A Holistic Approach to Urdu Language Word Recognition using Deep Neural Networks

Hashim Raza Khan
Department of Electronic Engineering
NED University of Engineering and Technology
Karachi, Pakistan
hashim@neduet.edu.pk

Muhammad Abul Hasan
Department of Bio-Medical Engineering
NED University of Engineering and Technology
Karachi, Pakistan
abulhasan@neduet.edu.pk

Majida Kazmi
Dpt. of Computer and Information Systems Engineering
NED University of Engineering and Technology
Karachi, Pakistan
majidakazmi@neduet.edu.pk

Nabeel Fayyaz
Department of Electrical Engineering
NED University of Engineering and Technology
Karachi, Pakistan
nabeelfayyaz@neduet.edu.pk

Hamza Khalid
Dpt .of Computer and Information Systems Engineering
NED University of Engineering and Technology
Karachi, Pakistan
hamza@neduet.edu.pk

Saad Ahmed Qazi
Dpt. of Computer and Information Systems Engineering
NED University of Engineering and Technology
Karachi, Pakistan
saadqazi@neduet.edu.pk

Abstract-Urdu is one of the most popular languages in the world. It is a Persianized standard register of the Hindi language with considerable and valuable literature. While digital libraries are constantly replacing conventional libraries, a vast amount of Urdu literature is still handwritten. Digitizing this handwritten literature is essential to preserve it and make it more accessible. Nevertheless, the scarcity of Urdu Optical Character Recognition (OCR) research limits a digital library's scope to a manual document search. The limited research work in this area is mainly due to the complexity of Urdu Script. Unlike the English language, the Urdu writing style is cursive, bidirectional, and character shapes and sizes highly vary depending on their position. Holistic word recognition is found to be a better solution among many other text segmentation techniques as it takes the complete word into account instead of segmenting it explicitly or implicitly. For this project, the data of five different Urdu words were collected for training and testing a convolutional neural network and 96% recognition accuracy was achieved.

Keywords-word recognition; Urdu; deep learning; CNN; cursive writting

I. INTRODUCTION

The Urdu language is an Indo-European language derived from the Farsi alphabet, which stems from the Arabic alphabet [1]. Having more than a hundred (100) million native speakers in the South Asian region, Urdu is one of the two official languages of Pakistan. Similar to Arabic, the Urdu language is bidirectional. However, Urdu has more isolated letters (38) than Persian (32) and Arabic (28). A great deal of history, literature, and lore of the South Asian region is present in the form of handwritten manuscripts in Urdu. To make them accessible to larger audiences through electronic media, the only possible solution is the digitization of the literature in searchable form. An obvious solution is manually sorting/annotation, which is a tedious and time-consuming task [2]. Usually, if we want to bring any hard copy document in e-format, scanning is performed, bringing the data in image format. Searching keywords from that image is not directly possible in Urdu, Persian, and Arabic. This problem can be addressed with the help of Optical Character Recognition (OCR) systems. OCR is defined as a process of classification of optical patterns in a digital image or as the conversion of images of typed, handwritten, or printed text into machine-encoded text [3]. Unlike English OCRs that first hit the market in the early 90s [4, 5], the earliest research conducted in Urdu OCRs was reported in 2003 [6]. The complications associated with Urdu script are:

- Urdu is a bidirectional language. It is written from right to left, and in some cases, such as numbers, it follows the left to the right direction.
- Each Urdu letter can take four distinct shapes due to four different places, i.e. initial, middle, final, and standalone, as shown in Figure 1 [7].
- Urdu is always written cursively. Inter- and intra- word spaces are present in the Urdu script.

Corresponding author: Hashim Raza Khan

- Words sometimes overlap each other and make the recognition process difficult.
- Some Urdu characters have dots associated with them which can vary in numbers and locations.
- Sometimes in handwritten documents, open loops may get closed [8].

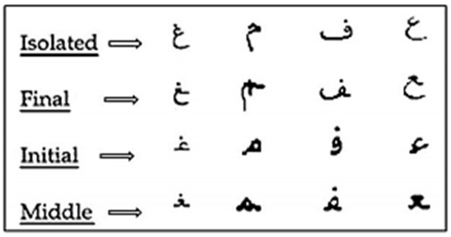

Fig. 1. Character variation depending on the position.

Research has been done on similar problem types, as mentioned above, in numerous languages, and text recognition has been experimented on text images using OCR [9]. OCR technique is used for converting images into editable text. However, for cursive scripts like Urdu, results are not satisfactory as isolated scripts. Hence, cursive scripts' pattern recognition is challenging [10]. Moreover, since Arabic and Persian languages' characters are subsets of Urdu language, it is practically impossible to use Arabic/Persian OCRs for Urdu language. On the other hand, the general problem with OCRs is that the text that occurs in natural scene images, historical documents, or images hugely varies in appearance and layout or images with background objects cause false-positive detections [11]. Due to these complications, converting an image with cursive text into editable text is not flawless, and accuracy is not satisfactory. Word (keyword) spotting is another technique that can be defined as a process of finding all instances of a query word that exist in a document image without wholly recognizing the document. In the past years, research in handwriting recognition with OCRs has developed to a level that generates commercial applications. However, the success rate is much higher in online than offline recognition [9]. Hence, in cases where the OCR system cannot assure human reader satisfaction, word spotting is a viable solution for information retrieval [2].

Over the last few decades, there has been rapid development in text spotting methods. The majority of methods split the complete process into two major stages [12]: Segmentation/text detection and word recognition. Segmentation/text detection generates candidate character or word bounding boxes/region proposals [3]. It can be defined as decomposing an image into many sub-images, each containing a character/word. Text detection tackles the standard text spotting pipeline and contributes to the whole process's error rate [12]. Several segmentation strategies can be broadly classified into three main categories: implicit segmentation, explicit segmentation, and holistic approach. Implicit segmentation (recognition-based segmentation) is a technique in which the system searches the images for components that match classes in its alphabet. However, this technique is adopted to integrate the processes of segmentation and recognition. Accordingly, Hidden Makrov Models (HMMs) based approaches have emerged, and authors in [13] proposed a method based on HMMs using an implicit segmentation approach and claimed word recognition accuracy of 88.2% on a lexicon of size 3,771. In explicit segmentation (classical approach), an input image is first partitioned into sub-images containing characters and then it is classified. Explicit segmentation needs to find all the interconnection between words, also known as ligatures, and dissect the word images through all detected ligatures. Authors in [14] proposed a direct segmentation approach for offline handwriting recognition. The accuracy of the segmentation approach was reported at 83.6%. However, over-segmentation and wrong segmentation are considerably high [15].

A holistic method tries to identify words instead of characters and considers them as units. It is therefore termed as Intelligible Word Recognition (IWR) instead of a character recognition technique. There are other significant reasons for its popularity as well: the holistic approach is parallel to human reading and can be applied to historical documents whose poor quality makes it challenging to analyze with the classical approach [4]. Due to the cursive handwriting style, changing the shape of characters depending on the Urdu script's position, the classification using a holistic approach is more efficient and optimal [1]. Besides these advantages, its major drawback is that the holistic approach is always restricted to a predefined lexicon since it does not deal directly with letters, but with words, so a specific lexicon of the word is a necessary constraint. Whenever we want to modify the lexicon, the training stage is mandatory. This property makes this approach suitable where the lexicon is statistically defined [15]. After applying machine learning techniques in the image recognition domain, the lexicon size can be made quite large, making this technique suitable for many applications. A lexicon of size of 90,000 words has been claimed in recent works of word recognition [7]. A typical optical character recognition pipeline starts with looking at the picture, locating text areas, identifying the lines of the text, and at the end, it tries to recognize each character or word present in that image. The text recognition step receives a single word or character's cropped image and identifies the depicted object. Character-based classifiers are used to classify individual characters, which are then integrated to generate an entire word. Different techniques are used for this classification task. Authors in [16, 17] use Convolutional Neural Networks (CNN) as character classifiers. Authors in [17] used a complicated combination of CNNs and HMMs for a fixed lexicon to generate the final recognition results. An alternate approach for this text detection task is whole word-based recognition or holistic word recognition, extracting features from the entire word before performing word classification [7].

Since we cannot use Arabic and Persian language recognition systems, our primary focus is to develop a word recognition system using a deep neural network for the Urdu language. To that end, we have used a holistic or whole word-based recognition approach. We have collected around 5298 samples of five Urdu words: tawhid, namaz, hajj, rehmat, and jannat for our project.

II. Data Preparation

A. Training and Testing Data

Data were collected from different people, with a diversity of writing styles, containing 5298 samples of 5 different words (approximately every word is written around 1000 times). Seventy five percent of the samples were selected for training, and 25% were set for testing purposes.

B. Data Pre-processing

Recognition results rely on the preprocessing phase. Unwanted noise can degrade the recognition accuracy. Our work in the preprocessing stages involves binarization and converting of the cropped images into a uniform size while creating the dataset. For data collection, we have used A4-sized papers with 26 rows and 10 columns. These papers were scanned at 300 dpi, and then individual characters were cropped using an open-source OCR helper tool. These cropped images were saved in binarized format in a text file, as shown in Figure 2. This text file has three channels (RGB), which carries redundant information, so information from only one channel was extracted, and then these arrays were concatenated to form a complete dataset. The OCR helper tool facilitates the dataset preparation process by providing an adjustable binarization threshold, height and width sensitivity, and the export size of the blob.

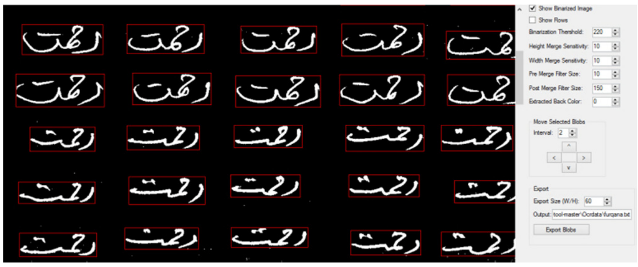
☑ Show Binarized Image
□ Show Rows
Binarization Thershold: 220
Height Merge Sensitivity: 10
Width Merge Sensitivity: 10
Pre Merge Filter Size: 10
Post Merge Filter Size: 150
Extracted Back Color: 0
Move Selected Blobs
Interval: 2
Export
Export Size (W/H): 60
Output: tool-master\Ocrdata\furqana.txt

Fig. 2. OCR helper tool for cropping and binarization.

III. The Deep Learning Model

A. Model Architecture

Neural networks are typically organized in layers, consisting of many interconnected nodes known as neurons. Generally, there are 3 types of layers: input, hidden, and output layers. The input layer receives the input pattern and communicates it to the next (hidden) layer. A hidden layer is an intermediate layer between the input and the output layers. The processing is performed here via weighted connections. The last output layer of the neural network usually consists of n nodes, where n is the total number of classes. A Convolutional Neural Network (CNN) is one of the most common types of artificial neural networks and it is considered a state-of-the-art technique in pattern recognition, outperforming several classifiers [18]. The CNN's architecture contains convolutional layers, pooling layers, and a fully connected layer. A CNN finds its application in a wide range of areas, e.g. natural language processing, video analysis, speech recognition, and pattern recognition [19].

The convolutional layer is the first layer in a CNN. This layer uses a number of n different filters or kernels (edge or curve detectors). Each of these can be intuitively modeled as a feature identifier. These filters then convolve with the original image's sub-images according to their dimension and extract features from that image. To understand the process of convolution, consider a 6×6 image and a 3×3 filter. To extract features, we must convolve them. This operation is shown in Figure 3.

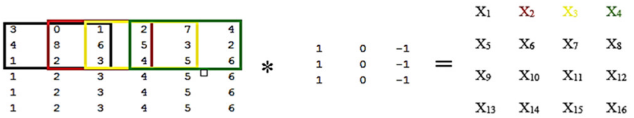

Fig. 3. A simple convolution operation.

The values of the output can be easily calculated as follows:

X_1=3*1+4*1+1*1+0*0+0*8+0*2+1*-1+6*-1+3*-1=-2

Output dimensions can be found as: n−f+1*n−f+1

where n is the input image dimension (6 in this case) and f is filter's dimension (3 in this case). The output dimensions will be 4×4.

Pooling or subsampling layer is used to decrease the features' resolution, making the feature more robust against noise and distortion and making the system computationally effective. A fully connected layer is often used as the last layer in a CNN. The working principle of a fully connected layer is the same as traditional multi-layer perceptons. Every neuron present in this layer receives input from every neuron of the previous layer and finally classifies the output.

B. Model Training and Testing

To develop an Urdu handwritten word recognition system, we created a deep feed-forward neural network architecture using the TensorFlow library. We started our experiment with 5 different word classes, each having approximately 1000 samples. We trained our network on a non-GPU machine with varying image sizes, i.e. (60×60, 50×50, and 30×30 pixels) and different numbers of neurons and got meaningful results comparable with [20].

C. Results

We trained our model for 5 different words and tested it for accuracy and loss by changing the number of neurons and input image sizes. Increasing the number of neurons resulted in better accuracy but increased training time. The highest accuracy achieved was 95.81%, with 60×60 image size and 2000 neurons in each layer. The words used to train the model were tawhid, namaz, hajj, rehmat, and jannat which were labeled as 0,1,2,3, and 4 respectively. Figure 4 shows the random inferences from the test set, where 'Tr' shows the original word and 'Pred' shows the predicted output.

1) 30×30 Pixel Images

The accuracies of individual words and of the complete development set on 30×30 images with different number of neurons is shown in Table I. Figure 5 shows the loss decrease and the accuracy increase with the number of epochs. Figure 6 shows the individual accuracy of each word with a CNN containing 500 neurons.

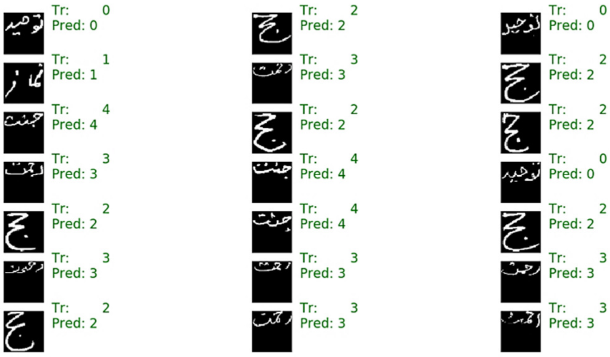

Fig. 4. Prediction on the test set with randomly selected words.

TABLE I. ACCURACY ON 30×30 IMAGES

Neurons	Accuracy (%)					
	Test set	Tawhid	Namaz	Hajj	Rehmat	Jannat
450	88.36	91.63	79.15	94.59	99.97	76.53
500	90.22	88.44	84.55	93.82	99.90	84.23
600	89.68	88.84	89.57	91.50	99.68	78.46
300	86.03	80.87	87.64	94.59	99.61	67.30
350	89.21	86.05	86.48	94.20	98.46	80.76

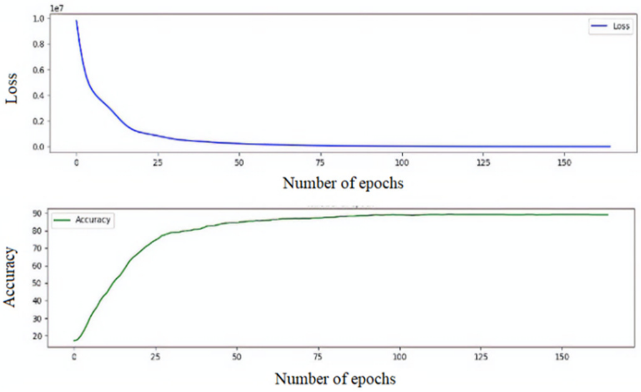

Fig. 5. Accuracy and loss with 500 neurons.

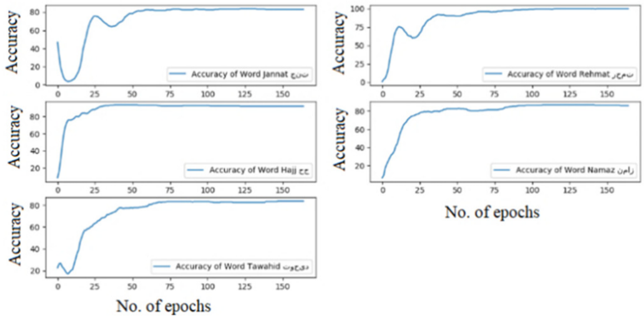

Fig. 6. Individual accuracies with 500 neurons.

2) 50×50 Pixel Images

The accuracies of individual words and of the complete development set on images of size 50×50 with different number of neurons is shown in Table II. Figure 7 shows the loss decrease and the accuracy increase with the number of epochs. Figure 8 shows the individual accuracy of each word with a CNN containing 1250 neurons.

TABLE II. ACCURACY ON 50×50 IMAGES

Neurons	Accuracy (%)					
	Test set	Tawhid	Namaz	Hajj	Rehmat	Jannat
1250	94.33	94.82	93.05	97.29	97.66	86.53
1400	94.25	97.61	94.59	98.45	97.50	80.76
1000	94.24	94.02	93.05	92.66	97.46	81.53
800	91.93	92.82	92.27	96.91	97.21	77.69

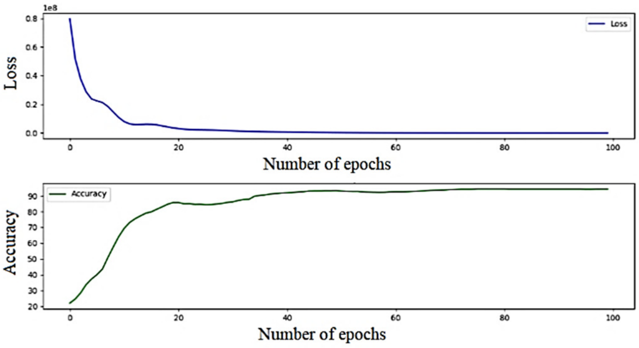

Fig. 7. Accuracy and loss of a CNN with 1250 neurons.

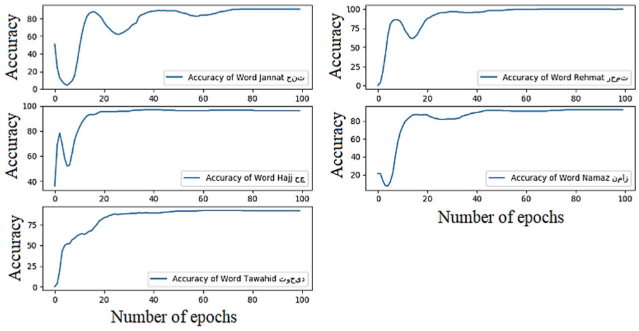

Fig. 8. Individual accuracies with 1250 neurons.

3) 60×60 Pixel Images

The accuracies of individual words and of the complete development set on images of size 60×60 with different number of neurons is shown in Table III. Figure 9 shows the loss decrease and the accuracy increase with the number of epochs. Figure 10 shows the individual accuracy of each word with a CNN containing 2000 neurons.

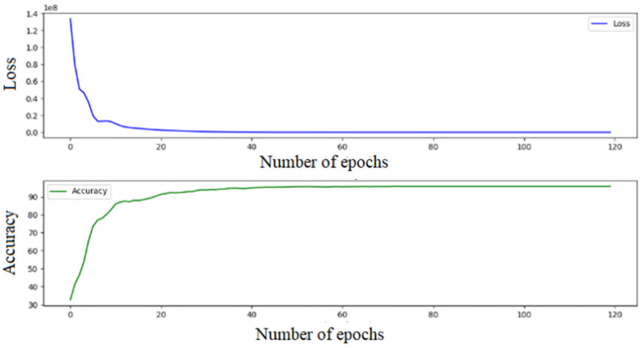

Fig. 9. Accuracy and loss with 2000 neurons.

TABLE III. ACCURACY ON 60×60 IMAGES

Neurons	Accuracy (%)					
	Test set	Tawhid	Namaz	Hajj	Rehmat	Jannat
400	87.82	80.87	87.25	94.98	98.60	75.76
800	92.63	93.22	88.03	97.68	98.94	84.23
1600	94.33	90.04	95.75	96.13	99.32	89.61
2000	95.81	95.21	96.52	98.45	99.61	88.84
2400	94.49	95.61	93.82	93.82	99.59	89.23

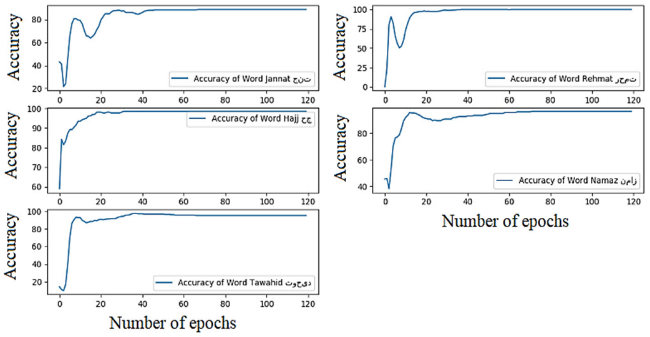

Fig. 10. Individual accuracies with 2000 neurons.

IV. CONCLUSION

This report describes a system for handwritten Urdu holistic word recognition using a deep neural network. This work's motivation was the existence of extensive valuable literature in the Urdu language in many South Asian libraries and the ongoing effort to digitalize these scripts. Research work has already been conducted in this area on different languages, but the digitization of Urdu script is a scarcely developed area. Some work has been done for Urdu word/character recognition using SVMs and HMMs.

In this work, a holistic Urdu handwritten word recognition has been developed using a deep neural network. We have collected more than 5000 handwritten samples to include a diversity of styles in our study. Approximately every word has 1000 samples. The images were saved in binarized format and the dataset was divided to training (75%) and testing (25%) subsets. A feed-forward deep neural network was trained for recognition/classification, and showed promising results. We have achieved accuracy up to 96% on the complete test set.

V. FUTURE RECOMMENDATIONS

Many different experiments, tests, and architectures can be tried in the future, including:

- Training neural networks for a larger lexicon of Urdu words.
- Using different architectures and configurations of neural networks for improved accuracy.
- Segmentation of Urdu words in handwritten script.
- Developing a complete end-to-end system of keyword search in handwritten Urdu script.

REFERENCES

[1] M. W. Sagheer, C. L. He, N. Nobile, and C. Y. Suen, "Holistic Urdu Handwritten Word Recognition Using Support Vector Machine," in *2010 20th International Conference on Pattern Recognition*, Aug. 2010, pp. 1900–1903, https://doi.org/10.1109/ICPR.2010.468.

[2] A. Abidi, A. Jamil, I. Siddiqi, and K. Khurshid, "Word Spotting Based Retrieval of Urdu Handwritten Documents," in *2012 International Conference on Frontiers in Handwriting Recognition*, Bari, Italy, Sep. 2012, pp. 331–336, https://doi.org/10.1109/ICFHR.2012.289.

[3] H. Bunke and P. S. P. Wang, *Handbook of Character Recognition and Document Image Analysis*. World Scientific, 1997, https://doi.org/10.1142/2757.

[4] A. Chaudhuri, K. Mandaviya, P. Badelia, and S. K. Ghosh, *Optical Character Recognition Systems for Different Languages with Soft Computing*. Springer International Publishing, 2017.

[5] N. H. Khan and A. Adnan, "Urdu Optical Character Recognition Systems: Present Contributions and Future Directions," *IEEE Access*, vol. 6, pp. 46019–46046, 2018, https://doi.org/10.1109/ACCESS.2018.2865532.

[6] U. Pal and A. Sarkar, "Recognition of Printed Urdu Script," in *Proceedings of the Seventh International Conference on Document Analysis and Recognition - Volume 2*, USA, Aug. 2003, pp. 1183-1187.

[7] Q. U. A. Akram and S. Hussain, "Improving Urdu Recognition Using Character-Based Artistic Features of Nastalique Calligraphy," *IEEE Access*, vol. 7, pp. 8495–8507, 2019, https://doi.org/10.1109/ACCESS.2018.2887103.

[8] Z. Ahmad, J. K. Orakzai, and I. Shamsher, "Urdu compound Character Recognition using feed forward neural networks," in *2009 2nd IEEE International Conference on Computer Science and Information Technology*, Beijing, China, Aug. 2009, pp. 457–462, https://doi.org/10.1109/ICCSIT.2009.5234683.

[9] V. Lavrenko, T. M. Rath, and R. Manmatha, "Holistic word recognition for handwritten historical documents," in *First International Workshop on Document Image Analysis for Libraries, 2004. Proceedings.*, Palo Alto, CA, USA, Jan. 2004, pp. 278–287, https://doi.org/10.1109/DIAL.2004.1263256.

[10] N. P. T. Kishna and S. Francis, "Intelligent tool for Malayalam cursive handwritten character recognition using artificial neural network and Hidden Markov Model," in *2017 International Conference on Inventive Computing and Informatics (ICICI)*, Coimbatore, India, Nov. 2017, pp. 595–598, https://doi.org/10.1109/ICICI.2017.8365201.

[11] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading Text in the Wild with Convolutional Neural Networks," *International Journal of Computer Vision*, vol. 116, no. 1, pp. 1–20, Jan. 2016, https://doi.org/10.1007/s11263-015-0823-z.

[12] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, Washington, DC, USA, Jun. 2004, vol. 2, https://doi.org/10.1109/CVPR.2004.1315187.

[13] P. R. Cavalin, A. de Souza Britto, F. Bortolozzi, R. Sabourin, and L. E. S. Oliveira, "An implicit segmentation-based method for recognition of handwritten strings of characters," in *Proceedings of the 2006 ACM symposium on Applied computing*, New York, NY, USA, Apr. 2006, pp. 836–840, https://doi.org/10.1145/1141277.1141468.

[14] M. Ghosh, R. Ghosh, and B. Verma, "A fully automated offline handwriting recognition system incorporating rule based neural network validated segmentation and hybrid neural network classifier," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 7, pp. 1267–1283, Nov. 2004, https://doi.org/10.1142/S0218001404003654.

[15] N. Sivashanmugam, "A Study of Various Segmentation Techniques for Cursive Handwritten Words Recognition," *International Journal of Modern Trends in Engineering and Research*, 2015.

[16] O. Alsharif and J. Pineau, "End-to-End Text Recognition with Hybrid HMM Maxout Models," *arXiv:1310.1811 [cs]*, Oct. 2013, Accessed: Apr. 23, 2021. [Online]. Available: http://arxiv.org/abs/1310.1811.

[17] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up Convolutional Neural Networks with Low Rank Expansions," *arXiv:1405.3866 [cs]*, May 2014, Accessed: Apr. 23, 2021. [Online]. Available: http://arxiv.org/abs/1405.3866.

[18] S. Hijazi, R. Kumar, and C. Rowen, "Using Convolutional Neural Networks for Image Recognition." Cadence Design Systems Inc, 2015.

[19] S. Sahel, M. Alsahafi, M. Alghamdi, and T. Alsubait, "Logo Detection Using Deep Learning with Pretrained CNN Models," *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6724–6729, Feb. 2021, https://doi.org/10.48084/etasr.3919.

[20] U. Khan, K. Khan, F. Hassan, A. Siddiqui, and M. Afaq, "Towards Achieving Machine Comprehension Using Deep Learning on Non-GPU Machines," *Engineering, Technology & Applied Science Research*, vol. 9, no. 4, pp. 4423–4427, Aug. 2019, https://doi.org/10.48084/etasr.2734.