ORIGINAL RESEARCH ARTICLE | UHD JOURNAL OF SCIENCE AND TECHNOLOGY

Offline Handwritten English Alphabet Recognition (OHEAR)

Hamsa D. Majeed, Goran Saman Nariman

Department of Information Technology, College of Science and Technology, University of Human Development, Kurdistan Region, Iraq

ABSTRACT

In most pattern recognition models, the accuracy of the recognition plays a major role in the efficiency of those models. The feature extraction phase aims to sum up most of the details and findings contained in those patterns to be informational and non-redundant in a way that is sufficient to fen to the used classifier of that model and facilitate the subsequent learning process. This work proposes a highly accurate offline handwritten English alphabet (OHEAR) model for recognizing through efficiently extracting the most informative features from constructed self-collected dataset through three main phases: Pre-processing, features extraction, and classification. The features extraction is the core phase of OHEAR based on combining both statistical and structural features of the certain alphabet sample image. In fact, four feature extraction portions, this work has utilized, are tracking adjoin pixels, chain of redundancy, scaled-occupancy-rate chain, and density feature. The feature set of 27 elements is constructed to be provided to the multi-class support vector machine (MSVM) for the process of classification. The OHEAR resultant revealed an accuracy recognition of 98.4%.

Index Terms: Alphabet Recognition, Handwriting Recognition, Multi-Class Support Vector Machine, Feature Extraction, Optical Character Recognition

1. INTRODUCTION

In the digital world, handwriting is one of the most appeared challenges faced in daily life. When handwriting is detected and transformed into a digital device, several pattern analysis problems will appear that need to be solved. The problems include handwriting recognition, script identification and recognition, signature verification, and writer identification. One of the most challenging and researchable fields among mentioned problems is handwriting recognition. The well-known system in this field is Optical Character Recognition (OCR) which transforms the uneditable text-image format of script into a machine-editable and manageable format of the script. In other words, OCR is a converter software of scanned scripts to a format that could be processed as a character by a computer. For the 1st time, OCR was invented by Carley in 1870 for processing scanned retina [1].

It is worth mentioning that nearly all of the OCR systems are script specific in the sight that they are restricted to recognizing a particular language or a writing system excluding several works that focused on multilinguistic handwriting recognition. However, most works focus on a specific script or language, but still, it has been broken down for more specificity which only covers special symbols, numerals, or characters within the same language or script.

After performed an in-depth review of several research articles including survey articles [2]–[4], we conclude that the entire process of alphabetic handwriting recognition could be classified under some separated classification types based on several factors as below.

Access this article online

DOI:10.21928/uhdjst.v6n2y2022.pp29-39	E-ISSN: 2521-4217 P-ISSN: 2521-4209

Corresponding author's e-mail: Hamsa D. Majeed, Department of Information Technology, College of Science and Technology, University of Human Development, Kurdistan Region, Iraq. E-mail: hamsa.al-rubaie@uhd.edu.iq

Received: 24-05-2022 Accepted: 15-08-2022 Published: 20-08-2022

1. Script writing system
2. Data acquisition (input modes) (online and offline)
3. Granularity level of documents
4. Source of the collected dataset
5. Script recognition process

The scriptwriting system type defines the selected language to be recognized in the proposed system. The languages which are in use today throughout the world have been defined under several different systems, more details can be found in Sinwar *et al.* [2], Ghosh and Shivaprasad [3], Pal [4], Ubul *et al.* [5].

The mechanism of data acquisition could be separated into two categories [2], [6], [7]: Offline and online handwriting recognition. In online handwriting recognition, a digital device with a touch screen without a keyboard must be involved like a personal digital assistant (PDA) or mobile. Where screen sensors receive the switching of pushing and releasing the pen on the screen together with the pen tip movements over the screen. While in offline mode, image processing is involved by converting an input image (from a scanner or a camera) of text to character code which is aimed to be utilized by a text processing application.

Granularity level of documents describes the stage of detailed information taken as initial input to the defined and proposed framework, as example, a full page or a single letter of text image uses as initial input.

There are two types of sources of collected dataset; public dataset (real-world dataset) and self-constructed dataset. The term "public dataset" refers to a dataset that has been saved in the cloud and made open to the public. MNIST, Keras, Kaggle, and others are examples. While the self-constructed dataset is the dataset that the researchers create and prepare on their own by scanning handwritten documents from different people.

The script recognition process is the primary section which is the practical part of the work. In general, it is formed from four main phases, namely, preprocessing (P), segmentation (S), feature extraction (F), and classification (C). The last two phases, F and C, are the common phases in the study, there is not any work without any of these two phases. However, there are many researches in literatures without P and/or S.

2. RELATED WORK

In this section, several works will be illustrated in the field of English alphabet handwritten recognition for bringing to light varied methodologies employed in each step to accomplish the recognition.

Starting with a review study [8] which summarizes eight research papers with their contributions, limitations coupled with strategies employed to enhance OCR systems. Here, we mention two of them and demonstrate their conclusion; Patel *et al.* [9] was working on the ANN (Artificial Neural Network). Characters were extracted using MATLAB. The module was analyzed pixel by pixel and transformed into a list of characters. To find edges, they used an edge and skew detection algorithm. Moreover, it became normalized thereafter. The authors claim that the accuracy is improved by increasing the hidden layers and neurons. Only 100 input neurons were used for testing which accounts for the work's limitation. The litterateurs of Gupta *et al.* [10] segment the input data at the word level into separated characters using AI and heuristic functions. Then, the feature vector is generated by extracting features from the segmented characters. As a property of vectors, blending three types of Fourier descriptors are utilized in parallel. Finally, SVM has been employed as a classifier. The authors claim that a piece of recognition error rates may arise from the usage of low-quality material and ink density diversity, as well, is another point that degrades document quality.

The authors of Karthi *et al.* [11] propose a system to recognize cursive handwriting English letters. The initial system input is in pdf format of both alphabet and cursive English letters which have been gathered from 100 different people and the total samples are 2K. This module is accomplished through four processes, namely, image preprocessing, skeletonization, segmentation points identification, and contour separation. The final module utilizes a convolutional neural network (CNN) for training the dataset to predict recognition. Support vector machine (SVM) is the system classifier. The accuracy rate of this work achieved 95.6%.

The investigation of pre-processing, feature extraction, and classifier techniques is emphasized in Ibrahim *et al.* [12]. The pre-processing initiates with normalizing image letters to 70X50 pixel dimensions by utilizing the nearest neighbor technique. Then, the binarization process is executed using Otsu's threshold sampling procedure. Character skeleton and contour algorithms have been employed to accomplish the feature extraction step. Further, both isolated and combined feature extraction procedures are involved in the experiments. The study employed two different classifications (Hibbert

Classifier) techniques which are support vector machine (SVM) and multilayer perceptron (MLP) classifiers. The recognition experiment outcomes obtained an accuracy of 97%.

In Parkhedkar *et al.* [13], a system has been produced that implements all four available steps of the handwritten recognition process. It takes a scanned document as initial input and proceeds through preprocessing for the oncoming step in which each letter of the word will be separated from the other (segmentation). Then, the Gabor feature is served for extraction of the features that will be passed through the KNN classifier on the final step. The accuracy rate of the developed project has not been given. Rather, the authors claim that multiple experiments have been established using publicly available data and the achieved accuracy is the highest in the experimentation studies when using constant data.

In Gautam and Chai [14], the proposed work uses the publicly available dataset EMINST and MINST. This means that no pre-processing and segmentation have been applied. The work only focuses on the last two steps, namely, feature extraction besides classification. The features of English letters and digits have been extracted by employing a hybrid proposal, which combines the zoning method and zig-zag diagonal scan. The feedforward NN (FFNN) is utilized as a classifier. Then, the back-propagation learning algorithm is used for the network training. The accuracy rate of English characters (e EMNIST) and English numbers (MNIST) recognition stands for 99.8% and 94%, respectively.

The litterateurs of Zanwar *et al.* [15] select 3410 samples of Chars74K which is another publicly available dataset. Independent component analysis (ICA) technique is used in feature extraction phase. Backpropagation neural networks have been employed in the final phase (classification). The recognition accuracy shows 98.21% of matching characters.

The authors of the previous study have improved their work [16] by hibernating two techniques at the feature extraction phase while the rest remained the same apart from the dataset that MNIST employed in this work. The new technique integrates detached component analysis and hybrid PSO and firefly optimization for effective selection of features and then applies a supervised learning technique called backpropagation neural network to perform classification. Recognition accuracy scores of 98.25% were recorded using the models.

3. PROPOSED METHOD

The proposed technique for offline handwritten English alphabet recognition (OHEAR) is revealed in this section. According to the aforementioned classification of handwritten recognition, Table 1 shows the used category of the classes for the presented method.

The selected input script to the model is the English alphabet (capital and small). The presented approach acquires data offline, which implies that scanned documents (images) are served as an entry to the model. Because the model operates at the character level, it takes character images as input. The used dataset nature is self-constructed, stating that it was manually gathered from 120 individuals, each of whom typed 52 characters from A to Z and a-z.

The contribution takes place in the general script recognition process phases which are the primary and the heart of such works. Apart from data acquisition which was mentioned before (commonly referred to as the first phase), it is divided into three major phases (PFC), which are pre-processing, feature extraction, along with classification. Each phase's output will be provided into the next. The phases are illustrated in Fig. 1 and described in the subsections that follow.

TABLE 1: Classification of the proposed technique

Classes	Nominated category
Script writing system	English alphabet
Data acquisition	Offline
Granularity level of documents	Character level
Source of the collected dataset	Self-constructed dataset
Script Recognition Process	PFC

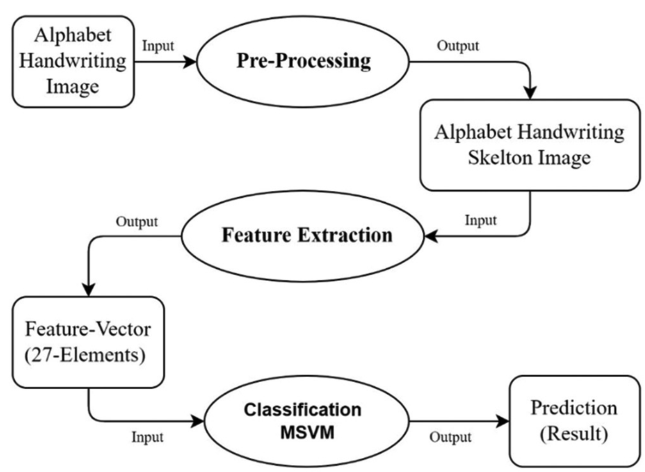

Fig. 1. Script recognition process of the proposed model.

3.1. Pre-Processing

This step is required and is a critical procedure because we are using a self-constructed dataset rather than public datasets. It should be carefully studied because the model's accuracy rate directly leans on the output quality of this phase. The reason being such a dataset used instead of using the small image size, cleaned, and noise-free public dataset is that it's truly close to data actuality in terms of real-world application.

The pre-processing procedure is broken down into six isolated processes, as shown in Fig. 2. The initial process is converting the inputs to grayscale for the purpose of size reduction which implies higher performance for the following processes without affecting accuracy.

The contrast enhancement manipulates and redistributes image pixels to improve the partitioning of hidden structural variations in pixel intensity to assemble a more distinct structural distribution.

The distribution of the pixels is calculated utilizing the histogram equalization (HE) approach, which represents the probability allocation of the image's gray levels (pixels).

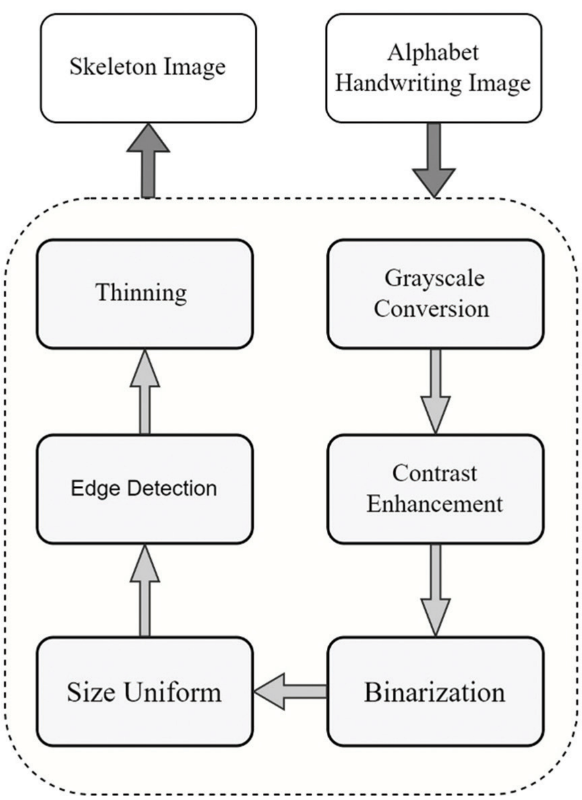

Fig. 2. The pre-processing phase of the proposed approach.

Adaptive thresholding, based on Otsu's approach, was used to convert the grayscale picture to a binary image (Binarization). This technique is used to divide the pixels into two classes: Foreground and background. Following the creation of the binary image, the sizes of all input images are uniform such that the output image only comprises the English letter. The compromised area refers to the region of interest (ROI).

After size uninformed, edge detection is the next step. It was done using the Canny approach, which locates all edges with the shortest distance between the detected edge and the processed letter's true edge. The final step of the pre-processing is for usage of skeletonization and thinning to produce the skeleton of the letter image. The thinning technique removes black foreground pixels, one at a time, until a skeleton of one-pixel width is obtained.

3.2. Feature Extraction

This phase is the uppermost critical and crucial because a proper feature extraction mechanism should be selected for a specified script. It is obvious that various scripts have distinct properties, therefore, factors that are effective in recognizing one script may not be effective in identifying another. The primary contribution of this study is the identification of features of English letter patterns that will be extracted and prepared for the oncoming and final phase of the recognition process. The feature vector is the output that consists of four segments as illustrated in Fig. 3. Each segment of the extracted feature vector is described below:

3.2.1. Tracking Adjoins Pixels

The first step in feature vector creation starts with studying the image details at the pixel level, discovering the starting point then tracking the flowing of each letter through the pixels owned by concerned image. Any pixel with more than 2 adjoins is represented as an intersection point, while the open-end point has precisely one adjoin as illustrated in Fig. 4 which is the English Letter H with two intersection points and four open-ended points.

3.2.2. Chain of Redundancy (CR)

The next feature is retrieved using Freeman Chain Code [17]. In the proposed OHEAR, the Chain code is employed to describe the form of English alphabets as a linked sequence of pixels in a restricted length and direction. This expression is based on clockwise 8-connectivity, as shown in Fig. 5a.

The skeleton image is tracked starting from the open-ended pixels and stopped at the last open-ended pixel. As for the intersection pixel, the tracking operation will be done

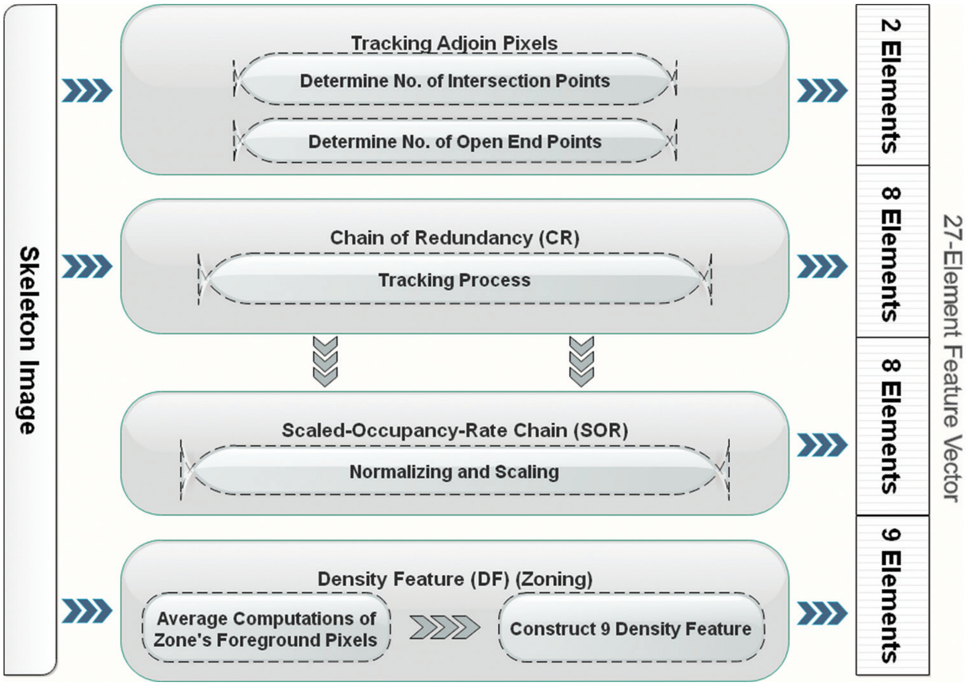

Fig. 3. Feature extracted process of OHEAR.

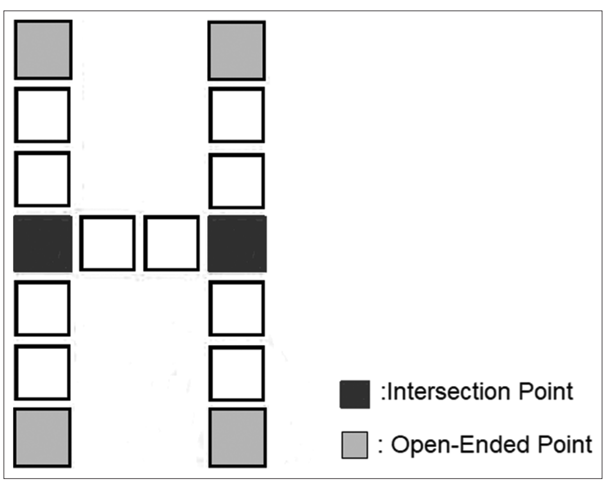

Fig. 4. Intersection points and open ended of letter H.

by proceeding in the alternative direction defined by that intersection point until it gets to the terminated open-end pixel. This process will continue until the entire pixels of the entered skeleton image of the English alphabet are tracked. A numbering method is employed to code the direction and length belonging to the pixels.

For instance, the generated chain code for the letter (S) is illustrated in Fig. 5b which shows that the starting pixel is the top-right open-ended one which indicates chain code 7 followed by three more 7s. Then, it turns to the left as 5 and

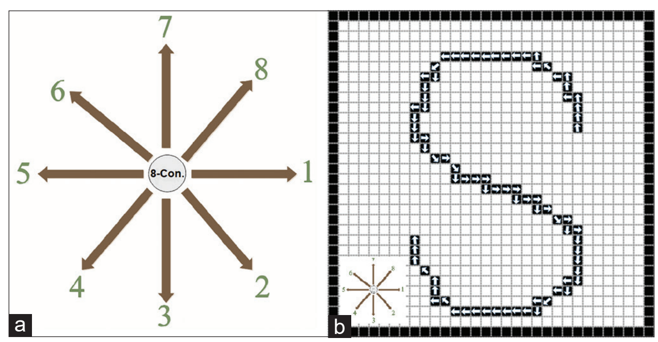

Fig. 5. (a) Eight directions of freeman chain code. (b) S letter with chain code directions.

so on. These chain code numbers will be adjusted for creating the Change of Redundancy (CR). CR consists of eight elements starting from index 1 to 8 which index numbers represent the directional numbers from the freeman chain code. For instance, in the full tracking process, 11, 3, and 19 times the chain code directions of 1, 2, and 3 have been repeated, respectively. In the result, the indexes 1, 2, and 3 of CR contain 11, 3, and 19. Finally, the CR with eight elements will be added to the feature vector as the second segment.

3.2.3. Scaled-Occupancy-Rate chain (SOR)

More valuable information can be retrieved from the above-generated data (CR) which involves the total pixels' number occupied by the English letter and considering the repetition of the individual number chain code directions. The Scaled-Occupancy-Rate chain (SOR) can generate a reasonable value to be added to the feature vector that could be generated, the Scaled-Occupancy-Rate chain (SOR).

SOR is a significant segment of the feature vector that gives weight to each chain code direction. For instance, the ideal CR of direction 3 (from Fig. 5a) for letters I and E is similar but the SOR of them is totally different, it gives 100% weight to the direction of 3 for I but much less for E.

SOR will be generated as follows, the division process applied to each index of CR on the total pixels number occupied by the skeleton image of the English letter, in other words, each index of CR is divided by the summation of CR's indexes values. For instance, from mentioned CR of S, the total pixels number of S's foreground is 76, so, the computation of the first and third indexes will be 11/76=0.144 and 19/76=0.25, respectively.

Finally, a scale factor of 10 will be hands-on to get a more practical value for classification objectives. For example, 0.144 and 0.25 will be 1.4 and 2.5, respectively. The final result with eight elements will be added to the feature vector as the third segment.

3.2.4. Density Feature (DF)

The final insertion to the feature set is the information extracted from the demanded character under the employment of the density feature. This segment of feature is achieved using the zoning technique which has been applied to the skeleton image of letters.

Zoning is a statistical feature extraction that calculates the density of foreground pixels by the zone's pixel numbers, each letter's image divided into 9 (3 × 3) zones. The zone's size of each is 10 × 10 denoting that the entered image will be resized to 90 × 90 before these divisions are applied as illustrated in Fig. 6.

This density feature (DF) will be calculated for all nine zones. Consequently, nine values will be generated and will be added to the feature vector as the last segment.

The ideal (S) illustrated in Fig. 6 takes all the nine zones, but in reality, the handwritten is dissimilar from the ideal state, the results from our dataset plotted in Fig. 7 demonstrate that with different handwritten styles, the zones' occupation will be changed accordingly. Fig. 7A3 shows that zone-3 and zone-7 will be discarded in the calculation of DF because they have zero density. It is alike the situation for Fig. 7B3 zone-3.

3.3. Classification

The latest phase of the proposed approach is the classification process which determines the recognition output of the given English letter's image. The multi-class support vector machine (MSVM) has been implied which is based on the support vector machine (SVM) technique.

The SVM is a well-known classifier, and it has obtained much traction in machine learning and statistics since it was first introduced. Vapnik's foundational work (1998) [18] set the groundwork for the theory of SVM generic statistical

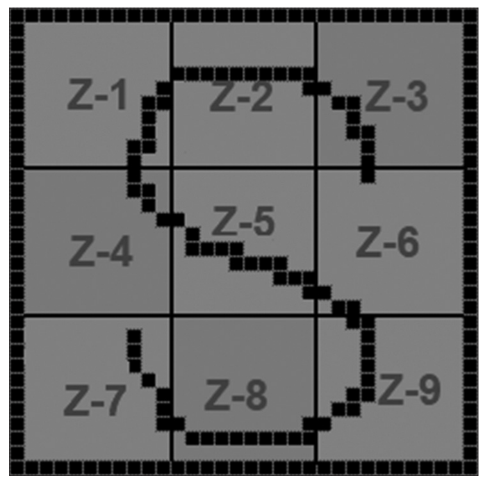

Fig. 6. Ideal resized and zoned S letter.

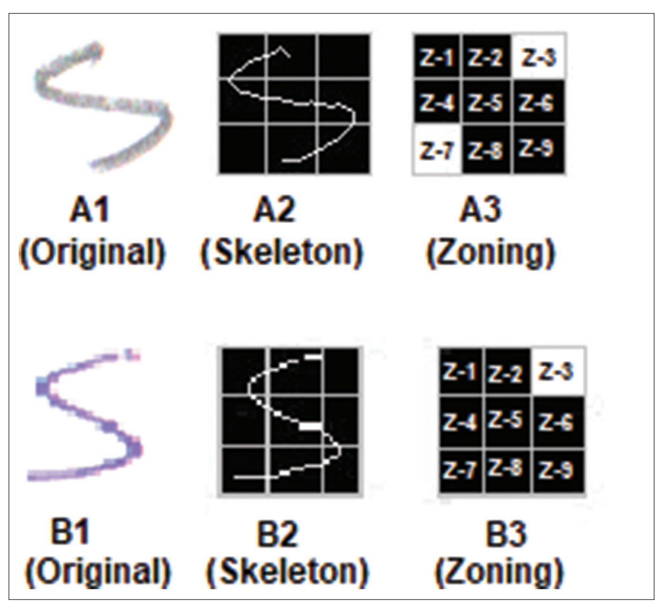

Fig. 7. Zone density and occupation of different handwritten styles.

learning, which, in turn, inspired several expansions. SVM is a binary classifier which means it only handles two-class classification issues. Therefore, it does not suit our work while having 52 English alphabet classes. More details about binary SVM can be found in Cristianini and Shawe-Taylor [19], Schoelkopf and Smola [20]. As a result of its limitation, the MSVM model has been developed to determine the dynamic process instability using multi-class classification. It has also found use in a variety of fields, including control chart pattern recognition besides industrial problem diagnosis [21], and is employed for many different language characters and numerals recognition such as Romaine, Thai, French, and Arabic Persian [1]. Furthermore, it is worth noting that, according to Ubul *et al.* [5], MSVM classifiers using various extracted features outperformed K-NN and NN classifiers in handwritten recognition field.

The feature vectors from the previous phase which were generated from 80% of the self-constructed dataset will be employed to train MSVM to create the classification model. This model creates 52 classes of small and capital English letters. The remaining 20% dataset are for testing operation.

4. RESULTS

The experimental outcomes have been established to assess the proposed model OHEAR performance. The model is implemented using MATLAB 2020a and the evaluation process had been performed through a constructed dataset consisting of 52 offline handwritten English alphabet from A (a) -to -Z (z) self-collected from 120 individuals, in a total of 6240 samples collected for capital and small letters together. With the aim of covering most of the various possibilities of the handwritten patterns, various types of writing objects (pen, pencil, and magic marker) with different colors and font sizes were applied to prove the effectiveness of the presented model regarding the recognition process.

The first set of results was in image form and from the share of preprocessing phase, as Fig. 2 presents, this phase goes through six stages starting from greyscale conversion to thinning, the outcome of this phase is illustrated in Fig. 8 for letter G.

Regardless of the entered image's color, it will be converted into the grayscale in the early steps of preprocessing phase, in the second stage, the brightness level is equalized yielding the contrast enhancement of that image. The oncoming stage shows the outcome of binary conversion through the

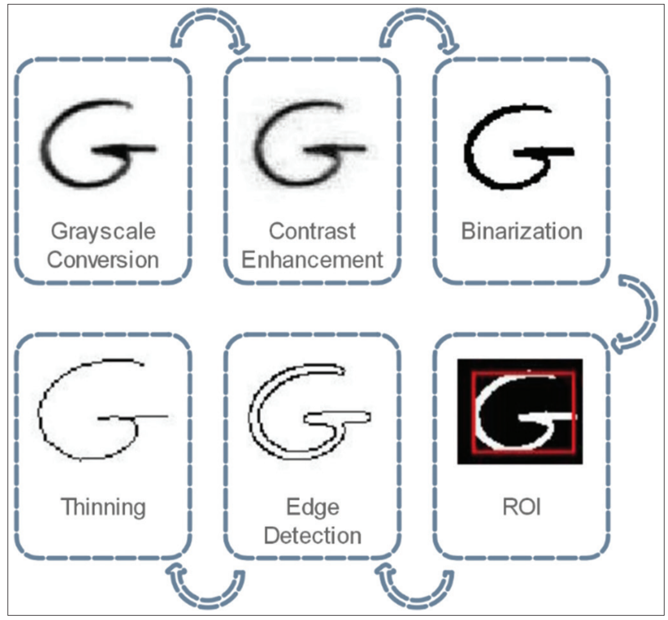

Fig. 8. Pre-processing phase results.

adaptive thresholding of the input. Size refinement is applied after binarization to determine ROI in a preparation step to the following stage where the edge of the interesting region is detected, the final stage represents the resultant thinning output to be ready for the oncoming phase of OHEAR which is the feature extraction. In this phase, the same stages are applied for all the 52 letters for each individual. In fact, it applied to all the collected dataset to the feature extraction state.

The second set of results was in the numbers form where the feature set has been extracted for each letter of the English alphabet through the OHEAR model where the statistical and structural features have been extracted and combined into one feature set with 27 elements.

As Fig. 3 revealed, the 27 elements of the extracted features are combined in four portions. In this section, those elements are translated into numbers in four tables, each table describes one of those portions in different capital with small letters. In fact, each table describes the outcomes of that portion of the feature set for all of the 52 letters, but for the publication requirements, the letters are distributed among the tables to show most of the outcomes of the letters. Consequently, all portions were gathered from the tables to define each letter in the dataset so as to create the feature set of that letter to be distinguished by the used classifier.

Table 2 illustrates the first portion of the feature set which is outlined by the pair (intersection and endpoints), the

TABLE 2: Intersections and endpoints

Character	A	a	B	b	C	c	D	d	E	e
No. of intersection points	2	2	1	1	2	2	0	2	1	0
No. of endpoints	2	1	0	1	0	0	0	2	3	1

TABLE 3: Chain of redundancy (CR)

Character	Chain of redundancy (CR)							
F	4	4	6	3	13	0	0	1
f	5	3	16	2	1	0	0	2
G	18	5	9	20	12	5	2	6
g	4	7	19	5	8	2	3	4
H	12	3	20	1	0	0	0	1
h	0	5	25	1	0	0	0	0
I	10	7	23	4	4	1	0	0
i	0	3	19	1	0	0	0	0
J	0	1	20	10	6	2	5	5
j	0	2	14	4	4	2	0	0

outcomes of A-to-E small and capital letters were illustrated, some challenges appear in this section of feature collection one of belonged long to the handwriting style in which the lines were not connected properly or more intersection points than normal created. Hence, this portion alone was not reliable enough and needs to have more features to be extracted, which lead to the second and third portions where their results are illustrated in Tables 3 and 4.

Tables 3 and 4 contain the portions: Chain of redundancy (CR) and Scaled-Occupancy-Rate chain, respectively, each portion has eight elements. The outcomes of F-to-J small and capital letters were illustrated in Table 2, while Table 3 shows the outcomes of K, L, M, Y, and P small and capital letters, the letters in Table 3 are not consecutive as trying to decrease the letters with a looks like letters as capital and small or looks as other letters in the same table. Those two portions increased the richness of the extracted characteristics from the letters with a minimum number of feature elements. Moreover, the combination of features' outcomes of the three tables so far improved the classification accuracy. Yet, some limitations floating to the surface of the process, because of existing different techniques in handwriting tracking the chain through the directions may differ for the same letter, for example, the straight line in a letter been written in bent way, or circles in some letter were not written completed, otherwise, some handwritten styles write circles where it should be a normal line, all these issues affect the chain creating process in those portions because it leans on the directions. These limitations have been solved by using another portion of combination which is the Density Feature.

Reaching Table 5 which reports the last portion of the feature set, the density chain provides the feature vector with the last nine elements. Those elements describe the density of nine zones for each letter, the results of Q, R, N, T, and U letters capital and small. Combing the outcomes of this portion with the previous chains boost the recognition accuracy, it gives occupied zones for each letter with the exact rate of that occupation in each zone, which advances the amount of information that extracted about each letter although there are some issues appear in some letters causing due to the writing direction sometimes it's in slant or diagonal way but when it's combined with the other features from the other portions it gives a cleared version of description to the classifier for recognition operation of that letter.

The next and final phase in the OHEAR model is classification, multi-class SVM is employed for this purpose in the proffered model, as a preparation step for this phase, all the features are gathered from the collected samples and then grouped into two packs of data, training data which contain 80% of the constructed dataset (96 samples out of 120 for each letter) fed to classifier for the purpose of training, while the remaining 20% labeled as test data (24 samples out of 120 for each letter) supplied to classifier for performance testing of the presented recognition model.

The recognition accuracy out of 100% has been measured for all the gathered samples. According to the outcomes from the self-constructed dataset used in this study, the handwritten English alphabet recognition accuracy in the proposed model can be classified into three groups:

First Group: The letters which achieved 100% accuracy throughout all the testes samples regardless of the font size, type of used pen, or its color, accompanied by the variety in how it's written or how straight it is (mostly slanted). The proposed combination of feature extraction mechanisms powered up the recognition ability of the classifier. Most of the letters (capital and small) belong to this group and this matter caused the raise of the total recognition accuracy of the proposed model.

Second Group: Portion of the letters which belong to this group, precisely (small letter of L, Capital letter of I, and z) are not fully recognized successfully, the classifier misclassifies one sample from the testing set of samples (i.e., 23 from 24 testing sample scored). This is due to the common way of handwriting those letters, commonly capital letter of I is similarly written as a small letter of L, beside the used way of writing the capital letter of Z with an extra line in the middle which confused the first portion of the feature vector.

Third Group: The letters (i and j) are the reason for this group creation, the classifier misclassifies two of the testing samples (i.e., scored 22 out of 24) for two major reasons, first, the dot (.) above the letters sometimes writing close to the letter, far, or lightly written in a way that excluded in the preprocessing phase. The second reason is produced by ROI determination, when the dot is written far from the letter, then it is considered out of the region of interest and excluded from the process.

Despite the fact that the model achieved an excellent recognition rate of (98.4%), there are still areas for improvement, such as reconsidering the mentioned issues in classification groups, which will be discussed in the following section.

The proposed combination of extracted features in this work is unique, for that matter, a comparison study has been made for the percentage of recognition rate achieved by other researchers that used different approaches for feature extraction as Table 6 illustrates. It is noticeable that the proposed model contributes remarkable efficient

TABLE 4: Scaled-occupancy-rate chain (SOR)

Character	Scaled-Occupancy-Rate chain (SOR)							
K	0.1875	0.3437	0.0312	0.0343	0.0937	0	0	0
k	0.2285	0.4571	0.3142	0	0	0	0	0
L	0.2142	0.0714	0.4285	0.2142	0	0	0	0.0714
l	0	0.1666	0.8333	0	0	0	0	0
M	0.0476	0.1309	0.4047	0	0	0	0.2023	0.2142
m	0.1904	0.0714	0.2857	0.0714	0	0	0.0952	0.2857
Y	0	0	0.2500	0.7250	0.0250	0	0	0
y	0.0212	0.1276	0.4042	0.2127	0	0	0	0.1702
P	0	0.0555	0.6944	0.2222	0	0	0	0
p	0.0681	0.0681	0.2045	0.1136	0.0101	0	0	0

TABLE 5: Density features

Character	Zones density values								
	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9
Q	14.166	15.111	12.277	14.166	15.111	16.055	10.622	22.133	0
q	10.818	17	11.333	17	27.818	5.6666	0	0	12.750
R	34.151	36.428	0	31.875	30.222	11.333	19.125	0	20.777
r	21.250	7.0833	18.888	21.250	28.333	0	17.163	4.3589	0
N	3.2692	8.1730	16.346	19.615	19.615	19.615	15.088	15.088	9.0532
n	13.909	23.181	4.2148	6.9545	23.181	23.181	0	18.545	6.3223
T	16.071	28.928	13.928	0	15	0	0	12	0
t	0	10.699	5.3496	13.730	35.664	10.699	0	23.181	12.482
U	20.863	0	11.590	25.500	0	23.181	11.590	23.181	9.2727
u	16.227	0	8.4297	25.500	9.2727	23.181	2.3181	13.909	14.752

TABLE 6: Illustrations of accuracy rates for various feature extraction techniques

Previous work	Feature extraction approach	Accuracy rate
Gautam and Chai [14]	Combination: Zoning method+zig-zag diagonal scan	94%
Zanwar *et al*. [16]	Combination: Detached component analysis+hybrid PSO	98.25%
Ibrahim *et al*. [12]	Combination: Features that are based on viewing capabilities+bit map feature.	97%
Zanwar *et al*. [15]	Independent component analysis (ICA) technique	98.21%
The proposed model (OHEAR)	Combination: Tracking adjoin pixels+chain of Redundancy+Scaled-Occupancy-Rate chain+and density feature	98.4%

recognition performance with a non-previously processed self-constructed dataset with different types of writing objects along with avoiding redundancy in the generated data for classification purposes.

5. CONCLUSION AND FUTURE CONSIDERATION

The most compacted and informative set of features has remarkable effectiveness to enhance the classifier – efficiency, recognition accuracy, and reliable classification accomplishment. This work presents an optimized feature extraction phase by employing both statistical and structural techniques to retrieve the features from constructed dataset self-collected for offline handwritten English alphabets through recognition (OHEAR) model. The extraction process goes through four stages: Tracking adjoins pixels, redundancy chain, adjusted scaled redundancy chain, and density feature.

The extracted feature set is provided to the multi-class SVM classifier which has been trained and tested using 120 sets of each capital and small letters of handwritten English alphabets. The proffered model achieved a recognition accuracy of 98.4%. Despite the good recognition rate, the experimental outcomes reveal some misclassification of some letters, those issues could be enhanced by making slight changing in the used features extraction techniques to raise the classification accuracy. Replacing the tracking adjoin pixels with another technique is a suggestion to overcome those misclassification issues, adopting the actual length of chain before redundancy calculation as a number in the features set are another possible suggestion besides expanding the threshold of ROI to include all the detailed characteristics of the letters while still, the increasing of the training set is always a valid option to improve the classification accuracy process. All over, reducing the total length of the feature vector with preserving the quality of the system and the level of validation rate is the goal looking forward to, on the other hand, employing another classifier is an important factor to achieve an optimum outcome from the proposed system. Moreover, the presented recognition model (OHEAR) can be extended for symbols, special characters, or other language recognition.

REFERENCES

[1] J. Mantas. "An overview of character recognition methodologies". *Pattern Recognition*, vol. 19, no. 6, pp. 425-430, 1986.

[2] D. Sinwar, V. S. Dhaka, N. Pradhan and S. Pandey. "Offline script recognition from handwritten and printed multilingual documents: A survey". *International Journal on Document Analysis and Recognition*, vol. 24, no. 1-2, pp. 97-121, 2021.

[3] D. Ghosh and A. P. Shivaprasad. "Handwritten script identification using the possibilistic approach for cluster analysis". *Journal of the Indian Institute of Science*, vol. 80,no. 3, pp. 215, 2000.

[4] U. Pal. "Automatic script identification: A survey". *J. VIVEK, Bombay*, vol. 16, no. 3, pp. 2635, 2006.

[5] K. Ubul, G. Tursun, A. Aysa, D. Impedovo, G. Pirlo and I. Yibulayin. "Script Identification of Multi-Script Documents: A Survey". *IEEE Access*, vol. 5, pp. 6546–6559, 2017.

[6] A. Priya, S. Mishra, S. Raj, S. Mandal and S. Datta. "Online and offline character recognition: A survey". *2016 International Conference on Communication and Signal Processing (ICCSP)*, pp. 0967-0970, 2016.

[7] X. Y. Zhang, Y. Bengio and C. L. Liu. "Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark". *Pattern Recognition*, vol. 61, pp. 348-360, 2017.

[8] B. M. Vinjit, M. K. Bhojak, S. Kumar and G. Chalak. "A Review on Handwritten Character Recognition Methods and Techniques". In: *2020 International Conference on Communication and Signal Processing* (*ICCSP*), 2020

[9] C. I. Patel, R. Patel and P. Patel. "Handwritten character recognition using neural network," *International Journal of Scientific and Engineering Research*. vol. 2, no. 5, pp. 1-6, 2011.

[10] A. Gupta, M. Srivastava and C. Mahanta. "Offline handwritten character recognition using neural network". In: *2011 IEEE International Conference on Computer Applications and Industrial Electronics* (*ICCAIE*), 2011.

[11] M. Karthi, R. Priscilla and K. S. Jafer. "A novel content detection approach for handwritten English letters". *Procedia Computer Science*. vol. 172, pp. 1016-1025, 2020.

[12] B. Ibrahim, H. Yaseen and R. Sarhan. "English character recognition system using hybrid classifier based on MLP AND *SVM". International Journal of Inventions in Engineering and Science Technology*, vol. 5, pp. 1-15, 2019.

[13] S. Parkhedkar, S. Vairagade, V. Sakharkar, B. Khurpe, A. Pikalmunde, A. Meshram and R. Jambhulkar. "Handwritten English character recognition and translate English to Devnagari words". *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 5, pp. 142-151, 2019.

[14] N. Gautam and S. S. Chai. "Zig-zag diagonal and ANN for English character recognition". *International Journal of Advanced Research in Computer Science*. vol. 8, no. 1-4, pp. 57-62, 2019.

[15] S. R. Zanwar, A. S. Narote and S. P. Narote. "English character recognition using robust back propagation neural network". In: *Communications in Computer and Information Science*. Springer, Singapore, pp. 216-227, 2019.

[16] S. R. Zanwar, U. B. Shinde, A. S. Narote and S. P. Narote. "Handwritten English character recognition using swarm intelligence and neural network". In: *Intelligent Systems, Technologies and Applications*. Springer, Singapore, pp. 93-102, 2020.

[17] H. Freeman. "Computer processing of line-drawing images". *ACM Computing Surveys*, vol. 6, no. 1, pp. 57-97, 1974.

[18] V. N. Vapnik and V. Vapnik. "*Statistical Learning Theory*". Vol. 1. Wiley, New York, 1998.

[19] N. Cristianini and J. Shawe-Taylor. "Background Mathematics".

In: *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, Cambridge, pp. 165-172, 2013.

[20] B. Schoelkopf and A. J. Smola. "Learning with kernels: Support vector machines, regularization, optimization, and beyond," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 16, no. 3, pp. 781-781, 2005.

[21] Y. Liu and H. Zhou. "MSVM recognition model for dynamic process abnormal pattern based on multi-kernel functions". *Journal of Systems Science and Information*, vol. 2, no. 5, pp. 473-480, 2014.