2017 14th IAPR International Conference on Document Analysis and Recognition

ICDAR2017 Robust Reading Challenge on COCO-Text

Raul Gomez[1], Baoguang Shi[2], Lluis Gomez[3], Lukas Neumann[4],
Andreas Veit[5], Jiri Matas[6], Serge Belongie[7] and Dimosthenis Karatzas[8]

***Abstract*— This report presents the final results of the ICDAR 2017 Robust Reading Challenge on COCO-Text. A challenge on scene text detection and recognition based on the largest real scene text dataset currently available: the COCO-Text dataset [1]. The competition is structured around three tasks: Text Localization, Cropped Word Recognition and End-To-End Recognition. The competition received a total of 27 submissions over the different opened tasks. This report describes the datasets and the ground truth, details the performance evaluation protocols used and presents the final results along with a brief summary of the participating methods.**

I. INTRODUCTION

The Robust Reading Competition series addresses the need to quantify and track progress in the domain of text understanding in unconstrained settings. The challenge on COCO-Text, organized in the context of the Robust Reading Competition series, focuses on English scene text. It consists of three tasks: Text localization, where the objective is to detect the presence of text and return its bounding box location in the image; cropped word recognition, where the objective is to automatically provide a transcription for a list of pre-localized word images; and end-to-end text recognition, where the objective is to both localize the text and provide its transcription.

This report presents the final results after analyzing the submissions received. The presentation is following the structure of the competition. First, the competition is briefly described in Section II, while Section III presents the dataset. Next, the performance evaluation protocol for each task is detailed and the results are presented and analyzed. Section IV is dedicated to Task 1, Section V to Task 2, and Section VI to Task 3. Overall conclusions are presented in SectionVII. Finally, in the appendix, the technical details of all participating methods are summarized.

II. COMPETITION PROTOCOL

The competition was run in results submission mode, meaning that the authors were requested to submit results over the test set images and not executables of their systems. At all times we relied on the scientific integrity of the authors to follow the rules of the competition. The authors were free to participate in as many tasks as they wished. They were allowed to make multiple submissions to the same task as well. In the case of submission of technically different methods from the same authors, each method is separately described and ranked in the final results presented here. In the case where the results of different variants of the same base method were submitted, the author had to blindly choose one of them as the participating one. A full ranking including all different variants submitted will be available on the competition Web[1]. The python code used for evaluating the performance of submitted methods will be available for download at the Robust Reading Competition portal. The code can be used in a standalone manner, but it will also be offered in a bundle with the validation data and a Web-based graphical interface that can be run on the local machine.

In total, 27 different method submissions were made to the different tasks of the competition from 26 individual participants (excluding different versions of the same method).

III. DATASET

A. COCO-Text

COCO-Text is based on the MS COCO dataset [4], which contains images of complex everyday scenes. The images were not collected with text detection and recognition in mind and thus contain a random selection of incidental text. In this sense, they relate to ICDAR 2015 Robust Reading Competition (RRC) - Challenge 4, on incidental text, referring to text that appears in the scene without the user having taken any specific prior action to cause its appearance or improve its positioning or quality in the frame [2]. Further, a considerable amount of images in the dataset do not contain text. That emulates a real life application and penalizes harder the false positive detections than other datasets.

Text in the COCO-Text dataset is annotated at the word level. Each word instance is annotated with its location in terms of an axis-aligned bounding box, fine-grained classification into machine printed text and handwritten text, classification into legible and illegible text, script of the text and transcriptions of legible text. In this competition only instances labeled as legible and as English script are considered. All other words are treated as *don't care*.

For this competition the COCO-Text annotations have been refined. We firstly generated candidate boxes around

[1] http://rrc.cvc.uab.es/?ch=5

*This work was supported by the Spanish project TIN2014-52072-P and the CERCA programme/Generalitat de Catalunya

[1] R. Gomez is with the Computer Vision Center, Universitat Autonoma de Barcelona and Eurecat.

[2] B. Shi is with the School of EIC, Huazhong University of Science and Technology

[3,8] L. Gomez and D. Karatzas are with the Computer Vision Center, Universitat Autonoma de Barcelona.

[4,6] L. Neumann and J. Matas are with the Center for Machine Perception, Czech Technical University.

[5,7] A. Veit and S.Belongie are with the Cornell University and Cornell Tech.

2379-2140/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDAR.2017.234

every annotation by sampling boxes with random widths and heights around it. The candidates, including the annotation itself, were evaluated by CRNN, which is an off-the-shelf text recognizer[2]. The output of the recognizer is the posterior confidence score of each text instance given its image and groundtruth text. The candidate with the highest confidence score was picked for every annotation. After that, candidates are ranked by their scores and divided into three groups based on their confidence scores. The first group comprises candidates with the highest scores. They were directly taken as the new annotations. The second group consists of candidates with lower scores. Two humans checked these candidates using a side-by-side comparison tool, and picked the candidates that are sufficiently good to replace the old annotations. Candidates in the the third group have the lowest scores. They are discarded and their original annotations were kept.

Using this method, we updated about 41k out of 173k annotations in the dataset. As part of submissions evaluation, we also validated that these GT improvements do not favor any specific submission, by comparing method results using the unrefined as well as the new GT. In particular, the ranking of the method remained the same and the methods accuracy was uniformly higher as a result of the better ground truth quality.

The annotations version used in this competition is v1.4 and the COCO-Text API version used is v1.3. The dataset, the annotations and the API are available for download through the RRC portal[3]. The dataset contains over $173,589$ labeled text regions in over $63,686$ images. This signifies an order of magnitude change from the 1,500 images and $7,548$ regions of the Incidental Scene Text dataset (Challenge 4) of RRC 2015. The splits are: $43,686$ training images, $10,000$ validation images and $10,000$ test images.

B. COCO-Text cropped words

The cropped words images are COCO-Text annotated word boxes padded by 2 pixels on all sides. Only legible English words longer than 3 characters are used. The splits are: $42,618$ training images, $9,896$ validation images and $9,837$ test images.

IV. TASK 1: TEXT LOCALIZATION

The aim of this task is to accurately localize text in terms of word bounding boxes. Participants are asked to run their systems to localize every word on every test image.

A. Performance Evaluation

Following the standard practice in object detection [3],[4], we calculate the Average Precision (AP) for each submission as shown in Eq. 1, where N_p is the number of regions proposed, N_gt is the number of GT regions, $P(k)$ is the precision at k and $M(k)$ is an indicator function equaling 1 if the item at rank k matches a GT region and zero otherwise. Each GT region can only be matched with one proposal. The metric is calculated at $IoU > 0.5$ and $IoU > 0.75$ respectively, where IoU stands for Intersection over Union between the Ground Truth and the evaluated region. Illegible or non-English text are treated as "don't care" objects. The AP at $IoU = 0.5$ is taken as the primary metric for ranking the submissions. This metric is equivalent to the "meanAP" metric adopted by PASCAL VOC, since we only have one category.

$$AP = \frac{\sum_{k=1}^{N_p} P(k) \times M(k)}{N_{gt}} \quad (1)$$

$$P(k) = \frac{\sum_{i=1}^{k} M(i)}{k} \quad (2)$$

Compared to reporting a single Precision, Recall and H-mean value, the metrics used in previous ICDAR Robust Reading competitions, Average Precision has certain advantages. It evaluates how a method ranks the proposed regions, instead of the performance of the method at a particular operating point. Therefore methods are evaluated without the need for selecting any a-priori a threshold for regions' scores.

B. Results and Discussion

15 methods from 14 different participants (excluding variants of the same method) were submitted to the text localization task. The participating methods are referred to by name in the ranking tables and in the text. Please see Tables III and IV for technical details of the participating methods. Table I shows the name of the methods ranked based on their Average Precision. The top-three performing methods, all from different authors and all employing CNN-based models, yield quite similar results. The competition winner is **Foo & Bar**, by Zheqi He and Yongtao Wang, from Peking University, which used a CNN for quadrangle regression. Note that for $IoU > 0.75$ this method remains at the top and with a larger margin, so its localizations seem to be of better quality.

The RRC Web provides tools to visually check the performance of a method image by image. This interface permits exploring the typical failing conditions for a particular method, and the quality of the regions proposed by a method compared to the GT bounding boxes (see Figure 1). Visually analyzing the results of the top ranking methods, we appreciate that they handle reasonably well handwritten text or text written in uncommon fonts. They also generally succeed in detecting text under bad illumination conditions or a difficult perspective. The most notable source of errors is the incorrect separation of different words inside text areas where both under-segmentation (proposing a region containing two or more words) and over-segmentation (splitting a word in different region proposals) can be observed (see Figure 2). Another important source of error seems to be small and short-length text.

An interesting analysis is to check if the methods succeed and fail on the same images. To do so, we have computed for each image the mean AP of the top 10 performing methods in the global ranking. In Figure 3 we have plotted how many

[2] http://github.com/bgshih/crnn

[3] http://rrc.cvc.uab.es/?ch=5&com=downloads

images fall in each mean AP range. The plot shows that there are many images where all the methods succeed. For 412 images (out of 2,793 test images with text) all the top 10 methods achieve $AP = 1$. However, the methods fail in different images: There are 49 images where no text is being localized by any method. For around half of those images this is partly due to bad quality annotations. The other half contain small or short-length text, which seems to be the most difficult to detect by the participating methods. Some of those images are shown in Figure 4.

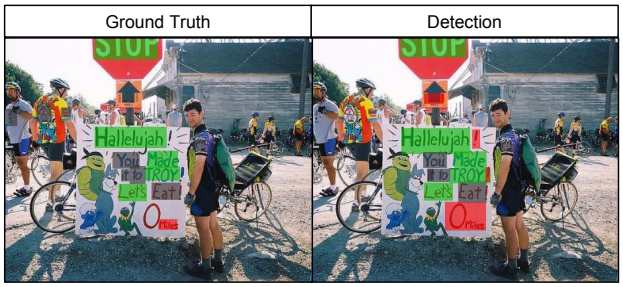

Fig. 1. Web-based visualization of the ground truth (left) and the detection results (right) of a method. On the left, correctly detected GT regions ($IoU > 0.5$) are shown in green, missed regions are shown in red and "don't care" regions in gray. On the right, detected regions matching the ground truth are shown in green, regions not matching the ground truth (false positives) are shown in red, and regions matching a "don't care" bounding box (not considered during evaluation) in gray.

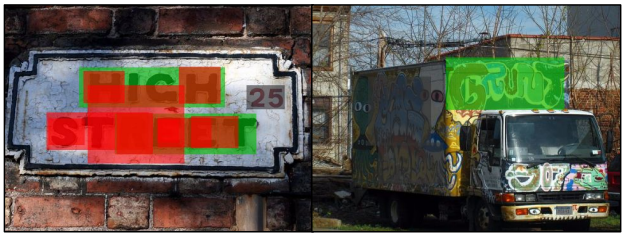

Fig. 2. On the left, the top ranking method **Foo & Bar** apart from proposing regions for the two machine printed and defined words, produces also a lot of smaller overlapping false positive regions. On the right, the same method succeeds in detecting a word in graffiti.

V. TASK 2: WORD RECOGNITION

The aim of this task is to recognize previously localized words based on their cropped word images. The recognition process is unconstrained in the sense that there was no lexicon provided.

A. Performance Evaluation

The metrics reported for this task include the percentage of Correctly Recognized Words (CRW) (both for case sensitive and case insensitive) and Total Edit Distance (TED) (both case sensitive and case insensitive). The Case-insensitive percentage of Correctly Recognized Words is taken as the primary ranking metric. The annotations contain Latin letters, numbers and other symbols that are all considered in the evaluation.

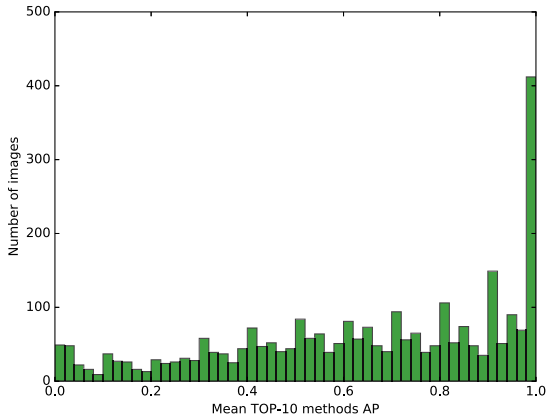

Fig. 3. Histogram of the mean-AP scores of the top-10 performing methods. Only test images with text (2,793 images) are used in this plot.

TABLE I
RANKING OF METHODS SUBMITTED TO TASK 1 – TEXT LOCALIZATION

Method Name	AP (IoU>0.5)	AP (IoU>0.75)
Foo & Bar	67.16	32.10
SRC-B-Machine Learning Lab	66.30	27.86
CCFLAB	64.67	28.29
Tencent-PRC & USTB-PRIR	61.95	20.53
UM	51.01	16.91
InfoNet	49.57	14.23
Text Detection DL	48.90	11.38
SCUT-DLVClab	48.76	11.64
SARI FDU RRPN	46.16	8.49
SCUT-DLVClab-HuangGroup	41.79	11.73
BRTRS-Detection	41.21	5.66
TextFCN	29.57	4.53
CNN-LSTM based text detection	6.19	0.53
RFCN	3.41	1.22
Cas-hoteye	0.26	0.00

B. Results and Discussion

8 methods from 8 different participants (excluding variants of the same method) were submitted to the word recognition task. The participating methods are referred to by name in the ranking tables and in the text. Please see Table V for technical details on the participating methods. Table VII shows the name of the methods ranked by %CRW. The competition winner is **HIK_OCR**, by Zhanzhan Cheng*, Gang Zheng*, Fan Bai, Yunlu Xu, Jie Wang, Ying Yao, Zhaoxuan Fan, Zhiqian Zhang and Yi Niu (*equal contribution), from Fudan University, which built on the sequence-sequence framework and used a CNN with an Edit Probability loss.

We analyzed how many methods recognized correctly each word of the dataset, in order to identify typical cases where methods consistently succeed or fail. To do so, we have plotted in Figure 5 how many words were recognized by a certain number of the top 5 ranked methods. The plot shows that there are many images where all top-5 methods succeed: For 3,675 images (out of 9,838 test images) all the top 5 methods achieve a correct recognition. There are also many images, 1,864, where all the methods fail. Those

Fig. 4. Example images where all the top 10 ranked methods failed to localize the text.

are images where the words do not appear completely or they are shown in a difficult perspective – some examples are shown in Figure 6.

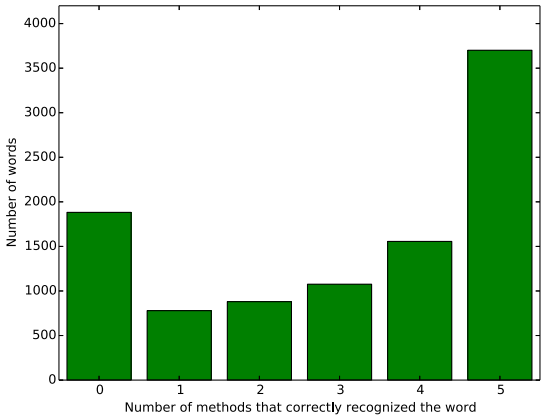

Fig. 5. Histogram of correctly recognized words of the top-5 ranked methods.

VI. TASK 3: END-TO-END RECOGNITION

The aim of this task is to both localize and recognize words in images. Only legible English words longer than 3 characters are considered. The rest are treated as "don't care" objects. The annotations contain Latin letters, numbers and other symbols. In the evaluation we considered symbols that appear in the middle of words (e.g. "e-mail", "128.0.0.1"), but ignored the symbols *!?.:,*"()[]/'_* at the beginning and at the end of both the ground truth transcription and the submitted results. Hence, in the case of the ground truth transcription "Hello!", both "Hello" and "Hello!" would be considered correct. If more than one of these symbols appear in the beginning or the end of a word, they all are removed. The evaluation is case-insensitive.

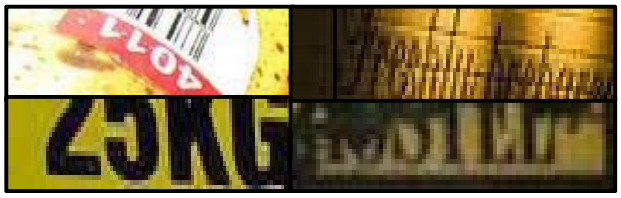

Fig. 6. Example images where all the top-5 ranked methods failed to recognize the words.

A. Performance Evaluation

Average Precision (AP) is calculated in the same way as in Task 1, except that the recognition results are taken into consideration in addition to localization. A detection is considered a true positive if two conditions are fulfilled: Its bounding box sufficiently overlaps with the matching ground truth box (correct localization) and its recognition matches the ground truth word (correct recognition. For localization, an Intersection over Union of $IoU > 0.5$ is required.

B. Results and Discussion

4 methods from 4 different participants (excluding variants of the same method) were submitted to the end-to-end task. The participating methods are referred to by name in the ranking tables and in the text. Please see Table VI for technical details on the participating methods. Table II shows the name of the methods ranked by their Average Precision. Note that 3 of the 4 participants also competed in the text localization task, although their end-to-end ranking is different. Also, the difference of performance between the top ranked methods is much bigger in the end-to-end task compared to the text localization task. The competition winner is **Tencent-PRC & USTB-PRIR**, by Chun Yang, Zejun Li, Jianwei Wu, Jiebo Hou, Chang Liu, Longhuang Wuand and Xu-Cheng Yin, from the University of Rochester, which first detects text regions, then extracts features from text lines and finally employ multiple LSTM-based models to read the text. The same authors submitted a method that end in 4th position in Task 1. The authors of the winning method of Task 1, **Foo & Bar**, are in 2nd position in the end-to-end task.

TABLE II
RANKING OF SUBMITTED RESULTS TO TASK 3 – END-TO-END RECOGNITION

Method Name	**AP** ($IoU > 0.5$)
Tencent-PRC & USTB-PRIR	43.58
Foo & Bar	27.01
WPS	18.82
CNN-LSTM based text recognition	0.73

VII. CONCLUSIONS

The ICDAR2017 Robust Reading Challenge on COCO-Text had a very high participation. Reading text in the wild is an active field of study and still an open problem due to the variety of text and all the different contexts it appears

in. At the same time, the field is evolving very fast and the methods are quickly becoming more efficient in recognizing more difficult text. That is why more challenging datasets, such as COCO-Text, are important to keep pushing the limit upwards.

The performance of the top-ranking methods in the text localization and word recognition tasks has been very close. That is a good indicator of where current state of the art is. However, it is interesting to notice that methods are not generally failing on the same images. While in many cases errors are due to over- or under-segmentation of text, the fact that annotations and performance evaluation is done at word granularity and penalizes methods that return, for example, whole text lines, is a recurring issue for which various, yet not satisfactory, solutions have been proposed to this date [9].

Another interesting conclusion is that text extraction methods tend to see text where there is none: The COCO-Text dataset includes a significant amount of images that do not contain any text, and are the cause for most of the false positive detections produced; yet, in most of the cases the confidence of these detections is low and hence the Average Precision metric is not severely affected.

REFERENCES

[1] A. Veit, T. Matera, L. Neumann, J. Matas, S. Belongie. COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images. arXiv preprint arXiv:1601.07140, 2016.

[2] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, D. Ghosh , A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, VR. Chandrasekhar, A. Lu, F. Shafait, S. Uchida, E. Valveny.: ICDAR 2015 robust reading competition. 13th International Conference on Document Analysis and Recognition (ICDAR).

[3] M. Everingham, L.V. Gool, C.K. Williams, J.M. Winn, A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision, 88, 3

[4] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, C.L. Zitnick. Microsoft COCO: Common Objects in Context. ECCV (2014).

[5] Y. Jiang, X. Zhu, X. Wang, S. Yang, W. Li, H. Wang, P. Fu, Z. Luo. R2CNN: Rotational Region CNN for Orientation Robust Scene Text Detection. arXiv:1706.09579v2 (2017).

[6] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, X. Xue. Arbitrary-Oriented Scene Text Detection via Rotation Proposals. arXiv:1703.01086v2 (2017).

[7] B. Shi, X. Bai, C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. CoRR, 2015.

[8] A. Graves, S. Fernandez, F. J. Gomez, and J. Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In ICML, 2006.

[9] C. Wolf and J. Jolion. Object count/Area Graphs for the Evaluation of Object Detection and Segmentation Algorithms. International Journal on Document Analysis and Recognition, 8(4):280-296, 2006.

[10] S. Ren, K. He, R. Girshick and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. NIPS. 2015.

[11] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu and Y. Wei. Deformable Convolutional Networks. NIPS. 2016.

[12] Z. Tian, W. Huang, P. Heand Y. Qiao. Detecting Text in Natural Image with Connectionist Text Proposal Network. European Conference on Computer Vision (pp.56-72). 2016.

TABLE III

DESCRIPTION OF METHODS PARTICIPATING IN TASK 1

Method Name	Authors	Affiliation	Brief Description
Foo & Bar	Zheqi He, Yongtao Wang	Peking University	An improvement of Faster RCNN [10] to meet the requirement of detecting quadrilateral object like text. The bounding box regression layer is replaced with a quadrangle regression layer, and the regression target and the loss function are modified accordingly. ResNet-101 is used as base net. To incorporate ResNet-101 and Faster R-CNN, conv4_x and conv5_x are disconnected from ResNet-101, the downsampling of conv4_x is removed and the region proposal network (RPN) and RoIPooling are inserted between them. The network first processes the whole image to produce a convolutional feature map. This map is used as the input of RPN to generate regions of interest (RoIs), each with an objectness score. These RoIs and the feature map generated by conv4_x are fed to the RoiPooling layer in order to get the fixed-size feature map. This feature map is fed to several convolutional layers (conv5_x). Conv5_x and layers after it play the roles of fully connected layers commonly seen in VGG networks, they calculate the feature map of each RoI and these feature map is pooled by a global average pooling (GAP). Finally, the output of GAP is fed into two sibling output layers: a classification layer to get the label of each ROI, and a quadrangle regression layer that outputs 8 real-valued numbers for each RoI, each set of 8 values encodes the coordinates of the vertices of the text region. The method is implemented under TensorFlow. The detection network is pre-trained on imagenet, no any other additional data was used.
SRC-B Machine Learning Lab	Yingying Jiang, Xiaobing Wang, Xiangyu Zhu, Shuli Yang, Wei Li, Zhenbo Luo	Samsung R&D Institute China - Beijing. Machine Learning Lab	Based on "R2CNN: Rotational Region CNN for Orientation Robust Scene Text Detection [5].
CCFLAB	Dai Yuchen	Shanghai Jiao-Tong University	Uses Deformable Convolutional Nets [11] as the base architecture. A resnet-101 is used as the backbone convolutional network for feature extraction. During feature extraction, deformable convolution layers are added to catch the text patterns with deformable convolutional kernels. Then region proposal network, which are 3x3 convolutions, generate regions of interest. Then a deformable ROI pooling layer is used to crop ROIs to fixed-size feature maps. Then these representation of ROIs are sent to the final classification and box-regression branches.
Tencent-PRC & USTB-PRIR	Jiebo Hou, Chang Liu, Longhuang Wu, Xu-Cheng Yin	University of Rochester and Precision Recommendation Center (PRC), Tencent.	An ensemble of multiple CNN models.
UM	Chng Chee Kheng	University of Malaya	A variant of faster RCNN [10] inception resnet v2. Modifications were done to change its anchor sizes.
InfoNet	Wenhua Cheng, Anbang Yao, Libin Wang, Dongqi Cai	University of Science and Technology of China	A novel fully convolutional neural network framework, which explicitly bridges the joint detection of informative text region centroid and informative text region context, presenting a new way towards straightforward scene text detection.
Text Detection DL	Liu Ming	Macau University of Science and Technology	Based on Faster RCNN [10]. The method is a modified version of FRCNN with some anchor and convolution feature extractor changes, basically borrowed from Resnet and FRCNN.
SCUTD LVC lab	Yuliang Liu, Sheng Zhang, Lianwen Jin	South China University o f Technology	A two stage deep learning method is used to roughly recall text first and then finely adjust the bounding box for compact detection. A novel network was carefully designed for multi-scale training, and a learnable method was used to generate quadrilateral proposals for tightly matching the scene text. Finally, a post-processing pipeline was designed for further improving the precision.
SARI FDU RRPN	Jianqi Ma, Weiyuan Shao, Yingbin Zheng, Hong Wang, Li Wang, Hao Ye, Xiangyang Xue	Shanghai Advanced Research Institute, CAS & Fudan University	Based on the Rotation Proposal Framework, from Arbitrary-Oriented Scene Text Detection via Rotation Proposals [6].
SCUTDLVC lab Huang Group	Jinrong Li, Zijian Zhou, Shuangping Huang, Zhengzhou Zhuang, Daihui Yang	South China University of Technology	A multi-branch convolution neural network (MBCNN) that has three branches with an harmonization mechanism to make all the three branches cooperate with each other. The core idea of this architecture is called Special Problem Special Solution.

TABLE IV

DESCRIPTION OF METHODS PARTICIPATING IN TASK 1

Method Name	**Authors**	**Affiliation**	**Brief Description**
SCUTDLVC lab Huang Group	Jinrong Li, Zijian Zhou, Shuangping Huang, Zhengzhou Zhuang, Daihui Yang	South China University of Technology	A multi-branch convolution neural network (MBCNN) that has three branches with an harmonization mechanism to make all the three branches cooperate with each other. The core idea of this architecture is called Special Problem Special Solution.
BRTRS Detection	Chengquan Zhang	BRTRS-Team. Institute of Deep Learning, Baidu Research, China	A robust detector with FCN is used to find all candidate characters. Then, text-lines are generated by a graph-based grouping method. Finally, a Bi-LSTM model is trained to split the text-line into several words.
TextFCN	Manuel Rota		Text Localization using a FCN (VGG as a base) which is trained to predict bounding boxes and bounding boxes divisions when two are too close.
CNN-LSTM based text detection	Lulu Xu	Sogou Inc.,China	The detection results are achieved by a combination of convolutional and recurrent nets. The feature maps are generated by a fully convolutional network based on VGG models. The FCN detects a text line by sliding a window in the last convolutional feature maps and outputs a sequence of a fixed-width text proposals densely. Then the sequential text line proposals are connected by a recurrent neural network.
RFCN	Dao Wu, Pengwen Dai	State Key Laboratory of Information Security, Chinese Academy of Sciences	Based on the RFCN framework.
Cas-hoteye	Dao Wu, Pengwen Dai	State Key Laboratory of Information Security, Chinese Academy of Sciences	Based on RPN, adopts the idea of vertical anchor mechanism and recurrent connectionist text proposals in Connectionist Text Proposal Network (CTPN) [12].

TABLE V

DESCRIPTION OF METHODS PARTICIPATING IN TASK 2

Method Name	Authors	Affiliation	Brief Description
HIK_OCR	Zhanzhan Cheng*, Gang Zheng*, Fan Bai, Yunlu Xu, Jie Wang, Ying Yao, Zhaoxuan Fan, Zhiqian Zhang, Yi Niu (*equal contribution)	Fudan University	The method is designed based on the sequence-sequence framework. In the encoder part, images are resized to 100x100, and features are extracted by using a CNN; In the decoder part, character sequence is generated by an attention-based decoder. The novelties of their method include 1) A complicated CNN-based model is proposed for the feature extraction. The model has a few special mechanisms, including mask spatial transform, for handling text of arbitrary placement; 2) Instead of softmax loss, an Edit Probability Loss is developed for training; 3) A self-adaption gate mechanism is adopted to capture global information.
Tencent-DPPR Team	Chun Yang, Zejun Li, Jianwei Wu, Jiebo Hou, Xu-Cheng Yin	University of Rochester	First, they use CNN to extract features from images. Second, they employ multiple LSTM-based models to generate different results and thus to obtain a candidate set for each image. Third, they design a heuristic mechanism to select the result with the maximum probability for each image.
HKU-VisionLab	Wei Liu, Chaofeng Chen, Bingbin Liu, Kwan-Yee Kenneth Wong	University of Hong Kong	They propose a Character-Aware Attention Network (Char-Net) for scene text with large spatial deformations. Their Char-Net consists of a hierarchical feature encoder and a LSTM-based decoder. The newly proposed encoder is able to encode the original text image from both word and character levels, which enables our Char-Net to handle severely distorted scene text. The whole neural network can be optimised in an end-to-end fashion. All the training data comes from public datasets for scene text recognition.
BRTRS-Recognition	Chengquan Zhang	BRTRS-Team. Institute of Deep Learning, Baidu Research, China	A CNN is used to extract features. Then, the RNN based encoder-decoder model is trained to get the result. Results from the attention based model and the CTC based model are combined to get the final output.
CCFLAB	Dai Yuchen	Shanghai Jiao-Tong University	In this method, a convolutional bidirectional-LSTM followed by a CTC loss layer is used for text recognition. A 34-layer Residual Network is used for feature extraction, and a single-layer bidirectional-LSTM is added for learning the text sequence. Finally, a CTC loss layer is used to do alignments.
3CNN 2BiLSTM CTC	Ma Long	Sogou Inc., China	The model for line recognition is based on a convolution recurrent neural network. For a test line, a fully convolution network (FCN) is used to extract features which is fed into LSTMS. The FCN model consisted of 18 convolution layers and 3 max-pool layers. For a test line with size 56*320, the last feature map of FCN has a size of 42*(56/8)*(320/8). Then the channels are concatenated, results in a 294*40 feature map. There are 2-layer bidirectional LSTM after FCN network, the predicted distributions is fed into Connectionist Temporal Classification (CTC) layer [8]. The proposed model is similar to CRNN [7]. The differences are that our model can treat sequences in arbitrary lengths, the width of the image is rescaled according to aspect ratio. The model is trained based on 150W lines that labeled by human. It takes about 20 epochs to reach a good model using a single NVIDIA(R) Tesla(TM) M40 GPU.
Enhancing Acc. Adding External Language Model	Ahmed Sabir	ALP Research Center. UPC Center for Language and Speech Technologies and Applications	This approach focuses on the integration of independent linguistic model to a pre-trained deep network. The advantage of trainable linguistic model is to enrich the probability of the words selected by the network taking into account external knowledge (in this case, a unigram language model learnt from freely available corpus). This hybrid approach opens the possibility of introducing higher-order trainable linguistic models. They apply our unigram language model (LM) over a deep CNN with a 90k-words pre-defined dictionary. The unigram model was trained on the Opensubtitles corpora. Opensubtitles is a database based on subtitles for movies. The corpus contains around 3 million words (combination of words and digits). They took only the five max probable words output form CNN softmax layer with pre-defined dictionary, and rerank them combining the softmax output with the unigram probabilities estimated from large-scale English corpora.
LSTM based text recognition	Lulu Xu	Sogou Inc.,China	For text line recognition, they implement an optimized convolutional recurrent neural network which was first proposed by Baoguang Shi et al. [7]. The proposed network can handle sequences in arbitrary lengths. In CRNN model, the feature extraction part is fully convolution network (FCN) based on a simplified Inception-Renset network which was build by 20 convolution layers and 3 max-pool layers. The FCN network is followed by recurrent neural network which was build by 4 LSTMs, two forward and two backward. RNN predicts each column of the last feature map in FCN, the predicted distributions is fed into Connectionist Temporal Classification (CTC) layer.

TABLE VI

DESCRIPTION OF METHODS PARTICIPATING IN TASK 3

Method Name	Authors	Affiliation	Brief Description
Tencent-PRC & USTB-PRIR	Chun Yang, Zejun Li, Jianwei Wu, Jiebo Hou, Chang Liu, Longhuang Wu, Xu-Cheng Yin	University of Rochester & Tencent-DPPR (Data Platform Precision Recommendation) Team.	They detect text regions using improved Rotation Region Proposal Networks. After that, they extract features from text lines and employ multiple LSTM-based models to generate different results for each image. Finally, they select the one with the maximum probability among all candidate results.
Foo & Bar	Zheqi He, Yongtao Wang, Xiang Bai	Peking University	The method uses quadrangle regression network for text detection, and then uses homography to transform quadrangle regions to rectangles and finally CRNN [7] for recognition.
WPS	Hin Lee		A word spotting system that combines a text-proposal extractor and a attention-based text recognizer.
CNN-LSTM based text recognition	Lulu Xu	Sogou Inc.,China	The detection results are achieved by a combination of convolutional and recurrent net. The feature map are generated by a full convolution network (FCN) based on VGG16 models which consist of 14 convolutional layers divided into 5 stages. The full convolutional network detects a text line by sliding a certain size of window in the last convolutional feature maps of FCN and outputs a sequence of a fixed-width text proposals densely. Then the sequential text line proposals are connected by a recurrent neural network. They exploit the long short-term memory (LSTM) architecture for the RNN layer. With an image post processing of the false positive proposes removal and component grouping, the final detections are achieved. For text line recognition, they implement an optimized convolutional recurrent neural network which was first proposed by Baoguang Shi [7]. The proposed network can handle sequences in arbitrary lengths. In CRNN model, the feature extraction part is fully convolution network (FCN) based on a simplified Inception-Renset network which was build by 20 convolution layers and 3 max-pool layers. The FCN network is followed by recurrent neural network which was build by 4 LSTMs, two forward and two backward. RNN predicts each column of the last feature map in FCN, the predicted distributions is fed into Connectionist Temporal Classification (CTC) layer.

TABLE VII

RANKING OF SUBMITTED RESULTS TO TASK 2 – CROPPED WORD RECOGNITION

Method Name	CRW (case insensitive)	TED (case insensitive)	CRW (case sensitive)	TED (case sensitive)
HIK_OCR	76.11%	899.10	41.72%	3,661.58
Tencent-DPPR Team & USTB-PRIR	70.83%	1,233.46	36.91%	4,022.12
HKU-VisionLab	59.29%	1,903.37	40.17%	3,921.94
BRTRS-Recognition	59.25%	2,282.49	28.18%	4,895.96
CCFLAB	42.66%	2,982.66	26.52%	4,743.28
3CNN_2BiLSTM_CTC	30.17%	4,395.42	12.19%	6,405.61
Enhancing Acc. Adding External Language Model	29.69%	5,555.89	17.88%	7,231.87
LSTM based text recognition	26.25%	4,638.83	10.11%	6,594.01