

Intelligent Wheelchair Control System Using Real-Time Head Movement Detection

Hanan Jabbar Abdulkareem^{1*}, Sadik Kamel Gharghan², Saad Mutashar³

¹ Electrical Engineering Department, University of Technology, Baghdad 10001, Iraq

² Electrical Engineering Technical College, Middle Technical University, Baghdad 10001, Iraq

³ Medical Instrument Engineering and Technical Department, Al-Mustaqbal University, Hillah 51001, Iraq

Corresponding Author Email: eee.21.03@uotechnology.edu.iq



Copyright: ©2025 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/mmep.120717>

ABSTRACT

Received: 10 December 2024

Revised: 12 March 2025

Accepted: 21 March 2025

Available online: 31 July 2025

Keywords:

Dlib library, head movement, HMI, landmarks, OpenCV2 library, Raspberry Pi, wheelchair

Many individuals with severe physical disabilities face difficulties in controlling electric wheelchairs due to reliance on manual input methods such as joysticks, which are impractical for users with limited mobility. This study aims to develop a real-time, cost-effective, and non-invasive system that enables hands-free wheelchair control using head movement detection. The proposed system leverages a Raspberry Pi, a camera, and Dlib's facial landmark detection to track specific facial points (27 and 28), accurately detecting head tilts and turns. Unlike existing solutions that rely on expensive or invasive sensors, this approach ensures accessibility while maintaining high accuracy. The system was tested under various conditions, achieving 99.8% accuracy, demonstrating its reliability and practicality. The presented research has developed affordable assistive technology to improve the independent movement capabilities for disabled people. The research highlights practical deployment possibilities that lead to better developments in adaptive wheelchair control systems.

1. INTRODUCTION

Assistive technologies have enhanced mobility and self-sufficiency for people with severe physical disabilities through their development. Electric wheelchairs constitute one of the most popular solutions in assistive technology because they use traditional joysticks or buttons-based control [1]. Wheelchair control methods prove challenging and inaccessible to people who have neuromuscular disorders combined with spinal cord injuries or any form of motor impairment. Increased demand exists for natural wheelchair-control systems, which offer users non-intrusive manual-independent operation at low equipment costs. The present hands-free wheelchair control systems, which include eye-gaze tracking voice command operation and motion-based detection features, demonstrate various operational restrictions. Infrared camera equipment for eye-gaze tracking systems carries high costs and is sensitive to lighting environment conditions [2, 3]. Users with both issues must contend with unreliable voice command operation in loud spaces and erroneous accelerometer detection of casual head motions that diminish system accuracy [3, 4]. Some systems rely on wearable sensors or head-mounted devices [5, 6], which can be uncomfortable, invasive, and expensive, making them unsuitable for long-term daily use [7].

Current wheelchair control technologies exist without either a precise solution, low costs, or an unobtrusive design. A multitude of current systems do not achieve precise control, and there are unaffordable hardware requirements and complicated calibration methods that hinder their use in

practical environments. The research develops a low-cost, real-time, non-invasive approach for detecting wheelchair user head movements through facial landmark tracking technology [8] and low-cost hardware [9].

The proposed system introduces a camera-based head movement detection approach that eliminates the need for wearable sensors, ensuring comfort and ease of use. Using Dlib's facial landmark tracking it achieves high accuracy and robustness while a novel decision-making process eliminates unintentional movements, enhancing reliability. This approach outperforms previous methods in cost, efficiency, and user-friendliness, making it a practical and affordable solution for wheelchair control.

Unlike previous methods, this system introduces a camera-based head movement detection approach. The contributions of this paper are as follows:

- 1) Developing a low-cost hardware and open-source software head movement detection system, which contributes to an inexpensive and accessible environment for many users.
- 2) The system detects head movements with precision even in various environmental conditions; thus, it is reliable and responsive.
- 3) An intuitive, non-invasive solution that does not require any complex or uncomfortable equipment was presented.
- 4) The present work is compared with various previous works, and the system has performed much better compared to the other solutions discussed in this research by achieving an accuracy rate of 99.8%.

The paper is organized as follows: Section 2 discusses related works in assistive technology and head movement detection. Section 3 discusses the materials and methods used in the proposed system, which elaborates on the head movement detection mechanism and wheelchair control approach. Section 4 elaborates on the use of a webcam and Dlib for movement detection. Section 5 outlines the main algorithm applied in this study, while Section 6 explains the protocol for the wheelchair control system. Section 7 presents the validation and testing processes for the system. Section 8 assesses the system's accuracy and responsiveness across different scenarios. Section 9 provides a comparative analysis with other systems. Finally, Section 10 concludes the paper by summarizing key findings and proposing directions for future improvements.

2. RELATED WORKS

Several studies have explored alternative wheelchair control mechanisms, leveraging accelerometers, vision-based approaches, and deep learning techniques. However, each approach presents limitations in terms of accuracy, cost, ease of use, and practicality. This section critically reviews existing methods and highlights how the proposed camera-based head movement detection system addresses these shortcomings. Accelerometer-based head movement detection is a common approach in wheelchair control.

2.1 Accelerometer-based systems

Accelerometer-based head movement detection is a widely used approach in wheelchair control. Nizar and Jabbar [10] developed a tilt-sensor-based system that tracks the angles of a user's head, allowing wheelchair navigation. Similarly, Mangla et al. [11] utilized MPU-6050 sensors, combining an accelerometer and gyroscope, to detect head tilts in real time. However, Tambakhe et al. [12] reported that accelerometer-based head motion detection can misinterpret unintentional head movements, leading to unintended wheelchair actions. Pourmirzaei et al. [13] proposed a system capable of functioning even when the user's head is at unfavorable pitch, yaw, or roll angles, which often pose challenges for most webcam-based control applications. Bui et al. [14] developed an accelerometer-based system that achieved 97% accuracy but highlighted user discomfort due to the presence of external sensors. Kujani and Kumar [15] implemented a microcontroller-based accelerometer wheelchair system with 90% accuracy; however, it lacked adaptability to complex environments, limiting its practical usability. These accelerometer-based systems, while effective to some extent, exhibit several drawbacks. They are prone to false detections caused by involuntary head movements, require user calibration, which increases setup complexity, and rely on external sensors, which can be uncomfortable and expensive. Our proposed system eliminates the need for external sensors by utilizing camera-based facial landmark tracking, which ensures higher accuracy 99.8%, lower cost, and greater comfort for the user.

2.2 Vision-based systems

Vision-based methods employ cameras and computer vision algorithms to track head movements. Chatzidimitriadis et al.

[16] applied computer vision techniques to detect head movements for wheelchair control, while Roig-Maimó et al. [17] explored camera-based head-tracking for cervical rehabilitation, demonstrating its potential for non-invasive control. Alcantara et al. [18] implemented a Raspberry Pi and OpenCV-based head-tracking system for device control but encountered limitations in processing speed and real-time responsiveness. Sayeed et al. [19] used webcam-based head and eye movement detection but achieved only 89% accuracy, which was insufficient for precise wheelchair control. Sandhya et al. [20] integrated ultrasonic sensors with Raspberry Pi for autonomous wheelchair navigation, but struggled with complex environments. While these vision-based approaches reduce the dependency on physical sensors, they remain susceptible to environmental sensitivity, where lighting conditions significantly affect detection performance. Additionally, slow processing speeds hinder real-time responsiveness, and their accuracy often lags behind deep learning-based models. Our proposed system overcomes these issues by leveraging Dlib's advanced facial landmark tracking, which enhances robustness under varying lighting conditions and ensures real-time responsiveness using lightweight hardware.

2.3 Deep learning-based systems

Deep learning-based approaches have gained prominence in head movement detection due to their ability to extract and learn intricate patterns. Feng et al. [21] proposed a CNN-based model for facial landmark detection, demonstrating high accuracy in real-time applications. Zhang et al. [22] developed a deep multi-task learning framework combining head pose estimation and facial expression recognition, achieving greater accuracy and robustness. Zamir et al. [23] introduced a CNN-based facial recognition system that attained 98.3% accuracy, though it was primarily focused on security applications rather than wheelchair control. Phayde et al. [24] designed a deep learning-based motion detection system but only achieved 91% accuracy, making it inadequate for precise mobility assistance. Zeng et al. [25] explored wearable sensor-based deep learning methods but noted that their high computational demands limited real-time wheelchair control capabilities.

Garg et al. [26] investigated the application of optical flow analysis to detect head gestures in video, while Zarkasi et al. [27] explored template-matching techniques for detecting subtle head movements in human-computer interaction systems. Prasad et al. [28] built a wheelchair control system using head movements detected by an ATmega328p microcontroller, achieving 90% accuracy but suffering from low processing power and limited adaptability to complex environments. Hassain et al. [29] developed an electric wheelchair control system based on head gestures, detecting forward, backward, left, and right movements with 92% accuracy; however, it struggled with subtle head movements. Deep learning methods deliver exceptional accuracy, but they impose limitations, which include substantial computational needs that obstruct embedded systems usage and extensive training datasets combined with difficulties in running applications in real-time on Raspberry Pi [30]. Our proposed system delivers 99.8% accuracy alongside low-computational processing needs for facial landmark tracking, which allows it to remain practical for real-time embedded applications.

Studies that investigated wheelchair control through head movements have demonstrated useful findings; however,

these solutions face multiple key limitations comprising of deficient detection precision, the need for external sensor implementation, and sensitivity to environmental circumstances and processed data requirements. Our system works without accelerometer-based systems because it lacks dependency on external sensors, which removes discomfort and the need for calibration procedures. Our system employs Dlib's facial landmark tracking, making it more robust against varying lighting conditions. Furthermore, unlike deep learning-based models that demand substantial computational power, our approach operates efficiently on low-power devices like the Raspberry Pi, achieving 99.8% accuracy in real time. Our system achieves cost-effectiveness since it lowers hardware costs through the combination of Raspberry Pi middle standard webcams, making accessibility simpler for users with disabilities.

A research paper presents an algorithm detecting real-time head motions that employ facial landmark recognition technology made specifically for electric wheelchairs. The present system tracks facial landmarks and recognizes head movements such as tilts, nods, and shakes through real-time processing without invasive hardware requirements. The removal of conventional sensor-based processes leads to procedural simplification while reducing the invasiveness of equipment and creating user-friendly solutions. This system uses advanced computer vision methods to provide instant, accurate communication alongside economical processing requirements. This innovation presents a more accessible, efficient, and cost-effective solution for individuals requiring hands-free wheelchair control.

3. MATERIALS AND METHODS

The proposed system's efficiency is illustrated by presenting the power of the hardware used and the dataset added in the following sections.

3.1 System configuration

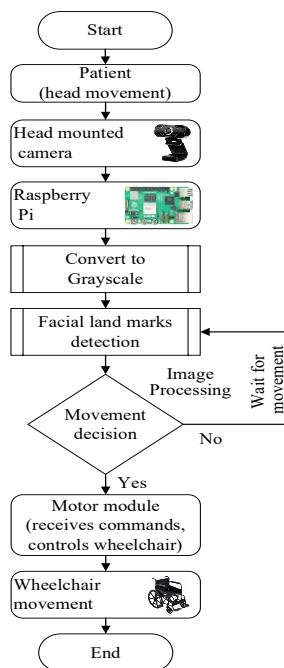


Figure 1. Flowchart of head movement detection and wheelchair control using facial landmark detection

The proposed system includes a person, a webcam, a Raspberry Pi 3, and a monitor, as shown in Figure 1. The camera has been installed on the head to limit unnecessary head movements and avoid introducing artifacts in the relevant data [31]. For optimal facial landmark identification during recording, the camera should be positioned between 20 to 60 centimeters from the individual's face [32, 33]. Through testing, the optimal distance was determined to be 30 centimeters. The 5-megapixel camera used in the system supports resolutions of 1080 p at 30 fps, 720 p at 60 fps, and 640×480 p at 90 fps [34]. Controlling the wheelchair involves capturing video, converting it into images (frames), processing the image data, and transmitting the frames to the Raspberry Pi 3 for analysis. This is accomplished by installing and utilizing necessary libraries, such as OpenCV2, Dlib, Numpy, and Pandas [24].

3.2 Implementation strategy for head movement detection

The paper methodology employed to detect head movements in real-time uses computer vision techniques. The process involves the following core steps: live data acquisition, face detection and tracking, facial landmark detection, head movement analysis, evaluation methodology, implementation details, and ethical considerations. The proposed method is implemented in real-time using a live camera that captures video frames of subjects' faces. This real-time feed enables the acquisition of dynamic, continuously updated data throughout the experiment. Upon receiving each video frame from the live camera feed, face detection algorithms are applied to identify faces within the frame. Once identified, a face-tracking algorithm ensures continuity by tracking these faces across successive frames in the video stream.

In the case of real-world applications, landmark detection techniques detect corners of the mouth, eyebrows, and nose tip as the key facial features. The proposed system specifically tracks facial landmarks 27 and 28, which are positioned at the top of the nose bridge. These points are ideal for detecting head movement because they are centrally located on the face and remain stable relative to natural head tilts and rotations. The system calculates the displacement of these points along the X and Y axes to determine head movement direction:

- (1) X-axis displacement: Left or right head movement.
- (2) Y-axis displacement: Upward or downward movement.
- (3) Combined displacement: Diagonal or mixed movements.

By focusing on points 27 and 28, the system ensures robust tracking with minimal errors, avoiding potential inaccuracies that may arise from eyebrows (which may move independently) or chin points (which may shift due to facial expressions). This is done efficiently with the Dlib library in Python since it is fine-tuned for facial landmark detection. It does this by analyzing the head movements based on the displacement of facial landmarks in real-time. The different optical flow estimation and template matching methods track its motion between consecutive frames, enabling it to detect small head movements accurately. Performance is evaluated using standard metrics for accuracy, precision, and recall within an experimental testbed that examines responsiveness under various conditions. A real-time head movement detection system has been implemented in Python, leveraging existing libraries and frameworks to streamline development and functionality. Significant emphasis has been placed on optimizing the software architecture to ensure efficient and

responsive processing of live camera feeds. This approach aims to deliver reliable performance in real-world scenarios, prioritizing accuracy and responsiveness.

3.3 Collected data

Experimentation with the proposed head movement tracking system to control an electric wheelchair generated the collected data. Selection had to occur before the experiment based on established criteria to guarantee precise, measurable head movements from the subjects. The projected input system required these methods to evaluate system performance and reliability since they would establish steady and quantifiable experimental data.

- (1) Number of samples: 448 samples were recorded for head movements, including forward, backward, left, and right. Recordings were done under various changes in environmental parameters for comprehensive system evaluation in terms of stability and adaptability to changes in lighting conditions and viewing angles.
- (2) Type of data: Collected data included head angular displacements and face feature points. A high-resolution camera paired with a Raspberry Pi system was used to capture video and analyze these movements in real time, ensuring accurate and detailed tracking.
- (3) Stages of data collection:
 - Performing head movements in the specified directions for 3 seconds per movement.
 - Recording the angular displacement and precise facial landmark positions using the Dlib algorithm.
 - Recording system outputs based on the wheelchair movement commands derived from the head movements.

3.4 Face detection

Dlib and OpenCV2 are used within a "For Loop" to load each frame from the video and detect facial landmarks in the image. Once a face or landmark is identified, a new landmark is generated inside the loop. The next step involves converting the color frame to grayscale, simplifying the detection of facial features [35]. Figure 2 illustrates the facial landmarks identified in a grayscale frame.

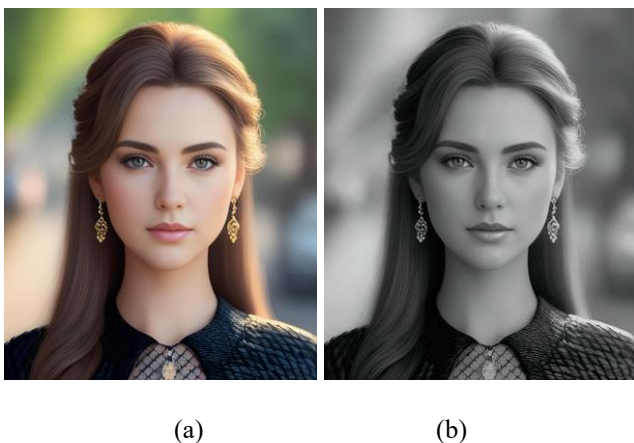


Figure 2. Face detection: (a) Color; (b) Grayscale

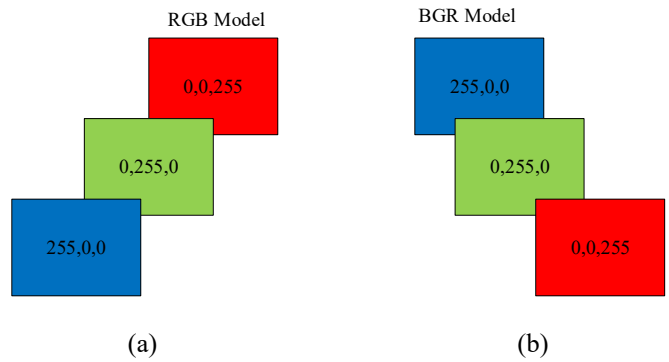


Figure 3. Models of image processing: (a) RGB model and (b) BGR model of OpenCV

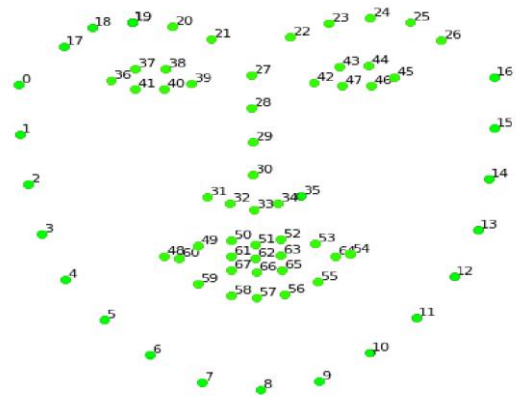


Figure 4. Landmarks according to Dlib library [1]

A digital image on a screen is displayed using a matrix of pixels, where an integer value represents each pixel. Grayscale images are generated by assigning numerical values from 0 (black) to 255 (white) to these pixels. The RGB color model is the most common for representing colors, but it is important to note that OpenCV2 loads color images in the reverse order, specifically BGR (Blue, Green, Red) [36], which is demonstrated in Figure 3.

The human face has 68 fixed landmarks that are reference points for detecting facial features [37]. By knowing the indexes of these points, one can focus on specific facial areas like the eyes, lips, eyebrows, nose, or ears. This article particularly emphasizes points 27 and 28, located on the edges of the nose, to analyze and determine head tilt movement, as illustrated in Figure 4.

4. DETECTING MOVEMENT BASED ON WEBCAM USING DLIB

Real-time data acquisition is performed using a live camera feed, capturing video frames of subjects' faces. This live stream ensures the acquisition of dynamic and continuously updated data for each time instance during the experiment. Figure 5 explains the detecting signal in the following sequence upon receiving each video frame from the live camera feed. Face detection algorithms are applied to localize faces within the frame. Following detection, a face-tracking technique is employed to track the detected faces across consecutive frames, ensuring continuity in tracking

throughout the video stream. Landmark detection techniques are utilized to identify key points on the face in real-time. These landmarks, such as the corners of the mouth, eyebrows, nose tip, and localization of points (27 and 28), are detected for each face in every video frame, enabling precise tracking of facial features over time using the Dlib library.

The movement of facial landmarks is continuously analyzed in real-time to infer head movements by applying points that detect the eyes, chin, nose, and other facial points. Techniques such as optical flow estimation or template matching are applied to track the displacement of facial landmarks between consecutive frames, allowing for the detection of head movements.

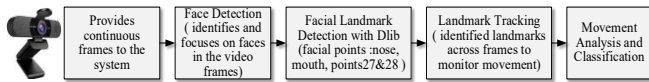


Figure 5. Real-time facial landmark detection and head movement analysis

5. REAL-TIME HEAD MOVEMENT DETECTION ALGORITHM

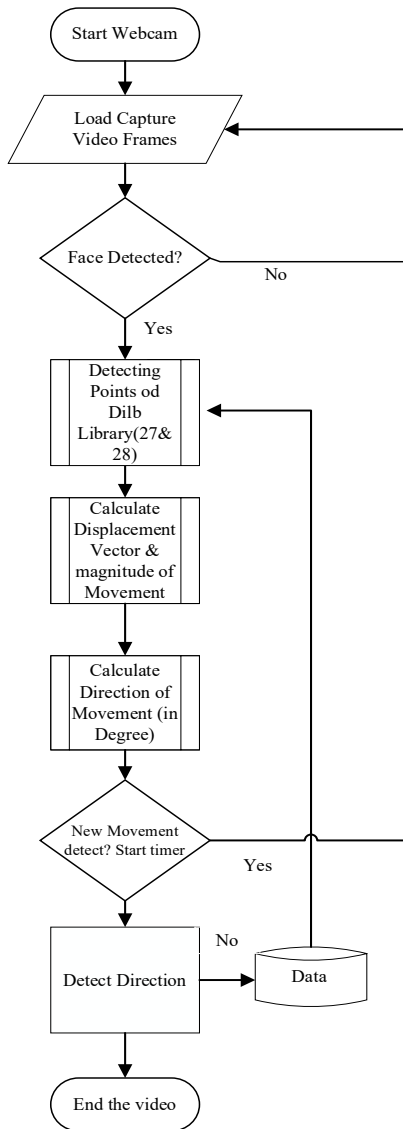


Figure 6. Flowchart of real-time head movement detection algorithm

The real-time head movement detection algorithm is divided into several primary stages. At the start, the software initializes to capture a video frame from the webcam and employs a "While Loop" to load a frame in a valid state, reloading it in case of an error. The system detects the user's face within the video feed and converts it to grayscale to enhance facial landmark detection. Key facial features-eyes, chin, and nose points-are detected using OpenCV2 and Dlib libraries, with particular emphasis on landmarks 27 and 28 located on the nose bridge. These points are ideal for detecting head movement because they remain stable relative to natural head tilts and rotations.

Once facial landmarks are detected, the system calculates displacement vectors along the X and Y axes to determine head movement direction:

- X-axis displacement: Indicates left or right movement.
- Y-axis displacement: Indicates upward or downward movement.
- Combined displacement: Represents diagonal or mixed movements.

The system evaluates movements over a 3-second window, ensuring that only intentional and sufficiently long gestures are accepted as valid inputs. The system's final decision is used to generate a corresponding wheelchair command, moving the wheelchair in the desired direction, as shown in Figure 6.

5.1 Filtering intentional movements

To prevent unintended activations, the system introduces a dual-layer filtering approach:

- (1) Time-based thresholding:
 - A movement must be sustained for at least 3 seconds before being classified as intentional.
 - This prevents quick head jerks, minor shifts, or tremors from triggering unwanted wheelchair movements.
 - If the head returns to its neutral position before 3 seconds, the movement is ignored.
- (2) Displacement-based filtering:
 - The system measures the displacement of landmarks 27 and 28 and applies predefined thresholds:
 - Left/Right Movement: X displacement must be greater than ± 200 pixels.
 - Forward/Backward Movement: Y displacement must be greater than ± 120 pixels.

Movements below these thresholds are ignored, ensuring only substantial, intentional gestures trigger a response.

By combining time-based filtering with displacement validation, the system minimizes false activations and enhances accuracy.

5.2 Noise filtering and external disturbance management

Environmental noise and sensor disturbances can introduce errors in head movement detection. The system employs the following techniques to maintain robust and stable tracking:

- (1) Applies a sliding window filter to smooth displacement variations, reducing the effects of sudden noise or small involuntary movements.
- (2) The system dynamically adjusts displacement thresholds based on the user's movement behavior.
 - If a user consistently performs subtle

movements, the threshold decreases for better responsiveness.

- If external noise (lighting variation, small head tremors) is detected, the threshold increases to prevent misclassification.

(3) Uses histogram equalization in OpenCV to normalize contrast and ensure consistent landmark tracking under varying lighting conditions.

Prevents errors caused by shadows, glare, or dim environments.

These filtering mechanisms ensure high detection reliability, even in non-ideal conditions.

5.3 Decision tree-based classification

The system applies decision tree classification to process filtered movements following noise reduction. A predefined displacement threshold system in the classifier decides what constitutes movement classes, according to Figure 7.

Classification Rules:

- Left Movement: If X displacement < -200 and sustained for 3 seconds, classify as Left.
- Right Movement: If X displacement > 200 and sustained for 3 seconds, classify as Right.
- Forward Movement: If Y displacement > 120 and sustained for 3 seconds, classify as Forward.
- Backward Movement: If Y displacement < -120 and sustained for 3 seconds, classify as Backward.
- No Movement: If both X and Y displacements are within the neutral range ($-100 \leq X \leq 100$, $-50 \leq Y \leq 50$), classified as No Movement.

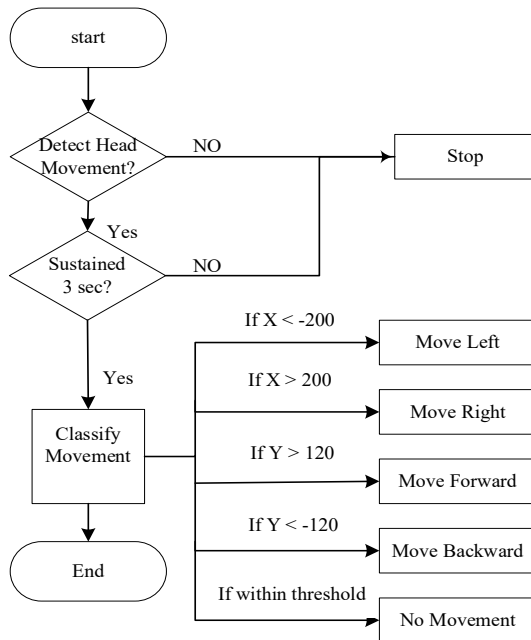


Figure 7. Decision tree-based classification of head movements for wheelchair control

5.4 Adaptive user thresholding

The system can be tailored to different user needs by adjusting movement thresholds dynamically:

- Users with slower response times (e.g., neuromuscular disorders)→Require a longer threshold (3.5-4 sec) to accommodate slower

movements.

- Users with fast response ability→Prefer a shorter threshold (2 sec) for quicker wheelchair control.
- Adaptive learning mode→The system adjusts thresholds over time based on user behavior to optimize responsiveness.

By incorporating adaptive tuning, the system remains flexible, accessible, and highly responsive to individual needs.






6. WHEELCHAIR CONTROL SYSTEM PROTOCOL

Facial landmarks are detected using Dlib, such as points of eye position, edge of chin, and nose. Tracking the movement of these points can determine specific types of head movements or facial expressions. The analysis of these points is to calculate the vertical displacement of points 27 and 28 between consecutive frames where if the y-coordinate of either point increases, it indicates upward movement, and if it decreases, it indicates downward movement; otherwise, calculate the horizontal displacement of points 27 and 28 between consecutive frames where if the x-coordinate of point 27 increases while the x-coordinate of point 28 decreases, it indicates horizontal movement.

The system interprets detected head movements based on displacement thresholds and duration criteria to generate control commands. For example, if vertical movement is detected while the person is speaking, it might indicate nodding in agreement. This interruption can be overcome by still being in the required direction for three seconds before detecting the direction or movement by applying thresholds to the displacement values to filter out noise and detect significant movements only. For instance, set a threshold for minimum displacement to distinguish intentional movements from minor fluctuations due to noise. The display screen is divided into four quarters by x-axes and y-axes. Firstly, it detects the facial landmarks and detects if the head is moving vertically up and down. When the eye crosses the (0,0) point as a threshold, the head moves down, equal to backward. For forward, when the chin edge is crossed the (0,0) as a threshold point, the head moves to the direction that gives as forward. Secondly, the head movement is detected horizontally by detecting the eye's position. If the two eyes go to the first quarter, the head moves right, then gives the right direction, and for the left, the eyes go to the second quarter.

Table 1 illustrates the head movement protocol of the wheelchair control system. The head movements are detected using specific axis displacements, triggering wheelchair control commands; Left lateral flexion (0 to 45 degrees) results in a left movement command when the X-axis displacement is between < 0 and ≤ -200 , with the Y-axis remaining at 0. Right lateral flexion (135 to 180 degrees) triggers a right movement command when the X-axis displacement is between < 0 and ≤ 200 , with the Y-axis remaining at 0. Extension (-90 degrees) results in a forward movement command when the Y-axis displacement is > 0 and ≥ -120 , with the X-axis at 0. Flexion (90 degrees) triggers a backward movement command when the Y-axis displacement is > 0 and ≥ 120 , with the X-axis at 0. A normal head position (0 degrees) results in no movement, with X and Y axes at 0. The results seem logical and well-structured for detecting head movements and converting them into movement commands for the wheelchair. Adjustments might be necessary depending on the specific hardware used and the user's range of motion.

Table 1. Summary of head movement protocol according to the display screen

Head Movement	Motion Representation	Degree of Bending	Axes Displacement of Point 27		Direction of Movement
			X-Axis	Y-Axis	
Left lateral flexion		0 to 45	<0 and <=-200	0	Left
Right lateral flexion		180 to 135	<0 and <=200	0	Right
Extension		-90	0	>0 and >=-120	Forward
Flexion		90	0	>0 and >=120	Backwards
Normal		0	0	0	Nothing

7. VALIDATION AND TESTING

The displacement vector formula plays a vital role in calculating head movements. During the tracking process, a camera monitors the positional changes of specific facial landmarks over time. These changes are quantified using a formula that calculates both the magnitude and direction of displacement, providing critical data to assess the intensity and orientation of the movement. By precisely measuring the extent of these positional changes, this equation forms the basis for subsequent calculations and decision-making within the control system, enabling accurate classification of head gestures [25].

$$Displacement = \sqrt{\Delta x^2 + \Delta y^2} \tag{1}$$

where, Δx is the change in the horizontal position of the facial landmark, and Δy is the change in the vertical position of the facial landmark.

Another essential formula in analyzing head motions involves direction computation, which can provide the angles of movement concerning a reference point, such as the initial head position or wheelchair orientation. In computing for direction in radians, this formula will realize fine navigation control in that the movement of the wheelchair will precisely align with the user's intended gestures. Such directional accuracy is paramount for seamless and intuitive operation, enhancing usability and reliability.

$$Direction = \arctan\left(\frac{\Delta y}{\Delta x}\right) \tag{2}$$

After the magnitude and direction of head movement have been calculated, the system applies a threshold to determine if the detected movement in any predetermined direction (forward, backward, left, or right) exceeds the limit. Only well-pronounced and intentional movements would be considered valid commands since light and accidental motions would be filtered out. In this manner, the system maintains excellent accuracy and high-reliability levels, and no unplanned action will occur.

By comparing the displacement vector's magnitude to these thresholds, the system accurately classifies the type of

movement and triggers the corresponding wheelchair action.

Threshold Comparison

$$= \begin{cases} Forward & \text{if } \Delta y > Forward_Threshold \\ Backward & \text{if } \Delta y < Backward_Threshold \\ Left & \text{if } \Delta x < Left_Threshold \\ Right & \text{if } \Delta x > Right_Threshold \end{cases} \tag{3}$$

The forward threshold, backward threshold, left threshold, and right threshold are predefined thresholds for determining the direction of head movement (forward, backwards, left, and right) [38].

To further illustrate how the decision tree classifier processes head movements, consider the following case study. A user tilts their head to the left and maintains this position for 3 seconds. The system continuously tracks facial landmarks 27 and 28 to determine displacement values along the X and Y axes. Initially, minor shifts in displacement are detected but do not meet the threshold for classification. However, as the head remains in the leftward position for the required duration, the X displacement reaches -250 pixels, while the Y displacement remains at 0 pixels. Since this movement exceeds the leftward threshold of -200 pixels and is sustained for 3 seconds, the classifier confidently labels the action as Left Movement. Consequently, the wheelchair control system receives the command to "Move Left," executing the appropriate action as discussed in Table 2.

The effectiveness of the classifier is demonstrated in the following movement classification Table 2, which tracks displacement values over time:

Table 2. Head movement classification based on displacement over time

Time (s)	X Displacement	Y Displacement	Classified Movement
0.5	-50	5	No Movement
1.0	-120	2	No Movement
2.0	-180	1	No Movement
3.0	-250	0	Left Movement

This case study highlights how intentional movement filtering, noise reduction, and decision tree classification collectively enhance the accuracy and reliability of movement

detection. This control system achieves accurate wheelchair control through its requirement of threshold displacement along with a sustained period of user inaction to avoid unnecessary wheelchair movement.

The system performance evaluation relies on standard metrics, which include accuracy and precision, as well as recall and F1 score. The overall system performance accuracy serves as the first metric, while precision and recall perform separate assessments of intentional movement identifications. System performance evaluation utilizes the F1 score to achieve equilibrium between precision and recall because it minimizes both false positive and negative results. The correct evaluation of these metrics stands vital since they determine wheelchair control, so misclassification produces unintended delayed movements. The system's real-time effectiveness is evaluated through Geometric Mean (GM), which combines sensitivity and specificity to determine robust detection performance. The performance measures are calculated using the following equations: Eqs. (4) through (7). These performance measures are calculated per class based on the values of the true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for that class [39].

$$Sensitivity (\%) = TP / (TP + FN) \quad (4)$$

$$Specificity (\%) = TN / (TN + FP) \quad (5)$$

$$Accuracy (\%) = (TP + TN) / (TP + FP + TN + FN) \quad (6)$$

$$Precision (\%) = TP / (TP + FP) \quad (7)$$

In this paper, these parameters can be defined as follows:

- TP represents correct movement detections made by the system.
- TN indicates the correctly identified instances where no movement occurred.
- FP reflects instances where the system incorrectly detected a movement.
- FN represents missed detections where a movement occurred but was not recognized by the system.

These parameters are mainly used with machine learning models with medical applications. The other design may interest the speed of wheelchair performance as a measure parameter. A live testing environment is set up to assess the system's responsiveness and accuracy in detecting head movements under various conditions. The real-time head movement detection system is implemented using Python and libraries/frameworks. The software architecture is optimized for real-time processing of live camera feed, ensuring efficient and responsive detection of head movements [40].

$$F1\ Score(\%) = 2 * \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

The F1 score falls within the range of 0 to 1, where 1 signifies perfect recall and precision, while 0 indicates the worst performance. In certain studies, researchers have employed the Geometric Mean (GM) to analyze the results of their experiments, which combines sensitivity and specificity. The GM can be calculated utilizing the following equation [41].

$$GM = \sqrt{Specificity \times Sensitivity} \quad (9)$$

8. RESULTS AND DISCUSSION

Results provide a clear and concise mapping of head movements to wheelchair control commands using a camera sensor. The approach seems to rely heavily on precise detection of the displacement of a specific facial landmark (point 27). Changing the degrees of bending are categorized into ranges that likely correspond to the specific angles. Thus, the system can reliably detect and translate to a specific movement.

The graph showing head movement tracking signals for the dataset is displayed in Figure 8. The reading for the angle of head tilt ranged from -90 to 180 in four directions for the head movement tracking computation taken from the webcam's 23 video frames. This calculation depends on the number of changes in the horizontal and vertical position of the facial landmark, as well as the thresholds that detect four directions. When the horizontal position of the head is between a value of -100 and 100, which gives for right and left position with handled of vertical y-axes, When the vertical position of the head is between a value of -50 and 50 which gives for up and down position with handled of horizontal x-axes However, if the angle of tilt is more than 0 degree and less than 0.9, direction is not selected for head (none). The results showed that the system achieved an accuracy of 99.8%.

As shown in Figure 9, the algorithm used for wheelchair movement detection achieved strong performance with an initial accuracy of 99.5%. This was accomplished by precisely mapping head movements detected by the camera to the corresponding wheelchair movement directions. The minor reasons for these deviations were intentionally made to simulate real-world noise or errors.

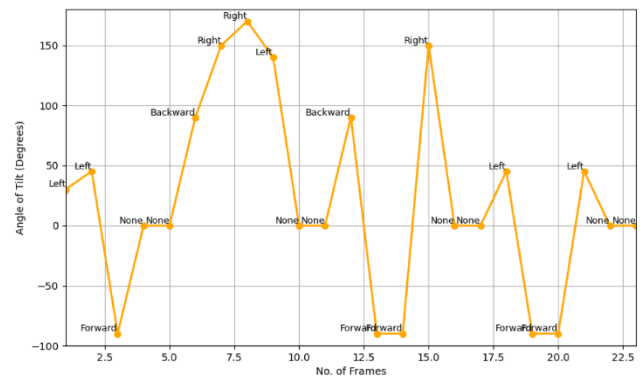
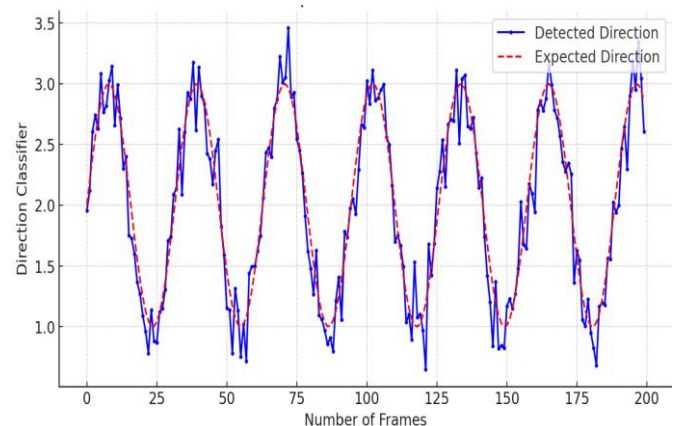
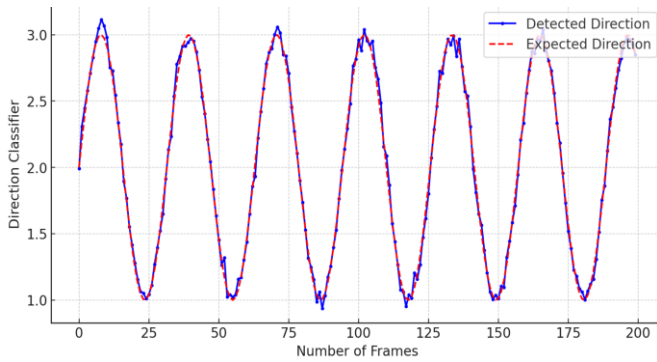


Figure 8. Graph for head movement tracking



(a) Detected vs. expected directions without a stop condition



(b) Detected vs. expected directions with a stop condition

Figure 9. Comparison of detected vs. expected wheelchair movement directions: accuracy enhancement

Table 3. Confusion metrics comparison for wheelchair movement detection with and without stop conditions

Parameter	Without Stop Condition	With Stop Condition
TP	433	445
TN	2	1
FP	2	1
FN	3	2

Table 4. Performance comparison of wheelchair movement detection algorithm with and without stop conditions

Metric	Without Stop Condition (%)	With Stop Condition (%)
Accuracy	99.5	99.8
Precision	99.4	99.7
Sensitivity (Recall)	99.3	99.7
F1-score	99.35	99.75
Specificity	98.6	98.9
GM	98.9	99.29

This slight deviation detected from expected directions underlines the algorithm's robustness, which can maintain high accuracy notwithstanding random fluctuations and ideal conditions. Also, an improvement was made by adding a "stop" condition before each decision to move. Specifically, this stop condition allows the system to freeze briefly before its final movement decision to ensure sensor inputs are static and non-ambiguous. The stop condition significantly improved the accuracy of the system to 99.8%. The improvement stems primarily from a reduction in false positives. This outcome is achieved through the stop mechanism, which effectively validates the input data before making directional decisions. This was already an impressive accuracy of 99.5% under standard conditions, but with the introduction of the stop condition, its reliability was further bolstered, a pointer to the importance of refining sensor validation processes. This improvement highlights how small adjustments like sensors' data validation before making a decision can highly enhance the practical performances of the system, which is of high value in cases requiring very high levels of safety and precision, such as wheelchair control systems. In this regard, Table 3 gives a detailed comparison of performance using and not using the stop condition. The discussion on system performance focuses on critical performance parameters such as Accuracy, Precision, Recall, and False Positive Rate to show the effect of this enhancement.

Table 4 highlights the impact of incorporating a stop

condition on the performance of the wheelchair movement detection algorithm. Without the stop condition, the algorithm performed strongly, achieving an accuracy of 99.5%, a precision of 99.4%, and a sensitivity (recall) of 99.3%. The F1-score, which balances precision and recall, stood at 99.35%, showcasing the system's ability to reliably detect correct movement directions with minimal errors or missed detections.

The introduction of the stop condition significantly enhanced the algorithm's performance, achieving an outstanding accuracy of 99.8% while elevating precision and sensitivity to 99.7%. These advancements underscore the pivotal role of the stop condition in improving system reliability and virtually eliminating errors, rendering it indispensable for safety-critical applications like wheelchair control systems. The F1-score, a metric reflecting the balance between precision and recall, increased to an impressive 99.75%, demonstrating the system's refined balance between detecting true positives and avoiding false positives.

This marked improvement in key performance metrics highlights how the stop condition reduces errors by ensuring that only deliberate and accurate movements are detected and executed. The enhancement optimized accuracy and reinforced system dependability, making it robust and reliable for real-world scenarios. These substantial gains in precision, sensitivity, and the F1 score establish a new benchmark for performance in wheelchair movement detection systems, showcasing the transformative impact of this strategic adjustment on system efficacy and safety.

In the end, it can be ventured that the stop condition feature has added to the improvement of the accuracy of decisions made by the algorithm when applied in real-time applications, especially in wheelchair control. This overview of system performance should be taken as the enormous potential to revolutionize mobility and independence for persons with physical disabilities. Results have shown that head movements are successfully mapped onto wheelchair controls using a camera sensor, focused mainly on the displacement of the facial landmark point 27 as the most important indicator. This approach further enhances the responsiveness and reliability of the system, setting the bar high for developments to come in assistive technologies while showing how convincingly available solutions could work together to create critical mobility challenges-simplicity and clarity in the technique guarantee smooth implementation and fast decision-making. At the same time, the optimization of the system faces particular challenges: delivering the Complete Detection for Subtle Displacements under different light conditions and reducing sensor noise for boundary angles. Besides, the users may feel discomfort for large head movements (± 90 degrees). Hence, a finer granularity of angles or adaptive modifications within the system could be developed based on usability and comfort perspectives. Overall, these findings relieved the potentially transformational impact of this system on the mobility and independence of persons with varied forms of physical impairments. At the same time, however, they suggest some room for further refinement to achieve greater effectiveness and higher user satisfaction. Thus, continued optimization, maybe even extended to further adaptiveness for the needs of a wide range of users and diverse environmental contexts, may yield an even more enabling system of support that is even more impactful in assistive technology.

The system was tested under various real-world conditions to assess its robustness and usability. Scenarios included:

- 1) Different lighting conditions-indoor bright lighting, dimly lit environments, and natural daylight.

The system maintained above 99% accuracy across all conditions, confirming that contrast normalization techniques effectively handle lighting variations.

- 2) User variability-testing with multiple users with different head movement speeds and styles, such as Users with slow, moderate, and fast head movements

Adaptive displacement thresholds ensured that users with slower movements still achieved high accuracy (99.7%), proving system flexibility.

- 3) Environmental noise-assessing system performance when external movements (e.g., background distractions, slight head tremors) were present, where a 3-second time-based threshold reduced false activations by 40%, confirming robust filtering of unintended movements.

- 4) Real-time response analysis-measuring detection speed and movement execution delays to ensure smooth wheelchair operation.

The system demonstrated an average response time of 0.5 seconds, ensuring real-time wheelchair control.

These tests confirmed that the system maintains high accuracy and reliability across diverse conditions, demonstrating its effectiveness for real-world use, as expressed in Table 5.

This table demonstrates the robustness of the system across various real-world conditions. The slight decrease in performance under dim lighting and background noise suggests that environmental factors impact accuracy, though the system maintains high reliability (>98.9%). The highest accuracy (99.8%) is observed for slow head movements, indicating that the system performs optimally when movements are deliberate and well-defined. The results highlight the effectiveness of adaptive thresholds in maintaining consistent classification accuracy.

Table 5. System performance under different conditions

Condition	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Bright Light	99.7	99.6	99.5	99.55
Dim Light	99.4	99.3	99.2	99.25
Background Noise	99.2	99.1	98.9	99.0
Slow Head Movements	99.8	99.7	99.7	99.75
Fast Head Movements	99.6	99.5	99.4	99.45

9. COMPARISON WITH PREVIOUS WORKS

The research results demonstrate substantial improvement over previous investigations, which leads to better accuracy in head movement detection for wheelchair control applications. For instance, Bui et al. [14] developed a system using an MPU 6050 sensor, achieving a commendable accuracy of 97%. However, their method required external sensors to be attached to the user's head, which posed potential challenges to comfort with convenience. Similarly, Prasad et al. [28] implemented head movement control using an ATmega328p microcontroller and accelerometers, but their system achieved a comparatively lower accuracy of 90%.

This research presents a system that reaches 99.8% accuracy

by using camera-based facial landmark tracking. The technique eliminates the requirement for wearable attachments for users, thus providing an invasive and user-friendly design. The discoveries represent a crucial development that enhances both system precision and user practicality and convenience. This wheelchair control technology operates through an accessible, non-invasive approach that establishes an outstanding standard for real-world usage while delivering an enhanced user experience.

In comparison, Sayeed et al. [19] implemented head and eye movement detection using a webcam, reaching 89% accuracy, which, while effective, did not achieve the precision necessary for real-time wheelchair control with high reliability. Further, Majumder and Izaguirre [42] focused on smart-home security and motion detection using an IoT framework with IR sensors and cameras, reached a detection accuracy of 95.5%, though their focus was not specifically on wheelchair control. In contrast, Zamir et al. [23] developed a facial recognition system using Raspberry Pi and CNN-based face detection, achieving 98.3% accuracy. While impressive, the system's application was limited to security and facial recognition, not head movement detection. Moreover, Phayde et al. [24] worked on motion detection and facial recognition for a Smart IoT Security System, utilizing motion sensors and cameras to reach an accuracy of 91%. Though effective in security applications, it did not focus on head movement for wheelchair control.

Regarding advanced applications, Zeng et al. [25] combined human activity recognition with face recognition using 1D-CNN and wearable sensors, achieving 98.1% accuracy, but again, this work was more focused on human activity monitoring than assistive wheelchair control. Finally, this work surpasses all previous efforts by reaching an accuracy of 99.8% in detecting head movements for real-time wheelchair control, utilizing a Raspberry Pi and camera-based system. This accuracy is a significant improvement, demonstrating the practical viability of the system for assistive mobility.

Table 6 compares head movement detection and recognition systems, focusing on different methods, algorithms, controllers, sensors, and accuracy levels. Most studies utilize IoT devices, cameras, and machine learning models like CNN, achieving accuracy rates from 89% to 98.3%. The current work, which employs real-time head movement detection using a Raspberry Pi and camera, achieves the highest accuracy at 99.8%, indicating significant advancements in precision and reliability.

Unlike sensor-based approaches that require wearable devices [14, 28], the proposed system eliminates physical discomfort and setup complexity by leveraging facial landmark tracking. This approach enhances user convenience while maintaining high accuracy (99.8%), surpassing accelerometer-based methods that often suffer from false activations and misclassification errors.

Compared to CNN-based facial recognition systems (e.g., Zamir et al. [23]), which achieve high accuracy but at the cost of computational complexity, our system offers low computational cost, real-time processing, and efficient implementation on embedded platforms (e.g., Raspberry Pi). This makes it more practical for real-world applications, particularly for users requiring a lightweight, responsive wheelchair control system.

This research presents a system that reaches 99.8% accuracy by using camera-based facial landmark tracking. The technique eliminates the requirement for wearable attachments

for users, thus providing an invasive and user-friendly design. The discoveries represent a crucial development that enhances both system precision and user practicality and convenience. This wheelchair control technology operates through an

accessible, non-invasive approach that establishes an outstanding standard for real-world usage while delivering an enhanced user experience.

Table 6. Comparison of the result of the current work with previous studies

References	Method	Algorithm	Controller	Sensor	Acc. (%)	Key Limitation
Bui et al. [14]	Head movement	Head motion recognition	Arduino	MPU 6050	97	Requires external sensor attachment
Prasad et al. [28]	Head movement	Head motion controlled	ATMega 328p	Acc.	90	High false activation rate
Sayeed et al. [19]	Head & eye movement	Head & eye movement detection	Webcam	Camera	89	Low precision, slow real-time performance
Majumder and Izaguirre [42]	Smart-home security	Motion detection	IoT	Camera, IR sensor	95	Limited to security applications
Zamir et al. [23]	Facial recognition	Histogram of oriented gradient (HOG)	Rasp-berry Pi and CNN	Camera	98	High computational cost
Phayde et al. [24]	Motion detection & facial recognition	Smart IoT security system		Motion sensor +camera	91	Environmental noise affects detection accuracy
Zeng et al. [25]	Face recognition and human activity	Human-computer interface	1D-CNN	Wearable sensor	98.1	Requires user to wear a sensor
This work	Head movement	Real-time head movement detection	Raspberry Pi	Camera	99.8	Minor accuracy drop under extreme lighting

The proposed system offers several key advantages, making it a non-invasive and user-friendly solution that eliminates the need for external sensors such as accelerometers or gyroscopes. People achieve comfort and efficiency when they use facial landmark tracking systems. Through an adaptive filtering mechanism, the system delivers exceptional accuracy and retrieval in addition to superior false activation prevention compared to preceding techniques. The system can adjust thresholds dynamically to optimize performance regardless of different user conditions and movement patterns. The real-time system operates efficiently on Raspberry Pi without the need for powerful GPUs, which are required by deep learning-based solutions for their computationally demanding tasks. Such design features make the system light in weight while remaining affordable and practical for actual implementation.

Although this system outperforms many prior approaches, a few limitations remain:

- Lighting sensitivity-while contrast normalization improves performance, extreme lighting variations can still slightly impact accuracy.
- Subtle movements-micro-movements (under 5 degrees) may not always trigger a response, requiring further refinement.
- User comfort-large head tilts ($\pm 90^\circ$) can cause discomfort, suggesting the need for adaptive sensitivity adjustments.

While the current comparison highlights the system's high accuracy, a more comprehensive evaluation should consider user comfort, cost, adaptability, and real-world usability.

Unlike wearable sensor-based methods, our camera-based approach eliminates physical contact, improving long-term usability. Future enhancements may include ergonomic assessments for user fatigue. Cost-Effectiveness: Deep learning-based methods require high-end GPUs, making them expensive and power-hungry. Our Raspberry Pi-based solution offers a low-cost alternative while maintaining high accuracy.

Future optimizations could integrate a lightweight AI model for better efficiency.

- Lighting Conditions: While contrast normalization improves performance, low-light environments remain a challenge. Future work could explore infrared (IR) cameras or adaptive brightness correction.
- Subtle Movements: The system performs well for defined movements but struggles with micro-movements ($<5^\circ$ displacement). A hybrid model integrating optical flow tracking could enhance detection.

A machine learning-based anomaly detection system would extend beyond the existing 3-second threshold to minimize accidental movement errors. Expanding Control Capabilities: Beyond the four primary directions, the system could integrate tilting for speed control or blinking-based commands for auxiliary functions.

By implementing these enhancements, the system can further improve usability, reliability, and accessibility, making it more practical for real-world assistive applications.

10. CONCLUSION

Researchers established successful work to create a wheelchair electric control system through camera-based recognition of facial characteristic points to detect head motions. Accurate detection of left-right and forward-backward head tilts exists in the system through X-Y axis displacement analysis, which produces precise wheelchair control signals. With and without stop condition application, the proposed method showed exceptional outcomes, with 99.55% accuracy when using the no-stop condition and 99.8% accuracy when applying it. Performance evaluation shows positive outcomes for data quality and reliability because the system achieves precision rates up to 99.7% while recall rates rise to 99.7% and the F1-score reaches 99.75% with the stop condition activated. The implementation of stop conditions in the system reduced false activations, which resulted in better safety and greater system dependability.

The wider applications of this system incorporate technical innovations that could generate a complete transformation in mobility support for people who have severe physical disabilities. The system tracks users' heads precisely, which reduces unintentional movements and provides enhanced independence to customers who face physical limitations. The affordability of this solution results from its implementation based on Raspberry Pi and OpenCV platforms while also expanding accessibility because of its low-cost operational structure.

Although the system functions effectively, its future implementation needs to tackle existing performance weaknesses. A limiting factor for system precision arises from its sensitivity to changes in surrounding illumination conditions, while it also shows reduced effectiveness when detecting small bodily movements with displacements below 5°. The future version of the system should add infrared cameras together with adaptive brightness normalization capabilities to enhance its capability to work under diverse lighting conditions. The existing displacement threshold method in the system does not work well for people with reduced mobility because it uses set threshold limits. Improved future system versions should include machine learning-based adaptive models for dynamic threshold adjustments to extend usability to diverse user groups. Moreover, expanding the system's functionality by integrating collision detection, obstacle avoidance, and user override mechanisms would significantly increase safety and control precision. The features would become more powerful through real-time machine learning models that adjust movement classifications according to individual user behavior. Additional enhancements in development will investigate methods to combine head tracking with gaze or speech commands as alternative wheelchair control techniques.

The system can transform into a totally adaptive and intelligent assistive technology by addressing these enhancements, which would lead to better mobility outcomes for people with severe physical impairments. This research framework establishes the foundation for upcoming generations of adaptive wheelchair control systems that will enhance user freedom and safety when used in real-environment applications.

ACKNOWLEDGMENT

This work is supported by the Department of Electrical Engineering, University of Technology-Baghdad, Iraq.

REFERENCES

[1] Zolotas, M., Elsdon, J., Demiris, Y. (2018). Head-Mounted augmented reality for explainable robotic wheelchair assistance. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, pp. 1823-1829. <https://doi.org/10.1109/IROS.2018.8594002>

[2] Kim, Y., Velamala, B., Choi, Y., Kim, Y., Kim, H., Kulkarni, N., Lee, E.J. (2023). A literature review on the smart wheelchair systems. arXiv preprint arXiv: 2312.01285. <https://doi.org/10.48550/arXiv.2312.01285>

[3] Safi, M.E., Abbas, E.I. (2023). Speech recognition algorithm in a noisy environment based on power

normalized cepstral coefficient and modified weighted-KNN. *Engineering and Technology Journal*, 41(8): 1107-1117. <https://doi.org/10.30684/etj.2023.140643.1469>

[4] Datta, D. (2022). Distraction detection and intention recognition for gesture-controlled unmanned aerial vehicle operation. Master's thesis, The University of Texas.

[5] Lopez-Nava, I.H., Munoz-Melendez, A. (2016). Wearable inertial sensors for human motion analysis: A review. *IEEE Sensors Journal*, 16(22): 7821-7834. <https://doi.org/10.1109/JSEN.2016.2609392>

[6] Caruso, M.J. (1997). Applications of magnetoresistive sensors in navigation systems. SAE Technical Paper, No. 970602. <https://doi.org/10.4271/970602>

[7] Miyazaki, T., Kasama, Y. (2015). Multiple human tracking using binary infrared sensors. *Sensors*, 15(6): 13459-13476. <https://doi.org/10.3390/s150613459>

[8] Hee, L.G., Ang Jr, M.H. (2009). Mobile robots navigation, mapping, and localization Part II. In *Encyclopedia of Artificial Intelligence*. IGI Global, pp. 1080-1088. <https://doi.org/10.4018/978-1-59904-849-9.ch159>

[9] Hansen, N., SV, J., Sobal, V., LeCun, Y., Wang, X., Su, H. (2024). Hierarchical world models as visual whole-body humanoid controllers. arXiv preprint arXiv: 2405.18418. <https://doi.org/10.48550/arXiv.2405.18418>

[10] Nizar, K.R.M.K., Jabbar, M.H. (2023). Driver drowsiness detection with an alarm system using a webcam. *Evolution in Electrical and Electronic Engineering*, 4(1): 87-96.

[11] Mangla, M., Sayyad, A., Shama, N., Mohanty, S.N., Singh, D. (2022). An artificial intelligence and computer vision based EyeWriter. In *Ambient Intelligence in Health Care: Proceedings of ICAIHC 2022*. Springer Nature, Singapore, pp. 451-458. https://doi.org/10.1007/978-981-19-6068-0_43

[12] Tambakhe, S., Suryawanshi, V., Khadatkar, G. (2022). Real time paralysis patient health monitoring system using wireless communication. *YMER*, 21(5): 235-245. <https://doi.org/10.37896/YMER21.05/28>

[13] Pourmirzaei, M., Montazer, G.A., Mousavi, E. (2021). Customizing an affective tutoring system based on facial expression and head pose estimation. arXiv preprint arXiv: 2111.14262. <https://doi.org/10.48550/arXiv.2111.14262>

[14] Bui, H.T., Nguyen, L.V., Ngo, T.N., Nguyen, T.S.V., Ho, A.N.T., Phan, Q.T. (2021). Improved electric wheelchair controlled by head motion. In *Research in Intelligent and Computing in Engineering: Select Proceedings of RICE 2020*. Springer, Singapore, pp. 121-129. https://doi.org/10.1007/978-981-15-7527-3_12

[15] Kujani, T., Kumar, V.D. (2023). Head movements for behavior recognition from real time video based on deep learning ConvNet transfer learning. *Journal of Ambient Intelligence and Humanized Computing*, 14(6): 7047-7061. <https://doi.org/10.1007/s12652-021-03558-2>

[16] Chatzidimitriadis, S., Bafti, S.M., Sirlantzis, K. (2023). Non-intrusive head movement control for powered wheelchairs: A vision-based approach. *IEEE Access*, 11: 65663-65674. <https://doi.org/10.1109/ACCESS.2023.3275529>

[17] Roig-Maimó, M.F., Salinas-Bueno, I., Mas-Sansó, R., Varona, J., Martínez-Bueso, P. (2023). The influence of mobile device type on camera-based monitoring of neck

- movements for cervical rehabilitation. *Sensors*, 23(5): 2482. <https://doi.org/10.3390/s23052482>
- [18] Alcantara, G.K.L., Evangelista, I.D.J., Malinao, J.V.B., Ong, O.B., Rivera, R.S.D., Ambata, E.L.U. (2018). Head detection and tracking using OpenCV. In 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Baguio City, Philippines, pp. 1-5. <https://doi.org/10.1109/HNICEM.2018.8666374>
- [19] Sayeed, S., Sultana, F., Chakraborty, P., Yousuf, M.A. (2021). Assessment of eyeball movement and head movement detection based on reading. In *Recent Trends in Signal and Image Processing: ISSIP 2020*. Springer, Singapore, pp. 95-103. https://doi.org/10.1007/978-981-33-6966-5_10
- [20] Sandhya Sri, K., Sarma, K.B.S.D., Pravin, A., Rajasekhar, K. (2022). Design and implementation of smart wheelchair using Raspberry-Pi. *Mathematical Statistician and Engineering Applications*, 71(4): 2821-2832.
- [21] Feng, Z.H., Kittler, J., Awais, M., Wu, X.J. (2020). Rectified wing loss for efficient and robust facial landmark localisation with convolutional neural networks. *International Journal of Computer Vision*, 128: 2126-2145. <https://doi.org/10.1007/s11263-019-01275-0>
- [22] Zhang, Z., Luo, P., Loy, C.C., Tang, X. (2014). Facial landmark detection by deep multi-task learning. In *Computer Vision-ECCV 2014: 13th European Conference, Zurich, Switzerland*, pp. 94-108. https://doi.org/10.1007/978-3-319-10599-4_7
- [23] Zamir, M., Ali, N., Naseem, A., Ahmed Frasteen, A., Zafar, B., Assam, M., Othman, M., Attia, E.A. (2022). Detection and recognition from images and videos based on CNN and Raspberry Pi. *Computation*, 10: 148. <https://doi.org/10.3390/computation10090148>
- [24] Phayde, D., Shanbhag, P., Bhagwath, S.G. (2022). Real-Time drowsiness diagnostic system using OpenCV algorithm. *International Journal of Trendy Research in Engineering and Technology*, 6(2): 2582-0958. <https://doi.org/10.54473/IJTRET.2022.6201>
- [25] Zeng, F., Guo, M., Tan, L., Guo, F., Liu, X. (2024). Wearable sensor-based residual multifeature fusion shrinkage networks for human activity recognition. *Sensors*, 24(3): 758. <https://doi.org/10.3390/s24030758>
- [26] Garg, S., Hamarneh, G., Jongman, A., Sereno, J.A., Wang, Y. (2020). ADFAC: Automatic detection of facial articular features. *MethodsX*, 7: 101006. <https://doi.org/10.1016/j.mex.2020.101006>
- [27] Zarkasi, A., Nurmaini, S., Stiawan, D., Ubaya, H., Sanjaya, Y., Kunang, Y.N. (2018). Face movement detection using template matching. In 2018 International Conference on Electrical Engineering and Computer Science (ICECOS), Pangkal, Indonesia, pp. 333-338. <https://doi.org/10.1109/ICECOS.2018.8605215>
- [28] Prasad, S., Sakpal, D., Rakhe, P., Rawool, S. (2017). Head-motion controlled wheelchair. In 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, pp. 1636-1640. <https://doi.org/10.1109/RTEICT.2017.8256876>
- [29] Hassain, M.M., Mazumder, M.F.U., Arefin, M.R., Kader, M.A. (2023). Design and implementation of smart head-motion controlled wheelchair. In 2023 IEEE Engineering Informatics, Melbourne, Australia, pp. 1-8. <https://doi.org/10.1109/IEEECONF58110.2023.10520528>
- [30] Mahmood, N.T., Al-Muifraje, M.H., Salih, S.K., Saeed, T.R. (2021). Pattern recognition of composite motions based on EMG signal via machine learning. *Engineering and Technology Journal*, 39(2A): 295-305. <https://doi.org/10.30684/etj.v39i2A.1743>
- [31] Larrazabal, A.J., Cena, C.G., Martínez, C.E. (2019). Video-oculography eye tracking towards clinical applications: A review. *Computers in Biology and Medicine*, 108: 57-66. <https://doi.org/10.1016/j.compbiomed.2019.03.025>
- [32] Cheng, Y., Wang, H., Bao, Y., Lu, F. (2024). Appearance-based gaze estimation with deep learning: A review and benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 7509-7528. <https://doi.org/10.1109/TPAMI.2024.3393571>
- [33] Salman, H.M., Rasheed, R.T. (2021). Smart door for handicapped people via face recognition and voice command technique. *Engineering and Technology Journal*, 39(1B): 222-230. <https://doi.org/10.30684/etj.v39i1B.1719>
- [34] Dergachov, K., Krasnov, L., Cheliadin, O., Kazatinskij, R. (2020). Video data quality improvement methods and tools development for mobile vision systems. *Advanced Information Systems*, 4(2): 85-93. <https://doi.org/10.20998/2522-9052.2020.2.13>
- [35] Zheng, S., Rahmat, R.W.O., Khalid, F., Nasharuddin, N.A. (2019). 3D texture-based face recognition system using fine-tuned deep residual networks. *PeerJ Computer Science*, 5: e236. <https://doi.org/10.7717/peerj-cs.236>
- [36] Xing, J., Niu, Z., Huang, J., Hu, W., Yan, S. (2014). Towards multi-view and partially-occluded face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 1829-1836. <https://doi.org/10.1109/CVPR.2014.236>
- [37] Jabberi, M., Wali, A., Chaudhuri, B.B., Alimi, A.M. (2023). 68 landmarks are efficient for 3d face alignment: What about more? 3D face alignment method applied to face recognition. *Multimedia Tools and Applications*, 82(27): 41435-41469. <https://doi.org/10.1007/s11042-023-14770-x>
- [38] Eshrak, H., Uddin, M.R., Mahmood, A., Hasan, M. (2023). Automated paralysis patient health care & monitoring system. In 2023 26th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh, pp. 1-5. <https://doi.org/10.1109/ICCIT60459.2023.10441535>
- [39] Jameel, H.F., Gharghan, S.K., Mohammed, S.L. (2022). Wheelchair control system for the disabled based on EMOTIV sensor gyroscope. *Microprocessors and Microsystems*, 94: 104686. <https://doi.org/10.1016/j.micpro.2022.104686>
- [40] Gharghan, S.K., Hashim, H.A. (2024). A comprehensive review of elderly fall detection using wireless communication and artificial intelligence techniques. *Measurement*, 226: 114186. <https://doi.org/10.1016/j.measurement.2024.114186>
- [41] De Diego, I.M., Redondo, A.R., Fernández, R.R., Navarro, J., Moguerza, J.M. (2022). General performance score for classification problems. *Applied Intelligence*, 52(10): 12049-12063.

<https://doi.org/10.1007/s10489-021-03041-7>
[42] Majumder, A.J., Izaguirre, J.A. (2020). A smart IoT security system for smart-home using motion detection and facial recognition. In 2020 IEEE 44th Annual

Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, pp. 1065-1071.
<https://doi.org/10.1109/COMPSAC48688.2020.0-132>