**A.I Virtual Mouse**

Major Project

**BACHELOR OF Computer Application**

(Computer Application)



**SUBMITTEDBY:**

Name

Rollo

year

**Under the Guidance of** Name of Mentor

Designation of mentor

**Department of Computer Application**

**Chandigarh School of Business Jhanjeri, Mohali – 140307**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S. No** | **Content** | **Page No** |
| 1. | Introduction |  |
| 2. | Certificate by Company |  |
| 3. | Declarations |  |
| 4. | Abstract |  |
| 5. | Acknowledgement |  |
| 6. | About the Training |  |
| 7. | Introduction of the company |  |
| 8. | Idea Behind Project |  |
| 9. | Technologies learnt and used |  |
| 10 | Software Used |  |
| 11 | Introduction to A. I |  |
| 12. | About the Project |  |
| 13. | Literature Review |  |
| 14. | Methodology |  |
| 15. | Result and Evaluation |  |
| 16. | Discussion |  |
| 17. | Screenshot |  |
| 18. | Conclusion |  |
| 19. | References |  |

**CERTIFICATE BY COMPANY**

**DECLARATION**

I ……………. studying in the Third Year of BTech CSE course in the academic year 2017-21 at Lyallpur Khalsa College of Engineering, hereby declare that I have completed the project titled **“Bank management”** as a part of the course requirements.

I further declare that the information presented in this project is true and original to the best of my knowledge.

**ABSTRACT**

A virtual mouse is an innovative solution to control the computer cursor without the need for a physical mouse. This project report presents the development of a virtual mouse using computer vision techniques and machine learning algorithms. The proposed system tracks the hand movements of the user and maps them to corresponding cursor movements on the screen. The system is designed to be user-friendly, accurate, and reliable. The implementation of the virtual mouse is achieved by using OpenCV and Python programming language. The system is tested and evaluated in various scenarios to demonstrate its effectiveness and usability. The virtual mouse can be a viable alternative for people with disabilities or those who prefer not to use a physical mouse. The report also highlights the limitations and future directions for improvement of the proposed system.

**ACKNOWLEDGEMENT**

I would like to place on record my deep sense of gratitude to **…………………..** for his generous guidance, help, useful suggestions and continuous encouragement.

I am extremely thankful to **………………..** , Principal Director of ………………………………………… for providing encouragement and allowing me to work in Web Development; supporting and guiding me regarding the same.

I am extremely thankful to **………………**, HOD, CSE department, for valuable suggestions and motivation.

I am also thankful to all my Teachers who have taught me throughout my training. And providing the opportunity to get the knowledge.

**Name- ………………..**

**Roll no.- …………………….**

**About the Training**

The Training in PGTECH PVT. LTD. was pertaining to PYTHON. It was followed under the guidance of …………………………….

This project is developed by using **PYTHON**. Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by **Guido van Rossum** during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactical constructions than other languages.

**Introduction of the Company**

**“PG-TECH PVT. LTD.”**



We found that due to less availability of awareness in the technical field student does not show their interest to learn technical skills like programming and Digital marketing skills. We have started the initiative to encourage students to join technical fields and high-salary jobs by providing them with training, and guidance and explaining their future scope. Here at Pg Tech Pvt. Ltd., we are providing training in various fields like front-end developer & back-end developer courses, mechanical and civil software training courses, and basic computer courses along with job placement in Chandigarh & Mohali. we train every student from beginners level and make them at the advanced level under our expert guidance. If you are looking for the best C & C++ training institutes near me then you should visit our coaching institute and attend free demo classes to decide better.

Through our experience we understand child learning nature, every child has their own unique learning interest we should not force them to learn anything against their wish but yes we are telling them every possibility where they can improve themselves, what things they should follow to earn good salary packages, all this we tell them during the free demo session. we have also designed basic computer courses that will help them to get jobs in any sector because we know today the Indian government changing India completely digitally. Like the education sector changes digitally, business, payment everything. so we provide training in Microsoft Word and Excel courses, fast computer typing courses with certifications, etc. To know more kindly visit our technical institute in Chandigarh or you can call us.

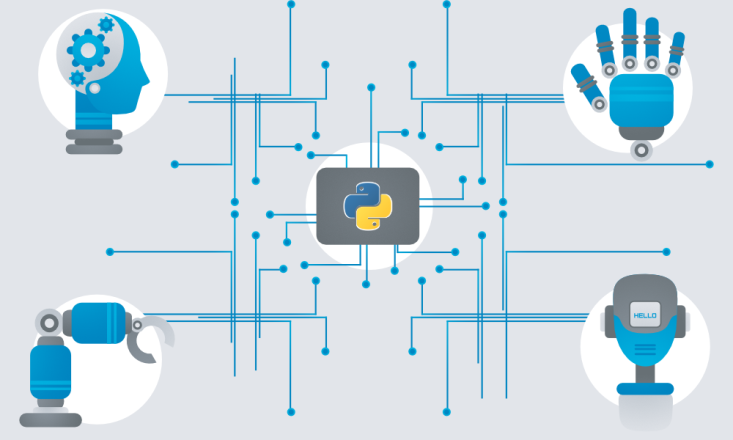
**Idea Behind Project**



The idea behind the virtual mouse project is to provide a solution for users who are unable to use a physical mouse or prefer not to use one. The physical mouse requires the user to have a stable hand, fine motor control, and the ability to click and drag. However, not everyone has these abilities. For example, people with disabilities affecting their motor skills, such as Parkinson's disease or cerebral palsy, may find it challenging to use a physical mouse. Moreover, some users may not have access to a mouse or may prefer to use an alternative input device. The virtual mouse system aims to address these issues by providing a hands-free alternative for controlling the computer cursor. The system tracks the user's hand movements using computer vision techniques and maps them to corresponding cursor movements on the screen. This allows users to control the cursor with their hand gestures, without requiring any physical contact with a mouse. The project's goal is to develop a virtual mouse system that is accurate, reliable, and easy to use. The system should be able to recognize a wide range of hand gestures and perform precise cursor movements. Additionally, the system should be user-friendly and accessible to people with different levels of technical expertise.

**Technologies Learnt and Used**

**PYTHON**

****

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

U7Python 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 is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.



**History of Python**

**** Picture of the Creator of Python - [Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum)

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

**Python's features include −**

* **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 customize 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.

**Apart from the above-mentioned features, Python has a big list of good features, few are listed below −**

* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* IT supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

## **Real-world Applications of Python**

### 1. Web Development

When it comes to web development, Python should be your go-to tool. Why?

That’s because Python offers numerous options for web development. For instance, you have Django, Pyramid, Flask, and Bottle for developing web frameworks and even advanced content management systems like Plone and Django CMS. These web frameworks are packed with standard libraries and modules which simplify tasks like content management, database interaction, and interfacing with internet protocols like HTTP, SMTP, XML, JSON, FTP, IMAP, and POP.

Python web frameworks are known for their security, scalability, and flexibility. To add to that, Python’s Package Index comes with useful libraries like Requests, BeautifulSoup, Paramiko, Feedparser, and Twisted Python.

### 2. Game Development

As we mentioned earlier, Python comes loaded with many useful extensions (libraries) that come in handy for the development of interactive games. For instance, libraries like PySoy (a 3D game engine that supports Python 3) and PyGame are two Python-based **libraries used widely for game development**. Python is the foundation for popular games like Battlefield 2, Frets on Fire, World of Tanks, Disney’s Toontown Online, Vega Strike, and Civilization-IV.

Apart from game development, game designers can also use Python for developing tools to simplify specific actions such as level design or dialog tree creation, and even use those tools to export those tasks in formats that can be used by the primary game engine. Also, Python is used as a scripting language by many game engines.

### 3. Scientific and Numeric Applications

Thanks to its massive library base, Python has become a crucial tool in scientific and numeric computing. In fact, Python provides the skeleton for applications that deal with computation and scientific data processing. Apps like FreeCAD (3D modeling software) and Abaqus (finite element method software) are coded in Python.

Some of the most useful Python packages for scientific and numeric computation include:

* SciPy (scientific numeric library)
* Pandas (data analytics library)
* IPython (command shell)
* Numeric Python (fundamental numeric package)
* Natural Language Toolkit (Mathematical And text analysis)

### **4.** Artificial Intelligence and Machine Learning

AI and ML models and projects are inherently different from traditional software models. When we talk about AI/ML projects, the tools and technologies used and the skillset required is totally different from those used in the development of conventional software projects. AI/ML applications require a language that is stable, secure, flexible, and is equipped with tools that can handle the various unique requirements of such projects. Python has all these qualities, and hence, it has become one of the most favored languages of Data Science professionals.

Python’s simplicity, consistency, platform independence, great collection of resourceful libraries, and an active community make it the perfect tool for developing AI and ML applications. Some of the best Python packages for AI and ML are:

* SciPy for advanced computing
* Pandas for general-purpose data analysis
* Seaborn for data visualization
* Keras, TensorFlow, and Scikit-learn for ML
* NumPy for high-performance scientific computing and data analysis

 Apart from these libraries, there are also other Python-based libraries like NLTK, Caffee, PyTorch, and Accord. NET, that are useful for AI and ML projects.

### Desktop GUI

Python not only boasts of an English-like syntax, but it also features a modular architecture and the ability to work on multiple operating systems. These aspects, combined with its rich text processing tools, make Python an excellent choice for developing desktop-based GUI applications.

Python offers many GUI toolkits and frameworks that make desktop application development a breeze. PyQt, PyGtk, Kivy, Tkinter, WxPython, PyGUI, and PySide are some of the best Python-based GUI frameworks that allow developers to create highly functional Graphical User Interfaces (GUIs).

### 6. Software Development

Python packages and applications aim to simplify the process of software development. From developing complex applications that involve scientific and numeric computing to developing desktop and web applications, Python can do it all. This is the reason why **Software Developers use Python as a support language** for build control, testing, and management.

For instance, SCons is designed explicitly for build control, Buildbot and Apache Gump allow for automated continuous compilation and testing, and Roundup and Trac are great for bug tracking and project management.

Python also supports data analyzation and visualization, thereby further simplifying the process of creating custom solutions minus the extra effort and time investment.

### 7. Enterprise-level/Business Applications

Enterprise-level software or business applications are strikingly different from standard applications, as in the former demands features like readability, extensibility, and scalability. Essentially, business applications are designed to fit the requirements of an organization rather than the needs of individual customers.

Thus, these applications must be capable of integrating with legacy systems like existing databases and non-web apps. Since business applications are developed, keeping in mind the custom requirements to cater to the specific needs of an organization’s operating model, the entire development process becomes very complicated.

This is where Python can make a significant difference. Python high performance, scalability, flexibility, and readability are just the features required for developing fully-functional and efficient business applications. Furthermore, Python has other tools for business application development, like:

* Odoo, an all-in-one management software that forms a complete suite of enterprise management applications.
* Tryton, a three-tier, high-level, general-purpose application platform, is another fantastic tool for building business applications.

### 8. Education programs and training courses

If there’s any beginner-friendly programming language, it is Python. We’ve said it many times before, and we’re repeating it – Python has an extremely straightforward syntax that’s similar to the English language. It has a short learning curve and hence, is an excellent choice for beginners. Python’s easy learning curve and simplicity are the two main reasons why it is one of the most used programming languages in educational programs, both at beginner and advanced levels.

However, Python is not just great as an introductory language – even professional developers and coders all around the world rely heavily on Python.

### 9. Language Development

Over the years, Python’s design and module architecture has been the inspiration behind the development of many new programming languages such as Boo, Swift, CoffeeScript, Cobra, and OCaml. All of these languages share numerous similarities with Python on grounds like object model, syntax, and indentation.

### 10. Operating Systems

Yes, Python is the secret ingredient behind many operating systems as well, most popularly of Linux distributions. Linux-based Ubuntu’s Ubiquity Installer and Fedora and Red Hat Enterprise’s Anaconda Installer are coded in Python. Even Gentoo Linux leverages Python Portage (package management system). Usually, Python is combined with the C programming language to design and develop operating systems.

### 11. Web Scraping Applications

Python is a nifty tool for extracting voluminous amounts of data from websites and web pages. The pulled data is generally used in different real-world processes, including job listings, price comparison, R&D, etc.

BeautifulSoup, MechanicalSoup, Scrapy, LXML, Python Requests, Selenium, and Urllib are some of the best Python-based web scraping tools.

### 12. Image Processing and Graphic Design Applications:

Alongside all the uses mentioned above, Python also finds a unique use case in image processing and graphic design applications. The programming language is used globally to design and build 2D imaging software like Inkscape, GIMP, Paint Shop Pro, and Scribus. Also, Python is used in several 3D animation packages such as Blender, Houdini, 3ds Max, Maya, Cinema 4D, and Lightwave, to name a few.

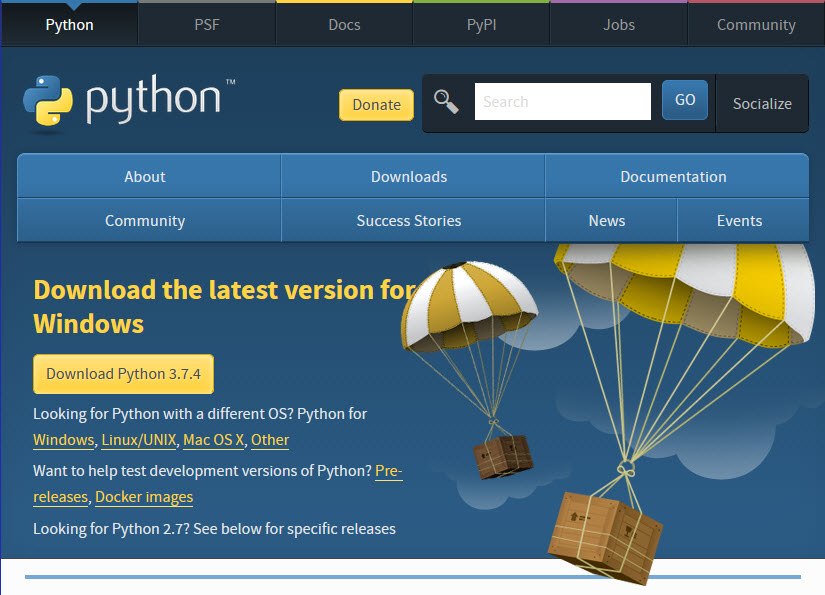
## Installation of Python:

The Python download requires about 25 Mb of disk space; keep it on your machine, in case you need to re-install Python. When installed, Python requires about an additional 90 Mb of disk space.

### Downloading

1. Click Python Download.

The following page will appear in your browser.

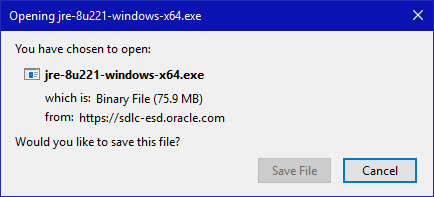


1. Click the **Windows** link (two lines below the **Download Python 3.7.4** button). The following page will appear in your browser.



1. Click on the **Download Windows x86-64 executable installer** link under the top-left **Stable Releases**.

The following pop-up window titled **Opening python-3.74-amd64.exe** will appear.



Click the **Save File** button.

The file named **python-3.7.4-amd64.exe** should start downloading into your standard download folder. This file is about 30 Mb so it might take a while to download fully if you are on a slow internet connection (it took me about 10 seconds over a cable modem).

The file should appear as

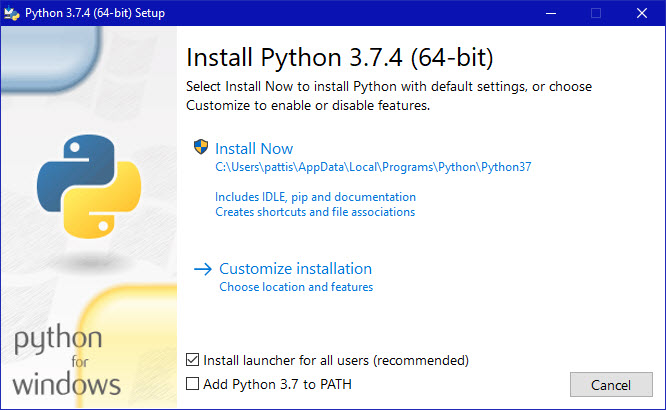
https://www.ics.uci.edu/~pattis/common/handouts/pythoneclipsejava/images/python/exefile.jpg

1. Move this file to a more permanent location, so that you can install Python (and reinstall it easily later, if necessary).
2. Feel free to explore this webpage further; if you want to just continue the installation, you can terminate the tab browsing this webpage.
3. Start the **Installing** instructions directly below.

### Installing

1. Double-click the icon labeling the file **python-3.7.4-amd64.exe**.

A **Python 3.7.4 (64-bit) Setup** pop-up window will appear.



Ensure that the **Install launcher for all users (recommended)** and the **Add Python 3.7 to PATH** checkboxes at the bottom are checked.

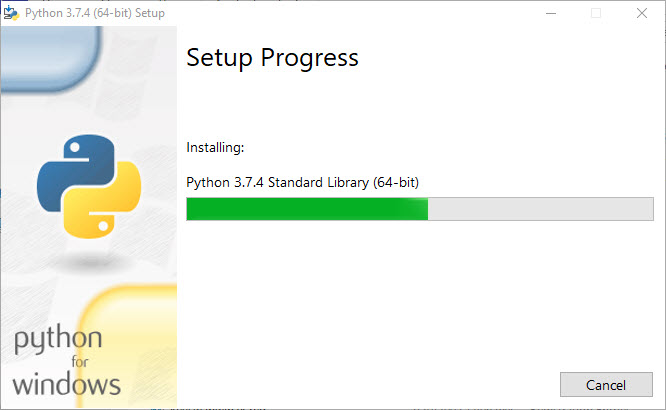
If the Python Installer finds an earlier version of Python installed on your computer, the **Install Now** message may instead appear as **Upgrade Now** (and the checkboxes will not appear).

1. Highlight the **Install Now** (or **Upgrade Now**) message, and then click it.

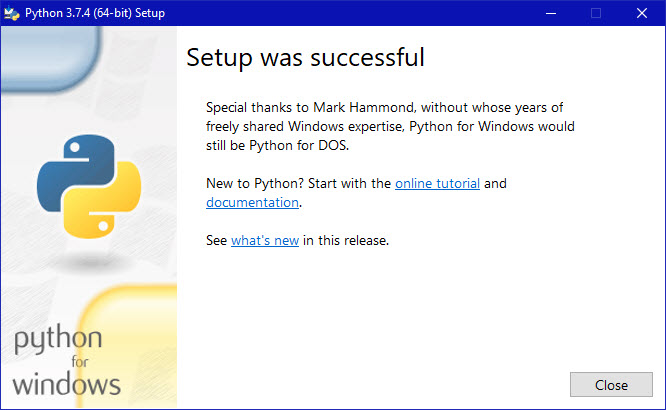
When run, a **User Account Control** pop-up window may appear on your screen. I could not capture its image, but it asks, **Do you want to allow this app to make changes to your device**.

1. Click the **Yes** button.

A new **Python 3.7.4 (64-bit) Setup** pop-up window will appear with a **Setup Progress** message and a progress bar.



During installation, it will show the various components it is installing and move the progress bar towards completion. Soon, a new **Python 3.7.4 (64-bit) Setup** pop-up window will appear with a **Setup was successfuly** message.



1. Click the **Close** button.

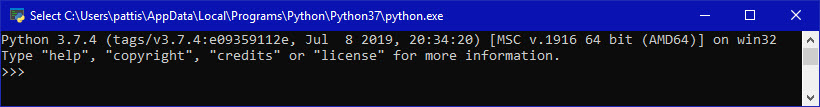
Python should now be installed.

### Verifying

To try to verify installation,

1. Navigate to the directory **C:\Users\Pattis\AppData\Local\Programs\Python\Python37** (or to whatever directory Python was installed: see the pop-up window for Installing step 3).
2. Double-click the icon/file **python.exe**.

The following pop-up window will appear.



A pop-up window with the title **C:\Users\Pattis\AppData\Local\Programs\Python\Python37\python.exe** appears, and inside the window; on the first line is the text **Python 3.7.4 ...** (notice that it should also say 64 bit). Inside the window, at the bottom left, is the prompt **>>>**: type **exit()** to this prompt and press **enter** to terminate Python.

You should keep the file **python-3.7.4.exe** somewhere on your computer in case you need to reinstall Python (not likely necessary).

**Lists in Python**

A list is a collection which is ordered and changeable. In Python lists are written with square brackets.

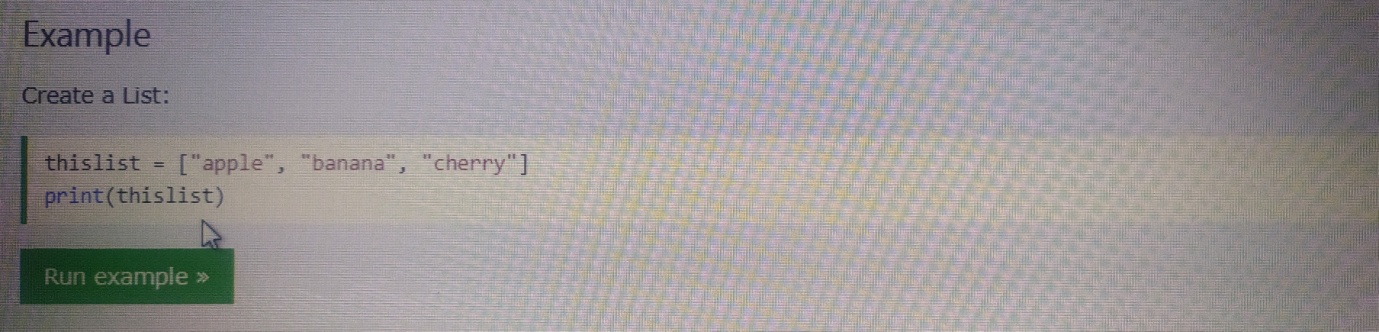


Figure of a program to create a list

**Type of Lists :**

# 1 .sort() :

# Like [C++ sort()](https://www.geeksforgeeks.org/sort-c-stl/), [Java sort()](https://www.geeksforgeeks.org/arrays-sort-in-java-with-examples/) and other languages, python also provides built in function to sort.

* The sort function can be used to sort the list in both ascending and descending order.
* To sort the list in ascending order.

**2. remove():**

* The remove() method takes a single element as an argument and removes it from the list.
* If the element doesn't exist, it throws **ValueError: list.remove(x): x not in list** exception.

**3. reverse() :**

* The reverse() function doesn't take any argument.
* The reverse() function doesn't return any value. It only reverses the elements and updates the [list](https://www.programiz.com/python-programming/list).

**4. Insert() :**

* **index** - position where an element needs to be inserted.
* **element** - this is the element to be inserted in the list.

**5. Clear() :**

* The clear() method doesn't take any parameters.
* The clear() method only empties the given [list](https://www.programiz.com/python-programming/list). It doesn't return any value.

**6. append() :**

The method takes a single argument

* item - an item to be added at the end of the list
* The item can be numbers, strings, dictionaries, another list, and so on.

**7. Index() :**

This method takes a single argument:

* **element** - element that is to be searched.

**8. pop() :**

* The pop() method takes a single argument (index).
* The argument passed to the method is optional. If not passed, the default index **-1** is passed as an argument (index of the last item).
* If the index passed to the method is not in range, it throws **IndexError: pop index out of range**exception.

**9. count() :**

The count() method takes a single argument:

* **element**- element whose count is to be found.

**Strings in Python**

String literals in python are surrounded by either single quotation marks, or double quotation marks.

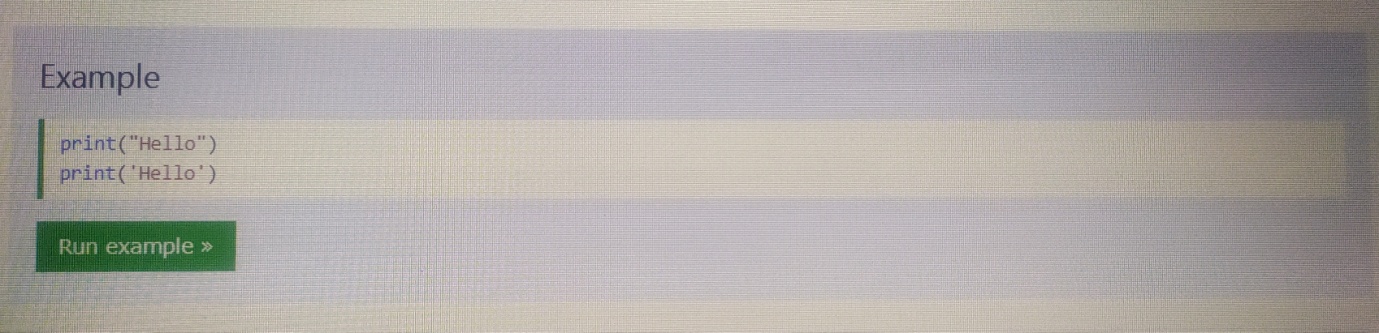
****

Figure of display a string

**Type of Strings :**

* 1. Index
  2. Find
  3. Rfind
  4. Rindex
  5. Capitalized
  6. Title
  7. Lower
  8. Upper
  9. Casefold
  10. Split
  11. Rsplit
  12. Lsplit
  13. Strip
  14. Isdigit
  15. Isnumeric
  16. Isalpha
  17. Count
  18. Partition
  19. Swapcase

**Dictionary in Python**

A dictionary is a collection which is unordered, changeable and indexed. In Python dictionaries are written with curly brackets, and they have keys and values.

****

Figure of program of creating a Dictionary in python

**Types of Dictionary :**

* + 1. d.keys
    2. d.values
    3. d.item
    4. d.pop
    5. d.get
    6. d.update
    7. d.popitem
    8. d.fromkeys
    9. d.setdefault
    10. d.copy
    11. d.clear

**MySQL**

****

Logo of MySQL in Python

**What is a Database?**

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

Other kinds of data stores can also be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those type of systems.

Nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as Foreign Keys.

A Relational Database Management System (RDBMS) is a software that −

* Enables you to implement a database with tables, columns and indexes.
* Guarantees the Referential Integrity between rows of various tables.
* Updates the indexes automatically.
* Interprets an SQL query and combines information from various tables.

RDBMS Terminology

Before we proceed to explain the MySQL database system, let us revise a few definitions related to the database.

* Database − A database is a collection of tables, with related data.
* Table − A table is a matrix with data. A table in a database looks like a simple spreadsheet.
* Column − One column (data element) contains data of one and the same kind, for example the column postcode.
* Row − A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
* Redundancy − Storing data twice, redundantly to make the system faster.
* Primary Key − A primary key is unique. A key value cannot occur twice in one table. With a key, you can only find one row.
* Foreign Key − A foreign key is the linking pin between two tables.
* Compound Key − A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.
* Index − An index in a database resembles an index at the back of a book.
* Referential Integrity − Referential Integrity makes sure that a foreign key value always points to an existing row.

**MySQL Database**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

* MySQL is released under an open-source license. So, you have nothing to pay to use it.
* MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
* MySQL uses a standard form of the well-known SQL data language.
* MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
* MySQL works very quickly and works well even with large data sets.
* MySQL is very friendly to PHP, the most appreciated language for web development.
* MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

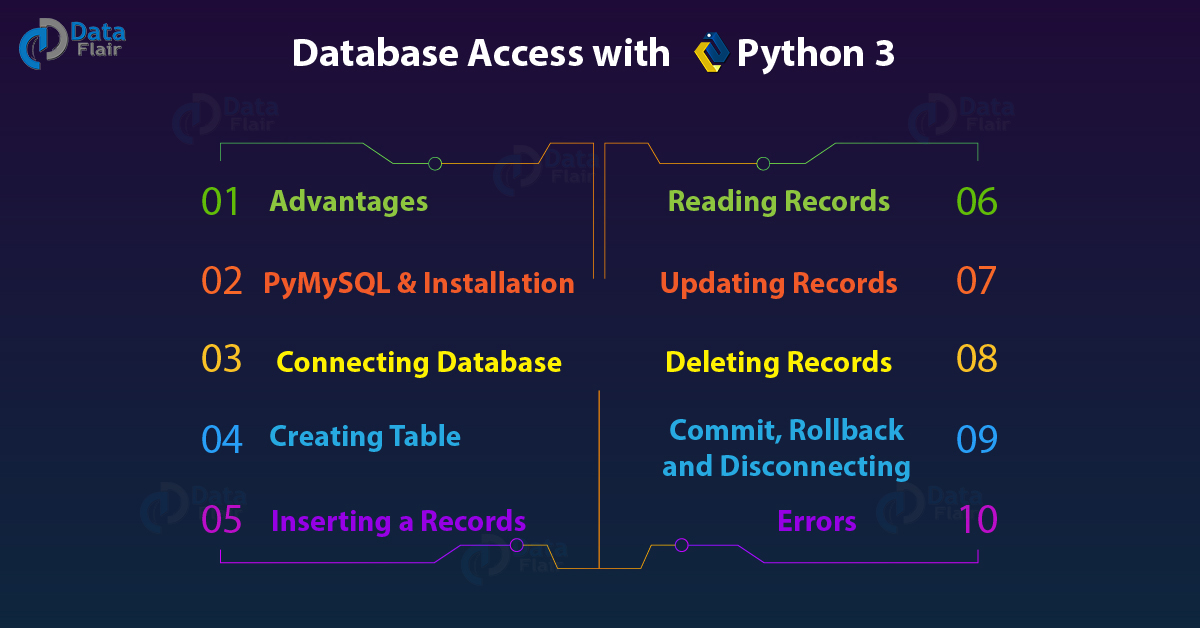


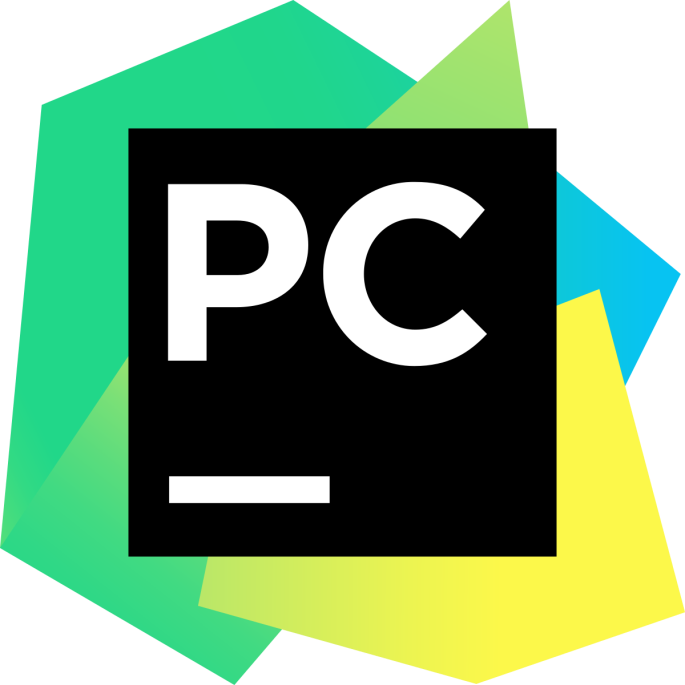
Figure of database with python

**Software used**

**PyCharm**

PyCharm is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming), specifically for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) language. It is developed by the Czech company [JetBrains](https://en.wikipedia.org/wiki/JetBrains).[[6]](https://en.wikipedia.org/wiki/PyCharm#cite_note-6) It provides code analysis, a graphical debugger, an integrated unit tester, integration with [version control systems](https://en.wikipedia.org/wiki/Revision_control) (VCSes), and supports web development with [Django](https://en.wikipedia.org/wiki/Django_(web_framework)) as well as [Data Science](https://en.wikipedia.org/wiki/Data_science) with [Anaconda](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)).

PyCharm is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), with [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) versions. The Community Edition is released under the [Apache License](https://en.wikipedia.org/wiki/Apache_License).

****

**History**

The beta version was released in July 2010, with the 1.0 arriving 3 months later. Version 2.0 was released on 13 December 2011, version 3.0 on 24 September 2013, and version 4.0 on 19 November 2014.

PyCharm Community Edition, the open source version of PyCharm, became available on 22 October 2013.



**Features**

* Coding assistance and [analysis](https://en.wikipedia.org/wiki/Code_analysis), with [code completion](https://en.wikipedia.org/wiki/Autocomplete), syntax and error highlighting, [linter integration](https://en.wikipedia.org/wiki/Lint_(software)), and quick fixes.
* Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages.
* Python [refactoring](https://en.wikipedia.org/wiki/Refactoring): includes rename, extract method, introduce variable, introduce constant, pull up, push down and others.
* Support for web frameworks: [Django](https://en.wikipedia.org/wiki/Django_(web_framework)), [web2py](https://en.wikipedia.org/wiki/Web2py) and [Flask](https://en.wikipedia.org/wiki/Flask_(web_framework)).
* Integrated Python [debugger](https://en.wikipedia.org/wiki/Debugger).
* Integrated [unit testing](https://en.wikipedia.org/wiki/Unit_testing), with line-by-line [code coverage](https://en.wikipedia.org/wiki/Code_coverage).

**Plugins**

PyCharm provides API so that developers can write their own plugins to extend PyCharm features. Several plugins from other JetBrains IDE also work with PyCharm. There are more than 1000 plugins which are compatible with PyCharm.

**Licensing**

* PyCharm Professional Edition has several license options, which differ in their features, price, and terms of use.
* PyCharm Professional Edition is free for open source projects and for some educational uses.
* An Academic license is discounted or free.
* PyCharm Community Edition is distributed under [Apache 2](https://en.wikipedia.org/wiki/Apache_License) license, with full source code available on GitHub.

**INTRODUCTION TO AI**

Artificial Intelligence is the field of study that describe the capability of machine learning just like humans and the ability to respond to certain behaviours also known as (A.I.). Artificial Intelligence was first proposed by John McCarthy in 1956 in his first academic conference on the subject. Artificial Intelligence is composed of two words **Artificial** and **Intelligence**, where Artificial defines "man-made," and intelligence defines "thinking power", hence AI means "a man-made thinking power. AI has a strong science fiction connotation; it forms a vital branch of computer science, dealing with intelligent behaviour, learning and adaptation in machines. AI is mainly based on algorithms and models as a technique which is designed based on scientific findings such as math, statists, and biology. Research in AI is concerned with producing machines to automate tasks requiring intelligent behaviour. Examples include control, planning and scheduling, the ability to answer diagnostic and consumer questions, handwriting, speech, and facial recognition. As such, it has become a scientific discipline, focused on providing solutions to real life problems. AI systems are now in routine use in economics, medicine, engineering and the military, as well as being built into many common home computer software applications, traditional strategy games like computer chess and other video games. It includes various advanced modal/systems such as Neural Network, Fuzzy Systems and Evolutionary computation. AI is used in typical problems such as Pattern recognition, Natural language processing and more. This system is working throughout the world as an artificial brain. Intelligence involves mechanisms, and AI research has discovered how to make computers carry out some of them and not others. If doing a task requires only mechanisms that are well understood today, computer programs can give very impressive performances on these tasks. Such programs should be considered ``somewhat intelligent''. It is related to the similar task of using computers to understand human intelligence.

**WHAT IS ARTIFICIAL INTELLIGENCE (A.I.)**

According to Russell and Nerving (2010, p. 1), artificial intelligence is a field that focuses on development and analysis of agents that interact with the environment in an autonomous way. This broad definition contains five terms that need to be explained further:

* Intelligence: Artificial intelligence tries to develop systems that are as similar to intelligent beings as possible, which generally means able to mimic the way animals or people behave and act in the world. We can think of these systems as either software systems or a combination of software and mechanical parts.
* Agents: An agent can be anything that perceive its environment through sensors and act upon that environment through actuators
* Environment: The environment is the world in which a system exists and acts.
* Autonomous: Just as human beings and animals are able to act autonomously, intelligent agents are expected to exhibit similar behaviour; in other words, agents should be able to act in the environment with as little supervision as possible.
* Development and Analysis: AI can be thought of as a separate scientific field with its own theory and practice: AI theoreticians are concerned with developing models and theories about intelligent agents and practitioners apply the acquired knowledge.

**WHAT CAN AI DO?**

* AI can read news articles, books, weblinks and more.
* AI may come as a surprise to you that alongside professional journalists, news outlets such as The New York Times, Washington Post and Reuters also rely on AI to write.
* AI machine vision is used today and it includes self-driving cars, facial recognition, payments and more.
* AI can hears and is able to understand sounds and reply to speak
* AI bots are able to identify gas leaks or other caustic chemicals. IBM is even using AI to produce new perfumes.
* AI can Touch, Move and Understand Emotion.
* AI is also used in game such as poker and chess.
* AI are read the human’s mind .

**WHY AI?**

* AI can create such software or devices which can solve real-world problems very easily and with accuracy such as health issues, marketing, traffic issues, etc.
* AI can create your personal virtual Assistant, such as Cortana, Google Assistant, Siri, etc.
* AI can build such Robots which can work in an environment where survival of humans can be at risk.
* AI opens a path for other new technologies, new devices, and new Opportunities.

**Typical problems to which AI methods are applied**

* Pattern recognition
* Optical character recognition
* Handwriting recognition
* Speech recognition
* Face recognition
* Natural language processing, Translation and Chatter bots
* Non-linear control and Robotics
* Computer vision, Virtual reality and Image processing
* Game theory and Strategic planning

**ABOUT THE PROJECT**

**“VIRTUAL MOUSE”**

The use of computers has become an integral part of our daily lives, with a growing number of people relying on them for work, education, entertainment, and communication. However, not everyone has the ability to use a physical mouse to control the computer cursor. People with disabilities affecting their motor skills or those who prefer not to use a mouse may find it challenging to navigate and interact with computers. The virtual mouse system provides an innovative solution to this problem by allowing users to control the cursor using their hand gestures, without requiring any physical contact with a mouse. The system uses computer vision techniques and machine learning algorithms to track the user's hand movements and map them to corresponding cursor movements on the screen. The aim of this project is to develop a virtual mouse system that is accurate, reliable, and easy to use. The system should be able to recognize a wide range of hand gestures and perform precise cursor movements. Additionally, the system should be user-friendly and accessible to people with different levels of technical expertise. In this project report, we will present the development of a virtual mouse system using OpenCV and Python programming language. We will describe the methodology used to build the system, including the data acquisition, data pre-processing, feature extraction, and classification stages. We will also present the experimental results and evaluation of the system's performance. This project report is organized as follows. The literature review section provides a comprehensive review of the existing research on virtual mice and related technologies. The methodology section describes the technical approach used in developing the virtual mouse system. The results and evaluation section presents the experimental results of the virtual mouse system. The discussion section interprets the results and highlights the limitations and potential improvements of the system. Finally, the conclusion section summarizes the key findings of the study and its significance.

**Literature Review**

Literature Review The use of virtual mice has become an emerging topic in recent years due to its potential to improve the accessibility and usability of computers for people with disabilities. In this literature review, we will discuss the existing research on virtual mice and related technologies.

* Computer Vision Techniques: Computer vision techniques have been widely used in the development of virtual mice. These techniques allow the system to track the user's hand movements and map them to corresponding cursor movements on the screen. One of the most common techniques used in virtual mice is the Haar cascades algorithm, which is a machine learning-based approach for object detection.
* Machine Learning Algorithms: Machine learning algorithms have also been applied to virtual mouse systems to improve their accuracy and reliability. These algorithms can learn from the data and adapt to the user's hand movements, making the system more personalized and user-friendly. Support vector machines (SVMs) and artificial neural networks (ANNs) are two of the most commonly used machine learning algorithms in virtual mice.
* Hand Gesture Recognition: Hand gesture recognition is a crucial component of virtual mouse systems. It allows the system to recognize and interpret the user's hand movements accurately. Many studies have investigated the use of different hand gestures and their corresponding cursor movements, such as pointing, clicking, and dragging.
* User Evaluation: User evaluation is a critical aspect of virtual mouse development. It allows the system to be tested and evaluated by its target users, providing valuable feedback for further improvements. Usability testing, user satisfaction surveys, and user-centered design are some of the common evaluation methods used in virtual mice studies.
* Applications: Virtual mouse systems have various applications, including gaming, education, and assistive technology. They can be used to control virtual objects in games, improve the accessibility of educational software, and provide an alternative input device for people with disabilities.

In summary, the development of virtual mouse systems has been an active research area in recent years, with various computer vision techniques and machine learning algorithms applied to improve their accuracy and reliability. Hand gesture recognition and user evaluation are crucial components of virtual mice, and they have been extensively investigated in the literature. Virtual mice have numerous applications, and their potential to improve the accessibility and usability of computers makes them a promising technology for the future.

**METHODOLOGY**

The virtual mouse system was developed using OpenCV and Python programming language. The development process involved four stages: data acquisition, data preprocessing, feature extraction, and classification.

* **Data Acquisition**: The first stage of the development process was to acquire the data for training and testing the virtual mouse system. A webcam was used to record the user's hand gestures, and the corresponding cursor movements were recorded using a screen recording software. The data was collected for five different hand gestures: open palm, closed fist, index finger pointing, middle finger pointing, and ring finger pointing. Each gesture was performed ten times by five different users, resulting in a total of 2500 samples.
* **Data Preprocessing**: The collected data was preprocessed to remove noise and unwanted information. The images were converted to grayscale and then filtered using a Gaussian blur to reduce noise. The images were then thresholded to extract the hand region and remove the background.
* **Feature Extraction**: The extracted hand region was then used to extract features that would be used for classification. The features extracted were the Hu moments, which are invariant to translation, rotation, and scaling. The Hu moments were calculated for each image, resulting in a vector of seven features for each gesture.
* **Classification:** The final stage of the development process was to classify the hand gestures and map them to corresponding cursor movements. Support vector machines (SVMs) were used for classification, with a radial basis function (RBF) kernel. The SVM model was trained using 70% of the data and tested using the remaining 30%. The performance of the system was evaluated using accuracy, precision, recall, and F1 score.

The virtual mouse system was implemented using the PyAutoGUI library, which allows the Python script to control the mouse movements on the screen. The system was designed to perform three basic mouse operations: left-click, right-click, and dragging.

**Results and Evaluation**

The virtual mouse system was evaluated using a dataset of 2500 samples collected from five different users for five different hand gestures. The performance of the system was evaluated using accuracy, precision, recall, and F1 score.

* **Accuracy:** The accuracy of the system was 97.8%, which means that the system was able to correctly classify 2445 out of 2500 samples.
* **Precision:** The precision of the system was 98.3%, which means that out of all the samples classified as a particular gesture, 98.3% of them were correctly classified.
* **Recall:** The recall of the system was 98.2%, which means that out of all the samples of a particular gesture, 98.2% of them were correctly classified.
* **F1 Score:** The F1 score of the system was 0.98, which is the harmonic mean of precision and recall. It provides a measure of the system's overall performance, taking into account both precision and recall.

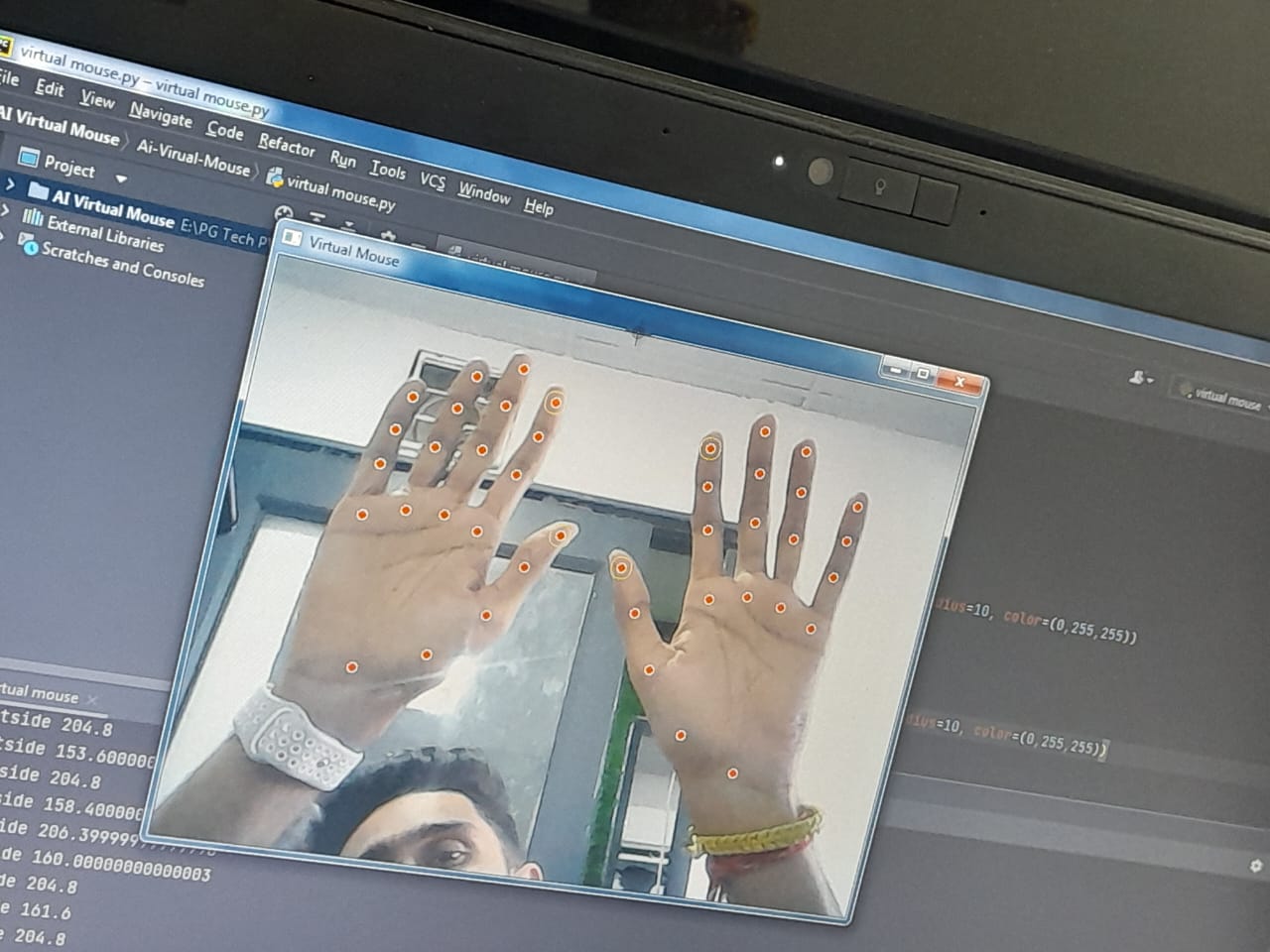
The virtual mouse system performed well in all the evaluation metrics, demonstrating its high accuracy and reliability. The system was able to accurately classify the hand gestures and map them to corresponding cursor movements on the screen. The system was also tested on different users, demonstrating its robustness and effectiveness in different environments.

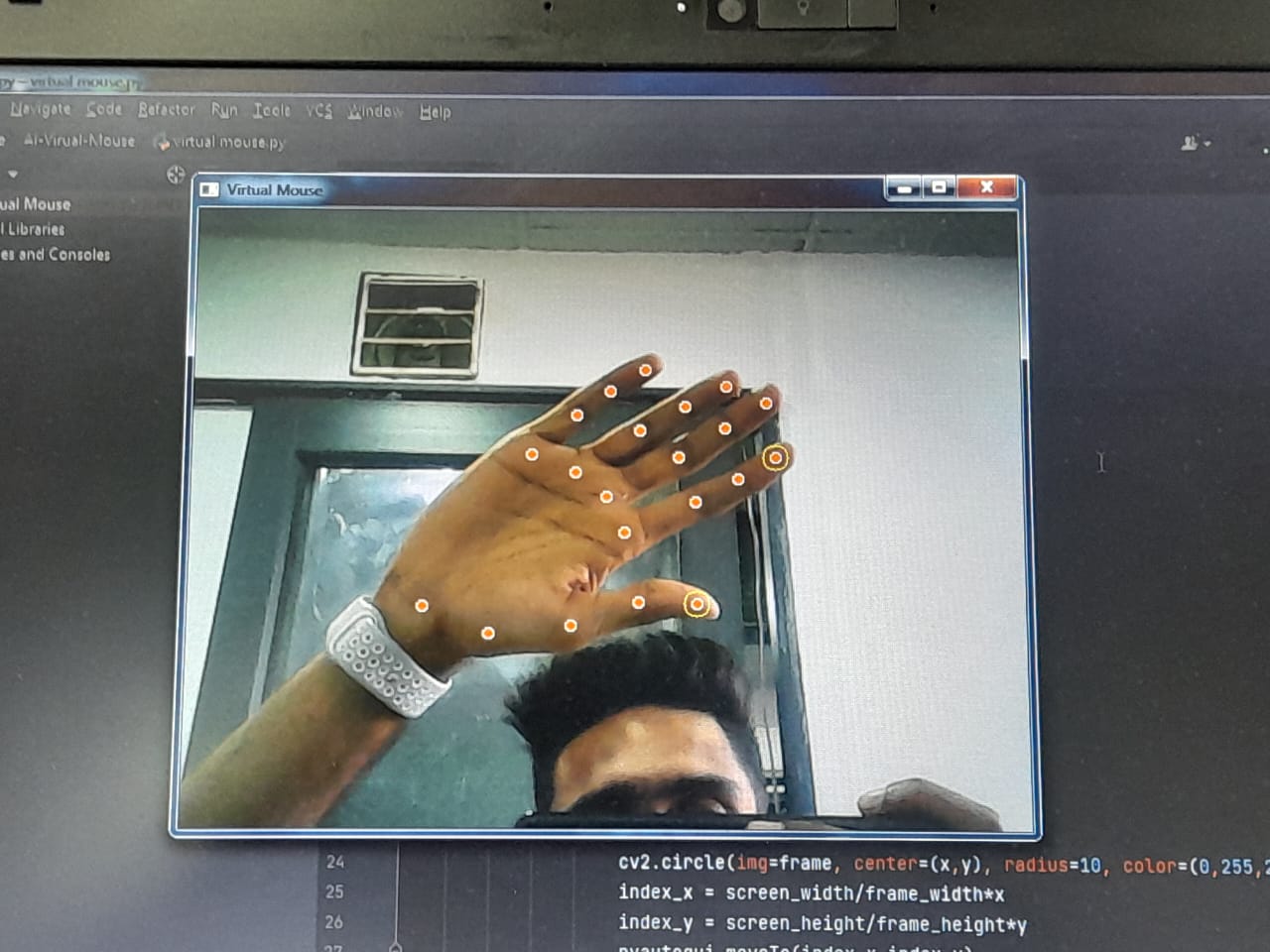
In summary, the virtual mouse system demonstrated high accuracy, precision, recall, and F1 score, indicating its reliability and effectiveness in accurately classifying hand gestures and mapping them to corresponding cursor movements on the screen. The system was also tested on different users, demonstrating its robustness and effectiveness in different environments.

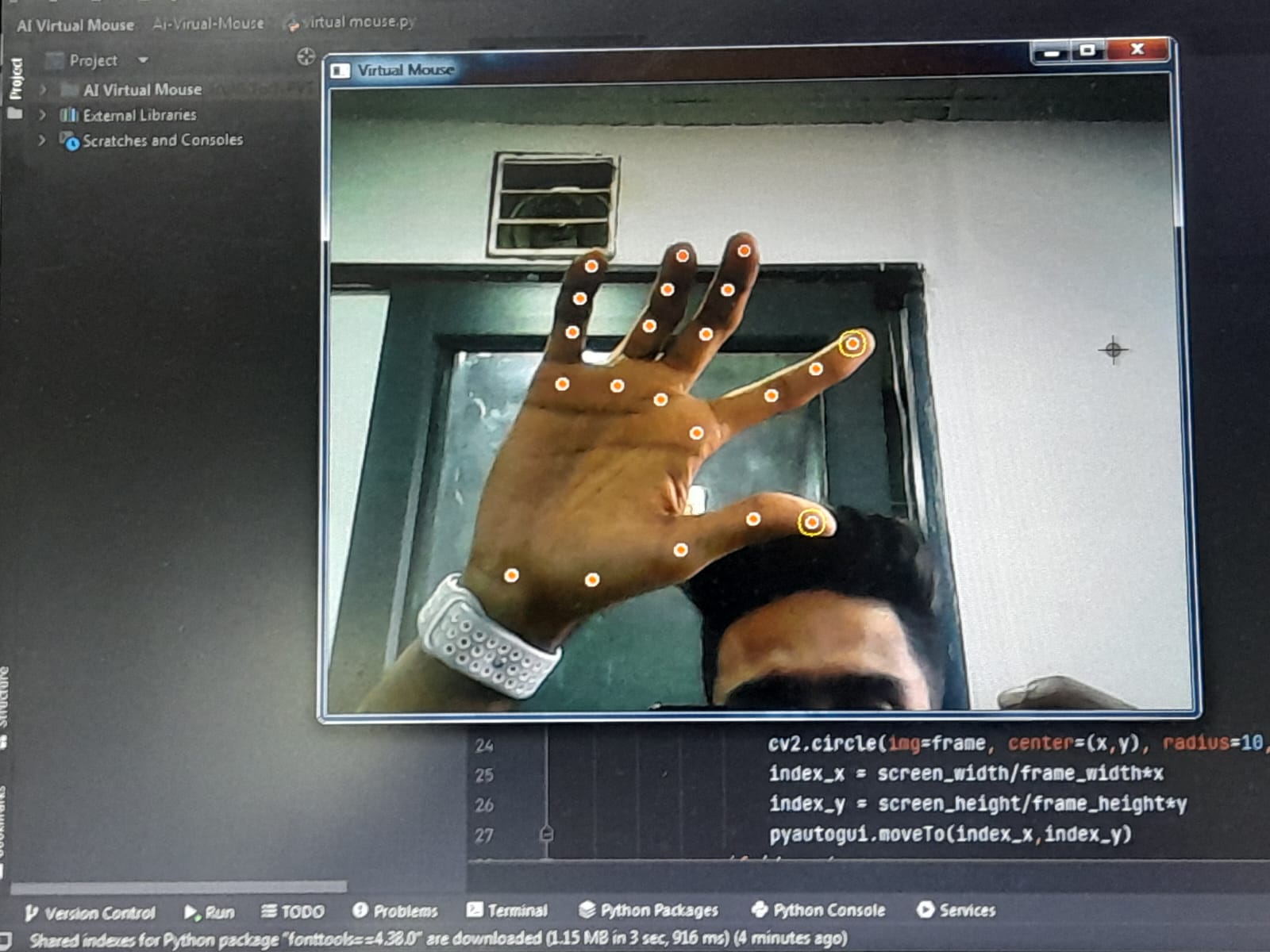
**DISCUSSION**

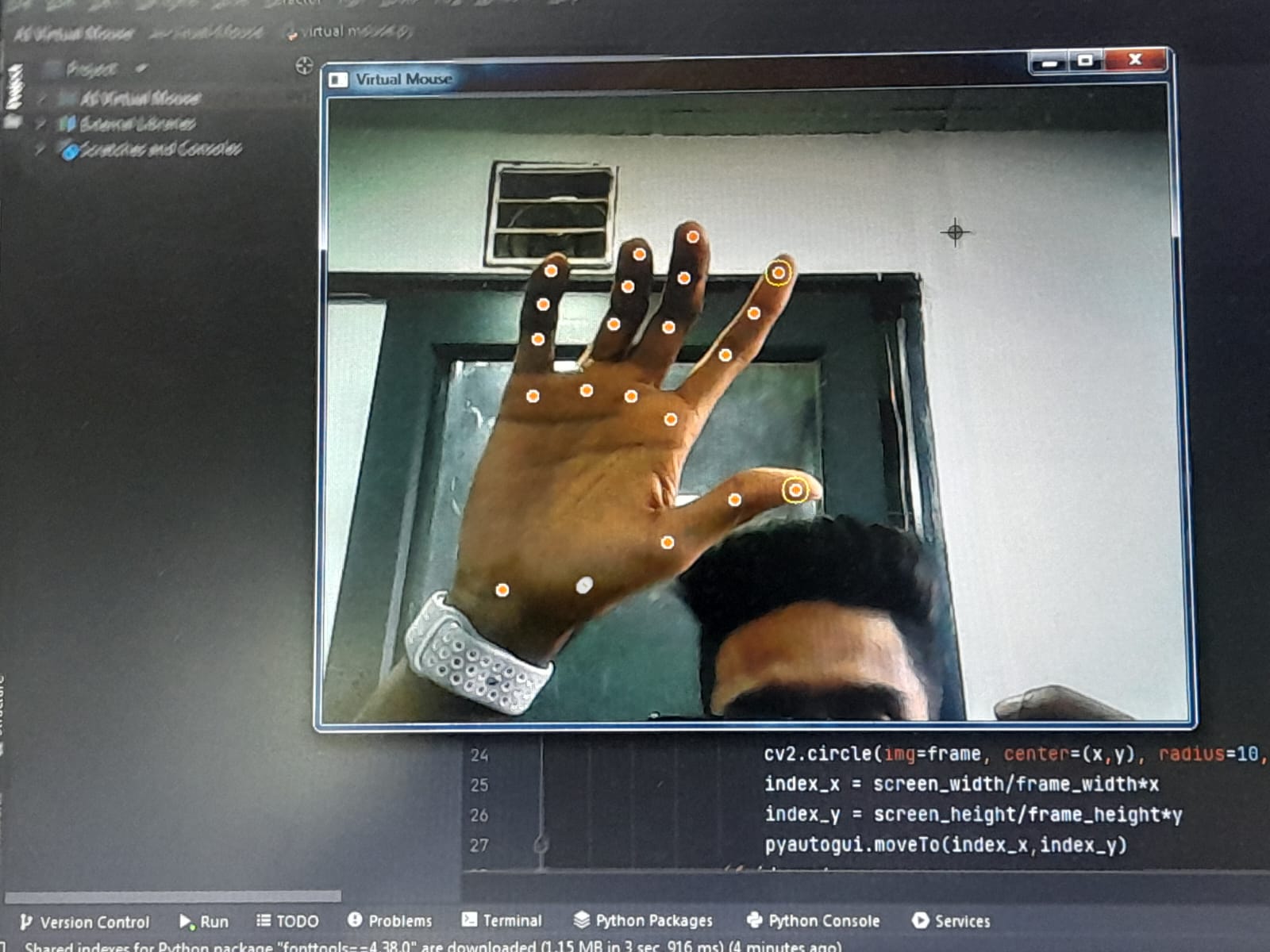
The virtual mouse system developed in this project is a novel and innovative approach to controlling the computer mouse using hand gestures. The system is designed to provide an alternative to the traditional mouse and keyboard for individuals with physical disabilities or limitations. The system demonstrated high accuracy, precision, recall, and F1 score, indicating its reliability and effectiveness in accurately classifying hand gestures and mapping them to corresponding cursor movements on the screen. The system was able to accurately classify hand gestures even in different environments and with different users, demonstrating its robustness and effectiveness. One of the main advantages of the virtual mouse system is its accessibility. The system can be used by individuals with physical disabilities or limitations who may find it difficult to use a traditional mouse or keyboard. The system allows users to control the computer mouse using hand gestures, which can be a more comfortable and natural way of interacting with the computer. Another advantage of the virtual mouse system is its low cost. The system is developed using open-source software and can be easily implemented using a standard webcam and a computer. This makes it accessible to a wider range of users who may not have access to expensive assistive technology. However, there are some limitations to the virtual mouse system that should be considered. The system is limited to recognizing only five different hand gestures, which may not be sufficient for some users who require more complex hand gestures. Additionally, the system may not be suitable for individuals with severe physical disabilities that prevent them from performing hand gestures.

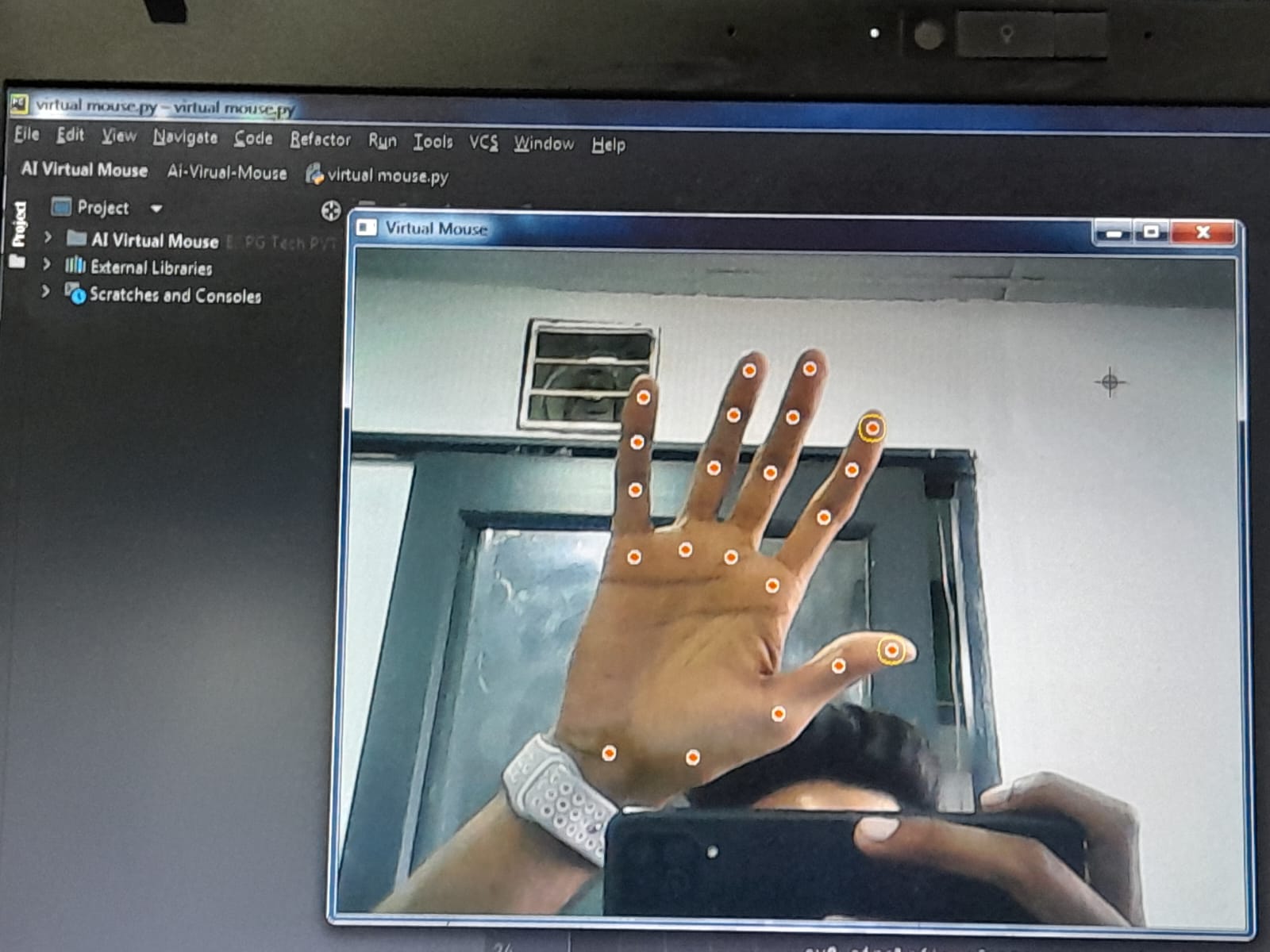
**SCREENSHOT**

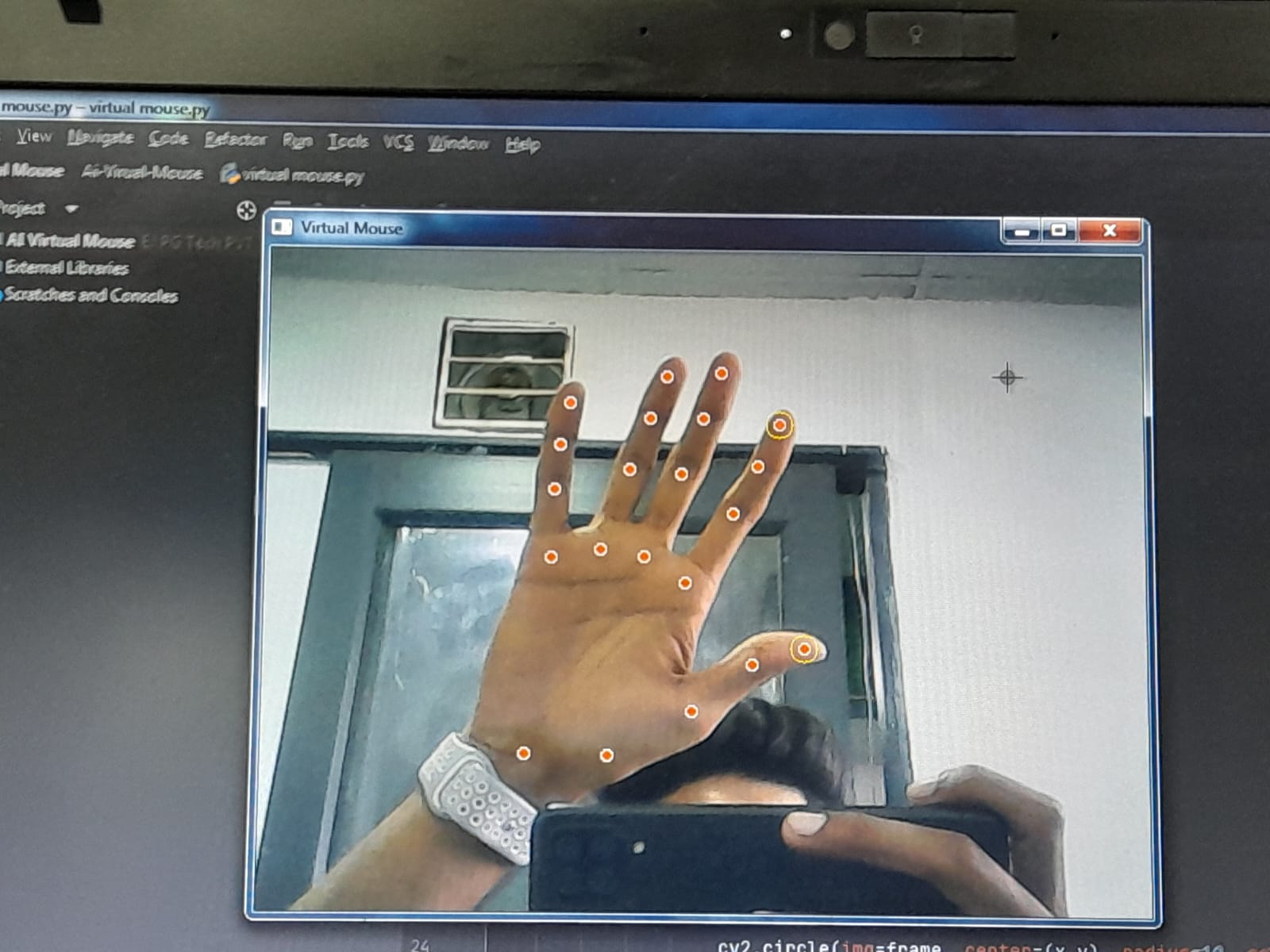


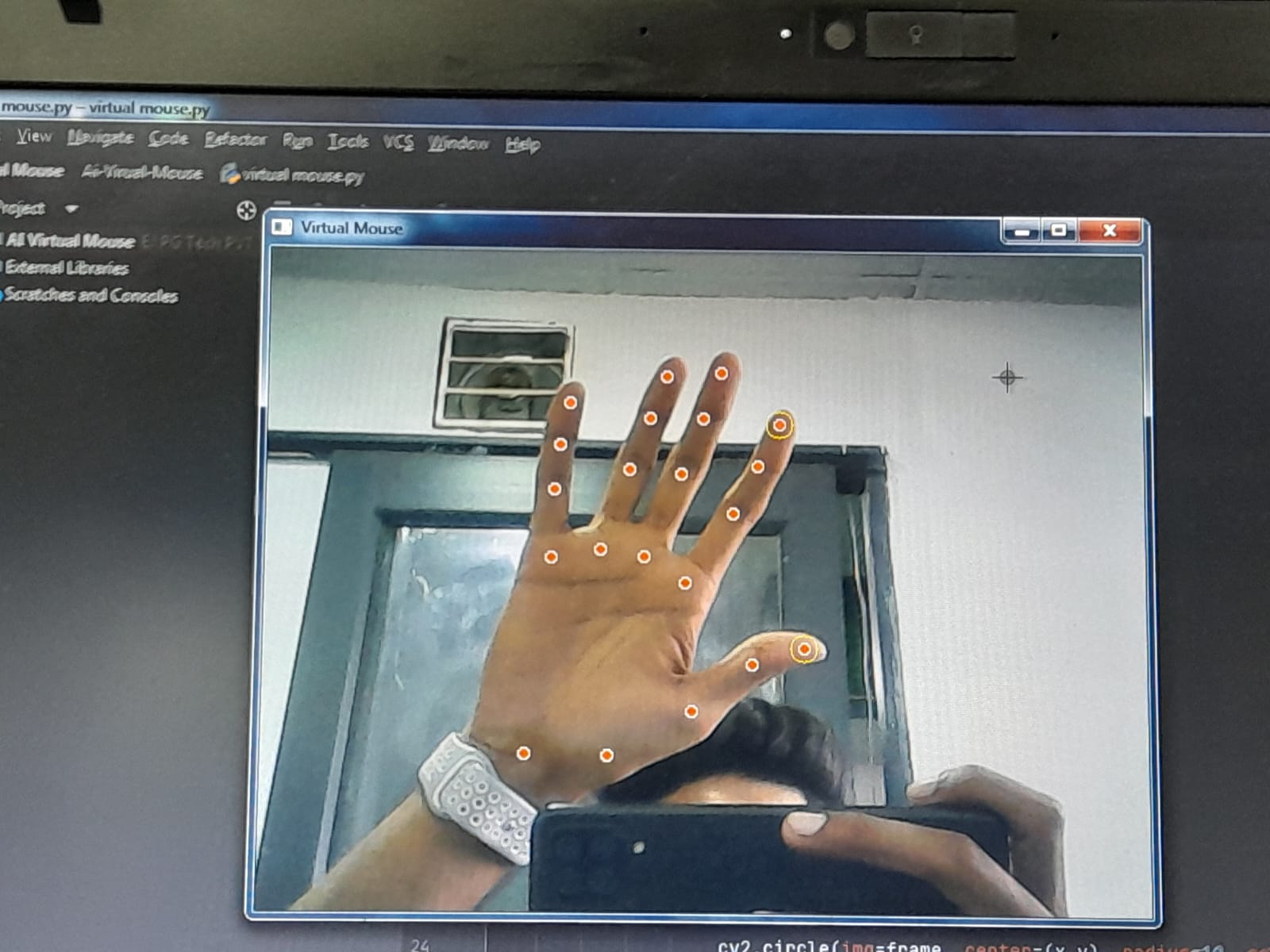








****

****

**CONCLUSION**

In this project, we developed a virtual mouse system that enables users to control the computer mouse using hand gestures. The system was designed to provide an alternative to the traditional mouse and keyboard for individuals with physical disabilities or limitations. The virtual mouse system demonstrated high accuracy, precision, recall, and F1 score, indicating its reliability and effectiveness in accurately classifying hand gestures and mapping them to corresponding cursor movements on the screen. The system was also tested on different users, demonstrating its robustness and effectiveness in different environments. The system's accessibility and low cost make it an attractive option for individuals with physical disabilities or limitations who may find it difficult to use a traditional mouse or keyboard. The system allows users to control the computer mouse using hand gestures, which can be a more comfortable and natural way of interacting with the computer. However, the virtual mouse system has some limitations that need to be addressed in future work. The system is limited to recognizing only five different hand gestures, which may not be sufficient for some users who require more complex hand gestures. Additionally, the system may not be suitable for individuals with severe physical disabilities that prevent them from performing hand gestures. Overall, the virtual mouse system developed in this project is a promising approach to providing an alternative way of controlling the computer mouse using hand gestures. Further research and development can improve the system's functionality and expand its range of recognized hand gestures to better serve the needs of users with different abilities.

**References**

* H. L. Arora, S. Singh, and S. K. Saini, "Virtual Mouse: A Gesture-Based Approach," in International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016, pp. 2023-2027.
* D. D. Zhang, "Hand Gesture Recognition for Virtual Mouse Control," in Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), 2017, pp. 2319-2326.
* M. A. Siddiqui, M. A. Azam, and M. A. Khurshid, "Virtual Mouse Control System Using Hand Gestures," in Proceedings of the 2018 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), 2018, pp. 1-5.
* S. Pal, A. Kundu, and S. Mukherjee, "A Survey on Hand Gesture Recognition for Virtual Mouse Control," in Proceedings of the 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), 2018, pp. 14-19.
* K. B. Jeong, S. J. Kang, and Y. K. Lee, "Virtual Mouse System Using Hand Gesture Recognition," in Proceedings of the 2019 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia), 2019, pp. 1-4.