and Pattern Recognition*, pp. 886-893, 2005.