



FEU INSTITUTE OF TECHNOLOGY

**Microprocessor-Based Interactive Mathematics
Learning Tool using Real-Time Computer-Vision**

By

GALVEZ, JOHN PATRICK B.

GUERRERO, JOHN CHRISTIAN S.

HERNANI, JON MICHAEL B.

Submitted in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Engineering

At

FEU Institute of Technology

February 26, 2022

Engr. Jamie Adrian D. Jamasali

Thesis Adviser

©2022 J. Galvez, J. Guerrero, and J. Hernani.

All Rights Reserved



The authors grant FEU Institute of Technology permission to reproduce and distribute the contents of this document in whole or in part.



APPROVAL AND ACCEPTANCE SHEET

The thesis entitled “**Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision**” prepared and submitted by:

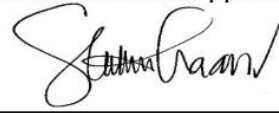
Galvez, John Patrick B.


Guerrero, John Christian S.

Hernani, Jon Michael B.

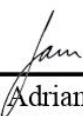
In partial fulfillment of the course requirement for the Degree of Bachelor of Science in Computer Engineering has been examined and is hereby recommended for approval.



Engr. Moises F. Jardiniano
Panelist 1

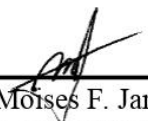

Engr. Stephanie Navarro
Panelist 2


Engr. Ricky Sandil
Head Panelist

Accepted as partial fulfillment of the requirements for the **Degree of Bachelor of Science in Computer Engineering.**


Engr. Jamie Adrian D. Jamasali
Thesis Adviser


Engr. Ethel Manansala
Course Adviser


Engr. Moises F. Jardiniano
Department Head

March 28, 2022
Date



ACKNOWLEDGEMENT

The researchers would like to extend their deepest gratitude to the following people for their help and support during the development of this study.

To our thesis adviser, Engr. Jamie Adrian Jamasali, for guiding us throughout the innovation of this study. Without his advice, wisdom, and encouragement, we won't be able to fulfill this project.

To our director, Engr. Moises Jardiniano, for inspiring not just us, but the whole computer engineering department and computer engineering students.

To our course adviser, Engr. Estrelita Manansala, for supervising the course, and guiding us in fulfilling the requirements.

To our teacher, Ma'am Jocelyn Marie Gahol, for giving us advice regarding the preferences of the grade school students when it comes to learning basic mathematics.

To our friends and family, for providing emotional, financial, and moral support throughout the whole semester.

And finally, to our Almighty God, for giving us the spiritual and emotional guidance throughout the study, for providing us the wisdom and knowledge within the day-to-day task up until the completion of the project.



TABLE OF CONTENTS

APPROVAL AND ACCEPTANCE SHEET	ii
ACKNOWLEDGEMENT	iii
TABLE OF CONTENTS	iv
LIST OF TABLES	viii
LIST OF FIGURES	viii
ABSTRACT	x
Chapter 1	1
INTRODUCTION	1
1.1 Background of the Study	2
1.2 Significance of the Study	3
1.3 Objectives	4
1.3.1 General Objective	4
1.3.2 Specific Objectives	4
1.4 Scope and Delimitations	5
1.5 Conceptual Framework	6



1.6	Definition of Terms	7
Chapter 2		8
REVIEW OF RELATED LITERATURE		8
2.1	Foreign Related Literature	8
2.1.1	Building Arduino-based Tangible Serious Games for Elementary Mathematics and Physics	8
2.1.2	Improving the Working Memory During Early Childhood Education Through the Use of an Interactive Gesture Game-Based Learning Approach	8
2.1.3	Fingermath – Arithmetic Hand Gesture Game to Improve Early Childhood Mathematics Learning	9
2.1.4	Object Detection, Tracking, and Recognition for Multiple Smart Cameras	9
2.1.5	Multiple object detection using OpenCV on an embedded platform	9
2.2	Local Related Literature	10
2.2.1	Microcontroller-Based Interactive Clock (iClock)	10
2.2.2	Microcontroller-Based Interactive Spelling Toy (iSpell)	10
2.2.3	Microcontroller-Based Interactive Math Toy (iMath)	10



2.2.4	Recognition of Baybayin Symbols (Ancient Pre-Colonial Philippine Writing System) using Image Processing	11
2.2.5	Proposed Image Pre-processing Techniques for Face Recognition Using OpenCV	11
2.3	Synthesis	12
2.4	Research Gap	13
Chapter 3		15
METHODOLOGY		15
3.1	Software Development Life Cycle	15
3.2	Block Diagram	16
3.3	System Flowchart	17
3.4	Proposed System	18
	Software System	18
3.4.1	Processing Unit	18
3.4.2	Camera	18
3.4.3	Push Button	18
3.4.4	TFT Liquid Crystal Display	18



3.4.5	Speaker	18
	Hardware System	19
3.4.6	Raspberry Pi	19
3.4.7	Camera	19
3.4.8	Push Button	19
3.4.9	TFT Liquid Crystal Display	19
3.4.10	Speaker	20
3.5	Testing Procedure	20
3.5.1	Displaying Basic Arithmetic Test	20
3.5.2	Detecting Hand Gestures Numbers Test	21
3.5.3	Reading Premade Numbers Test	22
3.5.4	Display Testing For Hand Gestures	24
3.5.5	Display Testing For Premade Numbers	25
3.5.6	Teaching and Correcting the Given Wrong Answer	27
3.5.7	Obtaining 90% success rate of all test	28
3.6	Design Constraints	29
3.6.1	Cost	29



3.6.2	Time	29
Chapter 4		30
RESULTS AND DISCUSSION		30
4.1	Displaying Basic Arithmetic	30
4.2	Detecting Hand Gestures Numbers Test	31
4.3	Reading Premade Numbers	32
4.4	Display Testing for Hand	34
4.5	Display Testing for Premade	35
4.6	Teaching and Correcting the Given Wrong Answer	36
4.7	Obtaining at least 90% success rate of all tests	38
Chapter 5		40
CONCLUSION		40
Chapter 6		41
RECOMMENDATION		41
References		42
APPENDIX		44
APPENDIX A	Curriculum Vitae of Ma'am Jocelyn Gahol	45



APPENDIX B Transcript of Interview with Ma'am Jocelyn Marie Gahol	49
APPENDIX C Flowchart	54
APPENDIX D Schematic Diagram	57
APPENDIX E Data Sheet	59
APPENDIX F Pictures	64
APPENDIX G User's Manual	75
APPENDIX H Program Code	79
APPENDIX I Proponents' Profile	136
APPENDIX J Grammarian's Profile	140
APPENDIX K Grammarian's Certificate	145

LIST OF TABLES

Table 0 — Research Gap for Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision and other similar projects	13
Table 1 — Results for Displaying Basic Arithmetic	30
Table 2 — Results for Detecting Hand Gestures Number Test	31
Table 3 — Results for Reading Premade Numbers	32
Table 4 — Results for Displaying Testing for Hand	35
Table 5 — Results for Display Testing For Premade	36



Table 6 — Results for Teaching and Correcting the Given Wrong Answer (PASSED)	37
Table 7 — Results for Teaching and Correcting the Given Wrong Answer (FAILED)	37
Table 8 — Results of Obtaining at least 90% success rate of all test	39

LIST OF FIGURES

Figure 1 — Conceptual Framework	6
Figure 2 — Block Diagram	16
Figure 3 — Flowchart	17
Figure 4 — Hand Landmarks	31
Figure 5 — Premade Testing Accuracy	33
Figure 6 — Premade Testing Loss	34
Figure 7 — Interview via Google Meet with Ma'am Jocelyn Marie Gahol	53
Figure 8 — System Flowchart	55
Figure 9 — Program Flowchart	56
Figure 10 — Schematic Diagram	58
Figure 11 — Raspberry Pi 4B	60
Figure 12 — LCD Screen 7inch	62
Figure 13 — Webcam	63
Figure 14 — Prototype	65
Figure 15 — G.U.I (Homepage)	65
Figure 16 — G.U.I (Teacher's Interface)	65



Figure 17 — G.U.I (Student’s Interface)	66
Figure 18 — Testing of the Hand tracking	66
Figure 19 — Testing of the Premade Numbers	67
Figure 20 — One of the proponents with the Grade 1 student	67
Figure 21 — Two of the proponents with the Grade 1 student and Grade 3 Student	68
Figure 22 — Testing of the prototype with the Grade 1 student	68
Figure 23 — Testing of the prototype with the Grade 1 student (2)	69
Figure 24 — Testing of the prototype with the Grade 1 student (3) (Hand tracking)	69
Figure 25 — Testing of the prototype with the Grade 3 student (Premade Numbers)	70
Figure 26 – Testing different backgrounds for answering Premade Numbers	70
Figure 27 – Testing different backgrounds for answering Premade Numbers	71
Figure 28 – Testing different backgrounds for answering Premade Numbers	71
Figure 29 – Testing different backgrounds for answering Premade Numbers	72
Figure 30 – Invalid Gesture (Thumbs-Up)	72
Figure 31 – Invalid Gesture (Rock On)	73
Figure 32 – Invalid Gesture (Call me)	73
Figure 33 – Invalid Gesture (Odd-Finger Up)	74
Figure 34 – Invalid Gesture (Isko-Ong Tandem)	74



ABSTRACT

When it comes to learning, there are beneficial effects of incorporating gestures in learning mathematics especially within the early ages of life. At first, we learn Math by counting and incorporating our fingers as a guide, as well as writing numbers along on a piece of paper. These gestures are known to be examples of hands-on learning experience for children. Aside from counting, they learn basic mathematical operations as well.

Teaching children mathematics is not an easy task. The child's attentiveness plays a huge factor in the learning process. The average attention span of the child from 2 to 10 years of age is 20 to 30 minutes only. Making the learning interactive can help in maximizing the learning experience of the child within the timespan. Thus, we aim to develop a compact device that is interactive with children when it comes to learning basic mathematical operations such as addition, subtraction, and multiplication.

This device has software that uses computer-vision and image detection as a way of teaching children math in an interactive way. The child has two different options of answering, by using hand gestures or premade printed numbers. The child is asked five random basic mathematical questions within a time limit. It has three levels of difficulty which the child must pass in order to progress each level. Furthermore, the software also has audio feedback as a way of letting the child know the correct answer. With these features, children are engaged in an interactive learning opportunity in mathematics.

Keywords: Interactive, Computer-vision, Image detection



Chapter 1

INTRODUCTION

Education is of utmost importance to every person in the world and also deserves to attain in their lifetime. Different teaching techniques are involved such as interactive or participative methods in order to have hands-on experience thus helping the learners understand better. There are a lot of ways to consider when teaching, such as giving examples, experimenting, demonstrating, etc. Tools are also used as a medium to aid in teaching such as PowerPoint presentations, supplementary materials, and video tutorials.

The researchers are interested in exploring the possibilities of incorporating the use of sensors and microprocessors as a medium for teaching basic mathematics to children. As microcontrollers are programmed with specific instructions, it is beneficial for these children to have a program that has the features to give exercises in basic Mathematics. Electronic hardware technologies such as microprocessors and other related sensors will significantly add a more interactive learning experience by having these tangible devices that have the capability to interact with the child. In return, the educator, e.g., teacher or a parent, and the learner will have a more pleasant experience learning the topics of basic mathematics.

The device will have two features of learning: mathematics. The device will be having interactive real-time hand tracking using OpenCV, and this will be the key tool for the child to interact with the device. Furthermore, the hand tracking will then communicate to the microprocessor, thus executing the instructions programmed to the microprocessor.



1.1 Background of the Study

Children of all periods within preschool and teenage times are getting progressed throughout their times. At this age, children are more likely to engage in different activities and challenges which results in learning new activities. Hands-on literacy allows someone to develop intelligence as they use their vision and other senses similar to touching, pulling, and playing. This type of literacy may apply to commodities as simple as introductory calculation.

According to a study in Goodwin University, “Learning by doing” is better for scholars because they aren't just physically engaged but also mentally which results in a more focused state of the pupil. Between the periods of five to seven, the right side of the child’s brain is developing, and this is the part of the brain where the brain has the functions of spatial and visual conditioning, therefore making it important to expose children within this age to interactive literacy (Middendorf, 2020). Wherein the left side of the brain is where Mathematics, direct thinking, and critical thinking processes. Being suitable to learn using a material that interacts through harkening, touching, reading, or speaking will stimulate and develop the brain making it salutary for the child.

According to a Michigan University research, children aged 2 to 10 had an average attention span of 4 to 6 minutes, ranging from 20 to 30 minutes. It's worth mentioning that some developmental experts determined that a 2-year-old could focus on a task for up to 10 minutes for every five minutes each year of his or her age. Other influences include their level of interest in the activity, adjacent distractions, and a less engaging experience.



1.2 Significance of the Study

The research entitled “Impacts of Technology Used on Children: Exploring Literature on the Brain, Cognition and Well-Being” stated that “children in the 21st century are avid users of technology - more so than generations past.” The more technology moves forward, the younger people are going to be exposed to it as time passes by. Technology can be a double-edged sword for the younger generations. It has its benefits and drawbacks. An example of a drawback is that children tend to focus their time playing games or watching entertainment videos. In terms of benefits, technology is a tool for children when it comes to learning. Thus, making this Interactive Mathematical Learning Tool for Children comes in handy. The groups that will be affected by this are:

Parents

This will help parents to aid them and make it easier for them to get the attention of their kid/s and teach them basic mathematics. It is designed as “entertaining while learning” because children like to have fun.

Elementary Teacher

This will be a help for the teachers in lower grades in two ways. First, this can be helpful for them if the children they are teaching are already exposed to the device, meaning their students have already learned mathematics. Thus, teaching them mathematics will be easy. Second, they can use this device to teach mathematics to their students who were not exposed to the device.



Children

As it is a device designed to be interactive with the children. The device will improve the analytic thinking of the child, attention span, and response. This will also challenge the child for future academic-related topics.

1.3 Objectives

1.3.1 General Objective

To develop a Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision.

1.3.2 Specific Objectives

- To display the basic arithmetic problem
- To detect specific hand gestures
- To detect premade numbers
- To display the numbers from the premade numbers
- To display the numbers from the specific hand gestures
- To produce audio feedbacks
- To obtain at least 90% success rate of the system



1.4 Scope and Delimitations

The learning tool is a box that can be carried around by children and used to perform basic arithmetic problems for practice. The learning toolbox is most suitable for children at grades 1 - 3. It can generate basic math operation problems.

There are three (3) levels of difficulty known as Levels 1 - 3. Basic arithmetic problems on elementary level will be based on the DepED curriculum. The device will only display five (5) problems randomly according to the level. It can only provide 1-digit answers up to only 2-digit answers. The device will produce the problems according to the progress of the child on the level of difficulties. For the child to continue to the next level, the child needs to score at least 60% of the total score which is 3 out of 5. There will be two-mode in answering, hand gesture mode and premade numbers mode. When using hand gestures, the camera can detect two (2) hands. When using the premade numbers, it will be placed in a container that will be faced to the camera for it to detect the given answer. The device will show the correct solution and answer on the LCD after answering each problem when the child answered it incorrectly. The system can also detect five (5) invalid gestures. It will inform the user at the end of the question when the user did an invalid gesture. Invalid gestures will not affect the score of the user.

Privilege will be given to a teacher or educator to change the level of difficulty according to the capability of the user to answer the arithmetic problems. A password will be prompted to change the level of difficulty. A 7-inch with a resolution of 1024x600 TFT LCD will display the basic arithmetic problems, the answer of the user, and the captured video in real time. It will also display the detected number using a hand gesture or premade number.

1.5 Conceptual Framework

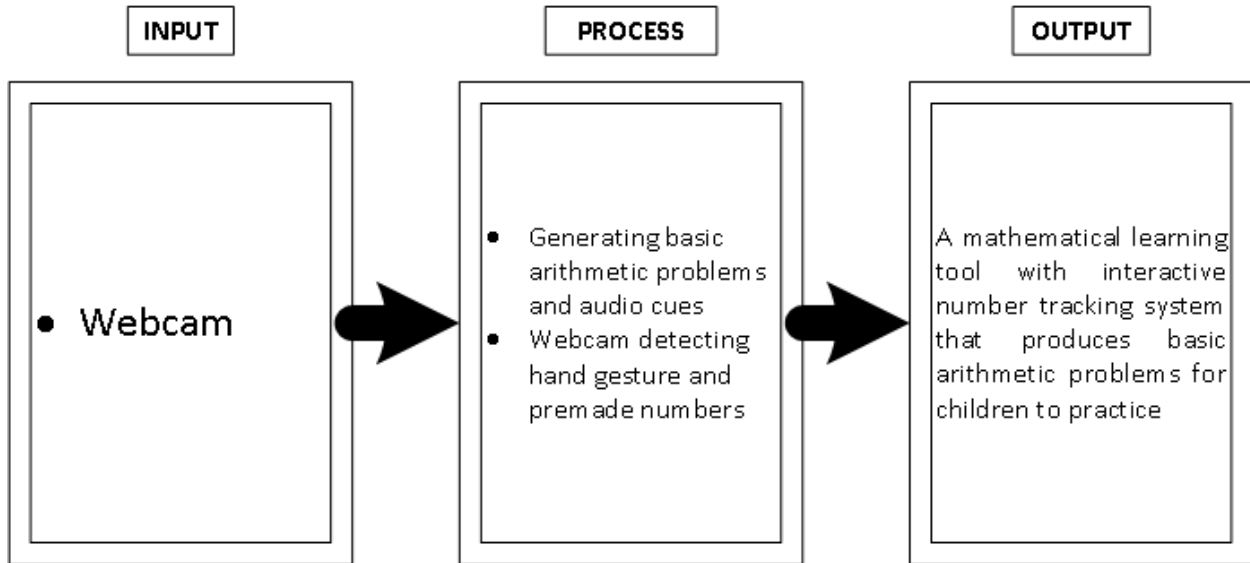


Figure 1 — Conceptual Framework

One of the ways to gain knowledge is through learning. Learning is achievable when someone is teaching others, for example, a student-teacher relationship. With this project, a child student can still learn elementary mathematical problems even without the presence of his educator. Children can answer the basic arithmetic problems by doing specific hand gestures in front of the camera. Detected written numbers or premade numbers by the camera will be displayed in the LCD and once it is correct, the device will automatically proceed into the next arithmetic problem. A learning tool box will be provided which produces basic arithmetic problems for children to practice.



1.6 Definition of Terms

The following terms are used and will be used throughout the proposed study:

Interactive – Involving actions with the input of the user.

Computer-vision - multidisciplinary scientific subject concerned with how computers can extract high-level information from digitized photos or films.

Image Processing - the use of a digital computer to run an algorithm on digital photographs.

Addition - the process of combining matrices, vectors, or other quantities according to the rules in order to obtain the overall sum.

Arithmetic - the field of mathematics concerned with number characteristics and manipulation.

Division - the process of dividing a matrix, vector, or other quantity by another according to set rules in order to get a quotient.

Microprocessor - an integrated circuit that contains all of the functionality of a computer's central processor unit.

Multiplication - the process of combining matrices, vectors, or other quantities according to defined rules in order to generate their product.

Subtraction - the process of subtracting one matrix, vector, or other quantity from another according to particular rules in order to derive the difference.

Tangible - Perceptible by the sense of touch.



Chapter 2

REVIEW OF RELATED LITERATURE

2.1 Foreign Related Literature

2.1.1 Building Arduino-based Tangible Serious Games for Elementary

Mathematics and Physics

The goal of this research is to create a physical gadget that can be used to incorporate fundamental mathematical principles into music and melody. The study focused on two basic concepts. The first one is a game that involves music in which different sounds are controlled by adjusting the distance between the hand and the ultrasonic sensor, similar to a Theremin. Different fundamental tones, such as an octave 7 main note or "Do Re Mi," will be produced as the hand distance from the sensor changes. The second study will look at the notion of alignment and distance. [3]

2.1.2 Improving the Working Memory During Early Childhood Education

Through the Use of an Interactive Gesture Game-Based Learning Approach

The benefits and uses of human-computer interaction is to improve children's extracurricular activities and learning. The study looks at gesture interactive game-based learning (GIGL) and employs a hand gesture recognition system in whether these sorts of apps are effective at improving the knowledge and fundamental mathematical principles in young children aged 5 to 6. The youngsters will finish nine mini-games, all of which are controlled by hand gestures. A PC connected to a Kinect and a TV screen that shows the applications make up the hardware. The results suggest that employing these technologies improved the child's working memory and essential math skills. [4]



2.1.3 Fingermath – Arithmetic Hand Gesture Game to Improve Early Childhood

Mathematics Learning

Some technology in the classroom may enhance and support the child's choices and ability to understand math in classrooms, as well as influence early childhood attitudes toward being more analytical, artistic, and imaginative. One approach for learning to calculate on one's fingers is the Jarimagic Method, in which young children habitually communicate with their fingers in the classroom while studying mathematics. As a consequence, the study integrates learning methods such as interactive games with hand gestures utilizing the Jarimagic methodology and tools scratch 2.0. The system comprises a software that displays number symbols ranging from 0 to 9 utilizing hand gestures. The curriculum will begin with varying levels of difficulty for the youngster's overall experience. [5]

2.1.4 Object Detection, Tracking, and Recognition for Multiple Smart Cameras

In a wide range of real-world applications, computer visions using cameras are one of the most often utilized types of innovation. Surveillance, video conferencing, and streaming are just a few examples. Using algorithms, such a gadget is capable of detecting, tracking, and recognizing things. This research focuses on the many applications of visual sensor networks, with a particular emphasis on the effective utilization of geometric limitations imposed by imaging sensors to develop distributed algorithms for object identification, tracking, and recognition. [6]

2.1.5 Multiple object detection using OpenCV on an embedded platform

Object detection now has a wide range of applications. It has been fueled by the growing processing power of software and hardware. The OpenCV Libraries are used in this study to create



a multiple object detection application. The suggested application is concerned with the machine learning and image processing systems, and The findings show of which the cases of image tracked applications can be found. One of the item detections in this investigation was on a motherboard that used a mid-range camera to display the different sections of it and was provided to the computer to recognize and name the different components. [7]

2.2 Local Related Literature

2.2.1 Microcontroller-Based Interactive Clock (iClock)

This device teaches and aids toddlers in reading the clock. This study also has different functionalities other than the clock such as teaching time management, scheduling, and it can even formulate basic mathematical questions. It has different modes of learning i.e Lesson Mode, Quiz Mode, and Puzzle Mode. Additionally, the interactive clock also has different difficulties, Beginner, Interactive, Expert, and Freehand. [8]

2.2.2 Microcontroller-Based Interactive Spelling Toy (iSpell)

This device helps educate toddlers in terms of memory retention of alphabets. This study has the capability to perceive the three sensory perceptions of the user: vision, hearing, and touch. The device has an audio response, large alphabets, and buttons for different quiz learning tests. Additionally, it comes with a miniature on the middle, the speaker on the right, and an overall seven buttons for different learning modes and tests. [9]

2.2.3 Microcontroller-Based Interactive Math Toy (iMath)

This device is an educational toy designed to aid children 4-6 years of age in learning basic math by teaching them how to identify and pronounce numbers. This covers the numbers zero to



one hundred. It also has three different modes of learning; lesson mode which teaches numbers one to nine and place values from ones, tenths, and hundredths. The next is solve mode which teaches basic arithmetic such as addition and subtraction and finally, quiz mode in which the device evaluates the child through test assessments. There is also beginner and advanced difficulty. The device incorporates physical learning blocks that compose numerical values, an audio feedback speaker, and a mini LCD. These devices engage touch, sight, and hearing/speaking which is beneficial for the child as the study becomes more interactive. [10]

2.2.4 Recognition of Baybayin Symbols (Ancient Pre-Colonial Philippine Writing System) using Image Processing

The focus of this work is on image processing and Machine Learning in utilizing Optical Character Recognition (OCR). In order for the Baybayin symbols to be detected, the researchers developed a Feed Forward Neural Network with a Dropout Method and a Convolutional Neural Network with a Dropout Method. Feed Forward Neural Networks are made up of a dense input layer, four dense hidden layers, and one dense output layer, as well as a CNN structure with three convolutional layers, two hidden layers, and one output layer. [11]

2.2.5 Proposed Image Pre-processing Techniques for Face Recognition Using OpenCV

This research tackles the enhanced processing techniques such as illumination, pose and illumination+pose. These techniques will be combined with eigenface, fisherface and LBPH face recognition using the OpenCV. To successful result, the researchers identified performance combinations, relevant speed, identification rate and its threshold are measured. The study used two subjects, frontal face from the Yale database and 20 individual faces. The result is that there



is a statistical difference among the speed and threshold level of algorithms with the enhanced processing techniques. While there is no significant interaction effect of the identification rate.

[12]

2.3 Synthesis

The concept of a device that is tangible and incorporating it to mathematical concepts and music is an innovative way to have an interactive learning experience for the child. The use of proximity sensors for hand distance input produces different tones which makes it interactive and gives feedback from the user. This study incorporates the use of hand gestures for learning. The next study, Arithmetic Hand Gesture Game to improve early childhood mathematics learning, thoroughly introduces the use of hand gesture in terms of learning basic math. They made handlike cartoons that show hand-gestured numbers which will be the learning reference for the child.

According to the study of “Object Detection, Tracking, and Recognition for Multiple Smart Cameras”, video cameras is one of the commonly used sensors in a large number of real-life applications such as a security camera, video conferencing and even to detect, track and recognize objects with the use of algorithms. One of the most common computer software programs in object detection is the OpenCV Libraries. According to the study of “Multiple object detection using OpenCV on an embedded platform”, the researchers used OpenCV Libraries to detect the different parts of a computer motherboard and it even displayed the basic parts of it.

Most of the local studies are focused on interactive learning for the child. The Microcontroller-Based Interactive Clock or iClock is a device that teaches children to read and has the capability to formulate quizzes for improved interactive learning experience. This study is identical to both Microcontroller-Based Interactive Spelling Toy (iSpell) and Microcontroller-



Based Interactive Math Toy (iMath) wherein the iSpell helps educate toddlers in terms of memory retention of alphabets using large alphabets and buttons for the quizzes while the iMath focuses more on children 4-6 years of age in learning basic math by teaching them how to identify and pronounce numbers. The device incorporates physical learning blocks that compose numerical values, an audio feedback speaker, and a mini LCD for interactive learning and giving quizzes.

Overall, similar to our study, the researchers will incorporate a learning tool using an OpenCV, a video camera for learning basic mathematics using interactive hand-tracking and number detection systems.

2.4 Research Gap

Table 0 — Research Gap for Microprocessor-Based Interactive Mathematics Learning

Tool using Real-Time Computer-Vision and other similar projects

Functions	Fingermath – Arithmetic Hand Gesture Game to Improve Early Childhood Mathematics Learning	Building Arduino-based Tangible Serious Games for Elementary Mathematics and Physics	Microcontroller-Based Interactive Spelling Toy (iSpell)	Microcontroller-Based Interactive Math Toy (iMath)
Hand Tracking Recognition	✓	✓	✗	✗
Object Detection	✓	✓	✗	✗



Audio Response	✓	✓	✓	✓
Interactive Learning	✓	✓	✓	✓
Mistake correction response	✗	✗	✓	✓
Has different levels of difficulties	✓	✗	✓	✓
Compact	✓	✓	✓	✓
Lightweight	✓	✓	✗	✓
Math based learning	✓	✓	✗	✓



Chapter 3

METHODOLOGY

3.1 Software Development Life Cycle

The Waterfall model is used as the Software Development Life Cycle (SDLC) and it is also referred to as a linear-sequential life cycle model. It is a model that finishes one phase after another. Overlapping phases will not be done in this model. This model has several phases which are planning, designing, implementation, testing, deployment, and maintenance phase.

- **Planning** - Researchers gather the data on how the learning box will function and the items needed which are camera, LCD screen, and a Raspberry Pi 3.
- **Designing** - Researchers determine the program language needed and assemble the materials to make the learning box function as expected.
- **Implementation** - Researchers create the program code needed for the hardware materials to function.
- **Testing** - Researchers begin to run the program code and observe how the hardware materials will function. If it does not function as expected, program code will be redesigned.
- **Deployment** - Researchers distribute the learning box to customers for the elementary level children to use.
- **Maintenance** - Researchers troubleshoot the learning box when there will be a problem with the device.



3.2 Block Diagram

The system will be using the cameras and the push buttons for the input in the microcontroller. The camera will handle the answers provided by the user and the push buttons will be the indicator of the difficulty. In the output, LCD and Speakers are the ones that will be doing the external part or output part. LCD will display the arithmetic question according to the difficulty that is chosen. In addition to that, the LCD will also display the input that the user gave to the system. Speakers will be the one handling the audio cues and the text to speech that will teach the user where their mistakes are and how they will correct it.

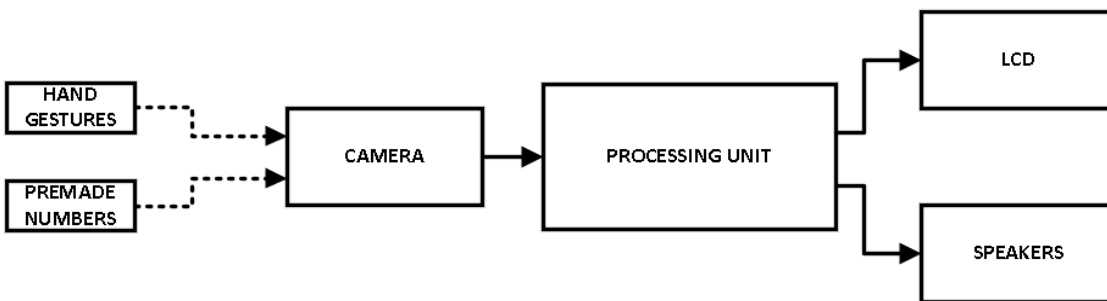


Figure 2 — Block Diagram



3.3 System Flowchart

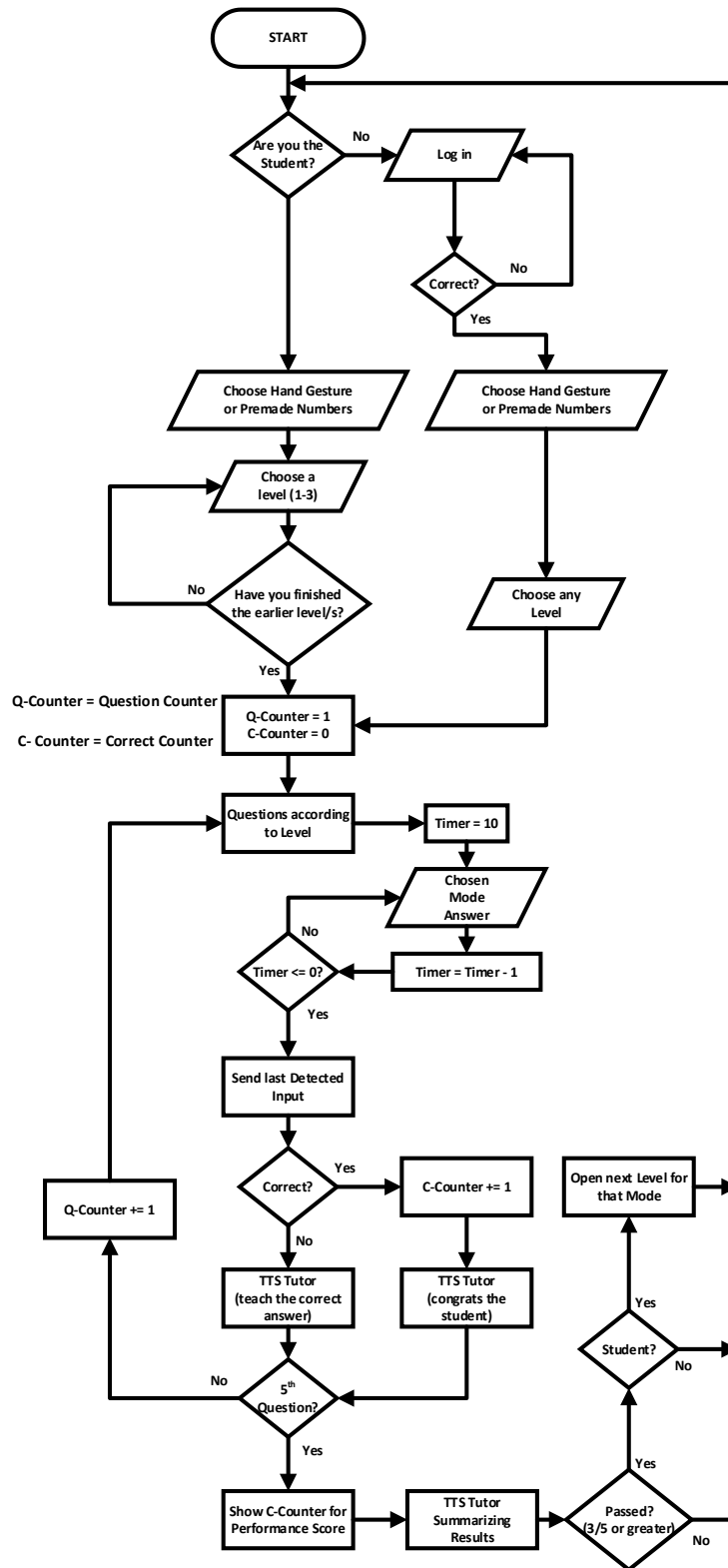


Figure 3 — Flowchart



3.4 Proposed System Software

For the software system of the device, the program that will detect the specific hand gestures and premade numbers is uploaded to the processing unit. Python is used as the programming language for the system to function.

3.4.1 Processing Unit

The processing unit sends instructions according to the program codes to the other materials to function.

3.4.2 Camera

The camera sends a signal to the processing unit when a specific hand gesture or a premade number has been captured.

3.4.3 Push Button

The program changes the level of difficulty of the problem when the user has pressed the button.

3.4.4 TFT Liquid Crystal Display

The program codes will be sent to the TFT Liquid Crystal Display to display the basic arithmetic problems and the captured specific hand gestures and premade numbers.

3.4.5 Speaker

The processing unit will then send instructions to the speaker to output audio cues when the basic arithmetic problem has been answered.



Hardware System

In the hardware system of the device, the Raspberry pi will be the main part of the hardware system. It will provide the other hardware materials instruction on what to do. A camera is installed to capture the specific hand gestures and premade numbers. A LCD screen is also installed to display the basic arithmetic problems and the detected specific hand gestures and premade numbers. Speaker will be used to deliver audio feedback.

3.4.6 Raspberry Pi

Raspberry Pi is the brain of the system. The system will function according to the instructions and program codes that will be uploaded to it.

3.4.7 Camera

The camera will capture the movement of the hand and premade numbers. Specific hand gestures and premade numbers are detected by the camera.

3.4.8 Push Button

The push button will be used to change among the levels of difficulty. Lower levels of difficulty can be skipped to proceed to the next level when the user is done in the lower level.

3.4.9 TFT Liquid Crystal Display

The LCD will display the basic arithmetic problems according to the levels of difficulty. It will also display the number, which can be done by doing specific hand gestures and premade numbers, then will be captured by the camera.



3.4.10 Speaker

The speaker will output the audio response whenever the user answers correctly the basic arithmetic problems or not.

3.5 Testing Procedure

3.5.1 Displaying Basic Arithmetic

Test Objectives:

To display different arithmetic problems in the LCD

Materials:

- Processing Unit
- LCD
- Camera

Procedure:

1. Turn on the device.
2. Run the program
3. Display Random arithmetic for Level 1 ranging 1 - 9 as an answer
4. Check if there is a display and corresponds to the range.
5. If yes, put a correct mark in table 1. Otherwise, put a wrong mark.
6. Repeat procedure 3 to 5 for 10 times
7. Display Random arithmetic for Level 2 ranging 5 - 20 as an answer
8. Check if there is a display and corresponds to the range.
9. If yes, put a correct mark in table 1. Otherwise, put a wrong mark.



10. Repeat procedure 7 to 9 for 10 times
11. Display Random arithmetic for Level 3 ranging 15 - 50 as an answer
12. Check if there is a display and corresponds to the range.
13. If yes, put a correct mark in table 1. Otherwise, put a wrong mark.
14. Repeat procedure 11 to 13 for 10 times
15. Calculate the success rate of the trials using the given formula

$$\text{Success Rate} = \frac{\text{Successful Trials}}{\text{Total Trials}} \times 100$$

3.5.2 Detecting Hand Gestures Numbers Test Objectives:

Test Objectives:

To detect what specific hand gestures the user is giving

Materials:

- Camera
- Processing Unit
- Computer Unit

Procedure:

1. Turn on the device.
2. Run the program
3. Count 1 to 10 using the user's hand, give 3 seconds per number.
4. Check if the displayed number in the LCD is also counting 1 to 10
5. If yes, put a correct mark in Table 2. Otherwise, put a wrong mark.



6. Repeat procedure 3 to 5 for 10 times.
7. Calculate the success rate of the trials using the given formula

$$\text{Success Rate} = \frac{\text{Successful Trials}}{\text{Total Trials}} \times 100$$

3.5.3 Reading Premade Numbers

Test Objectives:

To detect what the user is showing using the premade numbers.

Materials:

- Camera
- Plastic Container
- Premade Numbers
- Processing Unit
- Computer Unit

Procedure:

1. Turn on the device.
2. Run the program
3. Let the user show premade number 0 in the camera.
4. Check if the system reads it as 0. Give 3 seconds for this trial.
5. If yes, put a correct mark in table 3. Otherwise, put a wrong mark.
6. Let the user show premade number 1 in the camera.
7. Check if the system reads it as 1. Give 3 seconds for this trial.
8. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.



9. Let the user show premade number 2 in the camera.
10. Check if the system reads it as 2. Give 3 seconds for this trial.
11. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
12. Let the user show premade number 3 in the camera.
13. Check if the system reads it as 3. Give 3 seconds for this trial.
14. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
15. Let the user show premade number 4 in the camera.
16. Check if the system reads it as 4. Give 3 seconds for this trial.
17. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
18. Let the user show premade number 5 in the camera.
19. Check if the system reads it as 5. Give 3 seconds for this trial.
20. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
21. Let the user show premade number 6 in the camera.
22. Check if the system reads it as 6. Give 3 seconds for this trial.
23. If yes, put a correct mark in table 3. Otherwise, put a wrong mark.
24. Let the user show premade number 7 in the camera.
25. Check if the system reads it as 7. Give 3 seconds for this trial.
26. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
27. Let the user show premade number 8 in the camera.
28. Check if the system reads it as 8. Give 3 seconds for this trial.
29. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
30. Let the user show premade number 9 in the camera.
31. Check if the system reads it as 9. Give 3 seconds for this trial.



32. If yes, put a correct mark in Table 3. Otherwise, put a wrong mark.
33. Repeat procedures 3 to 32 for 10 times.
34. Calculate the success rate of the trials using the given formula.

$$\text{Success Rate} = \frac{\text{Successful Trials}}{\text{Total Trials}} \times 100$$

3.5.4 Display Testing For Hand

Gestures Objectives:

To display the hand gesture given by the user.

Materials:

- Camera
- LCD
- Processing Unit
- Computer Unit

Procedures:

1. Turn on the device.
2. Run the program for hand tracking
3. Enter level 1.
4. Answer the given problem using hand gesture numbers.
5. Check if the given user's answer is what the system reads.
6. If correct, put a correct mark in table 4. Otherwise, put a wrong mark.
7. Repeat procedure 4 to 6 for 10 times.



8. Enter level 2.
9. Answer the given problem using hand gesture numbers.
10. Check if the given user's answer is what the system reads.
11. If correct, put a correct mark in table 4. Otherwise, put a wrong mark.
12. Repeat procedure 4 to 6 for 10 times.
13. Enter level 3.
14. Answer the given problem using hand gesture numbers.
15. Check if the given user's answer is what the system reads.
16. If correct, put a correct mark in table 4. Otherwise, put a wrong mark.
17. Repeat procedure 4 to 6 for 10 times.
18. Calculate the success rate of the trials using the given formula.

$$\text{Success Rate} = \frac{\text{Successful Trials}}{\text{Total Trials}} \times 100$$

3.5.5 Display Testing For Premade

Numbers Objectives:

To display the given premade numbers by the user.

Materials:

- Camera
- LCD
- Processing Unit
- Computer Unit
- Premade Number



Procedure:

1. Turn on the device.
2. Run the program for premade numbers.
3. Enter level 1.
4. Answer the given problem using premade numbers.
5. Check if the given user's answer is what the system reads.
6. If correct, put a correct mark in Table 5. Otherwise, put a wrong mark.
7. Repeat procedure 4 to 6 for 10 times.
8. Enter level 2.
9. Answer the given problem using premade numbers.
10. Check if the given user's answer is what the system reads.
11. If correct, put a correct mark in Table 5. Otherwise, put a wrong mark.
12. Repeat procedure 4 to 6 for 10 times.
13. Enter level 3.
14. Answer the given problem using premade numbers.
15. Check if the given user's answer is what the system reads.
16. If correct, put a correct mark in Table 5. Otherwise, put a wrong mark.
17. Repeat procedure 4 to 6 for 10 times.
18. Calculate the success rate of the trials using the given formula.

$$\text{Success Rate} = \frac{\text{Successful Trials}}{\text{Total Trials}} \times 100$$



3.5.6 Teaching and Correcting the Given Wrong Answer

Objectives:

To let the user know where he got wrong and teach the right answer.

Materials:

- Camera
- LCD
- Processing Unit
- Computer unit
- Premade Numbers
- Speaker

Procedure:

1. Turn on the device.
2. Run the program.
3. Choose any input and any level.
4. In the 5 questions, get at least 4 or a perfect grade.
5. Check if after every question there is a response for correct or wrong answer.
6. Check if the device congrats the user after the 5 questions.
7. If yes, put a correct mark in table 6. Otherwise, put a wrong mark.
8. Repeat 2 and 3.
9. In the 5 questions, intentionally get a failing mark of 3 or below.
10. Check if after every question there is a response for correct or wrong answer.



11. Check if the device gives a warning bad feedback to the user after the 5 questions.
12. If yes, put a correct mark in Table 7. Otherwise, put a wrong mark
13. Calculate the success rate of the trials using the given formula.

$$\text{Success Rate} = \frac{\text{Successful Trials}}{\text{Total Trials}} \times 100$$

3.5.7 Obtaining 90% success rate of all test

Objective:

To gather all the data from the tables to get a high percentage of success rate.

Materials:

- Data from all the tables under 3.5 Testing Procedure
- Calculator

1. Get the success rates from Table 1 to Table 7 and record it in Table 8.
2. If there is a multiple success rate per table in Table 1 to 7, get their average before recording it in Table 8. Use the equation below.

$$\text{Success Rate Average} = \frac{\text{Summation of Success Rate per Table}}{\text{Total Number of Success Rate}} \times 100$$

3. After recording all the success rates, each row in table 8 should have 1 success rate coming from Table 1 to Table 7.
4. Compute for the total success rate of the system using the given equation below.

$$\text{Success Rate Average} = \frac{\text{Summation of Success Rate per Table}}{\text{Total Number of Success Rate}} \times 100$$



3.6 Design Constraints

3.6.1 Cost

The learning tool box will be a quite high-priced device especially for students. The cost of materials is estimated to be at least 6 thousand pesos. The price may vary but it is estimated at that price because the materials that will be used need to function as its best.

3.6.2 Time

The device requires a lot of program coding. Hence, there will be more trial and error in terms of coding to make the learning toolbox function perfectly. Errors will be found as the process starts. Solving all errors will take a few months to clear it up as the researchers monitor the progress of the device.



Chapter 4

RESULTS AND DISCUSSION

This chapter shows all the data gathered while doing different tests of the system. Data such as results from different testing procedures, observations, and conclusions about each test.

4.1 Displaying Basic Arithmetic

In this test, using the LCD, the group will display random arithmetic along with the camera display.

Table 1 — Results for Displaying Basic Arithmetic

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Level 1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Level 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Level 3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Total											100%

Observation:

In any level, 5 out of 5 trials are successful for displaying random arithmetic problems as shown in Table 1. In each trial in the 5 trials per level, a random arithmetic problem was displayed alongside the camera and it follows the range of the “supposed to be” answered per level.

4.2 Detecting Hand Gestures Numbers Test

In this test, it requires a user and camera wherein the software will detect the gestures of what number will the user show and display it as the input in the system.

Table 2 — Results for Detecting Hand Gestures Number Test

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Hand Gesture	✓	✓	✗	✓	✓	✓	✗	✓	✓	✓	80%

Observation:

The process of hand tracking utilizes two libraries that the researchers used to program the software. The hand landmarks will then be visible to the user while tracking the hands in real-time. The researchers then added instructions to the code in terms of detecting number gestures from 1 to 10.

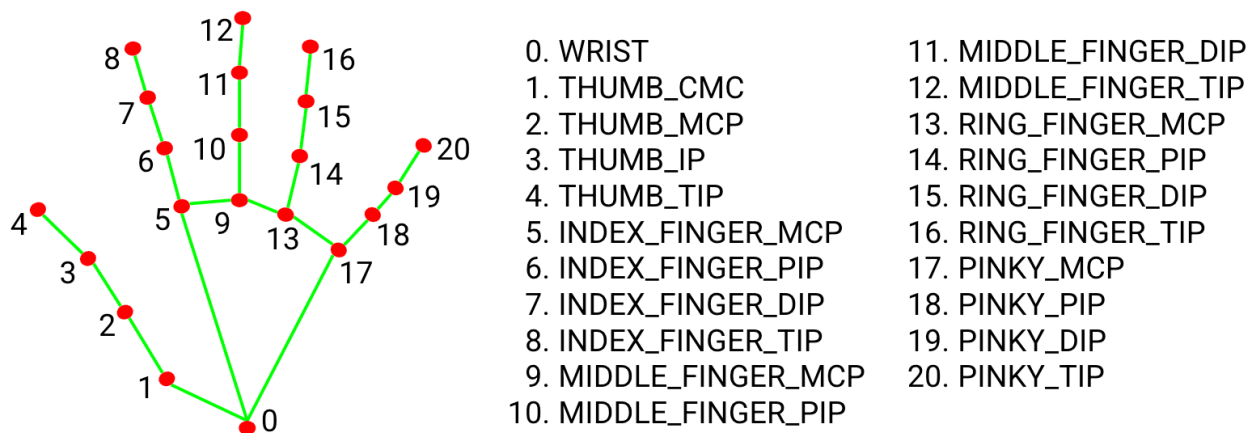


Figure 4 — Hand Landmarks



The researchers did different hand gestures in front of the camera to detect it. Out of 10 trials, the camera detected all the hand gestures, from 1 to 10, within the given time frame which is 10 seconds per number. Having said that, the researchers achieved a 100% success rate. Therefore, the camera can successfully detect hand gestures.

4.3 Reading Premade Numbers

In this testing, the test requires a user and a premade number or written numbers wherein the user will show the premade number in the camera and the system will receive it as an input.

Table 3 — Results for Reading Premade Numbers

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Premade 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Premade 1	✓	✗	✓	✓	✓	✓	✗	✓	✓	✓	80%
Premade 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Premade 3	✓	✓	✓	✗	✓	✓	✓	✓	✓	✗	80%
Premade 4	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	90%
Premade 5	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	90%
Premade 6	✓	✓	✓	✗	✗	✓	✓	✓	✓	✓	80%



Premade 7	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	90%
Premade 8	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓	80%
Premade 9	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	90%
Total											88%

Observation:

The process of developing the premade numbers uses machine learning. We trained the system with 1,017 datasets for each digit from 0 to 9, thus having a total of 10,170 datasets. After having trials and errors for each trained epoch, we managed to finalize the training to 38 epochs with an average of over 94% to 100% accuracy per digit.

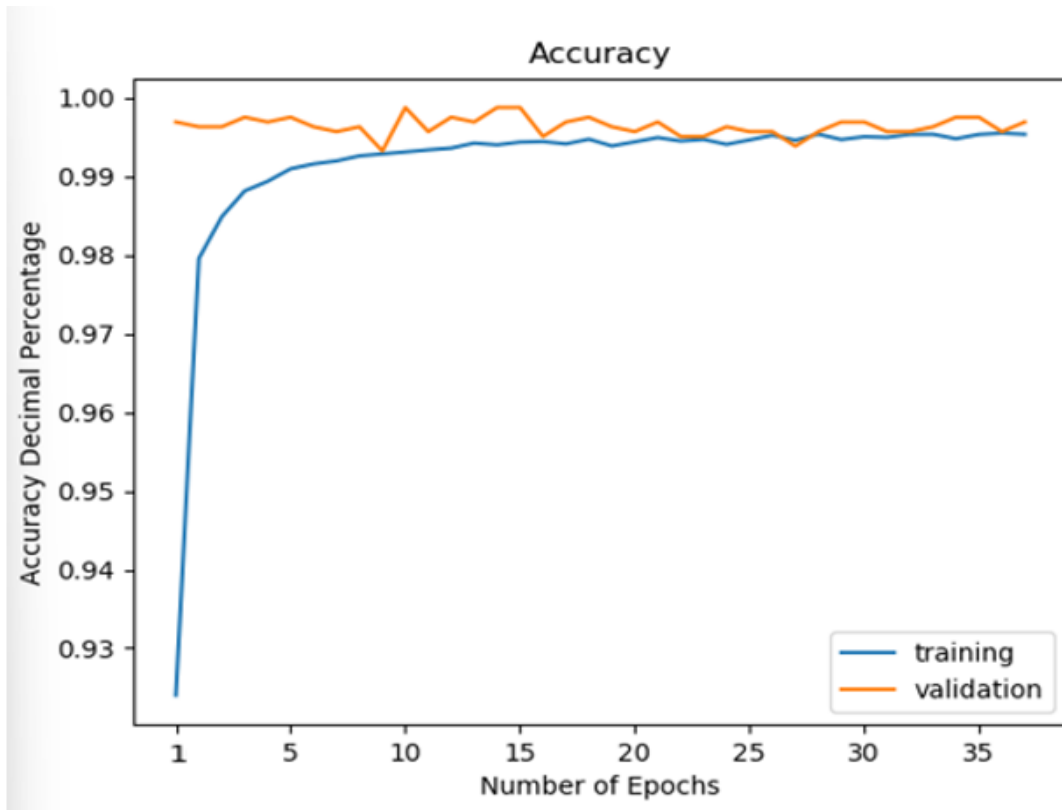


Figure 5 — Premade Testing Accuracy

Although some digits are having difficulty detecting, such as digit 7 is mistakenly detected as digit 1. As a result, we had errors as seen in Table 3, although we managed to compensate for this error by having the proctor overwrite the system whenever the child passes the level.

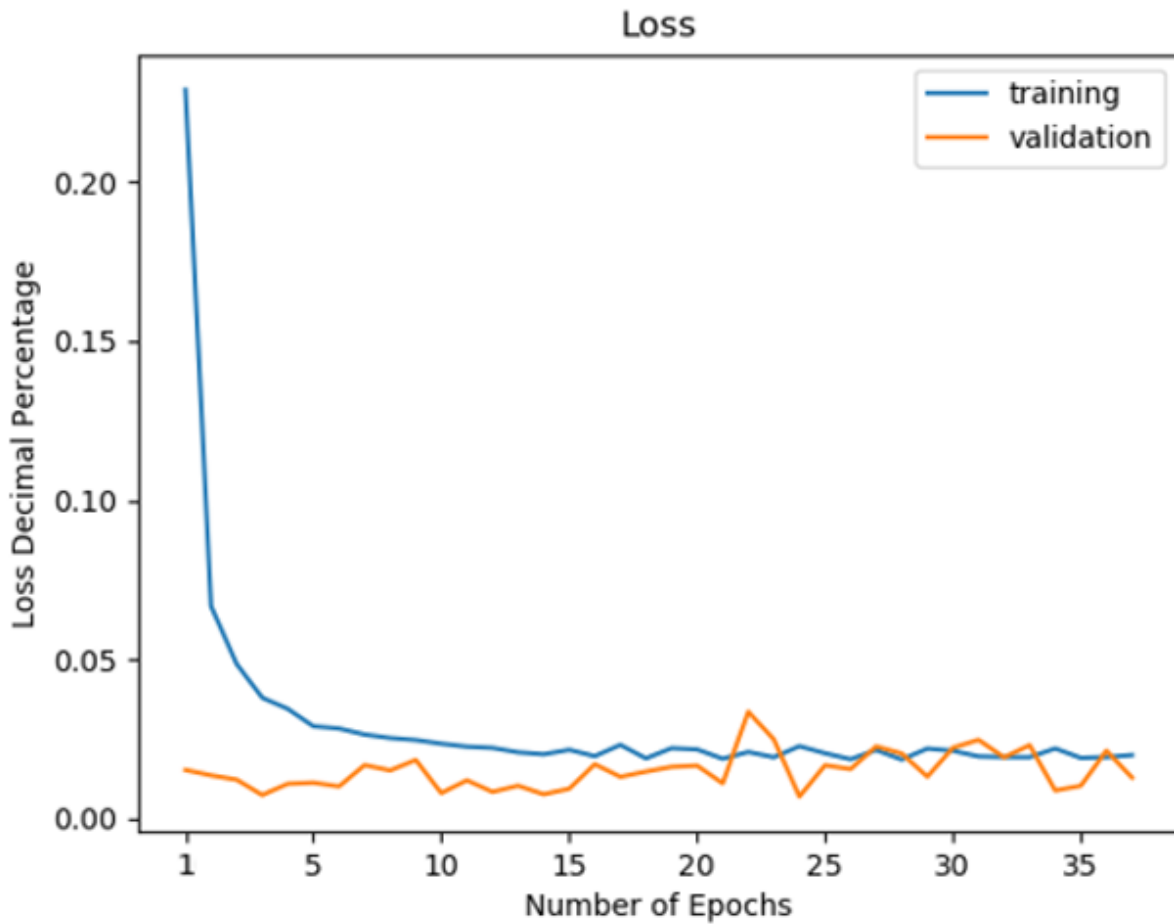


Figure 6 — Premade Testing Loss

4.4 Display Testing for Hand

In this testing, the user will now run the program for hand tracking and answer each level and the data will be gathered whenever the system read what the user wants to input.

Table 4 — Results for Displaying Testing for Hand

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Level 1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%



Level 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Level 3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Total											100%

Observation:

5 out of 5 trials are successful for the testing of displaying the numbers from the hand gestures as shown in Table 4. We answered all the 3 levels of difficulty and in those 5 trials, the prototype displayed on the LCD screen all the detected numbers from the user’s hand gestures.

4.5 Display Testing For Premade

In this testing, the user will now run the program for premade numbers and answer each level and the data will be gathered whenever the system reads what the user wants to input.

Table 5 — Results for Display Testing For Premade

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Level 1	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	90%
Level 2	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓	80%
Level 3	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	90%



Total	86.67%
--------------	---------------

Observation:

Each level has 5 trials to display the number detected using a premade number. 4 out of 5 trials only are successful. For Level 1, on the 2nd trial, the system did not display the expected number in the given time. Thus, this trial is marked as failed. As well for the Level 2, 4th trial and Level 3, 3rd trial. Giving only a total of 80% success rate for this testing.

4.6 Teaching and Correcting the Given Wrong Answer

In this testing, the system itself will talk using text-to-speech and give feedback on what the user answered. If the user's answer is right, the device will then respond with complimentary feedback. While if the user got the wrong answer, the device encourages the user and will reveal the correct answer. For every question, there will also be a response from the system.

Table 6 — Results for Teaching and Correcting the Given Wrong Answer (PASSED)

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Question 1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%



Question 4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Final Feedback	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Total											100%

Table 7 — Results for Teaching and Correcting the Given Wrong Answer (FAILED)

Trials	1	2	3	4	5	6	7	8	9	10	Success Rate (%)
Question 1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Question 5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Final Feedback	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100%
Total											100%



Observation:

The system provides a response whenever a question is answered correctly or incorrectly. A certain phrase will be said when the answer for the basic arithmetic problems is correct. Answering it incorrectly will also provide a certain phrase telling the user that the answer is incorrect and providing the correct answer for that question. The system also provides feedback if the user accumulates the required score to move to the next level.

4.7 Obtaining at least 90% success rate of all tests

In this part, all success rates will be gathered and checked if this project got at least 90% out of all the tests.

Table 8 — Results of Obtaining at least 90% success rate of all test

Table	1	2	3	4	5	6	7	Total Success Rate (%)
Success Rate per table (%)	100%	80%	88%	100%	86.67%	100%	100%	93.52%

Observation:

Based on Table 7, only testing process 3 and testing process 5 did not achieve at least 90% success rate. But by accumulating the success rate of all the testing procedures, the system obtained a 96.381% as shown in the table.



Chapter 5

CONCLUSION

Teaching children mathematics is surely not that easy nowadays, especially because of many distractions that surround them such as toys, digital devices, and their temper or attention span itself. This project makes sure that their attention and willingness to learn is targeted. As seen in Table 6, the device is surely programmed to interact with them through talking and giving them remarks that are responsible for making them annoyed or sad when they make a mistake.

The child user can also enhance their decision-making and their desire to learn more. As there are two modes to choose from, making them think what is best suited for them and maybe later, try the opposite mode for them to learn a different approach in answering mathematics.

After gathering all the data in the initial testing of the system, only in the part of reading premade numbers where result varies. As shown in Table 3, some premade numbers or written numbers are difficult for the system to read knowing some numbers look-alike such as 3 to 8 and 4 to 9. Thus, resulting in inputting an answer that is not supposed to be imputed as shown in table 5.

Ultimately, the prototype obtained at least 90% success rate thus the system is functioning well. The results that were obtained in the testing procedures are similar to the expected results for the system hence this prototype is successful and complete.



Chapter 6

RECOMMENDATION

To further improve the prototype, additional sets of arithmetic problems can be added. By adding more levels and increasing the difficulty of each added level, the scope of the prototype would not be limited to elementary students only.

For the design of the prototype, using a high-grade material for its case can provide more protection to its components. Providing a glass screen protector would help the LCD screen to receive less dirt, fingerprints, dust, and scratches. This will make sure that the device is more secure and less prone to hardware malfunction.

To improve the quality and save more time, making the premade number reading system be able to read 2 or more digits at the same time rather than inputting 1 digit at a time. Since premade numbers can be written or done in multiple digits together. Doing so will make 1 level less time consuming.

For the child user of this prototype, constant practice with basic arithmetic problems is essential. Practice provides more skill and mastery thus answering the arithmetic problems more quickly and accurately. So, it is suggested that after finishing a level, the child user should keep taking that same level for the purpose of mastering and learning. It is also suggested for the child to have a plain and clear background for a high efficiency for tracking hand gestures or capturing premade numbers.



References:

- [1] Makokha, A. (n.d.). Types of teaching methods. In M. Ongwae (Ed.), *A 14 days Teaching Methodology Course* (pp. 190–120). essay.
- [2] Helferich, E. (2020, June 24). *Hands-on learning helps brain development - here's why*. LUX BLOX. Retrieved October 7, 2021, from <https://www.luxblox.com/blogs/lux-corner/hands-on-learning-helps-brain-development-her e-s-why>.
- [3] GOODWIN UNIVERSITY. (2018, December 17). *Hands-on learning benefits for children*. Retrieved October 7, 2021, from <https://www.goodwin.edu/enews/benefits-of-hands>
- [4] Mollo, Luca & Bellotti, Francesco & Berta, Riccardo & De Gloria, Alessandro. (2016). Building Arduino-Based Tangible Serious Games for Elementary Mathematics and Physics. 60-69. 10.1007/978-3-319-50182-6_6.
- [5] T. Alzubi, R. Fernández, J. Flores, M. Duran and J. M. Cotos, "Improving the Working Memory During Early Childhood Education Through the Use of an Interactive Gesture Game-Based Learning Approach," in *IEEE Access*, vol. 6, pp. 53998-54009, 2018, doi: 10.1109/ACCESS.2018.2870575.
- [6] D. Yuliana and Suharjito, "Fingermath – Arithmetic Hand Gesture Game to Improve Early Childhood Mathematics Learning," 2019 International Conference on Information Management and Technology (ICIMTech), 2019, pp. 6-11, doi: 10.1109/ICIMTech.2019.8843756.
- [7] A. C. Sankaranarayanan, A. Veeraraghavan and R. Chellappa, "Object Detection, Tracking and Recognition for Multiple Smart Cameras," in *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1606-1624, Oct. 2008, doi: 10.1109/JPROC.2008.928758.
- [8] S. Guennouni, A. Ahaitouf and A. Mansouri, "Multiple object detection using OpenCV on an embedded platform," 2014 Third IEEE International Colloquium in Information Science and Technology (CIST), 2014, pp. 374-377, doi: 10.1109/CIST.2014.7016649
- [9] Fernandez, A., Garcia R., Sanchez L., Taruc, L., Villanueva, G. (2014). *Microcontroller-Based Interactive Clock (iClock)*. Manila.
- [10] Antonio, N., Dalangin, L., Faderon, T., Segovia, M., Vallespin, B. (2014). *Microcontroller-Based Interactive Spelling Toy (iSpell)*. Manila.
- [11] Aradillos, S., Marchan A., Pangadapun Y., Sudara G., Suhaili N. (2016). *Microcontroller-Based Interactive Math Toy (iMath)*. Manila.



[12] Daday, Mark. (2020). Recognition of Baybayin Symbols (Ancient Pre-Colonial Philippine Writing System) using Image Processing. *International Journal of Advanced Trends in Computer Science and Engineering*. 9. 594-598. 10.30534/ijatcse/2020/83912020.

[13] Abad Jr, Bienvenido. (2017). Proposed Image Pre-processing Techniques for Face Recognition Using OpenCV.

[14] *MediaPipe Hands*. MediaPipe. (2020). Retrieved February 21, 2022, from <https://google.github.io/mediapipe/solutions/hands.html>

[15] *7inch HDMI display*. LC Wiki. (n.d.). Retrieved February 21, 2022, from http://www.lcdwiki.com/7inch_HDMI_Display-C#Product_Description



APPENDIX



APPENDIX A

Curriculum Vitae of Ma'am Jocelyn Gahol



JOCELYN MARIE HALIMA GAHOL

Unit A Block 6 Lot 15 Hongkong Street Phase 1G Vista Verde North
Executive Village Llano, Caloocan City

+639606237260

celyngahol88@gmail.com

PROFESSIONAL TEACHER LICENSE NO.: 1253657



OBJECTIVES

To seek professional growth by excel in every way to tap my potentials making a highly competent, confident, and flexible individual and provide quality service as a professional individual.

PERSONAL INFORMATION

Age: 28 yrs. old
Birthday: July 23, 1993
Height: 5'3 feet
Weight: 60 kilograms
Sex: Female
Religion: Roman Catholic
Nationality: Filipino
Civil status: Single

EDUCATIONAL BACKGROUND

October 2016 – Present

Graduate Studies

Master of Arts in Education in Educational Measurement and Evaluation Philippine Normal University Taft Avenue, Manila

May 2013 – June 2013

Vocational Course Cookery

Novaliches Development Cooperative (training center) Novaliches, Quezon City

2009-2013

Tertiary Level

Bachelor of Secondary Education Major in Mathematics
Philippine Normal University Taft Avenue, Manila

2005-2009

Secondary Level

Holy Infant Montessori Center Novaliches, Caloocan City



1999-2005

Primary Level

Deparo Elementary School Deparo, Caloocan City

WORKING EXPERIENCES

Grade 11 and 12 General Mathematics, Statistics and Probability, Business Mathematics and Practical Research 2 Teacher -School Year 2018 - Present
Paso de Blas National High School, Valenzuela City

Mathematics 4 to 6, Science 6 and Values 6 Teacher – School Year 2015 – 2018
FEU Diliman – FERN College Diliman, Quezon City

Customer Advisor - December 2014 - April 2015
Security Bank Corporation E. Rodriguez - St. Luke's Medical Center Branch

Private Tutor- August 2014 - November 2014
Deparo, Caloocan City

Mathematics 8 and Values IV Teacher - School Year 2013-2014
Holy Infant Montessori Center Novaliches, Caloocan City

MAJOR ACHIEVEMENTS

Basic Education Research Fund Grantee and Presenter
Division of City Schools Valenzuela

Philippine Red Cross Youth Coach/Trainer
Philippine Red Cross Valenzuela City Chapter

Mathematics Educators Circle Member
Association of Math Educators of the Philippines

Mathematics 4 – 6 Curriculum Writer
FEU Diliman – FERN College – 2017 – 2018

Mathematics Grade School Department Task Force Coordinator
FEU Diliman – FERN College – School Year 2015 - 2018

Math Teacher Association of the Philippines (MTAP) Trainor
FEU Diliman – FERN College – School Year 2015 – 2018
Holy Infant Montessori Center - School Year 2013 – 2014

Grade School Student Council Adviser and Club Moderator
FEU Diliman – FERN College – School Year 2015 – 2018



CHARACTER REFERENCES

Ms. Elanie Scott V. Montemayor

Comworks Inc. - Senior Account Manager
+639989621570

Ms. Robelle R. Paderes

Paso de Blas National High School - Faculty
+639224266539

Mr. Mark Louie T. Silvestre

Colegio de San Agustin - Faculty
+639298288388

I hereby certify that I have given the following information to the best of my knowledge and belief.

A handwritten signature in black ink, appearing to read "Jocelyn Marie H. Gahol".

GAHOL, JOCELYN MARIE H.



APPENDIX B

Transcript of Interview with Ma'am Jocelyn Marie Gahol



Date: November 27, 2021

Interviewee: Ma'am Joeelyn Gahol (Teacher)

Interviewer: Our device is planned to be a interactive math learning device. It can generate basic math problems such as addition, subtraction, multiplication, and division. There will be two modes of answering these problems. First is the hand gesture mode, then the premade number mode. For the hand gesture mode, the device can detect what number the child is gesturing using one hand. On the other hand, the device can also detect premade number and determine what number it is. The problem and detected numbers will be displayed in the screen for the child to see.

Ma'am Celyn: Sorry, question, sino yung target respondents niyo dito? Elementary? Sino specifically sa grades school?

Interviewer: Buong elementary po kasi may level level po kasi.

Ma'am Celyn: Pero yung ano niyo lang, yung competency, MDAS lang? Paanong MDAS dun sa grade 6?

Interviewer: For grade 6 math problems po, yun po sa tingin namin ang pasok para sa mga grade 6 kasi wala pa po ata silang algebra.

Ma'am Celyn: Meron na ha. May algebra na, yes. Kaya I suggest ano kaya kung magfocus kayo sa lower grade. Sa grade 1-3 lang. Kasi kung ang target niyo lang talaga is mdas, yun lang talaga mga grade levels na masscope. Kasi pagdating ng grade 4, hindi na masyadong emphasized ang mdas eh. So, kung ang target is mdas, dapat ang target respondents niyo ay marunong mag mdas.

Interviewer: Maam actually po, example po for level 1, ang sakop niya lang is addition and subtraction. Wala pa pong multiplication and division.

Ma'am Celyn: Ahhh edi wag niyo na lang i-identify as mdas, ilagay niyo na lang as basic operation kasi kapag basic operation, it doesn't mean naman nandun yung apat. Kasi kung i-identify niyo agad yung apat, dapat makikita ko agad yung apat na operations so wag niyo siya ilabel talaga na ganon.

Ma'am Celyn: Magcheck din kayo sa internet, dapat ipattern niyo siya sa deped curriculum na dun talaga ma i-identify nae to yung specific competency, eto yung specific objective during grade 1, na eto yung tinatarget niyo para mamaster ng bata. Alam ko meron sa internet pero sigery to check and send after this.

Interviewer: Okay po Ma'am, we will try to research din po regarding on that lalo na sa deped curriculum.

Ma'am Celyn: Hand gesture, kailangan masesense ng camera or webcam, tama ba? Paano kapag medyo lowkey specs ng webcam tapos hindi niya clear ma identify or madetect yung hand gesture?



Interviewer: Kami po magproprovide ng webcam po. Kami bubuo ng lahat, buong device po.

Ma'am Celyn: Ang maganda nito ay yung ganyang tool ay magamit ng mga may kapansanan kasi hindi sila makapagsalita so instead, they can use gesture. Kaso mahirap maghanap ng respondents na ganon.

Ma'am Celyn: Bakit gesture? What's with the gesture?

Interviewer: Yun po napili naming kasi po naisip po namin na much better po na interactive ang learning experience ng bata and mas ma reretain niya po ang study niya due to it being interactive.

Ma'am Celyn: Actually, yun nga iniisip ko e, kasi at that age kasi, mas ineemphasize kasi ang skills for reading and writing which is medyo hindi natatarget ng tool ninyo. Yung tool is more on kinesthetic part e more on galaw, gesture.

Interviewer: Opo mam para din po active din po siya while answering and participating while natututo.

Ma'am Celyn: Oo pwede naman, pwede naman para magkaroon ng active participation.

Interviewer: Next question po naming mam is, What do you think is the best response po when the student got the wrong answer?

Ma'am Celyn: Ang tanong niyo ba sa akin is anong lalabas, im assuming na kunyari naggesture siya tapos correct, may magpo pop up na "wow" ganon?

Interviewer: One of the advises din po kasi samin ng panelist is pano daw po kung nagkamalim what is the better way po to show na mali siya, kung ano po audio cue dapat iparinig na hindi po nakakatrauma sa bata?

Ma'am Celyn: Gamit na lang siguro kayo ng happy face kapag tama siya, with matching paputok and "congratulations" and other way around kapag mali yung bata, lagyan niyo na lang ng cutie na face tapos lagyan ng "better luck next time" or "try again".

Interviewer: Kung 20 items po ba ay masyadong marami o sakto lang para sa elementary level?

Ma'am Celyn: Siguro kaya ang gawin niyo na lang, for example for grade 1, 10 items, grade 2 15 items, then grade 3 20 items ganon, ganon na lang siguro. Maganda kung tatlong level na lang kasi mga bata sila eh. Ganon ba gagawin niyo? Yung parang isang test lang lahat na?

Interviewer: Yes po, may progress po siya. Kaya po nag iisip po kami ng passing level bago po siya makapag next test or level.

Ma'am Celyn: Edi erase na yung 10 15 20 items. Gawin niyong 15 items lahat. 5 easy, 5 moderate, 5 difficult. Automatically the passing will be 3 over 5. What will happen if hindi siya makapass? Magreretake siya ng easy level?



Interviewer: Yes po maam. Sa dulo po kasi ng test, example after itake ng bata yung test and may 5 siyang mali out of 15, sa dulo po ng program namin is sasabihin kung saan siya nagkamali. Dun din po magkakaroon ng interactive learning, ituturo rin po ng device namin yung tamang sagot.

Ma'am Celyn: Bakit sa dulo pa? Actually sa philosophy of children education, kapag nagkamali raw ang isang bata, dapat feedback agad. Ganito na lang kaya gawin niyo. Talagang may magdidiscuss. Number 1 item tama, so nandiyan si happy face and paputok. Number 2 nagkamali, nandiyan si sad face pati yung sinasabi niyong discussion part. Kasi it will not take some time naman para madiscuss yung pagkakamali niya eh. Kasi parang kapag nagtuloy tuloy sa dulo, kasi ang mga batang yan ang iikli ng attention span niyan eh, let's say yung bata ang dami niyang pagkakamali so sa dulo kinig lang siya nang kinig sa dami ng kanyang pagkakamali, baka maboringan si bagets.

Ma'am Celyn: Paano kung naka 5 tries na yung bata at hindi pa rin siya makapasa pasa?

Interviewer: Parang problem na po ng magulang yun ah.

Ma'am Celyn: Btw ano ba target respondent niyo? Yung specific? Homo or parang mixed na may makulit tapos may matalino?

Interviewer: mixed po maam.

Ma'am Celyn: Kailan niyo ba to balak gawin?

Interviewer: Next semester pa po Ma'am for our thesis 2 which is the chapter 4 and 5 po.

Interviewers : question lang po mam, since tatalon po ng level. Ilan po ba ang best percentage para po makapasa siya mula level 1 to level 2 ?

Ma'am Celyn : Make it 5 items each round.

Interviewers : Pano po mam tig 10 items po per level

Ma'am Celyn Gahol : since ang gusto nyo is may level level, make it 5 items because it will make the quiz more exciting for the child. Erase na natin ung 10 15 20, gawin niyo nalang tig fa five each.

Interviewer : Much better din po yun Ma'am kasi mas nakakakuha din po kasi ng attention nila

Ma'am Celyn : Yes, so 3 ang passing grade per level.

Ma'am Celyn: I advise you guys to interview din pala at least 12 students. Like tanong niyo mga study habits niya sa maths. Kung saan siya mas madaling matuto sa math, pictures ba or sanay ba siya na mental math.

Interviewers : Last question Ma'am is after we made the prototype, will you recommend this po ba as a learning tool for elementary students?



Ma'am Celyn : Yes, lalo na provided niyo yung tool. I will look forward to that device.

Interviewers : Thank you so much Ma'am

Ma'am Celyn : Thank you din! See you soon.



Figure 7 — Interview via Google Meet with Ma'am Jocelyn Marie Gahol



APPENDIX C

Flowchart

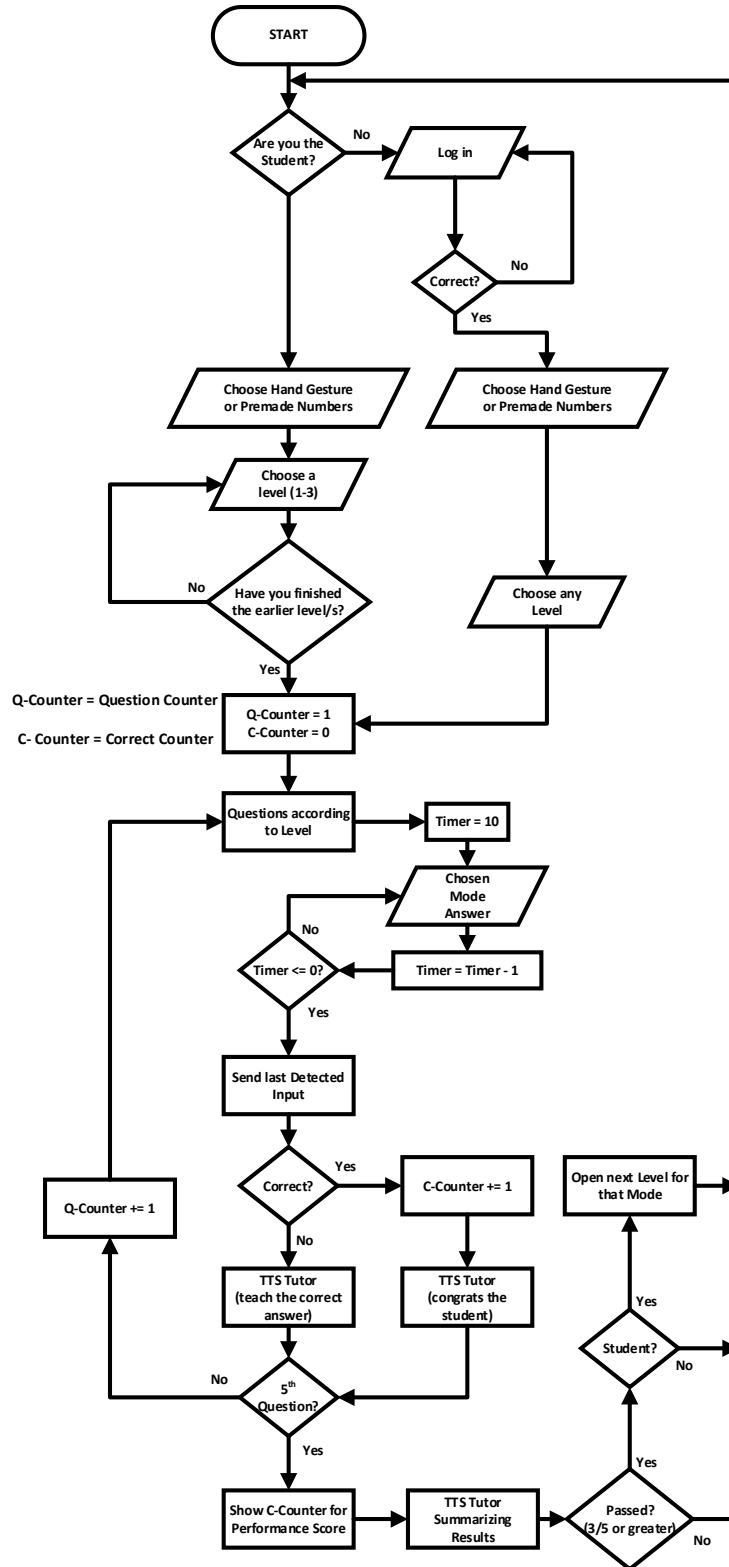


Figure 8 — System Flowchart

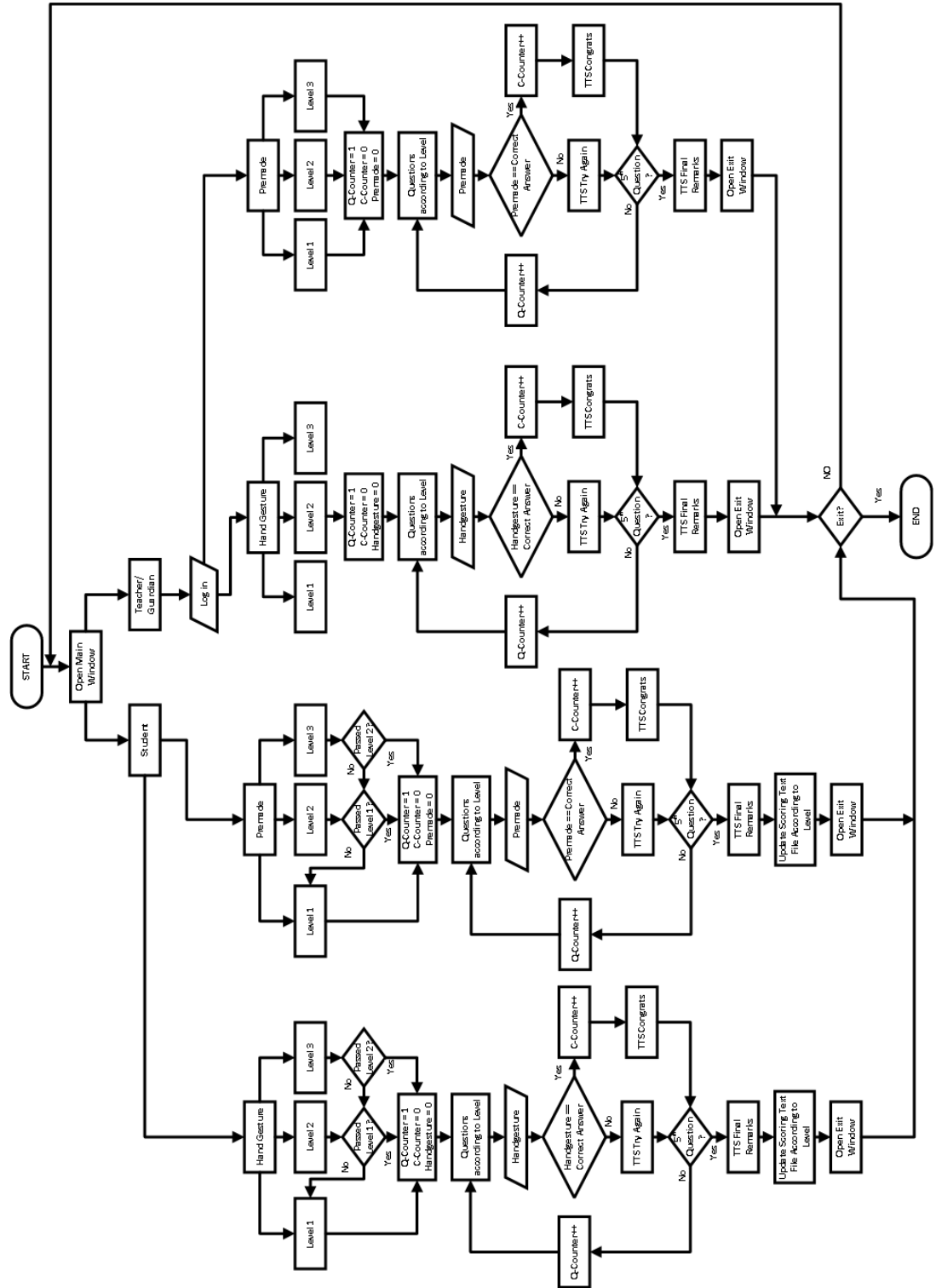


Figure 9 — Program Flowchart



APPENDIX D

Schematic Diagram

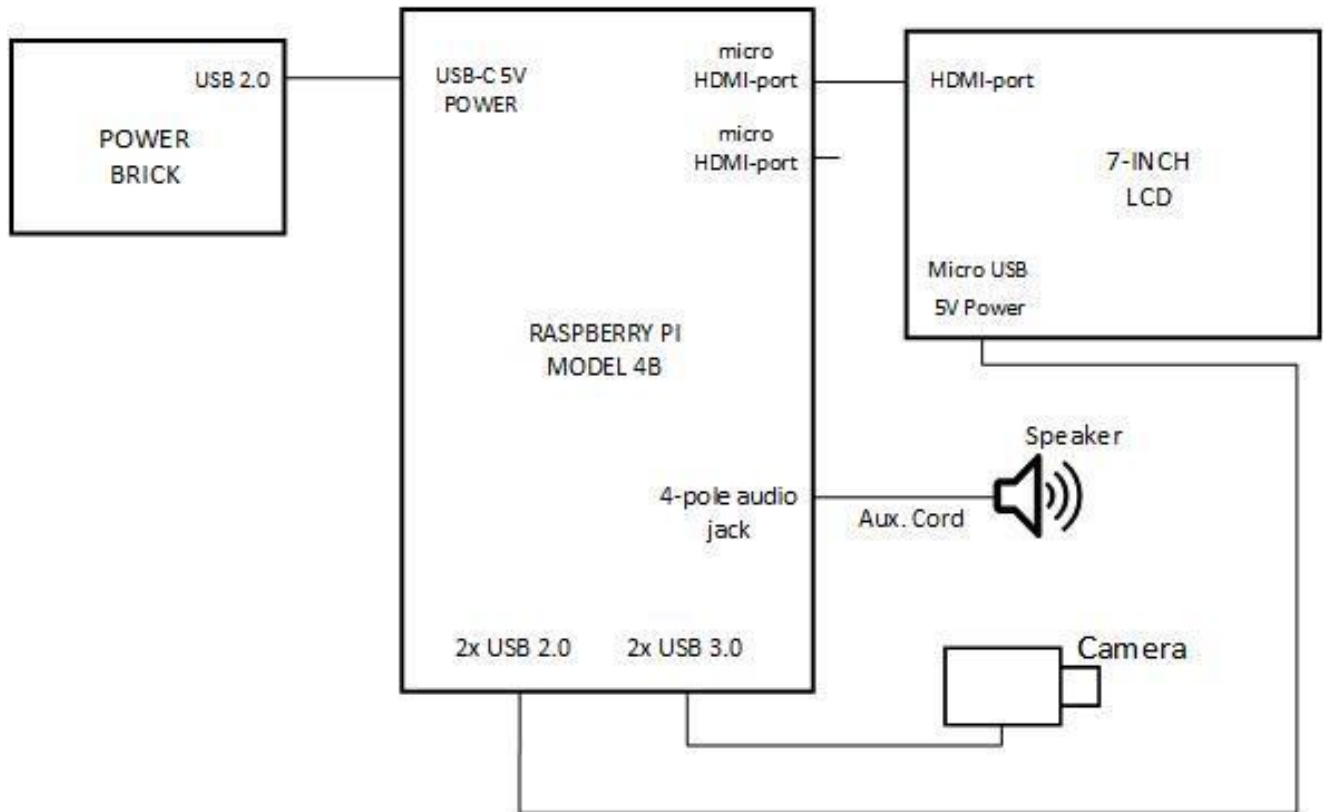


Figure 10 — Schematic Diagram



APPENDIX E

Data Sheet

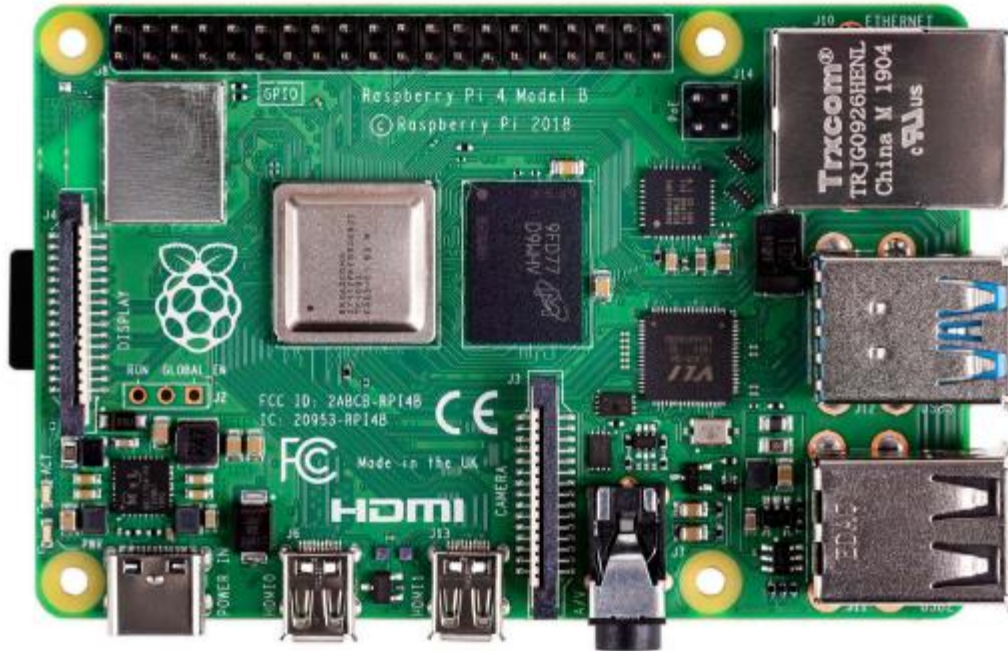


Figure 11 — Raspberry pi 4B

The Raspberry Pi 4 Model B is the most recent addition to the popular Raspberry Pi computer line. It outperforms the previous-generation Raspberry Pi 3 Model B+ in terms of CPU speed, multimedia performance, memory, and connection while maintaining backward compatibility and power consumption. The Raspberry Pi 4 Model B offers desktop performance equivalent to entry-level x86 PC systems to the end-user. A high-performance 64-bit quad-core processor, dual-display support at resolutions up to 4K via a pair of micro-HDMI ports, hardware video decode at up to 4Kp60, up to 4GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 5.0, Gigabit Ethernet, USB 3.0, and PoE capability are among the key features of this product (via a separate PoE HAT add-on).

Hardware

- Quad-core 64-bit ARM-Cortex A72 running at 1.5GHz
- 1, 2 and 4 Gigabyte LPDDR4 RAM options
- H.265 (HEVC) hardware decode (up to 4Kp60)
- H.264 hardware decode (up to 1080p60)
- VideoCore VI 3D Graphics
- Supports dual HDMI display output up to 4Kp60



Symbol	Parameter	Minimum	Maximum	Unit
VIN	5V Input Voltage	-0.5	6.0	V

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
V_{IL}	Input low voltage ^a	VDD_IO = 3.3V	-	-	TBD	V
V_{IH}	Input high voltage ^a	VDD_IO = 3.3V	TBD	-	-	V
I_{IL}	Input leakage current	TA = +85°C	-	-	TBD	μA
C_{IN}	Input capacitance	-	-	TBD	-	pF
V_{OL}	Output low voltage ^b	VDD_IO = 3.3V, IOL = -2mA	-	-	TBD	V
V_{OH}	Output high voltage ^b	VDD_IO = 3.3V, IOH = 2mA	TBD	-	-	V
I_{OL}	Output low current ^c	VDD_IO = 3.3V, VO = 0.4V	TBD	-	-	mA
I_{OH}	Output high current ^c	VDD_IO = 3.3V, VO = 2.3V	TBD	-	-	mA
R_{PU}	Pullup resistor	-	TBD	-	TBD	kΩ
R_{PD}	Pulldown resistor	-	TBD	-	TBD	kΩ

^a Hysteresis enabled

^b Default drive strength (8mA)

^c Maximum drive strength (16mA)



Figure 12 — LCD Screen 7inch

It has a 7" standard display, a 1024 600 hardware resolution, and a software configurable resolution of up to 1920x1080. It has a capacitive touch screen with maximum capability for 5 point touch. It offers backlight control alone, the backlight may be switched off to save power, and it supports Raspberry Pi, BB Black, Banana Pi, and other standard small PCs. It may be used as a general-purpose HDMI monitor, for example, by connecting it to a computer HDMI as a sub-display. It is used as a raspberry pi display, supporting Raspbian, Ubuntu, Kali-Linux, Kodi, win10 IOT, single-touch, free drive, and lastly, it works as a PC monitor, supporting win7, win8, win10 system 5 point touch (XP and previous version systems: single-point touch), and free drive.

Product Parameters

Size: 7.0 (inch)

Active Area : 154.2144*85.92(mm)

SKU: MPI7002

Product Size: 164.9*124.27 (mm)

Resolution: 1024*600 (dots)

Package Size: 211*150*53 (mm)

Touch: five-point capacitive touch

Rough Weight(Package containing): 405 (g)

Power Dissipation: 0.62A*5V @1024*600



Figure 13 — Webcam

A 1080p HD webcam that has 30 degrees rotatable adjustment. Its optical lens has high precision and no distortion within use. It is designed for both laptops and desktops and compatible with USB 2.0 with auto white balance, and auto color correction. This supports different operating systems.

Product Parameters :

Material: ABS.

Color: black

DSP chip: no driver

Image sensor: CMOS

Sensor pixels: 2 million

Product size: camera : 8cmx8cmx3cm, line length :140cm

Interface: USB 2.0

Frame rate: 30 FPS

Length: 150 cm

Brightness/color: automatically adjusts

Additional features: built-in microphone



APPENDIX F

Pictures



Figure 14 — Prototype

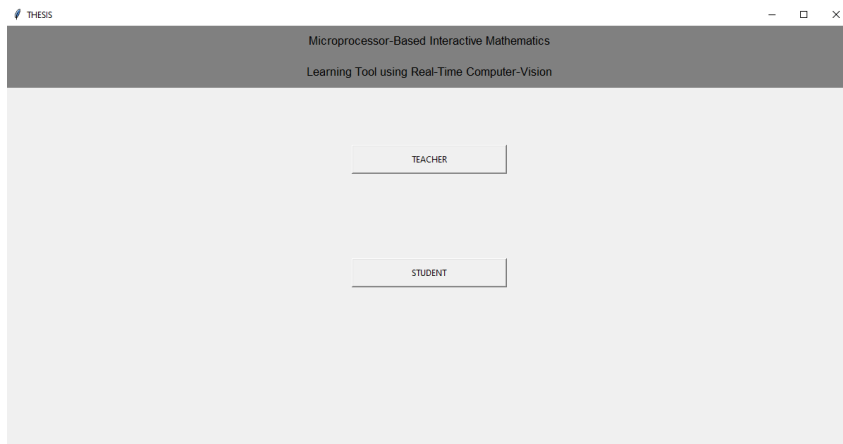


Figure 15 — G.U.I (Homepage)

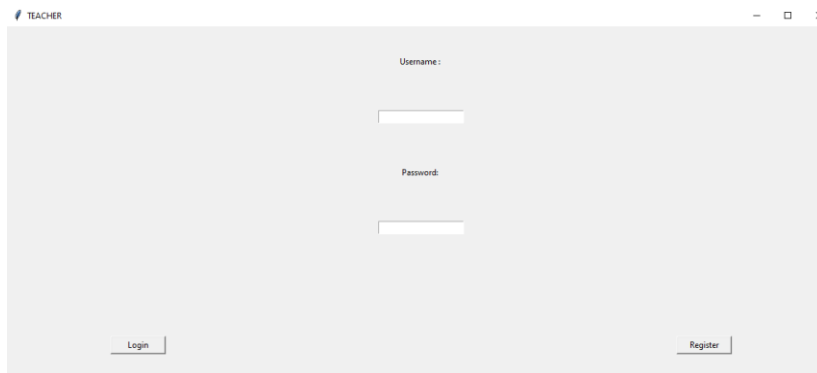


Figure 16 — G.U.I (Teacher's Interface)

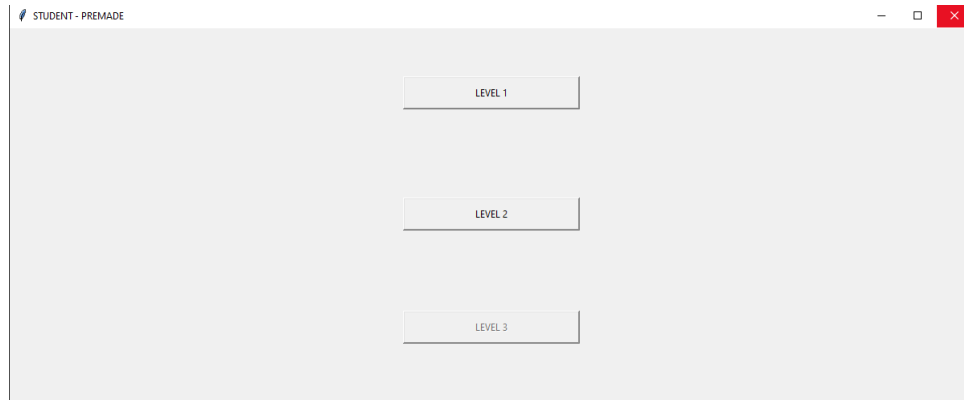


Figure 17 — G.U.I (Student's Interface)

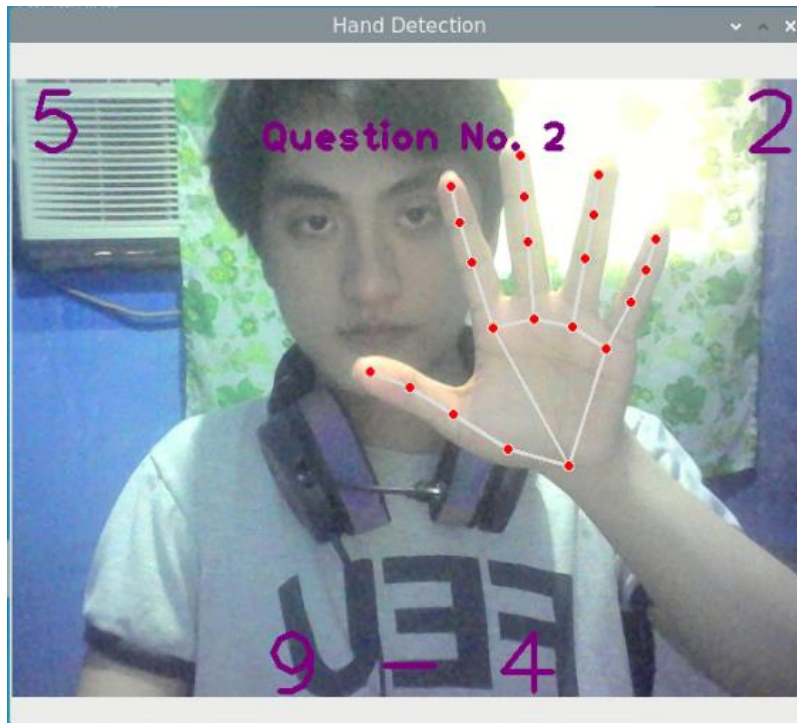


Figure 18 — Testing of the Hand tracking

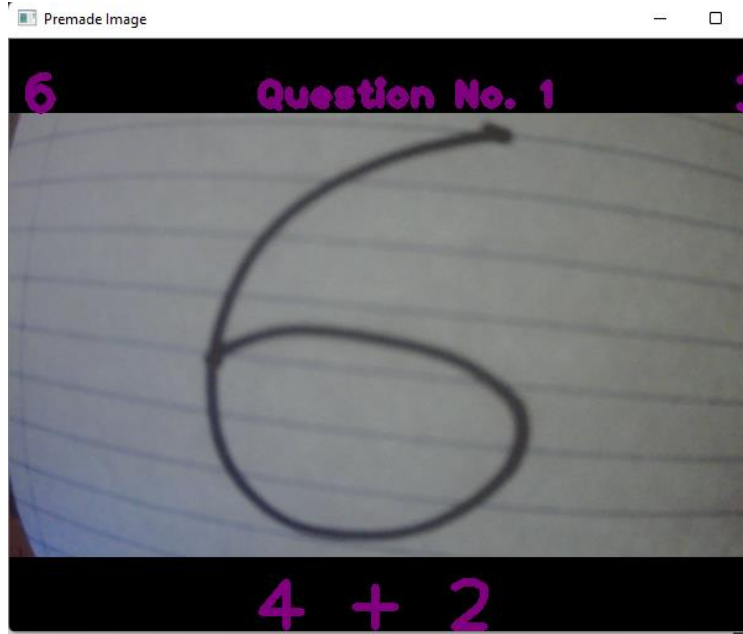


Figure 19 — Testing of the Premade Numbers



Figure 20 — One of the proponents with the Grade 1 student



Figure 21 — Two of the proponents with the Grade 1 student and Grade 3 Student

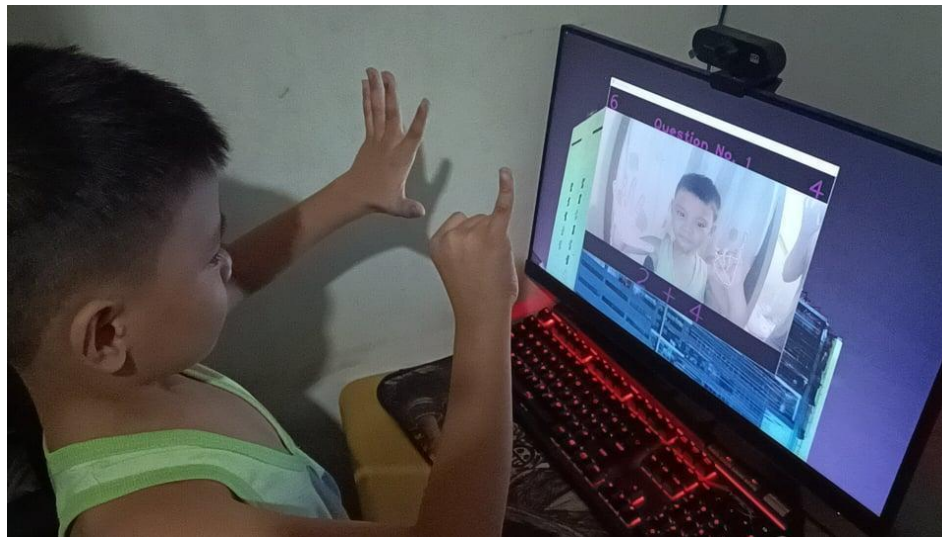


Figure 22 — Testing of the prototype with the Grade 1 student (Hand tracking)



Figure 23 — Testing of the prototype with the Grade 1 student (2) (Hand tracking)



Figure 24 — Testing of the prototype with the Grade 1 student (3) (Hand tracking)

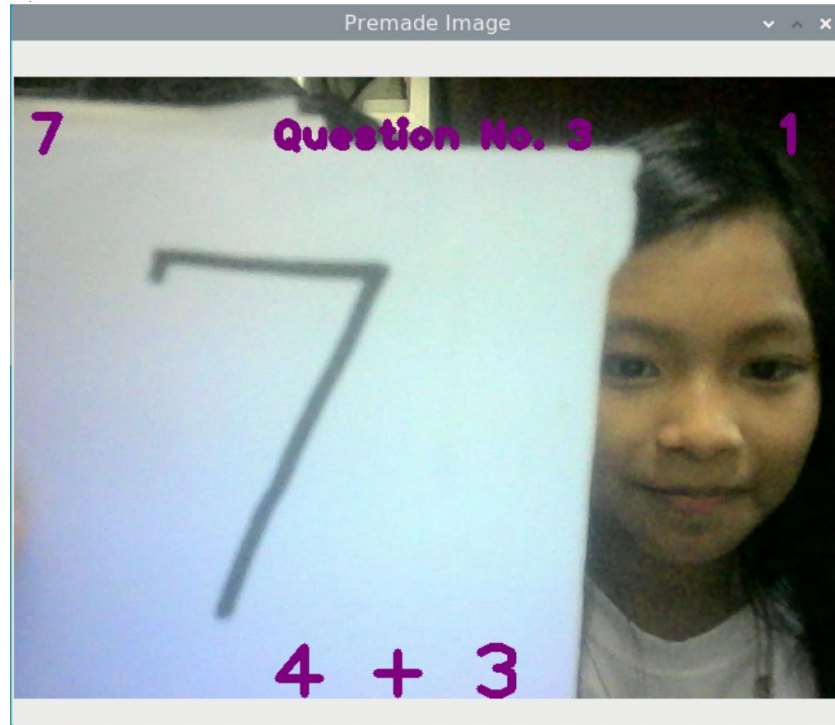


Figure 25 — Testing of the prototype with the Grade 3 student (Premade Numbers)

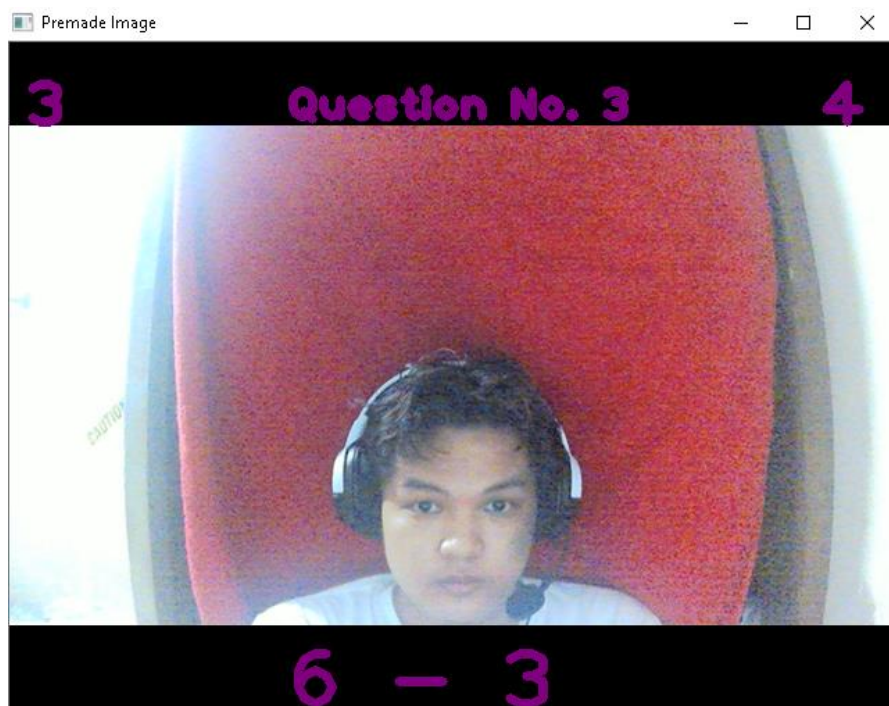


Figure 26 – Testing different backgrounds for answering Premade Numbers

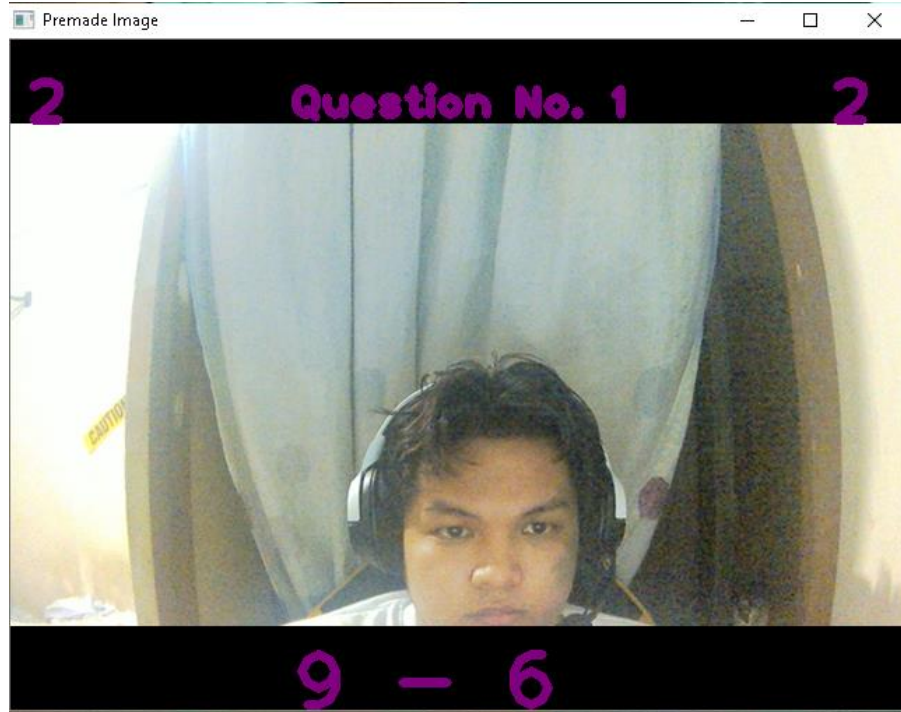


Figure 27 – Testing different backgrounds for answering Premade Numbers

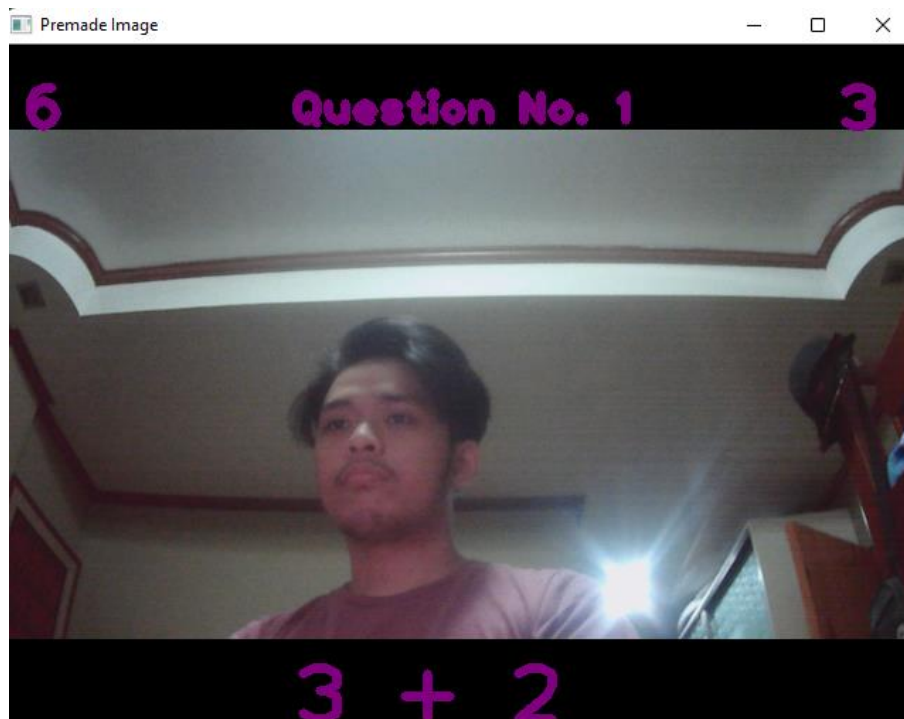


Figure 28 – Testing different backgrounds for answering Premade Numbers

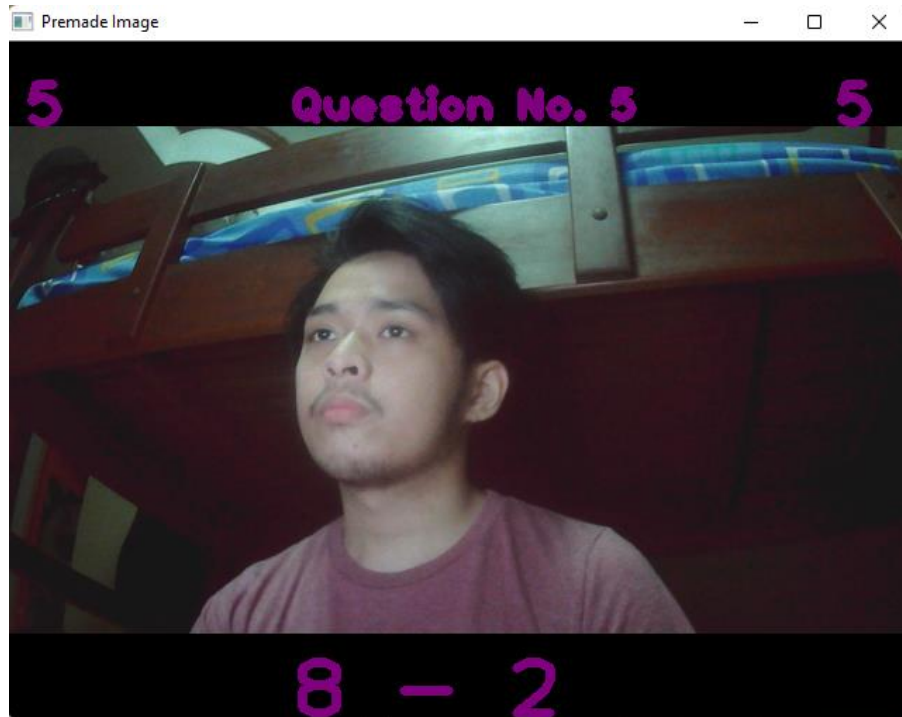


Figure 29 – Testing different backgrounds for answering Premade Numbers



Figure 30 – Invalid Gesture (Thumbs-Up)



Figure 31 – Invalid Gesture (Rock On)



Figure 32 – Invalid Gesture (Call me)



Figure 33 – Invalid Gesture (Odd-Finger Up)



Figure 34 – Invalid Gesture (Isko-Ong Tandem)



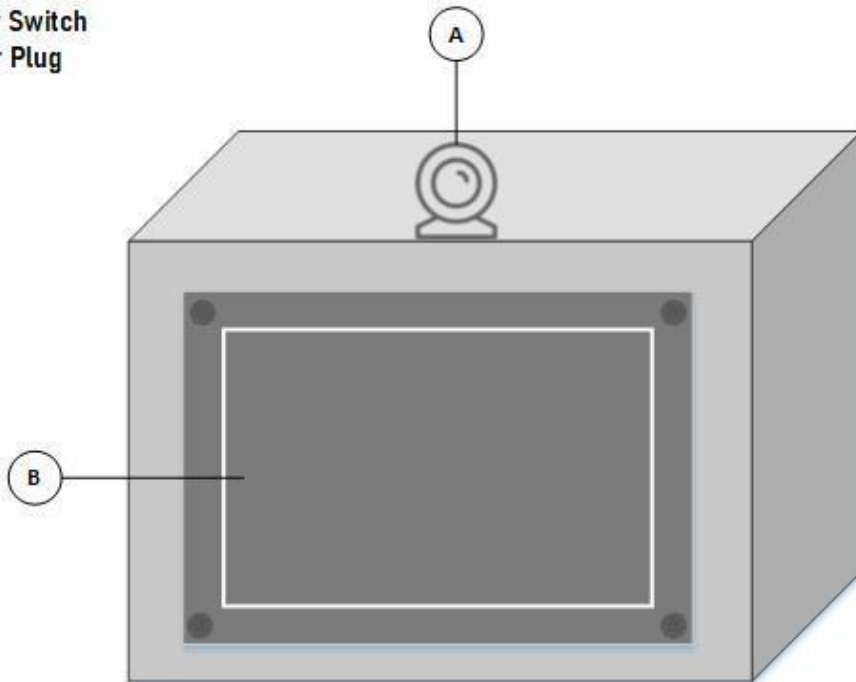
APPENDIX G

User's Manual

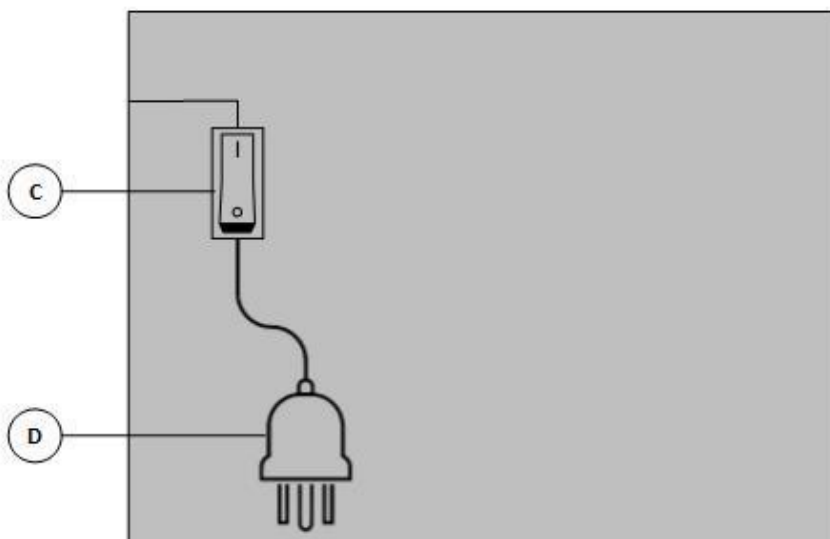


Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision

- A - Webcam
- B - Touchscreen LCD
- C - Power Switch
- D - Power Plug



Back of the Device





This project is a compact device that is intended for learning and giving quizzes based on basic mathematical arithmetic for children in grades 1 to 3. It is a plug and play device wherein you just need to plug it on the power socket for it to turn on. The user or the teacher will run the software using the touchscreen LCD, although the device also accepts peripherals such as the mouse and keyboard.

For running the executable file, the user needs to double click it on the desktop as seen on the LCD screen. The software will then run, and the Graphical user Interface will appear. First, there's the teacher mode and the student mode. Clicking the student mode accesses the two modes of answering the test, hand tracking and premade. Hand tracking uses the user's hand as an input to the system for answering the problem, wherein the premade is where the user will show the premade as a mode of answering the problem. Both of which have 3 levels of difficulties. The students need to have at least 3 points for every 5 questions in order to proceed to the next level.

When accessing the teacher mode, we must rerun the program again after the student finishes the test. The teacher has the access for all the difficulties of the system, wherein every difficulty is unlocked. The purpose for the teacher mode is that he or she can retest the student or can go advanced whenever the teacher decides to teach at a more advanced level. For turning off the device, the teacher will then open the terminal and input `sudo shutdown -h now`.



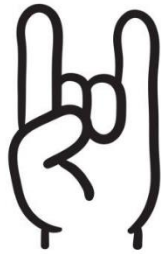
INVALID GESTURES



Isko-Ong Tandem



Thumbs-Up



Rock On



Call Me



Odd-Finger Up



APPENDIX H

Program Code



```
import time
import tkinter
from tkinter import *
import os
import cv2 as cv
import sys
from playsound import playsound
from gtts import gTTS
import pickle
import numpy as np
import mediapipe as mp
import random
import pygame as py
import keras

class level1():
    def __init__(self):
        self.mat = ["+", "-"]
        self.matt = random.choice(self.mat)
        self.nani3 = 0
        self.nani4 = 0
        self.naniTTS = "banani"
        if self.matt == "+":
            self.nani1 = random.randint(2, 9)
            self.nani2 = random.randint(2, 9)
            self.nani4 = self.nani1 + self.nani2
            self.naniTTS = "{} plus {}".format(self.nani1, self.nani2)
            while self.nani4 >= 10:
                self.nani1 = random.randint(2, 9)
                self.nani2 = random.randint(2, 9)
                self.nani4 = self.nani1 + self.nani2
                self.naniTTS = "{} plus {}".format(self.nani1, self.nani2)
            self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)
        elif self.matt == "-":
            self.nani1 = random.randint(2, 9)
            self.nani2 = random.randint(2, 9)
            self.nani4 = self.nani1 - self.nani2
            self.naniTTS = "{} minus {}".format(self.nani1, self.nani2)
            while self.nani4 <= 2:
                self.nani1 = random.randint(2, 9)
                self.nani2 = random.randint(2, 9)
                self.nani4 = self.nani1 - self.nani2
                self.naniTTS = "{} minus {}".format(self.nani1, self.nani2)
            self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)

    def getNani3(self):
        return self.nani3

    def getNani4(self):
        return self.nani4

    def getNaniTTS(self):
        return self.naniTTS

class level2():
```



```
def __init__(self):
    self.mat = ["+", "-"]
    self.matt = random.choice(self.mat)
    self.nani3 = 0
    self.nani4 = 0
    self.naniTTS = "banani"
    if self.matt == "+":
        self.nani1 = random.randint(3, 19)
        self.nani2 = random.randint(3, 19)
        self.nani4 = self.nani1 + self.nani2
        self.naniTTS = "{} plus {}".format(self.nani1, self.nani2)
        while self.nani4 > 20:
            self.nani1 = random.randint(3, 19)
            self.nani2 = random.randint(3, 19)
            self.nani4 = self.nani1 + self.nani2
            self.naniTTS = "{} plus {}".format(self.nani1, self.nani2)
        self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)

    elif self.matt == "-":
        self.nani1 = random.randint(5, 19)
        self.nani2 = random.randint(5, 19)
        self.nani4 = self.nani1 - self.nani2
        self.naniTTS = "{} minus {}".format(self.nani1, self.nani2)
        while self.nani4 <= 6:
            self.nani1 = random.randint(5, 19)
            self.nani2 = random.randint(5, 19)
            self.nani4 = self.nani1 - self.nani2
            self.naniTTS = "{} minus {}".format(self.nani1, self.nani2)
        self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)

def getNani3(self):
    return self.nani3

def getNani4(self):
    return self.nani4

def getNaniTTS(self):
    return self.naniTTS

class level3():
    def __init__(self):
        self.mat = ["+", "-", "*"]
        self.matt = random.choice(self.mat)
        self.mutt1 = [2, 3, 5]
        self.nani3 = 0
        self.nani4 = 0
        self.naniTTS = "banani"
        if self.matt == "+":
            self.nani1 = random.randint(10, 49)
            self.nani2 = random.randint(10, 49)
            self.nani4 = self.nani1 + self.nani2
            self.naniTTS = "{} plus {}".format(self.nani1, self.nani2)
            while self.nani4 > 50:
                self.nani1 = random.randint(10, 49)
                self.nani2 = random.randint(10, 49)
                self.nani4 = self.nani1 + self.nani2
                self.naniTTS = "{} plus {}".format(self.nani1, self.nani2)
```



```
self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)

elif self.matt == "-":
    self.nani1 = random.randint(10, 49)
    self.nani2 = random.randint(10, 49)
    self.nani4 = self.nani1 - self.nani2
    self.naniTTS = "{} minus {}".format(self.nani1, self.nani2)
    while self.nani4 <= 15:
        self.nani1 = random.randint(10, 49)
        self.nani2 = random.randint(10, 49)
        self.nani4 = self.nani1 - self.nani2
        self.naniTTS = "{} minus {}".format(self.nani1, self.nani2)
    self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)

elif self.matt == "*":
    self.nani1 = random.choice(self.mutt1)
    self.nani2 = random.randint(2, 9)
    self.nani4 = self.nani1 * self.nani2
    self.naniTTS = "{} times {}".format(self.nani1, self.nani2)
    while self.nani4 > 50:
        self.nani1 = random.choice(self.mutt1)
        self.nani2 = random.randint(2, 9)
        self.nani4 = self.nani1 * self.nani2
        self.naniTTS = "{} times {}".format(self.nani1, self.nani2)
    self.nani3 = "{} {} {}".format(self.nani1, self.matt, self.nani2)

def getNani3(self):
    return self.nani3

def getNani4(self):
    return self.nani4

def getNaniTTS(self):
    return self.naniTTS

class handLandmarkDetector():
    def __init__(self, mode=False, maxHands=2, modelC=1, detectionCon=0.5,
trackCon=0.5):
        self.mode = mode
        self.maxHands = maxHands
        self.modelC = modelC
        self.detectionCon = detectionCon
        self.trackCon = trackCon
        self.mpHands = mp.solutions.hands
        self.hands = self.mpHands.Hands(self.mode, self.maxHands,
self.modelC, self.detectionCon, self.trackCon)
        self.mpDraw = mp.solutions.drawing_utils

    def detect_landmarks(self, image, draw=True, draw_connections=True,
draw_default_style=False):
        imageRGB = cv.cvtColor(image, cv.COLOR_BGR2RGB)
        land_mark_data = []
        hand_classified_landmarks = [[], []]
        results = self.hands.process(imageRGB)
        landmarks = results.multi_hand_landmarks
        data = None
```



```
if landmarks:
    for hand_landmarks in landmarks:
        for id, landmark in enumerate(hand_landmarks.landmark):
            h, w, c = image.shape
            px, py = int(landmark.x * w), int(landmark.y * h)
            data = (id, px, py)
            land_mark_data.append(data)
            if draw and not draw_connections:
                self.mpDraw.draw_landmarks(image, hand_landmarks)
            elif draw and draw_connections and not draw_default_style:
                self.mpDraw.draw_landmarks(image, hand_landmarks,
self.mpHands.HAND_CONNECTIONS)
            elif draw and draw_connections and draw_default_style:
                self.mpDraw.draw_landmarks(image, hand_landmarks,
self.mpHands.HAND_CONNECTIONS,
self.mp_drawing_styles.get_default_hand_landmarks_style(),
self.mp_drawing_styles.get_default_hand_connections_style())
            if land_mark_data[0][1] > land_mark_data[4][1]:
                if len(land_mark_data) > 20:
                    hand_classified_landmarks[1] = land_mark_data[0:21]
                    hand_classified_landmarks[0] = land_mark_data[21::]
                else:
                    hand_classified_landmarks[1] = land_mark_data[0:21]
            elif land_mark_data[4][1] > land_mark_data[0][1]:
                if len(land_mark_data) > 20:
                    hand_classified_landmarks[0] = land_mark_data[0:21]
                    hand_classified_landmarks[1] = land_mark_data[21::]
                else:
                    hand_classified_landmarks[0] = land_mark_data[0:21]
            return hand_classified_landmarks, image

def count_up_fingers(self, data):
    fingers = [[], []]
    if len(data[1]) != 0: # right hand
        if (data[1][3][1] > data[1][4][1]): # thumb
            fingers[1].append(1) # finger array / append detection
        else:
            fingers[1].append(0)
            self.sirR = 1
        if (data[1][5][2] > data[1][8][2] and data[1][7][2] >
data[1][8][2]): # pointy
            fingers[1].append(1)
        else:
            fingers[1].append(0)
            self.sirR = 1
        if (data[1][9][2] > data[1][12][2] and data[1][11][2] >
data[1][12][2]): # middle
            fingers[1].append(1)
        else:
            fingers[1].append(0)
            self.sirR = 1
        if (data[1][13][2] > data[1][16][2] and data[1][15][2] >
data[1][16][2]): # ring
            fingers[1].append(1)
        else:
```




```
fingers[1].append(0)
self.sirR = 1
if (data[1][17][2] > data[1][20][2] and data[1][19][2] >
data[1][20][2]): # pinky
    fingers[1].append(1)
else:
    fingers[1].append(0)
    self.sirR = 1

if (data[1][7][2] > data[1][8][2] and data[1][19][2] >
data[1][20][2]): # pointy and pinky
    if (data[1][9][2] > data[1][12][2] and data[1][11][2] >
data[1][12][2]) or (
        data[1][13][2] > data[1][16][2] and data[1][15][2] >
data[1][16][2]) or (
        data[1][3][1] > data[1][4][1]):
        fingers[1].append(0)
    else:
        self.sirR = 0

if (data[1][3][1] > data[1][4][1]) and (data[1][19][2] >
data[1][20][2]): # thumb and pinky
    if (data[1][5][2] > data[1][8][2] and data[1][7][2] >
data[1][8][2]) or (
        data[1][9][2] > data[1][12][2] and data[1][11][2] >
data[1][12][2]) or (
        data[1][13][2] > data[1][16][2] and data[1][15][2] >
data[1][16][2]):
        fingers[1].append(0)
    else:
        self.sirR = 0

if (data[1][15][2] > data[1][16][2]) and (data[1][19][2] >
data[1][20][2]): #2 junks
    if (data[1][9][2] > data[1][12][2] and data[1][11][2] >
data[1][12][2]) or (data[1][5][2] > data[1][8][2] and data[1][7][2] >
data[1][8][2]) or (data[1][3][1] > data[1][4][1]):
        fingers[1].append(0)
    else:
        self.sirR = 0

if data[1][3][1] > data[1][4][1] and data[1][11][2] >
data[1][12][2] and data[1][19][2] > data[1][20][2]: # thumb and middle and
pinky
    if (data[1][5][2] > data[1][8][2] and data[1][7][2] >
data[1][8][2]) or (
        data[1][13][2] > data[1][16][2] and data[1][15][2] >
data[1][16][2]):
        fingers[1].append(0)
    else:
        self.sirR = 0

if data[1][11][2] > data[1][12][2]: # middle
    if (data[1][5][2] > data[1][8][2] and data[1][7][2] >
data[1][8][2]) or (data[1][13][2] > data[1][16][2] and data[1][15][2] >
data[1][16][2]) or data[1][19][2] > data[1][20][2] or data[1][3][1] >
data[1][4][1]:
```



```
        fingers[1].append(0)
    else:
        self.sirR = 0

    if data[1][3][1] > data[1][4][1]:
        if (data[1][5][2] > data[1][8][2] and data[1][7][2] >
data[1][8][2]) or (data[1][9][2] > data[1][12][2] and data[1][11][2] >
data[1][12][2]) or (data[1][13][2] > data[1][16][2] and data[1][15][2] >
data[1][16][2]) or (data[1][17][2] > data[1][20][2] and data[1][19][2] >
data[1][20][2]):
            fingers[1].append(0)
        else:
            self.sirR = 0

    if len(data[0]) != 0: # left hand
        if (data[0][3][1] < data[0][4][1]): # thumb
            fingers[0].append(1) # finger array / append detection
        else:
            fingers[0].append(0)
            self.sirL = 1
        if (data[0][5][2] > data[0][8][2] and data[0][7][2] >
data[0][8][2]): # pointy
            fingers[0].append(1)
        else:
            fingers[0].append(0)
            self.sirL = 1
        if (data[0][9][2] > data[0][12][2] and data[0][11][2] >
data[0][12][2]): # middle
            fingers[0].append(1)
        else:
            fingers[0].append(0)
            self.sirL = 1
        if (data[0][13][2] > data[0][16][2] and data[0][15][2] >
data[0][16][2]): # ring
            fingers[0].append(1)
        else:
            fingers[0].append(0)
            self.sirL = 1
        if (data[0][17][2] > data[0][20][2] and data[0][19][2] >
data[0][20][2]): # pinky
            fingers[0].append(1)
        else:
            fingers[0].append(0)
            self.sirL = 1

        if (data[0][7][2] > data[0][8][2]) and (data[0][19][2] >
data[0][20][2]): # pointy and pinky
            if (data[0][3][1] < data[0][4][1]) or (
                data[0][9][2] > data[0][12][2] and data[0][11][2] >
data[0][12][2]) or (
                data[0][13][2] > data[0][16][2] and data[0][15][2] >
data[0][16][2]):
                fingers[0].append(0)
            else:
                self.sirL = 0

        if (data[0][3][1] < data[0][4][1]) and (data[0][19][2] >
```



```
data[0][20][2]): # thumb and pinky
    if (data[0][5][2] > data[0][8][2] and data[0][7][2] >
data[0][8][2]) or (
        data[0][9][2] > data[0][12][2] and data[0][11][2] >
data[0][12][2]) or (
        data[0][13][2] > data[0][16][2] and data[0][15][2] >
data[0][16][2]):
        fingers[0].append(0)
    else:
        self.sirL = 0

    if (data[0][19][2] > data[0][20][2]) and (data[0][15][2] >
data[0][16][2]): #two jonks
        if (data[0][9][2] > data[0][12][2] and data[0][11][2] >
data[0][12][2]) or (data[0][5][2] > data[0][8][2] and data[0][7][2] >
data[0][8][2]) or (data[0][3][1] < data[0][4][1]):
            fingers[0].append(0)
        else:
            self.sirL = 0

    if (data[0][3][1] < data[0][4][1]) and (data[0][11][2] >
data[0][12][2]) and (
        data[0][19][2] > data[0][20][2]): # thumb and middle and
pinkie
        if (data[0][5][2] > data[0][8][2] and data[0][7][2] >
data[0][8][2]) or (
            data[0][13][2] > data[0][16][2] and data[0][15][2] >
data[0][16][2]):
            fingers[0].append(0)
        else:
            self.sirL = 0

    if (data[0][11][2] > data[0][12][2]): # middle
        if (data[0][3][1] < data[0][4][1]) or (
            data[0][5][2] > data[0][8][2] and data[0][7][2] >
data[0][8][2]) or (
            data[0][13][2] > data[0][16][2] and data[0][15][2] >
data[0][16][2]) or (
            data[0][17][2] > data[0][20][2] and data[0][19][2] >
data[0][20][2]):
            fingers[0].append(0)
        else:
            self.sirL = 0

    if (data[0][3][1] < data[0][4][1]):
        if (data[0][5][2] > data[0][8][2] and data[0][7][2] >
data[0][8][2]) or (data[0][9][2] > data[0][12][2] and data[0][11][2] >
data[0][12][2]) or (data[0][13][2] > data[0][16][2] and data[0][15][2] >
data[0][16][2]) or (data[0][17][2] > data[0][20][2] and data[0][19][2] >
data[0][20][2]):
            fingers[0].append(0)
        else:
            self.sirL = 0

    return fingers
```



```
def getsirickyright(self):
    return self.sirR

def getsirickyleft(self):
    return self.sirL

def niceone():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w,h))
    py.display.set_caption("NICE ONE")
    display = py.image.load(r'MP3\NICEONE.jpg')
    surface.fill((255,255,255))
    surface.blit(display,(0,0))
    py.display.update()
    playsound("MP3/NICEONE.mp3")

def nicetry():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w,h))
    py.display.set_caption("NICE TRY")
    display = py.image.load(r'MP3\NICETRY.jpg')
    surface.fill((255,255,255))
    surface.blit(display,(0,0))
    py.display.update()
    playsound("MP3/NICETRY.mp3")

def question1():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w, h))
    py.display.set_caption("Question 1")
    display = py.image.load(r'MP3\Q1.jpg')
    surface.fill((255, 255, 255))
    surface.blit(display, (0, 120))
    py.display.update()
    py.event.wait()

def question2():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w, h))
    py.display.set_caption("Question 2")
    display = py.image.load(r'MP3\Q2.jpg')
    surface.fill((255, 255, 255))
    surface.blit(display, (0, 120))
    py.display.update()
    py.event.wait()

def question3():
    py.init()
    w = 640
```



```
h = 480
surface = py.display.set_mode((w, h))
py.display.set_caption("Question 3")
display = py.image.load(r'MP3\Q3.jpg')
surface.fill((255, 255, 255))
surface.blit(display, (0, 120))
py.display.update()
py.event.wait()

def question4():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w, h))
    py.display.set_caption("Question 3")
    display = py.image.load(r'MP3\Q4.jpg')
    surface.fill((255, 255, 255))
    surface.blit(display, (0, 120))
    py.display.update()
    py.event.wait()

def question5():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w, h))
    py.display.set_caption("Question 5")
    display = py.image.load(r'MP3\Q5.jpg')
    surface.fill((255, 255, 255))
    surface.blit(display, (0, 120))
    py.display.update()
    py.event.wait()

def TY():
    py.init()
    w = 640
    h = 480
    surface = py.display.set_mode((w, h))
    py.display.set_caption("Thank You!")
    display = py.image.load(r'MP3\TY.jpg')
    surface.fill((255, 255, 255))
    surface.blit(display, (0, 120))
    py.display.update()
    py.event.wait()
    py.quit()

def handtracking():
    global screen1
    screen6.destroy()
    screen1 = Tk()
    screen1.title("STUDENT - HANDTRACKING")
    screen1.geometry("1200x600")
    btn1 = Button(screen1, text="LEVEL 1", height="2", width="30",
command=handtrackinglevel1)
    btn1.pack(side=TOP, pady=60)

    file = open('StudentScore/scorelevel1.txt', 'r')
```



```
verify = int(file.read())
file.close()
if verify >= 3:
    btn2 = Button(screen1, text="LEVEL 2", height="2", width="30",
command=handtrackinglevel2, state=NORMAL)
    btn2.pack(side=TOP, pady=50)
else:
    btn2 = Button(screen1, text="LEVEL 2", height="2", width="30",
state=DISABLED)
    btn2.pack(side=TOP, pady=50)

file = open('StudentScore/scorelevel2.txt', 'r')
verify2 = int(file.read())
file.close()
if verify2 >= 3:
    btn3 = Button(screen1, text="LEVEL 3", height="2", width="30",
command=handtrackinglevel3, state=NORMAL)
    btn3.pack(side=TOP, pady=50)
else:
    btn2 = Button(screen1, text="LEVEL 3", height="2", width="30",
state=DISABLED)
    btn2.pack(side=TOP, pady=50)

def register():
    screen4 = Toplevel(screen3)
    screen4.title("TEACHER")
    screen4.geometry("1200x600")

    global username
    global password
    global username1
    global password1
    username = StringVar()
    password = StringVar()

    lbl = Label(screen4, text="Username: ")
    lbl.pack(side=TOP, pady=80)
    username1 = Entry(screen4, textvariable=username)
    username1.pack(side=TOP, pady=20)
    lbl = Label(screen4, text="Password: ")
    lbl.pack(side=TOP, pady=80)
    password1 = Entry(screen4, textvariable=password)
    password1.pack(side=TOP, pady=20)
    btn = Button(screen4, text="Register", width=10, height=1,
command=register_user)
    btn.pack(side=TOP, pady=40)

def register_user():
    username_info = username.get()
    password_info = password.get()

    file = open(username_info, "w")
    file.write(username_info + "\n")
    file.write(password_info)
    file.close()

    username1.delete(0, END)
```



```
password1.delete(0, END)

def teacher():
    global screen3
    screen3 = Toplevel(screen)
    screen3.title("TEACHER")
    screen3.geometry("1200x600")

    global username_verify
    global password_verify
    global username2
    global password2
    username_verify = StringVar()
    password_verify = StringVar()

    lbl = Label(screen3, text="Username : ")
    lbl.pack(side=TOP, pady=40)
    username2 = Entry(screen3, textvariable=username_verify)
    username2.pack(side=TOP, pady=20)
    lbl = Label(screen3, text="Password: ")
    lbl.pack(side=TOP, pady=40)
    password2 = Entry(screen3, textvariable=password_verify)
    password2.pack(side=TOP, pady=20)
    btn = Button(screen3, text="Login", width=10, height=1,
command=login_verify)
    btn.pack(side=LEFT, padx=150, pady=40)
    btn = Button(screen3, text="Register", width=10, height=1,
command=register)
    btn.pack(side=RIGHT, padx=150, pady=40)

    # print("Login session started")

def login_verify():
    username3 = username_verify.get()
    password3 = password_verify.get()
    username2.delete(0, END)
    password2.delete(0, END)
    screen5 = Toplevel(screen3)
    screen5.title("TEACHER")
    screen5.geometry("400x150")

    list_of_files = os.listdir()
    if username3 in list_of_files:
        file1 = open(username3, "r")
        verify = file1.read().splitlines()
        if password3 in verify:
            lbl = Label(screen5, text="Login Success")
            lbl.pack(side=TOP, pady=20)
            btn = Button(screen5, text="DONE", width=10, height=1,
command=teachermodes)
            btn.pack(side=TOP, pady=20)

        else:
            screen5.destroy()

    else:
        screen5.destroy()
```



```
def teacherlevelshand():
    global screen2
    screen6.destroy()
    screen2 = Tk()
    screen2.title("TEACHER - HANDTRACKING")
    screen2.geometry("1200x600")

    btn = Button(screen2, text="LEVEL 1", height="2", width="30",
command=handtrackinglevel11)
    btn.pack(side=TOP, pady=60)
    btn = Button(screen2, text="LEVEL 2", height="2", width="30",
command=handtrackinglevel22)
    btn.pack(side=TOP, pady=50)
    btn = Button(screen2, text="LEVEL 3", height="2", width="30",
command=handtrackinglevel33)
    btn.pack(side=TOP, pady=60)

def teacherlevelspremade():
    global screen2
    screen6.destroy()
    screen2 = Tk()
    screen2.title("TEACHER - PREMADE")
    screen2.geometry("1200x600")

    btn = Button(screen2, text="LEVEL 1", height="2", width="30",
command=premadelevel11)
    btn.pack(side=TOP, pady=60)
    btn = Button(screen2, text="LEVEL 2", height="2", width="30",
command=premadelevel22)
    btn.pack(side=TOP, pady=50)
    btn = Button(screen2, text="LEVEL 3", height="2", width="30",
command=premadelevel33)
    btn.pack(side=TOP, pady=60)

def teachermodes():
    global screen6
    screen3.destroy()
    screen6 = Tk()
    screen6.title("TEACHER - MODES")
    screen6.geometry("1200x600")

    btn = Button(screen6, text="HANDTRACKING", height="2", width="30",
command=teacherlevelshand)
    btn.pack(side=TOP, pady=80)
    btn = Button(screen6, text="PREMADE", height="2", width="30",
command=teacherlevelspremade)
    btn.pack(side=TOP, pady=80)

def studentmodes():
    global screen6
    screen6 = Toplevel(screen)
    screen6.title("STUDENT - MODES")
    screen6.geometry("1200x600")

    btn = Button(screen6, text="HANDTRACKING", height="2", width="30",
command=handtracking)
```




```
btn.pack(side=TOP, pady=80)
btn = Button(screen6, text="PREMADE", height="2", width="30",
command=premade)
btn.pack(side=TOP, pady=80)

def premade():
    global screen10
    screen6.destroy()
    screen10 = Tk()
    screen10.title("STUDENT - PREMADE")
    screen10.geometry("1200x600")
    btn1 = Button(screen10, text="LEVEL 1", height="2", width="30",
command=premadelevel1)
    btn1.pack(side=TOP, pady=60)

    file = open('StudentScorePremade/scorelevel1.txt', 'r')
    verify = int(file.read())
    file.close()
    if verify >= 3:
        btn2 = Button(screen10, text="LEVEL 2", height="2", width="30",
command=premadelevel2, state=NORMAL)
        btn2.pack(side=TOP, pady=50)
    else:
        btn2 = Button(screen10, text="LEVEL 2", height="2", width="30",
state=DISABLED)
        btn2.pack(side=TOP, pady=50)

    file = open('StudentScorePremade/scorelevel2.txt', 'r')
    verify2 = int(file.read())
    file.close()
    if verify2 >= 3:
        btn3 = Button(screen10, text="LEVEL 3", height="2",
command=premadelevel3, width="30", state=NORMAL)
        btn3.pack(side=TOP, pady=50)
    else:
        btn2 = Button(screen10, text="LEVEL 3", height="2", width="30",
state=DISABLED)
        btn2.pack(side=TOP, pady=50)

def premadelevel1():
    width = 1080
    height = 720

    cap = cv.VideoCapture(0)
    cap.set(16, width)
    cap.set(18, height)

    screen.destroy()
    screen10.destroy()

    pickle_in = open("model_trained_38.p", "rb")
    model = pickle.load(pickle_in)
    global pcc1
    qc = 1
    pcc1 = 0

    while qc <= 5:
```



```
if qc == 1:
    question1()
elif qc == 2:
    question2()
elif qc == 3:
    question3()
elif qc == 4:
    question4()
elif qc == 5:
    question5()
neldan = level1()
nani3 = neldan.getNani3()
nani4 = neldan.getNani4()
TTS = neldan.getNaniTTS()
ad = gTTS(TTS)
wil = "Question No. {}".format(qc)
salisi = gTTS(wil)
salisi.save("qcounter.mp3")
playsound("qcounter.mp3")
os.remove("qcounter.mp3")
jc = 0
jopa = 0
noel = 7
ad.save("problem.mp3")
playsound("problem.mp3")
os.remove("problem.mp3")
while jopa >= 0:
    success, imgOriginal = cap.read()
    img = np.asarray(imgOriginal)
    img = cv.resize(img, (32, 32))
    img = preProcessing(img)
    img = img.reshape(1, 32, 32, 1)
    classIndex = int(model.predict_classes(img))
    cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
    cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
    cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
    cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)

    cv.imshow("Premade Image", imgOriginal)

    jopa = jopa + 1
    jepoy = jopa % 50

    if jepoy == 0:
        noel = noel - 1

    if jopa == 350:
        jc = int(classIndex)
        break
    if cv.waitKey(1) & 0xFF == ord('q'):
        break
cv.destroyAllWindows()
if jc == nani4:
```



```
pcc1 = pcc1 + 1
niceone()
correct = "{} is equal to {}".format(TTS, jc)
cclip = gTTS(correct)
cclip.save("correctclip.mp3")
playsound("correctclip.mp3")
os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}". The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")

    qc = qc + 1
    cap.release()
    TY()
    pcc011 = "Your grade is {} out of 5".format(pcc1)
    kerson = gTTS(pcc011)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen11
    screen11 = Tk()
    screen11.geometry("800x525")
    int_var = tkinter.IntVar(screen11, pcc1)
    lbl = Label(screen11, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen11, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if pcc1 >= 3:
        lbl = Label(screen11, text="Congratulations! You Passed!",
font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen11, text="Better luck next time", font=('Arial',
17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed1.mp3")
    lbl = Label(screen11, text="Do you want to exit?", font=('Arial', 12))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen11, text="Yes", height="4", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen11, text="No", height="4", width="10",
command=systemrestart11)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("StudentScorePremade/scorelevel1.txt").st_size == 0:
        pscoring1()
    else:
        pscoringverify1()
```



```
def pscoring1():
    file = open('StudentScorePremade/scorelevel1.txt', 'r+')
    file.write('{}'.format(pcc1))
    file.close()

def pscoringverify1():
    file = open('StudentScorePremade/scorelevel1.txt', 'r')
    verify = int(file.read())
    file.close()
    if pcc1 > verify:
        file = open('StudentScorePremade/scorelevel1.txt', 'w')
        file.write('{}'.format(pcc1))
        file.close()

def premadelevel11():
    width = 640
    height = 480

    cap = cv.VideoCapture(0)
    cap.set(8, width)
    cap.set(12, height)
    screen.destroy()
    screen2.destroy()

    pickle_in = open("model_trained_38.p", "rb")
    model = pickle.load(pickle_in)
    global pcc11
    qc = 1
    pcc11 = 0

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
            question5()
        neldan = level1()
        nani3 = neldan.getNani3()
        nani4 = neldan.getNani4()
        TTS = neldan.getNaniTTS()
        ad = gTTS(TTS)
        wil = "Question No. {}".format(qc)
        salisi = gTTS(wil)
        salisi.save("qcounter.mp3")
        playsound("qcounter.mp3")
        os.remove("qcounter.mp3")
        jc = 0
        jopa = 0
        noel = 7
        ad.save("problem.mp3")
        playsound("problem.mp3")
        os.remove("problem.mp3")
```



```
while jopa >= 0:
    success, imgOriginal = cap.read()
    img = np.asarray(imgOriginal)
    img = cv.resize(img, (32, 32))
    img = preProcessing(img)
    img = img.reshape(1, 32, 32, 1)
    classIndex = int(model.predict_classes(img))
    cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
    cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
    cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
    cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)

    cv.imshow("Premade Image", imgOriginal)

    jopa = jopa + 1
    jepoy = jopa % 50

    if jepoy == 0:
        noel = noel - 1

    if jopa == 350:
        jc = int(classIndex)
        break
    if cv.waitKey(1) & 0xFF == ord('q'):
        break
cv.destroyAllWindows()
if jc == nani4:
    pcc11 = pcc11 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
    cclip.save("correctclip.mp3")
    playsound("correctclip.mp3")
    os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")

    qc = qc + 1
    cap.release()
    TY()
    pcc111 = "Your grade is {} out of 5".format(pcc11)
    kerson = gTTS(pcc111)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
```



```
global screen11
screen11 = Tk()
screen11.geometry("800x525")
int_var = tkinter.IntVar(screen11, pcc11)
lbl = Label(screen11, text="Your grade is: ", font=('Arial', 17))
lbl.pack(side=TOP, pady=20)
lbl = Label(screen11, textvariable=int_var, font=('Arial', 17))
lbl.pack(side=TOP, pady=20)
if pcc11 >= 3:
    lbl = Label(screen11, text="Congratulations! You Passed!",
font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Passed.mp3")
else:
    lbl = Label(screen11, text="Better luck next time", font=('Arial',
17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Failed1.mp3")
    lbl = Label(screen11, text="Do you want to exit?", font=('Arial', 12))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen11, text="Yes", height="4", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen11, text="No", height="4", width="10",
command=systemrestart11)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("TeacherScorePremade/scorelevel1.txt").st_size == 0:
        pscoring11()
    else:
        pscoringverify11()

def pscoring11():
    file = open('TeacherScorePremade/scorelevel1.txt', 'r+')
    file.write('{}'.format(pcc11))
    file.close()

def pscoringverify11():
    file = open('TeacherScorePremade/scorelevel1.txt', 'r')
    verify = int(file.read())
    file.close()
    if pcc11 > verify:
        file = open('StudentScorePremade/scorelevel1.txt', 'w')
        file.write('{}'.format(pcc11))
        file.close()

def premadelevel2():
    width = 640
    height = 480

    cap = cv.VideoCapture(0)
    cap.set(8, width)
    cap.set(12, height)
    screen.destroy()
    screen10.destroy()

    pickle_in = open("model_trained_38.p", "rb")
    model = pickle.load(pickle_in)
```



```
pickle_in.close()
global pcc2
a = "Enter 1st Digit"
b = "Enter 2nd Digit"
c = "Enter your answer"
qc = 1
pcc2 = 0

while qc <= 5:
    if qc == 1:
        question1()
    elif qc == 2:
        question2()
    elif qc == 3:
        question3()
    elif qc == 4:
        question4()
    elif qc == 5:
        question5()
    neldan = level2()
    nani3 = neldan.getNani3()
    nani4 = neldan.getNani4()
    TTS = neldan.getNaniTTS()
    ad = gTTS(TTS)
    wil = "Question No. {}".format(qc)
    salisi = gTTS(wil)
    salisi.save("qcounter.mp3")
    playsound("qcounter.mp3")
    os.remove("qcounter.mp3")

    if nani4 < 10:
        jc = 0
        jopa = 0
        noel = 10
        ad.save("problem.mp3")
        playsound("problem.mp3")
        os.remove("problem.mp3")
        while jopa >= 0:
            success, imgOriginal = cap.read()
            img = np.asarray(imgOriginal)
            img = cv.resize(img, (32, 32))
            img = preProcessing(img)
            img = img.reshape(1, 32, 32, 1)
            classIndex = int(model.predict_classes(img))
            cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(c), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)

            cv.imshow("Premade Image", imgOriginal)
```



```
jopa = jopa + 1
jepoy = jopa % 50

if jepoy == 0:
    noel = noel - 1

if jopa == 500:
    jc = int(classIndex)
    break
if cv.waitKey(1) & 0xFF == ord('q'):
    break
cv.destroyAllWindows()
else:
    jc = 0
    nani5 = 0
    arr = []
    maykel = 0
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while maykel <= 1:
        if maykel == 0:
            playsound("MP3/firstdigit.mp3")
        elif maykel == 1:
            playsound("MP3/seconddigit.mp3")
        jc = 0
        jopa = 0
        noel = 10
        while jopa >= 0:
            success, imgOriginal = cap.read()
            img = np.asarray(imgOriginal)
            img = cv.resize(img, (32, 32))
            img = preProcessing(img)
            img = img.reshape(1, 32, 32, 1)
            classIndex = int(model.predict_classes(img))
            cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
            if nani5 == 0:
                cv.putText(imgOriginal, str(a), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
            else:
                cv.putText(imgOriginal, str(b), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
            cv.imshow("PremadeImage", imgOriginal)
            jopa = jopa + 1
            jepoy = jopa % 50

        if jepoy == 0:
            noel = noel - 1

    if jopa == 500:
```




```
        jc = int(classIndex)
        break
    if cv.waitKey(1) & 0xFF == ord('q'):
        break
cv.destroyAllWindows()
if maykel == 0:
    arr.append(jc)
    maykel = maykel + 1
    nani5 = nani5 + 1
    hu = "{} is your first digit".format(jc)
    huhu = gTTS(hu)
    huhu.save("huhuhu.mp3")
    playsound("huhuhu.mp3")
    os.remove("huhuhu.mp3")

elif maykel == 1:
    arr.append(jc)
    maykel = maykel + 1
    nani5 = 0
    hu = "{} is your second digit".format(jc)
    huhu = gTTS(hu)
    huhu.save("huhuhu.mp3")
    playsound("huhuhu.mp3")
    os.remove("huhuhu.mp3")
    j = arr[0] * 10
    jc = j + arr[1]
if jc == nani4:
    pcc2 = pcc2 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
    cclip.save("correctclip.mp3")
    playsound("correctclip.mp3")
    os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")

    qc = qc + 1
cap.release()
TY()
pcc022 = "Your grade is {} out of 5".format(pcc2)
kerson = gTTS(pcc022)
kerson.save("finalgrade.mp3")
playsound("finalgrade.mp3")
os.remove("finalgrade.mp3")
global screen12
screen12 = Tk()
screen12.geometry("800x525")
int_var = tkinter.IntVar(screen12, pcc2)
lbl = Label(screen12, text="Your grade is: ", font=('Arial', 17))
```



```
lbl.pack(side=TOP, pady=20)
lbl = Label(screen12, textvariable=int_var, font=('Arial', 17))
lbl.pack(side=TOP, pady=20)
if pcc2 >= 3:
    lbl = Label(screen12, text="Congratulations! You Passed!",
font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Passed.mp3")
else:
    lbl = Label(screen12, text="Better luck next time", font=('Arial',
17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Failed1.mp3")
    lbl = Label(screen12, text="Do you want to exit?", font=('Arial', 10))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen12, text="Yes", height="1", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen12, text="No", height="1", width="10",
command=systemrestart22)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("StudentScorePremade/scorelevel2.txt").st_size == 0:
        pscoring2()
    else:
        pscoringverify2()

def pscoring2():
    file = open('StudentScorePremade/scorelevel2.txt', 'r+')
    file.write('{}'.format(pcc2))
    file.close()

def pscoringverify2():
    file = open('StudentScorePremade/scorelevel2.txt', 'r')
    verify = int(file.read())
    file.close()
    if pcc2 > verify:
        file = open('StudentScorePremade/scorelevel2.txt', 'w')
        file.write('{}'.format(pcc2))
        file.close()

def premadelevel122():
    width = 640
    height = 480

    cap = cv.VideoCapture(0)
    cap.set(8, width)
    cap.set(12, height)
    screen.destroy()
    screen2.destroy()

    pickle_in = open("model_trained_38.p", "rb")
    model = pickle.load(pickle_in)
    pickle_in.close()
    global pcc22
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"
```



```
qc = 1
pcc22 = 0

while qc <= 5:
    if qc == 1:
        question1()
    elif qc == 2:
        question2()
    elif qc == 3:
        question3()
    elif qc == 4:
        question4()
    elif qc == 5:
        question5()
    neldan = level2()
    nani3 = neldan.getNani3()
    nani4 = neldan.getNani4()
    TTS = neldan.getNaniTTS()
    ad = gTTS(TTS)
    wil = "Question No. {}".format(qc)
    salisi = gTTS(wil)
    salisi.save("qcounter.mp3")
    playsound("qcounter.mp3")
    os.remove("qcounter.mp3")

    if nani4 < 10:
        jc = 0
        jopa = 0
        noel = 10
        ad.save("problem.mp3")
        playsound("problem.mp3")
        os.remove("problem.mp3")
        while jopa >= 0:
            success, imgOriginal = cap.read()
            img = np.asarray(imgOriginal)
            img = cv.resize(img, (32, 32))
            img = preProcessing(img)
            img = img.reshape(1, 32, 32, 1)
            classIndex = int(model.predict_classes(img))
            cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(c), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)

            cv.imshow("Premade Image", imgOriginal)

            jopa = jopa + 1
            jepoy = jopa % 50

            if jepoy == 0:
                noel = noel - 1
```



```
        if jopa == 500:
            jc = int(classIndex)
            break
        if cv.waitKey(1) & 0xFF == ord('q'):
            break
    cv.destroyAllWindows()
else:
    jc = 0
    nani5 = 0
    arr = []
    maykel = 0
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while maykel <= 1:
        if maykel == 0:
            playsound("MP3/firstdigit.mp3")
        elif maykel == 1:
            playsound("MP3/seconddigit.mp3")
        jc = 0
        jopa = 0
        noel = 10
        while jopa >= 0:
            success, imgOriginal = cap.read()
            img = np.asarray(imgOriginal)
            img = cv.resize(img, (32, 32))
            img = preProcessing(img)
            img = img.reshape(1, 32, 32, 1)
            classIndex = int(model.predict_classes(img))
            cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
            if nani5 == 0:
                cv.putText(imgOriginal, str(a), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
            else:
                cv.putText(imgOriginal, str(b), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
            cv.imshow("PremadeImage", imgOriginal)
            jopa = jopa + 1
            jepoy = jopa % 50

        if jepoy == 0:
            noel = noel - 1

        if jopa == 500:
            jc = int(classIndex)
            break
        if cv.waitKey(1) & 0xFF == ord('q'):
            break
    cv.destroyAllWindows()
```



```
        if maykel == 0:
            arr.append(jc)
            maykel = maykel + 1
            nani5 = nani5 + 1
            hu = "{} is your first digit".format(jc)
            huhu = gTTS(hu)
            huhu.save("huhuhu.mp3")
            playsound("huhuhu.mp3")
            os.remove("huhuhu.mp3")

        elif maykel == 1:
            arr.append(jc)
            maykel = maykel + 1
            nani5 = 0
            hu = "{} is your second digit".format(jc)
            huhu = gTTS(hu)
            huhu.save("huhuhu.mp3")
            playsound("huhuhu.mp3")
            os.remove("huhuhu.mp3")
            j = arr[0] * 10
            jc = j + arr[1]

    if jc == nani4:
        pcc22 = pcc22 + 1
        niceone()
        correct = "{} is equal to {}".format(TTS, jc)
        cclip = gTTS(correct)
        cclip.save("correctclip.mp3")
        playsound("correctclip.mp3")
        os.remove("correctclip.mp3")

    else:
        nicetry()
        wrong = "{} is not equal to {}. The correct answer is {}".format(TTS, jc, nani4)
        wclip = gTTS(wrong)
        wclip.save("wrongclip.mp3")
        playsound("wrongclip.mp3")
        os.remove("wrongclip.mp3")

    qc = qc + 1
    cap.release()
    TY()
    pcc222 = "Your grade is {} out of 5".format(pcc22)
    kerson = gTTS(pcc222)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen12
    screen12 = Tk()
    screen12.geometry("800x525")
    int_var = tkinter.IntVar(screen12, pcc22)
    lbl = Label(screen12, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen12, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if pcc22 >= 3:
        lbl = Label(screen12, text="Congratulations! You Passed!",
```



```
font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen12, text="Better luck next time", font=('Arial',
17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed1.mp3")
        lbl = Label(screen12, text="Do you want to exit?", font=('Arial', 10))
        lbl.pack(side=TOP, pady=120)
        btn = Button(screen12, text="Yes", height="1", width="10",
command=systemexit)
        btn.pack(side=LEFT, padx=97, pady=10)
        btn = Button(screen12, text="No", height="1", width="10",
command=systemrestart22)
        btn.pack(side=RIGHT, padx=97, pady=10)
        if os.stat("TeacherScorePremade/scorelevel2.txt").st_size == 0:
            pscoring22()
        else:
            pscoringverify22()

def pscoring22():
    file = open('StudentScorePremade/scorelevel2.txt', 'r+')
    file.write('{}'.format(pcc22))
    file.close()

def pscoringverify22():
    file = open('StudentScorePremade/scorelevel2.txt', 'r')
    verify = int(file.read())
    file.close()
    if pcc22 > verify:
        file = open('StudentScorePremade/scorelevel2.txt', 'w')
        file.write('{}'.format(pcc22))
        file.close()

def premadelevel3():
    width = 640
    height = 480

    cap = cv.VideoCapture(0)
    cap.set(8, width)
    cap.set(12, height)
    screen.destroy()
    screen10.destroy()

    pickle_in = open("model_trained_38.p", "rb")
    model = pickle.load(pickle_in)
    pickle_in.close()
    global pcc3
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"
    qc = 1
    pcc3 = 0

    while qc <= 5:
        if qc == 1:
```



```
        question1()
    elif qc == 2:
        question2()
    elif qc == 3:
        question3()
    elif qc == 4:
        question4()
    elif qc == 5:
        question5()
neldan = level3()
nani3 = neldan.getNani3()
nani4 = neldan.getNani4()
TTS = neldan.getNaniTTS()
ad = gTTS(TTS)
wil = "Question No. {}".format(qc)
salisi = gTTS(wil)
salisi.save("qcounter.mp3")
playsound("qcounter.mp3")
os.remove("qcounter.mp3")

if nani4 < 10:
    jc = 0
    jopa = 0
    noel = 10
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while jopa >= 0:
        success, imgOriginal = cap.read()
        img = np.asarray(imgOriginal)
        img = cv.resize(img, (32, 32))
        img = preProcessing(img)
        img = img.reshape(1, 32, 32, 1)
        classIndex = int(model.predict_classes(img))
        cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(c), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)

        cv.imshow("Premade Image", imgOriginal)

        jopa = jopa + 1
        jepoy = jopa % 50

        if jepoy == 0:
            noel = noel - 1

        if jopa == 500:
            jc = int(classIndex)
            break
    if cv.waitKey(1) & 0xFF == ord('q'):
```



```
        break
    cv.destroyAllWindows()
else:
    jc = 0
    nani5 = 0
    arr = []
    maykel = 0
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while maykel <= 1:
        if maykel == 0:
            playsound("MP3/firstdigit.mp3")
        elif maykel == 1:
            playsound("MP3/seconddigit.mp3")
        jc = 0
        jopa = 0
        noel = 10
        while jopa >= 0:
            success, imgOriginal = cap.read()
            img = np.asarray(imgOriginal)
            img = cv.resize(img, (32, 32))
            img = preProcessing(img)
            img = img.reshape(1, 32, 32, 1)
            classIndex = int(model.predict_classes(img))
            cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
            cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
            if nani5 == 0:
                cv.putText(imgOriginal, str(a), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
            else:
                cv.putText(imgOriginal, str(b), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
            cv.imshow("Premade Image", imgOriginal)
            jopa = jopa + 1
            jepoy = jepoy % 50

            if jepoy == 0:
                noel = noel - 1

            if jopa == 500:
                jc = int(classIndex)
                break
            if cv.waitKey(1) & 0xFF == ord('q'):
                break
    cv.destroyAllWindows()
    if maykel == 0:
        arr.append(jc)
        maykel = maykel + 1
        nani5 = nani5 + 1
        hu = "{} is your first digit".format(jc)
```




```
        huhu = gTTS(hu)
        huhu.save("huhuhu.mp3")
        playsound("huhuhu.mp3")
        os.remove("huhuhu.mp3")

    elif maykel == 1:
        arr.append(jc)
        maykel = maykel + 1
        nani5 = 0
        hu = "{} is your second digit".format(jc)
        huhu = gTTS(hu)
        huhu.save("huhuhu.mp3")
        playsound("huhuhu.mp3")
        os.remove("huhuhu.mp3")
        j = arr[0] * 10
        jc = j + arr[1]
if jc == nani4:
    pcc3 = pcc3 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
    cclip.save("correctclip.mp3")
    playsound("correctclip.mp3")
    os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")

    qc = qc + 1
    cap.release()
    TY()
    pcc033 = "Your grade is {} out of 5".format(pcc3)
    kerson = gTTS(pcc033)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen13
    screen13 = Tk()
    screen13.geometry("800x525")
    int_var = tkinter.IntVar(screen13, pcc3)
    lbl = Label(screen13, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen13, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if pcc3 >= 3:
        lbl = Label(screen13, text="Congratulations! You Passed!",
font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen13, text="Better luck next time", font=('Arial',
```



```
17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Failed1.mp3")
    lbl = Label(screen13, text="Do you want to exit?", font=('Arial', 10))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen13, text="Yes", height="1", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen13, text="No", height="1", width="10",
command=systemrestart33)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("StudentScorePremade/scorelevel3.txt").st_size == 0:
        pscoring3()
    else:
        pscoringverify3()

def pscoring3():
    file = open('StudentScorePremade/scorelevel3.txt', 'r+')
    file.write('{}'.format(pcc3))
    file.close()

def pscoringverify3():
    file = open('StudentScorePremade/scorelevel3.txt', 'r')
    verify = int(file.read())
    file.close()
    if pcc3 > verify:
        file = open('StudentScorePremade/scorelevel3.txt', 'w')
        file.write('{}'.format(pcc3))
        file.close()

def premadelevel33():
    width = 640
    height = 480

    cap = cv.VideoCapture(0)
    cap.set(8, width)
    cap.set(12, height)
    screen.destroy()
    screen2.destroy()

    pickle_in = open("model_trained_38.p", "rb")
    model = pickle.load(pickle_in)
    pickle_in.close()
    global pcc33
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"
    qc = 1
    pcc33 = 0

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
```



```
elif qc == 4:
    question4()
elif qc == 5:
    question5()
neldan = level3()
nani3 = neldan.getNani3()
nani4 = neldan.getNani4()
TTS = neldan.getNaniTTS()
ad = gTTS(TTS)
wil = "Question No. {}".format(qc)
salisi = gTTS(wil)
salisi.save("qcounter.mp3")
playsound("qcounter.mp3")
os.remove("qcounter.mp3")

if nani4 < 10:
    jc = 0
    jopa = 0
    noel = 10
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while jopa >= 0:
        success, imgOriginal = cap.read()
        img = np.asarray(imgOriginal)
        img = cv.resize(img, (32, 32))
        img = preProcessing(img)
        img = img.reshape(1, 32, 32, 1)
        classIndex = int(model.predict_classes(img))
        cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(c), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)

        cv.imshow("Premade Image", imgOriginal)

        jopa = jopa + 1
        jepoy = jopa % 50

        if jepoy == 0:
            noel = noel - 1

        if jopa == 500:
            jc = int(classIndex)
            break
        if cv.waitKey(1) & 0xFF == ord('q'):
            break
    cv.destroyAllWindows()
else:
    jc = 0
    nani5 = 0
```



```
arr = []
maykel = 0
ad.save("problem.mp3")
playsound("problem.mp3")
os.remove("problem.mp3")
while maykel <= 1:
    if maykel == 0:
        playsound("MP3/firstdigit.mp3")
    elif maykel == 1:
        playsound("MP3/seconddigit.mp3")
    jc = 0
    jopa = 0
    noel = 10
    while jopa >= 0:
        success, imgOriginal = cap.read()
        img = np.asarray(imgOriginal)
        img = cv.resize(img, (32, 32))
        img = preProcessing(img)
        img = img.reshape(1, 32, 32, 1)
        classIndex = int(model.predict_classes(img))
        cv.putText(imgOriginal, str(classIndex), (10, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(wil), (200, 55),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(noel), (585, 60),
cv.FONT_HERSHEY_PLAIN, 3, (128, 0, 128), 5)
        cv.putText(imgOriginal, str(nani3), (200, 480),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 5)
        if nani5 == 0:
            cv.putText(imgOriginal, str(a), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        else:
            cv.putText(imgOriginal, str(b), (220, 380),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        cv.imshow("Premade Image", imgOriginal)
        jopa = jopa + 1
        jepoy = jopa % 50

        if jepoy == 0:
            noel = noel - 1

        if jopa == 500:
            jc = int(classIndex)
            break
        if cv.waitKey(1) & 0xFF == ord('q'):
            break
    cv.destroyAllWindows()
    if maykel == 0:
        arr.append(jc)
        maykel = maykel + 1
        nani5 = nani5 + 1
        hu = "{} is your first digit".format(jc)
        huhu = gTTS(hu)
        huhu.save("huhuhu.mp3")
        playsound("huhuhu.mp3")
        os.remove("huhuhu.mp3")
```



```
        elif maykel == 1:
            arr.append(jc)
            maykel = maykel + 1
            nani5 = 0
            hu = "{} is your second digit".format(jc)
            huhu = gTTS(hu)
            huhu.save("huhuhu.mp3")
            playsound("huhuhu.mp3")
            os.remove("huhuhu.mp3")
            j = arr[0] * 10
            jc = j + arr[1]

    if jc == nani4:
        pcc33 = pcc33 + 1
        niceone()
        correct = "{} is equal to {}".format(TTS, jc)
        cclip = gTTS(correct)
        cclip.save("correctclip.mp3")
        playsound("correctclip.mp3")
        os.remove("correctclip.mp3")

    else:
        nicetry()
        wrong = "{} is not equal to {}. The correct answer is
        {}".format(TTS, jc, nani4)
        wclip = gTTS(wrong)
        wclip.save("wrongclip.mp3")
        playsound("wrongclip.mp3")
        os.remove("wrongclip.mp3")

    qc = qc + 1
    cap.release()
    TY()
    pcc333 = "Your grade is {} out of 5".format(pcc33)
    kerson = gTTS(pcc333)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen13
    screen13 = Tk()
    screen13.geometry("800x525")
    int_var = tkinter.IntVar(screen13, pcc33)
    lbl = Label(screen13, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen13, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if pcc33 >= 3:
        lbl = Label(screen13, text="Congratulations! You Passed!",
        font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen13, text="Better luck next time", font=('Arial',
        17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed1.mp3")
    lbl = Label(screen13, text="Do you want to exit?", font=('Arial', 10))
    lbl.pack(side=TOP, pady=120)
```



```
btn = Button(screen13, text="Yes", height="1", width="10",
command=systemexit)
btn.pack(side=LEFT, padx=97, pady=10)
btn = Button(screen13, text="No", height="1", width="10",
command=systemrestart33)
btn.pack(side=RIGHT, padx=97, pady=10)
if os.stat("TeacherScorePremade/scorelevel3.txt").st_size == 0:
    pscoring33()
else:
    pscoringverify33()

def pscoring33():
    file = open('TeacherScorePremade/scorelevel3.txt', 'r+')
    file.write('{}'.format(pcc33))
    file.close()

def pscoringverify33():
    file = open('TeacherScorePremade/scorelevel3.txt', 'r')
    verify = int(file.read())
    file.close()
    if pcc33 > verify:
        file = open('TeacherScorePremade/scorelevel3.txt', 'w')
        file.write('{}'.format(pcc33))
        file.close()

def handtrackinglevel1():
    capture = cv.VideoCapture(0)
    hand_detector = handLandmarkDetector()
    screen.destroy()
    screen1.destroy()
    global cc1
    qc = 1
    cc1 = 0
    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
            question5()
    neldan = level1()
    nani3 = neldan.getNani3()
    nani4 = neldan.getNani4()
    TTS = neldan.getNaniTTS()
    ad = gTTS(TTS)
    wil = "Question No. {}".format(qc)
    salisi = gTTS(wil)
    salisi.save("qcounter.mp3")
    playsound("qcounter.mp3")
    os.remove("qcounter.mp3")
    jc = 0
    jopa = 0
    noel = 7
```



```
ad.save("problem.mp3")
playsound("problem.mp3")
os.remove("problem.mp3")
while jopa >= 0:
    ret, image = capture.read()
    image = cv.flip(image, 1)
    image = cv.resize(image, (1080, 720))
    landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
    fingers = hand_detector.count_up_fingers(landmarks)
    fingers_up = int(fingers[0].count(1)) + int(fingers[1].count(1))
    cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
    cv.putText(image, str(wil), (280, 90), cv.FONT_HERSHEY_PLAIN, 4,
(128, 0, 128), 4)
    cv.putText(image, str(noel), (1000, 90), cv.FONT_HERSHEY_PLAIN,
6, (128, 0, 128), 4)
    cv.putText(image, str(nani3), (350, 720), cv.FONT_HERSHEY_PLAIN,
8, (128, 0, 128), 4)
    cv.imshow("Hand Detection", image)

    jopa = jopa + 1
    jepoy = jopa % 40
    if jepoy == 0:
        noel = noel - 1
    if jopa == 280:
        jc = int(fingers_up)
        break
    if cv.waitKey(1) == 27:
        break
cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

if jc == nani4:
    cc1 = cc1 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
    cclip.save("correctclip.mp3")
    playsound("correctclip.mp3")
    os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")
```



```
    qc = qc + 1
    TY()
    cc011 = "Your grade is {} out of 5".format(cc1)
    kerson = gTTS(cc011)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen7
    screen7 = Tk()
    screen7.geometry("800x525")
    int_var = tkinter.IntVar(screen7, cc1)
    lbl = Label(screen7, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen7, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if cc1 >= 3:
        lbl = Label(screen7, text="Congratulations! You Passed!",
font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen7, text="Better luck next time", font=('Arial',
17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed1.mp3")
    lbl = Label(screen7, text="Do you want to exit?", font=('Arial', 12))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen7, text="Yes", height="4", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen7, text="No", height="4", width="10",
command=systemrestart1)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("StudentScore\\scorelevel1.txt").st_size == 0:
        scoring1()
    else:
        scoringverify1()

def handtrackinglevel11():
    capture = cv.VideoCapture(0)
    hand_detector = handLandmarkDetector()
    screen.destroy()
    screen2.destroy()
    global cc11
    qc = 1
    cc11 = 0

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
```




```
question5()
neldan = level1()
nani3 = neldan.getNani3()
nani4 = neldan.getNani4()
TTS = neldan.getNaniTTS()
ad = gTTS(TTS)
wil = "Question No. {}".format(qc)
salisi = gTTS(wil)
salisi.save("qcounter.mp3")
playsound("qcounter.mp3")
os.remove("qcounter.mp3")
jc = 0
jopa = 0
noel = 7
ad.save("problem.mp3")
playsound("problem.mp3")
os.remove("problem.mp3")
while jopa >= 0:
    ret, image = capture.read()
    image = cv.flip(image, 1)
    image = cv.resize(image, (1080, 720))
    landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
    fingers = hand_detector.count_up_fingers(landmarks)
    fingers_up = int(fingers[0].count(1)) + int(fingers[1].count(1))
    cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
    cv.putText(image, str(wil), (280, 90), cv.FONT_HERSHEY_PLAIN, 4,
(128, 0, 128), 4)
    cv.putText(image, str(noel), (1000, 90), cv.FONT_HERSHEY_PLAIN,
6, (128, 0, 128), 4)
    cv.putText(image, str(nani3), (350, 720), cv.FONT_HERSHEY_PLAIN,
8, (128, 0, 128), 4)
    cv.imshow("Hand Detection", image)

    jopa = jopa + 1
    jepoy = jopa % 40
    if jepoy == 0:
        noel = noel - 1
    if jopa == 280:
        jc = int(fingers_up)
        break
    if cv.waitKey(1) == 27:
        break
cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

if jc == nani4:
    cc11 = cc11 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
```



```
        cclip.save("correctclip.mp3")
        playsound("correctclip.mp3")
        os.remove("correctclip.mp3")

    else:
        nicetry()
        wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
        wclip = gTTS(wrong)
        wclip.save("wrongclip.mp3")
        playsound("wrongclip.mp3")
        os.remove("wrongclip.mp3")

    qc = qc + 1
    TY()
    cc111 = "Your grade is {} out of 5".format(cc11)
    kerson = gTTS(cc111)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen7
    screen7 = Tk()
    screen7.geometry("800x525")
    int_var = tkinter.IntVar(screen7, cc11)
    lbl = Label(screen7, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen7, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)

    if cc11 >= 3:
        lbl = Label(screen7, text="Congratulations! You Passed!",
font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen7, text="Better luck next time", font=('Arial',
17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed1.mp3")
        lbl = Label(screen7, text="Do you want to exit?", font=('Arial', 12))
        lbl.pack(side=TOP, pady=120)
        btn = Button(screen7, text="Yes", height="4", width="10",
command=systemexit)
        btn.pack(side=LEFT, padx=97, pady=10)
        btn = Button(screen7, text="No", height="4", width="10",
command=systemrestart1)
        btn.pack(side=RIGHT, padx=97, pady=10)
        if os.stat("TeacherScore\scorelevel1.txt").st_size == 0:
            scoring11()
        else:
            scoringverify11()

def scoring1():
    file = open('StudentScore\scorelevel1.txt', 'r+')
    file.write('{}'.format(cc1))
    file.close()
```



```
def scoringverify1():
    file = open('StudentScore\scorelevel1.txt', 'r')
    verify = int(file.read())
    file.close()
    if cc1 > verify:
        file = open('StudentScore\scorelevel1.txt', 'w')
        file.write('{}'.format(cc1))
        file.close()

def scoring11():
    file = open('TeacherScore\scorelevel1.txt', 'r+')
    file.write('{}'.format(cc1))
    file.close()

def scoringverify11():
    file = open('TeacherScore\scorelevel1.txt', 'r')
    verify = int(file.read())
    file.close()
    if cc11 > verify:
        file = open('TeacherScore\scorelevel1.txt', 'w')
        file.write('{}'.format(cc11))
        file.close()

def handtrackinglevel2():
    capture = cv.VideoCapture(0)
    hand_detector = handLandmarkDetector()
    screen.destroy()
    screen1.destroy()
    global cc2
    qc = 1
    cc2 = 0
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
            question5()
        neldan = level2()
        nani3 = neldan.getNani3()
        nani4 = neldan.getNani4()
        TTS = neldan.getNaniTTS()
        ad = gTTS(TTS)
        wil = "Question No. {}".format(qc)
        salisi = gTTS(wil)
        salisi.save("qcounter.mp3")
        playsound("qcounter.mp3")
        os.remove("qcounter.mp3")
```



```
wil = "Question No. {}".format(qc)

if nani4 <= 10:
    jc = 0
    jopa = 0
    noel = 7
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while jopa >= 0:
        ret, image = capture.read()
        image = cv.flip(image, 1)
        image = cv.resize(image, (1080, 720))
        landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
        fingers = hand_detector.count_up_fingers(landmarks)
        fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
        cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(wil), (280, 90), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
        cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
        cv.putText(image, str(c), (350, 600), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
        cv.imshow("Hand Detection", image)

        jopa = jopa + 1
        jepoy = jopa % 40
        if jepoy == 0:
            noel = noel - 1
        if jopa == 280:
            jc = int(fingers_up)
            break
        if cv.waitKey(1) == 27:
            break
    cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

else:
    heart = 1
    jc = 0
    nani5 = 0
    arr = []
    maykel = 0
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
```



```
while maykel <= 1:
    if maykel == 0:
        playsound("MP3/firstdigit.mp3")
    elif maykel == 1:
        playsound("MP3/seconddigit.mp3")
    jc = 0
    jopa = 0
    noel = 7
    while jopa >= 0:
        ret, image = capture.read()
        image = cv.flip(image, 1)
        image = cv.resize(image, (1080, 720))
        landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
        fingers = hand_detector.count_up_fingers(landmarks)
        fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
        cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(wil), (280, 90),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 4)
        cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
        if nani5 == 0:
            cv.putText(image, str(a), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        else:
            cv.putText(image, str(b), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        cv.imshow("Hand Detection", image)
        jopa = jopa + 1
        jepoy = jopa % 40
        if jepoy == 0:
            noel = noel - 1
        if jopa == 280:
            jc = int(fingers_up)
            break
        if cv.waitKey(1) == 27:
            break
    cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

if maykel == 0:
    arr.append(jc)
    maykel = maykel + 1
    nani5 = nani5 + 1
    hu = "{} is your first digit".format(jc)
    huhu = gTTS(hu)
    huhu.save("huhuhu.mp3")
    playsound("huhuhu.mp3")
```



```
os.remove("huhuhu.mp3")

elif maykel == 1:
    arr.append(jc)
    maykel = maykel + 1
    nani5 = 0
    hu = "{} is your second digit".format(jc)
    huhu = gTTS(hu)
    huhu.save("huhuhu.mp3")
    playsound("huhuhu.mp3")
    os.remove("huhuhu.mp3")
    j = arr[0] * 10
    jc = j + arr[1]

if jc == nani4:
    cc2 = cc2 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
    cclip.save("correctclip.mp3")
    playsound("correctclip.mp3")
    os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")

qc = qc + 1
TY()
cc022 = "Your grade is {} out of 5".format(cc2)
kerson = gTTS(cc022)
kerson.save("finalgrade.mp3")
playsound("finalgrade.mp3")
os.remove("finalgrade.mp3")
global screen8
screen8 = Tk()
screen8.geometry("800x525")
int_var = tkinter.IntVar(screen8, cc2)
lbl = Label(screen8, text="Your grade is: ", font=('Arial', 17))
lbl.pack(side=TOP, pady=20)
lbl = Label(screen8, textvariable=int_var, font=('Arial', 17))
lbl.pack(side=TOP, pady=20)
if cc2 >= 3:
    lbl = Label(screen8, text="Congratulations! You Passed!",
font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Passed.mp3")
else:
    lbl = Label(screen8, text="Better luck next time", font=('Arial',
17))
```



```
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Failed1.mp3")
    lbl = Label(screen8, text="Do you want to exit?", font=('Arial', 10))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen8, text="Yes", height="1", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen8, text="No", height="1", width="10",
command=systemrestart2)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("StudentScore\scorelevel2.txt").st_size == 0:
        scoring2()
    else:
        scoringverify2()

def handtrackinglevel22():
    capture = cv.VideoCapture(0)
    hand_detector = handLandmarkDetector()
    screen.destroy()
    screen2.destroy()
    global cc22
    qc = 1
    cc22 = 0
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
            question5()
        neldan = level2()
        nani3 = neldan.getNani3()
        nani4 = neldan.getNani4()
        TTS = neldan.getNaniTTS()
        ad = gTTS(TTS)
        wil = "Question No. {}".format(qc)
        salisi = gTTS(wil)
        salisi.save("qcounter.mp3")
        playsound("qcounter.mp3")
        os.remove("qcounter.mp3")

        wil = "Question No. {}".format(qc)

        if nani4 <= 10:
            jc = 0
            jopa = 0
            noel = 7
            ad.save("problem.mp3")
            playsound("problem.mp3")
```



```
os.remove("problem.mp3")
while jopa >= 0:
    ret, image = capture.read()
    image = cv.flip(image, 1)
    image = cv.resize(image, (1080, 720))
    landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
    fingers = hand_detector.count_up_fingers(landmarks)
    fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
    cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
    cv.putText(image, str(wil), (280, 90), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
    cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
    cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
    cv.putText(image, str(c), (350, 600), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
    cv.imshow("Hand Detection", image)

    jopa = jopa + 1
    jepoy = jopa % 40
    if jepoy == 0:
        noel = noel - 1
    if jopa == 280:
        jc = int(fingers_up)
        break
    if cv.waitKey(1) == 27:
        break
cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

else:
    jc = 0
    nani5 = 0
    arr = []
    maykel = 0
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while maykel <= 1:
        if maykel == 0:
            playsound("MP3/firstdigit.mp3")
        elif maykel == 1:
            playsound("MP3/seconddigit.mp3")
    jc = 0
    jopa = 0
    noel = 10
    while jopa >= 0:
        ret, image = capture.read()
```




```
        image = cv.flip(image, 1)
        image = cv.resize(image, (1080, 720))
        landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
        fingers = hand_detector.count_up_fingers(landmarks)
        fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
        cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(wil), (280, 90),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 4)
        cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
        if nani5 == 0:
            cv.putText(image, str(a), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        else:
            cv.putText(image, str(b), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        cv.imshow("Hand Detection", image)
        jopa = jopa + 1
        jepoy = jopa % 40
        if jepoy == 0:
            noel = noel - 1
        if jopa == 280:
            jc = int(fingers_up)
            break
        if cv.waitKey(1) == 27:
            break
cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

if maykel == 0:
    arr.append(jc)
    maykel = maykel + 1
    nani5 = nani5 + 1
    hu = "{} is your first digit".format(jc)
    huhu = gTTS(hu)
    huhu.save("huhuhu.mp3")
    playsound("huhuhu.mp3")
    os.remove("huhuhu.mp3")

elif maykel == 1:
    arr.append(jc)
    maykel = maykel + 1
    nani5 = 0
    hu = "{} is your second digit".format(jc)
    huhu = gTTS(hu)
    huhu.save("huhuhu.mp3")
    playsound("huhuhu.mp3")
```



```
        os.remove("huhuhu.mp3")
        j = arr[0] * 10
        jc = j + arr[1]

    if jc == nani4:
        cc22 = cc22 + 1
        niceone()
        correct = "{} is equal to {}".format(TTS, jc)
        cclip = gTTS(correct)
        cclip.save("correctclip.mp3")
        playsound("correctclip.mp3")
        os.remove("correctclip.mp3")

    else:
        nicetry()
        wrong = "{} is not equal to {}. The correct answer is {}".format(TTS, jc, nani4)
        wclip = gTTS(wrong)
        wclip.save("wrongclip.mp3")
        playsound("wrongclip.mp3")
        os.remove("wrongclip.mp3")

    qc = qc + 1
    TY()
    cc222 = "Your grade is {} out of 5".format(cc22)
    kerson = gTTS(cc222)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen8
    screen8 = Tk()
    screen8.geometry("800x525")
    int_var = tkinter.IntVar(screen8, cc22)
    lbl = Label(screen8, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen8, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if cc22 >= 3:
        lbl = Label(screen8, text="Congratulations! You Passed!",
font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passed.mp3")
    else:
        lbl = Label(screen8, text="Better luck next time", font=('Arial',
17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed1.mp3")
    lbl = Label(screen8, text="Do you want to exit?", font=('Arial', 10))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen8, text="Yes", height="1", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen8, text="No", height="1", width="10",
command=systemrestart2)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("TeacherScore\\scorelevel2.txt").st_size == 0:
        scoring22()
```



```
else:
    scoringverify2()

def scoring2():
    file = open('StudentScore\scorelevel2.txt', 'r+')
    file.write('{}'.format(cc2))
    file.close()

def scoringverify2():
    file = open('StudentScore\scorelevel2.txt', 'r')
    verify = int(file.read())
    file.close()
    if cc2 > verify:
        file = open('StudentScore\scorelevel2.txt', 'w')
        file.write('{}'.format(cc2))
        file.close()

def scoring22():
    file = open('TeacherScore\scorelevel2.txt', 'r+')
    file.write('{}'.format(cc22))
    file.close()

def scoringverify22():
    file = open('TeacherScore\scorelevel2.txt', 'r')
    verify = int(file.read())
    file.close()
    if cc22 > verify:
        file = open('TeacherScore\scorelevel2.txt', 'w')
        file.write('{}'.format(cc22))
        file.close()

def handtrackinglevel3():
    capture = cv.VideoCapture(0)
    hand_detector = handLandmarkDetector()
    screen.destroy()
    screen1.destroy()
    global cc3
    qc = 1
    cc3 = 0
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
            question5()
    neldan = level3()
    nani3 = neldan.getNani3()
    nani4 = neldan.getNani4()
```



```
TTS = neldan.getNaniTTS()
ad = gTTS(TTS)
wil = "Question No. {}".format(qc)
salisi = gTTS(wil)
salisi.save("qcounter.mp3")
playsound("qcounter.mp3")
os.remove("qcounter.mp3")

wil = "Question No. {}".format(qc)

if nani4 <= 10:
    jc = 0
    jopa = 0
    noel = 7
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while jopa >= 0:
        ret, image = capture.read()
        image = cv.flip(image, 1)
        image = cv.resize(image, (1080, 720))
        landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
        fingers = hand_detector.count_up_fingers(landmarks)
        fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
        cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(wil), (280, 90), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
        cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
        cv.putText(image, str(c), (350, 600), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
        cv.imshow("Hand Detection", image)

        jopa = jopa + 1
        jepoy = jopa % 40
        if jepoy == 0:
            noel = noel - 1
        if jopa == 280:
            jc = int(fingers_up)
            break
        if cv.waitKey(1) == 27:
            break
    cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

else:
    jc = 0
```



```
nani5 = 0
arr = []
maykel = 0
ad.save("problem.mp3")
playsound("problem.mp3")
os.remove("problem.mp3")
while maykel <= 1:
    jc = 0
    jopa = 0
    noel = 7
    if maykel == 0:
        playsound("MP3/firstdigit.mp3")
    elif maykel == 1:
        playsound("MP3/seconddigit.mp3")
    while jopa >= 0:
        ret, image = capture.read()
        image = cv.flip(image, 1)
        image = cv.resize(image, (1080, 720))
        landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
        fingers = hand_detector.count_up_fingers(landmarks)
        fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
        cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(wil), (280, 90),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 4)
        cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
        if nani5 == 0:
            cv.putText(image, str(a), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        else:
            cv.putText(image, str(b), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        cv.imshow("Hand Detection", image)
        jopa = jopa + 1
        jepoy = jopa % 40
        if jepoy == 0:
            noel = noel - 1
        if jopa == 280:
            jc = int(fingers_up)
            break
        if cv.waitKey(1) == 27:
            break
    cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

if maykel == 0:
    arr.append(jc)
```



```
        maykel = maykel + 1
        nani5 = nani5 + 1
        hu = "{} is your first digit".format(jc)
        huhu = gTTS(hu)
        huhu.save("huhuhu.mp3")
        playsound("huhuhu.mp3")
        os.remove("huhuhu.mp3")

    elif maykel == 1:
        arr.append(jc)
        maykel = maykel + 1
        nani5 = 0
        hu = "{} is your second digit".format(jc)
        huhu = gTTS(hu)
        huhu.save("huhuhu.mp3")
        playsound("huhuhu.mp3")
        os.remove("huhuhu.mp3")
        j = arr[0] * 10
        jc = j + arr[1]

    if jc == nani4:
        cc3 = cc3 + 1
        niceone()
        correct = "{} is equal to {}".format(TTS, jc)
        cclip = gTTS(correct)
        cclip.save("correctclip.mp3")
        playsound("correctclip.mp3")
        os.remove("correctclip.mp3")

    else:
        nicetry()
        wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
        wclip = gTTS(wrong)
        wclip.save("wrongclip.mp3")
        playsound("wrongclip.mp3")
        os.remove("wrongclip.mp3")

    qc = qc + 1
    TY()
    cc033 = "Your grade is {} out of 5".format(cc3)
    kerson = gTTS(cc033)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen9
    screen9 = Tk()
    screen9.geometry("800x525")
    int_var = tkinter.IntVar(screen9, cc3)
    lbl = Label(screen9, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen9, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if cc3 >= 3:
        lbl = Label(screen9, text="Congratulations! You Passed!",
font=('Arial', 17))
```



```
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Passedall.mp3")
else:
    lbl = Label(screen9, text="Better luck next time", font=('Arial',
17))
    lbl.pack(side=TOP, pady=20)
    playsound("MP3/Failed.mp3")
    lbl = Label(screen9, text="Do you want to exit?", font=('Arial', 10))
    lbl.pack(side=TOP, pady=120)
    btn = Button(screen9, text="Yes", height="1", width="10",
command=systemexit)
    btn.pack(side=LEFT, padx=97, pady=10)
    btn = Button(screen9, text="No", height="1", width="10",
command=systemrestart3)
    btn.pack(side=RIGHT, padx=97, pady=10)
    if os.stat("StudentScore\scorelevel3.txt").st_size == 0:
        scoring3()
    else:
        scoringverify3()

def handtrackinglevel33():
    capture = cv.VideoCapture(0)
    hand_detector = handLandmarkDetector()
    screen.destroy()
    screen2.destroy()
    global cc33
    qc = 1
    cc33 = 0
    a = "Enter 1st Digit"
    b = "Enter 2nd Digit"
    c = "Enter your answer"

    while qc <= 5:
        if qc == 1:
            question1()
        elif qc == 2:
            question2()
        elif qc == 3:
            question3()
        elif qc == 4:
            question4()
        elif qc == 5:
            question5()
        neldan = level3()
        nani3 = neldan.getNani3()
        nani4 = neldan.getNani4()
        TTS = neldan.getNaniTTS()
        ad = gTTS(TTS)
        wil = "Question No. {}".format(qc)
        salisi = gTTS(wil)
        salisi.save("qcounter.mp3")
        playsound("qcounter.mp3")
        os.remove("qcounter.mp3")

        wil = "Question No. {}".format(qc)

        if nani4 <= 10:
```



```
jc = 0
jopa = 0
noel = 7
ad.save("problem.mp3")
playsound("problem.mp3")
os.remove("problem.mp3")
while jopa >= 0:
    ret, image = capture.read()
    image = cv.flip(image, 1)
    image = cv.resize(image, (1080, 720))
    landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
    fingers = hand_detector.count_up_fingers(landmarks)
    fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
    cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
    cv.putText(image, str(wil), (280, 90), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
    cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
    cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
    cv.putText(image, str(c), (350, 600), cv.FONT_HERSHEY_PLAIN,
4, (128, 0, 128), 4)
    cv.imshow("Hand Detection", image)

    jopa = jopa + 1
    jepoy = jopa % 40
    if jepoy == 0:
        noel = noel - 1
    if jopa == 280:
        jc = int(fingers_up)
        break
    if cv.waitKey(1) == 27:
        break
cv.destroyAllWindows()

sr = hand_detector.getsirickyright()
sl = hand_detector.getsirickyleft()

if sr == 0 or sl == 0:
    playsound("MP3/INVALIDGESTURE.mp3")

else:
    jc = 0
    nani5 = 0
    arr = []
    maykel = 0
    ad.save("problem.mp3")
    playsound("problem.mp3")
    os.remove("problem.mp3")
    while maykel <= 1:
        jc = 0
        jopa = 0
        noel = 7
        if maykel == 0:
```




```
        playsound("MP3/firstdigit.mp3")
    elif maykel == 1:
        playsound("MP3/seconddigit.mp3")
    while jopa >= 0:
        ret, image = capture.read()
        image = cv.flip(image, 1)
        image = cv.resize(image, (1080, 720))
        landmarks, image = hand_detector.detect_landmarks(image,
draw_default_style=False)
        fingers = hand_detector.count_up_fingers(landmarks)
        fingers_up = int(fingers[0].count(1)) +
int(fingers[1].count(1))
        cv.putText(image, str(fingers_up), (10, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(wil), (280, 90),
cv.FONT_HERSHEY_PLAIN, 4, (128, 0, 128), 4)
        cv.putText(image, str(noel), (1000, 90),
cv.FONT_HERSHEY_PLAIN, 6, (128, 0, 128), 4)
        cv.putText(image, str(nani3), (350, 720),
cv.FONT_HERSHEY_PLAIN, 8, (128, 0, 128), 4)
        if nani5 == 0:
            cv.putText(image, str(a), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        else:
            cv.putText(image, str(b), (350, 600),
cv.FONT_HERSHEY_PLAIN, 2, (128, 0, 128), 2)
        cv.imshow("Hand Detection", image)
        jopa = jopa + 1
        jepoy = jopa % 40
        if jepoy == 0:
            noel = noel - 1
        if jopa == 280:
            jc = int(fingers_up)
            break
        if cv.waitKey(1) == 27:
            break
    cv.destroyAllWindows()

    sr = hand_detector.getsirickyright()
    sl = hand_detector.getsirickyleft()

    if sr == 0 or sl == 0:
        playsound("MP3/INVALIDGESTURE.mp3")

    if maykel == 0:
        arr.append(jc)
        maykel = maykel + 1
        nani5 = nani5 + 1
        hu = "{} is your first digit".format(jc)
        huhu = gTTS(hu)
        huhu.save("huhuhu.mp3")
        playsound("huhuhu.mp3")
        os.remove("huhuhu.mp3")

    elif maykel == 1:
        arr.append(jc)
        maykel = maykel + 1
```



```
nani5 = 0
hu = "{} is your second digit".format(jc)
huhu = gTTS(hu)
huhu.save("huhuhu.mp3")
playsound("huhuhu.mp3")
os.remove("huhuhu.mp3")
j = arr[0] * 10
jc = j + arr[1]

if jc == nani4:
    cc33 = cc33 + 1
    niceone()
    correct = "{} is equal to {}".format(TTS, jc)
    cclip = gTTS(correct)
    cclip.save("correctclip.mp3")
    playsound("correctclip.mp3")
    os.remove("correctclip.mp3")

else:
    nicetry()
    wrong = "{} is not equal to {}. The correct answer is
{}".format(TTS, jc, nani4)
    wclip = gTTS(wrong)
    wclip.save("wrongclip.mp3")
    playsound("wrongclip.mp3")
    os.remove("wrongclip.mp3")

    qc = qc + 1
    TY()
    cc333 = "Your grade is {} out of 5".format(cc33)
    kerson = gTTS(cc333)
    kerson.save("finalgrade.mp3")
    playsound("finalgrade.mp3")
    os.remove("finalgrade.mp3")
    global screen9
    screen9 = Tk()
    screen9.geometry("800x525")
    int_var = tkinter.IntVar(screen9, cc33)
    lbl = Label(screen9, text="Your grade is: ", font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    lbl = Label(screen9, textvariable=int_var, font=('Arial', 17))
    lbl.pack(side=TOP, pady=20)
    if cc33 >= 3:
        lbl = Label(screen9, text="Congratulations! You Passed!",
font=('Arial', 17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Passedall.mp3")
    else:
        lbl = Label(screen9, text="Better luck next time", font=('Arial',
17))
        lbl.pack(side=TOP, pady=20)
        playsound("MP3/Failed.mp3")
        lbl = Label(screen9, text="Do you want to exit?", font=('Arial', 10))
        lbl.pack(side=TOP, pady=120)
        btn = Button(screen9, text="Yes", height="1", width="10",
command=systemexit)
```



```
btn.pack(side=LEFT, padx=97, pady=10)
btn = Button(screen9, text="No", height="1", width="10",
command=systemrestart3)
btn.pack(side=RIGHT, padx=97, pady=10)
if os.stat("TeacherScore\scorelevel3.txt").st_size == 0:
    scoring33()
else:
    scoringverify33()

def scoring3():
    file = open('StudentScore\scorelevel3.txt', 'r+')
    file.write('{}'.format(cc3))
    file.close()

def scoringverify3():
    file = open('StudentScore\scorelevel3.txt', 'r')
    verify = int(file.read())
    file.close()
    if cc3 > verify:
        file = open('StudentScore\scorelevel3.txt', 'w')
        file.write('{}'.format(cc3))
        file.close()

def scoring33():
    file = open('TeacherScore/scorelevel3.txt', 'r+')
    file.write('{}'.format(cc33))
    file.close()

def scoringverify33():
    file = open('TeacherScore/scorelevel3.txt', 'r')
    verify = int(file.read())
    file.close()
    if cc33 > verify:
        file = open('TeacherScore/scorelevel3.txt', 'w')
        file.write('{}'.format(cc33))
        file.close()

def systemrestart1():
    screen7.destroy()
    os.system('python "Login.py"')

def systemrestart2():
    screen8.destroy()
    os.system('python "Login.py"')

def systemrestart3():
    screen9.destroy()
    os.system('python "Login.py"')

def systemrestart11():
    screen11.destroy()
    os.system('python "Login.py"')

def systemrestart22():
    screen12.destroy()
    os.system('python "Login.py"')
```



```
def systemrestart33():
    screen13.destroy()
    os.system('python "Login.py"')

def systemexit():
    playsound("MP3/Goodbye.mp3")
    sys.exit()

def preProcessing(img):
    img = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
    img = cv.equalizeHist(img)
    img = img / 255
    return img

def main_screen():
    global screen
    screen = Tk()
    screen.geometry("1200x600")
    screen.title("THESIS")
    Label(text="Microprocessor-Based Interactive Mathematics", bg="grey",
width="300", height="2",
font=("Arial", 13)).pack()
    Label(text="Learning Tool using Real-Time Computer-Vision", bg="grey",
width="300", height="2",
font=("Arial", 13)).pack()
    Label(text="").pack()
    btn = Button(text="TEACHER", height="2", width="30", command=teacher)
    btn.pack(side=TOP, pady=60)
    btn = Button(text="STUDENT", height="2", width="30",
command=studentmodes)
    btn.pack(side=TOP, pady=60)
    playsound("MP3/Welcome.mp3")
    screen.mainloop()

print(keras.__version__)
main_screen()
```



APPENDIX I

Proponents' Profile



JOHN PATRICK B. GALVEZ

BS in Computer Engineering

Objective

To acquire an Intern position in the field of Software Engineering where I can apply my skills and abilities that I learned such as computer hardware and software understanding, programming, and critical thinking to solve software problems. To strengthen my knowledge about computer systems and improve communication skills with well-established company to provide support on achieving the company's goal.

Accomplished Projects

- Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision: A device that provide basic arithmetic problems and uses hand tracking and premade numbers | **Co-Founder, Programmer**
- Get the skills to start a career in network engineering, system administration, cybersecurity, and DEVOPS: A seminar on FEU Computer Engineering students' possible career paths | **Organizer**
- DVD Rental System: A computer program that can be used to rent movies that uses C++ Language | **Programmer**

Seminars

- **Web Application Security** | November 13, 2021
- **Start Right** | October 22, 2021
- **Computer Science Summit 5.0** | January 21, 2021
- **Thinking Future** | October 1, 2021

Education

BS in Computer Engineering

FEU Institute of Technology
P. Paredes St., Sampaloc, Manila
Expected Graduation - 2023

Senior High School

Polytechnic University of the Philippines
Anonas, Sta. Mesa, Manila 1016
2018

Personal Information

Address : 18 Q. Mejia St., Santolan, Pasig City 1610
 Mobile : +639 16 205 3144 (Globe)
 E-mail : jopagalvez896gmail.com
 Birthday : September 11, 2000
 Father : Rolando Z. Galvez
 Mother : Melani B. Galvez

Hardware and Software Proficiency

C Language, C++
 Java
 Python
 VHDL
 Adobe Photoshop
 Microsoft PowerPoint
 Microsoft Excel
 Cisco Packet Tracer
 Matlab
 Quartus II

Awards

Top 15th Performing BSCpE Student (SY 2020 – 2021)

Certifications

MTA Certified Introduction to Programming using Python 2020
 MTA Certified Introduction to Programming using Java 2021
 Cisco Certified Introduction to Packet Tracer 2021

Organizations

Member, Computer Engineering Organization | 2018 – 2019
 Member, Computer Engineering Organization | 2021 – 2022
 Member, Institute of Computer Engineers of the Philippines (Student Edition) | 2022 - Present

Other Interests

Guitar
 Video Games
 Anime and Manga





JOHN CHRISTIAN S. GUERRERO BS in Computer Engineering

Objective

I aim to obtain an intern position related to my degree, Computer Engineering, where I can employ all the skills and knowledge that I have learned and honed throughout my college years, together with my good communication skills, critical thinking skill, computer proficiency, and leadership experience to contribute to accomplishing the company's goal.

Accomplished Projects

- *DVD Rentals System: Data Structures & Algorithms for CpE Final Project* | **Lead Designer, Lead Programmer**
- *Step Down Power Supply: Fundamentals in Electronic Circuits Laboratory Final Project* | **Hardware Designer, Group Leader**
- *Get the Skills to Start a Career: Network Engineering, System Administration, Cybersecurity and DEVOPS: A webinar about the technical skills that CpE Students can learn on some career paths* | **Organizer**
- *Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision: a mathematical learning device for grade school students that uses real time computer vision* | **Lead Programmer**

Seminars and Webinars

- *UP Cursor Computer Science Summit 5.0: University of the Philippines* | **January 3, 2021**
- *An introduction to Virtualization: FEU – Institute of Technology* | **September 18, 2021**
- *Jumping into Cloude Nine: FEU – Institute of Technology* | **October 8, 2021**
- *ARTIFICIAL INTELLIGENCE IN INDUSTRY: "OPPORTUNITIES AND CHALLENGES": FEU – Institute of Technology* | **November 12, 2021**

Education

BS in Computer Engineering

FEU Institute of Technology
P. Paredes St., Sampaloc, Manila
Expected Graduation - 2023

High School

Pateros Catholic School
B. Morcilla St. Pateros, Metro Manila
2018

Personal Information

Address : 191 – G 28th Avenue, East Rembo, Makati City
 Mobile : 09396257867
 E-mail : jcguerrero.ftw@gmail.com
 Birthday : June 24, 1999
 Father : Frankie M. Guerrero
 Mother : Bellaflor S. Guerrero

Hardware and Software Proficiency

AutoCAD
 Adobe Premiere
 Adobe Photoshop
 Arduino IDE
 Cisco – Packet Tracer
 MATLAB
 Microsoft Office
 Oracle VM
 PyCharm
 Quartus II
 Sony Vegas

Windows Server 2008: Active

Directory and DHCP Server

Programming Language

C, C++, Java, Python, VHDL

Awards

Top 8th Performing BSCPE Student (SY 2020 – 2021)

Certifications

MTA: Introduction to Programming Using Python (2020)

Cisco Certified CCNA: Introduction to Networks (2021)

Cisco Certified: Introduction to Packet Tracer (2021)

Cisco Certified CCNA: Switching, Routing, and Wireless Essentials (2022)

Organizations

Member, Computer Engineering Organization (FEU TECH) | 2018 – Present

Member, Institute of Computer Engineers of the Philippines (Student Edition) | 2022 – Present

Other Interests

Video Editing

Guitar

Photoshoot

Anime and Manga

Online Games





JON MICHAEL B. HERNANI
BS in Computer Engineering

Objective

A detail-oriented B.S in Computer Engineering student from the FEU Institute of Technology with experiences mainly in the field of Object-Oriented Programming, Linux Operating systems, and Computer Vision. Highly organized, proficient with computers, and confident among online tasks. Also experienced in handling numerous documents required for academic and organizational paperwork. Lastly, a flexible team member and a team leader.

Accomplished Projects

- *Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision: A mathematical learning device for grade school students* | **Lead Developer, Group Leader**
- *DVD Rentals System: Data Structures & Algorithms for CpE Final Project* | **Lead Developer, Group Leader**
- *Get the Skills to Start a Career: Network Engineering, System Administration, Cybersecurity and DEVOPS: A webinar about the technical skills that CpE Students can learn on some career paths* | **Head Organizer, Co – Designer**
- *Step Down Power Supply: A project that aims to apply the concepts of Fundamentals in Electronics Circuit* | **Group Leader**

Seminars and Webinars

- *An introduction to Virtualization: FEU – Institute of Technology* | **September 18, 2021**
- *Roboutika, Arduino Workshop - FEU – Institute of Technology* | **February 3, 2021**
- *Jumping into Cloude Nine: FEU – Institute of Technology* | **October 8, 2021**
- *CMLI Aguhon Leadership Summit – Ateneo De Manila University* | **September 30, 2017**
- *Global Youth Summit - Mall of Asia Arena* | **March 3, 2017**

Education

BS in Computer Engineering

FEU Institute of Technology
P. Paredes St., Sampaloc, Manila
Expected Graduation – 2023

Senior High School

FEU Diliman
Sampaguita Avenue, Mapayapa Village, Brgy. Pasong Tamo, Quezon City
2018

Personal Information

Address: : Blk 17 Lot 13 Paris St. Casimiro Deparo Caloocan
 Mobile : +63 915 222 4148 (Globe)
 E-mail : jmhernani08@gmail.com
 Birthday : June 29, 1999
 Father : Jet Eliud A. Hernani
 Mother : Maribel B. Hernani

Hardware and Software Proficiency

Microsoft Office Suite – Word, Excel, PowerPoint
Adobe Creative Suite – Premiere Pro, Lightroom, After Effects
Programming Language – C#, C++, Python, Java
 AutoCAD

Awards

Global Finalist, Optum Stratethon Season 3, 2021
Rank 16, Top Performing Student BSCpE (SY 2020-2021)
Champion, ICpEP Quiz Bowl Challenge 2021
3rd placer, National MSRP Interscholastic Online Quiz Show 2021

Certifications

Cisco Certified CCNA: Introduction to Networks 2021
 Cisco Certified Introduction to Packet Tracer 2021
 MTA: Introduction to Programming Using Java – Certified 2021
 MTA: Introduction to Programming Using Python – Certified 2020

Organizations

Director of Means & Ways, Computer Engineering Organization | 2021 - Present
Member, Institute of Computer Engineers of the Philippines | 2021 – Present
Treasurer, FEU Diliman SHS Student Council | 2017-2018

Other Interests

Playing String Instruments
 Programming
 Solving Puzzles
 Cooking





APPENDIX J

Grammarian's Profile



GEMMARIE E. BELARDO, LPT

PROFILE

I am qualified and well organized English teacher. I had 5 years teaching experience in a classroom setting, 2 years as an online English teacher to Chinese professionals such as businessmen, doctors, and managers; teaching them proper pronunciation of words, correcting grammar errors and imparting new vocabulary to use in their daily conversation. I also taught English language in a university for a year. I am an English tutor for almost 10 years; teaching diverse group of students. I also had an experience in teaching special children. I can adapt my teaching style depending on the individual differences of students. I have a great passion for teaching especially grammar. I also became an editor of a book published last 2019.

CONTACT

PHONE:
0935-174-9636

ADDRESS:
B38Lot 26 Dreamland Subdivision,
Brgy. Hagonoy, Taguig City

EMAIL:
belardogemmarie@gmail.com

Current Job: Taguig's Tele-Aral Agent

EDUCATION

SILANGAN ELEMENTARY SCHOOL (PRIMARY)

Graduated: 2001

Awards Received: Most Behave, Most Polite, Top 4 of the class

Signal Village National High School (SECONDARY)

Graduated: 2005

Consistent Top 1 from 1st-2nd year and included in Pilot section from 3rd-4th year.

Polytechnic University of the Philippines-Taguig Campus (TERTIARY)

Graduated: 2010

Ms. Tioco's Scholar; PUP resident scholar; PUP consistent Dean's lister and part of President's lister; Best in Research Paper Award 2009 and 2010; Most Creative Award; Best in Visual Aids, 2008.

Taguig City University (Graduate Studies)

Masters in Educational Management

12 units

Undergraduate

WORK EXPERIENCE

Christian Harvest Academy – Elementary Teacher

May15-2010–March 31,2014

- Prepared lesson plans to teach primary students specific subjects and skills, like English, art, science or music
- Modeled good conduct and communication skills so students can learn how to communicate with and treat others
- Graded homework and other assignments so students receive proper feedback day-by-day and on their report cards
- Communicated with parents about their child's progress
- Pay attention to students who require extra help or guidance

Taguig City University – Part time Instructor

June28,2010–April 30,2011

- Ensured that first to second year college students learn proper grammar, writing, and reading comprehension.
- Combined various English teaching methods and techniques to help students improve their results.
- Trained in literature, writing, and reading to ensure their students are learning state educational standards.
- Taught students how to communicate information in a clear manner
- Elevated students' knowledge of language conventions in order to correct errors
- Conducted intensive trainings in communication and expository writing to improve their linguistic style.



I-Coach Educational Center – Pre school Head Teacher

June 15,2015 - August 30,2015

- Appoint staff who will enrich the pupils' experience through inspirational teaching and a commitment to extracurricular activity
- Ensure that proper standards of professional performance are established and maintained
- Promote, supervise, monitor and review arrangements for the professional development of academic and pastoral staff, including performance management
- Monitor and evaluate the quality of learning and teaching in the school, and to put in place appropriate initiatives developments

E-SAY Mobile English - Office and homebased online English Teacher

August 27,2015- January 31,2016

- Prepared and delivered lessons to improve students' reading, writing, and conversing skills.
- Assigned activities to facilitate students' consolidation of learning material.
- Managed interpersonal dynamics to ensure that the class is conducive to learning.
- Tailored teaching style in accordance with students' needs.
- Initiated conversation to the students to let them express their thoughts and so, check their linguistic and communication skills.

UNITEDHEALTH Group- OPTUM (Clinical Administrator)

July 13,2016- July 2019

- Managed the referral process, processing incoming and outgoing referrals and prior authorizations, including intake, notification and census roles
- Provided appropriate triage and care coordination notification cases for non-clinical assessment/intervention
- Handled resolution and inquiries from members and or over the phone

FREELANCE GRAMMARIAN

February 2017- Present

- Proofread hundreds of theses from different universities such as Ateneo De Manila, Lyceum of the Philippines, PUP, De La Salle University, University of the East, University of Sto Tomas, Batangas State University, and so on.
- Proofread materials for spelling, grammar, punctuation, syntax, usage, consistency, and brand voice
- Reviewed and markup written work
- Ensured the accuracy of content for print.



- Reviewed text for errors in spelling, grammar, punctuation, capitalization and syntax.

SOME of the thesis titles I have proofread and reviewed for grammars:

"MOSES AS FINE PARTICLE AND GASEOUS POLLUTANT FILTER"

**A Thesis Presented to the Faculty of Civil Engineering Department
Eulogio "Amang" Rodriguez Institute of Science and Technology**

EARIST, Manila

"Evaluation of Effectiveness of Calcium Carbonate (CaCO₃) and Silica (SiO₂) as an Additive in Mitigating Landslide"

**UNIVERSITY OF THE EAST
CALOOCAN**

"Home Appliances Automation with Electric Consumption Monitoring System"

Angeles University Foundation

"Understanding Solo Female Traveler and their Tour Preferences"

LYCEUM OF THE PHILIPPINES UNIVERSITY

"PREVALENCE OF LOW BACK PAIN AMONGST CALL CENTER AGENTS IN DAPITAN CITY"

"Disaster Preparedness Among Senior High School Students in the University of Santo Tomas"

"College Readiness among Health Allied Senior High School Students from a Selected University"

"DESIGN AND IMPLEMENTATION OF MEMBERSHIP MANAGEMENT FOR DEFTAC BETTERLIVING"

San Beda College Alabang

"THE COLLECTIVE VIEWS AND LIVED EXPERIENCES OF FILIPINO PRESCHOOL LEARNERS ABOUT KINDERGARTEN EDUCATION"

University of Santo Tomas

"GLASS BLENDED CONCRETE HOLLOW BLOCKS"

Far Eastern College – Silang, Inc.

"ODBC Task Manager: A Multi-Platform Secure and Intelligent Task Management System with Online Worship Service for Open Door Baptist Church"

Angeles University Foundation



BOOK EDITOR

Book Title: Uncovering Seafarer's Mental Health

Author: Captain Jess T. Cordova

Published: November 2019

- Proofread materials for spelling, grammar, punctuation, syntax, usage, consistency, and brand voice
- Reviewed and markup written work
- Ensured the accuracy of content for print.
- Reviewed text for errors in spelling, grammar, punctuation, capitalization and syntax.
- Edited manuscripts for clarity, style, consistency, grammar, and logic
- Changed and rewrote paragraphs to maintain story flow after coordinating with author
- Edited figures, tables, and overall story structure as necessary
- Handled all related editorial tasks for print and online publishing
- Performed technical research according to the subject of writing

SKILLS

Excellent written and verbal communication skills	★★★★★
Teaching skills	★★★★★
Time management and organizational skills	★★★★★
Interpersonal skills	★★★★★
English proficiency	★★★★★

PROFESSIONAL QUALIFICATIONS:

- Licensed Professional English teacher
- Undergraduate of Master Degree of Educational Management
- "School of Tomorrow" curriculum well trained
- Book editor of Uncovering Seafarer's Mental Health by Captain Jess T. Cordova

I hereby certify that all the statement given above are true and correct.

Gemmarie E. Belardo



APPENDIX K

Grammarian's Certificate



CERTIFICATION OF GRAMMARIAN

This to certify that the undersigned has reviewed and went through all the pages of the study entitled "**Microprocessor-Based Interactive Mathematics Learning Tool using Real-Time Computer-Vision**" by **GALVEZ, JOHN PATRICK B., GUERRERO, JOHN CHRISTIAN S. and HERNANI, JON MICHAEL B.**, aligned with the structural rules that govern the composition sentences, phrases, and words in the English language.

Signed

A handwritten signature in black ink, appearing to read "Belardo".

MS. GEMMARIË E. BELARDO, LPT.
Grammarian/ Proofreader
Professional English Specialist

Date Signed: March 27, 2022