Cross domain evaluation of text detection models.

ALI-GOMBE, A., ELYAN, E., MORENO-GARCÍA, C. and JAYNE, C.

2022

Cross Domain Evaluation of Text Detection Models

Adamu Ali-Gombe[1] Eyad Elyan[1] Carlos Moreno-García[1] Chrisina Jayne[2]

[1] Robert Gordon University Aberdeen United Kingdom
[2] Teesside University Middlesbrough United Kingdom

Abstract. Text detection is a very common task across a wide range of domains, such as document image analysis, remote identity verification, amongst others. It is also considered an integral component of any text recognition system, where the performance of recognition tasks largely depends on the accuracy of the detection of text components. Various text detection models have been developed in the past decade. However, localizing text characters is still considered as one of the most challenging computer vision tasks within the text recognition task. Typical challenges include illumination, font types and sizes, languages, and many others. Furthermore, detection models are often evaluated using specific datasets without much work on cross-datasets and domain evaluation. In this paper, we present an experimental framework to evaluate the generalization capability of state-of-the-art text detection models across different application domains. Extensive experiments were carried using different established methods: EAST, CRAFT, Tessaract and Ensembles applied to various publicly available datasets. The generalisation performance of the models was evaluated and compared using precision, recall and F1-score. This paper opens a future direction in investigating ensemble models for text detection to improve generalisation.

Keywords— Text Detection.

1 Introduction

Text detection separates text from non-text objects in a given image or video while recognition classifies and identifies text from images. An example is extracting labels, and annotations from engineering diagrams [8]. Text recognition is common across a wide range of applications. Examples include surveillance, number plate recognition, information retrieval, and others [16]. Similar to other computer vision tasks, text detection and recognition have seen significant progress in recent years due to the latest development in Deep Learning [4]. Traditional text recognition methods consist of localising individual characters, building a features space, and use specific machine learning algorithms for the classification tasks (recognition) [16]. In the deep learning era, it is common to see an end-to-end framework for text recognition, or simply splitting it into detection and recognition components [15].

Despite the significant achievement in text detection and recognition in recent years [24, 25, 12], localising and recognising text in specific domains still a very challenging task. For example, consider the Processing and Instrumentation Diagram (P&ID) which is commonly used in the Oil and Gas industry (Figure 1). Such diagrams contain various types of graphic elements (symbols, lines, other shapes), and text which in most cases overlap with other elements and thus makes the text detection task more challenging [8]. This scenario shows that the complexity of text detection/recognition can also be found in many other domains, including document verification, medical images, and others.

A relatively recent review paper [18] shows that, despite the significant progress in deep learning, and in particular applied to the computer vision domain, traditional text detection and recognition methods are still widely used in such complex scenarios. Nonetheless, they are largely ineffective as they are incapable of dealing with the aforementioned issues (i.e.shape overlap).

The typical approach in deep learning is to use a Convolution Neural Network (CNN) to detect and localise text [13]. For recognition, a Recurrent Neural Network (RNN) is commonly employed [12] to read words and sequences from image features. Other end-to-end systems combine both CNN and RNN to localise and recognise text from images [26]. Deep learning approaches have become more popular recently because learned features are more invariant to text recognition challenges. However, some challenges like text perspective are better handle with dedicated modules such as the rectification network in [26]. This can add to the complexity/scalability of the model in a typical application domain. The availability of public datasets and text detection challenges have elevated the performance of text recognition models. These datasets have provided text detection task in three main scenarios; text detection from a scanned document, focused text detection and text detection in the wild. Samples and task in these datasets are carefully curated and may not necessarily reflect the real-world domain.

In this paper, we evaluate the performance of state-of-the-art text detection models. We analysed how these models generalise on similar tasks in different datasets. We focused on the text detection task and we consider the text detection models: Efficient and Accurate Scene Text Detector (EAST) [33], Character Aware Region Awareness for Text Detection (CRAFT) [2], and Tesseract. We evaluate these models on ICDAR2013 [10], ICDAR2015 [9], MIDV-500 [1] datasets and P&ID diagrams. We also created an ensemble of the outputs from various models to address text detection challenges in P&ID. Cross dataset performances were also investigated, and the results show a drop in model performances but with noticeable improvement in some metrics for the ensemble.

The rest of the paper is organised as follows. In Section 2, related text recognition literature is reviewed and discussed. Section 3 presents the methods used in this work. Section 4 discusses in details the experimental set-up and the datasets used. Findings are discussed in section 5. Finally, we conclude and suggest future directions in Section 6.

2 Related Work

Traditional Optical Character Recognition (OCR) systems rely on features engineering to isolate a character. These are good at localising text from a scanned document and focused texts. However, they require preprocessing steps such as binarisation and de-blurring to de-noise images in most cases. Pre-processing steps could be domain-dependent and adds an extra step in the detection pipe. For instance, Strokelets [32] are a multiscale representation of different structural characteristics of characters ranging from arcs and corners to the character itself. Again, a region-based detector like Features pooling [11], combines pixel-wise low-level features by using a region-based pooling scheme. This was found to perform better than Histogram of Gradients (HOG) features in terms of both speed and accuracy. However, Curved text, perspective text and text detection in the wild are challenging for traditional OCR system. Thus, there is a renewed interest in modern deep learning approaches.

Deep learning based models are preferred over traditional approaches because they address text detection challenges better [8], [16]. These approaches could be categorised based on different characteristics such as prediction pipeline (single shot / text proposal networks), model type (discriminative / generative model), the task (document, perspective text and text in the wild) or bounding box post-processing technique (bounding box regression and binarisation methods). Loosely, we use the detection pattern to categorise deep learning approaches viz; character-based, word-based, line-based or a text segmentation-based approach.

2.1 Character-based detectors

Character-based methods rely on character features and shape in the detection pipeline to isolate, fine-tune or build the final text. Shi *et al.* [23] proposed Segment Linking (SegLink) approach to detecting oriented texts. Segments are parts of a word or text line while a link connects two adjacent segments belonging to the same word. Both were predicted using a VGG-16 backbone with convolutional predictors. A depth-first search was used to find connected segments from a word. SegLink is efficient in terms of speed and in detecting oriented texts. However, spaces between texts are not uniform, and SegLink fails in detecting text with large character spacing.

Similarly, Character Region Awareness Text Detector (CRAFT) [2] is a deep model that performs text detection using character-level detection with character affinity. CRAFT was designed to handle curved and long texts which are challenging cases for rigid word box predictors. CRAFT uses weak supervision to estimate character-level ground truths which are lacking in-text detection datasets. Affinity and region scores are used to guide the model during training. Craft out-performed state-of-the-art model on ICDAR2013, ICDAR2015, ICDAR2017 and MSRA-TD500 datasets.

2.2 Word-based detectors

Words are irregular in the wild with obvious challenges such as scale, perspective and curved sequences [26]. Irregular texts detection was addressed by Attentional Scene Text Recognizer with Flexible Rectification (ASTER) using a rectification network. The rectification step uses Spatial Transformer Networks (STN) [7] totransforms curved texts into regular horizontal text image before a recognition model is applied. An interesting aspect of this is that rectification does not require human annotation or character level detection. Experiments were conducted on ICDAR datasets [9, 10], CUTE80 [22], SVT-Perspective [20], Street View Text (SVT) [29] and IIIT5k-Words (IIIT5k) [17] and SynthText [5], and results reported show superior performances over existing techniques.

Using a fully convolutional network and NMS, Efficient and Accurate Scene Text Detector (EAST) [33] achieved state-of-the-art performance on ICDAR2015 dataset. EAST uses a novel loss function and does not require text region proposal, word partitioning or other intermediate steps. The Fully Convolutional Network (FCN) outputs text score maps and geometry from multiple channels which are passed to Non-Maximum Suppression (NMS) for post-processing. The model was designed to use rotated boxes or quadrangles for detection and a separate loss was used for each case. Results showed that EAST performed well in challenging scenarios such as irregular illumination, low resolution, orientation and perspective distortion.

2.3 Line-based detectors

Line-based detectors combine a sequence of text detections into text lines or directly localise text line as objects. Cascade Convolutional Text Network (CCTN) [6] detect whole text region and text line from a coarse low-resolution image to fine-grain as regions are enlarged. Interestingly, this approach does not rely on any post-processing. Rectangular convolutions with in-network fusion is employed to handle multi-shape and multi-scale text lines. The model consists of a coarse network that outputs per pixel heat map that indicates the location and probability of the text region. And a fine network that outputs two heats map representing a finer text line and text area. Experiments showed that CCTN has a high discriminative ability to distinguish text and no text line in multiple text variations while surpassing best results on ICDAR datasets.

Connectionist Text Proposal Networks (CTPN) [28] is an extension of Region Proposal Networks (RPN) [21] in text detection. Text line proposals are generated from convolution maps obtained from a VGG-16 network. Then a vertical anchor mechanism is used to predict text, non-text score and y-axis location of each proposal. An in-network recurrence layer is used to improve text context (using RNN with LSTM) to refine location in the vertical direction. Then side refinement is used to estimate offsets of each proposal in a horizontal direction. This essentially connects sequential text proposals. CTPN was effective on muli-lingual and multi-scale (i.e small scale text) problems and are quite fast at about 0.14s per image (GPU time). CTPN out-performed other existing detection methods on ICDAR, SWT and MULTILINGUAL [19] datasets.

2.4 Segmentation-based detectors

While other approaches rely on word/text/line proposals and eliminate false detection in post processes, segmentation methods approach detection in a holistic manner [31]. For instance, Liao *et. al* used a lightweight segmentation network with a novel Differentiable Binarisation (DB) module [14] to detect text from images. DB adaptively predict threshold values at different pixels to isolate text from background. The DB+segmentation set-up is trainable end-to-end, lightweight and consequently a very fast detector. At inference, bounding boxes are generated from binary and probability maps. Extensive experiments showed that DB is robust and effective in detecting curved and multi-lingual texts. Subsequently, DB became the choice detector in text recognition system such as PP-OCR [3] for its computational efficiency.

Yao *et. al* in [31] approached text detection as a semantic segmentation task. Using a single CNN, the authors detected texts from pixel-wise prediction maps and build a graph that predicts character properties such as scale, location, orientation and others. The framework predicts text regions, individual characters and the relationships between them at runtime. Experiments were conducted on COCO-Text, ICDAR2013, ICDAR2015 and MSRA-TD500 dataset. The results show that the model out-performed existing methods and was invariant to text orientation, font, scale and local distractors.

All these models excel in many different ways but are limited in certain context. For instance, contextual information is lost in selecting correct text boxes in proposals and also text bounding boxes have a much larger aspect ratio than objects. A common limitation of all segmentation approaches is that they fail to detect correctly text that is enclosed in another text.

3 Method

3.1 EAST

The EAST model utilised in this experiment uses a ResNet50 stem rather than PVANet2x. Although the same U-shape is maintained with the model divided into feature extraction branch, a feature merging branch and the output branch. The network outputs a score map (confidence) for pixel locations and a set of geometry representing the predicted text boxes. Again, the output geometry is based on a rotated box only. Outputs regions are further post-processed by binarising each region followed by a locality aware NMS to obtain the best possible text location. The loss function is a sum of the score map loss and the geometry loss, $L = L_s + \lambda_g L_g$ where λ_g is a hyper-parameter. Score map is evaluated using class-balanced cross entropy [30] as shown in Equation 1. The geometry loss is the sum of IOU using Axis-Aligned Bounding Box (AABB), and the rotation angle loss, $L = L_{AABB} + L_\theta$. This is shown in Equation 2.

$$\begin{aligned} L_s &= \textit{balanced-xent}(\hat{Y}, Y^*) \\ &= -\beta Y^* \log \hat{Y} - (1-\beta)(1-Y^*)\log(1-\hat{Y}) \end{aligned} \tag{1}$$

$$\begin{aligned} L_{AABB} &= -\log IoU(\hat{R}, R^*) \\ L_\theta &= 1 - \cos(\theta, \hat{\theta}^*) \end{aligned} \tag{2}$$

where $\hat{Y}$ is the predicted score map, Y^* is the ground truth, β is a balancing factor, $\hat{R}$ is the predicted AABB geometry, R^* is the ground truth, $\hat{\theta}$ is the predicted rotation angle and θ^* is the corresponding ground truth. See [33] for details on the loss functions.

3.2 CRAFT

CRAFT model is similar to EAST in terms of architecture. Both use the same U-Net structure however, CRAFT relied on a VGG-16 backbone. Again, CRAFT is built for character level detection. The final output of the model is a region score and an affinity score. The region score predicts the centre of a character while the affinity score is the centre probability of space between adjacent characters. As most text detection datasets use word-level annotations, character level annotations are generated using weak supervision. CRAFT uses connected component labelling to generate word boxes by finding the rotation angle. The loss function is shown in Equation 3.

$$L = \sum_p S_c(p) \cdot (||S_r(p) - S_r^*(p)||_2^2 + ||S_a(p) - S_a^*(p)||_2^2 \tag{3}$$

where $S_c(p)$ is the pixel-wise confidence, S_r and S_a are the predicted region score and affinity score, S_r^* and S_a^* are the pseudo-ground truth region score and affinity score. For more details, the reader is referred to [2].

3.3 Tesseract

The fourth model used in this experiment is Tesseract. Tesseract is an open-source OCR engine developed by Hewlett Packard and now maintained by Google. Tesseract-4 was used in our experiment which is extended with a deep learning engine. In particular, Tesseract-4 uses LSTM based recogniser[3] for more details). to predict text which is better than the traditional pipeline in 3.0. Images are processed by sliding window over the image. Each window is fed to the LSTM engine in a sequence [27]. Tesseract still requires ideal images for improved performances.

3.4 Outputs Ensemble

Our final model is an ensemble that combines outputs from the models described above. The aim is to boost performance by merging the outputs of participating

[3] https://tesseract-ocr.github.io/tessdoc/NeuralNetsInTesseract4.00

models. Different permutations of the models were considered, and a total of ten ensembles were created. Multiple detections and overlap conflicts were resolved through post-processing using three criteria. The first case coalesces the results based on a reference model. The reference model is the first model in the ensemble name. Outputs from the reference model and none overlapping boxes from the other model(s) are chosen as output candidates. Thus, overlapping boxes from other models are eliminated using a threshold value. The second scheme eliminates overlaps by selecting the box from the models with the highest confidence. And the third criterion averages overlapping boxes from all models in the ensembles. Results from all these three scenarios are reported and compared.

4 Experiment

4.1 Datasets

Four datasets were considered in this experiment, namely ICDAR2013 [10], ICDAR2015 [9], MIDV-500 [1] and P&ID [18] datasets.

ICADAR2013, which is a focused scene text localisation dataset, consist of 229 samples in train set and 233 samples in test set. Samples contain random images of text in sign post and written text from different scenes and backgrounds. Generally, text are focused in the image center.

ICDAR2015, which contains an incidental scene text localisation dataset, consist of 1500 samples with 1000 images for training and 500 samples for testing. Samples are challenging with cluttered scene and texts of different shape, size and orientation.

Meanwhile, MIDV-500 dataset consist of 15000 card samples from 50 countries from around the world. Card samples were generated from a single card from each country by taking a picture of the card in different view angle, cluttered environment, lightening condition and camera. Each sample creates a more challenging task for card detection, face localisation and text field OCR.

Finally, the P&ID dataset consists of engineering drawings cluttered with text and symbols. Thus P&IDs have text with varying fonts, font sizes and different orientations. The images are large (approximately 5239 by 7417) with uniform white background. The texts here are symbol names and standard acronyms which are mostly alpha-numeric. The models were evaluated on 155 P&ID drawings with 39538 ground-truth boxes in total.

4.2 Experimental Set-up

For the first experiment, we conducted a text localisation test using EAST. We used a pre-trained model from [4] (which we refer to as EAST-1). We also trained another EAST model from scratch on ICDAR2015 dataset only using the same protocol (also referred to as EAST-2.). A pre-trained CRAFT model was also evaluated on these datasets. We report the precision, recall and F1-score

[4] https://github.com/argman/EAST

on ICDAR2013, 2015 test sets using the official evaluation script from [5]. The evaluation was done without any samples pre-processing.

Our second set of experiments consisted of text localisation on MIDV-500. This was more challenging because ground truths are not available for all card samples text fields. However, the base card from which the samples were generated contains text annotation of the data fields. The dataset also provided a card annotation box (only) for all samples. To work around this, we used the provide card quadruple box to crop the card images from the samples. Then we applied perspective transform on the cropped card to neutralise any orientation change. Finally, cards are resized to the size of the base card in the categories. Then, ground truth from the base card is overlaid on the card crop serving as a pseudo ground truth. The number of text fields across the card type differs between 2-11. For evaluation, we discard partially occluded card but allowed for irregularly illuminated cards. In total, 12000 card samples were used to evaluate the models. A point to note is that no model was trained on this dataset. We used models trained from our initial detection experiments described earlier. Again, we ignored picture annotations, signature annotations and text detected by models that are not part of the annotated fields. In this case, false-negative detections were considered as missed field boxes and false positives are detections that are below the IOU threshold (0.5).

The third set of experiments were carried out on text detection in P&IDs. Again, no model was trained on this dataset hence, all images were used for evaluation. In these experiments, detected bounding boxes from the legend sections were discarded as no ground truth was available for comparison. IOU threshold was kept at 0.4 and we report the precision, recall and f1-score for each model.

5 Results and Discussion

Table 1 and 2 shows the quantitative performances of the methods described in section 3 on different datasets. We employed precision, recall and the f1-score to compare these performances. For ICDAR2013 and ICDAR2015, the official evaluation script from[6] was used. MIDV-500 and P&ID required us to write a separate evaluation script to meet the experimental requirements. Figures 1 and 2 shows sample detections from models.

Table 1: Text localisation results. The highest value for the selected metrics in each dataset is highlighted in bold

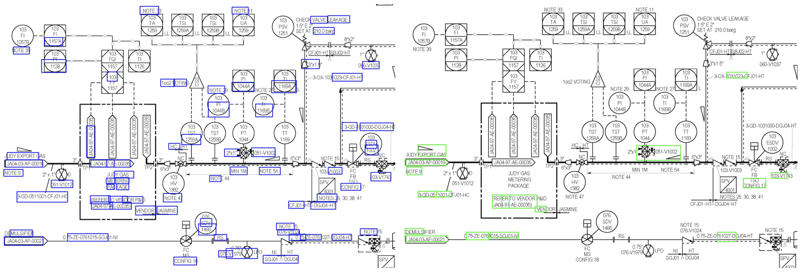
	ICDAR 2013				ICDAR 2015				MIDV-500				P&ID			
	East-1	East-2	CRAFT	Tesseract	East-1	East-2	CRAFT	Tesseract	East-1	East-2	CRAFT	Tesseract	East-1	East-2	CRAFT	Tesseract
Precision	0.88	0.80	**0.90**	0.41	0.84	0.84	**0.85**	0.05	0.49	**0.59**	0.35	0.32	**0.52**	0.50	0.45	0.13
Recall	**0.93**	0.76	0.92	0.27	0.77	0.77	**0.79**	0.04	**0.51**	0.41	0.50	0.15	**0.46**	0.34	0.12	0.20
F1-score	0.90	0.78	**0.91**	0.33	0.81	0.80	**0.82**	0.04	**0.50**	0.49	0.34	0.21	**0.49**	0.40	0.18	0.15

[5] https://rrc.cvc.uab.es/?ch=4&com=downloads

[6] https://rrc.cvc.uab.es/?ch=14

EAST-1 and EAST-2 performances are identical on ICDAR2015. However, EAST-1 showed better recall on ICDAR2013 which is no surprise given that the model was trained on a combined dataset conformed of ICDAR2013 and ICDAR2015. Again, ICDAR2013 contains focused text images and a smaller test set hence, performance from EAST-1 was high. On the other hand, this was not the case for EAST-2 where the performance dropped slightly. The results from EAST-2 on ICDAR2013 indicated the true performance of the model across a different dataset. The CRAFT and EAST models also performed well on ICDAR2015 with CRAFT obtaining slightly better precision, recall and f1-score among all models.

Tesseract performance was poor, particularly on ICDAR2015. The experiments highlighted the limitations of Tesseract in scene detection while the poor results on focused text detection indicated its reliance on traditional approaches to isolate text from background. The tesseract detector also had a lot of false detection and missed texts. Tesseract struggles because it relies on preprocessing and in most cases, there is no clear text-background separation in samples.



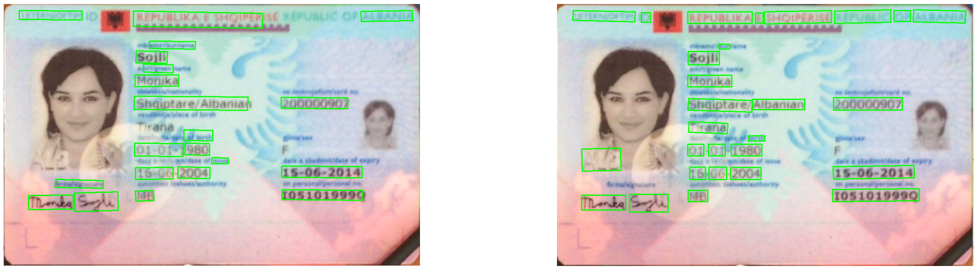
(a) EAST-1 (b) CRAFT

Fig. 1: Sample detection from cross-section of a P&ID.

More interesting to this research are the results from MIDV-500 and P&ID datasets which shows across domain performance. The results indicated a considerable drop in performance across models. In particular, on MIDV-500, apart from missing on complete word detection, some fields were detected halfway or a single field may be detected with two separate bounding boxes. The effect of text orientation is not significant here as cards were transformed to a natural horizontal position before detection but some text boxes appeared with a skewed orientation such as in Figure 2a (EAST-1). While the polygon points returned help with oriented texts, these conditions could contribute to bounding box shape distortion. These conditions can push IoU down and may have reduced the number of positive boxes detected. Furthermore, different text languages on cards have also contributed to low performance. For instance, the Chinese ID card is written in the Chinese alphabet which is drastically different from the mostly English training examples.

Table 2: Text localisation results on P&ID from outputs ensemble.

	Coalesce			Confidence			Average		
Models	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
EAST-1 + EAST-2	0.50	0.53	0.51	**0.52**	0.62	0.57	0.52	0.45	0.48
EAST-1 + CRAFT	0.51	0.6	0.55	**0.52**	0.62	0.56	0.52	0.45	0.48
EAST-1 + Tesseract	**0.52**	0.6	0.56	0.51	0.62	0.56	0.52	0.45	0.48
EAST-1 + EAST-2 + CRAFT	0.49	0.67	0.57	0.52	0.77	0.62	0.52	0.45	0.48
EAST-1 + EAST-2 + Tesseract	0.5	0.67	0.57	0.51	0.77	0.61	0.52	0.45	0.48
EAST-1 + EAST-2 + CRAFT +Tesseract	0.49	**0.79**	**0.61**	0.51	**0.87**	**0.64**	0.52	0.45	0.48
EAST-2 + CRAFT	0.49	0.48	0.48	0.49	0.5	0.5	0.5	0.34	0.4
EAST-2 + Tesseract	0.5	0.47	0.48	0.48	0.5	0.49	0.49	0.34	0.4
EAST-2 + CRAFT + Tesseract	0.49	0.62	0.54	0.48	0.67	56	0.49	0.34	0.4
CRAFT + Tesseract	0.45	0.19	0.27	0.45	0.21	0.28	0.46	0.12	0.19



(a) EAST-1

(b) CRAFT

Fig. 2: Sample detection from models.

Similarly, models performances dropped significantly on P&ID. This can be attributed to the challenging nature of the domain. Apart from text orientation, size and fonts, the resolution of diagrams in relation to the text size negatively affected performance. Figure 1a shows a sample detection from EAST-1 which was by far the best model on P&ID. In comparison, majority of the text within symbols were missed by Tesseract. Moreover, there were diagrams that Tesseract missed all none horizontal texts.

With ensemble outputs, there is a noticeable performance improvement. Overall, the best results from the P&ID are obtained when confidence is used as the voting criterion. That said, in terms of individual performances, no ensemble setup out-performed its peers in all three metrics. For instance, the best recall was obtained when all four models are combined based on confidence however, this was at the expense of a slight dip in precision.

6 Conclusion

In this paper, we analysed the performances of state-of-the-art text detection algorithms. Experiments were conducted to compare models trained on public datasets with varying text detection scenarios. Furthermore, the performances of these models were evaluated across the datasets using different metrics. The results indicated that despite the models trained on challenging scene text detection tasks, the performance dropped significantly when tested on text field detection on identity documents, and P&ID (Processing and Instrumentation Diagrams) with varying text fonts and background conditions. Hence, this highlighted some of the limitations of established text detection models in generalising to different text detection scenarios and domains. Future direction for this work is to investigate ensemble learning in the text detection across domains.

References

1. Vladimir Viktorovich Arlazarov, Konstantin Bulatovich Bulatov, Timofey Sergeevich Chernov, and Vladimir Lvovich Arlazarov. Midv-500: a dataset for identity document analysis and recognition on mobile devices in video stream. 2019.
2. Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
3. Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, et al. Pp-ocr: A practical ultra lightweight ocr system. *arXiv preprint arXiv:2009.09941*, 2020.
4. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT, 2016.
5. Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
6. Tong He, Weilin Huang, Yu Qiao, and Jian Yao. Accurate text localization in natural image with cascaded convolutional text network. *arXiv preprint arXiv:1603.09423*, 2016.
7. Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, 2015.
8. L. Jamieson, C. F. Moreno-Garcia, and E. Elyan. Deep learning for text detection and recognition in complex engineering diagrams. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, 2020.
9. Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2015.
10. Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. In *2013 12th International Conference on Document Analysis and Recognition*. IEEE, 2013.
11. Chen-Yu Lee, Anurag Bhardwaj, Wei Di, Vignesh Jagadeesh, and Robinson Piramuthu. Region-based discriminative feature pooling for scene text recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.
12. Chen-Yu Lee and Simon Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In *Proceedings of the IEEE Conference on CVPR*, 2016.
13. Minghui Liao, Baoguang Shi, and Xiang Bai. Textboxes++: A single-shot oriented scene text detector. *IEEE transactions on image processing*, 2018.
14. Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11474–11481, 2020.

15. Xiyan Liu, Gaofeng Meng, and Chunhong Pan. Scene text detection and recognition with advances in deep learning: a survey. *International Journal on Document Analysis and Recognition (IJDAR)*, 2019.
16. Shangbang Long, Xin He, and Cong Yao. Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision*, August 2020.
17. Anand Mishra, Karteek Alahari, and CV Jawahar. Top-down and bottom-up cues for scene text recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012.
18. Carlos Moreno-García, Eyad Elyan, and Chrisina Jayne. New trends on digitisation of complex engineering drawings. *Neural Computing and Applications*, June 2018.
19. Yi-Feng Pan, Xinwen Hou, and Cheng-Lin Liu. A hybrid approach to detect and localize texts in natural scene. *IEEE transactions on image processing*, 2010.
20. Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013.
21. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.
22. Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 2014.
23. Baoguang Shi, Xiang Bai, and Serge Belongie. Detecting oriented text in natural images by linking segments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.
24. Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2016.
25. Baoguang Shi, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Robust scene text recognition with automatic rectification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.
26. Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification. *IEEE transactions on pattern analysis and machine intelligence*, 2018.
27. Ray Smith. Tesseract ocr modernizationefforts.pdf. https://github.com /tesseract-ocr/docs/blob/master/das_tutorial2016/6ModernizationEfforts.pdf, 2016.
28. Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *European conference on computer vision*. Springer, 2016.
29. Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *2011 International Conference on Computer Vision*. IEEE, 2011.
30. Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, 2015.
31. Cong Yao, Xiang Bai, Nong Sang, Xinyu Zhou, Shuchang Zhou, and Zhimin Cao. Scene text detection via holistic, multi-channel prediction. *arXiv preprint arXiv:1606.09002*, 2016.
32. Cong Yao, Xiang Bai, Baoguang Shi, and Wenyu Liu. Strokelets: A learned multi-scale representation for scene text recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.
33. Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017.