International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Arbitrary Shape Hindi Text Detection For Scene Images

Khanaghavalle. G. R[1], Dr. N. Rajeswari[2],

[1]Student, Department of Computer Science and Engineering, Sri Venkateswara College of Engineering.
[2]Associate Professor, Department of Computer Science and Engineering, Sri Venkateswara College of Engineering.

Abstract - *Hindi Text Detection in scene images is a complex and challenging field of research. The main challenge in scene text detection is a complex background, different font size, and arbitrary orientations. Deep Neural Networks has gained its popularity in scene text detection. Many research works have been done in scene text detection for English. But these works don't perform well for Indian languages such as Hindi. Hindi characters in images generally exist in Devanagari scripts. Hindi Characters consist of many curved information and stroke features which makes it difficult to locate the text in scene images. Thus, we propose a novel Hindi Text Detector. Backbone network of our text detector is ResNet. The proposed work is able to locate Hindi Text which is of arbitrary shape. It also separates the text instances which are close to each other. The proposed work is evaluated against MLT-2019 Dataset for the Hindi language. Effectiveness of the proposed text detector is demonstrated by the experimental results*.

Key Words: Scene text detection, Deep Learning, ResNet, Hindi text detection.

1. INTRODUCTION

Scene text detection is a fundamental problem of detecting text from natural scene images. Text always contains semantic information that helps us to understand the environment. Scene text frequently appears in product packages, license plates, products bills, signage etc. Reading text in images finds its application in image-based geolocation, visual question answering system, and translation of text from images.

The need for ubiquitous access is increasing due to the rapid growth of the internet and broadcast. Therefore, huge databases are created with data such as videos and images captured by various electronic devices. These datasets have multiple scene images with different types of scripts. This makes the scene text detection task much more difficult. Computer Vision and Image Processing has gained its popularity in recent years and many researchers are working in this area to understand images with complex background and low-resolution images.

Text in an image always provides important information to the human [6]. Thus our focus is on images containing text in it. Text detection plays an important role in text recognition. "Robust Reading" competition is organized by the International Conference on Document Analysis and Recognition (ICDAR). ICDAR has released a series of datasets for the text detection task. ICDAR03 contains 507 natural scene images, ICDAR11 contains 229 training images and 255 testing images, ICDAR15 is composed of 1,670 images. Then the ICDAR17 dataset was created for multilingual text detection. They were composed of 7800 training images and 1800 testing images. ICDAR19 was also a multilingual text detection dataset with 10,000 images. The dataset is composed of textual images from 10 languages such as Arabic, English, French, Chinese, German, Korean, Japanese, Italian, Bangla and Hindi. Many researchers have achieved good detection results for most popular languages such as English. But text detection for Hindi remains unexplored. Thus, our work focuses on the Hindi language.

Hindi is India's national language which is the third most popular language in the world after English and Chinese. Hindi is most popularly written in Devanagari scripts. More than 300 million people use Devanagari script in the northern and central parts of India. The Devanagari script consists of 13 vowels, 34 consonants and 14 vowel modifiers. A whole letter is formed by connecting a modifier with consonants. Thus the shapes of composite characters are more complex than consonants. Hindi doesn't have uppercase or lowercase alphabets. The letters are generally written in left to the right direction. It also has a horizontal line on top of all characters. This line is called the header line. Hindi words have top, core and bottom strips as shown in Fig. 1.The top and core strips are separated by header line and the bottom strip is separated by a virtual line.

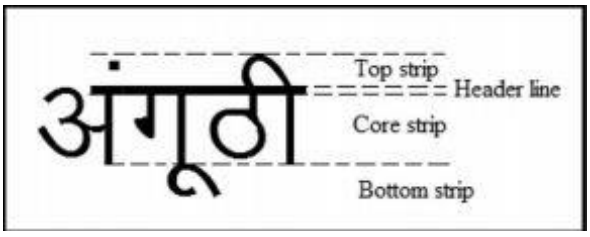

Fig -1: Strips of words in Devanagari script.[13]

It is observed in the literature that most text detection works for Hindi are carried out in document images using Optical Character Recognizer (OCR) [11]. Though OCR performs well for document images it cannot handle images with a complex background, arbitrary shaped text and the different size of the text. It also involves multiple processing steps, feature

extractions and filtering. This also requires much hard work in fine-tuning parameters which slow down the process of detection.

Inspired by the recent innovation in object detection, we propose our system to detect the text by predicting the bounding boxes with quadrilateral using deep neural networks. The proposed Hindi text detector algorithm is able to localize the text in the complex background images. The main contributions of the paper are as follows:

- ✓ Traditional methods fail to detect text of arbitrary shape in the complex background image. Thus we have used Deep Neural Networks to overcome this challenge.
- ✓ Our module is able to separate the text instances which are very close to each other.
- ✓ The proposed Hindi Text Detector is able to detect the stroke features of the text in the image efficiently and accurately.

2. RELATED WORKS

Scene text detection has become the most popular field of research in computer vision. Traditional methods manually extract features from scene images. These methods can be classified into sliding window-based methods and connected components based methods. Sliding window method detects text region by shifting the window in each position of the scene image [5]. Connected components based methods extract the text region first and then eliminate the non-textual noise using post-processing. The performance of these methods is limited in multi-oriented and low-resolution images.

Convolutional Neural Networks (CNNs) [9] gained its popularity in text detection. These networks achieved good performance accuracy and it was simple to implement. However, these methods have to classify a large number of the sliding window to get good performance. This results in high computational requirements. The YOLO is another popular convolutional neural network in object detection. It is a simple and efficient network that predicts multiple bounding boxes and class probabilities for the boxes. It directly finds the bounding boxes based on the feature maps of the images. Incremental learning method [3] used with deep convolutional neural networks increases the performance of the network by easing the training process.

Character-based text detection methods [8] work by finding the individual characters and then group them to form words. Extremal region proposed in [8] locates characters and then uses exhaustive search methods to group them. Word based text detection method [7] directly extracts words from the image which is similar to object detection methods. The candidate region with text was extracted and then a convolutional neural network is adopted to refine the bounding box regression. Text-line based text detection method detects the textual line and then extracts the words from it. In [16] MSER trees are applied to generate the text components from the image. Then a Deep Neural Network is used to classify the textual regions in order to detect the word.

Horizontal based text detection [17] tries to identify the horizontal text in the images. They generally use horizontal bounding boxes as their output. Some methods detect the horizontal text parts and merge them to form the word. Algorithms based on Adaboost [17] are proposed to detect the text region.

Multi-oriented text detection is more robust text detection when compared to horizontal text detection since these methods can detect text of arbitrary shapes. Several works have been carried out in this field. In [1], the author proposes two level classification scheme and two sets of features for text detection. One is the component level feature and the other one is the chain level feature. The components are extracted and analyzed to form candidate text regions.

DMPNet [19] uses quadrangle to detect text. Multiple quadrilateral sliding windows are applied to recall text. Sliding window with a higher overlapping threshold is used to judge positive or negative. The polygonal overlapping is computed using Monte-Carlo computational method. RRPN [12] has a Region Proposal algorithm which is used for object detection. It generates predefined anchor boxes which are mapped to the radar detection points. These points are then scaled to detect the actual object.

TextBoxes++ [10] uses text-box layers to detect words. It uses rotated rectangles to generate the arbitrarily oriented bounding boxes. The cascaded NMS is applied in the post-processing phase to accelerate the speed. Further, this module also combines a text recognition module.

EAST [18] adopts FCN with rotated rectangles to predict the text boxes. Non-maximal suppression is used to yield the final results. To merge the feature maps from different levels EAST uses the U-shape idea. Time complexity is reduced by a locality aware algorithm.

3. PROPOSED WORK

In this section, we first describe the overall pipeline of the proposed Hindi Text Detector as shown in Fig. 2.

3.1 Overall Architecture

The backbone architecture of our network is ResNet [3]. We adapt Region Proposal Network (RPN) with feature pyramid network to generate bounding boxes for our input image.

RPN uses a sliding window to generate region proposals. The sliding window is implemented over a 3 ×3 convolution layers which is replaced by FPN. The lower level features are concatenated to form higher-level features and thus we have three concatenated feature maps. Anchors of a single scale are assigned to each pyramid level. Each anchor is assigned with a label based on the Intersection-over-unions (IoU) with the ground truth bounding boxes. The IoU for the given ground truth bounding boxes is given positive label if the IoU is over 0.7. The IoU below 0.3 is given a negative label. Then we generate the bounding boxes for the positive label ground truth. Thus this produces an output image with bounding boxes in the textual region.

1.3 Implementation Details

We implemented our Hindi Text detector in ResNet50, ResNet101 and Resnet152 architectures. The ResNet we used is pre-trained on ImageNet dataset. We have used 800 images from IC19-MLT Dataset (Hindi) for training and 200 images from IC19-MLT Dataset (Hindi) for testing. The neural network is trained from scratch using the training images. During training, we ignored the blurred region labeled as DONT CARE. Bounding boxes are extracted by calculating the minimal rectangular area. All these experiments conducted in this paper are carried out in PC equipped with NVIDIA GPU. The whole training process took about 8 hours on IC19-MLT (Hindi) dataset which is the only standard dataset available for Hindi Text Detection.

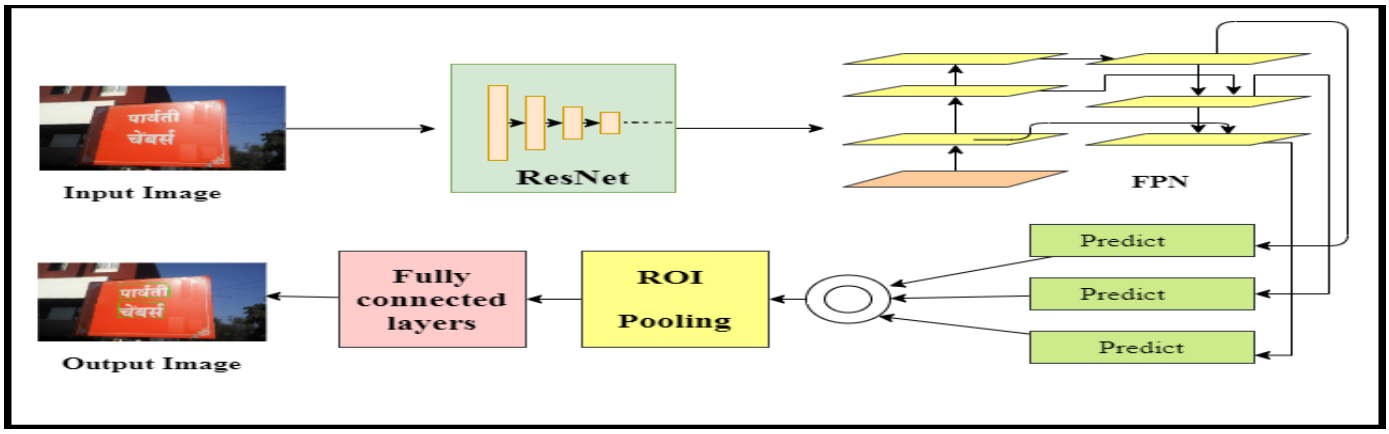

Fig -2 Architecture of Hindi Text Detector

1.2 Network Design

The basic network of Hindi Text Detector is inspired from FPN [15]. The pyramid takes an input of arbitrary size and produces output feature maps. The pyramid construction involves two ways such as bottom-up pathway and top-down pathway. Bottom-up pathway uses several feature maps to compute a feature hierarchy many layers produces an output of same size in the same network stage. Feature pyramid uses on pyramid for each stage. The output of the last layer of stage is chosen as the reference set of feature maps.

Feature activation is used at each stage of the residual block. The output of the last residual blocks is denoted as $\{C_1, C_2, C_3, C_4\}$. Top-down pathway builds higher resolution features by upsampling. The features from bottom-up pathways are used to enhance the high-resolution features. The lateral connection between the bottom-up pathway and the top-down pathway is used to merge the features between them. The upsampling is done iteratively until finest resolution map is obtained. $\{P_1, P_2, P_3, P_4\}$ are the final set of feature maps corresponding to $\{C_1, C_2, C_3, C_4\}$.

4. EXPERIMENTS AND RESULTS

This section describes the dataset used, evaluation measures and results.

A. Dataset

ICDAR19-MLT [2] is the largest multilingual dataset which has 10,000 images for scene text detection. It has images from 10 languages such as Arabic, English, French, Chinese, German, Korean, Japanese, Italian, Bangla and Hindi. It has 1000 scene text images for each language. Thus, for our work, we have used the Hindi scene text images.

B. Evaluation measures

The evaluation protocols for text detection rely on *precision* (P), *recall* (R), and *f-measure* (F). They are defined as:

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

$$F = 2 \times \frac{P \times R}{P + R}$$

(A)

Fig -3: Detection results in ICD19-MLT (Hindi dataset)

(B)

Fig -4: Detection results in ICD19-MLT (Hindi dataset)

Where TP, FP, and FN stands for number of hit boxes, wrongly identified boxes and missed boxes. A bounding box is considered as hit box when the Intersection over Union is larger than the given threshold.

C. Parameter setup

The following parameters were used for the network

Number of epochs = 100

Training batch size = 8

Testing batch size = 8

Activation function = ReLu

Cuda = True

Number of GPU = 1

Learning Rate = 0.01

D. Results

We have demonstrated the test examples of our Hindi Text Detector in Fig. 3 and Fig. 4 which the images from IC19-MLT dataset. From these examples, it is easily observed that our Hindi Text Detector is able to detect the textual regions without missing the strips and it is able to separate the text instances which are very close to each other. Hindi Text Detector is able to locate text which is smaller in size, arbitrary shapes and in different backgrounds.

Deeper the neural network better the performance of text detection. We have adapted ResNet as our backbone architecture. We implemented ResNet in three different architectures (50,101,152). Table 1 shows the accuracy obtained by different ResNet architectures. It clearly shows that the deeper neural network is able to achieve high accuracy from 74% to 78.5% with an improvement.

TABLE I

TEXT DETECTION RESULTS ON ICD19-MLT (HINDI) DATASET

Network	**Recall**	**Precision**	**F-measure**
ResNet-50	0.714	0.769	0.74
ResNet-101	0.734	0.795	0.762
ResNet-152	**0.753**	**0.82**	**0.785**

4. CONCLUSION

We have presented the Hindi Text detector, a convolution neural network for Hindi text detection. This is an efficient and stable network which is able to detect text in complex background images. The proposed method predicts the bounding boxes via quadrilateral representation. The proposed model has experimented with the benchmark dataset IC19-MLT (Hindi) and it is able to achieve best recall and F-measures. We have planned to expand our work to detect text in multiple Indian languages.

REFERENCES

[1] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, (2012). Detecting texts of arbitrary orientations in natural images, Proceedings of CVPR, 1083-1090.

[2] IC19-MLT Dataset link: https://rrc.cvc.uab.es/?ch=15

[3] J. Redmon, S. Divvala. R. Girshick, and A. Farhadi, (2016). You only look once: Unified, real-time object detection, Proceedings of CVPR, 779-788.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learing for image recognition, arXiv preprint, arXiv:1512.03385v1 .

[5] K. I. Kim, K.Jung, and J. H. Kim, (2003) Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm, IEEE Transaction on Pattern Analysis and Machine Intelligence, 1631-1639.

[6] K. S. Raghunandan, G. Hemantha Kumar, Umapada Pal, and Tong Lu, (2018). Multi-script-oriented text detection and recognition in video/scene/born digital images, IEEE Transactions on Circuits and Systems for Video Technology, 29, 1145 - 1162.

[7] L. Gomez and D. Karataz, (2017), TextProposals: A text-specific selective search algorithm for word spotting in the wild, Pattern Recognition, 28, 60-74.

[8] L. Neumann and J. Matas, (2012). Real-time scene text localization and recognition, Proceedings of CVPR, 3538-3545.

[9] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, (2016). Reading text in the wild with convolutional neural networks, International Journal of Computer Vision, 1-20.

[10] M. Liao, B. Shi and X. Bai, (2018). Textboxes++: A single-shot oriented scene detector, IEEE Transaction on Image Processing, 27, 3676-3690

[11] Parul Sahare and Sanjay B. Dhok, (2018) Multilingual character segmentation and recognition schemes for Indian Document Images, IEEE Access.

[12] Ramin Nabati and Hairong Qi, RRPN: Radar Region Proposal Network for Object Detection in Autonomous Vehicles, arXiv preprint, arXiv:1905.00526.

[13] R. Jayadevan, Satish R. Kolhe, Pradeep M. Patil, and Umapada Pal, (2011), offline recognition of Devanagari script: A survey, IEEE Transaction on system, man and cybernetics, 41,6, 786-796.

[14] S. Ren, K. He, R. Girshick, and J. sun, (2015). Faster R-CNN: Towards real-time object detection with region proposal networks, Proceedings of NIPS, 91-99.

[15] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie, Feature pyramid network for object detection, arXiv preprint, arXiv:1612.03144v2.

[16] W. Huang, Y. Qiao, and X. Tang, (2014). Robust scene text detection with convolution neural network induced MSER trees, Proceedings of ECCV, 497-511.

[17] X. Chen and A. L. Yuille, (2004). Detecting and reading text in natural scenes, Proceedings of CVPR, 366-373.

[18] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, Jiajun Liang, (2017), EAST: An Efficient and accurate Scene Text detector, arXiv preprint, arXiv:1704.03155v2.

[19] Y. Liu and L. Jin, (2016). Deep matching prior network: Towards tighter multi-oriented text detection, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 3454-3461.

[20] Z. Tian, W. Huang, T. He, P. He and Y. Qiao, (2016). Detecting text in natural image with connectionist text proposal network, Proceedings of ECCV, 56-72.