*Computers, Materials & Continua*
DOI: 10.32604/cmc.2022.026531
***Article***

Tech Science Press

# Automated Handwriting Recognition and Speech Synthesizer for Indigenous Language Processing

**Bassam A. Y. Alqaralleh[1,*], Fahad Aldhaban[1], Feras Mohammed A-Matarneh[2] and Esam A. AlQaralleh[3]**

[1]MIS Department, College of Business Administration, University of Business and Technology, Jeddah, 21448, Saudi Arabia
[2]Department of Computer Science, University College of Duba, University of Tabuk, 71491, Saudi Arabia
[3]School of Engineering, Princess Sumaya University for Technology, Amman, 11941, Jordan
*Corresponding Author: Bassam A. Y. Alqaralleh. Email: b.alqaralleh@ubt.edu.sa
Received: 29 December 2021; Accepted: 23 February 2022

**Abstract:** In recent years, researchers in handwriting recognition analysis relating to indigenous languages have gained significant internet among research communities. The recent developments of artificial intelligence (AI), natural language processing (NLP), and computational linguistics (CL) find useful in the analysis of regional low resource languages. Automatic lexical task participation might be elaborated to various applications in the NLP. It is apparent from the availability of effective machine recognition models and open access handwritten databases. Arabic language is a commonly spoken Semitic language, and it is written with the cursive Arabic alphabet from right to left. Arabic handwritten Character Recognition (HCR) is a crucial process in optical character recognition. In this view, this paper presents effective Computational linguistics with Deep Learning based Handwriting Recognition and Speech Synthesizer (CLDL-THRSS) for Indigenous Language. The presented CLDL-THRSS model involves two stages of operations namely automated handwriting recognition and speech recognition. Firstly, the automated handwriting recognition procedure involves preprocessing, segmentation, feature extraction, and classification. Also, the Capsule Network (CapsNet) based feature extractor is employed for the recognition of handwritten Arabic characters. For optimal hyperparameter tuning, the cuckoo search (CS) optimization technique was included to tune the parameters of the CapsNet method. Besides, deep neural network with hidden Markov model (DNN-HMM) model is employed for the automatic speech synthesizer. To validate the effective performance of the proposed CLDL-THRSS model, a detailed experimental validation process takes place and investigates the outcomes interms of different measures. The experimental outcomes denoted that the CLDL-THRSS technique has demonstrated the compared methods.

**Keywords:** Computational linguistics; handwriting character recognition; natural language processing; indigenous language

## 1 Introduction

Computational linguistics (CL) is the application of computer science to the comprehension, synthesis, and analysis of spoken and written language. The CL method is employed in text-to-speech (TTS) synthesizers, instant machine translation, interactive voice response (IVR) systems, speech recognition (SR) systems, language instruction materials, search engines, and text editors [1]. The interdisciplinary area of research needs expertise in neuroscience, machine learning (ML), artificial intelligence (AI), deep learning (DL), and cognitive computing. A computational understanding of language offers people with understanding into intelligence and thinking [2]. Computers are linguistically competent facilitates humans to interact with software and machines, as well as make the textual and other resources of the internet easily accessible in various languages. Most of the works in CL–have applied and theoretical elements –focus on enhancing the relationships among basic language and computers [3].

Generally, CL is utilized in large enterprises, universities, or governmental research labs. In the vertical companies, private sector, such as Caterpillar widely use CL for authenticating the precise translation of technical manuals [4]. Tech software companies, like Microsoft, usually hire CL to work on natural language processing (NLP), help computer programmers to develop voice user interface (VUI) that ultimately allows humans to interact with computing devices even though they were other people [5]. The NLP method is the application of AI to the English language. In addition, this problem would give an overview of the mathematical machinery behindhand the classification problems of indigenous language. Study in handwriting recognition analysis pertaining to Indic script was gaining more interest in the past few years. This is obvious through the presence of publicly available handwritten databases and effective machine recognition algorithms of some Indian scripts namely, Devanagari, Oriya Bengali, and Telugu, etc. [6].

Handwritten Character Recognition (HCR) consists of 2 different types: online and offline modes. Online mode translates the tip movement (strokes) of digital pen to list of coordinates, while offline mode employs character as scanned image. The difficult part of HCR is the difference from the writing pattern of individual person [7]. Even one individual handwriting could differ at various times. Conventional ML methods have been employed for an offline HCR for a longer period of time [8]. A common ML method of HCR comprises classifying, preprocessing, segmenting, and feature extraction. An offline HCR is trained initially by the group of characters (scanned image) and then a novel character images is provided as input; the scheme must be capable of recognizing it correctly [9]. HCR exposed its effectiveness in several applications like bank check reading, sorting of mails in post office, legal documents, and digitization of legacy document and form conversions or handwritten documents. The handwriting recognition challenge has been investigated by several approaches, namely: neural networks (NN), support vector machine (SVM), convolution neural network (CNN), and K-nearest neighbours (KNN) [10]. Arabic language is the commonly Semitic language, with probably 300 million speakers. It is graded as the fifth most predominant spoken language over the globe. Handwriting recognition for Arabic is considered as a challenging issue. It is generally written with the cursive Arabic alphabet from right to left. It comprises of 28 letters, everyone has distinct forms based on the position of the letter in the word. Moreover, the Arabic writing utilizes the diacritical marks denotes short vowels and other sounds, such as fat-ha, dhumma, and kasra. Arabic involves multiple ligatures, generated by the combination of two or many letters.

This paper introduces novel Computational linguistics with Deep Learning based Handwriting Recognition and Speech Synthesizer (CLDL-THRSS) for Indigenous Language. The presented CLDL-THRSS model involves two stages of operations namely automated handwriting recognition

and speech recognition. Firstly, the Capsule Network (CapsNet) based feature extractor is applied for recognizing handwritten Arabic characters. Next, optimal hyperparameter tuning of the CapsNet takes place by the use of the cuckoo search (CS) optimization algorithm. Besides, deep neural network with hidden Markov model (DNN-HMM) model is employed for the automatic speech synthesizer. In order to validate the effective performances of the proposed CLDL-THRSS model, a detailed experimental validation process takes place and investigates the results interms of different measures.

## 2 Literature Review

Jain et al. [11] presented a new architecture for detecting emotions of users in multi-language text data using emotion theory that deals with psychology and linguistics. The emotion extraction method is proposed on the basis of various features groups for a good understanding of emotion lexicon. Experiential research of three realtime events in fields such as healthcare, sports, and Political election are executed by the presented architecture. Altwaijry et al. [12] proposed an automated handwriting recognition technique for Arabic language. The presented model involves the CNN approach and is trained using the Hijja and Arabic Handwritten Character Dataset (AHCD) dataset. Lamghari et al. [13] presented a novel dataset for Arabic handwritten diacritics (DBAHD). It is developed for handling the Arabic handwriting recognition system using segmentation and machine learning. Alwajih et al. [14] developed DeepOnKHATT, a dedicated handwriting recognition model depending upon the bidirectional long short term memory and the connectionist temporal classification (BLSTM-CTC). It has the ability of accomplishing detection at the sentence level in real time. The presneted model has been validated using two publicly available datasets such as CHAW and Online-KHATT.

Hu et al. [15] designed a combined LSTM and ROM architecture. This presented method has the ability to represent the Spatio-temporal distribution as it makes use of LSTM ROM and. In order to decrease the dimension size of huge spatial data sets in LSTM, the singular value decomposition (SVD) and proper orthogonal decomposition (POD) methods have been presented. The LSTM prediction and training procedures are conducted over the reduction space. Lee et al. [16] introduced an emotional speech synthesizer based end-to-end neural system, called Tacotron. In spite of its benefit, we discovered that the original Tacotron suffers from the irregularity of the attention alignment and exposure bias problem. Next, tackle the problems by using residual connection and context vector at RNN.

## 3 The Proposed Model

This study has developed a novel CLDL-THRSS technique for the recognition and speech synthetisation of the Arabic handwritten characters. The proposed CLDL-THRSS technique encompasses preprocessing, segmentation, CapsNet based HCR, CS based hyperparameter tuning, DNN-HMM based speech synthesizer. The design of CS algorithm helps to appropriately adjust the hyperparameters of the CapsNet model.

### *3.1 Data Preprocessing*

The pre-processing was utilized to decrease the noise from the input images. The noise has cost the performance of the character detection technique. The noise arises as to the worse quality of the documents. In order to decrease the noise GF was employed from our technique. Usually, the filter was utilized for filtering redundant things or object from spatial regions or surfaces. In digital image modeling, typically the image was affected by different noises. The primary objective of the filter is for improving the quality of images with improving is for increasing interoperability of data presented

from the image to human vision. During the GF, the impulse response is Gaussian function. It alters the input signals with convolutional through Gaussian functions. The Gaussian smoothing function is a 2D convolutional function that is utilized for "blurring" the image and then removing noise and unwanted details. The Gaussian function was demonstrated under equation formula:

$$G(I) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{I^2}{2\sigma^2}} \tag{1}$$

whereas $\sigma$ implies the standard deviation (SD) of distribution and $I$ stands for the input images. According to the GF, the noise was eliminated in our suggested approach. Then, eliminating the noise in the input image the binarization procedure was implemented. Binarization is a technique of altering a gray scale image as to black as well as white images with thresholding technique. During the thresholding binarization technique, threshold value was utilized for assigning 0's and 1's to every pixel placed from the provided image. The skew recognition of scanned document images is most essential phase of their detection pre-processed. The skew of the scanned documents image requires the deviations of their text lines in the vertical or horizontal axis. Primarily remove the black text pixel to analysis.

### *3.2 Segmentation Process*

The segmentation is an essential stage from detection method as it removes meaningful regions to more investigation. It can be normally utilized to validate objects and boundaries as curves, lines, and so on. The scanned image was segmentation as to paragraphs utilizing spatial space recognition approach, paragraphs as to lines utilizing vertical histogram, lines as to word utilizing horizontal histograms. The accuracy of character detection was extremely dependent upon accuracy of segmented. The procedure of segmented mostly comprises the following:

- Recognize the text line from the page.
- Lastly, recognize separate characters from all words.

The most often utilized technique to line segmented grey scale images is the prediction profile approach. With summing up the prediction profiles together with the horizontal way of document, the gap amongst the text-lines remains recognized with define the prediction value. In last phase, the removed lines were segmentation as to character. For determining the boundaries amongst the character, it can execute a threshold values on length of the space from amongst the character. Afterward, to find the places of the space amongst characters is also remove the parts of line segments.

### *3.3 CapsNet Based Feature Extraction*

At the feature extraction process, the CapsNet receives the segmented data as input to carry out the HCR process. The CNN is most frequently utilized technique for 2D object classification. But, data as place and pose from the object was eliminated from the CNN because of their data routing process. For compensating for the limitations of CNN, a network framework named as CapsNet was presented [17]. The CapsNet is a deep network method containing capsules. The activation neuron signifies the features of modules from the objects. All capsules are responsible to determine a single module from the objects, and every capsule combined defines the entire framework of an object. Conversely, a few DNNs (for instance, DBN), this framework keeps object modules and spatial data. Related to CNN, the CapsNet is collected from multi-layer network. Fig. 1 illustrates the framework of CapsNet.

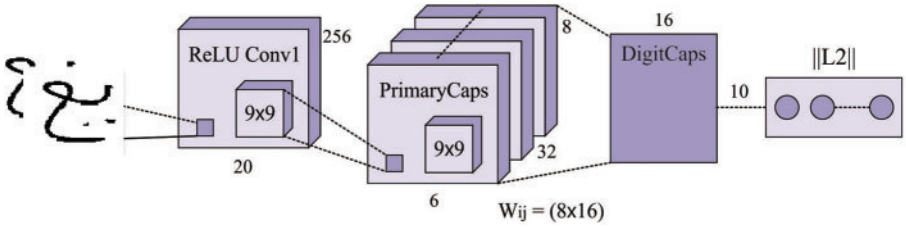


**Figure 1:** Structure of CapsNet

During the input and output of the capsules are vector. The length of output $u_j$ signifies the possibility of existence of their equivalent components, and the ways of vector $u_i$ encoded several properties (for instance, size and place) of their equivalent components. The prediction vector $\hat{u}$ implies the belief that encoded the connection amongst the $i^{th}$ capsules from the lower level capsule and $j^{th}$ capsules from the higher level capsule utilizing a linear transformation matrix $W_{ij}$ in Eq. (2):

$$\hat{u}_{j|i} = W_{ij} \cdot u_i \tag{2}$$

Specifically, the identified component existence and pose data were utilized for predicting the entire presence and pose data. In the trained procedure, the network slowly learns for adjusting the transformation matrix of capsule pair with equivalent connection amongst modules and entire objects.

During the higher level capsule, $s_j$ and $v_j$ indicates the input as well as output of capsules $j$ correspondingly $s_j$ denotes the sum values of predictive vectors $\hat{u}_{j|i}$ with equivalent weight $c_{ij}$ from low level capsule $i$. In Eq. (3), $c_{ij}$ stands for the coupling coefficients, where $\sum_j c_{ij} = 1$ and $c_{ij} \geq 0$. Once $c_{ij} = 0$, there is no data transfers amongst capsules $i$ and capsules $j$, but if $c_{ij} = 1$, each data of capsules $i$ has transferred to high level capsule $j$. As the length of output stands for a likelihood value, nonlinear squash function was utilized for ensuring that short vector was compressed that nearby 0 and the long vector was compressed that nearby 1. The squash function was illustrated in Eq. (4):

$$s_j = \sum_i c_{ij} \cdot \hat{u}_{j|i} \tag{3}$$

$$v_j = \frac{||s_j^2||}{1 + ||s_j^2||} \frac{s_j}{||s_j||} \tag{4}$$

$$c_{ij} = \frac{\exp(b_{ij})}{\Sigma_k \exp(b_{ik})}, b_{ij} \leftarrow b_{ij} + \hat{u}_{j|i} \cdot v_j \tag{5}$$

Once the lower and higher level capsules were consistent by its predictive, the value of $c_{ij}$ develops larger in Eq. (5), and it develops lesser once it can be inconsistent [18]. With changing the routing coefficients, the dynamic routing technique makes sure that the lower level capsule transmit its predictive vector to higher level capsule which is consistent with its predictive, thus the output of sub-capsules was sent to the correct parent capsule.

### *3.4 Cuckoo Search Based Hyperparameter Optimization*

The optimal hyperparameter tuning of the CapsNet system takes place using the CS approach. CS algorithm is a commonly employed bioinspired technique, developed by the inspiration of the breeding nature of the cuckoo birds. The cuckoo birds generally lay the eggs on other birds' nests and

trick them to take care of the eggs. But the other birds could identify the non-native eggs in some cases and therefore it removes them. The cuckoo bird tried to increase the probability of hatching of the own eggs by producing it identical to the host eggs with respect to color, shapes, or sizes or hurling other native eggs away from nest with a violent nature. If the cuckoo chick gets hatched, it might emit other eggs from the nest for increasing its individual feeding share [19].

Generally, the searching process starts with a particular set of nests since there exists only one solution for every nest. The population of the solution gets repetitively generated depending upon the idea of the identification of cuckoo egg (p) which is inspired by the removal of a proportion of solution in nest and replacing it with the newly produced ones. At the CS algorithm, the random walk can be utilized depending upon the Lévy flight distribution in producing new candidate solutions (cuckoos) from the existing ones as given in the following.

$$cuckoo_i^{(t+1)} = cuckoo_i^{(t)} + a \oplus Levy\,(\lambda) \tag{6}$$

where $cuckoo_i^{(t+1)}$ means the ith Cuckoo value at round $t$. The variables $a$ and $\lambda$ signify step size, and Lévy distribution coefficient $(1 < \lambda < 3)$, correspondingly.

A set of novel solutions are created from the present optimal solutions using Levy walk procedure since it enables the CS to carry out an effective local searching process with the ability of self-improvement. In addition, few of the newly produced solutions are farther from the present global best solutions [20]. It reduces the chance of trapping into local optima and assures exploration abilities. The design of CS algorithm ensured elitism as the best nest which can be presented during several iterations.

The CS technique develops a FF for attaining increased classification efficiency. It demonstrates a positive integer for signifying an optimum performance of the candidate solutions. During this analysis, the minimization of the classification error rate was regarded as FF provided in Eq. (7). A better result is a lower error rate and least solutions gain an improved error rate.

$$fitness\,(x_i) = ClassifierErrorRate\,(x_i) = \frac{numerb\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \tag{7}$$

### *3.5 Design of DNN-HMM Based Speech Synthesizer*

Finally, the design of speech synthesizer takes place using the DNN-HMM model. In convention GMM-HMM based handwritten character detection, the observation probability is modeled utilizing GMM in the maximal possibility condition. The possibility of such method was limited as GMM is statistically inefficient to demonstrate data which drive adjacent a non-linear manifold from the data space [21]. For overcoming this limitation, it can be present a hybrid DNN-HMM to handwritten character detection, in which the outcome of the DNN is capable of HMM as replacement of GMM. Fig. 2 demonstrates the structure of DNN. The HMM is signified as $= (A, B, \pi)$, entails of the subsequent elements:

1) The amount of states from the model represented as $Q$, the group of states represented as $S = \{s_1, s_2, \ldots, s_Q\}$, and $q_t$ the state at time $t$.
2) $A = \{a_{ij}\}$, the state transition probabilities distribution using

$$a_{ij} = P\left(q_{t+1} = s_j | q_t = s_i\right), 1 \le i,j, \le Q \tag{8}$$

3) $B = \{b_i\,(O_t)\}$, the observation probability, where $b_i\,(O_t)$ implies the probability of observing $O_t$ at state $s_i$. $B$ has demonstrated as a finite mixture:

$$b_i(O_t) = \sum_{m=1}^{M} c_{im} \aleph(O_t, \mu_{im}, U_{im}), \ 1 \le i \le Q \tag{9}$$

Whereas $c_{im}$ refers the mixture coefficients to the $m^{th}$ mixture from state $s_i$, and elliptically symmetric density with mean vectors $\mu_{im}$ or $\aleph$ some log-concave and covariance matrix $U_{im}$ to $m^{th}$ mixture components from state $i$. An important variance amongst DNN-HMM and GMM-HMM is utilizing DNN (before GMM) for estimating the observation probability. It can essentially utilize the DNN for modelling $(q_t|o_t)$, the posterior probabilities of state provided the observation vector $o_t$ that is feasible as $p(q_t)$ was simple for estimating in a primary state level alignment of trained set [22]. The comprehensively trained procedure to handwritten character detection is as follows:

a) To all handwritten character classes ($c = 1, \ \ldots, \ C$), left to right GMM-HMM $\lambda_c$ with $Q$ states were trained to utilize the trained sentences of class $c$.
b) To all the sentences $O = (O_1, O_2, \ \ldots, \ O_T)$ from the trained set $c$, the Viterbi technique of GMM-HMM, was carried out on $\lambda_c$ for obtaining a better state order $\left(q_1^c, \ldots, \ q_T^c\right)$, and all the states $q_t^c$ are allocated as label $L_i$ ($i \in (1, \ \ldots, \ C \times Q)$) based on a state-label mapping table.
c) Every trained sentence, along with its labeled state orders were utilized as input for training a DNN whose output is the posterior probability of $C \times Q$ output unit. The trained of the DNN was implemented utilizing BP technique with unsupervised pre-trained or discriminative pre-trained.

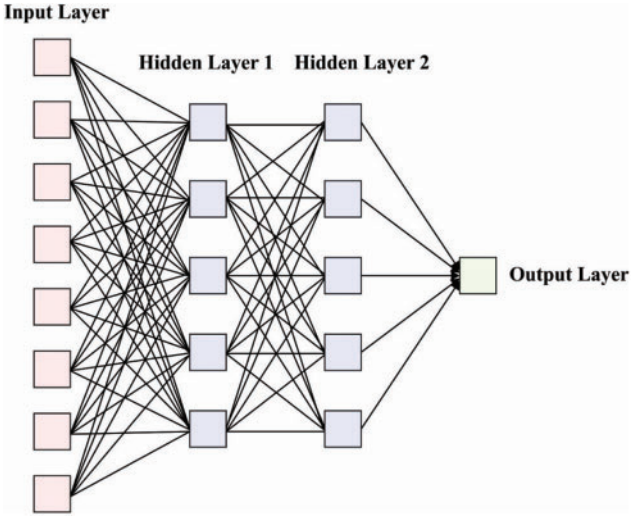


**Figure 2:** DNN structure

In order to detection procedure, to an input data $= (o_1, o_2, \ \ldots, \ o_T)$, one must evaluate the probabilities $p(O|\lambda_c)$ to all the handwritten character classes $c$, and obtain the last detection outcome. In GMM-HMM, these probabilities were attained using the Viterbi technique. In DNN-HMM is implement the subsequent process for calculating the probability $(O|\lambda_c)$.

a) An input feature order $O$ was primarily input as to DNN, attaining the posterior probability $\{p(L_i|O_t)\}_{i=1,\ldots,C\times Q}$ as output. Afterward, the later probabilities $p(q_t = S_k^c|O_t)$ is attained in $p(L_i|O_t)$, with mapping the label $L_i$ to state $k$ of models $c$

b) Based on the Bayesian rule is compute the likelihood $p(o_t|q_t)$ as

$$p\left(O_t|q_t\right) = \frac{p\left(q_t|O_t\right) p\left(O_t\right)}{p\left(q_t\right)} \tag{10}$$

During the execution, the prior likelihood of all the states, $p\left(q_t\right)$, was computed in (occurrences of) the trained set, and $p\left(O_t\right)$ is allocated as constants but the observation feature vector is considered as independent of everyone.

c) To all handwritten characters method $\lambda_c$, the Viterbi technique was executed for calculating the probability $p(O|\lambda_c)$ based on Eq. (10).

## 4 Experimental Validation

This subsection explores the result analysis of the CLDL-THRSS model using the benchmark images. The results are examined under several numbers of hidden layers. The proposed model is simulated using Python 3.6.5 tool.

Tab. 1 and Figs. 3–4 showcases a detailed recognition result analysis of the CLDL-THRSS technique under distinct layers. The experimental outcomes showcased that our CLDL-THRSS technique has accomplished maximum recognition performance in all layers. For instance, with layer-1, the CLDL-THRSS technique has provided precision, recall, accuracy, and F-score of 89.42%, 94.17%, 91.16%, and 92.69% respectively. Likewise, with layer-3, the CLDL-THRSS approach has offered precision, recall, accuracy, and F-score of 88.59%, 93.61%, 93.23%, and 94.90% respectively. Concurrently, with layer-5, the CLDL-THRSS methodology has provided precision, recall, accuracy, and F-score of 94.76%, 94.26%, 93.72%, and 91.73% correspondingly. At the same time, with layer-6, the CLDL-THRSS algorithm has given precision, recall, accuracy, and F-score of 88.46%, 93.95%, 89.83%, and 92.94% correspondingly. Lastly, with layer-7, the CLDL-THRSS system has provided precision, recall, accuracy, and F-score of 89.60%, 88.30%, 90.25%, and 94.26% correspondingly.

**Table 1:** Result analysis of CLDL-THRSS technique with different measures

| No. of hidden layers | Precision | Recall | Accuracy | F-score |
|---|---|---|---|---|
| Layer-1 | 89.42 | 94.17 | 91.16 | 92.69 |
| Layer-2 | 93.00 | 91.45 | 93.91 | 93.27 |
| Layer-3 | 88.59 | 93.61 | 93.23 | 94.90 |
| Layer-4 | 91.23 | 92.44 | 94.43 | 89.91 |
| Layer-5 | 94.76 | 94.26 | 93.72 | 91.73 |
| Layer-6 | 88.46 | 93.95 | 89.83 | 92.94 |
| Layer-7 | 89.60 | 88.30 | 90.25 | 94.26 |
| Average | 90.72 | 92.60 | 92.36 | 92.81 |

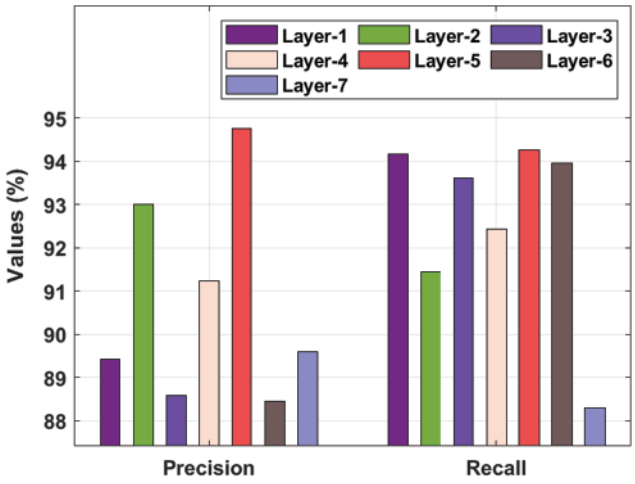


**Figure 3:** Precision and recall analysis of CLDL-THRSS technique

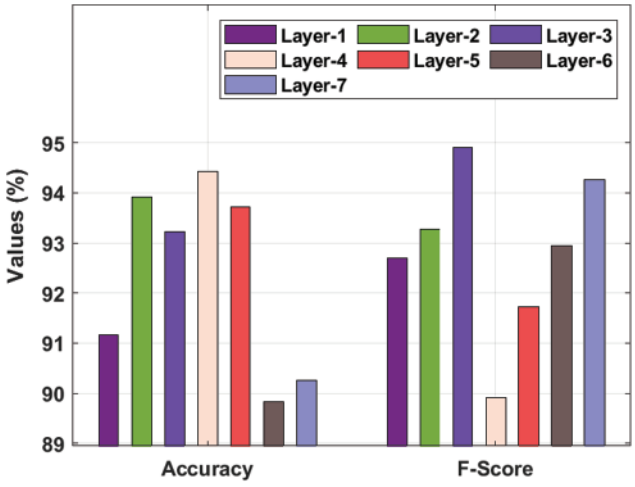


**Figure 4:** Accuracy and F-score analysis of CLDL-THRSS technique

Fig. 5 depicts the analysis of the CLDL-THRSS system on the test data set. The results demonstrated that the CLDL-THRSS technique has gained maximal efficacy with the maximum validation and training accuracy. It is detected that the CLDL-THRSS algorithm has gained superior validation accuracy over the training accuracy.

Fig. 6 exhibits the loss analysis of the CLDL-THRSS approach on the test data set. The outcomes established that the CLDL-THRSS method has resulted in a proficient outcome with reduced training and validation loss. It can be stated that the CLDL-THRSS algorithm has accessible lower validation loss over the training loss.

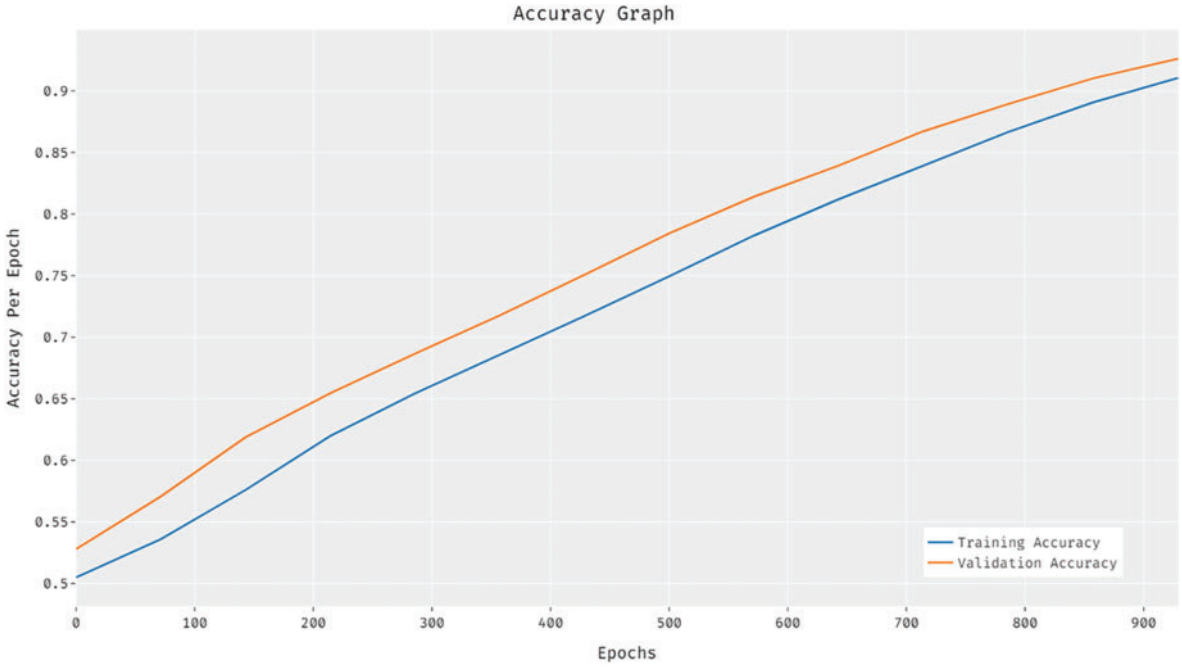


**Figure 5:** Accuracy graph analysis of CLDL-THRSS technique

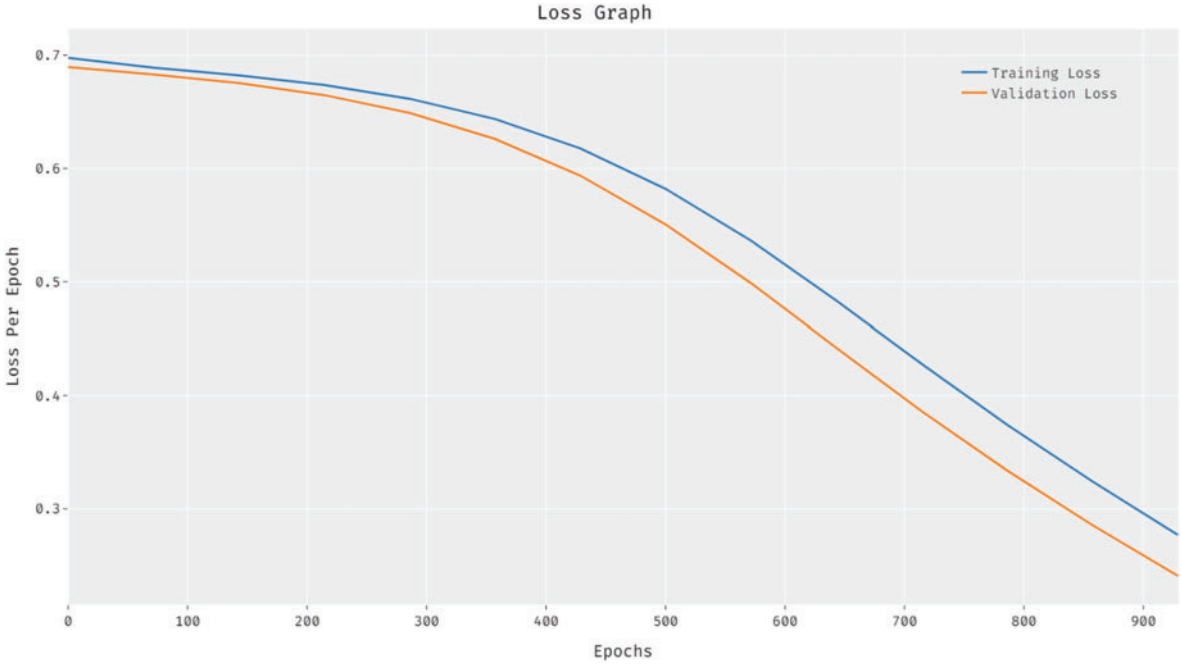


**Figure 6:** Loss graph analysis of CLDL-THRSS technique

A detailed recognition rate (RR) analysis of the CLDL-THRSS model with recent methods takes place in Tab. 2 and Fig. 7. The outcomes guaranteed the betterment of the CLDL-THRSS technique over the other techniques under all images. For instance, with image 1, the CLDL-THRSS technique

has offered increased RR of 94.69% whereas the NN and NN-EHO techniques have provided reduced RR of 84.11% and 92.52% respectively. Simultaneously, with image 2, the CLDL-THRSS technique has attained higher RR of 94.90% whereas the NN and NN-EHO techniques have obtained lower RR of 87.63% and 92.78% respectively. Concurrently, with image 3, the CLDL-THRSS technique has accomplished maximum RR of 95.54% whereas the NN and NN-EHO techniques have resulted in minimum RR of 87.50% and 92.71% respectively. Lastly, with image 4, the CLDL-THRSS technique has offered increased RR of 95.04% whereas the NN and NN-EHO techniques have provided reduced RR of 88.78% and 92.86% respectively.

**Table 2:** Recognition rate analysis of CLDL-THRSS technique with distinct images

| Images | Recognition rate (%) | | |
|---|---|---|---|
| | NN model | NN-EHO | CLDL-THRSS |
| Image 1 | 84.11 | 92.52 | 94.69 |
| Image 2 | 87.63 | 92.78 | 94.90 |
| Image 3 | 87.50 | 92.71 | 95.54 |
| Image 4 | 88.78 | 92.86 | 95.04 |

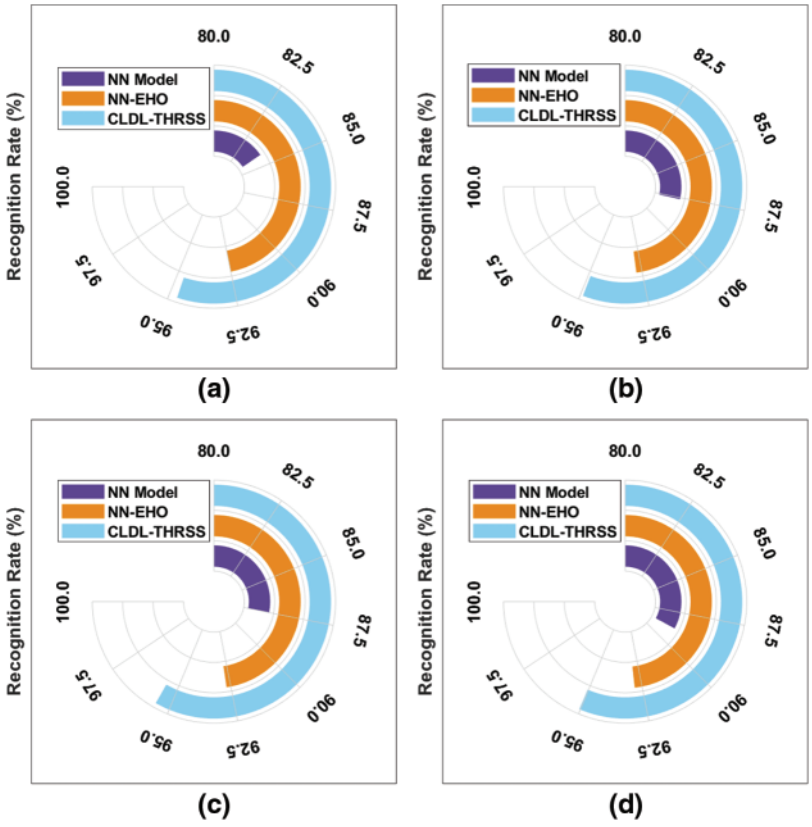


**Figure 7:** RR analysis of CLDL-THRSS technique with varying images

Tab. 3 and Fig. 8 demonstrates the brief recognition time (RT) analysis of the CLDL-THRSS technique on various set of images. The experimental results have ensured that the CLDL-THRSS technique has resulted to lower RT over the other methods.

**Table 3:** Recognition time analysis of CLDL-THRSS technique with different images

| Images | Recognition time (min) | | |
|---|---|---|---|
| | NN Model | NN-EHO | CLDL-THRSS |
| Image 1 | 1.789 | 1.637 | 1.569 |
| Image 2 | 1.655 | 1.491 | 1.414 |
| Image 3 | 1.671 | 1.527 | 1.432 |
| Image 4 | 1.860 | 1.607 | 1.489 |

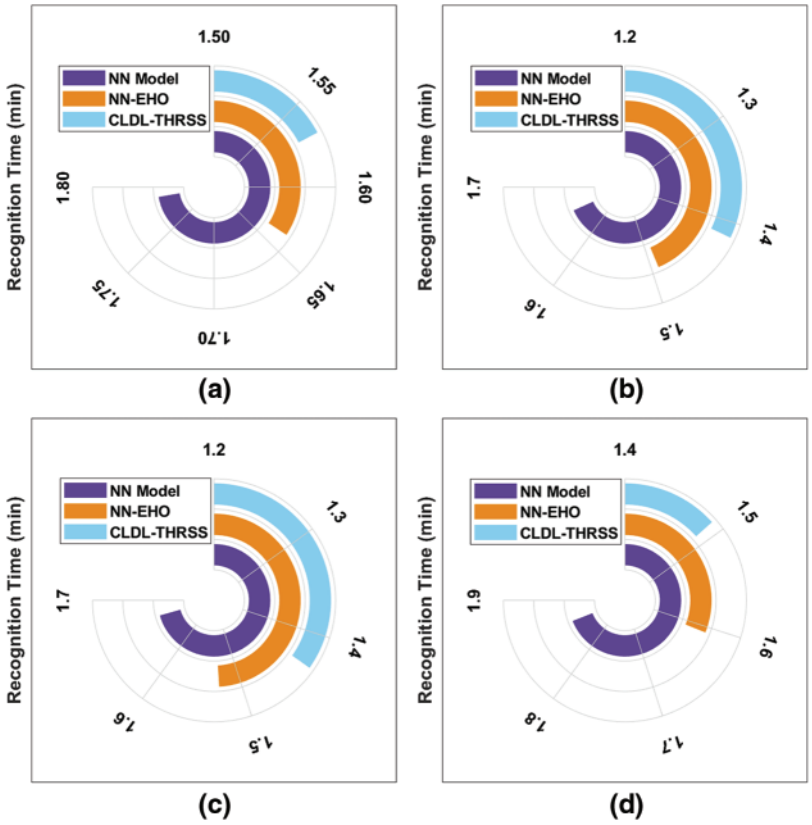


**Figure 8:** RT analysis of CLDL-THRSS technique with 4 images

For instance, with image 1, the CLDL-THRSS technique has required lesser RT of 1.569 min whereas the NN and NN-EHO techniques have needed RT of 1.789 and 1.637 min respectively. Meanwhile, with image 4, the CLDL-THRSS technique has reached to reduce RT of 1.860 min whereas the NN and NN-EHO techniques have accomplished higher RT of 1.607 and 1.489 min respectively.

Tab. 4 and Fig. 9 offer a detailed speech synthesizer performance analysis of the CLDL-THRSS technique [23]. The experimental values denoted that the CLDL-THRSS technique has accomplished effectual outcomes with the maximum recognition accuracy under all distinct hidden layers. For instance, with 1-hidden layer, the CLDL-THRSS technique has gained maximum recognition accuracy of 40.32% whereas the MLP-HMM, GMM-HMM, and NN-HMM techniques have obtained minimum recognition accuracy of 18%, 17.54%, and 10.03% respectively. Moreover, with 5-hidden layer, the CLDL-THRSS technique has attained increased recognition accuracy of 77% whereas the MLP-HMM, GMM-HMM, and NN-HMM techniques have obtained minimum recognition accuracy of 65.15%, 62.65%, and 39.18% respectively. Furthermore, with 7-hidden layer, the CLDL-THRSS technique has resulted in higher recognition accuracy of 62.42% whereas the MLP-HMM, GMM-HMM, and NN-HMM techniques have obtained minimum recognition accuracy of 59.68%, 48.98%, and 44.88% respectively.

**Table 4:** Comparative analysis of CLDL-THRSS technique under varying layers with existing approaches

| No. of hidden layers | CLDL-THRSS | MLP-HMM | GMM-HMM | NN-HMM |
|---|---|---|---|---|
| Layer-1 | 40.32 | 18.00 | 17.54 | 10.03 |
| Layer-2 | 63.78 | 53.76 | 51.48 | 27.57 |
| Layer-3 | 69.94 | 63.56 | 53.76 | 37.13 |
| Layer-4 | 77.45 | 63.10 | 62.19 | 38.27 |
| Layer-5 | 77.00 | 65.15 | 62.65 | 39.18 |
| Layer-6 | 69.25 | 65.61 | 59.91 | 42.83 |
| Layer-7 | 62.42 | 59.68 | 48.98 | 44.88 |

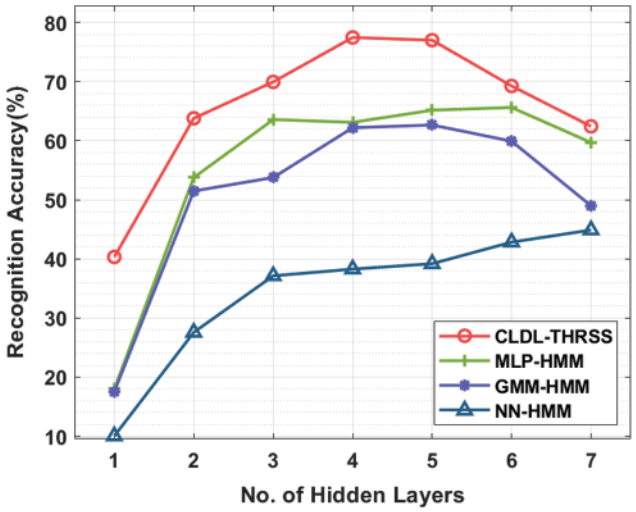


**Figure 9:** Recognition accuracy analysis of CLDL-THRSS technique

From the abovementioned figures and tables, it is apparent that the CLDL-THRSS technique has attained maximal detection performance over the other systems.

## 5 Conclusion

This study has presented a novel CLDL-THRSS procedure for the recognition and speech synthetization of the Arabic handwritten characters. The proposed CLDL-THRSS technique encompasses preprocessing, segmentation, CapsNet based HCR, CS based hyperparameter tuning, DNN-HMM based speech synthesizer. The design of CS algorithm helps to appropriately adjust the hyperparameters of the CapsNet model. In order to authenticate the effective performances of the proposed CLDL-THRSS model, a detailed experimental validation process takes place and investigates the outcomes interms of different measures. The experimental outcomes denoted that the CLDL-THRSS model has outperformed the compared methods. Therefore, the CLDL-THRSS technique is employed as a powerful mechanism for Arabic HCR and speech synthesizer for indigenous language. In future, hybrid DL models can be used for HCR and speech synthesizer.

**Funding Statement:** This Research was funded by the Deanship of Scientific Research at University of Business and Technology, Saudi Arabia.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] L. M. Lorigo and V. Govindaraju, "Offline Arabic handwriting recognition: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 712–724, 2006.

[2] M. T. Parvez and S. A. Mahmoud, "Arabic handwriting recognition using structural and syntactic pattern attributes," *Pattern Recognition*, vol. 46, no. 1, pp. 141–154, 2013.

[3] S. Sitaram, K. R. Chandu, S. K. Rallabandi and A. W. Black, "A survey of code-switched speech and language processing," arXiv preprint arXiv:1904.00784, 2019.

[4] P. M. ee, S. Santra, S. Bhowmick, A. Paul, P. Chatterjee *et al.,* "Development of GUI for text-to-speech recognition using natural language processing," in *2018 2nd Int. Conf. on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, Kolkata, India, pp. 1–4, 2018.

[5] M. Mager, X. G. Vasques, G. Sierra and I. Meza, "Challenges of language technologies for the indigenous languages of the Americas," arXiv preprint arXiv:1806.04291, 2018.

[6] A. Bharath, S. Madhvanath, "Online handwriting recognition for Indic scripts," in: Govindaraju, V., Setlur, S. (eds.) in *Guide to OCR for Indic Scripts Document Recognition and Retreival*, London: Springer, pp. 209–236, 2008.

[7] R. J. Kannan, R. Prabhakar and R. M. Suresh, "Off-line cursive handwritten tamil character recognition," in *2008 Int. Conf. on Security Technology*, Sanya, China, pp. 159–164, 2008.

[8] A. E. Sawy, M. Loey and E. B. Hazem, "Arabic handwritten characters recognition using convolutional neural network," *WSEAS Transactions on Computers*, vol. 5, pp. 11–19, 2017.

[9] 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.

[10] N. Shanthi and K. Duraiswamy, "A novel SVM-based handwritten Tamil character recognition system," *Pattern Analysis and Applications*, vol. 13, no. 2, pp. 173–180, 2010.

[11] V. K. Jain, S. Kumar and S. L. Fernandes, "Extraction of emotions from multilingual text using intelligent text processing and computational linguistics," *Journal of Computational Science*, vol. 21, pp. 316–326, 2017.

[12] N. Altwaijry and I. A. Turaiki, "Arabic handwriting recognition system using convolutional neural network," *Neural Computing and Applications*, vol. 33, no. 7, pp. 2249–2261, 2021.

[13] N. Lamghari and S. Raghay, "Recognition of arabic handwritten diacritics using the new database DBAHD," *Journal of Physics: Conference Series*, vol. 1743, no. 1, pp. 1–9, 2021.

[14] F. Alwajih, E. Badr, S. Abdou and A. Fahmy, "DeepOnKHATT: An end-to-end arabic online handwriting recognition system," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 35, no. 11, pp. 1–13, 2021, https://doi.org/10.1142/S0218001421530062.
[15] R. Hu, F. Fang, C. C. Pain and I. M. Navon, "Rapid spatio-temporal flood prediction and uncertainty quantification using a deep learning method," *Journal of Hydrology*, vol. 575, pp. 911–920, 2019.
[16] Y. Lee, A. Rabiee and S. Y. Lee, "Emotional end-to-end neural speech synthesizer," arXiv preprint arXiv:1711.05447, 2017.
[17] S. Sabour, N. Frosst and G. E. Hinton, "Dynamic routing between capsules," in *Proc. of the Advances n Neural Information Processing Systems 30: Annual Conf. on Neural Information Processing Systems*, Long Beach, CA, USA, pp. 3859–3869, 2017.
[18] H. Chao, L. Dong, Y. Liu and B. Lu, "Emotion recognition from multiband EEG signals using CapsNet," *Sensors*, vol. 19, no. 9, pp. 1–16, 2019.
[19] X. S. Yang and S. Deb, "Engineering optimisation by cuckoo search," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 1–17, 2010.
[20] E. Y. Bejarbaneh, A. Bagheri, B. Y. Bejarbaneh, S. Buyamin and S. N. Chegini, "A new adjusting technique for PID type fuzzy logic controller using PSOSCALF optimization algorithm," *Applied Soft Computing*, vol. 85, pp. 1–26, 2019.
[21] G. E. Dahl, D. Yu, L. Deng and A. Acero, "Context-dependent pretrained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.
[22] L. Li, Y. Zhao, D. Jiang, Y. Zhang, F. Wang *et al.,* "Hybrid deep neural network–hidden markov model (dnn-hmm) based speech emotion recognition," in *2013 Humaine Association Conf. on Affective Computing and Intelligent Interaction*, Geneva, Switzerland, pp. 312–317, 2013.
[23] S. Kowsalya and P. S. Periasamy, "Recognition of Tamil handwritten character using modified neural network with aid of elephant herding optimization," *Multimedia Tools and Applications*, vol. 78, no. 17, pp. 25043–25061, 2019.