2023 IEEE 6th International Conference on Computer and Communication Engineering Technology (CCET) | 979-8-3503-4068-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/CCET59170.2023.10335127

Bubble-Sheet Assessment Checker with Test Grader using Computer Vision through Raspberry Pi

Marianne G. Wata
School of Electronics, Electrical, and Computer Engineering
Mapúa University
Manila, Philippines

College of Engineering Education
University of Mindanao
Davao, Philippines
mariannewata14@gmail.com

Jocelyn F. Villaverde
School of Electronics, Electrical, and Computer Engineering
Mapúa University
Manila, Philippines
jfvillaverde@mapua.edu.ph

***Abstract*—Manual checking of test papers is still one of the most tedious tasks of being an educator. The immediate feedback of these exam results from the teachers enhances the student's learning experience since it also enables the teachers to identify the knowledge gap in the class efficiently. Thus, this study is mainly designed to develop an intelligent bubble-sheet test checker with a scoring system employing computer vision for automated checking using Python. This system requires the examiner to input the student's name and ID number into the system using the touchscreen monitor, then place the bubble sheet properly inside the chamber. The camera button must be pressed to capture the bubble sheet's image and undergo image processing; after a few seconds, the student's score is displayed. Canny edge detection, image segmentation, and inverse warp perspective are some of the image processing techniques involved in this study. To examine the system's accuracy, confusion matrix analysis and paired t-test are used. The confusion matrix analysis assesses the device's correctness in checking each letter on the shaded bubble sheet, which revealed a high accuracy rate of 96.80%. On the other hand, the paired t-test evaluates the accuracy of automated checking versus manual checking, which resulted in a p-value of 0.01.**

Keywords—bubble-sheet assessment, raspberry pi, computer vision, edge detection, image segmentation

I. Introduction

Learning assessment is recurrently one of the more challenging and arduous tasks in education. During and after a course of study, the instructor and the student must assess how successful they were in attaining the teaching and learning objectives. To do the evaluation, instructors can choose from several examination methods, each with various features. One of these examination formats is the Multiple-Choice Question (MCQ) test, a conventional type because of its credibility, substantiality, and grading efficiency [1]. When this type of evaluation is printed and given in a face-to-face setting, the examinees may encounter different types of forms to record their responses; the examinees could be instructed to write down the letter of their choice on a separate sheet of paper, encircle the letter of the best answer, or shade the circle (bubble) that corresponds your answer in the bubble sheet [2]. Even if MCQ tests are easier to check and grade than other assessment types, manual checking creates a risk of human errors. Teachers still find it an addition to their clerical workload, so they prioritize it the least frequently. Meanwhile, students find it very helpful if their instructors can give them their test results as soon as possible for their feedback and know if they have achieved the learning objectives. Thus, the innovation of a speedy and efficient device to automatically evaluate and score MCQ tests is significant to lessen the educator's burden and give the students prompt recommendations and advice for their learning.

Multiple-Choice Question exams are considered a standardized evaluation of learning, where the examinees must choose the best answer among the given alternatives or choices for a particular question [2]. This type of exam may be presented in various ways, like (a) writing down, (b) encircling, and (c) shading the circle of the letter corresponding to the student's choice [3]. The latter type of MCQ utilizes bubble sheets where examinees can highlight or shade the circle of the letter consistent with their choice using dark ink or pencil. Bubble sheets are commonly employed in summative tests and national examinations since the results can be generated in no time through an automated way of checking using different kinds of technology. One of these technologies is what Rusul and Emad utilized, the Modified Multi-Connect Architecture (MMCA), for their proposed Optical Markup Reading system to store the generated answers of the students that correspond to the correct answers on the answer key [4]. Another novel algorithm presented was the Graphical User Interface (GUI)-based OMR using Java that assisted the examiner in developing and creating their bubble sheets [5]. One study introduced a vertical bar in the middle of an OMR sheet to facilitate the operation of principal component analysis (PCA), which helped eliminate the problem of large tilt and rotation while taking the image of the bubble sheet. Implementing the PCA was deemed to boost the speed of evaluating the OMR sheet [6]. Aside from the mentioned methodologies, a cheaper approach is executed in this study wherein the researcher employs a Raspberry Pi (RPi) and its camera module that serves as the modern OMR Scanner. Raspberry Pi has been utilized in different fields of specialization, including medical [7] [8] [9], agricultural fields [10] [11] [12], and road safety [13]. With canny edge detection and image segmentation of OpenCV, it is easier and quicker for the system to check and score the bubble sheet. Canny edge was also used in several studies, including the detection of Entamoeba histolytica [14], the identification of the scope or edges of the answer sheet [15], the detection of common maize disease [16], and the counting of red blood cells in human urine [17].

The tedious task of manually checking the test papers and the time and effort the instructors consume to finish them are the main motivations of this study. The workload problems most teachers face should be minimized by taking advantage of what the world has in this generation, technology. Based on previous studies, a cost-efficient, user-friendly, speedy, and portable innovation of checking and scoring a bulk of students' examinations is greatly needed to help educators reduce the time they usually spend evaluating their students' works. This system can also give students immediate feedback and recommendation from their instructors.

Moreover, the main objective of this study is to develop a supervised modernistic system that will evaluate the MCQ test answers of the examinees and automatically generate and display the students' scores. It specifically focuses on (a) building a hardware setup and circuitry mainly comprised of Raspberry Pi (Rpi) and Rpi camera module, (b) designing a software system that will accept an answer key, capture the images of the shaded bubble sheets, and compare the answer key and answer sheets for test results using Python, and (c) evaluating its functionality in terms of its accuracy through the use of t-test and confusion matrix.

The output of this study shall be advantageous to the educators, the students, and the researchers. It would lessen the time it takes for the teachers to evaluate the student's work and minimize the burden of the teacher's workloads. This study would benefit students as they could get their examination feedback as soon as possible. Also, this paper is an addition to the body of knowledge that other researchers could use when investigating ways of automated checking summative tests.

This study is limited to assessing bubble sheet multiple-choice answers with 'A,' 'B,' 'C,' and 'D' as the options. The system can read bubble sheets with a maximum number of 100 items, and it can also be initially set to check any number of items, such as 25, 50, or 100-item answers. The answer key of a particular test will be keyed in first before the checking begins for the teacher's convenience. Lastly, this system has no maximum number of bubble sheets to be checked.

II. Materials and Methods

This section thoroughly explains the process implemented in the study involving the development of the improved bubble-sheet assessment checker with a test grader. The study's framework is based on applied research. It will aid the drawbacks of the existing OMR scanner [18] in the market nowadays by making it economical, reliable, and user-friendly using Python programming language and computer vision. Numerous recent studies have shown that Python is the most well-known programming language for Machine Learning and Data Science. Additionally, Computer Vision (CV) is an interdisciplinary scientific field concerned with computer processes in emerging a high-level intelligence of visual data, which has an analogous approach to the human visual system [19]. This field of artificial intelligence has been proven to be effective and reliable by numerous studies [20] [21] [22]. The details expounded in this section are based on various related studies considered from the beginning of this course.

A. *Conceptual Framework*

Fig. 1 shows the conceptual framework of this study, which exhibits the correlation between the three components and will serve as the outline of the course of this research. The entered answer key and the image of the bubble sheet shaded by the students are the required inputs of the system. The answer key is keyed in first through the system's user interface. The photo of the students' bubble sheets is captured by the Raspberry Pi (RPi) camera module. These inputs are then processed through computer vision using image processing algorithms. In addition, the tool's accuracy depends on various parameters, such as ambient light conditions, rotation angle, skewness, camera resolution, and background. Lastly, after comparing the answer key and the student's answer sheet, the exam results, with the corresponding names of the exam-takers, will be displayed in a tabular form through a 7-inch touchscreen monitor display.

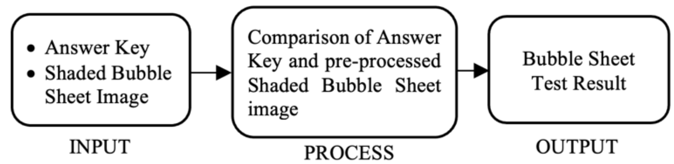


Fig. 1 Conceptual Framework

B. *Hardware Development*

Numerous studies stated that the accuracy of this device is based on a few factors, which include ambient light conditions, camera resolution, and background. Thus, the angle and position of the RPi camera, the distance of the camera from the bubble sheet, and the position of the LED tube lamp were considered before coming up with the final hardware structure.

Fig. 2 illustrates the 3D representation of the chamber where all the essential hardware and circuitry components are placed, whereas Fig. 3 shows the side view of the chamber to emphasize the placement of the camera inside the box. The chamber comprises two sections: the bottom box and the top box. The shaded bubble sheet is positioned on the bottom box, and the RPi camera module and the LED tube lamp are directly placed above it. To capture the entire area of the bubble sheet, the RPi camera module is placed on the center part of the ceiling of the bottom box. The LED tube lamp is positioned properly to avoid too much brightness inside the box. On the other hand, the top box contains the 7-inch LCD touchscreen monitor and the Raspberry Pi. The user interface of the LCD monitor is where the instructor interacts with the system.

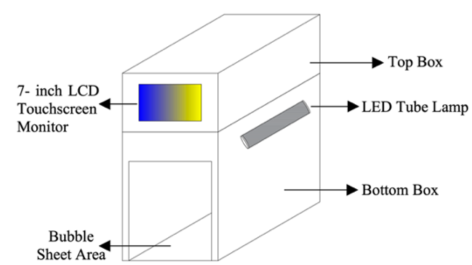


Fig. 2 3D Representation of the Hardware Set-Up

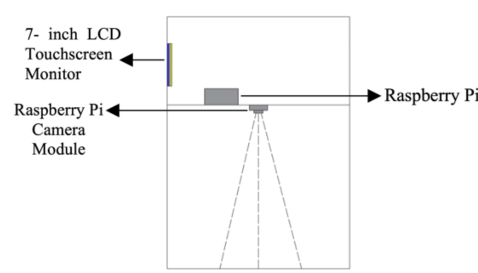


Fig. 3 Camera Set-Up

Additionally, Fig. 4 showcases the block diagram showing every component's connection to the Raspberry Pi. The RPi is considered the central processing unit of the system since all elements pass through it. It requires a 5V-power supply to operate. The user interface and the camera module are necessary to key in the answer key and capture the student's bubble sheet, respectively. These inputs are then processed and compared through image analysis using the Raspberry Pi with the assistance of an SD Card for storage purposes. Conclusively, the score of the students is displayed and saved through the LCD.

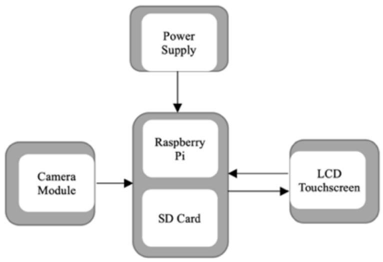


Fig. 4 Block Diagram

C. Software Analysis and Design

The researcher utilized Python programming language to implement computer vision using image processing algorithms. Python provides numerous computer vision frameworks and libraries for automating some tasks, which will be of great help in this study. Exhibited in Fig. 5 is the flowchart of the software system. It is used for the step-by-step procedure to achieve the objectives of the study.

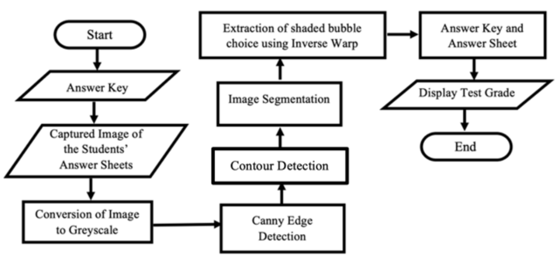


Fig. 5 System Flowchart

The process begins by entering the total number of test items and then inputting the corresponding lettered answers through the user interface. The system's interface was developed using Android Studio and integrated into the Raspberry Pi with the aid of Lineage OS. To initiate the analysis, another crucial input required is a photograph of the examinee's answer sheet, which is captured using the Raspberry Pi camera module through the interface's button.

Fig. 11 illustrates the image pre-processing procedure. Initially, the RGB data of the photo is converted into a greyscale image to accelerate image processing. This conversion reduces the channels from three to one, simplifying the algorithm's operations. The largest contour is also detected and shown in Fig. 11a. The boundaries and edges within the image are subsequently identified using canny edge detection presented in Fig. 11b, enabling precise edge detection. To determine the magnitude and orientation of the detected edges, a contour is applied, allowing for further analysis of the object, which is illustrated in Fig. 11c. Next is the image segmentation using thresholding and warp perspective in Fig. 11d, which is a way to segment the image to create binary images. These binary images are then used to trace the edges to apply the threshold. It will be split into vertical and horizontal groups to achieve better scaling using ROI. Image segmentation is used to split the converted image into quadrants based on a parallel criterion, and similar regions are merged to create the segmented result. The extraction of the shaded bubble sheet is done by utilizing the inverse warp perspective to determine the shaded bubble choices. The values extracted will be saved into a 2-dimensional array and be compared to the standard answer key obtained from the captured test image to calculate the result of the examinee's answer sheet. The comparison is made by submitting the data of the correct choices to the same algorithm of the correct answers on the bubble sheet answer key. There are specific color markings applied while checking the bubble sheets shaded by the examinees; green if the answer is correct and red if it is wrong. These markings are revealed in Fig. 11e. Finally, the results of the examinees will be saved to the database and displayed on the 7-inch touchscreen monitor in a tabular form.

D. Data Gathering and Experimental Procedures

This procedure started by giving orientation to a class of 30 students who will be given three sets of examinations. Set A involved the 25-item test, Set B had 50 items, and Set C contained a 100-item test. All students were asked to use the bubble sheets. During the exam, the students were required to have a Mongol No. 2 pencil with them, and individually, bubble sheets and exam questionnaires were handed to them.

When the examinations were done, the researcher checked the students' answer sheets exclusively. This was done by initially keying in the answer key of a particular exam to the developed system, and then using the Raspberry Pi camera module, the images of the shaded bubble sheets were captured. The GUI of the system enables the checker to input the name and ID number of the students first, and a button must be pressed to capture the photo of the student's bubble sheet and another button to save the result of the examination. After the paper is checked, the image of the bubbles in the sample bubble sheet is marked green and red and displayed on the 7-inch LCD monitor to denote the correct and wrong answers, respectively. The number of bubbles with green marks will be summed up, and the results will be displayed and stored in the database.

Both automated and manual checking was done on the bubble sheets to test the accuracy of the developed system. The score gained through checking done by the system should coincide with the scores in the manual checking. The letters of choice—A, B, C, & D, were also evaluated if they matched the actual letter of the correct answer. Only ten random student bubble sheets per set were picked to evaluate the correctness of the letters of choice. Afterward, the data collected in this experiment goes through statistical treatment.

E. Statistical Analysis

The statistical treatment of this study determines if the algorithm utilized in checking and grading the examinees' bubble sheets is effective. There will be two statistical treatments to be used in this study, the T-test and the confusion matrix. Paired T-test will compare the scores acquired in the manual and automated checking. It is calculated by dividing the difference of group averages by the standard error of difference, shown in (1). In addition, the confusion matrix classification determines the number of accurate predictions of the letter of choices—A, B, C, and D. It is calculated and determined using (2) by summing up all true values extracted from the functionality test and dividing it by the total number of items, multiplied by 100.

$$t = \frac{\overline{x_1} - \overline{x_2}}{\left(\frac{S_d}{\sqrt{n}}\right)} \tag{1}$$

$$Accuracy = \frac{Total\ Items\ Detected}{Total\ Number\ of\ Items} * 100 \tag{2}$$

III. Results and Discussion

The primary goal of this paper is to develop an automated examination-checking portable device with Raspberry Pi to ease the burden of the examiners of checking multiple test papers and returning the examinees' results in less than a week for immediate feedback. Hence, this section reveals the results of the assembled hardware that is appropriate for the produced software system. The test of accuracy for the algorithm used is also illustrated.

A. Graphical User Interface

Educators, both experienced and newbies, are the target market of this improved system. As a result, the researcher aims to produce a user-friendly software system that can be understood in no time. As shown in Fig. 6, the home screen of the software can be described as direct to the point. The "Start", "Records," and "Exit" buttons are displayed on the screen. If the "Start" button is pressed, the interface shown in Fig. 8 is displayed. The "Records" button displays all the saved data of the students, such as their names, ID numbers, and scores. Lastly, the "Exit" button terminates the program.

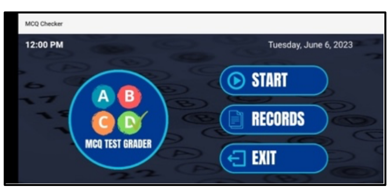


Fig. 6 Home Screen

Fig. 7 presents the interface once the "Start" button is pressed. Two buttons are then displayed— the "Answer Key" and "Process" buttons. Also seen in the interface are spaces or blanks where the user inputs the name and ID number of the student.

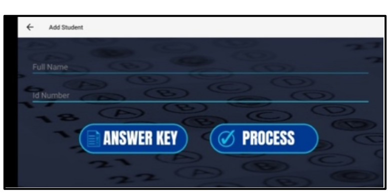


Fig. 7 Add Student Interface

Shown in Fig. 8 is the interface when the "Answer Key" button is pressed. The "Answer Key" button lets the user enter the number of test items to be checked, then the letters of choice- A, B, C, and D, are selected based on the test answer key through their corresponding radio buttons shown in Fig. 9.

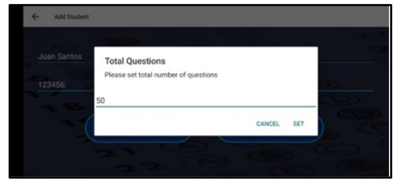


Fig. 8 Set Number of Items

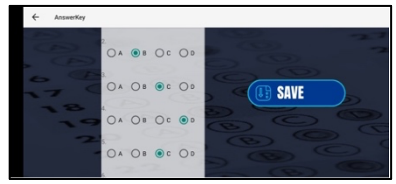
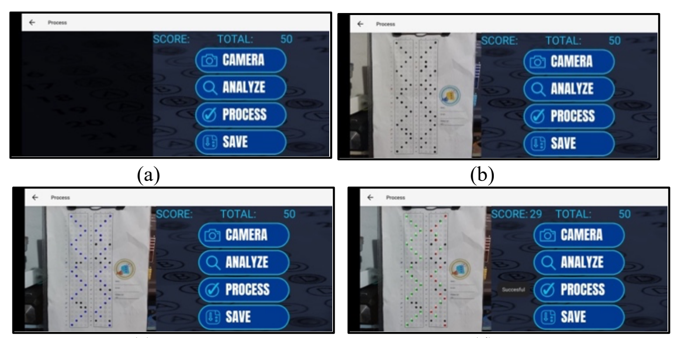


Fig. 9 Answer Key Interface

Furthermore, Fig. 10a is generally the interface where the checking of bubble sheets happens. This is revealed after the "Process" button is pressed. Multiple buttons can be seen in this interface. The "Camera" button is pressed when the user wants to capture the photo of the student's bubble sheet for checking (Fig. 10b). Once the image is captured, the "Analyze" button must be pressed to ensure the system recognizes all numbers and shaded bubbles. All shaded bubbles are marked blue once the analysis is done (Fig. 10c). The "Process" button makes the captured photo undergo image processing until the score is revealed. This button will not work unless the analysis is complete. Green and red marks are on the top of the shaded bubble for correct and incorrect answers, respectively, shown in Fig. 10d. The user can save the student's data to the system for record-keeping using the "Save" button.



(a) (b) (c) (d)

Fig. 10 Bubble Sheet Checking

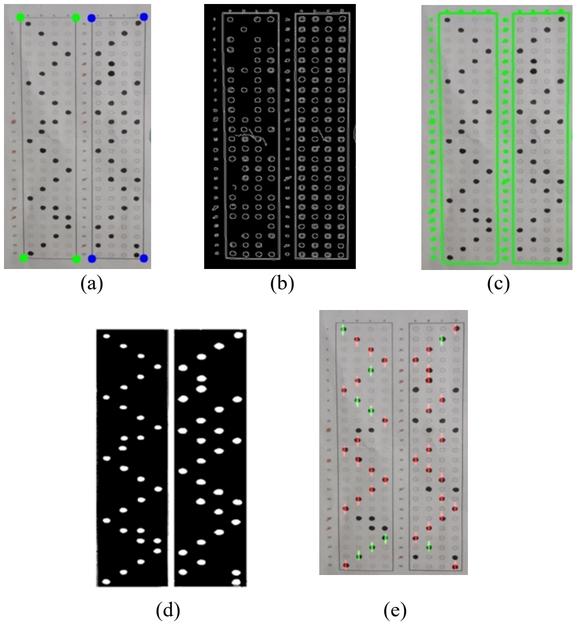


Fig. 11 a.) Original Image with Largest Contour b.) Canny Edge Detection c.) Contour Detection d.) Image Segmentation Mask e.) Test Result

It can be seen in Fig. 11e that particular numbers in a bubble sheet that have more than one shaded circle are ignored by the system since it exceeds the allowed number of answers per number.

Fig. 12 compares the answer key and the student's shaded answer. The values extracted from the image processing are stored in a two-dimensional array. Each letter of the student's shaded answer sheet is represented by numbers, where A, B, C, and D are equivalent to 0, 1, 2, and 3, respectively.

```
first box  [0, 1, 2, 3, 2, 1, 0, 1, 2, 3, 5, 1, 0, 1, 2, 3, 2, 1, 0, 5, 5, 3, 2, 1, 0]
answer key1[0, 1, 2, 3, 2, 1, 0, 1, 2, 3, 2, 1, 0, 1, 2, 3, 2, 1, 0, 1, 2, 3, 2, 1, 0]
second box [3, 2, 1, 0, 1, 1, 5, 2, 1, 5, 5, 2, 1, 0, 1, 2, 5, 2, 3, 2, 1, 0, 1, 5, 3]
answer key2[1, 2, 2, 1, 1, 0, 0, 2, 3, 2, 2, 1, 1, 1, 0, 2, 3, 1, 1, 2, 0, 1, 1, 1, 2]
```

Fig. 12 Comparison of the Answer Key and Student's Answer

B. Hardware Set-Up and Installation

The hardware component of the study's prototype presented in Fig. 13 is fabricated using wood suitable for controlling the lighting inside it. The box is designed with two boxes, the upper box housing the RPi 3 and the touchscreen LCD monitor, and the lower box contains the LED tube for lighting, the RPi camera module, and the platform where the bubble sheet must be placed. An exhaust fan is highly necessary on the upper box for the RPi to avoid overheating, which affects its performance.

Fig. 13 Device Prototype

C. Data Analysis

This study used two means of testing the accuracy of the developed device. One was the confusion matrix which examined the accuracy of the application for checking the letters of choice— A, B, C, and D if it matches the manual checking of the same letters, and the other one was the paired t-test that compared the results of the manual and device checking.

Shown in Table 1 is the confusion matrix analysis of which the overall accuracy was calculated using eq 2. Only ten random shaded bubble sheets for each set were included in this analysis since time was limited to manually check and compare each sheet's letter of choice with the automated checking. Between the manual and automated checking, choice A is checked accurately by the device. Choice B was detected as choice C 4 times out of 510 trials and was not detected 0.98% of the time. Choice C was not detected by the device 2.6% of the time, and it was perceived to be choice D eight times out of 380 trials. Lastly, 7 out of 350 times, choice D was distinguished as choice C by the device, and it was not spotted 22 times.

TABLE I. CONFUSION MATRIX ANALYSIS

		PREDICTED DATA				
		A	B	C	D	NONE
ACTUAL DATA	A	510	0	0	0	0
	B	0	501	4	0	5
	C	0	0	362	8	10
	D	0	0	7	321	22
	NONE	0	0	0	0	0
OVERALL ACCURACY		96.80%				

Shown in Table 2 is the paired t-test result to assess if there is a significant difference between manual and automated bubble sheet checking. The result indicated a significantly small difference between manual and device checking with a p-value of 0.010. The manual checking resulted in a mean of 38 and a standard deviation 16.6. On the other hand, a mean of 37.9 and a standard deviation of 16.6 for the device checking. With these descriptive results and using eq.1, the test statistic equals 2.639.

TABLE II. PAIRED T-TEST

Measure 1		Measure 2	t	df	p	Mean Difference	SE Difference
Manual Checking	-	Device Checking	2.639	89	0.010	0.156	0.059

IV. Conclusion and Future Works

In conclusion, the researcher has succeeded in developing an automated system for checking MCQ tests done on bubble sheets. With the aid of the Raspberry Pi and its camera module, this study has efficiently captured, analyzed, and displayed the test results of the students' bubble sheets by means of Python and OpenCV to implement canny edge detection, image segmentation, specifically thresholding, and warp perspective. Through the confusion matrix, the algorithm got a high accuracy rate of 96.80% in determining the correct shaded letters— A, B, C, and D. A p-value of 0.01 also reveals a small significant difference in the manual and automated checking of the same bubble sheets.

To enhance this study even further, it is recommended to use a higher-spec computer and larger memory capacity for faster processing time. Other researchers may also include the names and ID numbers of students in the image analysis to lessen the hassle of inputting them individually. Lastly, a specific log-in account for each instructor is suggested to separate the data from one user to another for easy access.

References

[1] M. Gierl, O. Bulut, Q. Guo and X. Zhang, "Developing, analyzing, and using distractors for multiple-choice tests in education: a comprehensive review," *Review of Educational Research,* vol. 6, no. 87, pp. 1082-1116, 2017.

[2] H. Ascencio, C. Peña, K. Vasquez, M. Cardona and S. Gutierrez, "Automatic multiple choice test grader using computer vision," *IEEE Mexican Humanitarian Technology Conference,* pp. 65-72, 2021.

[3] F. de Oliveira, "Optimized methodology using Multi-Choice question tests on paper from question authoring to grade publishin," in *3rd International Conference of the Portuguese Society for Engineering Education*, Aveiro, Portugal, 2018.

[4] R. &. A. K. E. Hasan, "An image processing oriented optical mark reader based on modify multi-conect architecture," 2015.

[5] S. Kumar and A. Rathee, "Implementation of OMR technology with the help of ordinary webcam," *International Journal for Modern Trends in Science and Technology,* vol. 6, no. 12, pp. 77-81, 2020.

[6] K. Rachchh and E. Gope, "Inclusion of vertical bar in the OMR sheet for image-based robust and fast OMR evaluation technique using mobile phone camera," *2nd Internationational Conference on Data Engineering and Communication Technology,* vol. 828, pp. 39-46, 2019.

[7] C. Widodo, K. Adi and I. Gunadi, "The use of raspberry pi as a portable medical image processing," *Journal Physics: Conference Series,* 2020.

[8] N. Linsangan, R. Javier, A. Baloloy and I. Villamor, "Portable non-ivasive glucometer using near-infrared sensor and raspberry pi," *3rd Internation Conference on Electrical, Telecommunication, and Computer Engineering,* pp. 35-39, 2020.

[9] C. Hortinela, N. Linsangan, F. Cruz, B. Redosendo, B. Asuncion, C. Leoncio and W. Chung, "Iris recognition using daugman algorithm on raspberry pi," in *EEE Region 10 Conference (TENCON)*, Singapore, 2016.

[10] J. Uy and J. Villaverde, "A durian variety identifier using canny edge and CNN," *7th International Conference onControl Science and Systems Engineering (ICCSSE),* pp. 293-297, 2021.

[11] M. Caya and J. Hong, "Development of an abaca fiber automated grading system using computer vision and deep convolutional neural network," *2021 IEEE Region 10 Conference,* pp. 99-104.

[12] M. Caya, A. Arcagua, D. Cleofas, R. Hornada and A. J. Ballado, "Soil survey implementation using raspberry pi through colorimetric determination of nitrogen, phosphorus, potassium, and ph level," in *ACRS 2019: Progress of Remote Sensing Technology for Smart Future*, South Korea, 2019.

[13] A. Yumang, C. Paglinawan, L. Andrada, E. Garcia and J. Hernandez, "Optimization of vehicle speed calculation on raspberry pi using sparse random projecti," in *International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management*, Baguio, Philippines, 2019.

[14] J. Villaverde, F. Bermudez, M. Palacio and J. Balbin, "Detection of entamoeba histolytica using canny edge and watershed transform," in *International Conference on Biomedical Engineering and Technology*, Tokyo, Japan, 2020.

[15] L. Largo, A. Jancinal, J. Guillermo and M. Wata, "Bubble sheet multiple choice mobile checker with test grader using optical mark recognition (OMR) algorithm," in *5th International Conference on Electronics and Electrical Engineering Technology (EEET)*, Beijing, China, 2022.

[16] M. V. Caya, D. J. Bonifacio, A. M. I. Pascual and J. C. Fausto, "Determination of Common Maize (Zea mays) Disease Detection using gray-level segmentation and edge-detection technique," in *Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, Manila, Philippines, 2020.

[17] M. Caya, D. Padilla, G. Ombay and A. Hernandez, "Detection and counting of red blood cells in human urine using canny edge detection and circle hough transform algorithms," in *Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, Laoag, Philippines, 2019.

[18] P. Jaiswal, "Integrating educational technologies to augment learners' academic achievement," *International Journal of Emerging Technologies in Learning,* vol. 2, no. 15, pp. 145-159, 2020.

[19] D. Sarkar, R. Bali and T. Sharma, Practical machine learning with python, India: APress, 2018.

[21] C. Lumogdang, S. Loyola, M. Wata, R. Angelia and H. Angelia, "Supervised machine learning approach for pork meat freshness identification," in *International Conference on Bioinformatics Research and Applications*, South Kores, 2019.

[22] A. Yumang, C. Manlises, M. Marcelo, A. Adriano and J. Reyes, "Indoor navigation system based on computer vision using CAMShift and D* algorithm for visually impaired," in *International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia, 2016.