IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 19, 2019, accepted May 7, 2019, date of publication May 14, 2019, date of current version May 24, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916616

Natural Scene Text Recognition Based on Encoder-Decoder Framework

LING-QUN ZUO[1], HONG-MEI SUN[1,2], QI-CHAO MAO[1], RONG QI[1], AND RUI-SHENG JIA[1,2]

[1]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[2]Shandong Province Key Laboratory of Wisdom Mine Information Technology, Shandong University of Science and Technology, Qingdao 266590, China

Corresponding author: Hong-Mei Sun (shm0221@163.com)

This work was supported in part by the Key Research and Development Program of Shandong Province, China, under Grant 2017GSF20115, and in part by the Natural Science Foundation of Shandong Province, China, under Grant ZR2018MEE008.

ABSTRACT Aiming at the situation that complex natural scene text is difficult to recognize a scene text recognition method based on an encoder–decoder framework is proposed. The method converts the natural text recognition into a sequence mark by combining the connection time classification (CTC) and attention mechanism under the encoder–decoder framework, in order to overcome the problem of character segmentation, using the correlation between image and text sequence. First of all, a convolutional neural network (CNN) is used to generate an ordered feature sequence from the entire word image. Then, the generated feature sequence is feature-coded using the bidirectional long short-term memory (Bi-LSTM) network. Finally, an integrated module of the CTC and attention mechanism is designed to decode and output the text sequence. The experiments show that compared with the comparison method, the recognition accuracy of the method is improved obviously.

INDEX TERMS Natural scene text recognition, encoder-decoder framework, CNN, Bi-LSTM.

I. INTRODUCTION

Optical Character Recognition (OCR) is a computer vision recognition technology that targets natural scene texts and has achieved remarkable success in many commercial applications. These applications include billboard reading, blind assistive technology, robot navigation, and more. Traditional OCR technology is suitable for recognizing high-quality documents with clear backgrounds, simple fonts, and neatly arranged uniformity. However, the scene OCR aims at solving scene text recognition, such as traffic signs, screens, bills, street scenes, goods, and so on. These text images are shown in Figure. 1. They have complex background, uneven illumination, low contrast, blurred fonts, font distortion, and multiple colors. Traditional OCR technology cannot be well recognition. Therefore, the study of scene text recognition methods has become a research hotspot in the field.

The technology of scene OCR mainly includes two categories: character-based recognition and whole word-based recognition. Based on character recognition, Bai *et al.* [1] learn the middle stroke features by clustering image blocks, then use HOG voting algorithm to detect characters, and finally use random forest classifier to classify; Wang *et al.* [2] proposed a new text recognition system based on the general target detection method of computer vision, which used the character confidence and the space constraint relationship between characters to give the most likely recognition result. Bissacco *et al.* [3], Alsharif and Pineau [4], Mishra *et al.* [5], and Jaderberg *et al.* [6] recognize each segmented character by first detecting a single character and then using a convolutional neural network as a classifier. The above method based on character recognition is confusing between characters and characters, cannot effectively utilize context information, and relies on an accurate character classifier, which seriously affects overall recognition performance. For the defects based on character recognition, Goel *et al.* [7], Rodriguez [8], Goodfellow *et al.* [9], and Jaderberg *et al.* [10] proposed a recognition method based on whole words. Goel et al.

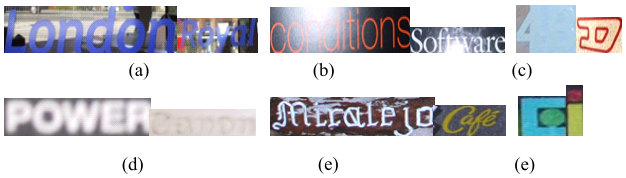


FIGURE 1. Text under 6 interference factors. (a) Complex background. (b) Uneven illumination. (c) Low contrast. (d) Font blur. (e) Distortion. (e) Multi-color.

The associate editor coordinating the review of this manuscript and approving it for publication was Amjad Mehmood.

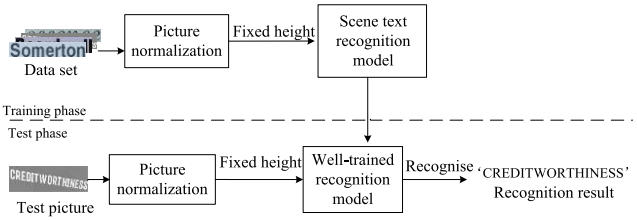


FIGURE 2. Scene text recognition flow chart.

use gradient-based feature maps to compare pre-made word images, and use dynamic neighborhood methods to determine the words contained; Rodriguez et al. used the integrated Fisher vector and SVM (Support Vector Machine) to establish the relationship between the picture and the entire word encoding; Goodfellow et al. used CNN to encode the entire picture, and then used a multi-position character classifier to perform text recognition; However, the above text recognition methods based on whole words are not end-to-end, and rely on detailed sample annotation, which is time-consuming and labor-intensive to train. Jaderberg et al. identified scene text recognition as an image classification problem, with each class corresponding to a word in a predefined large dictionary consisting of approximately 90K words. However, since the pre-refined dictionary is too large and the number of training samples is large, it is difficult to generalize to other cases with large numbers of classes; In order to solve this dilemma, Shi *et al.* [11], Lee and Osindero [12], Cheng *et al.* [13], and Shi *et al.* [14] used the scene text recognition as a sequence identification problem, Directly generate of tag sequences by a recurrent neural network [15] that connection time classification or attention mechanisms, which improves the accuracy of scene text recognition significantly. However, we note that existing attention-based methods do not perform well when dealing with complex or low-quality images. One of the main reasons is that existing methods cannot accurately align the feature areas and targets of these images. Call this phenomenon "attention drift".

In order to solve this problem, we design based on the Encoder-Decoder framework, using the correlation of image and text sequence to design Bi-LSTM to encode the image features, and then integrate the CTC and Attention modules to decode the output features. Compared with the single CTC mechanism or Attention mechanism, the decoding effect is better and the recognition accuracy is significantly improved.

II. MODEL DESIGN

A. MODEL

We use a combination of CTC-Attention mechanism in the Encoder-Decoder framework widely utilized in speech, image and video data in recent years to propose a new scene text recognition model. Scene text recognition can be divided into two parts, the training phase and the test phase, as shown in Figure. 2. In the training phase, all the training sets are first normalized to the image, and the purpose is to fix the input image to a scaled image with a constant height and width; then input the normalized image into the convolutional

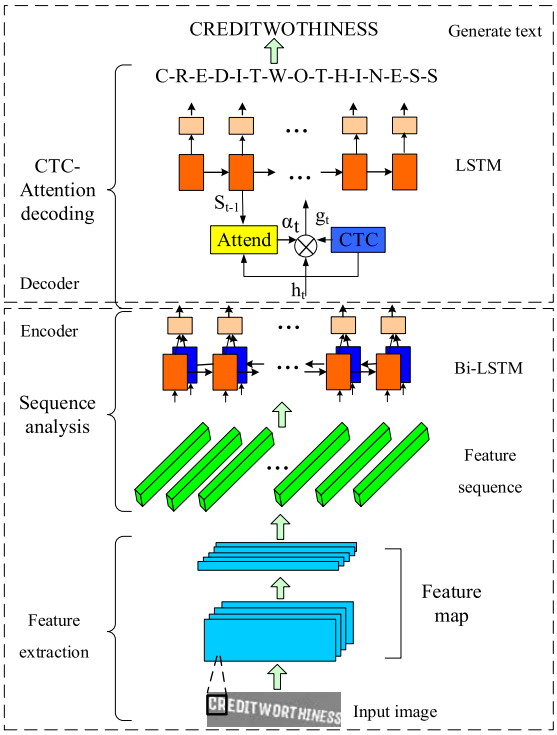


FIGURE 3. Text recognition model framework.

neural network for feature extraction. The extracted sequence is subjected to feature coding output through the Bi-LSTM network; then the CTC-Attention joint mechanism is used for training, and the parameters are gradually adjusted to optimize the model. In the test phase, you only need to input the image into the fully trained model to output the correct result.

B. MODEL FRAMEWORK

The scene text recognition framework in this paper includes the Encoder-Decoder process. The scene image to be recognized first extracts the feature sequence through the convolutional neural network of the Encoder code layer, and then uses the bidirectional long-term memory network to mark the feature. Finally, the labeled feature sequence is sent to the Decoder decoding layer, using the CTC-Attention joint mechanism in Bi-LSTM of the Decoder code layer decodes the output and generates the final recognition result. The process is shown in Figure. 3:

The character recognition model shown in Figure.3 includes three parts: feature extraction, sequence annotation, and decoding. The following describes each recognition process separately.

1) FEATURE EXTRACTION

The key of the scene text recognition model depend on whether the model has rich discriminating ability. To improve the discriminating ability, extracting features is a key step. Usually, the characters in a word are arranged in a line from left to right. First, we use a convolutional neural network as shown in Figure.4 to perform a series of convolution and

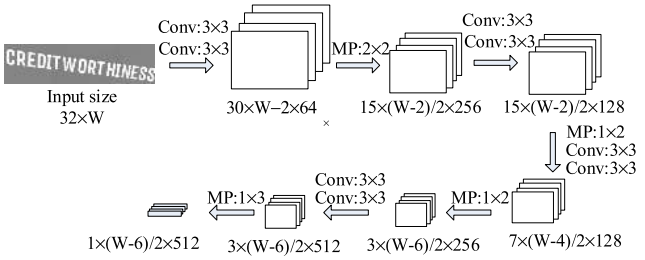


FIGURE 4. Convolutional neural network architecture.

pooling operations, Get a feature map with dimensions of $h_{conv} \times w_{conv} \times d_{conv}$ (h: height, w: width, d: depth), The size of the feature image output in the last step is $1 \times \frac{(w-6)}{2} \times 512$.

W is the width, Conv is the convolution operation, and MP is the maximum pooling operation.

2) SEQUENCE ANALYSIS

Divide the 512 feature maps along the row axis and perform a Map-to-sequence operation. After the splitting, the feature map is converted into a sequence of w_{conv} vectors, each having $h_{conv} \times d_{conv}$ dimensions, record as a sequence set $Q = \{q_1, q_2, q_3, \ldots, q_n\}$.The feature sequences q obtained from the convolutional neural network contain useful context information, which is important for word recognition. RNN has a strong ability to learn the ordered sequence recognition of images. In addition, the internal state change rate of RNN can be better adjusted by recursive weights, which helps to improve the local distortion of input data [16]. In order to expand the context feature, we use a multi-layer Bi-LSTM to perform bidirectional analysis of feature sequences, obtain long-term dependencies in two directions, and output new feature sequences Q of the same length, Recorded as $H = [h_1, h_2, \ldots, h_n]$, among them, n is the amount of w_{conv}.

3) CTC-ATTENTION JOINT MECHANISM DECODING

We use local attention to replace global attention, incorporate CTC modules for integration, and add a one-way LSTM network as the decoding network at the end. The integrated method of CTC-Attention can make full use of all the feature information of the current location through the codec mechanism in the Attention architecture, and can also utilize the feature information of the current location by calculating the global probability in the CTC, thereby Feature information is fully utilized. At the same time, the CTC-Attention mechanism integrates the CTC module in the Attention module, which not only accelerates the convergence speed of the network, but also improves the recognition performance of the network. The new feature sequence H marked by the Bi-LSTM network decoded by the decoding network under the CTC-Attention joint mechanism, and output the decoded character sequence $Y = \{y_1, y_2, y_3, \ldots, y_k\}$. The details are as follows:

The given sequence of feature $H = [h_1, h_2, \ldots, h_n]$, hidden variable $Z = \{z_t \in D \cup blank \,|t = 1, \ldots, W\}$, output a sequence of length L, $H = \{h_l \in D \,|l = 1, \ldots, W\}$. Where W is the number of encoding sequences; T is the number of characters; D is a dictionary containing all characters.

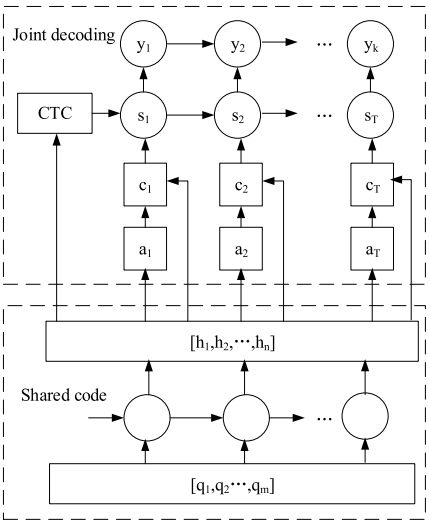


FIGURE 5. CTC-Attention decoding architecture.

CTC assumes that labels are independent, using Bayes' theorem to calculate the posterior probability distribution of the predicted sequence.

$$p_{ctc} = (Y\,|X) \approx \sum_{Z} \prod_{t} p(z_t\,|z_{t-1}, Y)p(z_t\,|X) \tag{1}$$

where $p(Y)$ is the language model of the character level, $p(z_t\,|X)$ is the probability of the hidden variable obtained from the known input feature, and $p(z_t\,|z_{t-1}, Y)$ is the conditional probability of the hidden variable predicted by the hidden variable output at the previous moment. The CTC algorithm assumes that the tags are conditionally independent, and each time the output is a single character probability, which causes the CTC to only predict the local information, ignoring the overall information, and therefore cannot effectively predict long text sequences. Relative to the local prediction of CTC, the attention mechanism directly predicts the text sequence without calculating the hidden variables and making assumptions that are independent of each other within the label, directly calculating the probability of the joint prediction sequence.

$$p_{atten}(Y\,|X) = \prod_{l} p(y_l\,|y_{1:l-1}, X) \tag{2}$$

In the formula (2), $p(y_l\,|y_{1:l-1}, X)$ represents the predicted probability that the input characteristic X and the first l output are obtained.

The attention mechanism does not introduce any constraints that lead to alignment, resulting in noise sensitivity and misalignment during decoding. Therefore, this paper designs a multi-task learning decoder based on Attention and CTC joint training [17]–[19], using Attention to decode the character-level semantics, and using CTC to achieve the constraints of Attention decoding. The CTC-attention decoding algorithm not only effectively solves the problem that the pure data-driven method is difficult to train for long-sequence input, but also extracts information for long characters. The CTC-Attention decoding framework is shown in Figure. 5.

TABLE 1. Text recognition network configuration.

Algorithm 1: Joint CTC/attention One-pass Decoding.

1: *procedure BeamSearch*(X, $L_{\max}$)
2: $\Phi_0 \leftarrow \{< sos >\}$
3: $\hat{\Phi} \leftarrow \varnothing$
4: *for* $l = 1 \ldots L_{\max}$ *do*
5: $\Phi_l \leftarrow \varnothing$
6: *while* $\Phi_{l-1} \neq \varnothing$ *do*
7: $g \leftarrow Head(\Phi_{l-1})$
8: $Dequeue(\Phi_{l-1})$
9: *for each* $c \in \Upsilon \cup \{< Eos >\}$ *do*
10: $Y \leftarrow g \cdot c$
11: $\alpha(Y) \leftarrow \lambda \log p_{ctc}(Y|X) + (1-\lambda) \log p_{atten}(Y|X)$
12: *if* $c =< Eos >$ *then*
13: $Enqueue(\hat{\Phi}, Y)$
14: *else*
15: $Enqueue(\Phi_l, Y)$
16: *if* $|\Phi_l| > BeamWidth$ *then*
17: $DeleteWorst(\Phi_l)$
18: *end if*
19: *end if*
20: *end for*
21: *end while*
22: *if* $EndDetect(\hat{\Phi}, l)$
23: *break*
24: *end if*
25: *end for*
26: *return* $\arg \max_{\hat{Y} \in \hat{\Phi}} \alpha(\hat{Y})$
27: *end procedure*

The CTC and Attention models share the coding network; q is the characteristic sequence of the input coding network; h is the implicit state corresponding to each input when coding; a is the attention weight vector; c is the decoded semantic vector; s represents the decoding network Hidden layer state; y is the predicted output. The maximum probability of joint maximization of CTC and Attention prediction can be expressed as:

$$\hat{Y} = \arg\max_{Y \in D} \{\lambda \log p_{ctc}(Y|X) + (1-\lambda) \log p_{atten}(Y|X)\}$$

The beam search algorithm for one-pass decoding is shown in Table 1. Φ_l and $\hat{\Phi}$ are initialized in lines 2 and lines 3 of the algorithm, which are implemented as queues that accept partial hypotheses of the length l and complete hypotheses, respectively. In lines 4–25, each partial ypothesis g in Φ_{l-1} is extended by each label c in the label set Υ. Each extended hypothesis, Y is scored in line 11, where CTC and attention-based scores are obtained by $\log p_{ctc}$ and $\log p_{atten}$. After that, if $c =< Eoc >$, the hypothesis h is assumed to be complete and stored in $\hat{\Phi}$ in line 13. If $c \neq< Eoc >$, Y is stored in Φ_l in line 15, where the number of hypotheses in Φ_l is checked in line 16.

If the number exceeds the beam width, the hypothesis with the worst score in Φ_l, *i.e.*,

$$Y_{worst} = \arg\min_{Y \in \Phi_l} \alpha(Y, X) \tag{3}$$

is deleted from Φ_l by DeleteWorst() in line 17.

We can optionally apply an end detection technique to reduce the computation by stopping the beam search before l reaches $L_{\max}$. Function EndDetect (Φ_l, l) in line 22 returns true if there is little chance of finding unique hypotheses with higher scores as l increases in future. Specifically, the function returns true

$$\sum_{m=0}^{M-1} \left[\left\{ \max_{Y \in \Phi_l : |Y| = l-m} \alpha(Y, X) - \max_{\hat{Y} \in \Phi_l} \alpha(\hat{Y}, X) \right\} < D_{end} \right] = M \tag{4}$$

where D_{end} and M are predetermined thresholds.

C. TRAINING

The model in this paper is an end-to-end training in multi-tasking to maximize joint probability conversion to minimize multitasking loss function. The objective function is as follows:

$$L = \lambda L_{\text{ctc}} + (1-\lambda) L_{atten} = \lambda[-\ln \mathrm{P}(l|x)] + (1-\lambda)[-\sum\nolimits_u (l_u|x, l_{1:u-1})]$$

where L_{ctc} is the loss function of CTC, L_{atten} is the loss function of the Attention; x is the input sequence, l is the true value sequence, $l_{1:u-1}$ is all the characters before the current true value label, and the variable parameter λ takes the value range $0 \leq \lambda \leq 1$.

III. EXPERIMENTAL DESIGN AND ANALYSIS

A. DATA SET

This paper uses the Synth90k dataset [10] containing 9 million synthetic scene text images as a training set, using SVT dataset [2], IIIT5K dataset [20], ICDAR 2003 dataset [21], ICDAR 2013 dataset [22] was tested as a test set.

Street View Text (SVT) is collected from Google Street View and contains 647 word images in its test set. Many images are severely corrupted by noise and blur, or the resolution is very low. Each image is associated with a 50 word dictionary.

The IIIT 5K-Words (IIIT5K) is collected from the Internet and contains 3,000 cropped word images in its test set. Each image is assigned a 50-word dictionary and a 1k-word dictionary, both of which contain ground-real words and other randomly selected words.

ICDAR 2003 (IC03) contains 251 scene images with text bounding boxes. Each image is associated with a 50-word dictionary defined by Wang *et al.* [2]. For a fair comparison, we discard images that contain non-alphanumeric characters or less than three characters. The resulting data set contains 867 cropped images. The dictionary includes a 50-word dictionary and a complete dictionary containing all the dictionary words.

ICDAR 2013 (IC13) is the successor to IC03, and most of its data is inherited from IC03. It contains 1015 cropped text images. No dictionary associated.

B. TEXT RECOGNITION NETWORK CONSTRUCTION

The construction of the identification network is as shown in Table 2. A $32 * 32$ sample picture is input, and the convolution layer extracts the feature information of the character by stacking four 3×3 size convolution pairs. Under the convolutional neural network, a two-layer Bi-LSTM network is built. Each Bi-LSTM network contains 256 hidden layers. The output of the memory layer is linearly projected to 256 dimensions before the next layer enters the next layer. The decoder is a CTC-Attention mechanism combined with a one-way long-term memory for decoding. The decoder recognizes 94 character classes, namely numbers, uppercase and lowercase letters, and 32 ASCII punctuation marks.

C. EXPERIMENTAL RESULTS AND ANALYSIS

The evaluation index is accuracy, and the calculation method is as follows:

$$accuracy = \frac{M}{N} \tag{5}$$

Among them: M represents the number of samples in the data set to identify the correct sample, and N represents the total number of samples in the data set.

The following is a horizontal comparison of the algorithm of this paper and several current mainstream algorithms on several public datasets. It is obvious that the algorithm of this paper is better than most mainstream algorithms on most public datasets. The comparison results are shown in Table 2.

From Table 3 we can see that compared with the current method, the proposed method achieves a good recognition effect in both the constrained and unconstrained dictionaries. Compared with the CTC or Attention mechanism alone, the CTC-Attention joint mechanism model has a significant improvement in model recognition accuracy, and the model training speed is faster, which proves the validity and superiority of the model. In particular, our approach is always superior to most existing methods in the case of restricted dictionary constraints. In terms of recognition accuracy, it exceeds the scene text recognition model proposed in the advanced representative [Cheng et al.]

Compared with [Cheng et al.], our method has improved the recognition accuracy by 0.4% on the IC03 of the ''1K'' vocabulary, and the recognition accuracy by 0.5% on the IIIT5K dataset of the ''full'' vocabulary. Compared with [Cheng et al.] on the SVT, IC03, and IIIT5K data sets of the ''50'' vocabulary, the recognition accuracy is basically the same. In the absence of dictionary constraints (None), our model recognition accuracy is higher than the above comparison method. On IIIT5k, the recognition accuracy of this method is 2.3% higher than that of the prior art [Cheng et al.]. On IC03, the recognition accuracy of this method is higher than that of [Cheng et al.] by 0.4. %, and the recognition

TABLE 2. Text recognition network configuration.

	Layers	Out Size	Configuration
Encoder	Block1	15*15	[3*3 conv , 64 s : 1 , p=0 3*3 conv , 64 s : 1 , p=1] 2*2 pool , 64 s : 2
	Block2	7*14	[3*3 conv , 128 s : 1 , p=1 3*3 conv , 128 s : 1 , p=1] 2*2 pool , 128 s : 2 , 1
	Block3	3*13	[3*3 conv , 256 s : 1 , p=1 3*3 conv , 256 s : 1 , p=1] 2*2 pool , 256 s : 2 , 1
	Block4	1*13	[3*3 conv , 512 s : 1 , p=1 3*3 conv , 512 s : 1 , p=1] 3*3 pool , 512 s : 3 , 1
	Bi LSTM 1	13	256 hidden units
	Bi LSTM 2	13	256 hidden units
Decoder	CTC-Attention LSTM	-	256 hidden units 256 Attention units
	CTC-Attention LSTM	-	256 hidden units 256 Attention units

" Outsize" represents the length *width of the convolutional layer output, and the sequence length of the loop layer output. S stands for the step size, and p stands for whether it is filled, that is, 0 is not filled, and 1 is filled.

accuracy of [Cheng et al.] increased by 1.3% on the SVT dataset. We observed that IIIT5k contains a lot of irregular text, especially curved text. In this paper, the CTC-Attention joint mechanism method, due to the powerful feature extractor and sequence learning network, has advantages in dealing with complex texts compared to the single CTC or Attention mechanism. Figure 6 lists some of the recognition results. Since most existing methods do not support dictionary-free state, they cannot perform recognition without a dictionary. In contrast, our model is available in both dictionary constraints and dictionary unconstrained settings.

D. DISCUSSION

In the scene text recognition method based on the Encoder-Decoder framework, the number of LSTM layers has a great influence on the recognition accuracy. The deeper LSTM layer number is particularly advantageous for the encoder, but the benefit to the decoder is small. The setting of the two-layer LSTM decoder can achieve good results, and for the encoder, the number of layers is increased to three. The layer tends to be stable. In fact, we tried to further increase the number of LSTM layers in the encoder and decoder, but found that there is a serious over-fitting problem. Tables 4 and 5 show that when changing the number of LSTM layers in the encoder and decoder, The effect of scene text recognition accuracy.

Since the traditional recognition algorithms mostly rely on the dictionary and are not comparable with the

TABLE 3. The accuracy of the scene text recognition on the standard data set.

algorithm	IC03			SVT		IIIT5K			IC13
	50	1k	None	50	None	50	1k	None	None
Wang et al.[2]	76.0	62.0	-	57.0	-	-	-	-	-
Wang et al.[26]	90.0	84.0	-	70.0	-	-	-	-	-
Mishra et.al.[5]	81.8	67.8	-	73.2	-	78.9	-	-	78.9
Novikova	82.8	-	-	72.9	-	64.1	57.5	-	-
TSM+CRF[25]	87.4	-	-	73.5	-	79.3	-	-	79.3
PhotoOCR[24]	86.1	-	-	90.4		85.5	-	-	85.5
Goel et al.[7]	89.7	-	-	77.3	-	-	-	-	-
Rodriguez [8]	-	-	-	70.0	-	-	57.4	-	-
Yao et al.[1]	88.5	80.3	-	75.9	-	-	69.3	-	-
Alsharif et al.[4] .	93.1	-	88.6	74.3	-	-	-	-	-
Su and lu [23]	92.0	82.0	-	83.0	-	-	-	-	-
Gordo[27]	-	-	-	90.7	-	93.3	86.6	-	-
Shi et al.[11]	97.8	96.2	89.9	95.5	81.9	86.7	93.8	81.9	86.7
Lee et al.[12]	97.9	97.0	88.7	96.3	80.7	90.0	94.4	78.4	90.0
Cheng et al.[13]	98.5	96.7	91.5	95.7	82.2	98.9	96.6	83.7	89.4
proposed	98.2	97.1	91.9	95.6	84.5	98.0	97.1	85.4	91.0

"50", "1k", and "Full" indicate the vocabulary used for constraint recognition, and None indicates no constraint.

(a)

(b)

FIGURE 6. (a) Correct and (b) incorrect samples recognized under the CTC-Attention joint mechanism.

TABLE 4. The effect of the number of LSTM layers of the coding layer on the recognition accuracy.

Encoder Layer	IIIT5K	SVT	IC03	IC13
1	83.8	81.3	90.2	89.0
2	84.0	81.9	91.2	90.1
3	84.2	83.0	91.5	90.9
4	83.3	82.1	90.4	89.8

proposed method, the two deep learning methods of CRNN and Attention-OCR are compared with the proposed CTC-Attention joint mechanism. A hyperparameter λ was introduced to balance the weight of the Attention model and CTC. When λ= 1, only use CTC decoding; When λ= 0, only use Attention for decoding. Between λis 0∼1, the CTC imposes different weight constraints on Attention. For the hyperparameter λ, we select 0, 0.2, 0.5, 0.7, 1, respectively, and compare the experimental data on the three data sets IC03, IC13, IIIT5k. The accuracy of the identification is shown in Table 5:

TABLE 5. Effect of the number of LSTM layers of the decoding layer on the recognition accuracy.

Decoder Layer	IIIT5K	SVT	IC03	IC13
1	82.9	81.5	90.2	89.5
2	84.9	82.1	91.2	90.1
3	84.2	83.4	90.5	89.5
4	83.7	89.9	89.7	89.7

TABLE 6. Comparison of recognition accuracy under 0, 1, 0.2, 0.5, and 0.7 (indicators are accuracy, in %).

Method	IC03	IC13	IIIT5K
CRNN (λ=1)	89.4	86.7	90.0
ATTENTION-OCR (λ=0)	88.7	87.2	91.3
CTC-ATTENTION (λ=0.2)	96.2	91.0	97.5
CTC-ATTENTION (λ=0.5)	93.8	89.9	94.0
CTC-ATTENTION (λ=0.7)	92.1	88.5	92.4

It can be seen from Table 5 that the separate CTC or Attention mechanisms are not as good as the joint CTC-Attention mechanism, and the CTC-Attention mechanism can be integrated in the decoding segment to improve the recognition of text. When $\lambda = 0.2$, the text recognition accurate is the highest. As λ increases, the overall recognition rate decreases.

In order to evaluate the algorithm complexity and the convergence speed of the model, every other training batch in the training process on the Synth90k data set outputs its test accuracy on the IC13 test set. The experiment selects the model with parameter $\lambda = 0.2$ and compares it with the test results of CTC and Attention. The results are shown in Figure 7.

It can be seen from Fig. 7 that in the case of convergence, the Attention method performs better than the CTC method on the test set, but the Attention method has the problem

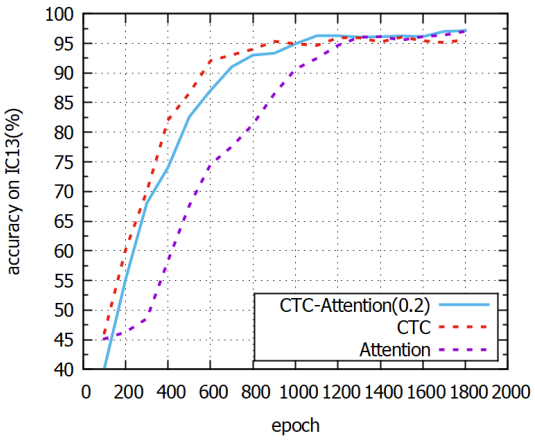


FIGURE 7. Model training curve comparison chart.

of high training time complexity and slow convergence. The CTC method training process converges relatively quickly, but its test accuracy is not as good as the Attention method. CTC-Attention accelerates the training speed of Attention by integrating the Attention layer set into the CTC module, and also improves the overall recognition performance. The performance of the convergence on the test set is better than the two methods.

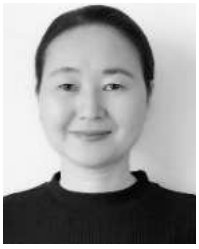
IV. CONCLUSION

Aiming at the problem that the existing scene text recognition method recognizes the poor effect of scene text, we propose an end-to-end network of CTC-Attention joint mechanism by analyzing the advantages and disadvantages of the recent mainstream CTC decoding model and Attention decoding model scene text recognition. Fully combining the advantages of both CTC and Attention, the CTC module is incorporated into the Attention mechanism to fully screen and utilize the feature information. In order to verify the effectiveness of the proposed method, we have done a lot of experiments on multiple benchmarks. The experimental results show that the CTC-Attention joint mechanism significantly improves the recognition performance of the model and has an advantage in identifying scene text images.

REFERENCES

[1] X. Bai, C. Yao, and W. Liu, "Strokelets: A learned multi-scale mid-level representation for scene text recognition," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2789–2802, Jun. 2016.
[2] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1457–1464.
[3] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "PhotoOCR: Reading text in uncontrolled conditions," in *Proc. IEEE Conf. Comput. Vis.*, Dec. 2013, pp. 785–792.
[4] O. Alsharif and J. Pineau, "End-to-end text recognition with hybrid HMM maxout models," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 46–69.
[5] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 1–11.
[6] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *Proc. Euro. Conf. Comput. Vis.*, 2014, pp. 512–528.
[7] V. Goel, A. Mishra, K. Alahari, and C. V. Jawahar, "Whole is greater than sum of parts: Recognizing scene text words," in *Proc. Int. Conf. Document Anal. Recog.*, Aug. 2013, pp. 398–402.
[8] J. A. Rodriguez, *Label Embedding for Text Recognition*. Bristol, U.K.: BMVC, 2013.
[9] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet. (2013). "Multi-digit number recognition from street view imagery using deep convolutional neural networks." [Online]. Available: https://arxiv.org/abs/1312.6082
[10] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," in *Proc. NIPS Deep Learn. Workshop*, 2014, pp. 1–10.
[11] B. Shi, X. Wang, P. Lv, C. Yao, and X. Bai, "Robust scene text recognition with automatic rectification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4168–4176.
[12] C.-Y. Lee and S. Osindero, "Recursive recurrent nets with attention modeling for ocr in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.(CVPR)*, Jun. 2016, pp. 2231–2239.
[13] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing attention: Towards accurate text recognition in natural images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5086–5094.
[14] B. Shi, X. Bai, and C. Yao. (2015). "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition." [Online]. Available: https://arxiv.org/abs/1507.05717
[15] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. 23rd Int. Conf. Mach. Learn.*, Jun. 2006, pp. 369–376.
[16] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.
[17] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Jul. 2017, pp. 4835–4839.
[18] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," in *Proc. Interspeech*, 2017, pp. 949–953.
[19] K. Xu, D. Li, N. Cassimatis, and X. Wang, "LCANet: End-to-end lipreading with cascaded attention-CTC," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit.*, May 2018, pp. 548–555.
[20] A. Mishra, K. Alahari, and C. V. Jawahar, "Top-down and bottom-up cues for scene text recognition," in *Proc. CVPR*, Jun. 2012, pp. 1–8.
[21] S. M. Lucas *et al.*, "ICDAR 2003 robust reading competitions: Entries, results, and future directions," *Int. J. Document Anal. Recognit.*, vol. 7, nos. 2–3, pp. 105–122, 2005.
[22] D. Karatzas *et al.*, "ICDAR 2013 robust reading competition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 1484–1493.
[23] B. Su and S. Lu, "Accurate scene text recognition based on recurrent neural network," in *Proc. Asian Conf. Comput. Vis.*, Aug. 2014, pp. 35–48.
[24] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "Photoocr: Reading text in uncontrolled conditions," in *Proc. IEEE Conf. Comput. Vis.*, Jun. 2013, pp. 785–792.
[25] C. Shi, C. Wang, B. Xiao, Y. Zhang, S. Gao, and Z. Zhang, "Scene text recognition using part-based tree-structured character detections," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, May 2013, pp. 45–69.
[26] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proc. Int. Conf. Pattern Recognit.*, Jun. 2012, pp. 3304–3308.
[27] A. Gordo, "Supervised mid-level features for word image representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Aug. 2015, pp. 2956–2964.

LING-QUN ZUO was born in Shandong, China, in 1991. He received the B.S. degree from Dezhou University, China, in 2017. He is currently pursuing the M.S degree with the Shandong University of Science and Technology. His research interests include image processing and deep learning.



HONG-MEI SUN is currently a Lecturer with the College of Computer Science and Engineering and the Leader of the Key Research and Development Projects of Shandong Province. She is the first author of four publications and the coauthor of five publications. Her research interests include microseismic monitoring technology and software engineering.

QI-CHAO MAO was born in Shandong, China, in 1995. He received the B.S. degree from the Shandong University of technology, China, in 2017, where he is currently pursuing the M.S degree. His research interests include image processing and deep learning.

RONG QI was born in Shandong, China, in 1994. He received the B.S. degree from the University of Jinan, China, in 2017. He is currently pursuing the M.S. degree with the Shandong University of Science and Technology. His research interests include image processing and deep learning.

RUI-SHENG JIA is currently a Full Professor with the College of Computer Science and Engineering, Shandong University of Science and Technology, China. He is the first author of 32 publications and the coauthor of 25 publications. His research interests include artificial intelligence, big data processing, information fusion, and microseismic monitoring and inversion.