IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 5, May 2015.
www.ijiset.com
ISSN 2348 – 7968

OMR Auto Grading System

Nithin T.
Research Asst. of Information Technology
Army Institute of Technology
Pune, Maharashtra, India
nithint_11484@aitpune.edu.in

T. Raj Shekhar
Research Asst. of Information Technology
Army Institute of Technology
Pune, Maharashtra, India
t.rajshekhar_11684@aitpune.edu.in

Md Nasim
Research Asst. of Information Technology
Army Institute of Technology
Pune, Maharashtra, India
mdnasim_11720@aitpune.edu.in

Omendra Singh Gautam
Research Asst. of Information Technology
Army Institute of Technology
Pune, Maharashtra, India
omendrsinghgautam_11667@aitpune.edu.in

Yuraj Gholap
Professor, Department of Information Technology
Army Institute of Technology
Pune, Maharashtra, India
ygholap@gmail.com

Abstract-
The project is based on an idea to grade an OMR sheet using a mobile utilizing the android platform.Even today large numberof institutes and colleges implement the idea of OMR sheet for evaluation of students based on multiple choice questions. Most of the standardized tests also use the same. Big institutes use expensive OMR software along with the machines associated to evaluate the OMR sheets. But all those 'not so rich' institutes and individual teachers would not have the financial credibility to afford the costly setup and would have to manage it using manpower. So our idea is to use a simple mobile application with simple interface that would be understandable to all. First we will discuss the OpenCV implementation of the image-processing algorithm and report on its challenges. Second we will give an overview of what was implemented on the android phone. Finally we will summarize the challenges and possibilities for future work.

Keywords:image-processing, OMR, homography, key-point detection

I. INTRODUCTION

In India, the multiple-choice questions have become an important part of the educational system. Important tests also use multiple- choice questions to judge the student's academic performance. The Indian Institute of Technology (IIT), the JEE MAINS and All Indian Pre-Medical Entrance Test(AIPMT) are just some of the many important tests conducted in INDIA to get the students on a common stage. Every year millions of students take these tests and they have toanswer various questions asked by highlighting the circles in OMR sheets. Current situation for evaluating these OMR sheets are veryoverpriced and need dedicated scanner, OMR software and buying customized OMR sheets. So small organizations, institutes, individual teachers and tutors cannot use this easy method of grading without spending lot of money. They resort to manually grading answer sheets. To check the OMR sheet test responses of a student takes 10 minutes on an average. The main theme of our project comes from the idea that we could build a mobile application that will assist the instructors in auto grading these answer sheets and saving their precious time. Now a day's, smartphones are very common. So there will be no extra cost associated with using this smartphone-based solution. And it is very easy to use without lot of steps or setup.

We have a possibility that our application would do the following activities:

(i) automatically checks the answer sheet in the photo,

(ii) detect responses for each question and

(iii) Compare student responses with reference answers.

ISSN 2348 – 7968

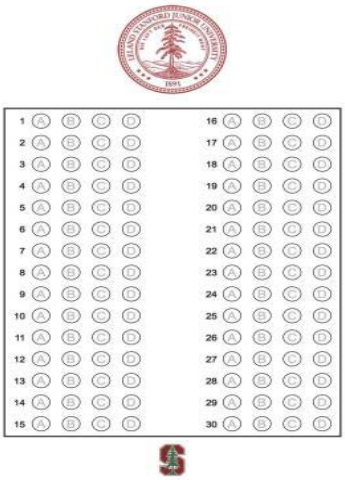

Fig. 1 Answer Sheet Format

We have made our algorithm assuming that we know the position of bubbles with respect to each question.The report is organized in parts as discussed. In Part II we provide the block diagram and explain the running sequence of our application. Part III will provide information about the Android implementation of our algorithm. Experimental results about our algorithm are mentioned in Part IV. Finally, Part V summarizes our results and lists possible improvements to our application.

II. APPLICATION FLOW

Fig. 2 shows the complete application flow. This section will describe how the whole application will works inside android smart phone. Inside this application when the main page appears on the screen there are three buttons. The first button named capture reference answer sheet, it is used to take the picture of the reference answer sheet and extract the correct answers from the image. The second button named capture student answersheet. It is used to take the picture of the student answer sheet and extract student responses to each question. The third and last button on the main page of the application called view result is used to compare the correct reference answers with student's answers, and generate the result table containing highlighted correct and wrong answers with different colours .

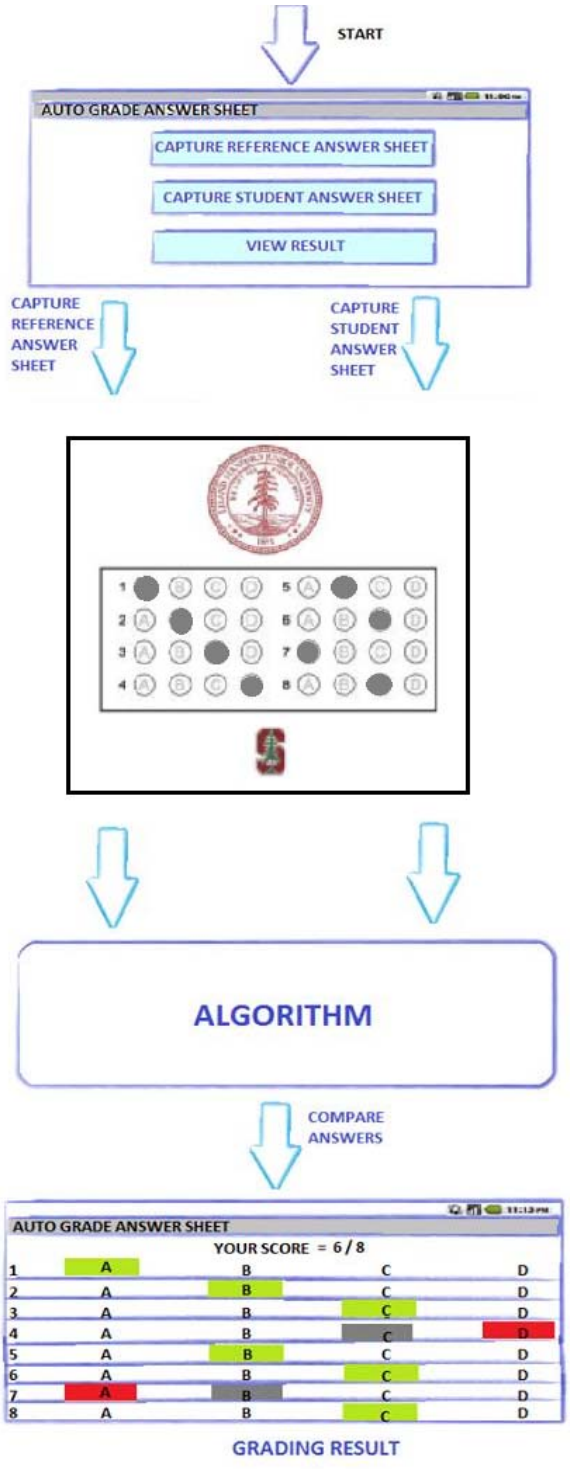

Figure 2. Application Flow

Fig. 2Application Flow

ISSN 2348 – 7968

A. Capture Reference Answer Sheet

This application will perform its functions when the picture is clicked using android smart phone camera. When the capture reference answer sheet button is pressed the application will use the camera of android smart phone to click a clear image of reference answer sheet with correct answers. To extract the correct answers from the clicked image, our main image processing algorithm is invoked. A list data structure is used to store the extracted correct answers which is used later on, when result is generated by view result button.

B. Capture Student Answer Sheet

The basic function performed by this button is almost same as Capture Reference Answer Sheet, except that the taken image is of student's answer sheet. Again a list data structure is used to store the extracted student answers from the clicked image using image processing algorithm.

C. View Result

There are two list data structures involved:-

(i) The list data structure which contain the correct answers from reference answer sheet and

(ii) The list data structure which contain correct as well as wrong answers from the student answer sheet

Both these are compared to generate a graded result table which will show correct answers and wrong answers with different colour highlighting.

(i) Correct answers are highlighted in green.

(ii) Incorrect answers are shown in red and correct answer for it is shown in grey colour.

After the answers are displayed,more answer sheets can be processed without taking the snapshots os reference answer sheets.

III. IMPLIMENATATION AND ALGORITHM

The algorithm is basically done in two stages:-

(i) The first stage include registration ofimage and its alignment according to a template image and later computing the transformation matrix using that.

(ii) Using the transformation matrix to locate the center pixels of all answer circles and analyzing whether the circle is filled or not.

A) Homography and registration of image

We first find the homography of the template image and the answer sheet to get the transformation matrix.Using the transformation matrix,we get the center point of answer circles in the answer sheet. The image can be registered using the following five steps.

Step 1

In this step,the key points of template image and answer sheet are detected. We have used the SURF detector as it seemed better than other existing detectors present. The SURF detector detects more features compared to its counterparts.an integer approximation is used to the determinant of Hessian blob detector, which can be computed very fast with an integral image

Step 2

In this step,descriptors for the image are computed using the key points which were found in the previous step.

Step 3

In this step,key points of the image captured is matched with the key points of the template. We have used Fast Library for Approximate Nearest Neighbors(FLANN) matcher to do the same.to discard the all the bad results,NearestNeighbor. Distance Ratio is used.in this,the minimum distance of the match is found at first. Then this distance is multiplied by a ratio to get a number. If the distance is more than this number,then that distance is considered as a bad result and is thus discarded.

Step 4

After the good matches are found,homography of the image is found to get the transformation matrix.

ISSN 2348 – 7968

Step 5

The transformation matrix is used to finally locate the center points of the answer circles. This center point location isprojected on the answer sheet to find whether answer is right or not.

B) Answer extraction

Step 6

The inverted binary image is used to count the white pixel around the center of each circle.in this step,the inverted binary image is obtained by using the threshold and applying the same to the original image.

Step 7

Here,a threshold number of white pixels is used to determine whether the circle is filled or not. If the number of white pixels are more than the threshold,then the circle is not filled otherwise not.

IV. RESULTS AND EXPERIMENTS

The quality of image is bound to be affected by the surrounding conditions. The images can be of poor illumination or may be affected by different orientations.The results of the experiment conducted in different conditions are given below.

(i) Images taken at the best possible alignment gives almost 100% result.

(ii) When the images are tilted without any rotation,the result still holds good but less than the first case.

(iii) When the images are taken with large tilt and rotation, 74% result came out to be accurate.

Rotation/Tilt Angle	Average Accuracy	Standard deviation
Flat/No rotation	100%	0
Flat/90 degrees	92%	0.15
Tilt/No rotation	85%	0.12
Flat/30 degrees	79%	0.3
Random tilt and	74%	0.33

TABLE I. WE SUMMARIZED OUR EXPERIMENT RESULT ON IMAGES WITH DIFFERENT ROTATION ANGLE AND TILT. WE ITERATE 20 TIMES FOR EACH CASE, AND CALCULATE AVERAGE AND STANDARD DEVIATION.

V. FUTURE WORK

Current algorithm only uses FLANN matcher to find key-point correspondence, and this turned out to be not optimal, because it does not take into account any geometry information around key-point. By introducing algorithms that uses geometry information around key-points, we expect to see more robust result even with large rotation. Our algorithm currently takes ~8 sec for processing a captured image. We can reduce this time by adjusting various parameters of SURF key point extraction. But we could not do these adjustments since we used Java wrapper for OpenCV. It does not have the flexibility offered by native OpenCV code. Current answer sheet format does not have student and test identification feature. We can use the same approach of filled circle detection for detecting test id and student id.

Advantages and Disadvantages

Advantage:-

1. Low cost of software used.
2. Faster response time.

Disadvantage:-

1. Image needto be taken in a specific orientation.

VI. ACKNOWLEDGEMENT

We are very thankful to the administration of Army Institute of Technology, Pune, India for providing all those resources which has helped us a lot to come up with this idea. We are very thankful to Asst. Prof of Information Technology, Prof. YurajGholap for his constant support. We are obliged to HOD Dr. SangeetaJhadav for her invaluable help and advice regarding the Android implementation of our algorithm.

VII. CONCLUSION

We have used the OpenCV,which is a library exclusively made for image processing. Although the most commonly used language is C++ for openCV,we have used openCV java to make the app. We have also used the android sdk for the same. This application brings in a lot of relief for small institutions and make the options based exams more regular.

ISSN 2348 – 7968

As we are implementing this on android,there is absolutely no case of any power consumption or energy wastage.This project along with the help of android platform will bring OMR sheet processing a drastic change where in it can bedone without expensive scanners. We hope that this application will help in replacing the expensive scanners in various image processing chores.

VIII. REFERENCES

Fast Approximate Nearest Neighbor search

[1] http://docs.opencv.org/2.4.9/modules/flann/doc/flann_fast_approximate_nearest_neighbor_search.html

[2] Digital Image Processing (3rd Edition) -Rafael C. Gonzalez

[3] Learning OpenCV-Computer Vision with the OpenCV Library- Gary Bradski, Adrian Kaehler

[4] High Level Computer Vision Using OpenCV-Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2011 24th SIBGRAPI Conference on 28-30 Aug. 2011

[5] Opencv implementation optimized for a cell broadband engine processor- Sugano, H. ; Dept. of Commun. & Comput. Eng., Kyoto Univ., Kyoto ; Miyamoto, R.

[6] B. Girod. “EE 368 Digital Image Processing Notes”. EE 368 Digital Image Processing. Spring 2011-2012

[7] http://morf.lv/modules.php?name=tutorials&lasit=2

Features2D + Homography to find a known object,

[8] http://docs.opencv.org/doc/tutorials/features2d/feature_homography/feature_homography.html#feature-homography