Computer Science • 25(4) 2024 https://doi.org/10.7494/csci.2024.25.4.6164

M Shanmuga Priya
Pavithra A
Leema Nelson

CHARACTER/WORD MODELLING: A TWO-STEP FRAMEWORK FOR TEXT RECOGNITION IN NATURAL SCENE IMAGES

Abstract *Text recognition from images is a complex task in computer vision. Traditional text recognition methods typically rely on Optical Character Recognition (OCR); however, their limitations in image processing can lead to unreliable results. However, recent advancements in deep-learning models have provided an effective alternative for recognizing and classifying text in images. This study proposes a deep-learning-based text recognition system for natural scene images that incorporates character/word modeling, a two-step procedure involving the recognition of characters and words. In the first step, Convolutional Neural Networks (CNN) are used to differentiate individual characters from image frames. In the second step, the Viterbi search algorithm employs lexicon-based word recognition to determine the optimal sequence of recognized characters, thereby enabling accurate word identification in natural scene images. The system is tested using the ICDAR 2003 and ICDAR 2013 datasets from the Kaggle repository, and achieved accuracies of 78.5% and 80.5%, respectively.*

Keywords scene text recognition, convolution neural network, character recognition, character/word modelling

Citation Computer Science 25(4) 2024: 637–652

Copyright © 2024 Author(s). This is an open access publication, which can be used, distributed and reproduced in any medium according to the Creative Commons CC-BY 4.0 License.

1. Introduction

Text recognition in natural scene images is a critical task in the fields of computer vision and machine learning, which aims to build computer software that automatically extracts text from natural scene images. This technology has widespread applications in areas such as automated identification of traffic signals, license plates, and autonomous robot navigation [10, 22]. While numerous studies have focused on text recognition, most have focused on documents or digital paper-based materials, neglecting the complexities of extracting text from natural scene images. The intricate nature of this task arises from the diverse layouts and styles of characters, encompassing factors such as font, shape, size, color, and position while contending with challenges such as noise, blur, occlusions, and non-uniform lighting [25]. Earlier studies have used, OCR engines, such as ABBYY and TESSERACT, for text recognition, but their efficacy in scene image processing has been limited.

The main aim of this study is to recognize text present in printed images [12,18]. Many researchers have acknowledged the effectiveness of deep learning architectures for text recognition tasks [2, 6, 14]. These architectures contain multiple layers for various purposes including input representation, feature extraction, and classification. Among these, convolutional neural networks (CNN) have attracted significant attention for computer vision applications [10, 16, 19, 25]. The CNN architecture has several layers, including input, middle, and output layers. The input layer processes the inputs, whereas the middle layers with convolution and pooling features extract relevant information. Finally, an output layer with one or more fully connected layers performs the final classification [8].

This study conducted a comprehensive comparative analysis of text recognition methods that employ various deep learning architectures. To evaluate the effectiveness of the proposed system, two widely used public datasets are employed: the ICDAR 2003 and ICDAR 2013 datasets. The ICDAR 2013 dataset consisted of 229 training images and 233 testing images, each annotated at the word level. This dataset includes a diverse range of character and word graphics captured in natural settings and is suitable for various applications such as banners, displays, navigation panels, clothing, and house numbers. The selection of the sample images from the ICDAR 2013 dataset is shown in Figure 1.

The datasets selected for this task are characterized by a diverse range of images that showcase various sizes, scales, orientations, font types, and styles. These images are rich in characters and are carefully chosen to provide a comprehensive representation of subject matter. The proposed framework can be effectively evaluated in diverse authentic scenarios by incorporating various sample types, thereby enhancing its adaptability and credibility in recognizing text from natural scene images.

The main aim of this study is to achieve three objectives. The initial objective is to identify individual characters present in natural scene images using a CNN. The second objective involves recognizing the sequential order of text present in natural scene images, which is determined using a Viterbi search to determine the optimal

character sequence. The final objective is to conduct extensive experiments on two complex scene text recognitions using benchmark datasets to demonstrate the performance of the proposed system.

Figure 1. Natural scene images taken from the ICDAR 2013 dataset

The problem addressed by the proposed system is scene text recognition, which involves recognition of text in natural scene images captured by cameras or other devices. This is a challenging task because of various factors such as varying lighting conditions, complex backgrounds, and different fonts and languages [8, 19, 24]. Traditional methods for text recognition rely on handcrafted features and complex modelling, which are time consuming and may not be able to handle variability in real-world scenes. Therefore, there is a need for an automated system that can accurately recognize text in scene images, even in challenging scenarios.

The remainder of this paper is organized as follows. Section 2 presents prior research efforts and contextualizes the advanced scene-text recognition techniques. Section 3 details the design and implementation of the proposed system, delving into its architecture, algorithms, and key functionalities, while also demonstrating the process of creating a blueprint for scene-text recognition using character/word modeling. Section 4 evaluates and discusses the experimental results and findings of the proposed system by assessing both the character and word recognition modules. Finally, Section 5 concludes the implementation of character/word modeling as a means of text recognition in natural images.

2. Related works

This section provides an overview of the most advanced methods for scene-text recognition. The two common processes in text recognition are character-based and word-based processes. Character-based recognition depends on the detection and recognition of each character to identify an entire word when the characters are combined. Character segmentation and recognition are the bases of traditional character-based

recognition systems [16]. Several studies have employed a sliding window strategy that incorporates various scales for character segmentation and recognition. The recognition of scene text is challenging due to the intricacy of the task and the impact that segmentation, an essential component, has on the overall recognition system. The absence of segmentation in other character-based techniques for scene-text recognition highlights the difficulties of this process.

Recent methods, such as connectionist temporal classification, create character predictions followed by a post-processing stage [10]. Post-processing techniques that incorporate linguistic knowledge can be employed to enhance the accuracy of scene-text recognition. For example, Thillou et al. (2005) utilized n-gram scores to restrict an inference algorithm's prediction of a correct word [19]. Shi et al. (2016) proposed a deep architecture that integrates a convolutional neural network (CNN) with a recurrent neural network (RNN) to identify scene texts in images. Convolutional layers, which collect characteristics from the input picture, and recurrent layers, which predict a label, make up this design distribution for each frame, and a transcription layer based on connectionist temporal classification converts the frame predictions into a label sequence [17].

The main objective of a word-based recognition system is to obtain features from a complete word picture, without implementing character segmentation. The second objective is to integrate or pool these features into a predetermined architecture to conduct word classification and subsequently recognition [5]. Chen et al. (2020) developed an adaptive embedding gate for attention-based scene-text recognition to detect text based on neurocomputing. Fisher vectors are combined with pyramidal histograms of characters [1]. To develop a word-based recognition system, these vectors are combined with densely extracted low-level descriptors and spatial pyramids [9, 24]. Word recognition was achieved using the maximum posterior estimate obtained from a finite-state-weighted transducer. To address scene-text recognition, a 90k-class-based CNN was designed, where each class corresponds to a word in the lexicon [5].

In general, there are two basic strategies for using image-specific lexicons. A dictionary or lexicon of terms is included to enhance the effectiveness of the word-based recognition system. This list of candidate words allows the system to fix some of its error [4]. To ensure accuracy, it is recommended to use a powerful algorithm that searches for the dictionary term closest to the anticipated word. This approach is particularly important for word-based recognition, because it simultaneously captures both low-level features and high-level linguistic priors. A character recognition module can also be implemented in conjunction with a word recognition module to improve the results.

3. System design

This system design demonstrates the process of creating a blueprint for the development of scene-text recognition using character/word modelling. In this study, a deep

learning-based system is developed to recognize text embedded in natural-scene images. Figure 2 shows the proposed framework, which consists of three modules: image preprocessing, character recognition, and word recognition. In the first module, the input image is preprocessed to enhance the text recognition. In the second module, characters with different variations are recognized from the images using a popular CNN-based architecture. Finally, in the third module, Viterbi search algorithms are used to determine the best character sequence, which guides the system in determining the exact word in the image.

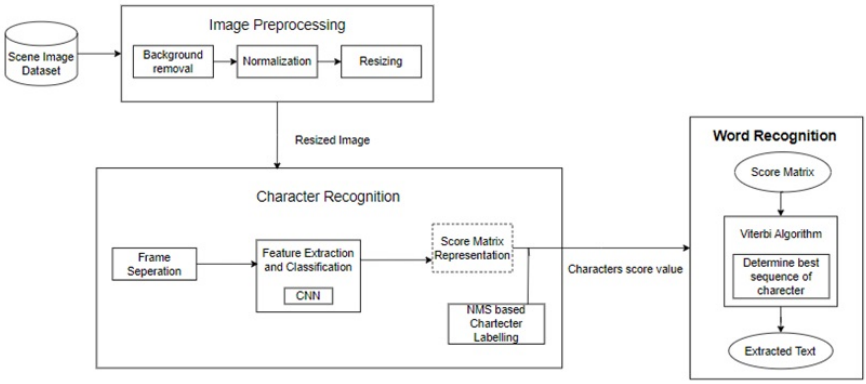

Figure 2. Overall system architecture

3.1. Image preprocessing module

The system accepts an image with text as input. The input image is preprocessed to enhance the quality and facilitate the extraction of text regions. Preprocessing refers to activities that involve pictures at the most fundamental level of abstraction. The input and output of the system are intensity images that are essentially identical to the original sensor data. Intensity images are typically depicted as matrices of image function values or brightness levels. The aim of preprocessing is to improve the picture data by removing unwanted distortions or enhancing the useful visual characteristics for further processing. Figure 3 shows the steps involved in image preprocessing, where the inputs are natural-scene images.

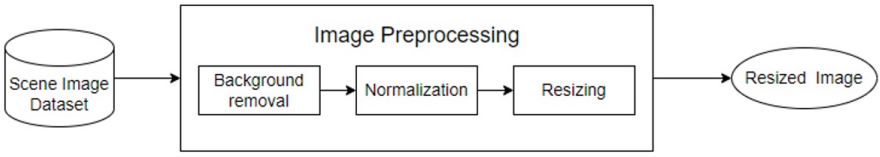

Figure 3. Image pre-processing

The steps involved are background removal, normalization, and image resizing. The image background removal step eliminates or alters the background from natural

scene images. Image normalization involves scaling the pixel values of an image to a fixed range or the mean and standard deviation, which can help mitigate the effects of lighting variations and other distortions. In the image-resizing step, each image is resized to a fixed height and width using a downsampling operation. This is because images captured from real-world scenes have character images of various scales, sizes, locations, and orientations. Therefore, to recognize characters in these images, it is mandatory to resize the image to a fixed height and width.

3.2. Character recognition module

In this module, the resized image is used as an input for character recognition. The image is divided into individual frames, which are then processed to identify the character region. The frames are separated from the resized image, and each frame is processed to localize the character region. To extract the important frame features, they must be processed using a cascaded CNN architecture. The CNN architecture consists of four stages of convolutional and subsampling layers used for feature extraction, whereas a fully connected layer is used for classification. The CNN architecture combines convolutional and subsampling layers to extract characteristics and predict class labels from input frames, motivated by the visual cortex structure. The filters detect specific characteristics in the input frame, thereby producing activation maps that are passed to the next layer in the CNN architecture [11]. Ultimately, the CNN outputs two crucial elements: the class label corresponding to the input frame and a score matrix representing the probabilities of the characters in each frame (p(char–I)). With characters falling into 62 possibilities (including 26 lowercase letters, 26 uppercase letters, and 10 numbers), it is important to note that a frame may encompass either the entire character or only part of it. Additionally, a character may span one or more frames at times. To address this challenge, a non-maximum-suppression method is applied. This method determines the frame containing the complete character based on the score value, thereby allowing the removal of redundant frames from classification inputs. The character-recognition process is illustrated in Figure 4.

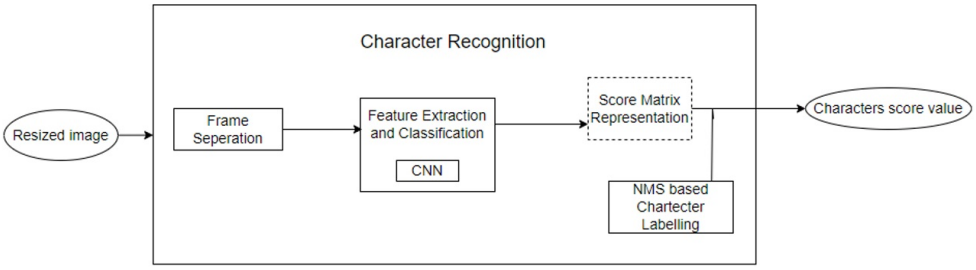

Figure 4. Character recognition

The score matrix depicted in Figure 5 represents the probabilities of the letters in the input image, which contains the word "NEW." The Viterbi search algorithm, a topic covered in the following module, selects letters with highlighted probabilities as the optimal sequence of characters to determine a word.

	N	n	...	e	E	...	W	w
0								
1								
...								
A								
...								
E				0.156	1.029			
...								
M								
N	0.456	0.021						
O								
...								
W							1.01	0.985

Figure 5. Score matrix for the word NEW

3.3. Word recognition module

In this module, a lexicon-based word recognition method is proposed to identify word W from an input image based on character sequence S generated from the character recognition module. The objective is to determine the sequence of characters with the highest probability. The character recognition module provides a set of N probabilities corresponding to the character class labels for each examined frame (N corresponds to the total number of character classes). The Viterbi search algorithm is used to determine the probability of an optimal sequence of characters. This algorithm is implemented to convert predicted probabilities into words. When confronted with the challenge of a sequence of M overlapping frames, it is essential to determine the most likely path for the optimal character sequence. The Viterbi search algorithm is effective in handling such scenarios using a score matrix to identify the character sequence with the highest score, ultimately declaring it as a recognized word [3, 23].

Figure 6 depicts the comprehensive process of the word recognition system, showing the recognized text within a boundary box. This box may overlap with the original image used for testing. To ensure clarity, the recognized word is transformed into text format and displayed separately.

Figure 7 provides a sample output of the text recognition process, offering a tangible representation of the system's capability to identify and display recognized words accurately.

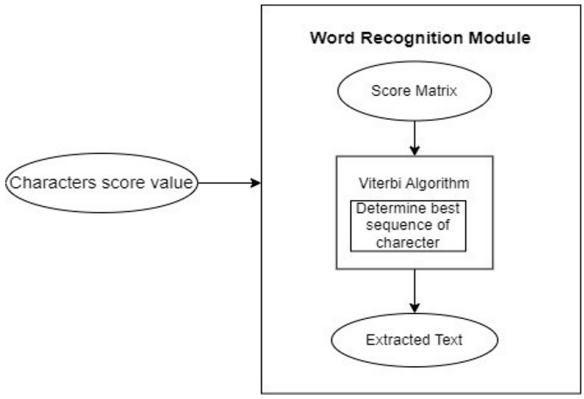

Figure 6. Word recognition

Figure 7. Output for word recognition

4. Results and discussion

This section discusses the experimental results and findings of the proposed system, by evaluating both the character and word recognition modules using the ICDAR 2003 and ICDAR 2013 datasets. The experiments are conducted on a computer with an Intel Core i7-7500U processor (2.9 GHz) and 8 GB RAM implemented using a Jupyter Source Notebook in Python. The performance assessment unfolds in two stages: character recognition, where various deep learning architectures, such as CNN, ANN, and RNN, are explored and compared, and word recognition, where the accuracy of the system is evaluated based on precision. The accuracy metric measures the ratio of correctly recognized characters or words to the total number of characters or words based on the ground truth.

Figure 8 illustrates the correct recognitions for each alphabetical character (A–Z, a–z) and number (0–9).

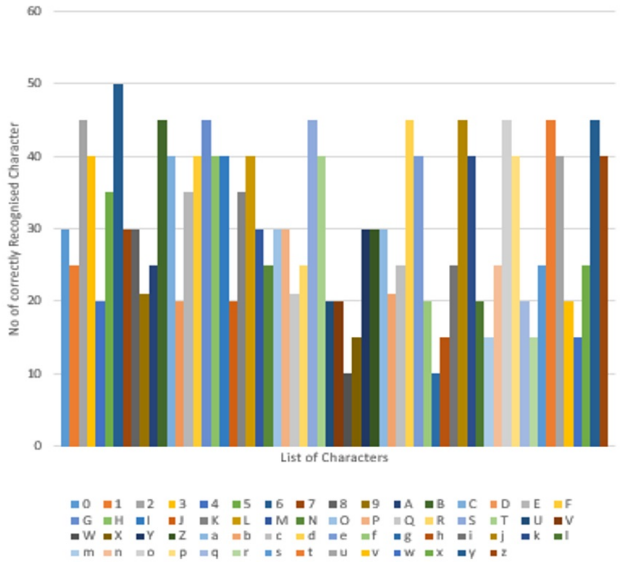

Figure 8. No. of times correctly recognised characters

The experimental results in Table 3 and Figure 9 highlight the superior performance of the proposed CNN-based text recognition method, which achieves the highest accuracy of 80.5% on the ICDAR 2003 dataset and 78.5% on the ICDAR 2013 dataset. Furthermore, the system performance is assessed by varying the width and height of the resized images, as listed in Tables 1 and 2.

Table 1
Word recognition accuracy in ICDAR 2003 different heights and widths

Height	Width	Accuracy [%]
80	80	79.8
90	90	80.7
100	100	81.5
150	150	77.2
200	200	76.3
300	200	71.5

Table 2
Word recognition accuracy in ICDAR 2013 with different heights and widths

Height	Width	Accuracy [%]
80	80	78
90	90	78.8
100	100	79.8
150	150	75.6
200	200	74.5
300	300	72.3

Table 3
Accuracy for test data using CNN

Datasets	Accuracies [%]
ICDAR 2003	80.5
ICDAR 2013	78.5

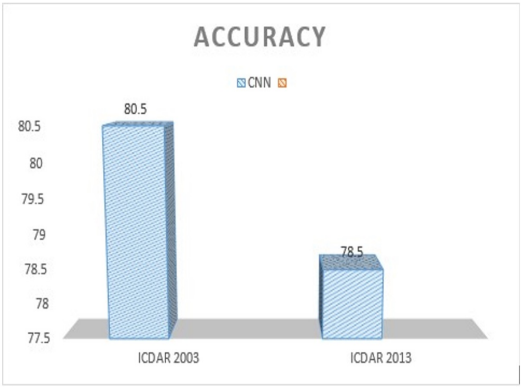

Figure 9. Accuracy for test data using CNN

Notably, the width and height of $(100, 100)$ demonstrates optimal accuracy, achieving 81.5% for the ICDAR 2003 dataset and 79.8% for the ICDAR 2013 dataset. Figure 10 and 11 show the word recognition accuracy for different heights and widths, revealing that the resized image of $(100, 100)$ outperforms other sizes.

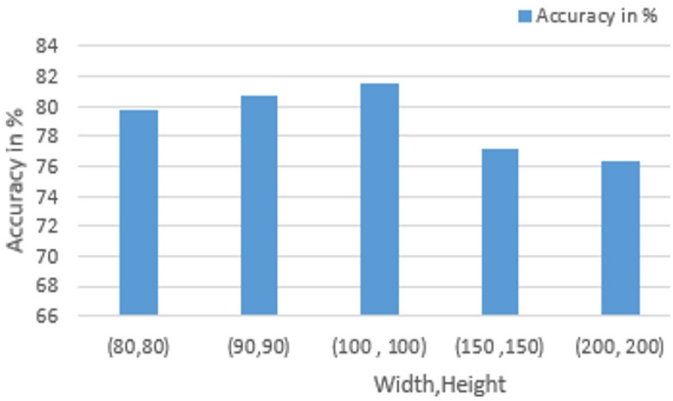

Figure 10. Word recognition accuracy in ICDAR 2003 with different heights and widths

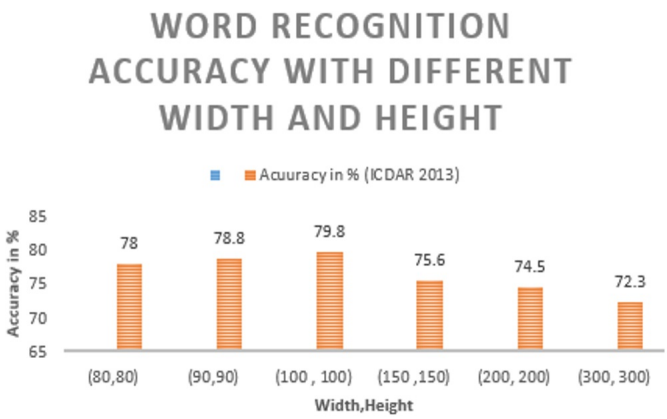

Figure 11. Word recognition accuracy in ICDAR 2013 with different heights and widths

The robustness of the system is tested against various image qualities, such as clear, blurred, and low-quality images, as shown in Table 4 and Figure 12.

Table 4
Precision value for different categories of Image data

Image category	ICDAR 2003 [%]	ICDAR 2013 [%]
Low quality images	20	15
Blurred images	65	70
Cleared images	97	98

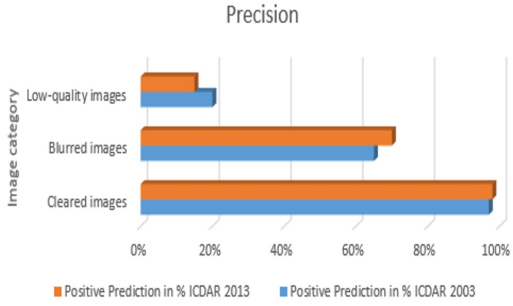

Figure 12. Precision value for different categories of image data

The system exhibited superior performance for clearer images. Comparisons between three different deep learning architectures CNN (Proposed Method), RNN, and ANN revealed that CNN outperformed the others. This is evident in Table 5 and Figure 13, where CNN achieved the highest accuracy in character recognition. Similarly, in word recognition, as shown in Figure 14 and Table 5, CNN outperformed the

other architectures. In summary, the proposed system, primarily utilizing a CNN, demonstrated remarkable accuracy in recognizing characters and words, showcasing resilience against varying image qualities, and confirming its efficacy in real-world scenarios.

Table 5
Word recognition accuracy

Datasets	CNN [%]	RNN [%]	ANN [%]
ICDAR 2003	80.5	78.9	70
ICDAR 2013	78.5	70.8	58.75

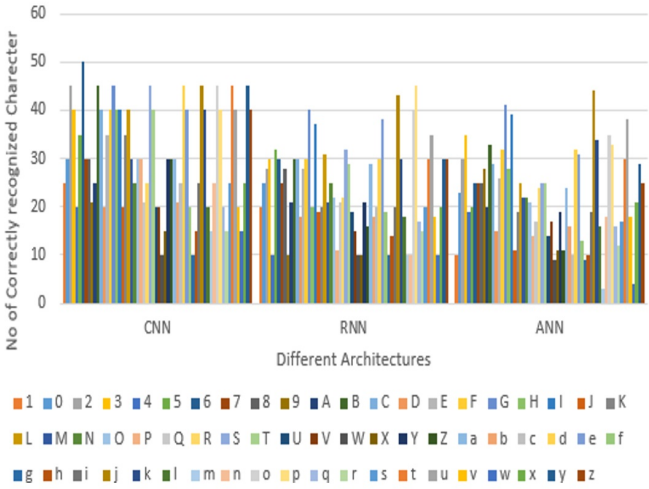

Figure 13. No. of correct character recognition using three different deep learning architectures

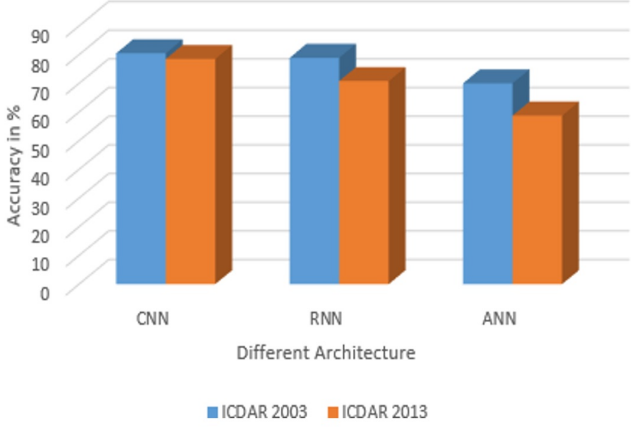

Figure 14. Word recognition accuracy

The result of the statistical test, as indicated by the obtained p-value of 0.235, demonstrated the level of significance associated with the contrast in accuracy between the two datasets under consideration. If the p-value is less than the predetermined significance level (e.g., 0.05), it can be reasonably inferred that there is a statistically significant difference in the performances of the two datasets. Statistical tests comparing the accuracies achieved using the ICDAR 2003 and ICDAR 2013 datasets are presented in Table 6.

Table 6
Statistical tests accuracy comparison using ICDAR 2003 and ICDAR 2013 datasets

S. No.	Dataset	Accuracy [%]	p-value
1	ICDAR 2003	80.5	0.235
2	ICDAR 2013	78.5	

The proposed system for scene text recognition demonstrated impressive accuracy and computational efficiency with notable speed, which makes it a strong candidate for practical applications. To provide a basis for comparison, the runtime of the system is assessed against those of state-of-the-art systems. The entire pipeline is implemented on an Intel Core i7 2.9 GHZ machine, yielding a runtime of approximately 0.30 seconds per image with GPU and 0.70 seconds per image with CPU. These results are competitive with other existing systems, such as the one reported in [4], which achieved an average runtime of 0.92 seconds per sample using a Core i5 2.80 GHZ processor. Another system, developed by Novikova et al. [15], had a speed of 0.85 seconds per image. In a separate study, [12] reported a runtime of approximately 0.25 seconds per image with GPU and 0.83 seconds per image with CPU only on a computer with an Intel Xeon E5 2.6 GHZ x 2 processor. The proposed word recognition system is compared with existing state-of-the-art techniques, as shown in Table 7.

Table 7
Comparison between proposed word recognition system with existing state of art systems

S. No.	Methods	ICDAR 2003 dataset accuracy [%]	ICDAR 2013 dataset accuracy [%]
1	[21]	62.00	70.00
2	[7]	66.19	–
3	[15]	–	72.40
4	[20]	80.56	–
5	[13]	78.20	–
6	[4]	78.44	82.31
7	[9]	81.70	79.40
8	Proposed system	80.50	78.50

5. Conclusion

This study implemented character/word modelling to recognize texts in natural images. This two-step process demonstrates significant advancements in scene-text recognition. In the first step, the CNN model is used for character recognition, and the Viterbi search algorithm is used for word recognition in the second step. The developed model is tested using ICDAR 2003 and ICDAR 2013 datasets. The results demonstrated the effectiveness of the cascaded CNN architecture for recognizing characters and words in various scenarios. Moreover, the developed model adapts to different image qualities, such as clear, blurred, and low-quality images, thereby highlighting its robustness in real-world applications. The results obtained from the comparative analysis show that existing deep learning techniques, such as RNN, ANN, and CNN, achieve the highest accuracy for character recognition. The developed model not only surpasses traditional optical character recognition (OCR) techniques but also shows promising results in challenging scenarios, making it suitable for applications such as document digitization, number plate recognition, image-based search engines, and augmented reality.

By exploring diverse convolutional neural network (CNN) architectures, including transformer-based models and innovative convolutional layers, it may be possible to enhance the performance in character and word recognition through improved feature extraction and context comprehension. This can be achieved by incorporating attention mechanisms and advanced techniques.

References

[1] Almazán J., Gordo A., Fornés A., Valveny E.: Word Spotting and Recognition with Embedded Attributes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36(12), pp. 2552–2566, 2014. doi: 10.1109/tpami.2014.2339814.

[2] Arafat S.Y., Iqbal M.J.: Urdu-Text Detection and Recognition in Natural Scene Images Using Deep Learning, *IEEE Access*, vol. 8, pp. 96787–96803, 2020. doi: 10.1109/access.2020.2994214.

[3] Bahi H.E., Zatni A.: Text recognition in document images obtained by a smartphone based on deep convolutional and recurrent neural network, *Multimedia Tools and Applications*, vol. 78(18), pp. 26453–26481, 2019. doi: 10.1007/s11042-019-07855-z.

[4] Bhunia A.K., Kumar G., Roy P.P., Balasubramanian R., Pal U.: Text recognition in scene image and video frame using color channel selection, *Multimedia Tools and Applications*, vol. 77, pp. 8551–8578, 2018. doi: 10.1007/s11042-017-4750-6.

[5] Chen X., Wang T., Zhu Y., Jin L., Luo C.: Adaptive embedding gate for attention-based scene text recognition, *Neurocomputing*, vol. 381, pp. 261–271, 2020. doi: 10.1016/j.neucom.2019.11.049.

[6] Coates A., Carpenter B., Case C., Satheesh S., Suresh B., Wang T., Wu D.J., Ng A.Y.: Text detection and character recognition in scene images with unsupervised feature learning. In: *2011 International Conference on Document Analysis and Recognition*, pp. 440–445, IEEE, 2011. doi: 10.1109/icdar.2011.95.

[7] Elagouni K., Garcia C., Mamalet F., Sebillot P.: Combining multi-scale character recognition and linguistic knowledge for natural scene text OCR. In: *DAS '12: Proceedings of the 2012 10th IAPR International Workshop on Document Analysis Systems*, pp. 120–124, IEEE, 2012. doi: 10.1109/das.2012.26.

[8] Goel V., Mishra A., Alahari K., Jawahar C.V.: Whole is greater than sum of parts: Recognizing scene text words. In: *2013 12th International Conference on Document Analysis and Recognition*, pp. 398–402, IEEE, 2013. doi: 10.1109/icdar.2013.87.

[9] Harizi R., Walha R., Drira F., Zaied M.: Convolutional neural network with joint stepwise character/word modelling based system for scene text recognition, *Multimedia Tools and Applications*, pp. 3091–3106, 2022. doi: 10.1007/s11042-021-10663-z.

[10] Jaderberg M., Simonyan K., Vedaldi A., Zisserman A.: Reading text in the wild with convolutional neural networks, *International Journal of Computer Vision*, vol. 116, pp. 1–20, 2016. doi: 10.1007/s11263-015-0823-z.

[11] Liao M., Zhang J., Wan Z., Xie F., Liang J., Lyu P., Yao C., Bai X.: Scene text recognition from two-dimensional perspective. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 8714–8721, 2019. doi: 10.1609/aaai.v33i01.33018714.

[12] Liu X., Kawanishi T., Wu X., Kashino K.: Scene text recognition with CNN classifier and WFST-based word labeling. In: *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 3999–4004, IEEE, 2016. doi: 10.1109/icpr.2016.7900259.

[13] Liu X., Kawanishi T., Wu X., Kashino K.: Scene text recognition with high performance CNN classifier and efficient word inference. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1322–1326, IEEE, 2016. doi: 10.1109/icassp.2016.7471891.

[14] Long S., He X., Yao C.: Scene Text Detection and Recognition: The Deep Learning Era, 2018, arXiv preprint arXiv:181104256. arXiv:1811.04256.

[15] Novikova T., Barinova O., Kohli P., Lempitsky V.: Large-lexicon attribute-consistent text recognition in natural images, *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision Florence, Italy, October 7–13, 2012 Proceedings, Part VI*, pp. 752–765, 2012. doi: 10.1007/978-3-642-33783-3_54.

[16] Portaz M., Kohl M., Chevallet J.P., Quénot G., Mulhem P.: Object instance identification with fully convolutional networks, *Multimedia Tools and Applications*, vol. 78(3), pp. 2747–2764, 2019. doi: 10.1007/s11042-018-5798-7.

[17] Shi B., Bai X., Yao C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39(11), pp. 2298–2304, 2016. doi: 10.1109/tpami.2016.2646371.

[18] Shivakumara P., Bhowmick S., Su B., Tan C.L., Pal U.: A new gradient-based character segmentation method for video text recognition. In: *2011 International conference on document analysis and recognition*, pp. 126–130, IEEE, 2011. doi: 10.1109/icdar.2011.34.

[19] Thillou C., Ferreira S., Gosselin B.: An embedded application for degraded text recognition, *EURASIP Journal on Advances in Signal Processing*, 370317, 2005. doi: 10.1155/asp.2005.2127.

[20] Wang D.H., Wang H., Zhang D., Li J., Zhang D.: Robust scene text recognition using sparse coding based features, 2015. ArXiv preprint arXiv:1512.08669., arXiv:1512.08669.

[21] Wang K., Babenko B., Belongie S.: End-to-end scene text recognition. In: *2011 International Conference on Computer Vision*, pp. 1457–1464, IEEE, 2011-11. doi: 10.1109/iccv.2011.6126402.

[22] Xu C., Yang J., Gao J.: Coupled-learning convolutional neural networks for object recognition, *Multimedia Tools and Applications*, vol. 78, pp. 573–589, 2019. doi: 10.1007/s11042-017-5262-0.

[23] Xue C., Huang J., Zhang W., Lu S., Wang C., Bai S.: Image-to-character-to-word transformers for accurate scene text recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45(11), pp. 12908–12921, 2023. doi: 10.1109/tpami.2022.3230962.

[24] Yuan J., Wei B., Liu Y., Zhang Y., Wang L.: A method for text line detection in natural images, *Multimedia Tools and Applications*, vol. 74, pp. 859–884, 2015. doi: 10.1007/s11042-013-1702-7.

[25] Zhang Z., Zhang C., Shen W., Yao C., Liu W., Bai X.: Multi-oriented Text Detection with Fully Convolutional Networks. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4159–4167, 2016. doi: 10.1109/cvpr.2016.451.

Affiliations

M Shanmuga Priya
Anna University, Chennai, Tamil Nadu, India, mscsepriya@annauniv.edu

Pavithra A
Anna University, Chennai, Tamil Nadu, India, spriya@cs.annauniv.edu

Leema Nelson
Chitkara University, Chitkara University Institute of Engineering & Technology, India, Punjab, leema.nelson@gmail.com

Received: 12.03.2024
Revised: 11.05.2024
Accepted: 5.06.2024