Unambiguous Text Localization and Retrieval for Cluttered Scenes

Xuejian Rong[†] Chucai Yi[§] Yingli Tian[†]
[†]The City College, City University of New York, USA
[§]HERE North America LLC, USA
{xrong, ytian}@ccny.cuny.edu chucai.yi@here.com

Abstract

Text instance as one category of self-described objects provides valuable information for understanding and describing cluttered scenes. In this paper, we explore the task of unambiguous text localization and retrieval, to accurately localize a specific targeted text instance in a cluttered image given a natural language description that refers to it. To address this issue, first a novel recurrent Dense Text Localization Network (DTLN) is proposed to sequentially decode the intermediate convolutional representations of a cluttered scene image into a set of distinct text instance detections. Our approach avoids repeated detections at multiple scales of the same text instance by recurrently memorizing previous detections, and effectively tackles crowded text instances in close proximity. Second, we propose a Context Reasoning Text Retrieval (CRTR) model, which jointly encodes text instances and their context information through a recurrent network, and ranks localized text bounding boxes by a scoring function of context compatibility. Quantitative evaluations on standard scene text localization benchmarks and a newly collected scene text retrieval dataset demonstrate the effectiveness and advantages of our models for both scene text localization and retrieval.

1. Introduction

Text instances such as characters, words and strings in a scene image provide the most concise and accurate natural language expressions to understand and explain the scene. Reading text information from a camera-based natural scene, named as scene text extraction, plays a significant role in scene understanding and its associated applications, such as navigation, localization, context retrieval, end-to-end machine translation, and wayfinding for visually impaired, etc. However, most existing scene text extraction approaches regard text instances as a generic category of objects, and attempt to encode text instances into separable

Figure 1: An example of unambiguous text localization and retrieval. Given a cluttered scene image and candidate text bounding boxes (in white, detected by the proposed DTLN), the proposed CRTR model is applied to retrieve a specific text instance (in color) based on a natural language description. It scores and ranks candidate boxes based on text attributes, spatial configurations, and context information.

feature representations from other categories of objects, and then assign all text instances existing in the scene to predefined prediction labels. It means that text instance could not contribute more than other objects to the understanding and description of a scene, even though the text is more related to context environment and semantically self-described.

Precisely, for a text instance in a natural scene image, current mainstream text extraction methods could generate their locations and sequential character codes, to which we refer as *spatial* and *literal* information afterward. However, to comprehensively describe and interpret a highly cluttered natural scene, higher level clues such as *semantic* and *con-*

textual information are necessary. There has been a lot of work exploring practical applications of scene text extraction such as shopping assistants in grocery stores [1, 2], especially for blind or visually impaired people. But text information would help scene understanding only if the user perceives where the text instances are from. For example, when a blind or visually impaired people is using scene text extraction in a grocery store to help find price of a product, he/she would prefer the shopping assistant application to generate natural language description like {*large words on a red sign saying "unbeatable price" above a basket of red apples at the right side*}, rather than a list of discrete and unordered words from text extraction, as shown in Fig. 1. Moreover, in daily life it is more natural for a human to refer to objects and scene text instances based on their attributes, appearances, and spatial configurations, since the fine recognition process usually occurs in brain after rough localization[1].

To better utilize text information in natural scenes, the relationships between text instances and their contexts are explored in this paper. We propose a new framework of text-based scene understanding, which combines the localization of text instances from a scene with the informative and unambiguous natural language description of the localized text instances. This kind of natural language descriptions is known as *referring expression* [3, 4, 5]. We know that context descriptions of text instances are effective on the understanding and description of the entire scene if the text instances are accurately localized. Being able to retrieve scene text instances from natural images is critical in a number of applications that use natural language interfaces, such as controlling a robot (e.g., {*Alexa, please read me the green note besides the fridge*}), or interacting with photo editing software (e.g., {*Picasa, please blur the white door numbers on the grey front door*}).In addition, it provides a valuable testbed for research on vision and language systems.

The contributions of this paper have three aspects. First, we propose a text-based framework of scene understanding, which combines the localization of text bounding boxes with the retrieval of text instances from context description. Second, we propose the relationship modeling between scene text instances and their context concepts in scene images. Third, a new large-scale dataset is constructed to evaluate the performance of unambiguous text instance retrieval. The proposed framework the first solution of jointly modeling image-based scene text localization with language-based description of the localized text instances. It significantly extends the conventional scene text retrieval task, and can be applied to understand and describe cluttered scenes.

In our proposed framework, spatial information and context descriptions of scene text instances benefit from each other. The scene text locations could provide pivotal and precise information for context descriptions of the entire or a region of the scene image, while context description could provide a more user-friendly way to incorporate the extracted text information and its context into practical applications.

[1] http://tinyurl.com/nerorec

2. Related Work

Generally, text detection and recognition, word image retrieval, image captioning and description, generation and comprehension of referring expressions can be seen as different directions of the same Visual-Linguistic super-task, which jointly models the natural language information and image content. We discuss these related areas as follows.

Text extraction in the wild. Scene text extraction consists of text localization and text recognition. As the state-of-the-art text recognition accuracy on cropped word image has been over 98% [6], the performance of text localization is the main bottleneck of text extraction in natural scenes. Most existing text localization methods [7, 8, 9, 10, 11] usually employed a bottom-up pipeline based on sliding window or connected components, which was usually hard-coded with less robustness and reliability, and their performance heavily relied on the low-level image filtering. Even though Convolutional Neural Networks (CNN) substantially improved generic object detections, text localization from cluttered scene image was still a challenging problem, due to the highly variant and undefined appearance and structure of scene text instances [12, 13, 14]. Recently, a new synthetic text dataset was proposed in [15] for training a fully convolutional regression network for text localization similar to YOLO [16], and achieved decent results on several popular datasets, though failures often occur on tiny or crowded text instances. Moreover, YOLO-alike approaches cannot predict more than two instances from one grid cell, while our proposed model is able to generate sets of predictions in variable lengths from a small region and handle the crowded instances in a high density. [17] aimed to connect sequential fine-scale text proposals horizontally using LSTM which achieved top performance on text localization. However, the strong assumption of horizontal text lines could be easily violated in practice applications.

Many deep neural networks [18, 19, 20] were proposed to effectively encode scene images or their sub-regions into feature representations for classification tasks, and these networks could be applied for scene text extraction. However, they ignored the relationships between text instances and their surrounding objects in cluttered scene images. In our proposed DTLN network, CNN is still employed to obtain deep convolutional representations of scene images, but we adopt Long Short Term Memory (LSTM) [21] based decoders to jointly model text instances and their context. This architecture worked very well on the generation of image captions [22] and machine translations [23]. With the

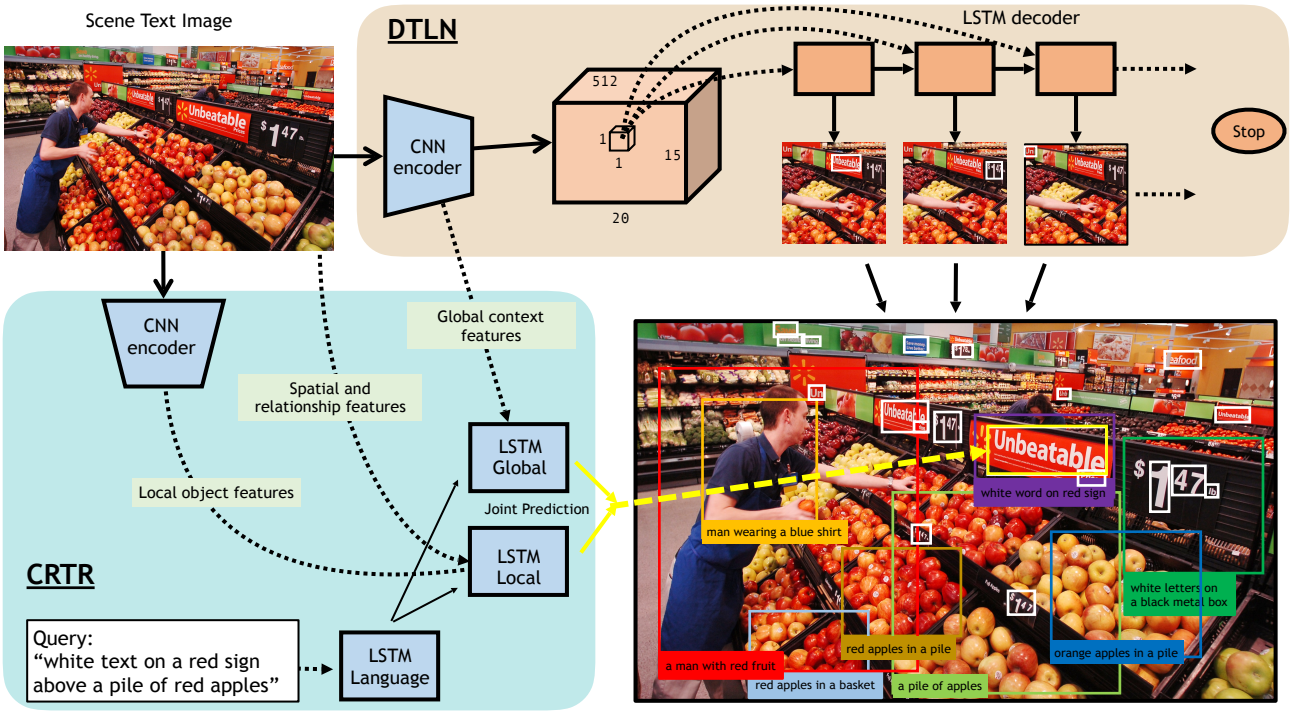


Figure 2: The architecture of the proposed Dense Text Localization Network (DTLN) and Context Reasoning Text Retrieval (CRTR) Models. For an input image, the DTLN model directly decodes the CNN features into a variable length set of text instance candidates. The CRTR model pools the information from three different LSTM models, and jointly scores and ranks the candidate text regions which are generated by DTLN.

help of LSTM network, our proposed DTLN could memorize previously generated text bounding boxes and avoid the repeated detection at multiple scales of the same target.

Image captioning and referring expression. Several approaches were proposed to explore the descriptions and explanations of scene images with natural language [24]. In the recent work [3], the image content was represented by hidden activations of a CNN, and then fed as input into LSTM framework for caption generation. However, these image captioning methods aimed to describe the entire image, without modeling spatial localization of text instances or some generic objects and their context. Our approach employs a similar network architecture to generate context descriptions of the localized text regions.

The context description is tightly related to the concept *referring expression* in the visual-linguistic research area. Referring expression generation had been a classic natural language processing problem. There were several important issues in this problem. It explored what types of attributes people typically used to describe visual objects, and also dealt with the usage of higher-order relationships (e.g., spatial comparison) [4]. However, referring expression for text instances of a scene image still remains unexplored, and our framework utilizes context descriptions of scene text instances as their referring expression to retrieve targeted text information from cluttered scene images.

The rest of the paper is organized as follows. Sec. 3 presents our proposed deep neural networks for dense scene text localization from image-based feature and scene text retrieval from language-based context description. Sec. 4 describes the experiments of localizing text instances on standard benchmark datasets, and the experiments of retrieving target text instances through their context descriptions on a self-constructed dataset. Sec. 5 concludes this paper.

3. Proposed Framework

3.1. Convolutional Encoding Network

Our framework employs the VGG-16 architecture [20] to encode a scene image I into a feature map in a $M \times N$ grid of 512 dimensional feature descriptors. In detail, VGG-16 network consists of 13 layers of 3×3 convolutions, and 5 interleaved layers of 2×2 max-pooling. We draw the network data before the final pooling layer as feature map, namely *conv5*. The feature map covers large receptive fields from the original scene image, and encodes object categories from ImageNet [25], which is then fed into a 2×2 average-pooling layer.

3.2. Dense Sequential Text Localization

Although scene text instances were often treated as one special category of object in the detection phase, their highly variant appearances/scales and self-description attribute significantly distinguish them from generic objects. Convolutional encoding network as described above encodes a strided region of the original scene image into a vector of 512 dimensional feature descriptors. According to the recent development of LSTM based language model [23, 26], we build a recurrent decoder to make joint predictions in sequence for all potential target objects, which are scene text instances in our framework. The combination of a CNN-based encoder with LSTM-based decoder plays a critical role in our framework. It enables the generation of coherent sets of predictions in variable lengths. These properties have been leveraged successfully to generate image captions [22], machine translation [23], and people detection [27]. The method in [27] worked well on people detection, but was not involved in the detection of objects with highly irregular and variant spatial configurations. Also, this method was mainly to solve the occlusion problems which rarely happen to scene text instances.

The ability to generate coherent sets is critically important in our task because there is no prior knowledge of how many text instances would appear in a local region, and our system needs to memorize previously generated text predictions and avoid repeated predictions of the same target.

Decoding process. The 512 dimensional feature descriptor summarizes the contents of the strided region and carries information about the sizes, positions and categories of the objects inside the strided region. An LSTM-based decoder would smartly extract target scene text instances from these CNN encoded feature descriptors. The LSTM-based decoder sequentially outputs new bounding boxes and their corresponding confidence scores. This score indicates the probability that a previously undetected text instance could be found at the location of the bounding box. The bounding boxes are produced in the ordering of descending confidence scores. When the LSTM-based decoder is unable to find more bounding boxes with higher confidence scores in the strided region, a stop symbol is produced to end the entire decoding process. All the output bounding boxes and confidence scores from all strided regions of the scene image are collected as the predictions of scene text instances.

Implementation details. According to the convolutional encoding network, there are $M \times N$ strided regions at a scene image, so the same number of $M \times N$ LSTM controllers run in parallel on $1 \times 1 \times 512$ grid cells. In our framework, we set $M = 15$ and $N = 20$ based on experimental results. The LSTM units have 500 memory states, no bias terms, and no output nonlinearities. At each step, we concatenate the VGG-16 feature maps with the output of the previous LSTM unit, and feed the result into the next LSTM unit. This network learns to regress exactly on bounding boxes of text instances through the LSTM decoder.

In training process, the LSTM-based decoder outputs an overcomplete set of bounding boxes along with their confidence scores. Bounding boxes with higher confidence score are preferred during matching with the ground truth. On COCO-TextRef dataset, we limit the overcomplete set to be top 5 predictions. In our experiments, more predictions largely increase the computational complexity, but not obtain obvious performance improvement.

In training process, hypotheses of text bounding boxes are generated in sequence. A text bounding box output by LSTM is represented by a 6 dimensional vector $\mathbf{b} = \{\mathbf{b}_{pos}, \mathbf{b}_c\}$, where $\mathbf{b}_{pos} = [\frac{b_x}{W}, \frac{b_y}{H}, \frac{b_w}{W}, \frac{b_h}{H}, \frac{b_w \cdot b_h}{W \cdot H}] \in \mathbb{R}^5$ is the relative position, width, height, and area size of the bounding box, and $b_c \in [0, 1]$ is a real-valued confidence. In LSTM, all hypotheses of text bounding boxes are associated with previous counterparts via the memory states.

Confidence scores lower than a pre-specified threshold are interpreted as a stop symbol at the testing phase. The higher confidence score b_c indicates that the bounding box is more likely to cover a true positive text instance. In practice, we use a Hungarian loss term for the output bounding boxes as in [27]. Typical detection errors such as false positives, missing detections, and repeated predictions of the same ground-truth instance are penalized in the training process.

Text region refinement. The proposed text localization method is trained to predict multiple bounding boxes within a grid cell. To handle an entire image in testing phase we generate predictions from each cell of a 15×20 grid of the image, and then recursively stitch and merge predictions from successive cells on the grid. Therefore, the proposed method can handle the dense and cluttered tiny text instances while still capturing large-size text instance that occupies a big area of the scene image.

3.3. Unambiguous retrieval of text instances

This subsection presents context reasoning model (CRTR) which retrieves scene text instances by natural language. In testing phase, given an image along with a natural language query and a set of candidate text bounding boxes (ground truth or generated by the proposed DTLN), the CRTR selects a subset of text bounding boxes from the outputs of DTLN that match the query context description.

Visual relationship modeling. Text instances in scene image are usually embedded in complex background with all sorts of contextual outliers, so it is difficult to model informative and unambiguous descriptions of the text instances if not take into account their relationships with the generic

objects in context. This makes sense intuitively: text instances in natural scenes are usually composed of printed or handwritten characters appearing on the surface of certain objects, and their visual relationships usually dominate the holistic interpretation of a natural scene image.

Since the set of relationships between text instances and context concepts (e.g., objects, stuff, persons) is tremendous and permutationally growing, we focus on the context concepts that are directly associated and interactive with text instances. However, it is still uneasy to obtain sufficient training examples to cover all this kind of relationship pairs. To simplify this problem and work out a minimum viable solution, we reduce the semantic space to contain only the relationships between single text instance and single context object, because the semantic space of all possible relationship pairs is much larger than that of individual text instance and context object. Visual relationship is represented as a language query as `{text-relationship-context}`, where `relationship` could be *spatial*, *preposition*, *comparative* or other possible categories (e.g., no *action* and *interaction* for text instances as the subject) [28] for text instances. To avoid ambiguities in the evaluations of the context descriptions of scene text instances, we focus on the prediction of their spatial relationships and text attributes, similar to the scheme in [4], as shown in Fig. 2.

Context reasoning text retrieval. Inspired by the architecture of LRCN [26] and SCRC [29], our Context Reasoning Text Retrieval (CRTR) model for natural language scene text instance retrieval consists of several components as illustrated in Figure 2. The model has three LSTM units denoted by LSTM$_{lang}$, LSTM$_{local}$ and LSTM$_{global}$, a local and a global CNN, and word embedding and prediction layers, concurrent with [26] and [29]. At testing, given an image I, a query text sequence S and a set of candidate text bounding boxes $\{b_{pos}\}$ in I, the network outputs a score s_i for the i-th candidate box b_{pos} based on local image descriptors x_{box} on b_{pos}, spatial configuration b_{pos} of the box with respect to the scene, and global contextual feature $x_{context}$. The local descriptor x_{box} is extracted by CNN$_{local}$ from local region I_{box} on b_{pos}, and the feature extracted by another network CNN$_{global}$ on the whole image I_{im} is employed as scene-level contextual feature $x_{context}$. The spatial configuration of $b_{pos} = [\frac{b_x}{W}, \frac{b_y}{H}, \frac{b_w}{W}, \frac{b_h}{H}, \frac{b_w \cdot b_h}{W \cdot H}] \in \mathbb{R}^5$ is an 5-dimensional representation similar to the one in DTLN.

In the query text sequence S, the words $\{w_t\}$ are represented as one-hot vectors and embedded through a linear word embedding matrix, and processed by LSTM$_{lang}$ as the input time sequence. At each time step t, LSTM$_{local}$ takes in $[h^{(t)}_{lang}, x_{box}, b_{pos}]$, and LSTM$_{global}$ takes in $[h^{(t)}_{lang}, x_{context}]$. Finally, based on $h^{(t)}_{local}$ and $h^{(t)}_{global}$, a word prediction layer predicts the conditional probability distribution of the next word based on local image region I_{box}, whole image I_{im}, spatial configuration b_{pos} and all previous input words.

For the other training settings, we follow [26] and [29]. VGG-16 net [20] trained on ImageNet dataset [25] is still used as the CNN architecture for CNN$_{local}$ and CNN$_{global}$ and we extract 1000-dimensional *fc8* outputs as x_{box} and $x_{context}$, and use the same LSTM implementation as in [26] and [29]. Each of the three LSTM units has 1000-dimensional state h_t. It is worth noting that the CNN$_{global}$ can share the features from the DTLN model. In testing phase, given an input image I, a query text S and a set of candidate text bounding boxes $\{b_{pos}\}$, the query text S is scored on i-th candidate box using the likelihood of S conditioned on the local image region, the whole image and the spatial configuration of the box, which can be computed as $s = p(S|I_{box}, I_{im}, \{b_c, b_{pos}\})$ and the candidate box with the highest score is retrieved ($b_c = 1$ for ground truth input, and $b_c \in [0, 1]$ for text localization input). In training phase, each instance is an image-bounding box-description tuple, which is constructed from the ground truth annotations as training instances (multiple tuples are constructed if there are multiple descriptions for the same text instance, or same description for multiple text instances in close proximity) in experiments. During training, the model parameters are initialized from the pretrained network, and fine-tuned using SGD with a smaller learning rate, allowing the network to adapt to natural language text retrieval domain. The whole CRTR network is trained end-to-end via back propagation.

4. Experiments

In Sec. 4.1 and 4.2 we introduce the details of the text localization datasets and the newly collected scene text retrieval dataset. Experiments and corresponding discussions are presented in Sec. 4.3 and 4.4.

4.1. Datasets for Text Localization

First, the proposed dense text localization method is trained and evaluated on standard benchmarks, including *SynthText* dataset, *ICDAR* 2013 dataset [30], and the Street View Text dataset [31]. Then the whole unambiguous text localization framework is evaluated on a newly collected COCO-TextRef dataset.

SynthText in the wild dataset. This is a dataset containing 800,000 synthetic training images, which were generated in [15]. Each image has word instances annotated with character and word-level bounding boxes.

ICDAR 2013 dataset. ICDAR (International Conference on Document Analysis and Recognition) 2013 dataset contains real world images of text on sign boards, books, posters and other objects with world-level axis-aligned

Table 1: Performance comparison between our proposed framework with previous scene text localization approaches on ICDAR 2013 [30] and SVT datasets [31] in terms of the measures of PASCAL Eval [32] and DetEval [33]. Precision (P) and Recall (R) at maximum F-measure (F) and the average computation time (T) are reported. Bold number indicates the best performance for each measure metric. Average time spent on these scene text localization approaches (the last column) demonstrates that the proposed DTLN achieves state-of-the-art F-measure while running in comparable speed as competing approaches.

	PASCAL Eval						DetEval						Time
	IC13			SVT			IC13			SVT			Avg.
	F	*P*	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*T/s*
TH-TextLoc [30]	-	-	-	-	-	-	0.67	0.70	0.65	-	-	-	-
Text Spotter [8]	-	-	-	-	-	-	0.74	0.88	0.65	-	-	-	0.3
Yin et al. [9]	-	-	-	-	-	-	0.76	0.88	0.66	-	-	-	0.43
Lu et al. [34]	-	-	-	-	-	-	0.78	0.89	0.70	-	-	-	-
Jaderberg [12]	0.76	0.87	0.68	0.54	0.63	0.47	0.77	0.89	0.68	0.25	0.28	0.23	7.3
Zhang et al. [35]	-	-	-	-	-	-	0.80	0.88	0.74	-	-	-	60.0
FCN [13]	-	-	-	-	-	-	0.83	0.88	0.78	-	-	-	2.1
FCRNall+filts [15]	0.84	**0.94**	0.76	0.63	0.65	0.60	0.83	**0.94**	0.77	0.27	**0.29**	0.26	1.27
Tian et al. [17]	**0.88**	0.93	**0.83**	**0.66**	**0.68**	**0.65**	-	-	-	-	-	-	**0.14**
DTLN	0.85	0.92	0.79	0.64	0.65	0.63	**0.85**	0.92	**0.78**	**0.28**	**0.29**	**0.27**	0.35

Figure 3: Example results of scene text localization. The green bounding boxes contain correct detections; Red bounding boxes contain false positives; Red dashed box (e.g., the one at the bottom-right image) contains the false negative.

bounding box annotations. It consists of 229 training images and 233 testing images.

Street View Text (SVT) dataset. This dataset consists of images harvested from Google Street View annotated with word-level axis-aligned bounding boxes. *SVT* is more challenging than the ICDAR data as it contains smaller and lower resolution text which exhibits high variability. It consists of 100 training images and 249 testing images.

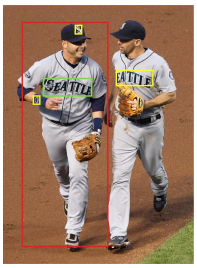
4.2. Data Construction for Text Instance Retrieval

To our knowledge, the largest dataset for evaluating object retrieval and referring expression is *ReferIt dataset* from [5]. However, this dataset did not provide any annotations and expressions for scene text instances, therefore we create a new large-scale dataset for evaluating the proposed framework.

We select the intersection parts of COCO-Text and Google Refexp Datasets to establish a new dataset contain-

query = "largest text on the closest object"

query = "white text around a bench"



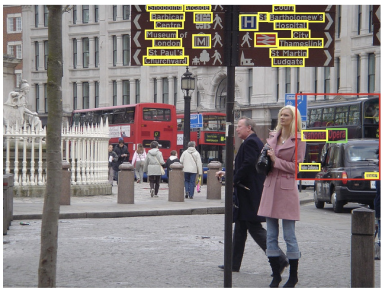
query = "largest text left to the right human"

query = "text on a motorcycle"

query = "text on the right cup"

query = "text on top of a red boat"

query = "blue text on the largest plane"



query = "most salient text on a bus"

Figure 4: Text region retrieval results of the proposed Context Reasoning Text Retrieval (CRTR) model on the COCO-TextRef dataset. At first, red boxes are employed to denote context concepts. Then green boxes are added to identify the successfully retrieved text regions associated with the context concepts. The remaining text regions are marked by yellow boxes.

ing both text instance annotations and background concept annotations with descriptions. Synthetic text instances are rendered on certain images through the method in [15] with corresponding descriptions manually labeled, when the number of natural text instances is much less than the context concepts. This dataset is referred as COCO-TextRef, which in total contains 6,638 images with 31,870 expressions (all in `{text-relationship-context}` style, and further filter out with human assessment), referring to 11,342 distinct objects. It contains 17,355 text instances and their literal transcriptions.

4.3. Text Localization Experiments

The proposed dense text localization network is trained on 800,000 images from the *SynthText in the Wild* dataset. Each image is resized to 480×640. VGG-16 weights are initialized with the weights pretrained on ImageNet [25], and fine-tuned to meet the new demands of the decoding process. All weights in the decoder are initialized from a uniform distribution. Training proceeds in parallel on all grid cells of one image at each iteration. All weights are tied between regions and LSTM steps. Training on the whole *SynthText in the Wild* dataset takes about 15 hours on a NVIDIA Titan X (Maxwell) GPU for 200,000 iterations.

The following criteria are used to evaluate text localization results. (1) The standard PASCAL VOC detect criterion: a detection is true positive if the Intersection over Union (IoU) between its bounding box and the ground truth exceeds 50%. (2) The DetEval [33] criterion: an evaluation metric which emphasizes more on detection quality and has been popularly used in ICDAR competitions. To further improve the performance, we follow the post-processing rou-

Table 2: The left table presents the Top-1 precision of our method compared with previous methods on annotated ground truth bounding boxes on the COCO-TextRef dataset. The right table presents the Top-1, Top-5 recalls of our method compared with previous methods with detected text regions generated by the proposed DTLN method.

Method	P@1
LRCN [26]	0.264
DenseCap [36]	0.291
SCRC [29]	0.457
CRTR	**0.582**

Method	R@1	R@5
LRCN [26]	0.083	0.213
DenseCap [36]	0.095	0.229
SCRC [29]	0.135	0.313
CRTR	**0.184**	**0.394**

tine introduced by [6] to filter out hard false positives. In detail, first we use a binary text/non-text random forest classification model to filter out non-text proposals; second, text region proposals are improved by CNN-based regression. Table 1 shows the performance of our DTPN model. The precision and recall at maximum F-measure, and the average computation time on both datasets of our basic model are reported. In conjunction with a simple binary text/no-text random-forest classifier [6] to further eliminate false-positive detections, it outperforms state-of-the-art methods in terms of recall and achieves comparable precision. Qualitative results are shown at Figure 3, which demonstrate that the proposed approach effectively tackles the relatively crowded scene text instances, and extracts them from the cluttered and complex background.

Based on the analysis of evaluation results and comparison with recent state-of-the-art word-based text detection methods like [15] and [17], our proposed DTPN performs equally well for sparse text instances, and performs better in detecting relatively dense and crowded ones. However, it still fails to handle some challenging cases, such as overexposure and large character spacing. Some failure cases are indicated by red solid (false positive) and dash (false negative) boxes in Figure 3.

4.4. Text Retrieval Experiments

Context Reasoning Text Retrieval (CRTR) model is evaluated on the newly collected COCO-TextRef dataset. Since DenseCap [36] solved a similar problem of region description and retrieval where text instances were treated as one special category of objects and denoted as *signs*, *words*, or *letters*, we fine-tune DenseCap with the COCO-TextRef dataset and adopt it as our baseline. We compare our method with LRCN [26] and SCRC [29], which are also fine-tuned on the COCO-TextRef dataset for the ability to retrieve text instances.

The CRTR model is evaluated for two scenarios. First, given a natural scene image and a natural language query, the model is to retrieve the corresponding text region from all annotated text regions in that image, which is similar to an object retrieval problem. And we evaluate our proposed CRTR model individually in this scenario. Second, as a more challenging but practical work, given an image and a natural language query, the model should retrieve a text region from a set of candidate text regions generated by the scene text localization methods. In both scenarios, we follow the standard PASCAL VOC detection criterion: a retrieved text region is considered as correct if $IoU > 50\%$, otherwise it is a false positive. This is equivalent to computing the *precision@1* measure (the percentage of the highest scoring text region being correct). We then average these scores over all images. Table 2 compares the evaluation results of our proposed CRTR model with previous object retrieval models tuned for text instance retrieval. We observe that CRTR outperforms most previous methods in terms of *precision@1* measure on individual text retrieval evaluation, and in terms of *recall@1* (the percentage of the highest scoring text bounding box proposals being correct) and *recall@5* (the percentage of at least one of top-5 highest scoring text bounding box proposals is correct) measures on joint text localization and text retrieval evaluation.

Fig. 4 shows examples of successfully retrieved text instances at top-1, where the highest scoring candidate region from our CRTR model overlaps with ground truth annotation by at least 50% IoU. It demonstrates that the proposed model effectively localizes and retrieves the targeted text region based on the input natural language queries. Also, the {text-relationship-context} modeling which the SCRC model did not explicitly handle substantively fills in the gap between image-based scene text localization and language-based scene understanding through the localized text instances, and boosts the performance.

5. Conclusion

To utilize text instances for understanding natural scenes, we have proposed a framework that combines image-based text localization with language-based context description for text instances. Context description enables the localized text information to be delivered in a more user-friendly way for many practical applications. Accurate localization of scene text regions ensures concise and accurate language description of a scene image, and effective retrieval of text instances from context description.

Our future work will focus on combining the models of scene text localization and scene text retrieval to produce an end-to-end system. The performance can also be further improved with pre-processing techniques such as image super-resolution [37, 38] and deblurring [39, 40].

Acknowledgements. This work was supported in part by NSF grants EFRI-1137172, IIP-1343402, and IIS-1400802.

References

[1] B. Xiong and K. Grauman. Text detection in stores using a repetition prior. *in WACV*, 2016.

[2] C. Yi, Y. Tian, and A. Arditi. Portable camera-based assistive text and product label reading from hand-held objects for blind persons. *IEEE Trans. on Mechatronics*, 2014.

[3] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg. Modeling context in referring expressions. *in ECCV*, 2016.

[4] J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy. Generation and comprehension of unambiguous object description. *in CVPR*, 2016.

[5] S. Kazemzadeh, V. Ordonez, M. Matten, and T. L. Berg. Referitgame: Referring to objects in photographs of natural scenes. *EMNLP*, 2014.

[6] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 2015.

[7] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in nature scenes with stroke width transform. *in CVPR*, 2010.

[8] L. Neumann and J. Matas. Real-time scene text localization and recognition. *in CVPR*, 2012.

[9] X. Yin, K. Huang, and H. Hao. Robust text detection in natural scene images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2014.

[10] X. Rong, C. Yi, X. Yang, and Y. Tian. Scene text recognition in multiple frames based on text tracking. *in ICME*, 2014.

[11] X. Yin, W. Pei, J. Zhang, , and H. Hao. Multi-orientation scene text detection with adaptive clustering. *in TPAMI*, 2015.

[12] M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. *in ECCV*, 2014.

[13] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-oriented text detection with fully convolutional networks. *in CVPR*, 2016.

[14] X. Rong, C. Yi, and Y. Tian. Recognizing text-based traffic guide panels with cascaded localization network. *in ECCV Workshop*, 2016.

[15] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localisation in natural images. *in CVPR*, 2016.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection. *in CVPR*, 2016.

[17] Z. Tian, W. Huang, T. He, Pan. He, and Y. Qiao. Detecting text in natural image with connectionist text proposal network. *in ECCV*, 2016.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *in NIPS*, 2012.

[19] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *in ICML*, 2015.

[20] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *in ICLR*, 2015.

[21] S. Hochreiter and J. Schmidhuber. Long short term memory. *Neural Computation*, 1997.

[22] A. Karpathy and Fei-Fei Li. Deep visual-semantic alignments for generating image descriptions. *in CVPR*, 2015.

[23] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. *in NIPS*, 2014.

[24] E. Krahmer and K. van Deemter. Computational generation of referring expressions. *Comp. Linguistics*, 2012.

[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and F. Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

[26] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. *in CVPR*, 2015.

[27] R. Stewart and M. Andriluka. End-to-end people detection in crowded scenes. *in CVPR*, 2016.

[28] C. Lu, R. Krishna, M. Bernstein, and F. Li. Visual relationship detection. *in ECCV*, 2016.

[29] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell. Natural language object retrieval. *in CVPR*, 2016.

[30] D. Karatzas. Icdar 2013 robust reading competition. *in ICDAR*, 2013.

[31] K. Wang and S. Belongie. Word spotting in the wild. *in ECCV*, 2010.

[32] M. Everingham, S. M. Ali Eslami, L. V. Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 2015.

[33] C. Wolf and J.-M. Jolion. Object count/area graphs for the evaluation of object detection and segmentation algorithms. *International Journal on Document Analysis and Recognition*, 2006.

[34] S. Lu, T. Chen, S. Tian, J. Lim, and C. Tan. Scene text extraction based on edges and support vector regression. *in IJDAR*, 2015.

[35] Z. Zhang, W. Shen, C. Yao, and X. Bai. Symmetry-based text line detection in natural scenes. *in CVPR*, 2015.

[36] J. Johnson, A. Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. *in CVPR*, 2016.

[37] R. Dahl, M. Norouzi, and J. Shlens. Pixel recursive super resolution. *arXiv:1702.00783*, 2017.

[38] Y. Xian and Y. Tian. Resolution enhancement in single depth map and aligned image. *in WACV*, 2016.

[39] X. Rong and Y. Tian. Adaptive shrinkage cascades for blind image deconvolution. *in DSP*, 2016.

[40] J. Pan, D. Sun, H. Pfister, and M. Yang. Blind image deblurring using dark channel prior. *in CVPR*, 2016.