Revista Dilemas Contemporáneos: Educación, Política y Valores.
http://www.dilemascontemporaneoseducacionpoliticayvalores.com/
Año: VI Número: Edición Especial Artículo no.:34 Período: Julio, 2019.

TÍTULO: Sistema de Lector Óptico de Marcas (OMRS) con el uso de Escáneres de Imágenes basados en web.

AUTORES:

1. Ph.D. Kamil Dimililer.
2. Master stud. Getinet Amare Mekonnen.

RESUMEN: El documento propone un nuevo sistema utilizando técnicas de procesamiento de imágenes. El sistema tiene ocho tipos de hojas de puntuación y su propio código de barras para clasificarlas. La hoja de puntuación contiene 15, 20, 30, 45, 60, 80, 100 y 125 preguntas con 5 opciones en cada una. Este sistema tiene dos tipos de usuarios: el primero es un alimentador y puede ver el resultado OMR, y el segundo ve resultado OMR. El sistema envía la hoja de calificación reconocida al delimitar el área de respuesta con diferentes colores: el verde para respuestas marcadas correctamente, el rojo para respuestas marcadas duplicadas y el amarillo para las preguntas sin marcar. Los resultados muestran que el sistema toma hojas de calificación y produce resultados con calificación de letras y letras numéricas equivalentes.

PALABRAS CLAVES: Procesamiento de imágenes, reconocimiento óptico de marcas, hoja de calificación, aplicación basada en web.

TITLE: Optical Mark Reader System (OMRS) using Web Based Image Scanners.

AUTHORS:

1. Ph.D. Kamil Dimililer.

2. Master stud. Getinet Amare Mekonnen.

ABSTRACT: In this paper, a new system is proposed by using image processing techniques. The proposed system has eight types of scoring sheets and its own unique barcode to classify them. The scoring sheet contains 15, 20, 30, 45, 60, 80, 100 and 125 questions with 5 choices in each. This system has two types of users where first one feed sheets and can see OMR result, and the second one only sees OMR results. The proposed system sends the recognized scoring sheet by bordering the answer area with different colors; these are green border for correctly marked answers, red border for duplicate marked answers, and yellow border for totally unmarked questions. Results obtained show that the proposed system takes scoring sheets and produces results with letter grade and equivalent numerical letters.

KEY WORDS: image processing, optical mark recognition, scoring sheet, web-based Application.

INTRODUCTION.

Optical mark recognition (OMR) is the process of scanning optical paper sheets to decide the appearance and non-appearance of marks in a prearranged location. Analysts have used it for almost more than 80 years to convert paper sheet data to computer readable format [5]. So, its inputs are well marked optical sheets to have accurate results for both software and hardware-based OMR. Optical mark recognition also known as optical mark reading or optical mark reader. It is a technology that extracts useful data from marked shapes such as fill-in squares, rectangles, triangles, circles and other fill-in shapes as well.

Even though OMR have many applications though out of our life in this world, the most common use of optical mark recognition is in research centers, universities, schools and industries where large number of hand filled documents needs to be processed such as time attendance, elections, surveys, questionnaires, examinations and replay cards with short period of time and painless.

If we compare to human beings, OMR can handle many documents with high accuracy and faster time. Therefore, we have no more doubt between the performances of human beings and scanned and web-based OMR. The scoring sheet paper quality, the limitation of pencils used for marking squares and weight of scoring sheet papers are some limitations of hardware based optical mark recognition [2], [7], [9], [10], [11], [12], [13].

This work has database tables that stores an answer key of each courses, course code and student id. The developed algorithm for recognizing optical marks is that receives scoring sheet images from ordinary scanner, apply pre-image processing techniques, recognize barcode, course code, student id and recognize students answer from marked squares as well. Scoring sheets have divided into five main parts such as barcode, student's id, course code, barcode and main body that contains multiple-choice squares.

After checking the validity of barcode, student's id and course code, the optical mark recognition algorithm compares recognized students answer with answer key of selected course retrieved from database table by the help of marked course code and recognized barcode. Then an OMR system counts how many marks are matched with the answer key; how many questions are not marked by students; count how many questions marked but not correct answer. Finally, the details of optical mark recognition data will send to html forms by post method for displaying its result to student's own web account and other concerned bodies.

DEVELOPMENT.

Literature review.

According to [8], they were conducted a research on an optical mark reader for Kolkata institute entrance multiple question examinations. 39 scanned scoring sheets out of 5000 were ambiguously erased so, this ambiguity leads to manual verification of those scoring sheets for fair evaluation, otherwise the system couldn't generate correct result as they were expected, and error rate is around 0.78%. Most developers said that java is not pure platform independent programming languages; however, they agree with only scripting languages are [14]. They have used java programming language with swing components for graphical user interface design and its implementation as well for software tool and then printer and scanner as hardware tool.

According to [6], they divide their system into three usual simple phases named as input, process and output. The first phase contains searching and selecting current working folder and answer key to be assign to recognition result. The second phase contains two modules i.e. recognition and analysis. Recognition phase is the key phase in their work that converts fill-in image data i.e. marked bubbles data to digital data (ASCII). The last phase provides to users the result of recognition.

Its objective is to reduce cost of implementation of OMR; cost of scanning sheet is around 0.25USD, to make the application available to wide range of scanners, to make the application that works on a certain level of damaged sheets, the damage includes stain of scoring sheets.

According to [5], they have focused on errors that will happen on during scanning and photocopying of questionnaire forms. They have recommended that the standard template questionnaire sheet would photocopied after getting one printed form for cost reduction purpose. Because the cost we pay for print and photocopy have a little difference per page even though have its own risk on OMR software they had developed. The errors that will happen during scanning and photocopied questionnaire forms are due to translation, scaling and rotation of both scanned and photocopied questionnaire forms and propose the appropriate solution as well. To solve such detected problems, they had proposed their solutions with four phases.

According to [4], their main aim is to solve the drawback of traditional OMR and the developed system works on both windows vista and XP. Their system named by low cost optical mark reader (LCOMR).

According to [3], he has presented an optical mark reader research paper for multiple-choice questions. The work tested by 50 multiple questions with 5 choices. Each question has 5 choices to fill students answer. His system starts the recognition process starting from students answer and continues to their id. After accomplishing the recognition process, the system sends it to students' email address for feedback. The average proposed speed of his system is 1.4 second per sheet including saving images to disk and excluding annotation. With annotation, the system takes 3.5 seconds per sheet. The type of tools used by the system is that: Intel core i5 processor laptop, MATLAB R2014b programming language, and scanner, A4 paper for preparing answer sheet and 100 total answer sheets with 50 questions in it as well.

Proposed method.

Depending on total amount of questions are in the sheet, there are eight types of scoring sheets are there in this system. These are 125, 100, 80, 60, 45, 30, 20, and 15 questions in it. From the literature review section of this work, most of the implementation algorithms are the aspect of mark, student id and course id detection for one type of scoring sheet. However, it is fair to say that bar code and web integrity plays an important role for this work. The need to add bar code to scoring sheet is that to differentiate which scoring sheet is currently being recognize among 8 types. In addition, the aim of web integration is to forward the recognition result remotely to concerned body i.e. students via their own web account.

Scoring Sheet Layout.

An overview of proposed answer sheet is illustrated in **Fig. 1**. Which shows the different parts of scoring sheet. The scoring sheet as shown in **Fig. 2** has eleven parts these are Barcode, Student id, Course code, Answer area, Date, Logo, Name of ministry(institution), Student signature, Invigilator signature, Instructions, Corner Points such as Top left corner point, Top right corner point, Bottom left corner point and Bottom right corner point.

Even though the scoring sheet contains the above 11 parts in it, only five parts are needed by the proposed system. The remaining six part are not needed by the proposed system rather they are needed by organizations. For instance, logo is needed to indicate the scoring sheet belongs to the specific organization. As long as I have used an image processing technique, the above five parts of scoring sheet is mandatory. These parts Corner Points (top left and right, bottom left and right points), Barcode, Student id, Course code and Answer area as well.

Scoring Sheet Registration.

This process should have done after scoring sheet is designed and before filled by students. As I mentioned earlier, there are eight types of scoring sheets were designed. To identify those sheets, its layout information with barcode have to register to database. On the other hand, this task helps us unregistered scoring sheets cannot process with this system.

Image Pre-processing.

Converting color image to grayscale is key for CPU time minimization [1]. Before going to do the actual or target aim of the work i.e. mark recognition, an image should pass through pre-image processing techniques. The reason behind that an image should pass through pre-image processing technique is that an OMR system enables to return correct result. Converting color sheet to grayscale, change the size of sheets, adjust sheet layout are some applied sheet pre-processing techniques.

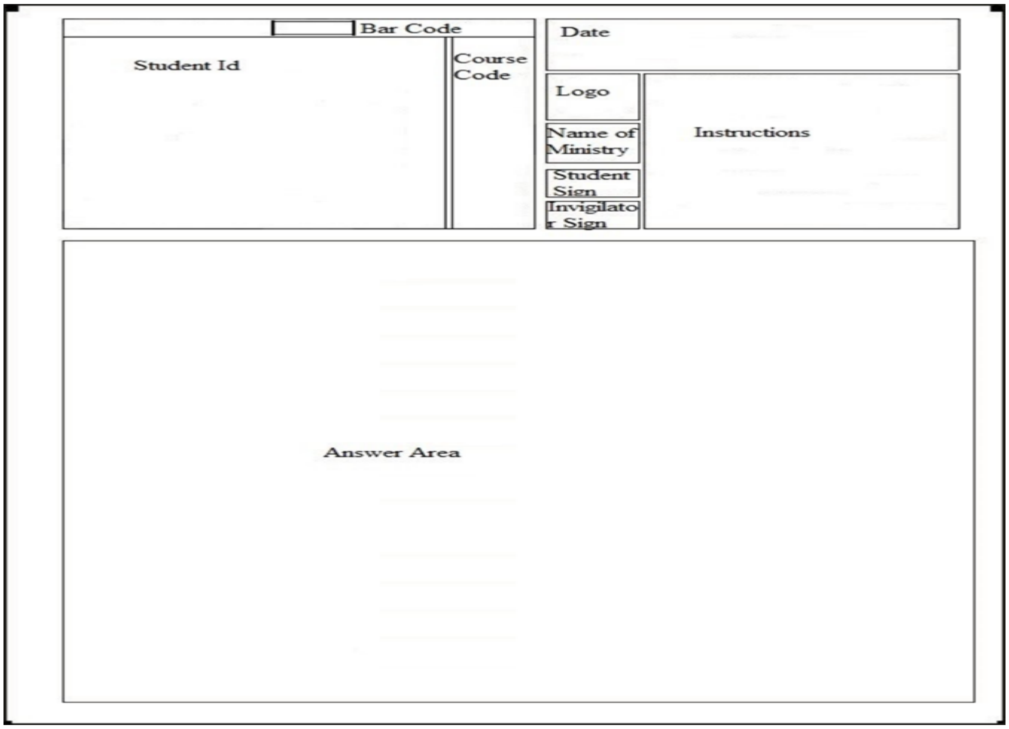


Fig. 3. Proposed Scoring Sheet.

Adjust Scoring Sheet Layout.

Before going to adjust scoring sheet layout, we should know the scoring sheet is in either correct layout or incorrect. The following algorithm does the scoring sheet layout is in correct or incorrect manner and make it correct manner for those not in correct layout:

1. Upload scoring sheet.
2. Convert it to grayscale image type.
3. Get height and width of scoring sheet.
4. Compare results that we obtained on step 3 to the height and width of template scoring sheet.
5. If we got approached result, the scoring sheet becomes either in correct layout or already rotated by 180^0 i.e. scanned in opposite direction.
6. Compare four layouts of scoring sheet that we got on step 5 with template image, if they are same, the sheet is in correct layout otherwise rotate it by 180 degrees.

7. If height and width of template and uploaded scoring sheet are not same, we need to find four corner points. We can find four corner points in the following way:

Top left corner point.

1. Create a loop equal to half of height and width for its row and column respectively.
2. Get intensity value of a pixel. The loop iterates until it gets more than 40% of grayscale pixel.
3. If the loop gets more than 40% of grayscale pixel, it will return the location of that pixel.

Top right corner point.

1. Create a loop with half of height for row side and column side goes from total width size to half of width by one decrement.
2. Get intensity value of a pixel. The loop iterates until it gets more than 40% of that of grayscale pixel.
3. If the loop gets more than 40% of grayscale pixel, it will return the location of that pixel.

Bottom left corner point.

1. Create a loop with half of width for column side and row side goes from total height size to half of height by one decrement.
2. Get intensity value of a pixel. The loop iterates until it gets more than 40% of that of grayscale pixel.
3. If the loop gets more than 40% of grayscale pixel; it will return the location of that pixel and this is bottom left corner point.

Bottom right corner point.

1. Create a loop that starts from total height and width size to half of height and width size respectively by one decrement.
2. Get intensity value of a pixel. The loop iterates until it gets more than 40% of that of grayscale pixel.

3. If the loop gets more than 40% of grayscale pixel; it will return the location of that pixel and this is bottom right corner point.
4. After collecting four corner points on step 7, make transformation to template layout.
5. If height and width of template scoring sheet image is equal to width and height of uploaded scoring sheet image respectively, that image is rotated both clockwise and anticlockwise direction by 90 degrees. So, rotate an image by 90 degrees and got to step 6.

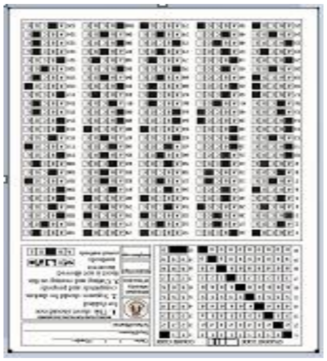


Fig. 5. 180^0 rotated sheet

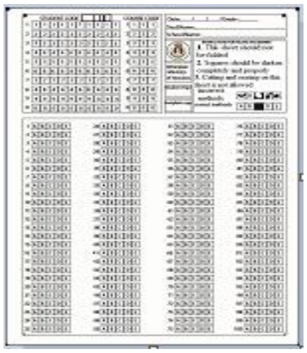


Fig. 4. Template sheet

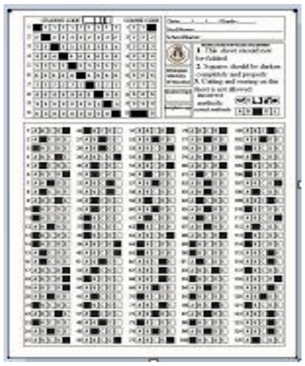


Fig. 7. Correctly scanned sheet

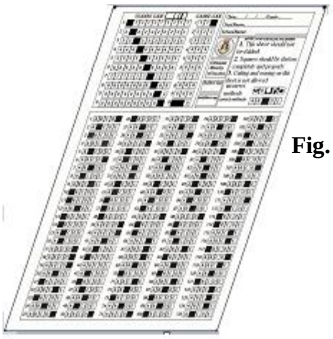


Fig. 6. Inclined scanned sheet

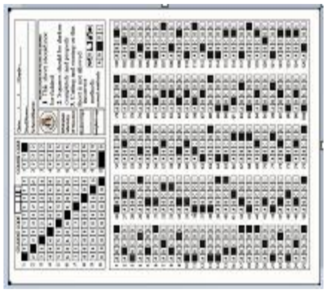


Fig. 9. +90 rotated scanned sheet

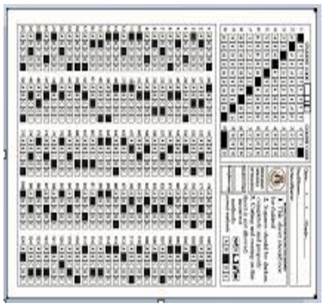


Fig. 8. -90 rotated scanned sheet

Course Code recognition.

After scoring sheet registration and correct barcode is detected, the next step is reading course code. The recognized course code and previously kept course code on database table have to it and that recognized sheet is belongs to which course, are answered by this step. An algorithm used to read course code is shown below:

1. Upload scoring sheet
2. Convert scoring sheet to grayscale image type
3. Retrieve barcode coordinate location from database table and read barcode
4. Check if the recognized barcode result will exist on previously inserted data to database table about barcode information.
5. If step 4 is correct (barcode information is existing on database table), the OMR system will retrieve course code from database table.
6. Retrieve the starting coordinate location of course code from database table that are kept on it during scoring sheet registration.
7. Get the intensity values of that region of image and count its gray shade intensity value is greater than 40%.
8. If shades of gray will greater than 50%, that region of image considered as marked (filled). This process continues up to 10 rows and 3 columns. The result of this step will be 30 columns with one row array i.e. the result gets from multiplication of ten rows by three columns.
9. Reshape an array that are returned from step 8 to 10 rows by 3 columns.
10. Find an array location that intensity values satisfy the gray level shade rule and check the recognized course code exactly equal to course code retrieved from database table.

11. If they are equal return, the system would read correct course code otherwise the system considers as invalid sheet is uploaded.

Student Code Recognition.

After recognizing valid course code and barcode, the next step is recognizing student's code. An algorithm that reads student's code is almost same as that of course code and listed below: Assume we have valid recognized course code and barcode:

1. Retrieve the starting coordinate values of student code square location starts from database table that are inserted during scoring sheet registration.
2. Then get intensity values of that region of image and count its gray shade intensity values is greater than 40%.
3. If shades of gray will greater than 50%, that region of image considered as marked (filled). This process continues up to 10 rows and 10 columns. The result of this step will be 100 columns with one row array i.e. the result gets from multiplication of ten rows by ten columns.
4. Reshape an array that are returned from step 3 to 10 rows by 10 columns.
5. Find an array index in which intensity values satisfy the gray level greater than 40%.
6. Accumulate results returned on step 5.
7. Check the result returned on step 6(recognized student code) is exist on previously kept on database table.
8. If the result exists on database table, an OMR system would return recognized student code.
9. Otherwise, the process ends that means the uploaded sheet is invalid either incorrect student code is filled by students or some other reason.

From the above algorithms we have student code, course code, total number of questions and barcode are known as well and needed by next recognition process i.e. students answer.

Students Answer Recognition.

This task will have done after successful completion of recognition of barcode, student and course code as well. Otherwise, an optical mark reader system considered, as an uploaded scoring sheet image is invalid and will reject it. An algorithm used to recognize students answer on scoring sheet is as shown below:

1. Retrieve the starting coordinate values of columns by using its barcode on database table.
2. Then get intensity values of that region of image and count its gray shade intensity values greater that 40%.
3. If shades of gray will greater than 50%, that region of image considered as marked (filled). This process will continue up to end of questions. The result of this step will not be same because it depends on the number of questions on sheet. As long as we have eight different types of questions, we can calculate its squares by multiplication of alternatives with total number of questions.
4. Accumulate results returned on step3 to numeric array.
5. Reshape an array that are returned on step 4 to total number of questions and choices per column. An OMR system gets total number of questions and alternatives from database table.
6. Because of student's mistake, questions might not be marked so, we need to find which number of question is marked or not marked. However, in this step let us return marked question number. Create two-dimensional array that we got from step 5 with length of first dimension equal to total number of questions and length of second dimension equal to its choices (alternatives).
7. If elements in step 6 array greater than 50 percent of total gray shade square, an OMR would append marked question numbers by instantiating a list until end of questions. As long as they obey an OMR system rules, whether all alternatives or two alternatives are marked (filled) equally per question, this system recognizes the first shade as student answer.

8. If elements in step 6 array greater than 40% of gray shade, an OMR system would append column or choices (alternatives) values.
9. Translate column values that are returned from step 8 to multiple-choice letters. if column values 0, 1, 2, 3 and 4 set letter “A”, “B”, “C”, “D”, and “E” respectively. If we think that every question number is marked exactly one choice, we would stop here. However, we should think in worst case rather than best: it may happen totally unmarked alternatives per question and more than one alternative gets marked per question.
10. Create a numeric array with length equal to number of questions.
11. If results returned by step 7 does not exist in results returned on step 4, create a dictionary with in existed keys and set ‘0’values. This loop iterates until length of results returned on step 4 and finally returns a dictionary with key question number and values its alternatives.
12. Sort dictionary values returned on step 11 in ascending order with keys.
13. Fetch answer keys from database table that satisfies a specific SQL query condition. Attributes used for SQL query comparison are course code, current academic year and then total number of questions.
14. Check length of arrays returned on step 12 and 13 are equal.
15. If step 14 returns correct, create a loop with length equal to either results returned on step 12 or 13 and check its similarity. This step returns total correct and incorrect result of students’ mark.
16. Convert numeric result returned on step 15 to letter grade.
17. Write results returned on steps 15 and 16 to database table.
18. OMR system notify to students for their own web account on notification icon and can see their results.
19. If errors might happen, students can send message back to an OMR system concerned body about that error.

From the above students answer recognition algorithm, we can get Question number, marked alternative (choice) letter even if it has more than one equal marked alternatives are there on same question and unmarked question number too.

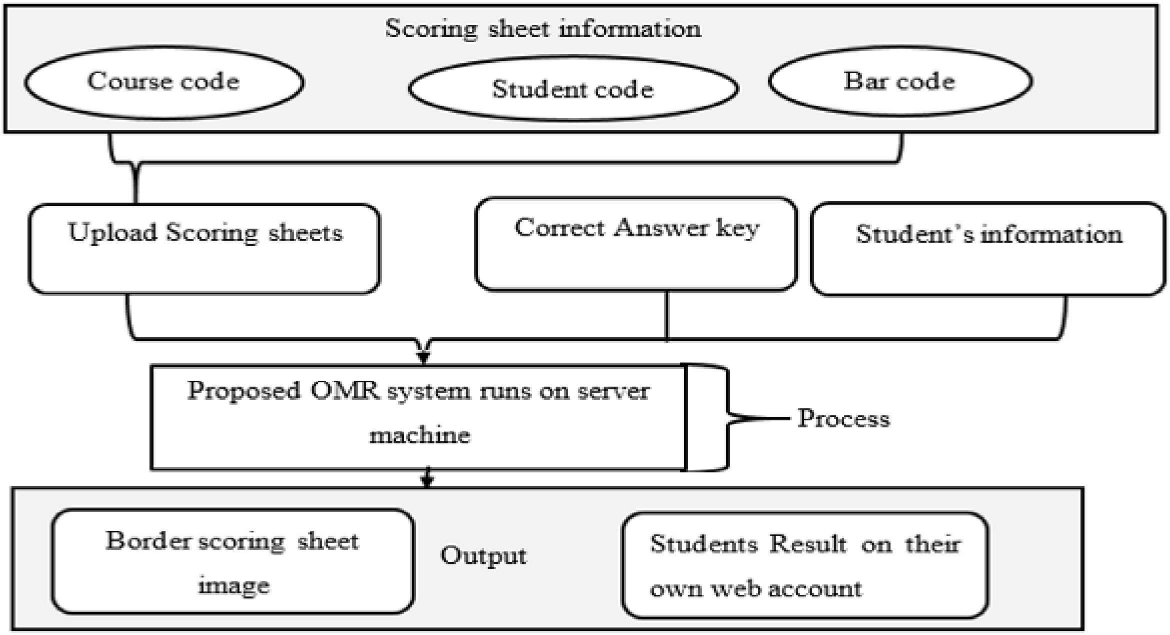

Scoring sheet information
Course code
Student code
Bar code
Upload Scoring sheets
Correct Answer key
Student's information
Proposed OMR system runs on server machine
Process
Border scoring sheet image
Output
Students Result on their own web account

Fig. 10. General workflow of proposed system.

Make Border for Squares.

The above work is used to convert marked image data to ASCII code. However, the following algorithm is used to make border marked squares in different color for visual aid. Colors used for bordering are shown below:

- Green color: - used for squares that are marked (filled) correctly.
- Red color: - used for squares that are marked (filled) incorrectly i.e. multiple squares filled per question.
- Yellow color: - used for squares that are total unmarked (unfilled) correctly per question.

1. Retrieve the starting horizontal and vertical location of squares from database table.
2. Count the number of gray shades that are greater than that of 40%.

Sum up the results returned on step 2.

3. If results, we got from step 3 are greater than 50% of total shade of squares, accumulate the starting vertices of that square.
4. If an algorithm did not get gray shades that satisfies this OMR system rule per question, an OMR system takes the starting point of that question and accumulate its results.
5. In this step, we do have vertices returned on step 4 and 5. Then convert a gray scale image to color image.

Create a loop in which its length is equal to results, we got from step 4 and draw a border around a square.

Results and discussion.

Eight different types of scoring sheets were designed and tested by the proposed web and image based OMR system. 30 total number of questions were used for testing the system and its details shown on Table 1 below.

Table 1. Sheets used for testing the system.

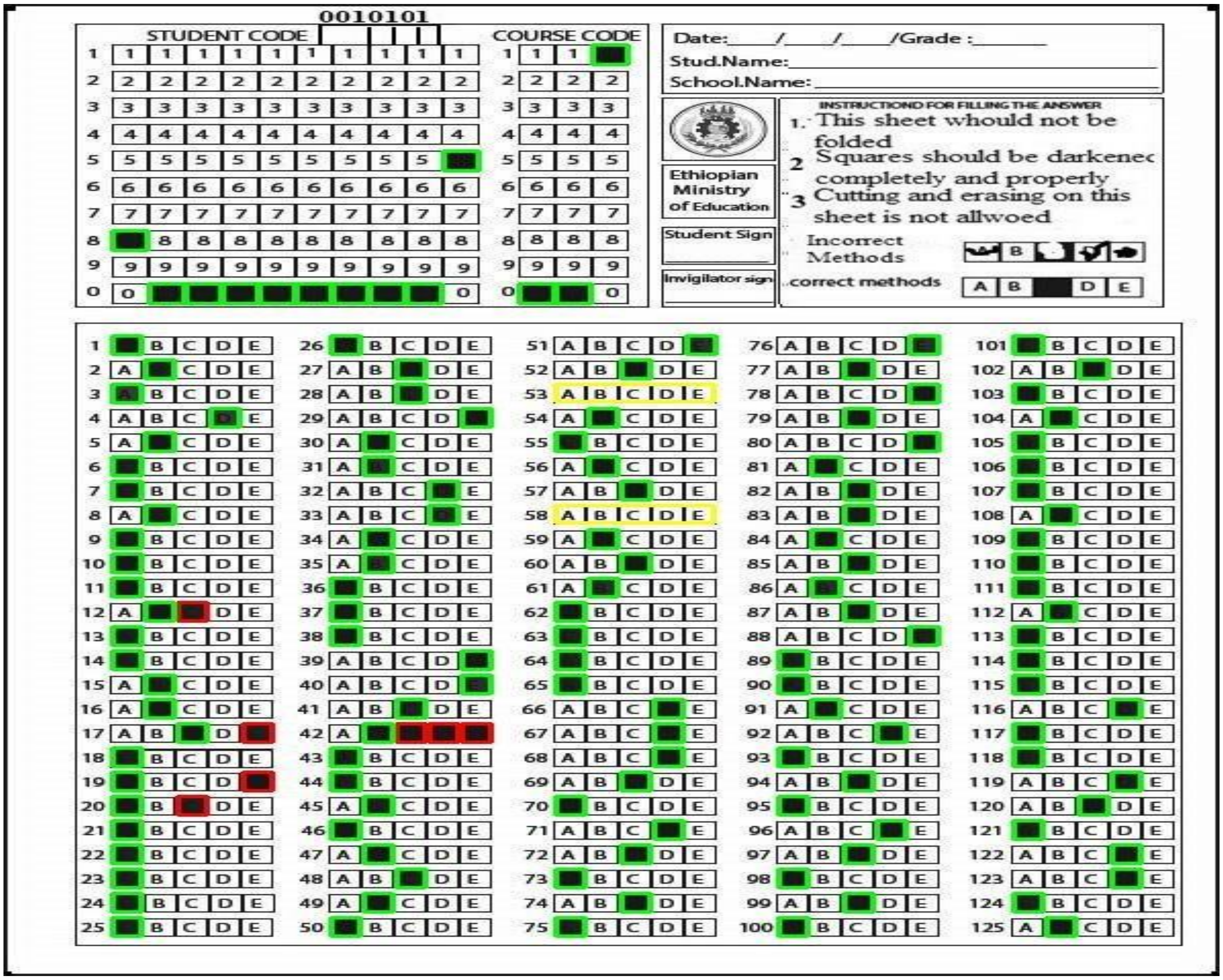
S.R.NO	Total number of questions per sheet	Sheets used for testing	Total Number of squares.
1	15	30	30*((15*5)+(10*10)+(3*10)) = 6150
2	20	30	30*((20*5)+(10*10)+(3*10)) = 6900
3	30	30	30*((30*5)+(10*10)+(3*10)) = 8400
4	45	30	30*((45*5)+(10*10)+(3*10)) = 10650
5	60	30	30*((60*5)+(10*10)+(3*10)) = 12900
6	80	30	30*((80*5)+(10*10)+(3*10)) = 15900
7	100	30	30*((100*5)+(10*10)+(3*10)) = 18900
8	125	30	30*((125*5)+(10*10)+(3*10)) = 22650
Total	475	240	102,450

As we saw on Table 1 column 3, we have used 240 total number of sheets for testing an OMR system. The system has worked with zero error rate, that means all sheets are recognized correctly and forwarded to the recognized student id to enable students see their results via their own web account.

Depending on the machine (computer) resources used for, an OMR system response time were varies. NEC (Nippon Electric Company), Intel® core i5-3210M, 2.50GHz, 64 based processor, 8.00 GB RAM, 500GB HDD and windows 10 home edition laptop computer is used for proposed system implememntation and tested by mobile phones, tablets, desktops and laptops as well.

Table 2. System response time in second(SRTs) for each question type.

Total Number of questions	SRTs1	SRTs2	SRTs3	SRTs4
15	0.1	0.7	1.6	1.65
20	0.14	1.05	2.0	2.15
30	0.2	1.4	2.5	2.5
45	0.30	2.0	3.4	3.6
60	0.4	2.4	3.6	4.4
80	0.5	3.1	4.5	5.0
100	0.6	3.9	4.8	6.0
125	0.8	5.1	6.0	7.0



0010101

STUDENT CODE

1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	■
6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7
8	■	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9
0	0	■	■	■	■	■	■	■	■	0

COURSE CODE

1	1	1	■
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
0	■	■	0

Date:____/____/____/Grade :_______

Stud.Name:________________________

School.Name:_______________________

Ethiopian Ministry of Education

Student Sign

Invigilator sign

INSTRUCTIOND FOR FILLING THE ANSWER

1. This sheet whould not be folded
2 Squares should be darkenec completely and properly
3 Cutting and erasing on this sheet is not allwoed

Incorrect Methods ▨ B ▨ ✓ ●

correct methods A B ■ D E

1	■ B C D E	26	■ B C D E	51	A B C D ■	76	A B C D ■	101	■ B C D E
2	A ■ C D E	27	A B ■ D E	52	A B ■ D E	77	A B ■ D E	102	A B ■ D E
3	■ B C D E	28	A B ■ D E	53	A B C D E	78	A B C D ■	103	■ B C D E
4	A B C ■ E	29	A B C D ■	54	A ■ C D E	79	A B ■ D E	104	A ■ C D E
5	A ■ C D E	30	A ■ C D E	55	■ B C D E	80	A B C D ■	105	■ B C D E
6	■ B C D E	31	A ■ C D E	56	A ■ C D E	81	A ■ C D E	106	■ B C D E
7	■ B C D E	32	A B C ■ E	57	A B ■ D E	82	A B ■ D E	107	■ B C D E
8	A ■ C D E	33	A B C ■ E	58	A B C D E	83	A B ■ D E	108	A ■ C D E
9	■ B C D E	34	A ■ C D E	59	A ■ C D E	84	A ■ C D E	109	■ B C D E
10	■ B C D E	35	A ■ C D E	60	A B ■ D E	85	A B ■ D E	110	■ B C D E
11	■ B C D E	36	■ B C D E	61	A ■ C D E	86	A ■ C D E	111	■ B C D E
12	A ■ ■ D E	37	■ B C D E	62	■ B C D E	87	A B ■ D E	112	A ■ C D E
13	■ B C D E	38	■ B C D E	63	■ B C D E	88	A B C D ■	113	■ B C D E
14	■ B C D E	39	A B C D ■	64	■ B C D E	89	■ B C D E	114	■ B C D E
15	A ■ C D E	40	A B C D ■	65	■ B C D E	90	■ B C D E	115	■ B C D E
16	A ■ C D E	41	A B ■ D E	66	A B C ■ E	91	A ■ C D E	116	A B C ■ E
17	A B ■ D ■	42	A ■ ■ ■ ■	67	A B C ■ E	92	A B C ■ E	117	■ B C D E
18	■ B C D E	43	■ B C D E	68	A B C ■ E	93	■ B C D E	118	■ B C D E
19	■ B C D ■	44	■ B C D E	69	A B ■ D E	94	A B ■ D E	119	A B C ■ E
20	■ B ■ D E	45	A ■ C D E	70	■ B C D E	95	■ B C D E	120	A B ■ D E
21	■ B C D E	46	■ B C D E	71	A B C ■ E	96	A B C ■ E	121	■ B C D E
22	■ B C D E	47	A ■ C D E	72	A B ■ D E	97	A B ■ D E	122	A B C ■ E
23	■ B C D E	48	A B ■ D E	73	■ B C D E	98	■ B C D E	123	A B C ■ E
24	■ B C D E	49	A ■ C D E	74	A B ■ D E	99	A B ■ D E	124	■ B C D E
25	■ B C D E	50	■ B C D E	75	■ B C D E	100	■ B C D E	125	A ■ C D E

Fig. 11. Screen shot of 125-question OMR result.

SRTs1(System Response Time in second), SRTs2, SRTs3 and SRTs4 columns shows that the time it took when an OMR system is (before switched to web, connected to database and without border), (before connected to database and switched to web, with border), (after connected to database, switched to web without border), (after connected to database, switched to web with border), respectively. As shown on Fig. 12 there is an OMR result having 125 questions in it. This sheet computed and recognized result are shown in Table 3. As we saw correctly marked answers were green bordered, incorrectly with red and totally unmarked answers were bordered in green color. We can see Fig 12, Table 3, 4 and 5 results when we login as Teacher, but we can see only Fig 12 and Table 3 result when we login as student with student id 8000000005 and his/her password.

Table 3. Computed results from **Fig. 12.**

Name	Recognized data
Total Number of incorrectly marked Question(s).	2
Total Number of correctly marked Question(s).	123
Recognized Student Id	*8000000005*
Recognized Course Code	*001*
Recognized Course Code	*0010101*
Total Student result	*21*
Letter grade	*['F']*
Numeric value	*['0.0']*

Table 4. Answer Keys retrieved from database table.

Number	F25K	S25K	T25K	F25K	L25K
1	E	C	B	C	C
2	C	A	C	B	C
3	C	B	E	C	A
4	A	C	C	B	B
5	D	C	A	C	C
6	B	E	C	B	C
7	C	D	B	C	E
8	A	C	C	E	A
9	A	C	E	C	C
10	B	A	C	E	C
11	C	A	E	C	C
12	C	C	C	C	D
13	D	C	A	E	C
14	E	C	C	E	C
15	A	D	B	A	A

16	A	B	B	C	C
17	C	A	C	A	B
18	C	B	C	C	C
19	D	C	A	B	A
20	C	C	C	B	C
21	C	E	C	C	C
22	D	D	A	C	B
23	B	C	C	A	C
24	C	C	B	C	C
25	A	A	C	C	C

Table 5. Recognized student answer by proposed OMR system.

Number	F25A	S25A	T25A	F25A	L25A
1	A	A	E	E	C
2	B	C	C	C	A
3	A	C	0	E	B
4	D	E	B	C	A
5	B	B	A	E	A
6	A	B	B	B	A
7	A	D	C	C	B
8	B	D	0	C	A
9	A	B	B	B	A
10	A	B	C	C	A
11	A	A	B	B	B
12	B	A	A	C	A
13	A	A	A	E	A
14	A	E	A	A	A
15	B	E	A	A	D
16	B	C	D	B	A
17	C	B	D	D	A
18	A	A	D	A	D
19	A	A	C	C	C
20	A	B	A	A	A
21	A	A	D	D	D
22	A	B	C	C	D
23	A	C	A	A	A
24	A	B	C	C	B
25	A	A	A	A	C

CONCLUSIONS.

An easy high precise web-based and scanned image-based optical mark reader system is proposed. Implementation results show that the speed of proposed system depends on, system functionality and total number of questions in the scoring sheets.

The proposed system uses ordinary scanners and printers that are available. A4 paper with 70 GSM is used for preparing scoring sheets, because of their high availability and common use. This system has two users; the one who feed the scoring sheet and can see the recognition result and the other only see the result of its recognition after compared with answer key. The system offers subsequent functionalities; convert black marks to computer understandable format, indicate which choices are marked by drawing border around the answer area, compare the recognized black marks with correct answer key that are stored in database table and calculate the result, compute the result into letter grade and letter grade equivalent, enables students can see their own result on web account.

Finally, students can send feedback to concerned body if the answer recognized by proposed system and students marked will not same. In another words, this system takes scoring sheets and finally produce its result with letter grade and numerical letter equivalency. As long as the proposed system is web based, any web enabled devices such as mobile phones, tablets, laptops, desktops computer can have used for accessing it.

BIBLIOGRAPHIC REFERENCES.

1. Dimililer, K., Ilhan, A. (2016). Effect of image enhancement on MRI brain images with neural networks. 12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS 2016, 29-30 August 2016, Vienna, Austria.

2. Ms. Sumitra B. Gaikwad. (2015). Image Processing Based OMR Sheet Scanning. International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE) Volume 4, Issue 3, March 2015.
3. Douglas Chai (2016). Automated Marking of Printed Multiple Choice Answer Sheets.2016 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE).
4. Hui Deng, Feng Wang, Bo Lian. (2008). A Low-Cost OMR Solution for Educational Applications. International Symposium on Parallel and Distributed Processing with Applications
5. Astha Gupta,Sandhya Avasthi. (2016) Image based low cost method to the OMR process for surveys and research. International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-7, July 2016 ISSN: 2395-3470 www.ijseas.com
6. Garima Krishna, Hemant Ram Rana, Ishu Madan, Kashif & Narendra Sahu. (2013). Implementation of OMR Technology with the Help of Ordinary Scanner International Journal of Advanced Research in Computer Science and Software Engineering
7. Nalan Karunanayake. (2015). OMR Sheet Evaluation by Web Camera Using Template Matching Approach. International Journal for Research in Emerging Science and Technology, vol. 2s, 8.
8. Rakesh S., Kailash Atal, & Ashish Arora. (2013). Cost Effective Optical Mark Reader International Journal of Computer Science and Artificial Intelligence. Jun. 2013, Vol. 3 Iss. 2, PP. 44-49.
9. Elmasri, R. (2011). Fundamentals of Database Systems, Arlington University of Texas: Addison-Wesley. Pratt, Philip J.; Last, Mary Z. (2014-09-08). Concept of Database Management (8 ed.). Course Technology.
10. JingTao Yao. (2008). An Introduction to Web-based Support Systems. J. T. Yao, An Introduction to Web-based Support Systems, Journal of Intelligent Systems, Vol. 17, No. 1-3, pp267-281, 2008.

11. Chris, W. (2018, December 30). How the World Wide Web (WWW) works? [Blog post]. Retrieved from https://www.explainthatstuff.com/howthewebworks

12. Patel, R., Sanghavi, S., Gupta, D., & Raval, M. S. (2015). CheckIt - A low cost mobile OMR system. TENCON 2015 - 2015 IEEE Region 10 Conference. doi:10.1109/tencon.2015.7372983

13. Omrhome. (2016). 5 Common Mistakes While Filling an OMR Sheet. Retrieved February 14, 2019, from https://create.piktochart.com/output/18411492-5-common-mistakes-while-filling-an-omr-sheet.

14. Rai.Skumar. (2013). Java is not very strictly platform independent. Retrieved February 15, 2019, from https://stackoverflow.com/questions/12337427/is-java-really-platform-independent.

15. Ever Y. K., Dimililer K., "The effectiveness of a new classification system in higher education as a new e-learning tool ", Quality & Quantity, Springer-Verlag, Vol. 52, Issue 1, pp. 573-582, Accepted: 2017, published: December 2018. DOI: 10.1007/s11135-017-0636-y

16. Dimililer K., "Use of Intelligent Student Mood Classification System (ISMCS) to achieve high quality in education", Quality & Quantity, Springer-Verlag, Vol. 52, Issue 1, pp. 651-662, Accepted: 2017, published: December 2018. DOI: 10.1007/s11135-017-0644-y

17. Sekeroglu B., Dimililer K., Tuncal K., "Student Performance Prediction and Classification using Machine Learning Algorithms", 2019 8th International Conference on Educational and Information Technology, ICEIT 2019.

DATA OF THE AUTHORS.

1. **Kamil Dimililer.** Assoc. Prof. BSc., MSc., PhD. Electrical & Electronic Engineering. Near East University, Nicosia, Cyprus. Chairperson in Automotive Engineering. Email: kamil.dimililer@neu.edu.tr

2. Getinet Amare Mekonnen. BSc. Software Engineering. MSc. Student. Debre Tabor University, Debre, Tabor, Ethiopia. Email: getinet.neu@gmail.com

RECIBIDO: 6 de junio del 2019. **APROBADO:** 19 de junio del 2019.