




Handwritten Mathematical Expression Recognition via GCAttention-Based Encoder and Bidirectional Mutual Learning Transformer

Xiaoxiang Han¹ , Qiaohong Liu² , Ziqi Han¹, Yuanjie Lin¹, and Naiyue Xu¹

¹ School of Health Science and Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

² School of Medical Instruments, Shanghai University of Medicine and Health Sciences, Shanghai 201318, China

769842320@qq.com

Abstract. Recognition of handwritten mathematical expressions to \LaTeX is an image-to-sequence task. Recent research has shown that encoder-decoder models are well suited for this challenge. Many innovative models based on this structure have been proposed, especially on the decoder. Such as attention mechanism and bidirectional mutual learning are used in the decoder. And our model also improves the encoder. We use the multi-scale fusion DenseNet as the encoder and add Global Context Attention. This attention mechanism combines the advantages of force-spatial attention and channel attention. The feature maps of the two scales output by the encoder are used as inputs to the two decoder branches. The decoder uses a two-way mutual learning Transformer, which can understand high-level semantic and contextual information, and can handle long sequences of information well. In order to save memory, the two decoder branches use a set of parameters, and the last two branches are distilled and learned from each other. In this way, not only the bidirectional decoders can learn from each other, but also the two decoder branches can learn from each other, which increases the robustness of the model. Our model achieves 56.80%, 53.34% and 54.62% accuracy on CROHME2014, 2016 and 2019, respectively, and 66.22% accuracy on our own constructed dataset HME100k.

Keywords: Mathematical expression · Handwriting recognition · Multi scale · Global context attention · Transformer · Bidirection mutual learning

1 Introduction

Mathematical formula recognition is an important part of OCR, however, it was not introduced by Anderson in his PhD thesis until the 1960s [1]. He proposed

This work is partially supported by the National Natural Science Foundation of China (Nos. 61801288).

a method of using syntax as a standard segmentation and using a top-down analysis method to identify mathematical formulas. Traditional methods of converting images to \LaTeX rely on specially designed syntax [16]. However, these grammars require a lot of prior knowledge to define the structure of mathematical expressions, the positional relationship of symbols and the corresponding parsing algorithms in advance, so that complex mathematical expressions cannot be recognized.

Compared with the traditional OCR problem, handwritten mathematical expression recognition is a more complex two-dimensional handwriting recognition problem. Its internal complex two-dimensional spatial structure makes it difficult to analyze, and the traditional method has a poor recognition effect. With the advancement of deep learning, encoder-decoder models have shown fairly effective performance on various tasks such as scene text recognition [4] and image captioning [26]. It also achieves significant performance improvements during HMER processing [5]. Zhang et al. [29] introduced the attention mechanism to undoubtedly increase the accuracy. They propose the Watch, Attend, and Parse (WAP) method, which employs a deep fully convolutional network (FCN) to encode handwritten images and a gated recurrent unit (GRU) with attention mechanism as the decoder to generate serial output. Zhang et al. developed DWAP-MSA [27] to try to use a multi-scale feature encoding to identify symbols of different sizes in handwritten mathematical expressions, so we borrowed this method in model design. \LaTeX is a markup language designed by humans and therefore has a cleaner and more defined syntactic structure. For example, the two parentheses “(” and “)” must be paired. When dealing with long \LaTeX sequences, as the distance increases, the dependency information captured between the currency symbol and the previous symbol becomes weaker and weaker, and it is difficult for RNN-based models to capture the relationship between two distant two brackets. And a major limitation of overlay attention is that it only uses historical alignment information without considering future information. Most models in the past only decode left-to-right, ignoring information on the right, so they may not take full advantage of long-range correlations and the grammar specification of mathematical expressions [2, 31]. Zhao et al. [31] designed a simple bidirectional Transformer decoder called BTTR, but there is no explicit supervision information for BTTR to learn from the opposite direction, and its decoders in both directions do not learn from each other, which limits its bidirectional learning ability.

2 Related Work

2.1 Image-to-Markup

Deng et al. [6] defined the image-to-markup problem as: transforming a rendered source image into a target rendering markup that fully describes its content and layout. The source, x , consists of an image. The target, y , consists of a sequence of tokens y_1, y_2, \dots, y_T , where T is the length of the output and each y is a token in the markup language.

2.2 CNN

In the past ten years, convolutional neural networks have continued to make efforts in many directions, and have made breakthroughs in speech recognition, face recognition, general object recognition, motion analysis, natural language processing and even brain wave analysis.

In 1962, Hubel and Wiesel's experiments [14] on cats found that the cat's visual cortex processes information in a hierarchical structure, that is to say, it extracts information layer by layer. The simplest information is extracted at the top, and then continuously. For simple information extraction, high-level abstract information is gradually obtained. Yann LeCun [19] was the first to use Convolutional Neural Network (CNN) for handwritten digit recognition and has maintained its dominance in the problem. In 2012, Alexnet [15] introduced a new deep structure and dropout method, which increased the error rate from more than 25% to 15%. It subverts the field of image recognition. In 2014, Karen et al. [20] used CNN to explore the relationship between the depth of the convolutional neural network and its performance. By repeatedly stacking 3×3 small convolution kernels and 2×2 maximum pooling layers, VGGNet successfully constructed 16 19-layer deep convolutional neural network. In the same year, Szegedy et al. [21] proposed GoogLeNet. It does not rely solely on deepening the network structure to improve network performance, but at the same time deepening the network (22 layers), it has made innovations in the network structure. It introduces the Inception structure to replace the traditional operation of simple convolution and activation. In 2015, He et al. [10] proposed ResNet, which made great innovations in the network structure and introduced the residual network structure. With this residual network structure, very deep networks can be designed, providing feasibility for advanced semantic feature extraction and classification. In 2017, Huang et al. [13] proposed DenseNet, which established the connection relationship between different layers, made full use of features, and further alleviated the problem of gradient disappearance. Moreover, its network is narrower and has fewer parameters, which effectively suppresses overfitting and reduces the amount of computation. The Dense block proposed by it solves the problem that the size of each input is different, and it is estimated that there is no need to force the input to be modified into a fixed shape. Due to the many advantages of this network, this paper uses DenseNet with the addition of the GCAttention module as the backbone network for feature extraction, that is, the encoder of our network.

2.3 Global Contextual Attention

The goal of capturing long-range dependencies is a global understanding of the visual scene, which is effective for many computer vision tasks, such as image classification, video classification, object detection, semantic segmentation, etc. Gao et al. [3] proposed a global contextual attention module, which is lightweight and can fully utilize global contextual information. The global context block can

be represented as

$$y_i = x_i + w_{v2}ReLU(LN(w_{v1} \sum_{\forall j} \frac{e^{w_k x_j}}{\sum_{\forall m} e^{w_k x_j}} x_j)) \tag{1}$$

2.4 Transformer

Recently transformers [23] have shown good performance on a variety of tasks [7, 8], it can avoid recursion, in order to allow parallel computing, and reduce performance degradation due to long-term dependencies. Since the hidden layer nodes of RNNs at time T depend on forward input and intermediate calculation results, this feature limits the parallel computing capability of RNNs.

The core of Transformer is Scaled Dot-Product Attention, which solves the problem that because the network inputs multiple vectors of different sizes, and there may be a certain relationship between the vectors, these relationships cannot be fully utilized. Its formula is as follows:

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

2.5 Mutual Learning

Zhang et al. [30] proposed a deep mutual learning (DML) strategy, in which a group of student networks, learns from and mentor each other throughout the training process. Different from the static pre-defined one-way transition path between teacher and student in distillation model.

Guo et al. [9] proposed an efficient online knowledge extraction method through collaborative learning, called KDCL, which can continuously improve the generalization ability of deep neural networks (DNNs) with different learning capabilities. Different from the two-stage knowledge distillation method, KDCL treats all DNNs as “students” and trains them collaboratively in one stage (knowledge is transferred among any students during collaborative training), thus achieving parallel computing, fast Computing and attractive generalization capabilities.

3 Methodology

3.1 Encoder

Since the size of images of handwritten mathematical expressions is usually random size, a model called Densely Connected Convolutional Network (DenseNet) [13] is used in our encoder. DenseNet is a type of FCN that connects all networks in a feed-forward fashion and enhances feature propagation and reuse by ensuring the maximum information flow between layers in the network, so FCN can handle images of any size.

There are many symbols of different scales in handwritten mathematical expressions, and using pure DenseNet will lose some details. The multi-scale dense network proposed by Gao et al. [12] utilizes information at all scales, which is obviously very expensive. This paper uses a DenseNet with a three-layer structure, and only the last layer of it is upsampled and the output of the second layer is fused. Finally, the encoder outputs the feature degrees of these two scales. This not only obtains multi-scale information, but also saves computational expenses. At the same time, we add a GCAttention layer after each pooling layer to make the network pay more attention to useful features, as shown in Fig. 1.

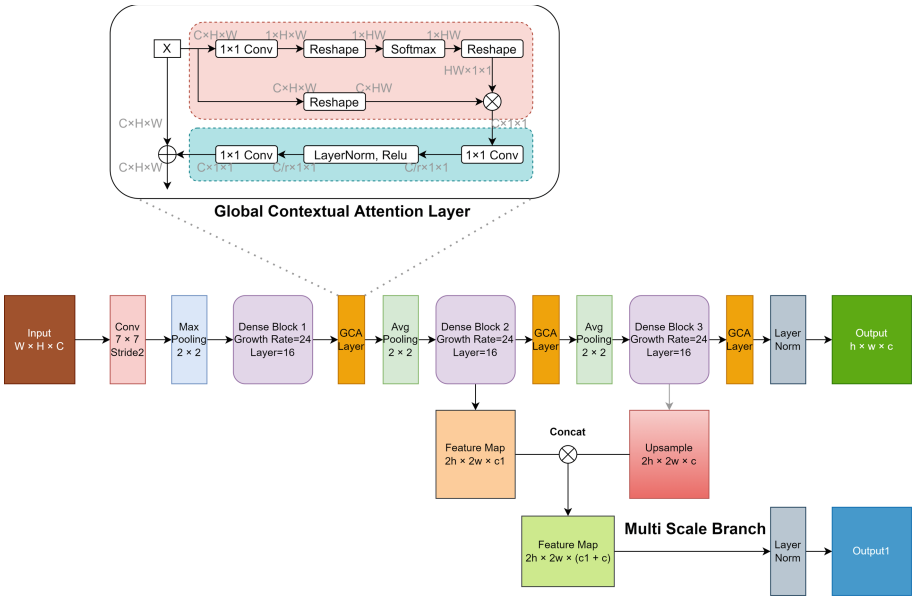


Fig. 1. The structure of the global contextual attention layer and the multi-scale encoder.

3.2 Decoder

We use the standard Transformer [23] as the decoder, and the feature maps output by the encoder are embedded into the decoder respectively. And these branches share a Transformer model to reduce the amount of parameters and computing power as shown in Fig. 2.

To achieve bidirectional training, we add $\langle sos \rangle$ and $\langle eos \rangle$ to the $\mathbb{L}\mathbb{T}\mathbb{E}\mathbb{X}$ sequence as start and end symbols, respectively. For example, a target $\mathbb{L}\mathbb{T}\mathbb{E}\mathbb{X}$ sequence

$$Y = \{Y_1, Y_2, \dots, Y_T\} \quad (3)$$

of length T , which is represented as

$$Y_{l2r} = \{\langle sos \rangle, Y_1, Y_2, \dots, Y_T, \langle eos \rangle\} \quad (4)$$

from left to right (L2R), is represented as

$$Y_{r2l} = \{\langle eos \rangle, Y_T, Y_{T-1}, \dots, Y_1, \langle sos \rangle\} \quad (5)$$

from right to left (R2L).

To quantify the difference in prediction distribution between the two directions and between the two branches, we introduce the Kullback-Leibler (KL) loss [11]. After optimization, this loss can minimize the distance of the probability distribution between different branches. The KL distance in both directions is calculated as follows:

$$\sigma(Z_{i,k}^{l2r}, S) = \frac{\exp(Z_{i,k}^{l2r}/S)}{\sum_{j=1}^K \exp(Z_{i,k}^{l2r}/S)} \quad (6)$$

$$L_{KL} = S^2 \sum_{i=1}^T \sum_{j=1}^K \sigma(Z_{i,k}^{l2r}, S) \log \frac{\sigma(Z_{i,k}^{l2r}, S)}{\sigma(Z_{T+1-i,j}^{r2l}, S)} \quad (7)$$

where σ represents the soft probability of one direction.

3.3 Positional Encoding

Since the Transformer model itself does not have any sense of position for each input vector, we do positional embeddings for both image and word vectors, which can effectively help the model identify areas that need attention. For the positional embedding of word vectors, we directly adopt the method of Transformer's original research [23]. For the positional embedding of word vectors, we directly adopt the method of Transformer's original research. It is defined as follows:

$$PE(pos, 2i) = \sin(pos/10000^{2i/d_{model}}) \quad (8)$$

$$PE(pos, 2i + 1) = \cos(pos/10000^{2i/d_{model}}) \quad (9)$$

where pos is the position and i is the dimension.

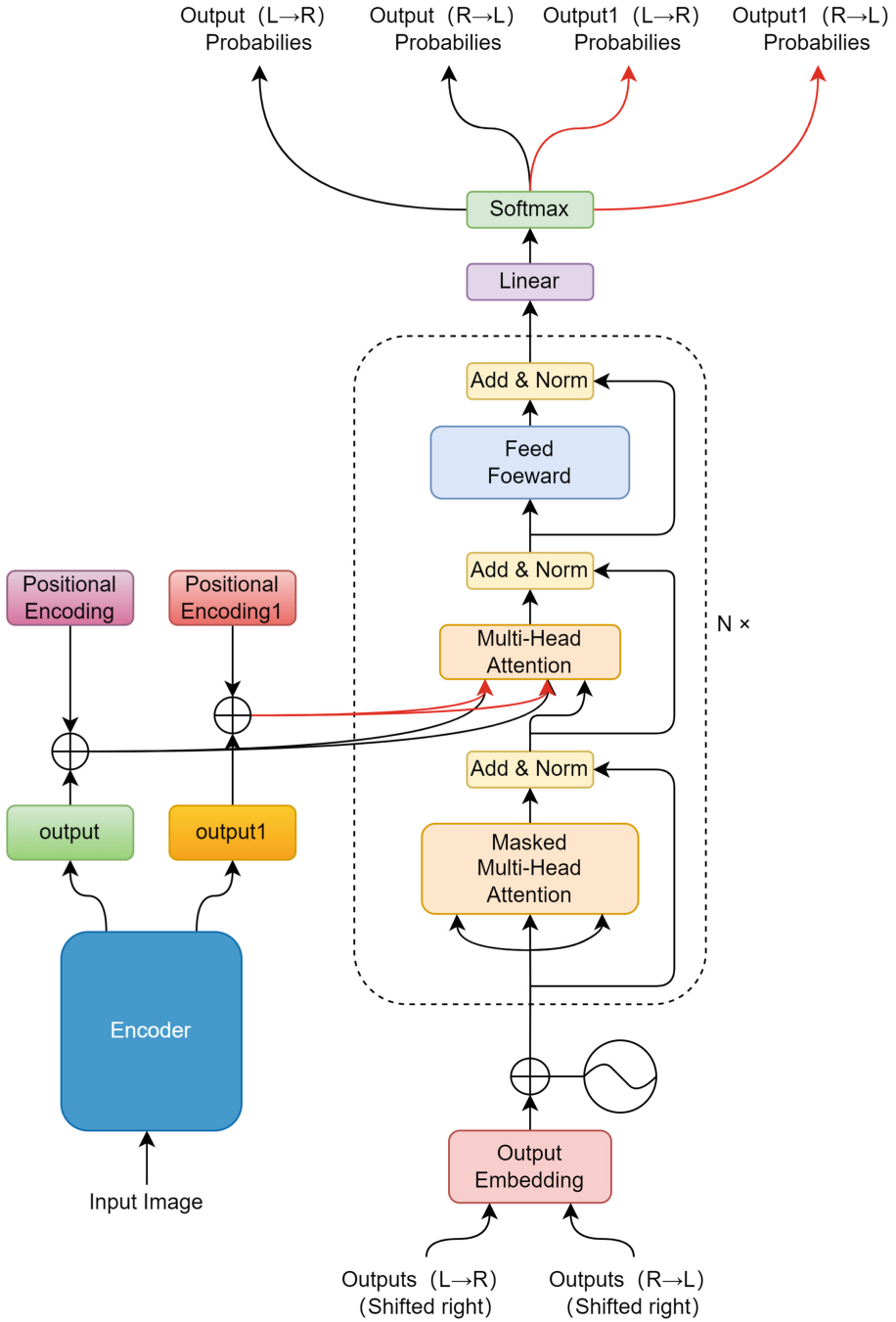


Fig. 2. The structure of the decoder.

Zhao et al. [31] describe a 2D normalized positional encoding for representing image positional features. The sinusoidal positional encoding $P_{pos,d/2}^W$ is first computed in both dimensions and then concatenated. Given a two-dimensional position tuple (x, y) and the same dimension d as the word position encoding, the image position encoding vector $P_{x,y,d}^I$ is represented as:

$$\bar{x} = \frac{x}{H}, \bar{y} = \frac{y}{W} \quad (10)$$

$$P_{pos,d/2}^W = [P_{\bar{x},d/2}^W; P_{\bar{y},d/2}^W] \quad (11)$$

where H and W are height and width of input images. Finally, the weighted summation of the cross-entropy loss and the KL distance of each output is performed.

4 Experiments

4.1 Datasets

We use Competition on Recognition of Online Handwritten Mathematical Expressions (CROHME2014) as our training set. It has 111 types of mathematical symbols and 8836 handwritten mathematical expressions, including numbers, almost all common operators. Then we take three public test datasets, as test sets, they are CROHME 2014, 2016 and 2019 with 986, 1147 and 1199 expressions, respectively.

In addition, we trained and tested on a dataset called HME10k. This dataset collects handwritten mathematical expressions in real handwriting scenarios from students, which is more diverse and richer than the CROHME dataset. But because it is a photo of a real scene, it is inevitable that there are many blurry images that are unrecognizable by humans, which is not helpful for our training, so we remove them. We divided this dataset into two parts, 80,000 training sets and 20,000 test sets.

4.2 Comparison with Prior Works

We compare our method with the previous state-of-the-art as shown in Table 1. All the methods shown in the table only use the 8836 training samples officially provided by CROHME, and do not use data augmentation to ensure the fairness of the performance comparison. These methods include PAL (Wu et al. [24]), PAL-v2 (Wu et al. [25]), WAP (Zhang et al. [29]), PGS (Le et al. [18]), DWAP (WAP with DenseNet as encoder), DWAP-MSA (DWAP with multi-scale attention) (Zhang et al. [27]), DWAP-TD (DWAP with tree decoder) (Zhang et al. [28]), DLA (Le [17]), WS WAP (weakly supervised WAP) (Truong et al. [22]) and BTTR (Zhao et al. [31]).

The results show that our method has a significant improvement in accuracy on CROHME 2014, which is 2.84% higher than BTTR, and at the same time, the accuracy on ≤ 1 and ≤ 2 is also improved by 5.25% and 6.57%, respectively.

Table 1. Comparison with prior works (in %). The results in the table are cited from their corresponding papers.

Dataset	Methods	ExpRate	≤ 1 error	≤ 2 error
2014	PAL	39.66	56.80	68.51
	WAP	46.55	61.16	65.21
	PGS	48.78	66.13	73.94
	PAL-v2	48.88	64.50	69.78
	DWAP-TD	49.10	64.20	67.8
	DLA	49.85	–	–
	DWAP	50.60	68.05	71.56
	DWAP-MSA	52.80	68.10	72.00
	WS WAP	53.65	–	–
	BTTR	53.96	66.02	70.28
	Ours	56.80	71.27	76.85
2016	PGS	36.27	–	–
	TOKYO	43.94	50.91	53.70
	WAP	44.55	57.10	61.55
	DWAP-TD	48.50	62.30	65.30
	DLA	47.34	–	–
	DWAP	47.43	60.21	63.35
	PAL-v2	49.61	64.08	70.27
	DWAP-MSA	50.10	63.80	67.40
	WS WAP	51.96	64.34	70.10
	BTTR	52.31	63.90	68.61
	Ours	53.34	67.56	74.19
2019	DWAP	47.70	59.50	63.30
	DWAP-TD	51.40	66.10	69.10
	BTTR	52.96	65.97	69.14
	Ours	54.62	68.97	74.64
HME100K	Ours	66.22	77.81	81.20

Table 2. Ablation study on the CROHME 2014 test sets (in %).

Mutual learning	GCAttention	Multi-scale	ExpRate
✗	✗	✗	48.36
✓	✗	✗	53.96
✓	✓	✗	55.22
✓	✓	✓	56.80

Our method also improves ExpRate by 1.03% and 1.66% compared to BTTR on CROHME 2016 and 2019, respectively., which proves the effectiveness of our model.

4.3 Ablation Study

In Table 2, the mutual learning in the first column indicates whether multi-scale and bidirectional mutual learning is used, the multi-scale in the second column indicates whether multi-scale is used in the encoder, and the GCAAttention in the third column indicates whether the encoder is added Global contextual attention layer.

First of all, we found that mutual learning has a great impact on the model. Without mutual learning, multi-scale and GCAAttention, the accuracy rate is only 48.36%, which is 8.44% different from the highest accuracy rate. Second, we found that under the combined effect of multi-scale and GCAAttention, our model also achieved certain results, which improved the accuracy by 2.84% compared to not using them.

4.4 The Program with GUI

Finally, we made a program (Fig. 3) with a GUI for the model trained on the dataset we built, where the user can use the mouse to write a mathematical expression on the drawing board, and then click the *Recognize* button to get the L^AT_EX expression.

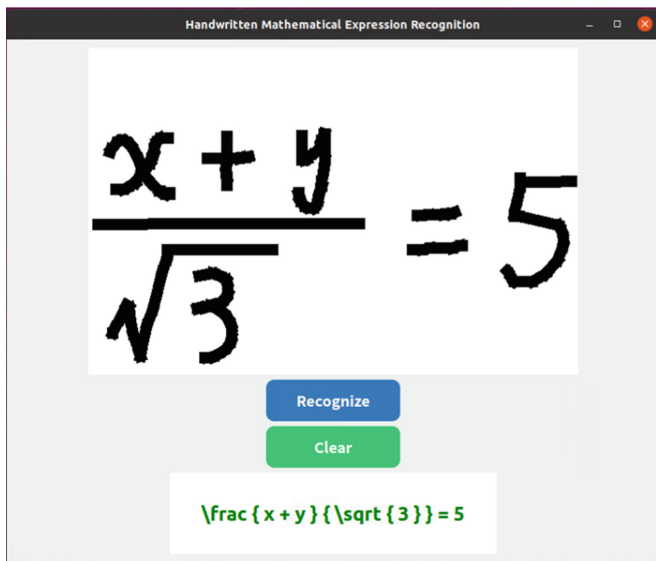


Fig. 3. The program with GUI for HMER.

5 Conclusion

In this paper, we improve the performance of models for recognizing handwritten mathematical expressions by introducing a global contextual attention mechanism and multi-branch mutual learning. We built a dataset ourselves and trained this model on it with good results. However, handwritten mathematical expressions are very complex, and each person's handwriting style is different. Whether it is a single character or the entire expression structure, there are certain differences in what everyone writes. Therefore, in the future we will collect more and more complex data to train our model and improve its robustness.

References

1. Anderson, R.H.: Syntax-directed recognition of hand-printed two-dimensional mathematics. In: Symposium on Interactive Systems for Experimental Applied Mathematics: Proceedings of the Association for Computing Machinery Inc., Symposium, pp. 436–459 (1967)
2. Bian, X., Qin, B., Xin, X., Li, J., Su, X., Wang, Y.: Handwritten mathematical expression recognition via attention aggregation based bi-directional mutual learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 113–121 (2022)
3. Cao, Y., Xu, J., Lin, S., Wei, F., Hu, H.: GCNet: non-local networks meet squeeze-excitation networks and beyond. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
4. Cheng, Z., Bai, F., Xu, Y., Zheng, G., Pu, S., Zhou, S.: Focusing attention: towards accurate text recognition in natural images. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5076–5084 (2017)
5. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint [arXiv:1412.3555](https://arxiv.org/abs/1412.3555) (2014)
6. Deng, Y., Kanervisto, A., Ling, J., Rush, A.M.: Image-to-markup generation with coarse-to-fine attention. In: International Conference on Machine Learning, pp. 980–989. PMLR (2017)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
8. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. arXiv preprint [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2020)
9. Guo, Q., et al.: Online knowledge distillation via collaborative learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11020–11029 (2020)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
11. Hinton, G., Vinyals, O., Dean, J., et al.: Distilling the knowledge in a neural network, **2**(7). arXiv preprint [arXiv:1503.02531](https://arxiv.org/abs/1503.02531) (2015)
12. Huang, G., Chen, D., Li, T., Wu, F., Van Der Maaten, L., Weinberger, K.Q.: Multi-scale dense networks for resource efficient image classification. arXiv preprint [arXiv:1703.09844](https://arxiv.org/abs/1703.09844) (2017)

13. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
14. Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *J. Physiol.* **160**(1), 106 (1962)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25 (2012)
16. Laviotte, S., Pottier, L.: Mathematical formula recognition using graph grammar. In: Document Recognition V, vol. 3305, pp. 44–52. SPIE (1998)
17. Le, A.D.: Recognizing handwritten mathematical expressions via paired dual loss attention network and printed mathematical expressions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 566–567 (2020)
18. Le, A.D., Indurkha, B., Nakagawa, M.: Pattern generation strategies for improving recognition of handwritten mathematical expressions. *Pattern Recogn. Lett.* **128**, 255–262 (2019)
19. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
20. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
21. Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
22. Truong, T.N., Nguyen, C.T., Phan, K.M., Nakagawa, M.: Improvement of end-to-end offline handwritten mathematical expression recognition by weakly supervised learning. In: 2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 181–186. IEEE (2020)
23. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
24. Wu, J.-W., Yin, F., Zhang, Y.-M., Zhang, X.-Y., Liu, C.-L.: Image-to-markup generation via paired adversarial learning. In: Berlingerio, M., Bonchi, F., Gärtner, T., Hurley, N., Ifrim, G. (eds.) ECML PKDD 2018. LNCS (LNAI), vol. 11051, pp. 18–34. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-10925-7_2
25. Bian, X., Qin, B., Xin, X., Li, J., Su, X., Wang, Y.: Handwritten mathematical expression recognition via attention aggregation based bi-directional mutual learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 113–121 (2022)
26. Xu, K., et al.: Show, attend and tell: neural image caption generation with visual attention. In: International Conference on Machine Learning, pp. 2048–2057. PMLR (2015)
27. Zhang, J., Du, J., Dai, L.: Multi-scale attention with dense encoder for handwritten mathematical expression recognition. In: 2018 24th International Conference on Pattern Recognition (ICPR), pp. 2245–2250. IEEE (2018)
28. Zhang, J., Du, J., Yang, Y., Song, Y.Z., Wei, S., Dai, L.: A tree-structured decoder for image-to-markup generation. In: International Conference on Machine Learning, pp. 11076–11085. PMLR (2020)
29. Zhang, J., et al.: Watch, attend and parse: an end-to-end neural network based approach to handwritten mathematical expression recognition. *Pattern Recogn.* **71**, 196–206 (2017)

30. Zhang, Y., Xiang, T., Hospedales, T.M., Lu, H.: Deep mutual learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4320–4328 (2018)
31. Zhao, W., Gao, L., Yan, Z., Peng, S., Du, L., Zhang, Z.: Handwritten mathematical expression recognition with bidirectionally trained transformer. In: Lladós, J., Lopresti, D., Uchida, S. (eds.) ICDAR 2021. LNCS, vol. 12822, pp. 570–584. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86331-9_37