

Cost-Efficient Optical Mark Recognition System for Evaluating Answersheets

Dr. S.S.Thakare

Ass. Prof, Dept. of ECE

Government College of Engineering, Amravati.

Dhanashri Vishwas Dhandare

Dept. of ECE

*Government College of Engineering, Amravati.
Amravati, Maharashtra, India*

Gunjan D. Paryani

Dept. of ECE

*Government College of Engineering, Amravati.
Amravati, Maharashtra, India*

Abstract—The idea behind this article is to create a system that can check and evaluate MLSS responses using a webcam. To date, many institutions and organizations have conducted various tests that provide users with separate questions and answers, the answers have multiple options, each option will be a square or circle, and the user has to mark or write with a pen or pencil. These Exams include JEE, CET, NEET etc. Many financial institutions and universities also refer to such exams because QA in MCQ format is helpful and saves time compared to long papers. Universities or famous schools use OMR technology to evaluate this data because there are millions of students and it is not possible to analyze every document accordingly. In terms of making mistakes, for example, people will rate wrong answers as right / right answers as wrong, people will be prejudiced against a student and evaluate him. Keeping records for a long time is difficult because it is affected by many changes in the weather (fire, humidity etc.) and people have to write all scores on different forms/documents. Moreover, higher education institutions use large printers to process and validate all forms, and colleges or universities cannot afford these machines.

Index Terms—OMR (Optical Mark Recognition), Open CV, Grassy thresholding, Binary View, Grayscale Image, Canny Algorithm, Contours.

I. INTRODUCTION

Multiple choice tests around the world have become a measure of student credibility. Many exams are offered not only in India but also in other countries in the form of MCQ, JEE MAINS (Entry Test) conducted by IITs in India, AIPMT (Tag All India Pre-Medical Entrance Examination), MHT conducted by Central Board of Ministries of Education. . -CET is an important exam to use this format for the exam. International exams such as TOEFL and GRE, GMAT are also based on MCQs. The reason the MCQ test is done is because it saves time and measurement, making it a good way to do the test. Many exams administered by the Research Selection and Placement Center or the Ministry of Education have multiple choice questions. All previous work is a process,

and the means of recognizing the response from the optical form is visual perception. OMR is a system for taking notes on multiple choice materials such as exams and tests. OMR technology detects the presence of marks on a piece of paper by sending light and measuring the intensity of light reflected from the circuit. However, he did not know the quality of the character. When conducting national tests, the cost of printing colored test papers and answers, sending them to test centers, sending back optical data and using special equipment to analyze and evaluate this information is prohibited. Using an expensive optical paper scanner will probably be the most expensive. Although there are many different OMR systems, the high cost of purchasing and maintaining these devices makes their application in developing countries difficult. However, optical data analysis can be done and measured faster and cheaper using standard scanners and software. Instead of using a mobile phone or tablet, the authors created a mobile application that allows students to evaluate their answers without using optical devices. The data is qualified, the products are presented as multiple choice with 100 students, analyzed and evaluated. All optical forms are analyzed in less than 20 seconds. This optical data is recognized by four tools, at least 90 using the OpenCV library. It accepts a 50-item multiple choice test definition process. However, nothing is done about the implementation and acceptance of the process. During the OMR process, we compare blank documents with images using the matching pattern. They indicate that the answers are correctly defined. The OMR application was developed with the help of the camera and the Python programming language. Use this application to optically analyze data by visualizing it using four predetermined parameters of the paper.

II. MATERIAL AND METHOD

A. Introduction to softwares used

The OMR system in this study was developed using Python software language and OpenCV library. Python is a programming language developed by Dutch programmer Guido Van Rossum. Python is an advanced programming language that is platform independent, interpretable and interactive. This language does not require a compiler like C, C++ and Java. Intelligence

III. APPLICATION FLOW

A. Image Capturing

The first thing you should do is turn on your laptop's webcam while the webcam is on, then all you have to do is hold down the area in front of the webcam and press the spacebar to capture the image. Please note when performing this task that the area should be placed in front of the webcam and should be placed without affecting the shadow of the page as it can affect the content of the filter water, why shadow will be discussed later when doing this task. a big problem.

B. Visual content

After taking the screenshot, the next task is to extract the appropriate part of the image, because we don't need the whole image. We draw a rectangle over the image and the images inside the rectangle will be extracted. The extracted image will be used later in processing the image as input.

C. Image Processing

After an image has been fed into an algorithm module, the next step is to preprocess it before applying the actual image algorithm to it. The next step is to give the real answer to our algorithm, after transforming the image it will check if the answer is the real answer and accordingly it will show the correct and error and also show the percentage. Here, if the answer is correct, it should be highlighted with a green frame, and if it is wrong, it should be highlighted with a red frame.

D. Recording Results

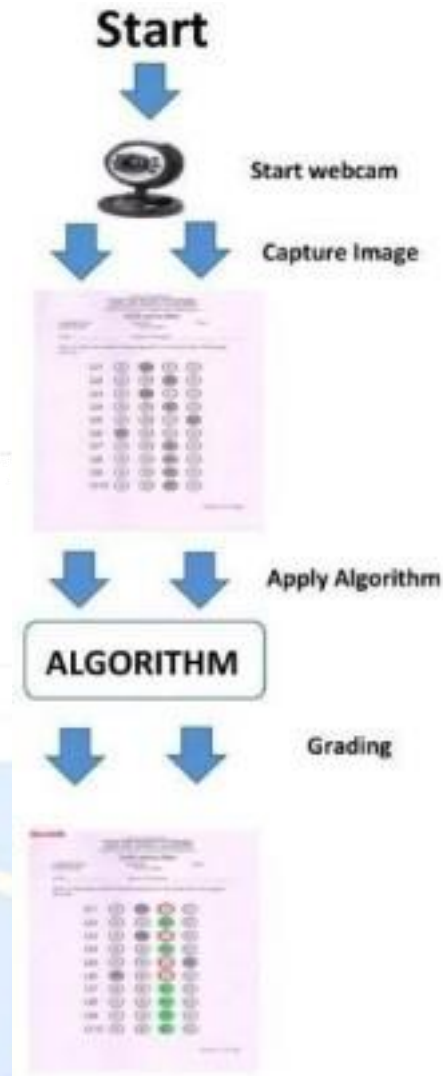
At this stage, after checking each answer sheet, the scores will be saved in an excel file. Filter content and use image processing algorithms.

1. Capture answer sheet image using webcam.
2. Perform extraction of image page.
3. Filter content and use image processing algorithms.
4. Determine the right and wrong and give points.
5. After the assessment, create an Excel sheet and save all scores.

IV. ALGORITHM AND IMPLEMENTATION

A. METHODOLOGY

- Step 1: As shown in Figure 2, the webcam is used to capture the response here, which requires opencv's VideoCapture and NamedWindow functions. Also, press the button to take a screenshot while holding the paper in front of the webcam. This can be done using a NamedWindow which examines the ascii status code to check if the desired button was clicked, and if so, captures the webcam's frame and writes it to the desired location.



- Step 2: The next step is to not need all the images. Answers with almost no background so the image should be cropped. Opencv and main events will be needed here. When the left button is clicked and its function is changed, all controls should be saved, after the button is released, select the junction area to make a rectangle and special crops when discarded. This creates the input for the OMR system.
- Step 3: After filtering the image, the next step is to convert the image to grayscale mode, this will help to easily find markups/highlights made by the user, so he can use the formula shown in it to complete this process.
 1. The post production image will be blurred so that we can reduce the noise which is image degradation. $gray = (red * 0.3 + green * 0.59 + blue * 0.11) (1)$
- Step 4: After converting to grayscale, the next step is to

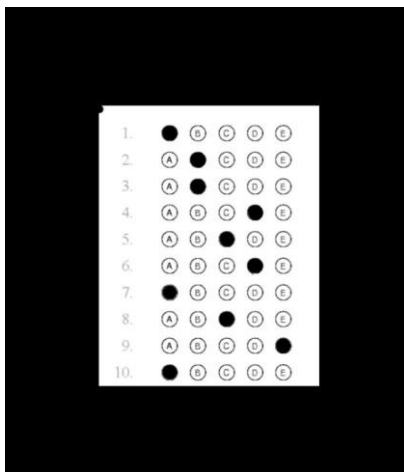


Fig. 2. Answersheet

Courtesy: <https://pyimagesearch.com/>

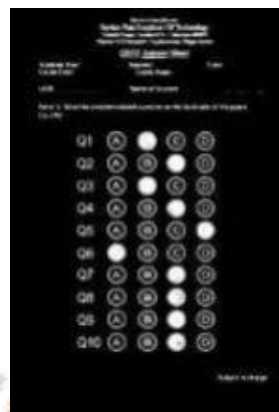


Fig. 3. Gray scale image

Courtesy: www.ijcaonline.org

extract the page from the given image. Use the Canny edge detection technique for this, it helps to find all the edges in the image, this algorithm is a multi-level algorithm, it goes through many steps and creates an image with all the edges (horizontal and vertical line)

- Step 5: After the contours are obtained, the previous image is A transformation is applied to give a bird's-eye view. This is done to get the four points of our test. Find these four points using the cv open library "cv.findContours()". Canny Edge Detection should be used before performing this step. Since the aim at this stage is to search for white patterns from a black background, this conversion can only be done using the previous steps. To get only the vertices instead of the continuous lines as a rectangle (page pattern) beyond the scale, it just strips the vertex points, thus removing the normal message of the points.
- Step 6: Once the contours are received, line them up so that the higher contours are at the front of our sequence and the smaller ones are at the end, the simple intuition behind the logic is that minus 4 Border contours are larger. the contour of the object in the picture from the present. The answer image here should be the main focus in all images.
- Step 7: Use the "fourpointtransform()" function to get a top-down view. This function is used for geometric transformation to extract the page shape from the whole image. It may be necessary to try various getPerspectiveTransform() methods for best results. The reason we do the transformation is to get a 90 degree view of our image.
- Step 8: After the image is binary, display it as shown in Figure 3 to convert the image to color black and white up to 256 gray levels. It is used as preprocessing before any OCR or OMR logic is applied. This can be done using the Otsu transform. As shown in the figure below,

a black and white image is created and binaryzation is done using the open cv's start() function and passing the herbaceous parameter to it.

- Step 9: The next step is to extract the bubble options from the answer page as shown in Figure 5, for this we use reconversion and binaryzation, there are several methods to find the bubbles in the image, here use opencv's HoughCircle() function. This loops through the contours to find all the circles in a left-to-right top-down approach. This saves it in an array where each index of the array has four options/bubble expressions; this means that every line is saved in every index of the array thus resulting in a nested array.
- Step 10: The next step is to do a task that checks if each keyword is correct. First time should reply us first. Now iterate through the contour sequence obtained earlier and compare each selection with our first answer to check if it is correct. But the first step is just to see the structure of the bubble, I still don't know which one is correct. After looking at the image in step 6, you can see that the user has highlighted or marked the selection bubble in white, while the other selections have a black and white border. And how can I find this free balloon? This is done using thresh() in open cv, with its parameters you can define if it is Whether the bubble below is a flag. If the actual answer in the first line is 3, the thing to do in the contour array is to select the first option arrcontour[0][0], if it doesn't move to arrcontour[0][1] check if there is a free balloon to move to the end of the line as there are 4 options or move on to the next option until the entire slot of the inner array [0,1,2,3] is reached. Then move to the next parameter of the bubble contour line[1][0] and repeat the previous step for each parameter. If the threshold of the real and bubble profile match, increase the counter to the real array. Finally, it has the right options for the final price. Here the correct answer is marked with a green

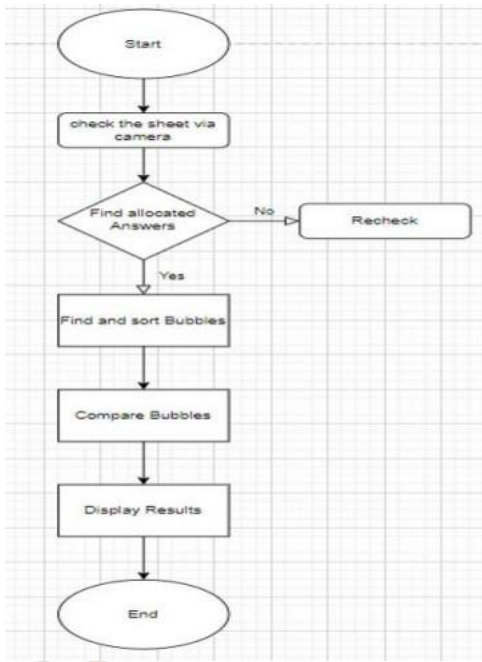


Fig. 4. Algorithm and Implementation
 Courtesy: www.ijariit.com

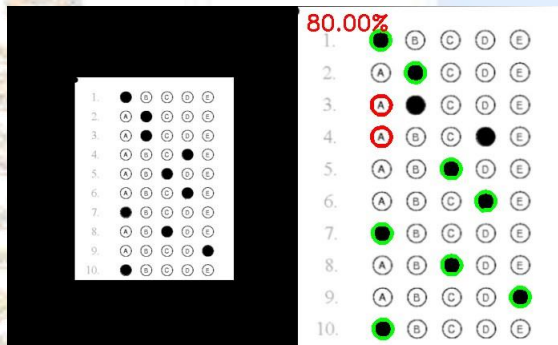


Fig. 5. Finding all bubbles Score

border and the wrong answer with a red border, as shown in Figure 6, which can be done using the drawContour() function of the open cv.

B. Why is the system Cost-Efficient?

When the status of the test is checked, it is forbidden to print the colored test and answer paper, send it to the test center, return the optical data, and use special tools for analyzing and evaluating this information. Using an expensive optical paper scanner will probably be the most expensive. Although there are many different OMR systems, the high cost of purchasing and maintaining these devices makes their application in developing countries difficult. It is known that schools are the places where this technology is used the most. Due to the limited budget of the school, it is very difficult to purchase and use these equipments. However, optical models of diagnosis can be checked and measured more quickly and

more cheaply using standard scanners and software. Here, installing OMR software on devices such as PCs, laptops and using webcams allows organizations to use the software at low cost.

V. EXPERIMENTAL RESULTS

To test the effectiveness of the system, scan the sample answer sheet and input the OMR form. The image is then processed using various visualization techniques and the results are displayed after use. This is done using the Python programming language and the OpenCV library.

VI. CONCLUSION

A cost-effective OMR software solution is proposed in this paper. It tries to overcome many shortcomings of the traditional OMR system. The system checks for different identifiers. In large groups, it reduces the time required to create system tests. The system solves the problem of spending more time calculating scores and gives teachers more time to prepare lessons and assist students by using software tools that streamline the assessment process. In addition, the system provides different benefits to groups of different sizes. It reduces the time spent sorting, evaluating answers and writing scores.

VII. REFERENCES

- [1] Taware, D. D., Kulkarni, A. V. (2020). Optical Mark Recognition Using Python and OpenCV. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.
- [2] Sharma, A., Goyal, S. (2020). Optical Mark Recognition (OMR) using Python and OpenCV. In 2020 6th International Conference on Signal Processing, Communication and Computational Research (SPCCR) (pp. 1-5). IEEE.
- [3]Gupta, D., Swaroop, A. (2021). OMR Sheet Processing using OpenCV and Python. In 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 33-37). IEEE.
- [4] Singh, N., Thakur, S. (2019). Optical Mark Recognition (OMR) System using Python and OpenCV. In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 515-519). IEEE.
- [5]Pandey, S., Sinha, P. (2019). OMR Sheet Scanning and Evaluation using OpenCV and Python. In 2019 4th International Conference on Internet of Things and Connected Technologies (ICIOTCT) (pp. 758-763). IEEE.
- [6]Pawar, S., Goswami, S. (2019). OMR Sheet Reading using Python and OpenCV. In 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.
- [7]Chirag, S., Kumar, V. (2018). OMR Sheet Scanning and Evaluation using Python and OpenCV. In 2018 IEEE Calcutta Conference (CALCON) (pp. 88-93). IEEE.
- [8]S. B. Gaikwad, "Image Processing Based OMR Sheet Scanning," International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), vol. 4, no. 3, pp. 519-522, 2015.

- [9]] K. Atal, A. Arora et al., "Cost effective optical mark reader." International Journal of Computer Science and Artificial Intelligence, vol. 3, no. 2, 2013
- [10] Parul, H. Monga, and M. Kaur, "A novel optical mark recognition technique based on biogeography based optimization," International Journal of Information Technology and Knowledge Management, vol. 5(2), pp. 331-333, 2012.
- [11] S. B. Gaikwad, "Image Processing Based OMR Sheet Scanning," International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), vol. 4, no. 3, pp. 519-522, 2015.
- [13]G. Krishna, R. H. Ram, I. Madan, Kashif, and N. Sahu, "Implementation of OMR Technology with the Help of Ordinary Scanner," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 4, pp. 714- 719, 2013.
- [14]I. A. Belag, Y. G. İltepe, and T. M. Elmalti, "An Image Processing Based Optical Mark Recognition with the Help of Scanner," International Journal of Engineering Innovation Research, vol. 7, no. 2, 2018.
- [15]Y. S. S. S. Reddy, A. S. Srinivas, and G. M. Krishna, "OMR Evaluation using Image Processing," International Journal of Innovations Advancement in Computer Science, vol. 7, no. 4, 2018
- [16] N. Kakade and R. C. Jaiswal, "OMR Sheet Evaluation Using Image Processing," Journal of Emerging Technologies and Innovative Research vol. 4, no. 12, pp. 640-643, 2017.
- [17] Learning OpenCV-Computer Vision with the OpenCV Library- Gary Bradski, Adrian Kaehler
- [18] High-Level Computer Vision Using OpenCVGraphics, Patterns and Images Tutorials (SIBGRAPI-T), 2011 24th SIBGRAPI Conference on 28-30 Aug. 2011
- [19] OpenCV implementation optimized for a cell broadband engine processor- Sugano H ; Dept.of Communication and Computer Engineering, Kyoto Univ, Kyoto; Miyamoto, R.
- [20] Digital Image Processing (3rd Edition) -Rafael C. Gonzalez