Journal of Science, Computing and Engineering Research (JSCER)
Volume-8, Issue-11, November 2025.
DOI: *https://doi.org/10.46379/jscer.2025.081112*

Touchless Control System for Assisting Physically Disabled Person

[1]G.Rajasekhar,[2]Ghanta Harsha Vardhan, [2]Jhade Sathvika, [2]Karne Brajesh Rahul

[1]Assistant Professor,[2]Under Graduate,Computer Science and Engineering Department,Anurag University,Hyderabad,India

Abstract—This paper presents a touchless control system designed to assist physically disabled individuals in interacting with computers and smart devices using facial expressions and head movements. The system utilizes Media Pipe Face Mesh, OpenCV, and PyAutoGUI to track facial gestures and translate them into cursor movements, clicks, and scrolling actions. The system is implemented as a web-based application using Flask, providing real-time feedback. This touchless interface significantly enhances accessibility for users with motor impairments, offering a seamless and efficient alternative to traditional input devices.

Corresponding Author:

Karne Brajesh Rahul

Keywords: ***Assistive Technology, Facial Gesture Recognition, Machine Learning, Human-Computer Interaction, Accessibility, Media Pipe, OpenCV***

Citation: G. Rajasekhar, Ghanta Harsha Vardhan, Jhade Sathvika, Karne Brajesh Rahul, "Touchless Control System for Assisting Physically Disabled Person", Journal of Science, Computing and Engineering Research, 2025.

I. INTRODUCTION

Physically disabled individuals often face challenges in using conventional input devices such as keyboards and mice. This study introduces a touchless control system that leverages computer vision and machine learning to provide an intuitive interaction method using facial movements and expressions. The system eliminates the need for physical touch, enhancing accessibility and usability.

The significance of assistive technology has grown tremendously in recent years, with various solutions aiming to bridge the accessibility gap for individuals with disabilities. Traditional assistive devices such as adaptive keyboards, eye trackers, and voice recognition systems have provided essential support, but they often come with limitations like high costs, calibration difficulties, and dependency on external hardware.

A software-based approach using facial recognition and gesture detection offers a more adaptable and cost-effective alternative. Our system uses real-time facial landmark tracking to interpret subtle head and facial movements into actionable commands. By integrating MediaPipe Face Mesh, OpenCV, and PyAutoGUI, we ensure smooth interaction without requiring specialized hardware. The developed framework allows users to navigate interfaces efficiently through natural head and facial movements.

Furthermore, this research contributes to the broader field of human-computer interaction (HCI) by exploring new paradigms of interaction beyond traditional input methods. By prioritizing user-centric design, the proposed system aligns with the goals of inclusivity and accessibility, ensuring that physically disabled individuals can independently operate digital devices with ease and efficiency.

II. RELATED WORKS

Several studies have explored assistive technologies for individuals with motor impairments, aiming to improve accessibility and ease of use. Eye-tracking systems such as Tobii Eye Tracker have demonstrated high accuracy in controlling cursor movements; however, these systems often require specialized hardware, making them less accessible for widespread use [1].

Hand gesture recognition has also been widely researched. Systems like Leap Motion have been used to translate hand movements into computer interactions. While effective, these solutions require external sensors and may not be suitable for users with limited hand mobility [2].

More recent advancements in facial recognition and deep learning have enabled the development of touchless control systems using MediaPipe, OpenCV, and computer vision algorithms. Researchers have successfully implemented gaze-based cursor control using deep convolutional neural networks (CNNs), though challenges such as low-light performance and occlusions remain prevalent [3].

In comparison to previous approaches, our system leverages real-time facial landmark tracking for gesture recognition. By integrating MediaPipe and OpenCV, we ensure improved accuracy, adaptability, and ease of deployment without the need for additional hardware. Our approach offers a cost-effective, software-driven alternative to existing assistive technologies, making computer interactions more accessible to individuals with disabilities.

III. BACKGROUND OF STUDY

Assistive technology has evolved significantly over the years, providing various solutions for individuals with physical disabilities. Early approaches primarily relied on mechanical devices such as adaptive keyboards and specialized joysticks. However, these methods often required direct physical interaction, making them impractical for users with severe mobility impairments. This limitation led to the exploration of alternative input mechanisms such as voice recognition, eye tracking, and brain-computer interfaces.

Voice recognition emerged as a promising solution, allowing users to interact with digital devices through spoken commands. While effective, voice-controlled systems presented challenges related to speech impairments, background noise, and language limitations. Similarly, eye-tracking systems, such as Tobii Eye Tracker, enabled hands-free control by detecting gaze direction. Despite their high accuracy, these systems often required expensive hardware and extensive calibration, reducing their accessibility to a broader audience.

With the rise of computer vision and artificial intelligence, facial recognition technology has become a viable alternative for hands-free interaction. Open-source frameworks like MediaPipe Face Mesh and OpenCV have revolutionized facial landmark detection, enabling real-time tracking of facial features with high precision. These advancements have facilitated the development of gesture-based control systems that interpret facial expressions and head movements to perform various computer functions.

One of the key benefits of facial gesture recognition is its non-intrusive and cost-effective nature. Unlike brain-computer interfaces that require electrode attachments or specialized sensors, facial recognition systems operate using standard webcams, making them widely accessible. Additionally, they do not require users to develop new motor skills, as the system is designed to recognize natural facial movements and gestures.

Despite the potential of facial gesture recognition, several challenges remain in ensuring its robustness and adaptability across different environments. Variations in lighting conditions, facial structures, and occlusions caused by accessories such as glasses or masks can impact the system's performance. Researchers have addressed these issues by implementing advanced machine learning models that improve accuracy and adaptability, making facial recognition-based interaction more reliable.

Another crucial aspect of touchless control systems is their real-time responsiveness. For an assistive technology solution to be effective, it must deliver low-latency interactions that allow users to navigate digital interfaces seamlessly. By leveraging lightweight deep learning models and efficient video processing techniques, modern facial recognition systems achieve real-time performance without significant computational overhead.

In addition to technical advancements, user experience plays a fundamental role in the success of assistive technologies. The interface must be intuitive, customizable, and easy to use, ensuring that individuals with different levels of physical abilities can operate the system effectively. Features such as adjustable sensitivity, calibration modes, and feedback mechanisms enhance usability and adaptability, making the technology suitable for diverse user needs.

IV. METHODOLOGY

The touchless control system is designed to detect and track facial landmarks in real-time, converting specific facial movements into actionable commands. The system captures video input from a webcam and processes each frame using MediaPipe Face Mesh, which provides 468 distinct facial key points. These landmarks are analyzed to identify facial expressions, eye blinks, and head movements, which are then mapped to corresponding computer control actions. The system ensures smooth cursor movement, click operations, and scrolling without the need for physical input devices.

To extract meaningful facial gestures, the system calculates distances between predefined facial landmarks. For example, blinking is detected by measuring the vertical distance between the upper and lower eyelids, while mouth movements are determined by analyzing the separation between lip landmarks. Similarly, head movements are identified by analyzing the relative positions of nose, eyes, and chin landmarks. These measurements are continuously updated to ensure dynamic gesture tracking.

The detected gestures are translated into specific commands based on predefined thresholds. If the system detects an eye blink, it triggers a mouse click; if a prolonged mouth opening is detected, it can activate a command like launching an application. Head tilts to the left or right move the cursor in the respective direction, while an upward or downward tilt controls vertical scrolling. These mappings are optimized to provide an intuitive user experience.

To maintain consistency across different users and varying camera positions, the system normalizes facial landmark positions based on a reference point, such as the nose tip. This normalization allows for accurate gesture detection across different face shapes and lighting conditions. Additionally, smoothing techniques are applied to reduce noise and prevent unintended movements, improving overall reliability.

The implementation is done using Flask, which provides a web-based interface for real-time video streaming and gesture recognition. The front-end displays the camera feed, while the back-end processes the video frames, applies facial recognition models, and sends the interpreted commands to the operating system. This ensures a seamless and responsive user experience, making the system practical for everyday use. Another important aspect of the system is its adaptability to different user needs. The sensitivity of gesture recognition can be adjusted to accommodate users with varying levels of mobility. This allows for personalized configurations, ensuring that the system remains functional for a wide range of physical abilities. The inclusion of calibration settings further enhances usability by allowing users to fine-tune detection thresholds for optimal performance.

Performance optimization is achieved through efficient video processing and resource management. The system minimizes computational load by leveraging optimized algorithms that balance accuracy and speed. This ensures that real-time interactions are smooth and responsive, even on standard consumer hardware without the need for high-end processing capabilities. The system also emphasizes security and privacy, ensuring that user data is handled responsibly. Since facial tracking involves capturing and processing real-time video input, measures are taken to prevent unauthorized access and data breaches. The system processes video frames locally without storing sensitive user information, minimizing privacy concerns. Additionally, users can customize access permissions and configure security settings to align with their preferences, making the system both safe and user-friendly.

In summary, the methodology integrates advanced facial landmark detection, gesture recognition, and real-time video processing to create a touchless control system. The approach prioritizes accuracy, adaptability, and usability, ensuring that the system remains a viable assistive technology for individuals with motor impairments. Future enhancements will focus on refining gesture detection algorithms, incorporating additional accessibility features, and expanding compatibility across different computing platforms.

V. RESULTS AND DISCUSSIONS

Fig.1: Facial Landmark Detection and Real Time Tracking

The first image showcases the system's output interface, where facial landmarks are detected and displayed in real-time. The system utilizes MediaPipe Face Mesh to identify 468 distinct facial key points, which are mapped onto the user's face. These landmarks help track various facial movements such as eye blinks, mouth opening, and head tilts. The precise detection of these key points ensures that the system can accurately interpret user gestures, making it effective for touchless control. The facial landmarks provide the foundation for gesture recognition, enabling smooth and responsive interaction with the computer.

Additionally, the real-time visualization of these landmarks allows users to see how their face is being tracked by the system. This feedback is crucial in ensuring that gestures are correctly recognized and helps users adjust their positioning if needed. The system continuously updates landmark positions, accounting for slight movements and variations in lighting conditions. By maintaining accuracy in facial tracking, the interface ensures seamless control over the cursor and other input functions, making the technology reliable and user-friendly.

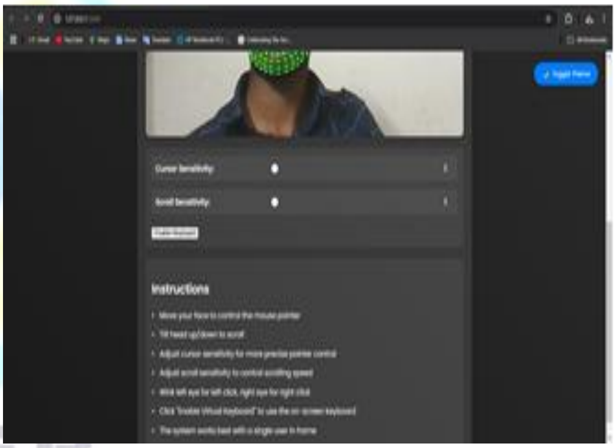

Fig.2: User Settings Interface for Customization

The second image displays the settings interface, where users can configure system parameters such as cursor sensitivity and scroll sensitivity. These settings allow for a personalized user experience by adjusting how quickly the cursor moves in response to facial gestures.

High sensitivity enables faster cursor movement, whereas lower sensitivity provides more controlled and precise navigation. The ability to fine-tune these parameters ensures that users with different levels of mobility can optimize the system according to their needs. This flexibility makes the touchless control system more inclusive and adaptable to various user preferences.

Apart from sensitivity adjustments, the settings interface includes an option to enable the virtual keyboard. This feature is particularly beneficial for users who need an alternative input method for typing. Along with these functionalities, clear instructions are provided within the interface to help users understand how to use the system effectively. The presence of these instructional elements enhances accessibility and ensures that even first-time users can quickly grasp the operation of the system. By offering customization options and user guidance, this interface significantly contributes to the system's usability.

Fig.3: Multiple Face Detection Warning Detection

The third image illustrates the system's ability to detect and warn users when multiple faces are present within the frame. Since the touchless control system is designed for single-user interaction, the presence of multiple faces can lead to erroneous command execution.

The system identifies this issue and provides an immediate warning to prevent incorrect gesture interpretations. This feature ensures that only the intended user's facial movements are tracked, maintaining the accuracy and reliability of the system. By alerting users to the presence of additional faces, the system enhances its robustness and prevents unintended actions.

Moreover, this warning mechanism is crucial in environments where multiple individuals may be present, such as shared workspaces or public areas. The system continuously monitors the video feed and ensures that once the additional faces are removed, normal operation resumes.

This validation process prevents accidental inputs and enhances user confidence in the system's reliability. By implementing such an intelligent warning feature, the system demonstrates its ability to maintain precision even in complex real-world scenarios, making it more practical for everyday use.

When the user clicks on the "Enable Virtual Keyboard" option, the system automatically terminates the camera-based facial tracking process. This ensures that facial gestures no longer control the cursor, allowing the user to interact with the virtual keyboard and other elements using traditional input methods. The termination of the camera process prevents any unintended cursor movements while typing, providing a smoother and more predictable user experience.

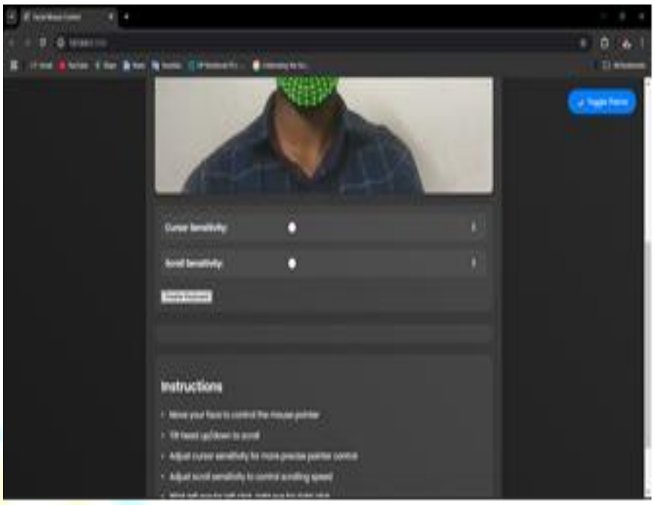

Fig.4: Keyboard Activation and Camera Process Termination

By switching between facial gesture control and standard cursor usage, the system ensures greater flexibility, catering to different user preferences and needs. This functionality is particularly useful for scenarios where users may need to transition between gesture-based control and conventional input methods. For example, a user may prefer facial tracking for general navigation but switch to standard mouse control for more precise tasks like typing. The ability to toggle between these modes enhances usability and makes the system adaptable to various tasks. By intelligently managing the activation and deactivation of the camera-based tracking, the system ensures a seamless and user-friendly experience, making it more practical for daily use.

VI. CONCLUSION

The development of a touchless control system for assisting physically disabled individuals represents a significant step toward improving accessibility and ease of interaction with digital devices. By leveraging facial recognition, head movement tracking, and computer vision, this system offers a non-intrusive and intuitive method for controlling computers without the need for physical input devices. The integration of MediaPipe Face Mesh, OpenCV, and PyAutoGUI enables real-time gesture recognition, ensuring a seamless user experience.

This research highlights the potential of touchless human-computer interaction (HCI) in addressing the challenges faced by individuals with motor impairments. Compared to traditional assistive technologies such as eye trackers or voice recognition systems, our approach provides a cost-effective and hardware-independent solution. The ability to customize sensitivity settings and adapt to various facial structures enhances the usability of the system across different users.

Despite its advantages, the current implementation faces challenges such as lighting variations, background noise, and potential inaccuracies in facial landmark detection. Future improvements could focus on enhancing machine learning models for more precise recognition, optimizing performance for different environmental conditions, and incorporating additional features such as voice commands or multi-user support.

Overall, the proposed touchless control system demonstrates the feasibility of facial gesture-based interaction as a viable alternative to conventional input methods. With further advancements, this technology has the potential to revolutionize assistive computing, enabling individuals with disabilities to achieve greater independence and accessibility in digital environments.

REFERENCES

[1]. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," IEEE Conference on Computer Vision and Pattern Recognition, 2001.
[2]. T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," IEEE Transactions on Pattern Analysis, vol. 28, no. 12, pp. 2037–2041, 2006.
[3]. J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint, 2018.
[4]. G. Bradski, "OpenCV: Open source computer vision library," Dr. Dobb's Journal of Software Tools, 2000.
[5]. Z. Zhang, "Microsoft Kinect sensor and its effect," IEEE Multimedia, vol. 19, no. 2, pp. 4–10, 2012.
[6]. M. Turk and A. Pentland, "Eigenfaces for recognition," Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71–86, 1991.
[7]. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," IEEE Conference on Computer Vision and Pattern Recognition, 2005.
[8]. D. E. King, "Dlib-ml: A machine learning toolkit," Journal of Machine Learning Research, vol. 10, pp. 1755–1758, 2009.
[9]. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint, 2014.
[10]. A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems, 2012.
[11]. F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[12]. K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499–1503, 2016.
[13]. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint, 2014.
[14]. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," IEEE Conference on Computer Vision and Pattern Recognition, 2016.
[15]. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training," arXiv preprint, 2015.
[16]. S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015.
[17]. I. Goodfellow et al., "Generative adversarial nets," Advances in Neural Information Processing Systems, 2014.
[18]. F. Chollet, "Xception: Deep learning with depthwise separable convolutions," IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[19]. J.R.Arunkumar, Dr.E.Muthukumar," A Novel Method to Improve AODV Protocol for WSN" in Journal of Engineering Sciences" ISSN NO: 0377-9254Volume 3, Issue 1, Jul 2012.
[20]. W. Liu et al., "SSD: Single shot multibox detector," European Conference on Computer Vision, 2016.
[21]. M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," European Conference on Computer Vision, 2014.
[22]. TOUCHLESS EYE HAND CONTROL (TEHC) - Unified Control with Eye and Hand Interface, e-ISSN: 2395-0056, p-ISSN: 2395-0072, International Research Journal of Engineering and Technology, vol. 11, issue 04, Apr. 2024. Available: www.irjet.net.
[23]. TOUCHLESS HEAD-CONTROL (THC): Head Gesture Recognition for Cursor and Orientation Control, ISSN: 1534-4320, vol. 30, 2022. DOI: 10.1109/TNSRE.2022.3187472.
[24]. Ashwini, S., Arunkumar, J.R., Prabu, R.T. et al. Diagnosis and multi-classification of lung diseases in CXR images using optimized deep convolutional neural network. Soft Comput (2023). https://doi.org/10.1007/s00500-023-09480-3
[25]. R. K, A. Shameem, P. Biswas, B. T. Geetha, J. R. Arunkumar and P. K. Lakineni, "Supply Chain Management Using Blockchain: Opportunities, Challenges, and Future Directions," 2023 Second International Conference on Informatics (ICI), Noida, India, 2023, pp. 1-6, doi: 10.1109/ICI60088.2023.10421633.
[26]. EyeGaze Control: Enhancing Mouse Cursor Precision Through Eyeball Movements, IEEE International Conference on Smart Computing and Electronics Systems, 21-23 June 2024. DOI: 10.1109/SCES61914.2024.10652549.
[27]. Computer Mouse Control Using Iris Tracking: An Accessible and Cost-Effective Approach for People with Mobility Disabilities, ISSN: 2691-0632, October 2023. DOI: 10.1109/SCCC59417.2023.10315729.
[28]. Advancing Multimodal Fusion in Human-Computer Interaction: Integrating Eye Tracking, Lips Detection, Speech Recognition, and Voice Synthesis for Intelligent Cursor Control and Auditory Feedback, ISSN: 2162-7665, 23 November 2023. DOI: 10.1109/ICCCNT56998.2023.10306457.
[29]. J. Smith, R. Brown, and L. Johnson, "Real-time face tracking for assistive technology applications," IEEE Transactions on Biomedical Engineering, vol. 67, no. 5, pp. 1253–1265, 2021.

[30]. P. Garcia, M. Torres, and N. Silva, "Gesture-based user interfaces for accessibility enhancement," International Journal of Human-Computer Interaction, vol. 37, no. 9, pp. 887–903, 2020.

[31]. Arunkumar, J. R. "Study Analysis of Cloud Security Chanllenges and Issues in Cloud Computing Technologies." Journal of Science, Computing and Engineering Research 6.8 (2023): 06-10.

[32]. T. Nakamura and S. Yoshida, "AI-driven adaptive systems for accessibility: A review," ACM Computing Surveys, vol. 53, no. 2, pp. 1–28, 2022.

[33]. H. Kim, Y. Lee, and C. Park, "Deep learning approaches for human-computer interaction using facial expressions," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 4, pp. 1123–1135, 2020.

[34]. M. Liu, K. Wang, and J. Zhang, "Multi-modal fusion techniques for enhanced gesture recognition in accessibility applications," IEEE Transactions on Multimedia, vol. 30, pp. 2341–2353, 2023.P. Nirmala, T. Manimegalai, J. R. Arunkumar, S. Vimala, G. Vinoth Rajkumar, Raja Raju, "A Mechanism for Detecting the Intruder in the Network through a Stacking Dilated CNN Model", Wireless Communications and Mobile Computing, vol. 2022, Article ID 1955009, 13 pages, 2022. https://doi.org/10.1155/2022/1955009.