https://doi.org/10.26682/sjuod.2022.25.2.24
Journal of University of Duhok., Vol. 25, No.2 (Pure Engineering Sciences), Pp 261-268, 2022

GRADING MULTIPLE CHOICE QUESTIONS BASED ON PREPARED QUESTIONS AND OPTIONS BOOKMARKS IN BUBBLE SHEET

SARDAR **O**MAR **S**ALIH
Dept. of Information Technology, Duhok Technical Institute, Duhok Polytechnic University

(Received: August 9, 2022; Accepted for Publication: November 8, 2022)

ABSTRACT

The process of evaluating students' answers is a time consuming and effort for teachers, therefore, based on this, Grading Multiple Choice Questions (G-MCQ) is proposed to auto-marking answer without human interaction. All the human does, is to use digital camera without using expensive ordinary document scanner and machine-read for this purpose, then, evaluating and marking each correct answer is algorithm duty. G-MCQ is based on a prepared bubble sheet that contains (54) questions with four circles options for each question, G-MCO is programmed using Python programming language, passes three main process , the first one, is a preparation of scanned document, then, second one, is to detect bookmarks, First Question Bookmark (FQB), Questions Bookmarks (QB) and Options Bookmarks (OB) positions, based on detecting FQB, QB and OB, the final one is started to detect answers which are circles positions of each question from instructor. The algorithm is tested with input images with PNG and JPG format, the result of detecting of accuracy is about 99%.

KEYWORDS: OMR (Optical Mark Recognition), Scanned Document (Bubble Sheet), Multiple Choice, Bookmarks

1 INTRODUCTION

The most difficult part in education for the teachers is to assess students, due to, it takes time, effort and cost, to reduce this, this paper is proposed to use auto-grading of MCQs from scanned documents. This paper uses computer vision techniques which are used wildly to auto arrange tasks for varsity applications such as component identification, inspection, quality control and license plate recognition. (Cardona, 2016) (Kulkarni et al., 2019) (Ugwu et al., 2022) as well as, using for optical character recognition (OCR) and pattern recognition.(Barik & Mondal, 2010)

There are several approaches have been used of computer vision to detect Multiple Choice Question MCQ answer, Abbas proposed 15 questions with answer sheet must not be rotated by 45 degrees and use rectangle for options , as he claimed some of student has a difficult to black answer (Abbas, 2009), Chinnasarn and Rangsanseri are used a histogram technique to detect answer by checking four options, then the highest black pixel on the answer is considered to be selected answer (Chinnasarn & Rangsanseri, 1999), Yimyam and Ketcham use mobile device to capture answer sheet and can obtain result of accuracy 96 (Yimyam & Ketcham, 2018), Spadaccini and Rizzo use existing framework from python library named Gamera framwork to analyze scanned document with cross over answer and he can obtain over 99% on recognition (Spadaccini & Rizzo, 2011). Fisteus team create a system named Eyegrade offers a truly low-cost solution with adding student ID recognition beside to OCR.(Fisteus et al., 2013). Zampirolli and their co-authors use matrix of 7 row for option and 2 of them as bookmark and 26 columns with two of them for bookmarks and use morphological opening operation to detect square options of size 10px (de Assis Zampirolli et al., 2010).My contribution to this approach as following:

- Answer sheet can be taken using mobile camera without depending on expensive ordinary scanner and marking machine
- Quality of the image (bubble sheet) is not affected on algorithm process
- Image taken in any angle cannot affect accuracy of algorithm but must be taken in bird-eye view.
- G-MCQ only checks answers position, this can avoid from noising.
- Accuracy is over 99% if a circle is blacked, sensitivity of detecting black circle can be increased and decreased.

sardar@dpu.edu.krd

2 METHODOLOGY

G-MCQ starts to scan answer document which is taken by digital camera, should be in the "bird-eye view [1]" bubble sheet paper located inside it as rectangle, Figure 1 (left side) in "bird eye view" and right side is a converted to top-down view , the algorithm tries to detect outside contour of the document and rotate as a rectangle with 90-degree , top-down view see Figure 1 (right side) to be readied for processing. The area before edge (outside contour) should be removed , in this process , the output paper is bordered with black line , this is also need to be removed , for this reason , two points , Left Start Point (LSP) and Top Start Point (TSP) which are the lines after black lines border need to be detected , depending on these lines LSP and TSP , the algorithm tries to find Top Margin (TM) , Left Margin (LM) and Right Margin (RM), after all these findings, cropping process is applied , then , it continuous to locate questions bookmarks which are (27) with 8 options bookmarks , the intersection between each QB and OB gives us circle location on the paper . The G-MCQ scans each of circles of the questions assigned by instructors, if it is blacked and matched with instructors correct answer, it counts as right answer. otherwise, it counts wrong answer. more than one answer in one question is not countable and considered to be wrong answer. In this paper the methods of algorithms and its procedures are explained with the method of testing.

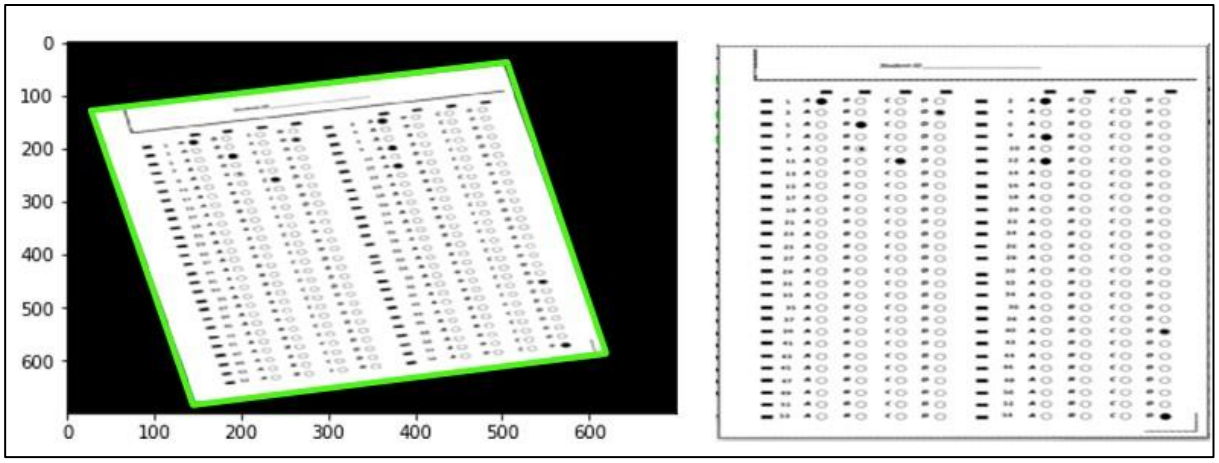
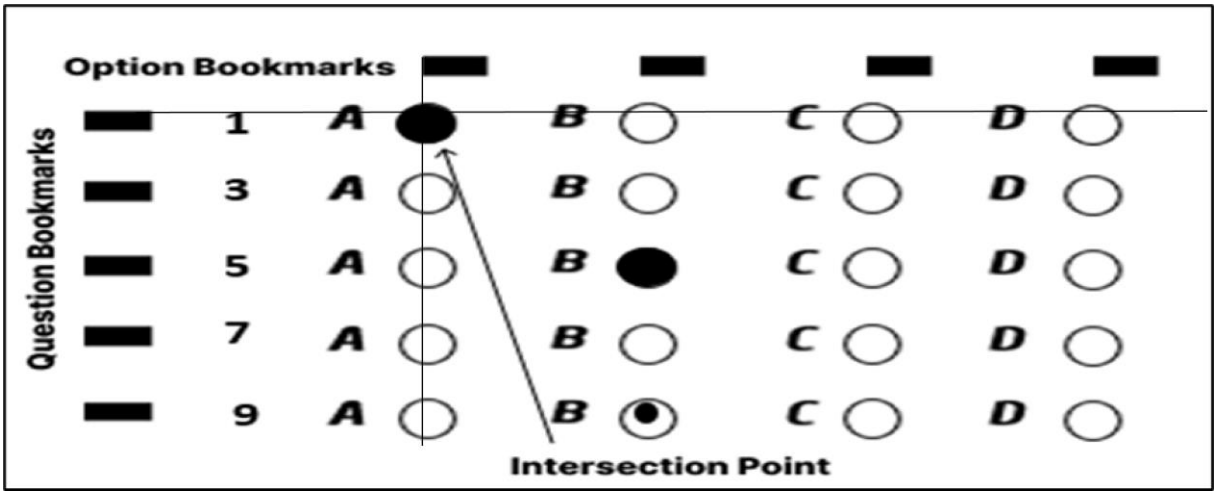


Figure 1 Remove background, left side is a bird eye view and right side is an output

3 ALGORITHM SECTIONS

3.1 DIGITIZE OF THE ANSWER SHEETS

After the questions are answered, circles are blacked, then, document is digitized by camera, this digitized document, then, converted to the binary image for processing, the binary image is a bunch of 0s and 1s represent only two colors black and white represent in array. Figure 2 shows bookmarks for both questions and answer with black circle.



3.2 REMOVE AREA BEFORE LEFT START POINT AND TOP START POINT

To finding outside contours of document "edge of the document" in bird eye view and rotate it, the "Moore Neighborhood" algorithm is used to remove black area (Sharma et al., 2013), in this case, outputted document gives us a border around them, Figure1 (left side, colored in green), therefore, this boarder needs to be removed, for this purpose LSP is positioned to be a start point for scanning document, LSP is a line after black border 'colored in green' , can be called (left vertical line).

The process of finding this line starts to store minimum value of these pixels is considered a LSP. same concept is applied to TSP, it stores each while pixel after black line and then, the minimum value is considered to be TSP, if there is no border both points are considered to be zero.

3.3 DETECTING LEFT MARGIN (LM) , RIGHT MARGIN (RM) AND TOP MARGIN (TM)

The aim of finding margins is to crop space from LST to the edge of Question Bookmarks (QB) and to crop space from TSP to the edge of Options Bookmarks (OB) (Figure 3 LM is 'dotted space'). To detect LM position in the binary image , the algorithm scans document from left to right and store leftmost black pixel in the array , then the minimum number in the array is considered Left Margin (LM), to find the Right Margin (RM) , it scans from right edge to the left and store rightmost black pixel in the array , the er in the array is considered to be RM , some process is applied to Top Margin , but it scans from top to down (left to right) and store topmost black pixel in the array , the maximum value in array is counted as TM. The following Figure shows LM, RM and TM which are detected.

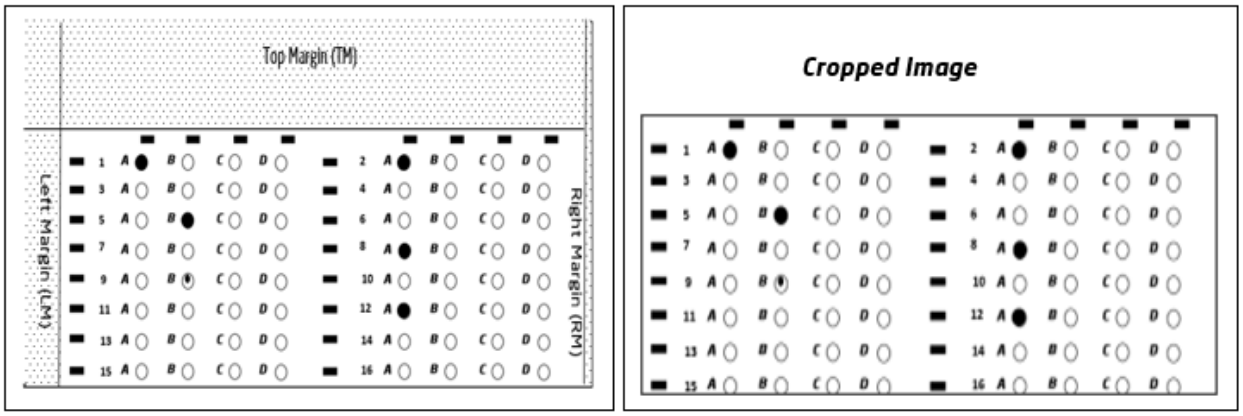


Fig. (3): Left side: Top, Left and Right Margin in dotted point, Right side :cropped document

Following is an arithmetic representation of LM, RM and TM

LM = $\min(x_{i,j})$, $x_{i,j}$ is left most black pixels , i =1, 2, 3, …rows and j=1, 2, 3, …columns

RM = $\max(x_{i,j})$, $x_{i,j}$ is right most black pixels , i =1, 2, 3, …rows and j=1, 2, 3, …columns

TM = $\min(x_{j,i})$, $x_{i,j}$ is top most black pixels , i =1, 2, 3, …rows and j=1, 2, 3, …columns

3.4 FINDING BOOKMARKS

In order to locate circle position, the First Question Bookmark (FQB), Options Bookmarks (OBs) and Questions Bookmarks (QBs) must be detected , the following is explanation of how these bookmarks can be detected .

3.4.1 Detecting First Question Bookmark (FQB)

The process of finding this point is as follow, it scans from LM to RM horizontally and from TM to down vertically, if any first black pixel is detected, it is considered an FQB. otherwise, the algorithm keeps looking for black pixel until to detect it. See Figure 4 FQB.

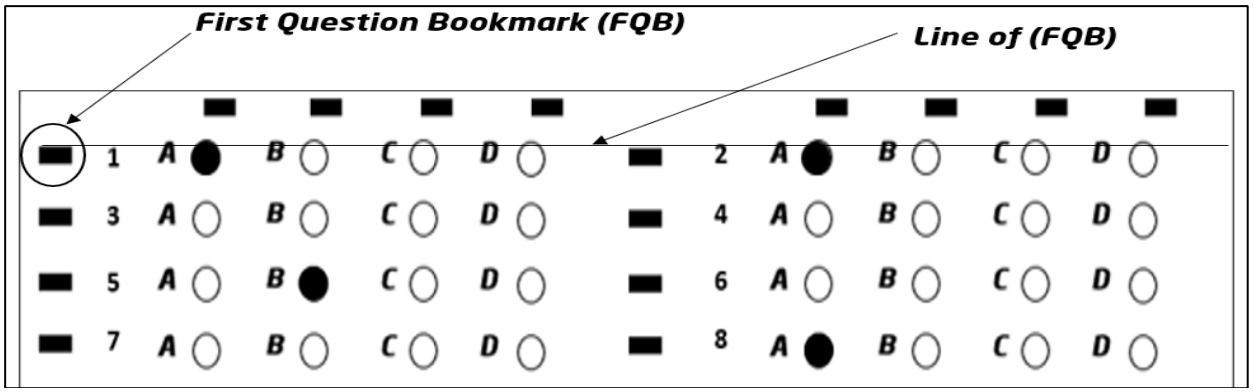


Fig. (4): First Question Bookmarks

3.4.2 Detecting Options Bookmarks

Bubble sheet shows (Figure 5) that options bookmarks (OB) are located between TM line and FQB line, the algorithm scans in this range, it scans from TM and goes down to reach black pixel , if there is, then stores in array and counted an option bookmark if black pixel is not found and reach to the line of FQB horizontally , then it goes one step to the left and do the same process , in case black pixel is found, it scans from right to left to pass black pixels, and start again for next OB .

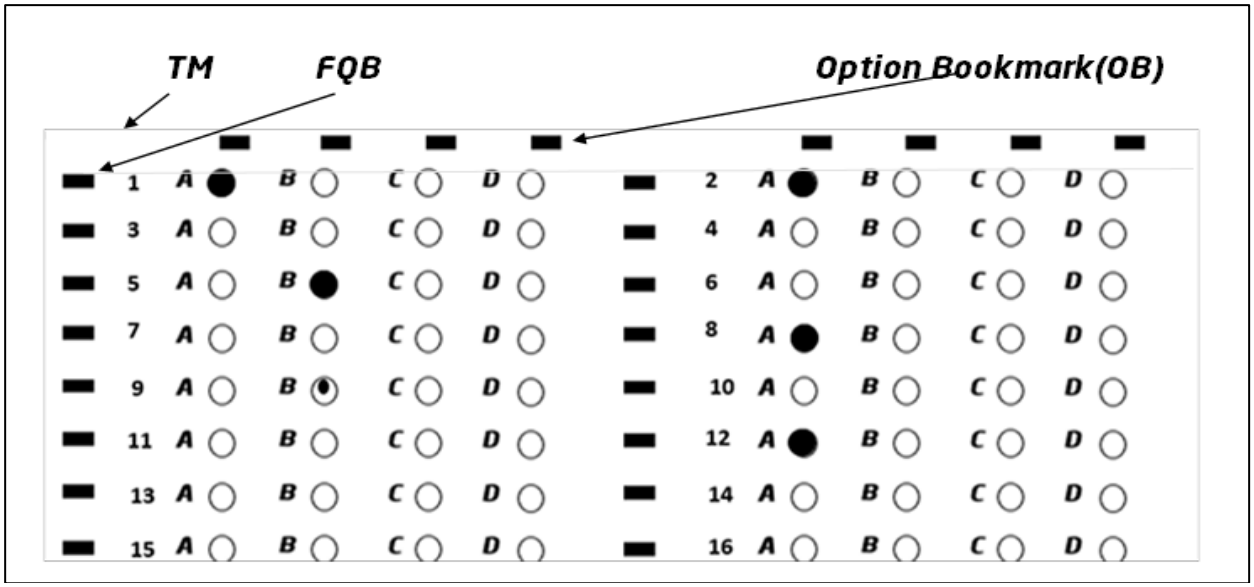


Fig. (5): Option Bookmarks

3.4.3 Detecting Questions Bookmarks (QB)

To find QB, the process of scanning starts from FQB and goes down to find black pixel and stores in array as QB, the algorithm tries to pass all back pixel and scans for next QB , the process is continues until

all QBs are detected, Figure 6 shows detected OB with green circle and QB and intersection between them can locate circle positions.

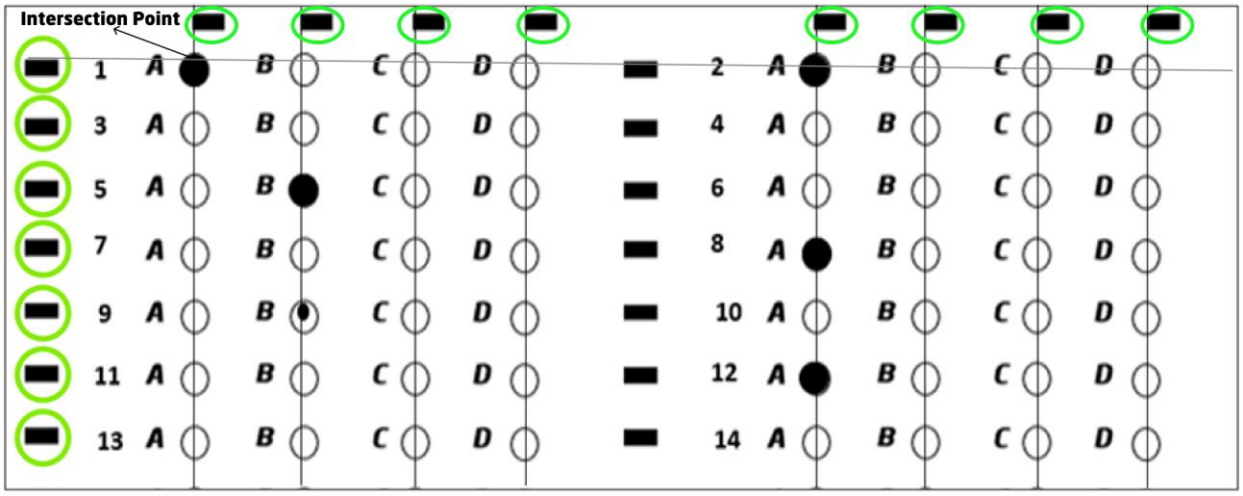


Fig. (6): Questions Bookmarks (QB)

3.4.4 Locating Circle Position

As it shown in Figure 6 that the intersection between each question bookmark and option bookmark can locate circle position, the algorithm checks numbers of black pixels in circle if its greater than 10 and less than 20, it considers a circle is blacked(selected), otherwise, the circle is not blacked (not selected) , this can be increased and decreased to gain sensitivity of black circle, algorithm looks at the instructor correct locations (correct answers), and count it. The Figure bellow left side shows correct answers from instructors and right side with red dot answered by student, the fill red circles indicate that the student answered this option and the green circles indicate instructors correct answers, as it shows in Figure 7, that student grade is 6 from 10.

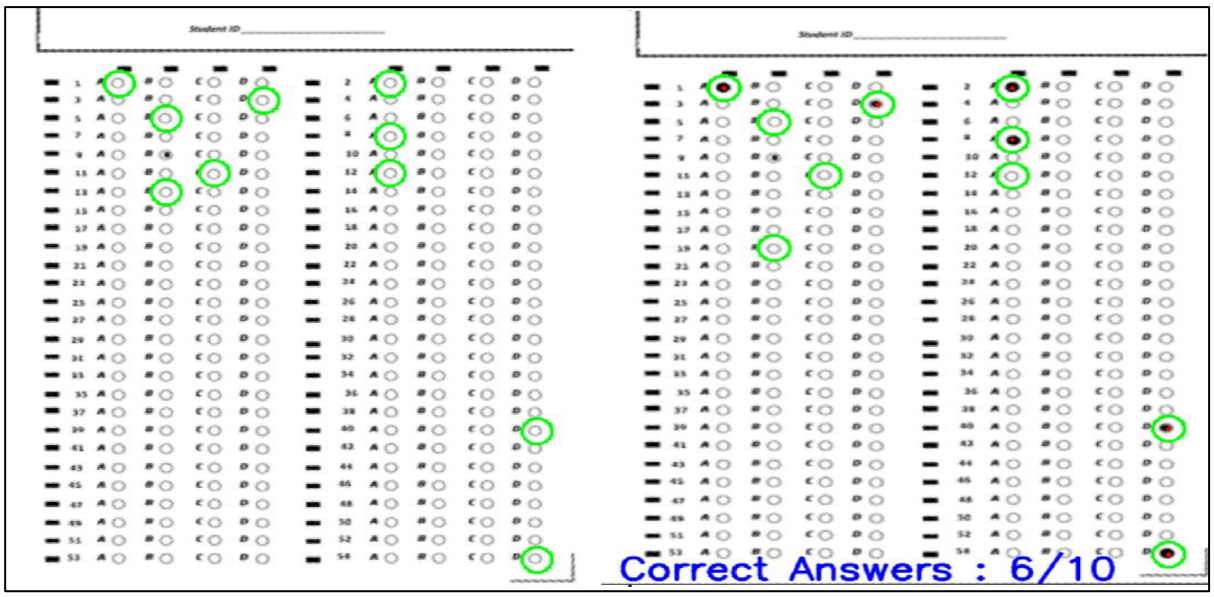


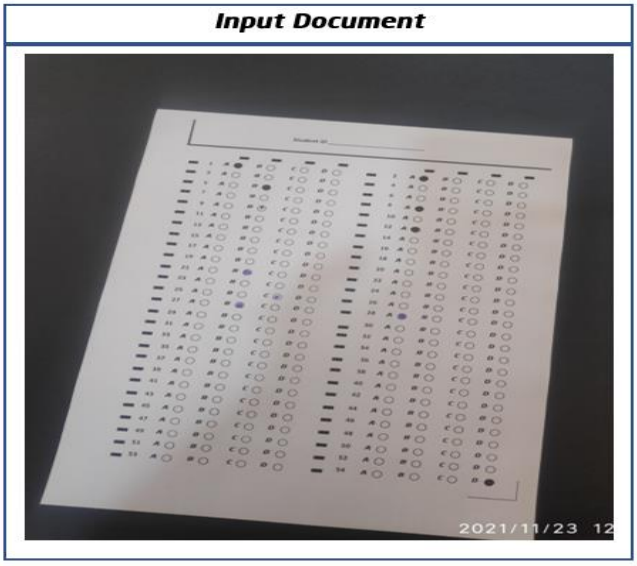
Fig.(7): Left side: Instructors answers and right-side sample of answered sheet

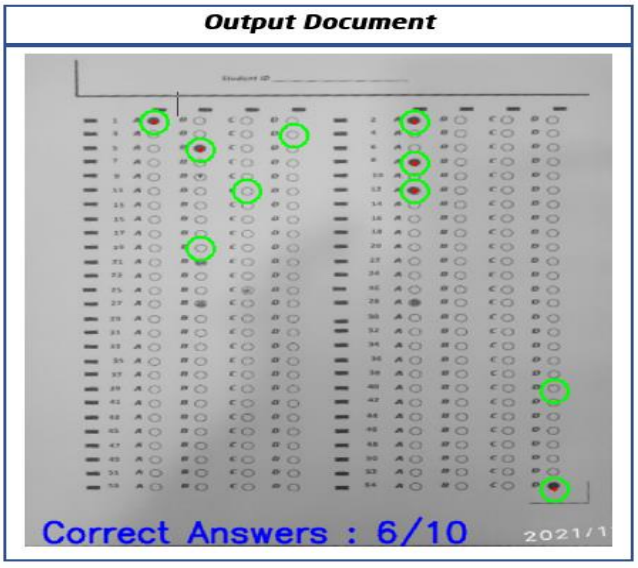
4 TESTING AND EVALUATION

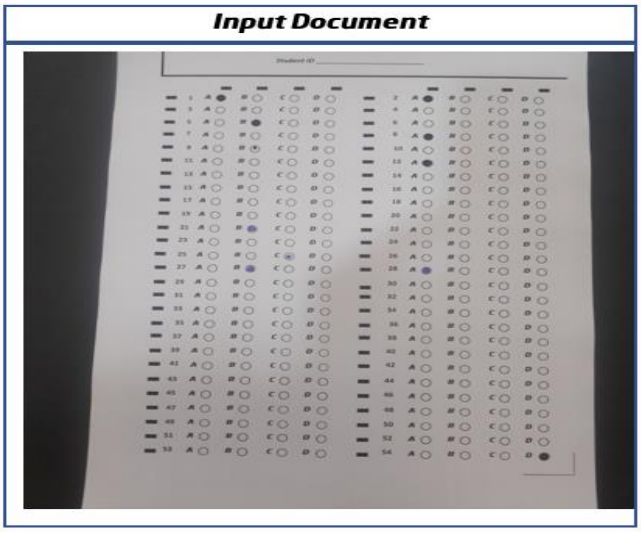
As it shows in Figure 7, that the algorithm can detect student selected options and score that answer with compared to instructors correct answers , in this part, some samples of PNG files that students answer inputted to the algorithm with their results , the algorithm is tested with variant of samples taken from mobile camera , the accuracy is depended on the how much circle is blacked , the sensitivity of algorithm can be increased and decreased by taking number of black pixel on blacked circle , algorithm accuracy is about 99% if prepared paper is used and well-blacked . The Figure 8 is some samples with their outputs, and the following table shows teachers correct answer for each question. Based on teacher's correct answer the samples bellow is evaluated.

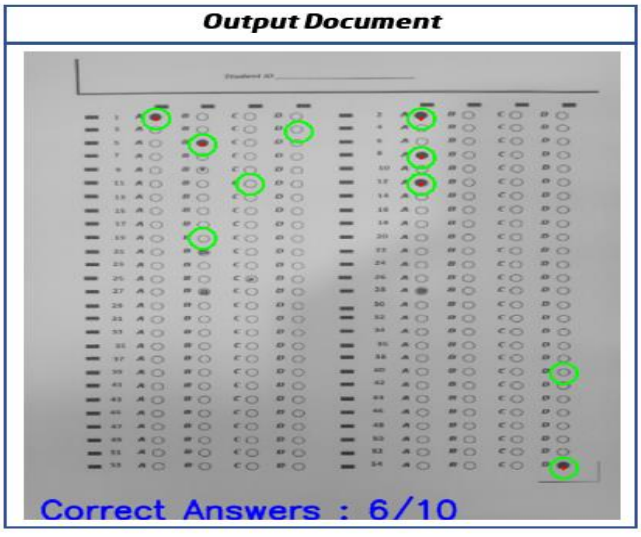
Table (1): Teachers correct answers

Question	correct options
1	1
2	1
3	4
5	2
8	1
11	3
12	1
19	2
40	4
54	4









sardar@dpu.edu.krd

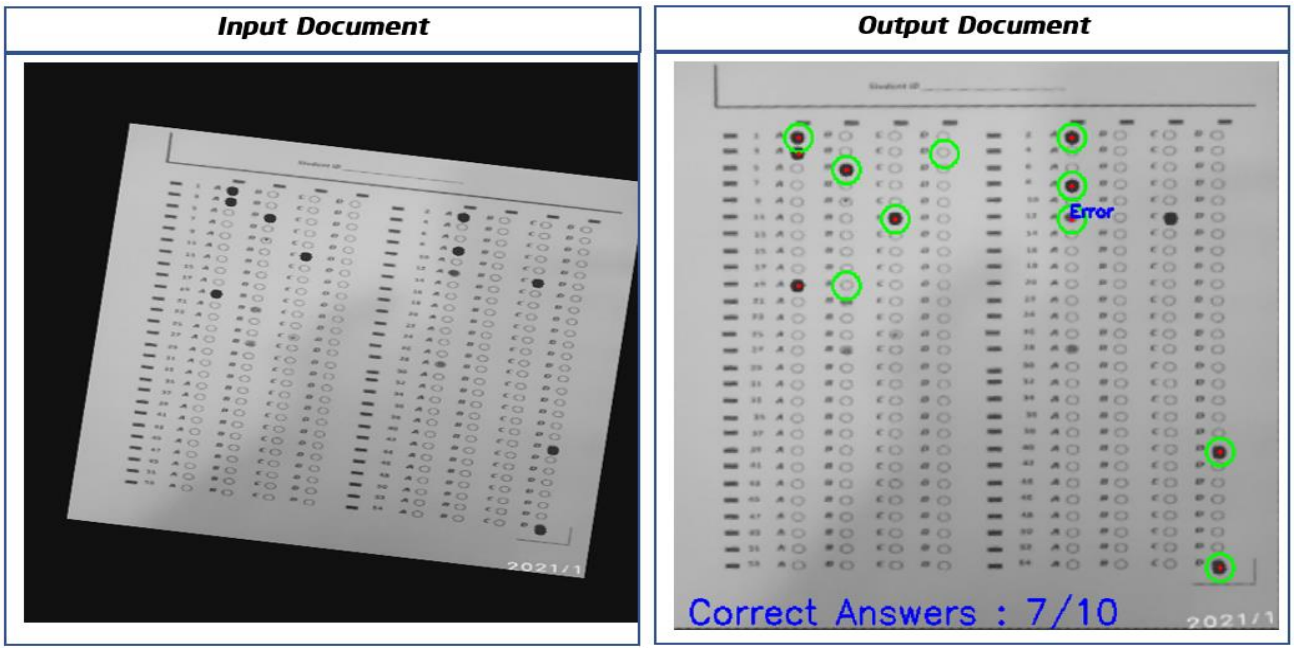


Fig. (8): samples of algorithm inputs and outputs

By looking at the samples above, it can be observed in question number 12 that student who is marked 7/10, two options are blacked, this question is ignored by the algorithm and labeled error.

5 CONCLUSION

The bookmarks position such FQB, QB and OB helps algorithm to find options location in the document. The G-MCQ scans each option locations stored and compared with instructor's correct answer which is inputted to the algorithm and calculate the correct answers, then, put the result on the bottom of paper. In case a circle is not well-blacked, this case could give us unselected option. This can be fixed by reducing number of black pixels for comparison, as a result, algorithm will be more sensitive to detect black-filled circles. Finally, algorithm is tested with variety of samples, and the result of accuracy which are conducted in the test is about 99% in case circles are well-blacked.

REFERENCES

Abbas. (2009). An Automatic System to Grade Multiple Choice Questions paper based exams. *Journal of University of Anbar for Pure Science*, *3*, 174–181.

Barik, D., & Mondal, M. (2010). Object identification for computer vision using image segmentation. *2010 2nd International Conference on Education Technology and Computer*, *2*, V2-170-V2-172. https://doi.org/10.1109/ICETC.2010.5529412

Cardona, M. (2016). Sugar crystals characterization for quality control inspection using digital image processing. *2016 IEEE 36th Central American and Panama Convention (CONCAPAN XXXVI)*, 1–6. https://doi.org/10.1109/CONCAPAN.2016.7942378

Chinnasarn, K., & Rangsanseri, Y. (1999). Image-processing-oriented optical mark reader. *Applications of Digital Image Processing XXII*, *3808*, 702–708. https://doi.org/10.1117/12.365883

de Assis Zampirolli, F., Gonzalez, J. A. Q., & de Oliveira Neves, R. P. (2010). Automatic correction of multiple-choice tests using digital cameras and image processing. *Universidade Federal Do ABC*.

Fisteus, J. A., Pardo, A., & García, N. F. (2013). Grading Multiple Choice Exams with Low-Cost and Portable Computer-Vision Techniques. *Journal of Science Education and*

Technology, *22*(4), 560–571. https://doi.org/10.1007/s10956-012-9414-8

Kulkarni, R., Kulkarni, S., Dabhane, S., Lele, N., & Paswan, R. S. (2019). An Automated Computer Vision Based System for Bottle Cap Fitting Inspection. *2019 Twelfth International Conference on Contemporary Computing (IC3)*, 1–5. https://doi.org/10.1109/IC3.2019.8844942

Sharma, P., Diwakar, M., & Lal, N. (2013). Edge Detection using Moore Neighborhood. *International Journal of Computer Applications*, *61*, 26–30. https://doi.org/10.5120/9910-4506

Spadaccini, A., & Rizzo, V. (2011). A multiple-choice test recognition system based on the gamera framework. *ArXiv Preprint ArXiv:1105.3834*.

Ugwu, E. M., Taylor, O. E., & Nwiabu, N. D. (2022). An Improved Visual Attention Model for Automated Vehicle License Plate Number Recognition Using Computer Vision. *European Journal of Artificial Intelligence and Machine Learning*, *1*(3), Article 3. https://doi.org/10.24018/ejai.2022.1.3.10

Yimyam, W., & Ketcham, M. (2018). The Grading Multiple Choice Tests System via Mobile Phone using Image Processing Technique. *International Journal of Emerging Technologies in Learning (IJET)*, *13*(10), 260–269.