JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 37, 259-277 (2021)
DOI: 10.6688/JISE.202101_37(1).0016

Understanding Engineering Drawing Images From Mobile Devices

XIAO-PIN ZHONG[1], HAI-JIAN CHEN[1], MENG-QIN LI[1] AND WEN-WEN ZENG[+,2]
[1]*College of Mechatronics and Control Engineering*
[2]*Library*
Shenzhen University
Shenzhen, 518060 P.R. China
E-mail: {xzhong; zengww}@szu.edu.cn; chen.haijian@foxmail.com; mengqin_li@126.com

Traditional engineering drawings are widely used but not easily digitally accessible or searchable. This paper presents a novel method for digital recognition of engineering drawings understanding. We investigated two tasks that determine the performance and accuracy of a recognition method: drawing classification and character sequence recognition. Engineering drawings consist of three types, and each type contains different geometric features. First, we propose a new method combining random sample consensus and geometric features to address the classification problem. The classification error of this method is less than 5%, and we designed a strategy that enables users to correct misclassifications. After precise classification of drawings, the feature information extractor can be applied effectively. Second, we use an end-to-end neural network combining the convolutional neural network (CNN) and recurrent neural network to recognize sequence labels. In contrast to traditional character recognition methods such as those that use support vector machine and CNN technology, the proposed end-to-end neural network architecture integrates character segmentation, feature extraction, and character recognition. The performance of this character recognition method on real-world engineering drawings was shown to be robust and competitive.

Keywords: text recognition, deep learning, end-to-end neural network, feature extraction, object detection

1. INTRODUCTION

"Construction according to plan" is an important principle of quality control in civil engineering. Engineering drawings are the basis for construction and inspection. However, investigations have found that when using traditional drawings there are at least but not limited to the following three disadvantages.

- Each set of engineering drawings contains dozens or even hundreds of papers. It is necessary to copy dozens or even hundreds of drawings that have been reviewed and stamped by relevant departments. The drawing cost is very high.
- A set of drawings is very heavy, and the drawings are at least divided into floor plans and node diagrams. It is not easily portable and highly inefficient when search and switch between those two types of drawings.
- These drawings are reproductions produced on blue paper coated with a photosensitive coating. They do great harm to our body. Besides they waste a lot of paper and are not friendly to environment.

Received March 5, 2019; revised May 30, 2019; accepted September 7, 2019.
Communicated by Chu-Song Chen.
[+] Corresponding author.

In recent years, the Chinese departments have implemented some reforms to use the archive's digitization of working drawings and the electronic review of drawings. However, due to Chinese national laws and regulations, only the drawings reviewed and confirmed with a stamp and signature by the relevant departments can be used as the basis for construction. Therefore, the current construction is still according to the "blueprint" but not a drawing of an electronic version of CAD.

In view of the shortcomings of the traditional printed drawings, this paper proposes a new method of using drawings on the mobile terminals, which can greatly reduce costs, improve efficiency, save paper and is harmless to human body. The method photographs the working drawings through the mobile terminal, recognizes the type of the drawings, and identifies the characters of field codes, so as to realize the automatic entry and the inquiry of drawings.

The proposed recognition method consists of two parts: drawing classification and sequence label recognition. The first step towards transforming an engineering drawing into a labeled image is scanning, which individuals perform using mobile phones with flexible image acquisition style, thus enhancing the complexity of the scanned image background. In addition, the scene complexity, uneven lighting, and image blurring, drawing degradation and distortion also present challenges to recognition methods for engineering drawings [35]. Because there are three types of engineering drawings, each with different geometric features, the drawings must be classified before labels are identified from their geometric features. In drawing classification, we focus on geometry extraction, including circle detection, table detection, and stroke-width detection. Most previously proposed classification systems [1, 10, 22, 25, 27, 33] detect circles and tables according to curvature, and the images are taken from a flatbed scanner, which, unlike a mobile phone camera, cannot detect paper degradation [26] or drawing distortion. Therefore, in the proposed method, we detect circles by using random sample consensus (RANSAC) technique. A RANSAC-table method is designed for detection of table distortion. Using these subtasks for geometry feature detection, a drawing can be precisely classified, and label location is extracted from the geometric features for further sequence recognition.

Related studies such as [3, 8] divide text recognition into text location, text segmentation, and single-character recognition. Single-character recognition involves machine learning [3] and convolutional neural networks [34]. In some state-of-the-art designs, networks combine text detection, text segmentation, and character recognition [20] or employ sequence recognition architecture [28]. Focusing on efficiency and drawing recognition performance, we designed a neural network architecture based on label segmentation and sequence recognition. This design is proven to be effective and efficient for engineering drawing recognition.

2. RELATED WORKS

With regard to the engineering drawings discussed in this paper, labels are identified according to one of three geometric shapes: a circle, table, or underline. The geometric features have to be extracted to precisely classify an engineering drawing before sequence label recognition can be performed.

Engineering drawing recognition methods involve two main tasks: engineering drawing classification and character sequence recognition. In this section, a brief introduction to related works on engineering drawing classification and character sequence recognition is presented.

2.1 Geometric Feature Extraction

In general, engineering drawings can be classified into one of three types, which are distinguished by label location. Different locations correspond with different geometric features.

Traditional circle detection methods are roughly comprised of the following categories: Hough transforms; geometric hashing; template matching; stochastic techniques, including RANSAC techniques; and genetic algorithms [1]. The Hough transform-based methods have been proven to be simple and effective for circle detection. The Hough transform detects a circle through a voting procedure implemented in a parameter space. However, the greater the complexity of the background and the smaller the radius of the circle, the more expensive the transform becomes to compute; additionally, the performance and efficiency of the Hough transform decreases. Moreover, the Hough transform is not sensitive to broken circles. When target circle has translation and scale change, template matching method needs a lot of search time, which makes this method difficult to be applied; Genetic algorithm has high requirement on image and weak robustness of environment; The geometric hashing approach is applied to match geometric features against a database of such features Geometric hashing encodes the model information in a pre-processing step and stores it in a hash table. The major disadvantage of the method is that the same subset has to be chosen for the model image as for the previously acquired images. RANSAC's advantage is its ability to do robust estimation of the model parameters, *i.e.*, it can estimate the parameters with a high degree of accuracy even when outliers are present in the data set so the comparatively efficient and robust RANSAC approach for detecting circles in engineering drawings is adopted for the method proposed in this paper.

Numerous methods have been proposed to address the problem of table detection, because tables include vital information such as summaries and comparisons. However, most traditional image processing methods are based on the use of a flatbed scanner [10, 22, 27, 33]. An approach for detecting the frame lines of a table using the Hough transform was presented in [33], whereas the method discussed in [27] was developed to detect different document layouts, and the approach presented in [22] relies on the detection of differences between table columns gaps and text line gaps. With the development of the mobile devices, methods such as that detailed in [26] have been developed that focus on table detection in camera-captured document images where the boundary of the table is clear and not connected to other information. However, because of the complexity of engineering drawings, a table boundary is not isolated; rather, it is connected to the background of the whole drawing. Therefore, in this paper we propose a method based on RANSAC to solve the problem of complex table detection.

2.2 Character Recognition

Character recognition refers to alphanumeric recognition of printed or handwritten characters. Character recognition approaches can be divided into two categories: single-character recognition and sequence character recognition.

2.2.1 Single-character recognition

Previous research on single-character recognition has mainly been devoted to building a robust character classifier that can adapt to various font sizes and backgrounds. Most single-character recognition methods divide character recognition into a sequence of distinct tasks: preprocessing, segmentation, and recognition [4]. Preprocessing is performed to remove noise from the camera or scanner. Preprocessing includes converting the image to gray scale, performing blur operation and threshold to extract the foreground. After extracting the foreground of the characters, single characters can be segmented using the boundary of each character or the projection of a histogram when the characters are connected. Single characters are then classified using pattern matching [16] or classifiers of a support vector machine (SVM) with manually selected features such as Hog [7].

The recognition performance of SVMs, however, is limited by manually selected low-level features in images [15]. To address this problem and enhance robustness, the present study proposes using numerous neural networks. Recent advances in deep neural network technology have enhanced the performance of single-character recognition systems. The pioneer of convolutional neural networks (CNN), Lecun, designed the first CNN architecture for isolated handwriting digit recognition in [19]. Lecun's contribution prompted the development of numerous recognition methods based on neural network architectures [15, 35]. These methods usually perform two tasks: character detection and recognition. Depending on the mode of detection and recognition, the methods can be classified as stepwise or integrated [35]. Single-character recognition methods mainly use stepwise methodologies and include four steps: text or character location, verification, segmentation, and recognition. In [6], a CNN-based approach that localizes and detects horizontal text lines is presented. In this approach, the network learns to extract and combine text features rather than using handcrafted features. In [5], an improved network architecture is proposed for narrowing the gap between machine and human performance. In [36], a deeper and slimmer CNN based on GoogLeNet [32] is presented improve the performance of end-to-end handwritten Chinese character recognition. These methods mainly treat isolated character recognition and subsequent word recognition separately. Next, we introduce methods that perform end-to-end recognition.

2.2.2 Sequence character recognition

The majority of recent developments in end-to-end sequence or scene text recognition can be classified into two categories. The first category is sequence end-to-end recognition, which detects a sequence in an image [29] and then outputs the sequence. The second category is end-to-end text spotting, which jointly addresses the techniques of text detection, segmentation, and recognition [21]. Methods in the second category contain a neural network architecture that integrates feature extraction, sequence modeling, and recognition into a unified framework. The network architecture has three parts: convolution layers, which extract the feature sequence; recurrent layers, which predict the sequence distribution; and a recognition layer, which translates the prediction into a result. The advantages of this unified architecture are that it is not confined to a predefined lexicon and that, in contrast to the architecture presented in [25], it can handle a sequence of arbitrary length. In [21], a unified network is presented that simultaneously localizes and recognizes text using a single forward pass, rendering image cropping and character grouping unnecessary. This network consists of convolutional layers, modified from the VGG network, as an encoder that generates a sequence of feature representations that can be used to create a recurrent neural network (RNN) decoder for sequence recognition.

3. METHODOLOGY

Recognition methods must address the complexity of engineering drawing backgrounds, the variety of engineering drawing categories, and the variety of image acquisition conditions that may affect by a camera. In this section, we describe our engineering drawing recognition method, which consists of geometric feature classification and end-to-end sequence recognition.

3.1 Geometric Feature Classification

As previously mentioned, civil engineering drawings comprise three types. Each type of engineering drawing contains many blocks, and each block includes a unique label (see example in Fig. 1). The difference between the three types of drawings is the location of the labels. Different label locations represent different geometric features. Fig. 1 (a) presents the first type of image with circle geometry. Fig. 1 (b) shows the second type of image with table geometry. Finally, Fig. 1 (c) illustrates the third type of image with underlined geometry. To classify engineering drawings, these geometric features must be precisely extracted. This paper presents several approaches for classifying engineering drawings according to whether they include circle, table, or underlined shapes.

As Fig. 1 (a) demonstrates, the image quality of engineering drawings varies from clear to blurry because of variations in camera focus. The intensity of the images also differs according to their original print quality and how well they have been conserved. The size of the target circle in each red bounding box varies according to the distance between the camera and the drawing. In addition, several distortions in the images increase the difficulty of circle detection.

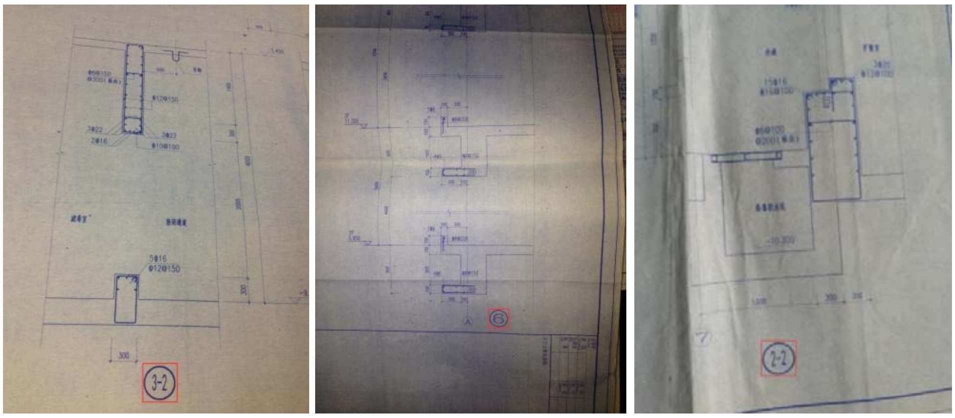
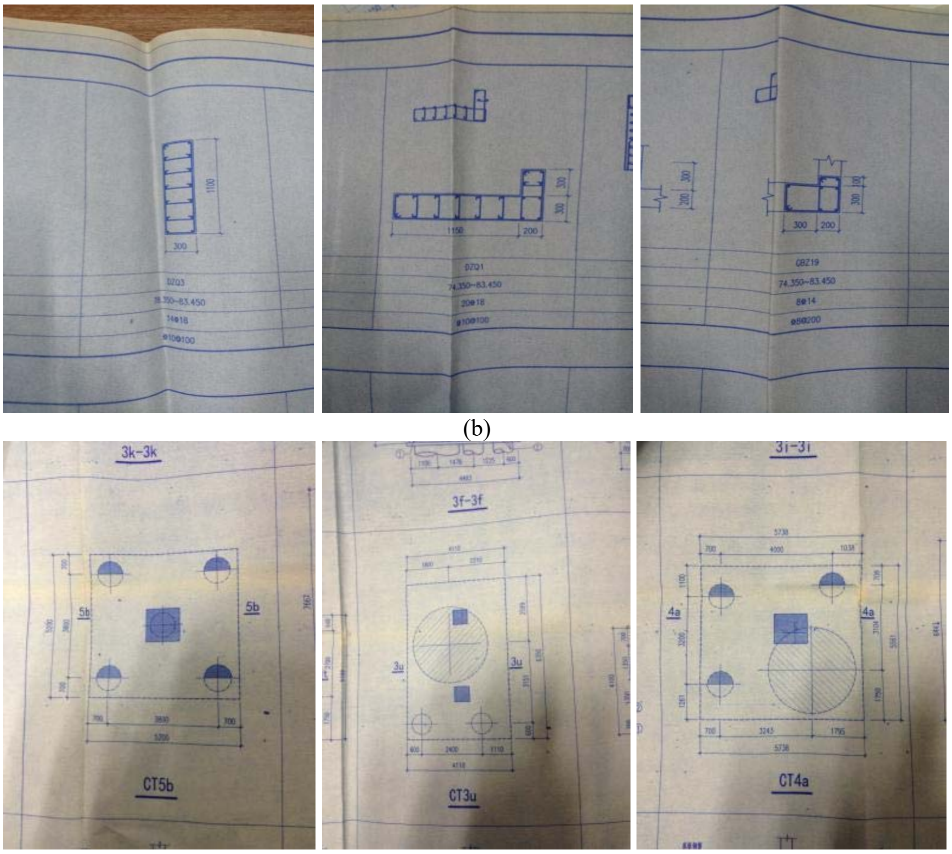


Fig. 1. (a) Type 1: images of engineering drawings with labels presented in circles.



(b)

(c)

Fig. 1. (b) Type 2: images of engineering drawings with labels presented in tables; (c) Type 3: images of engineering drawings with underlined labels.

We designed the following four steps to enhance circle detection: (1) thresholding; (2) identifying contours and deleting the objects that cannot be circles; (3) applying morphology methods to further delete horizontal and vertical lines; and (4) using the RANSAC method to locate the potential circles.

Applying this process, we begin with a raw image, which is then converted into a binary pattern. Because of the variation in light conditions, an adaptive threshold method is employed to ensure high binary performance. There are many Adaptive threshold methods. Among them Sauvola binarization is a good choice because this method can solve the problem of bad binarization of uneven illumination image especially in the case of document image. After obtaining the binary image, the contours of all the connected components of the binary image are calculated and the region whose ratio of width to height is too large or too small is deleted, where the ratio of width to height should be approximately one for a circle. Considering that engineering drawings contain numerous lines, the gradient of the object in the circle is calculated both horizontally and vertically. As shown in Fig. 2 when the horizontal gradient is calculated, the horizontal lines vanish because of the gradient in the horizontal direction is zero; the same phenomenon occurs for vertical lines under vertical gradient calculation. If an engineering drawing being analyzed under this method contains a circle, the results after deleting the objects inside the circle should be like those in Fig. 2 (b). In our procedure, to detect the circle further, three points from point sets in Fig. 2 (b) that are not in a line are randomly selected, and the unique circle is calculated accordingly. If there are not enough points from point sets of Fig. 2 (b) fall on the calculated circle, these three points will be thrown away from point sets and three new points are selected. The RANSAC procedure ends after enough points have been found to fall exactly on the calculated circle, indicating that the circle has been detected, or when no points remain on the point sets or too many iterations have been performed.

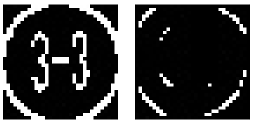


Fig. 2. Detected circle (left) before and (right) after removal of the inside objects.

0	0	1	0	0
1	1	1	1	1
0	0	1	0	0

Fig. 3. Dilate kernel used to enhance table lines.

Most traditional image processing methods are based on images obtained using flatbed scanners, and the simple Hough transform that these traditional approaches employ cannot detect the table in our engineering drawings, as Fig. 1 (b) shows. In this paper, we propose a method based on RANSAC to solve the problem of complex table detection. A flowchart of the proposed table detection method is shown in Fig. 4. As previously described, the raw image is converted into a binary image, and the gradient of the binary image in their direction is calculated to suppress vertical lines. Morphology methods are then applied using a diamond-shaped dilate kernel to enhance the lines of the tables. The kernel is defined as shown in Fig. 3.

As the dilate equation shows, the diamond-shaped kernel joins adjacent horizontal lines in case the table is broken after thresholding:

$$dst(x, y) = \max_{(x', y'):\mathrm{kernel}(x', y')\neq 0} src(x+x', y+y'). \tag{1}$$

To exclude objects in the image without table, the connected components are calculated, and after vertical gradient calculation and diamond-shaped dilation, the contours of the image are identified. Then, the objects whose bounding boxes are too small are deleted. Generally, the stroke width of an image after dilation is larger than its original stroke width; therefore, the skeleton of the dilated image is calculated in case of an expensive calculation. After obtaining the skeleton of the table, the RANSAC method for table detection is performed. Before the iterations reach the specified limit and while unused points remain in the image; (1) a threshold of slope is set two points in the skeleton image are randomly selected; if the slope between these two points is larger than the threshold, these points are skipped and another two points are selected until the slope between two points is less than the threshold; (2) the distance between each point in the skeleton image and the line calculated in Step 1 is calculated, and the number of these points close to the line is counted; if a sufficient number of points are close to the line, the line is saved and these points are thrown away to prevent double-checking; (3) when the iteration limit is met or there are no points left in the image, the lines are analyzed and whether they constitute a table in the engineering drawing is judged.

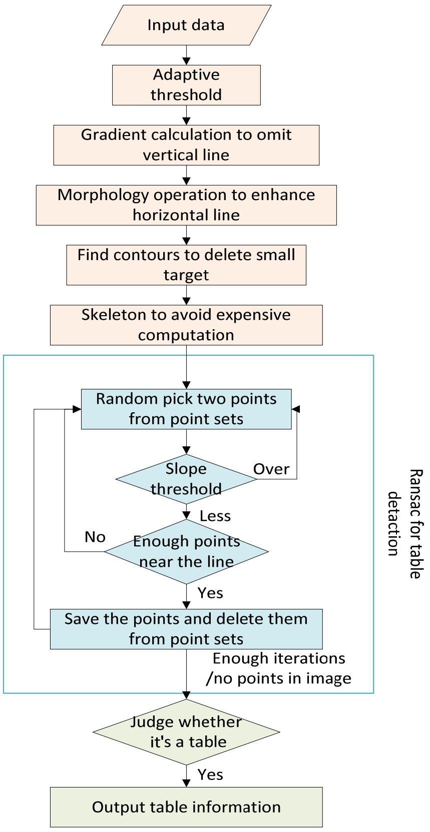

Input data
Adaptive threshold
Gradient calculation to omit vertical line
Morphology operation to enhance horizontal line
Find contours to delete small target
Skeleton to avoid expensive computation
Random pick two points from point sets
Slope threshold
Over
Less
No
Enough points near the line
Yes
Save the points and delete them from point sets
Enough iterations /no points in image
Ransac for table detection
Judge whether it's a table
Yes
Output table information

Fig. 4. Flowchart of the proposed table detection method.

After determining whether engineering drawings contain circles or tables, lines under drawing labels become easier to detect. According to the design of the engineering drawings presented in this paper, the underline, as shown in Fig. 1 (c), has a larger stroke width compared with other information in the drawing. To locate the label of the specific drawing, we use the stroke width transform (SWT), as devised in [9]. This process does not involve machine learning or elaborate tests. In brief, after Canny edge detection is applied to the input image, the thickness of each stroke that makes up objects in the image is calculated; thereafter, the line under the label is easily identified by calculating the ratio of height to width, because the line under the label has a relatively large stroke width.

3.2 Sequence Recognition

Traditional segmentation methods such as those based on SVM, Hog, SWT [9], and maximally stable external regions rely heavily on character detection and therefore result in inadequate character segmentation performance when an image contains uneven lighting, blurry text, drawing degradation, and distortion (Fig. 5). Thus, we propose an architecture consisting of neural networks rather than the traditional technologies. The network of the proposed sequence recognition method comprises three components: a convolution network for sequence feature extraction, a recurrent network for prediction of each sequence feature, and a recognition layer with connectionist temporal classification (CTC) loss [12].

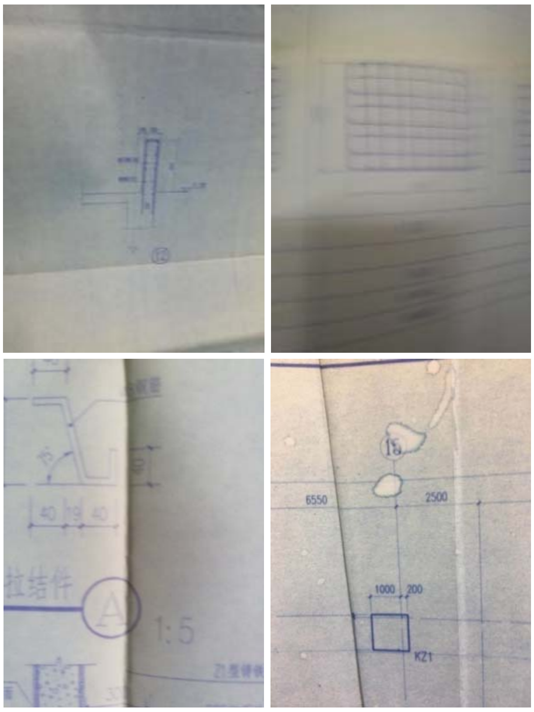


Fig. 5. Examples of uneven lighting, blurry text, drawing degradation, and a distorted image.

The feature extraction layer is inspired by VGG [16] and is constructed using convolutional and max-pooling layers. First, the input image containing the sequence text is fed into a CNN that is modified from VGG-16 network architecture. VGG-16 consists of 13 layers of convolutions followed by a rectified linear unit [23] (*i.e.*, 5 layers of max-pooling and 3 fully connected (FC) layers). Because VGG-16 is designed for large-scale image recognition and the sequence images described in this paper are relatively small, we remove some convolutional layers and max-pooling layers. To accelerate training, a batch normalization layer is added after the convolutional layers. Because the convolutional networks described in this paper are designed for feature extraction, we also delete the FC and softmax layers. Before the sequence images are fed into the CNN, they are preprocessed to a fixed height. The feature vectors of a sequence image after being processed by the CNN are generated from left to right. Consequently, the feature vectors after CNN processing are sequential and equal to each other in size; therefore, they can be considered as a sequence image feature and sent to the recurrent layers.

After receiving the sequential feature vectors from the CNN, a deep bidirectional RNN, bidirectional long short-term memory (BLSTM), predicts the sequence feature. As shown in [29], there are three advantages to recurrent layers: first, long short-term memory (LSTM) [11] is extremely capable of capturing contextual information within a sequence, and the BLSTM takes advantage of the input image bidirectionally; second, convolutional layers and recurrent layers can be trained within a unified network; third, the recurrent layers can operate on sequences of arbitrary lengths, traversing from the start to the end. In the recurrent layers in this study, the input feature vectors are considered a sequence and are fed into the RNN. As a precaution against gradient vanishing and explosion [2], we apply LSTM, which addresses the problem of exponential decay of gradient information other than vanilla RNN. The basic unit in the hidden layer of an LSTM network is memory blocks, each of which contain memory cells and multiplicative gates, including input gates, output gates, and forget gates. In general, the memory cells store information for a period of time and can be reset by the forget gates once they are no longer needed. Traditional LSTM only uses past information. However, information from both the left and right is important in an image-based sequence recurrent network. Hence, we combine the forward and backward memory cells into BLSTM, as shown in Fig. 6. Deep BLSTM has achieved significant performance in speech recognition [13]. To connect the back propagation of the CNN and LSTM, we concatenate the sequence vectors into maps, which are fed into the CNN during back propagation.

After the BLSTM has generated predictions, sequence recognition must be performed. In statistics, sequence recognition refers to matching the sequence label with the input image that has the highest probability of all the possible sequence labels. A method for training RNNs to label unsegmented sequences directly is presented in [12]. This method relies on the transformation of neural network outputs into a conditional probability distribution over label sequences. We adopt the CTC method for the prediction of label sequences. The formulation of CTC can be described as follows [12]: Let the input be denoted by i, sequence output as $o = o_1, \ldots, o_T$, and length of a sequence as T. o_k^t represents the probability of output label k at time t, which is interpreted as a distribution over the set $L'^T = L \cup \{blank\}$, where L indicates all the possible labels of the output sequence, (*e.g.*, all the English characters and Arabic numerals). π denotes the elements of L'^T.

$$p(\pi \mid i) = \prod_{t=1}^{T} o_{\pi_t}^{t} \tag{2}$$

After obtaining the elements of L'^T, we define the map B (*i.e.*, $L'^T \rightarrow L^{\leq T}$), which can be regarded as removing all blanks and repeated labels from π, (*i.e.* the elements of L'^T). For example, B maps '– – *h h*– *e*–*l*–*ll* –*o o* – –' onto '*hello*', where the '–' stands for '*blank*'. Finally, we define the conditional probability of a given label $l \in L^{\leq T}$ as the sum of the probabilities of all the π corresponding to it:

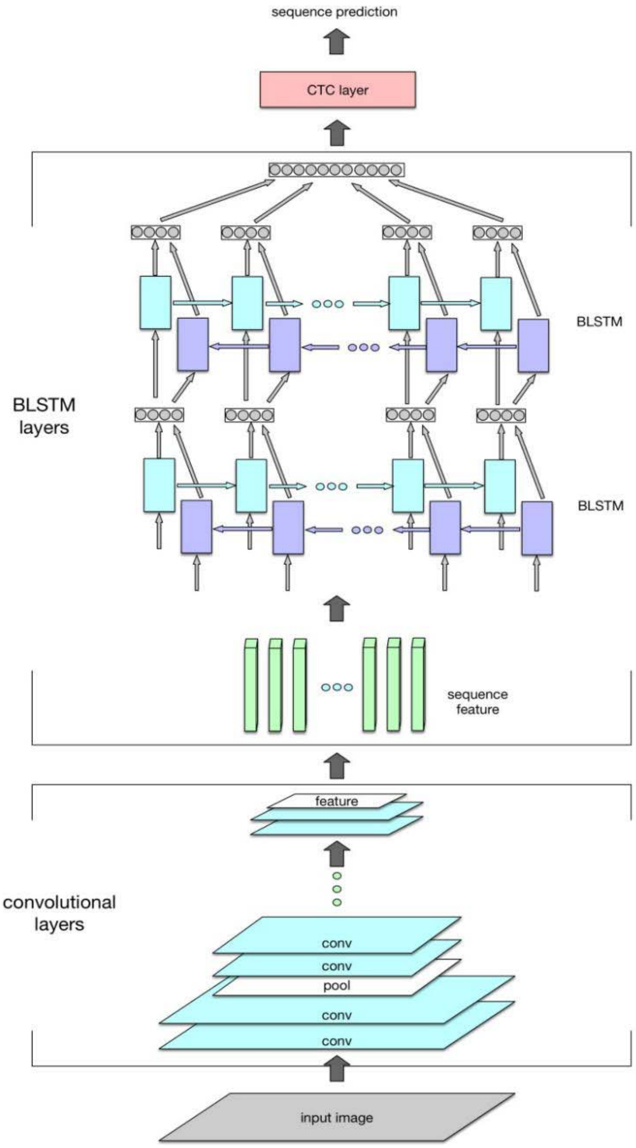


Fig. 6. Neural network architecture of convolutional layers and BLSTM layers.

$$p(l\,|\,i) = \sum_{\pi \in B^{-1}(l)} p(\pi\,|\,i). \tag{3}$$

The conditional probabilities of $p(l|i)$ can be calculated efficiently using a dynamic programming algorithm. Given Eq. (3), the output of the recognition should be the most probable labels for the input sequence, which can be described as $B(\arg\max)_{\ l\in L^{\leq T}} p(l|i)$. When training the dataset, the objective function can be regarded as in [12]. Because the cost function is calculated directly from an input image and the labeled se-quence, the network can be trained end-to-end, avoiding manual, individual character labeling. Furthermore, the convolutional networks and recurrent networks are trained using stochastic gradient descent. In BLSTM layers, back propagation through time is applied to calculate error differentials. The Adam [18] algorithm is performed to accelerate the training.

The whole flowchart about sequence location and recognition as shown in Fig. 7, where input image is engineering drawing and output is character sequence.

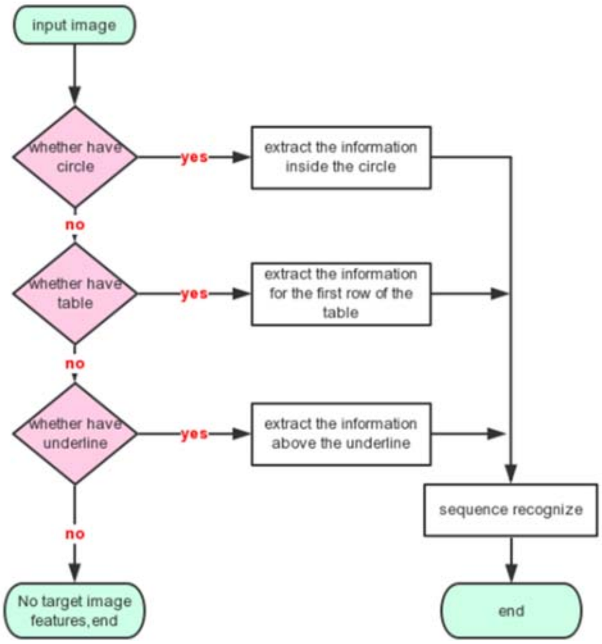

input image
whether have circle
yes
extract the information inside the circle
no
whether have table
yes
extract the information for the first row of the table
no
whether have underline
yes
extract the information above the underline
sequence recognize
no
No target image features, end
end

Fig. 7. Flowchart of sequence location and recognition.

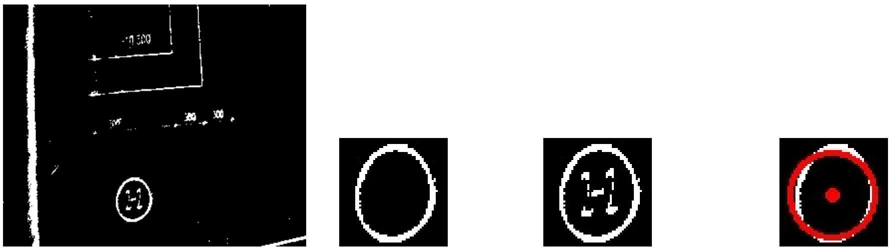
4. RESULTS AND DISCUSSION

In this section, we describe experiments conducted to evaluate the efficiency of the engineering drawing recognition method. The experiments consisted of drawing classification and drawing label recognition. The proposed method was designed using OpenCV and C++, and the network architecture Tensorflow was used in sequence recognition. We implemented the experiments using a workstation with a 3.20 GHz Intel(R) Core(TM) i5-4460 CPU and an NVIDIA(R) GeForce GTX 1080. The neural network is trained for nearly 20 hours.

4.1 Evaluation of Drawing Classification

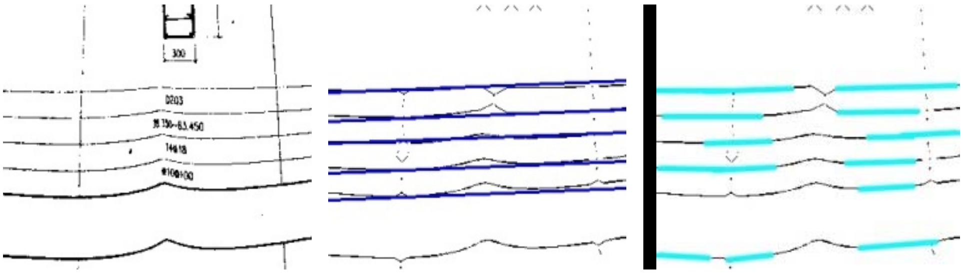
As previously described, drawing classification is related to geometric feature classification. In this experiment, the engineering drawing dataset was collected using a mobile phone. There were three drawing categories. The recognition method processed the raw images to detect their geometric features, determine the drawing's category, and locate the label near the geometric feature. Numerous experiments were performed to evaluate the proposed geometric feature classification system. Because the geometric features were located in the lower half of the raw images, we cropped the raw data when conducting geometric feature detection. After the adaptive threshold of the lower half data was reached, contours of the foreground were identified and small objects were removed to accelerate the detection. Figs. 8 and 9 display the geometric feature classification after performance of different traditional methods for circle detection and table detection, respectively.

Fig. 8 (a) presents the lower-half of a raw drawing image after the threshold was applied, indicating the input image. Fig. 8 (b) shows the result of identifying contours and deleting targets unlikely to be circles. Using the proposed method, we attempted to enhance the detection accuracy of proposed circle detection. Fig. 8 (c) shows the circle detection results obtained using RANSAC. As shown in Fig. 8 (d), the Hough circle transform could not find the circle because of its rough boundary and shape distortion.



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

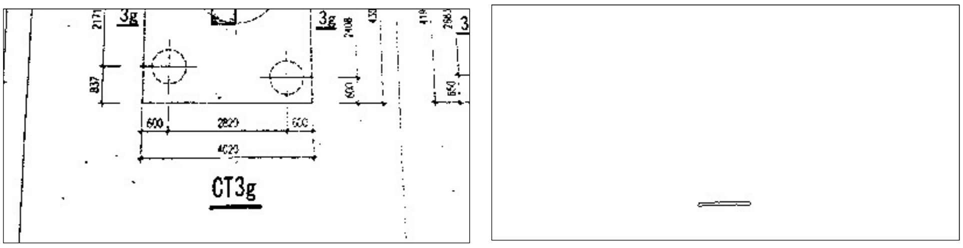
Fig. 8. (a) Lower-half raw image after the adaptive threshold was applied; (b) circle proposal region; (c) circle proposal region after removal of inner contours, where the red circle is the circle detected using RANSAC; and (d) results from the same input as (c) after applying Hough Circle detection.



(a) (b) (c)

Fig. 9. (a) Lower-half image after the adaptive threshold was applied; (b) Table proposal region from which small targets have been removed; the blue lines are the table using RANSAC-table; (c) Table proposal region obtained using the probabilistic Hough transform.

Fig. 9 (a) shows a lower-half table image after the adaptive threshold was applied, regarded as the input image; Fig. 9 (b) presents the result of RANSAC-table detection, and Fig. 9 (c) presents the results of using the probabilistic Hough transform. Comparing these two table detection results, we may conclude that traditional table detection cannot adapt to the distortion and breakage of lines and that RANSAC-table detection is more flexible regarding table distortion and the complexity of the engineering drawing background.



(a) (b)

Fig. 10. (a) Underlined engineering drawing; (b) after SWT detection.

SWT was used to detection underline as shown in Fig. 10. Fig. 10 (a) is the binary graph of the image after preprocessing. According to the calculation of the stroke width of the image, the connected component analysis method can usually be used to accurately locate the underline with wide stroke in the image, as shown in Fig. 10 (b).

The circle detection, stroke-width detection, and proposed RANSAC-table detection methods were tested and validated using numerous engineering drawing images. The datasets included images with uneven lighting, image blurring, drawing degradation, and varying degrees of distortion. The classification error obtained was less than 5%, and the classification data are available to download[1]. Additionally, we designed a strategy for users to correct misclassifications. In sum, the classification method proved to be practical and robust when applied to the engineering drawing understanding.

4.2 Evaluation of Sequence Recognition

After extracted and classified, the precise positioning of the engineering drawing character sequence is performed from engineering drawings so the corresponding mark of the drawing, that is, the character sequence can be obtained. Since the end-to-end neural network requires millions of marked data, the sample's label marked by manually is too large so in this article we use the character sequence generated by the script first. The specific operation of generating the training data has the following steps: extracting the background information of the multiple engineering drawings as the background of the generated image; randomly adding the characters conforming to the engineering drawing font, and using the same color as the engineering blueprint characters; adding the blur noise, deformation, illumination and other disturbances get the final composite map. The generated data include training data set and test data set has about 1.2 million end-to-end automatically labeled composite images. The ratio of the training set to the test set is 9:1. In [14] the authors released a synthetic dataset for text localization that requires high-performance text detectors and recognition. The dataset consists of millions of images synthesized by an engine to generate text to naturally meld with existing background images. We first applied the synthetic dataset to training to obtain accurate initial values for the sequence recognition networks. After millions of iterations, the accuracy of sequence recognition reached 91.3%. As a reported in [29] that CNN and LSTM architecture has achieved outstanding performance with regard to four popular text recognition benchmarks, as presented in Table 1.

[1] www.alors.cn/mengqinli/rawdata.zip

Table 1. Performance of different architectures on four popular benchmarks.

	architecture	End-to-end	IIIT5k	SVT	IC03	IC13	Size
Bissacco *et al.* [3]		×	–	78.0	–	87.6	–
Jaderberg *et al.* [17]	CNN	√	–	80.7	93.1	90.8*	490M
Jaderberg *et al.* [16]	CNN	√	–	71.7	89.6	81.8	304M
CRNN [29]	CNN + LSTM	√	81.2	82.7	91.9	89.6	8.3M

Table 2. Architecture details of the CNN+LSTM neural network.

Type	Kernel size	Stride (Vert and Horz)	Output dim.	H * W	Pad op.
Convolution	3*3	1	64	30*30	Same
Convolution	3*3	1	64	30*30	Same
MaxPooling	2*2	2	64	15*15	
Convolution	3*3	1	128	15*15	Same
Convolution	3*3	1	128	15*15	Same
MaxPooling	2*2	2,1	128	7*14	
Convolution	3*3	1	256	7*14	Same
Convolution	3*3	1	256	7*14	Same
MaxPooling	2*2	2,1	256	3*13	
Convolution	3*3	1	512	3*13	Same
Convolution	3*3	1	512	3*13	Same
MaxPooling	3*1	3,1	512	1*13	
BLSTM			512		
BLSTM			512		

In [17], the recognition accuracy for correctly cropped words is 98%, whereas the end-to-end text spotting F-score is only 69%, mainly due to incorrect and missed-word-region proposals [14].

As Table 2 shows, the network architecture in the present study consisted of the CNN, RNNs including BLSTM layers, and a CTC layer. Fig. 11 displays part of the sequence image for the engineering drawings, and the full dataset is available.[2] The sequence recognition accuracy was 92.97%, and the label accuracy reached 98.47%, which is highly competitive. Fig. 12 shows some results of sequence recognition. The samples including uneven lighting, blurry text, drawing degradation, and a distorted image. Our method can correctly identify most of the sequence, but some over blurry text has the wrong results as shown in Figs. 12 (e) and (f).

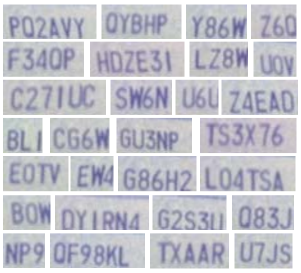


Fig. 11. Part of the sequence image from an engineering drawing.

[2] www.alors.cn/mengqinli/dataset.zip

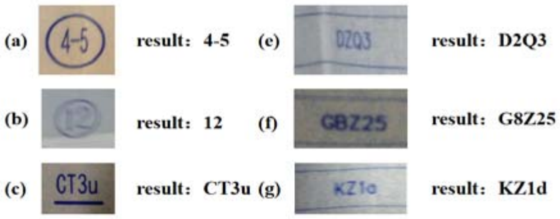


Fig. 12. Some result of sequence recognition.

5. CONCLUSIONS

In this paper, we presented a novel engineering drawing recognition method with image processing and neural networks. The method facilitates efficient image classification using a CNN and BLSTM. The RANSAC-table detection method was proposed for precise detection of tables in engineering drawings. The neural network architecture of this method combines modified VGG and two layers of BLSTM to extract the features of a sequence image. After numerous experiments, the CTC loss has proven to be efficient and effective in sequence recognition. In conclusion, the proposed method has been proven effective for the recognition of engineering drawings, and it has addressed the issues of document accessibility and searchability. Furthermore, this recognition method can be applied in other domains such as architecture, mechanics, transportation, and electrical engineering.

ACKNOWLEDGMENTS

This work was supported in part by the Shenzhen Science and Technology Program under Grant JCYJ20180305123922293 and the National Taipei University of Technology, Shenzhen University Joint Research Program under Grant 2019009. The authors would also like to thank the anonymous reviewers and Shenzhen Xiangshan Technology Ltd. providing lots of samples of engineering drawings.

REFERENCES

1. V. Ayala-Ramirez, C. H. Garcia-Capulin, A. Perez-Garcia, and R. E. Sanchez-Yanez, “Circle detection on images using genetic algorithms,” *Pattern Recognition Letters*, Vol. 27, 2006, pp. 652-657.
2. Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, Vol. 5, 1994, pp. 157-166.
3. A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, “Photoocr: Reading text in uncontrolled conditions,” in *Proceedings of IEEE International Conference on Computer Vision*, 2013, pp. 785-792.
4. P. K. Charles, V. Harish, M. Swathi, and C. Deepthi, “A review on the various techniques used for optical character recognition,” *International Journal of Engineering Research and Applications*, Vol. 2, 2012, pp. 659-662.

5. D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Convolutional neural network committees for handwritten character classification," in *Proceedings of IEEE International Conference on Document Analysis and Recognition*, 2011, pp. 1135-1139.
6. M. Delakis and C. Garcia, "Text detection with convolutional neural networks," in *Proceedings of International Conference on Computer Vision Theory and Applications*, Vol. 2, 2008, pp. 290-294.
7. J. Dong, A. Krzyżak, and C. Y. Suen, "An improved handwritten Chinese character recognition system using support vector machine," *Pattern Recognition Letters*, Vol. 26, 2005, pp. 1849-1856.
8. B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2963-2970.
9. B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform, in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2963-2970.
10. B. Gatos, D. Danatsas, I. Pratikakis, and S. J. Perantonis, "Automatic table detection in document images," in *Proceedings of IEEE International Conference on Pattern Recognition and Image Analysis*, 2005, pp. 609-618.
11. F. Gers, "Long short-term memory in recurrent neural networks," Ph.D. Thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2001.
12. A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd ACM International Conference on Machine Learning*, 2006, pp. 369-376.
13. A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proceedings of IEEE International Conference on Acoustics*, *Speech and Signal Processing*, 2013, pp. 6645-6649.
14. A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2315-2324.
15. P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," in *Proceedings of the 13th AAAI Conference on Artificial Intelligence*, 2016, pp. 3501-3508.
16. M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep structured output learning for unconstrained text recognition," ArXiv Preprint ArXiv:1412.5903, 2014.
17. M. Jaderberg, K. Simonyan, A. Vedald, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *International Journal of Computer Vision*, Vol. 116, 2016, pp. 1-20.
18. D. P. Kingma and J. Ba, "Adam: A method for Stochastic optimization," *CoRR abs/ 1412.6980*, 2014.
19. Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," *Advances in Neural Information Processing Systems*, Vol. 2, 1990, pp. 396-404.
20. H. Li, P. Wang, and C. Shen, "Towards end-to-end text spotting with convolutional recurrent neural networks," *ArXiv Preprint ArXiv:1707.03985*, 2017.

21. H. Li, P. Wang, and C. Shen, "Towards end-to-end text spotting with convolutional recurrent neural networks," *ArXiv Preprint ArXiv:1707.03985*, 2017.
22. S. Mandal, S. Chowdhury, A. K. Das, and B. Chanda, "A simple and effective table detection system from document images," *International Journal on Document Analysis and Recognition*, Vol. 8, 2006, pp. 172-182.
23. V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning*, 2010, pp. 807-814.
24. Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proceedings of NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011, pp. 5.
25. R. Scitovski and T. Marošević, "Multiple circle detection based on center-based clustering," *Pattern Recognition Letters*, Vol. 52, 2015, pp. 9-16.
26. W. Seo, H. I. Koo, and N. I. Cho, "Junction-based table detection in camera-captured document images," *International Journal on Document Analysis and Recognition*, Vol. 18, 2015, pp. 47-57.
27. F. Shafait and R. Smith, "Table detection in heterogeneous documents," in *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, 2010, pp. 65-72.
28. B. Shi, X. Bai, and C. Yao, "Detecting oriented text in natural images by linking segments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, 2017, pp. 2550-2558.
29. B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, 2017, pp. 2298-2304.
30. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ArXiv Preprint ArXiv:1409.1556*, 2014.
31. S. Singh, "Optical character recognition techniques: a survey," *Journal of Emerging Trends in Computing and Information Sciences*, Vol. 4, 2013, pp. 545-550.
32. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1-9.
33. Y. Tian, C. Gao, and X. Huang, "Table frame line detection in low quality document images based on Hough transform," in *Proceedings of IEEE 2nd International Conference on Systems and Informatics*, 2014, pp. 818-822.
34. T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proceedings of IEEE 21st International Conference on Pattern Recognition*, 2012, pp. 3304-3308.
35. Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 37, 2015, pp. 1480-1500.
36. Z. Zhong, L. Jin, and Z. Xie, "High performance offline handwritten Chinese character recognition using google net and directional feature maps," in *Proceedings of IEEE 13th International Conference on Document Analysis and Recognition*, 2015, pp. 846-850.

Xiaopin Zhong received the B.Sc. degree in Automation, the M.Sc. degree in Pattern Recognition and Intelligence from Xi'an Jiaotong University, and the Ph.D. degree in Mechatronic Wngineering from the Ecole Centrale de Lyon, in 2002, 2007, and 2010, respectively. He is currently an Associate Professor with the Laboratory of Machine Vision and Inspection, College of Mechatronics and Control Engineering, Shenzhen University, China. His research interests include computer vision and pattern recognition.

Haijian Chen is currently a Postgraduate at the College of Mechatronics and Control Engineering, Shenzhen University, China. His research interests include computer vision and 3D measurement.

Mengqin Li received the B.Sc. degree in Measurement-Control Technology and Instrumentation from Hubei University of Technology, and the M.Sc. degree in Control Engineering from Shenzhen university in 2015, and 2018, respectively. She is currently an image processing engineer in SFT Technology.

Wenwen Zeng received the B.Sc. degree in Communication Engineering from the University of Central Lancashire, the M.Sc. degree in Communication and Signal Processing from the University of Newcastle, and the Ph.D. degree in Photo-Electronic Engineering from Shenzhen University, in 2005, 2006, and 2016, respectively. She is currently a Librarian in Shenzhen University Library, China. Her research interests include computer vision and library information technology.