SN Computer Science (2024) 5:347
https://doi.org/10.1007/s42979-024-02713-8

ORIGINAL RESEARCH

A Deep Neural Network Based Holistic Approach for Optical Character Recognition of Handwritten Documents

Jayati Mukherjee[1]

Received: 4 June 2023 / Accepted: 13 February 2024

Abstract
Optical Character Recognition of handwritten document has been a research topic for last few decades now. Different type of classification schemes starting from template matching, structural analysis to deep neural network have been proposed by researchers. In this paper, a novel holistic approach is proposed for recognition of handwritten words. The approach is a hybrid model combining CNN and BLSTM layers which are responsible for extraction of spatial and temporal features respectively from the word images. Both the features are combined by compact bi-liner pooling. The CBP layer highlights the fine grained details which in turn help to achieve high recognition accuracy. The extracted feature is recognized by a connectionist temporal classification layer. The weights are learnt based on the database using the back propagation algorithm. The hybrid model is trained using three public databases CMATERdb2.1.2, IIIT-HW-Dev and IIIT-HW-Telugu containing Bengali, Devanagari and Telegu handwritten words respectively. The proposed model has achieved 96.42%, 94.79%, 95.07% accuracy on the Bengali, Devanagari and Telegu databases respectively.

Keywords Computer vision · Image analysis · Pattern recognition · Optical character recognition · Document analysis · Handwriting recognition

Introduction

Optical character recognition (OCR) is a process of reading document images and transforming it into machine readable format, which is ASCII code or Unicode. The first idea of OCR was found with the invention of 'retina scanner'. In early age, the idea of the OCR could be found in mechanical devices, which were able to recognize the limited number of characters with minimal speed and huge time. Later computer researchers started working on OCR systems. OCR is a systematic and sophisticated way of document handling. Not only this, OCR of documents helps in electronically editing the documents, key word spotting and keyword searching in the documents. Even for preservation of documents the OCR methodologies are applied on degraded documents. A degraded document can be defined as a document that either does not follow the typographical convention or get deformations due to some external reasons. A handwritten document also contains some sort of degradation like skew, tilted lines, un-even line gaps etc. In other words, the topographical convention of a document structure is not preserved in a handwritten document. Moreover, handwritten documents contain different type of writing due to variation of writers. So, the recognition of handwritten documents is a complex OCR problem to solve. These degradation creates challenges in the way of achieving accuracy in optical character recognition for handwritten documents. For English language OCR is a solved problem now. In European scripts such as Latin, Greek the OCR systems developed have achieved great success. In Indian languages like Hindi, Bengali, Tamil etc the works on OCR systems are going on for few decades now. Though rigorous work suggested by researchers, OCR of handwritten documents is still an unsolved challenging problem. The challenges increases in working with Indian scripts due to their complex nature and huge character sets.

An OCR system is mainly composed of three modules: Pre-processing, segmentation and classification. In some

This article is part of the topical collection "SWOT to AI-embraced Communication Systems (SWOT-AI)" guest edited by Somnath Mukhopadhyay, Debashis De, Sunita Sarkar and Celia Shahnaz.

✉ Jayati Mukherjee
jayati.uit93@gmail.com

[1] Department of Computer Science and Engineering,
Academy of Technology, Adisaptagram,
West Bengal 712121, India

Published online: 27 March 2024

OCR systems all these three phases are followed and the unit of recognition is character, those systems are called analytic system. In some of the systems the unit of recognition is word or line. Those system are called holistic system. In holistic system the character segmentation module is skipped. The advantage of holistic approach is that it removes the segmentation overhead and makes the OCR process faster. The most important step is the classification as the accuracy of the whole system majorly depends on the classification or recognition step. With time several technologies or algorithms were introduced for classification. The first approach was template matching. Later structural analysis, statistical learning methods, feature extraction methods etc were introduced.

In this article, I have proposed a novel holistic model for recognition of handwritten words. The proposed hybrid model consists of two major parts: feature extraction and recognition. In feature extraction a combination of Convolution Neural Network layer and Bi-directional long short term memory layer is used to extract the global and temporal feature from the word image. These two features are combined by a compact Bi-liner pooling layer, which generates the final feature vector. Recognition is done using a CTC layer. This hybrid model is tested on three public datasets CMATERdb2.1.2, IIIT-HW-Dev and IIIT-HW-Telugu containing Bengali, Devanagari and Telegu scripts respectively. The results have been compared of the existing methods on these datasets with the proposed model. The proposed model is script independent, which is an advantage of this model. The highlights of the article are:

- This article proposes a holistic recognition scheme for offline multiple handwritten Indian scripts.
- Its objective is to eliminate the need of script dependent classifiers and built a classifier which can be trained and tested on any script.
- In the proposed hybrid model, both spatial feature and temporal feature are extracted from the handwritten word image and combined by a compact bi-liner pooling layer. For recognition Connectionist Temporal Classification has been used.
- CNN and BLSTM, both the feature extractor enables end to end training and testing of the model.
- The model is tested on three public databases of Bengali, Devanagari and Telegu scripts and achieved better results than most of the existing classifiers on these datasets.

The organization of the article is as follows:

A detailed survey on present algorithms for character classification can be found in Sect. “Brief Survey on Classification Algorithms”. The description of the dataset can be found in Sect. “Database”. The proposed model is discussed in Sect. “Proposed Model”, which also contains the results achieved in testing phase and error analysis on the results. Finally the conclusion and future scope are in Sect. “Conclusion”.

Brief Survey on Classification Algorithms

The working principal of early OCR systems was based on template matching. In [1] the authors, Kelner and Glauberman, used this template matching procedure using a magnetic shift register in 1956. In template matching by calculating the differences between the test image and template images the match score is determined.After using reflected light for template matching, in 1962 [2], Hannan and group combined electronics and optical techniques to achieve the goal of OCR. They used electron tube instead of light and generated the match score. After mechanical and electronics, OCR became a field of work for computer researchers. The template matching was done using a logical matching scheme named pee-hole technique. In this technique, the binarized input character was matched with the strokes, black and white portions and for constant size and width the logic was true. This technique was adopted by Solatron Electronics Group Ltd. The main weakness of template matching procedure is that it demands to create a huge number of samples, when there is variation is character style and font as template matching is very rigid in nature. To overcome this, with time different improvements on template matching procedure have been proposed. One of the improvements of the template matching procedure for character recognition is correlation coefficient. In coefficient correlation the template is matched with the test character and similarity score is generated along with the location where the similarity score is maximum. Depending on this score the character is identified. In [3, 4], this template matching approach were used.

Until now template matching was going well for machine printed characters, but when hand written characters came into the scenario, the template matching procedure failed. Though, concept wise both the procedures are similar but in structural analysis a new idea was added, which is feature extraction. Feature can be defined as distinguishing characteristics to identify a character. The structural analysis procedure is still an open problem. With time different types of feature extraction techniques are developed to achieve the perfection and it is still going on. The first structural analysis was done using stroke feature. In 1954, Rohland [5] counted the number of black pixels in vertical scan and used it as a feature in structural analysis. In structural analysis the image is divided into partitions and both topological (cross count of black pixels) and geometrical (length and width) features are detected. Along with different features, different partitioning approaches was also proposed. In [6] Johnson and in [7] Dimond used different partitioning approaches. Sometimes, weights are assigned to each partition, similar approach was done by Kamensky [8].

Until now most of the proposed approaches discussed, were based either on structural analysis or template matching. Later combining both the approaches hybrid scheme was proposed in different studies. In hybrid scheme advantages of both the schemes were adopted. The advantage of template matching scheme is, it is very strong to global changes. Advantage of structural analysis scheme is, it can detect the local features and adopt the variation. In [9] a hybrid approach was proposed by Munson, where the images were divided into sub regions and the objects of matching were local features within the sub region. Similar idea can be found in [10] by Greanias et al. Later IBM 1287 was built following this idea. In early age, the features used were very basic features, for example shape features, intensity features, which were not enough to capture the discriminating feature. Later more prominent features were proposed, which increased the character recognition accuracy. Few of them are Histogram of Gradient feature [11], Shift invariant feature [12], wavlet feature [13] etc.

These three approaches discussed so far, are the foundation of all the proposed approaches for classification or recognition till now. With time algorithms have been developed either to find a match, or to extract the feature else combination of the two approaches. Now let us discuss some of the widely used algorithms, which are the following:

Hidden Marcov model (HMM): In HMM statistically derived parameters are used and from huge amount of training set the probability parameters are estimated. Some of the works on HMM are [14, 15].

Support vector machine (SVM): One of the most famous approaches of classification/ recognition is Support Vector Machine (SVM). SVM works best when it is used for two class classification problem. In case of multi-class normally the classes are divided into two class for the classification and the difficulty lies here to choose the distribution of the classes. Some of the recent works on SVM are reported on [16, 17].

Decision tree: Another method of character classification is decision tree or graph based method. In decision tree based methods, from the total number of classes refinement is done based on the discriminating features. As the tree proceeds downward through the branches the tree moves towards more specialized features from the generic features. Finally, the leaf nodes represents the class labels. Some of the important works using decision tree are [18, 19].

Fuzzy logic: Fuzzy logic is another well known scheme used for character recognition. The working of fuzzy logic mainly mimics the human reasoning. It process the variable in such a way that it allows multiple possible truth values for a same variable. In [20, 21], fuzzy logic based system are proposed for offline character recognition.

Clustering: Clustering methods are widely used classification algorithms that mainly works in the dimensional space and creates groups or clusters based on similar properties of the data points of that space. One of the popular clustering algorithm is K-means clustering algorithm that helps to create k number of disjoint cluster based on the number of observations. Each cluster has a cluster center and the cluster center must be well separated. A new data point is assigned to one of the k clusters. One example of k-means clustering algorithm can be seen in [22]. Variation of k-means clustering is fuzzy c-means clustering. Fuzzy c-means algorithm has been applied for Bulgarian character recognition in [23].

Artificial neural networks: Towards the end of nineteenth century a new scheme stared being widely used in OCR technologies, i.e., artificial neural networks. Artificial neural network follows supervised learning and uses labelled data for training. Some of the works for classification are [24] for Devanagari script, [25] for Arabic character. Other works using artificial neural network can be found on [26–28]. One of the variation of ANN is deep learning algorithm. Some of the basic and widely used deep learning networks are Convolutional Neural Network, Recurrent Neural Network, Long Short Term Memory, Generative Adversarial Networks (GANs), Autoencoders etc. and their applications are [29–33], respectively.

In the early age, when the OCR engines started to develop, researchers combined two or more classifiers and proposed hybrid schemes for OCR. It will not be possible to discuss all of the studies, I have tried to incorporate some of them. In [34] a hierarchical model is proposed combining k-means and SVM. In [35] deep learning and decision tree is combined. In [36] svm is hybridized with multi dimensional RNN. Another work [37] combines MLP and SVM. Another combination of CNN and SVM can be seen in [38]. In [39], a CNN is combined with XGBoost. So, now we have a brief idea of the classification approaches.

Database

In the proposed work, examined the proposed model has been examined using three public databases which are CMATERdb2.1.2 [40], IIIT-HW-Dev [41], IIIT-HW-Telugu [42]. The databases contains offline handwritten words in Bengali, Devanagari and Telegu. The details of the databases are discussed in the following sections:

- *CMATERdb2.1.2*: This database is developed containing word images written in Bengali script. This database contains 18000 word samples with lexicon size of 1298. Some of the word samples are given in Fig. 1.

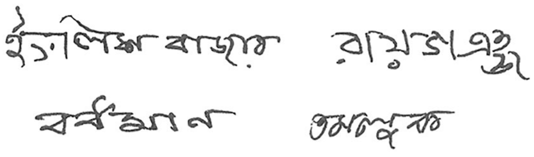

Fig. 1 Some sample images from CMATERdb2.1.2 database

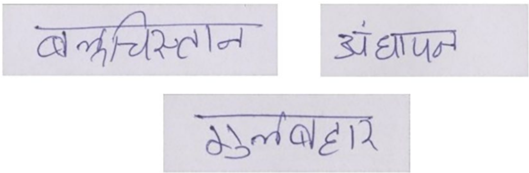

Fig. 2 Some sample images from IIIT-HW-Dev database

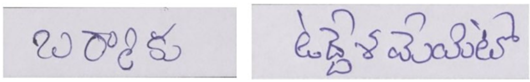

Fig. 3 Some sample images from IIIT-HW-Telegu database

- *IIIT-HW-Dev*: This dataset is developed by the researchers of CVIT, IIIT Hyderabad. In this database there are total 95,381 Devanagari words written by 12 individuals. Total vocabulary of 95,381 Devanagari words are present in the database. The database is divided into three subsets which are train, test and validation in the ratio of 70:15:15 and contain nearly 66766, 14307 and 14808 annotated word images. Some images are given in Fig. 2.
- *IIIT-HW-Telugu*: This dataset is developed by the researchers of CVIT, IIIT Hyderabad. This database contains handwritten words of Telegu words. Total 1,18,515 words are collected from 11 writers containing vocabulary of 12,945 words. The total words are divided into roughly 70:15:15 ratio, resulting 80637, 19980 and 17898 annotated word images in train, test and validation set respectively. Some images are given in Fig. 3.

Proposed Model

I have proposed a hybrid model for offline word recognition. As mentioned earlier, the model is a combination of CNN layers and BLSTM layers. The whole system overview of the proposed system is given in Fig. 4. The system can be divided into three main components: Feature extraction, training the classifier and testing the classifier or recognition. In feature extraction two types of features are extracted, spatial feature by convolutional layers and temporal features by Bi directional long short term memory layer. By combining these two types of features by Compact Biliner Pooling the final feature is formed. Compact biliner pooling helps in highlighting important features which in turn improves the recognition accuracy. The extracted feature next goes into the CTC layer for recognition. In the training phase the labelled word images are inputted into the classifier and the

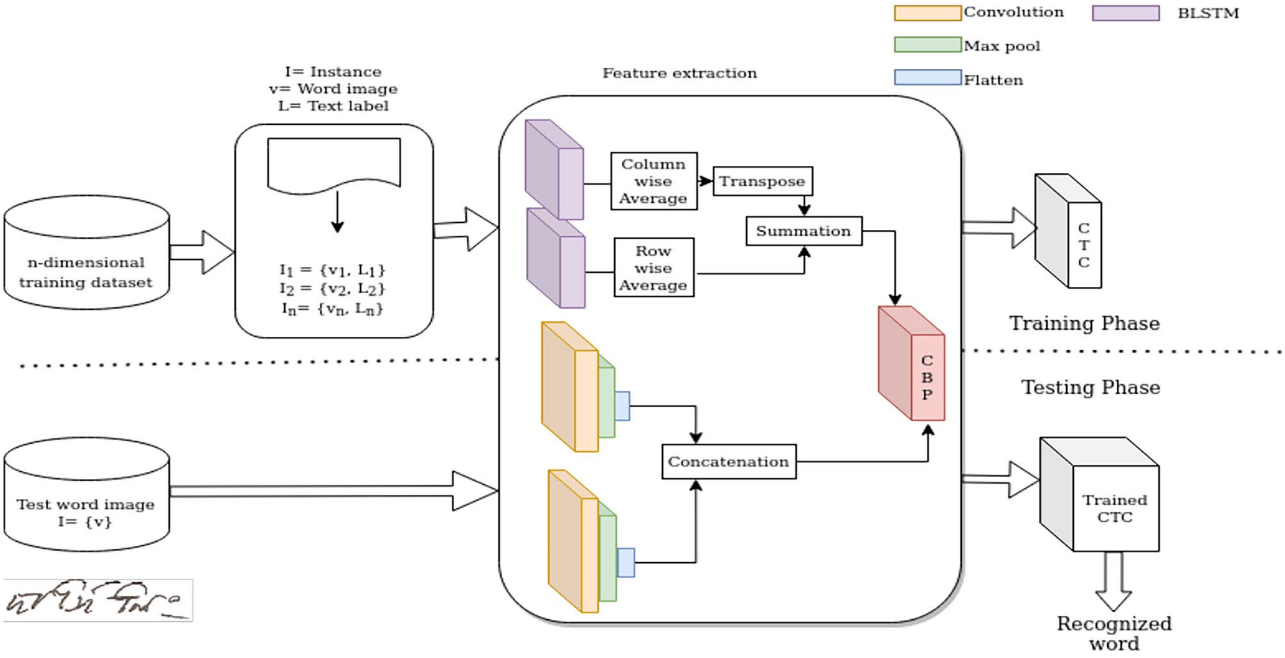

Fig. 4 Diagram of the Whole System

network in trained. In testing phase, the combined feature is extracted from the word image and the trained model recognizes the word image with the help of the trained CTC layer. In the following subsections the network and its components have been discussed in detail.

Feature Extraction

Feature extraction is one of the most important factors which affect the recognition accuracy drastically. In fact, a good feature extraction technique can produce a good result even with an average classifier. I have already discussed in section 2 that for years different types of feature extraction techniques are introduced by researchers. Starting from area, pixel intensity features now features like HOG, Shift invariant etc are being widely used.

The only aim of a feature extraction technique is to identify the discriminating feature and extract it so that the classifier can easily recognize the feature. So it can be said that along with strong classifier, strong feature extraction technique is equally important aspect of any OCR system. In feature extraction the main challenge is to choose the feature extraction technique that will work best for a specific task. Here manual intervention is needed, which may affect the system if wrong decision is taken. To overcome this, the machine extracted features were introduced which alleviate the need of hand crafted features. Such features are CNN features, LSTM features, BLSTM features. These feature extraction techniques extract the complex features automatically from the raw image. Additionally, this also discards the need of individual feature extraction unit as the recognizer and the feature extractor can be trained simultaneously.

To alleviate the need of handcrafted features and conduct the training simultaneously with the recognition module, a feature extraction technique combining CNN and BLSTM is proposed in this work. The CNN feature and BLSTM feature is combined with a compact bi linear pooling. The combined feature is the output of the feature extraction module. The feature extraction module is depicted in Fig. 5.

The input of the proposed word recognition system is binary word images of fixed size. A binary image is a combination of 0 and 1 valued pixels which is achieved using binarization. In CMATERdb2.1.2 the word images are already binary images but of variable size. So in CMATAR DB 2.1.2 as a pre-processing step only padding has been done with white pixels to fix the image size. The length of the images are fixed to 500 with padding in the right side. The width of the images are fixed to 200 with padding in the bottom. So the final image size after pre-processing is 500×200. IIIT-HW-Dev and IIIT-HW-Telegu databases contain RGB resized images of fixed length and width. So, Otsu's thresholding algorithm [43] has been used to get the binary word images. The word images of CMATERdb2.1.2 database, IIIT-HW-Dev database and IIIT-HW-Telugu database contain noise free handwritten images, so other than binarization and padding no other pre-processing method are followed.

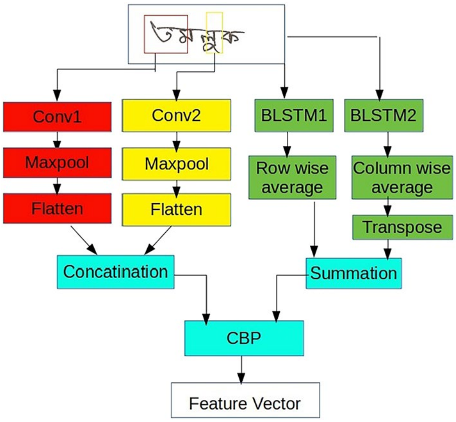

Fig. 5 Feature extraction module

As mentioned earlier, from the binarized two dimensional word image two types of features are extracted. By using the convolution layer the spatial feature or the shape, coordinates and regions of a word image are extracted. To capture more generic or global features the two CONV2d layer are used in parallel with different filter sizes. The configuration of the convolution layers are:

Conv1: Kernal size = maxnorm(5); Stride = 1; Dropout = 0.2;

Conv2: Kernal size = maxnorm(6); Stride = 1; Dropout = 0.2;

Following the CONV2d layers, maxpooling layers are used. The configuration of both the maxpool layers are same: Maxpool: pool = (2 × 2), Stride: 2 × 2; Maxpooling layers helps in dimension reduction as well as reduce the complexity of the extracted feature. The output of the two maxpool layers are flattened and a vector of one dimension is created. Both the 1D vectors are concatenated which creates the final global feature. Along with the global feature, to extract the temporal features two BLSTM layers of same dimension are used. The BLSTM layers are responsible to capture the correlations. In one of the BLSTM layer the word image is feed from top to bottom (BLSTM2) and in another BLSTM layer the word image is feed from left to right (BLSTM1). If the input word image is considered of (h × w). Then for BLSTM (Left to right) the input will be processed as (1 × height × width) and for BLSTM (Top to bottom) the input will be processed as (1 × width × height). Output of both the BLSTM layers is 2D matrices. The configurations of the BLSTM layers are:

BLSTM1: Hidden:16*w Drop: 0.3;

BLSTM2: Hidden:16*w Drop: 0.3;

From the 2D matrices, a row wise average operation is processed for BLSTM1 and a column wise average operation is processed for BLSTM2 layer. After the average operation on BLSTM2 layer a transpose operation is done to match the dimension of BLSTM1 and BLSTM2 layer so that the final summation operation can be applied on them. The global features extracted by the CNN layers and the temporal features extracted by BLSTM layers are combined by a compact biliner pooling. The bi linear pooling is a simple operation which results in the outer product of two input vectors. The advantage of bilinear pooling over simple mathematical operations is the resultant output vector focuses on fine grained details which in turn help to achieve high recognition accuracy. Moreover bi linear pooling has achieved state-of-the-art performance on wide range on machine learning task. One of the variations of bi-linear pooling is compact bi-linear pooling with an additional advantage of compactness. The global and temporal feature is transformed into sketch count vectors. After that the vectors are combined with a Fast Fourier Transform (FTT) convolution [44]. Another advantage of the CBP layer is that it allows back propagation which enables the proposed system to be an end to end model and feature extraction as an intrinsic part of the classification module. The output of the CBP layer is the final feature vector. The feature vector enters the CTC layer for recognition.

Training the Classifier

In training the classifier, the most important part is to train the CTC layer or the Connectionist temporal classification layer. The main advantage of CTC is that it allows sequential recognition discarding the need of segmentation specially in case of document processing. The model have been trained individually by the training set mentioned in Sect. "Database". The Devanagari and Telegu database are already divided into train, test and validation sets in 70:15:15 ratios. In [40], the database is divided into train and test set with 14,400 images and 3600 images respectively. A validation set from the training set by collecting 3600 images are created. This result in 10800 images in the training set. The input to the whole module is the input binary word image, the text label in the training phase. To train the CTC layer the characters of the script and a blank character are considered. By considering these characters the encoding is done in the CTC layer. The loss function calculated is CTC loss. In CTC loss first the probability of occurrence of the ground truth is computed and by taking negative logarithm the loss is computed. The loss is back propagated through the network to update the weights and produce the trained model. RMSProp optimizer and 0.001 learning rate while training the network have been used. The network training continued until the improvement in validation accuracy became static.

Testing the Classifier

In Sect. "Database", the experimental datasets are already discussed. The trained model have been tested on the test set of the databases. It is to be noted that the training and testing of the different scripts are done individually. The databases used to train and test the models are public databases and on these databases in literature several research work are proposed. The results obtained from the proposed model are compared with some of those with. In the testing phase the CTC uses the best path to decode the text. It is achieved in two steps. In the first step, max probability is calculated of the characters and the character with maximum probability is considered the desired character. In the second step the duplicate characters and blank characters are removed to get the actual word or character sequence.

Result and Discussion

As mentioned the model is tested on the three databases. The results obtained on the test set are discussed in Table 1.

For word recognition, two types of accuracies are calculated from the computed label error. First one is the character level accuracy: for a word image, the number of characters that are correctly identified is the character level accuracy. Second accuracy is calculated, which is word level accuracy. In a word image, if the correctly recognized characters by total number of characters are greater than a certain threshold then the line is considered as a correct word recognition. The accepted threshold for the work is 90%. The computer character level and word level accuracy as reported in Table 1.

These two are standard metrics for handwriting recognition tasks. In Table 2 The results on CMATARdb 2.1.2

Table 1 Classification accuracy obtained on the databases

Database	Test set size	Recognized words	Accuracy character	Accuracy word
CMATERdb2.1.2	3600	3294	96.42	91.5
IIIT-HW-Dev	14307	13562	94.79	90.84
IIIT-HW-Telugu	19980	17956	95.07	89.86

Table 2 Comparison on CMATARdb 2.1.2 database

Algorithm	Accuracy(%)
SVM	83.64
MLP	81.72
Proposed model	96.42 (Character level)
Proposed model	91.50 (Word level)

database reported on [40] are compared. From the table, it can be seen that the proposed method has achieved high accuracy on the dataset.

In Tables 3 and 4 I have compared the results on IIIT-HW-Dev database and IIIT-HW-Telugu database reported on [41] and [42] respectively. It can be observed that the method has performed better than most of the proposed methods. Though in case of Devanagari script it failed to perform better than Mixed-IAM-SCNN-BLSTM-Lexi model. Similarly, in recognition of Telegu script the IAM-SCNN-BLSTM-Lexi has performed better.

Error Analysis

Though the proposed model has performed really well compared to state-of-the-art methods, but for few word classes the error occurred. After examine the errors it has been found that the errors mainly occurred for the similarity in character shape. All the three scripts on which I have worked have this challenge. A small variation in one character shape represents another character. For example, in Bengali script if an arc is added with the characterব, then it represents another character ক. Similarly, if a dot is added with ব then it creates another character র.

Another challenge that have high impact on classification is the complex character shape. All Indian script contains a concept of compound characters, where two basic characters when combined together creates a modified shape. This increases the number of character classes as well as the challenges in recognition of such characters. Those characters drops the recognition accuracy.

Table 3 Comparison on IIIT-HW-Dev database

Algorithm	Accuracy(%)
CNN-BLSTM	82.44
SCNN-BLSTM	84.88
Synth-SCNN-BLSTM	89.98
IAM-SCNN-BLSTM	90.25
Mixed-SCNN-RNN	91.36
Mixed-IAM-SCNN-BLSTM-Lexi	95.10
Proposed model	94.79

Table 4 Comparison on IIIT-HW-Telugu database

Algorithm	Accuracy(%)
CNN-BLSTM	90.85
SCNN-BLSTM	92.17
Synth-SCNN-BLSTM	94.78
IAM-SCNN-BLSTM	95.48
IAM-SCNN-BLSTM-Lexi	99.7
Proposed model	95.07

The final and the most important factor, which contributes in the error is the imperfections that are present in a handwritten document. A handwritten document contains some sort of degradation like skew, tilted lines, un even line gaps etc. More over, handwritten documents contain different type of writing due variation of writers. So, the recognition of handwritten documents is a complex problem to solve.

Conclusion

In this present article, a holistic word recognition model is proposed. The model is tested on Bengali, Devanagari and Telegu scripts. In holistic word recognition approach a given word image is considered as a single and indivisible unit. For that purpose a hybrid model is proposed. The model extracts global feature from a word image by using convolution layer and temporal feature by using BLSTM layer. Both the features are combined by a Compact Bi-linear Pooling layer. The final feature vectors goes into the CTC layer which recognizes the word.

This proposed model is font and shape independent that is why it worked for all three scripts. The results achieved are better than the existing methods but there is still some scope of improvements. An error analysis has been done on the results and observed that the shape complexity, imperfections in a document, similarity in character shape are the main reasons that effects the accuracy. Additionally, any other preprocessing step other than binarization and padding have not been followed. No pre-processing like slant correction, skew correction are applied on the handwritten words, which are very common pre-processing that are applied for handwritten word recognition. Inclusion of such a pre-processing module like skew correction may increase the recognition accuracy in future. Additionally inclusion of post processing technique like dictionary learning will increase the total accuracy more.

Data availability Three databases have been used in this work. CMATARDB database is available at https://code.google.com/archive/p/cmaterdb/downloads?page=1. IIITHW-DEV Database is available at https://cvit.iiit.ac.in/research/projects/cvit-projects/indic-hw-data.

IITHW-TELEGU database is available at https://cvit.iiit.ac.in/research/projects/cvit-projects/indic-hw-data.

Declarations

Conflict of Interest The authors declare that there is no conflict of interest.

References

1. Glauberman M. Character recognition for business machines. Electronics. 1956;29(2):132–6.
2. Hannan W, Olsen T, Schanne J, Burton G, Heagerty W. Rca print reader investigation and tests. Raytheon Co Waltham Mass: Technical report; 1962.
3. Veeramachaneni S, Nagy G. Towards a ptolemaic model for ocr. In: Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings., 2003;vol. 3, pp. 1060–1060. Citeseer.
4. Mao J. A case study on bagging, boosting and basic ensembles of neural networks for ocr. In: 1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98CH36227), 1998;vol. 3, pp. 1828–1833. IEEE.
5. Rohland WS, Traglia PJ, Hurley PJ. The design of an ocr system for reading hand written numerals. In: Proceedings of the December 9–11, 1968, Fall Joint Computer Conference, Part II, 1968;pp. 1151–1162.
6. Johnson RB. Indicia-controlled record perforating machine. Google Patents. US Patent 2,741,312, 1956.
7. Dimond T. Devices for reading handwritten characters. In: Papers and Discussions Presented at the December 9–13, 1957, Eastern Joint Computer Conference: Computers with Deadlines to Meet, 1957; pp. 232–237.
8. Kamentsky LA. The simulation of three machines which read rows of handwritten Arabic numbers. IRE Trans Electron Comput. 1961;3:489–501.
9. Munson JH. Experiments in the recognition of hand-printed text, part I: character recognition. In: Proceedings of the December 9–11, 1968, Fall Joint Computer Conference, Part II, 1968; pp. 1125–1138.
10. Greanias EC, Meagher PF, Norman RJ, Essinger P. The recognition of handwritten numerals by contour analysis. IBM J Res Dev. 1963;7(1):14–21.
11. Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, 2005; pp. 886–893. IEEE.
12. Li Y, Fermuller C, Aloimonos Y, Ji H. Learning shift-invariant sparse representation of actions. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010; pp. 2630–2637. IEEE.
13. Pun C-M, Lee M-C. Extraction of shift invariant wavelet features for classification of images with different sizes. IEEE Trans Pattern Anal Mach Intell. 2004;26(9):1228–33.
14. Hassin AH, Tang X-L, Liu J-F, Zhao W. Printed Arabic character recognition using hmm. J Comput Sci Technol. 2004;19(4):538–43.
15. Das RL, Prasad BK, Sanyal G. Hmm based offline handwritten writer independent English character recognition using global and local feature extraction. Int J Comput Appl. 2012;46(10):45–50.
16. Pino R, Mendoza R, Sambayan R. Optical character recognition system for Baybayin scripts using support vector machine. PeerJ Comput Sci. 2021;7:360.
17. Katiyar G, Katiyar A, Mehfuz S. Off-line handwritten character recognition system using support vector machine. Am J Neural Netw Appl. 2017;3(2):22–8.
18. Raja S, John M. A novel Tamil character recognition using decision tree classifier. IETE J Res. 2013;59(5):569–75.
19. Shlien S. Multifont character recognition for typeset documents. Int J Pattern Recogn Artif Intell. 1988;2(04):603–20.
20. Hanmandlu M, Mohan KM, Chakraborty S, Goyal S, Choudhury DR. Unconstrained handwritten character recognition based on fuzzy logic. Pattern Recogn. 2003;36(3):603–23.
21. Mukherji P, Rege PP. Shape feature and fuzzy logic based offline Devanagari handwritten optical character recognition. J Pattern Recogn Res. 2009;4:52–68.
22. Mukherjee J, Parui SK, Roy U. Degraded Bangla Character Recognition by k-NN classifier. Int J Comput Sci Eng. 2019;07:42–7.
23. Krasteva R. Bulgarian hand-printed character recognition using fuzzy c-means clustering. Bulg Acad Sci Probl Eng Cybern Robot. 2002;53:112–7.
24. Keeni K, Shimodaira H, Nishino T, Tan Y. Recognition of Devanagari characters using neural networks. IEICE Trans Inform Syst. 1996;79(5):523–8.
25. Nawaz SN, Sarfraz M, Zidouri A, Al-Khatib WG. An approach to offline Arabic character recognition using neural networks. In: 10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003, vol. 3, 2003; pp. 1328–1331. IEEE.
26. Zhai X, Bensaali F, Sotudeh R. Ocr-based neural network for anpr. In: 2012 IEEE International Conference on Imaging Systems and Techniques Proceedings, 2012; pp. 393–397. IEEE.
27. Singh R, Yadav C, Verma P, Yadav V. Optical character recognition (ocr) for printed Devanagari script using artificial neural network. Int J Comput Sci Commun. 2010;1(1):91–5.
28. Desai AA. Gujarati handwritten numeral optical character reorganization through neural network. Pattern Recogn. 2010;43(7):2582–9.
29. Maitra DS, Bhattacharya U, Parui SK. CNN based common approach to handwritten character recognition of multiple scripts. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), 2015; pp. 1021–1025. IEEE.
30. Messina R, Louradour J. Segmentation-free handwritten Chinese text recognition with LSTM-RNN. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), 2015; pp. 171–175. IEEE.
31. Mukherjee J, Parui SK, Roy U. NN-based analytic approach to symbol level recognition for degraded Bengali printed documents. Sādhanā. 2020;45(1):1–22.
32. Hayashi H, Abe K, Uchida S. Glyphgan: style-consistent font generation based on generative adversarial networks. Knowl-Based Syst. 2019;186: 104927.
33. Gogna A, Majumdar A. Discriminative autoencoder for feature extraction: application to character recognition. Neural Process Lett. 2019;49(3):1723–35.
34. Liu W-C, Lin C-H. A hierarchical license plate recognition system using supervised k-means and support vector machine. In: 2017 International Conference on Applied System Innovation (ICASI), 2017; pp. 1622–1625. IEEE.
35. Sampath A, Gomathi N. Decision tree and deep learning based probabilistic model for character recognition. J Central South Univ. 2017;24(12):2862–76.
36. Bezerra BLD, Zanchettin C, de Andrade VB. A mdrnn-svm hybrid model for cursive offline handwriting recognition. In: International Conference on Artificial Neural Networks, 2012; pp. 246–254. Springer.

37. Azevedo WW, Zanchet C. A mlp-svm hybrid model for cursive handwriting recognition. In: The 2011 International Joint Conference on Neural Networks, 2011; pp. 843–850. IEEE.
38. Ahlawat S, Choudhary A. Hybrid CNN-svm classifier for handwritten digit recognition. Procedia Comput Sci. 2020;167:2554–60.
39. Mukherjee J, Roy U. Recognition of Degraded Bangla Documents Using Hybrid Deep Neural Network Model. In: 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2021; pp. 254–259. IEEE.
40. Bhowmik S, Malakar S, Sarkar R, Basu S, Kundu M, Nasipuri M. Off-line Bangla handwritten word recognition: a holistic approach. Neural Comput Appl. 2019;31(10):5783–98.
41. Dutta K, Krishnan P, Mathew M, Jawahar CV. Offline handwriting recognition on Devanagari using a new benchmark dataset. In: DAS 2018.
42. Dutta K, Krishnan P, Mathew M, Jawahar CV. Towards spotting and recognition of handwritten words in Indic scripts. In: ICFHR 2018.
43. Mukherjee J, Parui SK, Roy U. NN-based analytic approach to symbol level recognition for degraded Bengali printed documents. Sādhanā. 2020;45(1):1–22.
44. Cheikhrouhou A, Kessentini Y, Kanoun S. Multi-task learning for simultaneous script identification and keyword spotting in document images. Pattern Recogn. 2021;113: 107832. https://doi.org/10.1016/j.patcog.2021.107832.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.