Comparative Study of Optical Character Recognition Using Different Techniques on Scanned Handwritten Images

Anshika Singh and Sharvan Kumar Garg

Abstract Optical Character Recognition (OCR) is always a prominent area of research and has gained attention in academia and industries, and more advancements are being performed to improve accuracy and quality of text and images. OCR is a technique that converts the digital or optical image (which can be image of typed or printed document or scanned handwritten document or scene photo) into machine-encoded format. OCR takes an image containing text as input, scans the text character by character, analyzes it, and generates a text corresponding to the text in the image, thus reducing the storage size and making accessibility of data easy and faster. In this paper, we will be studying OCR using different techniques like Convolutional Neural Network, Recurrent Neural Network, pytesseract, Keras-OCR, EasyOCR and comparing their results on handwritten scanned images.

Keywords Optical character recognition · Convolutional neural network · Recurrent neural network · Pytesseract · Keras-OCR · EasyOCR

1 Introduction

Optical Character Recognition is a technique that automatically recognizes the characters existing in the digital images. Character Recognition is accomplished through feature extraction and classification steps. OCR is the most assuring technique in the area of pattern recognition and Artificial Intelligence (AI), used for conversion of handwritten words, letters, or characters into digital form that we can edit or search efficiently and store more optimally.

A. Singh (✉)
Department of Computer Science, Shaheed Rajguru College of Applied Sciences for Women, University of Delhi, Delhi, India
e-mail: anshika.research@gmail.com

S. K. Garg
Department of Computer Science and Engineering, Subharti Institute of Technology and Engineering, Swami Vivekan and Subharti University, Meerut, India

D. K. Sharma et al. (eds.), *Micro-Electronics and Telecommunication Engineering*,
Lecture Notes in Networks and Systems 617,
https://doi.org/10.1007/978-981-19-9512-5_38

OCR is utilized in various industries in the dockets, such as in the supply chain, in the medical field to label medicines and other equipment, and to enter data for checks such as passports, invoices, receipts and bank documents, license plates, business cards. It is very efficient in banking, legal, health care, and supply chains, among many others [1]. Motor vehicle number plates are automatically recognized through optical recognition features; the system uses cameras connected to the internet to analyze the plates [2]. The merits of the application of OCR include searchability, editability, accessibility, storability, translatability, and backups. The system has proven to work efficiently for businesses modernizing to the digital era.

Each person had a different handwriting style like shape, size, tilting, etc. So, we can write a character in ways. So, we will be applying different techniques or algorithms that will automatically recognize the characters or text in an input image. OCR comprises the following steps: taking an input text in the form of some image, optical scanning of the image, preprocessing (skew detection and correction, noise removal, binarization, etc.), segmentation (character or word segmentation), feature extraction (structural, global transform features, etc.), classification (CNN, RNN, SVM, etc.), text recognition, and generating output [3]. They proposed an OCR system with a faster training neural network which was developed, and each recognition stage was assigned with a training period of a short time. The character passed through the network is selected, and a similarity check is conducted by contrasting the pixels of objects. Final result is selected from the output having the highest score; the rate of recognition technique in this model is higher than that in the conventional network approach. Study indicates that to date, there is not even a single algorithm language which achieves 100% accuracy; filtering the noise at the preprocessing part improves algorithm quality [4]. OCR is applied in Captcha programs which reduces hacking risks. The internet performs various economic tasks like booking, registrations, education, and payments, among many others. The system is used in online libraries and institutional repositories to collect, preserve, or air intellectual data [1]. OCR helps in reducing paperwork for translation in multiple languages consequently decreasing the amount of storage and easy accessibility [5]. OCR is applied to process insurance papers and health forms to claim the insurance and maintain patient records in the medical sector [6].

2 Different Techniques to Achieve Image Text Recognition

2.1 Convolutional Neural Networks (CNNs)

CNN is an algorithm of deep learning which is inspired by visual system and has its applications in the area of Natural Language processing (NLP). It accepts an image as input and allocates importance in the form of trainable parameters (weight and bias) to different objects/features in the image and distinguishes them from each other. CNN is analogous to human brain connectivity patterns of neurons. Each neuron responds to a stimuli receptive field (a limited region of visual field). Entire visual area is covered by overlapping these fields.

An input image is represented as a matrix of pixel values having three channels in case of RGB scale, whereas a grayscale image has only one channel. CNN has different layers: Convolutional, ReLU, Pooling, Fully connected. In CNN, we compare the image piece by piece. These pieces are nothing but the features or filters. In the convolutional layer, we apply those filters on all possible positions on the image. Then, the features are aligned with the input image followed by multiplication of each pixel in the image and its corresponding pixel of feature, add them up. Then, divide the resultant by the total number of pixels in the feature, create the map, and set the filter values at that point. After this, move the filter to all possible positions and compare the feature values to find a match. Same procedure is followed for each filter and fed to ReLU layer. Rectified Linear Unit (ReLU) is a function that will activate node only when the input value is greater than a specific value of threshold, whereas if value of the input is less than or equal to 0, it will output 0. However, if the value of the input goes above a certain value of threshold, it becomes linear with dependent variable. All three layers having any negative values get eliminated in the ReLU layer. It is then fed to the Pooling layer which reduces the size of the image by applying the following steps: choose a window size and a step, move the window over the entire filtered image, and take a maximum value from each window. It is applied on all filters. We stack up these layers again to achieve more reduced matrix of pixels and finally fed to the Fully connected layer. Here, we take filters and reduced image and place them into a single list. We will find the elements in vector with high value. We sum up the values of those elements and compare them with those in the dataset. The value which is closer is the predicted character in the input image. This is how CNN works on input image.

CNN applies relevant filters to an image in order to capture spatial and temporal dependencies. It can be trained to better understand the complexity of an image. Hence, it is a good choice for applications of image classification, face recognition, speech recognition, scene labeling, text classification.

2.2 Recurrent Neural Networks (RNNs)

Unlike a feedforward neural network where inputs and outputs were independent, previous state output is fed into the current state in the Recurrent Neural network. RNN is suitable for sequential models or time-series data models and finds applications in the field of Natural Language processing, image captioning, machine translation, language translation, speech recognition, etc. For understanding sequential data, we take an example of Gmail; here, we write an email where we get edit suggestions like we type Thank and it will suggest you Thanks and Regards.

As its name says RNN is recurrent, the same task is performed for each element of a sequence with output dependent on prior computations. RNN applies the concept of "memory" (short-term memory) which holds the information of prior inputs for finding the next output in sequence. RNNs can make use of information in long sequences, but ideally they can look back only a few steps [7].

RNN had a number of input nodes, hidden nodes (with feedback loop so that information can pass back multiple times to the same node, also called recurrent units), and output nodes. Recurrent units process information for a set of known number of timesteps (each processing of input through hidden nodes), applying the activation function to each hidden state and input for that timestep. Hidden nodes have three parameters in RNN: input's weight, hidden unit's weight, and a bias. RNN uses modified backpropagation, i.e., backpropagation in time (BPTT) which unfolds in time to train the weights. BPTT computes the gradient vector [8]. There are more variants of RNN: Gated Recurrent Units (GRUs), Long Short-Term Memory (LTSM), Bidirectional Recurrent Neural Networks (BRNNs) [9].

When we talk about CNN and RNN, we need a dataset to train the network, but there are some other tools like pytesseract, Keras-OCR, EasyOCR which have predefined libraries.

2.3 Pytesseract

Pytesseract is also known as Python-tesseract. It is a tool in Python for Optical Character Recognition (OCR). It recognizes text present inside the image. It is a wrapper class for Tesseract-OCR Engine provided by Google [10]. Pytesseract is used to detect binary images and extracting characters. In number plate recognition, it recognizes the text in color images and in gray images with accuracy 61% and 70%, respectively [11].

2.4 Keras-OCR

Keras-OCR provides end-to-end sequential training steps for building a new OCR models [12]. It is used to digitize modern libraries to code articles into various categories, analyze texts syntactically, text and speech annotation. It is used to process handwritten images and classify them to specific categories [13].

2.5 EasyOCR

EasyOCR consists of OCR library which can read short text. It can also read multiple languages simultaneously in case they are compatible together. EasyOCR makes use of font files and template matching algorithms together to recognize broken or connected characters or badly printed characters [14].

EasyOCR is applied in imaging invoices, which helps track financial records, preventing piling up lists of payments. The system scans checks and writings on it without the need to involve a human. The system simplifies data collection in agencies and organizations, making data handling and processing simple [15].

3 Proposed Work

Our motive is to apply all the above techniques to handwritten scanned images to recognize the text embedded and study their behavior. The text recognition from the image process includes various steps as shown in Fig. 1.

3.1 Preprocessing

The purpose of this process lies in obtaining better accuracy in text recognition to obtain a better text recognition rate. Input can be in the form of RCB or gray image with possibilities of non-uniform background or some watermarks. So, it starts with the image enhancement process, i.e., noise elimination, minimizing blurring, contrast improvement followed by skew detection and correction, then perform thresholding to eliminate any watermarks or noise to extract the information from

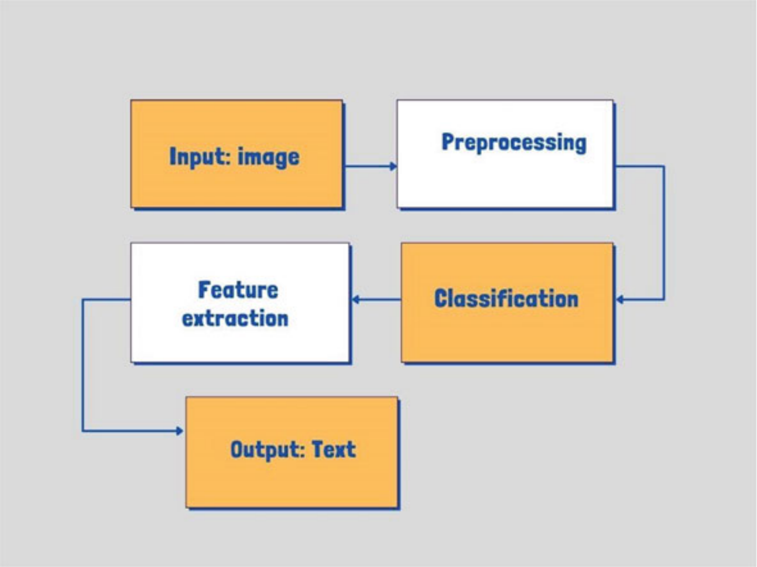


Fig. 1 Different steps in OCR

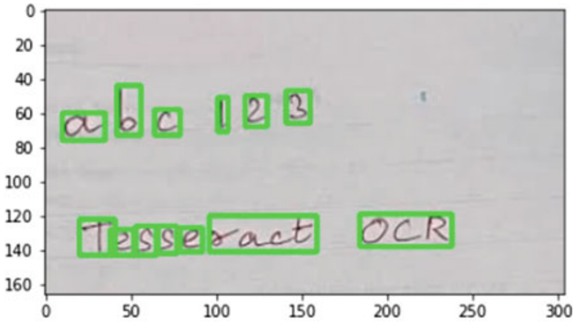
its background. Next step is segmentation for graphics isolation from text and separate individual characters, words, or sentences. After this, morphological operations to add pixels in case of eroded parts in characters present in preprocessed images. Processing of images is necessary for photography in feature classification. However, research should be done to improve digital imaging [16]. The use of these techniques normalizes images to a standard size, and this enhances image quality and recognition.

3.2 Feature Extraction

Extracting features is a vital step which eliminates redundancies from data. Accuracy of the classification can be improved by considering the most relevant features [17]. Features selected should be efficient enough to classify even among characters or symbols with close similarity. They must have different values for different classes and similar values for identical classes.

3.3 Classification and Text Recognition

Here, we implement CNN, RNN, pytesseract, Keras-OCR, EasyOCR on scanned image of handwritten text to recognize the embedded text in image and extract recognized text (Figs. 2, 3, 4, 5 and 6).



```
Q7QHSCSEW123W
```

Fig. 2 OCR using CNN

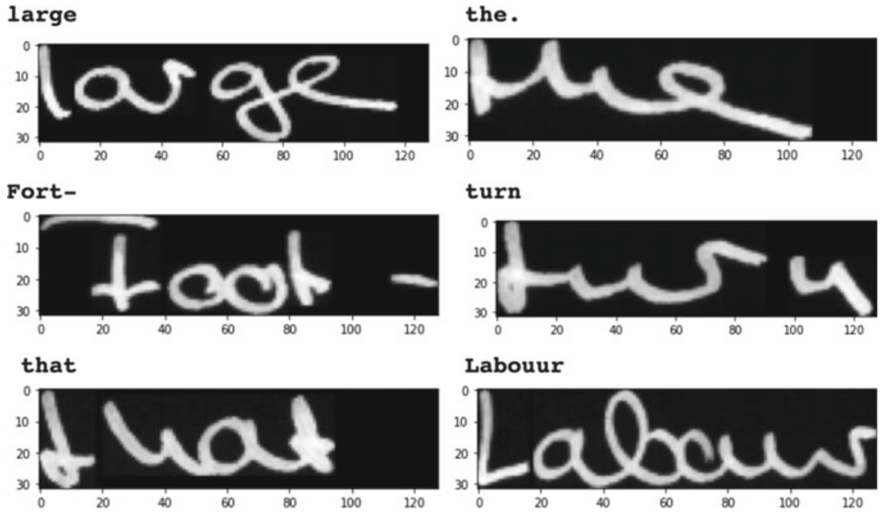


Fig. 3 OCR using RNN

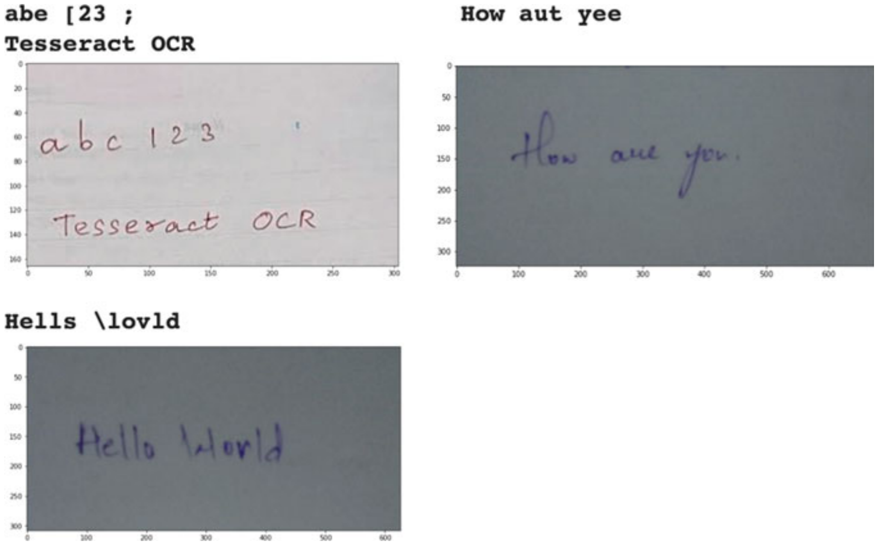
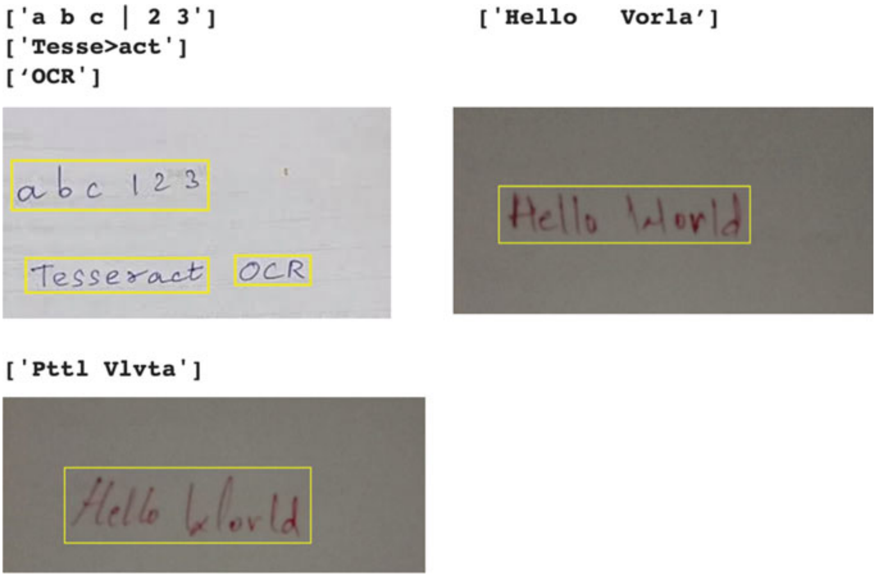


Fig. 4 OCR using pytesseract

4 Conclusion

We studied OCR using various techniques in this paper. OCR is divided into three major steps: preprocessing, feature extraction, and classification and text recognition. Here, in the classification step, we apply different techniques to classify the characters



```
['a b c | 2 3']
['Tesse>act']
[‘OCR']
```

```
['Hello  Vorla’]
```

```
['Pttl Vlvta']
```

Fig. 5 OCR using EasyOCR

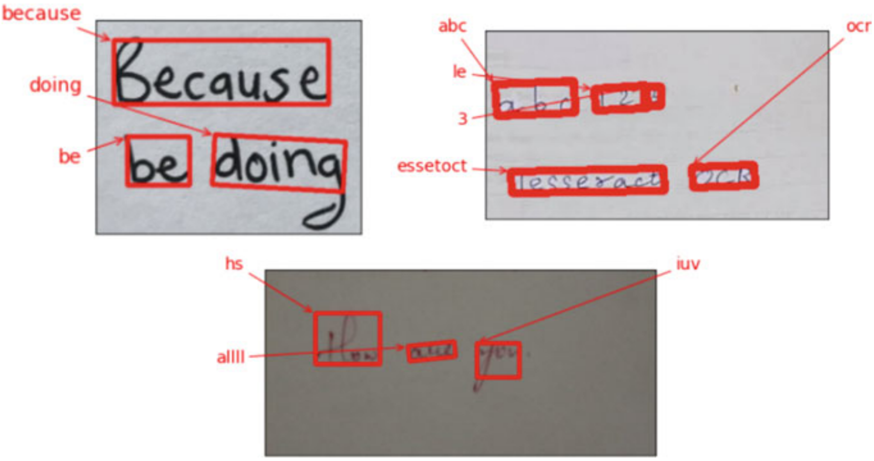


Fig. 6 OCR using Keras-OCR

or text embedded in an image. The merits of the application of OCR include searchability, editability, accessibility, storability, translatability, etc. OCR has several applications: number plate recognition to prevent vehicle theft, in Captcha programs to reduce hacking risks, data entry process of various business documents, for passport recognition and extraction of traveler information in airports, data entry for patient treatment summary in hospitals, etc. CNN and RNN are more time-consuming techniques as compared to pytesseract, Keras-OCR, and EasyOCR. As the results show that CNN and EasyOCR give the least accuracy among others, Keras-OCR works better than pytesseract, EasyOCR, and CNN. RNN is the best suitable technique in case of text recognition as it works well in the deformed style of writing the characters. However, accuracy of CNN can be enhanced by using larger datasets. In case of usage of predefined libraries and datasets, pytesseract performs better than EasyOCR for handwritten images and images of high resolution. While EasyOCR perform well in case of images with organized text, RNN and Keras-OCR works better in case of unorganized and multi-font or style of text embedded in images.

References

1. Rao, N. V., Sastry, A. S. C. S., Chakravarthy, A. S. N., & Kalyanchakravarthi, P. (2016). Optical character recognition technique algorithms. *Journal of Theoretical and Applied Information Technology, 83*(2), 275–282.
2. Dalarmelina, N. D. V., Teixeira, M. A., & Meneguette, R. I. (2019). A real-time automatic plate recognition system based on optical character recognition and wireless sensor networks for ITS. *Sensors, 20*(1), 55.
3. Sabu, A. M., & Das, A. S. (2018). A survey on various optical character recognition techniques. *Conference on Emerging Devices and Smart Systems (ICEDSS), 2018*, 152–155.
4. Dasari, M. S. K., Mehta, S., & DD, M. D. S. (2020). Optical character recognition of devanagari script using machine learning-a survey. *Journal of Xi'an University of Architecture & Technology*, 593–599. ISSN (1006-7930).
5. Sumathi, C.P., Santhanam, T., & Devi, G. (2012). A survey on various approaches of text extraction in images. *International Journal of Computer Science and Engineering Survey, 3*(4).
6. Surana, S., Pathak, K., Gagnani, M., Shrivastava, V., M. T. R., & Madhuri, G. S. (2022). Text extraction and detection from images using machine learning techniques: a research review. In *2022 International Conference on Electronics and Renewable Systems (ICEARS)* (pp. 1201–1207).
7. WILDML—Stanford Canvas. https://canvas.stanford.edu/files/1090843/download?download_frd=1
8. Towards Data Science. https://towardsdatascience.com/rnn-recurrent-neural-networks-how-to-successfully-model-sequential-data-in-python-5a0b9e494f92
9. Machine Learning Mastery. https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/
10. Pytesseract 0.3.9. https://pypi.org/project/pytesseract/
11. Patel, C., Patel, A., & Patel, D. (2012). Optical character recognition by open source OCR tool tesseract: A case study. *International Journal of Computer Applications, 55*(10), 50–56.
12. Keras-ocr. https://keras-ocr.readthedocs.io/en/latest/
13. Gulli, A., & Pal, S. (2017). Deep learning with Keras. Packt Publishing Ltd.
14. Open eVision. https://documentation.euresys.com/Products/Open_eVision/Open_eVision_2_5/en-us/Content/03_Using_Open_eVision/D1_EasyOCR_-_Reading_Texts/EasyOCR_-_Reading_Texts.htm
15. Sai Rakesh Kamisetty, V. N., Sohan Chidvilas, B., Revathy, S., Jeyanthi, P., Anu, V. M., & Mary Gladence, L. (2022). Digitization of data from invoice using OCR. In *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)* (pp. 1–10).

16. Alkoffash, M. S., Bawaneh, M. J., Muaidi, H., Alqrainy, S., & Alzghool, M. (2014). A survey of digital image processing techniques in character recognition. *International Journal of Computer Science and Network Security (IJCSNS), 14*(3), 65.
17. Arif, M., Hassan, H., Nasien, D., & Haron, H. (2015). A Review on Feature Extraction and Feature Selection for Handwritten Character Recognition. International Journal of Advanced Computer Science and Applications. 6(2).