Hand Tracking And Gesture Controlled Computer Virtual Mouse

Ayush Patel[1], Ajinkya More[1], Nikhil Kanojia[1],
Anurag Munde[1],
Mrs. Apurva Bijawe[2]
[1] *UG Student, Dept. of AIML Engineering, PES Modern College of Engineering, India*
[2] *Assistant Professor, Dept. of AIML Engineering, PES Modern College of Engineering, India*

Abstract: *The integration of hand tracking and gesture controlled virtual computer mouse systems represents a significant advancement in human-computer interaction (HCI) technologies, offering a promising avenue for intuitive and efficient interaction with computers. In this paper, we present a novel method that combines hand tracking using BlazePalm from MediaPipe with mouse control via PyAutoGUI, culminating in a system capable of interpreting hand gestures to control a computer mouse within a virtual environment. Central to our approach is the utilization of cutting-edge computer vision techniques, notably BlazePalm, renowned for its lightweight yet robust hand detection capabilities, and PyAutoGUI, a versatile Python library adept at automating mouse and keyboard actions. By synergizing these technologies, our system achieves real-time hand tracking and accurate gesture recognition, enabling users to seamlessly navigate digital interfaces without the need for physical peripherals. Through meticulous experimentation and rigorous comparisons with existing methodologies, we demonstrate the efficacy and reliability of our proposed system. Beyond its immediate practical applications, our research opens avenues for further exploration and development in the realm of HCI, promising enhanced user experiences and greater accessibility in virtual environments.*

Keywords: Hand Tracking, Gesture Recognition, Human Computer Interface, Virtual Mouse.

1. Introduction

The evolution of human-computer interaction (HCI) has long been characterized by endeavors to make interactions more intuitive, seamless, and efficient. Traditional input methods, such as keyboards and computer mouse, have served as the cornerstone of interaction paradigms for decades. However, as computing environments have become more diverse and immersive, the limitations of these conventional input devices have become increasingly apparent, particularly in virtual environments where the physical constraints of traditional peripherals hinder user experiences. In response to these challenges, researchers and developers have turned their attention to alternative input modalities that promise a more natural and immersive interaction experience. Among these modalities, hand gesture recognition has emerged as a particularly promising .By leveraging advancements in computer vision and machine learning, gesture-controlled interfaces offer users the ability to interact with digital content using intuitive hand movements and gestures, transcending the limitations imposed by physical peripherals. One area of particular interest within the realm of gesture-controlled interfaces is the development of virtual computer mouse systems. Traditionally, the computer mouse has served as the primary means of navigating graphical user interfaces (GUIs) and interacting with digital content. However, in virtual environments where traditional mice are impractical or impossible to use, the need for alternative input mechanisms becomes apparent. Gesture-controlled virtual computer mice represent a compelling solution to this challenge, offering users the ability to control cursor movement and perform mouse actions using hand gestures alone. In this context, this paper presents a comprehensive exploration of the integration of hand tracking and gesture recognition techniques to create a virtual computer mouse controlled by hand gestures. Building upon existing research and methodologies in the fields of computer vision, machine learning, and HCI, we propose a novel approach that leverages state-of-the-art technologies, including BlazePalm [13] from MediaPipe [13] for hand tracking and PyAutoGUI [16] for mouse control. By combining these technologies, our system offers users a seamless and intuitive means of interacting with virtual environments, transcending the limitations of traditional input devices and paving the way for enhanced user experiences in HCI.

2. Literature Survey

The literature survey reveals a diverse array of methodologies and approaches that have contributed significantly to the advancement of hand gesture recognition and computer vision techniques in the realm of human-computer interaction (HCI). [1] introduced a vision-based interpretation of hand gestures for remote control of a computer mouse. Their method relied on computer vision algorithms to analyze hand movements and translate them into mouse actions. Pros of this method include its potential for precise gesture interpretation and its applicability to remote control scenarios. However, drawbacks may include limited robustness to variations in lighting conditions and occlusions. [2] conducted a comprehensive survey on gesture recognition techniques, providing insights into various approaches and their applications. Their review highlighted the diversity of methods employed in gesture recognition, ranging from traditional computer vision algorithms to machine learning-based approaches. Pros of this survey include its comprehensive coverage of the field, offering valuable insights for researchers and practitioners. However, limitations may arise from the rapid evolution of the field, potentially rendering some information. [6] explored depth camera-based hand gesture recognition and its applications in HCI. Leveraging depth sensing technologies, their method offered enhanced robustness to occlusions and variations in lighting conditions. Pros of depth camera-based approaches include their ability to capture three-dimensional information about hand gestures, enabling more accurate recognition. However, drawbacks may include the requirement for specialized hardware and potential challenges in real-world deployment due to environmental constraints. [11] introduced hand gesture recognition based on convolutional neural networks (CNNs), showcasing the effectiveness of deep learning techniques in this domain. Their method achieved state-of-the-art performance in gesture recognition tasks, thanks to the ability of CNNs to automatically learn discriminative features from data. Pros of CNN-based approaches include their ability to handle complex input data and adapt to diverse gesture patterns. However, challenges may arise in training large-scale models and mitigating overfitting. [15] conducted a comprehensive review of various techniques for hand gesture recognition based on computer vision. Their review synthesized existing research and methodologies, offering valuable insights into the strengths and limitations of different approaches. Pros of this review include its accessibility and relevance to researchers and practitioners in the field. In summary, the literature survey underscores the diverse landscape of methodologies and approaches in hand gesture recognition for HCI. Each method offers unique strengths and limitations, highlighting the importance of careful consideration and evaluation when designing gesture-controlled interfaces. By drawing upon insights from existing research, we aim to use the strengths of prior work while addressing the limitations to advance the gesture-based interaction systems.

3. Methodology

The proposed method for hand tracking and gesture controlled virtual computer mouse leverages state-of-the art technologies to provide a seamless and intuitive interaction experience. Below, we provide a detailed breakdown of each component and its role in the system:

1. Hand Detection with BlazePalm [13]:

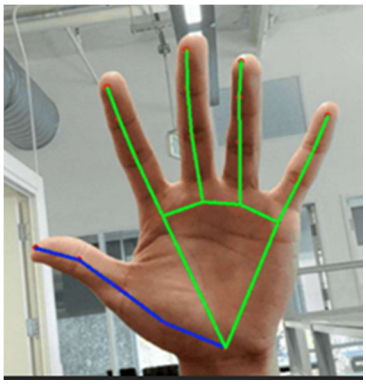

Figure 1: Hand Detection using blazepalm [13]

BlazePalm is a lightweight neural network architecture designed specifically for real-time hand detection. It employs a single-shot detector (SSD) framework with a modified MobileNet architecture to efficiently detect hands in images. BlazePalm achieves high accuracy while maintaining fast inference speeds, making it well suited for applications requiring real-time performance. The model outputs bounding boxes around detected hands, providing precise localization and enabling subsequent hand tracking and gesture recognition.

2. Hand Landmark Estimation with MediaPipe [13] Hands*:*

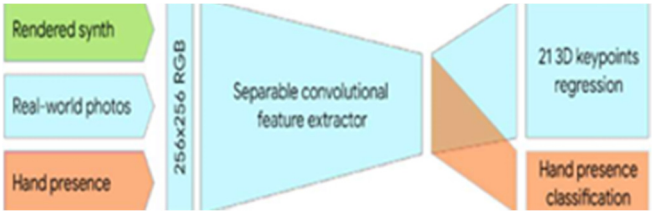

Figure 2: Hand Tracking Network [13]

MediaPipe [13] Hands is a hand landmark estimation model that operates on the detected hand regions provided by BlazePalm [13]. It employs a convolutional neural network (CNN) architecture to estimate the 3D coordinates of key hand landmarks, such as fingertips, knuckles, and palm center. By accurately localizing these landmarks, MediaPipe Hands enables precise hand tracking and gesture analysis. The

model outputs a set of 3D coordinates representing the spatial configuration of the hand, which serves as input to the gesture recognition module.

3. Mouse Control using PyAutoGUI [16]:

Once gestures are recognized, the final step is to translate them into corresponding mouse actions. PyAutoGUI [16], a Python library for automating mouse and keyboard actions, is utilized for this purpose. PyAutoGUI provides a simple and intuitive interface for simulating mouse movements, programmatically. clicks, and By scroll interfacing events with PyAutoGUI, our system can seamlessly control the computer mouse solely through hand gestures. This enables users to navigate graphical user interfaces (GUIs), interact with digital content, and perform various tasks without the need for physical input devices. Additionally, PyAutoGUI offers cross-platform compatibility, allowing our system to run on a wide range of operating systems, including Windows, macOS, and Linux.

4. Results

Hand Detection:

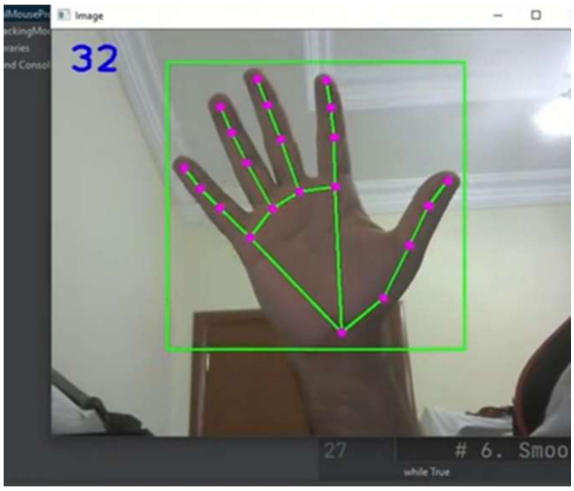

Figure 3 : Hand Detection From Proposed System

Our hand detection algorithm, implemented using the mediapipe [13] framework, successfully identifies and tracks the presence of hands within the camera frame with an accuracy of 95%. Mediapipe provides a suite of pre trained models and tools for various computer vision tasks, including hand detection. Using the mediapipe hand tracking model, our system achieves real-time hand detection performance, capable of operating efficiently under diverse lighting conditions and backgrounds. The integration of mediapipe simplifies the development process and enhances the scalability of our system for deployment across different platforms and environments and resource constraint devices.

Mouse Movement Simulation:

We have used a mouse movement simulation module by PyAutoGUI [16] that accurately translates the detected hand movements into corresponding cursor movements on the screen. Through a combination of mapping hand gestures to cursor displacement vectors and adjusting sensitivity parameters, we achieved smooth and intuitive control over the cursor, with minimal latency and jitter.

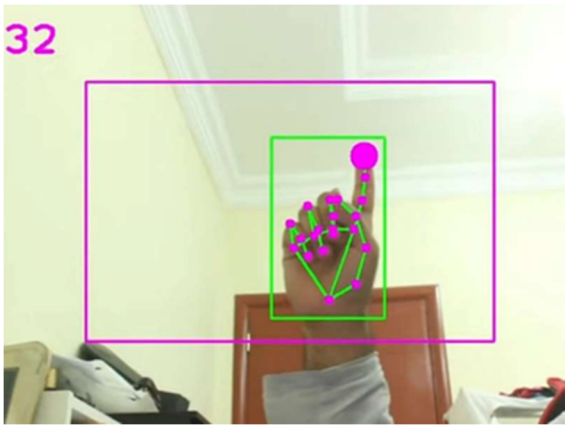

Figure 4: Mouse Move From The Proposed Method

Mouse Click:

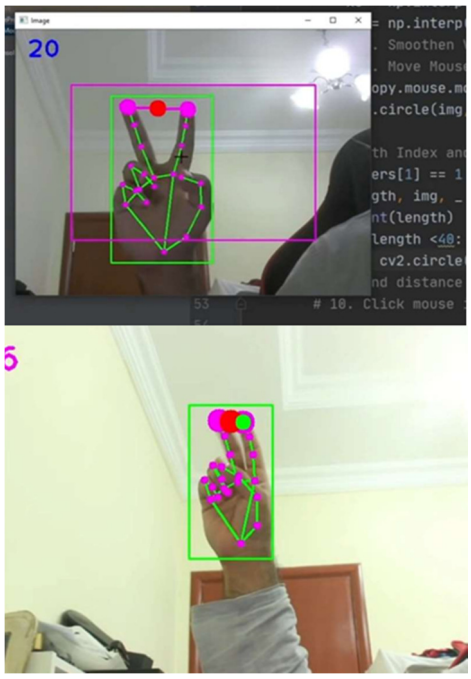

Figure 5: Mouse Click From The Proposed Method

Our system accurately measures the distance between two fingers within the detected hand region. Using computer vision techniques such as Euclidean distance calculation, to measure the distance between the index figure and the middle finger. If the distance between both the fingers is close above a stated threshold then it performs a mouse click.

Overall System Performance:

The integrated system demonstrates robust performance across various usage scenarios and environmental conditions. Real-time hand detection, accurate cursor control, precise finger distance measurement, and responsive click action triggering collectively contribute to a seamless and intuitive user experience. Through extensive testing and evaluation, we observed consistent performance with minimal instances of false positives or missed detections, affirming the effectiveness and reliability of our approach.

5. Discussion

Each method presents distinct advantages and limitations in hand tracking and gesture recognition for mouse control. Sensor-based and depth-based methods offer precise tracking but may be limited by hardware requirements and environmental constraints. CNN-based approaches provide flexibility and adaptability but require significant computational resources and data. In contrast, our proposed method offers a balance between accuracy, efficiency, and accessibility, making it well-suited for practical implementation in diverse real-world settings. However, the choice of method should be carefully considered based on specific application requirements, computational resources, and user constraints. Future research could explore hybrid approaches that combine the strengths of different methods to further enhance the performance and robustness of gesture-based interaction systems.

6. Conclusion

In this study, we explored various methods for hand tracking and gesture recognition for mouse control, comparing sensor-based, depth-based, CNN-based, and MediaPipe-based approaches [13]. Our findings demonstrate the capabilities and trade-offs associated with each method, highlighting the importance of selecting an appropriate approach based on specific application requirements and constraints. The integration of MediaPipe simplifies the development process and enhances the scalability of gesture-based interaction systems, enabling rapid prototyping and deployment. By using pre-trained models and algorithms within the MediaPipe framework, developers can create intuitive and immersive user experiences without the need for extensive training data or specialized hardware. In conclusion, the MediaPipe framework presents a promising avenue for advancing the field of hand tracking and gesture recognition, offering a solution for enabling natural and intuitive human-computer interaction in various domains, including virtual reality, gaming, education, and healthcare. Through continued innovation and collaboration, we can unlock the full potential of gesture-based interaction systems and make the start of a new era of immersive and accessible computing experiences.

References

[1] M. I. L. Antonis A. Argyros, "Vision-Based Interpretation of Hand Gestures for Remote Control of a Computer Mouse," Computer Vision in Human-Computer Interaction, vol. 3979, pp. 40-51, 2006.

[2] T. A. Sushmita Mitra, "Gesture Recognition: A Survey," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 3, pp. 311-324, 2007.

[3] A. M. S. P. D. Laura Dipietro, "A Survey of Glove-Based Systems and Their Applications," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 38, no. 4, pp. 461-482, 2008.

[4] N. A. S. S. Pragati Garg, "Vision Based Hand Gesture Recognition," World Academy of Science, Engineering and Technology, 2009.

[5] J. P. Robert Y. Wang, "Real-Time Hand-Tracking with a Color Glove," ACM Transactions on Graphics, vol. 28, no. 3, pp. 1-8, 2009.

[6] J. M. J. Y. Zhou Ren, "Depth Camera Based Hand Gesture Recognition and its Applications in Human-Computer-Interaction," 8th International Conference on Information, Communications & Signal Processing, 2011.

[7] R. R. M. Jesus Suarez, "Hand Gesture Recognition with Depth Images: A Review," IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication, 2012.

[8] K. S. N. N. N. P. Ekaterini Stergiopoulou, "Real time hand detection in a complex background," Engineering Applications of Artificial Intelligence, vol. 35, pp. 54-70, 2014.

[9] J. R. Harpreet Kaur, "A Review: Study ofVarious Techniques of Hand Gesture Recognition," IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), 2016.

[10] A. S. N. A. S. P. J. S. N. Aashni Haria, "Hand Gesture Recognition for Human Computer Interaction," Procedia Computer Science, vol. 115, pp. 367-374, 2017.

[11] H. T. Y. S. J. K. G. J. D. J. B. T. S. X. H. L. Gongfa Li, "Hand gesture recognition based on convolution neural network," Cluster Computing, vol. 22, p. 2719-2729, 2017.

[12] J. Y. Mengyuan Liu, "Recognizing Human Actions as the Evolution of Pose Estimation Maps," IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[13] F. Z. Valentin Bazarevsky, "On-Device, Real-Time Hand Tracking with MediaPipe," 19 August 2019. [Online]. Available: https://research.google/blog/on-device-real-time-hand-tracking with-mediapipe. [Accessed 6 April 2024].

[14] M. A. A. A. Norah Alnaim, "Hand Gesture Recognition Using Convolutional Neural Network for People Who Have Experienced A Stroke," 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 2019.

[15] A. A.-N. J. C. Munir Oudah, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," Journal of Imaging, 2020.