

## Implementation of OMR Technology

Sandeep kumar<sup>1</sup>, Dr. Anu Rathee<sup>2</sup>

<sup>1</sup>B. Tech Scholar, Department of IT, Maharaja Agrasen Institute of Technology, Delhi, India

<sup>2</sup>Assistant Professor, Department of IT, Maharaja Agrasen Institute of Technology, Delhi, India

\*\*\*

**Abstract** - In this modern time, people don't want to invest their time in something processing. They just want to give an input and take an output immediately. This project is actually an application that we are developing which enables implementation of OMR using an ordinary WebCam. It provides tools to the user to check and calculate on the base of layout they want. The user can give Input through Webcam of system and take correct checked calculate output, distribute it among others from whom information is desired.. The scanned image files will then be provided as input to the software, processing will be done, value of filled fields will be extracted and then the data will be manipulated as instructed by the user. The implementation is done using OpenCv and python. We are using Pycharm IDE so as to easily manage the graphical layout and all the resources. For the GUI Python was chosen mainly because of its platform independence and lightweight of the components in python.

**KEYWORDS:** *OpenCv, Python, Contours, Edge, Grey Scale.*

### I. INTRODUCTION

Optical Mark Reader (OMR), also called "mark sensing", is a method of scanning technology in which data is input via marks made in predefined positions on a form and entering data into a computer system. Therefore, OMR is best for handling discrete data, where values fall into a limited number of values. For examples, gender, occupation, religion, etc.

OMR is a technology that detects the absence or presence of a mark, but not the shape of the mark. OMR software interprets the output from the Webcam or scanner, and translates it into the desired ASCII output. Forms are scanned through an Webcam or OMR scanner. The forms contain small circles, referred to as bubbles, or boxes that are filled in by the respondent. Optical Mark Reader (OMR) reads marks written by pencil or ballpoint pen in the pre-defined positions on the questionnaire sheet. The OMR can judge the existence of written marks by recognizing their depth (darkness) on the sheet.

The simple model of thinking was applied, this work was intended to develop through three phases: (1) Input, (2) Process and (3) Output. First phase, there is only a module for covering the key objective of this phase that is an arrangement file system in searching and selecting the desired working folder and a file assigned to be the key answer sheet. Secondly, Process phase consists of two modules: recognizing and analysis. Recognizing module is the critical of whole program carrying the procedure that translates the image files to the digital data. The last Phase, called Output, responses in facilitating the end user to provide the necessary report. This is the simplest phase that follows the principle of data structure process.

A mark is response position on the questionnaire that is filled by survey collectors or respondents. If the position is filled (marked) then the scanner will registers significantly less light than the surrounding questionnaire. In order to be detected by a scanner, a mark has to be positioned correctly on the questionnaire and be significantly darker than surrounding of the questionnaire. Since most scanners use infrared light, an acceptable

## II. PROS & CONS OF OMR SHEET READER

### Pros:

- A fast method of inputting large amounts of data – up to 10,000 forms can be read per hour based on the quality of the machine used.
- Only one computer required to collect and process the data
- OMR has a better recognition rate than OCR because fewer mistakes are made by machines to read marks than by reading handwritten characters.
- The cost of inputting data and the chance of data input errors can be trim down because it is not necessary to type the details for data entry.
- OMR is much more accurate than data being keyed in by a person

### Cons:

- Documents for optical mark recognition are complicated to design.
- If the marks don't fill the space completely, or aren't in a dark enough pencil, they may not be read correctly
- The OMR reader needs to be reprogrammed for each new document design.
- OMR readers are relatively slow.
- The person placing marks on the documents must follow the instructions exactly.
- Any folding or dirt on a form may prevent the form being read correctly.
- Only suitable for entering one out of a selection of answers, not suitable for text input
- The OMR reader requires the answers to be on the prepared forms which will all be identical to one another. You can't just lift up a blank sheet of paper and mark your answers on it.

## III. VARIOUS OMR TECHNIQUES

There are some techniques mentioned which are as under;

### A. OMR WITH ORDINARY SCANNER

The rule of this evaluation is it will compare the given scanned image to its previously stored template. And then mark the answer sheet on the basis of the template and given criteria by the user. So here is an idea of system which make easier the OMR sheet evaluation technique which must be viable and efficien. For that system includes the main following modules:

- i) Answer Feed Module
- ii) Criteria Defining Module
- iii) Assessment Module
- iv) Result Storage Module
- v) Publish Result Module In order

to take the best output from the system the scanning should be performed very carefully and the image should not be tilted . It is prior that the scanned image is well and error free. According to system, four types of feasibility studies can be considered:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility
- Schedule Feasibility

### B. MULTI-CORE PROCESSORS FOR CAMERA BASED OMR

At present most of desktops, laptops, tablets, and even smart phones are shipped with multi-core processors. The efficient utilization of multi-core processors computation power can't be achieved by developing traditional applications with sequential algorithms. Parallel algorithms utilize the capabilities of these processors. They are very well suited for parallel processing. This work represents a low cost and fast solution for optical mark recognition system working in multi-core processor system .In this system a solution for camera based OMR is presented. This system turns on special designs of the answer sheet to add some marks which speed up the detection of bubbles. The system is insensitive to rotation scaling and illumination variations. In addition to that the flipped images can be processed and recognized without correction. The solution keep out of the way of heavy computational algorithms such as skew correction, circle detection and Hough transform, to increase the speed of the system .

### C. COMPUTER VISION BASED OMR SHEET EVALUATION USING OPENCV

The main aim of this work is completely removing the ordinary scanners by making use of a web camera as an input device of the OMR sheet. An OMR sheet is placed in front of webcam and the program takes its image. Then the program which is developed in OPENCV libraries, the open source C libraries used worldwide, further processes that image to extract the optical marks. This extraction includes several steps of image processing. OpenCV is the powerful tool used for image processing. OpenCV (Open Source Computer Vision Library) libraries are image processing and computer vision C libraries developed by Intel. OpenCV runs on Windows, Android, Linux and OS X. Open source computer vision library is providing functions required to run the webcam. It used the captured image after saving it and then loading it for further processing. Bloodshed DevC++ IDE is used for programming in C which can be easily configured to call the functions of the OpenCV. Operating system used is Microsoft Windows 7.

**Image Thresholding:** Thresholding is the simplest form of image segmentation and is used to create binary images. The images are in black and white. The black part is undetected part and the white part is detected part.

**Image Gridding and Division:** Image gridding involves making a grid as shown in the image. The grid will be drawn over the image in such a manner that each square or rectangle in the grid contains an optical mark or the black dot. After this we distinguish each black dot according to the rectangle in which it is contained. The grid is made to work even if the sheet is not at a fixed distance from the webcam and also works according to ratio of the distance. The grid adjusts itself. Now the grid is used to split up the problem into four major rectangles which can be processed separately. The first two columns hold thirteen questions and the last two hold twelve questions which makes hard to find a single large rectangle and then process it. Four different rectangles each for different set of questions make it easier to keep track of the small rectangles holding the black marks and thus it can be easily solved. Now that we have the holding rectangle of each of the black mark we will use IMAGE DIVISION to split up the image into separate rectangles for each mark. If we handle the first rectangle and cover it row wise we will first split up for all columns of first row, then second, then third and so on till the whole rectangle is split up. Consequently all the rectangles are split up. The software may become heavy and difficult to manage while working on so many images but OpenCV supplies with functions to determine whether to show, hide, create and put out these images after they serve their purpose. Also while we develop a professional version for this technique, it can be advance optimized.

#### IV. METHODOLOGY

Edge detection and contours are used to locate points in images where the color or brightness have great changes.

##### Canny

Canny Edge Detection is an algorithm used to extract edges from images, and since it looks quite straightforward, I believe we can start with it.

The algorithm has four stages:

- **First** — Performs noise reduction with a Gaussian Blur;
- **Second** — Gets the gradient direction and magnitude with a [Sobel kernel](#);
- **Third** — Applies non-maximum suppression, which removes unwanted pixels that are not part of a contour;

- **Fourth** — Applies the Hysteresis Thresholding that uses min and max values to filter the contours by the intensity gradient.

Even though it doesn't look so easy, the implementation of Canny detection is quite comfortable with OpenCV.

We start by importing the libraries.

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
```

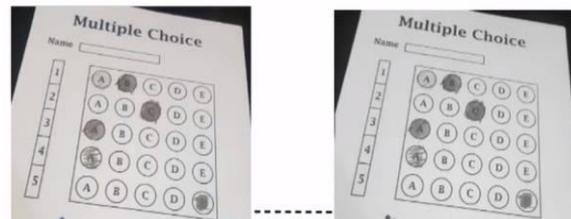
Now we can have a look at the image we want to extract the contours.

```
image = cv2.imread('Images.jpg')
image = cv2.cvtColor(image,cv2.COLOR_BGR2RGB)

fig, ax = plt.subplots(1, figsize=(12,8))
plt.imshow(image)
```

Original

Gray Scale



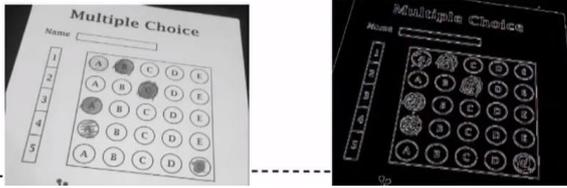
To use Canny, we need to convert our image to grayscale, then use the method .canny with the image, and the min and max thresholds.

I suggest that you experiment with those thresholds to see how the result changes.

```
g = cv2.cvtColor(image,cv2.COLOR_RGB2GRAY)
edge = cv2.Canny(g, 60, 180)fig, ax = plt.subplots(1, figsize=(12,8))
plt.imshow(edge, cmap='Greys')
```

Gray Scale

Edges



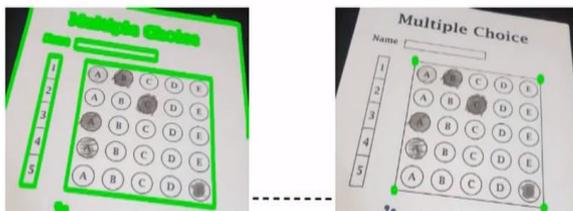
Find Contours and draw Contours, to get and plot the external contours found with Canny detection.

```
contours = cv2.findContours(edge, cv2.RETR_EXTERNAL,
```

```
cv2.CHAIN_APPROX_NONE)cv2.drawContours(image, contours[0], -1, (0,0,255), thickness = 2)fig, ax = plt.subplots(1, figsize=(12,8)) plt.imshow(image)
```

Contours

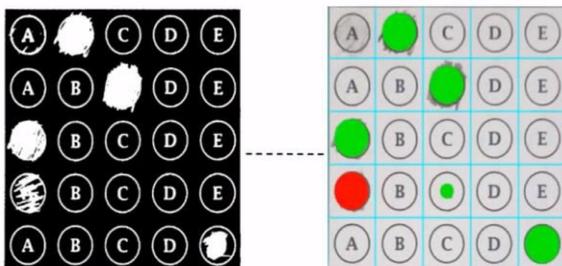
Biggest



After that find the bubble/circle which is correct filled with green and wrong with red.

Threshold

Find Marks



V. RESULT

By using the algorithms above we found our results to be nearly accurate. It was successful in reading the fields. However it won't work when the image is significantly damaged.

VI. CONCLUSION

Previous analysis presented earlier shows clearly the advantages of using automated OMR procedures instead of manual. The main advantage of the presented software is the ability to work with every single scanner of the market, needing no extra drivers whatsoever.

The program is also small and compact, easily fitting in a diskette. No special equipment is needed thus retaining a very low cost.

VII. REFERENCES

- [1] Infotronicx. (2010). OMR system.
- [2] Microsoft. (2010). Moving Java Applications to.NET. Available:<http://msdn.microsoft.com/enus/library/ms973842.aspx>
- [3] Fairley R. (2002) Software Engineering Concepts (For project size) New York: Tata Mac Graw Hill.
- [4] Pressman Roger S. (2001) Software Engineering- A Practitioner's Approach- Fifth Edition, New York: Tata Mac Graw Hill.
- [5] Jalote P. (2005) Software Project Management In Practices3rdEdition, United States of America: Springer Science + Business Media, Inc.
- [6] Software Project Management – A Unified Framework by Walker Royce.
- [7] Hoffer JA,George JE, Valacich JS, 1999, Modern System analysis and design.
- [8] A. Al-Marakeby, F. Kimura , M. Zaki, A. Rashid "Design of an Embedded Arabic Optical Character Recognition", International Journal of Signal Processing Systems March,2013.
- [9] David Doermann, Jian Liang, and Huiping Li "Progress in Camera-Based Document Image Analysis" International Conference on Document Analysis and Recognition (ICDAR'03)
- [10] Harshad B. Prajapat, Dr. Sanjay K. Vij "Analytical Study of Parallel and Distributed Image" Processing, International Conference on Image Information Processing (ICIIP 2011)
- [11] A. M. Smith, "Optical mark reading - making it easy for users", In Proceedings of the 9th annual ACM SIGUCCS conference on User services, United States, 1981, pp:257-263.

[12] K. Toida, “An Overview of the OMR technology: based on the experiences in Japan”, Workshop on Application of new information technology to population: Paper based data collection and capture, Thailand, 1999.

[13] Sabyasachi Das, “Optical Mark Recognition Technology for Rural Health Data Collection”, November 2010.

[14] Hui Deng, Feng Wang, Bo Liang, “A low-cost OMR solution for educational applications”, Parallel and Distributed Processing with Applications 2008, ISPA 2008.

[15] E. Greenfield, “OMR Scanners: Reflective Technology Makes the Difference”, Technological Horizons In Education, Vol. 18, pp: 1991.

[16] K. CHINNASARN, “An image-processing Oriented optical mark reader”, Applications of digital image processing XXII, Denver CO, 1999.

[17] Ngo Quoc Tao and Do Nang Toan “Some Characteristical aspects of Markreader Software Package for Automatic Mark Data Entry”, Circuits and Systems, 2002. APCCAS'02, pages 437 - 442 vol.2, 2002.