

Online Arabic Handwriting Characters Recognition using Deep Learning

Khalid Mohammed Musa Yaagoup^{1,2}, Mohamed Elhafiz Mustafa^{3,4}

Al Neelain University, Dept of Computer Science, Faculty of Computer Science and Information Technology, Sudan¹

Faculty of Computer Science and Information Technology, University of the Holy Quran and Islamic Sciences, Sudan²

College of Computer and Information Sciences, Jouf University, Sakaka, Kingdom of Saudi Arabia³

Faculty of Computer Science, Sudan University of Science and Technology, Khartoum, Sudan⁴

Abstract: Recent research shows that in a wide variety of fields, such as computer vision, speech recognition and natural language processing, deep learning has produced noticeably promising results. Automatic recognition of handwriting is a significant component for many applications in different fields. It is a complex subject that has gained a great deal of attention in the past three decades. Research has focused on the recognition of Latin languages' handwriting, fewer studies have been done for the Arabic language, a few problems still wait to be solved for Arabic handwritten characters. We presented a Convolutional Neural Network (CNN) model for the recognition of Arabic handwritten characters in this paper. The dataset is pre-processed before feeding it to the CNN model, it applied on database that contain 16800 of handwritten Arabic characters. The accuracy was raised to 96% as a test accuracy showing better results than other methods using the same database.

Keywords: Deep Learning, Convolutional Neural Network, Handwritten Characters, Pre-Processed.

I. INTRODUCTION

Recognition is a territory that covers different fields, for example, finger print recognition, image recognition, character recognition, numerals recognition, etc. The Handwritten Character Recognition System (HCR)[1] is a smart system capable of classifying handwritten characters as used by humans. Character classification[2, 3] is an important component of many computer vision and image issues, such as optical character classification. Recognition, recognition of license plates, etc. Due to the various handwriting types of the writers, the classification of handwritten characters is a more difficult task. In recent decades, handwritten Arabic character recognition (HACR) has attracted considerable interest. With the rapid development of deep learning algorithms, researchers have made major breakthroughs in this area. Arabic is a kind of Semitic language[4] used as the mother tongue of millions of people in countries in the Middle East. The Arabic alphabet characters consist of 28 basic letters and many letters look alike but are differentiated from one another by dots, these dots can be under or above the letter and they vary in the number, the number of the dots can be one, two, or three. The existence of dots and other strokes that differentiate similar characters make Arabic character recognition even more challenging[5] that illustrated in Fig. 1.

ص	ش	س	ز	ر	ذ	د	خ	ح	ج	ث	ت	ب	أ
sad	sheen	seen	zain	reh	thal	dal	khah	hah	jeem	theh	teh	beh	alef
ي	و	هـ	ن	م	ل	ك	ق	ف	غ	ع	ظ	ط	ض
yeh	waw	heh	noon	meem	lam	kaf	qaf	feh	ghain	ain	zah	tah	dad

Fig. 1 Arabic alphabet characters

Online Arabic handwriting recognition is an important technique for computer interface. It facilitates the data entry to small size portable computers (e.g., PDA, handheld PC, tablet laptop, and etc) which do not need for keyboards or keypads and use pen-based interfaces. Our study work we would like to contribute in the field of pattern recognition. The work aims to study the techniques of the Deep learning in order to apply them for the problem of online Arabic handwritten recognition. We intend to provide new techniques for this problem, and improve the previously known techniques for others.

Learning through deep models has recently gained considerable interest. Various deep learning architectures such as deep neural networks, deep convolutional neural networks, and recurrent neural networks have been applied to a wide range of fields like computer vision, automatic speech recognition, or natural language processing. The concept of deep learning is to discover the features in the input data in multiple levels of representation, in other words, higher level features represent more abstract semantics of the data. Deep learning techniques have been shown to produce state-of-the-art results on several difficult tasks [6-10]. For instance, computer vision tasks such as visual content-based image classification are a very challenging task, mainly due to intra-class variation, variation of size, variation of perspective, deformation or occlusion. Many attempts have also been made to recognize handwritten digits using convolutional machine learning techniques such as linear / nonlinear classifier, K-nearest neighbors, and SVMs on the popular MNIST dataset [11], or deep learning architectures such as deep neural networks and deep convolutional neural networks [12-14]. A recent application of machine learning for learning data representation is Deep Learning (DL). DL algorithms have taken the top place in the object recognition field due to the great performance improvement they have provided [15], [16]. It is based on algorithms using multilayer network such as deep neural networks, convolutional deep neural networks, deep belief networks, recurrent neural networks and stacked autoencoders. These algorithms make it possible for machines and computers to model our environment well enough to show intelligence. Autoencoders is an artificial neural network that is used to learn successful encoding where the input layer is equal to the output layer where the hidden layer is smaller [17,18]. The hidden layer provides a better representation of the input in the autoencoder than the original raw input, and the compression of the input data is often the hidden layer, which is the main feature of the input. A lot of training samples are required by CNN DL solutions, which place computational requirements on the system. However, the accelerating development and availability of low-cost computer hardware, high-speed networks and high-performance distributed computing applications has enabled the use of costly computing techniques. For example, Cecotti [19] used graphical processing units (GPUs) and high-performance clusters (HPCs) to use costly computational techniques to classify isolated characters from 9 databases.

There are several frameworks for Deep Learning. One of the most popular libraries is TensorFlow, that was released by Google in 2015 [20]. It is an open source code written in the programming language of C++ that can be used very well by GPUs. Keras [21], which is a higher-level API developed on top of TensorFlow, is another simpler application. For programming, Keras uses Python, which makes writing programs simpler than native TensorFlow codes.

The rest of the paper is structured as follows: a description of some of the related work done in the field is given in Section 2. Section 3 gives material and methods that we will discuss an Objectives of study, Dataset, Deep neural network model and Experimental setup. Section 4 provides an overview of the experiment and results, and in section 5 we list our conclusions.

II. RELATED WORK

Algorithms designed to recognize handwritten characters, primarily because of the variety of handwritten character shapes and types, are still less effective than those for printed characters. Recognition of Arabic characters is an significant problem, as it is a move that may be required in the more difficult problem of recognition of Arabic words or sentence [22]. Another difficult issue is character segmentation to break the word into characters. The character recognition problem is related to the simpler problem of Arabic numeral recognition which has recently attained great results [23].

Different methods have been proposed and high recognition rates are reported for the handwritten English and Chinese characters. In this section, however, we will present only the most competitive work related to solving the AHCR problem. An Optical Character Recognition (OCR) of Arabic Handwritten Characters was proposed using Hopfield's neural network [24]. They developed an OCR system and used eight Arabic letters to train their system. For eight Arabic letters with an effective recognition rate of 77.25 percent, they used a limited database.

Many algorithms in the past focused on finding structural features (such as the presence of loops, the orientation of curves, ...etc.) or statistical features (such as moments, histogram of gray level distribution, ...etc.) [25]. These features aim to optimize interclass variability while reducing intra-class variability and have been fed to the classifier. Some algorithms are known as segmentation-based recognition systems; their experimental results therefore depend on the word segment before the characters are recognized. In Al-abodi and Li [26], the IFN / ENIT database had been used to suggest a recognition system based on geometrical features of Arabic characteristics. The average recognition accuracy is 93.3%. Other works that used IFN/ENIT for segmentation-based character recognition such as [27] also achieved similar performance using three main modules: preprocessing, feature extraction and recognition. In this article, however, we will not discuss such a method. Although the IFN / ENIT database [28] is available, it is intended to classify words and involves letter segmentation before character recognition is carried out. In addition, it is considered to be small and does not contain sufficiently representative samples. This is why CNN DL architecture assessment was not found appropriate.

In 2014, Torki et al. [29] built their own database of about 9000 characters. They named the AlexU Isolated Alphabet (AIA9k) database. Then, they extracted three window-based gradient-based descriptors: Histogram of Oriented Gradients

(HOG) [30], Speeded-Up Robust Features (SURF) [31] and Scale Invariant Feature Transform (SIFT) [32]. In addition, two textured descriptors were extracted and 4 classifiers (Logistic regression, ANN, SVM-Linear and SVM-RBF) were evaluated on their database. Using SVM-RBF on SIFT features, the best achieved accuracy was 94.28%. The 75 characters that were misclassified are shown in Fig. 2. Although there are some characters that are not that hard to classify, some characters are actually confusing.



Fig. 2 Misclassified characters from the AIA9K dataset using the method of [29].

In 2015, Lawgali [33] published a survey about Arabic Character Recognition and none of the algorithms mentioned used deep learning. However, also in 2015, Elleuch [34] introduced an Arabic handwritten character recognition using Deep Belief Neural Networks. It does not require any feature engineering. The input is simply the raw data or the grayscale pixel value of the images.

The method was evaluated on a HACDB database[35] containing 6600 handwritten character shapes written by 50 persons. The dataset is divided into 5280 images in a training set and 1320 images in a test set. The result was promising on the character recognition task with 97.9% accuracy but dispiriting on the word recognition database with an accuracy of less than 60%.

Elleuch [36] continued to work on the DBN and stack of feature extractors such as Restricted Boltzmann Machine (RBM) and Auto-Encoder in 2017 and recorded character recognition results that were actually close to the previous work (97.8%). These are very positive outcomes and display the dominance of DL techniques in AHCR.

However, the HACDB database must be seen as a simple and clean database, and the key sections of the various letter types are well described in 66 different classes. On the other hand, it is harder to distinguish identical letters, which are only distinct by a dot, in character recognition. HACDB is much easier to classify as the AIA9k database and has three classes.

In 2017, El-Sawi et al. [37] collected the Arabic Handwritten Character Dataset (AHCD) of 16800 images of isolated characters. In order to train and test the dataset, they constructed a CNN Deep learning architecture. They used optimization methods to improve CNN 's results. Their proposed CNN gave an average 94.9% classification accuracy on testing data.

III. MATERIAL AND METHODS

A. Objectives

The proposed Model will have the following objectives:

- To take the arabic hand written characters as an input process the arabic character.
- Train the Model using Deep learning effectively to recognize the arabic character.
- To provide an ease of use interface for the users to draw the arabic characters with free hand.
- Model should be able to effectively predict the hand-written arabic characters in the presence of noise.
- To use Deep leaning model to enhancement the performance of Arabic handwritten character recognition.

Extract the most important features which represent the whole hand written Arabic character image. The key part of this research work is this process.



- Improve the accuracy of Arabic handwritten characters recognition using deep learning technology from more than (94.9%) of testing accuracy as in the database under study.
- Various techniques will be studied for extracting the features from the handwritten Arabic character. The goal is to develop an efficient extraction technique which leads to more accurate handwritten Arabic character recognition system.
- Various classifiers will be studied to measure the performance of the extracted features.

To solve the problem of latency in data processing, GPUs are used as suggested by Ciresan et al.[23], who trained and tested the CNN network using the Classifiers Committee and reduced the error rate of the MNIST dataset[24] to 2.7%. For this reason, as there is a GPU-enabled TensorFlow with support for CUDA acceleration, we decided to choose Keras and TensorFlow as the development environment. The experiment was performed on a 2.2 GHz Core(TM) i7-2670QM CPU with 8G memory and GPU NVIDIA GeForce GT 525M running on windows, a neural network has increased due to the availability of GPU's. Other cloud based services like Google Cloud platform Web Services which provide resources was performed to train a proposed CNN.

B. Dataset:

To get a good result, convolutional neural network needs a large training data of Arabic handwritten character images, so we train dataset (AHCD) consisting of 16,800 characters written by 60 participants, the age range is between 19 and 40 years, and 90 % of participants are right-handed. Each participant wrote each character (from 'alef' to 'yeh') ten times on two forms as shown in Fig.3.

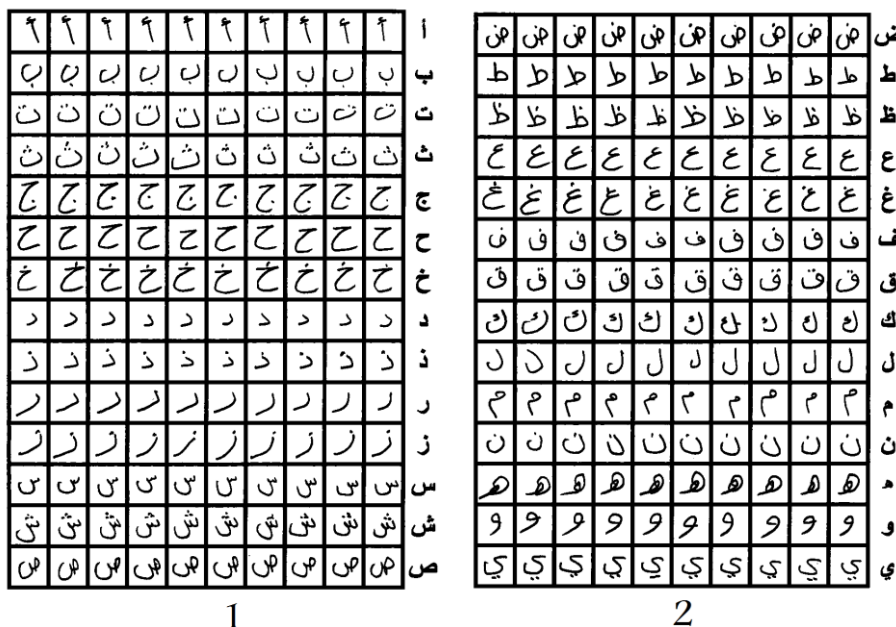


Fig. 3 Data collection for Arabic characters

The database is partitioned into two sets: a training set (13,440 characters to 480 images per class) as 80% from total, and a test set (3,360 characters to 120 images per class) as 20% from total. The training set and test set writers are exclusive. Orders for the inclusion of test set writers are randomized to ensure that test set writers are not from a single institution (to ensure test set variability). Creating the proposed database presents more challenges because it deals with many issues such as style of writing, thickness, dots number and position. Though written in the same place, some characters have distinct shapes. For example the character has different shapes in isolated position. The database is available for free at our google drive for researchers[38].

C. Deep neural network model(Convolutional Neural Network (CNN))

The neural network used for Arabic character recognition is outlined in this section. Any pattern recognition method usually involves three steps: preprocessing, feature extraction, and classification [39]. There is some way to preprocess the data using cleaning, allowing it to be transferred into the next step, feature extraction, where relevant information is extracted that helps in classification. Feature extraction is a challenging process that typically takes time and does not process raw visual data. The third step is classification, where either the manually extracted features are used to classify

the image into a specific class using an appropriate classifier [40], or some form of automatic feature extraction may be employed, such as CNNs [41].

CNN contains convolutional layers, pooling layers, and entirely related layers. On the input image, convolutional layers perform feature extraction, extracting features.. Our model architecture is composed of: input layer, four hidden layers and output layer. CNN are describe as following:

1) *Input layer:*

The input layer is an $H \times W \times D$ pixel image, where H is the height, W is the width, and D is the depth, in pixels. There is $D = 3$ on a colored RGB image, while $D = 1$ on a grayscale image. Our model has as input a $112 \times 112 \times 1$ grayscale image for convolutional layers as a first convolutional layer called input layer. after it performs (activation function using ReLU followed by Max pooling layer).

2) *Hidden layers:*

Our model composed consist of four hidden layers:

- First three layers are called convolutional layers, performs (activation function using ReLU followed by Max pooling layer) after each one.
- Convolutional layers (as a fifth convolutional layer).

After that doing flattening layer, flattening the model for fully connected layer.

Before output layer perform Fully connected layers or Dense layer by 512 neurons, and dowing ReLU as an activation function.

3) *Output layer:*

Output layer (Dense layer) as a number of the classes that we have according to our data set is 28 classes, which means that we have 28 output neurons, each represent one Alphabet characters from the dataset, after that doing Softmax as an activation function.

The model is constructed using the sequential model of the Keras library under tensorflow environment using python programming language. The input layer receives an images of size $(112 \times 112 \times 1)$, which means that the image is gray scaled and has height and width of 112 pixel. The raw pixel data of the character image will therefore be retained by the input layer.. The convolutional layer can perform a set of operations that result in a single value. We started the width of the convolutional layer by 112 and applied ReLU as an activation function for introducing nonlinearities into the model. We used the pooling layer to down sample the 56 by 56 image data to reduce the features' dimension so as to increase the performance of our model. Max pooling algorithm is used in the pooling layer to extract a 2×2 -pixel tiles from the feature map and return the maximum values while discarding other values. The role of our model's fully connected layer or dense layer is to perform the actual classification using the features extracted by the convolutional layer and sampled down by the max pooling algorithm. The number of the classes that we have according to our data set is 28 classes, which means that we have 28 output neurons, each represent one Alphabet characters from the dataset. The architecture of our proposed Model (CNN) is depicted in Fig. 4.

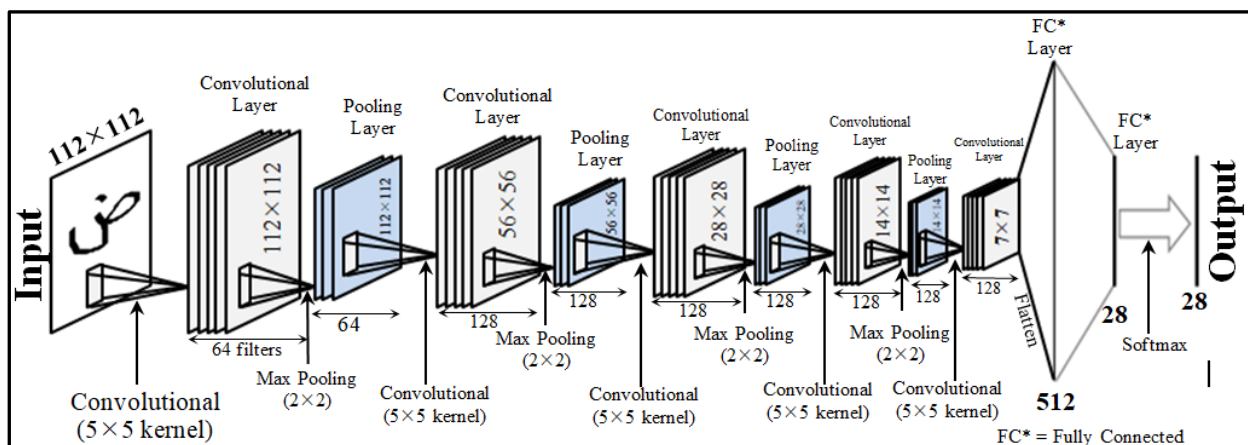


Fig. 4 The architecture of our proposed Model (CNN)

4) *Experimental setup*

We evaluate the performance of our proposed CNN model using the following measures:

- *Recall (R)*: is the proportion of the correctly classified images over the total number of images belonging to class x :

$$R = \frac{TP}{TP + FN} \quad (1)$$

– Precision (P): is the percentage of images that are classified correctly over the total number of classified images.:

$$P = \frac{TP}{TP + FP} \quad (2)$$

– F1 measure: That's a measure that includes Recall and Precision:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (3)$$

TP (true positive) is the total number of images that can be correctly labeled as belonging to class x , FP (false positive) is the total number of images that have been wrongly labeled as belonging to class x , FN (false negative) is the total number of images that have been incorrectly labeled as not belonging to class x , and TN (true negative) is the total number of images that have been incorrectly labeled as belonging to class x .

IV. RESULTS AND DISCUSSION

The results obtained by the proposed model (CNN) are incorporated in this section. To classify AHCD datasets, we will define the effects of applying the network. We will equate the findings obtained with those of other methods using the proposed method.

The size of the dataset is 16800. These groups are training and testing, separated into groups. The training data reflects 80 percent of the total data, which is 13440, while the test data represents for 20 percent, which is 3360. Prediction accuracy is used as a measure of performance. The accuracy of the test obtained is 95.7 percent, the miss-classification rate in Table 1 is 4.3 percent.

Table 1 Miss-Classification & correct-Classification rate & number of incorrect and correct recognition on testing data

Class	Character	Miss Classification	# of Missed	Correct Classification	# of Correct
0	alef "أ"	0.0%	0	100%	120
1	beh "ب"	0.8%	1	99.2%	119
2	teh "ت"	6.7%	8	93.3%	112
3	theh "ث"	5.8%	7	94.2%	113
4	jeem "ج"	2.5%	3	97.5%	117
5	hah "ح"	4.2%	5	95.8%	115
6	khah "خ"	1.7%	2	98.3%	118
7	dal "د"	6.7%	8	93.3%	112
8	thal "ذ"	5.8%	7	94.2%	113
9	reh "ر"	5.0%	6	95%	114
0	zain "ز"	12.5%	15	87.5%	105
1	seen "س"	5.8%	7	94.2%	113
2	sheen "ش"	1.7%	2	98.3%	118
3	sad "ص"	1.7%	2	98.3%	118
4	dad "ض"	5.0%	6	95%	114
5	tah "ط"	5.8%	7	94.2%	113
6	zah "ظ"	3.3%	4	96.7%	116
7	ain "ع"	2.5%	3	97.5%	117
8	ghain "غ"	1.7%	2	98.3%	118
9	feh "ف"	3.3%	4	96.7%	116
0	qaf "ق"	5.8%	7	94.2%	113
1	kaf "ك"	2.5%	3	97.5%	117
2	lam "ل"	1.7%	2	98.3%	118
3	meem "م"	1.7%	2	98.3%	118
4	noon "ن"	8.3%	10	91.7%	110
5	heh "ه"	5.8%	7	94.2%	113
6	waw "و"	7.5%	9	92.5%	111
7	yeh "ي"	4.2%	5	95.8%	115
Miss-Classification Rate= 4.3%					
Total Number of Miss-Classification = 144					
Total Number of Correct Classification = 3216					

Table 1 displays the missing classification rate and the number of missed classifications in the testing dataset. It can be seen that we achieve a very exciting average miss-classification rates of 0.0%, 0.8%, for class 0, 1 respectively, and 1.7%, for class 6, 12, 13, 18, 22, 23. The total of miss-classification is 144. This shows the great potential and recognition ability of Our CNN to classify Arabic handwritten characters. On the other hand, for characters with a high rate of miss-classification, we found that they have some common characteristics, which are summarized as:

- Some characters are very confusing with the similar strokes of characters shown in Table 2. The tiny distinction of stroke structure bring challenges for some similar character pairs, such as (dal) and (thal). The difference of (dal) and (thal) is the dot that above character (thal).
- Some characters are very confusable with the structure of other characters. The Arabic character (zain) have a highest number of wrong classification 15. The (zain) character is easily confused by the four Arabic (dal), (thal) and (reh) characters.
- Some character strokes are missing from the character structure. Other characters which have the additional touching of the stroke or stroke shown in Fig.5.

The above three phenomena may be the main reason why characters of such kinds of Arabic characters are very hard to be recognized, this reflects the difficulties that children have in writing these letters. Besides, we also note that the average miss-classification of these classes is commonly higher.

Table 2: Arabic characters stroke similarity

Similar characters	Master stroke
ب، ت، ث	ب
ج، ح، خ	ح
د، ذ	د
ر، ز	ر
س، ش	س
ط، ظ	ط
ص، ض	ص
ع، غ	ع
ف، ق	ف
ل، ك	ل



Fig. 5 Misclassified samples of characters. Ground-truthprediction is the lower label. The misclassified samples are due to extra stroke, missing stroke, touching stroke. For a suggested plan to make a successful estimate, most of these samples are difficult.

We also evaluate the performance of our proposed CNN model using measurements (precision , recall, F1 score), a 96% accuracy, a 96% recall and a 96% F1 score. In Table 3, the detailed metrics are reported per character.

Table 3 Experimental results on the AHCD datasets

Character	Precision	Recall	F1-score	Support
أ	1.00	0.98	0.99	122
ب	0.99	0.97	0.98	123
ت	0.93	0.93	0.93	120
ث	0.94	0.97	0.96	116
ج	0.97	0.97	0.97	120

Character	Precision	Recall	F1-score	Support
ح	0.96	0.96	0.96	120
خ	0.98	0.98	0.98	120
د	0.93	0.97	0.95	116
ذ	0.94	0.91	0.93	124
ر	0.95	0.92	0.93	124
ز	0.88	0.91	0.89	116
س	0.94	1.00	0.97	113
ش	0.98	0.94	0.96	126
ص	0.98	0.93	0.96	127
ض	0.95	0.97	0.96	118
ط	0.94	0.96	0.95	118
ظ	0.97	0.92	0.94	126
ع	0.97	0.93	0.95	126
غ	0.98	1.00	0.99	118
ف	0.97	0.94	0.95	123
ق	0.94	0.94	0.94	120
ك	0.97	0.95	0.96	123
ل	0.98	0.98	0.98	120
م	0.98	1.00	0.99	118
ن	0.92	0.98	0.95	112
هـ	0.94	0.95	0.95	119
و	0.93	0.97	0.94	115
ي	0.96	0.98	0.97	117
Accuracy			0.96	3360
Macro avg	0.96	0.96	0.96	3360
Weighted avg	0.96	0.96	0.96	3360

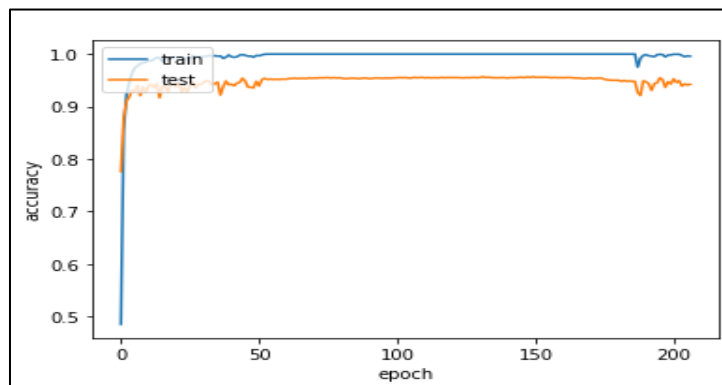


Fig. 6 Accuracy of our proposed Model (CNN)

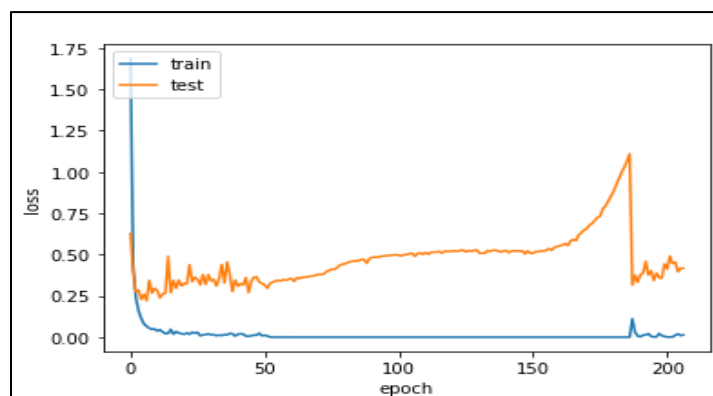


Fig. 7 Loss of our proposed Model (CNN)

Last, at the table. 4 showed the results obtained from CNN on the database we created and gathered. To get a good result, CNN needs huge training data, so we collect a large dataset of handwritten Arabic characters. To improve our CNN outcomes, we need more training data. If CNN were to use a small dataset such as [29,35], the rate of classification would not be good. It can be seen from Table.3 that the suggested method has a broad Arabic image character database and a good rate of classification (96 percent). The findings are better than those reported in the related work [24,26,29, 37], but the related work [35] has been trained and tested in a small database using its methods. It is often difficult to compare, however, because previous research has not experimented with large datasets. As shown in Fig 6 and Fig 7, the proposed method obtained 96 per cent accuracy on test data.

Table. 4 Comparison between Proposed Approach and Other Approach

Authors	Databases	Training Data & Testing Data	Accuracy
Hussien et al. [24]	Private dataset	Small database of eight Arabic letters	77.25%
Al-abodi and Li [26]	IFN/ENIT	60% Training images 40% Testing images	93.3%
Torki et al. [29]	AIA9k	9000 images	94.28%
A. Lawgali et al. [35]	HACDB	6600 images 5280 Training images 1320 Testing images	97.9%
A. El-Sawy. [37]	ACHD	16800 images 13440 Training images 3360 Testing images	94.9%
Our Approach	Our dataset(ACHD)	16800 images 13440 Training images 3360 Testing images	96%

V. CONCLUSION

Online handwriting recognition is a challenging problem, it is an active research area which always needs an improvement in accuracy. We present a review about Arabic character handwriting recognition and its importance. To recognize handwritten letters, we suggested a deep learning model. Our model achieved excellent results on the AHCD dataset, classification accuracy improved to 96% on testing dataset. In future work, we plan to study an Online arabic handwritten word recognition using deep learning architectures.

ACKNOWLEDGMENT

We are very grateful to our guide Mohamed Elhafiz Mustafa, Asst. Professor, Faculty of Computer Science, Sudan University of Science and Technology, for her valuable guidance and suggestions. We also wish to thank all our respected teachers and friends for their excellent contributions and support for the completion of this paper.

REFERENCES

- [1]. R. Sarkhel, N. Das, A.K. Saha, and M. Nasipuri, A multi-objective approach towards cost effective isolated handwritten Bangla character and digit recognition, *Pattern Recognition*, 58, pp. 172-189, 2016.
- [2]. M. Tounsi, I. Moalla, A.M. Alimi, and F. Lebougeois, Arabic characters recognition in natural scenes using sparse coding for feature representations, *Document Analysis and Recognition (ICDAR)*, 2015 13th International Conference on, pp. 1036-1040, 2015.
- [3]. D.K. Sahu and C.V. Jawahar, Unsupervised feature learning for optical character recognition, *Document Analysis and Recognition (ICDAR)*, 2015 13th International Conference on, pp. 1041-1045, 2015.
- [4]. M.N. Ibrahim, S.N. Bilmas, A. Babiker, and M. Idroas, A Framework of an Online Self-ased Learning for Teaching Arabic as Second Language (TASL), 2013 Fifth International Conference on Computational Intelligence, Modelling and Simulation, pp. 255-260, 2013.
- [5]. Murtada Khalafallah Elbashir, Mohamed Elhafiz Mustafa. "Convolutional Neural Network Model for Arabic Handwritten Characters Recognition". *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCCE)*, Volume 7, Issue 11, pages 1 – 5, 2018.
- [6]. LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86, no. 11, pp. 2278-2324, 1998.
- [7]. Ciresan, Dan, Ueli Meier, and Jürgen Schmidhuber. "Multi-column deep neural networks for image classification." In *Computer Vision and Pattern Recognition (CVPR)*, 2012 IEEE Conference on, pp. 3642-3649. IEEE, 2012.
- [8]. Ciresan, Dan Claudiu, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. "Deep, big, simple neural nets for handwritten digit recognition." *Neural computation* 22, no. 12, pp. 3207-3220, 2010.
- [9]. Deng, Li, and Dong Yu. "Deep learning: Methods and applications." *Foundations and Trends in Signal Processing* 7, no. 3–4, pp. 197-387, 2014.
- [10]. Deng, Li, Geoffrey Hinton, and Brian Kingsbury. "New types of deep neural network learning for speech recognition and related applications: An overview." In *Acoustics, Speech and Signal Processing (ICASSP)*, 2013 IEEE International Conference on, pp. 8599-8603. IEEE, 2013.
- [11]. MNIST. <http://yann.lecun.com/exdb/mnist/>

- [12]. LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86, no. 11, pp. 2278-2324, 1998.
- [13]. Cireşan, Dan, Ueli Meier, and Jürgen Schmidhuber. "Multi-column deep neural networks for image classification." In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 3642-3649. IEEE, 2012.
- [14]. Cireşan, Dan Claudiu, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. "Deep, big, simple neural nets for handwritten digit recognition." *Neural computation* 22, no. 12, pp. 3207-3220, 2010.
- [15]. G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Li and T. Hospedales, "When Face Recognition Meets with Deep Learning: An Evaluation of Convolutional Neural Networks for Face Recognition," *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 142–150, Santiago, Chile, 13–16 December 2015.
- [16]. K. Younis and A. Alkhateeb, "A New Implementation of Deep Neural Networks for Optical Character Recognition and Face Recognition," *Proc. of the New Trends in Information Technology (NTIT-2017)*, The University of Jordan, 25-27 April 2017.
- [17]. J. Maria, J. Amaro, G. Falcao, L. A. Alexandre, "Stacked Autoencoders Using Low-Power Accelerated Architectures for Object Recognition in Autonomous Systems," *Neural Processing Letters*, vol. 43, no. 2, pp. 445-458, 2016.
- [18]. Q. Xu and L. Zhang, "The effect of different hidden unit number of sparse autoencoder," *Proc. The 27th Chinese Control and Decision Conference (2015 CCDC)*, pp. 2464-2467, 2015.
- [19]. H. Cecotti, "Hierarchical K-nearest Neighbor with GPUs and a High-performance Cluster: Application to Handwritten Character Recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 31, no. 2, pp. 1–24, 2017.
- [20]. M. Abadi et al. "Tensorflow: Large-scale Machine Learning on Heterogeneous Distributed Systems," arXiv preprint arXiv:1603.04467, 2016.
- [21]. F. Chollet and Keras, GitHub Repository, [Online], Available: <https://github.com/fchollet/keras>, GitHub, 2015.
- [22]. M. Elleuch, N. Tagougui and M. Kherallah, "Arabic Handwritten Characters Recognition Using Deep Belief Neural Networks," *Proc. of the 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*, pp. 1-5, Mahdia, Tunisia, 16-19 March 2015.
- [23]. H. Alwzawy et al., "Handwritten Digit Recognition Using Convolutional Neural Networks," *International Journal of Innovative Research in Computer and Communication*, vol. 4, no. 2, 2016.
- [24]. R.S. Hussien, A.A. Elkhidir, and M.G. Elnourani, Optical Character Recognition of Arabic handwritten characters using Neural Network, *Proc. Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE)*, International Conference on, pp. 456-461, 2015.
- [25]. A. Lawgali, "A Survey on Arabic Character Recognition," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, no. 2, pp. 401-426, 2015.
- [26]. J. Al Abodi and X. Li, "An Effective Approach to Offline Arabic Handwriting Recognition," *Pattern Analysis and Applications*, vol. 40, no. 6, pp. 1883-1901, 2014.
- [27]. C. Boufenar, M. Batouche and M. Schoenauer, "An Artificial Immune System for Offline Isolated Handwritten Arabic Character Recognition. Evolving Systems," Springer-Verlag, pp.1-17, 2016.
- [28]. M. Pechwitz, S. S. Maddouri, V. Mrgner, N. Ellouze and H. Amiri, "Ifn/enit - database of Handwritten Arabic Words," *Colloque Inter. Francophone sur l'Écrit et le Document (CIFED)*, pp. 129–136, 2002.
- [29]. M. Torki et al., "Window-based Descriptors for Arabic Handwritten Alphabet Recognition: A Comparative Study on a Novel Dataset," arXiv preprint arXiv:1411.3519, 2014.
- [30]. N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886-893, 2005.
- [31]. H. Bay, A. Ess, T. Tuytelaars and L. van Gool, "Surf: Speeded-up Robust Features," *Journal of Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346-359, 2008.
- [32]. D. Lowe, "Distinctive Image Features F-ROM Scale-invariant Key Points," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.
- [33]. A. Lawgali, "A Survey on Arabic Character Recognition," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, no. 2, pp. 401-426, 2015.
- [34]. M. Elleuch, N. Tagougui and M. Kherallah, "Arabic Handwritten Characters Recognition Using Deep Belief Neural Networks," *Proc. of the 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*, pp. 1-5, Mahdia, Tunisia, 16-19 March 2015.
- [35]. A. Lawgali et al., "HACDB: Handwritten Arabic Characters' Database for Automatic Character Recognition," *European Workshop on Visual Information Processing (EUVIP)*, pp. 255-259, 2013.
- [36]. M. Elleuch et al., "Optimization of DBN using Regularization Methods Applied for Recognizing Arabic Handwritten Script," *International Conference on Computational Science (ICCS 2017)*, vol. 108, pp. 2292-2297, Zurich, 12-14 June 2017.
- [37]. A. El-Sawy, M. Loey and H. El-Bakry, "Arabic Handwritten Characters Recognition Using Convolutional Neural Network," *WSEAS Transactions on Computer Research*, vol. 5, pp. 11-19, 2017.
- [38]. <https://www.kaggle.com/mloey1/ahcd1>
- [39]. Impedovo S, Mangini FM, Barbuzzo D, A novel prototype generation technique for handwriting digit recognition. *Pattern Recognit* 47(3), pp. 1002–1010, 2014.
- [40]. Cruz RM, Cavalcanti GD, Ren TI, Handwritten digit recognition using multiple feature extraction techniques and classifier ensemble. In: 17th International conference on systems, signals and image processing, pp 215–218, 2010.
- [41]. Lauer F, Suen CY, Bloch G, A trainable feature extractor for handwritten digit recognition. *Pattern Recognit* 40(6), pp. 1816–1824, 2007.

BIOGRAPHY



Khalid Mohammed Musa Yaagoup: Lecturer in Faculty of Computer Science and Information Technology, University of the Holy Quran and Islamic Sciences, OCR, Deep learning.