Proceedings of 8th IOE Graduate Conference
Peer Reviewed
ISSN: 2350-8914 (Online), 2350-8906 (Print)
Year: 2020 Month: June Volume: 8

A Novel Approach for the Scene Text Recognition by Attentional Encoder Decoder Model

Aastha Pandey [a], Dibakar Raj Pant [b]

[a, b] *Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, TU, Nepal*
Corresponding Email: [a] 074msice001.aastha@pcampus.edu.np, [b] drpant@ioe.edu.np

Abstract
Texts in scene image contain rich and precise semantic information which helps in analyzing and understanding the corresponding environment. Text localization and recognition are two fundamental tasks for scene text recognition. Text localization is determining the position of text from input image with the position represented by a bounding box and text recognition is converting image regions containing text into machine-readable strings. There are variations in background, illumination, layout, size, font and color of texts in a scene image and presence of many patterns similar to characters which make the task challenging. The different deep learning methods have tradeoff between speed, accuracy and complexity. In this work, modified VGG16 is used along with You Only Look Once (YOLO) algorithm for text localization. Attention model with Convolutional Neural Network (CNN) is used in encoder part and attentional Long Short Term Memory (LSTM) is used in decoder part for reduction in the disturbance of background noise in scene text recognition. The images are first input to localization model which outputs the localized text regions and predicted bounding boxes. Next, the localized regions are fed to the recognition model. The final outputs obtained are the predicted texts of the scene image which are word level recognitions. Precision of 84.62%, recall of 78.91% and F1 score of 81.66% are obtained in text localization on ICDAR2015. Word recognition accuracies of 91.6% on ICDAR 2013 data and 88.7% on SVT data are obtained.

Keywords
Text localization, text recognition, YOLO, VGG16, encoder decoder, attention

1. Introduction

Texts in scene images provide information complementary to visual cues and convey rich semantic information. Recognizing texts in images is important for numerous applications such as image search [1], video retrieval, instant translation and industrial automation [2]. Scene text localization and recognition is an important research topic in computer vision. Scene text localization is the process of predicting the presence of text and localizing each instance of text present in natural scenes. Scene text recognition is the process of converting text regions into computer readable symbols. Several research have been done in the field of Optical Character Recognition (OCR). However, recognizing texts from natural images is still a challenging task due to the fact that scene text has high variation in character color, illumination, background and layouts. Also, there are many patterns similar to characters which can be falsely identified by feature extractor increasing the error rate of the model.

A model with good feature extracting ability and background noise suppression is required for scene text recognition. Before the deep learning era, there was requirement for repetitive pre-processing and post-processing steps to extract low-level or mid-level hand crafted image features due to which those methods had limited representation ability of handcrafted features and increased complexity of pipelines. The deep learning methods have aided in improvements in scene text recognition.

2. Related work

Early text detection and recognition research was an extension of document analysis and recognition research focusing on basic preprocessing, detection and OCR technology [3]. Scene text recognition is more challenging than OCR due to variation in

background, fonts, color so OCR methods do not perform well for scene text recognition. After deep learning era, there was the advantage of automatic feature learning and improved performance as compared to traditional methods with hand-crafted features. Wang et al. in [4] combined a multi-layer CNN with unsupervised feature learning to train character models, which were used in both text detection and recognition procedures. They used CNN based sliding window character classification and integrated the character responses with character spacings and a defined lexicon using a beam search algorithm to recognize words. Huang et. al. in [5] used Maximally Stable Extremal Regions (MSER) and Support Vector Machine (SVM) for text detection and recognition in scenes with the recognition problem splitted into detection and recognition procedure. Compared to sliding window methods, connected component methods proved more efficient and robust.

Several hybrid methods were used to make use of the advantages of different approaches. Huang et al. in [6] applied CNN to learn high-level features from the MSER components in image. Those components showed high discriminant ability and robustness against background noises. Sliding window method and NMS were incorporated in the CNN classifier to handle the problem of multiple characters connection. It was evaluated on the ICDAR 2011 dataset, and achieved over 78% in F-measure which was higher than previous methods. Qin et al. in [7] proposed a text detector based on the cascade of two CNNs. Text regions of interest were first produced by a FCN and then resized to a square shape with fixed size. In the next stage, a YOLO-like network was trained to generate rectangular bounding boxes for all words and finally NMS was used for removing overlapping bounding boxes.

In case of text recognition, character segmentation is considered the most challenging part due to the complex background and irregular arrangement of scene text and largely constrains the performance of the whole recognition system. Two major techniques adopted to avoid segmentation of characters are Connectionist Temporal Classification [8] and attention mechanism.

Gao et al. in [9] adopted the stacked convolutional layers to effectively capture the contextual dependencies of the input sequence which was characterized by lower computational complexity and easier parallel computation. They used CTC for sequence decoding. The method suppressed noise and increased performance. Lee et al. in [10] used a recursive recurrent neural network with attention modeling for lexicon-free scene text recognition. The model first passed input images through recursive convolutional layers to extract encoded image features then decoded them to output characters by recurrent neural networks. Attention-based mechanism performed soft feature selection for better image feature usage. In [11], Liu et al. proposed an efficient attention-based encoder-decoder model where the encoder part is trained under binary constraints to reduce computation. Shi et al. in [12], [13] proposed text recognition system which combined a Spatial Transformer Network (STN) [14] and an attention-based sequence recognition network. Shi et. al. used a thin-plate spline transformation to rectify the input irregular text image into a more canonical form. Zuo et. al. in [15], used an integrated module of the CTC and attention mechanism to decode and output the text sequence in an encoder-decoder based framework for scene text recognition to improve scene text recognition combining advantage of both CTC and attention.

YOLO is a faster object detector but it along is not suitable for text detection. VGG16 is slow to train but it has better accuracy and is easy to implement. Therefore, VGG16 is used with YOLO algorithm for feature extraction in text localization utilizing advantages of both YOLO and VGG16. In case of text recognition, attention is used in both encoder and decoder part in encoder-decoder framework to improve recognition than existing methods especially in images with background noise.

3. Methodology

3.1 Text localization

3.1.1 YOLO algorithm

YOLO is a single staged detector for object detection using only one forward propagation pass through a single convolutional neural network to the entire image. YOLO is extremely fast at testing because it requires only a single network evaluation. In YOLO algorithm, the input image is divided into an SxS grid. If the center of an object falls into a grid cell, then the grid cell is responsible for detecting that object. Each grid cell predicts bounding boxes and confidence scores for those boxes. If no object exists in that cell,

the confidence scores will be zero else the confidence score will be equal to the intersection over union (IoU) between the predicted box and the ground truth. Each bounding box consists of 5 predictions: x, y, w, h, and confidence score. The (x, y) coordinates represent the center of the box relative to the bounds of the grid cell. The width and height are predicted relative to the whole image. Finally, the confidence prediction represents the IoU between the predicted box and the ground truth box [16]. Non Maximum Suppression (NMS) takes list of proposal boxes, corresponding confidence scores, overlap threshold and outputs list of filtered proposals.

3.1.2 VGG16 network

VGG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" and was used in ImageNet Large Scale Visual Recognition (ILSVR) competition in 2014 [17] . VGG16 has convolution layers of 3x3 filter with a stride 1 and uses same padding and maxpool layers of 2x2 filter of stride 2. In the end, it has 2 fully connected layers followed by a softmax for output.

3.1.3 Implementation

YOLO with VGG16 is used for text localization. Input image is resized to 512x512 and grid size of 16x16 is used. Class is set to 1 because the model detects only text and there is no need of classification to various classes as in case of object detection. The last three fully connected layers of VGG16 have been removed and three convolutional layers are added to the last layer of the architecture.The text localization process is shown in Figure 1.

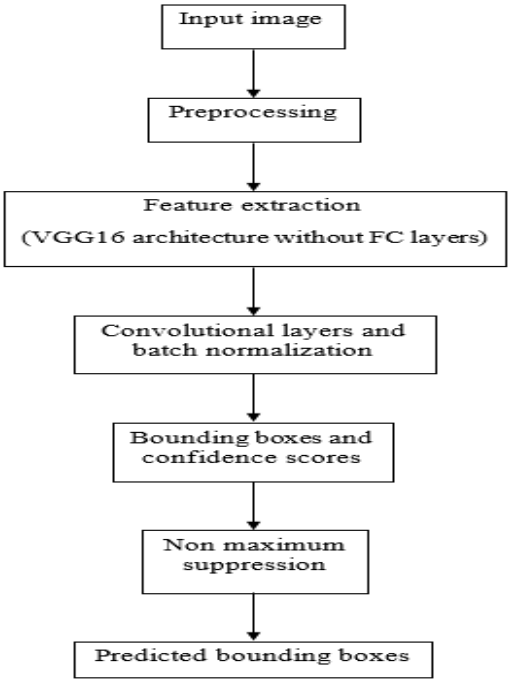

Figure 1: Text localization

3.2 Text recognition

3.2.1 Encoder-decoder framework

Encoder-decoder framework consists of a CNN to generate an ordered feature sequence from the entire word image. Then, the generated feature sequence is feature-coded using the bidirectional long short-term memory (Bi-LSTM) network. Next, LSTM is used along with attention mechanism to decode and output the text sequence.

3.2.2 Implementation

The input images to recognition model are cropped images with texts. Image is resized to 32x100. Then, the image is input into convolutional neural network to obtain feature maps. The CNN consists of series of convolution and maxpooling operations. After convolution network, attention network is used then maxpooling is done and fed to Bi-LSTM. The text recognition process is shown in Figure 2.

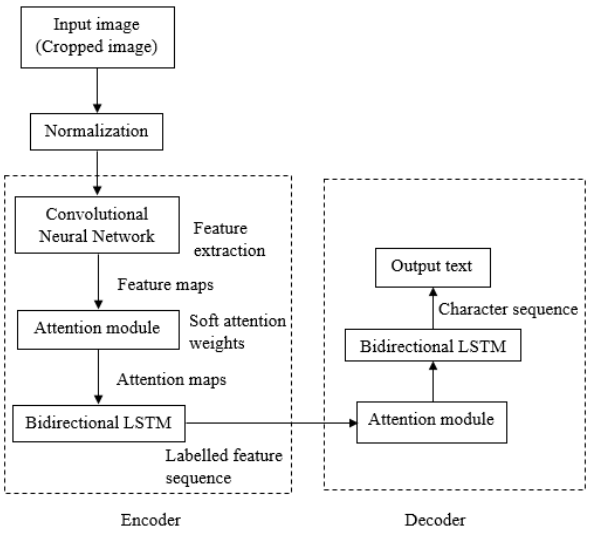

Figure 2: Text recognition

The attention block with down-sampling units, up-sampling units and sigmoid function is used. The

down-sampling unit contains a max pooling layer and a convolutional layer. The up-sampling unit contains a bilinear interpolation layer and a convolutional layer. The values are normalized by a sigmoid function and the attention maps are fused on corresponding feature maps with element-wise product [18]. The attentional CNN is shown in Figure 3.

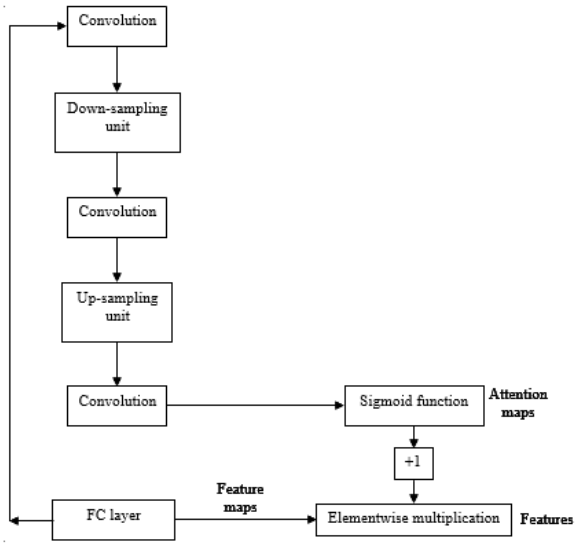

Figure 3: Attention model with CNN

Then, the feature map is converted into a feature sequence using BiLSTM network. It analyzes the feature sequence bidirectionally capturing long-range dependencies in both directions and outputs a new feature sequence.

The decoder is based on the attentional sequence-to-sequence model. At a time-step t, the decoder makes prediction based on the encoder output, internal state and symbol predicted at the last step t-1. The decoder computes a vector of attentional weights which indicates the importance of every item of the encoder output vector (H).

$$e_{t,i} = w^T tanh(W s_{t-1} + V h_i + b) \quad (1)$$

$$\alpha_{t,i} = exp(e_{t,i}) / \sum_{i'=1}^{n} exp(e_{t,i'}) \quad (2)$$

where s_{t-1} is internal state at t-1, y_{t-1} is symbol predicted in t-1, b is bias weight and α_t is vector of attentional weights; w,W,V are trainable weights. The decoder linearly combines the columns of encoder output into a glimpse vector (G).

$$g_t = \sum_{i=1}^{n} \alpha_{t,i} h_i \quad (3)$$

The glimpse vector, one hot embedding of y_{t-1} and s_{t-1} are then taken as an input to the LSTM cell of the decoder which produces an output vector and a new state vector. The output state is used to predict the current step symbol. The symbol predicted at the last step is incorporated in the computation at decoder so the decoder learns to capture the dependencies between its output characters. Then, beam search approach is used at every step.

4. Results

ICDAR 2013 and ICDAR 2015 datasets are used for training and testing of text localization which consists of horizontal English texts both focused and incidental. ICDAR 2013 and SVT datasets are used for training and testing of text recognition. ICDAR 2013 and SVT data are horizontal English texts. SVT consists of images with low resolution. The total number of images used for text localization is 1962 and the total number of images used for text recognition is 2590. Bi-LSTM network with 256 hidden units is used. The number of attention units used in decoder are 256. Sigmoid function is used in attentional CNN and Tanh function is used in attention in decoder. Beam search with k=5 is used. Training is done in google colab GPU using python with tensorflow and keras libraries. Adam optimizer is used with learning rate of 0.001 and cross-entropy loss is used. The relation of accuracy and loss with respect to number of epochs is shown in the plots in Figure 4 and Figure 5 respectively.

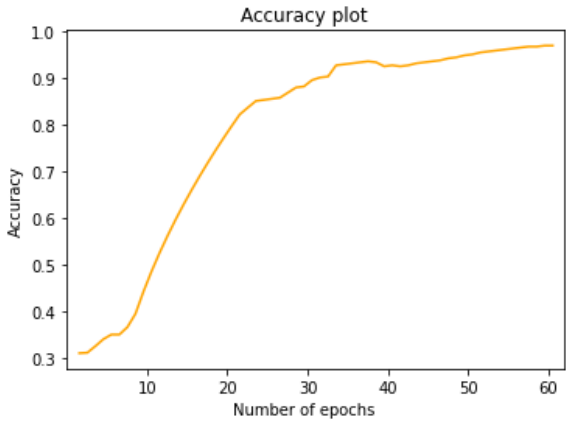

Figure 4: Accuracy plot

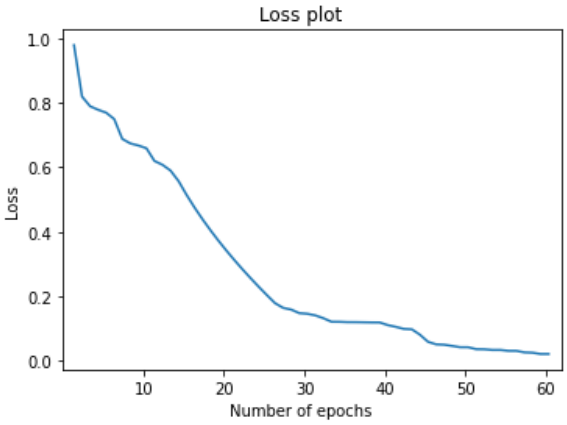

Figure 5: Loss plot

ICDAR evaluation protocol is used for evaluation of text localization method. In this evaluation method, predicted bounding boxes for text instances are matched to the ground truth boxes using IoU score. IoU threshold of 0.5 is used for the matching. Precision is calculated as the ratio of summation of the matched instances to predicted boxes. Recall is calculated as the ratio of summation of the matched instances to ground truth boxes. F1 score is given by the harmonic mean of precision and recall. For the evaluation of recognition method, word recognition accuracy is used which is the ratio of the correctly recognized words to the ground truths.

Figure 6: Text localization results

The text localization model is able to predict bounding boxes for texts region in scene image. The model detects text in word level as shown in Figure 6. Precision of 84.62%, recall of 78.91% and F1 score of 81.66% are obtained in text localization on ICDAR2015 data. The text recognition model takes the image with localized text region as input and produces the computer readable texts as output. The recognition results for different data are shown in Figure 7. Word recognition accuracies of 91.6% on ICDAR 2013 data and 88.7% on SVT data are obtained.The combined result is shown in Figure 8.

Figure 7: Text recognition results

Figure 8: Combined result

5. Conclusion

In this article, VGG16 with YOLO is used for text localization and attentional encoder-decoder framework is used for text recognition. The results obtained are good except for some complex characters and smaller texts in indistinguishable background. Use of attention in encoder part refined feature maps and provided robust feature extraction. This resulted in good text recognition in different data with different illumination, blurring and background. Precision of 84.62%, recall of 78.91% and F1 score of 81.66% are obtained in text localization on ICDAR2015. On ICDAR 2013, 91.6% word recognition accuracy is obtained and on SVT, 88.7% accuracy is obtained.

Acknowledgments

The authors are grateful to Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, TU for all the support and guidance in this research work.

References

[1] G. Schroth, S. Hilsenbeck, R. Huitl, F. Schweiger, and E. Steinbach. Exploiting text-related features for content-based image retrieval. In *2011 IEEE International Symposium on Multimedia*, pages 77–84, 2011.

[2] M. A. Chowdhury and K. Deb. Article: Extracting and segmenting container name from container images. *International Journal of Computer Applications*, 74(19):18–22, July 2013.

[3] A. Vinciarelli. Article: A survey on off-line word recognition. *Pattern Recognition*, 35(7):1433–1446, 2002.

[4] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng. End-to-end text recognition with convolution neural networks. In *IEEE International Conference on Pattern Recognition*, page 3304–3308, 2012.

[5] X. Huang, T. Shen, R. Wang, and C. Gao. Text detection and recognition in natural scene images. In *International Conference on Estimation, Detection and Information Fusion (ICEDIF)*, pages 44–49, 2015.

[6] W. Huang, Y. Qiao, and X. Tang. Robust scene text detection with convolution neural network induced mser trees. In *European Conference on Computer Vision*, pages 497–511, 2014.

[7] S. Qin and R. Manduchi. Cascaded segmentation-detection networks for word-level text spotting. In *International conference on document analysis and recognition*, 2017.

[8] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *23rd international conference on Machine learning*, page 369–376, 2006.

[9] Y. Gao, Y. Chen, J. Wang, and H. Lu. Reading scene text with attention convolutional sequence modeling, 2017.

[10] C. Lee and S. Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 2231–2239, 2016.

[11] Z. Liu, Y. Li, F. Ren, H. Yu, and W. Goh. Squeezedtext: A real-time scene text recognition by binary convolutional encoder-decoder network. In *AAAI*, 2018.

[12] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai. Robust scene text recognition with automatic rectification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 4168–4176, 2016.

[13] B. Shi, M. Yang, X. Wang, P. Lyu, X. Bai, and C. Yao. Aster: An attentional scene text recognizer with flexible rectification. In *IEEE transactions on pattern analysis and machine intelligence*, page 855–868, 2018.

[14] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial transformer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2017–2025. Curran Associates, Inc., 2015.

[15] L. Zuo, H. Sun, Q. Mao, R. Qi, and R. Jia. Natural scene text recognition based on encoder-decoder framework. *IEEE Access*, 7:62616–62623, 2019.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection, 2015.

[17] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[18] Y. Gao, Y. Chen, J. Wang, M. Tang, and H. Lu. Dense chained attention network for scene text recognition. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 679–683, 2018.