

ABSTRACT

Mathematical Equations are frequently entered by hand on computer which is slower than writing them on paper or writing software. Researchers usually go through an arduous process of learning the nuances and syntactic complexities of a mathematical tool in order to solve or process mathematical equations. Through this project we aim to develop a user friendly website that captures the image of mathematical equation, recognize the equation and present the user with required solution. We aim to make the whole experience of experimenting with equations very user friendly and to remove the hassle of learning a mathematical tool just for mathematical experimentation. Since mathematics itself is a very wide field, digitizing and evaluating all of the mathematical symbols becomes a very complex and tedious task. Therefore, only a subset of these mathematical symbols is considered in this paper which are digits (0-9), arithmetic operators (+, -, *, \div), characters(y). This Project will be entirely dedicated to the digitization and evaluation of handwritten mathematical expressions.

INDEX

CONTENTS	PAGE NUMBER
ABSTRCT	I
LIST OF FIGURES	IV
LIST OF TABLES	V
CHAPTER 1 INTRODUCTION	1
1.1 Motivation of the Work	2
1.2 Aim	2
1.3 Objective	2
CHAPTER 2 LITERATURE SURVEY	3
CHAPTER 3 SYSTEM ANALYSIS	6
3.1 System Study	6
3.2 Existing System	6
3.2.1 Disadvantages	7
3.3 Proposed System	7
3.3.1 Convolutional neural network	8
3.4 Software Environment	9
3.4.1 Functional Requirements	10
3.4.2 Non-Functional Requirements	10
3.5 Input Design	10
3.6 Output Design	11
3.7 System Requirements	11
3.7.1 Software Requirements	11
3.7.2 Hardware Requirements	11
CHAPTER 4 SYSTEM DESIGN	12
4.1 Architecture Diagram	12
4.2 Data Flow Diagrams	13
4.2.1 DFD Level-0	14

4.3 UML Diagrams	14
4.3.1 Use Case Diagram	14
4.3.2 Activity Diagram	15
4.3.3 Sequence Diagram	16
4.3.4 Class Diagram	17
CHAPTER 5 SYSTEM IMPLEMENTATION	18
5.1 Handwritten Mathematical Equation Solver	18
5.1.1 Data Collection	18
5.1.2 Pre-processing	18
5.1.3 Segmenting Pre-Processed Images	19
5.1.3.1 Contour based segmentation	19
5.1.3.2 Bounding Box	19
5.1.3.3 Resizing each contour	20
5.1.4 Building Model	21
5.2 Expression Evaluation	22
5.3 Web Application	23
5.4 Sample Code	23
CHAPTER 6 SYSTEM TESTING	29
6.1 Unit Testing	29
6.1.1 Test strategy and approach	29
6.1.2 Features to be tested	29
6.2 Integrate Testing	31
6.3 Acceptance Testing	33
CHAPTER 7 RESULTS AND OUTPUTS	35
7.1 Results	35
7.2 Output	35
CHAPTER 8 CONCLUSION	38
8.1 Limitations	38
CHAPTER 9 FUTURE SCOPE	39
CHAPTER 10 BIBILOGRAPHY	40

LIST OF FIGURES

DESCRIPTION	PAGE NUMBERS
Figure 3.3 : Block Diagram	7
Figure 3.3.1.1 : convolutional neural network	8
Figure 4.1.1 : Architecture Diagram	12
Figure 4.2.1 : DFD level 0 Diagram	14
Figure 4.3.1.1 : Use Case Diagram	15
Figure 4.3.2.1 : Activity Diagram	16
Figure 4.3.3.1 : Sequence Diagram	16
Figure 4.3.4.1 : Class Diagram	17
Figure 5.1.2.1 : Handwritten Image	18
Figure 5.1.2.2 : Threshold Image	19
Figure 5.1.2.3 : Binarized Image	19
Figure 5.1.3.2.1: Bounding box around each contour	20
Figure 5.1.3.3.1: Before Resize	21
Figure 5.1.3.3.2: After Resize	21
Figure 5.1.4.1 : Architecture of CNN	22
Figure 7.2.1 : Home Page	35
Figure 7.2.2 : Before Uploading	36
Figure 7.2.3 : After Uploading	36
Figure 7.2.4 : Before Drawing	37
Figure 7.2.5 : After Drawing	37

LIST OF TABLES

DESCRIPTION	PAGE NUMBERS
Table 6.1.2.1 : Unit Testing test cases	31
Table 6.2.1 : Integration Testing test cases	33
Table 6.3.1 : Acceptance Testing test cases	34