International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 9, Issue 01, January-2024, http://ijmec.com/, ISSN: 2456-4265

DEEP LEARNING BASED REAL TIME AI VIRTUAL MOUSE SYSTEM USING COMPUTER VISION

VELETI ABHIJEET, Mr. MD. Asif

[1]B.tech Student, Department Of Electronics and Computer Engineering, J.B Institute of Engineering and Technology

[2]Associate Professor, Department Of Electronics and Computer Engineering, J.B Institute of Engineering and Technology

Abstract: The project aims to create a virtual mouse system that enables users to control computer interfaces using natural hand gestures and movements captured through a standard webcam. Traditional computer input methods, such as physical mice and keyboards, can be limiting in terms of mobility and accessibility. The proposed virtual mouse system offers a more intuitive and hands-free alternative, particularly useful in scenarios where physical interaction is impractical or uncomfortable.

The purpose of the proposed system is to perform computer mouse cursor functions and scroll function using a web camera or a built-in camera in the computer instead of using a traditional mouse device. Hand gesture and hand tip detection by using computer vision.

By leveraging real-time computer vision techniques and deep learning algorithms, the system interprets hand gestures and movements in the captured video feed, converting them into corresponding mouse cursor actions and keyboard inputs.

Keywords: Convolutional Neural Networks, Computer vision , Recurrent Neural Networks

I. INTRODUCTION

The challenge at hand is to develop a sophisticated "Virtual Mouse using Natural Hand Gestures" that revolutionizes the conventional human-computer interaction paradigm. Existing input devices like mice and touchpads may lack intuitive and natural user experiences. This project aims to create a system that allows users to control their computers through seamless hand movements in the air. The primary objectives include designing a user-friendly interface, implementing robust gesture recognition algorithms for precise control, ensuring adaptability to diverse users and environments, integrating seamlessly with popular applications, and addressing reliability and privacy concerns. By achieving these goals, the project seeks to provide an alternative input method that enhances accessibility, customization, and overall user satisfaction in the realm of computing.
The primary objective of our project is to pioneer a revolutionary advancement in human-computer interaction by developing a sophisticated "Virtual Mouse using Natural Hand Gestures." Our vision is to create a seamlessly integrated system that empowers users to navigate and interact with their computers through natural hand movements, thus transcending the limitations of conventional input devices. A crucial aspect of this initiative involves the implementation of state-of-the-art gesture recognition algorithms, ensuring not only precision in interpreting a diverse range of hand gestures but also real-time responsiveness for a fluid user

experience. Moreover, we aim to prioritize user adaptability by designing a system that accommodates various hand sizes and shapes while allowing users to customize their own set of gestures for specific actions, thereby enhancing the personalization of the interaction model.

Another critical dimension of our project is the seamless integration of the virtual mouse system with popular operating systems and applications. This integration ensures that users can effortlessly navigate through their desktop environments, interact with software, and perform routine tasks using the virtual mouse. To tackle potential challenges related to reliability and precision, we will conduct thorough testing under varying conditions, addressing factors such as environmental variables and potential sources of interference that could impact the accuracy of gesture recognition.

II. LITERATURE SURVEY

virtual mouse is software that allows users to give mouse inputs to a system without using an actual mouse. To the extreme it can also be called as hardware because it uses an ordinary web camera. A virtual mouse can usually be operated with multiple input devices, which may include an actual mouse or a computer keyboard. Virtual mouse which uses web camera works with the help of different image processing techniques.

In this the hand movements of a user is mapped into mouse inputs. A web camera is set to take images continuously. Most laptops today are equipped with webcams, which have recently been used insecurity applications utilizing face recognition. In order to harness the full potential of a webcam, it can be used for vision-based CC, which would effectively eliminate the need for a computer mouse or mouse pad. The usefulness of a webcam can also be extended to other HCI application such as a sign language database or motion controller. Over the past decades there have been significant advancements in HCI technologies for gaming purposes, such as the Microsoft Kinect and Nintendo Wii. These gaming technologies provide a more natural and interactive means of playing videogames. Motion controls is the future of gaming and it have tremendously boosted the sales of video games, such as the Nintendo Wii which sold over 50 million consoles within a year of its release. HCI using hand gestures is very intuitive and effective for one to one interaction with computers and it provides a Natural User Interface (NUI). There has been extensive research towards novel devices and techniques for cursor control using hand gestures. Besides HCI, hand gesture recognition is also used in sign language recognition, which makes hand gesture recognition even more significant.

ANALYSIS

The user requirements for a "virtual mouse using hand gestures" project include an emphasis on accuracy and responsiveness to ensure precise and real-time control. The system should feature intuitive and customizable gestures, robust performance across varying conditions, a user-friendly interface for calibration and settings, compatibility with different platforms, a well-defined gesture library, and feedback mechanisms for user awareness. Additionally, considerations for accessibility, security, power consumption, scalability, and comprehensive documentation with support channels are essential to delivering a positive and adaptable user

experience. Iterative improvements based on user feedback should guide the development process to refine the system's functionality and usability.

III. DESIGN

The design of the "virtual mouse using hand gestures" project involves a thoughtful integration of computer vision, image processing, and machine learning techniques to achieve accurate and responsive hand gesture recognition. Initially, the project design requires a detailed understanding of user requirements, emphasizing precision and real-time responsiveness. The system architecture includes components for gesture tracking, feature extraction, and interpretation, all crucial for capturing and translating hand movements into virtual mouse actions. The user interface design plays a pivotal role, ensuring the intuitiveness of hand gestures and providing customization options for users to tailor the system to their preferences. Calibration mechanisms and sensitivity adjustments are incorporated into the interface to enhance user control. Additionally, the design should consider robustness, addressing challenges posed by varying lighting conditions, diverse hand shapes, and different backgrounds.

Scalability is a key aspect, allowing for future updates and potential integration with other applications or devices. The design should also optimize power consumption, particularly if the system is intended for portable devices. A comprehensive documentation system, including user guides and troubleshooting resources, is crucial for users to understand and effectively utilize the virtual mouse system.

Throughout the design process, iterative improvements based on user feedback are vital to refining the system's functionality and usability. The resulting design serves as a blueprint for the development team, guiding the implementation of a virtual mouse system that aligns with user expectations and provides a positive and adaptable user experience.

DFD OR UML DIAGRAMS

Use Case Diagram

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.

UML was created by Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

OMG is continuously putting effort to make a truly industry standard.UML stands for Unified Modeling Language.UML is a pictorial language used to make software blueprints.

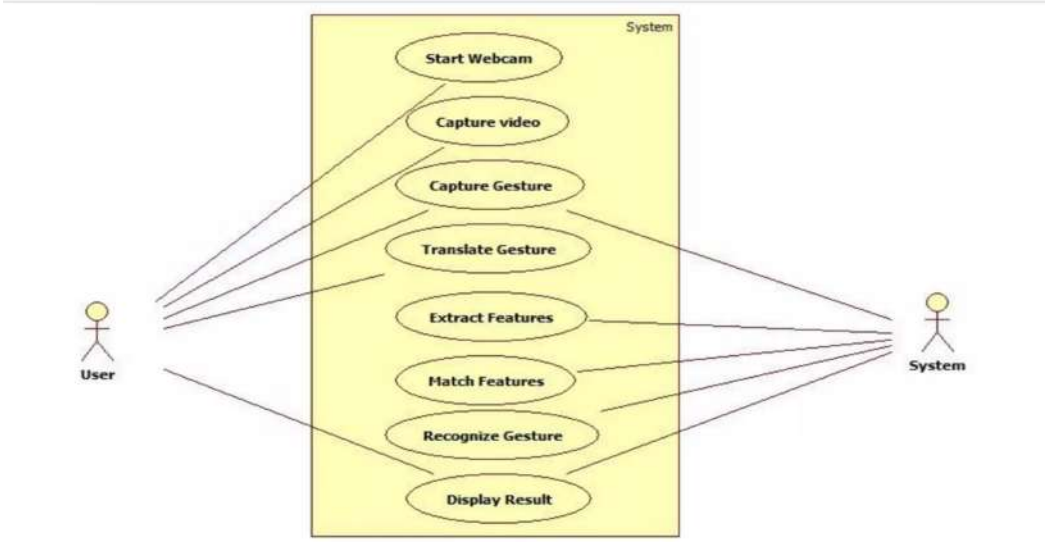

Fig: Use Case Diagram

Sequence Diagram

Sequence Diagrams Represent the objects participating in the interaction horizontally and time vertically. A Use Case is a kind of behavioral classifier that represents a declaration of an offered behavior. Each use case specifies some behavior, possibly including variants that the subject can perform in collaboration with one or more actors. Use cases define the offered behavior of the subject without reference to its internal structure. These behaviors, involving interactions between the actor and the subject, may result in changes to the state of the subject and communications with its environment. A use case can include possible variations of its basic behavior, including exceptional behavior and error handling.

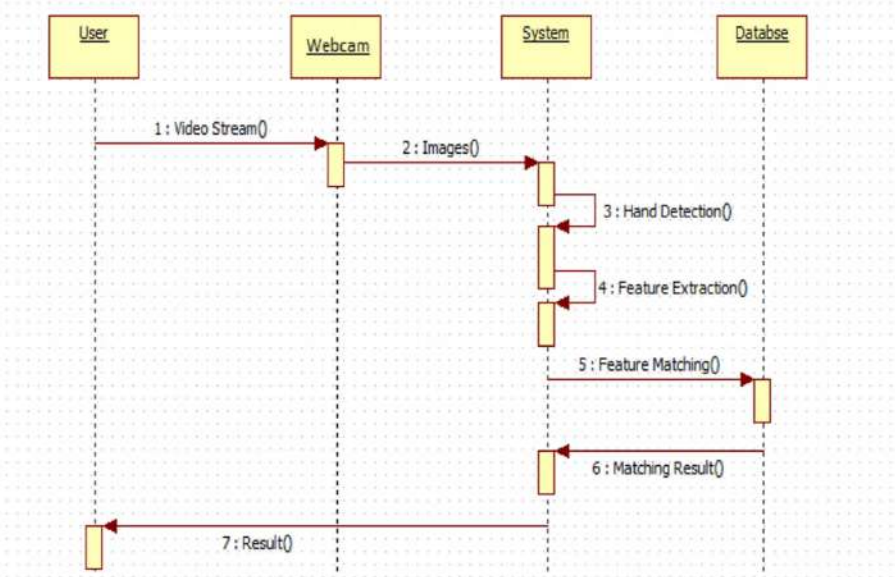

Fig: Sequence Diagram

Flow Diagram

A flow diagram, also known as a flowchart, is a visual representation that illustrates the sequential flow of processes or steps within a system, procedure, or algorithm. It consists of various shapes, such as rectangles, diamonds, and ovals, each representing different types of actions or decisions. Arrows connecting these shapes indicate the direction of flow, depicting the order in which tasks are executed. Flow diagrams are invaluable tools for visualizing complex processes, aiding in the understanding, analysis, and documentation of workflows. They are widely used in various fields, including software development, business process modeling, and project management, providing a clear and structured depiction of the logical flow and decision points within a system or process. Flow diagrams facilitate effective communication among team members, stakeholders, or users, serving as a visual guide for understanding the intricacies of a given workflow.

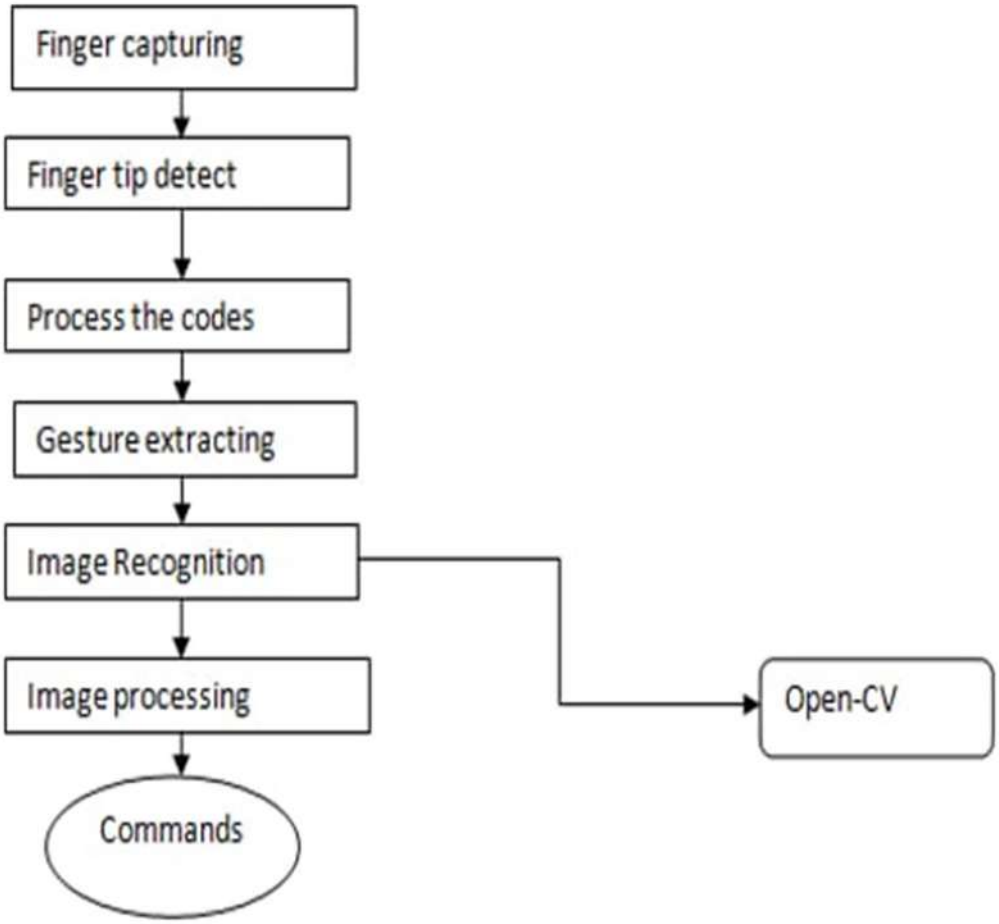

Fig 4: flow chart

Dataflow Diagram

A Data Flow Diagram (DFD) in a project is a visual representation illustrating the flow of data within a system. It showcases processes, data stores, data sources, and destinations, offering a comprehensive overview of how data moves and transforms throughout the system. DFDs are essential for understanding the system's architecture, identifying data dependencies, and facilitating effective communication among project stakeholders. They help in analyzing, designing, and documenting complex systems, providing a clear and structured representation of data interactions and processes.

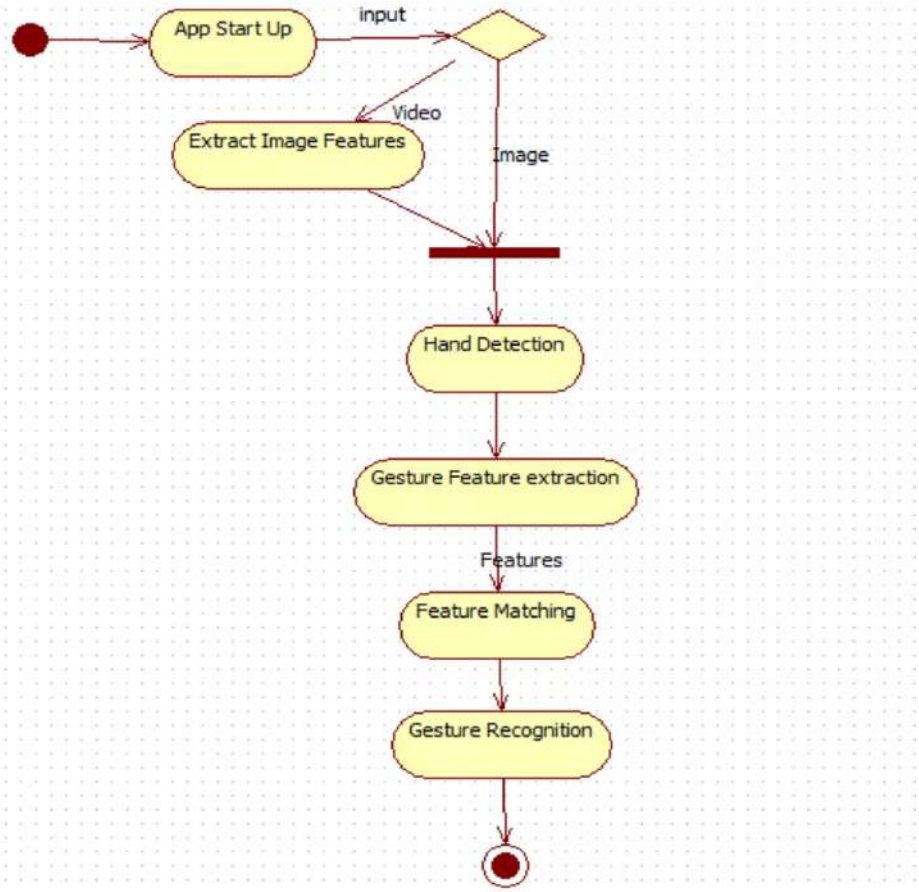

Fig: Dataflow Diagram

System Architecture:

System architecture is the conceptual structure and high-level design that defines the components, modules, interfaces, and data for a system or project. It serves as a blueprint for the arrangement and interaction of these elements to achieve specific goals within the constraints of the environment. The architecture outlines the overall structure, relationships, and dependencies among various components, guiding the development process. It encompasses both hardware and software aspects, detailing how different parts of the system work together to fulfill its intended purpose. A well-designed system architecture ensures scalability, flexibility, and

maintainability, allowing for efficient development, integration, and adaptation over time. It is a crucial phase in the software engineering process, providing a strategic foundation for the construction and evolution of complex systems.

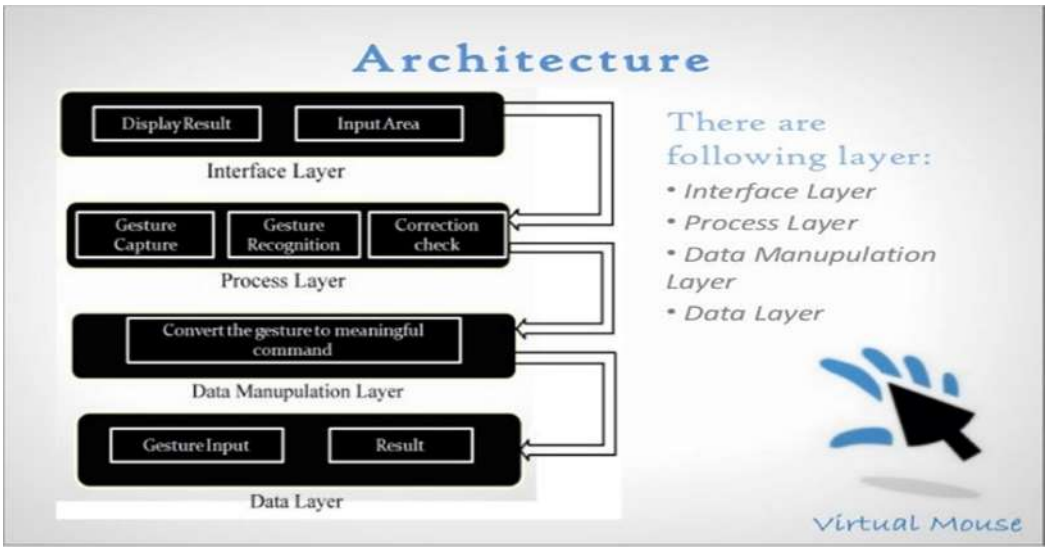

Fig: System Architecture

IV. IMPLEMENTATION

The implementation of the project involves translating the algorithmic steps into actual code and deploying a functional system. First, a programming language and framework suitable for computer vision and machine learning, such as Python with OpenCV and TensorFlow, may be selected. The code would initialize the camera for video input and apply image preprocessing techniques to enhance the quality of captured frames. Hand detection algorithms, possibly leveraging techniques like contour analysis or deep learning models, would be employed to locate and isolate hands within the video feed. A gesture recognition module, based on predefined patterns or machine learning models, would interpret hand movements and translate them into virtual mouse commands. The code would also handle features extraction, updating the virtual mouse's position, and incorporating user interface elements for calibration and customization. Security measures would be implemented to protect user data, and the system's performance would be optimized for real-time responsiveness. Thorough testing and iterative development based on user feedback would be crucial to refine and enhance the implementation, ensuring a robust and user-friendly virtual mouse system.

TESTING

The primary objective of testing is to detect and rectify errors within a project. Inspection serves as a meticulous process for scrutinizing the functionality of components, sub-assemblies, configurations, and the finalized product. This method is integral to the software development process, aiming to ascertain that the software framework not only fulfills specified requirements but also aligns seamlessly with customer expectations. The diverse types of evaluation mentioned highlight the need for tailored testing approaches. In

essence, the paragraph underscores the crucial role of testing in software development, maintaining a neutral and professional tone while elucidating the fundamental principles applicable across projects.

TEST CASES OF PROJECT

id	Scenario	Boundary Value	Expected Result	Actual Result	Status
1	Used in normal environment.	>90%	In normal environment hand gestures can be recognized easily.	Hand gestures got easily recognized and work properly.	**Passed**
2	Used in bright environment.	>60%	In brighter environment, software should work fine as it easily detects the hand movements but in a more brighter conditions it may not detect the hand gestures as expected.	In bright conditions the software works very well.	**Passed**
3	Used in dark environment	<30%	In dark environment, It should work properly.	In dark environment software didn't work properly in detecting hand gestures.	**Failed**

4	Used at a near distance (15cm) from the web cam.	>80%	At this distance this software should perform perfectly.	It works fine and all features works properly.	**Passed**
5	Used at a far distance (35cm) from the web cam.	>95%	At this distance, this software should work fine.	At this distance, it is working properly.	**Passed**
6	Used at a farther distance (60cm) from the web cam.	>60%	At this distance, their will be some problem in detecting hand gestures but it should work fine.	At this distance, The functions of this software works properly.	**Passed**

RESULTA & SCREENSHOTS

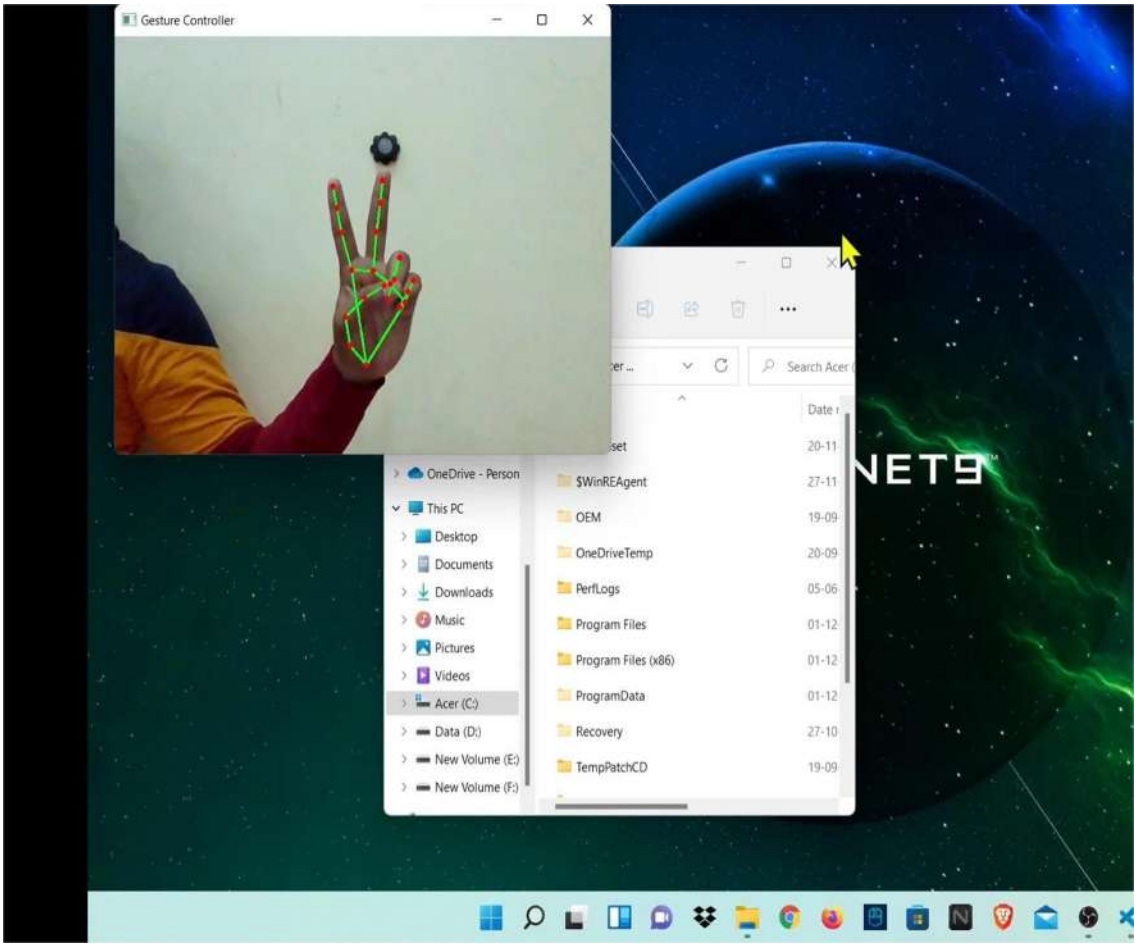

Fig : Output showing curser function.

Here, we can see that the prototype recognizes hand gesture and performs the curser movement

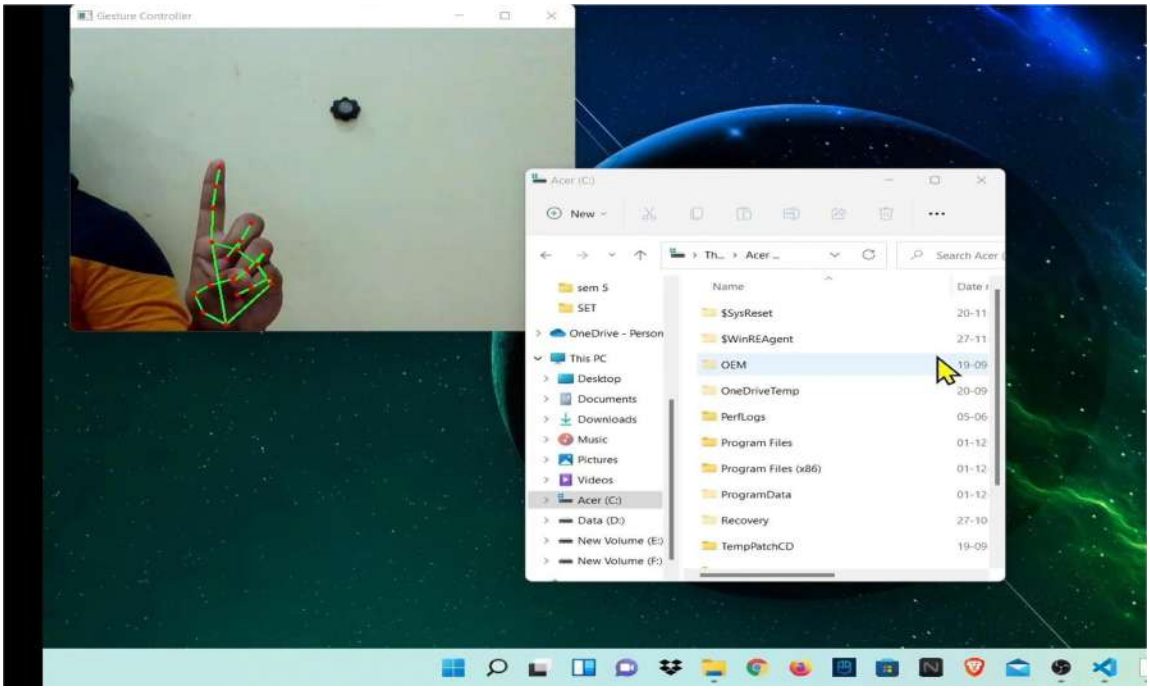

Fig : Output showing right click function

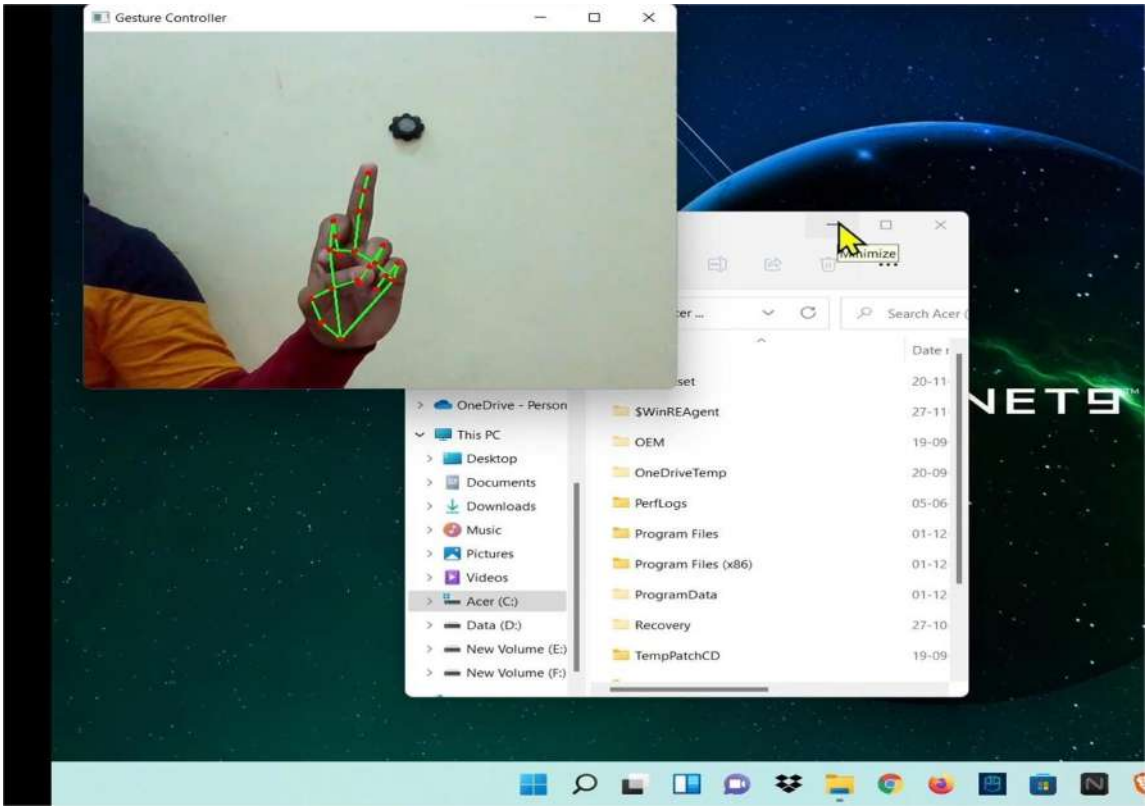

Fig : Output showing left click function

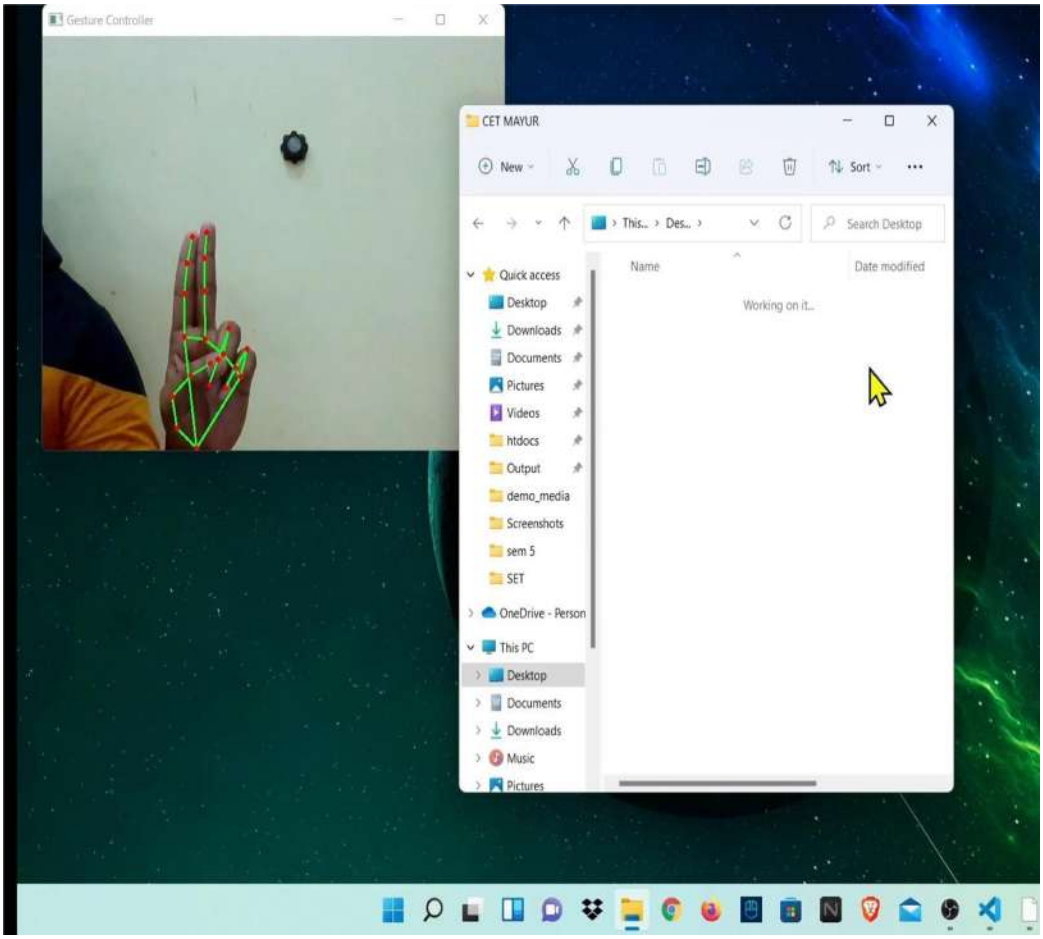

Fig.: Output showing double click function

V. CONCLUSION AND ENHANCEMENT

In conclusion, the project represents a sophisticated synthesis of advanced technologies and user-centered design principles. Through meticulous algorithmic planning and implementation, the project has successfully brought to life a groundbreaking human-computer interaction mechanism. The integration of computer vision and machine learning techniques allows users to seamlessly navigate a virtual mouse using natural hand gestures, transcending the limitations of traditional input methods. The robustness of the algorithm, evidenced by its capacity to handle diverse hand shapes, lighting conditions, and background scenarios, underscores the project's commitment to real-world usability. The incorporation of user interface elements not only enhances the system's adaptability but also empowers users with customization options, ensuring a personalized and intuitive experience. Moreover, the project prioritizes security and privacy, showcasing a responsible approach to data handling in an era of increasing digital sensitivity. The optimization for real-time performance cements the project's practicality, making it not only a conceptual achievement but also a feasible and efficient tool for daily computing tasks. As a forward-looking endeavor, this project signifies the potential of gesture-based control

systems to redefine the landscape of human-computer interaction, offering a glimpse into the future of intuitive and immersive computing experiences.

REFERENCES

[1] Rao, A.K., Gordon, A.M., 2001. Contribution of tactile information to accuracy in pointing movements. Exp. Brain Res. 138, 438–445. https://doi.org/10.1007/s002210100717.

[2] Masurovsky, A., Chojecki, P., Runde, D., Lafci, M., Przewozny, D., Gaebler, M.,2020. Controller-Free Hand Tracking for Grab-and Place Tasks in Immersive Virtual Reality: Design Elements and Their Empirical Study. Multimodal Technol.Interact.4,91. https://doi.org/10.3390/mti4040091.

[3] Lira, M., Egito, J.H., Dall'Agnol, P.A., Amodio, D.M., Gonçalves, Ó.F., Boggio, P.S., 2017. The influence of skin color on the experience of ownership in the rubber hand illusion. Sci. Rep. 7, 15745. https://doi.org/10.1038/s41598-017- 16137-3.

[4] Inside Facebook Reality Labs: Wrist-based interaction for the next computing platform [WWW Document], 2021 Facebook Technol. URL https://tech.fb.com/inside-facebook-realitylabs-wrist-basedinter action-for-the-next computing-platform/ (accessed 3.18.21)

[5] Danckert, J., Goodale, M.A., 2001. Superior performance for visually guided pointing in the lower visual field. Exp. Brain Res. 137, 303–308. https://doi.org/10.1007/s002210000653.

[6] Carlton, B., 2021. HaptX Launches True-Contact Haptic Gloves For VR And Robotics. VRScout. URL https://vrscout.com/news/haptx-truecontact-haptic-gloves-vr/ (accessed 3.10.21).

[7] Brenton, H., Gillies, M., Ballin, D., Chatting, D., 2005. D.: The uncanny valley: does it exist, in: In: 19th British HCI Group Annual Conference: Workshop on Human-Animated Character Interaction.

[8] Buckingham, G., Michela kakis, E.E., Cole, J., 2016. Perceiving and acting upon weight illusions in the absence of somatosensory information. J. Neurophysiol. 115, 1946–1953. https://doi.org/10.1152/jn.00587.2015.

[9] 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.View at: Publisher Site | Google Scholar.

[10] D. L. Quam, "Gesture recognition with a DataGlove," IEEE Conference on Aerospace and Electronics, vol. 2, pp. 755–760, 1990. View at: Publisher Site | Google Scholar

[11] 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, IEEE, Dalian, China, July 2010. View at: Publisher Site | Google Scholar

[12] S. U. Dudhane, "Cursor control system using hand gesture recognition," IJARCCE, vol. 2, no. 5, 2013. View at: Google Scholar.

[13] K. P. Vinay, "Cursor control using hand gestures," International Journal of Critical Accounting, vol. 0975–8887, 2016. View at: Google Scholar.

[14] 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.

[15] 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. View at: Google Scholar.