An Innovative Idea Of Matrices Problem Solver Using Image Processing

Ameer Hamza Sajjad[*], Muhammad Saqib[†], Sehrish Munawar Cheema[‡], Taskeen Haider[§] and Bushra Younas[¶]

[*],[†],[§]Department of Software Engineering, University of Gujrat, Sialkot, Pakistan
[‡]Department of Software Engineering, University of Management and Technology, Sialkot 51040, Pakistan
[¶]Department of Mathematics, University of Sialkot, Pakistan
[*]xamzasajjad@gmail.com, [†]miansaqib7862@gmail.com, [‡]sehrishcheema@gmail.com,
[§]taskeenhaider126@gmail.com, [¶]bushra.younas@uogsialkot.edu.pk

Abstract— Math is the cradle of all creations, without which the world cannot move an inch. Mathematical functions are used in software engineering, civil and mechanical engineering and in many other fields as well. A problem faced by the students is to derive the solution for a matrix problem using pen and paper-based approach that is time taking process, also for a solver to find an exact formula to solve specific matrix is problematic. Our major goal is to bridge the gap between traditional pen-based methods and technology by developing an android application solver using image processing for matrix problems. Matrix related operations like determinants, transpose, ad-joint, inverse, Cramer rule will be solved in step by step manner for user better understanding. While calculating determinants if matrices include any trigonometric terms as elements it will be solved by considering trigonometric formulas. User will scan the image of matrix problem statement holds matrices up to 10x10 dimensions. Matrix recognition is categorized into two classes i.e. computer printed matrix and hand written matrix. Our application will recognize only computer printed form. After getting the image from camera, OCR will get text from the image and our algorithm will solve the required problem and show the result to user in step by step manner. Our research work is a new dimension for using image processing to solve matrix problems.

Index Terms—Android applications, OCR technology, Matrices problems, Computer assisted learning, Mathematics, Education

I. INTRODUCTION

Today's machine has the capability of looking and understanding things, just like as human [1]. This is achieved through image processing and machine learning; the process of learning begins with observation of data to make better decision in future. The primary aim is to allow the computers learn automatically without human intervention. This application is aimed at using this capability for looking new age problems. Mathematical equations have become bread and butter for engineers and scientist [1]. Mathematics is a broad field; in this application we have set a specific domain to solve the matrices questions. Matrices play important role in all branches of science, engineering, social science and management [2]. In matrices we have set further domain like determinants, adjoint, inverse and system linear equation. There are multiple algorithms and different techniques for solving functions in a faster and efficient manner. Many programming languages (c, c++, java, python etc.) give support for mathematical equation solving with pre-defined function and libraries [1].

Now a days computer programs play a very important role in solving equations. There are many applications and websites available in the market, which can solve a given equation and return result. These applications and websites are manually it is very time-consuming process, user enter value one by one if a matric is 4x4 then a user enters 16 values. Just think about the scenario where they want to solve the same question. This application is unique in this way and this is a new dimension of using image processing to solve matrix problem and provide solution more than 2/3 methods of single problem if it's is possible. To be precise, our application captures an image of printed matrix from the built-in camera of user mobile up to 10x10 dimensions, the main reason to select the dimension is that most of students and engineers uses up to 10x10 dimension of matrices. The image is pre-processed to arrive at the point of interest and user can crop the image. After the resizing of image, noise of the picture will be removed and any other background information is removed. After that OCR will extract text from image [3] [4] and the text of matrix will be available on the edit filed, if user wants to edit the text otherwise the user selects list of above operations that are defined.

II. LITERATURE REVIEW

During the detailed literature survey, a few related research works identified that smartphones, online applications and tools have become integral part of students and teachers lives all around the world [5]. These applications and tools have transformed the way for people to communicate and work together. It was a challenge for technologists, researchers and educators to explore the mobile techniques to support online learning [6]. In front of the idea of inclusion of mobile learning in educational process, many case studies were examined

to estimate the consequences of online mathematic learning tools and applications at all educational levels [7].

Kalloo and Mohan developed a mobile application named MobileMath for secondary school students to enhance their performance in algebra. This application works using internet and supports students by providing game-based learning, tutorials, quizzes, examples problems. These game-based learning improve their algebraic skill set. A short survey on about 57 students was conducted. Data was collected via interviews, questionnaire, pre and post-tests. Three evaluation studies represented students increasing learning level especially by game-based learning [8].

Markus Hohennwarter et al started a project at the University of Salzburg in 2002. They designed an application named "GeoGebra" by combining the features of geometry software and computerized algebra systems [9]. This application was designed for teachers for demonstration, to prepare lecture material and visualization for mathematics. For years they use computer algebra system (CAS) for computation of symbols. GeoGebra was revised for symbolic features to compute undefined variables, fractions and equations. Latest version of GeoGebra came with a fully dynamic three-dimensional (3D) graphics and geometry [10].

In [11] Amey Chavan and Asad Naik demonstrated an android application that uses the OCR technique to recognize the captured image by built-in camera of app. This application recognizes simultaneous linear equation image and compute the solution by binarize the image using OTSU algorithm. Text recognition is done by Google Tesseract. They also demonstrate the method for optimizing image for OCR accuracy. Once the user confirms the equation system solves it using parser and Cramer rule.

Every year researchers present efficient algorithms, theorems and paradigm to perform mathematical calculations in an intellectual way to improve the performance of mathematical implementation like sorting and searching [12] [13]. Researchers are investigating and designing more accurate theorems to compute the matrices determinant. In [14] a simple Method of Compression was presented that reduce a determinant to another of lower order; this would lessen the labor of its evaluation. Wearable learning is basically a game-based learning in which student can play the mathematical games. This will help the student to engage with computation thinking [15].

Image processing is a technique used to segment the images and perform specific function on image. Optical character recognition or optical character reader (OCR) is a technique used to extract the text from an image. Sometimes OCR called as a tesseract. The text of images can be either printed form or handwritten form, for this SVM trained model was created which is used to extract and recognize hand written text using fuzzy technique. All these techniques were developed and implemented in an application to make it easy for a user to solve the math's problem. User can take picture of image and using image processing and tesseract algorithms to run and solve the required question. These techniques were all used in an application because many students who has poor thinking of any formula or cannot remember steps to solve the questions. This application can help in to remember how to solve the question [16] [17] [18].

Popular programming languages provide wide-ranging pre-defined functions and libraries to support mathematical equation solving. In [1] they develop an android application that feed an image of mathematical expression using built-in camera and produces a java code corresponding to feed expression that would be utilized for multiple inputs of their needs. Along with the code, this application also displays the basic plot of function that will be helpful for choosing solution to entered problem. There are many applications in the market which can compute the result of a mathematical equations.

In article [19] Armend Salihu they designed and demonstrated a theorem to compute the determinants of square matrices of order 5 and 6. Presented theorem was a combination of Farhadian's Duplex Fraction theorem and Salihu's method to reduce the order of determinants to a second order. In short, this article leads the possibility to calculate determinants of higher order.

Many applications have been proposed and developed focusing to different aspects of mathematics like algebra, linear equations, laws of arithmetic's, geometry, matrices and general math skill learning and improving apps [7] shown in figure 1. We found that many of these applications have no accuracy and precision. A valuable comparison among few of them, related systems and applications have presented in table I.
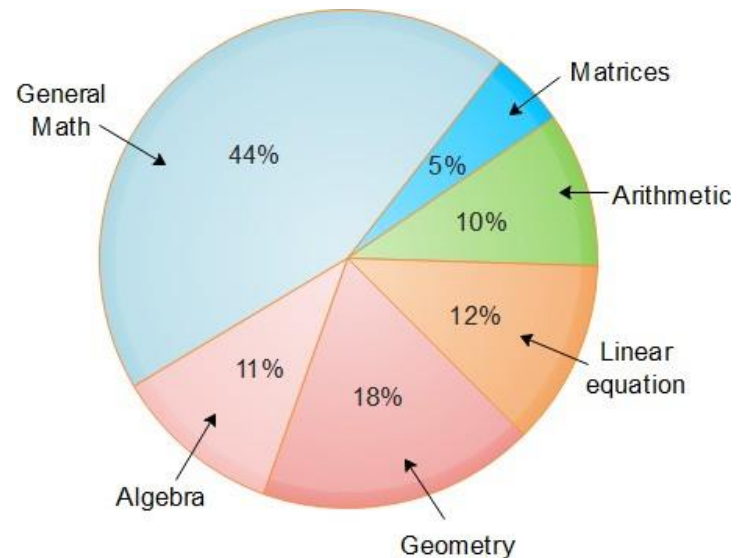

Figure 1: Development of math application in %.

III. APPLICATION DESIGN AND ARCHITECTURE

The proposed system has three tiers referred as application tier, logic tier, data tier shown in Figure 10. At top level application tier present all the interfaces to interact with user. The next tier is logic and connectivity tier that encompass all logic of problem solver which is based on major features provided by system. At the bottom data tier provides repository for intended system.

TABLE I: Comparison with existing Systems

	Features	
References	Techniques	Problem Domain
[1]	OCR, code generation and function plotting	Logarithm, trigonometric, polynomial and exponential problem solving
[20]	Online survey in Hungarian and Slovakian Languages both are launched in Slovakia	Lack usage of mobile devices in mathematics field
[21]	Dodgson condensation, Chois condensation method	Matrix containing more than 3x3 dimensions solving
[22]	Image processing, Artificial intelligence with fuzzy and KNN techniques	Algebraic, Binomial, Polynomial and Exponential equations solving
[9]	GeoGebra technique (A way of calculating algebra, geometry, statics and calculus using an application)	Making objects according to math rules
[16]	Image Processing, OCR and SVM model trained for hand written recognition	Solve arithmetic operations and linear equation
[23]	RETAIN design model used	Good way to teach students about the basic of algebra
[24]	Manually data entering	Matrix operations solving using an offline application
[25]	Online matrix solving with manual input	Inverse, Adjoint, Addition, Multiplication, Subtraction, Transpose solving problems handling
[26]	Model base OCR for hand written text extraction, trained model for plotting graph, image processing and artificial intelligence	Almost cover algebra up to 4 variables

IV. PROPOSED SOLUTION

Keeping in view the math's learner problem related to matrix, our application will provide an easy platform to solve matrix questions with in our domain. Many students of higher education and secondary education, engineers and scientist are using matrix operation. Many of them might not think about a formula or steps to solve the matrix question. Like if a student wants to solve a determinant using cofactor and he cannot remind the steps he can easily remind them by using

"MatMath", where user can find step by step solution of the given question. We have included determinants (Using Butterfly method, Minor and Cofactor method, Dodgson method, Compression method), Cramer rule, Transpose, Adjoint, Inverse. These are the most common operations of matrix that we mostly used. We will solve matrix up to 10x10 dimension and for Cramer rule up to 3x3 dimension.

- Design Algorithms

  In this application we are focusing on different methods to solve determinants. After getting the command of determinant from user, system will ask for the given list of determinant methods solving. If a user selects butterfly method then system will solve the question with butterfly method, this method is solved using algorithm A. If a user selects any other methods then, different algorithms will be called according to the selected method. Standard method of solving determinant is the simplest and most common method of solving determinant. In this application we have also included the standard method of solving determinant. Solving the determinant through standard method include the following activity flow. After the selection of standard method from method selection menu, new activity will start. First the system will check if the number of rows and number of columns are equal which means that matrix is square matrix, in case if they are not equal system will ask the user to check there
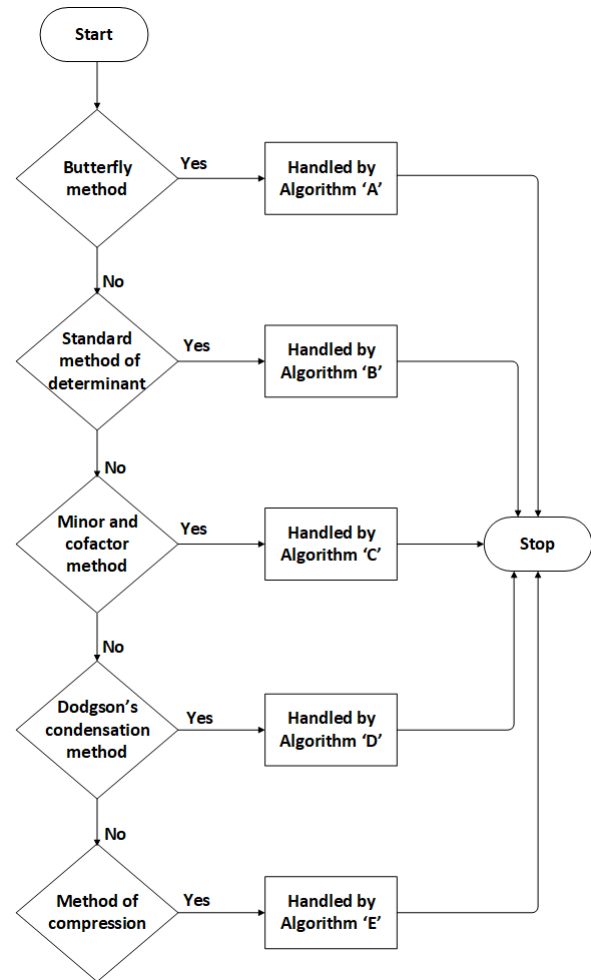

Figure 2: Core algorithm of determinant.

question again or this question cannot be solved with standard method of solving determinant. After successful checking of rows and columns it's time to start solving the problem. First step include in this method is to skip 1st row and 1st column, after that the first element of row 1st is multiplied by the corresponding elements and value is store in variable A. Next step is to skip 1st row and 2nd column, after that first element of column 2 is multiplied by the corresponding elements and value is store in variable B and so on further elements are skipped according to mathematical rule.
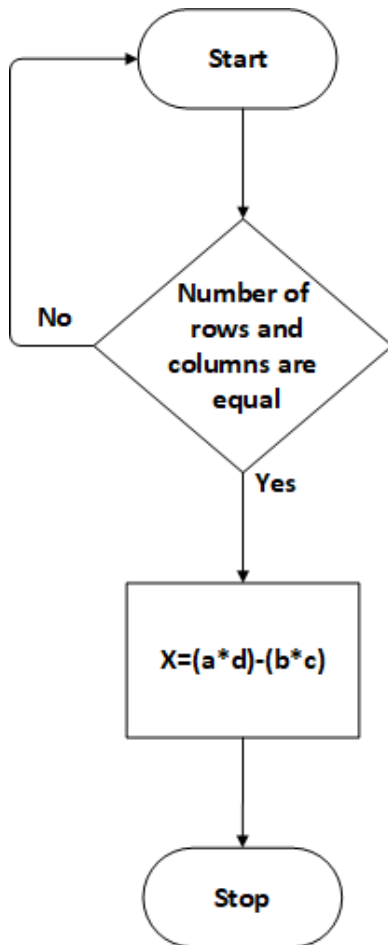

Figure 3: View of algorithm A.

Next step is to skip 1st row and 3rd column, after that first element of 3rd column is multiplied by the corresponding elements and value is store in C. After that solution will be calculated using formula result= A-B+C. Where A is 1st element of 1st row, B is 2nd element of 1st row and C is 3rd element of 1st row.

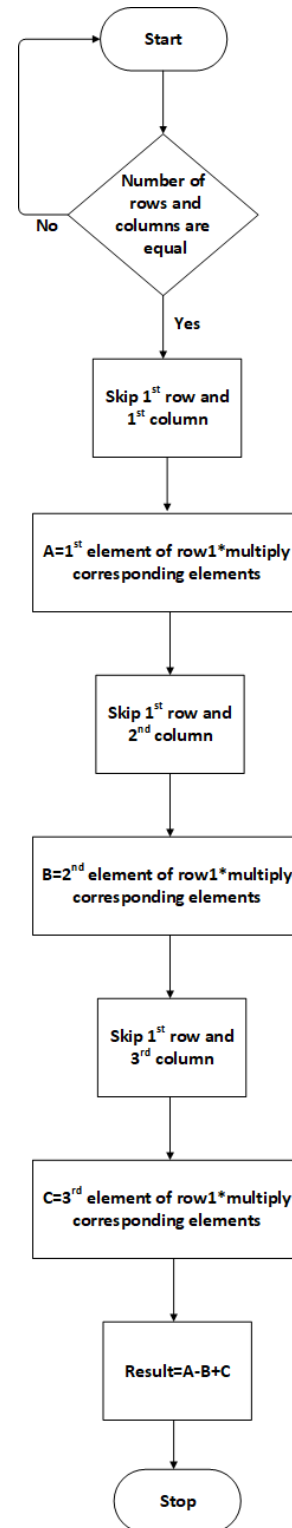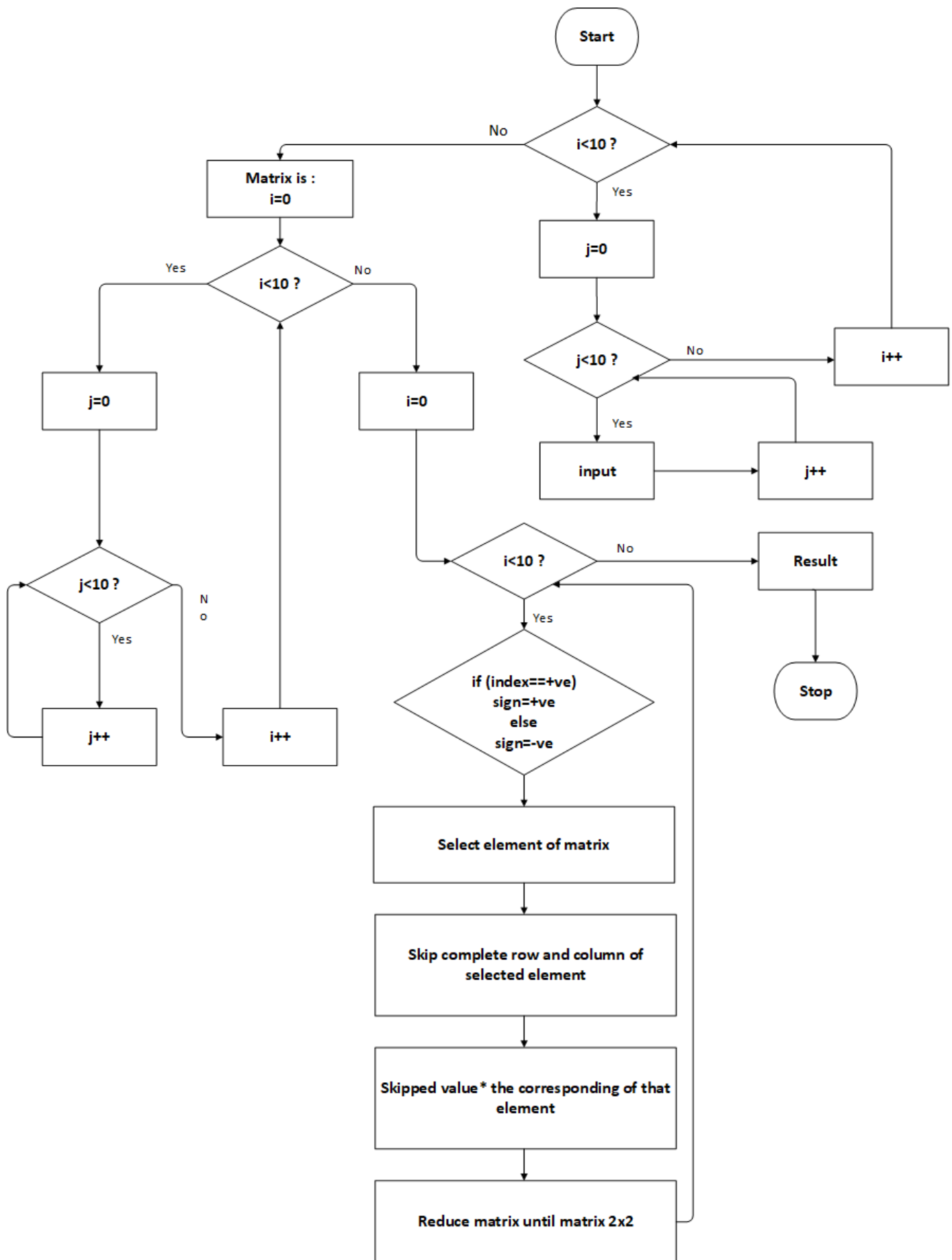
Figure 4: View of algorithm B.

Figure 5: View of algorithm C.

A. Capture and scan image

At the first time, the proposed application requires permission for the user to take image from camera or gallery. If the result is not shown after two to three seconds, then either the math problem is not supported or the application is unable to properly recognize the text (for example due to poor light conditions or camera not being clean) is shown in figure 6.
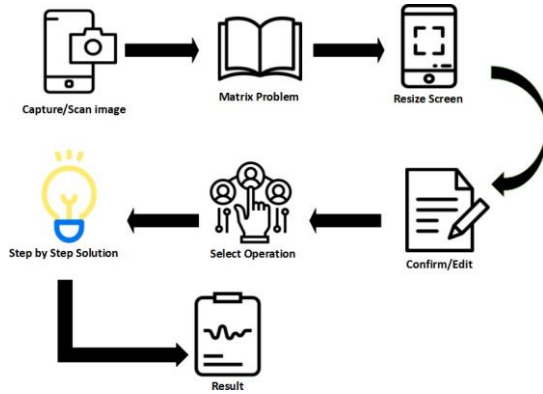

Figure 6: Functional architecture.

B. Crop Image

At initial step, if a user wants to resize the window then user will tap and hold with one finger at the any corner of rectangle and drag it any direction to resize the screen window.
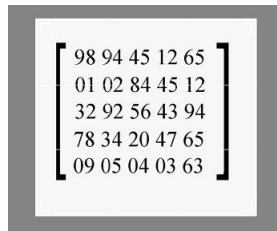

Figure 7: Crop image.

C. Noise Removal

Background noise is one of the deadliest backlogs for an application [1]. It is a very problematically situation when the image has some irrelevant content, firstly it is correctly identified and then remove it. Fourier transform [27], Wiener filter and Gabor filter are used to remove background noise from the image and only the text is retained. Otsu's adaptive thresholds was used to threshold the image. Color image can also be a threshold. There are many approaches to separate the threshold for each of RGB value of image and combine with AND operation [28]. The algorithm supposed that the image is to be threshold, containing two classes of pixels and then the threshold converts the image into two-bit image just containing white and black pixel.
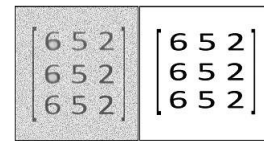
Before Noise   After Noise

Figure 8: Noise removed.

D. OCR

OCR main purpose is to extract text from an image after the noise removal. There is notable pit fall to use OCR, for that we have added an edit text field where user can check the given image and given text extracted from OCR are same, if the extracted text is wrong user can manually edit the question or capture the image again.

E. Edit Text

The scanned text is shown at the edit text field. If the user wants to modify a scanned problem, tap the edit icon and edit the text manually.

$$A = \begin{bmatrix} 3 & 7 \\ 6 & 4 \end{bmatrix} \qquad A = \begin{bmatrix} 2 & 7 \\ 6 & 4 \end{bmatrix}$$

Before edit of matrix *A*.   After edit of matrix *A*.

Figure 9: Edit text.

Before the edit, the 1st element of matrix is 3 and after the text edit of matrix: the 1st element is 2.

F. Select method

After the modify text, application allow the user to select the method (for example a user want to calculate the determinant of a matrix it provides a list of method).

1) Butterfly:

$$B = \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix}$$

B= (1)(4)-(2)(3)
B= 4-6
B= -2

In order to find the determinant of 2x2 matrix, At first multiply the diagonal element of left to right and then subtract the diagonal product of right to left element of matrix as shown in Figure 3. If matrix is 3x3 then, it applies Sarru's method.

2) Standard method of solving determinant: The standard method to find the determinant of a 3x3 matrix is breakdown into smaller 2x2 determinant then 2x2 determinant which can be easily handle as shown in Figure 4.

$$B = \begin{vmatrix} 1^+ & 2^- & 3^+ \\ 4^- & 5^+ & 6^- \\ 7^+ & 8^- & 9^+ \end{vmatrix}$$

3) Minor and Cofactor: The minor Mij of the element aij in a given determinant is the determinant of order (n – 1 x n – 1) obtained by deleting the ith row and jth column of Anxn. The scalars Cij = (-1)i+j Mij are called the cofactor of the element aij of the matrix A.

The value of the determinant in matrix can also be found by its minor elements or cofactor, as a11M11 – a12M12 + a13M13 or a11C11 + a12C12 + a13C13. Hence the det of matrix is the sum of the elements of any row or column multiplied by their corresponding cofactor as shown in Figure 5.

4) Dodgson's condensation: The method then consisted of four steps:

1. Use elementary row and column operations to remove all zeros from the interior of A.

2. Find the 2×2 determinant for every four adjacent terms to form a new $(n–1) \times (n–1)$ matrix B.

3. Repeat this step to produce an $(n–2) \times (n–2)$ matrix, and then divide each term by the corresponding entry in the interior of the original matrix A, to obtain matrix C.

4. Continue "condensing" the matrix down, until a single number is obtained. This final number will be det A [29].

5) Method of Compression: In this method we can reduce a determinant to another of lower order and we find a new determinant of a lower order, when multiply by one of its original element and select any element as a master element of its row and column, divide the master element into the other elements of its [14]

G. Step by step solver

In this application user facilitate to see more than one solution of the same problem. After the selection of method, text will be show in the edit text field. That text is sent into python (it supports mathematic libraries and function for example, numpy: this library is used to solve mathematical problems and provide some built-in functions like det(A) where A is a matrix of any dimension and det(A) is calculate the determinant of matrix. There is notable pit fall to use built-in functions of numpy library which is, that it provides only answers not step by step solution. Algorithms that are used to solve the problem will be design and selected method will be called to the same function and it will be solved after that output sent into android application. Application has some additional feature for example, In "MatMath" we have added a mathematical dictionary. The user will find any definitions and formulas about mathematics and it will also use calculator to solve arithmetic problem which will be stored in history panel.

H. Algorithm to compute determinant of 4x4 matrix

$$A = \begin{bmatrix} 25 & -15 & 23 & -5 \\ -15 & -10 & 19 & 5 \\ 23 & 19 & -15 & 9 \\ -5 & 5 & 9 & -5 \end{bmatrix}$$

1) Step 1: Create a function to solve 3x3 matrix.

$$def\ three\ by\ three\ (m):$$

$$one = (m[0][0]) * (m[1][1] * m[2][2] - m[2][1] * m[1][2])$$

$$two = (-m[0][1]) * (m[1][0] * m[2][2] - m[2][0] * m[1][2])$$

$$three = (m[0][2]) * (m[1][0] * m[2][1] - m[2][0] * m[1][1])$$

det = one+two+three
return det

```
a = [
    [m[1][1], m[1][2], m[1][3]],
    [m[2][1], m[2][2], m[2][3]],
    [m[3][1], m[3][2], m[3][3]]
]
a = (m[0][0]) * (three_by_three(a))
```

- Step1
  Get remaining matrix from parent matrix.

$$A = \begin{bmatrix} -10 & 19 & 5 \\ 19 & -15 & 9 \\ 5 & 9 & -5 \end{bmatrix}$$

- Step 2

  Passing A as an argument to function

$$three\_by\_three\ (A)$$

- Step 3
  Given function calculates the matrix shown in Step 1.

- Step 4
  After performing Step-3, calculated value is multiplying by 1st element of skipped row and column.

```
b = [
    [m[1][0], m[1][2], m[1][3]],
    [m[2][0], m[2][2], m[2][3]],
    [m[3][0], m[3][2], m[3][3]]
]
b = (-m[0][1]) * (three_by_three(b))
```

- Step 1
  Get the remaining matrix from parent matrix.

$$B = \begin{bmatrix} -15 & 19 & 5 \\ 23 & -15 & 9 \\ -5 & 9 & -5 \end{bmatrix}$$

- Step 2
  Passing B as an argument to function

  three by three (*B*)

- Step 3
  Given function calculates the matrix shown in Step 1.

- Step 4
  After performing Step-3, calculated value is multiplying by 1st element of skipped row and column.
  c=[
      [m[1][0], m[1][1], m[1][3]],
      [m[2][0], m[2][1], m[2][3]],
      [m[3][0], m[3][1], m[3][3]]
    ]
    c = (m[0][2]*(*three_by_three*(c)))

- Step 1
  Get remaining matrix from parent matrix.

$$C = \begin{bmatrix} -15 & -10 & 5 \\ 23 & 19 & 9 \\ -5 & 5 & -5 \end{bmatrix}$$

- Step 2
  Passing C as an argument to function
  three_by_three(C)

- Step 3
  Given function calculates the matrix shown in step 1.

- Step 4
  After performing Step-3, calculated value is multiplying by 1st element of skipped row and column.

  d=[
      [m[1][0], m[1][1], m[1][2]],
      [m[2][0], m[2][1], m[2][2]],
      [m[3][0], m[3][1], m[3][2]]
    ]
    d = (-m[0][3]) * (*three_by_three*(d))

- Step 1

$$D = \begin{bmatrix} -15 & -10 & 19 \\ 23 & 19 & -15 \\ -5 & 5 & 9 \end{bmatrix}$$

- Step 2
  Passing D as an argument to function

  three _ by _three (*D*)

- Step 3
  Given function calculates the matrix shown in Step 1.
- Step 4

After perform Step-3, calculated value is multiplying by 1st element of skipped row and column.

Sample 1

$$\begin{bmatrix} -15 & 10 \\ 23 & 19 \end{bmatrix}$$

Output: -515

Sample 2

$$\begin{bmatrix} -5 & 1 & 3 \\ 3 & 9 & -1 \\ 0 & 1 & -4 \end{bmatrix}$$

Output: -188

Sample 3

$$\begin{bmatrix} -2 & 1 & 3 & 2 \\ 3 & 9 & -1 & 1 \\ 0 & 1 & -4 & 2 \\ 3 & 1 & 0 & -1 \end{bmatrix}$$

Output: 220

Sample 4

$$\begin{bmatrix} -2 & 1 & 3 & 2 & 0 \\ 3 & 9 & -1 & 1 & 1 \\ 0 & 1 & -4 & 2 & -2 \\ 3 & 1 & 0 & -1 & 0 \\ 2 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Output: 394

Sample 5

$$\begin{bmatrix} -2 & 1 & 3 & 2 & 0 & 9.45 \\ 3 & 0 & -1.7 & -1 & 1 & 0 \\ 1 & -1.2 & -4 & 2 & -2 & 1 \\ 3 & 1.1 & 2 & -1 & 1 & 0 \\ 2.2 & 0 & 0 & 0 & 1 & -5.5 \\ 1.8 & 0 & 0.5 & 1 & 0 & 1 \end{bmatrix}$$
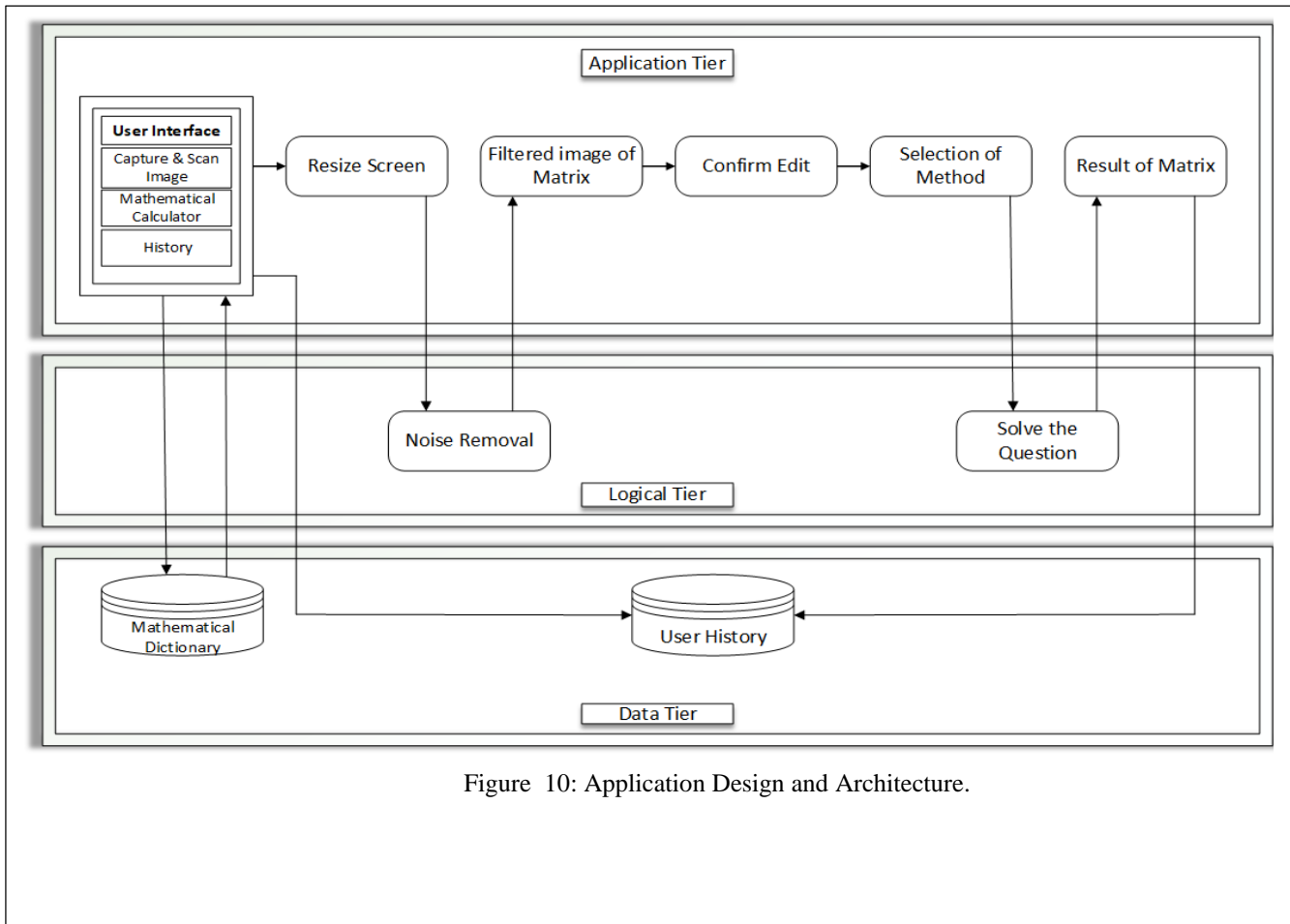
Output: 194.469

214

Figure 10: Application Design and Architecture.

Sample 6

$$\begin{bmatrix} 2 & 1/9 & -1 & 0 & -1 & 0 & 0 \\ 1 & 1.8 & 2/2 & 1.1 & 0 & -1 & 1 \\ 0 & 1 & 1.1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 3.1 & 0 & 2 \\ 3 & 1 & 9.1 & 3.2 & 0 & 0 & 0 \\ 1 & 9 & 1 & 1 & 1 & 1.6 & 0 \\ 0 & 1 & 1 & 7 & 2/8 & 1 & 1 \end{bmatrix}$$

Output: 23575.5735

Sample 7

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 2 & 0 \\ 1 & 1 & 1 & 1 & 1 & 8 & 1 & 1 \\ 0 & 17/10 & 1 & 0 & 2 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 7 & 1 & 1 & 21/10 & 3 & 1 \\ 9 & 1 & 4 & 0 & 2 & 1 & 0 & 1 \\ 9 & 1 & 3 & 1 & 2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 4 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Output: -203.52

215

Given sample are just an example to show the output of matrices according to matrix dimensions, if a user wants to see the step by step solution, print() function will be called at each step of code. If a matrix includes any float number like 2.2, these types of matrices can also be solved with same algorithm.

V Conclusion

In this paper we have proposed an android application which solves an image based mathematical matric problem, applying OCR technology and put solution in a step by step manner to dictate user in a convenient way. It is concluded that by using this application named "MatMath" students

with any level of mathematical knowledge would get encouragement to learn matrices. Keeping in view the upcoming trends in online math learning applications and learning courses, our work fits perfectly in this trend. It shows promising educational usage and impact in the field of mathematics. As the intended application is under development phase, for our further research by evaluating innovative methods to compute determinants functionality, solve a matrix which have trigonometric function by applying OCR and accuracy of our android application would be enhanced. In this way probability of errors could be diminished. As a result, application will provide more precise and variety of solutions to one specific problem to be solved.

References

[1] S. K. Vasudevan, S. Abhishek, V. Kumar, T. Aswin, and P. R. Nair, "An innovative application for code generation of mathematical equations and problem solving," *Journal of Intelligent & Fuzzy Systems*, no. Preprint, pp. 1–10, 2019.

[2] J. DUTTA and S. Pal, "Generalization of a new technique for finding the determinant of matrices," *Journal of Computer and Mathematical Sciences Vol*, vol. 2, no. 2, pp. 170–398, 2011.

[3] S. Mori, H. Nishida, and H. Yamada, *Optical character recognition*. John Wiley & Sons, Inc., 1999.

[4] J. Memon, M. Sami, R. A. Khan, and M. Uddin, "Handwritten optical character recognition (ocr): a comprehensive systematic literature review (slr)," *IEEE Access*, vol. 8, pp. 142 642–142 668, 2020.

[5] K. Fabian and K. J. Topping, "Putting "mobile" into mathematics: Results of a randomised controlled trial," *Contemporary Educational Psychology*, p. 101783, 2019.

[6] D. Frohberg, C. Göth, and G. Schwabe, "Mobile learning projects–a critical analysis of the state of the art," *Journal of computer assisted learning*, vol. 25, no. 4, pp. 307–331, 2009.

[7] A. Drigas and M. Pappas, "A review of mobile learning applications for mathematics," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 9, no. 3, pp. 18–23, 2015.

[8] V. Kalloo and P. Mohan, "Mobilemath: An innovative solution to the problem of poor mathematics performance in the caribbean," *The Caribbean Teaching Scholar*, vol. 2, no. 1, 2012.

[9] D. Majerek, "Application of geogebra for teaching mathematics," *Ad-vances in Science and Technology Research Journal*, vol. 8, no. 24, pp. 51–54, 2014.

[10] "Geogebragraphingcalculator (5.0.564.0)," 2015. [Online]. Available:https://play.google.com/store/apps/details?id=org.geogebra.android

[11] A. Chavan and A. Naik, "Linear equation solver in android using ocr,"*vol*, vol. 3, pp. 42–44, 2013.

[12] S. M. Cheema, N. Sarwar, and F. Yousaf, "Contrastive analysis of bubble & merge sort proposing hybrid approach," in *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*. IEEE, 2016, pp. 371–375.

[13] A. Ali, M. Shakil, H. Rafique, and S. M. Cheema, "Connection time estimation between nodes in vdtn," *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, vol. 10, no. 1, pp. 339–345, 2019.

[14] G. Macloskie, "A method of solving determinants," *The Annals of Mathematics*, vol. 1, no. 1/4, pp. 74–76, 1899.

[15] I. Arroyo, M. Micciollo, J. Casano, E. Ottmar, T. Hulse, and M. M. Rodrigo, "Wearable learning: multiplayer embodied games for math," in *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 2017, pp. 205–216.

[16] A. Sikka and B. Wu, "Camera based equation solver for android devices," *EE368*, 2012.

[17] S. J. Jang, "Ocr related technology trends," *European Journal of Engineeringand Technology Vol*, vol. 8, no. 1, 2020.

[18] A. Mathur, A. Pathare, P. Sharma, and S. Oak, "Ai based reading system for blind using ocr," in *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*. IEEE, 2019, pp. 39–42.

[19] A. Salihu, "A method to compute the determinant of square matrices of order five and six," *Open Journal of Mathematical Science*, vol. 3, pp. 84–93, 2019.

[20] L. Korenova and I. Veress-Bágyi, "The usage of mobile devices in the students' mathematics learning," in *17th Conference on Applied Mathematics, APLIMAT*, 2018, pp. 606–620.

[21] F. F. Abeles, "Dodgson condensation: the historical and mathematical development of an experimental method," *Linear Algebra and its applications*, vol. 429, no. 2-3, pp. 429–438, 2008.

[22] D. Doz and E. Doz, "Photomath inc., photomath, 2018 (versione 5.0.3)."

[23] N. M. Siew, J. Geofrey, and B. N. Lee, "Students' algebraic thinking and attitudes towards algebra: the effects of game-based learning using dragonbox 12+ app," *The Research Journal of Mathematics and Technology*, vol. 5, no. 1, pp. 66–79, 2016.

[24] "Matrix operations calculator(3.0)," 2013. [Online]. Available: https:

[25] "Determinant(2.1),"2016.[Online].Available:https://play.google.com/ store/apps/details?id=net.andromo.dev6860.app138441

[26] "Photomath (6.0.3)," 2015. [Online]. Available: https://play.google.com/store/apps/details?id=com.microblink.photomath

[27] S. Greenberg, M. Aladjem, and D. Kogan, "Fingerprint image enhance-ment using filtering techniques," *Real-Time Imaging*, vol. 8, no. 3, pp. 227–236, 2002.

[28] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic imaging*, vol. 13, no. 1, pp. 146–166, 2004.

[29] A. Rice and E. Torrence, "Condensation method for evaluating determinants."[Online].Available:https://www.maa.org/sites/default/files/pdf/Mathhorizons/pdfs/nov 2006 pp.12-15.pdf