Front.Comput.Sci.
DOI

REVIEW ARTICLE

# Scene Text Detection and Recognition: Recent Advances and Future Trends

**Yingying ZHU, Cong YAO, Xiang BAI***

School of Electronic Information and Communications,
Huazhong University of Science and Technology (HUST),
Wuhan, 430074, China

**Abstract** Text, as one of the most influential inventions of humanity, has played an important role in human life, so far from ancient times. The rich and precise information embodied in text is very useful in a wide range of vision-based applications, therefore text detection and recognition in natural scenes have become important, active research topics in computer vision and document analysis. Especially in recent years, the community has seen a surge of research efforts and substantial progresses in these fields, though a variety of challenges (e.g. noise, blur, distortion, occlusion and variation) still remain. The purposes of this survey are three-fold: (1) introduce up-to-date works, (2) identify state-of-the-art algorithms, and (3) predict potential research directions in the future. Moreover, this paper provides comprehensive links to publicly available resources, including benchmark datasets, source codes, and online demos. In summary, this literature review can serve as a good reference for researchers in the areas of scene text detection and recognition.

**Keywords** Text detection, text recogntion, natural image, algorithms, applications

## 1 Introduction

As a product of human abstraction and manipulation, text in natural scenes directly carries high level semantics. This property makes text present in natural images and videos a special, important source of information. The rich and precise information embodied in text can be very beneficial to a variety of vision-based applications, such as image search [1], target geolocation [2], human-computer interaction [3], robot navigation [4] and industrial automation [5]. Consequently, automatic text detection and recognition, offering a means to access and utilize textual information in images and videos, have become active research topics in computer vision and document analysis.

However, localizing and reading texts in natural scenes are extremely difficult tasks. The major challenges in scene text detection and recognition can be roughly categorized into three types [6, 7]:

- **Diversity of scene text:** In contrast to characters in document images, which are usually with regular font, single colour, consistent size and uniform arrangement, texts in natural scenes may bear entirely different fonts, colours, scales and orientations, even in the same scene.
- **Complexity of background:** The backgrounds in natural scene images and videos can be very complex. Elements like signs, fences, bricks and grasses are virtually undistinguishable from true text, and thus are easily to cause confusions and errors.
- **Interference factors:** Various interference factors, for instance, noise, blur, distortion, low resolution, non-uniform illumination and partial occlusion, may give rise to failures in scene text detection and recognition.

To tackle these challenges, a rich body of approaches have been proposed and substantial progresses have been achieved in recent years [8–20]. In almost all these methods, the study on representation is the main research subject, since represen-

Received month dd, yyyy; accepted month dd, yyyy

E-mail: xbai@hust.edu.cn

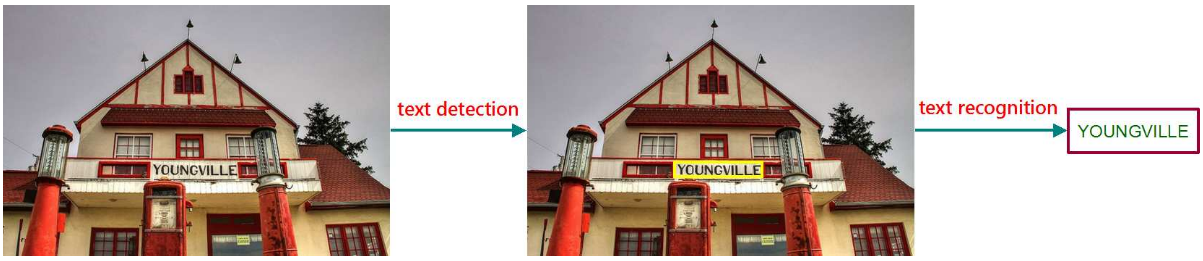

**Fig. 1** Illustration of scene text detection and recognition.

tation is the key to the effectiveness and robustness of these algorithms. In scene text detection and recognition, representation involves the way and manner of describing and modelling text and background in natural scenes.

In this paper, we present a comprehensive literature review of works on scene text detection and recognition in the past few years, mainly from the perspective of representation. This survey is dedicated to: (1) introduce up-to-date works and summarize recent advances, (2) compare different methods and highlight state-of-the-art algorithms, and (3) analyse development tendency and predict future research directions. Moreover, it provides links to useful resources, including benchmark datasets, source codes, and online demos.

There are already several excellent review papers [21–23] in the fields of scene text detection and recognition. However, these review papers are somewhat outmoded, since they were published about 10 years ago and missed numerous important, influential works that are proposed in recent years. The only two near-term surveys we are aware of are the works of Zhang *et al.* [24] and Uchida *et al.* [25]. The survey of Zhang *et al.* [24] has mainly focused on papers related to scene text detection, but ignored methods on text recognition. The work of Uchida *et al.* [25] reviewed methods for text detection and recognition in images as well as videos, but it was partial to works from the document analysis community and neglected several up-to-date works from the computer vision community, which have introduced new insights and ideas [7, 19, 20, 26]. Different from the previous review papers [21–25], this article provides a comprehensive survey on scene detection and recognition in static images, with a special emphasis on the latest advances in these areas.

The remainder of this article is structured as follows. First, we review recent works in the fields of scene text and recognition in Sec. 2. Then, we describe related benchmark datasets and evaluation methods in Sec. 3. In Sec. 4, we present our observations, thoughts and opinions on the current research status of scene text detection and recognition. Finally, conclusion remarks and promising research directions in the future are given in Sec. 5.

## 2 Recent Advances in Scene Text Detection and Recognition

In recent years, text detection and recognition in natural images have become active research topics in the communities of computer vision, pattern recognition and even document analysis. Researchers from these communities have proposed a large amount of novel ideas and approaches for the extraction of textual information from in natural images and videos.

These methods can be broadly divided into three categories: (1) text detection, (2) text recognition, (3) end-to-end text recognition, as demonstrated in Fig. 1. The first category of methods [9, 12, 27–30] concern how to discover and locate the regions possibly containing text from natural images, but do not need to preform recognition. The second category of methods [7, 14, 15, 31–33] suppose that texts have been detected, and only focus on the process of converting the detected text regions into computer readable and editable symbols. The third category of methods [10, 11, 13, 17, 19, 34] aim at constructing end-to-end text recognition systems that accomplish both the detection and recognition tasks.

### 2.1 Related Works on Scene Text Detection

In the past two decades, researchers have proposed numerous methods for detecting texts in natural images or videos. There are mainly three types of methods: texture based methods, component based methods and hybrid methods.

**Texture based method** [8, 35–37] treat texts as a special type of texture and make use of their textural properties, such as local intensities, filter responses and wavelet coefficients, to distinguish between text and non text areas in the images. These methods are usually computationally expensive as all locations and scales should be scanned. In addition, these methods mostly handle horizontal texts and are sensitive to rotation and scale change.

In an early work, Zhong *et al.* [35] proposed a method for text localization in color images. Horizontal spatial vari-

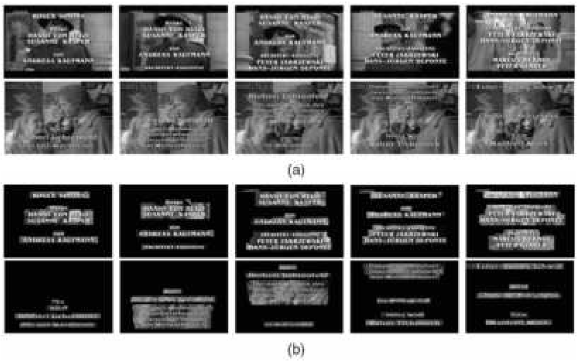

**Fig. 2** Text detection examples of the algorithm of Kim *et al* [36]. This algorithm is a representative work of early stage methods for text detection. It is only applicable to relatively simple scenes.

ance was utilized to roughly localize texts and then color segmentation was performed within the localized regions to find texts. Later, Li *et al.* [38] introduced a text detection system to detect and track texts in videos. In this system, images are decomposed by using the mean of wavelet coefficients, and the first-order and second-order moments as local features.

Kim *et al.* [36] trained a SVM classifier to classify each pixel by directly using the raw pixel intensity as local feature. Text areas were sought via adaptive MeanShift [39] in probability maps. The method produces excellent detection results in images or videos (Fig. 2) with simple backgrounds, but it is difficult to generalize this method to complex natural scene images or videos.

To handle multilingual texts (mainly Chinese and English) in videos, Lyu et al. [40] proposed a coarse-to-fine multiscale search scheme. The scheme used properties such as strong edge and high contrast of texts to distinguish between text and non-text regions. Moreover, this algorithm provides a local adaptive binary strategy to segment detected text areas. Similar to many other approaches, this method involves numerous rules and parameters, so it is very hard for it to deal with videos of different qualities and texts of different types.

Different from conventional methods, Zhong *et al.* [41] proposed an interesting algorithm that can directly detect text in the Discrete Cosine Transform (DCT) domain. The advantage of this algorithm lies in its high efficiency, as it is not necessary to decode the image before detection. However, the detection accuracy of this method is limited.

In order to speed up the text detection procedure, Chen *et al.* [8] proposed a fast text detector. The detector is a cascade Adaboost [42] classifier, in which each weak classifier is trained from a set of features. The feature pool includes mean strength, intensity variance, horizontal difference, vertical difference, and gradient histogram. The detection efficiency of this method is significantly higher than other algorithms [43–45], but the detection accuracy on real-world images is limited.

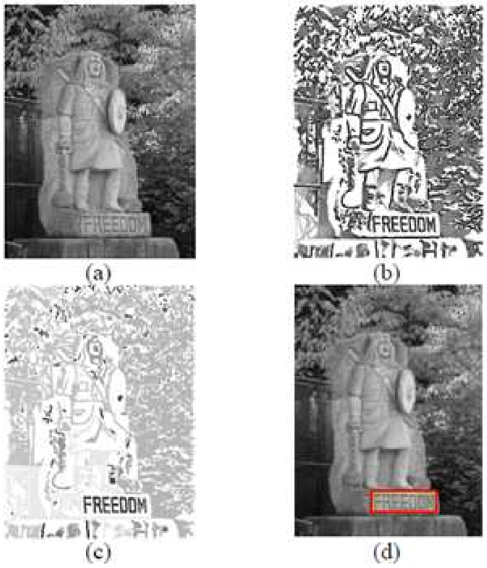

**Fig. 3** Text detection examples of the algorithm of Epshtein *et al.* [9]. This work proposed SWT, an image operator that allows for direct extraction of character strokes from edge map.

Recently, Wang *et al.* [46] proposed a method for locating specific words from natural scenes. Firstly, single characters are detected by sliding window. Then, possible combinations are scored according to the structural relationships between characters. Finally, the most similar combinations are selected from the given list as the output results. Unlike traditional text detection methods, this algorithm can only detect words in the given list, incapable of handling words out of the given list. In reality, however, a word list that contains all possible words is not always available for each image. This makes the applicability range of the method narrow, compared to other text detection methods.

**Component based methods** [9, 12, 28, 29, 47] first extract candidate components through a variety of ways (e.g., color clustering or extreme region extraction), and then filter out non-text components using manually designed rules or automatically trained classifiers. Generally speaking, these methods are much more efficient, because the number of components to be processed is relatively small. In addition, these methods are insensitive to rotation, scale change and font variation. In recent years, component based methods have become the mainstream in the field of scene text detection.

The method proposed by Jain *et al.* [47] decomposed images into several non-overlapping components by color clustering, grouped components into text lines through component analysis, and then removed non-text components according to geometric rules. Because of the artificially defined

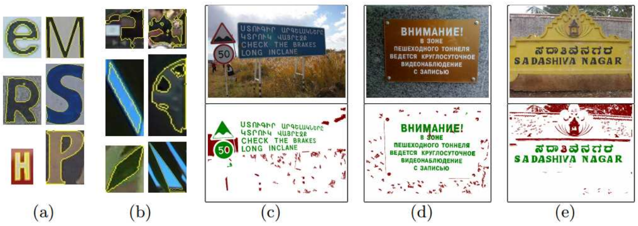

**Fig. 4** Text detection examples of the algorithm of Neumann *et al.* [10]. This work is the first that introduces MSER into the field of scene text detection.

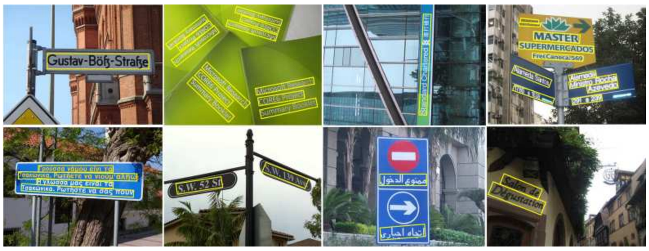

**Fig. 5** Text detection examples of the algorithm of Yao *et al.* [12]. Different from previous methods, which have focused on horizontal or near-horizontal texts, this algorithm is able to detect texts of varying orientations in natural images.

rules and parameters, this method performs poorly on complex natural images.

Making use of the property that characters have nearly constant stroke width, Epshtein *et al.* [9] proposed a new image operator: Stroke Width Transform (SWT). This operator provides an easy way to recover character strokes from edge maps and is able to efficiently extract text components of different scales and directions from complex scenes (Fig. 3). However, this method also comes with a series of human-defined rules and parameters, and only considers horizontal texts.

Neumann *et al.* [10] proposed a text detection algorithm based on Maximally Stable Extremal Regions (MSER). This algorithm extracts from the original images MSER regions as candidates, and eliminates invalid candidates using a trained classifier (Fig. 4). At a later stage, the remained candidates are grouped into text lines through a series of connection rules. However, such connection rules can only adapt to horizontal or nearly horizontal texts, therefore this algorithm is unable to handle texts with larger inclination angle.

SWT [9] and MSER [10] are two representative methods in the field of scene text detection, which constitute the basis of a lot of subsequent works [12–14, 29, 30, 34, 48, 49].

The great success of sparse representation in face recognition [50] and image denoising [51] has inspired numerous researchers. For example, Zhao *et al.* [52] constructed a sparse dictionary from training samples and used it to judge whether a particular area in the image contains text. However, the generalization ability of the learned sparse dictionary is restricted, so that this method is unable to handle issues like rotation and scale change.

Different from the aforementioned algorithms, the approach proposed by Yi *et al.* [28] can detect tilted texts in natural images. Firstly, the image is divided into different regions according to the distribution of pixels in color space, and then regions are combined into connected components according to the properties such as color similarity, spatial distance and relative size of regions. Finally, non-text components are discarded by a set of rules. However, the prerequisite of this method is that it assumes the input images consists of several main colors, which is not necessarily true for complex natural images. In addition, this method relies on a lot of artificially designed filtering rules and parameters, so that it is difficult to generalize to large-scale complex image data sets.

Shivakumara *et al.* [53] also proposed a method for multi-oriented text detection. The method extracted candidate regions by clustering in the Fourier-Laplace space and divided the regions into distinct components using skeletonization. However, these components generally do not correspond to strokes or characters, but just text blocks. This method can not directly compare with other methods quantitatively, since it is not able to detect characters or words directly.

Based on SWT [9], Yao *et al.* [12] proposed an algorithm that can detect texts of arbitrary orientations in natural images (Fig. 5). This algorithm is equipped with a two-level classification scheme and two sets of rotation and rotation-invariant features specially designed for capturing the intrinsic characteristics of characters in natural scenes.

Huang *et al.* [29] presented a new operator based on Stroke Width Transform, called Stroke Feature Transform (SFT). In order to solve the mismatch problem of edge points in the original Stroke Width Transform, SFT introduces color consistency and constrains relations of local edge points, producing better component extraction results. The detection performance of SFT on standard datasets is significantly higher than other methods, but only for horizontal texts.

In [30], Huang *et al.* proposed a novel framework for scene text detection, which integrated Maximally Stable Extremal Regions (MSER) and Convolutional Neural Networks (CNN). The MSER operator works in the front-end to extract text candidates, while a CNN based classifier is applied to correctly identify true text candidates and separate the connections of multiple characters in components. This algorithm achieves significantly enhanced performance over conventional methods.

**Hybrid methods** [27, 54] are a combination of texture based methods and component based methods, which make use of the advantages of these two types methods. In the method propsed by Liu *et al.* [54], edge pixels of all possible text regions were extracted using an elaborate edge detection strategy, and the gradient and geometrical properties of region contours are verified to generate candidate text regions, followed by a texture analysis procedure to distinguish true text regions from non-text regions.

Unlike [54], the hybrid method proposed by Pan *et al.* [27] extracts candidate components from multi-scale probability maps. The probability maps are estimated by a classifier, which is trained on a set of texture features (HOG features [55]) computed with a group of predefined patterns. A Conditional Random Filed (CRF) model [56], combining unary component properties and binary contextual relationships, is utilized to discriminate text components from non-text components. Like most other algorithms, these two methods can only detect horizontal texts.

The advantages and limitations of the existing methods for scene text detection are summarized in Tab. 1.

## 2.2 Related Works on Scene Text Recognition

Since the properties of natural images are greatly different from document images, there would be many obstacles if directly applying traditional character recognition methods to natural images. For example, these methods may produce plenty of false alarms and gibberish, when running on natural images.

In order to tackle these problems, Sawaki *et al.* [57] proposed a method which can automatically create character templates according to the characteristics of natural images. Zhou *et al.* [58, 59] used surface fitting classifier and specifically designed character recognition algorithm to identify characters in Internet images (including simple synthetic images and natural images). However, these algorithms were not evaluated on complex natural images, so the adaptability of these methods have not been sufficiently validated.

In [60], de Campos *et al.* tested, compared and analysed the current commonly used feature descriptors and classification algorithms in computer vision and pattern recognition. In addition they released a image dataset, called Chars74K, for evaluating character recognition algorithm. Chars74K has been widely accepted and used in the field of character recognition in natural images. However, unlike the mainstream character recognition methods, which treat word as the basic unit, the method of de Campos *et al.* [60] only consider the problem of individual character recognition.

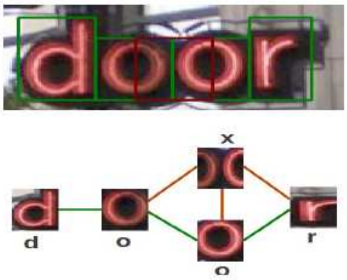

**Fig. 6** Bottom-up and top-down cues for scene text recognition [31]. In this method, cues from low level (character detection) and high level (language prior) are integrated into a unified framework.

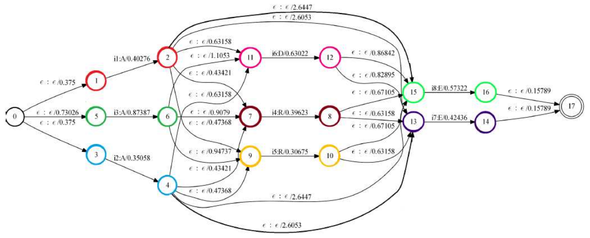

**Fig. 7** Weighted Finite-State Transducers for scene text recognition [14]. This work is similar with [31], since it also used cues from both low level and high level.

Mishra *et al.* [31] employed bottom-up and top-down cues for scene text recognition, which works in an error correction manner [61]. Due to the presence of complex backgrounds in natural scenes, it is very difficult to directly segment characters from local background. So this method uses sliding window to detect possible characters, and treat the detection results as the bottom-up information. The top-down information comes from the statistics of a large dictionary. The bottom-up and top-down information are integrated in a unified model through Conditional Random Field (CRF) [56]. One of the advantages of this method is that is can tolerate errors in character detection. As shown in Fig. 6, the region between two 'o's is regarded as the character 'x', but according to the prior information, the possibility of 'oor' is higher than 'oxr', so the word is identified as 'door' finally.

Recently, Mishra *et al.* proposed a new text recognition method [15] based on the algorithm in [31]. This method introduces an error correction model, which take full advantage of higher order prior information, further boosting the recognition accuracy.

Novikova *et al.* [14] proposed to characterize character appearance and the relationship between characters via a unified probabilistic model. Unlike the algorithm in [31], character candidates are extracted using MSER [62]. This method adopts Weighted Finite-State Transducers [63] as the proba-

**Table 1** Comparison of existing text detection methods.

| **Algorithm** | **Year** | **Strength** | **Weakness** |
|---|---|---|---|
| Zhong *et al.* [35] | 1995 | • can detect text in natural images | • only applicable to simple images |
| Jain *et al.* [47] | 1998 | • can detect text in natural images | • only applicable to simple images<br>• rely on manually defined rules |
| Kim *et al.* [36] | 2003 | • can detect text in natural images and videos | • only applicable to simple images<br>• only applicable to horizontal texts |
| Li *et al.* [38] | 2000 | • can detect and track texts in videos | • only applicable to horizontal texts |
| Chen *et al.* [8] | 2004 | • can detect text in complex images<br>• fast | • only applicable to horizontal texts |
| Lyu *et al.* [40] | 2005 | • can detect text in videos<br>• can handle multilingual texts | • only applicable to horizontal texts |
| Liu *et al.* [54] | 2006 | • can detect text in natural images | • only applicable to horizontal texts |
| Wang *et al.* [46] | 2010 | • can detect text in complex natural images | • only applicable to horizontal texts<br>• require a lexicon for each image |
| Epshtein *et al.* [9] | 2010 | • can detect text in complex natural images<br>• can handle multilingual texts<br>• reasonably fast | • only applicable to horizontal or near-horizontal texts<br>• rely on manually defined rules |
| Neumann *et al.* [10] | 2010 | • can detect text in complex natural images<br>• reasonably fast | • only applicable to horizontal or near-horizontal texts |
| Yi *et al.* [28] | 2011 | • can detect texts of different orientations<br>• can handle multilingual texts | • only applicable to simple natural images<br>• rely on manually defined rules |
| Shivakumara *et al.* [53] | 2011 | • can detect texts of different orientations | • produce text blocks, instead of words or lines<br>• rely on manually defined rules |
| Yao *et al.* [12] | 2012 | • can detect texts of different orientations<br>• can handle multilingual texts<br>• reasonably fast | • rely on manually defined rules |
| Huang *et al.* [29] | 2013 | • can detect text in complex natural images<br>• robust | • only applicable to horizontal texts<br>• rely on manually defined rules |
| Huang *et al.* [30] | 2014 | • can detect text in complex natural images<br>• excellent performance | • only applicable to horizontal texts |

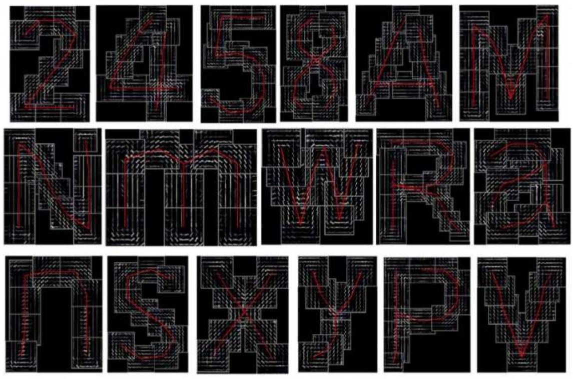

Fig. 8 Part-based tree-structured model for scene text recognition [32]. The structures of characters are manually designed and the parts are annotated by hand.

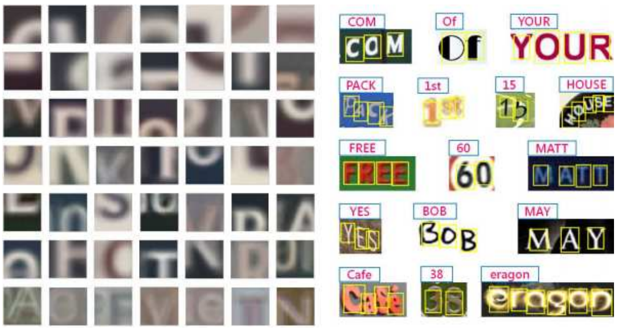

Fig. 9 Strokelets for scene text recognition [7]. In contrast to [32], the parts of characters (i.e. strokelets) are automatically learned from training data.

bilistic model (see Fig. 7) and searches the most likely word by an efficient reasoning algorithm. However, the procedure of this method is complicated and its word recognition performance has no obvious advantage over other error correction methods that also utilize statistic language models [15, 31].

Rodriguez-Serrano *et al.* [64] explored a new approach for text recognition, in which label embedding was utilized to directly perform matching between strings and images, bypassing pre- or post-processing operations.

In the past two years, part based text recognition algorithms [7, 32] have become very popular. Shi *et al.* [32] proposed a part-based tree-structured model to recognize characters in cropped images. This algorithm is robust to nose, blur, partial occlusion and font variation. However, this algorithm depends on the detailed annotation information, including character models and part annotations (Fig. 8).

In [7], Yao *et al.* proposed a novel representation, call Strokelets, which consists of a set of multi-scale mid-level elements (Fig. 9). Strokelets can be automatically learned from character level labels and are able to capture the structural properties of characters at different granularities. Moreover, strokelets provide an alternative way to accurately identify individual characters and compose a histogram feature to effectively describe characters. The scene text recognition algorithm based on strokelets has proven to be both effective and robust.

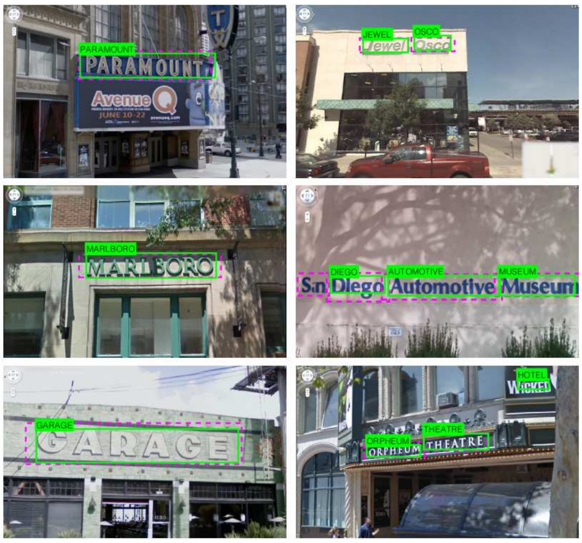

Fig. 10 End-to-end recognition examples of [11]. This algorithm accomplishes both text detection and recognition, but requires that a lexicon is given for each test image in advance.

The advantages and limitations of the existing methods for scene text recognition are summarized in Tab. 2.

## 2.3 Related Works on End-to-End Text Recognition

The above-mentioned methods only concern one aspect of the problem in text information extraction [22] (either text detection or text recognition). There are a variety of methods that attempt to construct a unified framework for both text detection and recognition.

Base on the work in [46], Wang *et al.* [11] proposed an end-to-end text recognition system[1] (see Fig. 10). Inspired by general object detection algorithms in computer vision, this method treats words as a special kind of object, and characters as parts of the object. It searches the most possible detection and recognition results by modelling each single character and the spatial relationship between characters. Experiments show that this method obtains excellent performance on multiple standard datasets. However, this algorithm can only handle words that are within the given word list, thus it is not applicable to images without a word list.

The first real end-to-end text recognition system for natural images is proposed by Neumann *et al.* [10], which does not requires a word list. This system extracts the character candidates via MSER and eliminates non-text candidates through a trained classifier. The remaining candidates are fed

[1] Code available at http://vision.ucsd.edu/ kai/grocr/

**Table 2** Comparison of existing text recognition methods.

| Algorithm | Year | Strength | Weakness |
|---|---|---|---|
| Zhou *et al.* [59] | 1997 | • can recognize characters in natural images | • only applicable to simple images<br>• sensitive to font variation |
| deCampos *et al.* [60] | 2009 | • can recognize characters in natural images<br>• insensitive to font variation | • only applicable to single characters, instead of words |
| Mishra *et al.* [31] | 2012 | • can recognize words in complex natural images | • sensitive to font variation |
| Mishra *et al.* [15] | 2012 | • can recognize words in complex natural images | • rely on large lexicon<br>• sensitive to font variation |
| Novikova *et al.* [14] | 2012 | • can recognize words in complex natural images | • rely on large lexicon<br>• sensitive to blur and occlusion |
| Rodriguez-Serrano *et al.* [64] | 2013 | • can recognize words in complex natural images<br>• retrieval based word recognition<br>• efficient | • require a lexicon for each image |
| Shi *et al.* [32] | 2013 | • can recognize words in complex natural images<br>• insensitive to blur, occlusion, font variation | • rely on manually designed character models<br>• require detailed annotations for parts |
| Yao *et al.* [7] | 2014 | • can recognize words in complex natural images<br>• can learn character parts from training data<br>• insensitive to blur, occlusion, font variation | • inefficient |

into a character recognition module, which is trained using a large amount of synthetic characters. Based on [10], Neumann *et al.* [65] introduced new feature extraction methods and combination strategies, which significantly improves the accuracy and efficiency of this system. Later, Neumann *et al.* [13] further extend the methods in [10, 65] to attain real-time text detection and recognition[2] (Fig. 11).

Recently, Neumann *et al.* [66] presented a new system for scene text localization and recognition, which combines the advantages of sliding-window based and component based methods. In this system, character parts (strokes) are modelled by oriented bar filters. These oriented bar filters are utilized to perform both character detection and recognition.

Based on [12], Yao *et al.* [34] constructed an end-to-end system that accomplishes scene text detection and recognition concurrently. This is the first work that can localize and read texts of arbitrary orientations in natural images.

The great success of deep learning methods in various computer vision tasks [67–71] has enlightened researchers in the area of scene text detection and recognition. Coates [72] and Wang *et al.* [73] used CNN with unsupervised pre-training for text detection and character recognition. Bissacco *et al.* [17] built a system, called PhotoOCR, which is able to read characters in uncontrolled conditions. The core

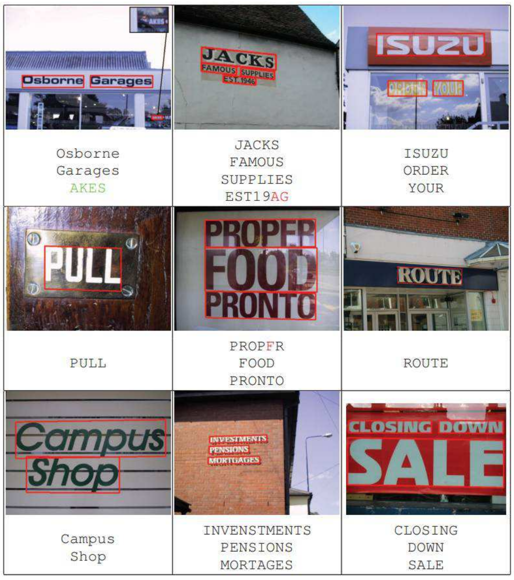

**Fig. 11** End-to-end recognition examples of [13]. This is the first real end-to-end system for scene text recognition.

[2] Demo available at http://textspotter.org/

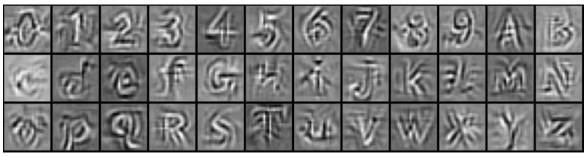

**Fig. 12** Filters learned from characters in natural images [19]. As can be seen, the learned filters can capture the shapes of the corresponding characters.

of PhotoOCR is a DNN model running on HOG features, instead of image pixels. Jaderberg *et al.* [19] proposed a new CNN architecture, which allows feature sharing (Fig. 12) for character detection, character classification and bi-gram classification. As we will show in Sec. 3, deep learning based systems, once trained with tremendous data, generally outperform conventional methods by a considerable margin. There are mainly two drawbacks of these methods: (1) They all only handle horizontal or near-horizontal texts; (2) The computational burdens of these algorithms are extremely high. These two drawbacks may constrain the promotion and application of such algorithms.

The advantages and limitations of the existing systems for end-to-end text recognition are summarized in Tab. 3.

### 2.4 Related Applications and Products of Scene Text Detection and Recognition

In recent years, text detection and recognition in natural scenes have become active research topics. Consequently, a lot of relevant theories, models, algorithms and systems have been proposed developed. Meanwhile, researchers in related fields such as robot and multimedia, adopt these technologies in applications in robot navigation, image search and object recognition, achieving satisfactory results.

Researchers from the GRASP laboratory in the University of Pennsylvania have successfully endowed a robot called 'Graspy'[3] with the ability of locating and reading characters in natural scenes. While moving in the room, this robot can perceive the surrounding environment, recognize the characters, doorplates and signs on the wall, and infer its position according to such information.

Tsai *et al.* [1] has developed a document search system for smart phones. This system allows the user to take a picture of the interested document, and then it will automatically read the title of the document and return the document stored on the server. Karaoglu *et al.* [74] incorporated textual information in natural images into the traditional object recognition framework, which further improves the recognition accuracy.

In addition, some commercial products also have functionalities related to scene text detection and recognition. For example, the Google Goggles app [75] is able to read characters on books, CDs and products. The Amazon Firefly app can recognize web addresses and phone numbers in natural scenes.

## 3 Benchmark Datasets and Evaluation Protocols

Public datasets and relevant assessment standards establish solid reference substances for algorithm development and comparison. The recent progresses in scene text detection and recognition have largely been driven by the datasets and evaluation methods in these fields.

In this section, we will describe in detail the widely used datasets and protocols for performance evaluation in scene text detection and recognition. Moreover, we will identify the current state-of-the-art algorithms on each benchmark dataset where applicable.

### 3.1 Benchmark Datasets

**ICDAR 2003 and 2005.** The ICDAR 2003 Rubust Reading Competition [76] held in 2003 is the first that formally releases a benchmark [4] for scene text detection and recognition. This dataset contains 509 fully annotated text images. 258 images from the dataset are used for training and 251 for testing. It was also used in the ICDAR 2005 Text Locating Competition [43].

**ICDAR 2011 and 2013.** The ICDAR 2011 [77] and 2013 [78] Robust Reading Competitions were held successively to track the recent advances in scene text detection and recognition. The datasets in 2011[5] and 2013[6] are inherited from the benchmark used in the previous ICDAR competitions but have undergone extension and modification, since there are some problems with the previous dataset (for instance, imprecise bounding boxes and inconsistent definitions of "word").

**OSTD.** The Oriented Scene Text Database (OSTD)[7] was proposed by Yi *et al.* [28]. This dataset contains in total 89 images of logos, indoor scenes and street views and can be used for evaluating detection algorithms for multi-oriented texts in natural scenes.

[3] http://phys.org/news/2011-05-university-pennsylvania-pr2-robot-video.html

[4] http://algoval.essex.ac.uk/icdar/Datasets.html

[5] http://robustreading.opendfki.de/wiki/SceneText

[6] http://dag.cvc.uab.es/icdar2013competition/

[7] http://media-lab.engr.ccny.cuny.edu/ cyi/project_scenetextdetection.html

**Table 3** Comparison of existing end-to-end text recognition methods.

| Algorithm | Year | Strength | Weakness |
|---|---|---|---|
| Neumann *et al.* [10] | 2010 | • the first real end-to-end system for scene text recognition | • only applicable to horizontal texts |
| Wang *et al.* [11] | 2011 | • good performance on street view images<br>• robust | • only applicable to horizontal texts<br>• require a lexicon for each image |
| Neumann *et al.* [13] | 2012 | • high recall on complex natural images<br>• fast | • only applicable to horizontal texts |
| Coates *et al.* [72] | 2011 | • good performance on complex natural images<br>• robust | • only applicable to horizontal texts<br>• require a large volume of training data |
| Bissacco *et al.* [17] | 2013 | • excellent performance on complex natural images<br>• robust<br>• fast | • only applicable to horizontal texts<br>• require a large volume of training data |
| Yao *et al.* [34] | 2014 | • good performance on complex natural images<br>• can handle texts of different orientations | • inefficient |
| Jaderberg *et al.* [19] | 2014 | • excellent performance on complex natural images<br>• robust | • only applicable to horizontal texts<br>• require a large volume of training data<br>• inefficient |

**MSRA-TD500.** The MSRA Text Detection 500 Database (MSRA-TD500)[8] is a benchmark for evaluating detection algorithms for multi-oriented texts in natural scenes, which was first introduced in the work of Yao *et al.* [12]. This dataset includes 500 images with horizontal as well as slant and skewed texts in complex natural scenes.

**SVT.** The Street View Text (SVT) dataset[9] [11, 46] is a collection of outdoor images with scene texts of high variability. SVT contains 350 full images and also has word-level annotations (no character bounding boxes). This dataset can be used for both cropped word recognition as well as full image word detection and recognition.

**NEOCR.** The NEOCR dataset[10] [79] includes images with multi-oriented texts in natural scenes. It contains 659 real world images with 5238 annotated bounding boxes. This is a multilingual dataset, as the texts in this database are in different languages, for example, English, Hungarian, Russian, Turkish and Czech.

**KAIST.** The KAIST scene text dataset[11] [80] consists of 3000 images that are taken in indoor and outdoor scenes, under different lighting conditions. This database is also a multilingual benchmark, which includes texts in Korean and English. Besides, it provides binary masks for each character in the images. Thus, this dataset can be used in both text localization and segmentation tasks.

**Chars74K.** The Chars74K dataset[12] was released by de Campos *et al.* [81] to evaluate recognition algorithms for individual characters in natural images. This dataset contains symbols of both English and Kannada language. The *Good-Img* subset includes 636 images, in which a portion of Latin letters and Arabic numbers are labelled by the annotators.

**SVHN.** The Street View House Numbers (SVHN) Dataset[13] [82] is a large real-world database with more than 600,000 digits in natural scenes. The digits are cropped house numbers harvested from Google Street View images. This benchmark is primarily used for developing and evaluating digit recognition algorithms.

**IIIT 5K-Word.** The IIIT 5K-Word dataset[14] [15] is the largest and most challenging benchmark in this field to date. This database includes 5000 images with text in both natural scenes and born-digital images. It is challenging because of the variation in font, color, size, layout and the presence of noise, blur, distortion and varying illumination. 2000 images are used for training and 3000 images for testing.

[8] http://www.iapr-tc11.org/mediawiki/index.php/MSRA_Text_Detection_500_Database_(MSRA-TD500)

[9] http://vision.ucsd.edu/ kai/svt/

[10] http://www.iapr-tc11.org/mediawiki/index.php/NEOCR:_Natural_Environment_OCR_Dataset

[11] http://www.iapr-tc11.org/mediawiki/index.php/KAIST_Scene_Text_Database

[12] http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/

[13] http://ufldl.stanford.edu/housenumbers/

[14] http://cvit.iiit.ac.in/projects/SceneTextUnderstanding/IIIT5K.html

## 3.2 Evaluation Protocols

### 3.2.1 Evaluation Protocols for Text Detection Algorithms

In scene text detection, there are three important metrics in performance assessment: precision, recall and F-measure. Precision measures the ratio between the true positives and all detections, while recall measures the ratio the true positives and all true texts that should be detected. F-measure, as an overall, single indicator of algorithm performance, is the harmonic mean of precision and recall.

**Evaluation Protocol of ICDAR 2003 and 2005.** In the ICDAR robust reading competitions in 2003 [76] and 2005 [43], the organizers designed more flexible definitions of precision and recall than those used in the information retrieval community.

The match $m$ between two rectangles is the ratio of the area of intersection of the rectangles and that of the minimum bounding rectangle containing both. The set of rectangles estimated by each algorithm are called *estimates* and the set of ground truth rectangles provided in the ICDAR dataset are called *targets*. For each rectangle, the match with the largest value is found. Hence, the best match for a rectangle $r$ in a set of rectangles $R$ is defined as:

$$m(r; R) = \max\{m(r, r') | r' \in R\}. \tag{1}$$

Then, the definitions of precision and recall are:

$$precision = \frac{\sum_{r_e \in E} m(r_e; T)}{|E|}, \tag{2}$$

$$recall = \frac{\sum_{r_t \in T} m(r_t; E)}{|T|}, \tag{3}$$

where $E$ and $T$ are the sets of ground truth rectangles and estimated rectangles, respectively. F-measure $f$ is a combination of the two above measures, *precision* and *recall*. The relative weights of precision and recall are controlled by a parameter $\alpha$, which is usually set to 0.5 to give equal weights to precision and recall:

$$f = \frac{1}{\dfrac{\alpha}{precision} + \dfrac{1-\alpha}{recall}}. \tag{4}$$

The performances of different text detection methods evaluated on the ICDAR 2003 dataset are shown in Tab. 4. Note that the ICDAR 2003 and 2005 competitions used the same dataset for performance evaluation and algorithm comparison.

**Table 4** Performances of different methods evaluated on ICDAR 2003.

| Algorithm | Precision | Recall | F-measure |
|---|---|---|---|
| Huang *et al.* [29] | **0.81** | **0.74** | **0.72** |
| TD-Mixture [12] | 0.69 | 0.66 | 0.67 |
| TD-ICDAR [12] | 0.68 | 0.66 | 0.66 |
| Epshtein *et al.* [9] | 0.73 | 0.60 | 0.66 |
| Yi *et al.* [83] | 0.71 | 0.62 | 0.63 |
| Neumann *et al.* [65] | 0.65 | 0.64 | 0.63 |
| Yi *et al.* [28] | 0.71 | 0.62 | 0.62 |
| Chen *et al.* [8] | 0.60 | 0.60 | 0.58 |

On this dataset, the algorithm of Huang *et al.* [29] outperforms other methods by a large margin. This algorithm augments the SWT feature with color information and introduces covariance descriptors.

**Evaluation Protocol of ICDAR 2011 and 2013.** One problem with the evaluation method used in the ICDAR robust reading competitions in 2003 and 2005 is that it is unable to handle the cases of one-to-many and many-to-many matches, which occur very often in practice. Thus this protocol always underestimates the performance of text detection algorithms. To address this problem, the organizers of ICDAR 2011 [77] and 2013 [78] adopted the evaluation method proposed by Wolf *et al.* [84].

The protocol of Wolf *et al.* [84] considers three matching cases: one-to-one, one-to-many and many-to-many. Precision and recall are defined as follows:

$$precision(G, D, t_r, t_p) = \frac{\sum_j Match_D(D_j, G, t_r, t_p)}{|D|}, \tag{5}$$

$$recall(G, D, t_r, t_p) = \frac{\sum_i Match_G(G_i, D, t_r, t_p)}{|G|}. \tag{6}$$

$G$ and $D$ represent ground truth rectangle set and detection rectangle set, respectively. $t_r \in [0, 1]$ is the constraint on area recall and $t_p \in [0, 1]$ is the constraint on area precision. The typical values of $t_r$ and $t_p$ are 0.8 and 0.4.

$Match_D$ and $Match_G$ are functions which take different types of matches into consideration. Specifically, $Match_D$ and $Match_G$ are defined as:

$$Match_D(D_j, G, t_r, t_p) = \begin{cases} 1 & \text{if one-to-one match} \\ 0 & \text{if no match} \\ f_{sc}(k) & \text{if many } (\rightarrow k) \text{ matches} \end{cases} \tag{7}$$

**Table 5** Performances of different methods evaluated on ICDAR 2011.

| Algorithm | Precision | Recall | F-measure |
|---|---|---|---|
| Huang *et al.* [30] | **0.88** | **0.71** | **0.78** |
| Yin *et al.* [49, 85] | 0.863 | 0.683 | 0.762 |
| Neumann *et al.* [86] | 0.854 | 0.675 | 0.754 |
| Koo *et al.* [87] | 0.814 | 0.687 | 0.745 |
| Unified [34] | 0.822 | 0.657 | 0.730 |
| Huang *et al.* [29] | 0.82 | 0.75 | 0.73 |
| Neumann *et al.* [66] | 0.793 | 0.664 | 0.723 |
| Shi *et al.* [88] | 0.833 | 0.631 | 0.718 |
| Kim *et al.* [77] | 0.830 | 0.625 | 0.713 |
| Neumann *et al.* [13] | 0.731 | 0.647 | 0.687 |
| Yi *et al.* [89] | 0.672 | 0.581 | 0.623 |
| Yang *et al.* [77] | 0.670 | 0.577 | 0.620 |
| Neumann *et al.* [77] | 0.689 | 0.525 | 0.596 |
| Shao *et al.* [77] | 0.635 | 0.535 | 0.581 |

**Table 6** Performances of different methods evaluated on MSRA-TD500.

| Algorithm | Precision | Recall | F-measure |
|---|---|---|---|
| Kang *et al.* [26] | **0.71** | 0.62 | **0.66** |
| Yin *et al.* [49] | **0.71** | 0.61 | **0.66** |
| Unified [34] | 0.64 | 0.62 | 0.61 |
| TD-Mixture [12] | 0.63 | **0.63** | 0.60 |
| TD-ICDAR [12] | 0.53 | 0.52 | 0.50 |
| Epshtein *et al.* [9] | 0.25 | 0.25 | 0.25 |
| Chen *et al.* [8] | 0.05 | 0.05 | 0.05 |

$$Match_G(G_i, D, t_r, t_p) = \begin{cases} 1 & \text{if one-to-one match} \\ 0 & \text{if no match} \\ f_{sc}(k) & \text{if many } (\rightarrow k) \text{ matches} \end{cases} \tag{8}$$

where $f_{sc}(k)$ is a parameter function controlling the amount of punishment, which is inflicted in case of scattering, i.e. splits or merges. In practice, $f_{sc}(k)$ is set to a constant 0.8.

The performances of different text detection methods evaluated on the ICDAR 2011 dataset are shown in Tab. 5. Since the ICDAR 2013 dataset is brand new and the recent algorithms have not been fully evaluated on it, we only report the quantitative results on the ICDAR 2011 dataset.

As can be seen from Tab. 5, the method proposed by Huang *et al.* [30] obtains state-of-the-art performance on this benchmark. This method is a mixture of MSER and deep learning technique.

**Evaluation Protocol of MSRA-TD500.** In [12], Yao *et al.* proposed a novel protocol that is more suitable for assessing detection algorithms that are designed for texts of arbitrary orientations. Minimum area rectangles [90] are used in the protocol of [12] because they are much tighter and more accurate than axis-aligned rectangles. However, a problem imposed by using minimum area rectangles is that it is difficult to judge whether a text line is correctly detected. It is not trivial to directly compute the overlap ratio between the estimated rectangle $D$ and the ground truth rectangle $G$. Instead, Yao *et al.* [6, 12] proposed to calculate the overlap ratio using axis-aligned rectangles $G'$ and $D'$, which are obtained by rotating $G$ and $D$ round their centers $C_G$ and $C_D$, respectively. The overlap ratio between $G$ and $D$ is defined as:

$$m(G, D) = \frac{A(G' \cap D')}{A(G' \cup D')}, \tag{9}$$

where $A(G' \cap D')$ and $A(G' \cup D')$ denote the areas of the intersection and union of $G'$ and $D'$. Obviously, the overlap ratio computed in this way is not accurate. Besides, the ground truth rectangles annotated are not accurate either, especially when the texts are skewed. Because of the imprecision of both ground truth and computed overlap ratio, the definitions of precision and recall used in the ICDAR protocol do not apply. Alternatively, Yao *et al.* [6, 12] used the original definitions of precision and recall.

Similar to the evaluation method for the PASCAL object detection task [91], in the protocol of [6, 12] detections are considered true or false positives based on the overlap ratio between the estimated minimum area rectangles and the ground truth rectangles. If the included angle of the estimated rectangle and the ground truth rectangle is less than $\pi/8$ and their overlap ratio exceeds 0.5, the estimated rectangle is considered a correct detection. Multiple detections of the same text line are taken as false positives. The definitions of precision and recall are:

$$precision = \frac{|TP|}{|E|}, \tag{10}$$

$$recall = \frac{|TP|}{|T|}, \tag{11}$$

where $TP$ is the set of true positive detections while $E$ and $T$ are the sets of estimated rectangles and ground truth rectangles.

Moreover, to accommodate difficult texts (too small, occluded, blurry, or truncated) that are hard for text detection algorithms, the protocol of [12] introduces an elastic mechanism which can tolerate detection misses of difficult texts. The basic criterion of this elastic mechanism is: *if the difficult texts are detected by an algorithm, it counts; otherwise, the algorithm will not be punished.* Accordingly, the annotations of the images in the proposed dataset should be changed. Each text line considered to be difficult is given an additional “difficult” label. Thus the ground truth rectangles can be categorized into two sub sets: ordinary sub set $T_o$ and difficult sub set $T_d$; ditto, the true positives $TP$ can also be categorized into ordinary sub set $TP_o$, which is the set of rectangles matched with $T_o$, and difficult sub set $TP_d$, which is the set of rectangles matched with $T_d$. After incorporating the elastic mechanism, the definitions of precision, recall and F-measure become:

$$precision = \frac{|TP_o| + |TP_d|}{|E|} = \frac{|TP|}{|E|}, \tag{12}$$

$$recall = \frac{|TP_o| + |TP_d|}{|T_o| + |TP_d|} = \frac{|TP|}{|T_o| + |TP_d|}, \tag{13}$$

$$f = \frac{2 \cdot precision \cdot recall}{precision + recall}. \tag{14}$$

The performances of different text detection methods evaluated on the MSRA-TD500 dataset are shown in Tab. 6. As can be observed, the methods of Kang *et al.* [26] and Yin *et al.* [49] achieve state-of-the-art performance on this database. A key thing to notice here is that both of these two methods adopted adaptive clustering strategy for text line grouping.

### 3.2.2 Evaluation Protocols for Text Recognition Algorithms

In scene text recognition, the performance of an algorithm is measured either by character level recognition rate or word level recognition rate.

**Evaluation Protocol of ICDAR.** The ICDAR competitions [43, 76–78] also include text recognition tasks. There are two metrics for performance assessment: word level recognition rate and normalized edit distance. The former is quite strict, as it requires that each character is correctly recognized. The latter is relatively looser, since it can tolerate partial local errors for each word.

Tab. 7 shows the performances of different recognition algorithms evaluated on the ICDAR 2003 dataset. The performances of the algorithms are measured by word level recognition rate, which is more often used for quantitative comparison. Since the datasets used in ICDAR 2011 and 2013 are almost the same as the ICDAR 2003 dataset and majority of the recent works reported results on this dataset, we only demonstrate quantitative results on it, for the sake of simplicity and clarity.

As can be seen, the Deep Features method proposed by Jaderberg *et al.* [19] achieves significantly better performance than other methods. This method is based on deep learning and requires large volume of training data.

**Evaluation Protocol of SVT.** The metric for performance evaluation of the SVT dataset [11, 46] is also word level recognition rate, similar to that of the ICDAR datasets. The performances of different recognition algorithms evaluated on the SVT dataset are demonstrated in Tab. 8. The top performers on this benchmark are PhotoOCR [17] and Deep Features [19], both of which are deep learning based algorithms and harvest extra data for training.

**Evaluation Protocol of Chars74K.** The primary purpose of the Chars74K dataset [81] is to assess the classification accuracy of different algorithms on cropped characters, so de Campos *et al.* used average recognition rate at character level as the measure for performance evaluation.

**Evaluation Protocol of IIIT 5K-Word.** Similar to the evaluation method for the ICDAR competitions, the protocol used for the IIIT 5K-Word dataset [15] adopted word level recognition rate as a measure of algorithm performance. This dataset comes with three types of lexicons (small, medium, and large) for each test image. So there are three settings for performance assessment, corresponding to the three types of lexicons.

The performances of different text recognition algorithms evaluated on the IIIT 5K-Word dataset are depicted in Tab. 9. On this benchmark, the Strokelets method proposed by Yao *et al.* [7] generally outperforms other competing algorithms. The Strokelets method is the first work that introduces automatically learned multi-scale mid-level representation into the domain of scene text recognition. This open doors for a set of new techniques to be used in scene text recognition and enables new possibilities for further improvement in performance.

## 4 Discussions

We have presented a comprehensive literature review on scene text detection and recognition. We introduced inspiring ideas and influential works, and described public datasets and evaluation protocols in these fields.

As can be seen from the previous sections, scene text detection and recognition are active research topics with rapid development and consistent growth. A great many of excellent works have been proposed and significantly pushed the performance envelope of text detection and recognition. For example, the best text detection performance of on the ICDAR 2011 has increased drastically from 0.581 to 0.78 in F-measure, in a very short term (2011 to 2014); the highest recognition accuracy on the large, challenging dataset IIIT

**Table 7** Performances of different algorithms evaluated on ICDAR 2003.

| Dataset | ICDAR 2003(FULL) | ICDAR 2003(50) |
|---|---|---|
| Deep Features [19] | **91.5** | **96.2** |
| Strokelets [7] | 80.33 | 88.48 |
| Discriminative Feature Pooling [33] | 76 | 88 |
| CNN [73] | 84 | 90 |
| Whole [92] | - | 89.69 |
| TSM+CRF [32] | 79.30 | 87.44 |
| TSM+PLEX [32] | 70.47 | 80.70 |
| Multi-Class Hough Forests [93] | - | 85.70 |
| Large-Lexicon Attribute-Consistent [14] | 82.8 | - |
| Higher Order [15](with edit distance) | - | 80.28 |
| Higher Order [15](without edit distance) | - | 72.01 |
| Pairwise CRF [31](with edit distance) | - | 81.78 |
| Pairwise CRF [31](without edit distance) | - | 69.90 |
| SYNTH+PLEX [11] | 62 | 76 |
| ICDAR+PLEX [11] | 57 | 72 |
| ABBYY9.0 [94] | 55 | 56 |

**Table 8** Performances of different algorithms evaluated on SVT.

| Dataset | SVT |
|---|---|
| PhotoOCR [17] | **90.39** |
| Deep Features [19] | 86.1 |
| Discriminative Feature Pooling [33] | 80 |
| Strokelets [7] | 75.89 |
| CNN [73] | 70 |
| Whole [92] | 77.28 |
| TSM+CRF [32] | 73.51 |
| TSM+PLEX [32] | 69.51 |
| Large-Lexicon Attribute-Consistent [14] | 72.9 |
| Higher Order [15](with edit distance) | 73.57 |
| Higher Order [15](without edit distance) | 68.00 |
| Pairwise CRF [31](with edit distance) | 73.26 |
| Pairwise CRF [31](without edit distance) | 62.28 |
| SYNTH+PLEX [11] | 57 |
| ICDAR+PLEX [11] | 56 |
| ABBYY9.0 [94] | 35 |

**Table 9** Performances of different algorithms evaluated on IIIT 5K-Word.

| Lexicon | Small | Medium | Large |
|---|---|---|---|
| Strokelets [7] | **80.2** | **69.3** | 38.3 |
| Label Embedding [64] | 76.1 | 57.4 | - |
| Higher Order [15](with edit distance) | 68.25 | 55.50 | 28 |
| Higher Order [15](without edit distance) | 64.10 | 53.16 | **44.30** |
| Pairwise CRF [31](with edit distance) | 66 | 57.5 | 24.25 |
| Pairwise CRF [31](without edit distance) | 55.50 | 51.25 | 20.25 |
| ABBYY9.0 [94] | 24.33 | - | - |

5K-Word with small lexicons has been boosted from 55.5 to 80.2, in an even shorter term (2012 to 2014).

Nevertheless, numerous challenges are still to be addressed. There is still no single algorithm that can handle all the difficulties encountered in real-world scenarios. Various factors, such as noise, blur, occlusion, non-uniform illumination, serious distortion and complex clutter, would still make the systems struggle or even fail.

Moreover, the practicability and applicability of the current algorithms are very limited. All the algorithms except for [12, 28, 34, 49] can only deal with horizontal or near-horizontal texts in natural scenes. This is a severe shortcoming, since a considerable portion of the texts in real-world scenes are non-horizontal. Such limitation would make it impossible to capture the textual information embodied in non-horizontal texts and thus seriously restricts the practicability and applicability of these methods.

There are more than 100 types of frequently-used languages all over the world, but majority of the existing methods and benchmarks (except for [12,27,79–81]) have focused on texts in English. In this age of globalization, it is urgent and indispensable to build systems that are able to handle multilingual texts and serve the people in the whole world.

An emerging trend in scene text detection and recognition is the usage of deep learning and big data [17, 19, 73, 82, 95–98]. These methods indeed brought new ideas into these fields and improved the performance. However, they simply adopted techniques form other domains and used tremendous training examples, but gave few insights about the problem itself. The performance boost might be largely due to the large amount of training data, which is not publicly available. Therefore, it is interesting to investigate and uncover the reasons behind the success of such deep learning based systems.

The essential properties of scene text lie in the structure of individual characters and the relationships between characters. The key for building effective systems for scene text detection and recognition is to capture and utilize the structure of individual characters and the relationships between characters. We believe this direction is promising and worth of further exploration.

## 5 Conclusions

Text is born as an explicit carrier of high level semantics. This unique property makes text different from other generic visual cues, such as contour, color and texture. The rich and precise information embodied in text can assist a wide range of real-world applications. Therefore, detecting and recognizing texts in natural scenes have become important and vibrant research areas in computer vision. This literature review is aimed at tracing the recent advances in scene text detection and recognition, and providing other researchers with a full reference to useful resources in these fields.

Through the efforts of multitudinous researchers, considerable progresses have been made in scene text detection and recognition in recent years. However, there are plenty of problems that should be addressed in the future. To build practical systems that can accurately and robustly extract textual information from natural scenes, there is still a long way to go. We believe the following aspects are worthy to explore in the next decade:

- **Muiti-Orientation:** In real-world scenarios, texts can be in different orientations. The works of Yao *et al.* [12, 34], Shivakumara *et al.* [53] and Yi *et al.* [28] have successfully made the community realize the significance of multi-oriented text detection and recognition. But majority of the researchers in the fields only paid their attention to horizontal texts. In order to take full advantage of textual information in natural scenes, it is necessary to read texts of different orientations.
- **Muiti-Language:** Most of the existing methods are concerned with text in English, while several works involved texts in other languages (for example, Chinese [12], Kannada [60] and Korean [80]). Considering practicability, it is crucial to develop detection and recognition systems that can handle texts of different languages.
- **Deep Learning+Big Data:** The combination of deep learning methods and large amount of training data seems dominate the fields of scene text detection and recognition. Previous deep learning based methods simply adopted mature techniques from other domains and achieved performance boosts over conventional algorithms. Further improvement in detection and recognition accuracy can be achieved, if the deep learning framework is employed to discover and model the characteristics of scene text from large volume of data.

## References

1. S. Tsai, H. Chen, D. Chen, G. Schroth, R. Grzeszczuk, and B. Girod. Mobile visual search on printed documents using text and low bit-rate features. In *Proc. of ICIP*, 2011.
2. D. B. Barber, J. D. Redding, T. W. McLain, R. W. Beard, and C. N. Taylor. Vision-based target geo-location using a fixed-wing miniature air vehicle. *Journal of Intelligent and Robotic Systems*, 47(4):361–382, 2006.
3. B. Kisacanin, V. Pavlovic, and T. S. Huang. *Real-Time Vision for Human-Computer Interaction*. Springer, Heidelberg, 2005.
4. G. N. DeSouza and A. C. Kak. Vision for mobile robot navigation: A survey. *IEEE Trans. PAMI*, 24(2):237–267, 2002.
5. Y. K. Ham, M. S. Kang, H. K. Chung, R. H. Park, and G. T. Park. Recognition of raised characters for automatic classification of rubber tires. *Optical Engineering*, 34(1):102–109, 2005.
6. C. Yao, X. Zhang, X. Bai, W. Liu, Y. Ma, and Z. Tu. Rotation-invariant features for multi-oriented text detection in natural images. *PLoS One*, 8(8), 2013.
7. C. Yao, X. Bai, B. Shi, and W. Liu. Strokelets: A learned multi-scale representation for scene text recognition. In *Proc. of CVPR*, 2014.
8. X. Chen and A. Yuille. Detecting and reading text in natural scenes. In *Proc. of CVPR*, 2004.
9. B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *Proc. of CVPR*, 2010.
10. L. Neumann and J. Matas. A method for text localization and recognition in real-world images. In *Proc. of ACCV*, 2010.
11. K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In *Proc. of ICCV*, 2011.
12. C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu. Detecting texts of arbitrary orientations in natural images. In *Proc. of CVPR*, 2012.
13. L. Neumann and J. Matas. Real-time scene text localization and recognition. In *Proc. of CVPR*, 2012.
14. T. Novikova, , O. Barinova, P. Kohli, and V. Lempitsky. Large-lexicon attribute-consistent text recognition in natural images. In *Proc. of ECCV*, 2012.
15. A. Mishra, K. Alahari, and C. V. Jawahar. Scene text recognition using higher order language priors. In *Proc. of BMVC*, 2012.

16. J. J. Weinman, Z. Butler, D. Knoll, and J. Feild. Toward integrated scene text reading. *IEEE Trans. on PAMI*, 36(2):375–387, 2013.
17. A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. PhotoOCR: Reading text in uncontrolled conditions. In *Proc. of ICCV*, 2013.
18. T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan. Recognizing text with perspective distortion in natural scenes. In *Proc. of ICCV*, 2013.
19. M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. In *Proc. of ECCV*, 2014.
20. J. Almazan, A. Gordo, A. Fornes, and E. Valveny. Word spotting and recognition with embedded attributes. *IEEE Trans. PAMI*, 36(12):2552–2566, 2014.
21. D. Chen, J. Luettin, and K. Shearer. A survey of text detection and recognition in images and videos. Technical report, IDIAP, August 2000.
22. K. Jung, K. Kim, and A.K. Jain. Text information extraction in images and video: a surveyâĂİ. *PR*, 37(5):977–997, 2004.
23. J. Liang, D. Doermann, and H. Li. Camera-based analysis of text and documents: a survey. *IJDAR*, 7(2):84–104, 2005.
24. H. Zhang, K. Zhao, Y. Z. Song, and J. Guo. Text extraction from natural scene image: A survey. *Neurocomputing*, 122(1):310–323, 2013.
25. S. Uchida. Text localization and recognition in images and video. *Handbook of Document Image Processing and Recognition*, pages 843–883, 2014.
26. L. Kang, Y. Li, and D. Doermann. Orientation robust text line detection in natural images. In *Proc. of CVPR*, 2014.
27. Y. Pan, X. Hou, and C. Liu. A hybrid approach to detect and localize texts in natural scene images. *IEEE Trans. Image Processing*, 20(3):800–813, 2011.
28. C. Yi and Y. Tian. Text string detection from natural scenes by structure-based partition and grouping. *IEEE Trans. Image Processing*, 20(9):2594–2605, 2011.
29. W. Huang, Z. Lin, J. Yang, and J. Wang. Text localization in natural images using stroke feature transform and text covariance descriptors. In *Proc. of ICCV*, 2013.
30. W. Huang, Y. Qiao, and X. Tang. Robust scene text detection with convolution neural network induced mser trees. In *Proc. of ECCV*, 2014.
31. A. Mishra, K. Alahari, and C. V. Jawahar. Top-down and bottom-up cues for scene text recognition. In *Proc. of CVPR*, 2012.
32. C. Shi, C. Wang, B. Xiao, Y. Zhang, S. Gao, and Z. Zhang. Scene text recognition using part-based tree-structured character detection. In *Proc. of CVPR*, 2013.
33. C. Y. Lee, A. Bhardwaj, W. Di, V. Jagadeesh, and R. Piramuthu. Region-based discriminative feature pooling for scene text recognition. In *Proc. of CVPR*, 2014.
34. C. Yao, X. Bai, and W. Liu. A unified framework for multi-oriented text detection and recognition. *IEEE Trans. Image Processing*, 23(11):4737–4749, 2014.
35. Y. Zhong, K. Karu, and A. K. Jain. Locating text in complex color images. *Pattern Recognition*, 28(10):1523âĂŞ–1535, 1995.
36. K. I. Kim, K. Jung, and J. H. Kim. Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm. *IEEE Trans. PAMI*, 25(12):1631–1639, 2003.
37. J. Gllavata, R. Ewerth, and B. Freisleben. Text detection in images based on unsupervised classification of high-frequency wavelet coefficients. In *Proc. of ICPR*, 2004.
38. H. P. Li, D. Doermann, and O. Kia. Automatic text detection and tracking in digital video. *IEEE Trans. Image Processing*, 9(1):147–156, 2000.
39. B. Leibe and B. Schiele. Scale-invariant object categorization using a scale-adaptive mean-shift search. *Pattern Recognition*, pages 145–153, 2004.
40. M. R. Lyu, J. Song, and M. Cai. A comprehensive method for multilingual video text detection, localization, and extraction. *IEEE Trans. CSVT*, 15(2):243–255, 2005.
41. Y. Zhong, H. Zhang, and A. K. Jain. Automatic caption localization in compressed video. *IEEE Trans. PAMI*, 22(4):385–392, 2000.
42. P. Viola and M. Jones. Fast and robust classification using asymmetric adaboost and a detector cascade. In *Proc. of NIPS*, 2001.
43. S. M. Lucas. Icdar 2005 text locating competition results. In *Proc. of ICDAR*, 2005.
44. V. Wu, R. Manmatha, and E. M. Riseman. Finding text in images. In *Proc. of 2nd ACM Int. Conf. Digital Libraries*, 1997.
45. C. Wolf and J. M. Jolion. Extraction and recognition of artificial text in multimedia documents. *Formal Pattern Analysis and Applications*, 6(4):309–326, 2004.
46. K. Wang and S. Belongie. Word spotting in the wild. In *Proc. of ECCV*, 2010.
47. A. Jain and B. Yu. Automatic text location in images and video frames. *Pattern Recognition*, 31(12):2055âĂŞ–2076, 1998.
48. H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, and B. Girod. Robust text detection in natural images with edge-enhanced maximally stable extremal regions. In *Proc. of ICIP*, 2011.
49. X. C. Yin, X. Yin, K. Huang, and H. Hao. Robust text detection in natural scene images. *IEEE Trans. on PAMI*, 36(5):970–983, 2014.
50. J. Wright, A. Y. Yang, and A. Ganesh A et al. Robust face recognition via sparse representation. *IEEE Trans. PAMI*, 31(2):210–227, 2009.
51. M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Processing*, 15(12):3736–3745, 2006.
52. M. Zhao, S. Li, and J. Kwok. Text detection in images using sparse representation with discriminative dictionaries. *Image and Vision Computing*, 28(12):1590–1599, 2010.
53. P. Shivakumara and T. Q. Phan abd C. L. Tan. A laplacian approach to multi-oriented text detection in video. *IEEE Trans. PAMI*, 33(2):412–419, 2011.
54. Y. Liu, S. Gotoand, and T. Ikenaga. A contour-based robust algorithm for text detection in color images. *IEICE Transactions on Information and Systems*, 89(3):1221–1230, 2006.
55. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. of CVPR*, 2005.
56. J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

In *Proc. of ICML*, 2005.

57. M. Sawaki, H. Murase, and N. Hagita. Automatic acquisition of context-based images templates for degraded character recognition in scene images. In *Proc. of ICPR*, 2000.
58. J. Zhou and D. Lopresti. Extracting text from www images. In *Proc. of ICDAR*, 1997.
59. J. Zhou, D. P. Lopresti, and Z. Lei. Ocr for world wide web images. In *Proc. of SPIE*, 1997.
60. T. E. de Campos, B. R. Babu, and M. Varma. Character recognition in natural images. In *Proc. of VISAPP*, 2009.
61. R. Smith. Limits on the application of frequency-based language models to ocr. In *Proc. of ICDAR*, 2011.
62. J. Matas, O. Chum, and M. Urban M et al. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 20(10):761–767, 2004.
63. M Mohri, F Pereira, and M Riley. Weighted finite-state transducers in speech recognition. *Computer Speech and Language*, 16(1):69âĂŞ–88, 2013.
64. J. A. Rodriguez-Serrano and F. Perronnin. Label embedding for text recognition. In *Proc. of BMVC*, 2013.
65. L. Neumann and J. Matas. Text localization in real-world images using efficiently pruned exhaustive search. In *Proc. of ICDAR*, 2011.
66. L. Neumann and J. Matas. Scene text localization and recognition with oriented stroke detection. In *Proc. of ICCV*, 2013.
67. Y. LeCun, B. Boser, J. S. Denker, R. E. Howard, W. Habbard, L. D. Jackel, and D. Henderson. Handwritten digit recognition with a back-propagation network. In *Proc. of NIPS*, 1990.
68. C. Farabet, C. Couprie, , L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE Trans. on PAMI*, 35(8):1915–1929, 2013.
69. Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proc. of CVPR*, 2014.
70. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. of CVPR*, 2014.
71. C. Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-supervised nets. In *Proc. of AISTATS*, 2015.
72. A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. J. Wu, and A. Y. Ng. Text detection and character recognition in scene images with unsupervised feature learning. In *Proc. of ICDAR*, 2011.
73. T. Wang, D. J. Wu, A. Coates, and A. Y. Ng. End-to-end text recognition with convolutional neural networks. In *Proc. of ICPR*, 2012.
74. S. Karaoglu, J. C. van Gemert, and T. Gevers. Object reading: Text recognition for object recognition. In *Proc. of ECCVW*, 2012.
75. Google Goggles. https://play.google.com/store/apps.
76. S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. ICDAR 2003 robust reading competitions. In *Proc. of ICDAR*, 2003.
77. A. Shahab, F. Shafait, and A. Dengel. ICDAR 2011 robust reading competition challenge 2: Reading text in scene images. In *Proc. of ICDAR*, 2011.
78. D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez i Bigorda, S. Robles Mestre, J. Mas, D. Fernandez Mota, J. Almazan Almazan, and L. P. de las Heras. ICDAR 2013 robust reading competition. In *Proc. of ICDAR*, 2013.
79. R. Nagy, A. Dicker, and K. Meyer-Wegener. NEOCR: A configurable dataset for natural image text recognition. In *CBDAR Workshop at ICDAR*, 2011.
80. S. H. Lee, K. Jungz M. S. Cho, and J. H. Kim. Scene text extraction with edge constraint and text collinearity link. In *Proc. of ICPR*, 2010.
81. T. E. de Campos, B. R. Babu, and M. Varma. Character recognition in natural images. In *Proc. of VISAPP*, 2009.
82. Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *Proc. of NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.
83. C. Yi and Y. Tian. Text extraction from scene images by character appearance and structure modeling. *CVIU*, 117(2):182âĂŞ–194, 2013.
84. C. Wolf and J. M. Jolion. Object count/area graphs for the evaluation of object detection and segmentation algorithms. *IJDAR*, 8(4):280–296, 2006.
85. X. C. Yin, X. Yin, K. Huang, and H. Hao. Accurate and robust text detection: a step-in for text retrieval in natural scene images. In *Proc. of SIGIR*, 2013.
86. L. Neumann and J. Matas. On combining multiple segmentations in scene text recognition. In *Proc. of ICDAR*, 2013.
87. H. Koo and D. H. Kim. Scene text detection via connected component clustering and nontext filtering. *IEEE Trans. on Image Processing*, 22(6):2296–2305, 2013.
88. C. Shi, C. Wang, B. Xiao, Y. Zhang, and S. Gao. Scene text detection using graph model built upon maximally stable extremal regions. *Pattern Recognition Letters*, 34(2):107–116, 2013.
89. C. Yi and Y. Tian. Text detection in natural scene images by stroke gabor words. In *Proc. of ICDAR*, 2011.
90. H. Freeman and R. Shapira. Determining the minimum-area encasing rectangle for an arbitrary closed curve. *Comm. ACM*, 18(7):409–413, 1975.
91. M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.
92. V. Goel, A. Mishra, K. Alahari, and C. V. Jawahar. Whole is greater than sum of parts: Recognizing scene text words. In *Proc. of ICDAR*, 2013.
93. G. Yildirim, R. Achanta, and S. Susstrunk. Text recognition in natural images using multiclass hough forests. In *Proc. of VISAPP*, 2013.
94. ABBYY FineReader 9.0. http://www.abbyy.com/.
95. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. In *NIPS Workshop on Deep Learning*, 2014.
96. B. Su and S. Lu. Accurate scene text recognition based on recurrent neural network. In *Asian Conference on Computer Vision (ACCV)*, 2014.
97. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading

text in the wild with convolutional neural networks. In *arXiv*, 2014.

98. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Deep structured output learning for unconstrained text recognition. In *arXiv*, 2014.

**Yingying Zhu** received the B.S. degree in electronics and information engineering from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2011. She is currently a Ph.D. student with the School of Electronic Information and Communications, HUST. Her research areas mainly include text/traffic sign detection and recognition in natural images.

**Cong Yao** received the B.S. and Ph.D. degrees in electronics and information engineering from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2008 and 2014, respectively. He was a Visiting Research Scholar with Temple University, Philadelphia, PA, USA, in 2013. His research has focused on computer vision and machine learning, in particular, the area of text detection and recognition in natural images.

**Xiang Bai** received the B.S., M.S., and Ph.D. degrees from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2003, 2005, and 2009, respectively, all in electronics and information engineering. He is currently a Professor with the School of Electronic Information and Communications, HUST, where he is also the Vice Director of the National Center of Anti-Counterfeiting Technology. His research interests include object recognition, shape analysis, scene text recognition, and intelligent systems.