Received: 16 May 2024, Revised: 02 July 2024, Accepted: 09 July 2024, Online: 23 July 2024

DOI: https://doi.org/10.55708/js0307005

A Computational Approach for Recognizing Text in Digital and Natural Frames

Mithun Dutta[1], Dhonita Tripura[1], Jugal Krishna Das[2]

[1]Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati-4500, Bangladesh

[2]Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka-1342, Bangladesh

*Corresponding author: Mithun Dutta, mithundutta92@gmail.com

ABSTRACT: Acquiring tenable text detection and recognition outcomes for natural scene images as well as for digital frames is very challenging emulating task. This research approaches a method of text identification for the English language which has advanced significantly, there are particular difficulties when applying these methods to languages such as Bengali because of variations in script, morphology. Text identification and recognition is accomplished on multifarious distinct steps. Firstly, a photo is taken with the help of a device and then, Connected Component Analysis (CCA) and Conditional Random Field (CRF) model are introduced for localization of text components. Secondly, a merged model (region-based Convolutional Neural Network (Mask-R-CNN) and Feature Pyramid Network (FPN)) are used to detect and classify text from images into computerized form. Further, we introduce a combined method of Convolutional Recurrent Neural Network (CRNN), Connectionist Temporal Classification (CTC) with K-Nearest Neighbors (KNN) Algorithm for extracting text from images/ frames. As the goal of this research is to detect and recognize the text using a machine learning-based model a new Fast Iterative Nearest Neighbor (Fast INN) algorithm is now proposed based on patterns and shapes of text components. Our research focuses on a bilingual issue (Bengali and English) as well as it producing satisfactory image experimental outcome with better accuracy and it gives around 98% accuracy for our proposed text recognition methods which is better than the previous studies.

KEYWORDS: Iterative, Component, Recognition, Bilingual

1. Introduction

Recently, a notable surge in research efforts dedicated to text recognition, a critical aspect within various image processing and vision algorithms. Scene text recognition, in particular, emerges as a challenging yet highly advantageous endeavor involving the identification of text within natural images. Deciphering text embedded in digital photographs poses a significant challenge. Moreover, image text recognition not only forms the foundation of information retrieval but also plays a crucial role in enabling effective human-machine communication. The potential applications of a system capable of identifying and extracting text from real-world images are extensive. Additionally, databases comprise texts of diverse types, including manually edited caption texts and scene texts with various orientations, thereby introducing layers of complexity to text detection and recognition. Notably, viewers often prioritize text when interpreting images, emphasizing the significance of text detection and recognition in aiding human comprehension of intricate visual compositions.

Numerous methodologies have been developed to address the challenges associated with text detection and recognition in digital frames and natural scene images characterized by varying orientations, scripts, font sizes, and other factors. Previous studies suggest that many existing methods focus on specific data types and address particular issues such as complex backgrounds, low contrast, or multiple scripts and orientations, resulting in suboptimal performance when confronted with data influenced by multiple adverse factors [1, 2, 3]. The primary challenges stem from various sources: frames captured with low-resolution cameras which make low contrast inputs; images also taken with high-resolution devices may exhibit high contrast input but frequently feature complex backgrounds, leading to an increased incidence of false positives; images sourced

from digital platforms often showcase a multitude of character components.

In this study, we employed mask-RCNN for character and text detection and recognition of both Bangla and English sentences. R-CNN (Region-based Convolutional Neural Network), including its variant Mask R-CNN, represents a class of machine learning models tailored explicitly for computer vision tasks, particularly object detection. Mask R-CNN propagates Faster R-CNN by introducing an additional pulse for outputting object masks alongside class labels and bounding-box offsets, thereby facilitating the extraction of finer spatial layouts of objects. Additionally, our proposed method incorporates the Fast Iterative Nearest Neighbor (Fast INN) algorithm, which utilizes shape information to detect candidate components. Subsequently, Fast INN extracts shape of script. The novelty of our approach is to explore these fundamental concepts to address the ongoing challenges of text detection and recognition without imposing rigid constraints, thereby aiming for enhanced performance and versatility.

2. Literature Review

In [1], the authors proposed a method for detecting and tracking text on a variety of large and small text blocks. This is consists of two different modules: first one is a sum of squared difference (SSD) and another one is a contour-based module. The main purpose of the paper proposed [2] text recognition, which submitted a petition on the text lines detection.

In [3, 4], the authors proposed the text detection method which is carried out through edge detection, local thresholding, and hysteresis edge recovery. In [5, 6], the researches proposed an SVM classifier which used to identify text from the selected features object. In [7, 8], the authors addressed a model that is based on invariant features. In [9, 10], the authors proposed signal extraction techniques based on RWT method for extracting test information. They also proposed text line extraction based on integrated K-shortest path optimization. In [11, 12], the authors used the NLP to analyze and become acquainted the data set and also used both Naive Bayes and logistic regression algorithms to determine the best accuracy system.

3. Proposed System

3.1. Proposed Framework

In this section, we outline the operational procedures of the proposed machine learning-based text recognition framework and delve into its logical execution. The experiment integrates a preprocessing technique tailored to mitigate unwanted noise and artifacts effectively. Initially, we employ the Connected Component Analysis (CCA) model to determine adjacent pixels based on predefined pixel connectivity criteria. Additionally, we utilize edge detection methodologies. Typically, false values in the image are associated with background pixels, while valid values represent foreground or object pixels. The connected component analysis methodology segments the image into regions, identifies relevant areas, and extracts text from these regions. Subsequently, the image is partitioned into smaller components, which are then classified based on their geometric attributes.

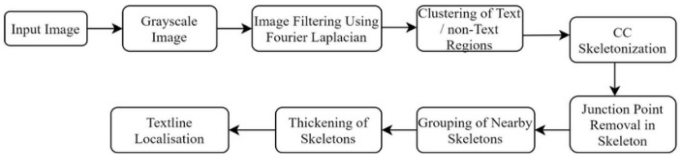

Figure 1: Text localization based on connected component analysis

Text localization involves the utilization of conditional random field (CRF), a statistical modeling technique employed when class labels for different inputs which are independent. Specifically, in image segmentation, the assignment of a class label to a pixel is contingent upon the labels of its neighboring pixels. To calculate the confidence of region texts, translate the output, conditioned variable (s), s ∈ {acc, rej}, and t can be calculated based on the Bayes' theorem, where Pt (s|x) are calculated is as follows.

$$P_t(x|s) = \frac{P_t(s|x)P_t(x)}{\sum_x^1 P_t(s|x)P_t(x)} = \frac{P_t(s|x)P_{t-1}(x|accept)}{\sum_x^1 P_t(s|x)P_{t-1}(x|accept)}$$

Image Segmentation: Niblack's binarization formula [13] is defined as:

$$b(x) = \begin{cases} 0 & , \ if\ gray(x) < \mu_r(x) - k \cdot \sigma_r(x); \\ 255 & , \ if\ gray(x) > \mu_r(x) + k \cdot \sigma_r(x); \\ 100 & , \ other, \end{cases}$$

And an implementation, P (Y |X) is as:

$$P(Y|X) = \frac{1}{Z(X)} \exp\big(-E(X, Y, N, \lambda)\big),$$

Where Z(X) is the normalization constant which is used to solve the problem, and the best label Y ∗ might be considered by maximizing conditional probability P (Y |X) to minimize the energy.

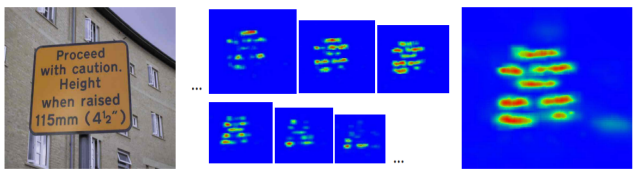

Figure 2: (a) original image (b) text confidence maps (c) text confidence map for the original image

Then, Mask R-CNN is an advance computer vision model used for object instance segmentation. A text detector based on Mask R-CNN is used, and fully convolutional

networks mainly inspire the methods. First, CNN is adopted to detect text blocks from which character candidates are extracted.

Then, FPN (Feature Pyramid Network) is used to predict the corresponding segmentation masks. Convolutional Neural Network (CNN) is a one kind of AI based neural network which uses for recognizing and processing image components that is optimized to process pixel data. It integrates data components detection task where the goal is to detect object through the formation of bounding box prediction of an image and a semantic segmentation task, that may classify every pixel into pre-defined object categories.

This is an execution of Mask R-CNN on Python 3, and the model generates bounding boxes and segmentation masks for each data object in the frame, which is also based on FPN.

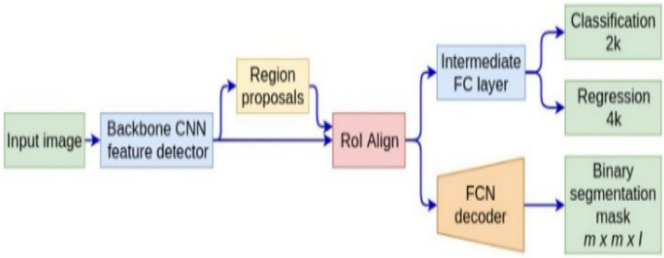

Figure 3: Flowchart of segmentation with Mask R-CNN

Mask R-CNN combines object detection and instance segmentation by using a Feature Pyramid Network (FPN) and the Region of Interest Align (ROIAlign) layer, Mask R-CNN which achieves strong performance and accuracy rate.

Now, a combined method of Convolutional Recurrent Neural Network (CRNN), Connectionist Temporal Classification (CTC) with K-Nearest Neighbors (KNN) Algorithm for extracting text from images. Extracting text of different shapes and sizes, various directions and orientations from images, especially from web pages and sites which is augmented reality assistance systems, and content moderation in social media platforms with the combined method. Combining Convolutional Recurrent Neural Networks (CRNN) with Connectionist Temporal Classification (CTC) allows CRNN to handle variable-length sequences without requiring an explicit alignment between input images and text outputs.

Suppose a given sequence $X = [x1, x2, x3, x4, x5\ldots, xT]$, such as any voice, and by mapping the sequence to its corresponding outcome sequence $Y = [y1, y2, y3, y4, y5\ldots,yU]$. Our main object is to find out an accurate mapping between $X's$ and $Y's$.

In particular:

- X and Y are different in length or dimension.
- The lengths ratio of X and Y may vary from each to other.
- No accurate alignment of X and Y.

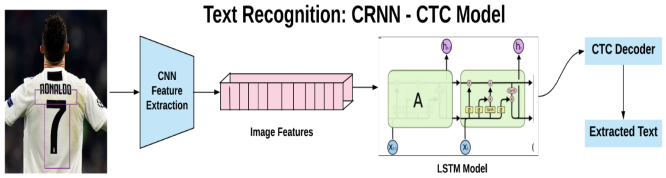

Figure 4: Text Recognition Pipeline Architecture

In CTC, a conditional probability $p(\pi|x)$ can be labeled as follows:

$$p(\pi|x) = \prod_{t=1}^{T} y_{\pi_t}^t, \forall \pi \epsilon L'^T$$

$$p(s|x) = \sum_{\pi \epsilon B^{-1}(8)}^{1} p(\pi|x)$$

CTC loss function is described as

$$L_{CTC} = -\ln p(s|x)$$

Further, the k-nearest neighbor (KNN) classifier uses proximity to make classifications or predictions about the grouping of an individual data point and fundamentally relies on a distance metric. The most common choice is the Minkowski distance dist(x,z)=(d Σ r=1|xr−zr|p)1/p. Finally, the proposed method, Fast Iterative Nearest Neighbor algorithm (Fast INN), is applied to recognize the text. The FINN algorithm is suggested by the inspiration of KNN, which is one of the simplest Machine Learning algorithms. The proposed algorithm estimates the similarity between the new case/data and the available cases and places the new case in the category that is most similar to the available data. During the training phase, the FINN algorithm stores the datasets, and when it gets new data, it then classifies that data into a category that is similar to the latest data. As the FINN algorithm helps to identify the nearest points or the groups for a query point and to determine the closest groups or the nearest points for a query point, we need to calculate some distance metrics (Euclidean distance, Manhattan distance, Minkowski distance, etc.).

Our Fast Iterative Nearest Neighbor (Fast I-NN) working can be explained through the following algorithm:

Step-1: Select the number of the neighbors (Suppose the number is N)

Step-2: Calculate the distance (Euclidean) of N number of neighbors

Step-3: Take the N nearest neighbors as per the calculated Euclidean distance.

Step-4: Among these N neighbors (adjacent data points), count the number of the data points in each category.

Step-5: Assign the neoteric data points to that category

Step-6: Find for which the number of the neighbor is maximum.

Step-7: Calculate the distance is minimum.

Step-8: Compare the new data pixel with residual data points to that category.

Step-9: Predict and Print the output.

3.2. *Datasets*

In this study, research will use most of the original dataset images, which are taken with a Kodak DX7590 (5.0 MP) still frame camera and a Sony DCR-SR85E handy camera in still mode, as well as from Kaggle. From different background images are taken like roadways, digital banner, signboard, wall writings, etc. Data may be prepared for a machine learning approach using Python language through Jupyter Notebook. The proposed text recognition method is admired in both cases qualitatively and quantitively. Datasets include original images of dimension either (321x481) or (481x321) pixels, and from the datasets randomly, 80 percent of images are used for training and 20 percent of images for testing. For Augmentation, we used to tilt the image with different angle, zooming, rotation, shearing width and height shifting, and horizontal and vertical flipping techniques in a Python setup environment.

3.3. *Configure and Train the Model*

The model, Default-Trainer, Default-Predictor, Color-Mode, and Visualizer were all imported. We also imported mask-RCNN with FPN, as well as other models and the associated checkpoints. We defined the dataset and other settings, such as total number of workers, batch size, as well as the number of classes, by importing Default-Trainer. We trained the model further after initializing it using pre-trained weights. We've trained our new model in Jupyter Notebook and Google Colab wherein it could dash and make advantages. The system generated an output folder, into which the trainer saved the training checkpoints to count the maximum and best checkpoints for the trained model. The trained software automatically saves the checkpoints in the output directory during training.

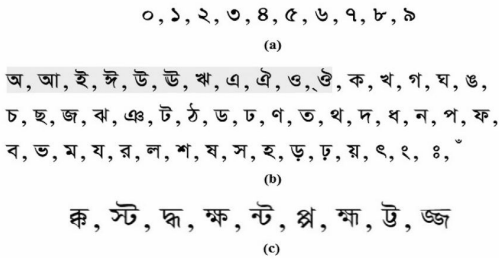

Figure 5: Bangla character (a) Digits (b) Vowels and consonants (c) Compound/joint characters

Figure 6: Frequently used grapheme roots of Bangla Char

3.4. *Eliminate unwanted Character*

This system eliminates multiple-identified characters from the object array after replacement. Machine learning detection models can predict the same object or predict the same instance with a different class at another time. Consequently, we must remove redundant objects from the array, keeping only one instance wherein the same instance is not recognized more than once.

3.5. *Recognize the Text*

All the images previous step is used for both to train and test [9, 10]. Our proposed method has been accustomed for text line recognition. The following figure shows the step of capturing to recognize.

Figure 7: From image capture to text recognition

4. Experimental Result

4.1. *Performance Evaluation*

Maximum accuracy rate indicates the perfect prediction for the localization and recognition of text. A

minimal false positive rate indicates fewer redundant pixels in unwanted regions resulting in a higher accuracy rate. This provides a balanced measure of the model's ability to correctly identify both positive and negative instances. In this paper, we have adapted the performance of machine learning models. The figure below reincarnates the results of the trained machine learning. Here, Figure a. shows accuracy rate of performance, and Fig b. shows false negative prediction, Fig c. indicates accuracy of the foreground classification, Fig d. shows the accuracy rate of mask RCNN, Fig e. represents the false negative of mask RCNN, and lastly, Fig f. shows the total loss of the mask RCNN.

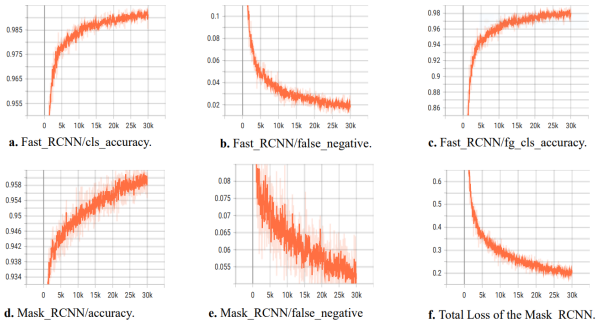

Figure 8: Result of the Trained Machine Learning Model

4.2. Experimental Result and Discussion

Our proposed model for future bilingualism text recognition will perform very well. We have used a huge number of data set to acquire a better performance and a well predicted result. To achieve the performance and print the result we had to perform our operational work properly in several sectors from where quite a few data were lost. Using the proposed algorithm through python language significant percentage accuracy rate is achieved. To calculate the sample size and accuracy rate we need the total number of successfully recognized images. The following table depicts the results of our study of all steps together.

Table 1: Summarizes the accuracy in terms of its gradations

Gradations	Text Detection	Text Extraction	Text Recognition
Total Data Samples	19200	19200	19200
Correct for Bangla	18430	18110	17998
Accuracy for Bangla	95.98%	94.32%	93.74%
Correct for English	19110	19030	18810
Accuracy for English	99.53%	99.11%	97.97%

5. Conclusion

In this research, we have well-acquainted and analyzed various text detection techniques for natural scenes as well as for digital images. In here, we introduced a unique model for detecting and recognizing the Bangla and English text using machine learning and the proposed Fast Iterative Nearest Neighbor (FINN) method. Before the system detects text, images are initially taken from the nature or surrounding environment and from that image texts are extracted. After that, text recognition was done by a merge proposed method where it gave false positive results or was unable to recognize some words or parts of them [13]. Then, the proposed method works effectively on curved or slanted Bengali and English texts as well. Pre-processing techniques make this model more robust against any random artifacts or unwanted blurring effects. However, the system gives about 98% accuracy for the proposed text recognition model.

References

[1] H. Li, D. Doermann, and O. Kia, "Automatic Text Detection and Tracking in Digital Video," *IEEE Transactions on Image Processing*, vol. 9, no. 1, 147-156, Jan. 2000, doi: 10.1109/83.817607.

[2] D. Chen, J.-M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," *Pattern Recognition*, vol. 37, 595-608, 2004, doi:10.1016/j.patcog.2003.06.001.

[3] P. Shivakumara, T. Q. Phan, and C. L. Tan, "A Laplacian Approach to Multi-Oriented Text Detection in Video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, 412-420, Feb. 2011, doi:10.1109/TPAMI.2010.166.

[4] M. R. Lyu, J. Song, and M. Cai, "A Comprehensive Method for Multilingual Video Text Detection, Localization, and Extraction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, 243-255, Feb. 2005, doi:10.1109/TCSVT.2004.841653.

[5] X.-C. Yin, Z.-Y. Zuo, S. Tian, and C.-L. Liu, "Text Detection, Tracking and Recognition in Video: A Comprehensive Survey," *IEEE Transactions on Image Processing*, 1-24, 2015, doi: 10.1109/TIP.2016.2554321.

[6] Q. Ye, Q. Huang, W. Gao, and D. Zhao, "Fast and robust text detection in images and video frames," *Image and Vision Computing*, vol. 23, 565-576, 2005, doi:10.1016/j.imavis.2005.01.004.

[7] K. S. Raghunandan, P. Shivakumara, S. Roy, G. H. Kumar, U. Pal, and T. Lu, "Multi-Script-Oriented Text Detection and Recognition in Video/Scene/Born Digital Images," *IEEE Transactions on Circuits and Systems for Video Technology*, doi: 10.1109/TCSVT.2018.2817642, 2018.

[8] M. Cai, J. Song, and M. R. Lyu, "A New Approach for Video Text Detection," in *Proceedings. International Conference on Image Processing*, 2002, 117-120, doi: 10.1109/ICIP.2002.1037973.

[9] C. S. Shin, K. I. Kim, M. H. Park, and H. J. Kim, "Support Vector Machine-Based Text Detection in Digital Video," in *Proceedings of IEEE ICIP*, 2000, 634-641.

[10] H. Wang, S. Huang, and L. Jin, "Focus On Scene Text Using Deep Reinforcement Learning," in *Proceedings of the 24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, Aug. 2018, 3759-3766, doi: 10.1109/ICPR.2018.8545022.

[11] Y. Wang, "Extraction Algorithm of English Text Information from

Color Images Based on Radial Wavelet Transform," *Special Section on Gigapixel Panoramic Video with Virtual Reality*, Aug. 2020, doi 10.1109/ACCESS.2020.3020621.

[12] O. Y. Ling, L. B. Theng, A. Chai, and C. McCarthy, "A Model for Automatic Recognition of Vertical Text in Natural Scene Images," in *2018 8th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia, 2018, doi: 10.1109/ICCSCE.2018.8685019.

[13] M. Dutta, A. Mohajon, S. Dev, D. S. Bappi, and J. K. Das, "Text Recognition of Bangla and English Scripts in Natural Scene Images," *International Journal of Advanced Research in Science and Technology*, vol. 12, no. 10, 1137-1142, 2023.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (https://creativecommons.org/licenses/by-sa/4.0/).