REAL TIME VIRTUAL AI MOUSE

FINAL PROJECT REPORT

TEAM

Nikhil Pillai (20BEC0687)
Roopreet Saluja (20BEC0247)
Nida Ghani (20BCE2850)
Josh George Jacob (20BEC0304)

REAL TIME VIRTUAL AI MOUSE

I. INTRODUCTION :

The size of the commonplace devices we use has decreased as a result of developments in wireless or communication as well as augmented reality. By utilizing hand motions and hand tip detection, the suggested computerized vision-based artificially intelligent virtual system in this study mimics mouse capabilities. The main objective of the system is to eliminate the need for a traditional mouse device by performing mouse pointer, zoom in and zoom out capabilities using a webcam or a computer's built-in camera. By employing the computer's built-in camera to track the tip of a finger making a hand motion, the AI virtual mouse technology enables us to move the pointer and zoom simultaneously.
Using a bluetooth-enabled or wireless cursor requires additional hardware, such as a computer connection dongle and a power source for the device. However, in the study, a user moves his hands while using the computer's built-in camera to use the mouse. Python was used as the programming language to create this model. The model uses the MediaPipe package to track the tips of the hands, while the screeninfo, win32api, and win32con packages are used to navigate the computer's window screen and perform actions. left- and right-clicks, for example. The suggested model's results demonstrated a very high level of precision.

II. LITERATURE SURVEY :

The cursor location in **Paper [1]** is controlled manually, without the aid of any electrical machinery. In contrast, a variety of hand motions will be employed to carry out operations like clicking and dragging objects. To use the suggested approach, a camera will be all that is required for input. The two pieces of software needed to put the suggested approach into practice are Python and OpenCV. The system's display will show the camera's output so that a user can alter it. For configuring this system, Python will be used along with NumPy, math, and mouse.

Teaching techniques that rely on cutting-edge technological tools to improve communication and interaction between teachers and students are described in **Paper [2]**. We have provided a device that aims to be a virtual marking tool and has features akin to those of a mouse to aid in this. The existing Virtual Marker has been altered to serve as both a pointer for the mouse and a pointer that offers all of the functions of every mouse. In this study, we demonstrated a hardware version of a virtual mouse that, by making the existing "virtual marker" very responsive in real time, increases its effectiveness.

Paper [3] discusses the enhancement of relationships through the implementation of technologically reliant creative products in educational approaches. To aid in this, we have provided a product that aims to be a virtual marking tool and also includes mouse-like features that give it all of the mouse's capabilities. In this study, we propose a hardware version of a virtual pointer that outperforms the existing "virtual marker" in terms of performance by being extremely responsive in real time.

According to **Paper [4]**, current research on how people and machines interact has made great strides. In this project, we used a Human PC Interaction approach with the goal of including the use of a hardware mouse and manipulating the mouse coordinates with hand gestures and color recognition. Hand movements were captured using a camera and color recognition technology. We replaced the current hardware with gestures to connect with the computer system in an effort to remove barriers between humans and computers.

The goal of **Paper [5]** is to use gesture recognition in place of hardware like a mouse to move the cursor on the screen. Several technologies have been used recently to create a virtual mouse.

Problem Description: The proposed AI can employ virtual mouse solutions to address the issue in the real world when there isn't enough room to use a real mouse or a person has hand issues and can't use one. Utilizing the proposed AI virtual mouse is one way to solve these issues, even if hand gesture and hand tip detection are already used to manage the PC mouse's capabilities via a webcam or built-in camera.

Objective: The primary goal of the proposed system is to offer an alternative to the current standard mouse device for carrying out and controlling mouse operations. Using a webcam that recognizes hand movements and hand tips and then processes the captured frames to carry out the desired mouse operation, including left and right clicking, makes this possible.

III. THEORITICAL ANALYSIS :

The numerous scenarios and functions employed in the system are explained Figures .

The system's camera:
Based on screenshots made with a laptop or desktop computer, the proposed AI virtual mouse technology. When the Python-based video recording object is built using the OpenCV image processing library, as shown in Figure 3, the web camera will start recording video. Webcam frames are sent to the virtual AI system, which analyzes them.

Capturing and processing the video:
The camera that the AI virtual controller system uses to record each frame till the show is over. To locate the hands in the video frame by frame, the frames are transformed from BGR to RGB as shown in the following code.

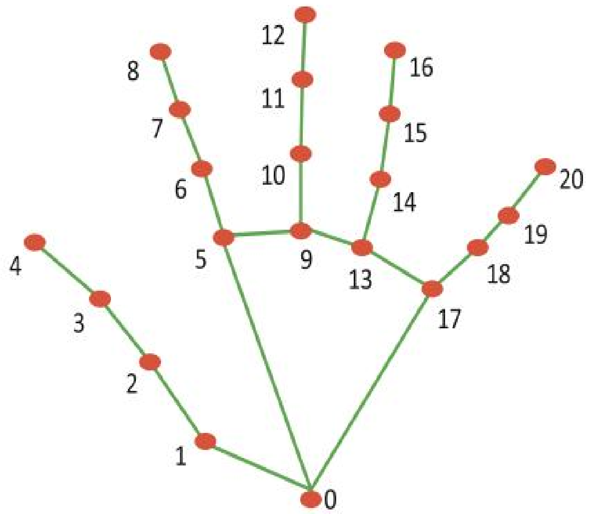


Figure 1: Landmarks or co-ordinates in the hand.

0. WRIST	11. MIDDLE_FINGER_DIP
1. THUMB_CMC	12. MIDDLE_FINGER_TIP
2. THUMB_MCP	13. RING_FINGER_MCP
3. THUMB_IP	14. RING_FINGER_PIP
4. THUMB_TIP	15. RING_FINGER_DIP
5. INDEX_FINGER_MCP	16. RING_FINGER_TIP
6. INDEX_FINGER_PIP	17. PINKY_MCP
7. INDEX_FINGER_DIP	18. PINKY PIP
8. INDEX_FINGER_TIP	19. PINKY DIP
9. MIDDLE FINGER MCP	20. PINKY TIP
10. MIDDLE FINGER	

Identifying which finger:
We can tell which finger is up right now by using the MediaPipe's tip Id for the specific finger we located and the related positions of the up fingers, as illustrated in Figure. Based on that conclusion, the indicated mouse function is subsequently carried out.

Computer vision is used to detect the tips of the hands and control the mouse based on gestures:
for the mouse pointer's movement within the computer's window. The AutoPy module in Python is used to move the mouse cursor around the computer screen if the pointed finger is up using tip Id 1 or if the other pointer finger and the center finger are up using tip Id 2. to make a left mouse button click. if the thumb on Id 0 and the first finger on tip 1 are both up.To reach the right button, simply click the mouse once. if the tips of the index and middle fingers, or Ids 1 and 2, are elevated simultaneously.

Figure 2: Video recording using the webcam (computer vision).

Figure 3: Detection of which finger is up

IV. FLOWCHART :

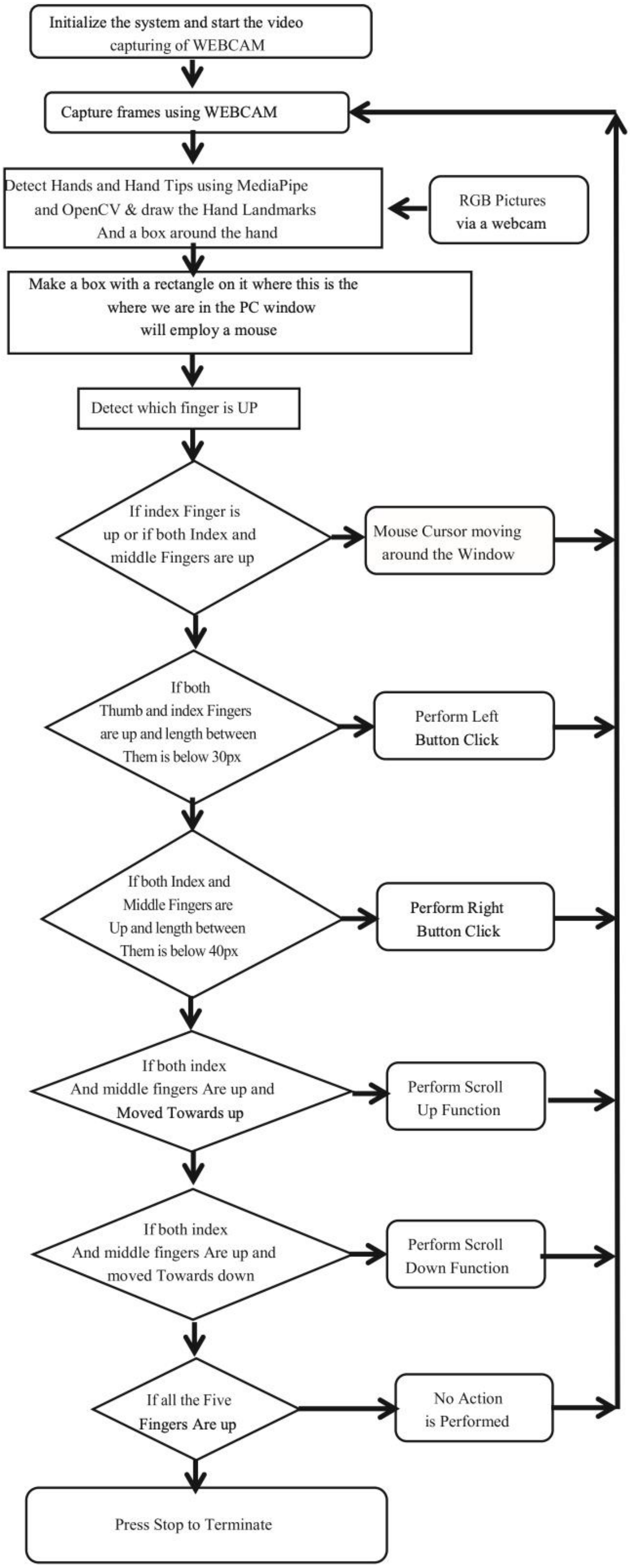


Figure 4: Virtual mouse system flowchart.

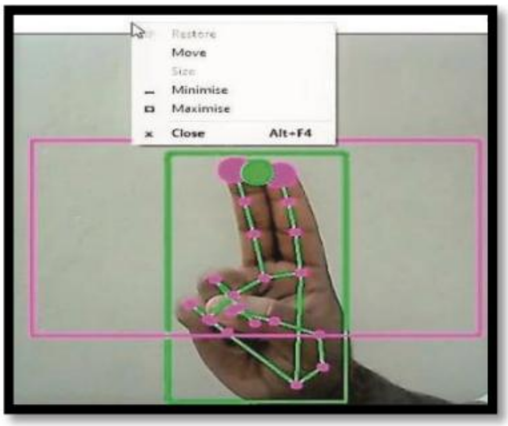


Figure 5: Gesture for the computer to move the cursor

Figure 6: Gesture for right click

V. ALGORITHMS USED FOR TRACKING HAND

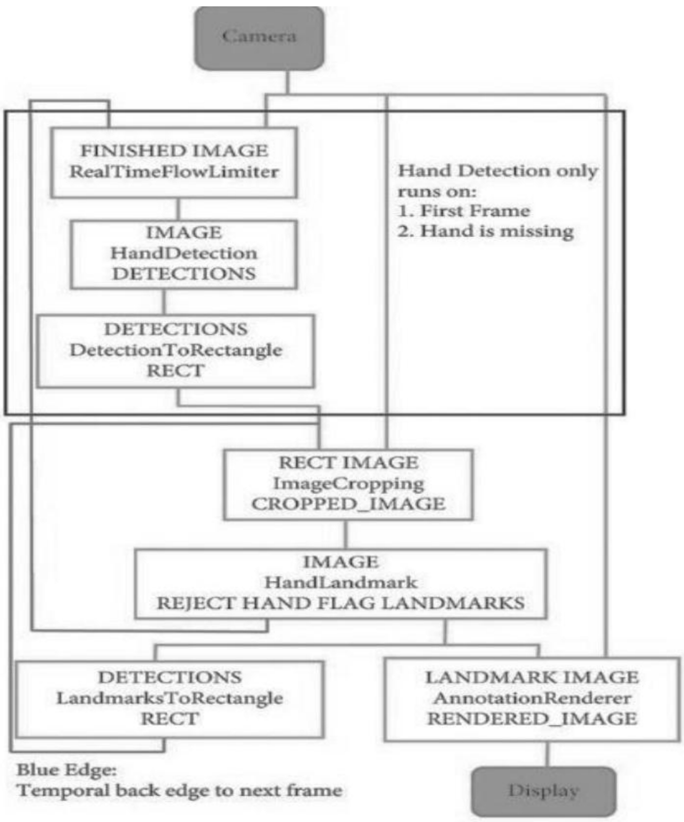


Figure 7: Hand recognition graph.

MediaPipe(MP):

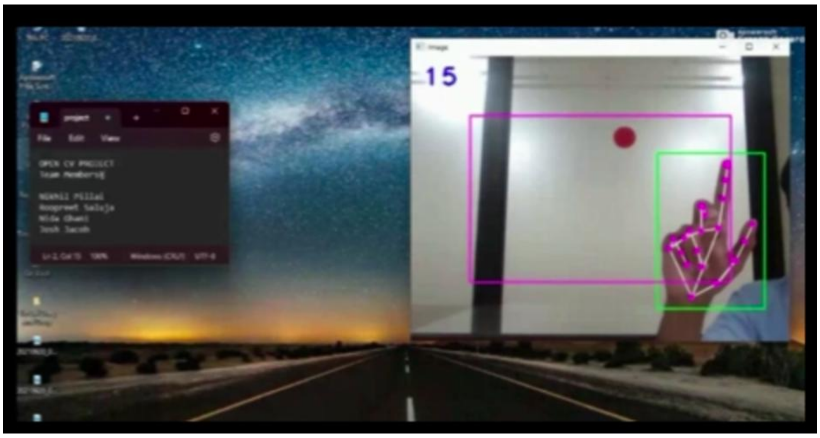
When using the OpenCV library for computer vision, the MP framework is used for hand tracking and gesture detection.Machine learning algorithms are used by the program to identify, track, and record hand motions.MediaPipe is a Google architecture that is made accessible for free and used in a machine learning pipeline. The MediaPipe framework can be utilized for cross-platform development because it was developed using time series data. The MediaPipe architecture is multimodal and supports a variety of audio and video formats. The MediaPipe framework is used by the developer to design systems for application-related goals as well as to build and analyze graph-based systems. The pipeline configuration is where the actions in the MediaPipe-using system are carried out.With the aid of a single-shot detector device, a palm or hand can be instantly detected and identified. The concept of a single-shot detector is utilized by the MediaPipe. The hand detection module first trains a model for palm detection since learning the palms is simpler than learning the hands.

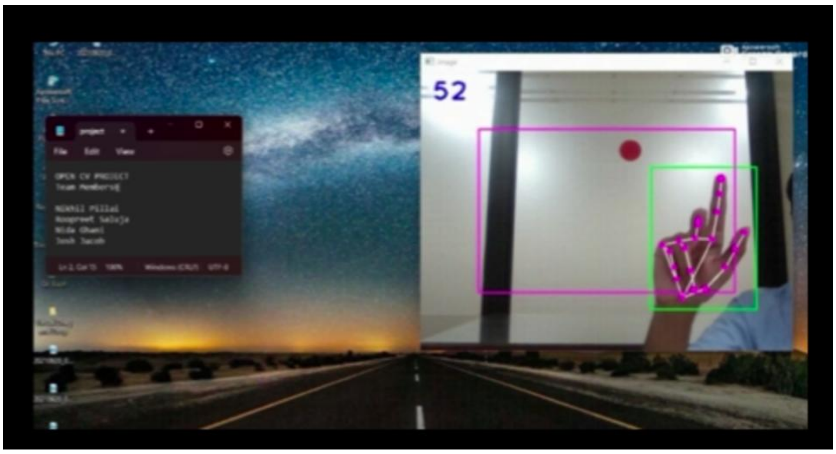
OpenCV:

Image-processing tools for object detection are part of the OpenCV computer vision toolbox. The Python's OpenCV package may be deployed to develop real-time computer vision software. The OpenCV library is used for image and video processing and also for analytical methods like face recognition.

VI. RESULTS AND DISCUSSION :

Output ScreenShots :





The idea of enhancing human-computer interaction using cv is shown through the suggested mouse system.Testing of the AI mouse is difficult to compare since there are so few datasets available. The webcamera has been positioned at different distances from the users to monitor hand motions and identify finger tips in an effort to assess their abilities under varied illumination conditions.

Artificial intelligence (AI) virtual mouse system has a success rate of nearly 99%.Based on its 99% accuracy, we can say that the suggested AI mouse has worked as intended. The "Right Click" gesture is the most challenging for the computer to comprehend, hence accuracy is low. Because the gesture required to carry out the specific mouse action is more difficult, the right click's precision is poor.

All of the other gestures have high and excellent accuracy as well. In comparison to other methods for virtual mice, our model performed 99% accurately and very effectively.

VII. ADVANTAGES AND DISADVANTAGES :

Advantages:

1. Enhanced Precision: Real-time AI mouse can utilize machine learning algorithms to improve cursor precision and accuracy, resulting in smoother and more responsive movements. This can be particularly beneficial for tasks that require fine-grained control, such as graphic design or gaming.
2. Adaptive Sensitivity: AI-powered mice can dynamically adjust sensitivity based on contextual factors. For instance, the mouse can automatically increase sensitivity during fast movements or decrease it for precise actions, allowing for greater flexibility and control.
3. Gesture Recognition: With AI capabilities, the mouse can recognize complex gestures made by the user. This enables the execution of specific actions or commands by simply moving the mouse in a particular pattern, making navigation and interaction more intuitive and efficient.
4. Personalized Experience: Real-time AI mice can learn and adapt to individual user preferences over time. They can analyze usage patterns, understand user behavior, and customize settings accordingly, resulting in a more personalized and comfortable user experience.

Disadvantages:

1. Cost: The integration of AI capabilities into computer mice can increase their cost compared to traditional mice. Advanced sensors, algorithms, and additional hardware components contribute to the higher price point, which may limit accessibility for some users.
2. Power Consumption: AI algorithms require computational resources, which can lead to increased power consumption. Real-time AI mice might require additional power or frequent battery replacements, impacting their overall battery life and potentially reducing the convenience of wireless usage.
3. Complexity: AI mice may come with more complex software interfaces and settings compared to regular mice. Users might need to spend more time learning and adjusting settings to fully utilize the AI features. This added complexity could be overwhelming for some users, especially those who prefer simpler and more straightforward devices.
4. Reliability and Compatibility: As with any technology, there can be occasional issues with reliability or compatibility. AI algorithms may not always perform optimally in certain situations, leading to unexpected behavior or inaccuracies. Additionally, compatibility with different operating systems or software applications may vary, potentially limiting the full utilization of AI capabilities.

   It's worth noting that the advantages and disadvantages mentioned above are general considerations, and the specific implementation of an AI mouse can vary across different products and manufacturers.

VIII. APPLICATIONS :

1. Accessibility: Virtual AI mouse technology can significantly improve accessibility for individuals with physical disabilities or limited mobility. It allows users to control the computer cursor using alternative input methods, such as head tracking, eye tracking, or gestures. This can empower people with disabilities to interact with computers and perform tasks more independently.
2. Virtual Reality (VR) and Augmented Reality (AR): In VR and AR environments, virtual AI mice can serve as an intuitive and natural way to interact with virtual objects. By capturing hand or finger movements, the virtual AI mouse can provide a seamless and immersive experience, enabling users to navigate menus, manipulate objects, or perform actions within the virtual world.
3. Gesture Recognition: Real-time virtual AI mice can interpret hand or finger gestures and translate them into computer commands. This technology can find applications in gaming, presentations, or any scenario where intuitive gesture-based interaction is desired. Users can perform actions like swiping, pinching, or rotating their hands to control the cursor or trigger specific commands.
4. Remote Control: Virtual AI mice can be used to control computers or devices remotely. With the help of AI algorithms, users can manipulate the virtual cursor on a remote computer or mobile device, enabling them to access and operate applications and files from a distance. This can be particularly useful for remote collaboration, presentations, or accessing computers in inaccessible locations.

IX. CONCLUSION :

The final AI mouse has a few issues, including a modest loss in right click precision and the model's limited ability to drag on clicking to pick text. To solve these and other issues, we propose an AI virtual mouse technology in our later study. The virtual control of keyboard and mouse actions may also be used in future applications of human-computer interaction.
Instead of using a hardware mouse to control mouse cursor capabilities, an AI system will take over the primary function of hand gestures. It is feasible to employ a live webcam or an embedded camera that detects hand and fingertip movements and analyzes these frames to execute prepared mouse instructions to increase the precision of the suggested system. The model's outcomes show that the suggested virtual cursor system has performed remarkably well, has greater precision than the earlier models, and successfully addresses the majority of their flaws. The suggested model can be used in real-world scenarios because it is more realistic. Furthermore, as the suggested mouse technology can be used to operate electronically with hand gestures instead of a traditional physical mouse.

X. REFERENCES :

[1] J. Katona, “A review of human–computer interaction and virtual reality research fields in cognitive Info Communications,” Applied Sciences, vol. 11, no. 6, p. 2646, 2021.

[2] D. L. Quam, “Gesture recognition with a DataGlove,” IEEE Conference on Aerospace and Electronics, vol. 2, pp. 755–760, 1990.

[3] D.-H. Liou, D. Lee, and C.-C. Hsieh, “A real time hand gesture recognition system using motion history image,” in Proceedings of the 2010 2nd International Conference on Signal Processing Systems, July 2010.

[4] S. U. Dudhane, “Cursor control system using hand gesture recognition,” IJARCCE, vol. 2, no. 5, 2013.

[5] K. P. Vinay, “Cursor control using hand gestures,” International Journal of Critical Accounting, vol. 0975–8887, 2016.

[6] L. Thomas, “Virtual mouse using hand gesture,” International Research Journal of Engineering and Technology (IRJET, vol. 5, no. 4, 2018.

[7] P. Nandhini, J. Jaya, and J. George, “Computer vision system for food quality evaluation—a review,” in Proceedings of the 2013 International Conference on Current Trends in Engineering and Technology (ICCTET), pp. 85–87, Coimbatore, India, July 2013.

[8] J. Jaya and K. Thanushkodi, “Implementation of certain system for medical image diagnosis,” European Journal of Scientific Research, vol. 53, no. 4, pp. 561–567, 2011.

[9] P. Nandhini and J. Jaya, “Image segmentation for food quality evaluation using computer vision system,” International Journal of Engineering Research and Applications, vol. 4, no. 2, pp. 1–3, 2014.

[10] J. Jaya and K. Thanushkodi, “Implementation of classification system for medical images,” European Journal of Scientific Research, vol. 53, no. 4, pp. 561–569, 2011.

[11] J. T. Camillo Lugaresi, “MediaPipe: A Framework for Building Perception Pipelines,” 2019, https://arxiv.org/abs/1906.08172.

[12] Google, MP, https://ai.googleblog.com/2019/08/on-devicereal-time-hand-tracking-with.html.

[13] V. Bazarevsky and G. R. Fan Zhang. On-Device, MediaPipe for Real-Time Hand Tracking