LETTER

Improving Faster R-CNN Framework for Multiscale Chinese Character Detection and Localization

Minseong KIM¹, Nonmember and Hyun-Chul CHOI¹ a), Member

SUMMARY
Faster R-CNN uses a region proposal network which consists of a single scale convolution filter and fully connected networks to localize detected regions. However, using a single scale filter is not enough to detect full regions of characters. In this letter, we propose a simple but effective way, i.e., utilizing variously sized convolution filters, to accurately detect Chinese characters of multiple scales in documents. We experimentally verified that our method improved IoU by 4% and detection rate by 3% than the previous single scale Faster R-CNN method.

key words: Chinese character localization, multiscale object detection, deep learning

1. Introduction

Recently developed detection and recognition techniques based on deep neural networks are very useful in digitalizing handwritten documents. As the first step of document digitalization, the task of character detection and localization is very important for successful further procedure of character recognition and translation. Detecting and localizing characters of various sizes, writing styles, and severely bad conditions is a challenging task. Especially for Chinese characters, they consist of several components that can be misunderstood as independent characters and, because of this, we need to localize the accurate regions of Chinese characters.

The previously used state-of-the-art detector, Faster R-CNN [1], consists of a feature extractor based on convolutional neural network (CNN) and a localizer based on region proposal network (RPN). VGG [2] or Resnet [3] which is pre-trained with ImageNet dataset [4] is commonly used as the feature extractor. RPN consists of a 3×3 convolution layer, a network for confidence score, and another network for bounding box prediction.

However, RPN with a 3×3 single convolution layer in Faster R-CNN has a limitation in localizing variously sized characters. For instance, the previous RPN looks only some parts of the character when localizing a character larger than 3×3 convolution filter as shown in Fig. 1 (a) and RPN looks background as well as the character region when localizing a small character as shown in Fig. 1 (b). This makes RPN localize the inaccurate region of the character. Such an inaccurate localization as omitting a part of the Chinese character or including a large background region may cause severe performance drop in the further procedure of character recognition and document translation. So, we need to improve the previous method to find accurate bounding boxes of Chinese characters.

In this letter, we propose a simple but effective way, i.e., using multiple variously sized convolution filters as shown in Fig. 2, to improve the character detection and localization performance of the previous Faster R-CNN framework. Whole character feature in case of Fig. 1 (a) can be included by adding our large convolution filters and background features in case of Fig. 1 (b) can be excluded by adding our small convolution filters.

Previously, two methods [5], [6] were proposed to deal
with multi-scale object detection. FPN [5] gradually accumulated up-sampled higher level features into lower level features by using skip-connection and concatenation. Then, the concatenated feature of each level went through a single detector/classifier to get object candidates. DSOD [6], to the opposite direction, gradually accumulated down-sampled lower level features into higher level features by using skip-connection and concatenation. These two methods utilized multi-level feature to cover multi-scaled objects but they need to repeat detector/classifier operations for each level feature and have to deal with a large number of candidates from multiple level features in its non-maximum suppression process.

In contrast, our method uses the single highest level feature map without complex skip-connect operations but applies simple multi-convolution kernels of different sizes to the feature map for multi-scale feature. This makes our method use features of the same abstraction level for multi-scale objects and fast multi-scale dealing without increasing in number of candidates.

In the remained of this letter, we will firstly explain the details of our method, secondly experimentally verify its effectiveness, and finally conclude this work.

2. Improved Faster R-CNN with Multiple Convolution Filters

In this section, we briefly introduce the previous Faster R-CNN [1] and describe the details of our Chinese character detector based on it.

The previous Faster R-CNN is designed to output the coordinates of object bounding boxes which include center position \((x, y)\), width \(w\), and height \(h\), and the confidence score of the detected objects as probability values \(p \in [0, 1]\) by using its region proposal network (RPN) [1], [7]. For RPN training and feature extractor fine-tuning, two objective functions, i.e., coordinate loss and score loss [1], were used. Coordinate loss is the normalized L2 distance (Eq. (1)) between the predicted bounding box \((x_t, y_t, w_t, h_t)\) and the ground truth box \((x_i, y_i, w_i, h_i)\).

\[
L_{\text{coordinate}} = \left(\frac{x - x_t}{w_t}\right)^2 + \left(\frac{y - y_t}{h_t}\right)^2 + \left(\frac{w - w_t}{w_t}\right)^2 + \left(\frac{h - h_t}{h_t}\right)^2.
\] (1)

Score loss is the binary cross entropy (Eq. (2)) between the predicted confidence score \(p_t\) and the ground truth value \(p_i \in 0, 1\).

\[
L_{\text{score}} = -(p_i \cdot \log(p) + (1 - p_i) \cdot \log(1 - p)).
\] (2)

Using these two losses, the RPN was trained to minimize \(L_{\text{total}}\) (Eq. (3)), the weighted summation of \(L_{\text{coordinate}}\) and \(L_{\text{score}}\) with the weights \(\lambda_{\text{coordinate}}\) and \(\lambda_{\text{score}}\).

\[
L_{\text{total}} = L_{\text{coordinate}} \cdot \lambda_{\text{coordinate}} + L_{\text{score}} \cdot \lambda_{\text{score}}.
\] (3)

However, in the previous Faster R-CNN framework, only \(3 \times 3\) convolution filter (corresponding to \(48 \times 48\) pixels on image) among the variously sized filters in Fig. 2 was used to make the input of RPN from the output of the feature extractor, where each pixel of feature map is corresponding to \(16 \times 16\) pixels on image due to downsampling procedure of four pooling layers in feature extractor. This single sized filter can cover only parts of a large character or include backgrounds around a small character as shown in Fig. 1. This may result in a performance drop of the Faster R-CNN. For an improved detection and localization performance, we add three more convolution filters of different sizes, i.e., \(1 \times 1, 5 \times 5,\) and \(7 \times 7\), and concatenate the output responses of four convolution filters before injecting it into RPN as shown in Fig. 2. This modification utilizes features corresponding to variously sized regions from \(16 \times 16\) pixels (\(1 \times 1\) filter) to \(112 \times 112\) pixels (\(7 \times 7\) filter) in an input image. The number and sizes of the filters are experimentally decided and will be described in Experiments section. By using multiple convolution filters of various sizes, RPN can output bounding boxes and confidence scores based on both parts and whole regions of variously sized objects with or without background regions.

3. Experiments

For Faster R-CNN structure, we used Resnet34 [3] pretrained with Imagenet dataset [4] as the feature extractor (Fig. 2), where it is known for higher classification performance [7] and less parameters than VGG [2]. When training RPN, this feature extractor was also finely tuned for the task of Chinese character detection and localization. In loss calculation of Eq. (3), the experimentally determined weights, \(\lambda_{\text{coordinate}} = 1\) and \(\lambda_{\text{score}} = 1\), were used. For the other settings, we followed the previous Faster R-CNN framework [1]. All experiments were operated with Pytorch v0.4.0, CUDA v10.0, cuDNN 7.6.2 on NVIDIA GTX 1080 Ti device.

At first, as shown in Fig. 3, we synthesized Chinese documents from CASIA handwritten text dataset [8] and 100 background images scanned from real Chinese documents to train and test networks. The CASIA dataset and background images were divided into two groups without overlap, 80% for synthesizing training images and the other 20% for test images. When synthesizing documents, we considered possible variations in character size, brightness, number of characters in a document, and overlap of characters. For character size, we randomly sized each character to have its short length between 16 pixels (corresponding to \(1 \times 1\) filter) and 112 pixels (corresponding to \(7 \times 7\) filter). For character brightness, we randomly selected a value between 0.5 and 1.1 per character and multiplied the weight to the pixel values of its corresponding character. We randomly selected a number between 10 and 160 and put the number of characters in a document. And we made overlap between characters by the randomly selected intersection over union (IoU) up to 0.05. For augmentation of a small number of
background images, we randomly resized the shorter length of each image between 1000 pixels and 1400 pixels maintaining its aspect ratio, randomly cropped its 80% area, and horizontally flipped it or not. Finally, we synthesized 20,000 documents for training and 500 documents for testing as the way above mentioned. In addition, to measure the performance for each character size, we made additional 100 documents of $N \times N$ pixel-sized characters for $N = 16 \times K$, $K = 1..7$.

As the first experiment, we tested multiple convolution filters of our method on 100 documents of each sized characters to find how the filters independently affect the performance of Faster R-CNN and how many filters are effective in improving its detection performance. To exclude the effect of fine-tuned features, we used Resnet34 pretrained with ImageNet dataset\textsuperscript{[4]} itself without fine-tuning and trained only RPN. Table 1 presents averaged true positive rates of RPN with $K \times K$ convolution filter(s). RPN with a single $3 \times 3$ filter (a) achieved the lowest performance (0.57) and achieved 9\% improvement (0.66) with an additional $1 \times 1$ filter (b). With additional larger filters (c,d), the detection performance remained or slightly increased. Based on this result, it is verified that using multiple filters is effective in improving the detection performance of RPN even without fine-tuning of feature extractor. And for further experiments, we used convolution filters up to $7 \times 7$ at which true positive rate was saturated.

As the second experiment, we compared the localization performances of three different methods, i.e., Faster R-CNN with a single-scale convolution filter (Single scale RPN)\textsuperscript{[1]}, with an additional regressor network (Single scale RPN + Regressor), and with multiple convolution filters (Ours). The networks of the three methods were trained with the 20,000 training documents and tested on the 500 test documents. Resnet34 of these networks were also finely tuned when training RPN with the total loss of Eq. (3). Here, ‘Single scale RPN + Regressor’ has an additional regressor network which refines the detection result of the previous Faster R-CNN (Single scale RPN). This regressor outputs the difference ($\Delta x, \Delta y, \Delta w, \Delta h$) between a bounding box of RPN ($x, y, w, h$) and its ground truth box ($x_t, y_t, w_t, h_t$) for a given input Resnet34 feature inside the bounding box of RPN. This regressor network was trained to minimize IoU loss\textsuperscript{[9]} of Eq. (4) between updated bounding box ($x_r = x + \Delta x, y_r = y + \Delta y, w_r = w + \Delta w, h_r = h + \Delta h$) and its ground truth box.

$$L_{IoU} = 1 - \frac{w' \cdot h'}{w_r \cdot h_r + w_t \cdot h_t - w' \cdot h'},$$

$$w' = \min(x_t + w_t, x_r + w_r) - x',$$

$$h' = \min(y_t + h_t, y_r + h_r) - y',$$

$$x' = \max(x_t, x_r), y' = \max(y_t, y_r).$$

As the localization performance indicator, we measured IoU between the ground truth boxes of Chinese characters and the detected bounding boxes of the aforementioned three different methods. Figure 4 shows IoU cumulative density function (CDF) of the bounding boxes with confidence score $p \geq 0.5$, where $p = 0.5$ is the minimum threshold value for assuming a detected bounding box as a Chinese character. As CDF is getting close to bottom right, IoU performance is also getting better because the number of bounding boxes with higher IoU increases. As shown in Fig. 4, CDFs of the three methods gradually increase to 1.0 but Single scale RPN (blue line) steepest increases at around $IoU = 0.8$ and is almost saturated at $IoU = 0.9$. Adding a regressor to Single scale RPN (Single scale RPN + regressor, orange line) made CDF go to lower right than Single scale RPN and resulted in a more portion of higher IoU. RPN with multiple convolution filters (Ours, green line) shows CDF closest to bottom right. This means that our method has the best character localization performance among the three methods and a much higher density of large IoU (+3\% on average) then Single scale RPN, the state-of-the-art method.

Figure 5 shows two examples of Chinese character localization results. The colored boxes represent the detected bounding boxes of the three methods and ground truth boxes. Each colored number shows IoU between the same colored bounding box and ground truth box. In case of detecting isolated Chinese character as shown in Fig. 5 (a), bounding box of our method (green color) includes almost all region of the Chinese character and has the highest IoU, 0.82, while bounding boxes of single scale RPN (blue color)
and regressor (orange color) missed some parts of the Chinese character and have relatively low IoU, 0.68 and 0.69 respectively. In case of overlapping characters as shown in Fig. 5(b), our method (green color) localized two characters separately with very high IoU values (0.93 and 0.87) while the other methods (blue or orange color) incorrectly localized characters as two bounding boxes with a large overlap and low IoU values. This superiority of our method in character localization can be thought due to using multiple convolution filters that can capture the whole regional feature (Fig. 5(a)) by using large-sized filters and avoid small overlapping regions by using small-sized filters (Fig. 5(b)).

As the last experiment, we applied the three methods to 100 test images per character scale and compared their detection rates. We counted the detected bounding boxes with IoU ≥ 0.7 as the true positives, where CDF starts to increase in Fig. 4 and measured true positive rate at false alarm = 5.

Table 2 presents true positive rates of all scales, 16×K for K = 1, 7, and incremental compared to single scale RPN and Fig. 6 shows the measured true positive rates in graphs. Our method shows the highest values for all scales and, even for characters of 48×48 pixels which is corresponding to 3×3 convolution filter, our method has slightly better performance than Single scale RPN. For relatively small characters of 16×16 or 32×32 pixels, our method achieved the higher true positive rates, 0.98 for 32×32 pixels and 0.89 for 16×16 pixels, than Single scale RPN did (0.94 for 32×32 pixels and 0.78 for 16×16 pixels). This is due to that RPN with only 3×3 convolution filter uses features on mixed region of character and background as shown in Fig. 1(b) while our method uses additional 1×1 convolution filter which utilizes features on character-only regions. For large characters of 112×112 pixels, our method has 7×7 convolution filter which covers a whole regional character feature and this might result in higher true positive rate than the other methods which use only 3×3 convolution filter covering partial regions of character as shown in Fig. 1(a). On average, our method achieved a 3% higher true positive rate than Single scale RPN and 1.4% higher true positive rate than Single scale RPN + Regressor.

4. Conclusion

We proposed to use multiple variously sized filters for region proposal network (RPN) to improve Chinese character detection and localization performance in the Faster R-CNN framework. After analyzing the performance of RPN according to the sizes of convolution filters, we chose to use four convolution filters of N×N for N = 1, 3, 5, 7. These filters utilized the features of small regions and the whole region of Chinese characters and this resulted in the improvement of detection rate and localization accuracy. Well designed experiments verified that our method achieved a 3% higher averaged true positive rate and 4% higher averaged intersection over union (IoU) than the previous Faster R-CNN did.

Acknowledgments

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2017M3C1B6071500) and in part by the 2018 Yeungnam University Research Grant.

References

[1] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” Neural Information Processing System (NIPS), Montreal, Canada, pp.91–99, Dec. 2015.

[2] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” International Conference on
Learning Representations (ICLR), San Diego, USA, May 2015.

[3] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, pp.770–778, June 2016, DOI: 10.1109/CVPR.2016.90

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, pp.248–255, June 2009, DOI: 10.1109/CVPR.2009.5206848

[5] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature Pyramid Networks for Object Detection,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, pp.2117–2125, July 2017, DOI: 10.1109/CVPR.2017.106

[6] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, “DSOD: Learning Deeply Supervised Object Detectors from Scratch,” IEEE International Conference on Computer Vision (ICCV), Venice, Italy, pp.1919–1927, Oct. 2017, DOI: 10.1109/ICCV.2017.212

[7] T. Wilkinson, J. Lindström, and A. Brun, “Neural Ctrl-F: Segmentation-Free Query-by-String Word Spotting in Handwritten Manuscript Collections,” 2017 IEEE International Conference on Computer Vision (ICCV), Venice, pp.4443–4452, Oct. 2017, DOI: 10.1109/ICCV.2017.475

[8] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, “CASIA Online and Offline Chinese Handwriting Databases,” 2011 International Conference on Document Analysis and Recognition, Beijing, Sept. 2011, pp.37–41, DOI: 10.1109/ICDAR.2011.17

[9] M.A. Rahman and Y. Wang, “Optimizing intersection-over-union in deep neural networks for image segmentation,” Advances in Visual Computing, Lecture Notes in Computer Science, vol.10072, pp.234–244, Springer International Publishing, Cham, 2016.