SCIENCE CHINA
Information Sciences

• LETTER •

March 2024, Vol. 67, Iss. 3, 139101:1–139101:2
https://doi.org/10.1007/s11432-023-3935-8

Scene text recognition via dual character counting-aware visual and semantic modeling network

Ke XIAO[1], Anna ZHU[1*], Brian Kenji IWANA[2] & Cheng-Lin LIU[3,4]

[1]*School of Computer Science and Artificial Intelligence, Wuhan University of Technology, Wuhan* 430070*, China;*
[2]*Human Interface Laboratory, Kyushu University, Fukuoka* 819-0395*, Japan;*
[3]*State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing* 100190*, China;*
[4]*School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing* 100049*, China*

Received 14 July 2023/Revised 6 November 2023/Accepted 26 December 2023/Published online 5 February 2024

Scene text recognition (STR) is drawing increasing attention nowadays due to its wide application in real life. Character counting information, as auxiliary information, has been shown to be effective in boosting text recognition performance. However, most previous methods only utilize it for visual feature enhancement [1, 2]. It can also benefit the semantic models by providing useful global clues for sequence-to-sequence character prediction. Given the previously output characters, the language model (LM) can provide the character probability distribution $p(y_t|y_0,\ldots,y_{t-1})$ for the next character prediction through a statistic rule of reading words. If considering text length n, we can get $p(y_t|y_0,\ldots,y_{t-1}) = \sum_n p(n)p(y_t|y_0,\ldots,y_{t-1},n)$. If n is known as prior, the LM could generate a finer constraint on the sequence, making the character prediction more precise.

In this study, we rethink character counting in STR from a principled viewpoint. The STR model aims to output the predicted word Y: $(y_0, y_1,\ldots,y_t)$ with the maximum probability given image representation V. In our framework, we use the law of total probability to expand $P(Y|V)$ with a predicted text length T. Given ground truth Y^*, the optimization goal of training the model is to maximize (1):

$$
\begin{aligned}
&\log P(Y^*|V) = \log \sum_{T=1}^{\max(T)} P(T|V)P(Y^*|V,T)\\
&\overset{\text{Step1}}{=} \log \sum_{T=1}^{\max(T)} P(T|V)\prod_{t=1}^{T}\frac{P(y_t|v_t,T)P(y_t|Y_{t-1},T)}{P(y_t|T)}\\
&\overset{\text{Step2}}{=} \log P(T^*|V) + \sum_{t=1}^{T^*}\log\frac{P(y_t^*|v_t,T^*)P(y_t^*|Y_{t-1},T^*)}{P(y_t^*|T^*)}\\
&\overset{\text{Step3}}{=} \log P(T^*|V)+\sum_{t=1}^{T^*}\log P(y_t^*|v_t,T^*)P(y_t^*|Y_{t-1},T^*)-\alpha\\
&= -(L_{\rm cc} + L_{\rm rec}),
\end{aligned}
\tag{1}
$$

where $\max(T)$ is the maximum length of given text images, and $Y_{t-1} = \{y_0,\ldots,y_{t-1}\}$. $L_{\rm cc}$ and $L_{\rm rec}$ are the corresponding character counting loss and character prediction loss, respectively. The inference of Step1 is provided in Appendix A. Step2 holds because the correct labels can only be predicted when the correct character count T^* is given. $\alpha = P(y_t^*|T^*)$ in Step3 is a constant and could be ignored in training and test phrases. Because during training, the labels y_t^* and T^* are known. During the test phase, however, the character sequence given length T^* is dependent on the text image, and the probability of sequence depends on a universe LM. If this external LM is available, it is used in recognition only (not in training). Whereas, in most studies of STR, the external LM is not used. Hence, we can ignore it. From the above formulation, we can see the loss for predicting character y_t at time step t has relations with both visual feature v_t, previous linguistic context Y_{t-1}, and text length T^*.

The proposed model. Based on the formulated problem, we propose a character counting aware scene text recognizer. The full model can be found in Appendix B. Given an input image, a backbone network first extracts the local representations V^0. A character counting involved encoder is built upon the Transformer architecture. The positional information and an additional character counting token V_c^0 are added to the encoder. V_c^0 is initialized by averaging features of V^0. It outputs context-enhanced local representations V and a global character counting feature V_c. They are further fed into two independent decoders, a counting-aware semantic decoder, and a vision decoder.

The encoder is a stack of L identical character counting aggregated (CCA) layers, each of which includes a global aggregated attention (GAA) block and a feed-forward neural network (FFN). GAA block is designed to achieve context information for each local representation and also capture a comprehensive global representation. It is implemented by the multi-head self-attention [3] (MHSA). Both features V^l and the global feature V_c^l are fed into MHSA in the l layer and then followed by a residual connection and a layer-normalization (LN) process. Finally, the FFN is used to get the output of the $(l+1)$th layer.

The vision decoder consists of L' identical layers, where

* Corresponding author (email: annazhu@whut.edu.cn)

each contains the multi-head cross attention (MHCA) module to extract positional enhanced visual features and a gate mechanism to transfer the complementary character counting information for character prediction. V^L and V_c^L from the encoder are input into the vision decoder. The positional embedding vector p_t, which is projected from one-hot vectors at each time step t, is passed into an MHCA module as the query. The features of V^L are projected to the key and value. We could obtain $\hat{p}_t^{l+1} = \mathrm{MHCA}(p_t^l, V^L, V^L)$ after cross-attention in the lth layer. For the first layer, p_t^0 is p_t. Then, a gating mechanism, defined as $\sigma = \mathrm{sigmoid}((p_t^l)^{\mathrm{T}} V_c^L)$, is used to control the importance of global information by the query p_t^l and the count information V_c^L. After that, we adaptively fuse the global representation to update the output from MHCA as $\overline{p}_t^{l+1} = \hat{p}_t^{l+1} + \sigma \times V_c^L$. Finally, the refined output of layer l is obtained by using residual connections and LN $p_t^l = \mathrm{LN}(p_t^l + \overline{p}_t^{l+1})$.

The counting-aware semantic decoder is designed in two ways, an long short-term memory (LSTM)-based or a Transformer-based decoder. The first one consists of a double-layer stacked LSTM network with 512-dimensional hidden units and an attention module. The first layer of the stacked LSTM network takes the previously predicted character embedding as input and operates from left to right over the word sequence. The hidden states of the first layer is defined as $h_t^0 = \mathrm{LSTM}(y_{t-1}, h_{t-1}^0)$, where LSTM is the recurrent unit, and y_{t-1} is the decoded output result at time step $t-1$. We set y_0 as a special start token $\langle$start$\rangle$. The global counting information V_c^L is integrated into the second stacked layer to output $h_t^1 = \mathrm{LSTM}(\mathrm{MLP}[h_t^0; V_c^L], h_{t-1}^1)$, where $\mathrm{MLP}[\cdot;\cdot]$ is the integration operation using a multilayer perceptron (MLP). h_t^1 is then input into the consequent attention module as the query feature vector to compute attention $\alpha_i^t = \mathrm{softmax}(h_t^{1\mathrm{T}} V_i^L)$. This attention enables the STR model to learn a language model involving character-level counting that represents output class dependencies. Finally, a glimpse vector G_t aggregates the context-aware visual information V^L for the character prediction during decoding by $G_t = \sum_i \alpha_i^t V_i^L$.

The alternative Transformer-based decoder is composed of stacked L'' identical masked MHSA layers and one MHCA layer. The previously predicted character embeddings are concatenated with the character counting V_c^L and order embeddings, and then input to the masked MHSA layers. The last L''th layer outputs the interacted character embedding $h_{t-1}^{L''}$, and it is further input to MHCA layer as the query to compute attention $\alpha_i^t = \mathrm{softmax}(h_{t-1}^{L''\mathrm{T}} V_i^L)$. The glimpse vector G_t is achieved in the same way as in the LSTM-based decoder.

Finally, a fusion [4] module is conducted on the counting-aware features p_t and G_t for character prediction via an element-wise gate mechanism.

Except for L_{rec} for auto-regressive character prediction and L_{cc} for character counting, we additionally regard connectionist temporal classification (CTC) with character predictions as output labels as a regularizer [5] and stack it onto the encoder for STR modeling. Overall, the total loss function can be expressed by the sum of losses $L_{\mathrm{all}} = L_{\mathrm{rec}} + L_{\mathrm{cc}} + \lambda L_{\mathrm{ctc}}$.

Experimental results. Our experiments mainly include the implementation, ablation studies, comparison experiments, character counting accuracy, and generalization of fancy text images. More details can be found in Appendix C. We report the test accuracy on the regular and irregular datasets with state-of-the-art (SOTA) methods in Table 1. The result demonstrates our method achieves comparable accuracy with SOTA methods. Specifically, our method using an LSTM-based semantic encoder (i.e., $\mathrm{Ours}_{\mathrm{LSTM}}$) gets the best performance on IIIT5K. RF-LN [1] and ACE [2] also utilized the counting information in the vision-based STR model. However, our model is better than them on all test sets since we inject the character counting information in both vision and language models, which could further enhance the recognition accuracy.

Table 1 Accuracy (%) comparison with SOTA STR methods on six standard benchmarks[a)]

Method	Regular			Irregular		
	IIIT5K	SVT	IC13	SVTP	IC15	CUTE
CRNN	78.2	80.9	89.4	–	–	–
NRTR	90.1	91.5	95.8	86.6	79.4	80.9
ACE	82.3	82.6	89.7	70.1	68.9	82.6
RobustScanner	95.3	88.1	94.8	79.5	77.1	90.3
SEED	93.8	89.6	92.8	81.4	80.0	83.6
SCATTER	93.2	90.9	94.1	86.2	82.0	84.8
RF-LN	94.0	87.7	93.5	84.7	76.7	77.8
SRN	94.8	91.5	95.5	85.1	82.7	87.8
ABINet-LV	96.2	93.5	**97.4**	<u>89.3</u>	<u>86.0</u>	89.2
S-GTR	95.8	<u>94.1</u>	96.8	87.9	84.6	92.3
CornerTrans.	95.9	**94.6**	96.4	**91.5**	**86.3**	92.0
SVTR-L	96.3	91.7	<u>97.2</u>	88.4	86.6	**95.1**
CDistNet	96.4	93.5	<u>97.2</u>	88.7	86.0	<u>93.4</u>
$\mathrm{Ours}_{\mathrm{LSTM}}$	**97.4**	93.6	96.8	87.8	84.0	91.3
$\mathrm{Ours}_{\mathrm{Transformer}}$	<u>96.9</u>	93.6	<u>97.2</u>	89.0	84.8	92.4

a) The best results are in bold; the second best results are underlined.

Conclusion. In this work, we study character counting in STR from a new viewpoint, giving a principled framework showing that the counting information is involved in both visual decoding and semantic decoding. Based on the principled framework, we propose a novel scene text recognizer with a dual character counting-aware visual and semantic modeling network, where the counting information is fused in both vision and language branches. Experimental results demonstrate the effectiveness of our model.

Acknowledgements This work was supported by Open Project Program of the National Laboratory of Pattern Recognition (NLPR) (Grant No. 202200049).

Supporting information Appendixes A–C. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Jiang H, Xu Y L, Zhan Z, et al. Reciprocal feature learning via explicit and implicit tasks in scene text recognition. In: Proceedings of the 16th International Conference on Document Analysis and Recognition, 2021. 287–303
2 Xie Z, Huang Y, Zhu Y, et al. Aggregation cross-entropy for sequence recognition. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 6531–6540
3 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017. 6000–6010
4 Yue X Y, Kuang Z H, Lin C H, et al. RobustScanner: dynamically enhancing positional clues for robust text recognition. In: Proceedings of European Conference on Computer Vision, 2020. 135–151
5 Zhang B, Haddow B, Sennrich R. Revisiting end-to-end speech-to-text translation from scratch. In: Proceedings of International Conference on Machine Learning, 2022. 26193–26205