Text-Attentional Convolutional Neural Network for Scene Text Detection

Tong He, Weilin Huang, *Member, IEEE*, Yu Qiao, *Senior Member, IEEE*, and Jian Yao, *Senior Member, IEEE*

***Abstract*— Recent deep learning models have demonstrated strong capabilities for classifying text and non-text components in natural images. They extract a high-level feature globally computed from a whole image component (patch), where the cluttered background information may dominate true text features in the deep representation. This leads to less discriminative power and poorer robustness. In this paper, we present a new system for scene text detection by proposing a novel text-attentional convolutional neural network (Text-CNN) that particularly focuses on extracting text-related regions and features from the image components. We develop a new learning mechanism to train the Text-CNN with multi-level and rich supervised information, including text region mask, character label, and binary text/non-text information. The rich supervision information enables the Text-CNN with a strong capability for discriminating ambiguous texts, and also increases its robustness against complicated background components. The training process is formulated as a multi-task learning problem, where low-level supervised information greatly facilitates the main task of text/non-text classification. In addition, a powerful low-level detector called contrast-enhancement maximally stable extremal regions (MSERs) is developed, which extends the widely used MSERs by enhancing intensity contrast between text patterns and background. This allows it to detect highly challenging text patterns, resulting in a higher recall. Our approach achieved promising results on the ICDAR 2013 data set, with an F-measure of 0.82, substantially improving the state-of-the-art results.**

***Index Terms*— Maximally stable extremal regions, text detector, convolutional neural networks, multi-level supervised information, multi-task learning.**

Manuscript received October 1, 2015; revised February 14, 2016; accepted March 10, 2016. Date of publication March 28, 2016; date of current version April 14, 2016. This work was supported in part by the National Natural Science Foundation of China under Project 61503367 and Project 41571436, in part by the Guangdong Natural Science Foundation under Grant 2015A030310289, in part by the Shenzhen Research Program under Grant JSGG20150925164740726, Grant JCYJ20150925163005055, and Grant CXZZ20150930104115529, in part by the National High-Tech Research and Development Program of China under Grant 2015AA042303, and in part by the Guangdong Research Program under Grant 2014B050505017 and Grant 2015B010129013. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiaochun Cao. *(Corresponding author: Weilin Huang.)*

T. He is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, China (e-mail: tong.he@siat.ac.cn).

W. Huang and Y. Qiao are with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Multimedia Laboratory, The Chinese University of Hong Kong, Hong Kong (e-mail: wl.huang@siat.ac.cn; yu.qiao@siat.ac.cn).

J. Yao is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, China (e-mail: jian.yao@whu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2547588

I. Introduction

TEXT detection and recognition in natural images have received increasing attention in computer vision and image understanding, due to its numerous potential applications in image retrieval, scene understanding, visual assistance, etc. Though tremendous efforts have recently been devoted to improving its performance, reading texts in unconstrained environments is still extremely challenging and remains an open problem, as substantiated by recent literature [1], [3]–[5], where the leading performance on the detection sub task is of 80% F-measure on the ICDAR 2011 [4], and current result of unconstrained end-to-end recognition is only 57% accuracy on the challenging SVT dataset [3]. In this work, we focus on the detection task that aims to correctly localize exact positions of text-lines or words in an image. Text information in natural images may have significant diversity of text patterns in highly complicated background. For example, text can be in a very small size, low quality, or low contrast, and even regular ones can be distorted considerably by numerous real-world affects, such as perspective transform, strong lighting, or blurring. These pose fundamental challenges of this task where detecting correct character components is difficult.

The Extremal Regions (ERs) or Maximally Stable Extremal Regions (MSERs) based methods achieved the state-of-the-art performance on scene text detection [1], [5]. The ERs detector exhibits great advantage in detecting challenging text patterns, and often results in a good recall. However, low-level nature of the ERs or MSERs makes them easily to be distorted by numerous real-world affects and complicated background. This results in a huge number of non-text components, which may be many orders of magnitude larger than the number of true text components, e.g. over 10^6 ERs and 10^3 MSERs per image, compared to dozens of characters in an image [6]. Therefore, correctly filtering out these non-text components is critical to the success of such methods.

Many previous methods focus on developing hand-crafted features based on a number of heuristic image properties (e.g. intensity variance, sharp information or spatial location) to discriminate text and non-text components [6], [7]. These low-level features inherently limit their generality on highly challenging text components. They also reduce the robustness against text-like components, which often have similar low-level properties as the true texts, such as bricks, windows or leaves. These facts pose main challenge of current scene text detection systems, and severely harm their performance in both precision and recall.

Recently, a number of deep models have been developed for text component filtering/classification [1], [2], [8], or word/character recognition [2], [8]–[10], by leveraging advances of deep image representation. Huang *et al.* [1] applied a two-layer CNN with a MSERs detector for text detection. Jaderberg *et al.* [8] and Wang *et al.* [2] employed CNN models for both detection and recognition tasks. These text classifiers, building on conventional CNN, are generally trained with binary text/non-text label, which is relatively less informative to learn a meaningful text feature. Furthermore, they compute a high-level deep feature globally from an image patch for discriminating text and non-text. Such feature commonly includes a large amount of background information that possibly dominates in the representation, so as to reduce its discriminative power and robustness. As shown in Fig. 1 (a), a number of background components are classified incorrectly as text by the CNN used in [1] and [2], while some ambiguous text components may be misidentified as the non-text ones.

Our goal is to design a text CNN model that particularly focalizes to text-related regions and specific characteristics of text within an image component. In particular, this is achieved by training a CNN with more informative supervised information, such as text region mask, character label and binary text/non-text information. We show that such meaningful supervision would be greatly helpful to make a reliable binary decision on text or non-text. To realize this capability and incorporate additional supervised knowledge, we propose a novel Text-Attentional Convolutional Neural Network (Text-CNN) for text component filtering. The Text-CNN is incorporated with a newly-developed Contrast-Enhanced MSERs (CE-MSERs) to form our text detection system. Our contributions are summarized as follow.

Firstly, we propose the Text-CNN that particularly focuses on extracting deep text feature from an image component. Applying a deep CNN for text/non-text classification is not new, but to our knowledge, this is the first attempt to design a CNN model specially for text-related region and text feature computing, by leveraging multi-level and rich supervised information. The additional low-level supervised information enables our Text-CNN with better discriminative power to identify ambiguous components, and stronger robustness against background, as shown in Fig. 1.

Secondly, we introduce a deep multi-task learning mechanism to learn the Text-CNN efficiently, where each level of the supervised information is formulated as a learning task. This allows our model to share learned features between multiple tasks, making it possible to learn more discriminative text features by using the additional low-level supervision. This greatly facilitates convergence of the main task for text/non-text classification.

Thirdly, we develop a new CE-MSERs detector, which extends the widely-used MSERs method by enlarging the local contrast between text and background regions. The CE-MSERs detector is capable of detecting highly ambiguous components which are often missed or confused by the original MSERs. We incorporate the CS-MSERs with our Text-CNN filter to form a new text detection system which achieves the state-of-the-art results on the ICDAR 2011 and ICDAR 2013 benchmarks.

(a) The confident scores by CNN [1], [2] (Yellow) and Text-CNN (Red)

(b) Image (c) CNN [1], [2] (d) Text-CNN

Fig. 1. Comparisons of confident maps/scores between the proposed Text-CNN and the CNN filter of [1] and [2]. (a) The confident scores of a number of text and non-text components: the Text-CNN in RED bars and CNN of [1] and [2] in YELLOW bars. (b-d): the Text-CNN in (d) shows stronger robustness against background components, and higher capability for discriminating ambiguous text components than the CNN of [1] and [2] in (c).

The rest of paper is organized as follows. A brief review on related studies is given in Section II. Then the proposed text detection system, including the Text-CNN model and CE-MSERs detector, is described in Section III. Experimental results are compared and discussed in Section IV, followed by conclusions in Section V.

II. Related Work

Existing work for scene text detection can be roughly categorized into two groups, sliding-window and connected component based methods [11]. The sliding-window methods detect text information by moving a multi-scale sub-window through all possible locations in an image [2], [4], [8], [12]–[17]. Then a pre-trained classifier is applied to identify whether text information is contained within the sub-window. For example, Wang, *et al.* [15] explored Random Ferns classifier [18] with a histogram of oriented gradients (HOG) feature [19] for text detection. Similarly, Pan *et al.* [17] generated text confident map by using a sliding-window with a HoG feature and a WaldBoost [20] which is a boosted cascade classifier. The main difficulties for this group of methods lie in designing a discriminative feature to train a powerful classifier, and managing computational flexibility by reducing the number of the scanning windows.

The connected component methods have achieved great success in text detection and localization [1], [5]–[7], [21]–[30]. They separate text and non-text information at pixel-level by running a fast low-level detector. The retained pixels with similar properties are then grouped together to construct possible text components. The ERs/MSERs [31], [32] and Stroke Width Transform (SWT) [25] are two representative methods in this group. Extending from the original SWT, Huang *et al.* [22] proposed a Stroke Feature Transform (SFT), by incorporating important color cues of text patterns for pixel tracking. In [1], [5], [23], and [26], the MSERs detector has demonstrated strong capability for detecting challenging text patterns, yielding a good recall in component detection. In [24], Characterness was proposed by incorporating three novel cues: stroke width, perceptual divergence, and HoG at edges. Recently, Jaderberg *et al.* [3] applied the EdgeBox proposals [33] for generating text components. Sun, *et al.* proposed a robust text detection system by developing a color-enhanced contrasting extremal region (CER) for character component detection, followed by a neural networks classifier for discriminating text and non-text components [30].

In order to handle multi-oriented text lines, a two-level approach was proposed for sequentially detecting component candidates and text chain candidates [7]. Then two filters building on Random Forest [34] were developed to effectively classify corresponding two-level candidates. Extending from this work, Yao *et al.* [35] proposed a unified framework for multi-oriented text detection and recognition, where same features were used for both tasks. Then a new dictionary search approach was developed to improve recognition performance. In [21]. Yin *et al.* presented a new multi-orientation text detection system by proposing a novel text line construction method. They designed a number of sequential coarse-to-fine steps to group character candidates based on a pre-trained adaptive clustering.

The connected component based methods exhibit great advantage in speed by fast tracking text pixels in one pass computation, with complexity of $O(N)$. However, low-level nature of these methods largely limits their capability, making them poorer robust and discriminative. Therefore, a sophisticated post-processing method is often required to deal with large amount of generated components, which causes main challenge of this group of methods.

A powerful text/non-text classifier or component filter is critical to success of both the sliding-window and connected component based methods. Huge efforts have been devoted to developing an efficient hand-crafted feature that could correctly capture discriminative characteristics of text. Chen and Yuille [13] proposed a sliding-window method by using the Adaboost classifiers trained on a number of low-level statistical features. Similarly, Zhang *et al.* [4] proposed a symmetry-based text line detector that computes both symmetry and appearance features based on heuristic image properties. For the connected component approaches, in [25], a number of heuristic rules were designed to filter out the non-text components generated by the SWT detector. Extending from this framework, a learning based approach building on Random Forest [36] was developed by using manually-designed histogram features computed from various low-level image properties [7]. To eliminate the heuristic procedures, Huang *et al.* [22] proposed two powerful text/non-text classifiers, named Text Covariance Descriptors (TCDs), which compute both heuristic properties and statistical characteristics of text stokes by using covariance descriptor [37]. However, low-level nature of these manually-designed features largely limit their performance, making them hard to discriminate challenging components accurately, such as the ambiguous text patterns and complicated background outliers.

Deep CNN models are powerful for image representation by computing meaningful high-level deep features [38]–[41]. Traditional CNN network (such as the well-known LeNet) has been successfully applied to text/document community for digit and hand-written character recognition [42], [43]. Recently, the advances of deep CNN have also been adopted to design the challenging scene text detection/recognition systems [1]–[3], [8]. A CNN model was employed to filter out non-text components, which are generated by a MSERs detector in [1] or the Edgebox in [3], while Wang *et al.* [2] and Jaderberg *et al.* [8] applied a deep CNN model in the sliding-window fashion for text detection. Gupta *et al.* developed a new Fully-Convolutional Regression Network (FCRN) that jointly performs text detection and bounding-box regression [44]. Though these deep models have greatly advanced previous manually-designed features, they mostly compute general image features globally from a whole image component/patch mixing with cluttered background, where background information may be computed dominantly in feature learning processing. They only use relatively simple text/non-text information for training, which is significantly insufficient to learning a discriminative representation.

Our work is closely related to recently proposed approach by Huang *et al.* [1], who incorporated a two-layer CNN with a MSERs detector for text detection. Our method improves upon this approach by developing a novel Text-CNN classifier and an improved CE-MSERs detector, which together lead to a significant performance boost. Our Text-CNN model is specially designed to compute discriminative text features from general image components, by leveraging more informative supervised information that facilitates text feature computing. This enables it with strong robustness against cluttered background information, and remarkably sets us apart from previous CNN based methods for this task.

III. Proposed Approach

The proposed system mainly includes two parts: a Text-Attentional Convolutional Neural Network (Text-CNN) for text component filtering/classfication, and a Contrast-Enhanced MSERs (CE-MSERs) detector for generating component candidates.

A. Text-Attentional Convolutional Neural Network

Given an ambiguous image component, people can easily discriminate it in text or non-text, with much more informative knowledge about it, such as pixel-level text region segmentation and character information

(e.g., 'a', 'b', 'c', etc.). Such low-level prior information is crucial for people to make a reliable decision. The text region segmentation allows people to accurately extract true text information from cluttered background, while the character label (assuming that people understand this language) helps them make a more confident decision on ambiguous cases.

Current text component filters are mostly learned with just binary text/non-text labels, which are insufficient to train a powerful classifier, making it neither robust nor discriminative. We wish to train a more powerful deep network for text task. We aim to 'teach' the model with more intuitive knowledge about the text or character. These important knowledge include lower-level properties of the text, such as explicit locations of text pixels and labels of characters. Toward this, we propose the Text-CNN by training it with multi-level highly-supervised text information, including text region segmentation, character label and binary text/non-text information. These additional supervised information would 'tell' our model with more specific features of the text, from low-level region segmentation to high-level binary classification. This allows our model to sequentially understand *where*, *what* and *whether* is the character, which is of great importance for making a reliable decision.

However, training a unified deep model that incorporates multi-level supervised information is non-trivial, because different level information have various learning difficulties and convergence rates. To tackle this problem, we formulate our training process as a multi-task learning (MTL) [45] problem that treats the training of each task as an independent process by using one of the supervised information. Deep CNN model is well suited for the MTL by sharing features [46]. We formulate our model as follows.

1) Problem Formulation: Given total N training examples, denoted as $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, the goal of traditional CNN is trying to minimize

$$\arg\min_{\mathbf{W}} \sum_{i=1}^{N} \mathcal{L}(y_i, f(\mathbf{x}_i; \mathbf{W})) + \Psi(\mathbf{W}) \tag{1}$$

where $f(\mathbf{x}_i; \mathbf{W})$ is a function parameterized by $\mathbf{W}$. $\mathcal{L}(\cdot)$ denotes a loss function, which typically chooses the hinge loss for classification task, and the least square loss for regression task. The $\Psi(\mathbf{W})$ is regularization term. The training procedure just tries to find a mapping function that connects the input image patch and (single-task) output labels (e.g., 0/1 for binary classification) without any extra information. Many existing CNN-based models follow this formulation by simply using the binary text/non-text information [1], [2], [8].

We observe that Eq. (1) can be readily extended to the traditional MTL problem by maximizing overall loss of all tasks. However, the traditional MTL considers equal contribution or importance for all tasks. In our model, three tasks are significantly different. The lower-level text region learning and character label classification are much more complicated than the main task of text/non-text classification, leading to different learning difficulties and convergence rates. To this end, our goal is to optimize the main task with assistance of two auxiliary tasks. Therefore, directly applying the traditional MTL to our problem is non-trivial.

It has been shown that deep neural networks can benefit from learning with related tasks simultaneously [46]. Recently, Zhang *et al.* [47] developed an efficient task-constrained deep model for facial landmark detection, where additional facial attributes information are utilized to facilitate learning of the main detection task. Motivated from this work, we formulate our problem as a MTL problem with one main task ($\mathcal{L}^m$) and two auxiliary tasks ($\mathcal{L}^a$):

$$\begin{aligned}\arg\min_{\mathbf{W}^m, \mathbf{W}^a} & \sum_{i=1}^{N} \mathcal{L}^m(y_i^m, f(\mathbf{x}_i; \mathbf{W}^m)) \\ & + \sum_{i=1}^{N} \sum_{a \in A} \lambda^a \mathcal{L}^a(y^a, f(\mathbf{x}_i; \mathbf{W}^a)),\end{aligned} \tag{2}$$

where λ^a and $\mathbf{W}^a$ denote importance factor and model parameters of the auxiliary tasks, respectively. Regularization terms are omitted for simplification. In our model, the main text/non-text task and the auxiliary character label task are the classification problem, while learning of text region is a regression problem. Our three tasks can be further detailed as follows:

$$\mathcal{L}^B(y_i^b, f(\mathbf{x}_i; \mathbf{W}^b) = y_i^b \log(p(y_i^b | \mathbf{x}_i; \mathbf{W}^b)), \tag{3}$$

$$\mathcal{L}^L(y_i^l, f(\mathbf{x}_i; \mathbf{W}^l) = y_i^l \log(p(y_i^l | \mathbf{x}_i; \mathbf{W}^l)), \tag{4}$$

$$\mathcal{L}^R(\mathbf{y}_i^r, f(\mathbf{x}_i; \mathbf{W}^r) = \|\mathbf{y}_i^r - f(\mathbf{x}_i; \mathbf{W}^r)\|^2. \tag{5}$$

Three tasks use the same input image patch, $\mathbf{x}_i \in \mathbb{R}^{32\times32\times3}$. The main differences between them are in output labels. $y_i^b = \{0, 1\} \in \mathbb{R}^2$ (i.e. text/non-text), and $y_i^l = \{0...9, A...Z, a...z\} \in \mathbb{R}^{62}$ are the labels of main task and character label task, respectively. $\mathbf{y}_i^r = \{0, 1\} \in \mathbb{R}^{32\times32}$ is a binary mask with the same size of the input image (in 2D), indicating explicit pixel-level segmentation of text. For the text region task, the model estimates probability of the text at each pixel: $f(\mathbf{x}_i; \mathbf{W}^r) = [0, 1] \in \mathbb{R}^{32\times32}$. It minimizes the L2 distance between the ground truth mask and the estimated probability map. Notice that both auxiliary tasks are only explored in the training process. In the test process, only the main task works for text/non-text classification.

The details of Text-CNN are presented in Fig. 2. It includes three convolutional layers (with kernel size of 9×9, 7×7 and 5×5, respectively), followed by two fully-connected layers of 1024D. The second convolutional layer is followed by a max pooling layer with the pooling kernel of 3×3. The last fully-connected layer is followed the outputs of main text/non-text task (2D) and the character label task (62D). The text region regression is trained by using an additional sub network branched from the second convolutional layer of the main network, before the non-invertible max pooling operation. It includes two deconvolutional layers [48] duplicated from the 1st and 2nd convolutional layers of the main network, in an invert order. The output of the (2nd) deconvolutional layer is a 32×32 probability map, corresponding to pixel locations of the input image. λ_1 and λ_2 are two manually-setting parameters used to balance the two auxiliary tasks and main task.

The architecture of our model was designed carefully. First, the pooling layer was designed to balance computational complexity and model accuracy. In general, the pooling layer

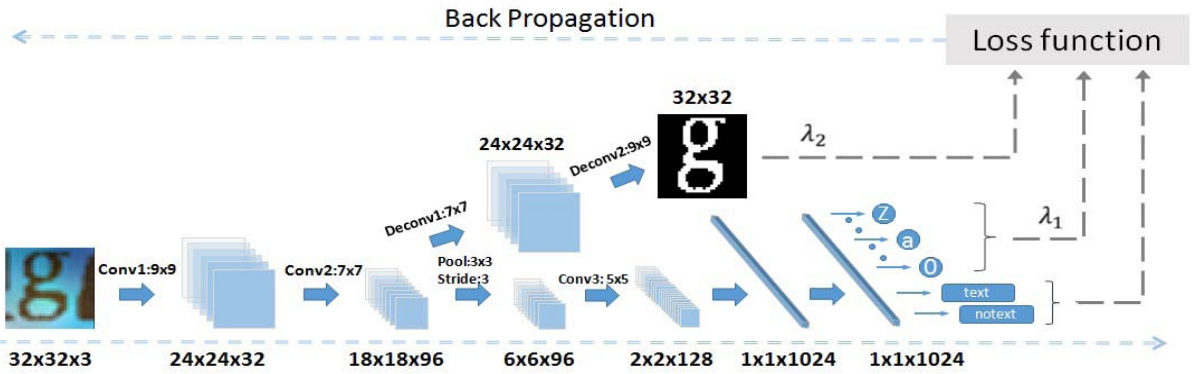


Fig. 2. Structure of the Text-Attentional Convolutional Neural Network.

is developed to reduce model parameters and complexity, but at the cost of spatial information loss. Since we use small image patch (32 × 32) as our model input (compared to the 227 × 227 input image in regular image classification tasks), our architecture does not need more pooling layers to reduce its spatial resolution. Second, since the pooling operation is non-invertible, so that it is difficult to apply it before the deconvolution layers starting at the 2nd convolution layer. Therefore, we do not use the pooling layer in the 1st convolution layer. In our experiments, removing the pooling operation in the 2nd convolution layer did not further improve the performance. But adding an additional pooling layer in the 3rd convolution reduced the accuracy, since the output map was already in a small size, 2×2.

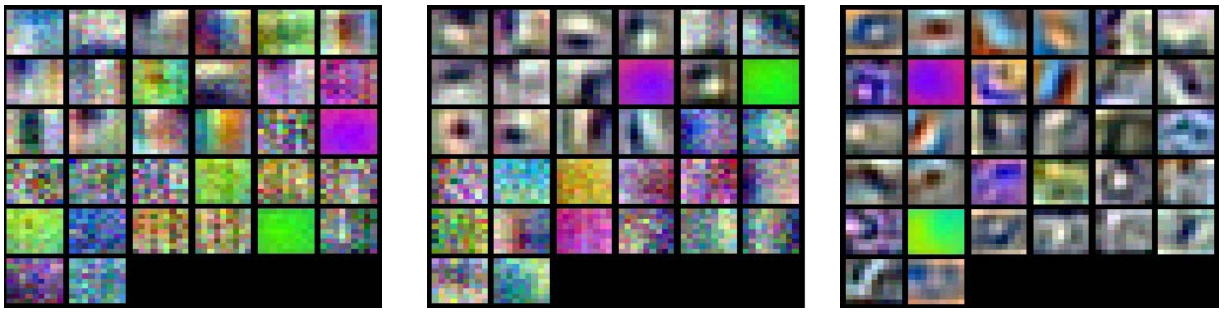
2) Training the Text-CNN: Our Text-CNN is trained by using stochastic gradient algorithm which has been widely used for many conventional deep learning models. However, directly applying it to each of our tasks independently can not achieve a joint optimization. The main reason is that our three tasks have significantly different loss functions with highly unbalanced output layers (e.g. 2D, 62D and 1024D). Our Text-CNN is also different from previous MTL model for facial landmark detection, where different tasks are heterogeneous [47]. An important property of our model is that the three tasks have strong hierarchical structure, and can be optimised sequentially from the low-level region regression to the high-level binary classification. Such hierarchical structure inherently follows basic procedure of our people to identify a text/non-text component. People should first be able to segment a text region arcuately from cluttered background. Then people could make a more confident decision if they recognize the label of a character. Therefore, a reliable high-level decision is strongly built on robust learning of the multi-level prior knowledge.

The training process starts from joint learning of the two auxiliary tasks for text region regression and character recognition. The text region task aims to regress a binary mask of the input image through the deconvolution network. It enables the model with meaningful low-level text information which is important to identify the text and non-text regions at pixel level. Joint optimization of both auxiliary tasks endows the network to learn shared features of multiple characters having a same label, while learning discriminative features between characters of different labels.

Then the region task is stopped when the main task starts. The main reason for early stoping the regression task is that the low-level knowledge has been embedded into the model successfully after a number of training iterations (e.g. 30,000 iterations in our training process). By jointly learning, the low-level pixel information can be abstracted correctly by the higher-level character label. Still keeping the region task would cause the model overfitting to the low-level task, due to significant deference of their convergence rates. The character task continues with training of the main task until the model is finally optimized.

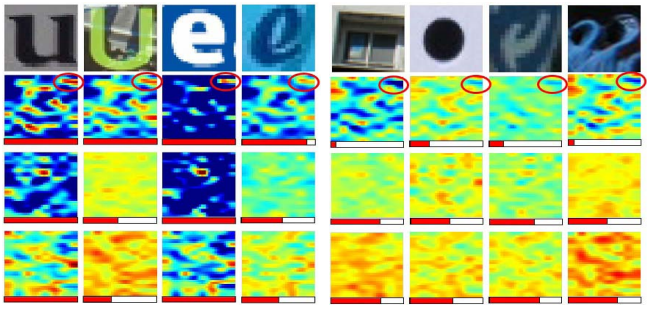
The label task brings meaningful character-level information (including the low-level character region and its abstracted label) to the main task. Such information may enable the main task with an additional strong feature that discriminates the text and non-text reliably, e.g. a model can identify a component as text confidently, if it knows what exactly this character is. This is crucial to identify some highly confused components, and thus is the key to the robustness and discriminant of our model. Essentially, the label task keeps smooth of the whole training process, by transforming the low-level knowledge to the high-level binary decision. From the viewpoint of parameter optimization, the auxiliary tasks actually work as an important pre-training for our Text-CNN, which has been proven to be importance for seeking a better model optimization [48].

To demonstrate efficiency of our design, we present the low-level filters (from the 1st convolutional layer) learned by three different CNN models in Fig. 3. The positive impact of each auxiliary task for learning the Text-CNN are shown clearly. Experiment details will be described in Section IV. As shown, the filters learned by the conventional CNN may capture main low-level structures of the character images. However, the filter maps are distorted heavily by an amount of noise, indicating that the binary supervision can not provide informative low-level information for learning robust filters. Obviously, the noise is reduced considerably by using the additional character label task, which provides more detailed character information that facilitates learning of the network. As expected, our Text-CNN with both auxiliary tasks learns a number of



(a) Conventional CNN (b) Text-CNN with only label task (c) Text-CNN

Fig. 3. Learned filters (1st-conv. layer) of (a) the conventional CNN, (b) the Text-CNN with only label task, and (c) with both auxiliary tasks.



(a) (b)

Fig. 4. Performance of the Text-CNN (2nd row), Text-CNN just with character label (3rd row), and the conventional CNN (4th row) on a number of ambiguous text and background components (1st row). The feature map (32×32) is reshaped from the last fully-connected layer of each model, with its confident score under each map (in red bar, the higher score denotes higher probability to be the text). One of the key locations for discriminating text and non-text has been indicated by a red circle in each map. (a) Text components. (b) Non-text components.

high-quality flitters, which not only preserve rich low-level structure of the text, but also are strongly robust against the background noise.

Robustness and discriminative capability of the Text-CNN are further demonstrated in Fig. 4, where we show (reshaped) feature maps of the last fully-connected layers on a number of challenging samples. As demonstrated by the confident scores (red bars under the feature maps) and the key discriminative features (located by red circles), the Text-CNN consistently yields high confident scores on these highly ambiguous text components, and generates very low scores for those extremely complicated background components, including many text-like strokes. By contrast, the conventional CNN only works well on clear text components. It has confused scores on the ambiguous ones and complicated background components. The advances of Text-CNN are further demonstrated in the confident maps shown in Fig. 5.

B. Contrast-Enhanced MSERs

As discussed, the MSERs algorithm has strong ability to detect many challenging text components, by considering each of them as a 'stable' extremal region. It has achieved remarkable performance on current text detection systems [1], [5]. However, low-level nature of the MSERs detector largely limits its performance. First, the text components generated by the MSERs are easily distorted by various complicated background affects, leading to numerous incorrect detections, such as connecting the true texts to the background components, or separating a single character into multiple components. This makes it difficult to identify the true texts in the subsequent filtering step. Second, some text components are ambiguous with low-contrast or low-quality characters. They may not be defined as the 'stable' or extremal regions, and thus are discarded arbitrarily by the MSERs. Importantly, it is difficult or impossible to recover these components in the subsequent processes, leading to a significant reduction on recall.

To improve the performance of current MSERs based methods, a crucial issue is to enable it to detect as much true text components as possible. Toward this, we develop an efficient approach that enhances local stability of the text regions, which turns out to mitigate low-level distortions effectively. Thus we wish to enhance region-level contrast of natural images. Motivated from recent methods for salient detections [49], we propose a two-step contrast enhancement mechanism. In the first step, we apply cluster based approach developed in [50] for computing the cluster cues globally from the image, e.g. the contrast and spatial cues. Then the contrast region is defined by fusing both cues as [49]. This approach can strongly enhance the contrast of most dominant or large regions, but it may discard some small-size components as well, as shown in Fig. 6 (a).

To circumvent this problem, we develop the second step to improve contrast of the small-size regions. We compute dominated colors from the remained non-contrast regions. Then the contrast regions are computed in the remained regions by using the color space smoothing method as [50]. The resulted contrast region maps computed by both steps are shown in Fig. 6. Finally, the original MSERs algorithm is implemented on both contrast region maps and together with the original image, to generate final CE-MSERs components. Details of the CE-MSERs are described in Alg. 1.

The detection results by the MSERs and CE-MSERs are compared in Fig. 6. Obviously, the CE-MSERs collect more true text components, some of which are extremely challenging for original MSERs detector. To further demonstrate efficiency of the CE-MSERs, we compute character-level

(a) (b) (c) (d)

Fig. 5. Confident maps of the conventional CNN, the CNN of [1] and [2] (CNN-W), and Text-CNN. (a) Image. (b) CNN. (c) CNN-W. (d) Text-CNN.

(a) (b) (c) (d)

Fig. 6. Contrast-region maps of the CE-MSERs, and comparison of CE-MSERs and MSERs. We discard the bounding boxes which are not matched with the ground truth for a better display. (a) Contrast Regions (step one). (b) Contrast Regions (step two). (c) CE-MSERs. (d) MSERs.

Algorithm 1 CE-MSER Proposals Generation

input: a natural image
Transform the color space from RGB to Lab.
Step I
Cluster all pixels into K classes using k-means.
Compute contrast cue of cluster as [49].
Compute spatial cue of cluster as [49].
Multiply both cues to get the contrast value
Generate contrast region map I.
Step II
Quantize the remaining color from 255^3 to 12^3.
Compute color histogram to select the dominant ones.
Enhance contrast of similar colors by color space smoothing [50].
Generate contrast region maps II.
Implement MSERs on the contrast region map I, II and original image.
return CE-MSER components.

recall (by $\geq$ 0.5 overlapping) on the ICDAR 2011 dataset. Our CE-MSERs improve the recall of original MSERs from 88.3% to 91.7%, which is considerable for current low-level text detectors. This turns out to boost the performance on final detection, with 6% improvement on recall, as shown in Table III. More discussions will be presented in Section IV.

The pipeline of full text detection system is as follows. The CE-MSERs detector is first applied to a natural image

Fig. 7. Image samples and their text region masks from the *CharSynthetic*.

to generate text components. Then the proposed Text-CNN is adopted to filter out the non-text components. Finally, text-lines are constructed straightforward by following previous work in [1] and [7]. First, two neighboring components are grouped into a pair if they have similar geometric properties, such as horizontal locations, heights and aspect ratios. Second, we merge the pairs sequentially if they include a common component, and at the same time, have similar orientations. A text-line is constructed until no pair can be merged further. Optionally, a text-line is broken into multiple words based on the horizontal distances of characters or words.

IV. Experiment and Results

We evaluate efficiency of the proposed Text-CNN and CE-MSERs individually. Then our system is tested on four benchmarks for scene text detection: the ICDAR 2005 [51], ICDAR 2011 [52], ICDAR 2013 [53] and MSRA-TD500 [7] databases.

A. Experimental Setup and Datasets

The Text-CNN is trained on character-level by using two datasets, *CharSynthetic* and *CharTrain*. In the *CharSynthetic*, we synthetically generated 80,141 character images, each of which has both character mask and its label.

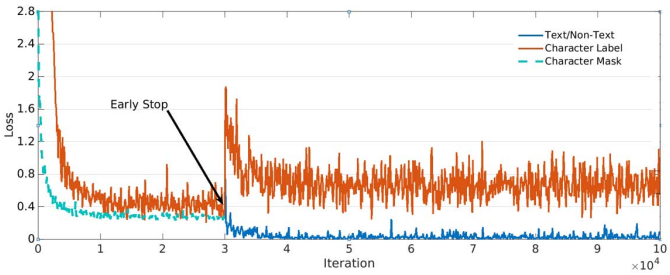


Fig. 8. Training loss of the Text-CNN. The character label task runs through the whole training process. The training data is changed from the $CharSynthetic$ to the $charTrain$ after the early stop at 30K iterations.

TABLE I

WE EVALUATE PARAMETERS OF THE TEXT-CNN: λ_1 AND λ_2. IN THE STAGE ONE (FIRST 30K ITERATIONS), WE FIX $\lambda_1 = 1$ AND CHANGE THE VALUE OF λ_2. IN THE STAGE TWO, WE VERY THE VALUE OF λ_1

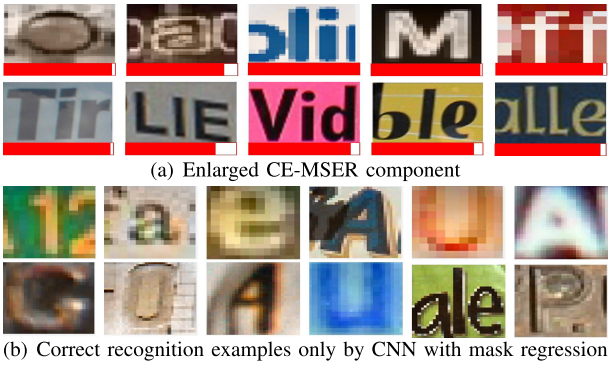
λ_2 (Stage 1)	0.9	0.7	0.5	0.3	0.1
$P_{charLabel}$	1.6%	1.8%	6.4%	84.5%	84.3%
D_{mask}	147.5	141.7	131.8	19.8	21.1
λ_1 (Stage 2)	0.9	0.7	0.5	0.3	0.1
$P_{charLabel}$	87.5%	87.3%	87.4%	87.5%	86.1%
$P_{Text/Non-Text}$	92.6%	92.3%	92.6%	92.6%	92.3%

Some examples are shown in Fig. 7. This dataset is applied for jointly training the low-level text region and character label tasks. The *CharTrain* includes 17,733 text (character) and 85,581 non-text samples, which are cropped from training set of the ICDAR 2011 dataset [52]. Each sample has binary text/non-text information and 62-class character label, which are used to joint train the character label task and the main task. The training of Text-CNN starts from joint learning of mask regression and character recognition tasks, by using the *CharSynthetic*. The loss parameters are manually set to $\lambda_1 = 1$ and $\lambda_2 = 0.3$. The mask regression task is stopped at 30K iterations. Then the main task starts, together with the continued character label task, whose loss parameter is then changed to $\lambda_1 = 0.3$. The tasks are trained with further 70K iterations by using the *CharTrain*. The curves of multiple training losses are presented in Fig. 8, where they converge stably at 30K and 100K iterations.

We investigate the performance of Text-CNN with various values of λ_1 and λ_2 in Table I. The experiments were conducted as follow. In stage one (first 30K iterations), we fixed $\lambda_1 = 1$ and changed the value of λ_2. We computed 62-class character recognition accuracy ($P_{charLabel}$), and L2 distance (D_{Mask}) between the estimated mask and ground true mask on 20,036 synthetically generated character images. Examples of the estimated masks are shown in Fig. 9. In stage two, we computed the 62-class and text/non-text classification accuracies on the $testChar$, by changing the value of λ_1. As can be found, the model can not converge well when the value of λ_2 is larger than 0.5 in the stage one, while the impact of λ_1 is much less significance in the stage two.

For testing the Text-CNN, we collect a character-level dataset: the *CharTest*, which includes 5,751 character images and 11,502 non-text components. The character images are

Fig. 9. Examples of the estimated masks.



(a) Enlarged CE-MSER component

(b) Correct recognition examples only by CNN with mask regression

Fig. 10. (a) Examples of enlarged CE-MSER component by its long side, with corresponding Text-CNN score. (b) Some component examples for character recognition, which are all classified correctly by the CNN with mask regression, but are all failed without the mask regression.

cropped manually from the ICDAR 2011 test set, while the non-text ones are randomly selected from the image background of same dataset.

As mentioned, the input of Text-CNN is a 32×32 patch. We do not directly resize the detected component, instead we enlarge each CE-MSER region to a square area by using its long side, as shown in Fig. 10 (a). Therefore, each input patch may include neighboring characters or parts of them, allowing the Text-CNN to classify it more reliably by exploring surrounding context. This is particularly important to those components which are easily confused with cluttered background, such as 'I', '1', 'l', etc. This operation is implemented on both training and test data. On the other hand, the original CE-MSER or MSER components may include multiple characters, while a single character may also be detected by multiple components due to tree structure of the MSER. The Text-CNN aims to retain all possible components containing true characters. It is able to classify a multi-character component as a positive detection, as shown in Fig. 10 (a). In general, the confident score decreases when a component has more than three characters.

Our text detection system is evaluated on the ICDAR 2005, 2011 and 2013 Robust Reading Competition benchmarks and the MSRA-TD500 database. The ICDAR 2005 [51] has 258 images in the training set, and 251 images for testing with size varying from 307 × 97 to 1280 × 960.

The ICDAR 2011 [52] includes 229 and 255 images for training and testing, respectively. There are 299 training images and 233 test images in the ICDAR 2013 [53]. The MSRA-TD500 database [7] has 500 images containing multi-orientation text lines in different languages (e.g. Chinese, English or mixture of both). The training set contains 300 images, and the rest is used for testing.

B. Results and Comparisons

To investigate the efficiency of each proposed component, we first evaluate the Text-CNN on text component classification. Then the contribution of individual Text-CNN or CE-MSERs to our final detection system is investigated. Finally, the performance of final system are compared extensively with recent results on four benchmarks.

1) Text-CNN on Text Component Classification: The experiments were conducted on the *CharTest*, and results are presented in Table II. To demonstrate the impact of different types of supervisions, we conducted several experiments which were trained and tested on a same dataset for fair comparisons. The baseline is conventional CNN trained by just using text/non-text information. As can be seen, the Text-CNN outperforms the conventional CNN substantially, reducing error rate from 9.8% to 6.7% by using informative low-level supervision. The individual character mask and character label information improve the CNN by 1.4% and 2.0%, respectively.

We further investigate their error rates on separated character and non-character subsets of the *CharTest*. As can be found in Table II, the main errors are raised in the character set by classifying true characters into non-text components. There were two observations from the results. First, the character label information is of great importance for improving the accuracies in the character set, whose error rate is reduced from 28.3% to 17.5%. It may be due to that some ambiguous character structures are easily confused with text-like background objects (e.g. bricks, windows and leaves) which often have similar low-level mask structures as true characters or character strokes. The mid-level character label information is indeed helpful to discriminate them by telling the model what stroke structures of the true characters are. This turns out to improve the final recall. Second, the mask information helps to reduce the error in classifying background components into the characters, resulting in a reduction of false alarms. We also compare our model against the CNN classifier used in [1] and [2] (referred as CNN-W). The performance of Text-CNN is also over 8.6% of the CNN-W trained purely on text/non-text supervision. The strong ability of the Text-CNN is further demonstrated in Fig. 11, where it shows strong robustness against a large amount of building windows. These windows have similar low-level properties as the text patterns, and the character mask and label information are greatly helpful to discriminate them, which turns out to yield a high precision.

In addition, we evaluate the impact of mask regression on character recognition task. The experiments were conducted on the *CharTest*, with 5,751 character images of 62 classes. We compare performance of the conventional CNN and mask pre-trained CNN, which achieve 14.2% and 12.7% error rates respectively. Obviously, the mask regression also improves the performance of CNN on character recognition. Some challenging examples which are only recognized successfully by the CNN with mask regression are presented in Fig. 10(b). This further confirms the efficiency of mask regression for pre-training the CNN model.

TABLE II

INDEPENDENT IMPACT OF CHARACTER MASK OR CHARACTER LABEL FOR TEXT AND NON-TEXT COMPONENT CLASSIFICATION AND RECOGNITION. EXPERIMENTS WERE CONDUCTED ON ICDAR 2011 TEST SET WITH CHARACTER-LEVEL ANNOTATION (THE *CharTest* SET)

Method	CharTest	Text (5751)	Non-Text (11550)
Text/Non-Text Classification (Error Rate)			
CNN-W	8.6%	–	–
CNN	9.8%	28.3%	2.2%
CNN-Mask	8.4%	24.7%	1.8%
CNN-Label	7.8%	17.5%	4.0%
Text-CNN	6.7%	18.6%	1.8%
62-Class Character Recognition			
CNN	–	14.2%	–
CNN-Mask	–	12.7%	–

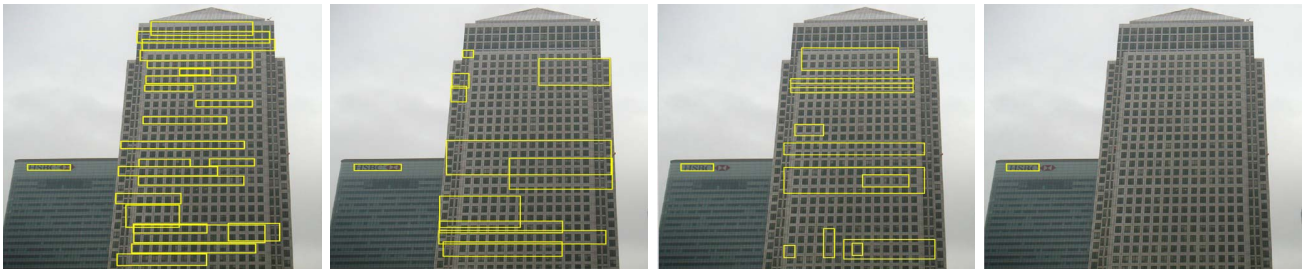
TABLE III

INDEPENDENT CONTRIBUTION OF THE TEXT-CNN AND CE-MSERS (ON THE ICDAR 2011 DATASET)

Method	*Precision*	*Recall*	*Fmeasure*
	with Text-CNN		
MSERs	0.89	0.68	0.78
CE-MSERs	0.91	0.74	0.82
	with CE-MSERs		
CNN	0.85	0.71	0.77
Text-CNNL	0.88	0.72	0.79
Text-CNN	0.91	0.74	0.82

2) Evaluation on the Individual CE-MSERs or Text CNN: We conducted experiments on the ICDAR 2011, and experimental results are presented in Table III. As discussed, our CE-MSERs detector improves the original MSERs with an about 3% character-level recall. This leads to a surprising improvement on the recall of final detection (from 68% to 74%). The large improvement indicates that the CE-MSERs is able to detect some challenging text patterns, which are of importance for detecting or retaining the whole text-lines or words missed by the original MESRs based methods. For example, a word can be easily broken into two separate words in text-line construction, if a character is missed in the middle location. The broken words are considered as false detections, reducing both precision and recall. This can be verified by the improved precision by the CE-MSERs. Both improvements lead to a 4% improvement on the F-measure.

The advantages of Text-CNN are demonstrated on the remarkable improvement on precision. The Text-CNN improves conventional CNN substantially from 85% to 91%, where the single character task obtains a 3% improvement. Similarly, strong capability of the Text-CNN also results in an increase of recall, since a word is not correctly detected if it includes the background components which are not filtered out robustly.



(a) (b) (c) (d)

Fig. 11. Detection results of the conventional CNN, CNN of [1] and [2] (CNN-W) and Text CNN, by using the same CE-MSERs detector. (a) CNN. (b) CNN-W. (c) Text-CNNL. (d) Text-CNN.

TABLE IV

EXPERIMENTAL RESULTS ON THE ICDAR 2005 DATASET

Method	Year	*Precision*	*Recall*	*Fmeasure*
Our method	–	**0.87**	**0.73**	**0.79**
MSERs-CNN [1]	2014	0.84	0.67	0.75
SFT-TCD [22]	2013	0.81	0.74	0.72
Yao *et al.* [7]	2012	0.69	0.66	0.67
Chen *et al.* [29]	2012	0.73	0.60	0.66
Epshtein *et al.* [25]	2010	0.73	0.60	0.66
Yi and Tian [56]	2013	0.71	0.62	0.63
Neumann & Matas [26]	2011	0.65	0.64	0.63
Zhang & Kasturi [57]	2010	0.73	0.62	–
Yi & Tian [28]	2011	0.71	0.62	0.62

TABLE V

EXPERIMENTAL RESULTS ON THE ICDAR 2011 DATASET

Method	Year	*Precision*	*Recall*	*Fmeasure*
Our method	–	**0.91**	0.74	**0.82**
Zhang *et al.*[4]	2015	0.84	**0.76**	0.80
MSERs-CNN [1]	2014	0.88	0.71	0.78
Yin *et al.* [5]	2014	0.86	0.68	0.76
Neumann & Matas [58]	2013	0.85	0.68	0.75
SFT-TCD [22]	2013	0.82	0.75	0.73
Neumann & Matas [27]	2013	0.79	0.66	0.72
Shi *et al.* [59]	2013	0.83	0.63	0.72
Neumann & Matas [6]	2012	0.73	0.65	0.69
González *et al.* [60]	2012	0.73	0.56	0.63
Yi & Tian [28]	2011	0.67	0.58	0.62

TABLE VI

EXPERIMENTAL RESULTS ON THE ICDAR 2013 DATASET. OUR PERFORMANCE IS COMPARED TO THE LAST PUBLISHED RESULTS IN [4]

Method	Year	*Precision*	*Recall*	*Fmeasure*
Our method	–	**0.93**	0.73	**0.82**
Zhang *et al.*[4]	2015	0.88	**0.74**	0.80
iwrr2014 [61]	2014	0.86	0.70	0.77
USTB TexStar [5]	2014	0.88	0.66	0.76
Text Spotter [6]	2012	0.88	0.65	0.75

TABLE VII

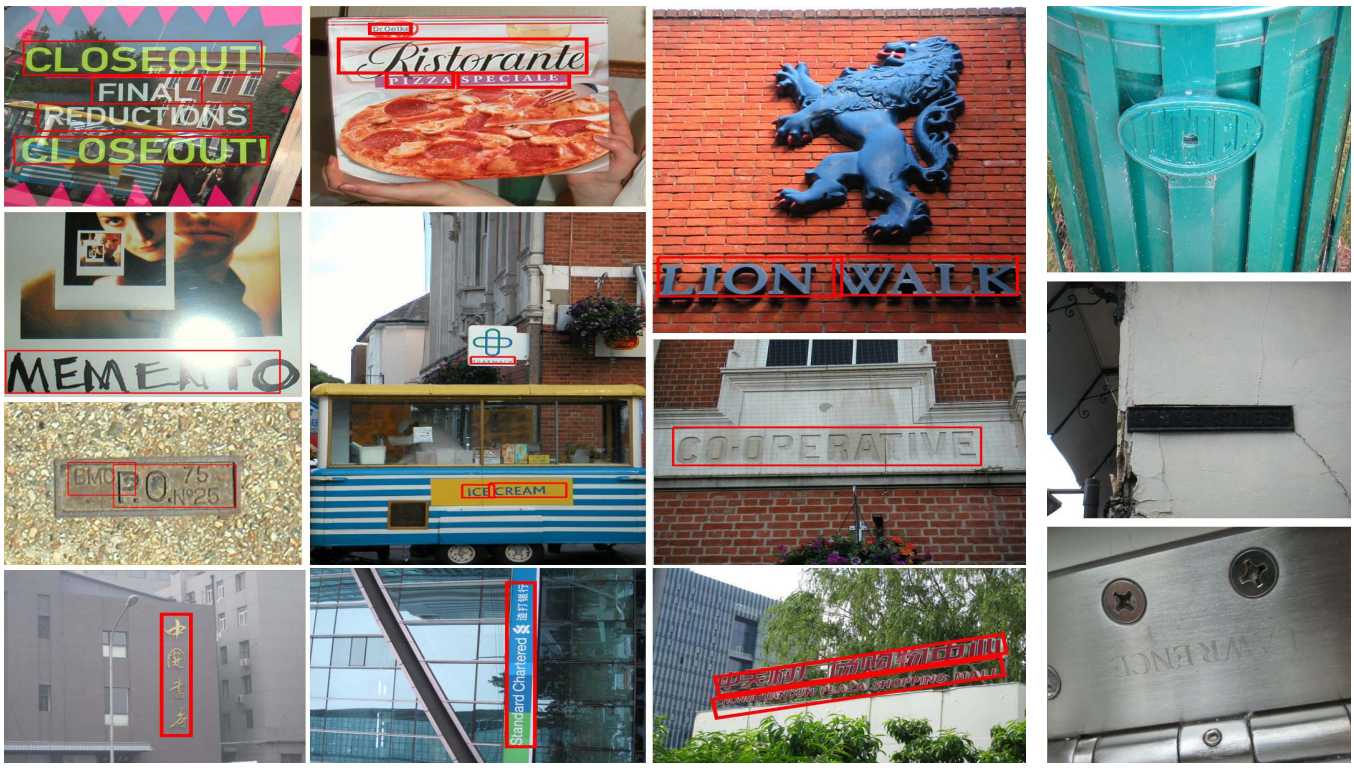
EXPERIMENTAL RESULTS ON THE MSRA-TD500 DATASET

Method	Year	*Precision*	*Recall*	*Fmeasure*
MSRA-TD500				
Our method	–	0.76	0.61	0.69
Yin *et al.* [21]	2015	**0.81**	**0.63**	**0.71**
Yin *et al.* [5]	2014	0.71	0.61	0.66
Kang *et al.* [23]	2014	0.71	0.62	0.66
Yao *et al.* [35]	2014	0.62	0.64	0.61
Yao *et al.* [7]	2012	0.63	0.63	0.60
ICDAR2011				
Our method	–	**0.91**	**0.74**	**0.82**
Yin *et al.* [21]	2015	0.84	0.66	0.74
ICDAR2013				
Our method	–	**0.93**	**0.73**	**0.82**
Yin *et al.* [21]	2015	0.84	0.65	0.73

The running time of the CE-MSER and Text-CNN was evaluated on the ICDAR 2011. The average time for the CE-MSER is about 4.1s per images by our MATLAB implementation, compared to 1.2s of the original MSER (also implemented in MATLAB). The average time for the Text-CNN is about 0.5s per image implementing in CAFFE framework [54] with a single GPU. The implementation of CE-MSER can be accelerated significantly with more engineering optimization, e.g. about 0.3s/image by the MSER, as reported in [55].

3) Evaluation on Full Text Detection: The full evaluation of the proposed system was conducted on three benchmarks. We follow standard evaluation protocol of the ICDAR 2011 by using the DetEval tool offered by authors of [62], and the ICDAR 2013 standard in [63]. The performance of the proposed approach in terms of *Precision*, *Recall* and *F-measure*, is compared in Table IV, V and VI. Our detection system achieves promising results on all three datasets, which improve recent results considerably. In the ICDAR 2005 and 2011 datasets, the proposed approach outperforms the MSERs-CNN [1] substantially with 4% improvement in F-measure. The large improvement on recall is mainly beneficial from strong ability of the enhanced CE-MSERs detector which robustly identifies more distorted text patterns. While the Text-CNN improves discriminative power, so that it reduces the number of false alarms dramatically, and thus boosts the precision. Comparing to the closest performance achieved by Zhang *et al.* [4] on the ICDAR 2011, the Text-CNN obtains a 7% improvement on precision. Finally, the proposed system is further evaluated on the ICDAR 2013 dataset. It achieves a high precision of 0.93 with 0.73 recall and 0.82 F-measure, which advance recent published results with 0.88 precision, 0.74 recall, and 0.80 F-measure [4].

The detection results on a number of challenging images are shown in Fig. 12. The correct detection samples demonstrate



(a) (b)

Fig. 12. Successful detection results on changeling examples, and failure cases. (a) Successful examples. (b) Failure cases.

high performance of our system with strong robustness against multiple text variations and significantly cluttered background. The failure cases have extremely ambiguous text information and are easily confused with its background. They are even hard to human detection.

4) Evaluation on Multi-Orientation and Multi-Language Text: The generality of the proposed method to multi-language and multi-orientation text lines is further investigated. We conducted experiments on the MSRA-TD500 dataset. For training the Text-CNN, we synthetically generate 30,000 Chinese character images and corresponding masks. There are totally 2,300 character classes including 2,238 Chinese character classes and 62 English characters in the $CharTrain$. The training process follows previous experimental settings.

The results are reported in Table VII, with comparisons against state-of-the-art performance on this benchmark [21], [23], [35]. As can be found, our method achieves a F-measure of 0.69, significantly outperforming recent methods of [35] (with 0.61) and [23] (with 0.66). Our result comes tantalizingly close to the best performance of 0.71 F-measure achieved by Yin *et al.* [21]. Notice that all three methods [21], [23], [35] compared are specially designed for multi-oriented text detection. On the other hand, we also compare our performance with Yin *et al.*'s approach [21] on near-horizontal text in the Table VII, where our method obtains large improvements, by 8-9% on the ICDAR 2011 and 2013. These results demonstrate the efficiency and generality of our method convincingly.

In fact, we aims to solve fundamental challenge for this task by providing more accurate or reliable character candidates. It could be readily incorporated with a more sophisticated method for text line construction (e.g., [21], [23]), and then better performance can be expected.

V. Conclusion

We have presented a new system for scene text detection, by introducing a novel Text-CNN classifier and a newly-developed CE-MSERs detector. The Text-CNN is designed to compute discriminative text features from an image component. It leverages highly-supervised text information, including text region mask, character class, and binary text/non-text information. We formulate the training of Text-CNN as a multi-task learning problem that effectively incorporates interactions of multi-level supervision. We show that the informative multi-level supervision are of particularly importance for learning a powerful Text-CNN which is able to robustly discriminate ambiguous text from complicated background. In addition, we improve current MSERs by developing a contrast enhancement mechanism that enhances region stability of text patterns. Extensive experimental results show that our system has achieved the state-of-the-art performance on a number of benchmarks.

References

[1] W. Huang, Y. Qiao, and X. Tang, "Robust scene text detection with convolution neural network induced MSER trees," in *Proc. 13th Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 497–511.

[2] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proc. 1st Int. Conf. Pattern Recognit. (ICPR)*, 2012, pp. 3304–3308.

[3] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *Int. J. Comput. Vis.*, vol. 116, no. 1, pp. 1–20, 2015.

[4] Z. Zhang, W. Shen, C. Yao, and X. Bai, "Symmetry-based text line detection in natural scenes," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2558–2567.

[5] X. C. Yin, X. Yin, K. Huang, and H. W. Hao, "Robust text detection in natural scene images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 970–983, May 2014.

[6] L. Neumann and K. Matas, "Real-time scene text localization and recognition," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3538–3545.

[7] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1083–1090.

[8] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 512–528.

[9] P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," in *Proc. 30th AAAI Conf. Artif. Intell. (AAAI)*, 2016.

[10] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "PhotoOCR: Reading text in uncontrolled conditions," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2013.

[11] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1480–1500, Jul. 2015.

[12] K. InKim, K. Jung, and J. HyungKim, "Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1631–1639, Dec. 2003.

[13] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun./Jul. 2004, pp. II-366–II-373.

[14] S. M. Hanif and L. Prevost, "Text detection and localization in complex scene images using constrained AdaBoost algorithm," in *Proc. 10th Int. Conf. Document Anal. Recognit. (ICDAR)*, 2009, pp. 1–5.

[15] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 1457–1464.

[16] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2012, pp. 1–11.

[17] Y. F. Pan, X. Hou, and C. L. Liu, "A hybrid approach to detect and localize texts in natural scene images," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 800–813, Mar. 2011.

[18] M. Ozuysal, P. Fua, and V. Lepetit, "Fast keypoint recognition in ten lines of code," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.

[19] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.

[20] J. Šochman and J. Matas, "WaldBoost—Learning for time constrained sequential detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 150–156.

[21] X. C. Yin, W. Y. Pei, J. Zhang, and H. W. Hao, "Multi-orientation scene text detection with adaptive clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1930–1937, Sep. 2015.

[22] W. Huang, Z. Lin, J. Yang, and J. Wang, "Text localization in natural images using stroke feature transform and text covariance descriptors," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 1241–1248.

[23] L. Kang, Y. Li, and D. Doermann, "Orientation robust text line detection in natural images," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 4034–4041.

[24] Y. Li, W. Jia, C. Shen, and A. van den Hengel, "Characterness: An indicator of text in the wild," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1666–1677, Apr. 2014.

[25] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2963–2970.

[26] L. Neumann and K. Matas, "Text localization in real-world images using efficiently pruned exhaustive search," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2011, pp. 687–691.

[27] L. Neumann and J. Matas, "Scene text localization and recognition with oriented stroke detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 97–104.

[28] C. Yi and Y. Tian, "Text string detection from natural scenes by structure-based partition and grouping," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2594–2605, Sep. 2011.

[29] 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. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2012, pp. 2609–2612.

[30] L. Sun, Q. Huo, W. Jia, and K. Chen, "A robust approach for text detection from natural scene images," *Pattern Recognit.*, vol. 48, pp. 2906–2920, Sep. 2015.

[31] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image Vis. Comput.*, vol. 22, no. 10, pp. 761–767, 2004.

[32] D. Nistér and H. Stewénius, "Linear time maximally stable extremal regions," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2008, pp. 183–196.

[33] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 391–405.

[34] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[35] C. Yao, X. Bai, and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Trans. Image Process.*, vol. 23, no. 11, pp. 4737–4749, Nov. 2014.

[36] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Proc. IEEE 11th Int. Conf. Comput. Vis. (ICCV)*, Oct. 2007, pp. 1–8.

[37] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 589–600.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[39] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[40] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–14.

[41] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[42] Y. Le Cun *et al.*, "Handwritten digit recognition with a back-propagation network," in *Proc. Neural Inf. Process. Syst. (NIPS)*, 1989, pp. 396–404.

[43] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[44] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural image," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016.

[45] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2007, pp. 1–8.

[46] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[47] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Learning deep representation for face alignment with auxiliary attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[48] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 818–833.

[49] H. Fu, X. Cao, and Z. Tu, "Cluster-based co-saliency detection," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3766–3778, Oct. 2013.

[50] M. M. Cheng, G. X. Zhang, N. J. Mitra, X. Huang, and S. M. Hu, "Global contrast based salient region detection,"2011 in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 409–416.

[51] S. M. Lucas, "ICDAR 2005 text locating competition results," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, 2005, pp. 80–84.

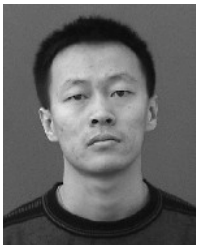
[52] A. Shahab, F. Shafait, and A. Dengel, "ICDAR 2011 robust reading competition challenge 2: Reading text in scene images," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, 2011, pp. 1491–1496.

[53] D. Karatzas *et al.*, "ICDAR 2013 robust reading competition," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, 2013, pp. 1484–1493.

[54] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[55] M. Buta, L. Neumann, and J. Matas, "FASText: Efficient unconstrained scene text detector," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1206–1214.

[56] C. Yi and Y. Tian, "Text extraction from scene images by character appearance and structure modeling," *Comput. Vis. Image Understand.*, vol. 117, no. 2, pp. 182–194, Feb. 2013.
[57] J. Zhang and R. Kasturi, "Character energy and link energy-based text extraction in scene images," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, 2010, pp. 308–320.
[58] L. Neumann and J. Matas, "On combining multiple segmentations in scene text recognition," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, 2013, pp. 523–527.
[59] C. Shi, C. Wang, B. Xiao, Y. Zhang, and S. Gao, "Scene text detection using graph model built upon maximally stable extremal regions," *Pattern Recognit.*, vol. 34, no. 2, pp. 107–116, 2013.
[60] Á. González, L. M. Bergasa, J. J. Yebes, and S. Bronte, "Text location in complex images," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, 2012, pp. 617–620.
[61] A. Zamberletti, L. Noce, and I. Gallo, "Text localization based on fast feature pyramids and multi-resolution maximally stable extremal regions," in *Proc. Workshop Asian Conf. Comput. Vis. (ACCV)*, 2014, pp. 91–105.
[62] C. Wolf and J.-M. Jolion, "Object count/area graphs for the evaluation of object detection and segmentation algorithms," *Int. J. Document Anal. Recognit.*, vol. 8, no. 4, pp. 280–296, 2006.
[63] *ICDAR 2015 Robust Reading Competition*, accessed on Aug. 2015. [Online]. Available: http://rrc.cvc.uab.es



Tong He received the B.S. degree in marine technology from the Tianjin University of Science and Technology, Tianjin, China, in 2013, where he is currently pursuing the M.S. degree with the School of Remote Sensing and Information Engineering, Wuhan University. He has been with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, where he is a Visiting Graduate Student. His research interests are text detection, text recognition, general object detection, and scene classification.

Weilin Huang (M'13) received the the B.Sc. degree in computer science from the University of Shandong (China), the M.Sc. degree in internet computing from the University of Surrey (U.K.), and Ph.D. degree in electronics engineering from the University of Manchester, U.K., in 2012. He is currently a Research Assistant Professor with the Chinese Academy of Science, and a joint member with the Multimedia Laboratory, Chinese University of Hong Kong. His research interests include computer vision, machine learning, and pattern recognition. He has served as a Reviewer for several journals, such as the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART B, and *Pattern Recognition*.

Yu Qiao (SM'13) received the Ph.D. degree from the University of Electro-Communications, Japan, in 2006. He was a JSPS Fellow and Project Assistant Professor with the University of Tokyo from 2007 to 2010. He is currently a Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests include pattern recognition, computer vision, multimedia, image processing, and machine learning. He has authored over 90 papers. He received the Lu Jiaxi Young Researcher Award from the Chinese Academy of Sciences in 2012.

Jian Yao received the B.Sc. degree in automation from Xiamen University, China, the M.Sc. degree in computer science from Wuhan University, China, and the Ph.D. degree in electronics engineering from The Chinese University of Hong Kong, in 2006. From 2001 to 2002, he was a Research Assistant with the Shenzhen R&D Centre, City University of Hong Kong. From 2006 to 2008, he was a Post-Doctoral Fellow with the Computer Vision Group, IDIAP Research Institute, Martigny, Switzerland. From 2009 to 2011, he was a Research Grantholder with the Institute for the Protection and Security, Citizen, European Commission–Joint Research Centre, Ispra, Italy. From 2011 to 2012, he was a Professor with the Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, China. Since 2012, he has been a Chutian Scholars Program Distinguished Professor with the School of Remote Sensing and Information Engineering, Wuhan University, China, and the Director of the Computer Vision and Remote Sensing Laboratory. He has authored over 70 papers in international journals and proceedings of major conferences and holds ten patents. His current research interests mainly include computer vision, machine vision, image processing, pattern recognition, machine learning, LiDAR data processing, SLAM, and robotics.