Unambiguous Text Localization, Retrieval, and Recognition for Cluttered Scenes

Xuejian Rong, *Member, IEEE,* Chucai Yi, *Member, IEEE,* and Yingli Tian†, *Fellow, IEEE*

Abstract—Text instance as one category of self-described objects provides valuable information for understanding and describing cluttered scenes. The rich and precise high-level semantics embodied in the text could drastically benefit the understanding of the world around us. While most recent visual phrase grounding approaches focus on general objects, this paper explores extracting designated texts and predicting unambiguous scene text information, i.e., to accurately localize and recognize a specific targeted text instance in a cluttered image from natural language descriptions (referring expressions). 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 text detections at multiple scales 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. Third, a recurrent text recognition module is introduced to extend the applicability of aforementioned DTLN and CRTR models, via text verification or transcription. Quantitative evaluations on standard scene text extraction benchmarks and a newly collected scene text retrieval dataset demonstrate the effectiveness and advantages of our models for the joint scene text localization, retrieval, and recognition task.

Index Terms—Natural Language Description, Text Detection, Text Retrieval, Text Recognition, Deep Neural Network, Referring Expression

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 camera-based natural scenes, named as scene text extraction, plays a significant role in scene understanding and its associated applications, such as navigation, geo-localization, context retrieval, end-to-end machine translation, and way-finding for visually impaired. However, most existing scene text extraction approaches regard text instances as a generic category of objects, and attempt to encode text instances into separable 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 information 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 *contextual* 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 people who are blind or visually impaired. But text information would help scene understanding only if the user perceives where the text instances are from and related. For example, when people with visual impairments are using scene text extraction in grocery stores or supermarkets to help find the price of a product, they would usually prefer a shopping aid to generate natural language descriptions 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 Figure 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 the brain after rough localization [3].

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* [4], [5], [6]. We know that contextual descriptions of text instances are effective on the understanding and description of the entire scene if the text instances are accurately localized.

High-quality and user-specific scene text detection from referring expressions can underpin many vision-language applications which rely on natural language interfaces, such

† *Corresponding author.*
X. Rong and Y. Tian are with the Department of Electrical Engineering, The City College, City University of New York, New York, NY, 10031.
E-mail: {xrong, ytian}@ccny.cuny.edu.
C. Yi received his Ph.D. degree in computer science in 2014 from The Graduate Center, City University of New York. He is now working at Google on augmented reality, Mountain View, CA, 94043.
E-mail: gschucai@gmail.com

Fig. 1: An example of unambiguous text localization, retrieval, and recognition. 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, which can be further transcribed to machine-readable character codes. The CRTR model scores and ranks candidate boxes based on text attributes, spatial configurations, and context information.

as controlling a robot (*e.g.*, {*Alexa, please read me the green note beside the fridge*}, or {*Alexa, please read me the price of non-fat CHOBANI Greek yogurt on the top shelf*}), or interacting with photo editing software (*e.g.*, {*Picasa, please blur the white door numbers on the grey front door*}, or {*Premiere, please transcribe/mosaic all the identity information if photo IDs or credit cards appear in this video*}). In addition, it provides a valuable testbed for research on vision and language systems. The proposed framework could also help dramatically boost the efficiency of the whole scene text reading process while avoiding the exhaustive search and recognition of all text instances which are very time-consuming.

This paper extends our earlier work [7] by integrating a scene text recognition module into the unambiguous text extraction pipeline for text transcription and verification. To our best knowledge, the proposed approach is the first solution of jointly modeling image-based scene text localization with a language-based description of the localized text instances, and still the state-of-the-art as indicated in [8]. It significantly extends the conventional scene text retrieval task, and can be applied to understand and describe cluttered scenes. The same CNN-RNN encoder-decoder architecture is naturally employed for text recognition along with preceding text localization and retrieval modules to handle text sequences in arbitrary lengths, involving no character segmentation or horizontal scale normalization. More analysis and more results are also presented.

The contributions of this paper have four 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 the contextual description, and the recognition process afterward. Second, we propose the relationship modeling between scene text instances and their context concepts in scene images. Third, a recurrent text recognition model is introduced to extend the text detection and retrieval approach through further transcription and verification. Last, a new large-scale dataset is constructed to evaluate the performance of unambiguous text instance retrieval. Our framework could retrieve targeted scene text instances according to their contextual descriptions, and generate contextual descriptions to uniquely pinpoint the text instances in the scene images in a CNN-RNN encoder-decoder manner.

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

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 subfields of the same Visual-Linguistic super-task, which jointly models the natural language information and image content. We first summarize the connections between most recent work and our earlier work on this task, and then discuss these related areas as follows.

Connections of later work with our preceding approach. As discussed before, our earlier work [7] is the first framework of jointly modeling scene text detection with corresponding language-based descriptions, and still the state-of-the-art as indicated in [8]. Our proposed approach in this manuscript comprises three modules (scene text detection, retrieval, and recognition), and more approaches have recently emerged especially for detection [9], [10], [11], [12], [13], [14], [15], [16] and recognition [17], [18], [19], [20]. Theoretically, many of these latest developments in scene text detection and recognition areas can be adopted in our proposed framework to boost the final performance without requiring a significant modification of the network architectures. However, some original designs such as sequential recurrent localization in [7] are still advantageous in terms of flexibility. Several recent follow-up tasks such as scene text visual question answering (VQA) [21], unambiguous scene text segmentation [22], and object assisted scene text spotting [23], are conceptually similar to the proposed task in [7] but introduced different evaluation metrics and testing data. We further introduce the recent work in detail in the rest subsections to provide a better context for our proposed framework in this manuscript.

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% [24], the performance of text localization is the main bottleneck of text extraction in natural scenes. Most existing text localization methods [25], [26], [27], [28], [29] usually employed a bottom-up pipeline based on sliding

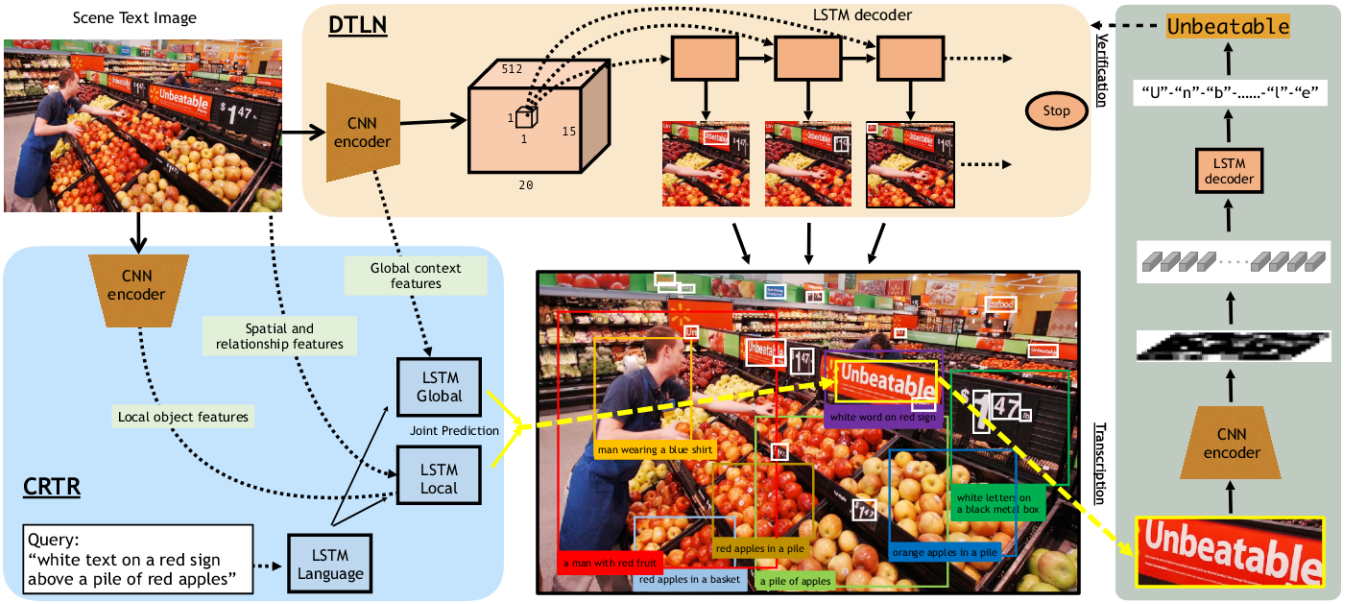


Fig. 2: The architecture of the proposed Dense Text Localization Network (DTLN) and Context Reasoning Text Retrieval (CRTR) Models. For an input image, the DTLN model directly decodes the CNN features into a variable length set of text instance candidates. The CRTR model pools the information from three different LSTM models, and jointly scores and ranks the candidate text regions which are generated by DTLN. The rightmost recurrent text recognition module helps further extend the applicability of DTLN and CRTR models, by transcribing the retrieved text regions to precise literal character codes for verification.

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 [30], [31], [32]. Recently, a new synthetic text dataset was proposed in [33] for training a fully convolutional regression network for text localization similar to YOLO [34], 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. [35] 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 [36], [37], [38] 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) [39] based decoders to jointly model text instances and their context. This architecture worked very well on the generation of image captions [40] and machine translations [41]. With the LSTM network, DTLN could memorize previously generated text bounding boxes and avoid repeated detection at multiple scales of the same target.

Recently, [9], [10], [11], [12], [42] further boost the scene text detection performances, which are either based on direct regression or text segmentation. [9] proposed a multi-scale shape regression network that is capable of locating text lines of different lengths, shapes, and curvatures in scenes. [11] proposed to model cross-domain shifts with adversarial training for both text detection and recognition tasks. [10] is able to effectively detect text areas by exploring each character region and affinity between characters, which results in finer bounding boxes. In comparison, DTLN still demonstrates more potentiality and controllability as a sequential localization model.

In the recognition process, the diversity of scene text instances (*e.g.*, different colors, scales, orientations, fonts, and languages) usually makes text recognition a challenging problem. The complexity of background (cluttered elements like signs, fences, bricks often have similar textures and structures to true text) and various interference factors (*e.g.*, noise, blur, uneven illumination, low resolution, and partial occlusion) also drastically degrade the performance of final recognition results. To address these problems, some attempts have been made in the last decade. For instance, it was proposed in [43], [44] to first detect and segment individual characters, and then recognize them with CNN models trained on labeled character images. Such methods often heavily rely on the performance of accurate character segmentation, and are prone to be affected by common degradation. Other approaches such as [24], [30] were pro-

posed to treat scene text recognition as an overcomplete image classification problem, and assign a class label to an entire input text region as an English word from a dictionary (90,000 predefined classes in total). This kind of methods usually results in a large trained model and is difficult to be generalized to text instances with various appearances. Several pioneering work such as [45], [46], [47] further verified the effectiveness of text recognition with recurrent networks. In our proposed framework, text recognition module consistently follows the CNN-RNN encoder-decoder architecture associated with DTLN and CRTR modules. It naturally deals with text sequences in arbitrary lengths, involving no character segmentation or horizontal scale normalization.

For a more comprehensive view and knowledge about the history of the scene text detection and recognition approaches emerging in the deep learning era, we also refer readers to the recent survey [8].

Alignment of images with language. Learning correspondences between sentence structural semantics and image regions has been explored with the visual-semantic alignment. This architecture has been used for applications in image retrieval and caption generation [48], [49]. With new datasets proposed which provide bounding box-level natural language annotations [5], recent work has also investigated region-wise image captioning and description for the tasks of natural language object retrieval [50], dense captioning [51], scene graph parsing [52], and visual common sense reasoning [53]. Our proposed framework has a similar idea that aligns a language triplet with regions of pixels in the image. Typically, existing approaches do not explicitly represent relations between noun phrases in a sentence to improve visual-semantic alignment. We believe that understanding these relations will lead to better scene understanding including phrase grounding and comprehension, as well as scene graph generation and reasoning.

Image captioning and referring expression. Many approaches have been proposed to explore the descriptions and explanations of scene images with natural language [54]. In the recent work [4], 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 contextual descriptions of the localized text regions.

The contextual 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 with 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) [5]. However, referring expression for text instances of a scene image still remains unexplored, and our framework utilizes contextual descriptions of scene text instances as their referring expression to retrieve targeted text information from cluttered scene images.

Visual relationship detection Visual relationship detection, as a classic research topic, has been investigated by numerous studies in the last decade. In the early days, researchers mainly focused on specific types of phrases [55], [56] or couple visual phrases with other vision tasks for better performance [57], [58]. Recently, more attention are paid to general visual relationship detection [56], [59], [60], [61], [62], [63], [64]. Lu *et al*. [65] utilized the language prior in the relationship detection between the targets and related components. Li *et al*. [59] model the dependencies among subject, predicate, and object branches with the message passing structure. Xu *et al*. [60] built up a fully-connected graph to iteratively pass messages along the scene graph. Liang *et al*. [66] aimed to detect the relationship between objects and their attributes by reinforcement learning. However, the relations between text instances and the surrounding background objects have not yet been specifically studied. Also, existing approaches do not scale well to a large number of relations as such visual explanations grow combinatorially. And our proposed methods can significantly alleviate this problem.

Triplet learning has been addressed in various tasks such as mining typical relations (knowledge extraction) [67], reasoning [68], object detection [69], or image retrieval [70]. In this work, we address the task of relationship modeling in scene text segmentation from language-based explanations. Early work on human-object interactions [71] models the triplet in the form $(person, action, object)$. Recently, the work in [65] tried to generalize the similar setting to non-human subjects by developing a language model sharing knowledge among visual detections related to each other. Inspired by the idea but different from these approaches, we restrict the $subject$ to be a text instance and cover a broader class of predicates that include prepositions and comparatives. In our work this combinatorial challenge can be addressed by developing a new visual representation with better generalization into unseen triplets $\{$*text-predicate-object*$\}$ *and* without depending on a strong language model.

Grounding visual explanations. Our proposed framework is an innovative combination of the recent work on object localization and segmentation from natural language descriptions, i.e., referring expression comprehension. In those work, the task is to localize/segment a target object in a scene based on its natural language referring expression (by drawing a bounding box over it, or pixel wisely assigning the foreground label to it).

The methods of [50] and [5] are built upon image captioning frameworks such as LRCN [72] or mRNN [73], and localize objects by selecting the bounding box where the expression has the highest probability. The authors of [7] firstly proposed a natural language-based scene text extraction method, but the framework is not trained end-to-end and cannot output pixel-wise text annotations. In [74], the authors proposed a model to localize a textual phrase by attending to a region on which the phrase can be best reconstructed. In [75], a joint embedding space of visual features and words is learned to localize target object by searching the closest region in the joint embedding space. [76] proposes an end-to-end training method for generating object segmentation mask from natural language descriptions. The proposed model encodes the given expression into a real-valued vector using LSTM networks [39], and extracts a

spatial feature map from the image using a Convolutional Network. Then it performs pixel-wise classification based on the encoded referring expression and feature map to output an image mask covering the visual entity described by the expression. Liu et al. [77] further propose to learn the word-to-image interaction instead of modeling image and sentence features independently. The proposed method achieves top results on general object segmentation with language explanations, and also shows that the combination of visual and linguistic features for scene text segmentation is worth exploring.

To the best of our knowledge, all previous methods of natural language-based detection and retrieval can only return a bounding box or segmentation mask of the generic objects, and no prior work has learned to directly localize text instances given a natural language description as a query. Our previous work [7] pioneered the task of natural language-based scene text detection and retrieval, and we expect to further improve the framework by integrating the text recognition capability to make it a unified end-to-end scene text extraction pipeline (scene image in, text transcription out).

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

3 Proposed Framework

3.1 Convolutional Encoding Network

Given an image with scene text instances, an informative and discriminative feature representation plays a significant role in unambiguous text localization. The feature representation should preserve the spatial layout of the objects into this image to enable the correct spatial prediction for text instances. This can be accomplished through a fully convolutional network model similar to FCN-32s [78], where the image is fed into a series of convolutional (and pooling) layers to obtain a feature map as an output containing encoded spatial information.

In our work, the network is further modified to encode region information into a better feature representation for varieties of text instances. Given an input scene text image of size $W \times H$, a $w \times h$ spatial feature map (where $w = 20$ and $h = 15$) is obtained with adaptive pooling in each position of the feature map containing D_{im} channels (D_{im} dimensional local descriptors). The grid setting is heuristically determined by common image ratios and can be changed accordingly. For each position on the spatial feature map, in order to obtain a more robust feature representation, $L2$-normalization is applied to the D_{im} dimensional local descriptor. In this way, a $w \times h \times D_{im}$ spatial feature map is extracted as the representation of each image.

To enable the model to reason about spatial relationships as shown in Figure 2, two extra channels are concatenated with the feature maps: the x and y coordinates of each spatial location. In this coordinate system, the top-left corner and the bottom-right corner of a feature map are represented as $(-1, -1)$ and $(+1, +1)$, respectively. In this way, we obtain a $w \times h \times (D_{im} + 2)$ representation containing both local image descriptors and spatial coordinates.

In our implementation, the VGG-16 architecture [38] is adopted as a fully convolutional network by converting fully-connected layers `fc6`, `fc7` and `fc8` to fully convolutional layers. This design helps aggregate the information of context concepts from neighboring regions of text instances, and reason about the interaction between visual text entities and context concepts.

3.2 Dense Sequential Text Localization

Scene text instances are distinguished from generic objects by their high variant appearances and scales, and self-descriptive attributes, even though they were usually treated as one special category of generic object in detection task. A strided region of the scene image is encoded into a 512 dimensional feature vector by the convolutional encoding network as described above. According to the recent development of LSTM-based language model [41], [72], 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 [40], machine translation [41], and people detection [79]. The method in [79] works well on people detection, but is not involved in the detection of objects with highly irregular and variant spatial configurations. Also, this method is 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 content of a strided region, including the sizes, positions, and categories of objects inside that region, is summarized by the 512 dimensional feature descriptor. An LSTM-based decoder smartly extracts target scene text instances from the 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 is tending to output 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. For the COCO-TextRef dataset, we limit the overcomplete set to be the top 5 predictions. In our experiments, more predictions largely increase the computational complexity, but not obtain obvious performance improvement.

Also during training, 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 [79]. Typical detection errors such as false positives, missed detections, and repeated predictions of the same ground-truth instance are penalized in the training process.

Text region refinement. There are multiple bounding boxes predicted by our proposed localization method within each cell of the 15×20 grid, and then the predictions from successive and neighboring cells are recursively stitched and merged. Specifically, for the current set of all accepted bounding box predictions, some of the new bounding box predictions may correspond to previous predictions. Therefore, any new boxes having nonzero intersection with accepted boxes are removed to avoid adding false positives, conditioned on the constraint that previously accepted boxes may destroy at most one new box. 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 our context reasoning text retrieval model (CRTR) which retrieves scene text instances by natural language descriptions. In the 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 and noises, 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) [65] for text instances.

To avoid ambiguities in the evaluations of the contextual descriptions of scene text instances, we focus on the prediction of their spatial relationships and text attributes, similar to the scheme in [5], as shown in Figure 2. This setting helps alleviate two problems in the visual relation modeling. First, the appearances of objects could significantly vary due to the interactions with other objects. However, text instances usually keep stable appearance in most cases, and they rarely change status relative to surrounding objects. Second, it is usually difficult and time-consuming to build the exhaustive explanation annotations for the object interactions (*i.e.*, the combinatorial challenge). Fixating the text instances as the subject of the pair-wise relationship explanations significantly narrows down the query space.

Context reasoning text retrieval. Inspired by the architecture of LRCN [72] and SCRC [50], our Context Reasoning Text Retrieval (CRTR) model, for scene text instance retrieval from natural language descriptions, 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 [72] and [50]. In testing process, 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

TABLE 1: Performance comparison between our proposed framework with previous scene text localization approaches on ICDAR 2013 [80] and SVT datasets [81] in terms of the measures of PASCAL Eval [82] and DetEval [83]. 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
	ICDAR13			SVT			ICDAR13			SVT			Avg.
	F	*P*	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*T/s*
TH-TextLoc [80]	-	-	-	-	-	-	0.67	0.70	0.65	-	-	-	-
Text Spotter [26]	-	-	-	-	-	-	0.74	0.88	0.65	-	-	-	0.3
Yin *et al*. [27]	-	-	-	-	-	-	0.76	0.88	0.66	-	-	-	0.43
Lu *et al*. [84]	-	-	-	-	-	-	0.78	0.89	0.70	-	-	-	-
Jaderberg [30]	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*. [85]	-	-	-	-	-	-	0.80	0.88	0.74	-	-	-	60.0
FCN [31]	-	-	-	-	-	-	0.83	0.88	0.78	-	-	-	2.1
FCRNall+filts [33]	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*. [35]	**0.88**	0.93	**0.83**	**0.66**	**0.68**	**0.65**	-	-	-	-	-	-	0.14
Liao *et al*. [86]	0.85	0.88	0.83	-	-	-	0.86	0.89	0.83	-	-	-	0.73
CRAFT *et al*. [10]	-	-	-	-	-	-	**0.95**	**0.97**	**0.93**	-	-	-	**0.12**
DTLN (ours)	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

region I_{box} on b_{pos} (i.e., I_{box} is the cropped image patch based 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. The word embedding module is pretrained from general image captioning task on MS-COCO dataset, and two additional symbols are added as start word and stop word in the query sentence. At each time step t, LSTM$_{local}$ takes in $[h^{(t)}_{lang}, x_{box}, b_{pos}]$, and LSTM$_{global}$ takes in $[h^{(t)}_{lang}, x_{context}]$. Here h_{lang} represented the encoded feature of the query sentence. The words in the query text sequence are encoded as one-hot vectors and embedded through a linear word embedding matrix. 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. Specifically, the word prediction layer indicates a Softmax layer for predicting the conditional probability distribution of the next word based on all current and previously predicted information.

For the other training settings, we follow [72] and [50]. VGG-16 net [38] trained on ImageNet dataset [87] 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 [72] and [50]. 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. Specifically, CNN_{local} (the local convolutional network which tackles the localized word patches) and and CNN_{global} (the global convolutional network which tackles the whole image) are both initialized from the fully convolutional VGG network [38]. We actually found that the model could achieve a better result if we use a variant of the VGG architecture for CNN_{local} (using more convolutional filters), but keep the current design to make the whole approach more unified.

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 training objective for CRTR is to minimize the sum of negative log-likelihood of conditional probability (the {`image-bounding box-description`} tuple w.r.t. I_{box}, I_{im}, and b_{pos}). The whole CRTR network is trained end-to-end via back propagation.

Fig. 3: Example results of scene text localization. The green bounding boxes contain correct detections; The red bounding boxes contain false positives; The yellow bounding boxes contain false negatives.

3.4 Recurrent Text Recognition

After the targeted text region has been retrieved, we intend to further transcribe it into a sequence of computer-readable characters, namely, text recognition. Recognizing the retrieved text instance can facilitate more real-world applications, and the transcribed *literal* information can further assist the understanding of natural scenes along with the *semantic* and *contextual* information obtained in the previous stages. The text recognizer is able to provide extra recognition outputs, and also regularizes the text localization and retrieval process with precise *literal*-level awareness.

However, even if we have already significantly facilitated the recognition process by focusing on the retrieved text region instead of all potential ones within a natural scene image, a successful transcription is still very challenging. To solve these problems, we consistently follow the CNN-RNN encoder-decoder architecture for text recognition along with previous text localization and retrieval modules. It naturally handles text sequences in arbitrary lengths, involving no character segmentation or horizontal scale normalization. The main components, including a convolutional sequence feature encoder and a recurrent sequence feature decoder similar to CRNN [88], are described in detail at the rest of this section. The verification and filtering of text detection and retrieval results can also benefit from the text recognition module.

Convolutional sequence feature encoder. Given a retrieved text region as input, the VGG-16 architecture is employed to obtain a feature map from which a sequence of feature vectors is extracted as the input for the following recurrent decoder. Specifically, feature vectors are sampled from each column of the feature maps for decoding. Since all the convolutional layers are translation invariant, each column of the feature map corresponds to a rectangle region of the original image, *i.e.*, receptive field. And the rectangle regions are in the same order of their corresponding columns on the feature maps from left to right. Therefore, each vector in the feature sequence is associated with a receptive field, and can be considered as the descriptor of that region.

Different from general CNN encoders which tend to extract one whole holistic representation from an image patch containing the text string, the sequence feature encoder conveys deep features into sequential representation, which is invariant to the large length variation of text instances, and compatible with the recurrent sequence decoding process afterward.

Recurrent sequence modeling and transcription. After the sequential feature vectors have been generated, a stacked bi-directional LSTM is adopted to traverse the sequence feature and decode them into distributions which correspond to all vectors in the feature sequence. The recurrent layers in LSTM are capable of capturing contextual information

within a sequence, which are more effective and stable than dividing a text sequence into individual characters for independent processing. For instance, wide characters may require several successive vectors to fully describe. In addition, some ambiguous characters are easier to distinguish when observing their contexts, *e.g.*, it is easier to recognize "*il*" by contrasting the character heights than recognizing each of them separately. At the bottom of the recurrent layers, the sequence of propagated differentials are concatenated into maps, inverting the operation of converting feature maps into feature sequences, and fed back to the convolutional layers for unified training.

With the predicted distributions over the set of all labels, a transcription process is applied for converting the per-vector predictions made by RNN into a computer readable character sequence. Mathematically, transcription is to find the character sequence with the highest probability conditioned on the per-vector predictions. These sequences are further corrected and refined based on the pre-defined dictionary to generate the final recognition output. The whole encoding-decoding process for text recognition on the retrieved text instance is illustrated in Figure 2.

Localization boosting with recognition results. The text recognition output is further adopted to help eliminate false-positive localization results that are unlikely to be meaningful words, i.e. localization boosting with recognition. Particularly, when a lexicon is present, the recurrent recognizer is capable of removing irrelevant and non-matching bounding boxes (w.r.t given words) effectively.

In practice, the number of output bounding boxes of DTLN has been enough to produce a redundant set of word candidates. The compatibility of each bounding box patch to a particular word is measured, resulting in generating a probability-based matching score for filtering.

4 Experiments

In Sections 4.1 and 4.2, we introduce the details of the text localization and recognition datasets, and the newly collected scene text retrieval dataset. Experimental results and corresponding discussions are presented in Sections 4.3, 4.4, and 4.5.

4.1 Datasets for Text Localization and Recognition

First, the proposed dense text localization method is trained and evaluated on standard benchmarks, including *SynthText* dataset [33], *ICDAR* 2013 dataset [80], and the Street View Text dataset [81]. Then the whole unambiguous text localization framework is evaluated on a newly collected COCO-TextRef dataset.

SynthText in the wild dataset [33]. This is a dataset containing 800,000 synthetic training images, which were generated in [33]. Each image has word instances annotated with character and word-level bounding boxes.

ICDAR 2013 dataset [80]. 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 bounding box annotations. It consists of 229 training images and 233 testing images.

ICDAR 2015 dataset [89]. ICDAR 2015 dataset was used in the Robust Reading Competition under ICDAR 2015. It contains incidental scene text images that are captured without preparation before capturing. 2077 text image patches are cropped from the dataset for the text recognition task, where a large amount of cropped scene texts suffer from perspective and curvature distortions.

Street View Text (SVT) dataset [81]. 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.

COCO-Text dataset [90]. COCO-Text is a large-scale dataset for text detection and recognition in natural images, based on MS-COCO dataset. It contains more than 63,000 images and 173,000 text instances. However, there exist many annotated text instances are illegible for transcription and recognition, due to too small text height or too much occlusion. Our dataset ignores these kinds of illegible text samples.

4.2 New COCO-TextRef Dataset Construction for Text Instance Retrieval

Although there are many datasets for the evaluations of scene text detection and recognition, semantic object retrieval, phrase grounding, and image captioning respectively, no benchmark dataset is available with both bounding-box level scene text annotations and corresponding natural language descriptions upon we started our work. Very recently one dataset for Scene Text Visual Question Answering (ST-VQA) [21] has emerged for the ICDAR 2019 competition but as a different task this dataset still does not provide proper information we need for the targeted task in this manuscript. The ReferIt dataset [6] has been widely used in image captioning and natural language object retrieval. However, it does not provide any image-based annotations or language-based referring expressions for the scene text instances. The ICDAR datasets [80], [89], [91] contains real-world images of text on sign boards, books, posters and other objects. However, most text bounding boxes from these datasets are in extremely focused view and rarely contain useful context concept entities.

We aggregate the information from two existing datasets which benefit from each other, to create a new large-scale dataset for evaluating the proposed scene text detection, retrieval, and recognition framework. Specifically, we select to build our own dataset upon the intersection parts of COCO-Text and Google Refexp Datasets to establish a new dataset containing both text instance annotations and background concept annotations with descriptions. The images commonly shared by COCO-Text and Google Refexp datasets are first selected as the base to build up the dataset. In this case, we already have the annotations of 1) Scene text bounding boxes; 2) Scene text transcriptions; 3) Object bounding boxes and corresponding class labels; 4) Object referring expressions, for each image. Then a list of possible triplet-style ({`text-relationship-context`}) referring expressions are automatically build based on the

query = "largest text on the closest object"

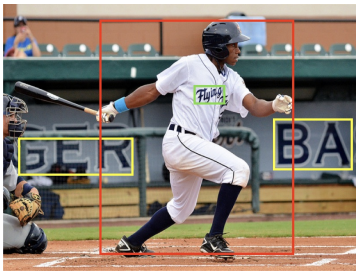
recognition result = "nok"

query = "white text around a bench"

recognition result = "Vango"

query = "largest text left to the right human"

recognition result = "SEATTLE"



query =
"most salient text on the player swinging a bat"

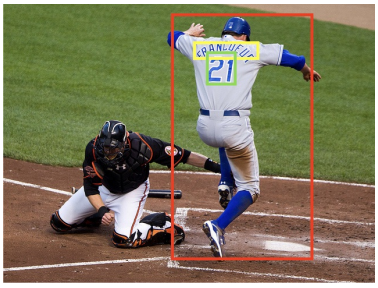
recognition result = "Flying"

query = "largest text on top of a red boat"

recognition result = "RX55"

query = "text on a motorcycle"

recognition result = "POLICE"



query = "digits on the right man"

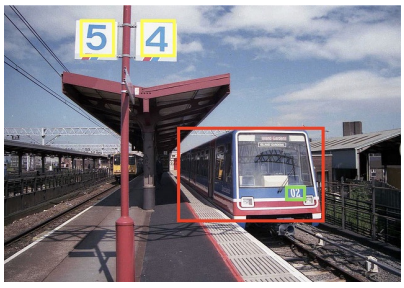
recognition result = "21"

query = "digits on the leftmost man"

recognition result = "8"

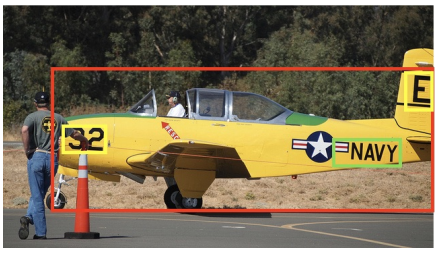
query = "blue text on the largest plane"

recognition result = "DELTA"



query = "text on the front of a train"

recognition result = "02"



query = "text on the right of a yellow plane"

recognition result = "NAVY"

query = "white text on a front black motocycle"

recognition result = "TRIUMPT"

Fig. 4: Text region retrieval and recognition 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. The input queries and output recognition results are listed below each image.

relationship between the scene text instances and objects, and concatenated with existing object-level referring expressions. We finally manually go through all the generated annotations and remove the unreasonable expressions.

Synthetic text instances are rendered on certain images through the method in [33] 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 [87], 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 an NVIDIA Titan X (Maxwell) GPU for 200,000 iterations.

TABLE 2: This table presents the Top-1 precision of our method compared with previous methods on annotated ground truth bounding boxes on the COCO-TextRef dataset.

Method	P@1
LRCN [72]	0.264
DenseCap [51]	0.291
SCRC [50]	0.457
CRTR (ours)	**0.582**

Evaluation protocol. 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 [83] 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 routine introduced by [24] 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 DTLN 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 [24] to further eliminate false-positive detections, it outperforms state-of-the-art methods in terms of recall and achieves comparable precision. In addition, text retrieval and recognition results are shown at Figure 3, demonstrating that the proposed approach effectively tackles the relatively crowded scene text instances, and extracts them from the cluttered and complex background.

TABLE 3: This 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 on the COCO-TextRef dataset.

Method	R@1	R@5
LRCN [72]	0.083	0.213
DenseCap [51]	0.095	0.229
SCRC [50]	0.135	0.313
CRTR (ours)	**0.184**	**0.394**

Based on the analysis of evaluation results and comparison with recent state-of-the-art word-based text detection methods like [33] and [35], our proposed DTLN performs equally well on 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 (false positive) and yellow (false negative) boxes in Figure 3. Intuitively, these failure cases are most likely due to the extreme size and challenging font types/styles. And one main weakness of our proposed text localization module is that the sequential generation procedure might not well handle all these various challenging cases, especially when compared with bottom-up segmentation-based methods.

The classical evaluation protocols for text detection and end-to-end recognition all rely on $precision$ (P), $recall$ (R), and $f\text{-}measure$ (F), which are defined as:

$$\begin{aligned} P &= \frac{TP}{TP+FP} \\ R &= \frac{TP}{TP+FN} \\ F &= 2 \times \frac{P \times R}{P+R} \end{aligned} \tag{1}$$

where TP, FP, and FN is the number of hit boxes, incorrectly identified boxes, and missed boxes, respectively. For text detection, a detected box b is considered as a hit box if the IoU between b and a ground truth box is larger than a given threshold (as mentioned above based on the standard PASCAL VOC criterion or DetEval criterion). The hit boxes in end-to-end recognition require not only the same IoU restriction but also correct recognition results. Since there is a trade-off between precision and recall, $f\text{-}measure$ is the most commonly adopted measurement for overall performance assessment.

4.4 Text Retrieval Experiments

Context Reasoning Text Retrieval (CRTR) model is evaluated on the newly collected COCO-TextRef dataset. Since

TABLE 4: Text recognition results (f-measure value) on ICDAR 2013, ICDAR 2015, and SVT datasets with the recurrent recognition model. For ICDAR 2013 and ICDAR 2015, a full lexicon (90K words) is adopted for the evaluation. The bounding box is fixed with ground truth in *ICDAR15 rec* experiment. The methods marked by "*" are published on the ICDAR 2017 Robust Reading Competition website [91].

Methods	SVT e2e	ICDAR13 rec	ICDAR13 e2e	ICDAR15 rec
Jaderberg *et al*. [24]	0.56	0.76	-	-
FCRNall+filts [33]	0.53	0.85	-	-
Deep2Text II+*	-	0.80	0.77	-
SRC-B-TextProcessingLab*	-	0.81	0.80	-
Adelaide-ConvLSTMs*	-	0.83	0.80	-
TextBoxes (adopting CRNN)	0.64	**0.87**	0.84	0.681
Busta et al. [92]	-	-	0.77	0.674
Li et al. [15]	-	-	0.85	0.690
Liu et al. [93]	-	0.91	-	0.742
Shi et al. [94]	-	**0.92**	-	**0.761**
Proposed	0.65	0.87	0.86	0.694

DenseCap [51] 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 [72] and SCRC [50], 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 and Table 3 compare 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.

Figure 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.

4.5 Text Recognition Experiments

The performance of text recognition is evaluated by detected results that are refined by recognition, *i.e.*, how much improvement on detection performance can we obtain by introducing the text recognition module. And the evaluation of end-to-end performance concerns both detection and recognition results. The ICDAR 2013 and SVT datasets are adopted for the evaluation. As shown in Table. 4, our method outperforms all existing approaches, including the most recent ICDAR 2017 competition results published on the ICDAR website [91]. On the SVT dataset, it also significantly outperforms the leading method [33] by over 12%. Furthermore, when collaborated with a recognition model, *DTLN* achieves state-of-the-art performance on ICDAR end-to-end recognition benchmarks, w.r.t. the related top-performing approaches.

Some challenging scene text recognition examples from COCO-TextRef dataset, as shown in Figure 5, demonstrate the robustness and effectiveness of the encoder-decoder based recurrent text recognition model. In summary, the text recognition module significantly extends the applicability of the proposed DTLN and CRTR models, by introducing

the flexibility of filtering and verifying the text detection results before text retrieval, and generating precise literal information based on the text retrieval results.

TABLE 5: Refined detection results with recognition on ICDAR 2013 dataset: precision (P), Recall (R), and f-measure (F). "Det" – DTLN; "Rec" – recognition without lexicon; "Rec-lex" – recognition with the given strong lexicon.

Datasets	ICDAR13		
	R	P	F
Det	0.79	0.92	0.850
Det + Rec	**0.81**	0.93	0.866
Det + Rec-lex	**0.81**	**0.95**	**0.874**

4.6 Ablation Study on Text Recognition Module

Furthermore, we evaluate how the newly integrated scene text recognition module helps differentiate the scene text instances and the backgrounds, resulting in boosting the text localization performance.

Table 5 presents the results on ICDAR 2013 dataset, for both scene text recognition and end-to-end scene text extraction tasks. The recognition module dramatically improves the detection performance on ICDAR 2013 dataset, especially when a lexicon is available. Specifically, the detection precision benefits more from the recognition regularization. Note that it is still difficult for the current recurrent text recognizer to transcribe vertical or significantly irregular text instances, and extra performance improvement can be expected by integrating a stronger recognition module.

5 Conclusion

Our proposed framework combines vision-based localization and language-based contextual description to extract text information from images, helping the deep understanding of image-based natural scenes. Image-based localization ensures the accurate hit and retrieval of text strings from language-based description, while contextual description enables the user-friendly delivery of the localized text instances. The text recognition model further extends the applicability of the whole text extraction approach.

In future, scene text localization, retrieval, and recognition will be combined into an end-to-end trainable system. The detection and recognition performance can also be further improved with pre-processing techniques such as image super-resolution [95], [96] and deblurring [97], [98].

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

Fig. 5: Challenging samples of scene text from COCO-TextRef dataset, which are correctly recognized by the recurrent text recognition model: "AMER", "YALLE", "plus", "FAIRHAVEN", "ASTED", "LIBERTY", "FORCE", "CLEARANCE", "193", "SWEET", "Sauza", "PROCTOR", "Kawasaki", "Patcham", and "TELEPHONE".

References

[1] Bo Xiong and Kristen Grauman. Text Detection in Stores Using a Repetition Prior. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 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] Wikipedia. Cognitive neuroscience of visual object recognition. https://tinyurl.com/nerorec, 2017.

[4] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg. "Modeling Context in Referring Expressions". *in ECCV*, 2016.

[5] J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy. "Generation and Comprehension of Unambiguous Object Description". *in CVPR*, 2016.

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

[7] Xuejian Rong, Chucai Yi, and Yingli Tian. Unambiguous Text Localization and Retrieval for Cluttered Scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[8] Shangbang Long, Xin He, and Cong Yao. Scene text detection and recognition: The deep learning era. *arXiv preprint arXiv:1811.04256*, 2018.

[9] Chuhui Xue, Shijian Lu, and Wei Zhang. MSR: Multi-Scale Shape Regression for Scene Text Detection. *in IJCAI*, 2019.

[10] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character Region Awareness for Text Detection. *in CVPR*, 2019.

[11] Fangneng Zhan, Chuhui Xue, and Shijian Lu. GA-DAN: Geometry-Aware Domain Adaptation Network for Scene Text Detection and Recognition. *in ICCV*, 2019.

[12] Chuhui Xue, Shijian Lu, and Fangneng Zhan. Accurate Scene Text Detection through Border Semantics Awareness and Bootstrapping. *in ECCV*, 2018.

[13] Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. An end-to-end textspotter with explicit alignment and attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5020–5029, 2018.

[14] Michal Busta, Lukas Neumann, and Jiri Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[15] Hui Li, Peng Wang, and Chunhua Shen. Towards end-to-end text spotting with convolutional recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5238–5246, 2017.

[16] 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*, pages 5551–5560, 2017.

[17] Fan Bai, Zhanzhan Cheng, Yi Niu, Shiliang Pu, and Shuigeng Zhou. Edit Probability for Scene Text Recognition. *in CVPR*, 2018.

[18] Zhanzhan Cheng, Yangliu Xu, Fan Bai, Yi Niu, Shiliang Pu, and Shuigeng Zhou. Aon: Towards arbitrarily-oriented text recognition. *in CVPR*, 2018.

[19] Shijian Lu Fangneng Zhan. ESIR: End-to-end Scene Text Recognition via Iterative Image Rectification. *in CVPR*, 2019.

[20] Mingkun Yang, Yushuo Guan, Minghui Liao, and Xin He. Symmetry-constrained Rectification Network for Scene Text Recognition. *in ICCV*, 2019.

[21] Ali Furkan Biten et al. Scene Text Visual Question Answering. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2019.

[22] Xuejian Rong, Chucai Yi, and Yingli Tian. Unambiguous scene text segmentation with referring expression comprehension. *IEEE Transactions on Image Processing*, 29:591–601, 2019.

[23] Shitala Prasad and Adams Wai Kin Kong. Using object information for spotting text. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[24] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading Text in the Wild with Convolutional Neural Networks. *International Journal of Computer Vision*, 116(1):1–20, 2016.

[25] B. Epshtein, E. Ofek, and Y. Wexler. "Detecting Text in Nature Scenes with Stroke Width Transform". *in CVPR*, 2010.

[26] L. Neumann and J. Matas. "Real-time Scene Text Localization and Recognition". *in CVPR*, 2012.

[27] X. Yin, K. Huang, and H. Hao. "Robust Text Detection in Natural Scene Images". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2014.

[28] X. Rong, C. Yi, X. Yang, and Y. Tian. "Scene Text Recognition in Multiple Frames based on Text Tracking". *in ICME*, 2014.

[29] X. Yin, W. Pei, J. Zhang, , and H. Hao. "Multi-orientation Scene Text Detection with Adaptive Clustering". *in TPAMI*, 2015.

[30] M. Jaderberg, A. Vedaldi, and A. Zisserman. "Deep Features for Text Spotting". *in ECCV*, 2014.

[31] Zheng Zhang, Chengquan Zhang, Wei Shen, Cong Yao, Wenyu Liu, and Xiang Bai. Multi-Oriented Text Detection with Fully Convolutional Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[32] X. Rong, C. Yi, and Y. Tian. "Recognizing Text-based Traffic Guide Panels with Cascaded Localization Network". *in ECCV Workshop*, 2016.

[33] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic Data for Text Localisation in Natural Images. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

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

[35] Z. Tian, W. Huang, T. He, Pan. He, and Y. Qiao. "Detecting Text in Natural Image with Connectionist Text Proposal Network". *in ECCV*, 2016.

[36] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[37] S. Ioffe and C. Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". *in ICML*, 2015.

[38] K. Simonyan and A. Zisserman. "Very Deep Convolutional Networks For Large-Scale Image Recognition". *in ICLR*, 2015.

[39] S. Hochreiter and J. Schmidhuber. "Long Short Term Memory". *Neural Computation*, 1997.

[40] A. Karpathy and Fei-Fei Li. "Deep Visual-Semantic Alignments for Generating Image Descriptions". *in CVPR*, 2015.

[41] I. Sutskever, O. Vinyals, and Q. V. Le. "Sequence to Sequence Learning with Neural Networks". *in NIPS*, 2014.

[42] Dan Deng, Haifeng Liu, Xuelong Li, and Deng Cai. Pixellink: Detecting scene text via instance segmentation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[43] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng. "End-to-end Text Recognition with Convolutional Neural Networks". *in ICPR*, 2012.

[44] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. "PhotoOCR: Reading Text in Uncontrolled Conditions". *in ICCV*, 2013.

[45] Bolan Su and Shijian Lu. Accurate Scene Text Recognition Based on Recurrent Neural Network. *in ACCV*, 2014.

[46] Bolan Su and Shijian Lu. Accurate recognition of words in scenes without character segmentation using recurrent neural network. *Pattern Recognition*, 63:397–405, 2017.

[47] Shangxuan Tian, Ujjwal Bhattacharya, Shijian Lu, Bolan Su, Qingqing Wang, Xiaohua Wei, Yue Lu, and Chew Lim Tan. Multilingual scene character recognition with co-occurrence of histogram of oriented gradients. *Pattern Recognition*, 51:125–134, 2016.

[48] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.

[49] Andrej Karpathy, Armand Joulin, and Li F Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, pages 1889–1897, 2014.

[50] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell. "Natural Language Object Retrieval". *in CVPR*, 2016.

[51] J. Johnson, A. Karpathy, and Li Fei-Fei. "DenseCap: Fully Convolutional Localization Networks for Dense Captioning". *in CVPR*, 2016.

[52] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2018.

[53] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. *arXiv preprint arXiv:1811.10830*, 2018.

[54] E. Krahmer and K. van Deemter. "Computational Generation of Referring Expressions". *Comp. Linguistics*, 2012.

[55] W. Choi, Y. Chao, C. Pantofaru, and S. Savarese. "Understanding indoor scenes using 3d geometric phrases". *in CVPR*, 2013.

[56] B. Dai, Y. Zhang, and D. Lin. "Detecting visual relationships with deep relational networks". *in CVPR*, 2017.

[57] M. Sadeghi and A. Farhadi. "Recognition using visual phrases". *in CVPR*, 2011.

[58] M. Kumar and D. Koller. "Efficiently selecting regions for scene understanding". *in CVPR*, 2010.

[59] Yikang Li, Wanli Ouyang, Xiaogang Wang, and Xiaoou Tang. "ViP-CNN: Visual Phrase Guided Convolutional Neural Network". *in CVPR*, 2017.

[60] D. Xu, Y. Zhu, C. Choy, and L. Fei-Fei. "Scene graph generation by iterative message passing". *arXiv preprint arXiv:1701.02426*, 2017.

[61] B. Plummer, A. Mallya, C. Cervantes, J. Hockenmaier, and S. Lazebnik. "Phrase localization and visual relationship detection with comprehensive linguistic cues". *arXiv preprint arXiv:1611.06641*, 2016.

[62] H. Zhang *et al.* . "Visual Translation Embedding Network for Visual Relation Detection". *arXiv:1702.08319*, 2017.

[63] B. Zhuang, L. Liu, C. Shen, and I. Reid. "Towards context-aware interaction recognition". *in ICCV*, 2017.

[64] B. Zhuang, Q. Wu, C. Shen, I. Reid, and A. Hengel. "Care about you: towards large-scale human-centric visual relationship detection". *arXiv preprint arXiv:1705.09892*, 2017.

[65] C. Lu, R. Krishna, M. Bernstein, and F. Li. "Visual Relationship Detection". *in ECCV*, 2016.

[66] X. Liang, L. Lee, and E. Xing. "Deep variation-structured reinforcement learning for visual relationship and attribute detection". *in CVPR*, 2017.

[67] Mark Yatskar, Vicente Ordonez, and Ali Farhadi. Stating the obvious: Extracting visual common sense knowledge. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 193–198, 2016.

[68] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning with neural tensor networks for knowledge base completion. In *Advances in neural information processing systems*, pages 926–934, 2013.

[69] Mohammad Amin Sadeghi and Ali Farhadi. Recognition using visual phrases. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1745–1752. IEEE, 2011.

[70] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

[71] V. Ramanathan et al. Learning semantic relationships for better action retrieval in images. *CVPR*, 2015.

[72] 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.

[73] J. Mao *et al*. . "Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN)". *in ICLR*, 2015.

[74] A. Rohrbach *et al*. . "Grounding of Textual Phrases in Images by Reconstruction". *in ECCV*, 2016.

[75] B. Plummer *et al*. . "Flickr30k entities: Collecting region-to-phrase correspondences for riche image-to-sentence models". *in ICCV*, 2015.

[76] R. Hu, M. Rohrbach, and T. Darrell. "Segmentation from Natural Language Expressions". *in ECCV*, 2016.

[77] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille. Recurrent multimodal interaction for referring image segmentation. *arXiv preprint arXiv:1703.07939*, 2017.

[78] J. Long and T. Darrell E. Shelhamer. "Fully Convolutional Models for Semantic Segmentation". *in CVPR*, 2015.

[79] R. Stewart and M. Andriluka. "End-to-end People Detection in Crowded Scenes". *in CVPR*, 2016.

[80] D. Karatzas. "ICDAR 2013 Robust Reading Competition". *in ICDAR*, 2013.

[81] K. Wang and S. Belongie. "Word Spotting in the Wild". *in ECCV*, 2010.

[82] 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.

[83] 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.

[84] S. Lu, T. Chen, S. Tian, J. Lim, and C. Tan. "Scene Text Extraction based on Edges and Support Vector Regression". *in IJDAR*, 2015.

[85] Zheng Zhang, Wei Shen, Cong Yao, and Xiang Bai. Symmetry-based text line detection in natural scenes. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015.

[86] M. Liao *et al*. . "TextBoxes: A Fast Text Detector with a Single Deep Neural Network". *in AAAI*, 2017.

[87] 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.

[88] 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*, 39(11):2298–2304, 2017.

[89] Xinyu Zhou, Shuchang Zhou, Cong Yao, Zhimin Cao, and Qi Yin. Icdar 2015 text reading in the wild competition, 2015.

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

[91] ICDAR. ICDAR 2017 Robust Reading Competition Website. http://rrc.cvc.uab.es, 2017.

[92] Michal Busta, Lukas Neumann, and Jiri Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2204–2212, 2017.

[93] Wei Liu, Chaofeng Chen, and Kwan-Yee K Wong. Char-net: A character-aware neural network for distorted scene text recognition. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

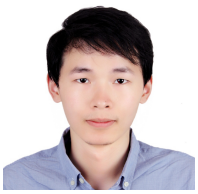
[94] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2035–2048, 2018.

[95] R. Dahl, M. Norouzi, and J. Shlens. "Pixel Recursive Super Resolution". *arXiv:1702.00783*, 2017.

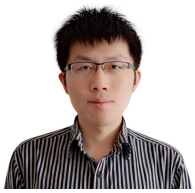
[96] Y. Xian and Y. Tian. "Resolution Enhancement in Single Depth Map and Aligned Image". *in WACV*, 2016.

[97] X. Rong and Y. Tian. "Adaptive Shrinkage Cascades for Blind Image Deconvolution". *in DSP*, 2016.

[98] J. Pan, D. Sun, H. Pfister, and M. Yang. "Blind Image Deblurring Using Dark Channel Prior". *in CVPR*, 2016.



Xuejian Rong is a researcher currently working at Facebook on 3D vision and computational photography. Previously, he was a PhD student studying at City University of New York under the supervision of Prof. Yingli Tian. He worked as research interns in Facebook Research and Siemens Research. He received the B.E. degree from Nanjing University of Aeronautics and Astronautics with honors thesis in 2013. Xuejian's research interests are in visual recognition and machine learning with a focus on deep learning based scene text extraction and understanding. He also worked in the areas of image degradation removal such as image deblurring and denoising.



Chucai Yi is currently working at Google corporate on computer vision research and development. Before joining Google, he worked at HERE Map Technologies and Amazon corporates. He received his Ph.D. degree in Computer Science in 2014 from The Graduate Center of City University of New York. As an academic researcher, his research work includes object/signage detection and recognition, human involved event detection, 3D point cloud processing, and camera calibration. As a professional software developer, he has been designing and implementing end-to-end applications and services that apply machine learning and computer vision techniques to solve specific problems on large-scale data.

Yingli Tian (M'99 – SM'01 – F'18) received the B.S. and M.S. degrees from Tianjin University, China, in 1987 and 1990, and the Ph.D. degree from Chinese University of Hong Kong, Hong Kong, in 1996. After holding a faculty position at National Laboratory of Pattern Recognition, Chinese Academy of Sciences, Beijing, she joined Carnegie Mellon University in 1998, where she was a postdoctoral fellow at the Robotics Institute. She then worked as a research staff member in IBM T. J. Watson Research Center from 2001 to 2008. She is one of the inventors of the IBM Smart Surveillance Solutions. She has been with the City University of New York (CUNY) since 2008, and currently she is a CUNY distinguished professor in the Department of Electrical Engineering at the City College and the department of Computer Science at the Graduate Center. Her research focuses on a wide range of computer vision problems from object recognition, scene understanding, human behavior analysis, facial expression recognition, to medical imaging analysis and assistive technology. She is a fellow of IEEE.