
Software Requirements Specification

for

AI VIRTUAL MOUSE

Prepared by

ADITI SINHA

SHIVANSH SINGH

RAUNAK JAHAN

ANUDARSHI DUTTA

DATE : 24.03.2023

Table of Contents

Table of Contents	1
1. Introduction	2
1.1 Purpose	2
1.2 Product Scope	2
1.3 References	3
2. Overall Description	3
2.1 Product Perspective	3
2.2 Product Functions	3
2.3 User Classes and Characteristics	3
2.4 Operating Environment	4
2.5 User Documentation	4
2.6 Assumptions and Dependencies	4
3. External Interface Requirements	6
3.1 User Interfaces	Error! Bookmark not defined.
3.2 Operating System Interfaces	Error! Bookmark not defined.
3.3 Hardware Interfaces	6
4. System Features	6
4.1 Gesture Recognition	4
4.2 Cursor Movement	4
4.3 Mouse Click	4
4.4 Calibration	4
5. Other Nonfunctional Requirements	7
6. Other Requirements	8

1. Introduction

With the development technologies in the areas of augmented reality and devices that we use in our daily life, these devices are becoming compact in the form of Bluetooth or wireless technologies. This project develops an AI virtual mouse system that makes use of the hand gestures and hand tip detection for performing mouse functions in the computer using computer vision.

1.1 Purpose

A software programme called the AI virtual mouse system is to develop an alternative to the regular and traditional mouse system to perform and control the mouse functions and this can be achieved by capturing hand gestures and enables users to control their computer's cursor function using hand motions for that are photographed by a camera such as left click, right click, and scrolling function.

The system recognizes hand gestures and maps them to cursor movements using machine learning methods that are implemented in Python. The purpose of the project is to develop a software application that enables users to control their computer cursor through hand gestures captured by a camera. The system will use machine learning algorithms implemented in Python to recognize hand gestures and map them to cursor movements.

1.2 Product Scope

The AI virtual mouse will use computer vision technology to track the user's hand movements and gestures to simulate mouse movements and clicks.

- Object detection: Use computer vision techniques to detect objects, including hands or other objects, that are used to control the virtual mouse.
- Hand tracking: Track the movement of hands or other objects and use that information to control the virtual mouse.
- Gesture recognition: Recognize different hand gestures and use them to perform specific actions such as left-click, right-click, scroll, etc.
- Machine learning-based prediction: Use machine learning algorithms to predict the user's intent and improve the accuracy and responsiveness of the virtual mouse.

1.3 References

OpenCV official documentation: <https://docs.opencv.org/>.

MediaPipe documentation: <https://google.github.io/mediapipe/>. AutoPy documentation: <https://pypi.org/project/autopy/>.

NumPy documentation: <https://numpy.org/doc/stable/>.

2. Overall Description

2.1 Product Perspective

Users with physical or other disabilities who find it difficult to use a conventional mouse and keyboard may be able to use an AI virtual mouse as an alternative input method. This might aid in improving computing accessibility for a broader variety of users.

By eliminating the need for actual mouse motions, an AI virtual mouse may enable users to use their computer more effectively. This could be especially helpful for precise jobs like graphic design or video editing. When travelling or using a cramped laptop keyboard, for example, using a physical mouse may not be feasible or handy. In these circumstances, an AI virtual mouse may be used. By doing so, productivity might increase and customer annoyance might decrease.

Overall, an AI virtual mouse using Python and ML could provide a range of benefits for users, including increased accessibility, efficiency, customization, convenience, and innovation.

2.2 Product Functions

The main function of an AI virtual mouse using AI and ML would be to provide an alternative input method for users who have difficulty using traditional mouse and keyboard inputs due to physical or other disabilities. The virtual mouse would use machine learning algorithms to track the user's hand movements and translate them into virtual mouse movements on the computer screen. Some key functions of an AI virtual mouse using AI and ML are Hand tracking , Gesture recognition, Calibration , Accessibility.

2.3 User Classes and Characteristics

People with physical disabilities: Users with physical disabilities such as limited mobility, tremors, or muscular dystrophy could benefit from an AI virtual mouse as it would provide an alternative input method that does not require physical dexterity or fine motor skills.

Students and researchers: Users who need to perform repetitive tasks or navigate large amounts of data could benefit from an AI virtual mouse. By using machine learning algorithms to automate tasks or improve navigation, an AI virtual mouse could save time and improve productivity.

Older adults: Older adults who may not be familiar with traditional mouse and keyboard inputs could benefit from an AI virtual mouse, as it provides a more intuitive input method that relies on natural hand movements.

Gamers: Gamers require fast and responsive input methods for gameplay, and an AI virtual mouse could provide customizable settings that allow them to optimize their gameplay experience.

2.4 Operating Environment

Hardware requirements: The hardware requirements for an AI virtual mouse would depend on the type of input method used, such as a camera or a motion sensor. The hardware would need to be capable of capturing hand movements accurately and quickly.

Operating system: The AI virtual mouse would need to be compatible with the operating system of the computer or device it is being used on. This would include compatibility with Windows, MacOS, and Linux.

Software libraries: An AI virtual mouse using Python and ML would rely on software libraries such as OpenCV, Media-pipe, and Autopy to perform hand tracking, gesture recognition, and virtual mouse movements. These libraries would need to be installed and configured on the computer or device.

Security and privacy: The AI virtual mouse would need to be designed with security and privacy considerations in mind, as it would potentially be capturing and transmitting user data. Encryption and o

2.5 User Documentation

Installation guide: Provide step-by-step instructions on how to install the necessary software libraries and drivers needed to use the AI virtual mouse. Include screenshots and links to resources if possible.

Configuration guide: Explain how to configure the AI virtual mouse to fit the user's preferences. This could include settings such as mouse sensitivity, tracking speed, and gesture recognition.

Usage guide: Provide clear instructions on how to use the AI virtual mouse, including how to move the virtual mouse, click and drag, and perform gestures. Include tips and tricks for optimizing the user's experience.

Troubleshooting guide: Provide solutions to common issues that users may encounter, such as tracking errors or software conflicts. Include detailed instructions on how to resolve these issues, and provide resources for further assistance if necessary.

Security and privacy guide: Explain how the AI virtual mouse handles user data, and provide information on how to protect sensitive information.

Frequently Asked Questions (FAQs): Include a list of frequently asked questions and answers to help users quickly find solutions to common issues.

2.6 Assumptions and Dependencies

Assumptions:

The required tools and software for the AI virtual mouse are accessible to users on a computer or other device.

Users are familiar with the fundamentals of using a computer and software and are at ease performing simple repairs.

For hand tracking and gesture detection, users have good lighting.

Dependencies:

There are many useful software libraries, including OpenCV, Mediapipe, Autopy, and others.

A camera or motion sensor built into the computer or gadget can precisely detect hand movements.

The AI virtual mouse programme is compatible with the operating system of the computer or device.

3. External Interface Requirements

3.1 User Interface:

The AI virtual mouse software will have a user interface that will allow users to interact with the virtual mouse. The interface will include options to customize the settings of the virtual mouse, such as sensitivity, speed, and click behavior.

3.2 Operating System Interface:

The AI virtual mouse software will need to interact with the operating system to control the mouse pointer and perform mouse actions. This interface will be different for each operating system, and the software will need to be compatible with Windows, macOS, and Linux.

3.3 Hardware Interface:

The AI virtual mouse software will need to be compatible with different types of hardware, such as laptops and desktops. The software will also need to work with different types of input devices, such as keyboards and touch-pads.

4. System Features

4.1 Gesture Recognition :

The system shall recognize the following hand gestures:

- Moving the hand left/right to move the cursor left/right
- Moving the hand up/down to move the cursor up/down
- Making a fist to click the mouse button
- Opening the hand to release the mouse button
- Holding the hand steady for a certain period to perform a right-click

4.2 Cursor Movement :

The system shall move the cursor on the screen in response to the recognized gestures with the following characteristics:

- The cursor movement shall be smooth and precise
- The cursor speed shall be adjustable by the user
- The cursor movement shall be proportional to the hand movement

4.3 Mouse Click :

The system shall simulate the left and right mouse clicks using the recognized gestures as specified

4.4 Calibration :

The system shall provide a calibration step for each user to adjust the gesture recognition parameters to their specific hand size and movements.

5. Other Nonfunctional Requirements

● Performance:

To give users a seamless and effective experience, the AI virtual mouse should be quick to respond and have minimal latency.

● Usability:

The AI virtual mouse should have an intuitive UI that is simple to learn and use and doesn't call for specialized training.

● Reliability:

To ensure that users can depend on the AI virtual mouse for their tasks, it must operate consistently and precisely without crashing or malfunctioning.

● Scalability:

The AI virtual mouse should be able to manage a high volume of users or traffic without experiencing any performance degradation.

● Security:

To secure user data and stop unauthorised access, the AI virtual mouse should be built with robust security features like encryption and authentication.

● Maintainability:

To reduce downtime and make sure it remains current with technology, the AI virtual mouse should be simple to maintain and update, with a modular design and clear documentation.

6. Other Requirements

- **Accuracy:** The AI virtual mouse should follow hand movements accurately and reliably before converting them to mouse movements on the screen. The virtual mouse should be simple and precise for users to manipulate with their hands.
- **Speed:** There should be little to no lag or delay between the user's movements and the matching mouse movements on the screen. The AI virtual mouse should react rapidly to hand movements. The customer experience will be seamless and easy thanks to this.
- **Security and privacy:** Security and privacy should be taken into consideration when designing the AI virtual cursor. To avoid theft or unauthorised access, user data should be secured and encrypted.
- **Compatibility:** The AI virtual mouse ought to work with a variety of hardware setups and running systems. Users will be able to use the virtual mouse on their chosen device and operating system thanks to this.