HmwkCheck: A Homework Auto-Checking System based on Arithmetic Operation Recognition using Smartphones

Lingyu Zhang
Kuang Yaming Honors School,
Nanjing University
Nanjing, China
171240524@smail.nju.edu.cn

Yafeng Yin
State Key Laboratory for Novel Software Technology,
Nanjing University
Nanjing, China
yafeng@nju.edu.cn

Lei Xie
State Key Laboratory for Novel Software Technology,
Nanjing University
Nanjing, China
lxie@nju.edu.cn

Sanglu Lu
State Key Laboratory for Novel Software Technology,
Nanjing University
Nanjing, China
sanglu@nju.edu.cn

ABSTRACT

The homework for low-grade pupils often contains simple arithmetic problems, i.e., four arithmetic operations. To evaluate the learning quality of pupils, teachers and parents often need to check the homework manually, which is time and labor consuming. In this paper, we propose a homework auto-checking system HmwkCheck, which checks the four arithmetic operations automatically. Specifically, HmwkCheck utilizes the embedded camera of a smartphone to capture the homework as an image, and then processes the image in the smartphone to detect, segment and recognize both printed characters and handwritten characters. We implement HmwkCheck in an Android smartphone. The experiment results show that HmwkCheck can check homework efficiently, i.e., the average precision, recall and F1-score of character recognition achieve 94.03%, 93.41% and 93.72%, respectively.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; *Ubiquitous and mobile computing*.

KEYWORDS

Homework auto-checking; Arithmetic operation recognition; Image processing; Smartphone

ACM Reference Format:
Lingyu Zhang, Yafeng Yin, Lei Xie, and Sanglu Lu. 2020. HmwkCheck: A Homework Auto-Checking System based on Arithmetic Operation Recognition using Smartphones. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers (UbiComp/ISWC '20 Adjunct), September 12–16, 2020, Virtual Event, Mexico.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3410530.3414393

UbiComp/ISWC '20 Adjunct, September 12–16, 2020, Virtual Event, Mexico

ACM ISBN 978-1-4503-8076-8/20/09.
https://doi.org/10.1145/3410530.3414393

1 INTRODUCTION

Homework is often adopted in education, and it can be used to evaluate the teaching quality of teachers and learning quality of students. However, correcting the homework manually can be time and labor consuming, especially for the homework assigned towards a larger number of students frequently, e.g., the arithmetic operation problems assigned for low-grade pupils. Therefore, the automatic homework checking approaches were proposed to reduce human cost. With a scanned image, the optical character recognition (OCR) [3] was a typical technology used to recognize printed characters. By taking a picture of homework, image processing [4] was used to detect and recognize handwritten answers. When given the image, the existing approaches usually send the image to a server for character recognition.

Different from the existing approaches, we aim to recognize both printed characters and handwritten answers in arithmetic operations. In addition, considering the popularity of smartphones, we aim to perform homework auto-checking locally in a smartphone. Without transmitting images on the Internet, we can further protect the user's privacy. As shown in Fig. 1, the homework is four arithmetic operation. At first, we use the embedded camera of a smartphone to capture an image of the homework. Then, we use image processing to recognize the characters and calculate the arithmetic expression to check the homework.

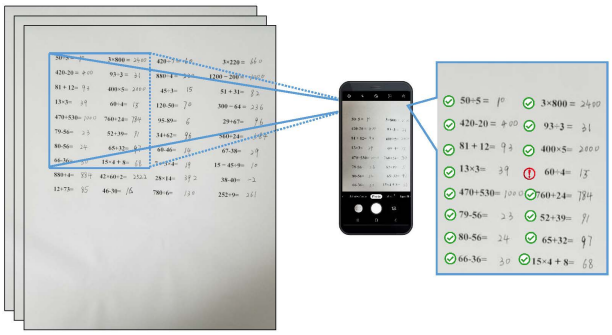

Figure 1: Homework auto-checking using smartphones

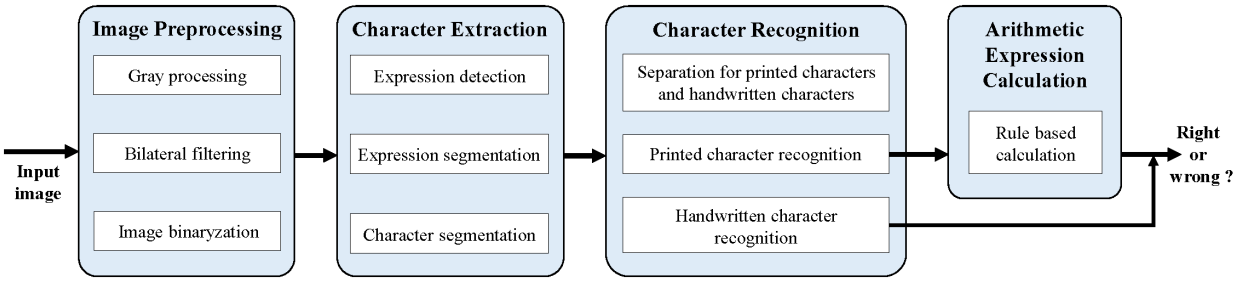

Figure 2: System overview

However, there are some challenges in the problem. Usually, the picture taken by the smartphone is not as clear as the scanned image, and characters may have some deformation. Besides, the printed characters and handwritten ones usually have different styles, even for the same numeral. In addition, the computing power of a smartphone is limited. Therefore, to achieve the goal, we first introduce image preprocessing to distinguish the characters and background. Then, we extract each character based on horizontal projection and vertical projection. After that, we separate the arithmetic expression and recognize printed characters and handwritten ones individually. We also choose suitable image sizes to make the system work on a smartphone.

2 RELATED WORK

Four arithmetic operations including numerals and operators often appear in the homework of low-grade pupils. To recognize and calculate arithmetic operations automatically, Meng et al. [5] used a BP neural network and template matching to recognize printed numerals and operators. To recognize handwritten-style characters, Jiang et al. [2] used the end-to-end learning technology to recognize arithmetic operations in fixed-size or varied-size images, where the characters were CAPTCHA-style. Khalighi et al. [3] proposed a novel OCR system to recognize and calculate handwritten Persian arithmetic expression. Instead of only recognizing numerals and operators, Li et al. [4] proposed BAGS, an automatic homework grading system based on the pictures taken by smart phones. BAGS detected answer areas in the answer sheets and recognized the handwritten characters (e.g., words). In BAGS, the images were processed in another computers or servers instead of smartphones. Different from the existing work, we provide a homework auto-checking system based on images taken by smartphones, aiming to recognize both printed characters and handwritten characters in arithmetic operations. Besides, our system runs on the easy-to-get smartphone locally without transmitting images or processing images offline.

3 SYSTEM DESIGN

To check the homework consisted of four arithmetic operations, it is necessary to detect the characters, recognize the characters and verify the calculation of arithmetic expression. Therefore, the proposed system HmwkCheck consists of four components, i.e., image preprocessing, character extraction, character recognition, arithmetic expression calculation, as shown in Fig. 2.

3.1 Image Preprocessing

As shown in Fig. 1, the four arithmetic operations are in white paper, and the picture taken by the smartphone is a RGB image. To remove noises and distinguish characters with the background in the image, we first preprocess the image. As shown in Fig. 2, we first process the raw image, i.e., picture taken by the smartphone, with gray-scale image processing. Then, we use a bilateral filter to remove noises while keeping the edges in the image. After that, we perform image binaryzation to separate the foreground (i.e., characters) and background, i.e., the characters are in black color while the background is in white color. The preprocessed image will be used for the following character extraction.

3.2 Character Extraction

To extract the characters, we need to detect the arithmetic expression and separate each character. Specifically, with the binarized image, we use the horizontal projection of pixels in each row to detect the arithmetic expression. Supposed the coordinate of a pixel in an image is (x_i, y_j), $i \in [1, w]$, $j \in [1, h]$, where w and h represent the width and height of the image, respectively. If the number n_p of black pixels (x_i, y_p), $i \in [1, w]$ in the pth row satisfies $n_p > \epsilon_p$, the row is treated as 'Expression-Row', where ϵ_p is set to 2 by default. By connecting the consecutive 'Expression-Rows', we can get the arithmetic expressions, as shown in Fig. 3(a). To further segment the arithmetic expressions in same rows, e.g., "$28 \times 14 = 392$" and "$38 \times 40 = 1520$", we introduce the vertical projection of each pixels in each column, where the principle of vertical projection is similar to that of horizontal projection. After that, we can get each arithmetic expression, as shown in Fig. 3(b). In addition, to segment each character from the arithmetic expression, we repeatedly use the vertical projection. Finally, we can extract each arithmetic expression and its corresponding characters, as shown in Fig. 3(c).

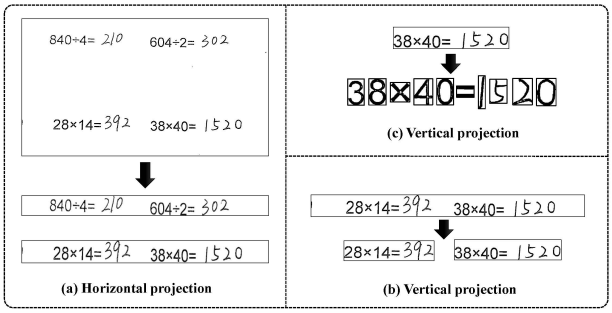

Figure 3: An example of character segmentation

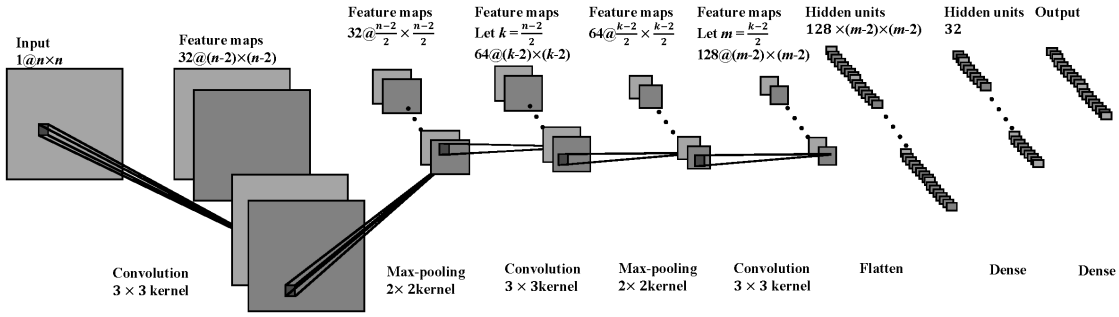

Figure 4: The CNN based character recognition

3.3 Character Recognition

After character extraction, we will recognize each detected character. Due to the different styles of printed numerals and handwritten ones, e.g, the different styles of printed '9' and handwritten '9', we recognize the printed characters and handwritten ones separately, to avoid the interference between them. It is worth noting that the printed characters include numerals '0'-'9' and five operators '+', '−', '×', '÷', '=', while the handwritten characters include numerals '0'-'9'.

Firstly, we utilize the structure of the arithmetic expression, i.e., 'P' = 'H', to separate printed characters and handwritten ones. Here, 'P' means the printed part consisted of numerals and operators, '=' is the equal sign, and 'H' means the handwritten part consisted of numerals. Therefore, if we can recognize the equal sign '=', we can separate printed characters and handwritten ones in an arithmetic expression. To achieve this goal, we first use the 'printed-model' to recognize the characters in an expression from left to right, while treating each character as a printed one by default. Once we recognize the '=', we will change to use the 'handwritten-model' for the following handwritten character recognition, as shown in Fig. 5.

In regard to the 'printed-model' and 'handwritten-model' used for printed and handwritten character recognition respectively, they both use the convolutional neural network (CNN) shown in Fig. 4. For the CNN in 'printed-model', the input image size of each character is '28×28' (i.e, $n = 28$) and the output corresponds to 15 classes , while for the CNN in 'handwritten-model', the input image size of each character is '56×56' (i.e, $n = 56$) and the output corresponds to 10 classes. For the segmented image containing the character, it is filled with background color to satisfy the above image size requirement. Then, the detected character will be sent to the corresponding CNN for further recognition.

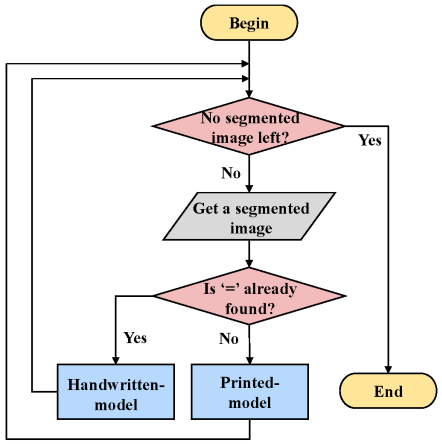

Figure 5: The process of recognizing characters

3.4 Arithmetic Expression Calculation

When each character is recognized, we will calculate the arithmetic expression to verify whether the answer is right. Specifically, we use the rules of four arithmetic operations to calculate the answer based on the recognized printed numerals and operators. Then, we compare the calculated answer with the handwritten one. If they are the same, then the handwritten answer is right. Otherwise, the handwritten answer is treated as wrong.

4 SYSTEM IMPLEMENTATION AND PERFORMANCE EVALUATION

In this paper, we aim to recognize four arithmetic operations and provide a homework auto-checking system running on a smartphone for low-grade pupils. In the following subsections, we will show the system implementation and evaluate the performance of HmwkCheck.

4.1 System Implementation

We implement HmwkCheck in an Android smartphone. As shown in Fig. 6(a), HmwkCheck first takes a picture of the homework. Then, it preprocess the image, as the binarized image shown in Fig. 6(b). After that, it extracts and recognizes the characters, and then calculates the arithmetic expression to verify whether the handwritten answer is right, as the colored text shown in Fig. 6(b). The arithmetic expression with right answer is shown in green, while that with wrong answer is shown in red. Sometimes, the red expression may be caused by the recognition error, i.e., the actual handwritten answer is right. Take the red expression in Fig. 6(b) as an example, the handwritten '6' is wrongly recognized as '8', while the actual arithmetic expression is right. Neverthless, we make the potential wrong answer with red to raise concern.

4.2 Performance Evaluation

To evaluate the performance of HmwkCheck, we invite 20 volunteers to perform the following experiments. As shown in Fig. 1, the homework containing four arithmetic operations is in white paper. The space between the expressions or characters is comparable to

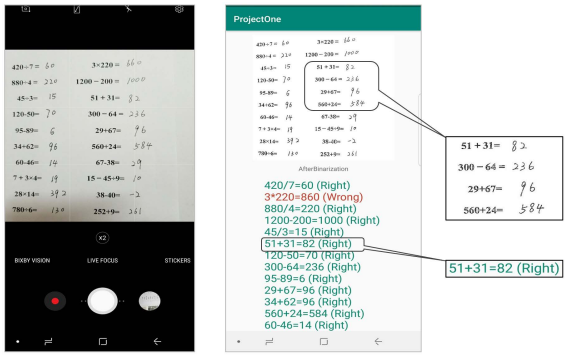

Figure 6: System implementation

that of the homework for low-grade pupils. Each volunteer calculated all the 40 expressions in a piece of paper and wrote the answers. Unless otherwise specified, we use the A4-sized white paper and Samsung Note 8 smartphone to take pictures. To train the 'printed-model', we print out numerals and operators, use smartphone to take pictures, and get 360 images for each printed character. To train the 'handwritten-model', we use the public USPS data set [1] and randomly select 2150 images for each handwritten character. Considering that the calculation of arithmetic expression has certain rules which are fixed, we focus on the evaluation of character recognition.

In Fig. 7, we show the recognition performance for printed characters, handwritten characters, and both of the them. For printed characters, they have fixed typefaces, thus can be recognized well, i.e., the precision, recall and F1-score achieve 96.22%, 97.01% and 96.61%, respectively. The failure of recognizing some printed characters may be caused by the interference in the input image, e.g., smudges around the printed characters. For handwritten characters, different people can write the same character in different styles, thus the recognition performance of handwritten characters is not as good as that of printed characters, i.e., the precision, recall and F1-score are 88.24%, 84.34% and 86.25%, respectively. In addition, some illegible handwritten characters also decrease the recognition performance. For example, two handwritten characters having overlap may be segmented as one character, the strokes in one character away from each other may be segmented as two characters, which can lead to the recognition error. Overall, HmwkCheck can recognize characters efficiently, the average precision, recall and F1-score of character recognition achieve 94.03%, 93.41% and 93.72%, respectively.

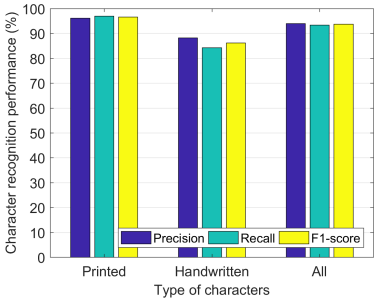

Figure 7: Character recognition performance

5 CONCLUSION

In this paper, we propose a homework auto-checking system HmwkCheck focusing on the recognition of four arithmetic operations for low-grade pupils. We use the embedded camera of a smartphone to take a picture of the homework, and then process the image to recognize characters. After that,we calculate the arithmetic expression to check the handwritten answer. We implement HmwkCheck in an Android smartphone. The experiment results show that we can recognize characters accurately.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China under Grant Nos. 61802169, 61872174, 61832008, 61902175, 61906085; JiangSu Natural Science Foundation under Grant Nos. BK20180325, BK20190293; the Key R&D Program of Jiangsu Province under Grant No. BE2018116. This work is partially supported by Collaborative Innovation Center of Novel Software Technology and Industrialization. Yafeng Yin is the corresponding author.

REFERENCES

[1] Jonathan J. Hull. 1994. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16, 5 (1994), 550–554.

[2] Y. Jiang, H. Dong, and A. El Saddik. 2018. Baidu Meizu Deep Learning Competition: Arithmetic Operation Recognition Using End-to-End Learning OCR Technologies. *IEEE Access* 6 (2018), 60128–60136.

[3] S. Khalighi, P. Tirdad, H. R. Rabiee, and M. Parviz. 2009. A Novel OCR System for Calculating Handwritten Persian Arithmetic Expressions. In *2009 International Conference on Machine Learning and Applications*. 755–758.

[4] Xiaoshuo Li, Tiezhu Yue, Xuanping Huang, Zhe Yang, and Gang Xu. 2019. BAGS: An automatic homework grading system using the pictures taken by smart phones. *CoRR* abs/1906.03767 (2019). arXiv:1906.03767 http://arxiv.org/abs/1906.03767

[5] Wang Meng, Bian Guang-rong, Zheng Kai, He Ninghui, and Zhang Yan-lei. 2017. Recognition Technology for Four Arithmetic Operations. *TELKOMNIKA Telecommunication Computing Electronics and Control* 15, 1 (2017), 306–313.