SCENE TEXT RECOGNITION

K VAMSI
M22RM002
IIT Jodhpur,India
m22rm002@iitj.ac.in

RAHUL MADAAN
M22RM008
IIT Jodhpur,India
m22rm008@iitj.ac.in

GAURAV CHOUDHARY
M22CS055
IIT Jodhpur,India
m22cs055@iitj.ac.in

Abstract

The field of scene text recognition has become increasingly popular in recent years, as recognizing text in natural scenes presents a significant challenge compared to scanned documents. This paper proposes a novel technique that utilizes first-order Histogram of Oriented Gradient (HOG) through a spatial pyramid,EAST,EASY OCR to address the issue of text recognition in photos of natural scenes and on the internet.Further More,Deep learning-based approaches, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have shown promising results in scene text recognition, and have been combined with various techniques to handle specific challenges, such as attention mechanisms, geometric transformations, and multi-scale processing. The importance of scene text recognition is further underscored by its applications in various domains, including autonomous driving, content-based image retrieval, and cultural heritage preservation.In This paper we have reviewed recent advances in scene text recognition, with a particular focus on deep learning-based methods and their applications.

keywords: EAST, CNN, RNN

1. Introduction

The text in an image is an important source of information in our daily life, which can be extracted and interpreted for different purposes.Scene text recognition has emerged as a challenging and important problem in computer vision, with a wide range of applications in various domains such as autonomous driving, content-based image retrieval, and cultural heritage preservation. Text in natural scene images usually contains rich and valuable semantic information.Thus robust reading of text in uncontrolled environments has become a crucial task that is essential to many computer vision applications.Reading scene text from scene images typically can be divided into two stages: text detection and text recognition.Different from traditional Optical Character Recognition that transcribes characters or words from scanned documents, scene text recognition is difficult due to a wide variety of factors,such as variability of font and color, distortion, occlusion, low resolution, cluttered background,etc. Furthermore, the problem we focus on is unconstrained text recognition,which is more challenging than constrained recognition.

Traditional methods of scene text recognition relied heavily on hand-crafted features and statistical models to recognize text in natural scenes. These methods often struggled with variations in font, size, orientation, and lighting conditions, which are common in natural scenes. To address these challenges, researchers developed a range of techniques such as stroke-width transform, MSER, and connected component analysis. These techniques were combined with various classifiers such as SVM, HMM, and CRF to recognize text in natural scenes.

In recent years, deep learning-based approaches, particularly CNNs and RNNs, have shown great promise in scene text recognition, and have been combined with various techniques to address specific challenges. However, despite the recent progress, scene text recognition remains a challenging task, and there is a need to develop more effective and efficient methods for recognizing text in natural scenes.Most prevalent approaches to date combine Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). In these methods, the CNN is regarded as a feature extractor or an encoder for images. The RNN, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit outputs character sequences for language modeling. Benefiting from the strong capacity of RNNs in modeling language sequences, these kinds of method achieve promising performance.

Here In his paper we have implemented some existing traditional and deep learning based scene text recognition algorithms.we have reviewed recent advances in scene text recognition, with a focus on deep learning-based methods and their applications.

2. Performance Metric

Word Error Rate (WER) is a performance metric used to evaluate the accuracy of speech recognition or OCR systems. It measures the percentage of words that are incorrectly recognized or transcribed, compared to the total number of words in the reference or ground truth text.

The WER calculation involves comparing the recognized or transcribed text to the ground truth text, and measuring the number of substitutions, deletions, and insertions needed to convert one into the other. The resulting WER value can range from 0 to 1, with lower values indicating better accuracy.While WER is commonly used to evaluate speech recognition systems, it can also be used for text recognition tasks such as OCR. A lower WER value indicates higher accuracy and better performance of the system.

3. Dataset

we have collected custom dataset from various enviornments.we have used custom dataset dataset for OCR and EAST Models.we have used ICDAR dataset for FOTS Model.ICDAR (International Conference on Document Analysis and Recognition) is a series of conferences organized for the past 27 years by the International Association of Pattern Recognition (IAPR) to bring together researchers, scientists, and practitioners in the field of document analysis and recognition.ICDAR datasets are widely used in the field of document analysis and recognition for research and benchmarking purposes. These datasets include images of documents, such as books, journals, newspapers, forms, and handwritten notes, and annotations such as text, layout, and structure. Some of the well-known ICDAR datasets include ICDAR 2011, ICDAR 2013, ICDAR 2015, and ICDAR 2019. These datasets have been widely used for tasks such as text detection, text recognition, layout analysis, and document image segmentation.

4. METHODS IMPLEMENTED

we have tried to implement some classical methods to recognize the text in the image.However,Over the past few years, deep learning-based methods have made significant progress in scene text recognition. These methods can be broadly classified into two categories: seq2seq-based and segmentation-based methods. In addition, there are also classification-based methods that are often overlooked.

Seq2seq-based methods are based on the encoder-decoder architecture and are designed to directly predict the text sequence from the input image. These methods typically use recurrent neural networks (RNNs) or transformer networks as the encoder and decoder. One of the most popular seq2seq-based methods is the attention-based encoder-decoder model, which uses a mechanism to selectively focus on relevant parts of the input image during decoding. Another popular approach is the fully convolutional sequence recognition model, which uses a (CNN) as the encoder and an RNN as the decoder.

Segmentation-based methods, on the other hand, first segment the text regions from the input image and then recognize the text within each segmented region. These methods typically use segmentation techniques such as the EAST and sliding window approach or the connectionist text proposal network (CTPN) to detect and segment text regions. The recognized text within each region is then obtained using a classification-based or seq2seq-based method.

Classification-based methods, which are often overlooked, directly classify each character or word in the input image without explicitly segmenting the text regions. These methods typically use CNNs or a combination of CNNs and RNNs to classify each character or word.

In addition to these three categories of methods, some popular modules are often used in scene text recognition, such as the histogram of oriented gradients (HOG) and the spatial transformer network (STN).

Overall, the progress in deep learning has led to significant advances in scene text recognition, with each category of methods having its strengths and limitations. The choice of method often depends on the specific application and the requirements for accuracy, speed, and robustness.

4.1. EAST

EAST (Efficient and Accurate Scene Text detection) is a deep learning-based text detection method that was proposed byinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang [2]. EAST uses a fully convolutional neural network (FCN) to detect text regions in natural scene images. The network is designed to predict the score map and the geometry map of the text regions simultaneously. The score map indicates the likelihood of a pixel being part of a text region, while the geometry map provides information about the shape and orientation of the text region. The architecture of the EAST network consists of a backbone network, a feature merger module, and two output layers for the score map and the geometry map. The backbone network is based on the VGG16 architecture and is used to extract features from the input image.

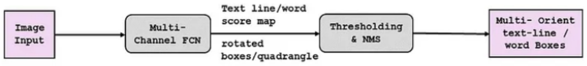


Image By Paritosh Mahto

Figure 1. EAST Model Framework

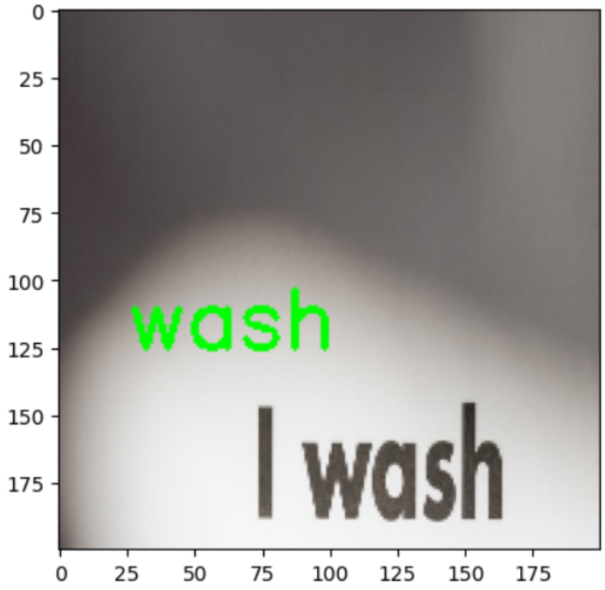


Figure 2. EAST Text Recognition Output Image

It is a fast and accurate scene text detection method and consists of two stages: 1. It uses a complete convolutional network (FCN) model to directly generate pixel-based word or text line predictions

2. After generating text predictions (Rotate a rectangle or quad) and the output is sent to the non-maximum suppression to produce the final result. The feature merger module is used to combine features from different scales of the backbone network to improve the detection accuracy.

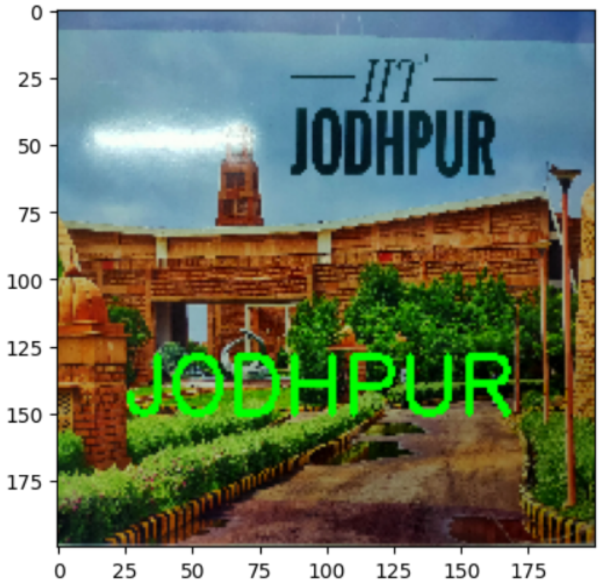


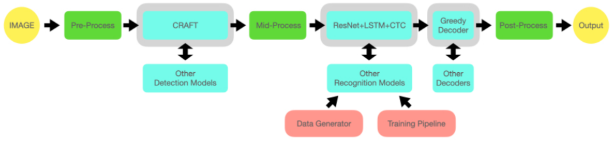
Figure 3. EAST Text Recognition Output Image

The output layers are convolutional layers that predict the score map and the geometry map. One of the key features of EAST is its efficiency. The network is designed to be computationally efficient, allowing it to process images in real-time on a CPU. This is achieved by using a fully convolutional architecture that avoids the need for computationally expensive operations such as fully connected layers.

To detect text regions using EAST, the input image is first resized to multiple scales and fed into the network. The output score map and geometry map are then used to predict the text regions. Non-maximum suppression (NMS) is used to eliminate overlapping text regions.The output of the EAST is shown below.Here In our implementation we have got quite good **WER=0.66** but not as expected,reason is insufficient data set to pass through the network.

4.2. EASY OCR

Easy OCR is based on deep learning algorithms and pre-trained models that can recognize text in various languages, fonts, and styles.EasyOCR works by first performing image preprocessing to enhance the image quality, such as de-skewing, binarization, and noise reduction. Then, the image is fed into a pre-trained neural network that is capable of recognizing text from the image. The neural network outputs the recognized text along with a confidence score for each character.EasyOCR uses a combination of convolutional neural networks (CNN) and recurrent neural networks (RNN) to recognize text from images.EasyOCR uses a multilingual recognition system that supports more than 70 languages, including Arabic, Chinese, English, French, German, Hindi, Japanese, Korean, Russian, Spanish, and many more.



Source: Github

Figure 4. EASY OCR Model Frame Work

steps in easy ocr:

1.In Easy Ocr we defined a function that takes in the OCR result and the path of the image and returns the image with the recognized text highlighted.The function first reads the image using the OpenCV function.

2.It then extracts the top-left and bottom-right coordinates of the bounding box of the recognized text and the text itself from the OCR result.

3.It then defines a font to use for the text and draws a rectangle around the bounding box using the OpenCV rectangle() function. It then puts the recognized text inside the

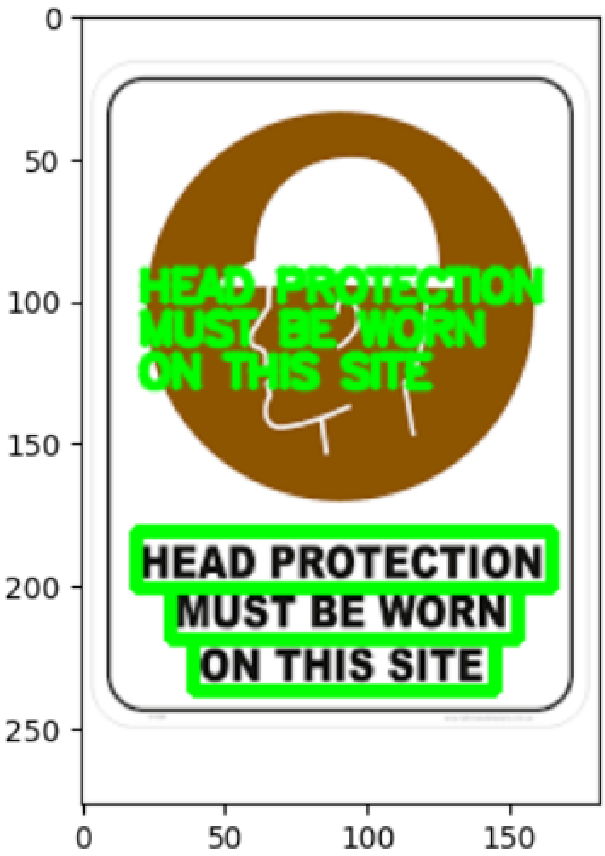


Figure 5. EASY OCR Text Recognition OUPUT Image

bounding box using the putText() function.Finally, it returns the image with the highlighted text.EASY OCR was able to predict the text with an **accuracy** of around **0.97**.

4.3. FOTS

FOTS is an end-to-end trainable framework that detects and recognizes all words in a natural scene image simultaneously. The overall architecture of FOTS consists of four parts: 1.Shared Convolutions 2.Text Detection Branch 3.RoI Rotate 4.Text Recognition Branch The input image is fed into the shared convolutions network and shared features are extracted.The shared features are passed as input to the text detection branch, which detects and localize the text with bounding box around the text in the image.The shared features and text detection branch predicted bounding boxes are fed into the RoI rotate operator to extract the axis aligned text proposal features.

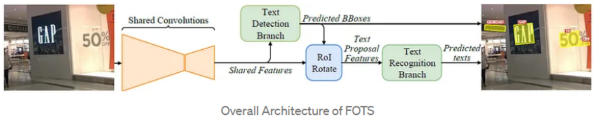


Figure 6. FOTS Model Frame Work

Extracted axis aligned text proposal features are then fed to the text recognition branch which contains Recurrent Neural Network(RNN) encoder and Connectionist Temporal Classification(CTC) decoder to recognize the text inside the bounding box.

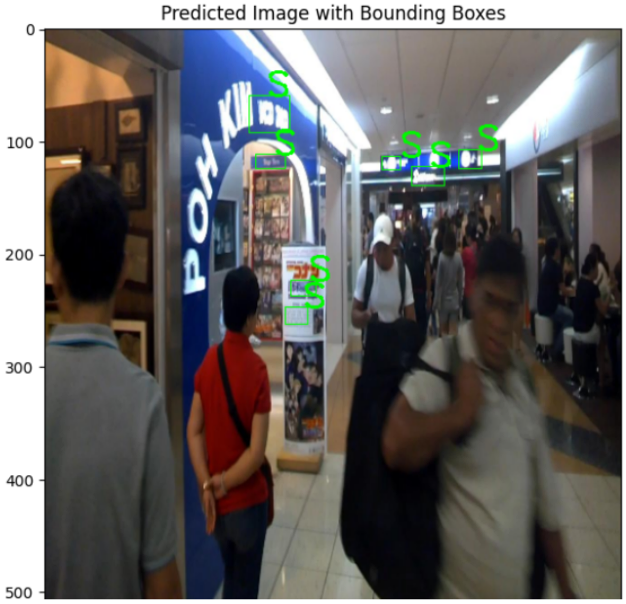


Figure 7. FOTS Model Recognition output

Given any input image to FOTS text spotting system, the final inference pipeline work as follows

- 512 x 512 x 3 reshaped input image is fed into the detector model which will predict the text pixels and localize them in the image. In other words detectors returns the score map geo map of the input image.
- Using thresholding NMS on these maps, the bounding box with maximum IoU is obtained.
- ROIRotate takes the input from the predicted bounding box and transform it to axis parallel box .
- The text region inside the bounding box is reshaped to 64 x 128 x 3 and fed into the recognition model to predict the text label.

Output of text detection branch and text recognition branch are merged to the input image to display the final prediction.Here we correctly identified the text but we were not able to extract the text exactly as shown in the figure below.

4.4. Important Links

Collab Links:

1.Here is a link to –EAST-EASY-OCR.
2.Here is a link to –FOTS-DATA-PRE.
3.Here is a link to –FOTS-DETECTION-MODEL.

4.Here is a link to –FOTS-END-MODEL.
Youtube Link:
Here is a link to –You-Tube
Github Link:
Here is a link to –Git-Hub

5. FUTURE WORK

In future we will try to try to implement Scene text recognition methods, such as thresholding, edge detection, and feature extraction, as well as deep learning-based approaches, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) and decoder-encoder methods. These methods can be combined to create hybrid models that leverage the strengths of both traditional and deep learning methods. Additionally, we Will try to implement various pre-processing steps, such as image normalization and text localization, which can also be applied to improve the accuracy of the recognition process.we will try implement this methods for other languages as well along with English.

References

[1] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. Fots: Fast oriented text spotting with a unified network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[2] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. Fots: Fast oriented text spotting with a unified network, 2018.

[3] Cong Yao Shangbang Long1, Xin He. Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision (2021)*, 2021.

[4] Zhi Rong Tan, Shangxuan Tian, and Chew Lim Tan. Using pyramid of histogram of oriented gradients on natural scene text recognition. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2629–2633. IEEE, 2014.

[5] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: An efficient and accurate scene text detector, 2017.

[3] [5] [4] [1] [2]