PEOPLES CO-OPERATIVE ARTS AND SCIENCE

COLLEGE, MUNNAD

(Affiliatied To Kannur University)



April 2024

MCS4pr04: PROJECT REPORT

VIRTUAL DRAW USING AI HAND GESTURE

Submitted on partial fulfillment of the requirement for the award of the degree in Master of Computer Science with specialized in Artificial Intelligence from Kannur University .

Submitted by

JISHNA M

Reg. No: C2PSAI2005

UNDER THE GUIDENCE OF

Mr./Mrs.BINDU K NAIR

DEPARTMENT OF MASTER OF COMPUTER SCIENCE 2022-2024

PEOPLES CO-OPERATIVE ARTS AND SCIENCE

COLLEGE MUNNAD

(Affiliated To Kannur University)



CERTIFICATE

This is to certify that this project report entitled “VIRTUAL DRAW USING AI HANDGESTURE” is a bona fide record on partial fulfillment for the degree of the Master of Computer Science with specialized in Artificial Intelligence to the Kannur University through PG Department of Computer Science, Peoples Co-Operative Arts and Science College, Munnad, Kasaragod done by JISHNA.M (Reg.No: C2PSAI2005) in the year 2024.

Place : Munnad

Date : Head of the department

Examiner 1:

Examinar 2:

DECLARATION

I hereby declare that the report of the project work entitled “VIRTUAL DRAW USING AI HANDGESTURE” which is being submitted to the Department of Computer science, peoples co-operative arts and Science College Munnad ,Kannur University, in partial fulfilment of the requirement for the award of the degree of Masters in Computer Science with specialization in Artificial Intelligence, is a bonafide report of the work carried out by me. The material contained in this report has not been submitted to any University or Institute for the award of any degree.

REGISTER NUMBER:

NAME:

INDEX

|  |  |  |
| --- | --- | --- |
| 1 | Abstract | 5 |
| 2 | Introduction | 6 |
| 3 | Objective | 8 |
| 4 | Background/Approach  Future scope | 9 |
|  | System Analysis | 10 |
| 5 | Existing System | 10 |
| 6 | Proposed System | 11 |
| 7 | Feasibility Study | 12 |
| 8 | Hardware configuration | 13 |
| 9 | Software use | 13 |
|  | System testing and maintenence | 14 |
| 10 | System Testing | 14 |
| 11 | White Box testing | 14 |
| 12 | Black Box testing | 16 |
| 13 | Grey Box testing | 17 |
| 14 | System Maintenence | 18 |
| 15 | System Design | 19 |
| 16 | Media Pipe | 19 |
| 17 | Python | 21 |
| 18 | pycharm | 22 |
| 19 | Open cv | 24 |
| 20 | Gesture recogontion | 26 |
| 21 | Metaverse | 32 |
| 22 | Drawing hand pose detection | 33 |
| 23 | conclusion | 40 |
| 24 | Drawback | 41 |
| 25 | Code | 42 |
| 26 | Reference | 46 |

ABSTRACT

Gesture recognition is an emerging technology field. The presentaion becomes virtual because the drawing happens in air. Web camera is used to extract the gestures of hand in order to achieve fast and stable gesture recognition in real time without any distance restrictions.The virtual draw application’s fundamental goal is to deliver an AI based tool that allows user to present anything on screen using hand movement.using this application we can draw virtually.

In the present time technologies has been very prerequisite for us. In this application help to teach students easly because we can present slides text and diagram and using hand gesture opertes slides virtually.In this we are painting by a human being is a drawing of his sentiment and the responses of the world and deliver a message or giving the statement to the entire of humans who get that part. Utilizing the OpenCV module and mediapipe, we created this application. to make this as a real time application. An application like this is created using the Python programming language, a top machine learning tool. In this we drawing and virtually operates the images by capturing hand motion and the interaction between hand and camera. Python is the language that was used for this project and it is more libraries that are extensive and the syntax is simple to utilizing and the fundamentals and need to be utilized in any language that supports OpenCv. Processes for palm detection and hand tracking have been employed to achieve the project's objective. This gap is exploited by the project, which emphasis on creating a converter for moving images to text may be used as software for clothing-mounted computers that the ability to write in the air. In order to write from above, The system will use computer vision to track the movements of the finger.

INTRODUCTION

Around 2000 BC, the practice of writing was first discovered. The cave wall was first written on by Neolithic people. Soon after, stone replaced these walls, then fabric, and we now present a written report. In a slow manner transitioning to a form that is primarily digital, based on ability and adapt the technology to exploring it in the serval ways and organizing the electronic text papers created with modern technology. using the QWERTY keyboard, These digital text documents are editable and programmable. electronic text papers created with modern technology. using the QWERTY keyboard, These digital text documents are editable and programmable.. Electronic text and keyboards are gradually displacing traditional writing with pens and paper. With the help of our technology, also known as open cv and mediapipe, it is possible to drawing in the air. in traditional way teachers using teach black bord.but we living in 21 the century,here technology move high speed.humans will create a virtual world.it is simmple and usefull appllication for teachers and students.according to the technology develoment we have to developing.Virtual AI painter using OpenCV and Mediapipe is an application which prints the motion of an object. By using the great feature to track , the person can draw on the screen by moving the hand by the human in this project in front of the webcam or camera . This is the real time webcam which is introduced by tracking the motion of the object and helps the person to make a things easy which are both fascinating and testing. OpenCV Stands for Open Source Computer Vision was launched in August 1999 at the Computer Vision and Pattern Recognition conference which is a programming language which contains library of different types of functions only for computer vision. It have almost 3,000 functions and 14 million downloads. We can explain this in a easy language or in a proper way ,it is a library which is used for processing the image properly. It is used to do all the types of operations which is exactly to the occupied images .To create a digital canvas for drawing on. To recognize by the human finger. To performing the phonological operations. To design a user interface that connects the human hand to the system. Teaching chemical calculations or organic reaction mechanisms is much easier on a whiteboard (TRUST ME!) than making fancy PPT slides. It also allows for more creativity and dialogue among the students, as they can contribute their own ideas and feedback on the board. On the other hand, PowerPoint presentation is more suitable for structured and complex teaching, where the teacher needs to follow a certain flow and present a lot of information and parameters in a clear and organised way. I use animations (e.g.: SN1 and SN2 reaction mechanisms ) which make my students remember complex topics. These also work best for KS3 science as slides remind me to what limit I can teach certain topics. PowerPoint slides also allow for more consistency and reliability in the presentation, as the teacher can prepare the slides beforehand and slides can be uploaded on learning platforms such as Teams or Google Classroom for future reference when pupils revise the topics taught.Using virtual environment it make easier than other teaching methods.

Sketching On Air is possible through our trending technology namely open cv ,python. Open cv is mainly known as an open source computer vision and machine learning software . Python is one of the high-level-general-purpose programming language. Object-oriented approach mainly to help programmers to write clear, logical code for small as well as large – scale projects.

OBJECTIVE

-> In this application help to teach students easly because we can present slides text and diagram and using hand gesture opertes slides virtually.

-> The virtual drawing application’s fundamental goals is to deliver an AI based tool that allows users to draw anything on screen using hand movements.

-> we can operate images using fingers. we can change images,zoom in and zoom out ,change the ink color everythings are done only use fingers

->It creates a space for the users to draw anything they wish using a single finger while using two fingers to make a choice in a very convinent and effortless manner.

->The primary goals of gesture recognition research applied to Human Computer Ineraction is to create systems,which can identify specific human gestures and use them to convey information or controlling devices.

->In the future scope,various hand gestures can be recognized and applied as input to the computer.The hand gesture representing numbers can also be converted into commands to perform related task in real time.

->The main purpose of using hand gestures is that it provides a more schematic way of controlling the robot and with this features,robot can be used as a wheelchair or etc.,As human hand gestures are nature,with the help of wireless communicatin,it is easier to ineract with the robot in a more-friendly way.

BACKGROUND/APPROACH

Many online learning platform such as BYJUS,Coggnitive class with help of their tutorial and some technology such as graphical animation there make their video more explanatory.In our approach we tried to make a prototype tool which would be alternative for such kind of software.Our tool would be very effective and it will enhance the online learning.

And as as it would be cost effective and it could be use by any teacher to make their teaching videos much explanatory.

Along with that we tried to make it more simple and user friendly and with minimum hardware requirement so that a person not having any prior knowledge about computer could use this.

FUTURE SCOPE /WORK

This system could be used as an alternative for teaching software used by teachers and students.If further interpreter various virtual based physical game could be made.Controlling the robot using gesture considered as one of the interesting applications in this field proposed a system controlling a robot using hand pose signs.The orders could be given to robot to execute some task,where each sign has a specific meaning and represents different function.

To guarantee this, the interface is very straightforward and easier to understand by the human. Humans are capable of drawing in accordance with their choice what they actually want to draw without any problems or any type of difference. In the future, it is really very useful for making the kids learn things and draw in schools in an interactive and appropriate way or it gives the clear vision to kids to explore it and get it easily and faster.

In the future scope,various hand gestures can be recognized and applied as input to the computer.The hand gesture representing numbers can also be converted into commands to perform related task in real time.

SYSTEM ANALYSIS

EXISTING SYSTEM

The cave wall was first written on by Neolithic people. Soon after, stone replaced these walls, then fabric, and we now present a written report. In a slow manner transitioning to a form that is primarily digital, based on ability and adapt the technology to exploring it in the serval ways and organizing the electronic text papers created with modern technology. using the QWERTY keyboard, These digital text documents are editable and programmable. electronic text papers created with modern technology. using the QWERTY keyboard, These digital text documents are editable and programmable.. Electronic text and keyboards are gradually displacing traditional writing with pens and paper. . in traditional way teachers using teach black bord.but we living in 21 the century,here technology move high speed.humans will create a virtual world.it is simmple and usefull appllication for teachers and students.according to the technology develoment we have to developing. With the help of our technology, also known as open cv and mediapipe, it is possible to paint in the air.

Teaching chemical calculations or organic reaction mechanisms is much easier on a whiteboard (TRUST ME!) than making fancy PPT slides. It also allows for more creativity and dialogue among the students, as they can contribute their own ideas and feedback on the board. On the other hand, PowerPoint presentation is more suitable for structured and complex teaching, where the teacher needs to follow a certain flow and present a lot of information and parameters in a clear and organised way. I use animations (e.g.: SN1 and SN2 reaction mechanisms ) which make my students remember complex topics. These also work best for KS3 science as slides remind me to what limit I can teach certain topics. PowerPoint slides also allow for more consistency and reliability in the presentation, as the teacher can prepare the slides beforehand and slides can be uploaded on learning platforms such as Teams or Google Classroom for future reference when pupils revise the topics taught.Using virtual environment it make easier than other teaching methods.

PROPOSED SYSTEM

Virtual AI painter using OpenCV and Mediapipe is an application which prints the motion of an object. By using the great feature to track , the person can draw on the screen by moving the hand by the human in this project in front of the webcam or camera . This is the real time webcam which is introduced by tracking the motion of the object and helps the person to make a things easy which are both fascinating and testing.

The virtual drawing and presntation application’s fundamental goal is to deliver an AI based tool it allows users to draw anything on screen using hand movements.Using this application we can draw virtually.We can draw pictures and write.This application is useful for Students,childrens,teachers.

Many online learning platform such as BYJUS,Coggnitive class with help of their tutorial and some technology such as graphical animation there make their video more explanatory.In our approach we tried to make a prototype tool which would be alternative for such kind of software.Our tool would be very effective and it will enhance the online learning. To guarantee this, the interface is very straightforward and easier to understand by the human. Humans are capable of drawing in accordance with their choice what they actually want to draw without any problems or any type of difference. In the future, it is really very useful for making the kids learn things and draw in schools in an interactive and appropriate way or it gives the clear vision to kids to explore it and get it easily and faster. Teaching chemical calculations or organic reaction mechanisms is much easier on a whiteboard (TRUST ME!)than making fancy PPT slides. It also allows for more creativity and dialogue among the students, as they can contribute their own ideas and feedback on the board. On the other hand, PowerPoint presentation is more suitable for structured and complex teaching, where the teacher needs to follow a certain flow and present a lot of information and parameters in a clear and organised way. I use animations (e.g.: SN1 and SN2 reaction mechanisms ) which make my students remember complex topics. These also work best for KS3 science as slides remind me to what limit I can teach certain topics. PowerPoint slides also allow for more consistency and reliability in the presentation, as the teacher can prepare the slides beforehand and slides can be uploaded on learning platforms such as Teams or Google Classroom for future reference when pupils revise the topics taught.Using virtual environment it make easier than other teaching methods.

FEASIBILITY STUDY

The feasibility study is major factor, which contributes to the analysis and development of the system. The decision of the system analyst whether to design a particular system or not depends on its feasibility study.

1. Operational Feasibility:-

o The introduction to this system is not going to hamper any user of the system.

o The proposed system is very flexible and user friendly

o The proposed system produces best results and gives high performance. It can be implemented easily .So this project is operationally feasible.

1. Technical Feasibility:-

o This feasibility deals with technicality of the system.

o No efficient manpower is required to handle the system.

1. Economic Feasibility:-

o Economic Feasibility deals about the economic impact faced by the organization to implement a new system.

o Economic Feasibility in this project: - The cost to conduct a full system investigation is possible.

- There is no additional manpower requirement.

- There is no additional cost involved in maintaining the proposed system.

HARDWARE CONFIGURATIONS

Processor : Intel Core i3

Processor Speed : 1GHz to 2GHz

RAM : 8GB DDR4|256GB M.2SSD

SOFTWARE CONFIGURATIONS Operating System:Windows.

Software Used

1. Python 3.10.9-amd64 Programming Language
2. PyCharm - Version : 2022.3.3 Community edition
3. MediaPipe Framework - Version :0.9.2.1.
4. OpenCV - Python library - Version : 4.7.0.68
5. OpenCV-contrib-python-version-4.7.0.72
6. Pip-version-22.3.1
7. Cvzone-version-1.5.6
8. Numpy-version-1.24.2

SYSTEM TESTING & MAINTENANCE

SYSTEM TESTING

Software Testing Fundamentals (STF) is a platform to gain (or refresh) basic knowledge in the field of Software Testing. If we are to „cliché‟ it, the site is of the testers, by the testers, and for the testers. Our goal is to build a resourceful repository of Quality Content on Quality.

The box approach: Software testing methods are traditionally divided into white- and black-box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

WHITE BOX TESTING :

White-box testing (also known as clear box testing, glass box testing, transparent box testing and structural testing, by seeing the source code) tests internal structures or workings of a program, as opposed to the functionality exposed to the end-user. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g. in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test.

Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements.

Techniques used in white-box testing include:

API testing – testing of the application using public and private APIs (application programming interfaces)

Code coverage – creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)

Fault injection methods – intentionally introducing faults to gauge the efficacy of testing strategies.

The white box testing contains various tests, which are as follows:

* Path testing
* Loop testing
* Condition testing
* Testing based on the memory perspective
* Test performance of the program

Steps for whitebox testing

* Design all test scenarios, test cases and prioritize them according to high priority number.
* This step involves the study of code at runtime to examine the resource utilization, not accessed areas of the code, time taken by various methods and operations and so on.
* In this step testing of internal subroutines takes place. Internal subroutines such as nonpublic methods, interfaces are able to handle all types of data appropriately or not.
* This step focuses on testing of control statements like loops and conditional statements to check the efficiency and accuracy for different data inputs.
* In the last step white box testing includes security testing to check all possible security loopholes by looking at how the code handles security.

BLACK BOX TESTING:

Black-box testing treats the software as a "black box", examining functionality without any knowledge of internal implementation, without seeing the source code. The testers are only aware of what the software is supposed to do, not how it does it. Black-box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, state transition tables, decision table testing, fuzz testing, model-based testing, use case testing, exploratory testing and specification-based testing.

Specification-based testing aims to test the functionality of software according to the applicable requirements. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behaviour), either "is" or "is not" the same as the expected value specified in the test case. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external descriptions of the software, including specifications, requirements, and designs to derive test cases. These tests can be functional or nonfunctional, though usually functional.

Specification-based testing may be necessary to assure correct functionality, but it is insufficient to guard against complex or high-risk situations. One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight. “Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.

GREY BOX TESTING :

Grey-box testing involves having knowledge of internal data structures and algorithms for purposes of designing tests, while executing those tests at the user, or black-box level. The tester is not required to have full access to the software's source code.[30][not in citation given] Manipulating input data and formatting output do not qualify as grey-box, because the input and output are clearly outside of the "black box" that we are calling the system under test. This distinction is particularly important when conducting integration testing between two modules of code written by two different developers, where only the interfaces are exposed for test.

However, tests that require modifying a back-end data repository such as a database or a log file does qualify as grey-box, as the user would not normally be able to change the data repository in normal production operations. Grey-box testing may also include reverse engineering to determine, for instance, boundary values or error messages.

By knowing the underlying concepts of how the software works, the tester makes betterinformed testing choices while testing the software from outside. Typically, a grey-box tester will be permitted to set up an isolated testing environment with activities such as seeding a database. The tester can observe the state of the product being tested after performing certain actions such as executing SQL statements against the database and then executing queries to ensure that the expected changes have been reflected. Grey-box testing implements intelligent test scenarios, based on limited information. This will particularly apply to data type handling, exception handling, and so on.

SYSTEM MAINTENANCE:

An integral part of software is the maintenance one, which requires an accurate maintenance plan to be prepared during the software development. It should specify how users will request modifications or report problems. The budget should include resource and cost estimates. A new decision should be addressed for the developing of every new system feature and its quality objectives. The software maintenance, which can last for 5–6 years (or even decades) after the development process, calls for an effective plan which can address the scope of software maintenance, the tailoring of the post-delivery/deployment process, the designation of who will provide maintenance, and an estimate of the life-cycle costs. The selection of proper enforcement of standards is the challenging task right from early stage of software engineering which has not got definite importance by the concerned stakeholders.

SYSTEM DESIGN

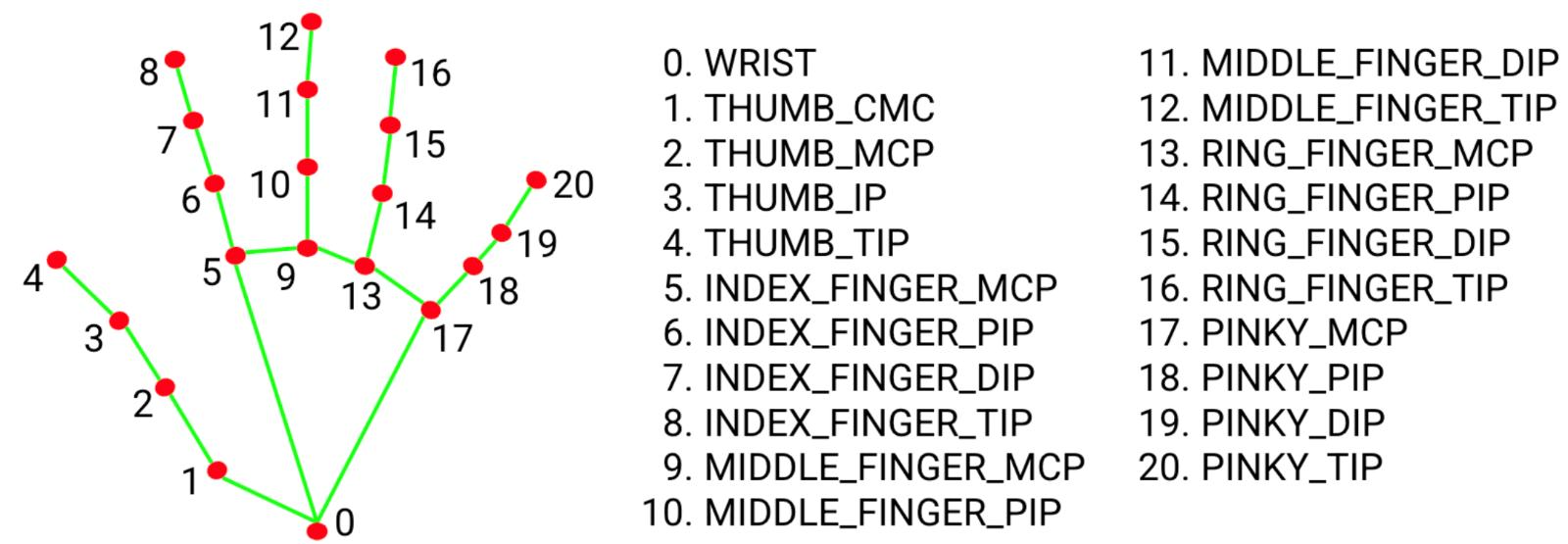
Mediapipe

MediaPipe Solutions provides a suite of libraries and tools for you to quickly apply artificial int.

MediaPipe Gesture

The MediaPipe Gesture Recognizer task lets you recognize hand gestures in real time, and provides the recognized hand gesture results along with the landmarks of the detected hands. You can use this task to recognize specific hand gestures from a user, and invoke application features that correspond to those gestures. This task operates on image data with a machine learning (ML) model, and accepts either static data or a continuous stream. The task outputs hand landmarks in image coordinates, hand landmarks in world coordinates, handedness (left/right hand), and the hand gesture categories of multiple hands. Models The Gesture Recognizer uses a model bundle with two pre-packaged model bundles: a hand landmark model elligence (AI) and machine learning (ML) techniques in your applications. You can plug these solutions into your applications immediately, customize them to your needs, and use them across multiple development platforms. MediaPipe Solutions is part of the MediaPipe open source project, so you can further customize the solutions code to meet your application needs.

bundle and a gesture classification model bundle. The landmark model detects the presence of hands and hand geometry, and the gesture recognition model recognizes gestures based on hand geometry. Hand landmark model bundle The hand landmark model bundle detects the keypoint localization of 21 hand-knuckle coordinates within the detected hand regions. The model was trained on approximately 30K real-world images, as well as several rendered synthetic hand models imposed over various backgrounds. See the definition of the 21 landmarks below:



The hand landmarker model bundle contains palm detection model and hand landmarks detection model. Palm detection model localizes the region of hands from the whole input image, and the hand landmarks detection model finds the landmarks on the cropped hand image defined by the palm detection model. Since palm detection model is much more time consuming, in Video mode or Live stream mode, Gesture Recognizer uses bounding box defined by the detected hand landmarks in the current frame to localize the region of hands in the next frame. This reduces the times Gesture Recognizer triggering palm detection model. Only when the hand landmarks model could no longer identify enough required number of hands presence, or the hand tracking fails, palm detection model is invoked to relocalize the hands.

Python

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. Python is a programming language that lets you work quickly and integrate systems more efficiently. Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

• Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

• Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects Python’s Features includes-

• Easy-to-learn − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.

• Easy-to-read − Python code is more clearly defined and visible to the eyes.

• Easy-to-maintain − Python's source code is fairly easy-to maintain.

• A broad standard library − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.

• Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

• Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

• Extendable − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customise their tools to be more efficient.

• Databases − Python provides interfaces to all major commercial databases.

• GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

• Scalable − Python provides a better structure and support for large programs than shell scripting.

PyCharm

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python language.

It is developed by the Czech company JetBrains.PyCharm Community Edition is totally free and open-source, available under the Apache 2.0 license.

PyCharm makes it easier for programmers to write various web applications in Python supporting widely used web technologies like HTML, CSS, JavaScript, TypeScript and CoffeeScript PyCharm is available in three editions: Professional, Community, and Educational (Edu).

The Community and Edu editions are open-source projects and they are free, but they have less features.

PyCharm Edu provides courses and helps you learn programming with Python.

An **IDE** consists of an editor and a compiler that we use to write and compile programs. It has a combination of features required for developing software.

The presence of an IDE makes the development process and programming much easier. It interprets what we are typing and suggests the relevant keyword to insert. We can distinguish between a class and a method as the IDE allocates different colors to them. The IDE also gives different colors for the right and the wrong keywords. If we are writing a wrong keyword, it tries to predict the keyword that we intend to write and auto completes it.

|  |
| --- |
|  |

The major reasons for using an IDE for development are given below:

* An IDE consists of a **text editor** window where we can write our programs.
* It consists of a **project editor** window where we store all the necessary files of a software project.
* We can provide various inputs and check the efficiency of our program by inspecting the output we receive on the output window.
* If any error occurs, then the IDE shows warnings and suggestions on the output window so that we can resolve it.
* An IDE has a rack of modules and packages in one place that helps add features in our software applications.
* It helps increase efficiency in creating software.

Now, do you want to know which IDE we use for Python programming and application development? The most popular and widely used IDE for Python application development and programming is**PyCharm**.

PyCharm is a hybrid platform developed by JetBrains as an IDE for Python. It is commonly used for Python application development. Some of the unicorn organizations such as Twitter, Facebook, Amazon, and Pinterest use PyCharm as their Python IDE!

### **Features of PyCharm**

Below, we have compiled some of the essential features provided by PyCharm.

**1. Intelligent Code Editor**

**2. Code Navigation….**

**etc**

opencv

OpenCV, short for Open Source Computer Vision Library, is an open-source computer vision and machine learning software library. Originally developed by Intel, it is now maintained by a community of developers under the OpenCV Foundation.

Opencv is a huge open-source library for computer vision, machine learning, and image processing. Now, it plays a major role in real-time operation which is very important in today’s systems. By using it, one can process images and videos to identify objects, faces, or even the handwriting of a human.

When it is integrated with various libraries, such as NumPy, python is capable of processing the opencv array structure for analysis. To Identify an image pattern and its various features we use vector space and perform mathematical operations on these features.

NumPy

NumPy is a Python library used for working with arrays.

It also has functions for working in domain of linear algebra, fourier transform, and matrices.

NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy stands for Numerical Python. In Python we have lists that serve the purpose of arrays, but they are slow to process.

NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.

The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.

Arrays are very frequently used in data science, where speed and resources are very important.

NumPy arrays are stored at one continuous place in memory unlike lists, so processes can access and manipulate them very efficiently.

This behavior is called locality of reference in computer science.

This is the main reason why NumPy is faster than lists. Also it is optimized to work with latest CPU architectures.

NumPy is a Python library and is written partially in Python, but most of the parts that require fast computation are written in C or C++. The source code for NumPy is located at this github repository <https://github.com/numpy/numpy>

Os module

OS module in Python provides functions for interacting with the operating system. OS comes under Python’s standard utility modules. This module provides a portable way of using operating system-dependent functionality.

The os and os.path modules include many functions to interact with the file system.

Python-OS-Module Functions

Here we will discuss some important functions of the Python os module :

Handling the Current Working Directory

Creating a Directory

Listing out Files and Directories with Python

Deleting Directory or Files using Python

Handling the Current Working Directory

Consider Current Working Directory(CWD) as a folder, where Python is operating. Whenever the files are called only by their name, Python assumes that it starts in the CWD which means that name-only reference will be successful only if the file is in the Python’s CWD

# Gesture recognition

 bookmark\_border

The MediaPipe Gesture Recognizer task lets you recognize hand gestures in real time, and provides the recognized hand gesture results and hand landmarks of the detected hands. These instructions show you how to use the Gesture Recognizer with Python applications.

You can see this task in action by viewing the [Web demo](https://mediapipe-studio.webapps.google.com/demo/gesture_recognizer) For more information about the capabilities, models, and configuration options of this task.

## Setup

This section describes key steps for setting up your development environment and code projects specifically to use Gesture Recognizer. For general information on setting up your development environment for using MediaPipe tasks, including platform version requirements, see the [Setup guide for Python](https://ai.google.dev/mediapipe/solutions/setup_python).

### Packages

The MediaPipe Gesture Recognizer task requires the mediapipe PyPI package. You can install and import these dependencies with the following:

$ python -m pip install mediapipe

## Setup

This section describes key steps for setting up your development environment and code projects specifically to use Gesture Recognizer. For general information on setting up your development environment for using MediaPipe tasks, including platform version requirements, see the [Setup guide for Python](https://ai.google.dev/mediapipe/solutions/setup_python).

### Packages

The MediaPipe Gesture Recognizer task requires the mediapipe PyPI package. You can install and import these dependencies with the following:

$ python -m pip install mediapipe

Specify the path of the model within the Model Name parameter, as shown below:

base\_options = BaseOptions(model\_asset\_path=model\_path)

## Create the task

The MediaPipe Gesture Recognizer task uses the create\_from\_options function to set up the task. The create\_from\_options function accepts values for configuration options to handle. For more information on configuration options, see [Configuration options](https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer/python#configuration_options).

The following code demonstrates how to build and configure this task.

These samples also show the variations of the task construction for images, video files, and live video streams.

import mediapipe as mp

BaseOptions = mp.tasks.BaseOptions

GestureRecognizer = mp.tasks.vision.GestureRecognizer

GestureRecognizerOptions = mp.tasks.vision.GestureRecognizerOptions

VisionRunningMode = mp.tasks.vision.RunningMode

# Create a gesture recognizer instance with the image mode:

options = GestureRecognizerOptions(

base\_options=BaseOptions(model\_asset\_path='/path/to/model.task'),

running\_mode=VisionRunningMode.IMAGE)

with GestureRecognizer.create\_from\_options(options) as recognizer:

# The detector is initialized. Use it here.

Configuration options

This task has the following configuration options for Python applications:

Option NameDescriptionValue RangeDefault Valuerunning\_modeSets the running mode for the task. There are three modes:

IMAGE: The mode for single image inputs.

VIDEO: The mode for decoded frames of a video.  
  
LIVE\_STREAM: The mode for a livestream of input data, such as from a camera. In this mode, resultListener must be called to set up a listener to receive results asynchronously.{IMAGE, VIDEO, LIVE\_STREAM}IMAGEnum\_handsThe maximum number of hands can be detected by the GestureRecognizer.Any integer > 01min\_hand\_detection\_confidenceThe minimum confidence score for the hand detection to be considered successful in palm detection model.0.0 - 1.00.5min\_hand\_presence\_confidenceThe minimum confidence score of hand presence score in the hand landmark detection model. In Video mode and Live stream mode of Gesture Recognizer, if the hand presence confident score from the hand landmark model is below this threshold, it triggers the palm detection model. Otherwise, a lightweight hand tracking algorithm is used to determine the location of the hand(s) for subsequent landmark detection.0.0 - 1.00.5min\_tracking\_confidenceThe minimum confidence score for the hand tracking

to be considered successful. This is the bounding box IoU threshold between hands in the current frame and the last frame. In Video mode and Stream mode of Gesture Recognizer, if the tracking fails, Gesture Recognizer triggers hand detection. Otherwise, the hand detection is skipped.0.0 - 1.00.5canned\_gestures\_classifier\_optionsOptions for configuring the canned gestures classifier behavior. The canned gestures are ["None", "Closed\_Fist", "Open\_Palm", "Pointing\_Up", "Thumb\_Down", "Thumb\_Up", "Victory", "ILoveYou"] Display names locale: the locale to use for display names specified through the TFLite Model Metadata, if any.

 Max results: the maximum number of top-scored classification results to return. If < 0, all available results will be returned.

 Score threshold: the score below which results are rejected. If set to 0, all available results will be returned.

 Category allowlist: the allowlist of category names. If non-empty, classification results whose category is not in this set will be filtered out. Mutually exclusive with denylist.

 Category denylist: the denylist of category names. If non-empty, classification results whose category is in this set will be filtered out. Mutually exclusive with allowlist.

* Display names locale: any string
* Max results: any integer
* Score threshold: 0.0-1.0
* Category allowlist: vector of strings
* Category denylist: vector of strings
* Display names locale: "en"
* Max results: -1
* Score threshold: 0
* Category allowlist: empty
* Category denylist: empty

custom\_gestures\_classifier\_optionsOptions for configuring the custom gestures classifierbehavior.

 Display names locale: the locale to use for display names specified through the TFLite Model Metadata, if any.

 Max results: the maximum number of top-scored classification results to return. If < 0, all available results will be returned.

 Score threshold: the score below which results are rejected. If set to 0, all available results will be returned.

 Category allowlist: the allowlist of category names. If non-empty, classification results whose category is not in this set will be filtered out. Mutually exclusive with denylist.

 Category denylist: the denylist of category names. If non-empty, classification results whose category is in this set will be filtered out. Mutually exclusive with allowlist.

* Display names locale: any string
* Max results: any integer
* Score threshold: 0.0-1.0
* Category allowlist: vector of strings
* Category denylist: vector of strings
* Display names locale: "en"
* Max results: -1
* Score threshold: 0
* Category allowlist: empty
* Category denylist: empty

result\_callbackSets the result listener to receive the classification results asynchronously when the gesture recognizer is in the live stream mode. Can only be used when running mode is set to LIVE\_STREAMResultListenerN/AN/A

## Prepare data

Prepare your input as an image file or a numpy array, then convert it to a mediapipe.Image object. If your input is a video file or live stream from a webcam, you can use an external library such as [OpenCV](https://github.com/opencv/opencv) to load your input frames as numpy arrays.

## Run the task

The Gesture Recognizer uses the recognize, recognize\_for\_video and recognize\_async functions to trigger inferences. For gesture recognition, this involves preprocessing input data, detecting hands in the image, detecting hand landmarks, and recognizing hand gesture from the landmarks.

The following code demonstrates how execute the processing with the task model.

Note the following:

* When running in the video mode or the live stream mode, you must also provide the Gesture Recognizer task the timestamp of the input frame.
* When running in the image or the video model, the Gesture Recognizer task will block the current thread until it finishes processing the input image or frame.
* When running in the live stream mode, the Gesture Recognizer task doesn’t block the current thread but returns immediately. It will invoke its result listener with the recognition result every time it has finished processing an input frame. If the recognition function is called when the Gesture Recognizer task is busy processing another frame, the task will ignore the new input frame.

## Handle and display results

The Gesture Recognizer generates a gesture detection result object for each recognition run. The result object contains hand landmarks in image coordinates, hand landmarks in world coordinates, handedness(left/right hand), and hand gestures categories of the detected hands.

The following shows an example of the output data from this task:

The resulted GestureRecognizerResult contains four components, and each component is an array, where each element contains the detected result of a single detected hand.

* Handedness

Handedness represents whether the detected hands are left or right hands.

* Gestures

The recognized gesture categories of the detected hands.

* Landmarks

There are 21 hand landmarks, each composed of x, y and z coordinates. The x and y coordinates are normalized to [0.0, 1.0] by the image width and height, respectively. The z coordinate represents the landmark depth, with the depth at the wrist being the origin. The smaller the value, the closer the landmark is to the camera. The magnitude of z uses roughly the same scale as x.

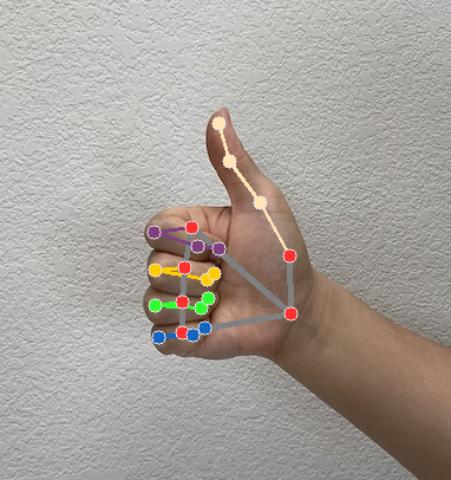
* World Landmarks

The 21 hand landmarks are also presented in world coordinates. Each landmark is composed of x, y, and z, representing real-world 3D coordinates in meters with the origin at the hand’s geometric center.

Note the following:

* When running in the video mode or the live stream mode, you must also provide the Gesture Recognizer task the timestamp of the input frame.
* When running in the image or the video model, the Gesture Recognizer task will block the current thread until it finishes processing the input image or frame.
* When running in the live stream mode, the Gesture Recognizer task doesn’t block the current thread but returns immediately. It will invoke its result listener with the recognition result every time it has finished processing an input frame. If the recognition function is called when the Gesture Recognizer task is busy processing another frame, the task will ignore the new input frame.

The following images shows a visualization of the task output:



## 

## 

## 

## Metaverse Virtual Influencer

A metaverse virtual influencer is a computer-generated character or avatar with a distinct personality, often created using AI and 3D modeling, which actively engages with audiences in virtual environments.

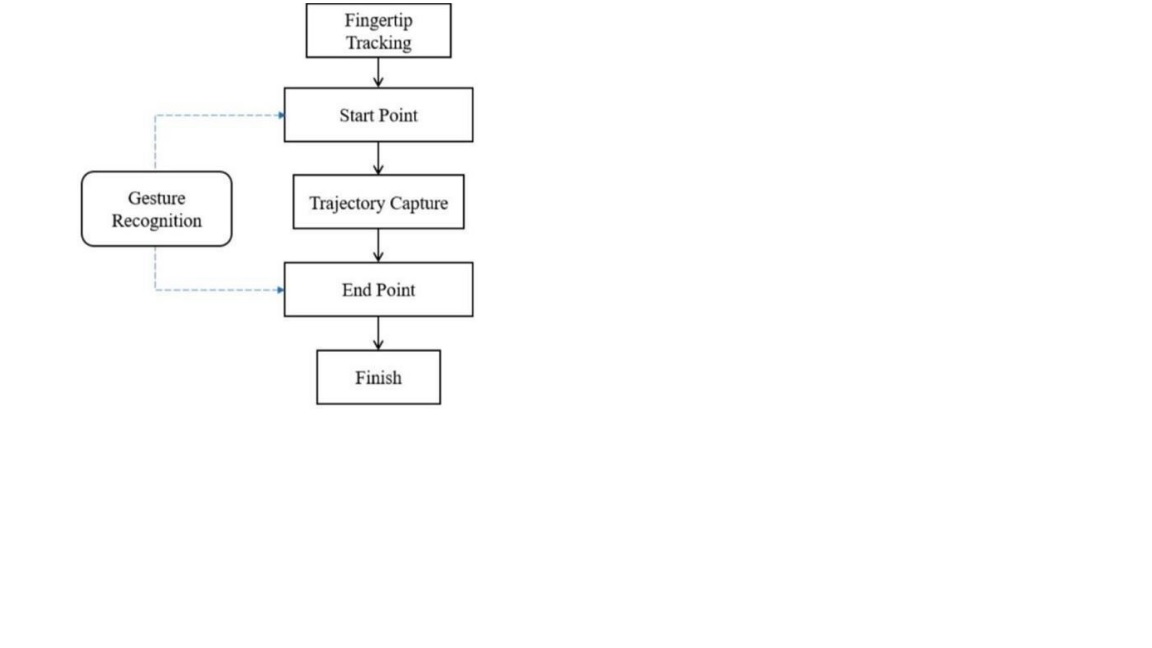
These digital beings can be found in social media, virtual worlds, and gaming platforms, where they share content, endorse and [advertise products](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10026852/), and build communities.

While they aren’t real individuals, they amass substantial followings by providing entertainment, information, and relatability to users. Their influence extends to fashion, gaming, technology, and more, shaping trends and sparking conversations within the metaverse.

Drawing Hand Pose Detection

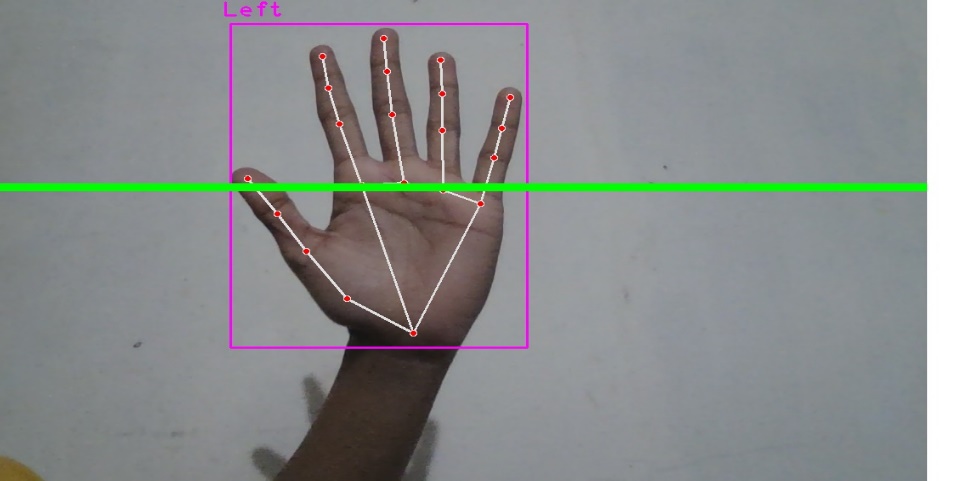
Acknowledging an important stage is to recognize the generating hand's position, as well as accept that through the other signals is an essential step in determining to create an airborne. And all the conventional writing, whenever the write goes down, and the write moves upward, creating it within the discussion is not assigned out as an arrangement of this writing incident . The Main framework acknowledges the location of that hand recognizing the supplied amount.

Usin this system we can draw virtually.we track our hand with the help of mediapipe.



Hand Region Segmentation

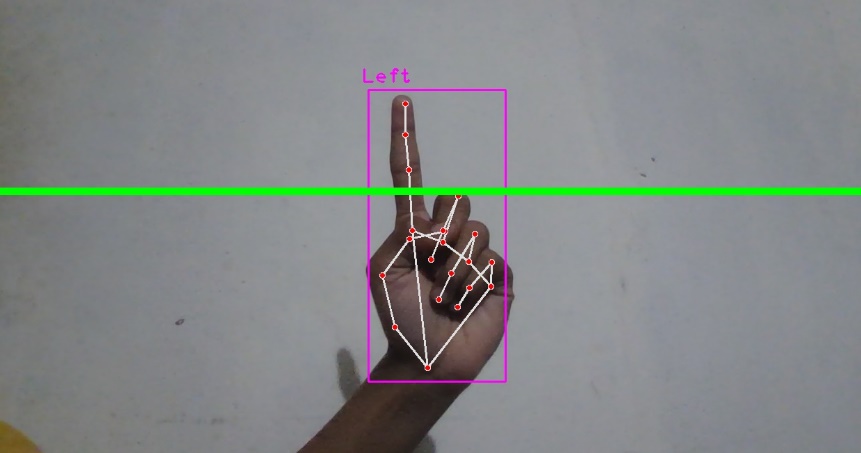
So Firstly the hand tracking and segmentation which are the main or important steps for any kind of hand gesture recognition system in this. They are having the aim to develop or to organize the robust and make it hand segmentation algorithm, it also catches the hand using over procedure, division of the hand zone has been done by hiring a two step approach in this. The division of skin and subtraction of the starting point and having the best combined picture of the hand is created from the input of two. They offered a procedure which works effective in realtime and also gives a quite exact division. The fact that have a skin colors shift especially from kind to kind, it was observed.



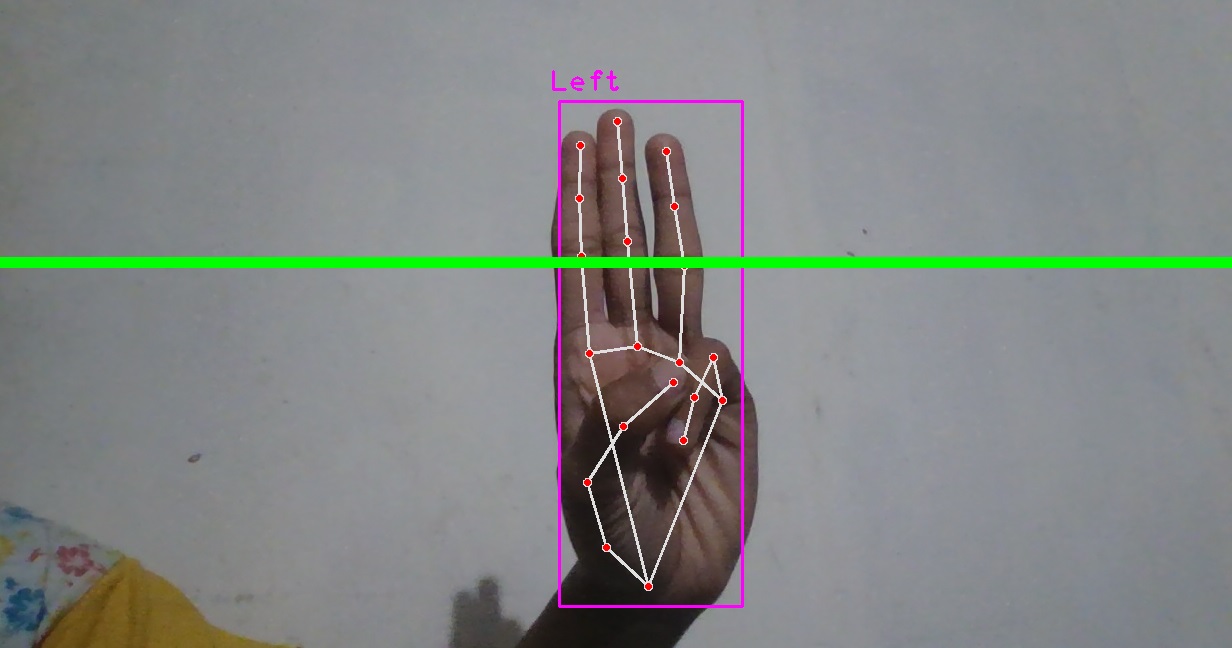
If you lift the index finger and middle finger together,you can move any where on the screen.



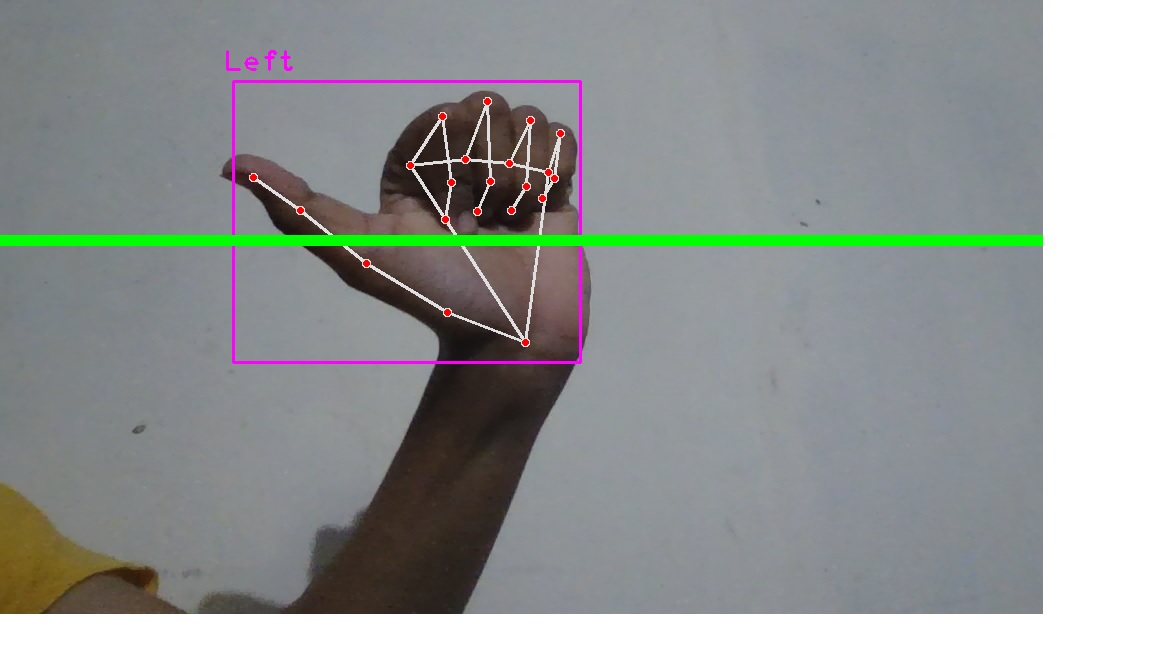
If we lift the index finger alone,we can draw.

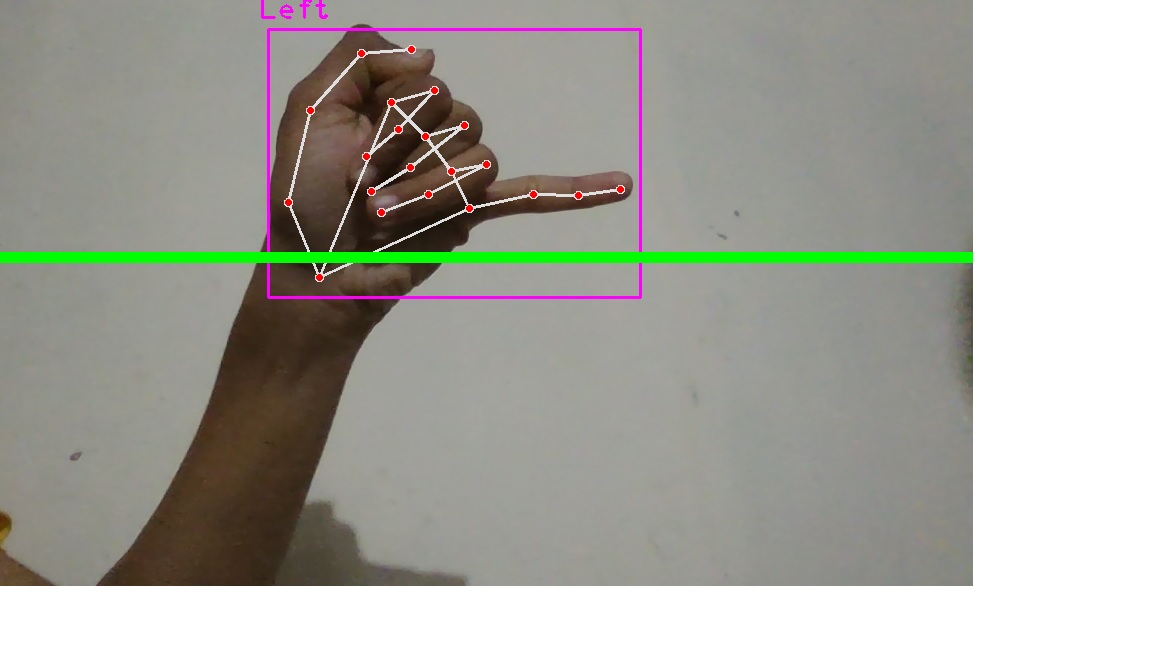


To remove the drawing,the middle finger,index finger and ring finger should be lifted together.The three finger removing based on what we draw after each entry.



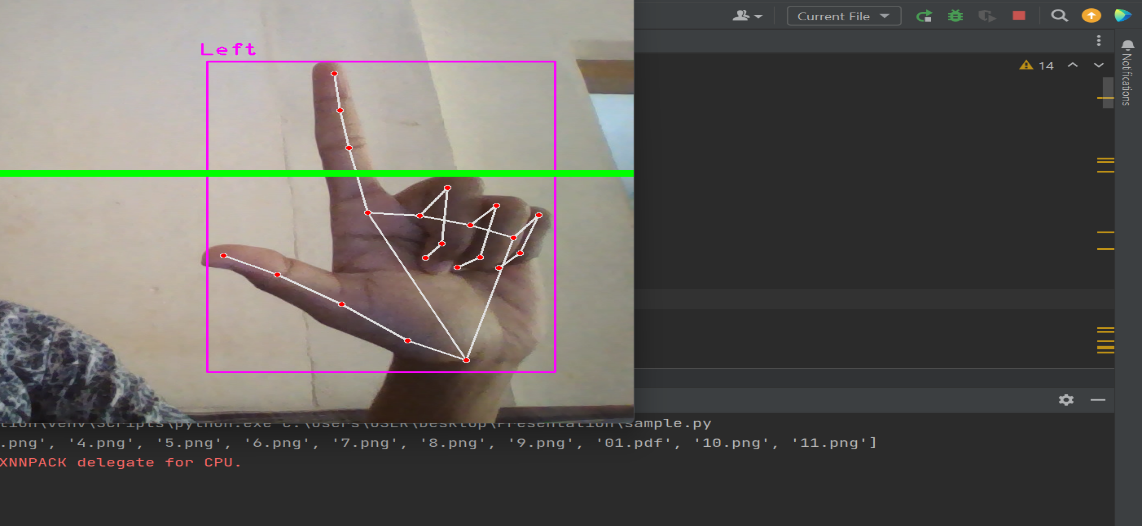
You have to go to another slide:



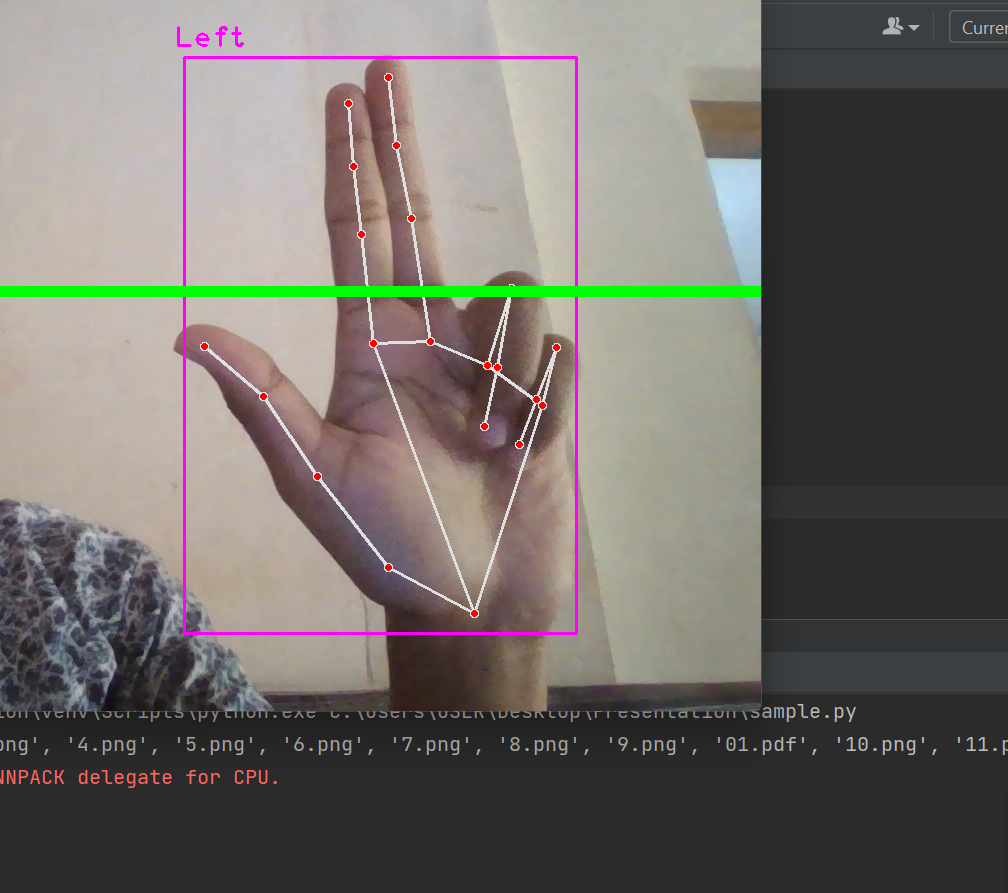


Ink color selection:

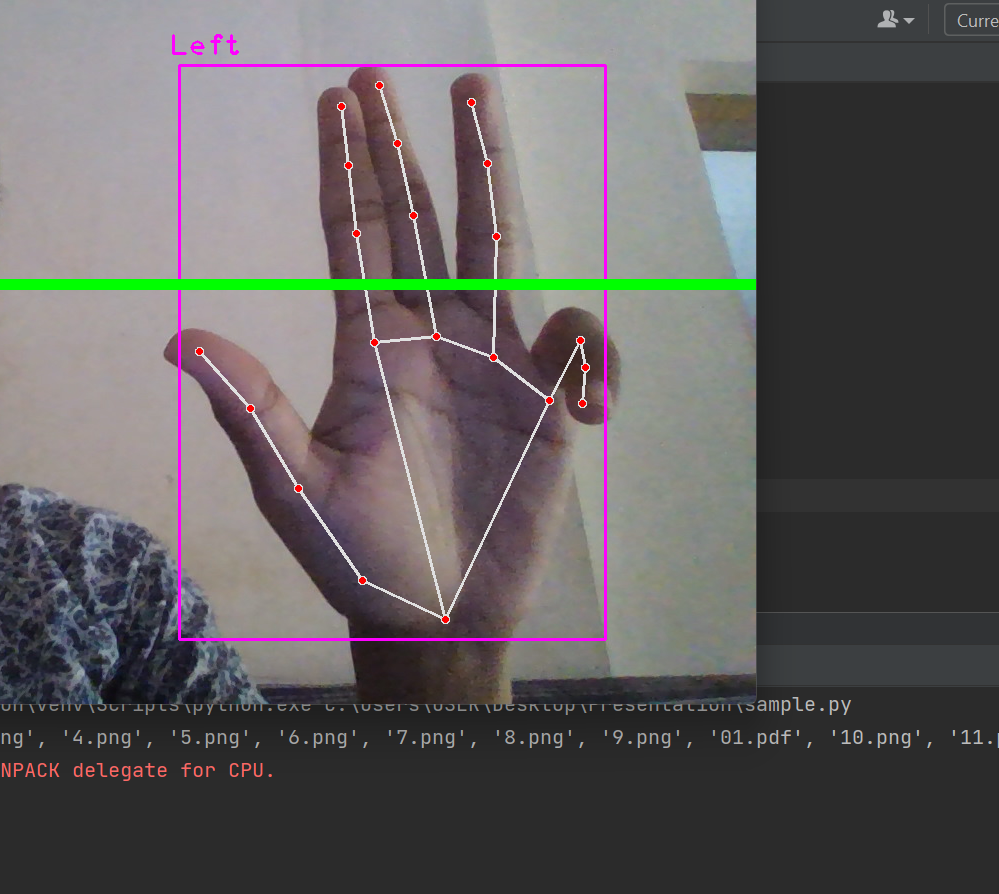
Select red color:-



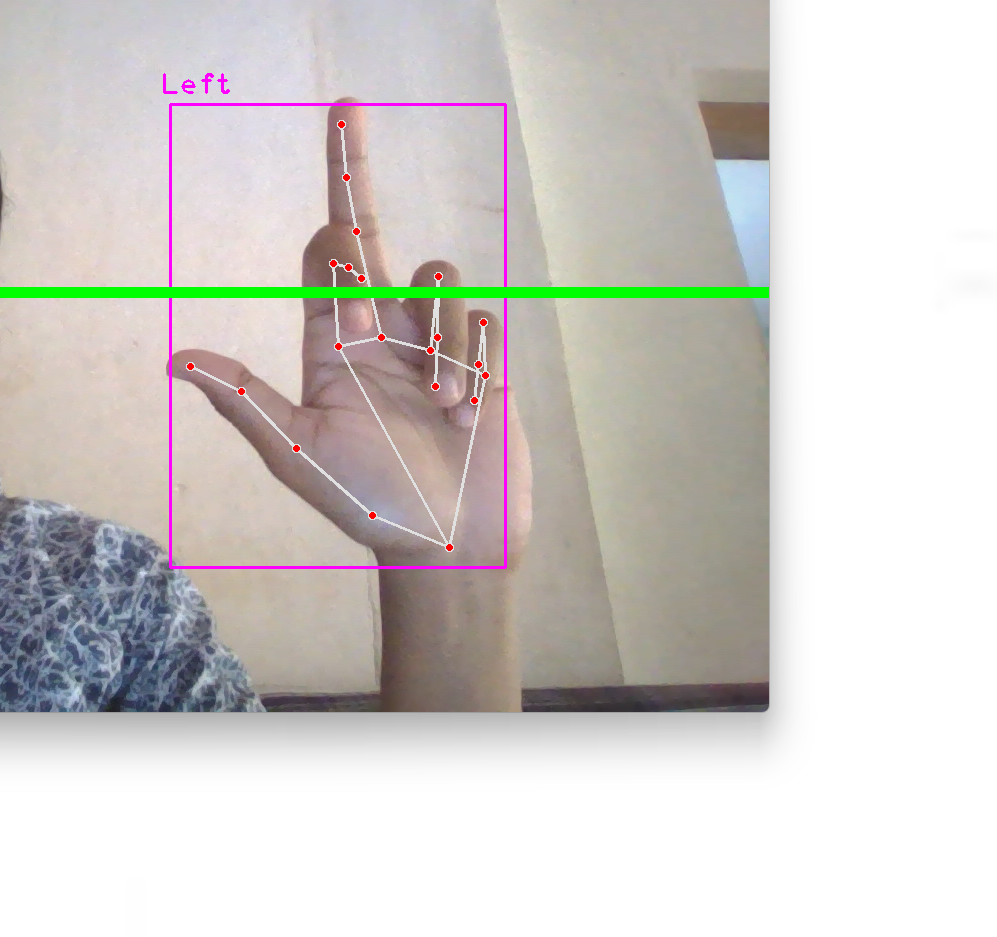
Select green:-



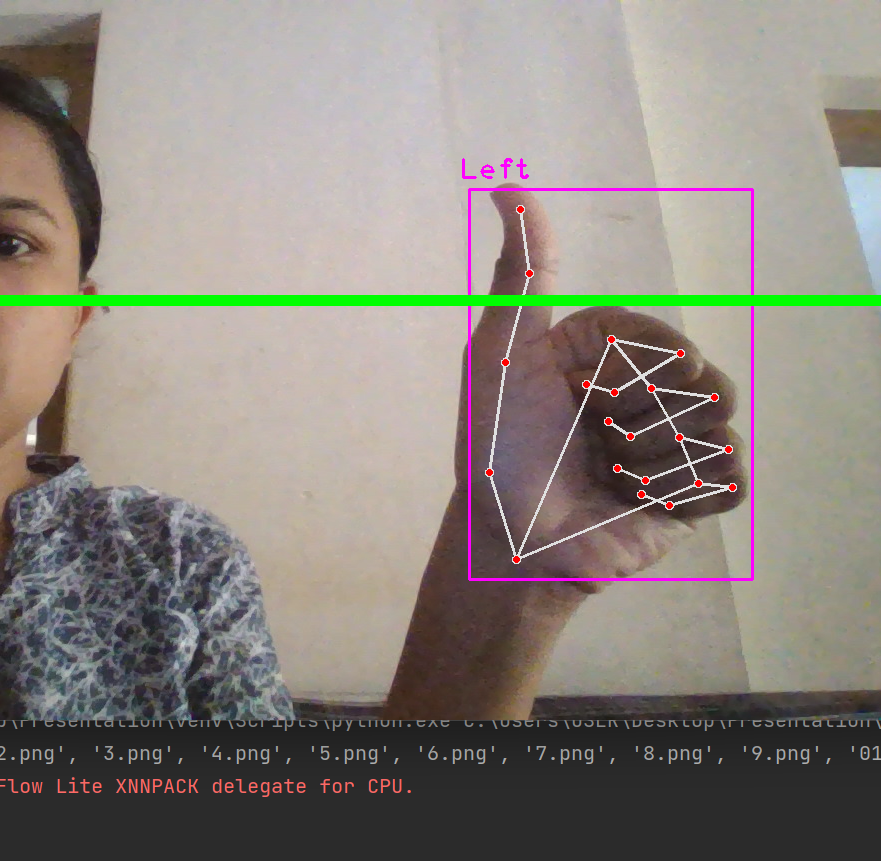
Select blue:-



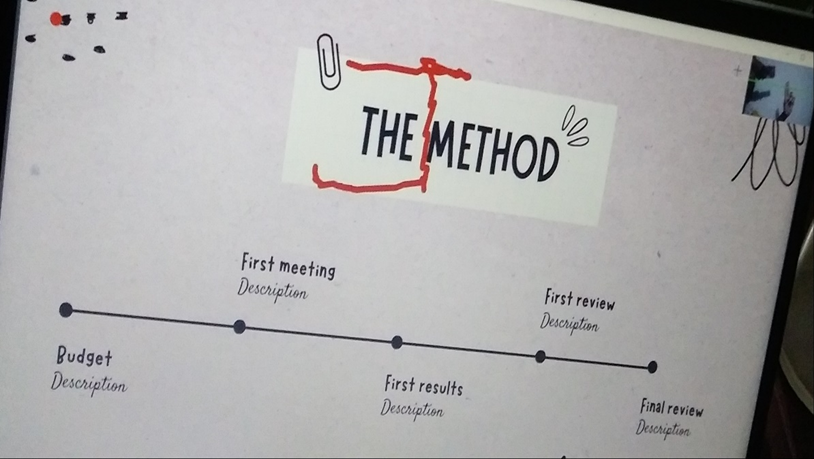
Zoom in:-

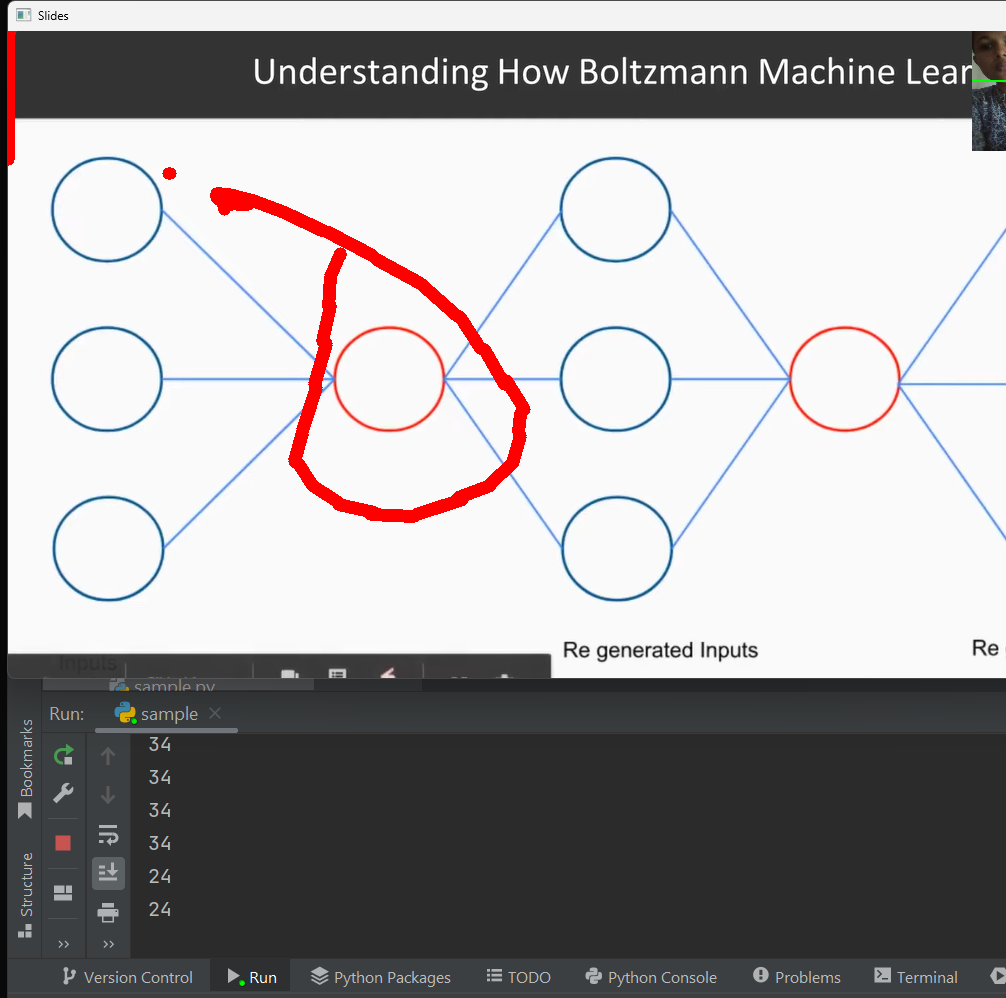


Zoom out:-



Result:-





CONCLUSION

This concept allows the individual to experience a unique and reciprocal world where they one might be able to draw.we can draw and which are shows on the screen and in this we select the virtual AI screen and painter is an application which is developed by using the library OpenCv and mediapipe and where we are also having more libraries and the different algorithm which are in built and also make there interface more active and effective. In this project we also used python programming language which also have many libraries that are defined in that and this project can also be put to in many fields which never required lots of experience in typing and it may also very structured as it took a little amount of time to send to the persons minds and we can make a space for the persons to draw anything whatever they actually wants by using with a single finger and lift the index finger and middle finger together,you can move any where on the screen. To remove the drawing,the middle finger,index finger and ring finger should be lifted together.The three finger removing based on what we draw after each entry. This system could be used as an alternative for teaching software used by teachers and students.If further interpreter various virtual based physical game could be made.Controlling the robot using gesture considered as one of the interesting applications in this field proposed a system controlling a robot using hand pose signs.The orders could be given to robot to execute some task,where each sign has a specific meaning and represents different function. . Teaching chemical calculations or organic reaction mechanisms is much easier on a whiteboard (TRUST ME!)than making fancy PPT slides. It also allows for more creativity and dialogue among the students, as they can contribute their own ideas and feedback on the board. On the other hand, PowerPoint presentation is more suitable for structured and complex teaching, where the teacher needs to follow a certain flow and present a lot of information and parameters in a clear and organised way. I use animations (e.g.: SN1 and SN2 reaction mechanisms ) which make my students remember complex topics. These also work best for KS3 science as slides remind me to what limit I can teach certain topics. PowerPoint slides also allow for more consistency and reliability in the presentation, as the teacher can prepare the slides beforehand and slides can be uploaded on learning platforms such as Teams or Google Classroom for future reference when pupils revise the topics taught.Using virtual environment it make easier than other teaching methods.

Drawbacks

->This application not suitable for inputting large amounts of data,not very accurate.

->Selecting detailed objects can be difficult with fingers it is more expensive than alternatives such as mouse and can soon become faulty if misused.

->Loss of human touch:Another potential drawback of using AI art is is that it can lead to a loss of the human touch and personal coonection that is often present in traditional art.

->Electricity is want.

->

Code

from cvzone.HandTrackingModule import HandDetector  
import cv2  
import os  
import numpy as np  
  
# Parameters  
width, height = 1280, 720  
gestureThreshold = 300  
folderPath = "Presentation"  
  
# Camera Setup  
cap = cv2.VideoCapture(0)  
cap.set(3, width)  
cap.set(4, height)  
  
# Hand Detector  
detectorHand = HandDetector(detectionCon=0.8, maxHands=1)  
  
# Variables  
imgList = []  
delay = 30  
buttonPressed = False  
counter = 0  
drawMode = False  
imgNumber = 0  
delayCounter = 0  
annotations = [[]]  
annotationNumber = -1  
annotationStart = False  
hs, ws = int(120 \* 1), int(213 \* 1) # width and height of small image  
ink\_color = (0, 0, 255) # Initial ink color (red)  
  
# Get list of presentation images  
pathImages = sorted(os.listdir(folderPath), key=len)  
print(pathImages)  
  
# Zoom variables  
zoom\_level = 1.0 # Initial zoom level  
min\_zoom = 0.5 # Minimum zoom level  
max\_zoom = 2.0 # Maximum zoom level  
  
while True:  
 # Get image frame  
 success, img = cap.read()  
 img = cv2.flip(img, 1)  
 pathFullImage = os.path.join(folderPath, pathImages[imgNumber])  
 imgCurrent = cv2.imread(pathFullImage)  
  
 # Find the hand and its landmarks  
 hands, img = detectorHand.findHands(img) # with draw  
 # Draw Gesture Threshold line  
 cv2.line(img, (0, gestureThreshold), (width, gestureThreshold), (0, 255, 0), 10)  
  
 if hands and buttonPressed is False: # If hand is detected  
  
 hand = hands[0]  
 cx, cy = hand["center"]  
 lmList = hand["lmList"] # List of 21 Landmark points  
 fingers = detectorHand.fingersUp(hand) # List of which fingers are up  
  
 # Constrain values for easier drawing  
 xVal = int(np.interp(lmList[8][0], [width // 2, width], [0, width]))  
 yVal = int(np.interp(lmList[8][1], [150, height-150], [0, height]))  
 indexFinger = xVal, yVal  
  
 if cy <= gestureThreshold: # If hand is at the height of the face  
 if fingers == [1, 0, 0, 0, 0]:  
 print("Left")  
 buttonPressed = True  
 if imgNumber > 0:  
 imgNumber -= 1  
 annotations = [[]]  
 annotationNumber = -1  
 annotationStart = False  
 if fingers == [0, 0, 0, 0, 1]:  
 print("Right")  
 buttonPressed = True  
 if imgNumber < len(pathImages) - 1:  
 imgNumber += 1  
 annotations = [[]]  
 annotationNumber = -1  
 annotationStart = False  
  
 if fingers == [0, 1, 1, 0, 0]:  
 cv2.circle(imgCurrent, indexFinger, 12, ink\_color, cv2.FILLED)  
  
 if fingers == [0, 1, 0, 0, 0]:  
 if annotationStart is False:  
 annotationStart = True  
 annotationNumber += 1  
 annotations.append([])  
 print(annotationNumber)  
 annotations[annotationNumber].append(indexFinger)  
 cv2.circle(imgCurrent, indexFinger, 12, ink\_color, cv2.FILLED)  
  
 else:  
 annotationStart = False  
  
 if fingers == [0, 1, 1, 1, 0]:  
 if annotations:  
 annotations.pop(-1)  
 annotationNumber -= 1  
 buttonPressed = True  
  
 # Color selection gesture  
 if fingers == [1, 1, 0, 0, 0]:  
 ink\_color = (0, 0, 255) # Red color  
 elif fingers == [1, 1, 1, 0, 0]:  
 ink\_color = (0, 255, 0) # Green color  
 elif fingers == [1, 1, 1, 1, 0]:  
 ink\_color = (255, 0, 0) # Blue color  
  
 # Zoom gestures  
 if fingers:  
 if fingers == [1, 0, 1, 0, 0]: # Two fingers up  
 zoom\_level += 0.1 # Increase zoom level  
 if zoom\_level > max\_zoom:  
 zoom\_level = max\_zoom  
  
 if fingers == [1, 0, 0, 0, 0]: # Thumb up  
 zoom\_level -= 0.1 # Decrease zoom level  
 if zoom\_level < min\_zoom:  
 zoom\_level = min\_zoom  
  
 else:  
 annotationStart = False  
  
 if buttonPressed:  
 counter += 1  
 if counter > delay:  
 counter = 0  
 buttonPressed = False  
  
 for i, annotation in enumerate(annotations):  
 for j in range(len(annotation)):  
 if j != 0:  
 cv2.line(imgCurrent, annotation[j - 1], annotation[j], ink\_color, 12)  
  
 imgSmall = cv2.resize(img, (ws, hs))  
 h, w, \_ = imgCurrent.shape  
 imgCurrent[0:hs, w - ws: w] = imgSmall  
  
 # Resize the image based on the zoom level  
 imgCurrentZoomed = cv2.resize(imgCurrent, None, fx=zoom\_level, fy=zoom\_level)  
  
 # Display the zoomed image  
 cv2.imshow("Slides", imgCurrentZoomed)  
 cv2.imshow("Image", img)  
  
 key = cv2.waitKey(1)  
 if key == ord('q'):  
 break

REFERENCE

-> https://opencv.org/

-><https://www.jetbrains.com/pycharm/>

-> <https://www.python.org/>

-><https://developers.google.com/mediapipe/solution/vision/gesture_recoganizer/web_js>

-> https://google.github.io/mediapipe/solutions/ pose#python-solution-api

-><https://www.section.io/engineering-education/creating-a-hand-tracking-module/>

-> <https://ieeexplore.ieee.org/abstract/document/9742419>

-> <https://ieeexplore.ieee.org/abstract/document/9382703>

-><https://ieeexplore.ieee.org/abstract/document/9340297>

-> <https://ieeexplore.ieee.org/abstract/document/9641587>

-> <https://ieeexplore.ieee.org/abstract/document/9753939>

-> https://ieeexplore.ieee.org/abstract/document/9775918