International Journal of Applied Mathematics Electronics and Computers

IJAMEC

ISSN:2147-8228 http://dergipark.gov.tr/ijamec **Original Research Paper**

An Image Processing Oriented Optical Mark Recognition and Evaluation System

Zeki KÜÇÜKKARA*[1], Abdullah Erdal TÜMER[2]

Accepted : 28/12/2018 Published: xx/xx/201x **DOI: 10.18100/ijamec.2018447788**

Abstract: In this study, a fast, reliable and cheap method is suggested for the recognition and evaluation of the marks of a multiple-choice test on the images that are obtained via the scanning of the optical forms printed on a standard sheet of paper with an ordinary scanner. This method is called the recognition of optical marks and is the process of capturing the data on the multiple-choice forms. The application of recognition has been developed by using the software language, Python and the image processing library, OpenCV. When the answer sheet is loaded in the application, incorrect answers are marked as red while the correct ones as green and with the calculation of the correct/incorrect answers and blanks, the result is printed on the optical form image. This method is economical, fast and quite successful. As a result of three examinations in Konya with the participation of 35.250 students, 105.750 optical forms were scanned with a scanner. The recognition success has been calculated as 99,76 %. The empirical studies have shown that the suggested system is more successful than the conventional optical mark recognition systems regarding accuracy, reliability and performance.

Keywords: Image Processing, Multiple Choice Test, OpenCV, Optical Mark Recognition, Python, QR Code

1. Introduction

In today's technology, there are lots af applications in our life related to computer –based image processing and computerized recognition. Automation in cockpit, mobile robot applications, implementations in agriculture, surveillance of friend and foes in military zones, biomedical and medicine, geographical data systems, security systems, applications in design and production processes can be held as examples for these areas [1]. Obtaining data about the objects on images and drawing a conclusion from them has been taking the high attention of researchers. Image processing can be defined as the process of seeing and evaluating the image by a machine despite not being as well as it is carried out by human eye and brain. The image processing applications are very limited in educational technologies. Therefore, image processing techniques have recently begun to be seen for various purposes in education as well.

Many tests or questionnaires are applied in many disciplines like education. In many examinations carried out by Assessment Selection and Placement Center (OSYM) or Ministry of Education (MEB) multiple choice questions are asked. When a country-wide examination is held, the publication of question papers and answer sheets in color, its transportation to the exam centers, recalling the optical forms and the recognition and evaluation of these forms with special devices are really expensive. Especially, the use of optical form scanners with high prices appears to cost the most. Each of the operations above is a process and the technique to recognition the answers from optical forms is the recognition of optical marks. Optical Mark Recognition (OMR) is a technique capturing the data marked on multiple choice documents such as questionnaires and tests [2].

[1,2] *Computer Engineering, Necmettin Erbakan University, Konya-42250, TURKEY*

* *Corresponding Author: Email: zekikucukkara@gmail.com*

The OMR technology, by sending light on the document, perceives the existence or non-existence of the mark according to the density of the light reflected by circles. However, it cannot perceive the shape of the mark [3]. Even though there are a lot of and various OMR systems, that it is expensive to buy and maintain these devices makes it more difficult to use in developing countries [4]. Schools are known to be the places where this technology is mostly used. It is really difficult to purchase and use these devices with the limited economical capacity of the schools. However, it is possible to implement and evaluate examinations with optical forms faster and more cheaply by means of a standard scanner and a software.

The Authors in [4] developed a mobile application via which they could evaluate students' answer sheets without connecting an optical device but using their mobile phone or tablet PC. In a multiple-choice exam with 100 student participants, such operations as the justification of the form, revealing its attributions, recognition and its evaluation were conducted. Each optical form was recognition in 20 seconds. Recognition by four devices, these optical forms were figured out as 90 % success at least and 99.7 % at most. In [5], the author developed a system for optical mark recognition using C# and OpenCV library. Recognition process is applied for a 50-item multiple choice test. However, no operation about the success and confirmation of the process was carried out. The authors in [6] compared a blank form with a scanned form with the template matching technique in the process of OMR. They indicated the recognition of the answers successfully. In [7], the application of OMR was developed with the help of a scanner using NetBeans IDE and Java languages. With this application, the optical forms were recognition by justifying them according to four reference points pre-determined on the form. The authors were able to demonstrate that the success level on undamaged forms was high while it was low on damaged ones. In [8], the scanned optical form images were recognition by determining key points on the optical form and through justification of the image regarding these points. In [9], the authors developed OMR software using image processing techniques with OpenCV and Python.

In this work, they didn't recognition all sections on the form (the owner, the booklet group, whether the candidate sat for the exam or not etc.), however just a limited area in which the answers took place. They stated the success percentage in the work. No operation about the number of correct/incorrect answers and blanks was fulfilled. In [10], the authors loaded the image of the optical form with correct answers in the database. They designed a system that compares the loaded image with the students' optical forms. In the system each question and answer area were operationalized in AND with its answer sheet. The threshold was pegged to 150 and they found the success rate as 88%. In this work, an application that prints out the information that is in a QR Code on the optical form in a file together with the answers on the optical form marked by the student has been developed by using image processing algorithm and techniques. In the application, the student's answers were compared with the answer sheet. After incorrect answers were identified as red and the correct ones as green, they were written on the image of the student's answer sheet. These operations were carried out using Python software language with OpenCV, Imutils and ZBar libraries. The recognition success percentage of the system was found as 99.76.

The structure of the study is as follows: The image processing algorithm and techniques, and libraries are given in Sec. 2. In Sec. 3, the comparison and evaluation of results with other optical form reader devices are given. Finally, Sec. 4 draws the main conclusions.

2. Material and Method

In this work, the system of OMR has been developed out using Python software language with OpenCV, Imutils and ZBar libraries. Python is a programming language written by a Dutch programmer named Guido Van Rossum. Python; is an independent platform, interpretable, interactive object-oriented high-level programming language [25]. In this language; There is no need for a compiler as in C, C ++, Java. OpenCV (Open Source Computer Vision) is an open source library introduced by intel for image and video analysis. On the image obtained with a scanner, the following operations were performed in order:

- Conversion of the image to grayscale
- Betterment and crystallization of the image
- Binary conversion with Otsu thresholding
- Morphological operations
- Detecting the edge
- Contour extraction and corner finding
- Geometrical correction
- Bubble finding
- Comparing bubbles
- Printing out the results in a file and on the image

Figure 1 shows the flowchart of the proposed OMR System.

An image is accepted as a two dimensional vector series and defined as two-dimensional function like f(x,y). An area f(x,y) function indicates regarding any coordinate on the image is called as pixel. Pixel is regarded as the smallest element of the digital image [11,12]. In digital image processing methods, the image is investigated under three headings. It is classified as binary image, grayscale and color image [12]. The image of the optical answer sheet is converted from RGB to Gray. While converting into gray, the average method that is put into practice with the multiplication of RGB channels with predetermined factors. This method is calculated using the formula in Equation 1. [13,14]

$$\text{Gray} = (\text{red} * 0.3 + \text{green} * 0.59 + \text{blue} * 0.11) \qquad (1)$$

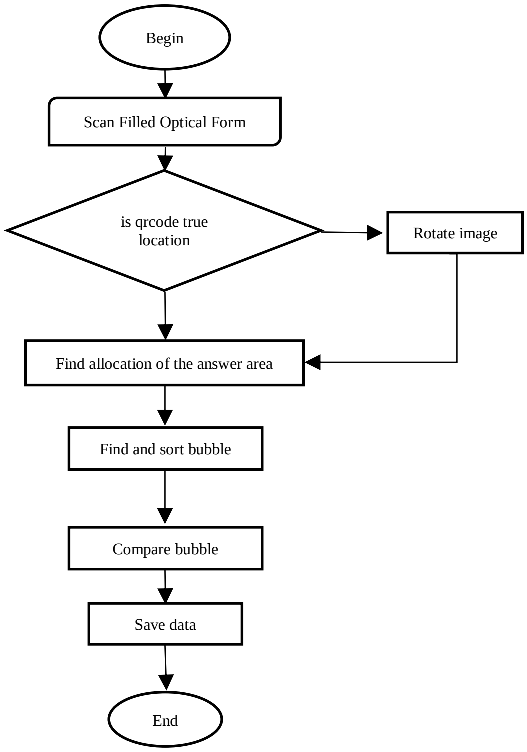

Fig. 1. Flowchart of the developed OMR System

The detailed pseudo-code of the developed OMR system is presented Figure 2.

```
FUNCTION qroku(img):
    data <- decode(img)
    qrcode <- data[0][0]
    location <- data[0][1]
       IF(location = mylocation):
          OUTPUT qrcode
       ELSE:
          rotate(img)
      ENDIF
ENDFUNCTION
FUNCTION cevapalanibul(img):
    img <- threshold(img)
    opening <- morphology(img)
    canny <- cannydetector(opening)
             for gray in split(canny):
                         contours <- findContours(canny)
                         for cnt in contours:
                                     IF len(cnt) = 4 AND
contourArea(cnt) = myarea:
                         x,y,w,h <- boundingRect(cnt)
                         cevapalani <- img[y:y+h,x:x+w]
                         OUTPUT cevapalani
                         ENDIF
             ENDFOR
             ENDFOR
ENDFUNCTION
```

```
FUNCTION degerlendir(img):
   cnts <- findContours(img)
   for cnt in cnts:
            (x, y, w, h) <- boundingRect(cnt)
            ar <- w / h
            IF ar=dairecapi
                        dairecnts.append(cnt)
            ENDIF
   ENDFOR
   dairecnts <- sort_contours(dairecnts,method="top-to-bottom")
   for (q, i) in enumerate(np.arange(len(dairecnts), 4)):
            satircnt <- sort_contours(dairecnts[i:i + 4])
            for (j, c) in enumerate(satircnt):
                        mask <- np.zeros(img.shape)
                        drawContours(mask, [c])
                        mask <- bitwise_and(img, img, mask=mask)
                        IF array[i] < countNonZero(mask)
                                    array[i] = c
                        ENDIF
            ENDFOR
   ENDFOR
ENDFUNCTION
```

Fig. 2. The detailed pseudo-code of the developed OMR system

The gray in the formula represents the average value of a pixel. Converted into gray, the image has been purified from pixel mistakes through cleansing pepper-pot appearances with Median Blur Filter. In Figure 3, the image of a Colorful Answer Sheet and in Figure 4 the image of Gray scaled image of an optical form filtered with Median Blur are shown.

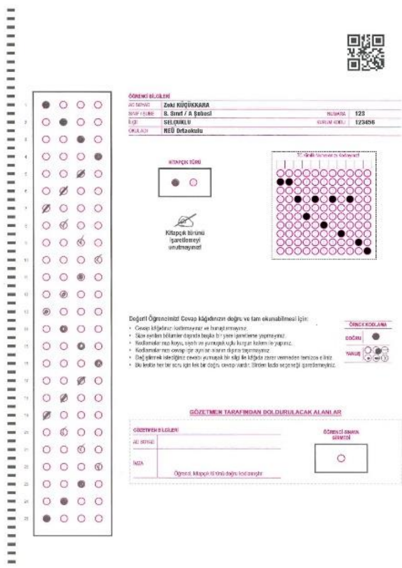

Fig. 3. Colorful Optic Form

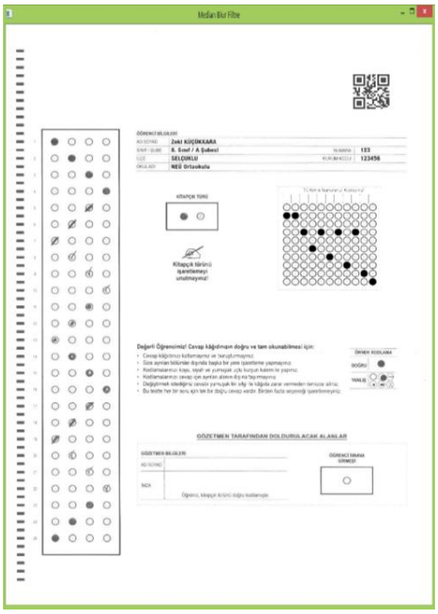

Fig. 4. Gray scaled and Filtered Image

A thresholding value is used while converting a gray scaled image into binary image. On the gray scaled image, each pixel is evaluated by processing. In the image processing method, the thresholding value is determined by the user. Throughout the process, the pixels below the thresholding value are appointed as 0, that is, black; the others as 255, that is, white. In the developed application, there is no pre-determined thresholding value. This is because of the fact that the light on the image will not be delivered equally regarding the angle of the light source if a mobile phone or tablet PC is used while the images are shot. Therefore, the object and the background will not be distinguished. To avoid this, it is very important to determine a gray scale thresholding. Various techniques are recommended for this operation. In an ideal occasion, histogram possesses a deep and sharp valley between the two peaks representing the background and the objects. Henceforth, it is possible to determine the threshold from the bottom of this valley [15-16]. However, it is hard to detect the bottom of the valley delicately for most images.

Especially, if the valley is wide and flat; the noise is much or the peaks are not equal, it is very difficult to detect the valley. Otsu's method, decreasing the intraclass variations to the least, seeks a threshold obtaining better results when two peaks of the histogram of the original image belong to, one to the signal or the foreground and the other to the background. Otsu's threshold is found by seeking throughout all the ranges of the image's pixel values until the intraclass variations reach the minimums. As described, the threshold detected by Otsu's method, either foreground or the background, is detected more profoundly by a class having a wide range. Thus, Otsu's method can generate the smallest results when the histogram of the image has more than two peaks or one of the classes has a wide range [17]. As the thresholding values must be dynamically determined in line with to Otsu's method and the gray scaled image should be converted into binary system [18,19,20,21], this operation has been carried out. In Figure 5, the binary converted image with Otsu's thresholding method is displayed. After this operation, corner finding technique is used in order to define the borders of the objects on the binary image.

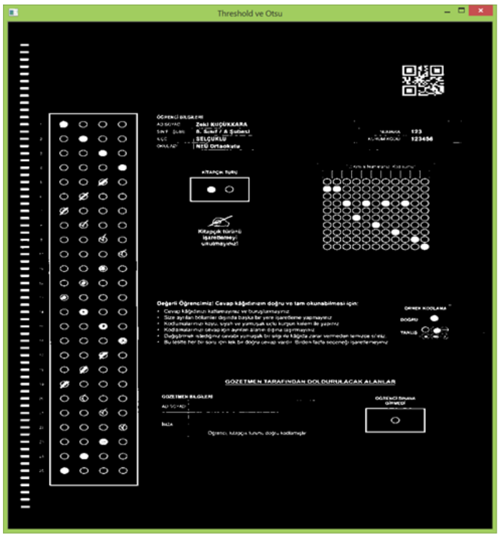

Fig. 5. Binary image

Named as Canny Edge Detector and used to find the corners, this method was developed by J. Canny in 1986 [22,23]. Canny Edge Detector is one of the most commonly used image processing devices that detects the edges very precisely. It is considered to be as the standard edge detecting method in industry. Canny accepted the edge detecting problem as a problem of signal processing optimization, thus developed it in order to optimize an objective function. The solution of this problem was a hard-exponential function but Canny found out many ways to predict the edge detecting problem and to optimize it [23]. In figure 6, the applied form of Canny edge detector algorithm on a binary image is displayed. Canny algorithm has been used to detect the squares around the areas which were marked on the form.

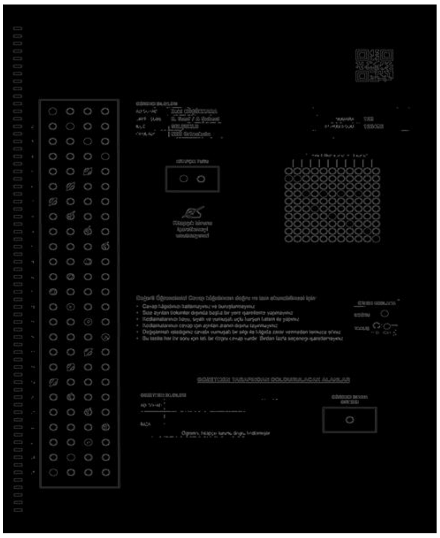

Fig. 6. Image processed with Canny algorithm

Although the algorithms like Camry Edge detector are able to be used to detect the edge pixels separating different parts on an image, they cannot indicate anything about the entities related to themselves. The next step is to combine these edge pixels with the contours. A contour is the list of the points representing a curve on a figure or image [22]. It is crucial to break the image into pieces while classifying or segmenting it [24]. Contours can be defined as a curve that compounds all the constant points (along the border) having the same color or density. They are useful tools used to perceive objects, identify them and analyze figures. In Figure 6, the applied form of corner detecting and contour extraction is shown. With the help of this operation, squares and circles are determined on the image. Figure 7 displays the rectangles whose corners and contours were extracted.

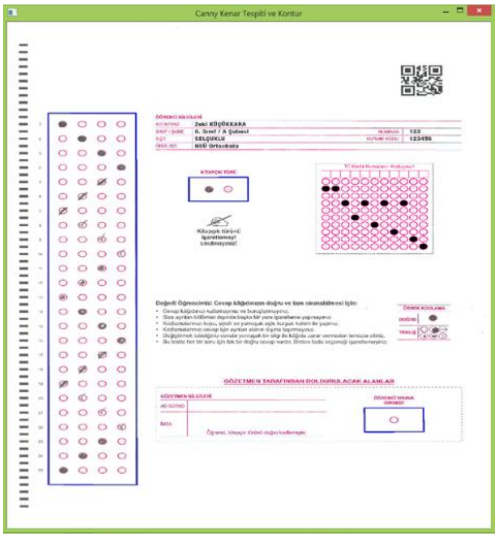

Fig. 7. Contour extracted

In Figure 7, the allocation of the answer area has been detected on the image. Figure 8 indicates the binary conversion of the allocation of the answer area and the colored form of the contour operation on the image.

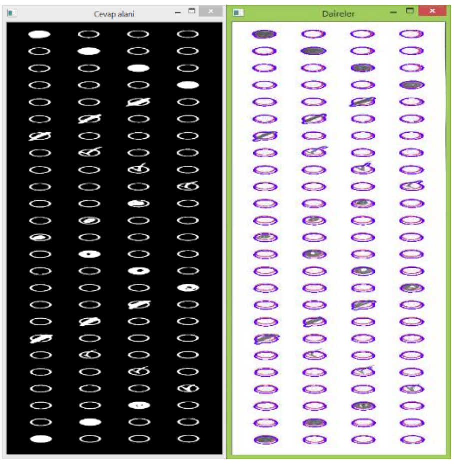

Fig. 8. Binary image (on the left), contour extracted image (on the right)

In Figure 8, after the contour operation, the contours were ordered from left to right and top to bottom in accordance with the coordinate system. In turn, the answers on each line were compared to the ones on the answer sheet. So, whether the circular areas were correct, incorrect or blank was determined. Correct answers were marked as green while the incorrect ones as red. The result was written on the image. Finally, it was typed in a file and saved. In Figure 9, the evaluated form of the answer sheet is displayed. In Figure 10, the written form of the result including the numbers of correct, incorrect answers and blanks on the optical answer sheet is shown.

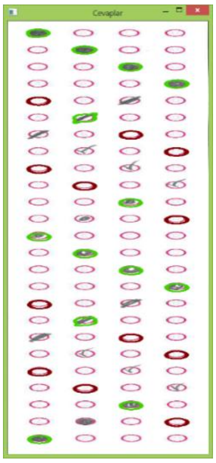

Fig. 9. The comparison and evaluation of marks with the answers

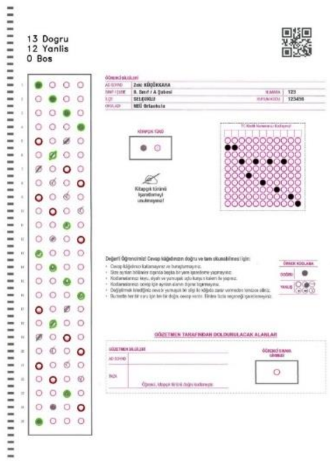

Fig. 10. Final State of Student Optical Answer Sheet

3. Result and Discussion

The system for Optical Mark Recognition of has been developed with Python programming language. The software was applied in the evaluation of three different examinations with the participation of 35.270 students in 570 schools in Konya. 105.750 optical answer sheets were recognition with the software. To compare the success of the developed OMR system, 105.750 optical forms were compared with named Insight 700c scanner and the optical reader software of this scanner belong to Scantron Company. Information about the students was both written on the forms and saved in QR Codes. The marked optical forms were scanned with a scanner. Because of student oriented mistakes (such as the harm on the QR Code), the QR Codes of 250 answer sheets weren't read. This decreased the success rate of the software. Together with the unread answer sheets, the success rate of the whole recognition process was found 99.76 %. Of the 105.500 students with recognition QRCodes, 750 couldn't be recognition due to student-oriented reasons (combining circles, deleting the edge lines around of the answer area or deleting the circles etc.). When these were also regarded, the success rate was calculated as 99.29 %. This problem is natural when a standard optical form scanner is used as well. To sum up, 1000 answer sheets out of 105.750 couldn't be read out and the success rate of the software was calculated as 99.1 %.

Each optical form was recognition in 2.4 seconds with the application. The 105750 optical form is proportionally divided into 11 different folders. The developed OMR system (application) was run separately in each folder. When 11 different applications were used on the same computer (with i7 CPU, 16 gb ram), the recognition of all optical answer sheets lasted 8 hours (28.800 seconds). Approximately an image was recognition in 0.34 second. Alongside with the success of this application regarding accuracy and reliability, other advantages of this software can be ordered as follows:

- A black and white print out of an answer sheet can be used instead of a special, colored and expensive optical form paper.
- The optic forms can be scanned with a standard scanner.
- The same operations can be carried out without purchasing an optical form reader or software.

4. Conclusion

This work presents a system for Optical Mark Recognition developed for multiple choice tests with the programming language Python. 105.750 optical forms obtained after 3 different examinations for the 5th grades of 571 schools in 31 districts in Konya were evaluated with this software. As it is easy to use, the software can be easily used by teachers or school managers as well. Hence, fast, reliable and effective software like this will not only enable the personnel to save time but also it will be very cheap. Furthermore, students will be able to learn the results earlier. When the software is used in many other schools, it will also make it possible to gather lots of data about the students' marks. It is possible for the software to be used in multiple choice examinations by the users in other cities. The software has proven to be efficient to use while evaluating the answer sheets of multiple-choice exams when the success rates are considered. Thanks to its success, the software has been actively used in the Assessment and Evaluation Center of Konya Provincial Directorate of National Education. This study showed some important results for research in OMR, OCR, classification approaches and image processing techniques. The developed application has been tested in the school exams of real-life students. It can be developed and used in future applications such as reading of questionnaires with similar systems, voting forms, attendance forms, product evaluation, reading and evaluation of university entrance forms. The proposed system can also be developed and implemented on a mobile phone instead of a computer.

References

[1] G. Samtaş and M. Gülesin, "Sayısal Görüntü İşleme ve Farklı Alanlardaki Uygulamaları," *Electronic Journal of Vocational Colleges*, vol. 2, no. 1, pp. 85-97, 2011.

[2] 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.

[3] Anonymous. (2018, Erişim Tarihi: 29.8.2018). *ICR, OCR and OMR - A Comparison of Technologies*.

[4] A. Yüksel, İ. Çankaya, M. Yalçınkaya, and N. Ateş, "Mobile based optical form evaluation system," *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi* vol. 22, pp. 94 - 99, 2016.

[5] 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.

[6] D. Patel and S. Zaid, "Efficient System For Evaluatıon Of Omr Sheet-A Survey," *International Journal of Advanced Research in Engineering, Science & Management,* vol. 3, no. 7, 2017.

[7] 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.

[8] I. A. Belag, Y. Gültepe, 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.

[9] 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.

[10] 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.

[11] R. C. Gonzales and R. E. Woods, Digital Image Processing, 3 ed.: Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 2002. [Online]. Available.

[12] D. Karakuş, "Görüntü Analiz Yöntemleri İle Kayaçların Yapısal Özelliklerinin Tanımlanması," Doktora Tezi, Fen Bilimleri Enstitüsü, Dokuz Eylül Üniversitesi, 2006.

[13] Ö. F. Boyraz and M. Z. Yıldız, "Mobil Damar Görüntüleme Cihaz Tasarımı," presented at the 4th International Symposium on Innovative Technologies in Engineering and Science - ISITES2016, (Alanya/Antalya - Turkey), 2016.

[14] T. Helland. (2018, Erişim Tarihi : 07.09.2018). *Seven grayscale conversion algorithms*.

[15] M. L. Mendelsohn and J. M. S. Prewitt, "The Analysıs of Cell Images," *Annals of the New York Academy of Sciences,* vol. 128, no. 3, pp. 1035-1053, 1966.

[16] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 9, no. 1, pp. 2 - 66, 1979.

[17] N. Senthilkumaran and S. Vaithegi, "Image Segmentation By Using Thresholding Techniques For Medical Images," *Computer Science & Engineering: An International Journal (CSEIJ),* vol. 6, no. 1, pp. 1-13, 2016.

[18] J. S. Weszka, R. N. Nagel, and A. Rosenfeld, "A Threshold Selection Technique," *IEEE Transactions on Computers,* vol. 23, no. 12, pp. 1322-1326, 1974.

[19] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image Analysis Using Mathematical Morphology," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 9, no. 4, pp. 532-550, 1987.

[20] M. Sezgin and M. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging,* vol. 13, no. 1, pp. 146-165, 2004.

[21] C. Yu, C. Dian-ren, Y. Xu, and L. Yang, "Fast Two-Dimensional Otsu's Thresholding Method Based on Integral Image " presented at the 2010 International Conference on Multimedia Technology (ICMT), Ningbo, China, 2010.

[22] G. Bradski and A. Kaehler, *Learning OpenCV*. USA: O'Reilly Media, Inc., 2008.

[23] S. Vijayarani and M. Vinupriya, "Performance Analysis of Canny and Sobel Edge Detection Algorithms in Image Mining," *International Journal of Innovative Research in Computer and Communication Engineering,* vol. 1, no. 8, pp. 1760-1767, 2013.

[24] D. Auroux, L. D. Cohen, and M. Masmoudi, "Contour Detection and Completion for Inpainting and Segmentation Based on Topological Gradient and Fast Marching Algorithms," *International Journal of Biomedical Imaging,* vol. 2011, p. 20, 2011.

[25] I. Culjak, A. Abram, T. Pribanic, H. Dzapo, and M. Cifrek, "A brief introduction to OpenCV," presented at the 2012 Proceedings of the 35th International Convention MIPRO, Opatija, 2012.