Mathematical Biosciences and Engineering

MBE, 19(5): 4976–4990.
DOI: 10.3934/mbe.2022233
Received: 17 December 2021
Revised: 14 February 2022
Accepted: 27 February 2022
Published: 15 March 2022

http://www.aimspress.com/journal/MBE

Research article

A retrieval and ranking method of mathematical documents based on CA-YOLOv5 and HFS

Xinpeng Xu[1,2], Xuedong Tian[1,2,*] and Fang Yang[1,2]

[1] School of Cyber Security and Computer, Hebei University, Baoding 071002, China
[2] Institute of Intelligent Image and Document Information Processing, Hebei University, Baoding 071002, China

* **Correspondence:** Email: xuedong_tian@126.com.

Abstract: In a retrieval system for mathematical documents based on mathematical expressions, the input and matching of mathematical expressions are key steps that affect the system's usability, accessibility and efficiency because of their special attributes. Therefore, this paper mainly focuses on improving the input efficiency and matching accuracy of mathematical expressions. This paper proposes a method for retrieval and ranking of mathematical documents based on CA-YOLOv5 and HFS (hesitation fuzzy set) by utilizing the advantages of CA (coordinate attention) model and YOLOv5 in target detection and the superiority of HFS in multiattribute decision-making. By embedding the CA model into the YOLOv5 network, the mathematical expressions in layout images are extracted and recognized to form mathematical query expressions. These expressions are then analyzed to obtain similarity evaluation features and matched with the candidate mathematical expressions indexed with the same features in a library of mathematical documents by employing the HFS as the similarity evaluation measure. Experiments were performed based on the TFD-ICDAR2019v2 dataset and the NTCIR dataset. The F1-score of the mathematical expression detection result was 76.54%, the MAP (mean average precision) of the mathematical documents retrieval result was 71.73%, and the average nDCG of mathematical documents ranking was 80.89%.

Keywords: mathematical document retrieval; mathematical expressions; YOLOv5; HFS; CA

1. Introduction

Mathematical documents are important media that store information about science and technology, and mathematical expressions play an indispensable role in expressing the information in such documents. Using mathematical expressions to quickly and effectively find relevant mathematical documents is an important way for scientific and technological workers to obtain necessary information. At present, mathematical document retrieval based on mathematical expressions is still faced with two key challenges in practical applications: the fast and easy input of mathematical query expressions and mathematical expression matching considering variations in syntax and semantics.

Research on the detection of mathematical expressions in document images has achieved valuable results [1−5]. Gao et al. [6] used the LIBSVM algorithm to classify text lines as independent expression lines and non-independent expression lines and then detected the embedded expressions and independent expressions. To address the misclassification of text lines, the team proposed a learning-based merging strategy to merge incorrectly split text lines on the basis of the projected contour cutting results. In the merging strategy, they used the layout of text lines, textual features and the features in consecutive lines to detect lines that were misclassified [7]. Lin et al. [8] combined four novel features of PDF documents and proposed a method to directly use data extracted from PDF documents to detect mathematical expressions. Gao et al. [9] proposed a solution based on AlexNet and Bi-LSTM for the detection and recognition of mathematical expressions in PDF document images. Phong et al. [10] proposed a mathematical variable classification method based on CNNs (AlexNet and ResNet-50) for the detection of mathematical variables in embedded expressions. Then, the team proposed a unified mathematical expression detection system [11] to detect mathematical expressions in document images. First, this method was used for layout analysis involving entire document images, and it improved the accuracy of text line segmentation and word segmentation. Then, the features extracted by FFT and CNN (AlexNet and ResNet-18) models were used to detect independent expressions and embedded expressions in document images, respectively. This method combined manually extracted features and deep learning features for mathematical expression detection, and the detection accuracy was greatly improved compared with that of previous methods.

In studies of scientific document retrieval based on mathematical expressions, a variety of results have been obtained [12−18]. Considering the importance of the contextual features of mathematical expressions, Wang and Tian [19] proposed a method based on BERT to calculate the contextual similarity of mathematical expressions for scientific document retrieval. First, NTCIR data were preprocessed, and the correspondence between mathematical expressions and scientific documents was saved. Then, BERT was used to calculate the context similarity of the results returned by the mathematical expression similarity calculation module and finally output the retrieval results of scientific documents based on the similarity scores. This method was evaluated based on the NTCIR dataset with Chinese scientific documents added, and it performed reasonably well. Hussain and Khoja [20] proposed a method for retrieving scientific documents based on the semantic information from mathematical expressions to optimize the sorting of scientific documents. The variables, constants and operators in the expressions with unified symbols were replaced, and weights were assigned to the semantic subtrees of the expressions to enhance the retrieval results. This method was evaluated based on the NTCIR-12 and arXiv corpora, and for the top-5 documents, the precision of Wikipedia formula queries reached 47% and 44%, respectively. Xu et al. [21], in an effort to overcome the shortcomings of similarity calculations using only text information, proposed a method to calculate

the similarity of scientific documents by combining text information and mathematical expression information. This method used the formula coverage in document pairs to measure formula similarity and the distances between feature words in the documents to measure text similarity. Finally, text similarity and formula similarity were used to calculate the similarity of the scientific documents. The experimental results showed that compared with the traditional vector space method, this method improved the precision of document similarity calculations and was more suitable for cross-language document similarity calculations. Pathak et al. [22] proposed a method to retrieve mathematical expressions based on the context of scientific documents. First, "context-formula" pairs were extracted by using a pattern-based method and stored in a knowledge base. Then, Apache Lucene was used to create an inverted index for the context in the knowledge base and to coordinate with the index and the knowledge base to obtain the retrieval results for expressions related to the text query. This method used the context of mathematical expressions to aid in retrieval, which could improve the precision of matching to a certain extent. By combining the methods of natural language processing (NLP) and mathematical language processing (MLP), Scharpf et al. [23] realized the classification and clustering of documents containing mathematical content, thus laying a foundation for the efficient retrieval of mathematical documents. The study aims to assess the impact of choice and combined encoding of natural and mathematical languages on the classification and clustering of documents containing mathematical content. For the coarse-grained classification of the primary MSC subject number (pMSCn), Schubotz et al. [24] proposed a method combined with machine learning to automate this process. The method reduces the effort while maintaining classification accuracy, contributing to research in mathematical documents retrieval.

This paper proposes a mathematical document retrieval and ranking method based on CA-YOLOv5 and HFS [25] to improve the performance of mathematical document retrieval systems by combining the ability of CA-YOLOv5 to quickly and accurately detect targets and HFS multiattribute decision-making. The main contributions of this research are as follows:

(1) In the proposed mathematical query interface, the automatic input of mathematical query expressions is achieved by using YOLOv5 [26] for the mathematical expression detection task and utilizing CA [27] model to obtain the target location information.

(2) In the mathematical matching stage, FDS is used to normalize mathematical expressions, and HFS algorithm is introduced to calculate the similarity between pairs of mathematical expressions, thus enabling our method to adapt to variable forms of mathematical expressions and improving the performance of mathematical document retrieval.

The remainder of this paper is organized as follows. A system overview is given in Section 2. In Section 3, the mathematical query interface module is proposed. In Section 4, the mathematical matching module is introduced. In Section 5, we present the experimental results and discuss them. Finally, conclusions are summarized in Section 6.

2. Overview of the mathematical document retrieval system based on CA-YOLOv5 and HFS

The workflow of the mathematical document retrieval system is shown in Figure 1. The mathematical query interface uses the pretrained CA-YOLOv5 to automatically detect and recognize mathematical expressions in layout images. The mathematical matching module parses each symbol in a mathematical query expression into an n-tuple attribute feature by using the FDS algorithm. Then, a mathematical query feature index is established to match the mathematical expressions in the dataset.

Finally, the results of mathematical document retrieval and sorting are obtained.

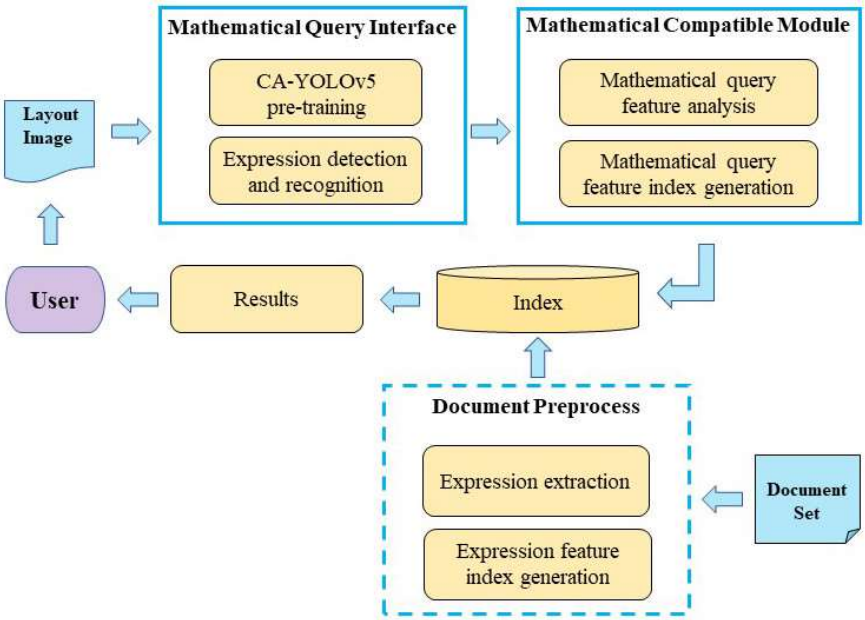

Figure 1. Flow chart of the mathematical document retrieval system.

3. Mathematical query interface

The structure of the mathematical query interface is shown in Figure 2. YOLOv5 is an end-to-end object detection network based on the YOLO [28] series of neural networks. The CA model is used to capture the positions of mathematical expressions in layout images, and by embedding the module into the YOLOv5 network architecture, the ability of YOLOv5 to extract the positional features of mathematical expressions is enhanced.

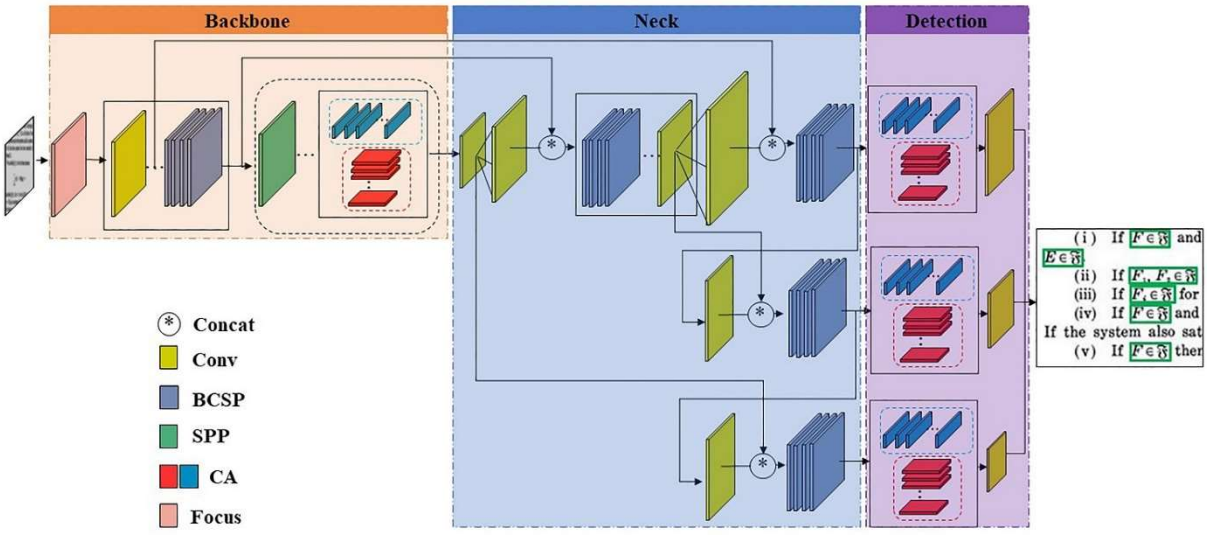

Figure 2. The structure of the mathematical query interface.

3.1. Backbone

The backbone of the system mainly includes five modules: Focus, Conv, BCSP, SPP, and CA. Among them, the CA module aggregates the positional features of mathematical expressions in the horizontal and vertical spatial directions so that the attention block captures long-distance dependencies in one direction while retaining the positional information in the other direction. The module structure is shown in Figure 3.

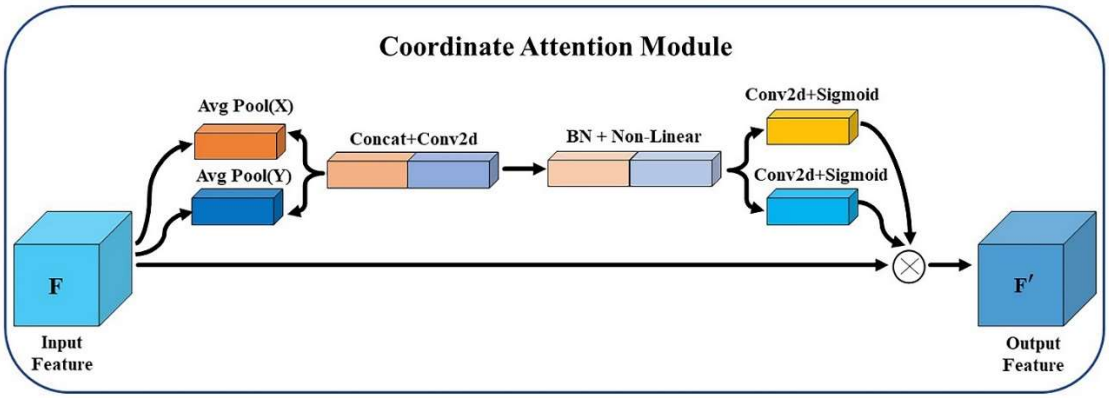

Figure 3. The structure of the CA module.

First, based on the feature matrix $\mathrm{F}\epsilon\mathbb{R}^{(C\times H\times W)}$ of the input document image, each channel is calculated in the horizontal and vertical directions by using two pooling kernels with spatial ranges $(H, 1)$ and $(1, W)$, as shown in Eqs (1) and (2).

$$\mathrm{z}^h(h)=\frac{1}{W}\sum_{0\leqslant i<W}x(h,i) \tag{1}$$

$$\mathrm{z}^w(w)=\frac{1}{H}\sum_{0\leqslant j<H}x(j,w) \tag{2}$$

Then, z^h and z^w are concatenated in the spatial dimension and transformed through 1×1 convolution, as shown in Eq (3), where $Concat(\mathrm{z}^h,\mathrm{z}^w)$ represents the concatenation of the features z^h and z^w in the spatial dimension, $Conv2d$ is the 1×1 convolutional layer, and Θ is the nonlinear activation function.

$$\mathrm{f}=\Theta(Conv2d(Concat(\mathrm{z}^h,\mathrm{z}^w))) \tag{3}$$

And then, f is split into two separate tensors $\mathrm{f}^h\epsilon\mathbb{R}^{C/r\times H}$ and $\mathrm{f}^w\epsilon\mathbb{R}^{C/r\times W}$ along the spatial dimension, r is the reduction ratio for controlling the block size. The 1×1 convolutional layer $Conv2d$ is used to separately transform f^h and f^w to tensors with the same channel number to the input F. As shown in Eqs (4) and (5).

$$\mathrm{g}^h=Sigmoid(Conv2d(\mathrm{f}^h)) \tag{4}$$

$$\mathrm{g}^w=Sigmoid(Conv2d(\mathrm{f}^w)) \tag{5}$$

Finally, the output is shown in Eq (6), where $\mathrm{F}'\epsilon\mathbb{R}^{C\times H\times W}$ is the output, F is the input feature, $\otimes$ represents elementwise multiplication.

$$F' = F \otimes g^h \otimes g^w \tag{6}$$

3.2. Neck

The neck part of the system mainly uses a PANet structure. Through bottom-up path augmentation, accurate localization signals in lower layers are used to enhance the entire feature hierarchy, thereby shortening the information path between lower layers and topmost features and enhancing the flow of pixel information in mathematical expressions.

3.3. Detection

The detection module includes the CA module and the Conv module, which are mainly used to output the position of the mathematical expression. In order to better locate mathematical expressions of different sizes, the CA module and Conv module are added to detect the corresponding mathematical expressions adaptively.

4. Mathematical matching module

The module workflow is shown in Figure 4 and mainly includes two parts: mathematical expression analysis and similarity calculation. First, an expression entered into the mathematical query interface is parsed by FDS and stored in the database in the form of five tuple attributes. Then, the HFS is used to calculate the similarity between mathematical expressions, and the relevant mathematical documents are matched and sorted according to the similarity scores.

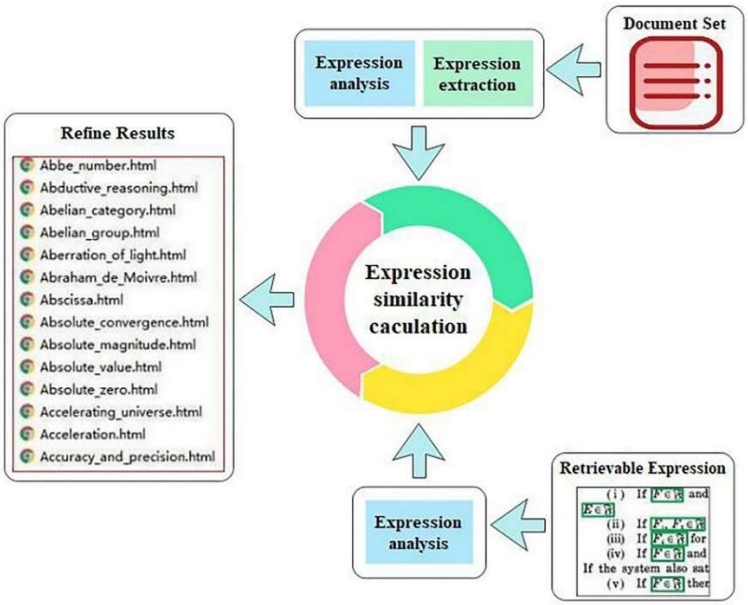

Figure 4. Flow chart of the mathematical matching module.

4.1. Mathematical expression analysis

Considering the syntactical and semantic variations in mathematical expressions, the FDS [29]

algorithm is used to analyze mathematical expressions. The analysis process is shown in Figure 5. First, the relevant symbols are separately obtained in LaTeX. Then, if a symbol is ordinary, we use the BaseAnalysis module to integrate numbers and letters and extract information; otherwise, according to the matching results based on the special symbol database, different types of special symbols are processed in different ways with the FunctionAnalysis module. In this paper, each symbol in mathematical expressions is parsed into a five-tuple attribute (level, flag, count, ratio, operator).

The meanings of the five-tuple attributes are described as follows:

(1) "level" is the level of the current mathematical symbol, which is based on the position of the horizontal baseline. For example, in the mathematical expression $a+c^{d}/b$, the level values of a, $+$, $\square/\square$, b, c and d are 0, 0, 0, 1, 1, 2 respectively.

(2) "flag" refers to the relationship of the current mathematical symbol to its nearest prior in the higher level. Its value is from 1 to 7 respectively represents the up, superscript, subscript, down, inclusion, left superscript and left subscript. And the flag values of the symbols in main baseline are 0.

(3) "count" is the sequential position of the current mathematical symbol in the mathematical expression.

(4) "ratio" represents the frequency of the operator in the mathematical expression.

(5) "operator" refers to whether the current symbol is an operator or not. If it is, the operator value is 1, otherwise it is 0.

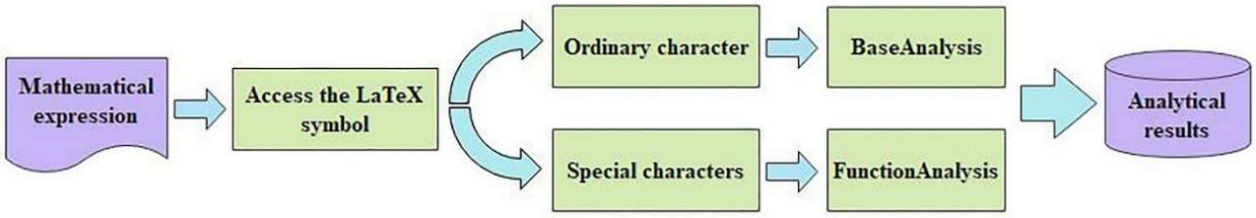

Figure 5. Flow chart of mathematical expression analysis.

4.2. Mathematical expression similarity calculations

In this section, the HFS is used to calculate the similarity between the expressions entered by the user and the expressions in the reference dataset. The definitions of relevant parameters and membership degrees are shown in Table 1. The implementation strategy is shown in Algorithm 1.

Table 1. Parameter and membership degree definitions.

Parameters/membership degrees	Description
F_Q	the mathematical query expression
$F_{D_i}(i=1,2,\cdots,N)$	the mathematical expression dataset
$S_{Qt\_q}(t\_q=1,2,\cdots,c\_Q)$	the t_q-th symbol of the mathematical query expression, c_Q being the total number of symbols
$S_{Dt\_d}(t\_d=1,2,\cdots,c\_D)$	the t_d-th symbol of the expression in the dataset, c_D being the total number of symbols
$M_{lev}(S_{Qt\_q},S_{Dt\_d})$	the membership degree between the levels of mathematical symbols
$M_{fla}(S_{Qt\_q},S_{Dt\_d})$	the membership degree between the flags of mathematical symbols
$M_{cou}(S_{Qt\_q},S_{Dt\_d})$	the membership degree between the counts of mathematical symbols
$M_{rat}(S_{Qt\_q},S_{Dt\_d})$	the membership degree between the ratios of mathematical symbols
$M_{ope}(S_{Qt\_q},S_{Dt\_d})$	the membership degree between the operators of mathematical symbols
$SUM(term)$	the sum of the attribute values of the five-tuple of mathematical symbols, $term$ being the attribute value
$SIM(F_Q,F_{Di})$	the similarity between mathematical expressions

Algorithm 1: Mathematical expression similarity calculation algorithm

Input: F_Q, $F_{D_i}(i=1,2,\cdots,N)$
Output: $SimExpList$ // A collection of expressions similar to F_Q

1. $S_{Qt\_q}(t\_q=1,2,\cdots,c\_Q)$
2. $S_{Dt\_d}(t\_d=1,2,\cdots,c\_D)$
3. for qe in S_{Qt_q}:
4. for fs in S_{Dt_d}:
5. if $qe==fs$:
6. $vec=[M_{rat}(qe,fs),M_{lev}(qe,fs),M_{ope}(qe,fs),M_{fla}(qe,fs),M_{cou}(qe,fs)]$
7. $list_{mem}.\text{add}([qe,qe.id,vec])$
8. else:
9. $list_{mem}.\text{add}([qe,qe.id,[0,0,0,0,0]])$
10. for mem in $list_{mem}$:
11. if $mem.qe$ not in $list_{fs}.qe$:
12. if $mem.id$ not in $list_{fs}.id$:
13. $list_{fs}.\text{add}(mem)$
14. if $SUM(mem.vec)/5 \geq SUM(list_{fs}.vec)/5$:
15. $list_{fs}.vec=mem.vec$
16. $SimExpList=SIM(list_{fs},list_{qe})$
17. **RETURN** $SimExpList$
18. **END**

5. Results and discussion

5.1. Experimental datasets

Our experiment used the TFD-ICDAR2019v2 dataset[1], NTCIR dataset and Chinese scientific documents (CSD) dataset for CA-YOLOv5[2] pretraining and mathematical document retrieval, respectively. The TFD-ICDAR2019v2 dataset contains 795 English PDF document images and a total of 38,181 annotated mathematical expressions. The NTCIR dataset contains 31,742 English documents, with a total of 518,929 mathematical expressions. Furthermore, to make the experimental data more convincing, we also add CSD dataset to expand the NTCIR dataset, which contains 10,372 documents and 121,495 mathematical expressions.

5.2. Experiment results

5.2.1. Results of mathematical query expression positioning

The results of mathematical query expression positioning are shown in Figure 6. Notably, CA-YOLOv5 can fairly accurately detect the mathematical expressions contained in the layout images. However, there are also some problems in the detection process. For example, "$c=0$" in the image is detected as "$e\ c=0$" because the font of the character c is similar to that of the preceding word "$case$", resulting in overdetection. Moreover, incomplete detection (only a part of an expression is detected) occurs in some cases. Based on the above analysis, most of the detection errors are caused by the failure to effectively split or merge some expressions during the detection process.

As a result, complete IoU (CIoU) was used as the evaluation metric for the results of the mathematical query expression positioning analysis. The description of the CIoU evaluation metric is shown in Eq (7).

$$\mathrm{CIoU}=\mathrm{IoU}-\frac{\rho^2(b,b^{gt})}{c^2}-\alpha\nu \tag{7}$$

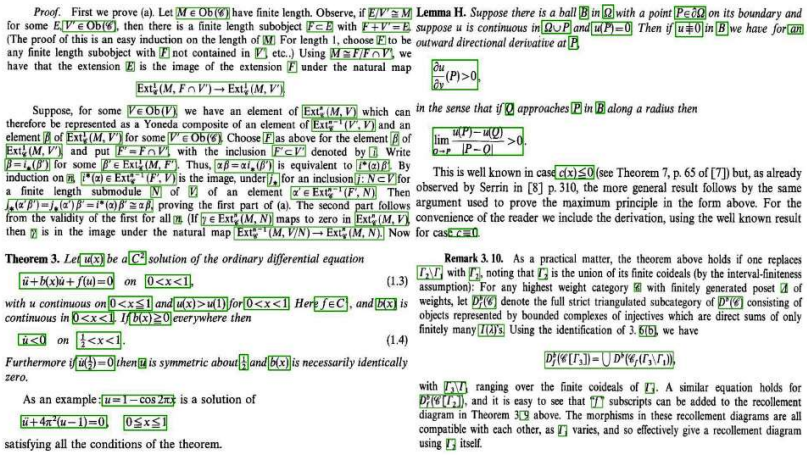

Figure 6. The results of mathematical query expression positioning.

[1] https://github.com/fireae/TFD-ICDAR2019/tree/master/TFD-ICDAR2019v2
[2] https://pan.baidu.com/s/17y4Cg-MDhpBLmZ-Xuoxfpg?pwd=spcv

In the equation, IoU represents the intersection over union of the ground truth and the anchors, $\rho^2(b,b^{gt})$ represents the Euclidean distance between the central points of the anchors and the ground truth, and c represents the diagonal length of the smallest enclosed area that can contain both the anchors and the ground truth. ν represents the blending degree of the aspect ratio of the anchors and the ground truth (for the expression in Eq (8)), α is the balance factor (formula is shown in Eq (9)).

$$\nu = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{8}$$

$$\alpha = \frac{\nu}{1-\mathrm{IoU}+\nu} \tag{9}$$

where w^{gt} is the width of the ground truth, h^{gt} is the height of the ground truth, w is the width of the corresponding anchor, and h is the height of the corresponding anchor.

In mathematical expression detection tasks, both the precision of detection and the recall rate must be considered. Therefore, this paper uses the F1-score to evaluate the system's detection performance. In Table 2, the detection performance of the proposed method is compared to that of the RIT 2 system [30], RIT 1 system [30] and Michiking system [30] used in the TFD-ICDAR2019 competition.

Table 2. Evaluation of the CA-YOLOv5 test results.

Method	Precision (%)	Recall (%)	F1-score (%)
RIT 2	**83.14**	67.00	75.41
RIT 1	74.40	68.47	71.32
Michiking	36.87	27.00	31.18
Ours	78.53	**74.66**	**76.54**

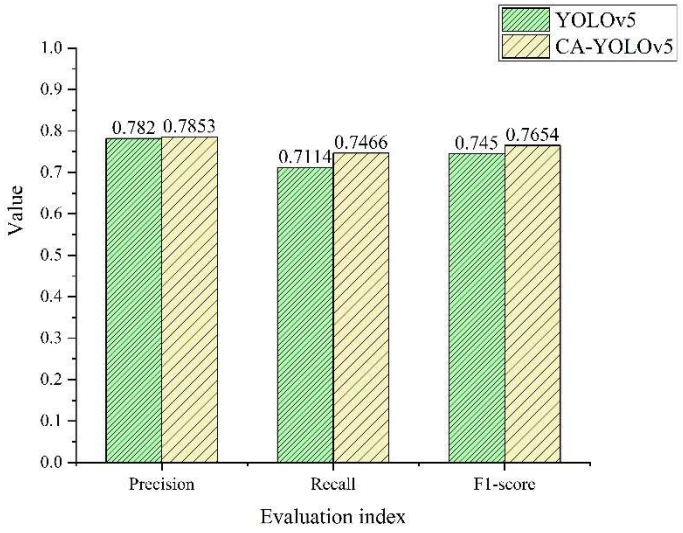

Figure 7. Ablation evaluation results of CA-YOLOv5 and YOLOv5.

Ablation study: When the CA model was introduced into the YOLOv5 network structure for mathematical expression detection, the detection performance was greatly improved. Compared with that of YOLOv5, the recall rate of our method increased by 3.52%, and the F1-score increased by 2.04%. A detailed comparison of results is shown in Figure 7. Therefore, the CA model can improve the performance of mathematical expression detection to a certain extent. Specifically, the CA model

can capture the long-term dependence of mathematical symbol pixels in one spatial direction and retain important positional information in the other direction, thereby improving the detection performance for long mathematical expressions and multiline mathematical expressions.

5.2.2. Mathematical document retrieval and ranking results

To increase the accuracy of the retrieval results from mathematical documents, ten mathematical query expressions obtained in the mathematical expression detection experiment are selected for retrieval. The mathematical query expressions and their LaTeX forms are listed in Table 3.

Table 3. Mathematical query expressions and their LaTeX forms.

Expressions	LaTeX
$f_{\infty}=0$	f_{\infty}=0
$f(u)\leqslant\lambda u$	f(u) \leq \lambda u
$\lambda>0$	\lambda>0
$\|Au\|\geqslant\|u\|$	|A u| \geq|u|
$u\epsilon K\cap\partial\Omega_1$	u \in K \cap \partial\Omega_{1}
$a^2+b^2=c^2$	a^{2}+b^{2}=c^{2}
$C=x/(x+y)$	C=\frac{x}{(x+y)}
$f(xy)=x+y$	f(x y)=x+y
$S=\pi r^2$	S=\pi r^{2}
$\log_a x$	\log _{a} x

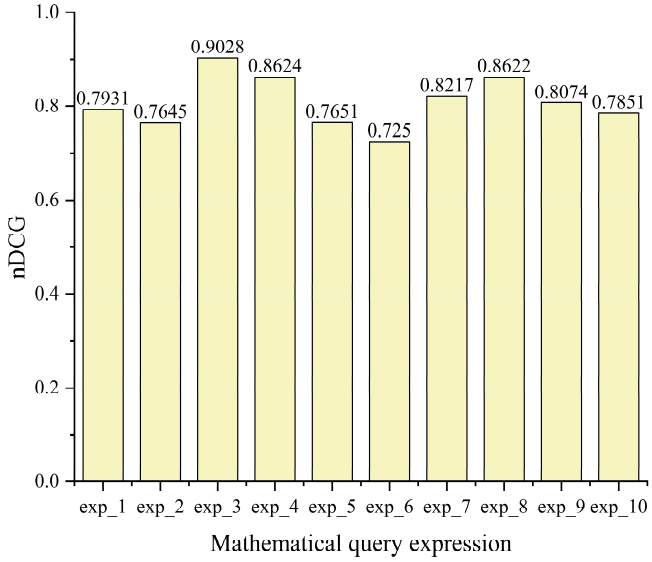

Figure 8. nDCG of mathematical document retrieval results under different expressions.

Since the mathematical document results retrieved by mathematical query expressions usually return a multi-result sequence, and the result sequence is in order, the DCG index is used to measure the ranking result. Eq (10) shows the corresponding formula, and nDCG is used to standardize the

retrieval results based on Eq (11).

$$\mathrm{DCG} = \sum_{i=1}^{P} \frac{2^{rel_i} - 1}{\log_2^{(i+1)}} (P \geqslant 1) \tag{10}$$

$$\mathrm{nDCG} = \frac{\mathrm{DCG}}{\mathrm{IDCG}} \tag{11}$$

i represents the ordinal number of the retrieval result. rel_i represents the classification of the ith retrieval result as excellent, good or bad; these classifications are associated with scores of 3, 2, and 1, respectively. P is the total number of retrieval results. IDCG represents DCG under ideal conditions.

In this experiment, mathematical document retrieval is performed using the mathematical query expressions in Table 3, and different retrieval results are obtained. The average nDCG of all mathematical documents is 80.89%, and the nDCG of mathematical document retrieval results under different mathematical query expressions is shown in Figure 8.

To explore the retrieval performance of our method, we use SearchOnMath [31] to conduct a comparative experiment. SearchOnMath is a mathematical document retrieval system based on mathematical expressions, but compared with our method, this system requires the manual input of mathematical query expressions in LaTeX format for retrieval, which is inconvenient. This paper uses the mathematical query expressions in Table 3 to compare the nDCG values obtained for our method and SearchOnMath, and the results are shown in Figure 9.

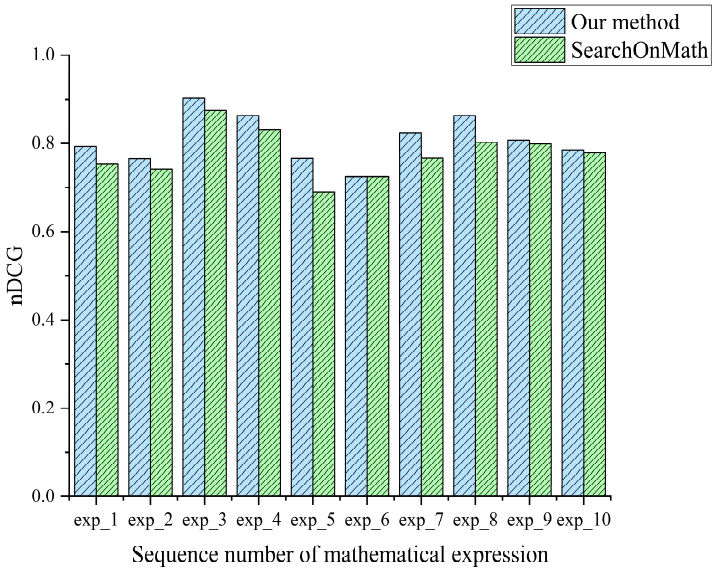

Figure 9. Comparison between the proposed method and SearchOnMath.

6. Conclusions

This paper proposes a mathematical document retrieval and ranking method based on CA-YOLOv5 and HFS for the input and matching of mathematical expressions. First, we use CA-YOLOv5 to automatically detect mathematical expressions in layout images and input them into the relevant retrieval module. Then, the membership degrees of the symbol attributes in each mathematical expression are calculated, and the similarity of the mathematical expressions is calculated based on the HFS. Finally, mathematical documents are sorted according to the similarity of mathematical expressions. This method integrates the advantages of CA-YOLOv5 and HFS, and improves the input

efficiency and matching accuracy of the mathematical document retrieval system to a certain extent.

However, this method has some limitations. In the future, we will improve the method from the following three perspectives:

(1) Continue to improve the CA-YOLOv5 network architecture, reduce the time required for mathematical expression detection, and improve the precision of mathematical expression detection;

(2) Add a Chinese dataset to the training set of CA-YOLOv5 to improve detection performance and enhance model applicability;

(3) Include information based on mathematical expressions and context, title, author, etc. in retrieval to further improve the accuracy of mathematical document retrieval and the relevance of the retrieval results.

Acknowledgments

This work is supported by the Natural Science Foundation of Hebei Province, China (No. F2019201329), and the Key Project of the Science and Technology Research Program in University of Hebei Province, China (No. ZD2019131).

Conflicts of interest

The author declares that there are no conflicts of interest in the publication of this article.

References

1. W. Chu, F. Liu, Mathematical formula detection in heterogeneous document images, in *2013 Conference on Technologies and Applications of Artificial Intelligence*, (2013), 140–145. https://doi.org/10.1109/TAAI.2013.38
2. P. Mali, P. Kukkadapu, M. Mahdavi, R. Zanibbi, ScanSSD: Scanning single shot detector for mathematical formulas in PDF document images, preprint, arXiv:200308005.
3. W. Ohyama, M. Suzuki, S. Uchida, Detecting mathematical expressions in scientific document images using a U-Net trained on a diverse dataset, *IEEE Access*, **7** (2019), 144030–144042. https://doi.org/10.1109/ACCESS.2019.2945825
4. B. H. Phong, L. T. Dat, N. T. Yen, T. M. Hoang, T. L. Le, A deep learning based system for mathematical expression detection and recognition in document images, in *12th International Conference on Knowledge and Systems Engineering*, (2020), 85–90. https://doi.org/10.1109/KSE50997.2020.9287693
5. B. H. Phong, T. M. Hoang, T. L. Le, Mathematical variable detection based on convolutional neural network and support vector machine, in *2019 International Conference on Multimedia Analysis and Pattern Recognition*, (2019), 1–5. https://doi.org/10.1109/MAPR.2019.8743543
6. X. Lin, L. Gao, Z. Tang, X. Lin, X. Hu, Mathematical formula identification in PDF documents, in *2011 International Conference on Document Analysis and Recognition*, (2011), 1419–1423. https://doi.org/10.1109/ICDAR.2011.285
7. X. Lin, L. Gao, Z. Tang, J. Baker, M. Alkalai, V. Sorge, A text line detection method for mathematical formula recognition, in *2013 12th International Conference on Document Analysis and Recognition*, (2013), 339–343. https://doi.org/10.1109/ICDAR.2013.75
8. X. Lin, L. Gao, Z. Tang, J. Baker, V. Sorge, Mathematical formula identification and performance evaluation in PDF documents, *Int. J. Doc. Anal. Recog.*, **17** (2013), 239–255. https://doi.org/10.1007/s10032-013-0216-1

9. L. Gao, X. Yi, Y. Liao, Z. Jiang, Z. Yan, Z. Tang, A deep learning-based formula detection method for PDF documents, in *2017 14th IAPR International Conference on Document Analysis and Recognition*, (2017), 553–558. https://doi.org/10.1109/ICDAR.2017.96
10. B. H. Phong, T. M. Hoang, T. L. Le, A. Aizawa, Mathematical variable detection in PDF scientific documents, *Intell. Inform. Database Syst.*, **11432** (2019), 694–706. https://doi.org/10.1007/978-3-030-14802-7_60
11. B. H. Phong, T. M. Hoang, T. L. Le, A hybrid method for mathematical expression detection in scientific document images, *IEEE Access*, **8** (2020), 83663–83684. https://doi.org/10.1109/ACCESS.2020.2992067
12. R. Deveaud, J. Mothe, M. Z. Ullah, J. Y. Nie, Learning to adaptively rank document retrieval system configurations, *ACM Trans. Inform. Syst.*, **37** (2019), 1–41. https://doi.org/10.1145/3231937
13. K. Yamada, H. Murakami, Mathematical expression retrieval in PDFs from the Web using mathematical term queries, in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, **12144** (2020), 155–161. https://doi.org/10.1007/978-3-030-55789-8_14
14. P. Sojka, M. Růžička, V. Novotný, MIaS: math-aware retrieval in digital mathematical libraries, in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, (2018), 1923–1926. https://doi.org/10.1145/3269206.3269233
15. M. Schubotz, N. Meuschke, T. Hepp, H. S. Cohl, B. Gipp, VMEXT: a visualization tool for mathematical expression trees, *Intell. Comput. Math.*, **10383** (2017), 340–355. https://doi.org/10.1007/978-3-319-62075-6
16. M. Líška, P. Sojka, M. Růžička, Combining text and formula queries in math information retrieval, in *Proceedings of the First International Workshop on Novel Web Search Interfaces and Systems*, (2015), 7–9. https://doi.org/10.1145/2810355.2810359
17. W. Zhong, S. Rohatgi, J. Wu, C. L. Giles, R. Zanibbi, Accelerating substructure similarity search for formula retrieval, *Adv. Inform. Retrieval*, **12035** (2020), 714–727. https://doi.org/10.1007/978-3-030-45439-5
18. D. Stalnaker, R. Zanibbi, Math expression retrieval using an inverted index over symbol pairs, *Int. Soc. Opt. Photonics*, **9402** (2015), 940207. https://doi.org/10.1117/12.2074084
19. X. Tian, J. Wang, Retrieval of scientific documents based on HFS and BERT, *IEEE Access*, **9** (2021), 8708–8717. https://doi.org/10.1109/ACCESS.2021.3049391
20. S. Hussain, S. Khoja, Retrieval of mathematical information with syntactic and semantic structure over Web, *J. Inform. Sci. Engineering*, **36** (2020), 75–89.
21. J. Xu, C. Xu, Computing similarity of Sci-Tech documents based on texts and formulas, *Data Anal. Knowl. Discov.*, **2** (2018), 103–109. https://doi.org/10.11925/infotech.2096-3467.2018.0211
22. A. Pathak, P. Pakray, R. Das, Context guided retrieval of math formulae from scientific documents, *J. Inform. Optimization Sci.*, **40** (2019), 1559–1574. https://doi.org/10.1080/02522667.2019.1703255
23. P. Scharpf, M. Schubotz, A. Youssef, F. Hamborg, N. Meuschke, B. Gipp, Classification and clustering of arXiv documents, sections, and abstracts, comparing encodings of natural and mathematical language, in *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, (2020), 137–146. https://doi.org/10.1145/3383583.3398529
24. M. Schubotz, P. Scharpf, O. Teschke, A. Kühnemund, C. Breitinger, B. Gipp, AutoMSC: Automatic Assignment of Mathematics Subject Classification Labels, *Int. Conf. Intell. Comput. Math.*, (2020), 237–250. https://doi.org/10.1007/978-3-030-53518-6_15

25. V. Torra, Hesitant fuzzy sets, *Int. J. Intell. Syst.*, **25** (2010), 529–539. https://doi.org/10.1002/int.20418
26. G. Jocher, K. Nishimura, T. Mineeva, R. Vilariño: YOLOv5, 2020. Available from: https://github.com/ultralytics/yolov5
27. Q. Hou, D. Zhou, J. Feng, Coordinate attention for efficient mobile network design, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2021), 13713–13722. https://doi.org/10.1109/CVPR46437.2021.01350
28. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2016), 779–788. https://doi.org/10.1109/CVPR.2016.91
29. X. Tian, A mathematical indexing method based on the hierarchical features of operators in formulae, *Adv. Eng. Res.*, **119** (2017), 49–52.
30. M. Mahdavi, R. Zanibbi, H. Mouchere, C. Viard-Gaudin, U. Garain, ICDAR 2019 CROHME+ TFD: Competition on recognition of handwritten mathematical expressions and typeset formula detection, in *2019 International Conference on Document Analysis and Recognition*, (2019), 1533–1538. https://doi.org/10.1109/ICDAR.2019.00247
31. C. Wang, Y. Yang, F. Deng, H. Lai, A review of text similarity approaches, *Inform. Sci.*, **37** (2019), 1007–7634. https://doi.org/10.13833/j.issn.1007-7634.2019.03.026

AIMS Press