IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 2, 2020, accepted June 17, 2020, date of publication June 22, 2020, date of current version July 7, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3004286

An Enhanced Offline Printed Arabic OCR Model Based on Bio-Inspired Fuzzy Classifier

SAAD MOHAMED DARWISH AND KHALED OSAMA ELZOGHALY
Institute of Graduate Studies and Research, Alexandria University, Alexandria 21526, Egypt
Corresponding author: Saad Mohamed Darwish (saad.darwish@alexu.edu.eg)

ABSTRACT In the recent few years, there was a concentrated search on Arabic Optical Character Recognition (OCR), especially the recognition of scanned, offline, machine-printed documents. However, Arabic OCR consequences are dissatisfying and are still a developed research area. Finding the best feature extraction techniques and selecting an appropriate classification algorithm lead to supreme recognition accuracy and low computational overhead. This paper presents a new Arabic OCR model by integrating both of Genetic Algorithm (GA) and the Fuzzy K-Nearest Neighbor classifier (F-KNN) in a unified framework to enhance the identification accuracy. GA is utilized as a feature selection algorithm that has better convergence and spread of solutions with candid variation preservation mechanism. The F-KNN algorithm is more appropriate to classify ambiguous or uncertain data objects in the sense that every object belongs to all classes with different degrees of membership. The suggested model semantically fuses bio-inspired based feature vectors with fuzzy KNN classifier to build accurate membership function for each class. Experimental results compared to other approaches revealed the effectiveness of the suggested model and demonstrated that the feature selection approach increased the identification accuracy process.

INDEX TERMS Arabic OCR, fuzzy classification, feature selection, GA.

I. INTRODUCTION

Optical character recognition is the automatic recognition of characters from images with lots of applications such as document recovery, zip code recognition, car plate recognition, and many banking and business applications. In general, OCR is divided into online and offline character recognition systems [1]. Online OCR recognizes characters as they are entered and utilizes the speed, direction of individual pen strokes and order to achieve a high level of identification accuracy in recognizing handwritten text. However, offline OCR is more complex. This kind of recognition must get over many complexities, such as similarities of distinct character shapes, interconnections of neighboring characters, and character overlaps. Although offline systems are less precise than online setups, they are broadly used in specialized applications such as interpreting handwritten postal addresses on envelopes and reading currency amounts on bank checks. Furthermore, offline OCR saves money, time, and has the ability to rewrite old and historical documents in electronic format [2] [3]. Consequently, obstacles encountering offline OCR, and the increasingly urgent need for OCR applications, make offline OCR an exhilarating field of research.

OCR system aims to accomplish a high recognition rate, overcome the poor quality of scanned images, particularly in historical documents, and adapt style and size variations within the same document. Regardless of other languages, Arabic OCR is still developing because of the complicated nature of Arabic words structure and syntax. Some of these complexities are that [3]: (1) every character has two or four shapes where the form of each letter relies on its location in the word as shown in Table 1. (2) The shape of some characters is similar, but difference arises with the position and dots number such as (ب ت ث), which can be written either above or below the characters. (3) The characters are written connected to each other. Yet, some characters cannot be accompanying to latter characters that cause a word to have many connected components; these are called Pieces of Arabic Words (PAWs). Moreover, special marks called diacritics, written above or below the character, are used to adjust the character accent (see Fig. 1).

The performance of the OCR relies on the quality of the input text, processing of text image, and the different classification techniques used to improve the identification

The associate editor coordinating the review of this manuscript and approving it for publication was Kathiravan Srinivasan.

TABLE 1. Arabic character forms.

Name	Isolated	Initial	Medial	Final
Alif	أ	---------		ا
Baa	ب	بم	بم	ب
Taa	ت	تح	تح	ت
Thaa	ث	ثن	ثن	ث
Jiim	ج	جم	جم	مج
Haa	ح	حم	حم	مح
Khaa	خ	خم	خم	مخ
Daal	د			د
Zaal	ذ	---------		ذ

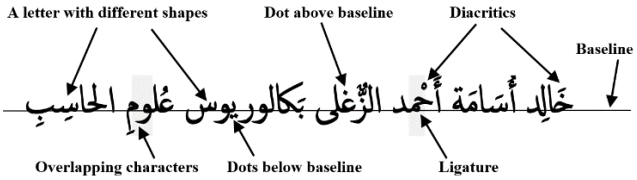

FIGURE 1. Printed Arabic script characteristics.

rate. Generally, the OCR system involves six stages: image acquisition (scanning), segmentation, preprocessing, feature extraction, classification, and post-processing, as shown in Fig. 2 [4]. The two major factors that affect the OCR recognition rate are: (1) a set of representative features from word images and (2) a robust classification algorithm [5]. The selection of a stable and representative set of features is the core of OCR system design. This operation selects the most important features of a word and joins them in a feature vector, yet simultaneously ignores the unimportant ones. OCR classification techniques can be broadly grouped into three categories [5], [6]: heuristic (e.g., fuzzy logic), template matching (e.g., dynamic time warping), and learning-based methods (e.g., neural networks). These algorithms, until now do, not reach a suitable and fitting consequence. With Arabic OCR as they are not generalized training data well and sensitive to common types of distortions.

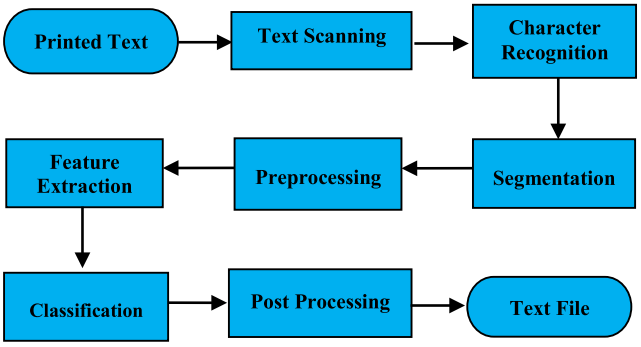

FIGURE 2. Optical character recognition steps.

Feature selection is the process of getting the most applicable inputs for a predictive model. These techniques can be used to recognize and ignore unnecessary, unimportant, and redundant features that do not participate or decrease the accuracy of the predictive model [6]. The Genetic Algorithm (GA) is one of the most advanced and strong algorithms for feature selection. This is a stochastic method for function optimization based on the mechanics of natural genetics and biological evolution. As mentioned and shown in this article, we try to clarify how genetic algorithms can be applied by selecting the most relevant features in order to optimize the performance of the predictive model.

There are many advantages of genetic algorithms over other optimization algorithms. Two of the most notable are the ability to deal with complex problems and parallelism. Genetic algorithms can deal with various types of optimization, whether the objective (fitness) function is stationary or non-stationary (changes with time), linear or nonlinear, continuous or discontinuous, or with random noise. Because multiple offsprings in a population act like independent agents, the population (or any subgroup) can explore the search space in many directions simultaneously. This feature makes it ideal for parallelizing the algorithms for implementation. GA, like all other random-search oriented optimization algorithms, does not require any information about the structure of the function to be optimized and uses it as Black Box. Classical optimization methods should use some information. The GA is a well-established and popular algorithm with recognition applications as it yields good optimization for "noisy" environments [7]–[9].

There are fuzzy classifier models inspired by the concept of "fuzzifying" conventional classifiers. A typical representative of this group is the K-Nearest Neighbor classifier (K-NN). In the classical K-NN, the object x is labelled as the majority of its K nearest neighbors in a reference data set. The approximations of the posterior probabilities for the classes are crude, given by the proportion of neighbors out of k voting for the respective class. Fuzzy K-NN uses the distances to the neighbors as well as their soft labels, if these are obtainable. Fuzzy k-nearest neighbor is based on that the pattern set is extended to a fuzzy set to assigns class membership to a pattern instead of assigning the pattern to a specific class [10]. In general, fuzzy algorithms are oftentimes powerful, in the sense that they are not very sensitive to changing environments and erroneous or forgotten rules. Furthermore, the reasoning process is often easy, in contrast with the precise computationally systems, so computing power is saved. This is a very interesting feature, particularly in real-time systems. However, there are some emerging problems regarding how to obtain near good Arabic OCR that overcomes the curious nature of Arabic characters, accomplishing a high- level of recognition rate and dealing with numerous font styles. All these challenges constitute the motivation of this research.

II. CONTRIBUTION

The cursive nature of the Arabic characters makes it more difficult to reach a high accuracy in character recognition since even printed Arabic characters are in cursive form.

The main contribution of this research is to establish a new Arabic OCR model that deals with a printed and segmentation-free approach for images of Arabic words by adopting both GA and Fuzzy KNN. GA is utilized in the feature selection process because of its capability to exploit accumulating information about an initially unknown search space in order to bias subsequent search into promising subspaces. Besides, the proposed model exploits fuzzy KNN as a classification algorithm to deal with Arabic characters' cursive nature. The ''fuzzification'' process guarantees voting from different samples belonging to more than one class, using the membership function that may be considered as weighted voting. The model aims to reach the least recognition error, the shortest running time, and the simplest structure.

As far as our knowledge, this is the first model that fuses both of well-known bio-inspired feature selection and fuzzy classification for Arabic OCR. The suggested model fuses the near good extracted features with a fuzzy KNN classifier by means of building a precise membership function for each class through features-related training data. Herein, the contribution extends to aspects that play a key role in refining fuzzy clustering, including the local search process of building the membership function that relies on some parameters for sample' features.

Current methods of OCR recognition, in general, depend on the use of extracted features as samples for the fuzzy classifier based on default membership functions and sample's distance from its KNN. Unlike these methods, the proposed model considers how to build the membership function of the fuzzy classifier based on the samples' features vectors to enhance classification accuracy. Herein, a histogram-based method is utilized to build memberships of those KNN in the possible classes. This is the first work in which a fuzzy classifier's membership function has been built based on the semantic fusion between two methods: histogram and fuzzy nearest neighbor to handle the cursive nature of Arabic words for recognition applications.

The structure of this paper is prepared as follows: a short survey about prior studies is discussed in Section II. In Section III, the proposed model is provided in detail. The experimental results that show the performance of the suggested model and the assessment are given in Section V. Then, the paper concludes with final remarks on the study and the future work in Section IV.

III. RELATED WORK

Research in the Arabic OCR domain has attracted tremendous interest in the past few years, mainly due to its challenging nature in electively satisfying both aims without degrading one another [11]–[14]. For example, the authors in [15] built an Arabic OCR system using Scale Invariant Feature Transform (SIFT) as features for classification of letters in conjunction with the online failure prediction method. The system scans every word with increasing window sizes; segmentation points are set where the classifier achieves maximal confidence. By exploiting the polymorphism of Arabic letters, one can accurately predict the correctness of the segmentation.

To highlight the influence of image descriptors, the research in [16] focuses on enhancing the extracted feature stage by selecting the efficient feature subsets using different feature selection techniques. These techniques ranked the 96 possible features based on their importance. The work proved that the NSGA selects the best subset of features compared to the other four methods. The system also concluded that the Support Vector Machine (SVM) classifier has the best classification accuracy.

The idea of the partial segmentation process has been utilized in [15] for identifying Arabic machine-printed texts using the Hausdorff distance. The stroke width transform was used to calculate the size and the font style to define a set of multi-size sliding windows to search and recognize characters within the given shape of a PAW. The process evaluates the likenesses of the two sub-images (character and sliding window) using Hausdorff distance. The system gave acceptable results of high-level recognition rate for the Arabic Printed Text Image (APTI) database and Printed Arabic Text Set A01 (PATS-A01) database. However, the process of time-consuming comes from increasing the number of sliding windows in every image. To handle the problem of word segmentation, the authors in [16] defined each shape of an Arabic word as a separate class, without word segmentation. The features extracted for every word consisted of twenty vertical sliding windows to get structural and geometrical representations of Arabic words. The last phase was the classification phase, where the multi-class SVM was applied. The system was examined using different datasets of Arabic words and reached a recognition rate of 98.5%.

As stated in [17], many works were introduced that utilizes fuzzy logic within Arabic OCR applications. In [18], some of these approaches, features are modeled by fuzzy linguistic variables, and fuzzy rules are then used for classification. A structural method for feature extraction is employed in another work, and then fuzzy relations for classification are introduced. Combining a fuzzy linguistic model and non-fuzzy one allows a simple qualitative representation of the feature knowledge for the Arabic characters. Other approaches [19], [20] were also reported in the literature.

Reference [21] listed an approach for the Arabic OCR using neural networks to classify features. This algorithm creates tokens that characterize the characters. The suggested method mainly depends on extracting a set of features for every character. It then provides all the extracted information to the recognition and assembly phases. However, the average recognition rate is only 87%. In [22], to resolve the problem and overcome the difficulty of Arabic handwriting recognition, the artificial neural network successfully applied, and the ANN obtained 99.62% to the percentage for recognition using handwriting database.

In recent years, deep learning has been received much attention from researchers [23], [24]. The authors in [24] deployed a deep learning approach based on

Multi-Dimensional Long Short-Term Memory (MDLSTM) networks and Connectionist Temporal Classification (CTC). The MDLSTM has the advantage of scanning the Arabic text-lines in all directions (horizontal and vertical) to cover dots, diacritics, strokes, and fine inflammation. However, the application of the deep neural network is facing some difficulties, including hyper-parameter tuning is non-trivial, needs a big dataset for proper training, still a black box, and is comparatively slow.

In [23], the authors presented a generic OCR system based on deep Siamese convolution neural networks (CNNs) and support vector machines (SVM). Supervised deep CNNs achieve a high level of accuracy in classification tasks. However, fine-tuning a trained model for a new set of classes requires a large amount of data to overcome the problem of dataset bias. The classification accuracy of deep neural networks (DNNs) degrades when the available dataset is insufficient. Moreover, using a trained deep neural network in classifying a new class requires tuning the network architecture and retraining the model. Our proposed model handles all these limitations. The deep Siamese CNN is trained for extracting discriminative features. The training is performed once using a group of classes. The OCR system is then used for recognizing different classes without retraining or fine-tuning the deep Siamese CNN model. Only a few samples are needed from any target class for classification.

From the survey conducted, it has been inferred that the current methods for Arabic OCR rely on hard classifiers for classification. The current fuzzy rule-based classification methods mainly have low accuracy. Furthermore, the semantic association between Arabic OCR features and classifiers is missed. Different from the existing methods, the suggested model relies on a fuzzy classifier to deal with the vagueness of the extracted features not by using feature's fuzzification but using features to build the fuzzy membership function of each class. This is done to deal with the major limitation in selecting the optimal distinctive features from different words; which, despite the use of the genetic algorithm to extract it, we often do not get the optimal features.

IV. PROPOSED MODEL

The block diagram that summarizes the main components of the proposed Arabic OCR model is depicted in Fig. 3. The model utilizes GA to select the optimal features and fuzzy KNN Classifier for recognition of off-line Arabic characters without assigning a hard-crisp membership for each class. The system contains two main phases: training and testing phases. The following subsections discuss the system's components in detail with the clarification of each step's objective.

A. IMAGE ACQUISITION

Although there are many common Arabic databases, the proposed model used well-researched printed databases that have a high-quality resolution, many sizes, and font styles, Firstly PATS-A01 database [25] contains 2766 text line images

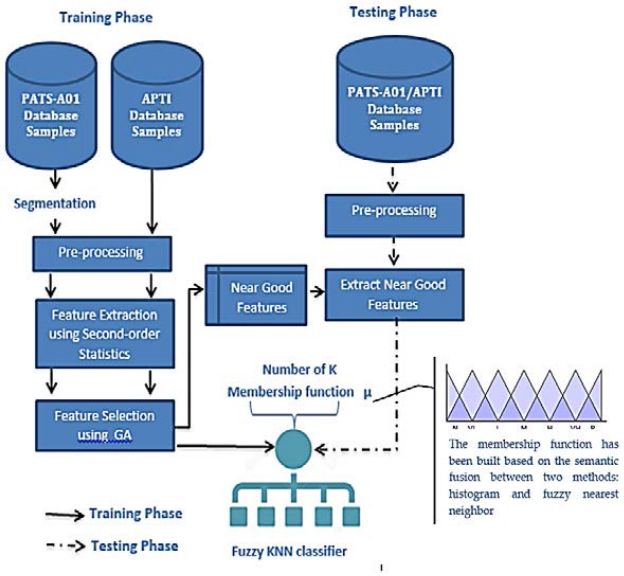

FIGURE 3. The proposed Arabic OCR model by fusing GA and FKNN.

in eight fonts. Secondly, the APTI database [26] includes 113,284 text images, 10 Arabic fonts, 10 font sizes, and 4 font styles. The samples are different in size, font type, orientation, and noise degree. Since PATS-A01 images are li

B. SEGMENTATION

Since word segmentation is the major source of errors in recognition, the proposed model avoids this step and uses pre-segmented images (segmentation-free words) [1]. However, images from the PATS-A01 database are lines of words; these will be segmented manually. Given a digital text image, a line segmentation algorithm locates and extracts each text line from the image for further processing. The challenges for line segmentation are mentioned as follows: (1) Overlapping line boundaries, (2) Touching lines, (3) Broken lines, (4) Lack of baseline information, (5) Curvilinear text, (6) Piecewise linear text, (7) Touching characters and words within a line. See [27] for more information.

C. PREPROCESSING

Pre-processing aims to produce a clear version of every image for the OCR model [25], [28]. In this step, data is subjected to many preliminary processing phases. Each sample's image follows five operations to prepare for feature extraction, as shown in Fig. 4. These operations are: (a) transforming the image to grayscale and then to binary format, (b) removing noise from the image by applying a suitable median filter, (c) removing all small objects by applying morphologic open and close operations, (d) correcting the image if it is rotated, (e) resizing image to appropriate dimensions in order to handle the scale problem since some of the characters in the text may have various sizes and scales.

FIGURE 4. Preprocessing operations.

D. FEATURE EXTRACTION

The main goal of the feature extraction stage is to maximize the recognition rate with the minimum number of features that are stored in a feature vector. The underlying idea of this stage is to extract features from word images that achieve a high degree of similarity among samples of the same classes and a high degree of variation among samples of other classes [6], [13]. As stated in [5], feature extraction methods based on second-order statistics achieved higher differentiation rates than the power spectrum (transform-based), and structural methods. From these second-order statistics, image moments achieved the best results [14]. Consequently, the proposed model employs a set of fourteen features extracted from Gray Level Co-occurrence Matrix (GLCM) that are dependent on invariant moments; because they are translation and scale-invariant [5], [6], [13], [16].

In general, the first-order statistics of an image, concerned with properties of individual pixels, obtained from mean and standard deviation. As known, the second-order statistics of an image can be obtained from GLCM, which accounts for the spatial inter-dependency or co-occurrence of two pixels at specific relative positions. Co-occurrence matrices are calculated for the directions of 0°, 45°, 90°, and 135°. For every matrix, the fourteen features that include angular second moment, correlation, contrast, the sum of squares or variance, inverse difference moment, sum average, sum entropy, sum variance, difference entropy, difference variance, information measure of correlation and cluster tendency are obtained. The homogeneity, entropy, contrast, and energy are sensitive to the choice of the direction. The entropy and homogeneity supply the indication on the dominancy values of the main diagonal on the basis of the frequencies. The energy supplies the information on the randomness of the spatial distribution. See [29] for more details.

The advantage of the co-occurrence matrix calculations is that the co-occurring pairs of pixels can be spatially related in diverse orientations regarding distance and angular spatial relationships, as on considering the relationship among two pixels at a time. As a consequence, the combination of grey levels and their positions are exhibited apparently. In comparison with deep learning-based feature extraction, CNN has a problem of overfitting, and it is mostly computationally expensive because it has to take a large dataset for training. So, you give much data, CNNs are stronger and more willing to give you better performance, you give less data CNNs is very weak.

E. FEATURE SELECTION

Feature subset selection problem is concerned with finding a subset of the original features of a dataset, such that an induction algorithm was running on data that only including the selected features that will produce a predictive model that has the highest possible accuracy. It is important to select a subset of those features which are most relevant to the prediction problem and are not redundant [30]. In general, a feature f_i is relevant if a change in the feature's value can result in a change in the value of the predicted (class) variable. A feature f_i is powerfully relevant if the use of f_i in the predictive model eliminates the uncertainty in the classification of instances. A feature f_i is weakly relevant if f_i becomes strongly relevant when a subset of the features is removed from the set of available features. By implication, a feature is irrelevant if it is not powerfully relevant, and it is not weakly relevant. A feature f_i is redundant relative to the class variable C and a second feature f_i if f_i has stronger predictive power for f_i than for the class variable C. The reduction of the number of features decreases the size of the instance space, and therefore also decreases the complexity of the prediction problem [7].

The proposed model utilizes the obvious feature selection that includes the use of a distinct step to select those features that are considered relevant for a predictive modeling task. As a rule, the suggested model needs to extract the best features that optimize classification results and highlight the discrepancy among different classes. The goal in optimization is to find the best possible solution or solutions to a problem, with respect to one or more criteria. Therefore, genetic algorithm is utilized to select the best features and reduce the dimensionality of the training dataset. The GA is a well-established and popular algorithm with recognition applications as it yields good optimization for "noisy" environments [7]–[9]. This is what distinguishes GA in dealing with the extraction of features in the Arabic words, which has a lot of noise represented in the great overlap between the words due to the presence of many of the similar letters.

Genetic Algorithms (GAs) are stochastic optimization methods based on the mechanics of natural evolution and natural genetics [8], [9]. They work with a population of individuals, each representing a practical solution in the research space. A fitness score (namely the objective function) measures the adaptability of individuals in their environment. For group individual, the set of parameters are coded into a finite-length character string (chromosome). The convergence of the population to a global optimum of the space comes from applying respectively three genetic operators: selection, crossover, and mutation. However, for simple genetic algorithms, all the individuals in the population converge to a single solution representing the global solution of (see Algorithm 1).

In this case, an instance of a GA-feature selection optimization problem can be described in a formal way as a four-tuple (R, Q, T, f) defined as [8], [31]–[34]:

- R is the solution space (initial population – a combination of n-dim second-order statistics feature vectors) where n represents the number of vectors features (vector's element). Each bit is signified as a gene that represents the absence or existence of the feature within the vector. Every feature vector is represented as a chromosome.
- Q is the feasibility predicate (different operators- selection, crossover, and mutation). The crossover is the procedure of exchanging the parent's genes to generate one or two offspring that transfer inherent genes from both parents to raise the diversity of the mutated individuals [34]. Herein, a single point crossover is employed because of its easiness. The essential aim of mutation is to avoid dropping into a locally optimal solution of the solved problem [34]. Uniform mutation is employed for its simple implementation. The selection operator retains the best fitting chromosome of one generation and selects the fixed numbers of parent chromosomes. In all, probability-based tournament selection is the most common selection method in genetic algorithm due to its efficiency and simple implementation.
- T is the set of feasible solutions (new generation populations). With these new generations, the fittest chromosome will represent the character vector with a set of salient elements. This vector will specify the optimal feature combination explicitly in accordance with the identification accuracy
- f is the objective function (fitness function). The individual who has higher fitness will win to be added to the predicate operators' mate. Herein, the fitness function is computed based on accuracy the recognition accuracy of class matching.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

$$Presision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

In which, True Positives (TP) stands for the number of correctly classified samples, False Positives (FP) defines the number of wrongly classified samples, True Negatives (TN) represents the number of correctly rejected samples, and False Negatives (FN) is the number wrongly rejected samples. For evaluation of the classification per class, recall and precision measures were used: precision is the proportion of positive predictions that are correct, and recall is the proportion of positive samples that are correctly predicted positive [34].

F. CLASSIFICATION USING FUZZY K-NN

Classification is the decision-making process in the OCR Model that makes use of the features extracted from the earlier stage. The classification algorithm is taught with the training dataset; then it is fed with the testing dataset to recognize the different classes (each class is a word). Reaching a high identification rate needs a powerful classification technique that outperforms its contemporaries' techniques in terms of speed, simplicity, and recognition rate. The suggested model utilizes Fuzzy KNN (F-KNN) classifier.

The similarity among the KNN and F-KNN algorithms is that both of them are used to assigning a class label to a newly unclassified data object. In the KNN algorithm, each newly unclassified object is assigned to the closest class with a full membership degree of 1. While the F-KNN algorithm is more appropriate to classify ambiguous or vague data objects in the sense that every object belongs to all classes with varied degrees membership [35], [36]. The fuzzy K-nearest neighbor algorithm assigns class membership to a sample vector rather than assigning the vector to a particular class. The main advantage is that no random assignments are made by the algorithm. Besides, the vector's membership values must provide a level of assurance to go along with the resultant classification.

The basis of this algorithm is to assign membership as a function of the vectors distance from its K-nearest neighbors and those neighbors' memberships in the possible classes [10], [19]. Let $W = \{x_1, x_2, \ldots, x_N\}$ be a set of N labeled samples. Also, let $u_i(x)$ be the assigned membership of the vector x, and u_{ij} be the membership in the i th class of the j th vector of the labeled sample set. $u_i(x)$ is computed by [37]:

$$u_i(x) = \frac{\sum_{j=1}^{K} u_{ij} \left(1 / \left\|x - x_j\right\|^{2/(m-1)}\right)}{\sum_{j=1}^{K} \left(1 / \left\|x - x_j\right\|^{2/(m-1)}\right)} \tag{4}$$

The variable m determines how heavily the distance is weighted when calculating each neighbor's contribution to the membership value. If m is two, then the participation of every neighboring point is weighted by the common distance from the point being classified. As m increases, the neighbors are more evenly weighted, and their relative distances from the point being classified have less effect. As m reaches one, the closest neighbors are weighted far more heavily than those farther away, which ha the influence of decreasing the number of points that contribute to th membership value of the point being classified. As seen by (4), the assigned memberships of x are influenced by both their class memberships and the inverse of the distances from the nearest neighbors. The inverse distance helps to weight a vector's membership more if it is closer and less if it is farther from the vector under consideration [36]–[39].

In our case, the histogram-based method is employed to buil u_{ij}. In general, histograms of features provide information regarding the distribution of input feature values. A multidimensional histogram of n-dimensional feature vectors from the wor's image can be constructed for each class. The histogram thus generated can be modeled by a mixture of parameterized functions such as Gaussians. The parameterized mixture can then be used as the membership function for the particular class/image. This method is easy to

implement, and memberships once generated can be used for classification in the testing phase. So, the suggested model semantically fuses between the histogram of features (to compute u_{ij}) and fuzzy nearest neighbor (to compute $u_i(x)$) membership function generation techniques to build accurate memberships assigned to the sample vector.

V. EXPERIMENTAL RESULTS

In this section, the accuracy of the suggested model was tested, and the consequences were compared with the results of related state-of-the-art Arabic OCR systems on the same benchmarked databases. The testbed dataset contains 1200 word images (around 50.000 characters) for PATS-A01 and APTI (960 training samples and 240 as a testing sample) [26]. APTI Database is the large-scale benchmarking of open-vocabulary, multi-font, multi-size, and multi-style text recognition systems in Arabic. The database is synthetically generated using a lexicon of 113'284 words, 10 Arabic fonts, 10 font sizes and 4 font styles. The database contains 45'313'600 single word images totaling to more than 250 million characters. The images of APTI are generated using 10 different fonts. These fonts have been selected to cover different complexity of shapes of Arabic printed characters, going from simple fonts with no or few overlaps and ligatures to more complex fonts rich in overlaps, ligatures, and flourishes (Diwani Letter or Thuluth). Different sizes are also used in APTI. We also used 4 different styles, namely plain, italic, and bold and combination of italic and bold. Overall, the APTI Database contains 45'313'600 single words images, taking into account the full lexicon where the different combinations of fonts, style, and sizes are applied.

The first Printed Arabic Text Set A01 (PATS-A01) consists of 2766 text line images. The text of 2751 line images of this set was selected from two standard classic Arabic books. The text of the remaining 15 line images is added from minimal Arabic script. The line images are available in eight fonts: Arial, Tahoma, Akhbar, Thuluth, Naskh, Simplified Arabic, Andalus, and Traditional Arabic. The model tests only four of the eight fonts in this database, which are Arial, Naskh, Simplified, and Tahoma. The individual text lines of the PATS-A01 database were segmented manually to separate them into words. Training classes were 24 (13 classes for PATS-A01, and 11 class for APTI) different Arabic words in different sizes, orientations, noise degrees, and fonts, as in Fig.5.

FIGURE 5. Arabic words samples.

The experiments were conducted on an Intel Core i7–5500U, 2.4 GHz processor, 8 GB DDR3 RAM laptop, and Windows 10 operating system. The code was written in Python language using Python 3.6 software. The adopted GA configuration parameters are population type: bit strings, population size: 100, number of generations: 200, Crossover ratio: 0.8, Mutation ratio: 0.1, fitness function based on accuracy, selection scheme tournament of size 2, and finally Elite count is 2. Many criteria were used in the evaluation of the model, these criteria are training time, defined as the time consumed in the training phase, testing time, which is the time consumed in predicting all testing data, and training/testing, that is precision and recall measures were used: (1) Precision is the proportion of correct positive predictions. (2) Recall is the proportion of positive samples that are properly predicted positive.

A. EXPERIMENT 1: THE EFFECT OF USING GA ON ACCURACY

The first set of experiments was performed to compare the identification accuracy of the proposed model that employs GA to determine the optimal features and the traditional version of the model without using GA (i.e., using 14 features from second-order statistics). A set of features is extracted from each word image forming a feature vector for each word. Each feature vector is then classified individually using a fuzzy 3-Nearest-Neighbor (3NN) classifier. The results shown in Table 2 revealed that the use of the 6optimal features $[f_4, f_6, f_7, f_8, f_{12}, f_{14}]$ with fuzzy 3NN classifier generates a further identification rate improvement of 1.29 % for the same method without feature selection phase, and 2.03% improvement for PATS-A01 and APTI respectively.

TABLE 2. The identification accuracy rates with and without GA for $K = 3$.

Dataset		Accuracy (%)	Testing Time for all samples average(sec)	Training Time for all samples average (sec)
PATS-A01 (650 samples for all classes)	With GA	98.69	15	98
	Without GA	97.04	30	60
	With PSO	98.51	17	130
	With BAT	98.43	16	100
	With ACO	98.58	19	102
APTI (550 samples for all classes)	With GA	95.37	12	73
	Without GA	93.34	25	15

The performance improvement comes from the correct identification of word image because of using GA to extract optimal features (discriminative features) with the help of the objective fitness function that mixes the recognition error. In general, increasing the number of neighbors in the F-KNN classifier may decrease the identification accuracy (overfitting of the training phase), in addition to the increasing of the

computational cost. Also, APTI contains images with a small Arial font in contrast with PATS-A01 that contains images with a big Arial font; so, the accuracy for APTI decreases compared with the second dataset. As expected, using only six features, on average, for each sample will decrease the time required for identification in the test phase as compared with fourteen features (on average 56 % decreasing in time). For the training phase, the GA module consumes more time for feature selection, about 63% increasing.

Furthermore, another subset of experiments was accomplished to verify the efficiency of the genetic algorithm for feature extraction compared to other meta-heuristics algorithms such as particle swarm optimizer (PSO), BAT, and Ant Colony Optimization (ACO) [40], [41]. We have replaced the GA-based feature extractor module in the proposed model with a well-known optimization–based feature extractor as a Blackbox with their default configurations.

The results in Table 2 confirm the research hypothesis that using GA with the fuzzy classifier will enhance, to some extent, the recognition accuracy of Arabic words, compared with other packages that incorporate different optimization methods for feature extraction with fuzzy classifier. The results confirm that the differences in accuracy of identification are very small between different mechanisms of extracting features. The increase for GA does not exceed 0.001%. One explanation for these results is the use of a fuzzy classifier which has the ability based on the membership function to classify new samples based on the extracted features. The proposed membership function depends on both histogram and fuzzy nearest-neighbor techniques that can handle the overlapping between Arabic words (noisy environment).

B. EXPERIMENT 2: THE EFFECT OF USING FKNN CLASSIFIER

The second set of experiments was running to validate the role of the F-KNN as a classifier to enhance classification accuracy as compared with traditional KNN. As revealed from Table 3, matching features learned from the FKNN classifier achieves better classification performance than direct matching using baseline (KNN) algorithm. FKNN classifier enhances the recognition accuracy of up to 4% for PATS-A01 and 6% for APTI. One possible justification for this reduction in accuracy for the APTI dataset is that it contains images with a small Arial font, and resizing will degrade the quality of the image. Furthermore, some limitations are facing our model due to overlapping fonts such as Diwani and Thuluth fonts that are significantly affecting accuracy compared to the other fonts. The increased accuracy in the case of FKNN comes at the expense of the time needed for the computation. The fuzzy KNN classifier module with $k = 3$ needs twice the time as the conventional classification module needs.

Another set of experiments was implemented to verify the efficiency of the combination between the GA and the Fuzzy KNN as a classifier in the field of Arabic OCR, although both of them are not new in much research. We have replaced the fuzzy classifier in the proposed model with well-known classifiers as a Blackbox with their default configurations. These classifiers include Support vector Machine [SVM], Hidden Markova Model [HMM], Artificial Neural Network [ANN], Decision Tree [DT], Random Forest [RF], and finally Gradient boosting [GBoosting].

TABLE 3. Comparative study between fknn, knn, and Standard Classifiers with optimal features module.

Dataset	Classifier Type	Accuracy (%)	Average Classifier Time in the training phase (Sec)
PATS-A01 (650 samples for all classes)	FKNN	98.69	16
	KNN	91.04	7
	SVM	93.08	11
	HMM	87.23	14
	ANN	91.60	20
	DT	70.90	8
	RF	79.20	13
	GBoosting	80.45	15
APTI (550 samples for all classes)	FKNN	95.37	10
	KNN	89.34	5

The results in Table 3 confirm the research hypothesis that using the fuzzy classifier based on discriminative features extracted using GA will enhance the recognition accuracy. At least, the suggested combination achieved a 6 % increase in accuracy compared to the nearest combination that shields between the genetic algorithm and the SVM classifier. This increase was achieved due to the method used to construct the membership function within the fuzzy classifier that made the proposed model able to distinguish words in the Arabic language despite the great similarity in their letters and overlapping between words.

Algorithm 1 *Genetic Algorithm Pseudo Code*

$t = 0$
Generate Initial Population $[R(t)]$;
Evaluate Population $[R(t)]$;
WHILE *not termination* ***DO***
$R'(t) = Variation\ [R(t)]$;
Evaluate population $[R'(t)]$; 0
$R(t+1) = Apply\ GA\ Operators\ [R'(t)Q]$;
$t = t+1$
END WHILE

C. EXPERIMENT 3: PERFORMANCE ACCURACY WITH DIFFERENT K

The third set of experiments is conducted to clarify the effect of parameter k of the fuzzy KNN classifier on the recognition accuracy of the proposed model. In general, the standard approach to choose k is to try different values of k and see

Algorithm 2 *Fuzzy K Nearest Neighbor*

BEGIN
Input x, of unknown classification.
Set K, $1 \leq K \leq n$.
Initialize $i = 1$.
DO UNTIL (K -nearest neighbors to x found)
Compute distance from x to x_i.
IF *($i \leq K$)* ***THEN***
Include x_i in the set of K-nearest neighbors.
ELSE IF *(x_i closer to x than any previous nearest neighbor)* ***THEN***
Delete the farthest of the K -nearest neighbors
Include x_i in the set of K -nearest neighbors.
END IF
END DO UNTIL
Initialize $i = 1$.
DO UNTIL *(x assigned membership in all classes)*
Compute $u_{ij}(x)$ based on Histogram of classes' features
Increment i.
END DO UNTIL
END

which provides the best accuracy on your particular data set. So, a different number of k is considered with the stability of the rest of the model variables. The selection of k is made by selecting the best top minimum distance nearest neighbours. As shown in Table 4, the greater the k value, the greater the accuracy, but with a slight increase (only 1 to 2 % difference between the use of $k = 1$ and 5). This slight increase at the expense of cost, which is often measured by the time required to implement the program (about 11, and 8 sec is required to increase k from 1 to 5 for both dataset respectively). It can be concluded that $k = 3$ is the best optionthat achieves high accuracy at an acceptable time.

TABLE 4. Results of proposed model identification for different *K*.

Test Set	*K*	Accuracy (%)	Testing Time (s)
PATS-A01	1	96.62	4.23
	3	98.07	8.92
	5	97.32	15.14
APTI	1	94.32	2.80
	3	95.30	7.01
	5	95.09	9.91

As the proposed model mainly depends on the features selection module to associate each sample with a reduced vector that encodes the most salient characteristics that able to distinguish samples. This vector effectively handles intra and inter-based variations. Smoothing is typically a desirable property for generalization. While it technically relies on the characteristics of the dataset, increasing k should reduce overfitting, but once k is too large, the smoothing effect you intuited results in decreased variance, which will affect overall performance negatively.

D. EXPERIMENT 4: THE RELATIONSHIP BETWEEN ACCURACY AND NUMBER OF SAMPLES

The fourth set of experiments was performed to show how the recognition rate of the suggested model relies on the number of word's image samples per word because if the word has more enrolled samples, the chance of correct hit increases. The maximum allowed limit of word's image is 60 (for both PATS-A01 and APTI) per class and through which they appear different operations on the image such as rotation, scaling, and noise. In Table 5, as expected, the recognition rate increases as the number of samples grows as a result of the increase in inter-class word's image variability. The accuracy rate grows nearly by 2% on average for every increase by 5 of the number of samples in the dataset.

As shown from Table 5, increasing the number of samples within each class does not affect largely in improving accuracy up to 60 samples, since the suggested model relies on extracting characteristic features from the pattern word image, which does not vary much based on the font type and style. Combining all samples to learn the proposed model increases accuracy up to 99%; due to the GA performance in choosing the best features that represent the word's image in general. This increase is done at the cost of the time taken to train the model. But this time is negligible compared to the time consumed in the testing phase. In the training phase, the optimal feature selection module takes the most time.

TABLE 5. Relationship between accuracy rate and the number of samples.

Test Set	No of samples	Accuracy (%)	Testing Time (s)
PATS-A01	5	80.98	3.12
	10	88.48	5.51
	20	90.36	10.82
	30	98.02	14.21
	50	98.69	24.74
	60	98.70	28.54
APTI	5	80.02	1.09
	10	87.41	1.92
	20	90.33	3.78
	30	97.08	4.97
	50	97.10	8.66
	60	97.20	11.87

E. EXPERIMENT 5: THE IDENTIFICATION ACCURACY RATES AGAINST IMAGE TRANSFORMATION (SCALE-ROTATION-NOISE)

Although the proposed model relies on a mechanism to perform pre-treatment of words (pre-processing phase), which helps a great deal in improving the accuracy of recognition, and in order to verify the effectiveness of each of both

GA module to select optimal features and fuzzy classifier this set of experiments was running to assess recognition performance in case of disable pre-processing phase.

TABLE 6. Relationship between accuracy rate and scaling factor.

Scaling Ratio (%)	PATS-A01 Accuracy (%)	APTI Accuracy (%)
50	22.7	15.15
70	47.7	30.72
90	75.2	70.41
95	93.36	86.20
105	92.8	91.30
110	89.2	87.70
130	78.1	75.32
150	68.2	66.90

The first sub-experiment was performed to illustrate how the verification rate of the proposed model is robust against image resizing. Resizing operation scales an image at a scaling factor between 50 and 150 by the bicubic interpolation method. Bicubic interpolation is often chosen over bilinear or nearest-neighbor interpolation because bicubic interpolations are smoother and have fewer interpolation artifacts. As is evident in Table 6, resizing image dramatically affects the accuracy of recognition. In general, image reduction results from merging pixels, which in turn leads to loss of some details and features. Also, the image enlargement leads to the appearance of many artifacts that also leads to loss of some details and features. As the images within the APTIdataset is small in size, therefore its accuracy is generally reduced in the case of zooming in and out.

The second sub-experiment was conducted to show how the verification rate of the proposed model depends on the rotation angle of the word's image. Rotation operation rotates the image by an angle in degrees in a counterclockwise direction around its center point (rotation angles from 3 To 60 degrees). As shown in Table 7, at every rotated angle of the image, the chance of a correct hit decreases. Up to 3 degrees of the rotation angle, the returns in performance are, however, diminishing for every new rotation angle to the image because rotation moves the pixels out of place and thus the extracted features differ from the features of the original image depending on the degree of rotation. Also, as the images within the API dataset are small in size, therefore; its accuracy is generally reduced in the case of increasing rotation angle as compared with another dataset.

TABLE 7. Verification rate as a function of image's angel rotation.

Rotation Degree	PATS-A01 Accuracy (%)	APTI Accuracy (%)
3	83.65	78.41
5	75.96	72.72
7	74.03	62.50
10	69.23	62.30
20	57.69	56.81
40	53.48	45.45
60	47.7	21.15

The third sub-experiment was running to validate how the verification rate of the proposed model depends on the noise amount of the image. In this case, Gaussian noise is added to the image (noise amount between 1 and 10). As shown in Table 8, as expected, at every amount of noise, the chance of a correct hit decreases. Up to 1 degree of noise amount, the returns in performance are, however, diminishing for every new amount of noise of word's image. As noise changes the pixels' gray level, so, a difference appears in the extracted features. We get the same difference in accuracy between the two datasets.

TABLE 8. Relationship between accuracy rate and noise amount.

Noise Amount	PATS-A01 Accuracy (%)	APTI Accuracy (%)
1	91.75	87.13
2	89.71	83.85
5	75.28	69.45
7	71.33	63.82
10	67.16	58.12

F. EXPERIMENT 6: COMPARATIVE STUDY

The last set of experiments was fulfilled to validate the efficiency of the suggested model as compared to state-of-the-art models listed in Table 9 using the PATS-A01 dataset. The model in [40] relies on a widely used Hausdorff distance-based classifier for recognition. However, the main drawback of this measurement is its lack of robustness, which makes it inappropriate for noisy input data. In [25], a classifier based on a support vector machine is employed. Choosing a kernel must be according to previous knowledge of invariances. However, the linear kernel function does not fit the unpredictable invariances of the words in the current datasets. The classifier system in [15] depends on some heuristic penalties and segmentation techniques that significantly affect the SIFT descriptor accuracy. In the case of the comparison, the default parameters are set for each of the re-implemented methods that were compared.

The results confirm the superiority of the suggested model. Despite the convergence of the results of the proposed model with the results of the SVM-based recognition model but the suggested model is independent of any descriptors, and it uses a powerful set of translation and scale-invariant features. In general, SVM does not perform very well when the data set has more noise i.e., target classes are overlapping. In cases where the number of features for each data point exceeds

TABLE 9. Comparison analysis on pats−A01 dataset.

Recognition Models	Max. Testing Accuracy	Execution Time (Testing)
R. Saabni Method [42]	0.970	10
A. Al Tameemi, et al. Method [25]	0.978	17
A. Stolyarenko and N. Dershowitz, Method [15]	0.951	20
Deep learning [23] based on Siamese convolution neural networks and SVM (60 samples per class).	0.975	22
Deep learning [24] based on Dimensional long short-term memory networks and connectionist temporal classification (60 samples per class)	0.982	21
The proposed Model (six features for each word with K=3 for F-KNN 30 samples per class)	0.988	10

the number of training data samples, the SVM will underperform [3]. Furthermore, one characteristic of the Hausdorff distance is that it heavily punishes single outliers, which is a severe drawback in many cases [40].

To verify the efficiency of the proposed model compared to methods that rely on the use of deep neural networks as one of the most famous tools for extracting features, another set of experiments was conducted to compare the proposed model with recent works in [23] and [24] that differ for each other in the type of CNN and classifier as illustrated in Table 9. Both methods were re-implemented and running on the PATS-A01 dataset. Although the results have largely converged with these methods. However, we have a smaller number of training samples. Moreover, the size of the high-level feature vector used for classification is roughly 4 the size of their feature. In general, the decrease of the feature set from 14 features to only six features results in an increase in the accuracy and decrease in time. Thus, F-KNN is a powerful classification technique with an accuracy of 98.69% and short running time. Moreover, applying GA reduced the complexity by 57%, increased accuracy, and cut the time by half, by selecting the best features.

VI. CONCLUSION

Arabic offline OCR for printed text is a very challenging and an open area of research. This paper developed an Arabic OCR for printed words based on a combination of the FKNN classifier and the GA. In the beginning, the model used fourteen features dataset. After applying GA, the datasets were reduced to six features dataset; then, data was fed to the FKNN, which is fast and straightforward. GA is utilized in the feature selection process because of its ability to exploit accumulating information about an initially unknown search space in order to bias subsequent search into promising subspaces. Besides, the suggested model exploits fuzzy KNN as a classification algorithm to deal with the cursive nature of the Arabic characters. The ''fuzzification'' process ensures voting from different samples belonging to more than one class, using the membership function, which may be considered as weighted voting. The model aims to reach the least recognition error, the shortest running time, and the simplest structure. The model achieves a high recognition accuracy of 98.69% for different samples in a very short time.

One of the advantages of the proposed identification model is its dependence on the fuzzy classifier to deal with Arabic words overlapping. The strength of the fuzzy KNN classifier depends primarily on the method of constructing the membership function, which was done through the semantic fusing of both histogram and fuzzy nearest neighbor, for the first time, to improve the performance of the classification (context-based classification). However, the proposed model fails to recognize free (handwritten) words, as these samples need a feature vector that must include geometrical characteristics. Future work includes utilizing more complex Arabic font's datasets, especially Diwani font, and trying to solve the diacritics problem to achieve promising results. Furthermore, enhancing the suggested model to handle Arabic handwritten words.

REFERENCES

[1] A. Lawgali, ''A survey on arabic character recognition,'' *Int. J. Signal Process., Image Process. Pattern Recognit.*, vol. 8, no. 2, pp. 401–426, Feb. 2015.
[2] L. M. Lorigo and V. Govindaraju, ''Offline arabic handwriting recognition: A survey,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 712–724, May 2006.
[3] K. Jumari and M. A. Ali, ''A survey and comparative evaluation of selected off-line arabic handwritten character recognition systems,'' *Jurnal Teknologi*, vol. 36, no. 1, pp. 1–18, Jun. 2002.
[4] I. Bouazizi, F. Bouriss, and Y. Salih-Alj, ''Arabic reading machine for visually impaired people using TTS and OCR,'' in *Proc. 4th Int. Conf. Intell. Syst., Modeling Simulation*, Jan. 2013, pp. 225–229.
[5] M. M. Mohamad, H. Hassan, D. Nasien, and H. Haron, ''A review on feature extraction and feature selection for handwritten character recognition,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 6, no. 2, pp. 204–213, 2015.
[6] S. Ismail and S. Abdullah, ''Geometrical-matrix feature extraction for online handwritten characters recognition,'' *J. Theor. Appl. Inf. Technol.*, vol. 49, no. 1, pp. 1–8, Mar. 2013.
[7] B. Oluleye, A. Leisa, J. Leng, and D. Dean, ''A genetic algorithm—Based feature selection,'' *Brit. J. Math. Comput. Sci.*, vol. 4., no. 21, pp. 889–905, 2014.
[8] A. El-Sawy, M. Hussein, E. Zaki, and A. A. Mousa, ''An introduction to genetic algorithms: A survey, a practical issues,'' *Int. J. Sci. Eng. Res.*, vol. 5, no. 1, pp. 252–262, 2014.
[9] F. Dai, N. Kushida, L. Shang, and M. Sugisaka, ''A survey of genetic algorithm-based face recognition,'' *Artif. Life Robot.*, vol. 16, no. 2, pp. 271–274, Sep. 2011.
[10] S. M. Darwish, A. A. El-Zoghabi, and O. A. Hassen, ''A modified walk recognition system for human identification based on uncertainty eigen gait,'' *Int. J. Mach. Learn. Comput.*, vol. 4, no. 4, pp. 346–353, 2014.
[11] F. Solamani and A. Mohamed, ''Off-line optical character recognition system for arabic handwritten text,'' *J. Pure Appl. Sci.*, vol. 18, pp. 52–58, Nov. 2019.
[12] I. A. Doush, F. AIKhateeb, and A. H. Gharibeh, ''Yarmouk arabic OCR dataset,'' in *Proc. 8th Int. Conf. Comput. Sci. Inf. Technol. (CSIT)*, Jul. 2018, pp. 150–154.
[13] S. Elsaid, H. Alharthi, R. Alrubaia, S. Abutile, R. Aljres, A. Alanazi, and A. Albrikan, ''Arabic real-time license plate recognition system,'' in *Proc. 1st Int. Conf. Comput.*, 2019, pp. 126–143.
[14] M. Awel, M. Ahmed, and A. Abidi, ''Review on optical character recognition,'' *Int. J. Comput. Appl.*, vol. 6, no. 5, pp. 3666–3669, 2019.

[15] A. Stolyarenko and N. Dershowitz, "OCR for arabic using SIFT descriptors with online failure prediction," *J. Imag.*, vol. 3, no. 1, pp. 1–10, Jun. 2011.
[16] G. Abandah and T. Malas, "Feature selection for recognizing handwritten arabic letters," *Eng. Sci. J.*, vol. 37, no. 2, pp. 1–20, Oct. 2010.
[17] M. Alghamdi and W. Teahan, "Printed arabic script recognition: A survey," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 9, pp. 415–428, 2018.
[18] O. Hachour, "The combination of fuzzy logic and expert system for arabic character recognition," in *Proc. 3rd Int. IEEE Conf. Intell. Syst.*, Sep. 2006, pp. 189–191.
[19] M. Z. Khedher and G. Al-Talib, "A fuzzy expert system for recognition of handwritten arabic sub-words," in *Proc. 9th Int. Symp. Signal Process. Appl.*, Feb. 2007, pp. 1–4.
[20] M. A. Abed, H. A. A. Alasadi, Z. S. Baha Al-Deen, and A. Naser Ismai, "Fuzzy logic approach to recognition of isolated arabic characters," *SSRN Electron. J.*, vol. 8, no. 2, pp. 1–9, 2010.
[21] F. Alotaibi, M. T. Abdullah, R. B. H. Abdullah, R. W. B. O. K. Rahmat, I. A. T. Hashem, and A. K. Sangaiah, "Optical character recognition for quranic image similarity matching," *IEEE Access*, vol. 6, pp. 554–562, 2018.
[22] A. A Zaidan, B. B Zaidan, H. A. Jalab, H. O. Alanazi, and R. Alnaqeib, "Offline arabic handwriting recognition using artificial neural network," 2010, *arXiv:1006.2809*. [Online]. Available: http://arxiv.org/abs/1006.2809
[23] G. Sokar, E. E. Hemayed, and M. Rehan, "A generic OCR using deep siamese convolution neural networks," in *Proc. IEEE 9th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, Nov. 2018, pp. 1238–1244.
[24] R. Ahmad, S. Naz, M. Afzal, S. Rashid, M. Liwicki, and A. Dengel, "A deep learning based arabic script recognition system: Benchmark on KHAT," *Int. Arab J. Inf. Technol.*, vol. 17, no. 3, pp. 299–305, May 2020.
[25] A. M. A. Tameemi, L. Zheng, and M. Khalifa, "Off-line arabic words classification using multi-set features," *Inf. Technol. J.*, vol. 10, no. 9, pp. 1754–1760, Sep. 2011.
[26] F. Slimane, R. Ingold, S. Kanoun, A. M. Alimi, and J. Hennebert, "A new arabic printed text image database and evaluation protocols," in *Proc. 10th Int. Conf. Document Anal. Recognit.*, Jul. 2009, pp. 946–950.
[27] D. Gupta and S. Bag, "An efficient character segmentation approach for handwritten hindi text," in *Proc. 5th Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Feb. 2018, pp. 730–734.
[28] D. Doermann, and K. Tombre, *Handbook of Document Image Processing and Recognition*. Springer, 2014.
[29] A. Porebski, N. Vandenbroucke, and L. Macaire, "Haralick feature extraction from LBP images for color texture classification," in *Proc. 1st Workshops Image Process. Theory, Tools Appl.*, Nov. 2008, pp. 1–8.
[30] P. Kaur and J. Kaur, "Finger vein biometric information authentication system using modified genetic algorithm," *Int. J. Comput. Eng. Res.*, vol. 3, no. 11, pp. 41–46. 2013.
[31] S. Ramberger, "Genetic algorithm with niche," Eur. Org. Nucl. Res., Switzerland, France, Tech. Rep., 2000, pp. 1–9.
[32] D. Rosiyadi, S.-J. Horng, P. Fan, X. Wang, M. K. Khan, and Y. Pan, "Copyright protection for E-Government document images," *IEEE MultimediaMag.*, vol. 19, no. 3, pp. 62–73, Jul. 2012.
[33] S. Sivanandam and S. Deepa, *Introduction to Genetic Algorithms*. Springer, ch 2, 2007.
[34] R. H. Sheikh, M. M. Raghuwanshi, and A. N. Jaiswal, "Genetic algorithm based clustering: A survey," in *Proc. 1st Int. Conf. Emerg. Trends Eng. Technol.*, Jul. 2008, pp. 314–319.
[35] S. Elkasrawi and F. Shafait, "Printer identification using supervised learning for document forgery detection," in *Proc. 11th IAPR Int. Workshop Document Anal. Syst.*, Apr. 2014, pp. 146–150.
[36] M. Amirfakhrian and S. Sajadi, "Fuzzy k-nearest neighbor method to classify data in a closed area," *Int. J. Math. Model. Comput.*, vol. 3, no. 2, pp. 109–114, 2013.
[37] J. Derrac, S. García, and F. Herrera, "Fuzzy nearest neighbor algorithms: Taxonomy, experimental analysis and prospects," *Inf. Sci.*, vol. 260, pp. 98–119, Mar. 2014.
[38] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy K-nearest neighbor algorithm," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-15, no. 4, pp. 580–585, Jul./Aug. 1985.
[39] W. Pedrycz, "Conditional fuzzy clustering in the design of radial basis function neural networks," *IEEE Trans. Neural Netw.*, vol. 9, no. 4, pp. 601–612, Jul. 1998.
[40] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inf. Sci.*, vol. 237, pp. 82–117, Jul. 2013.
[41] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, "A survey on new generation metaheuristic algorithms," *Comput. Ind. Eng.*, vol. 137, Nov. 2019, Art. no. 106040.
[42] R. Saabni, "Efficient recognition of machine printed arabic text using partial segmentation and hausdorff distance," in *Proc. 6th Int. Conf. Soft Comput. Pattern Recognit. (SoCPaR)*, Aug. 2014, pp. 284–289.

SAAD MOHAMED DARWISH received the B.Sc. degree in statistics and computer science from the Faculty of Science, Alexandria University, Egypt, in 1995, the M.Sc. degree in information technology from the Department of Information Technology, Institute of Graduate Studies and Research (IGSR), Alexandria University, in 2002, and the Ph.D. degree from Alexandria University, for a thesis in image mining and image description technologies. Since June 2017, he has been a Professor with the Department of Information Technology, IGSR. He is the author or coauthor of more than 100 papers publications in prestigious journals and top international conferences. He has supervised around 60 M.Sc. and Ph.D. students. His research and professional interests include image processing, optimization techniques, security technologies, database management, machine learning, biometrics, digital forensics, and bioinformatics. He received several citations. He has served as a reviewer for several international journals and conferences.

KHALED OSAMA ELZOGHALY was born in Alexandria, Egypt. He received the B.S. degree in computer science from the Faculty of Computer and Information Sciences, Egypt, in 2012, the Artificial Intelligence Courses from Northampton University, U.K., in 2013, and the IBM Artificial Intelligence Course, in 2019. He is currently pursuing the Ph.D. degree with the Department of Information Technology, Institute of Graduate Studies and Research, Alexandria University, Egypt. He has contributed in many scientific articles in the field of image processing and medical imaging. His research interest includes artificial intelligence.