Journal of Disability Research

2025 | Volume 4 | Pages: 1–13 | e-location ID: e20240080
DOI: 10.57197/JDR-2024-0080

Designing a Novel CNN–LSTM-based Model for Arabic Handwritten Character Recognition for the Visually Impaired Person

Manel Ayadi[1,*], Nesrine Masmoudi[2], Latifa Almuqren[1], Hadeel Saeed Alshahrani[3] and Raneem Oudah Aljohani[4]

[1]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh 11564, Saudi Arabia
[2]Management Information System Department, Taibah University, Yanbu 46421, Saudi Arabia
[3]College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh 11564, Saudi Arabia
[4]College of Computer Science and Engineering, Taibah University, Yanbu 46421, Saudi Arabia

Correspondence to:
Manel Ayadi*, e-mail: mfayadi@pnu.edu.sa
Nesrine Masmoudi, e-mail: nmasmoudi@taibahu.edu.sa
Latifa Almuqren, e-mail: laalmuqren@pnu.edu.sa
Hadeel Saeed Alshahrani, e-mail: 442002377@pnu.edu.sa
Raneem Oudah Aljohani, e-mail: TU4451880@taibahu.edu.sa

Received: December 25 2023; Revised: June 8 2024; Accepted: June 9 2024; Published Online: January 3 2025

ABSTRACT

The field of healthcare has undergone a radical change as a consequence of the latest advancements in deep learning. Recently, the development of visual substitutes for visually impaired people (VIP) has significantly aided research on assistive technology (AT). However, there is still little usage of ATs to understand the fundamental meaning of various written substances. This research presents a novel hybrid model of convolutional neural network (CNN) and long short-term memory (LSTM) for Arabic handwritten character recognition (AHCR) to present AT for VIP. This technique offers a practical way to improve accessibility for those who are visually impaired. The CNN's convolutional layers are used to capture both local and global patterns by extracting hierarchical information from the intricate and varied shapes in Arabic characters. After feeding these features into the LSTM network, the model comprehends the sequential nature of Arabic writing by capturing contextual information. Combining these two architectures allows the model to take advantage of temporal as well as spatial correlations, which improves recognition accuracy for complex Arabic letters. In this work, text-to-speech technology is also used to turn the recognized text into audio. To validate the model performances, we employed the publically available Arabic Handwritten Characters Dataset, which includes a range of writing situations and styles. The proposed CNN–LSTM model outperforms conventional methods for AHCR and achieves the highest accuracy of 98.07% over the state-of-the-art approaches.

KEYWORDS

DL, CNN, LSTM, visual impairment, Arabic character recognition, text-to-speech

INTRODUCTION

In the last 10 years, machine learning (ML) techniques and applications have helped to create significant developments in the assistive technology (AT) space. Researchers are making use of these developments to continually enhance people's quality of life, particularly for those who are disabled or have serious health issues (Bhattamisra et al., 2023). Visual ability plays a crucial role in performing various activities of daily life. Loss of sight is a major problem for anyone. Globally, more than 1 billion people live with some form of disability (World Health Organization, 2011). According to the World Health Organization, one-sixth of the world's population is visually impaired (Pydala et al., 2023). A person may have mild, moderate, severe, or complete blindness as their level of vision impairment. Recent developments in deep learning (DL) have made research projects and innovative solutions for those with visual impairments more well liked (Khosrobeigi et al., 2022). For those who are visually impaired, these solutions can provide safety, confidence, and self-sufficiency when performing daily duties. Improving accessibility for the blind and visually impaired is a constant motivator in the ever-changing field of AT (Ang et al., 2016). Research studies on AT cover a variety of conditions, including cognitive, auditory, and visual impairments. The AT converts identified elements, such as text or objects, into audio

translations *via* a text-to-speech conversion mechanism (Dokania and Chattaraj, 2022). Through the usage of an aural interface, users can access the content by having spoken words corresponding to the identified written or visual content. Text-to-speech technology improves accessibility for people who might have trouble understanding textual or visual information by using synthesized speech to communicate the meaning of detected items (Alahmadi et al., 2023). Digital engagement and autonomous reading are greatly facilitated by the ability to accurately recognize handwritten characters. The complex character of Arabic script poses particular obstacles in the context of Arabic handwritten character recognition (AHCR) (El-Awadly et al., 2023). However, because Arabic is a complicated alphabet with multiple forms for the same letter, there is a dearth of comparable studies on Arabic literature, making it a difficult task (Varma and Zisserman, 2003). The advancement of optical character recognition (OCR) for handwritten text, particularly Arabic script, is greatly dependent on artificial intelligence (AI). By examining visual characteristics and finding patterns that correspond to particular letters, AI approaches like ML, DL, and neural networks (NNs) teach OCR systems to recognize and interpret handwritten text more accurately. AI-powered OCR has several applications, such as digitizing old documents, automating data entry, and helping people with vision impairments. It can also separate text into individual characters and recognize language and script (Zhang et al., 2018). To detect intricate patterns and associations, DL employs NNs, whereas ML techniques utilize statistical models. This allows OCR to distinguish between the distinctive characteristics of each character and the context in which it appears (Li et al., 2021).

There is a clear research space in AHCR for the visually handicapped. The goal is to develop sophisticated models that can handle the complexities of Arabic script without sacrificing readability. While existing systems achieve impressive progress in character identification, they are not always able to meet the particular difficulties presented by handwritten Arabic characters (Maalej and Kherallah, 2020). By presenting the proposed novel convolutional neural network–long short-term memory (CNN–LSTM)-based model specifically created to meet the unique demands of the visually impaired, this research seeks to close this gap. The shortcomings of the existing character recognition models and OCR technologies highlight the research gap. Arabic characters have complicated forms and contextual variances that make them difficult for many conventional models to interpret effectively, even after being trained on a variety of scripts. This discrepancy is made worse when these models are modified to account for users' visual impairments, highlighting the need for a customized solution. Furthermore, the current state of AHCR is deficient in the thorough integration of state-of-the-art technologies, like CNN and LSTM architectures (Alrobah and Albahli, 2022). Although individual designs have shown effectiveness in specific scenarios, the synergistic combination of CNN and LSTM remains uncharted ground in the field of visually impaired Arabic character recognition.

The restricted investigation of transfer learning strategies in AHCR, particularly about Arabic script, is another area of study deficiency. There is a lack of research in this particular area about the possible advantages of using pre-trained vision transformer models for feature extraction and generalization. This disparity restricts the flexibility and resilience of current models when faced with the variety of handwriting styles common in Arabic scripts (Zerdoumi et al., 2022). The proposed CNN–LSTM model fills these research gaps and attempts to establish new standards for inclusivity and accuracy while also introducing a fresh take on AHCR. Transfer learning is incorporated with a special emphasis on the complexities of Arabic script, which adds a great deal to the body of knowledge and is a revolutionary step toward giving blind people a sophisticated and usable reading and comprehension aid. DL has changed a number of healthcare areas in the last few years, from therapy recommendations to medical imaging examinations. The CNNs and LSTMs have exposed peculiar operations in tasks containing illness analysis, medication growth, and persistent consequence forecasts (Ganesan et al., 2022). For instance, CNNs have achieved considerable improvement in processing medical images, even in outdoor areas, which speeds up the analysis and decision-making process. Similarly, LSTMs are outstanding at classifying temporal correlations in sequential data, which assist in predicting the type of disease, its history, and its stages. Furthermore, DL models are widely used to improve healthcare and enhance the quality of life (Zhou et al., 2021). Through the use of novel methods such as the CNN–LSTM model, it will assist visually impaired people (VIP) to read and understand the text efficiently. The main contributions of this research are highlighted as follows:

- Due to the complex shapes and contextual variations in Arabic handwritten characters, an accurate recognition system is a challenging task. The combination of CNN and LSTM architectures considerably improves the accuracy by effectively capturing both spatial features and sequential features. This novel approach results in a more accurate recognition system compared to conventional techniques.
- The proposed model is designed to work in real time, making it suitable for incorporation into portable devices and AT. The capability to process information in real time supports VIP in reading and comprehending the handwritten Arabic language in routine circumstances.
- The proposed model is capable of providing recognition outcomes in several output modalities, including text and audio. This adaptability enables various likes and supports among the visually impaired population, offering greater usage and accessibility.
- The proposed model is flexible and scalable, allowing it to be utilized not only with Arabic but also with handwritten text in other languages that possess similar characteristics. The model offers a significant addition to the wider domain of handwritten text recognition due to its versatility.

The remainder of the paper is organized as follows. The Related Works section presents related works and discusses prior potential solutions for vision impairment. The Methodology section describes the proposed methodology, elaborating on the CNN–LSTM-based model's approach

to identifying Arabic handwritten letters adapted for those with visual impairments. The Experiments section covers the experimental phase, which includes a description of the trials, training techniques, and specific parameters used in the execution of the proposed model. The Conclusion section concludes this work and provides insight for future efforts.

RELATED WORKS

In this section, we discuss the different state-of-the-art techniques used by AHCR for VIP. In recent years, major advances in AHCR have been made in response to the growing demand for inclusive technologies. The necessity to improve accessibility for VIP is driving this progress (Younis, 2017). Vision impairment is a prevalent ailment that varies in intensity. The provision of visual substitution through equipment has been made possible through the integration of AT. This enables the VIP to understand their surroundings (Tapu et al., 2020; Swathi et al., 2021). The study conducted by Shelton and Ogunfunmi (2020) combined their trained model with a webcam to distinguish things in real time. Lastly, their application took advantage of text-to-speech conversion to audibly speak what their trained model identified, enabling users to understand their environments. They tried two changes to the original architecture to enhance its performance for their application of image recognition for the visually handicapped after receiving preliminary findings from the retrained AlexNet. The fully connected layers underwent the first modification, and the convolutional layers underwent the second modification. According to their findings, 88% of exterior object data and 92% of internal object data were recognized. The research published by El-Sawy et al. (2017) gathered the Arabic Handwritten Characters Dataset (AHCD) containing 16,800 isolated character pictures. To train and test the dataset, they used a CNN DL architecture. They employed optimization strategies to improve the CNN's performance. On testing data, their proposed CNN had an average classification accuracy of 94.9%. The work conducted by Aichaoui et al. (2022) integrated AraBART with the SPIRAL dataset, which comprises eight types of error for training and testing. The best recall was allegedly 0.863 for space-related issues. However, an unequal distribution of error categories was observed, which could explain why the findings were not better. However, they claimed that this high result highlights the possibility of robust outcomes. If the model is trained on more datasets, as well as the possibility of using more text embeddings. The study conducted by Awni et al. (2022) investigates the performance of three deep CNNs for detecting Arabic handwritten words that have been randomly initialized. Then, for the same job, they assessed the ResNet-18 model's performance, which was pre-trained using the dataset provided by ImageNet. Finally, utilizing the ResNet-18 model, we suggest a method for progressively transferring mid-level word picture representations across two successive stages. To determine the most successful approach to implementing transfer learning, they did four distinct series of studies utilizing IFN/ENIT (v2.0p1e) and AlexU-W, two well-known offline Arabic handwritten word datasets. Their findings show that employing ImageNet as a source dataset enhances recognition accuracy of the 10 most commonly incorrectly classified terms of the IFN/ENIT dataset by 14%, while their suggested technique improves recognition accuracy by 35.45%. They achieved a recognition accuracy of up to 96.11% in the entire dataset, which is approximately a 2.5% enhancement over previous state-of-the-art techniques. The work done by Maalej and Kherallah (2018) proposes a new system that is built on the combination of two deep NNs. First, a CNN extracts feature from raw images automatically, followed by a bidirectional long short-term memory network and a connectionist temporal classification layer for sequence labeling. This model is validated using an enhanced IFN/ENIT database developed using data augmentation techniques. This hybrid design produces enticing performance. It outperforms both handcrafted feature-based techniques and automatic feature extraction models. According to the findings of the experiments, the recognition rate is 92.21%. The work done by Boualam et al. (2022) tested the model on a set that was split from the same augmented set used for training. The development of such high accuracy is not an unusual phenomenon; nonetheless, it may indicate an overfitted model because no genuine generalization testing was undertaken. As a result, the authors could do considerably more robust testing of their model using the other unused parts of the IFN/ENIT dataset. Furthermore, they inverted the word error rate reporting to 91.79% rather than 8.21%, which may cause some confusion. The work conducted by ElAdel et al. (2015) introduced an NN architecture based on the fast wavelet transform and the AdaBoost algorithm. The Arabic handwritten character classification system was learned and tested using the IESK-arDB dataset, which contains 6000 segmented characters. The categorization rate for the various character groupings is 93.92%. The work published by Alahmadi et al. (2023) utilized the Microsoft Common Objects in Context dataset to validate the work presented in their proposed model. To improve the training process, image preprocessing techniques were used, and hand annotation guarantees that every image is accurately labeled. Through the use of text-to-speech conversion, the module gives VIP audio information to help them recognize obstacles. Following 4000 training iterations, the model attains an accuracy of 96.34% on test photos taken from the dataset, with a loss error rate of 0.073%.

The ability to detect and recognize text from images has become a popular topic. Several apps are created for text identification using DL approaches (Abbadeni et al., 2013). However, in the case of the Arabic language, very limited research is conducted, while most studies focus on English or other extensively used languages (Patel, 2013). The research published by Lawgali et al. (2014) presented a new segmentation-based framework for AHCR. An artificial neural network was employed to detect the outline of a character using data gathered from discrete cosine transform. Their approach was tested by the IFN/ENIT database, which has 6033 characters. The average rate of recognition is 90.73%. Despite advances in AHCR, there is still a significant gap in addressing the special demands of visually impaired individuals who use Arabic characters. Existing models frequently lack the resilience required to handle the

wide range of writing styles, ligatures, the rich morphological complexities of the Arabic language itself, and variations found in Arabic script (Alrobah and Albahli, 2022; Fakhet et al., 2022; Alyahya et al., 2023). This paper seeks to fill that void by providing a novel CNN–LSTM-based model suited for AHCR with an emphasis on improving accessibility for the visually impaired. The proposed model is used to capture both temporal and spatial correlations in the handwritten text images (Tayal et al., 2021). CNNs are good at extracting local features from images, while LSTMs work well at processing sequential data that have long-range dependencies, which makes them useful for tasks like handwriting recognition.

METHODOLOGY

Due to the complex ligatures and cursive writing style of Arabic characters, AHCR poses distinct difficulties. The capability to reliably identify and comprehend handwritten text can sufficiently improve the self-reliance and written communication of VIP. In this context, the proposed work introduces a novel CNN–LSTM-based model that combines CNN and LSTM to overcome these problems and assist VIP in recognizing Arabic handwritten characters. In this work, we observed that CNNs are useful for retrieving spatial characteristics of the Arabic text from the images. Within the framework of AHCR, the CNN part of the model is adept at obtaining fine information from Arabic characters, like curves, strokes, and letter-to-letter connections. Convolutional filter layers recognize various characteristics at different abstraction levels, ranging from edges to complex patterns, which are unique to Arabic letters. The LSTM component observes the spatial information retrieved by the CNN and comprehends it in the setting of adjacent letters. This sequential understanding is essential for identifying handwriting in cursive style because a character's look can be modified by its nearby letters. The proposed model can be incorporated into any portable device, like camera-equipped reading aids or mobile phones. These gadgets can be used to record handwritten notes, text, or any other type of text; understand it instantly; and turn it into audible speech. Utilizing this approach, people with visual impairments can read handwritten letters from their friends, relatives, or colleagues on their own by enabling the recognition of Arabic handwriting. This encourages their personal and professional autonomy and lessens their reliance on sighted support.

Figure 1 shows the proposed methodology; in the first step, the system receives its input in the form of an image comprising Arabic text. This input image has been preprocessed to ensure consistent size and uniform dimensions. The normalized image is then processed employing several convolutional layers. In this study, the CNN takes the image and extracts its spatial characteristics. Next, the LSTM network receives these features. The LSTM network acquires temporal connections and contextual information by processing the feature sequence. The character with the highest likelihood is chosen as the recognized character. Utilizing text-to-speech technologies, the detected text is converted into audible speech.

Data collection

In this research, we used the publicly available AHCD. During the selection process, we gave special consideration to datasets that are not only large and typical of Arabic handwritten characters but also specifically designed to meet the needs of blind people. The Arabic script's contextual

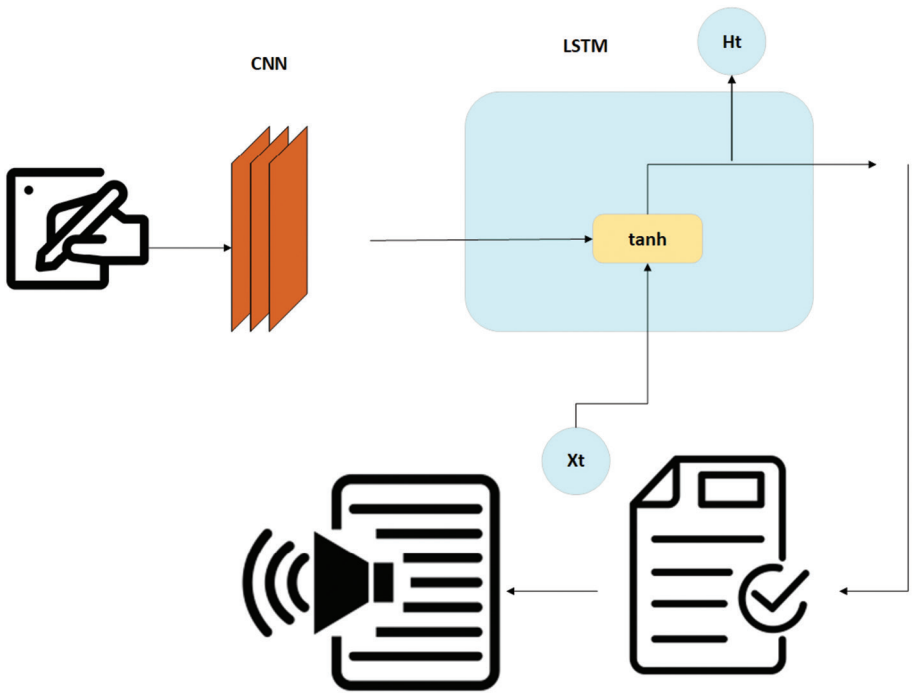

Figure 1: Steps performed in the proposed methodology. Abbreviations: CNN, convolutional neural network; LSTM, long short-term memory.

variations, ligatures, and writing styles are all represented in the dataset's wide variety of handwritten characters.

Data preprocessing

We performed thorough data preprocessing before training the proposed model to achieve optimal model performance. We preprocessed the AHCR dataset using a wide range of Python packages. We used Pillow (PIL) for image loading and manipulation, NumPy for numerical operations, OpenCV for image processing tasks including color conversion and scaling, and TensorFlow and Keras for the development of a DL model as part of our preprocessing pipeline. To improve dataset variety, we also used the Augmentor package for image augmentation. These well-chosen packages made it easier to prepare the dataset for the CNN–LSTM model's training and assessment later on, guaranteeing reliable character recognition that is appropriate for those with visual impairments. Normalization of photographs, scaling to a standard format, and augmentation procedures to allow for variances in writing styles are all part of the process.

Normalization of images

The process of normalization entails scaling the image's pixel values to a conventional range, such as [0, 1] or [−1, 1]. To promote steady convergence during model training, it is imperative that certain features not take precedence over others during the training process.

Resizing to a consistent format

Resizing all photos to a standard format guarantees that they have the same dimensions and can be consistently input into the model. This phase is critical because CNNs have fixed-size input layers, and having images of variable sizes may cause issues during training.

Augmentation techniques

Augmentation is the process of introducing random adjustments to images, such as rotation, flipping, or brightness and contrast changes. The goal of augmentation is to increase the variety of the training dataset artificially (Eltay et al., 2021). By exposing the model to several modified versions of the same image, it becomes more robust and capable of generalizing to previously unknown data. This is especially useful when dealing with writing style changes that may not be fully reflected in the original dataset.

Handling imbalances in the dataset

Imbalances in the dataset emerge when particular classes are not adequately represented or heavily represented in comparison with others. Oversampling and undersampling strategies are used to tackle this. Oversampling entails making more copies of instances from the underrepresented class, and undersampling entails reducing the number of instances from the overrepresented class (Nassiri et al., 2022). We employed the synthetic minority oversampling technique to produce synthetic samples of Arabic handwritten characters from the minority class to address imbalances in the dataset. To mitigate the class imbalance, we used a random undersampling strategy with replacement in our Arabic handwritten character dataset. This involved randomly eliminating instances from the majority class. Balancing the dataset in this way prevents the model from being biased toward the majority class during training, providing equal representation and correct learning for all classes. The Arabic handwritten characters are the primary emphasis of the system's use of CNNs for precise text detection and recognition from visual inputs (Alahmadi et al., 2023). The recognized text is then shown on the user's screen, resulting in an interface. The system incorporates text-to-speech conversion methods, which enable the text to be identified to be converted into speech with ease. Text-to-speech systems improve accessibility for VIP by generating speech. These sounds feel natural by anticipating and synthesizing the auditory patterns associated with certain language aspects. This enables visually impaired users to access written content through an auditory interface. This all-inclusive paradigm stresses a comprehensive and inclusive user experience for people with visual impairments, while also addressing the complexities of AHCR. Table 1 represents examples of preprocessing steps.

Table 1: Examples of simplified preprocessing step representations.

Preprocessing step	Explanation	Example
Image normalization	Pixel values should be scaled to a defined range.	Original pixel values: [100, 150, 200] Normalized pixel values: [0.2, 0.5, 0.8]
Resizing to a standardized layout	Resize the image to a consistent scale.	Original image size: 100 × 100 pixels Resized image size: 64 × 64 pixels
Methods of augmentation	For data diversity, introduce random transformations.	Original image: 100 × 100 Augmented image: 64 × 64
Managing dataset imbalances	To rectify class imbalances, both an oversample and an undersample should be used.	Original dataset: Class A (1000 instances), Class B (200 instances) Balanced dataset: Class A (1000 instances), Class B (1000 instances)

Convolutional neural network

The CNN is the most widely used and effective DL method in image processing. It has various applications in text analysis, object detection, and recognition (Agarap, 2018; Tayal et al., 2021). CNNs are made up of numerous fundamental layers, which are then followed by the corresponding activation functions. Actually, there are three basic layers that make up the CNN framework: the fully connected layer, the pooling layer, and the convolutional layer. We accomplished the task of collecting hierarchical features from input images that fall to the CNN component. It consists of several convolutional layers, which are followed by a pooling layer, which together make up the feature extraction process. The purpose of these layers used in this work is to capture the unique characteristics and spatial patterns seen in Arabic handwritten characters. The proposed CNN-based LSTM framework is shown in Figure 2.

Input layer

The input layer receives a fixed dimension of grayscale images. Each pixel in these images indicates the grayscale color's strength, which is displayed as matrices of pixel values. During preprocessing, each image is resized to 64 × 64 pixels. If the width and height of the input images are W_{in} and H_{in}, respectively, then the input layer can be signified as a matrix X with a size of $W_{in} \times H_{in}$.

Convolutional layer

The convolution layer consists of various filters that scan the input image to extract features. Each filter has a specific design. Following the convolutional process, an activation function called rectified linear unit (ReLU) is employed to produce non-linearity. Let W_f and H_f be the filter width and height, respectively, and let F be the number of filters. The following yields the convolution operation as denoted by Equation (1):

$$C_i = \acute{O}(W_i * X + b_i), \tag{1}$$

where σ is the activation function, b_i is the bias, $*$ indicates convolution, and C_i is the output feature map of the ith filter. The weights of the filter are W_i. Similarly, let us look at a straightforward example where a little portion of the input image I is subjected to the 3 × 3 filter K as represented in Equation (2):

$$F(x, y) = \sum_{i,j=1}^{3} I(x+i-1, y+j-1) \cdot K(i, j), \tag{2}$$

where $F(x, y)$ represents the feature map value at (x, y) position. Pixel intensity in the input image is represented

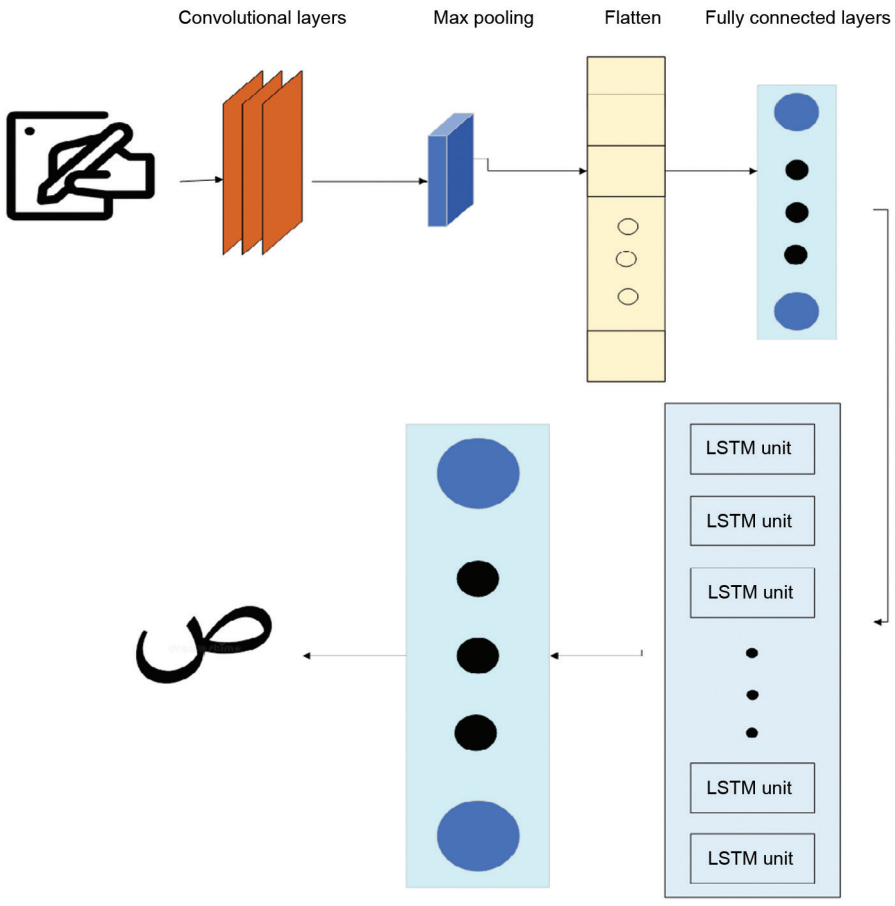

Figure 2: The proposed CNN-based LSTM framework. Abbreviations: CNN, convolutional neural network; LSTM, long short-term memory.

by $I(x + i - 1, y + j - 1)$. The filter weights are shown by $K(i, j)$. By calculating the weighted sum of the local region's pixel intensities, this procedure highlights specific spatial patterns that the filter K represents. Thus, a low-level characteristic that the filter identified is represented by the output $F(x, y)$. Subsequent layers then use the outputs of earlier layers as input. As we proceed further into the network, it enables them to learn progressively more complicated patterns.

Pooling layer

An NN's pooling layer is in charge of downsampling the input feature maps' spatial dimensions to minimize their size while preserving crucial data. Typical pooling layer types, such as max pooling or average pooling, combine local data in a methodical way to produce a condensed representation that helps with feature extraction and network computing efficiency (Younis, 2017). In this work, downsampling is accomplished with max pooling as shown in Equation (3), where the pooled feature map is denoted by P_i:

$$P_i = \max(C_i). \tag{3}$$

Long short-term memory network

The LSTM networks are a type of recurrent neural network (RNN), which is capable of grasping order dependencies in predicting sequences (Graves, 2012). The LSTM network is used to learn temporal connections and contextual information based on the sequential pattern of Arabic letters. It receives output from the CNN components and sequentially analyzes it, taking into account each character's context about its neighboring characters.

Input sequence

In the input sequence, the feature vector is obtained, which is produced by the CNN model. In this study, T represents the CNN output features' sequence length. The LSTM receives the input sequence as given in Equation (4):

$$\text{Seq} = (P_1, P_2, \ldots, P_L). \tag{4}$$

A set of values known as Seq is represented by this equation. P_1, P_2, P_L, where L represents the sequence's length, are the elements that make up the sequence. Each P_i represents a distinct element in the sequence. The output feature maps from the CNN pooling layer are used to generate this sequence (Seq). The sequence's P is all related to the pooling feature map at a particular time step (i), which can be any value between 1 and L. This sequence is processed by the LSTM network, which improves recognition performance overall by capturing temporal dependencies and patterns within the feature maps. Thus, the suggested model, $\text{Seq} = (P_1, P_2, \ldots, P_L)$ is a representation of the feature map sequence that is obtained from the CNN and input into the LSTM for further processing.

LSTM cell

The purpose of LSTM networks is to extract temporal relationships from textual data. The network is applied to Arabic character data in order to acquire the associations between characters in a sequence, which is then used to comprehend the structure and semantics of the Arabic language. The input sequence is processed by the LSTM cell as defined in Equation (5):

$$h_t, c_t = \text{LSTM}(P_t, h_{t-1}, c_{t-1}), \tag{5}$$

where the hidden state is denoted by h_t, the cell state is represented by c_t, and the time step is indicated by t.

Hybrid model integration

The CNN component's output serves as the input sequence for the LSTM component, resulting in a combined CNN–LSTM architecture. The CNN represents spatial information, while the LSTM represents sequential dependencies. The model's prediction for the given handwritten character is represented as the final output.

Fully connected layer

After being flattened, the LSTM output is joined to a fully connected layer as Equation (6) indicates:

$$y = \text{softmax}(W_{fc} \cdot \text{flatten}(h_T) + b_{fc}), \tag{6}$$

where W_{fc} stands for the weight's matrix, b_{fc} stands for the bias, and flatten stands for the flattening operation on the LSTM output. Y stands for the output prediction in this equation.

Loss function

The loss function measures the error between predicted and actual values. Training entails modifying the model's parameters to minimize this error, improving the model's predictive accuracy. We employed a cross-entropy loss function in the proposed study. It measures the performance of a classification model whose output is a probability value between 0 and 1. The cross-entropy loss increases as the predicted probability diverges from the actual label. Equation (7) signifies the loss function:

$$L = -\sum_{i}^{N} y_i \cdot \log(\hat{y}_i), \tag{7}$$

where N is the number of classes in this case. The ground truth is represented by y_i, and $\hat{y}_i$ is the estimated probability.

Optimizer

The last argument needed to assemble the model before training phenomena is an optimizer. The Keras library contains several optimizer variations, including Adam, root mean

square, stochastic gradient descent (SGD), and others. Adam (Kingma and Ba, 2014) is used to recognize hand gestures. During training, the "Adam" optimizer is used to lower the loss that is determined at the end of each epoch. The adaptive estimate of first- and second-order moments serves as the foundation for the SGD method that this optimizer employs. This method works well for issues requiring sophisticated data/parameter processing since it is computationally efficient, requires less memory, and is invariant to diagonal rescaling of gradients. In this study, we used the Adam optimizer, a widely used stochastic optimization algorithm, to fine-tune the parameters of the proposed CNN–LSTM model. Equation (8) governs the optimization process:

$$\theta_t = \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \in}, \tag{8}$$

where to update the proposed model's parameters θ_t, we alter them based on the gradients' average direction $\hat{m}_t$ and scale $\hat{v}_t$. The learning rate α controls the size of the step, and we add a little value $\in$ to avoid any problems with dividing by very small amounts. Adam is the perfect fit for our model because of its great computational efficiency, minimal memory needs, and invariance to diagonal rescaling of gradients. These qualities enable us to successfully explore the high-dimensional parameter space and improve the performance of the suggested model.

EXPERIMENTS

This section discusses the overall results of the experiment. We go over the training process, experimental design, and evaluation standards to show how the proposed framework performs the process of character recognition and text-to-speech conversion.

Training process

This empirical study aims to assess the efficacy of the suggested methodology in developing a hybrid CNN–LSTM-based AHCR model tailored to the needs of VIP. We put into practice a brand-new CNN–LSTM model. We commence the procedure by gathering data, leveraging the AHCD, which is made freely available. We concentrate on datasets that are particularly created to meet the requirements of those who are blind. This dataset captures the various writing styles, ligatures, and contextual variants that come with using Arabic script. We next carry out a thorough data preprocessing to maximize model performance. We utilize multiple Python packages, including PIL, NumPy, OpenCV, TensorFlow, and Keras, to standardize image sizing, apply augmentation techniques to improve dataset diversity and normalize images. Specifically, managing dataset imbalances *via* oversampling and undersampling techniques is essential to avoid biases during training. Using input photos, the CNN component is intended to extract hierarchical features. Using pooling layers for downsampling and convolutional layers for feature extraction, the CNN focuses on identifying distinct features and spatial patterns in Arabic handwritten letters. Concurrently, the LSTM part is presented to acquire contextual knowledge and temporal relationships through the sequential patterns present in Arabic script. After receiving the CNN's output, the LSTM examines each character in turn, taking into account its context for its neighboring characters. Combining the CNN and LSTM components creates a hybrid model that makes use of both the LSTM's capacity to capture sequential dependencies and the CNN's spatial information in a synergistic way. The output of the softmax function is the prediction for a given handwritten character, and the final output of the model is obtained through a fully linked layer. The recognized text is converted into audio signals using text-to-speech conversion technology. Thus, VIP can easily listen to the audio text. A cross-entropy loss function is used to measure the difference between expected and actual values to train the model. To reduce this loss and increase the model's forecast accuracy repeatedly, the parameters are adjusted during the training phase. This strategy is essential for improving the efficacy of the suggested CNN–LSTM-based model for AHCR, particularly when it comes to meeting the special requirements of people with visual impairments. We use precision, recall, and F1-score metrics to evaluate the performance of the proposed model. These metrics provide us feedback, to further improve the accuracy and efficiency of the proposed model by revealing how efficiently it knows Arabic handwritten characters.

Evaluation criteria

In this section, we will employ the theoretical assessment equations listed below, denoted as Equations (9)–(12). We aim to evaluate the system's text detection and recognition capabilities and calculate this proposed system's accuracy value using the following formulas.

Accuracy

This is a basic statistic that calculates the proportion of accurately predicted occurrences to total instances:

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N}, \tag{9}$$

where T_p is the total Arabic characters that were recognized correctly, F_N is the number of Arabic characters omitted, F_P is the quantity of non-Arabic characters that are mistakenly recognized as Arabic, and T_N is the quantity of non-Arabic characters successfully identified.

Precision

This measure shows how accurate positive forecasts are. The calculation is as follows:

$$\text{Precision} = \frac{T_P}{T_P + F_P}. \tag{10}$$

Table 2: Experimental configuration.

Experimental configuration	Explanations
Model architecture	CNN–LSTM
Dataset	AHCD
Data preprocessing	Image standardization, augmentation, normalization
Python packages used	Pillow, NumPy, OpenCV, TensorFlow, Keras
Handling imbalanced data	Oversampling and undersampling techniques
CNN	Extracts hierarchical features
LSTM	Captures contextual knowledge and temporal relationships
Training loss function	Cross-entropy
Training phase	Iterative parameter adjustment for loss reduction
Evaluation metrics	Accuracy, precision, recall, F1-score
Special considerations	Tailored to the needs of visually impaired individuals
Frameworks used	TensorFlow and Keras
Data splitting	The entire dataset is divided into three parts
Training set	80% of the dataset
Validation set	10% of the dataset
Test set	10% of the dataset
Learning rate	0.001
Batch size	64
Number of epochs	20
Convolutional layers	2
LSTM units	128
Dropout rate (CNN)	0.25
Dropout rate (LSTM)	0.5
Optimizer	Adam
Loss function	Cross-entropy
Image size	64 × 64 pixels
Augmentation techniques	Random rotation
Oversampling technique	Synthetic minority oversampling technique
Undersampling technique	Random undersampling with replacement

Abbreviations: AHCD, Arabic Handwritten Characters Dataset; CNN, convolutional neural network; LSTM, long short-term memory.

Recall

This metric, which is computed as follows, assesses how well the model captures all positive instances. It is determined as follows:

$$\text{Recall} = \frac{T_P}{T_P + F_N}. \tag{11}$$

F1-score

The harmonic mean of recall and precision is known as the F1-score. It offers a harmony between recall and precision. It is computed as follows:

$$\text{F1-score} = \frac{2 \times P \times R}{P + R}. \tag{12}$$

The configuration that has been used to train the hybrid CNN–LSTM model for AHCR is shown in Table 2.

RESULTS AND DISCUSSION

In this section, we explain the suggested model's outcomes and comparisons with several baseline techniques. We thoroughly assess the performance measures of the proposed model using various metrics, such as accuracy, precision, recall, and F1-score. We also demonstrate the merits and limitations of the proposed model in comparison with state-of-the-art works. The CNN-based LSTM network is deployed in three different steps on the AHCR dataset for VIP, as displayed in Table 3. The CNN model's initial implementation demonstrated good performance with an accuracy

Table 3: Proposed model with relevant baseline techniques.

Model	Dataset	Precision (%)	Recall (%)	F-score (%)	Accuracy (%)
CNN	AHCD	89.87	90.9	90.38	90.32
LSTM	AHCD	93.04	95.07	94.04	94.65
CNN–LSTM	AHCD	96.43	98.32	97.37	98.07

Abbreviations: AHCD, Arabic Handwritten Characters Dataset; CNN, convolutional neural network; LSTM, long short-term memory.

of 90.32%, highlighting its ability to recognize Arabic handwritten letters. The CNN's convolutional layers allowed it to extract hierarchical features, allowing it to recognize various spatial patterns in the characters. However, the CNN had several limitations, particularly when it came to preserving the sequential dependencies inherent in Arabic script, which might cause problems when dealing with ligatures and contextual variants.

The LSTM model has been established in response to the CNN model's limitations and has shown significant improvement. The LSTM tackled the difficulty of collecting sequential patterns in Arabic characters with a precision of 93.04%, a recall of 95.07%, an F-score of 94.04%, and an accuracy of 94.65%. Because the LSTM excels at grasping temporal links and context, it is better suited to recognizing characters with complex structures. This capability is evident in improved precision and recall, implying fewer false positives and false negatives. The benefits of the LSTM compensated for the sequential constraints of the CNN, resulting in a more robust model adapted to the demands of visually impaired users.

The CNN–LSTM model managed better than the two separate models, with precision, recall, F-score, and accuracy values of 96.43%, 98.32%, 97.37%, and 98.07%, respectively. The hybrid model produced a synergistic impact by merging the sequential understanding of LSTM with the spatial information extraction of CNN. By addressing the shortcomings of both CNN and LSTM, this combination created a potent AHCR system. The CNN–LSTM model showed that it could reduce false positives and false negatives at the same time since its precision and recall were higher than those of the individual models. The most successful method for satisfying the unique needs of people with visual impairments was to take a comprehensive approach that made use of both spatial and sequential information. This allowed for accurate and dependable character recognition in a variety of writing styles and contextual variations.

A graphic depiction of the training and testing accuracy for each of the three models CNN, LSTM, and CNN–LSTM is shown in Figure 3. The graphic illustrates the accuracy values that demonstrate the performance of each model in the training and testing stages of the ML process. It shows that in the initial stages, the CNN model is quite good at recognizing patterns in space, but it faces trouble capturing the

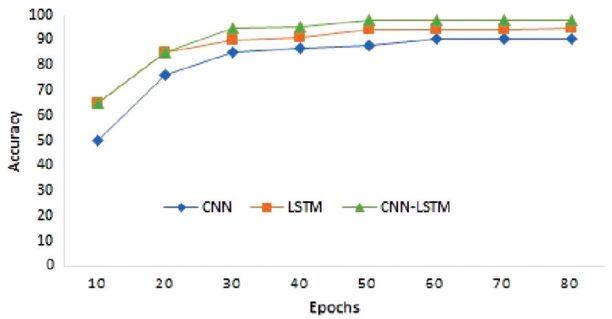

Figure 3: Training and testing accuracy of CNN, LSTM, and CNN–LSTM. Abbreviations: CNN, convolutional neural network; LSTM, long short-term memory.

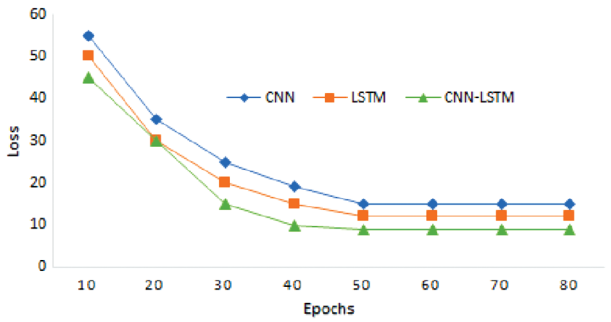

Figure 4: Training and testing loss of CNN, LSTM, and CNN–LSTM. Abbreviations: CNN, convolutional neural network; LSTM, long short-term memory.

sequential relationships in Arabic letters. The LSTM is better at identifying the intricate structures in Arabic characters since it is good at catching sequential patterns. The proposed method, which is specifically designed to meet the demands of visually impaired individuals, effectively utilizes the strengths of both architectures by utilizing LSTM after CNN and combining them in the CNN–LSTM model. This results in superior performance in AHCR. Similarly, the training and testing losses for the CNN, LSTM, and CNN–LSTM models are shown in Figure 4.

Comparison with baseline approaches

Table 4 shows a comparison of the suggested approach to current state-of-the-art approaches. We compare the results of the proposed methodology with the best methods on different Arabic datasets for VIP. A thorough evaluation of pertinent research, each providing its conclusions on datasets about the Arabic language, is part of the comparative analysis. The study conducted by Alrobah and Albahli (2022) focused on the application of a CNN–support vector machine (SVM) approach to the Arabic language utilizing the Hijja dataset. The goal of the study was to use the combined capabilities of CNNs and SVMs to extract features and classify data effectively. The study yielded an astounding accuracy of 96.30%, demonstrating the ability of CNN-SVM to navigate the difficulties of Arabic language data. By using the DL technique on the Flickr8k dataset, the model developed in 2022 was capable of AHCR, with an accuracy rate of 86%. In 2022, the Arabic Corpus dataset for audio data was examined using a CNN–ReLU model, obtaining an impressive accuracy of 90%. In the same year, the EfficientNetB4 model was used on the Arabic Alphabets Sign Language dataset, yielding a 95% accuracy. An augmented reality approach using the AHT2D dataset produced a noteworthy accuracy of 95% in the follow-up study in 2023. In 2023, the Mask R-CNN technique on the Pascal VOC 2007 and Pascal VOC 2012 datasets yielded an accuracy of 83.90%. Although this method is an improvement, it might have trouble picking up on minute differences in AHCR.

In comparison with these baseline methodologies, our suggested AHCD model greatly outperforms them. We obtain 98.07% accuracy, outperforming state-of-the-art achievements in Arabic language processing. Through the merging of CNN and LSTM networks, the AHCD model is specifically

Table 4: Comparisons of the proposed model with baseline technique.

Study	Year	Dataset	Method	Language	Accuracy (%)
Alrobah and Albahli (2021)	2021	Hijja	CNN–SVM	Arabic	96.30
Ganesan et al. (2022)	2022	Flickr8k	DL	Arabic	86.00
Bhatia et al. (2022)	2022	Arabic Corpus dataset	CNN–ReLU model	Arabic	90.00
Zakariah et al. (2022)	2022	ArASL	EfficientNetB4	Arabic	95.00
Ouali et al. (2023)	2023	AHT2D	Augmented reality	Arabic	95.00
Alzahrani and Al-Baity (2023)	2023	Pascal VOC 2007, 2012	Mask R-CNN	Arabic	83.90
Proposed model	2023	AHCD	CNN–LSTM	Arabic	98.07

Abbreviations: AHCD, Arabic Handwritten Characters Dataset; ArASL, Arabic Alphabets Sign Language; CNN, convolutional neural network; DL, deep learning; LSTM, long short-term memory; ReLU, rectified linear unit; SVM, support vector machine.

designed to capture complicated connections in sequential data. This combination enables our model to extract spatial data successfully using CNN and capture temporal dependencies using LSTM. Our approach excels in several areas. To begin, the model exhibits remarkable accuracy, suggesting its ability to grasp and handle Arabic language data. Second, by employing a new CNN–LSTM architecture, our model can capture both short-term and long-term dependencies, ensuring a more comprehensive comprehension of the data's sequential structure. This capacity is especially important in jobs involving context, such as natural language processing.

Discussion

The proposed model is tested on the AHCD, which shows robustness across a variety of character shapes and styles. This adaptability demonstrates our model's usefulness in a wide range of Arabic language processing applications. Various factors are taken into account for user experience testing. For example, in evaluating accessibility, the AHCR system integrates with visually impaired users' workflows by taking into account accessibility features like braille displays and screen readers. The ease of use is taken by assessing the AHCR system user interface to find any usability obstacles that VIP face while using the model. The accuracy is evaluated in identifying Arabic handwritten characters under various circumstances and distinct writing styles. Through continuous user experiences and feedback, we iteratively improve the proposed system's performance.

In this work, we captured the diversity of Arabic handwritten characters in the actual world, including differences in writing styles, writing contexts, and environmental factors. However, more efforts are needed to fully capture the variations in writing styles and the impacts of context on writing. In this work, we observed that a smaller or less diverse dataset may provide overfitting or biased results, which may have an impact on the model's performance and generalizability. Handwriting is a dynamic process, which varies continuously in terms of writing styles, sizes, and forms. This makes it difficult to accurately recognize characters, particularly in the case of limited samples and when written in various contexts. Addressing these limitations and challenges is crucial for ensuring the practical relevance and effectiveness of the proposed CNN–LSTM model in real-world applications. Strategies for mitigating these challenges include data augmentation and transfer learning. Data augmentation is employed to increase the model resilience for various samples using syntactic data, which reflects a range of handwriting styles and variances. Transfer learning is used to improve the training process to adapt the proposed model to new writing styles by deploying pre-trained models.

In conclusion, our proposed CNN–LSTM-based model, AHCD, marks a big step forward in Arabic language processing. Its remarkable precision, combined with its capacity to capture both spatial and temporal relationships, distinguishes it as a cutting-edge solution that exceeds previous baseline methodologies and establishes a new benchmark for excellence in the sector. The proposed CNN–LSTM-based model, created in 2023 on the AHCD, surpassed the baseline approaches. The shortcomings of the baseline methods were solved by the novel hybrid architecture, which combined CNN and LSTM. This customized method works well for identifying Arabic characters, which makes it particularly useful in applications where the visually impaired need to identify characters accurately. The strength of this work is combining both CNN and LSTM to extract local features through CNN and maintain the temporal dependencies between words using LSTM. The limitation of this work is the computational complexity due to the convolution process used for feature extraction. Also, the performance of the model heavily relies on the quality and quantity of the training data.

CONCLUSION

In this study, we developed the CNN–LSTM-based model to recognize Arabic text and provide assistance for VIP. An auditory interface is also added for the detected Arabic characters, to improve its usability. The model takes the efficacy of integrating the CNN and RNN by achieving a remarkable accuracy of 98.07% on the AHCD. The proposed methodology surpasses baseline techniques by tackling the unique constraints given by Arabic script, setting the groundwork for more inclusive technological solutions. This research not only advances the science of character recognition but also has the potential to improve the quality of life for people with visual impairments.

In the future, we plan to create a custom dataset that contains several Arabic handwriting styles. We intend to investigate transfer learning methodologies adapted to specific activities, as well as to improve user experience by introducing

real-time involvement. Furthermore, we plan to test the model's robustness in a variety of environmental scenarios.

ACKNOWLEDGMENTS

The authors extend their appreciation to the King Salman Center for Disability Research for funding this work through Research Group no. KSRG-2022-066.

COMPETING INTERESTS

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The data will be made available on request.

REFERENCES

Abbadeni N., Ghoneim A. and Alghamdi A. (2013). Program educational objectives definition and assessment for quality and accreditation. *Int. J. Eng. Pedagogy*, 3(3), 33-46.

Agarap A.F. (2018). Deep learning using rectified linear units (ReLU). 1, 1-7, arXiv preprint arXiv:1803.08375.

Aichaoui S.B., Hiri N., Dahou A.H. and Cheragui M.A. (2022). Automatic building of a large Arabic spelling error corpus. *SN Comput. Sci.*, 4(2), 108.

Alahmadi T.J., Rahman A.U., Alkahtani H.K. and Kholidy, H. (2023). Enhancing object detection for VIPs using YOLOv4_Resnet101 and text-to-speech conversion model. *Multimodal Technol. Interact.*, 7(8), 77.

Alrobah N. and Albahli S. (2021). A hybrid deep model for recognizing Arabic handwritten characters. *IEEE Access*, 9, 87058-87069.

Alrobah N. and Albahli S. (2022). Arabic handwritten recognition using deep learning: a survey. *Arab. J. Sci. Eng.*, 47(8), 9943-9963.

Alyahya H.M., Ismail M.M.B. and Al-Salman A. (2023). Intelligent ResNet-18 based approach for recognizing and assessing Arabic children's handwriting. In: *2023 International Conference on Smart Computing and Application (ICSCA)*, IEEE, Hail, Saudi Arabia, 5-6 February 2023, pp. 1-7.

Alzahrani N. and Al-Baity H.H. (2023). Object recognition system for the visually impaired: a deep learning approach using Arabic annotation. *Electronics*, 12(3), 541.

Ang L.-M., Seng K.P. and Heng T.Z. (2016). Information communication assistive technologies for visually impaired people. *Int. J. Ambient Comput. Intell.*, 7(1), 45-68.

Awni M., Khalil M.I. and Abbas H.M. (2022). Offline Arabic handwritten word recognition: a transfer learning approach. *J. King Saud Univ.-Comput. Inf. Sci.*, 34(10), 9654-9661.

Bhatia S., Devi A., Alsuwailem R.I. and Mashat A. (2022). Convolutional neural network based real time Arabic speech recognition to Arabic Braille for hearing and visually impaired. *Front. Public Health*, 10, 898355.

Bhattamisra S.K., Banerjee P., Gupta P., Mayuren J., Patra S. and Candasamy M. (2023). Artificial intelligence in pharmaceutical and healthcare research. *Big Data Cogn. Comput.*, 7(1), 10.

Boualam M., Elfakir Y., Khaissidi G. and Mrabti M. (2022). Arabic handwriting word recognition based on convolutional recurrent neural network. In: *WITS 2020: Proceedings of the 6th International Conference on Wireless Technologies, Embedded, and Intelligent Systems*, Springer, Fez, Morocco.

Dokania H. and Chattaraj N. (2022). An assistive interface protocol for communication between visually and hearing-speech impaired persons in internet platform. *Disabil. Rehabil. Assist. Technol.*, 19, 1-14.

ElAdel A., Ejbali R., Zaied M. and Amar C.B. (2015). Dyadic multi-resolution analysis-based deep learning for Arabic handwritten character classification. In: *2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, Vietri sul Mare, Italy, pp. 807-812.

El-Awadly E.M.K., Ebada A.I. and Al-Zoghby A.M. (2023). Arabic handwritten text recognition systems and challenges and opportunities. *Egypt. J. Lang. Eng.*, 10(2), 84-103.

El-Sawy A., Loey M. and El-Bakry H. (2017). Arabic handwritten characters recognition using convolutional neural network. *WSEAS Trans. Comput. Res.*, 5(1), 11-19.

Eltay M., Zidouri A., Ahmad I. and Elarian Y. (2021). Improving handwritten Arabic text recognition using an adaptive data-augmentation algorithm. In: *Proceedings, Part I 16 of the Document Analysis and Recognition–ICDAR 2021 Workshops*, Lausanne, Switzerland, 5-10 September 2021, Springer.

Fakhet W., El Khediri S. and Zidi S. (2022). Guided classification for Arabic characters handwritten recognition. In: *2022 IEEE/ACS 19th International Conference on Computer Systems and Applications (AICCSA)*, IEEE, Abu Dhabi.

Ganesan J., Azar A., Alsenan S.A., Kamal N.A., Qureshi B. and Hassanien A. (2022). Deep learning reader for visually impaired. *Electronics*, 11(20), 3335.

Graves A. (2012). Long short-term memory. In: *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer, Berlin, Heidelberg; pp. 37-45.

Khosrobeigi Z., Veisi H., Hoseinzade E. and Shabanian H. (2022). Persian optical character recognition using deep bidirectional long short-term memory. *Appl. Sci.*, 12(22), 11760.

Kingma D.P. and Ba J. (2014). Adam: a method for stochastic optimization. 1, 1-15, arXiv preprint arXiv:1412.6980.

Lawgali A., Angelova M. and Bouridane A. (2014). A framework for Arabic handwritten recognition based on segmentation. *Int. J. Hybrid Inf. Technol.*, 7(5), 413-428.

Li D., Wang R., Chen P., Xie C., Zhou Q. and Jia X. (2021). Visual feature learning on video object and human action detection: a systematic review. *Micromachines*, 13(1), 72.

Maalej R. and Kherallah M. (2018). Convolutional neural network and BLSTM for offline Arabic handwriting recognition. In: *2018 International Arab Conference on Information Technology (ACIT)*, IEEE, Werdanye, Lebanon, pp. 1-6.

Maalej R. and Kherallah M. (2020). Improving the DBLSTM for on-line Arabic handwriting recognition. *Multimed. Tools Appl.*, 79, 17969-17990.

Nassiri N., Lakhouaja A. and Cavalli-Sforza V. (2022). Evaluating the impact of oversampling on Arabic L1 and L2 readability prediction performances. In: *Proceedings of NISS 2021, Networking, Intelligent Systems and Security*, Springer, Kenitra, Morocco.

Ouali I., Halima M.B. and Wali A. (2023). An augmented reality for an arabic text reading and visualization assistant for the visually impaired. *Multimed. Tools Appl.*, 82, 43569-43597.

Patel A. (2013). *Arab Nahdah: The Making of the Intellectual and Humanist Movement*. Edinburgh University Press.

Pydala B., Kumar T.P. and Baseer K.K. (2023). Smart_Eye: a navigation and obstacle detection for visually impaired people through smart app. *J. Appl. Eng. Technol. Sci.*, 4(2), 992-1011.

Shelton A. and Ogunfunmi T. (2020). Developing a deep learning-enabled guide for the visually impaired. In: *2020 IEEE Global Humanitarian Technology Conference (GHTC)*, IEEE, Seattle, Washington, USA, pp. 1-8.

Swathi K., Vamsi B. and Rao N.T. (2021). A deep learning-based object detection system for blind people. In: *Proceedings of SMART-DSC, Smart Technologies in Data Science and Communication*, Springer, Guntur, India, pp. 223-231.

Tapu R., Mocanu B. and Zaharia T. (2020). Wearable assistive devices for visually impaired: a state of the art survey. *Pattern Recognit. Lett.*, 137, 37-52.

Tayal, A., Gupta J., Solanki A., Bisht K., Nayyar A. and Masud M. (2021). DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases. *Multimed. Syst.*, 28, 1417-1438.

Varma, M. and A. Zisserman. (2003). Texture classification: are filter banks necessary? In: *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.1-8, IEEE, Madison, Wisconsin.

World Health Organization. (2011). World report on disability 2011. World Health Organization, Geneva.

Younis K.S. (2017). Arabic hand-written character recognition based on deep convolutional neural networks. *Jordanian J. Comput. Inf. Technol.*, 3(3), 186.

Zakariah M., Alotaibi Y.A., Koundal D., Guo Y. and Mamun Elahi M. (2022). Sign language recognition for Arabic alphabets using transfer learning technique. *Comput. Intel. Neurosc.* 2022, 4567989.

Zerdoumi S., Jhanjhi N.Z., Ahmed R., Hashem I.A.T. and Gabralla L.A. (2022). Adaptive auto-encoder for extraction of Arabic text: invariant, font, and segment. Research Square, v1, 2022, pp.1-43. 10.21203/rs.3.rs-2190247/v1.

Zhang Z., Wang H., Liu S. and Xiao B. (2018). Consecutive convolutional activations for scene character recognition. *IEEE Access*, 6, 35734-35742.

Zhou S.K., Greenspan H., Davatzikos C., Duncan J.S., van Ginneken B., Madabhushi A., et al. (2021). A review of deep learning in medical imaging: imaging traits, technology trends, case studies with progress highlights, and future promises. *Proc. IEEE Inst. Electr. Electron. Eng.*, 109(5), 820-838.