Scanner-Based Optical Mark Recognition

Chatree Saengtongsrikamon*, Phayung Meesad**, Sunantha Sodsee*

Abstract

The objective of this work was to develop Optical Mark Recognition (OMR) software for implementation in the simple scanner and for its usage as an OMR machine. It was developed using Java language. The software will help the assessor to capture and score responses of the multiple choices-answer sheets with accuracy and efficiency. Moreover, it was proposed to be a replacement for the costly OMR machine. The developed software was evaluated by simulating an actual implementation. It interpreted 1,000 answer sheers scanned by 5 different scanners in 4 different resolutions. The evaluation's results were favorable, with an estimated number of mistakes were less than 1 per 1,000 or 0.1 percent. As a result, the quality and accuracy of the software that has been developed is thus within acceptable limits.

Keyword: Optical Mark Recognition, Optical Mark Reader

1. Introduction

It is undeniable that the method of assessment is very important in any education system. The assessment is a powerful force in driving the way students learn. From the students' perspective, only the most important knowledge in a subject is assessed. Also, the teachers usually apply an appropriate assessment method to ensure that their students gain the maximum knowledge from the instruction. So by changing the assessment method, the various subject teachers can affect the way students learn the subject content.

Different ways of assessment provide the measurement of a student's capabilities in different contexts. Stiggins [1] groups the different methods of assessment into 4 main categories : Selected Response, Essays, Performance Assessment and Personal Communication.

The Multiple Choice Questions (MCQs), which is a method in the Selected Response category, is the most common method chosen to assess the students in primary and high school of Thailand because it facilitates the assessment of a broad range of learner knowledge in a short period of time. A large number of MCQs can be developed for a given content area, which provides a wide range of concepts that can be assessed consistently. Moreover, MCQs allow for test reliability if they are provided by a well-trained assessor.

To facilitate the assessor to capture and score responses of the student in MCQs, the answer sheets are implemented. This techniques help the assessor to reduce the time spend in looking for the students' responses in whole exam paper but only looking for it in the area provided in answer sheet that correspond to their answers the exam paper.

Optical mark recognition is the process of capturing data from the preparation form by recognizing the reflection patterns from the marked positions on a page. The reader or scanner is able to detect a marked area by shining a beam of light on the page to obtain the reflection patterns[2], [3]. One of the most familiar applications of optical marking recognition is the use of 2B-pencil to color the bubble on answer sheets in multiple choice question examinations. Students mark their answers by darkening circles marked on a pre-printed sheet. This sheet is then automatically graded by a scanning machine.

The scanning machine, which is also known as the OMR machine, can maintain a throughput of 1,800 to 10,000 forms per hour. This activity can be controlled and processed by a single machine that is more than enough for the primary and secondary schools. Indeed, the OMR machines are not scanners in the sense that they do not form an image of the sheets that pass through. Instead, the OMR device simply detects whether predefined areas are blank or have been marked. The fact that OMRs do not form an image differentiates them from desktop scanners that are often used to capture and form the image. This isalso the reason why OMR machines have the capacity to read sheets at a very high rate.

Although OMR machine provides a fast and accurate way to collect and input data, it is not suited for everyone's needs, primarily due to the cost of the machine and the MCQ-scoring sheets, which need to be bought in bulk by Print Services and is normally very costly. The reduction in cost of information technological devices have brought about the replacement of the pricey Optimal Mark Readers with simple scanners that cost less than 3,000 baht to score the multiple choice answer sheets. However, the scanner on its own is insufficient to meet this requirement, and accompanying software need to be developed to support this particular task.

OMR machines have no problems dealing with normal scanning problems that may arise such as Skew, Orientation, Scale factor, Offset and Contrast [4]. However, regular scanners are not designed to overcome such scanning problems. OMR machines traditionally minimize these problems from occurring by using a suitable hardware mechanism that fixes the problems of paper feeding. However, it is undeniable that the scanner has no mechanism to prevent such problems. Consequently, good OMR software should be combined with the suitable algorithm that can work on addressing these problems.

2. Methodology

In order to realize the principal purpose of developing software that is able to support the concept of this work, some processes/models would need to be adopted. Indeed, there are many software development models that have been adopted for the software development process. These models are

*Department of Information Technology Management , Faculty of Information Technology, KMUTNB.

**Department of Teacher Training in Eleetrical Engineering , Faculty of Technical Education, KMUTNB.

referred to as “Software Development Process Models”. This study has selected the “Waterfall Model” as the main structure of development.

The Waterfall Model was one of the first Process Model to be introduced and implemented successfully in Software Engineering. In the “Waterfall” approach, the whole process of software development is divided into separate process phases [5]: Requirement Specifications, Software Design, Implementation & Testing, Integration & System Testing and Deployment & Maintenance. Each of these phases has end products that flow on to the next phase to follow on development.This characteristic brings about the name “Waterfall Model”.

2.1 Requirement Specifications

All the requirements of the system that need to be developed are captured in this phase. The requirements are a set of functionalities and constraints that the end user expects from the system. These requirements are analyzedfor their validity and the possibility in the system. Finally, a Requirement Specification document is created which serves as a guideline for the next phase of the model.

The basic requirement specifications of the proposed software are:

2.1.1 The application should recognize and score a specific multiple-choice question answer sheet form correctly with acceptable reliability.

2.1.2 The application should be able to accept the grey level images input provided by the normal scanner in multiple resolution (at least 150 psi or higher)

2.1.3 The major scanning problems as listed below should be addressed and interpreted properly:

2.1.3.1 Skew
2.1.3.2 Orientation
2.1.3.3 Scale factor
2.1.3.4 Offset

The above basic requirement specifications come from the requirement of the general education institute’s need and the experience of author who has assigned to take care of professional OMR machine of Army Command and General Staff College.

2.2 Software Design

Before commencing work on the actual coding, it is critical to understand what we are going to create and what it should look like. In this phase, the requirement specifications identified from the Requirements phase are studied and a system design is prepared. The system design helps in specifying both hardware and system requirements, and also in defining the overall system architecture. The system design specifications as identified will serve as input for the next phase of the model.

Figure 1 depicts the implementation of the draft design specification application.

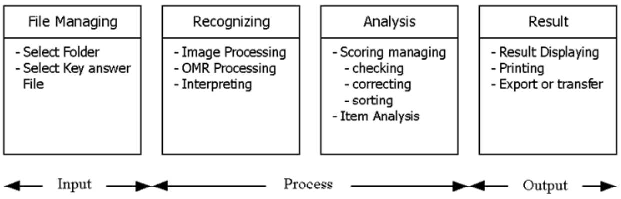


Figure 1: Design Specification

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 most important methodology for this module is image processing which is applied in the OMR field. The only one responsibility of this module is reading the marked area in the image file, then provide text/character answer accorded to the marked area. The second module arranges the procedure in compare the entire digital data produced by the earlier procedures in a file with the digital answers in the key file. The comparison allows the program to know the number of correct answers and record it as the total score of that answer sheet.

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.

2.3 Implementation & Unit Testing

After accepting the system design specification, the work is divided into modules/units and actual coding will commence. The system is first developed in small programs called units. Each unit is developed and tested for its functionality; this is referred to as Unit Testing.Unit testing mainly verifies if the modules/units meet their specifications. The units that meet the specifications are then integrated in the next phase.

In this work, the Design specification that has been identified earlier can be broken down into 5 modules as depicted in figure 2.

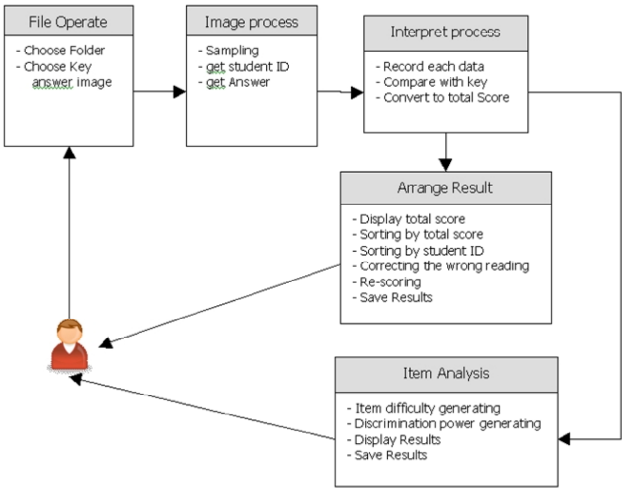


***Figure 2** : Design Modules*

The most important and complicated unit of the all above is Image process unit. It is considered as the main unit, which must translate the scanned images into the digital data correctly. There are two underlying logics that are used to translate the image to the digital data. The first is recognition concept that recognizes the selected answer of each item by counting the number of black pixels in the area. The process begins by first divides the area between two assigned points (see P1 and P2 in figure 3) into five small areas. Subsequently, the process detects the selected answer by comparing the number of black pixels in each area and selecting the area that has the greatest number of black pixels back.

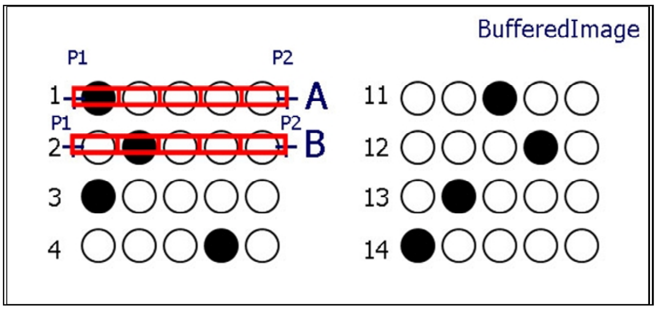


Figure 3: Recognition Concept

Second concept is that of scanning-problem detection and correction. The important factor that allows the first concept recognizes the selected choice correctly is the precision of assigning point P1 and P2. In the right shape and right position of the answer sheet form it is very easy to provide the set of points P1 and P2 of each item in advance. However, the set of assigned points will only work well if the images from the scanning have no error. Unfortunately, it is very hard to avoid all the various types of scanning errors. Once the errors happen, the set of assigned point is no longer aligned with that of the image. There are two solutions to deal with this problem, namely (1) transferring the image back to the proper position, and (2) moving the set of assigned point to the image that is out-of-alignment. The first solution has to pay more effort to transfer than the second one. As a result, this work is designed to apply the second solution.

The set of assigned points (P1 and P2 of each item) is generated after the four tick marks at the corners of the answer sheet are detected. This technique works as the solution to solve the major problems of scanning such as skew, orientation, offset and scale by regenerating set of points P1 and P2 by refering to four corner-tick marks. Figure 4 illustrates the regeneration of assigned points.

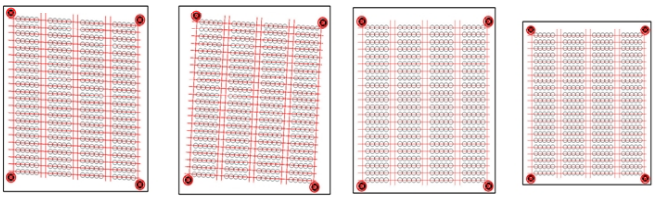


Figure 4: Detection and Correction concept

Figure 4 elaborates the broad concept applied to deal with the main problem of scanning. For more detail of implementation to this work will discuss step by step as the following sections

2.3.1 Detect the tick marks' location

As mention, the tick marks at the corner are used as the main reference to recognize the other points of the whole page so the most importance first step of the process must be the accurately detection of all tick marks. The equation 1 is used to find the x and y location of each tick marks

$$T(x,y)=\left(\frac{\sum Bx}{n},\frac{\sum By}{n}\right) \qquad \text{Eq 1}$$

where $T(x,y)$ is the location (x,y) of tick mark.

Bx is the x location of the black pixel in the detection area.

By is the y location of the black pixel in the detection area.

n is the number of black pixel in the detection area.

2.3.2 Locate the upper and lower horizontal referent point After detection the tick marks' location, the distance from top-left to top-right tick marks is divided uniformly into 8 points. (see Figure 5) and name it U1, U2, U3, … U8.

Same process as early mention, the points L1 to L8 are assigned from the low-left and low-right tick marks in the similar ratio.

2.3.3 Find item's referent point

Item's references are the point used to find out the result of each item in whole paper that is mentioned early on. Each point is detected by dividing the range between point Un and Ln (n=1,2,3,…,8) into 25 points and name it Pn-m (m=1,2,3,…,25). Those points are assigned as P1 and P2 of each item for translating the items' result by recognition concept.

2.4 Integration & System Testing

As specified above, the modules in the system are developed separately and tested individually for their functionalities. Once that is completed, the various modules are integrated into a complete system in the Integration phase and a system connectivity test is done to check that all modules/units are well-connected and the system as a whole behaves in accordance with the design specifications. Once the system connectivity has been validated, the System design can proceed to the next phase. Figure 5 describes the functions of each module after the Integration phase.

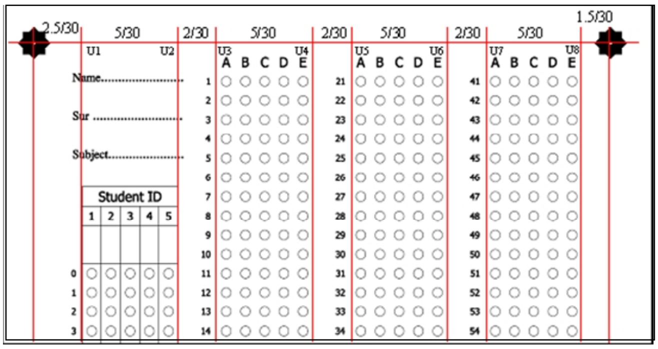


Figure 5: Locate the upper point

For this program, the validation process was not only designed for ensuring that the entire units work together properly but also guaranteed the results of interpreting with

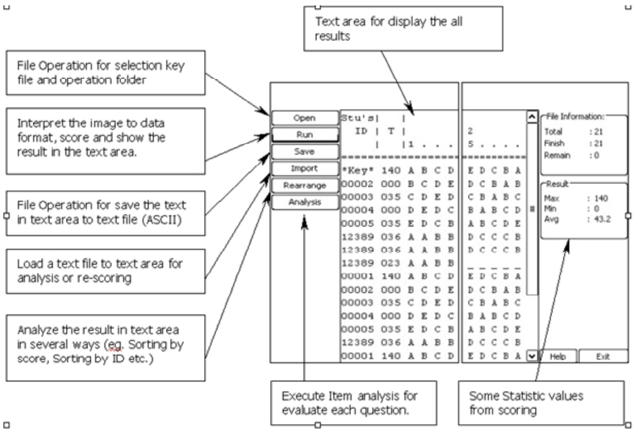

Text area for display the all results
File Operation for selection key file and operation folder
Interpret the image to data format, score and show the result in the text area.
File Operation for save the text in text area to text file (ASCII)
Load a text file to text area for analysis or re-scoring
Analyze the result in text area in several ways (eg. Sorting by score, Sorting by ID etc.)
Open
Run
Save
Import
Rearrange
Analysis
File Information:
Total : 21
Finish : 21
Remain : 0
Result
Max : 140
Min : 0
Avg : 43.2
Help
Exit
Execute Item analysis for evaluate each question.
Some Statistic values from scoring

Figure 6 : Modules integration

accuracy and reliability. The software was evaluated by executing repeatedly though the simulated using. To achieve the objective of evaluation, the program executed the number of images, which are scanned from the answer sheets marked by different students. This experiment was used to check the ability of program in overcoming the problem via human's behaviours. Because particular person always has his own style in coloring bubbles on the answer sheet as a result, this may effect on the recognition process of program.

Moreover, the application must be evaluated in dealing with the all-major problems of scanning via simple scanner. The testing, designed for looking into this solution, is arranged with scanning the entire answer sheets, which were mentioned earlier, by different scanners in different resolutions. In this evaluation, 5 different scanners were provided for scanning all papers with 4 resolutions (600 dpi, 300 dpi, 150 dpi and 75 dpi).

2.5 Deployment & Maintenance

This phase of the "Waterfall Model" is virtually a never-ending phase. Generally, problems with the system that has been developed, which are not found during the development life cycle will surface only after the System is deployed and utilized. As such, these issues can only be identified and resolved after the system has been deployed. Again, not all the problems will surface immediately upon implementation, but they arise from time to time and will need to be solved there and then maintenance is an essential part of this process.

For this study, this phase will be limited as the number of assessors is limited As a result, it is anticipated that some of problems of application will not be identified and resolved. However, there is still a requirement to run through this phase of the application, as even in the limited form, new requirements could still be identified that may need to be incorporated withinthe Design requirement specifications.

3. Experimental Results

3.1 System Design

The software comprises of 6 major functions: Open, Run, Save, Import, Rearrange, and Analysis. The Open function is used for selecting the working folder and identifying the key answer file. The answer sheet files are image files saved in ".jpg" format. Those files can be created with the scanner, camera or other devices that can change the paper answer sheets to the image files. The Run function is used to execute the process of recognizing image files and compare each image file in the working folder with the key image file in order to score those files one by one. The Save function translates the data in text area produced by the 'Run function' to text file. The end user can read this text file with any text editor software (e.g. notepad, wordpad, Microsoft Word etc) or convert to a Microsoft Excel spreadsheet to facilitate any analysis of the data if they so wish. The Import function helps user to load the text file created by the 'Save function' into area text for changing, correcting or analyzing. The two important sub functions in the Rearrange function are sorting and re-scoring. The first one is used in arranging the result in the area text by score or ID number. The re-scoring is used when changes to the answers have been

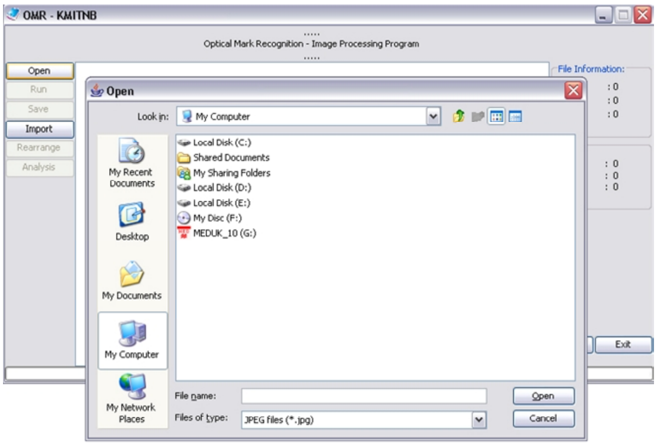
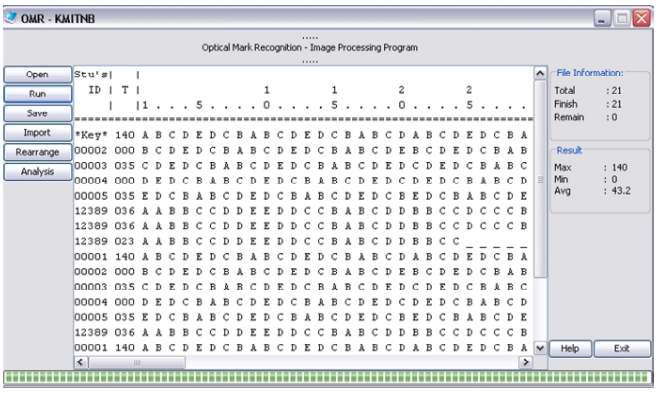


Figure 7 : Open Function

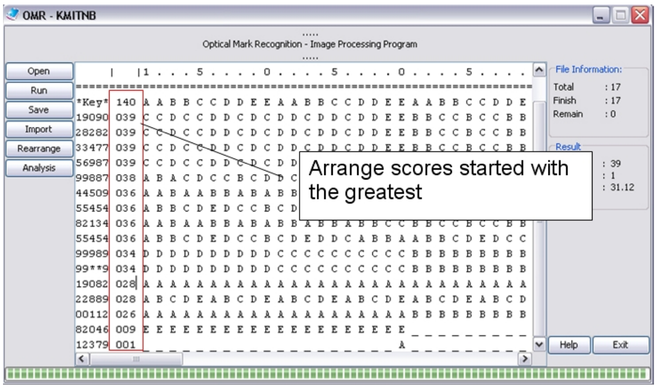


```
Stu's|    |
  ID | T  |                 1         1         2         2
     |    |1 . . . 5 . . . . 0 . . . . 5 . . . . 0 . . . . 5 . . . .
==================================================================
*Key* 140 A B C D E D C B A B C D E D C B A B C D A B C D E D C B A
00002 000 B C D E D C B A B C D E D C B A B C D E B C D E D C B A B
00003 035 C D E D C B A B C D E D C B A B C D E D C D E D C B A B C
00004 000 D E D C B A B C D E D C B A B C D E D C D E D C B A B C D
00005 035 E D C B A B C D E D C B A B C D E D C B E D C B A B C D E
12389 036 A A B B C C D D E E D D C C B A B C D D B B C C D C C C B
12389 036 A A B B C C D D E E D D C C B A B C D D B B C C D C C C B
12389 023 A A B B C C D D E E D D C C B A B C D D B B C C _ _ _ _ _
00001 140 A B C D E D C B A B C D E D C B A B C D A B C D E D C B A
00002 000 B C D E D C B A B C D E D C B A B C D E B C D E D C B A B
00003 035 C D E D C B A B C D E D C B A B C D E D C D E D C B A B C
00004 000 D E D C B A B C D E D C B A B C D E D C D E D C B A B C D
00005 035 E D C B A B C D E D C B A B C D E D C B E D C B A B C D E
12389 036 A A B B C C D D E E D D C C B A B C D D B B C C D C C C B
00001 140 A B C D E D C B A B C D E D C B A B C D A B C D E D C B A
```

```
File Information:
Total   : 21
Finish  : 21
Remain  : 0

Result
Max     : 140
Min     : 0
Avg     : 43.2
```

Figure 8: Sample Results from Run Function



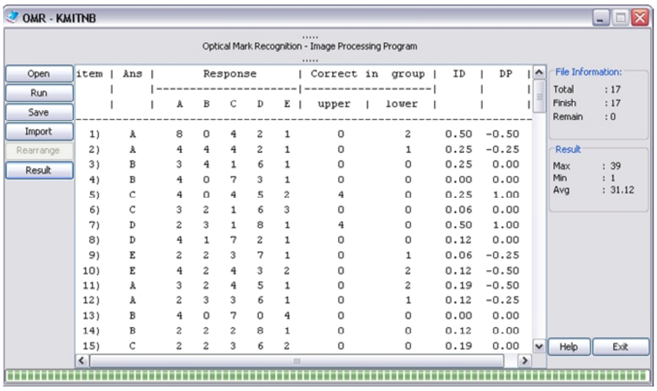
```
     |    |1 . . . 5 . . . . 0 . . . . 5 . . . . 0 . . . . 5 . . . .
==================================================================
*Key* 140 A A B B C C D D E E A A B B C C D D E E A A B B C C D D E
19090 039 C C D C C D D C D C D D C D D C D D E E B B C C B C C B B
28282 039 C C D C C D D C D C D D C D D C D D E E B B C C B C C B B
33477 039 C C D C C D D C D C D D C D D C D D E E B B C C B C C B B
56987 039 C C D C C D D C D C D D
99887 038 A B A C D C C B C D D C
44509 036 A A B A A B B A B A B B
55454 036 A B B C D E D C C B C D
82134 036 A A B A A B B A B A B B A B B A B B C C B B C C B C C B B
55454 036 A B B C D E D C C B C D E D D C A B B A A B B C D E D C C
99989 034 D D D D D D D D D D C C C C C C C C C C B B B B B B B B B
99**9 034 D D D D D D D D D D C C C C C C C C C C B B B B B B B B B
19082 028 A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
22889 028 A B C D E A B C D E A B C D E A B C D E A B C D E A B C D
00112 026 A A A A A A A A A A A A A A A A A A A A B B B B B B B B B
82046 009 E E E E E E E E E E E E E E E E E E E E _ _ _ _ _ _ _ _ _
12379 001 _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ A _ _ _ _ _ _ _ _ _
```

```
File Information:
Total   : 17
Finish  : 17
Remain  : 0

Result
        : 39
        : 1
        : 31.12
```

Arrange scores started with the greatest

Figure 9: Sample Results of Rearrange Function



```
.....
Optical Mark Recognition - Image Processing Program
.....
item | Ans |        Response         | Correct in  group |  ID   |  DP   |
     |     |-------------------------|-------------------|       |       |
     |     |  A    B    C    D    E  |  upper  |  lower  |       |       |
-------------------------------------------------------------------------
  1)    A     8    0    4    2    1       0         2     0.50   -0.50
  2)    A     4    4    4    2    1       0         1     0.25   -0.25
  3)    B     3    4    1    6    1       0         0     0.25    0.00
  4)    B     4    0    7    3    1       0         0     0.00    0.00
  5)    C     4    0    4    5    2       4         0     0.25    1.00
  6)    C     3    2    1    6    3       0         0     0.06    0.00
  7)    D     2    3    1    8    1       4         0     0.50    1.00
  8)    D     4    1    7    2    1       0         0     0.12    0.00
  9)    E     2    2    3    7    1       0         1     0.06   -0.25
 10)    E     4    2    4    3    2       0         2     0.12   -0.50
 11)    A     3    2    4    5    1       0         2     0.19   -0.50
 12)    A     2    3    3    6    1       0         1     0.12   -0.25
 13)    B     4    0    7    0    4       0         0     0.00    0.00
 14)    B     2    2    2    8    1       0         0     0.12    0.00
 15)    C     2    2    3    6    2       0         0     0.19    0.00

File Information:
Total    : 17
Finish   : 17
Remain   : 0

Result
Max   : 39
Min   : 1
Avg   : 31.12
```

Figure 10 : Sample Result of Item analysis process

done and a re-scoring of the answers is needed. The last function, which is the Analysis function, allows the user to conduct an Item analysis.

3.2 Evaluation Results

After developing the proposed software, this program is tested by the author to ensure that all functions work correctly. The next step is the evaluation. The main propose of the evaluation is to check the software for accurate results. This is the most important process in checking for software reliability as incorrect results could potentially result in major problems and is thus highly undesirable.

The software was evaluated by executing the runs repeatedly. The images were scanned by 5 different scanners. The scanner models are namely: Canon MX318, Lexmark-X6170, Epson CX5500, HP LaserJet3020, and Canon LiDE20. Each scanner was used to scan the test images in 4 different resolutions (600 dpi, 300 dpi, 150 dpi and 75 dpi). After scanning 50 test-answer sheets marked by 50 different students with 5 scanners in 4 resolutions, the 1,000 images (50x5x4) were produced and ready to be executed by the program. As a result, there were 1,000 images interpreted via this valuation and there were no wrong responses detected from the execution. Consequently, The results of the evaluation were favorable, with an estimated number of mistakes of less than 1 per 1,000 or 0.1 percent. The quality and reliability of the software that has been developed is thus within acceptable limits.

4. Conclusion and Future Work

The Optical Mark Recognition (OMR) was developed for facilitating the MCQ scoring. The OMR machine or Optical Mark Readers are used to capture and score responses on scoring sheets that are then used by the assessor for analysing and reporting. It is a powerful tool in scoring with accuracy and efficiency, but it is also a very expensive tool that has not seen a substantial reduction in cost even as new technologies have constantly evolved to bring technological costs lower than ever. The new OMR machines are able to perform faster and with better accuracy than the old ones. However, even as the new model is priced at the same price as the earlier versions, this is still too expensive to be commonly used in the smaller sized schools or education institutes. The problem is made worse, as not only are the machines expensive, the scoring sheets are also costly.

The problem is proposed to be resolved by using a normal scanner as a replacement to the costly OMR machine. The proposed software is able to work properly with reliability and accuracy. Moreover, the operating cost is further reduced as the scanner does not require the special OMR-scoring sheets, as it is able to work with plain photocopying paper.

After the evaluation, the program can be improved by the following features:

- Adding a 'Print' function, which allows the users to print the report or the blank answer sheets directly from the software.

- Adding a 'Create Sheet-form' function. The current template adopted for the answer sheet is to facilitate the purpose of this study and testing As such, it may not be suitable for all occasions. The 'Create Sheet-form' function allows the users to produce their own answer sheet format that differs from the standard template, such as increasing or decreasing the number of items from the template figure of 140, or adding more to the current 5 MCQ choices.

- Adding a 'Grade' function, which allows the students' overall MCQ grade to be scored automatically.

5. References

[1] Stiggins, R. **Student-involved assessment for learning.** Upper Saddle River, N.J., Prentice Hall, 2005.

[2] Bergeron, Bryan P. (1998, August). **Optical mark recognition.** Postgraduate Medicine online. June 7, 2006.

[3] Haag, S., Cummings, M., McCubbrey, D., Pinsonnault, A., Donovan, R. **Management Information Systems for the Information Age**, 3rd ed., Canada: McGraw-Hill Ryerson, 2006.

[4] Gonzalez, R. C. and Wintz, P. **Digital Image Processing**, 2nd ed., Addison-Wesley, Reading, Massachusetts, pp. 30-31 and 130-134 , 1987.

[5] McConnell, Steve . **Software Estimation: Demystifying the Black Art**. N.C., Microsoft Press, 2006.

[6] Heywood John, **Assessment in Higher Education**. Jessica Kingsley Publishers, London, Pages 350-372, 2000.

[7] TAWIP. Image Capture, **Image Recognition and Image Processing**.[Online],(n.d.).,Available from : http://www.tawpi.org/image-capture-recognition-processing.html

[8] Payne DA. **The Assessment of Learning — Cognitive and Affective**. Lexington, Ky: D.C. Heath and Company: 255,272, 1974.

[9]. Lowe Doug. **Java™ All-in-One Desk Reference For Dummies**. River Street, Hoboken, Wiley Publishing, 2005.

[10] French, Christine L. "A Review of Classical Methods of Item Analysis". **Presened at the annual meeting of the Southwest Educational Research Association**, New Orleans, 2001.

[11] Raksakietisak Sunee. **The Development of Online Item banking and Testing System**. Masters Thesis, Faculty of Science, Srinakharinwirot University, 2003.

[12] Pornsiriprasert Nisarath. **Design and development of page segmentation program for character recognition**. Masters Thesis, Chulalongkorn University, 2002.