



# MobText: A Compact Method for Scene Text Localization

Luis Gustavo Lorgus Decker<sup>1a</sup><sup>b</sup>, Allan Pinto<sup>1</sup>, Jose Luis Flores Campana<sup>1</sup>, Manuel Cordova Neira<sup>1</sup>, Andreza A. dos Santos<sup>1</sup>, Jhonatas S. Conceição<sup>1</sup>, Marcus A. Angeloni<sup>2</sup>, Lin Tzy Li<sup>2</sup> and Ricardo da S. Torres<sup>3</sup><sup>c</sup>

<sup>1</sup>RECOD Lab., Institute of Computing, University of Campinas, 13083-852, Brazil

<sup>2</sup>AI R&D Lab, Samsung R&D Institute Brazil, 13097-160, Brazil

<sup>3</sup> Department of ICT and Natural Sciences, Norwegian University of Science and Technology (NTNU), Ålesund, Norway  
luisgustavo.decker@gmail.com, allan.pinto@ic.unicamp.br, j209820@dac.unicamp.br, {manuel.diduchi, andi.apsantos, jhonatassantosdejesus17}@gmail.com, {m.angeloni, lin.tzy.li}@samsung.com, ricardo.torres@ntnu.no

Keywords: Scene Text Detection, Mobile devices, Object Detector Networks, MobilenetV2, Single Shot Detector


Abstract: Multiple research initiatives have been reported to yield highly effective results for the text detection problem. However, most of those solutions are very costly, which hamper their use in several applications that rely on the use of devices with restrictive processing power, like smartwatches and mobile phones. In this paper, we address this issue by investigating the use of efficient object detection networks for this problem. We propose the combination of two light architectures, MobileNetV2 and Single Shot Detector (SSD), for the text detection problem. Experimental results in the ICDAR'11 and ICDAR'13 datasets demonstrate that our solution yields the best trade-off between effectiveness and efficiency and also achieved the state-of-the-art results in the ICDAR'11 dataset with an f-measure of 96.09%.

## 1 Introduction

Reading text in images is still an open problem in computer vision and image understanding research fields. In fact, this problem has attracted a lot of attention of these communities due to large number of modern applications that can potentially benefit from this knowledge, such as self-driving vehicles (Yan et al., 2018; Zhu et al., 2018), robot navigation, scene understanding (Wang et al., 2018), assistive technologies (Yi et al., 2014), among others.

Several methods have been recently proposed in the literature towards localizing textual information in scene images. In general, the text reading problem is divided into two separated tasks, localization and recognition, in which the former seeks to localize delimited candidate regions that contain textual information, while the second is responsible for recognizing the text inside the candidate regions found during

<sup>a</sup>Part of results presented in this work were obtained through the “Algoritmos para Detecção e Reconhecimento de Texto Multilíngue” project, funded by Samsung Eletrônica da Amazônia Ltda., under the Brazilian Informatics Law 8.248/91.

<sup>b</sup> <https://orcid.org/0000-0002-6959-3890>

<sup>c</sup> <https://orcid.org/0000-0001-9772-263X>



Figure 1: Examples of textual elements with different font sizes and styles.

localization task. In both tasks, the inherent variability of a text (e.g., size, color, font style, background clutter, and perspective distortions), as illustrated in Fig. 1, makes text reading a very challenging problem.

Among the approaches for localizing texts in images, the deep-learning-based techniques are the most promising strategy to reach high detection accuracy.

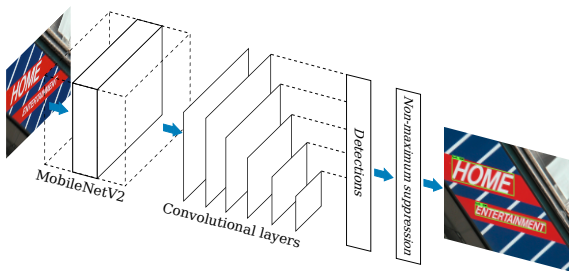


Figure 2: Overview of the proposed method for text localization.

He et al., for example, presented a novel technique for scene text detection by proposing a Convolutional Neural Network (CNN) architecture (He et al., 2016) that focuses on extracting text-related regions and specific characteristics of text. The authors introduced a deep multi-task learning mechanism to train the Text-CNN efficiently, where each level of the supervised information (text/non-text label, character label, and character mask) is formulated as a learning task, besides a pre-processing method which enhances the contrast of small-size region improving the local stability of text regions.

Although the proposed CNN presented a reasonable efficiency in detecting candidate regions, with a processing time of about 0.5 seconds per image, the pre-processing step requires about 4.1 seconds per image, which may prevent a real-time detection.

Another venue that may render outstanding results in terms of effectiveness consists in combining different deep learning architectures to benefit from complementary information to make a better decision. In this vein, (Zhang et al., 2016) introduced an approach based on two Fully Convolutional Network (FCN) architectures for predicting a salient map of text regions in a holistic manner (named as *Text-Block FCN*), and also for predicting the centroid of each character. Similarly, Tang et al. (Tang and Wu, 2017) proposed an ensemble of three modified VGG-16 networks: the first extracts candidate text regions (CTR); the second network refines the coarse CTR detected by the first model, segmenting them into text; and finally, the refined CTR are served to a classification network to filter non-text regions and obtain the final text regions. The CTR extractor network is a modified VGG-16 that, in the training process, receives the edges of the text as supervisory information in the first blocks of convolutional layers and the segmented text regions in the last blocks. Both strategies present issues in terms of computational efficiency that could make their use unfeasible in restrictive computing scenarios (e.g., mobile devices).

Towards having a truthfully single stage text detection, Liao et al. (Liao et al., 2018) proposed an end-to-end solution named TextBoxes++, which handles arbitrary orientation of word bounding boxes, whose architecture inherits from the VGG-16. Similarly to TextBoxes++, Zhu et al. proposed a deep learning approach (Zhu et al., 2018) also based on the VGG-16 architecture, but for detecting text-based traffic sign. Both techniques presented outstanding detection rates, though rely on the VGG-16 architecture, which could be considered inadequate for restrictive computing scenarios due its model size. In contrast, lighter CNN architectures, such as MobileNet (Howard et al., 2017), present a very competitive alternative for this scenario, with a model size of 4.2 millions of parameters and the FLOPS of 569 millions, for instance.

With those remarks, we propose a novel method for text localization considering efficiency and effectiveness trade-offs. Our approach combines two light architectures that were originally proposed for object detection – MobileNetV2 (Sandler et al., 2018) and SSD (Liu et al., 2016) – and adapts them to our problem. The main contributions of this paper are: (i) the proposal of an effective method for text localization task in scene images, which presented better or competitive results (when compared with state-of-the-art methods) at a low computational costs in terms of model size and processing time; and (ii) a comparative study, in the context of text localization, comprising widely used CNN architectures recently proposed for object detection.

## 2 Proposed Method

Fig. 2 illustrates the overall framework of our approach for text localization, which uses MobileNetV2 as feature extractor and then SSD (convolutional layers) as multiple text bonding boxes detector. Next, We will detail the CNN architectures used, then explain the learning mechanism adopted for finding a proper CNN model for the problem.

### 2.1 Characterization of text regions with MobileNetV2

The MobileNetV2 is a new CNN specifically designed for restrictive computing environments that includes two main mechanisms for decreasing the memory footprints and the number of operations while keeping the effectiveness of its precursor architecture, the MobileNet (Sandler et al., 2018): the linear bottlenecks and the inverted residuals.

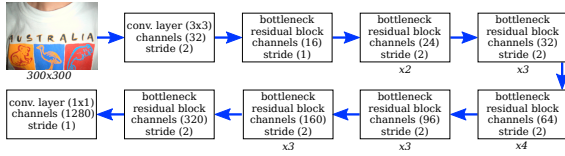


Figure 3: MobileNetV2 architecture used in this work and its parameters. More detail on Bottleneck residual block in (Sandler et al., 2018).

Fig. 3 shows the MobileNetV2 architecture used to characterize text candidate regions. The *bottleneck residual block* implements the optimization mechanisms aforementioned considering the convolutional operations with a kernel of size  $3 \times 3$ . The first bottleneck block uses an expansion factor of 1, while the remaining blocks use an expansion factor of 6, as suggested by Sandler et al. (Sandler et al., 2018).

## 2.2 Detecting multiple text bounding boxes via SSD

The localization of text regions in scene images is challenging due to inherent variability of the text, such as size, color, font style, and distortions. The text localization should handle multiple scales and bounding boxes with varying aspect ratio. Although several authors consider the image pyramid for performing multi-scale detection, it is quite costly, which may be impractical in a restrictive computing scenario. Thus, we use the Single Shot detector (SSD) framework (Liu et al., 2016), a state-of-the-art method for object detection. The SSD approach includes a feature pyramid mechanism that allows the identification of text regions in multiple scales. Specifically, in the framework, the authors adopt a top-down fusion strategy to build new features with strong semantics while keeping fine details. Text detections are performed based on multiple new constructed features respectively during a single forward pass. All detection results from each layer are refined by means of a non-maximum suppression (NMS) process (Neubeck and Gool, 2006).

## 2.3 Using Linear Bottlenecks and Inverted Residuals Bottlenecks for Memory efficiency

Besides the use of depthwise separable convolutions operations, MobileNetV2 introduced linear bottlenecks in the convolutional blocks. This reduces the number of parameters of a neural network and captures the low-dimensional subspace, supposing that such low-dimensional subspace is embedded in

a manifold formed by a set of activation tensors. In (Sandler et al., 2018), Sandler et al. showed empirical evidences that the use of linear layers is important to prevent non-linearity added from destroying information. Experiments conducted by the authors showed that non-linear bottlenecks, built with rectified linear units, can decrease the performance significantly in comparison with linear bottlenecks. By using the idea of Inverted Residual bottlenecks, the authors achieved better memory use, reducing a significant amount of computation. We follow this idea in this paper.

## 2.4 Learning

The main decisions we took in the learning phase of our network are described below.

**Objective function** Similar to (Liu et al., 2016), we use a multi-task loss function to learn the bounding boxes locations and text/non-text predictions (Eq. 1). Specifically,  $x_{ij}$  indicates a match ( $x_{ij} = 1$ ) or non-match ( $x_{ij} = 0$ ) between  $i$ -th default bounding boxes,  $j$ -th ground-truth bounding boxes;  $N$  is the number of matches; and the  $\alpha$  parameter is used to weight the localization loss ( $\mathcal{L}_{loc}$ ) and the confidence loss ( $\mathcal{L}_{conf}$ ). The used loss function can be defined as:

$$\mathcal{L}(x, c, l, g) = \frac{1}{N} (\mathcal{L}_{conf}(x, c) + \alpha \mathcal{L}_{loc}(x, l, g)) \quad (1)$$

We adopted the smooth L1 loss (Girshick, 2015) for  $\mathcal{L}_{loc}$  between the predicted box ( $l$ ) and the ground truth box ( $g$ ), and a sigmoid function for  $\mathcal{L}_{conf}$ . Plus, we consider  $\alpha = 1$  in the same fashion as (Girshick, 2015).

**Hard example mining** The hard example miner is a mechanism used to prevent imbalances between negative and positive examples in the training phase. During the search for text during the training, we usually have several non-text bounding boxes and few text bounding boxes. To mitigate the training with imbalanced data, we sort the negative bounding boxes according to their confidence, selecting the negative samples with higher confidence value, considering a ratio proportion of 3:1 with the positive samples.

## 3 Experimental Protocol

This section presents the datasets, metrics, and protocols used for evaluating the proposed method.

### 3.1 Datasets

We evaluated the proposed methods in two datasets widely used for evaluating text localization methods: ICDAR'11 Karatzas et al. (2011), that contains 551 digitally created images, such as headers, logos, captions, among others, and ICDAR'13 Karatzas et al. (2013), containing 462 born-digital or scene text images (captured under a wide variety, such as blur, varying distance to camera, font style and sizes, color, texture, etc). We also used the SynthText Gupta et al. (2016) dataset to help training our network due to the small size of the ICDAR's datasets. We have not used ICDAR'15 Karatzas et al. (2015) and some other newer datasets, because our method is not tailored to the prediction of oriented bounding boxes, which is required to handle such multioriented datasets.

### 3.2 Evaluation Metrics

**Effectiveness** We evaluated the effectiveness of the methods in terms of recall, precision, and f-measure. Here, we consider a correct detection (true positive) if the overlap between the ground-truth annotation and detected bounding box, which is measured by computing the intersection over union, is greater than 50%. Otherwise, the detected bounding box is considered an incorrect detection.

**Efficiency** The efficiency aspects considered both the processing time and the disk usage (in MB). We used the GNU/Linux *time* command to measure the processing time, while the disk usage considered the size of the learned models. All experiments were performed considering a Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz with 12 cores, a Nvidia GTX 1080 TI GPU, and 64GB of RAM.

### 3.3 Evaluation Protocols

The experiments were divided into three steps: training, fine-tuning, and test. For the training step, we used four subsets of the SynthText dataset. This dataset comprises of images with synthetic texts added in different backgrounds and we selected samples of the dataset considering 10 (9.25%), 20 (18.48%), and 30 (27.71%) images per background, then finally the whole dataset. The resulting subsets were again divided into train and validation, using 70% for training and 30% for validation. Using these collections, we trained a model with random initialization parameters for 30 epochs. For the fine-tuning step, we took the model trained in SynthText

and continued this training using ICDAR'11 or ICDAR'13 training subsets, stopping when we reached 2000 epochs. The number of epochs was defined empirically. Finally, for the test step, we evaluated each fine-tuned model in the test subset of ICDAR'11 or ICDAR'13.

**Experimental Setup** We conducted the training of the proposed method considering a single-scale input, and therefore, all input images were resized to  $300 \times 300$  pixels. The training phase was performed using a batch size of 24 and we used the RMSprop optimizer (Tieleman and Hinton, 2012) with a learning rate of  $4 \times 10^{-3}$ . We also use the regularization L2-norm, with a  $\lambda = 4 \times 10^{-5}$ , to prevent possible overfitting.

### 3.4 State-of-the-art object detection methods for comparison

This section provides an overview of the chosen methods for comparison purpose. For a fair comparison, we selected recent approaches specifically designed for a fast detection, including SqueezeDet and YOLOv3. We also use methods for text localization that presents good compromise among effectiveness and efficiency as baselines, which are briefly described in this section.

**TextBoxes** This method consists of a Fully Convolutional Network (FCN) adapted for text detection and recognition (Liao et al., 2017). This network uses the VGG-16 network as feature extractor followed by multiple output layers (text-boxes layers), similar to SSD network. At the end, the Non-maximum suppression (NMS) process is applied to the aggregated outputs of all text-box layers.

**TextBoxes++** This method extends the TextBoxes method (Liao et al., 2017) toward detecting arbitrary-oriented text, instead of only (near)-horizontal bounding boxes (Liao et al., 2018). TextBoxes++ also brings improvements in the training phase, which leads to a further performance boost, in terms of accuracy, especially for detecting texts in multiple scales. In this work, the authors combine the detection scores of CRNN recognition method (Shi et al., 2017) with the TextBoxes++ to improve the localization results and also to have an end-to-end solution.

**SSTD** Single-shot text detector proposed by He et al. (He et al., 2017) designed a natural scene text detector that directly outputs word-level



bounding boxes without post-processing, except for a simple NMS. The detector can be decomposed into three parts: a convolutional component, a text-specific component, and a box prediction component. The convolutional and box prediction components are inherited from the SSD detector (Liu et al., 2016) and the authors proposed a text-specific component which consists of a text attention module and a hierarchical inception module.

**SqueezeDet** This network was proposed to detect objects for the autonomous driving problem, which requires a real-time detection (Wu et al., 2017). The SqueezeDet contains a single-stage detection pipeline, which comprises three components: (i) a FCN responsible for generating the feature map for the input images; (ii) a convolutional layer responsible for detecting, localizing, and classifying objects at the same time; and (iii) the non-maximum suppression (NMS) method, which is applied to remove the overlapped bounding boxes.

**YOLOv3** This is a convolutional network originally proposed for the object detection problem (Redmon and Farhadi, 2018). Similarly to SSD network, the YOLOv3 predicts bounding boxes and class probabilities, at the same time.

## 4 Results

This section presents the experimental results of the proposed method (SSD-MobileNetV2) and a comparison with the state-of-the-art methods for text localization. Table 1 shows the results for the evaluated methods considering the ICDAR'11 dataset. In this case, the SSD-MobileNetV2 method achieved the best results with Precision, Recall, and F-measure values of 97.40%, 94.81%, and 96.09%, respectively. On the other hand, the SqueezeDet network presented the lowest Precision and F-measure among the evaluated methods (56.36% and 66.01%, respectively). In turn, the TextBoxes achieved the lowest results of Recall (71.93%).

With regard to ICDAR'13 dataset, the SSTD methods presented the highest Recall (82.19%), and F-measure (86.33%), while the YOLOv3 reached the best results in terms of Precision (Table 1). Note, however, that the SSD-MobileNetV2 yields very competitive results for this dataset as well, in terms of Precision.

As we could observe, the proposed approach presented some difficult in localizing scene text in the ICDAR'13 dataset. In comparison with results achieved

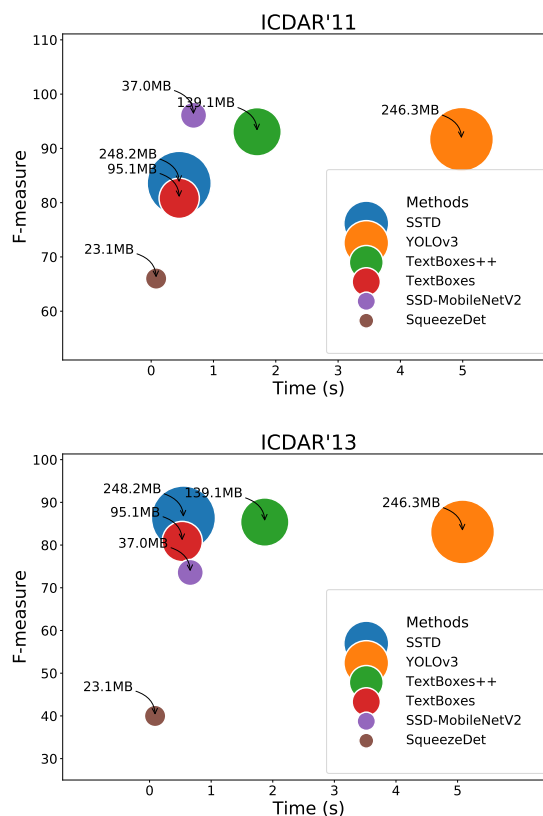


Figure 4: Comparison results among the evaluated methods considering aspects of efficacy and efficiency.

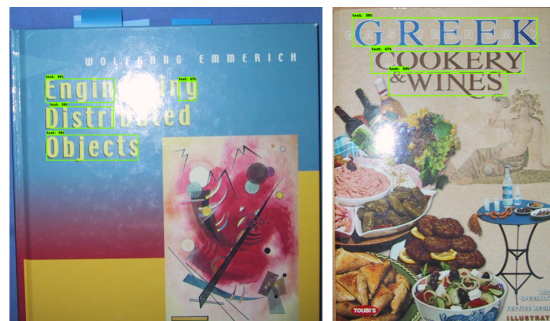


Figure 5: Two high resolution examples of ICDAR'13 dataset with both medium-sized text (detected by our method) and small-sized (not detected)

for the ICDAR'11, the precision and recall rates decreased 9.36 and 31.61 percentage points, respectively, which suggest that our network did not localize several candidate regions containing texts.

To understand the reasons that led the proposed method to have this difficult in localizing text for the ICDAR'13 datasets, we performed an analysis of failure cases taking into account the relative area of missed bounding boxes. Fig. 4 presents a box-plot graph that shows the distribution of the relative area

Table 1: Comparison of effectiveness among the evaluated deep learning-based methods for the ICDAR’11 and ICDAR13 dataset.

Datasets Methods	ICDAR11			ICDAR13		
	P (%)	R (%)	F (%)	P (%)	R (%)	F (%)
SSD-MobilenetV2	97.40	94.81	96.09	88.38	66.67	76.00
SSTD	89.28	78.53	83.56	90.91	82.19	86.33
TextBoxes	92.15	71.93	80.80	88.84	74.16	80.83
TextBoxes++	95.76	90.51	93.06	90.49	80.82	85.38
YOLOv3	94.27	89.21	91.67	92.01	75.71	83.07
SqueezeDet	56.36	79.66	66.01	29.41	62.47	39.99

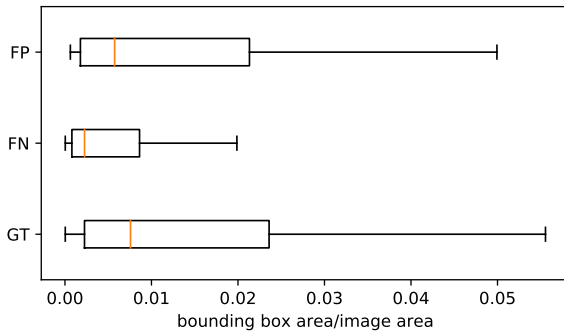


Figure 6: Comparison among distributions of relative areas of bounding boxes from Ground-Truth (GT), False Negatives (FN) cases, and False Positive (FP) cases. We omitted the points considered outliers for a better visualization.

of bounding boxes (i.e., ratio of bounding box area to image area) for the ground-truth, false positive cases, and false negative cases.

As we can observe, the missed bounding boxes (false negative cases) have a small relative area. More precisely, 75% of false negative cases (third quartile of FN box-plot) have a relative area up to 0.01 and correspond to 50% of the bounding box present in the ground-truth (median of GT box-plot). This results suggest to us that high resolution images with relatively small text (see Fig. 5) are specially challenging to our method. To overcome this limitation, future investigations can be conducted to devise an architecture to better localize bounding boxes with multiple scales such as Feature Pyramid Networks (FPNs), as proposed by (Lin et al., 2017).

In term of the efficiency of the presented methods, Fig. 4 summarizes the results considering the metrics used to assess the effectiveness of the evaluated methods, in terms of F-measure, along with the metrics for measuring the efficiency of those methods, considering the ICDAR’11 and ICDAR’13 datasets.

Regarding the efficiency (processing time and disk usage), the proposed method (SSD-MobilenetV2) yielded very competitive results, taking only 0.45

and 0.55 seconds per image, considering the ICDAR’11 and ICDAR’13, respectively. Comparing SSD-MobilenetV2 with the baseline methods originally proposed for text localization (TextBoxes, TextBoxes++, SSTD), the proposed method presented the very competitive results with a processing time of 0.67 seconds per image and with disk usage of about 37.0MB. In contrast, the most effective baseline methods, the SSTD and TextBoxes++ networks, presented competitive and worse results in terms of effectiveness and processing time, respectively, in comparison with the proposed method. Regarding the disk usage, the SSD-MobileNetV2 also presented the best balance between accuracy and model size.

Now, when compared with the state-of-the-art approaches for object detection, the proposed method also presented competitive results. In this case, the fastest approach for text localization was the SqueezeDet network, which takes about 0.1 seconds per image, on average. However, when we take into account the trade-off between efficiency and effectiveness, we can safely argue that the proposed method presented a better compromise between these two measures. Fig. 7 and Fig. 8 provide some cases of success (first column) and of failure of the proposed method for the ICDAR’11 and ICDAR’13 datasets. For the first (see Fig), the proposed method was able to localize textual elements with different font styles and even multi-oriented texts. For the latter, the proposed method was able to localize text in several contexts such as in airport signs, traffic signs, text in objects, among others. Failures are due to compression artifacts, the high similarity between the background and the text colors, small texts, and lighting conditions.

## 5 Conclusions

How to perform efficient and effective text detection in scene images in restrictive computing environments? To address that research problem, we



Figure 7: Examples of success (first row) and failure (second row) cases of the proposed approach for the ICDAR'11 dataset. Green bounding boxes indicate the regions correctly localized (true positives cases), while red bounding boxes show candidate regions were not detected by our method (false negatives cases).

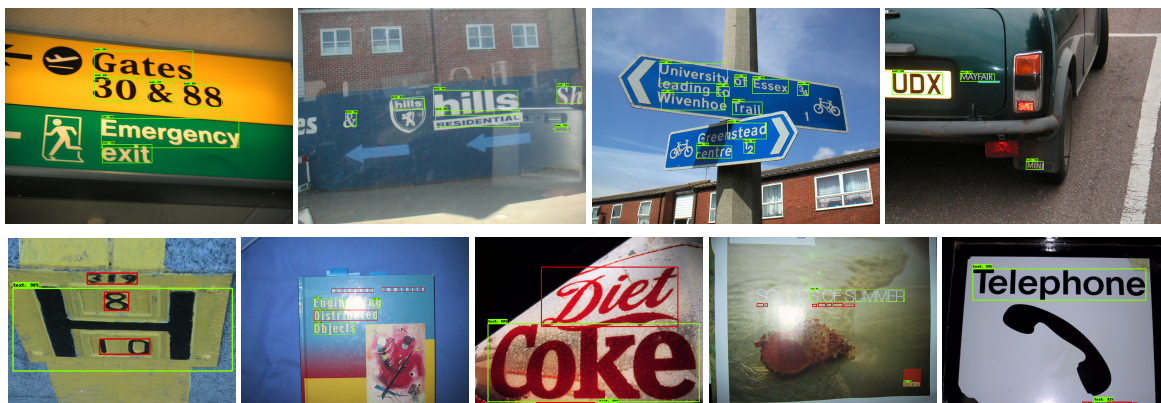


Figure 8: Examples of success (first row) and failure (second row) cases of the proposed approach for the ICDAR'13 dataset. Green bounding boxes indicate the regions correctly localized (true positives cases), while red bounding boxes show candidate regions were not detected by our method (false negatives cases).

presented a new method based on the combination of two light architectures, MobileNetV2 and Single Shot Detector (SSD), which yielded better or comparable effectiveness performance when compared with state-of-the-art baselines despite having a low processing time and small model size. Compared with other object detector solutions, our methods is the most promising. Our findings disagree with the discussion provided in (Ye and Doermann, 2015), as we demonstrated that adapting object detector networks for text detection is a promising research venue.

Future research efforts will focus on better char-

acterizing both small and large candidate regions to localize text in multiple scales such as Feature Pyramid Networks.

## REFERENCES

Girshick, R. (2015). Fast r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*.

Gupta, A., Vedaldi, A., and Zisserman, A. (2016). Synthetic data for text localisation in natural images. In *2016 IEEE*

- Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2315–2324.
- He, P., Huang, W., He, T., Zhu, Q., Qiao, Y., and Li, X. (2017). Single shot text detector with regional attention. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3066–3074.
- He, T., Huang, W., Qiao, Y., and Yao, J. (2016). Text-attentional convolutional neural network for scene text detection. *IEEE Transactions on Image Processing*, 25(6):2529–2541.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861.
- Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V. R., Lu, S., Shafait, F., Uchida, S., and Valveny, E. (2015). Icdar 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1156–1160.
- Karatzas, D., Mestre, S. R., Mas, J., Nourbakhsh, F., and Roy, P. P. (2011). Icdar 2011 robust reading competition - challenge 1: Reading text in born-digital images (web and email). In *2011 International Conference on Document Analysis and Recognition*, pages 1485–1490.
- Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., Bigorda, L. G. i., Mestre, S. R., Mas, J., Mota, D. F., Almazan, J. A., and de las Heras, L. P. (2013). Icdar 2013 robust reading competition. In *Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, ICDAR '13*, pages 1484–1493, Washington, DC, USA.
- Liao, M., Shi, B., and Bai, X. (2018). Textboxes++: A single-shot oriented scene text detector. *IEEE Transactions on Image Processing*, 27(8):3676–3690.
- Liao, M., Shi, B., Bai, X., Wang, X., and Liu, W. (2017). Textboxes: A fast text detector with a single deep neural network. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 4161–4167.
- Lin, T., Dollr, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). SSD: Single shot multibox detector. In Leibe, B., Matas, J., Sebe, N., and Welling, M., editors, *Computer Vision – ECCV 2016*, pages 21–37, Cham. Springer International Publishing.
- Neubeck, A. and Gool, L. V. (2006). Efficient non-maximum suppression. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 850–855.
- Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *CoRR*, abs/1804.02767.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4510–4520.
- Shi, B., Bai, X., and Yao, C. (2017). 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*, 39(11):2298–2304.
- Tang, Y. and Wu, X. (2017). Scene text detection and segmentation based on cascaded convolution neural networks. *IEEE Transactions on Image Processing*, 26(3):1509–1520.
- Tieleman, T. and Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31.
- Wang, L., Wang, Z., Qiao, Y., and Van Gool, L. (2018). Transferring deep object and scene representations for event recognition in still images. *International Journal of Computer Vision*, 126(2):390–409.
- Wu, B., Iandola, F., Jin, P. H., and Keutzer, K. (2017). SqueezeDet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 446–454.
- Yan, C., Xie, H., Liu, S., Yin, J., Zhang, Y., and Dai, Q. (2018). Effective uighur language text detection in complex background images for traffic prompt identification. *IEEE Trans. Intelligent Transportation Systems*, 19(1):220–229.
- Ye, Q. and Doermann, D. S. (2015). Text detection and recognition in imagery: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(7):1480–1500.
- Yi, C., Tian, Y., and Arditi, A. (2014). Portable camera-based assistive text and product label reading from hand-held objects for blind persons. *IEEE/ASME Transactions on Mechatronics*, 19(3):808–817.
- Zhang, Z., Zhang, C., Shen, W., Yao, C., Liu, W., and Bai, X. (2016). Multi-oriented text detection with fully convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Zhu, Y., Liao, M., Yang, M., and Liu, W. (2018). Cascaded segmentation-detection networks for text-based traffic sign detection. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):209–219.