Interpreting Doctors' Notes Using Handwriting Recognition and Deep Learning Techniques

Maheswari G[1], Ramya.R[2], Babitha.S[1], Sriram.R[2], Jennis .M[2] and RamPrasad .P[2]

1Assistant Professor, Department of Information Technology, Hindustan Institute of Technology and Science, Chennai,India
2Assistant Professor, Department of Artificial Intelligence and Data Science PSNA College of Engineering and Technology

E-mail : mahe.jul13@gmail.com, ramyarajamanickam2596@gmail.com, sbabitha@gmail.com, msriram2018@psnacet.edu.in, Jennisj2020@psnacet.edu.in, ramjul2002@gmail.com

2024 Ninth International Conference on Science Technology Engineering and Mathematics (ICONSTEM) | 979-8-3503-6509-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICONSTEM60960.2024.10568799

Abstract- This research provides a deep learning approach and handwriting recognition to understand medical notes. The system processes and interprets textual medical notes using connectionist temporal classification (CTC) and optical character recognition (OCR) techniques. The written text is transcribed using the OCR component, and then the CTC algorithm is utilized to analyze the text and give significance to particular note portions. The suggested system's ability to correctly understand and arrange written medical information is demonstrated by training it on prescription notes from doctors taken from a dataset. According to the final forecast, the system can read medical notes with high accuracy, making medical professionals' usage and accessibility of these records easier. The suggested paradigm offers fast and precise access to patient information, which could increase the efficacy and efficiency of medical care.

Keywords— Optical Character Recognition, Connectionist Temporal Classification, Doctors' Notes, Accuracy, Interpreting.

I. INTRODUCTION

Machine learning's deep learning subfield focuses on using multi-layered neural networks for prediction and data-driven learning. Deep learning provides various benefits over machine learning techniques. First, by extracting features from unprocessed input data, deep learning algorithms eliminate human feature engineering requirements. Second, deep learning models are more resilient to changes in the data because they can manage high data volumes and noise. Thirdly, deep learning models can learn hierarchical data representations, which enables them to function well in tasks requiring comprehension of the relationships or context of the data. Furthermore, deep learning models can enhance their performance over time through continual learning. Deep learning has proven effective in many applications, including speech recognition, self-driving cars, computer vision, and natural language processing.

Deep learning— especially convolutional neural networks (CNNs) and recurrent neural networks (RNNs)—has proven especially effective in handwriting identification tasks. By learning spatial hierarchies of features through several layers of convolution and pooling operations, CNNs can extract information from handwriting images. RNNs, on the other hand, are made to handle sequential input, like handwriting. By utilizing long short-term memory (LSTM) layers or gated recurrent units (GRU), they may identify temporal dependencies in the data.

Large datasets of handwritten text can be used to train CNNs and RNNs to recognize and transcribe new handwriting samples accurately by teaching them the characteristics and patterns of various handwriting styles. Before supplying the input image to the deep learning model, pre-processing methods like image enhancement and segmentation can be applied to make it better. Moreover, several research have combined CNNs and RNNs to leverage the advantages of both architectures and enhance recognition performance. Deep learning methods are dependable when used for handwriting recognition tasks. If research and development are done more, they might get even better.

Character recognition is identifying handwritten characters and converting them into a text encoded by a computer. This applies to several industries, including data automation and digitization—text produced by hand and forensic uses [1]. This method entails several processes to create the final forecast, including data collection, data preparation for analysis, feature extraction, and classification [2]. Handwriting recognition systems use many algorithms. Among them is the CNN method, a deep learning approach to image recognition that requires no explicit feature engineering or extraction but yields good results compared to traditional manual feature extraction techniques. [3], Experimental evaluations have shown the faster RCNN proposed by Albahli et al. [4] to be more efficient and accurate in identifying and categorizing handwritten numerals.

Recognition systems for numerous widely spoken languages, including Arabic, have been proposed. [5], Chinese [6], Tamil, etc. These languages are generally more cursive and, hence, challenging for recognition. In terms of handwritten cursive note recognition, OCR and CTC performed better. OCR is a superior deep learning method for handwriting recognition since it can accommodate many handwriting styles and font kinds. OCR systems can recognize the characteristics and patterns of many handwriting styles because they have been trained on extensive databases of handwriting samples. As a result, even when handwriting samples are written in various styles or font kinds, they can reliably identify and transcribe them. Furthermore, OCR systems may be trained to identify handwriting in several languages, increasing their versatility and adaptability to various use cases. Additionally, OCR systems can be combined with other pre-processing methods, including segmentation and image enhancement, to boost image quality

Our research presents a deep learning approach and handwriting recognition system: OCR, which can convert different kinds of scanned documents, including images from different scenarios, into editable text [7], and CTC, which can train sequence-to-sequence models to interpret doctor's notes without the need for aligned input-output pairs, thus leading to better patient care.The format of this document is as follows: In the second section of this study, we provide a summary of pertinent studies. In Section 3, the pipelined strategic linked dominant set algorithm is explained. Performance measurements are discussed in Section 4, and possible areas for future improvement are outlined in Section 5, which concludes the study.

II.RELATED WORKS

Ahamed et al.[8] a deep learning algorithm that is effective at recognizing Arabic handwritten characters has been suggested. The model is created to work with both single-font and multiple-font types of texts, and it focuses on two architectures: SVM and CNN. SVM determines the best boundary that divides the data into different classes, and CNN uses convolutional layers to learn and extract features from images. They also suggested a brand-new training rule based on the maximum interval minimum classification error backpropagation approach. The suggested model was evaluated on many databases. Moreover, it was discovered that the model's findings outperform conventional techniques for Arabic text recognition.

Kayani et al. [9] released a brand-new method for selective text encryption and identification in scanned document images. The main objectives of this model are efficiency and security. The approach combines optical OCR and NLP for text recognition while utilizing multiple data modalities. The text is located and then encrypted using a hybrid chaotic map and the Advanced Encryption Standard in Cipher Block Chaining mode.

The simulation results show the effectiveness and dependability of the suggested encryption technique. Espana et al.[10] We have presented a recognition method that combines convolutional blocks and one-dimensional bidirectional LSTM networks with CTC decoding. The main aim of the model is to recognize the Spanish language. They used a dataset containing handwritten sentences extracted from forms written by over 1600 individuals. The collection contained over 13,690 sentences from a vocabulary of around 3,280 words.

Tapotosh et al.[11] have proposed the use of CNN architecture to classify handwritten Bengali characters. They have used a dataset of over 230 Bengali handwritten images. White was used as the foreground colour in the first conversion of the photographs to black and white. With recognition accuracies of 96.55% and 96.20%, respectively, DenseNet121 and InceptionNetV3 also did well. All the designs fared equally well when the models were tested using characters that people could find challenging to identify. The ability to recognize perplexing characters is taken into account.

Chauhan et al.[12] have proposed a self-controlled (RDP) algorithm for feature extraction technique and 1-dimensional CNN for recognition. The reduced size of the convolution and shorter feature vectors in the proposed model have the advantage of faster training time, which is great for smaller devices. Hidden Markov models with directional characteristics produced the best results in the literature, and their script-independent feature extraction approach increases recognition accuracy over those findings.

Ghamdi et al. [13] have proposed a method for Arabic text recognition divided into four stages. They are classification, character segmentation, feature extraction, and pre-processing. The text image skeletons were first created using a thinning process. The following step introduces a novel chain code technique that uses an agent-based model to extract characteristics from images of Arabic text.

Using a character segmentation approach, the gathered attributes divide linked Arabic words into individual characters. In the fourth step, a PPM compression-based technique classifier detects Arabic text.

Prashanth et al. [14] proposed a handwritten Devanagari text recognition method. The model consists of three different CNN architectures. They are CNN, MLCNN and ACNN. MLCNN architecture was proposed by modifying the architecture of CNN. They used a dataset of over 38,700 images of the Devanagari language, which were collected from over 3000 subjects. They conducted a list of experiments with the collected dataset, and they had a minimum loss of 0.01%. This model has filled a significant percentage between the performance of the identification systems in the real world and its practical applications.

Bonyani et al. [15] have proposed a system for Persian language recognition. They developed a model with DenseNet and Xception architectures. They used three datasets to evaluate this architecture, namely HODA, Sadri, and Iranshahr, which have a wide range of handwriting styles and variations of letters based on position. The proposed architectures achieved an accuracy of 99.49% on the HODA dataset. On the Sadri dataset, the accuracy was. On the Iranshahr dataset, the accuracy rate for words was 98.99%. These results outperformed the existing proposed models. The evaluation parameters, like computation times, were also better than the alternative models.

III. Methodology

Handwriting recognition of doctor’s notes using deep learning involves Optical Character Recognition (OCR), Feature Extraction, Multi- Dimensional Recurrent Neural Network (MDRNN), and Connectionist Temporal Classification (CTC). OCR is a computer vision technique that enables the extraction of text from images and its conversion into machine-readable form. This technology is utilized in the project to process scanned doctor's notes and extract the handwritten text for further analysis. MDRNN is a type of recurrent neural network that can handle sequences of inputs with multiple dimensions. In the project, this deep learning algorithm is utilized to recognize and categorize the extracted handwritten text. CTC is a training algorithm in deep learning that predicts sequences of outputs from input sequences. It is used to train the MDRNN model to accurately recognize and classify the handwritten text in the doctor's notes.

The combination of these algorithms provides a powerful solution for the interpretation of doctor's handwritten notes. The OCR component extracts the text from the images, the MDRNN component recognizes and categorizes the handwritten text, and the CTC component trains the model to improve its accuracy. The end result is an efficient and accurate system for medical record management. The use of OCR, MDRNN, and CTC in this project provides a cutting-edge solution for the interpretation of doctor's handwritten notes. These algorithms are essential for enhancing the accuracy and efficiency of managing medical records, which can benefit patient care and the general state of healthcare.

The first step in recognizing doctors’ notes is to convert the scanned image of Prescription of patient into a machine-readable format, such as a text file or a sequence of characters (as shown in Fig 1.). To recognize the characters in an image, OCR algorithms often combine image processing techniques such as thresholding and morphological operations with pattern recognition techniques such as template matching. The OCR algorithm utilized will be determined by the project's specific requirements as well as the nature of the input data.

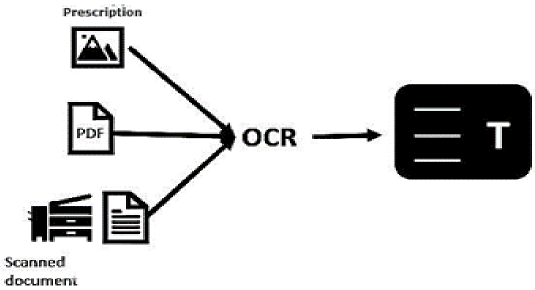


Fig 1. Scanned image is converted into digital text.

In mathematical terms, the OCR process can be described as a function, f(x), where x is the input image and f(x) are the output machine- readable text. The function f(x) takes the image x as input and applies a series of image processing and pattern recognition techniques to produce the output machine-readable text. Depending on the particular OCR technique employed, the form of the function f(x) will change. The goal of optical character recognition (OCR) is to convert handwritten text into a machine-readable format. The typical steps in the process include feature extraction, classification, segmentation, and pre-processing.

Pre-processing include sharpening the input image's characteristics, reducing noise, and cleaning it up. The process of breaking a picture up into smaller parts, such words, lines, and individual characters, is called segmentation. The process of removing relevant data from the segments—like shape, size, and stroke width—is known as feature extraction.

Finally, the recognized characters are combined to form words and lines, which are then output as machine-readable text. The accuracy of OCR for handwriting recognition depends on several factors, including the quality of the input image, the choice of feature extraction method, and the performance of the classifier. Nevertheless, the technology has made significant progress in recent years and has found applications in fields such as digital archives, handwriting analysis, and signature verification.

By extending the standard recurrent neural network (RNN) to accommodate multi-dimensional inputs including photos, videos, and audio signals, the MDRNN is a particular form of neural network design. Convolutional and recurrent layers are used in the architecture of MDRNNs to capture both the spatial and temporal relationships in the data. They have been utilized effectively in a number of applications, such as handwriting recognition, natural language processing, and image and video processing.

The CTC algorithm is a technique used for jobs like voice and handwriting recognition when only the input data and the intended transcription are supplied, with no knowledge of how well the two are aligned. Because each character might take up a different amount of space in handwriting, simple strategies like giving each character an equal amount of space are ineffective.

In the context of handwriting recognition, the intended output is represented as Y = [y1,y2,...,yU] while the input picture regions for a particular phrase are represented as X = [x1, x2,...,xT]. The objective is to properly calculate Y from X. The CTC method provides a distribution across all potential Ys from the input X, allowing enabling prediction of the outcome. (as shown in Fig2.).

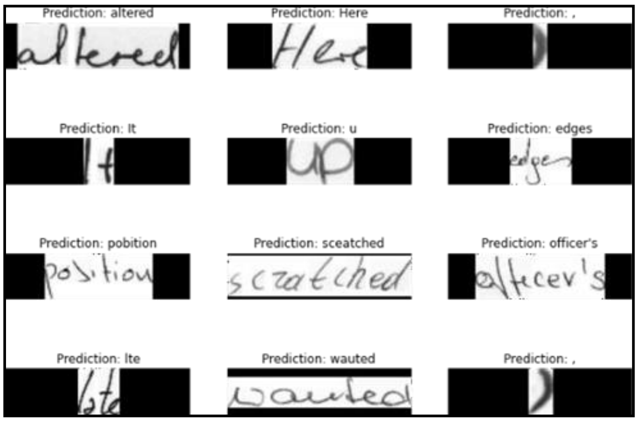


Fig 2. Model recognizing space between words, matching with the given text and producing output.

Algorithm for constructing the model

The following methods are commonly used in the multi-step process of identifying doctor's notes utilizing deep learning ideas and handwriting recognition.

Step 1: To recognize doctor's notes, follow these steps first. The handwritten notes are scanned by the OCR process, which transforms them into machine-readable text.

Step2: Following the notes' conversion to text, the following step involves extracting linguistic components crucial to the recognition task. For the deep learning model, this usually means turning the text into a numerical representation that it can use as input.

Step 3: This deep learning algorithm is the main one that used to recognize the doctor's notes.The MDRNN is trained on the recovered features to learn how to associate the input data with the target labels. The MDRNN is an example of a recurrent neural network; due to its ability to process multi-dimensional input data, it is highly effective at tasks such as handwriting recognition.

Step 4: The MDRNN is trained with the CTC method in order to predict the right label sequence for every input sample. Because it enables the network to accommodate variable-length input sequences and generate predictions at each time step, CTC is a well-liked approach for training recurrent neural networks in sequence labeling tasks.

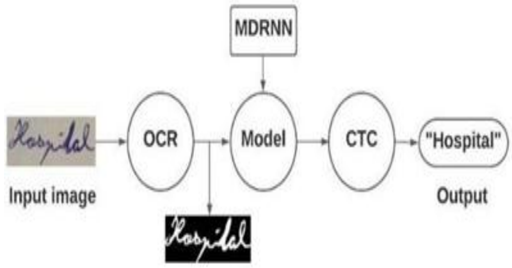


Fig 3. Workflow

In terms of the model's architecture, the MDRNN usually consists of multiple layers:

Input layer: The data that has been extracted from features is fed into the model at the input layer.

Recurrent layer: This is the primary layer of the MDRNN. It is made up of several recurrent neurons that process input data and keep track of a hidden state that represents the information they have already processed.

Output layer: Using the hidden state from the recurrent layer, the output layer forecasts the target labels. The network's output is transformed into a probability distribution over the target labels using the SoftMax layer.

These are the primary procedures and techniques for using deep learning and handwriting recognition to the identification of doctor's notes. Depending on the particular issue being handled and the kind of data being used, the model architecture and training procedure's specifics will change

Pseudo Code for Pre-processing

The pseudo code for pre-processing is Begin for subdir in each subdirectory in data_Ram:
processed_subdir =
create_processed_subdir(Processed_Cletus, subdir) for image_file in each image_file in subdir:
image = process _image (image_file) images .append (image)
label = get_label(subdir) if label not in label_dict:
label_dict[label] = next_label next_label += 1
labels.append(label_dict[label])
save_processed_image(processed_subdir, image_file, image)

A function that loads and preprocesses a collection of photographs kept in a directory is defined by the provided code. The function applies some image pre-processing steps, like turning the image into black and white, lowering noise, resizing it to a fixed size with a DPI of 300, and inverting the black and white pixels, iterating through each subdirectory in the supplied directory and creating a corresponding subdirectory inside a new directory to store pre-processed images. Next, the code retrieves the image's label from the subdirectory and uses a dictionary to transform it to an integer. The pre-processed image is appended to the pre-processed image list, and the label associated with it is appended to the label list. After processing every image, the function divides the data into training and testing sets, converts the labels into categorical values, and returns the training and testing sets of images together with their labels as numpy arrays. As a preliminary step, this function can be applied to picture classification tasks.

Pseudo Code for the model

The pseudo code for constructing the model is def
create_mdrnn_model(input_shape, num_classes):
print('Creating MDRNN model...') inputs =
Input(shape=input_shape) # Convolutional layers
conv = TimeDistributed(Conv2D(32, (3, 3),
activation='relu'))(inputs) conv =
TimeDistributed(MaxPooling2D((2, 2)))(conv)
conv = TimeDistributed(Conv2D(64, (3, 3),
activation='relu'))(conv) conv =
TimeDistributed(MaxPooling2D((2, 2)))(conv)
conv = TimeDistributed(Flatten())(conv) # LSTM layer
lstm = Bidirectional(LSTM(128,
return_sequences=True))(conv) lstm = Dropout(0.2)(lstm)
lstm = Bidirectional(LSTM(64))(lstm) # Output layer
output = Dense(num_classes, activation='softmax')(lstm)
model1 = Model(inputs=inputs, output=output)
return model

Define the function to create the CTC loss def
ctc_loss(y_true, y_pred):
# Create the placeholder for the labels
labels = placeholder(dtype='int32', shape=(None, None))
# Calculate the CTC loss
loss = ctc_batch_cost(labels, y_pred, y_true.shape[1],
y_pred.shape[1]) # Return the loss
return loss

Two functions that are helpful in creating an MDRNN for sequence recognition issues are defined in the provided code. The MDRNN model's layers, such as the convolutional and LSTM layers, are defined using the create_mdrnn_model function. These layers process input sequences and provide a probability distribution across a range of output classes. The difference between the true and predicted label sequences is measured by the CTC loss function for the model, which is defined by the ctc_loss function. When combined, these functions offer all the parts required to construct and train an MDRNN model for problems involving sequence recognition.

IV. Result and Analysis

In this paper, we suggested a deep learning approach combined with handwriting recognition to understand medical notes. The OCR algorithm is used by the system to digitize handwritten text, and the MDRNN and CTC algorithms are used to identify and categorize the text. We assembled the dataset for this study ourselves, which included prescription drugs.

Moreover, a performance analysis of the system showed that obtaining high accuracy was mostly dependent on the MDRNN and CTC algorithms. The MDRNN method was tasked with modeling the handwritten text's temporal dependencies, while the CTC algorithm handled the text sequences' variations in length. Additionally, the OCR algorithm was crucial to the functionality of the system by precisely converting the handwritten text into digital text.

We pre-processed our dataset using TensorFlow Keras and OpenCV (cv2). Image thresholding, median filtering, and scaling are just a few of the image processing operations available in the OpenCV computer vision library. Conversely, TensorFlow Keras is a high-level deep learning model development and training API. It helped us build and hone our MDRNN-CTC handwriting recognition model. We were able to quickly pre-process our dataset and create a dependable and accurate model for identifying medical prescriptions by merging these two potent libraries (Fig. 4). The dataset that has been annotated is displayed below (Fig 5.).

Fig 4. Pre-processed image using OpenCV and TensorFlow

```
a01-000u-01-00 ok 156 395 932 441 100 VBG nominating
a01-000u-01-01 ok 156 901 958 147 79 DTI any
a01-000u-01-02 ok 156 1112 958 208 42 AP more
a01-000u-01-03 ok 156 1400 937 294 59 NN Labour
a01-000u-01-04 ok 156 1779 932 174 63 NN life
a01-000u-01-05 ok 156 2008 933 237 70 NNS Peers
a01-000u-02-00 ok 157 408 1106 65 70 BEZ is
a01-000u-02-01 ok 157 541 1118 72 54 TO to
a01-000u-02-02 ok 157 720 1114 113 63 BE be
a01-000u-02-03 ok 157 916 1136 281 46 VBN made
a01-000u-02-04 ok 157 1281 1117 80 59 IN at
a01-000u-02-05 ok 157 1405 1140 64 35 AT a
a01-000u-02-06 ok 157 1544 1115 339 96 NN meeting
a01-000u-02-07 ok 157 1936 1106 91 74 INO of
a01-000u-02-08 ok 157 2092 1121 302 65 NN Labour
a01-000u-03-00 err 156 430 1290 177 59 NPTS M Ps
a01-000u-03-01 err 156 705 1296 431 54 NR tomorrow
a01-000u-03-02 err 156 1154 1346 9 10 . .
```

Fig 5. Annotation of the dataset.

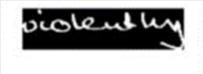
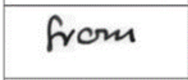
Raw image	Processed image	Annotation	Output after training
		a01-011u-05-07 ok 167 1830 1648 389 109 RB violently	violently
		a01-000u-00-06 ok 154 1896 757 173 72 IN from	From
		a01-011x-00-05 err 168 1795 720 230 90 JJ United	United

Fig 6. Sample Dataset

Fig 6 includes the procedure of the model. The first column contains the raw image. The pre-processed image is shown in the next column, followed by the actual text in the image (the "annotation" column) and the predicted text (the "output" column) from the handwriting recognition model. This table compares the predicted text with the actual text in order to evaluate the performance of the model.

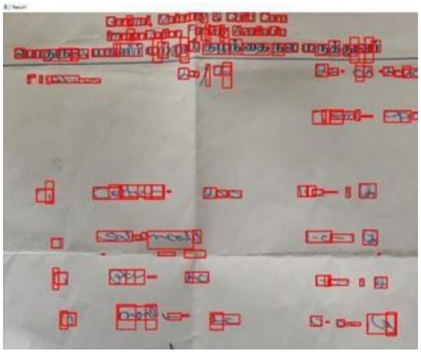


Fig 7. Text identified using OCR

The workflow for the model of handwriting recognition in doctors consists of several phases. The input image of the handwritten text is first sent to the model. The image is pre-processed to make it grayscale, resize, and remove noise in order to better the text. Next, OCR technology is used to identify the text in the image (as seen in Fig. 7). The output of the OCR is input into an MDRNN, a type of neural network used for sequence modeling. Using the OCR's output as its input, the MDRNN generates a number of hidden states. Then, a CTC algorithm is applied to these hidden states to get the final result.

V.Conclusion

In conclusion, the goal of our study was to create a deep learning model that could identify handwritten medical notes in order to make the process of converting manual notes into digital text easier. To accomplish the goal, the team used OCR, MDRNN, and CTC algorithms. To separate the handwritten notes into distinct words and characters, the OCR algorithm was utilized for pre-processing. Afterwards, the characters were identified using MDRNN, and the characters were subsequently translated into words using CTC. It was possible to correctly identify the text from the handwritten notes by combining these strategies. This experiment emphasizes how crucial it is to use deep learning methods for handwriting identification in the medical profession since they can result in quicker and more accurate medical record documentation, which will eventually improve patient care. Future research in this area can investigate the application of additional sophisticated algorithms and methods to raise the precision and effectiveness of handwriting recognition software.

References

[1] Joseph, F.J.J. Effect of supervised learning methodologies in offline handwritten Thai character recognition. Int. j. inf. tecnol. 12, 57–64 (2020).https://doi.org/10.1007/s41870-019-00366-y

[2] Hamida, S., El Gannour, O., Cherradi, B. et al. Handwritten computer science words vocabulary recognition using concatenated convolutional neural networks. Multimed Tools Appl (2022). https://doi.org/10.1007/s11042-022-14105-2

[3] Mushtaq, F., Misgar, M.M., Kumar, M. et al. UrduDeepNet: offline handwritten Urdu character recognition using deep neural network. Neural Comput & Applic 33, 15229–15252 (2021). https://doi.org/10.1007/s00521-021-06144-x

[4] Albahli, S., Nawaz, M., Javed, A. et al. An improved faster-RCNN model for handwritten character recognition. Arab J Sci Eng 46, 8509–8523 (2021). https://doi.org/10.1007/s13369-021-05471-4

[5] Altwaijry, N., Al-Turaiki, I. Arabic handwriting recognition system using convolutional neural network. Neural Comput & Applic 33, 2249–2261 (2021). https://doi.org/10.1007/s00521-020-05070-8

[6] D. Peng et al., "Recognition of Handwritten Chinese Text by Segmentation: A Segment-annotation-free Approach," in IEEE Transactions on Multimedia, https://doi.org/10.1109/TMM.2022.3 146771

[7] Misgar, M.M., Mushtaq, F., Khurana, S.S. et al. Recognition of offline handwritten Urdu characters using RNN and LSTM models. Multimed Tools Appl 82, 2053–2076 (2023). https://doi.org/10.1007/s11042-022-13320-1

[8] Ali, A. A. A., & Mallaiah, S. (2021). Intelligent handwritten recognition using hybrid CNN architectures based-SVM classifier with dropout. Journal of King Saud University - Computer and Information Sciences. https://doi.org/10.1016/j.jksuci.2021. 01.012

[9] Kayani, M., Ghafoor, A. & Riaz, M.M. Multi-modal text recognition and encryption in scanned document images. J Supercomput (2022). https://doi.org/10.1007/s11227-022-04912-7 5

[10] España-Boquera, S., Castro-Bleda, M.J. A Spanish dataset for reproducible benchmarked offline handwriting recognition. Lang Resources & Evaluation 56, 1009– 1022 (2022). https://doi.org/10.1007/s10579-022- 09587-3

[11] Tapotosh Ghosh, Abedin, MHZ., Al Banna, H. et al. Performance Analysis of State of the Art Convolutional Neural Network Architectures in Bangla Handwritten Character Recognition. Pattern Recognit. Image Anal. 31, 60–71 (2021). https://doi.org/10.1134/S105466182 1010089

[12] Singh, S., Chauhan, V.K. & Smith, E.H.B. A self-controlled RDP approach for feature extraction in online handwriting recognition using deep learning. Appl Intell 50, 2093– 2104 (2020). https://doi.org/10.1007/s10489-020- 01632-4

[13] Al Ghamdi, M.A. A Novel Approach to Printed Arabic Optical Character Recognition. Arab J Sci Eng 47, 2219–2237 (2022). https://doi.org/10.1007/s13369-021- 06163-9

[14] Prashanth, D.S., Mehta, R.V.K., Ramana, K. et al. Handwritten Devanagari Character Recognition Using Modified Lenet and Alexnet Convolution Neural Networks. Wireless Pers Commun 122, 349–378 (2022). https://doi.org/10.1007/s11277-021-

08903-4
[15] Bonyani, M., Jahangard, S. & Daneshmand, M. Persian handwritten digit, character and word recognition using deep learning.IJDAR 24, 133– 143 (2021). https://doi.org/10.1007/s10032-021- 00368-2