



An Intelligent Vision-Based Wheelchair Control System Using Facial Landmark Tracking for Individuals with Severe Motor Disabilities

Violeta Amjed Yousif^{1*}, Emad A. Mohammed¹, Akram M Zeki²

¹ Department of Computer Techniques Engineering, Technical Engineering College for Computer and Artificial Intelligence, Mosul, Northern Technical University, Mosul 41001, Iraq

² Kulliyah of Engineering, International Islamic University Malaysia, Kuala Lumpur 53100, Malaysia

Corresponding Author Email: violetta.amjad@ntu.edu.iq

Copyright: ©2026 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/jesa.590816>

ABSTRACT

Received: 3 June 2026

Revised: 10 August 2026

Accepted: 21 August 2026

Available online: 31 August 2026

Keywords:

intelligent wheelchair control, computer vision, facial landmark tracking, human-machine interface, MediaPipe Face Landmarker

Powered wheelchairs are an important mobility aid for individuals who cannot use traditional manual controls sufficiently; however, some users with severe upper-limb motor impairment may still find joystick-based control challenging. This constraint motivates the development of contactless control interfaces that convert voluntary facial actions into wheelchair commands. In this context, this paper presents an intelligent vision-based prototype wheelchair command interface for hands-free control using facial gestures. The system uses a Raspberry Pi 5 and MediaPipe Face Landmarker for real-time facial landmark tracking. It uses head movements to navigate, detects eye blinks for emergency STOP and power control, and recognizes mouth-opening gestures to control external wireless devices. In this paper, the X-Y joystick signals of the wheelchair platform used in this study are simulated using dual MCP4725 digital-to-analog converters (DACs), and HC-12 communication is used for auxiliary-device switching. The prototype was assessed in a controlled indoor environment with 20 able-bodied participants performing 600 gesture trials at an approximate fixed distance of 40–45 cm from the camera. The system obtained an overall gesture-classification accuracy of 96.5% with command-wise accuracies of 95–98%, as well as a macro-averaged precision, recall, and F1-score of 0.966, 0.965, and 0.965. The mean software command-response latency was 67.5 ms, and no packet loss was observed in 100 controlled indoor HC-12 transmissions up to 50 m. The results presented here offer evidence of the technical feasibility of the proposed multimodal command interface under the investigated conditions. Nevertheless, additional validation with the target user population and under a broader range of environmental and dynamic operating conditions is required before clinical or practical mobility deployment.

1. INTRODUCTION

Mobility assistance technologies have greatly enhanced the quality of life for individuals with severe motor disabilities by enabling greater independence in daily activities. Powered wheelchairs are one of the most effective technologies for restoring personal mobility [1]. However, traditional methods of wheelchair control, including hand-operated joysticks [2], chin controls [3], and sip-and-puff interfaces [4], often require residual motor abilities that many users with advanced neuromuscular disorders, high-level spinal cord injuries, or amyotrophic lateral sclerosis may not have [5]. Such users are thus faced with major difficulties in safely and efficiently operating wheelchairs, thus limiting their autonomy and social participation.

Recent advances in computer vision, embedded computing and artificial intelligence have created new opportunities to develop non-contact human-machine interfaces (HMIs) based on facial movements and gestures [6]. However, many of the existing vision-based wheelchair systems have limitations such as high computational complexity, reliance on

specialized sensors, limited compatibility with commercial wheelchair electronics, inadequate response times and lack of integrated environmental control capabilities [7]. In addition, some solutions rely on cloud processing or expensive hardware, limiting their applicability in real-world deployments [8].

The convergence of affordable embedded computing, high-resolution consumer cameras, and lightweight machine learning inference frameworks has opened new avenues for non-contact, vision-based HMIs in assistive technology [9]. Gesture-based and gaze-based control systems exploit natural motor repertoires head movements, eye blinks, and facial expressions that remain largely intact even in advanced stages of neuromuscular disease [10]. MediaPipe Face Landmarker provides an exceptionally efficient facial landmark detection pipeline capable of localizing 478 three-dimensional key points across the face at frame rates exceeding 30 fps on Raspberry Pi-class hardware [11], making it uniquely suited to embedded assistive applications [12].

The main contribution of this work is not in the development of a new facial-landmark algorithm or hardware

component, but rather in the system-level integration of complementary control functions into a unified vision-based wheelchair command interface. The proposed system integrates head orientation, normalized nose displacement, Eye Aspect Ratio (EAR)-based blink detection, and mouth-opening gestures into a single webcam-based MediaPipe pipeline for multimodal, contact-free control. It also uses two MCP4725 DACs to recreate the analog X-Y joystick signals native to the wheelchair controller used in this study without changing its internal motor-control electronics. In addition, wheelchair navigation and HC-12-based auxiliary device switching are embedded into a single offline architecture based on a Raspberry Pi 5 embedded system, which enables the control of mobility and environmental devices via the facial interface.

Recently, many research efforts have been made in the development of intelligent wheelchair control systems based on computer vision, embedded systems, HMIs, and assistive technologies [13]. Various techniques have been proposed to improve the mobility and independence of people with severe motor disabilities by means of facial gestures [14], eye movements [15], head tracking [16], brain-computer interfaces [17], and wireless communication technologies [18]. Although promising results have been reported in the literature, there are still many challenges such as limited real-time performance, high system complexity, dependence on specialized hardware, sensitivity to environmental conditions, and insufficient integration with existing wheelchair platforms [19].

In recent years, vision-based wheelchair control systems have attracted a lot of interest for hands-free mobility of severely motor-impaired people. Ritu et al. [20] developed a wheelchair controlled by facial gestures using computer vision and facial landmark detection. In the proposed system, users can control the wheelchair by facial movement. The proposed system demonstrated the feasibility of non-contact human-machine interaction for assistive mobility applications. The research showed positive gesture recognition and more user access. However, the system was susceptible to variations in lighting conditions and occlusions in the face that may affect the recognition accuracy in real-world scenarios. Moreover, the work was mainly focused on wheelchair navigation, without controlling any auxiliary device or compatibility with the commercial wheelchair electronics available.

Haddoun et al. [21] proposed a wheelchair system based on an MPU-6050 sensor and Bluetooth wireless communication controlled by head motion for people with severe motor disabilities. The system converted the head movements to navigation commands and obtained an accuracy of directional control of more than 90% by tilting the head 45°. The best operating speed was found to be 1.87 km/h at a PWM value of 180. Experiments showed that the wireless communication is stable and the real-time response is reliable when the wheelchair is navigating. The proposed solution was low-cost and effective as a mobility aid but required a cap fitted with wearable sensors and did not include non-contact interaction, computer-vision and environmental control functions. These restrictions show that more natural and flexible vision-based wheelchair control systems are needed.

Pienaar et al. [22] proposed a low-cost eye tracking algorithm to control an assistive wheelchair using MediaPipe facial landmarks. The gaze was computed in real-time and used to translate eye movements into commands to control a wheelchair. The system was a low-cost non-contact human-

machine interface for people with severe motor impairments. The results of the experiments indicated that the gaze tracking was stable under varying illumination conditions and could operate reliably in real-time at 30 fps. The proposed approach improves user comfort and accessibility by eliminating the need for wearable sensors and reducing the hardware complexity. However, the system had to be calibrated for each user, and it was mainly based on eye-gaze control, which could be affected by visual fatigue and long use. Moreover, the proposed study overlooked additional environmental control capabilities or sophisticated multimodal facial interaction, which opens a path for more comprehensive vision-based assistive mobility systems.

Abdullah and Mohammed [23] proposed an intelligent robotic hand control system based on the combination of gesture recognition using YOLO and landmark tracking using the MediaPipe platform on Raspberry Pi 4. They trained the system on a custom dataset of 6000 labeled images and showed impressive performance. The YOLOv11s model achieved 99.2%, 99.1%, 99.3%, and 99.1% for precision, recall, F1-score, and mAP@0.5 scores, respectively. Two robotic hands with 10 servos each were controlled by Arduino Mega. The study was not focused on wheelchair navigation, but rather on robotic hand actuation. However, successful implementation of MediaPipe, embedded processing and real-time control confirms the feasibility of using facial landmark tracking and single-board computers in assistive mobility applications. However, the proposed system did not incorporate wheelchair navigation, user safety mechanisms, and environmental control functions. These are still big challenges for assistive wheelchair systems.

Wameed and Alkamachi [24] proposed a computer vision-based robotic control system for real-time hand gesture recognition using the MediaPipe framework and Raspberry Pi 4. The system detected and classified hand gestures based on 21 3D hand landmarks and transmitted control commands wirelessly to a mobile robot. The experimental results showed 100% accuracy of gesture recognition under the best conditions (illuminance of 100 lux, the distance of 1 m, and tilt angles of 0°–30°) with an average response time of 0.133 s. However, the recognition accuracy fell dramatically under challenging conditions, to 12–15% at a tilt angle of 90° and 70% at longer distances of 4 m for some gestures. Results showed promising performance of MediaPipe for real-time embedded control applications but also showed that it is sensitive to environmental factors such as hand orientation, distance and illumination, which can challenge the system's robustness for real-world applications.

This paper presents a prototype intelligent vision-based wheelchair command interface for hands-free navigation using facial gestures. The proposed system uses a Raspberry Pi 5 and the MediaPipe Face Landmarker framework for real-time tracking of facial landmarks and gesture recognition. Head movements turn into directional wheelchair commands, and eye-blink and mouth-opening gestures provide emergency STOP, power control and wireless auxiliary-device functions. The proposed system combines computer vision, embedded processing, dual MCP4725-based joystick signal emulation, and HC-12 wireless communication in a single contactless architecture. The prototype was experimentally interfaced with the original joystick-control pathway of the wheelchair and tested under controlled indoor conditions. Therefore, the present work shows that the proposed multimodal command interface is feasible in the conditions evaluated.

2. VISION-BASED WHEELCHAIR CONTROL SYSTEMS

The proposed intelligent wheelchair control system is based on the integration of computer vision, facial gesture recognition, and embedded control technologies as shown in Figure 1 [25]. These technologies are the theoretical basis for the implementation of the real-time non-contact interaction between the user and the wheelchair [26]. To develop a reliable and efficient assistive mobility system, one must understand the principles of facial landmark detection, gesture interpretation and embedded communication [27]. In this section, the concepts used in the system, such as computer vision-based facial landmark tracking, facial gesture recognition and embedded control with wireless communication, are presented [28].

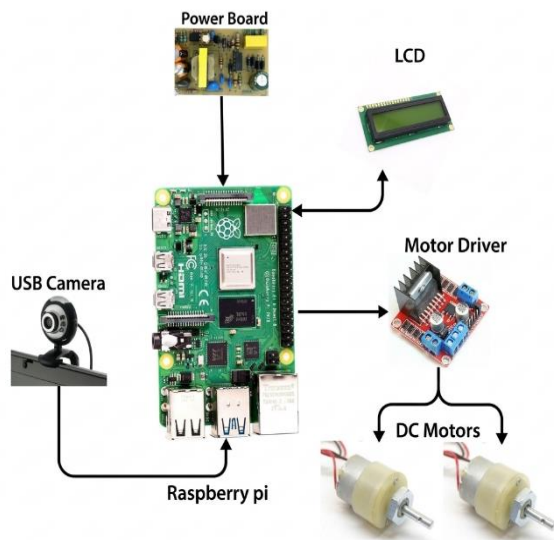


Figure 1. Smart wheelchair autonomous controlled system

2.1 Computer vision and facial landmark detection

Computer vision is an area of artificial intelligence that allows machines to interpret and understand visual information from cameras and image sensors. Computer vision is a non-contact human-machine interface for assistive mobility applications by extracting meaningful features from facial expressions, head movements, eye blinks and other visual cues [29]. Computer vision enables machines to understand visual information by extracting significant features from images and video streams [30]. In the proposed system, the MediaPipe Face Landmarker framework is used to detect and track 478 three-dimensional facial landmarks in real-time, which gives the geometric information required for facial gesture recognition and wheelchair control.

2.2 Facial gesture recognition and human-machine interaction

Facial gesture recognition is performed by analyzing facial landmarks to estimate head orientation, eye blinks and mouth movements. Head movements are translated into navigation commands, and the EAR algorithm is used for blink detection and emergency stop functions [31]. They use sustained mouth-opening gestures as an additional control channel for auxiliary device operation, allowing a natural and non-contact human-

machine interface [32].

2.3 Embedded control and wireless communication

The proposed system is based on Raspberry Pi 5 as the core processing unit for real-time facial gesture recognition and command generation. Two MCP4725 digital-to-analog converters (DACs) translate navigation commands into analog joystick-equivalent signals for interfacing with the wheelchair controller. The HC-12 transceiver is a wireless communication device, operating in the 433 MHz frequency band, which ensures stable serial communication over long distances with low power consumption [33]. The control commands are sent from the Raspberry Pi to the Arduino Nano receiver to control the external devices via the relay modules. The integrated architecture enables accurate wheelchair navigation and improves the system's ability to control environmental and smart home devices, while ensuring operational flexibility, reliability and user safety [34].

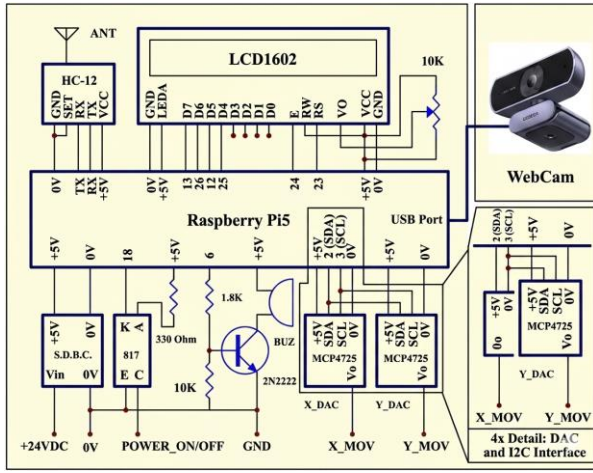
3. PROPOSED SYSTEM ARCHITECTURE

The proposed system is an intelligent assistive mobility system for non-contact wheelchair control through face gesture interaction. It integrates computer vision, embedded processing and wireless communication technologies in a single architecture that allows real-time navigation, emergency control and auxiliary device operation for people with severe motor disabilities. The overall architecture includes a number of hardware and software subsystems that are interconnected and work together to capture user inputs, interpret facial gestures, generate commands for control and safe system operation [35].

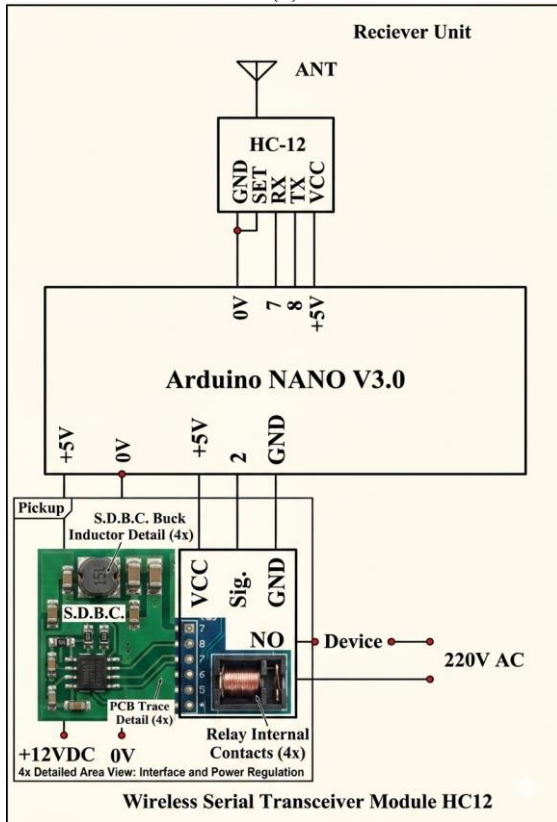
The operational workflow begins with real-time image acquisition by means of a UGREEN USB webcam connected to a Raspberry Pi 5. The video stream that is captured is processed with the MediaPipe Face Landmarker framework that extracts 478 three-dimensional facial landmarks at 30 fps. These landmarks are used to estimate head orientation and detect blink and mouth gestures, which are then used to control wheelchair navigation and auxiliary commands [36]. The generated commands are converted into analog signals that correspond to joystick inputs through MCP4725 DACs and then sent to the wheelchair controller [37].

3.1 Hardware and software architecture

The hardware architecture comprises a Raspberry Pi 5 as the central processing unit, a UGREEN USB webcam, two MCP4725 12-bit DACs, an HC-12 wireless transceiver, an Arduino Nano, an LCD1602 display, an active buzzer, and a PC817 optocoupler. The complete hardware configuration, electrical interconnections and the wireless receiver unit are shown in Figure 2. The Raspberry Pi 5 performs image acquisition, facial-gesture processing, and command generation. Two MCP4725 DACs emulate the analog X-Y joystick signals and interface with the original joystick-control pathway of the NOUR AL DALIL NR 6081-46 powered wheelchair [38]. The Arduino Nano-based receiver processes HC-12 wireless commands for auxiliary-device control. The GPIO and communication-bus assignments of the integrated peripherals are summarized in Table 1.



(a)



(b)

Figure 2. Hardware architecture of the proposed vision-based wheelchair control system: (a) main control and interface circuit; (b) wireless auxiliary-device receiver unit

Table 1. The peripheral interface configuration of the proposed vision-based wheelchair control system

GPIO/Address	Interface	Peripheral
0x60	I ² C	MCP4725 DAC (X-axis)
0x61	I ² C	MCP4725 DAC (Y-axis)
GPIO14, GPIO15	UART	HC-12 Transceiver
GPIO23–26	Parallel GPIO	LCD1602
GPIO18	Digital GPIO	PC817 Optocoupler
GPIO6	Digital GPIO	Active Buzzer

The system is functionally organized in three layers, including an input layer for facial image acquisition, a processing layer for facial landmark analysis and control decision generation, and an output layer for wheelchair and auxiliary device control. The MCP4725 DACs offer 12-bit resolution (4096 output levels) and communicate with the

Raspberry Pi via the I²C interface using GPIO2 (SDA) and GPIO3 (SCL). The HC-12 transceiver uses a UART interface through GPIO14 (T_{XD}) and GPIO15 (R_{XD}) for wireless serial communication. An active buzzer and LCD1602 in 4-bit parallel mode driven through a transistor-based GPIO interface provide user feedback. A PC817 optocoupler isolates the power control loop for the wheelchair electrically. The prototype is powered by the wheelchair's 24-V battery through a DC-DC buck converter that provides the regulated 5-V supply needed by the electronic subsystems. The configuration results in a compact, fully embedded prototype assistive-control platform [39].

The software architecture is implemented in Python as a real-time processing pipeline. Video frames acquired by the webcam are processed using MediaPipe Face Landmarker to extract 478 facial landmarks [40]. These landmarks are used to estimate head orientation, calculate the EAR for blink detection, and determine mouth-opening events. The validated facial actions are subsequently mapped to directional navigation, emergency STOP/power-control, and auxiliary-device commands according to the predefined decision rules. Navigation commands are converted into joystick-equivalent analog outputs via the dual MCP4725 DACs, whereas auxiliary commands are transmitted through the HC-12 wireless link when required. The resulting hardware–software integration was implemented with the existing joystick-control electronics of the wheelchair used in this work [41].

3.2 Communication and control interfaces

The system proposed here uses I²C and UART communication protocols. Table 1 summarizes the I²C interface used to communicate with the two MCP4725 DAC modules that generate the analog voltages corresponding to the joystick positions. The Raspberry Pi 5 is connected to the HC-12 wireless transceiver through the UART interface, allowing wireless communication to the Arduino Nano receiver [42]. The results of the gesture recognition are mapped onto predefined commands for the wheelchair like FORWARD, BACKWARD, LEFT, RIGHT and STOP. The corresponding DAC values and output voltages are shown in Table 2, where each gesture is mapped to a particular joystick-equivalent control signal. The system's architecture enables simultaneous navigation of the wheelchair and wireless control environment.

Table 2. Calibrated digital-to-analog converter (DAC) output levels for joystick-equivalent commands of the tested NOUR AL DALIL NR 6081-46 wheelchair

Gesture	X-DAC Value	Y-DAC Value	Output Voltage	Wheelchair Action
Centre	2048	2048	2.5/2.5 V	STOP
Forward	2048	573	2.5/0.7 V	FORWARD
Backward	2048	3686	2.5/4.5 V	BACKWARD
Left	573	2048	0.7/2.5 V	LEFT
Right	3686	2048	4.5/2.5 V	RIGHT

The proposed prototype is implemented on a NOUR AL DALIL NR 6081-46 powered wheelchair manufactured by Foshan Nour Aldalil Trading Limited, China and integrated with its original joystick-control pathway. Two MCP4725 DACs simulate the analog X–Y joystick signals with experimentally calibrated neutral and directional control levels of approximately 2.5 V and 0.7/4.5 V, respectively (Table 2),

without replacing the wheelchair's internal motor-control electronics. The compatibility demonstrated in this study is specific to the NR 6081-46 wheelchair/controller configuration and should not be generalized to other commercial wheelchairs, which may require model-specific electrical profiling, voltage calibration and safety validation prior to integration.

3.3 Safety and isolation mechanisms

The proposed prototype includes both hardware and software safety mechanisms. As depicted in Figure 2, the PC817 optocoupler is used for electrical isolation within the wheelchair power-control path. At the command-interface level, the fail-safe logic resets both DAC channels to the neutral 2.5-V state in the event of loss of valid facial tracking or activation of the emergency STOP command. The protection scheme described is specific to the prototype interface tested and is not a substitute for the manufacturer-specific electrical protection requirements for other wheelchair controllers [43].

This prevents voltage faults and ground loop problems. The emergency-stop and auxiliary-device functions are implemented by two independent control channels in an effort to avoid command conflict. The validated triple-blink sequence detected using the EAR algorithm is solely assigned to the wheelchair safety and power-control path. The triple-blink event immediately forces DAC outputs to neutral, clears any active directional latch, and puts the wheelchair in the STOP state as the wheelchair moves. When the wheelchair is already stopped, the validated blink sequence can be used to control the power of the wheelchair through the electrically isolated PC817 interface. In contrast, auxiliary-device control is only acquired by a sustained mouth-opening gesture of at least 2 s, generating a toggle command sent through the HC-12 wireless link to the Arduino Nano and relay unit. Thus, the mouth gesture changes the auxiliary device from OFF to ON or ON to OFF depending on its previous state and does not affect the wheelchair navigation or power. The safety-related blink processing is performed before the auxiliary-device branch, which prioritizes the emergency STOP function and prevents confusion between the two command paths [44].

4. SOFTWARE ALGORITHM AND CONTROL LOGIC

The proposed control framework is a real-time decision-making architecture that continuously processes facial landmark information in order to generate wheelchair navigation and auxiliary control commands [45]. The general sequence of operation of the system is shown in Figure 3. It summarizes the whole workflow from image acquisition to command execution. Firstly, the system performs initialization of hardware and communication, then it acquires the facial landmarks continuously and extracts features. Then, facial gestures such as head movements, eye blinks and mouth opening are analyzed and interpreted to determine the intended command of the user. The recognized commands are then converted to commands for wheelchair navigation or wireless control operations to be sent to the corresponding output interfaces. Visual and audio feedback mechanisms are also used in order to enhance the reliability of the system and the awareness of the user [46].

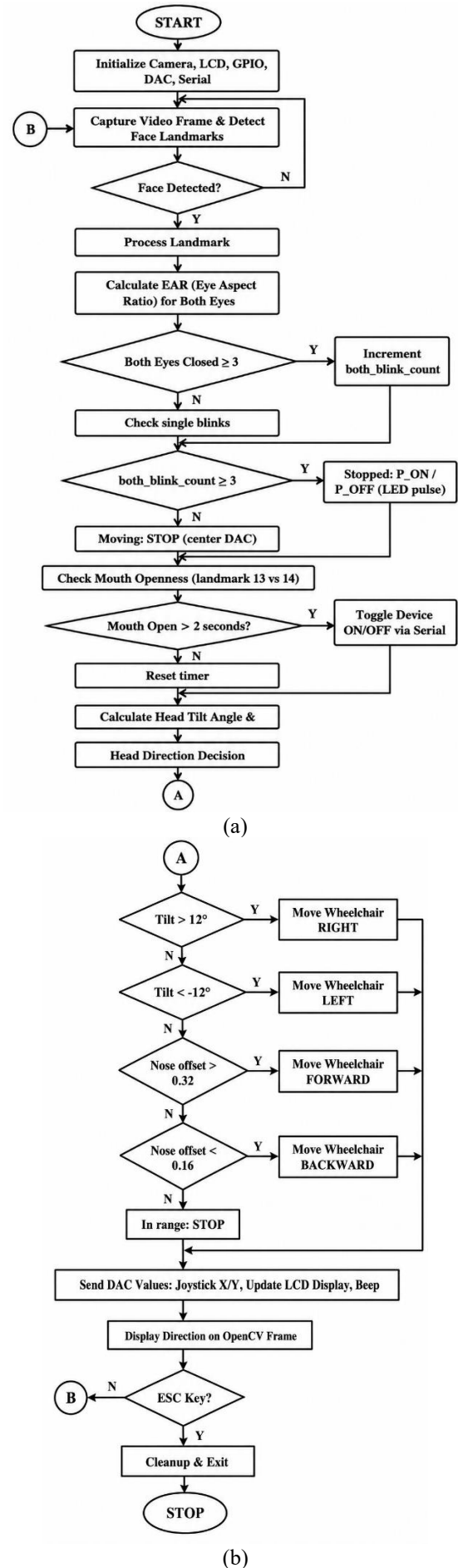


Figure 3. Flowchart diagram of the proposed vision-based wheelchair control system algorithm

4.1 System initialization

The control process starts with the initialization phase, during which all the sensing, communication and control subsystems are activated and configured to ensure reliable system operation. Figure 3 shows the flowchart of the embedded controller, where communication is established with the webcam, DAC, wireless transceiver, display module and feedback devices and the real-time operating loop begins. At this stage the system also initializes the facial landmark detection framework and the predefined decision thresholds for gesture interpretation. After all modules are verified and ready, the system enters the facial landmark acquisition stage, which allows for continuous monitoring of the user's facial movements and guarantees stable and safe operation during the control process.

4.2 Facial landmark acquisition

After the system initialization, the facial landmark acquisition phase continuously acquires the image frames from the webcam and processes them to detect the user's face in real-time. The detected face region is processed by the MediaPipe Face Landmarker framework, which extracts 478 three-dimensional facial landmarks corresponding to key facial features such as eyes, nose, mouth and facial contours. The landmark locations provide the geometric information for gesture analysis and decision-making later on. In case of a valid face, the extracted landmarks are passed to the gesture recognition module; if no valid face or reliable facial landmarks are detected, the control algorithm immediately enters the fail-safe STOP condition, returns the joystick-equivalent DAC outputs to their neutral values, and suspends the gesture-driven navigation until valid facial tracking is reestablished. No further face-loss timeout is applied beyond the first invalid processing cycle. This stage guarantees stable and consistent acquisition of facial information for accurate wheelchair control [47].

4.3 Gesture recognition and decision logic

In this section, the extracted facial landmarks are evaluated to identify the user's intentions. The MediaPipe framework extracts 478 three-dimensional facial landmarks at 30 fps for real-time gesture recognition. The geometric parameters defined in Eqs. (1) and (2) are used to estimate the head orientation [48]. The extracted facial landmarks are continuously evaluated to decide the intended user command. The inter-ocular tilt angle and normalized nose-eye offset are used to estimate head orientation, with thresholds of $\pm 12^\circ$ for LEFT/RIGHT steering and 0.16 and 0.32 for BACKWARD/FORWARD control, respectively. These thresholds were subsequently used (fixed) in testing, having been calculated prior to the formal 600-trial evaluation. Thresholds were not optimized using any observations from the reported evaluation dataset. The angular and normalized geometric features are applied in an effort to reduce the sensitivity to inter-subject differences in facial dimensions and camera-scale variations. The eye-blink activity is calculated using the EAR formulation in Eq. (3) with the threshold of 0.21 [49]. A confirmed three-blink sequence is a safety command and is given highest priority. It immediately issues a STOP command if the wheelchair is moving, returns both DAC channels to neutral output and cancels any active

FORWARD/BACKWARD latch. The blink channel is dedicated only to wheelchair stop/power control and is not used to operate the auxiliary device. An independent sustained mouth opening of at least 2 s generates a MOUTH-TOGGLE event, which is transmitted via the HC-12 interface. This event changes the auxiliary-device state from ON to OFF, or OFF to ON, as the case may be. Therefore, the emergency-stop function and the auxiliary-device functions are designed with different facial events, different output interfaces and explicitly separated decision branches, so as to avoid command conflict during operation.

$$\theta_{tilt} = \arctan2(y_R - y_L, x_R - x_L) [degree^\circ] \quad (1)$$

$$nose_offset = (y_{nose} - y_{eye_center}) / face_height \quad (2)$$

$$EAR = (\|p_2 - p_6\| + \|p_3 - p_5\|) / (2 \times \|p_1 - p_4\|) \quad (3)$$

The gesture recognition layer recognizes six control events: FORWARD, BACKWARD, LEFT, RIGHT, STOP and MOUTH-TOGGLE. These events shall be distinguished from the operational states that result from them in the system. The CENTER condition is not an independently classified facial gesture, but rather a neutral head-pose region in which neither the lateral nor the longitudinal decision threshold is crossed, and the wheelchair output remains at the joystick-center value. Also, Device ON and Device OFF are not two different facial expressions. A single MOUTH-TOGGLE event is produced by a sustained mouth-opening gesture of at least 2 s. If the auxiliary device is OFF, this event will turn it ON. If the auxiliary device is ON, the same event will turn it OFF. Thus, the auxiliary device is controlled with two-state toggle logic; wheelchair navigation is a separate function from the head-orientation and blink-based control events. This separation prevents ambiguity between the gesture classes and the physical states generated by the control algorithm.

Table 3 summarizes the facial landmarks, decision thresholds, temporal validation criteria, command priorities, and conflict-handling rules used for all navigation, safety, and auxiliary-control events, providing a reproducible specification of the implemented decision logic. The reported sign convention is in accordance with the image coordinates used by the processing pipeline. All conditions were evaluated using the fixed parameters defined before the formal evaluation. The implementation evaluated used a triple-blink detector based on event counts instead of an additional fixed global time window. For a blink to be accepted, both eyes had to be under the EAR threshold for at least 3 consecutive frames, and then opened again. Three such valid blink events triggered the STOP/power-control action, and the blink counter was reset.

4.4 Command execution and user feedback

After the gesture is recognized and classified, the corresponding control command is executed through the appropriate output interface. The navigation commands are translated into analog joystick-type signals and sent to the wheelchair controller [50]. The wheelchair can then move in the desired direction in real-time. Wireless commands are sent to the remote-control unit to turn ON/OFF external devices for auxiliary functions. Also, visual and auditory feedback mechanisms are used to inform the user about the current

system status and command execution results. This feedback loop increases user awareness, confirms successful command

delivery, and improves the overall safety and reliability of the proposed assistive mobility system [51].

Table 3. Facial-landmark parameters and control-decision rules

Command	Feature/Landmarks	Threshold/Rule	Temporal/Debounce	Priority/Action
LEFT	Head roll (33, 263)	$(\theta_{tilt} < -12^\circ)$	5 frames; $\pm 2^\circ$ hysteresis	Directional
RIGHT	Head roll (33, 263)	$(\theta_{tilt} > +12^\circ)$	5 frames; $\pm 2^\circ$ hysteresis	Directional
FORWARD	Normalized nose offset (1; 33, 362)	>0.32	5 frames; ± 0.02 hysteresis	Directional
BACKWARD	Normalized nose offset (1; 33, 362)	<0.16	5 frames; ± 0.02 hysteresis	Directional
CENTER	Head roll + nose offset	Within neutral ranges	Continuous	Neutral DAC output
STOP/Power	Both-eye EAR	$EAR < 0.21$	≥ 3 closed frames/blink; 3 blinks; ≤ 1.5 s inter-blink	Highest priority; STOP/neutral or power toggle
MOUTH-TOGGLE	Lip gap (13, 14)	>0.03	≥ 2 s; 3 s lockout	Auxiliary device ON $\leftrightarrow$ OFF
Face loss	Face-detection flag	No valid face	800 ms grace period	Fail-safe STOP; neutral DAC

5. EXPERIMENTAL RESULTS AND DISCUSSIONS

The performance of the proposed intelligent wheelchair control system was experimentally tested in a controlled indoor environment for real-time assistive mobility applications. For the experimental evaluation, the distance between the user's face and the camera system was maintained at about 40–45 cm for stable facial landmark detection and reliable gesture recognition performance. Experiments were performed in normal indoor illumination condition of about 500 lux and further observations under reduced lighting condition were made to investigate the robustness of facial landmark detection. The participants repeatedly performed predefined facial gestures for wheelchair navigation and auxiliary control commands. The experimental results provide a quantitative evaluation of the proposed prototype in terms of gesture-recognition performance, command-response latency, wireless command delivery, and wheelchair-interface operation under controlled indoor conditions.

5.1 Participant characteristics and experimental protocol

The gesture-recognition experiment involved 20 able-bodied participants without any motor disabilities. The participants were 20–49 years old: eight participants were 20–29 years old, seven were 30–39 years old, and five were 40–49 years old. The study group consisted of 11 males and nine females. Eighteen participants performed the experiments without glasses and two participants wore glasses during testing. Each participant performed five repetitions of six predefined control events such as FORWARD, BACKWARD, LEFT, RIGHT, STOP and MOUTH-TOGGLE resulting in 30 trials per participant and 600 evaluation trials in total. The present cohort of participants was used to evaluate the technical feasibility and command-interface performance of the prototype and should not be considered representative of the intended clinical population with severe motor disabilities. The experimental setup was kept at a face-to-camera distance of 40–45 cm under nominal indoor illumination of about 500 lux. Most importantly, the choice of the threshold was disentangled from the formal performance assessment. The decision thresholds were established prior to the formal 600-trial evaluation and remained fixed throughout the experiment. None of the trials

included in Tables 4–7 was used to optimize or change these thresholds. Final thresholds were $\pm 12^\circ$ lateral head tilt, 0.16 and 0.32 normalized nose–eye offset, 0.21 EAR-based blink detection, and 2 s sustained mouth opening. [In the evaluation, all 20 participants were provided with the same fixed thresholds and no subject-specific retuning was performed] To reduce inter-subject variation, we used geometric features that are intrinsically normalized or ratio-based such as the inter-ocular roll angle, the nose-eye displacement normalized by face height, and the EAR. Therefore, the reported classification results are actually evaluations using fixed decision criteria and not threshold fitting on the test observations.

5.2 Gesture classification accuracy

The experimental tests were carried out with 20 participants to evaluate the performance of the gesture classification. Each participant performed each gesture classes (Forward, BACKWARD, Left, Right, Stop, and Device Control) five times, resulting in 100 trials per gesture class. The classification results show that all commands are recognized with high recognition rate (95% to 98%). The stop command was the most accurate due to the specific pattern of blinks used to trigger it. The accuracies for the directional head-movement commands were slightly lower due to natural variations of the user's posture and the orientation of the face in operation. The overall average classification accuracy across all gesture classes is 96.5%, as shown in Table 4, which confirms the effectiveness of the MediaPipe facial landmark framework for reliable real-time wheelchair control. The confusion matrix of the gesture-recognition results is shown in Table 5, which indicates the distribution of correctly and incorrectly classified samples for each gesture class. Most of the classes achieved high true-positive rates and the few misclassifications were mainly between similar directional commands, which can be explained by natural variations in head orientation and user posture [52]. Overall, the matrix provides efficient discrimination for the navigation and auxiliary-control gestures with very low inter-class confusion. From a safety perspective, however, classification errors may have different operational consequences. Misclassifying an intended motion command as STOP or neutral is a conservative error that mainly affects usability, whereas interpreting STOP as a

directional command may cause unintended wheelchair motion and is potentially dangerous. Therefore, the confusion matrix in Table 5 should be evaluated not only by overall recognition accuracy but also by considering the direction and safety impact of each error. The implemented fail-safe logic further protects the system by returning the DAC outputs to the neutral state when valid facial landmarks are lost, while the triple-blink STOP command takes precedence over navigation commands. However, these results represent command-interface evaluations under controlled indoor conditions rather than full mobility-safety validation.

Table 4. Mapping of recognized facial gestures to digital-to-analog converters (DACs) output commands of the proposed control system

Gesture Class	Correct Classification	Total Trials	Accuracy (%)
FORWARD	98	100	98.0
BACKWARD	95	100	95.0
LEFT	97	100	97.0
RIGHT	96	100	96.0
STOP (Blink-Based)	97	100	97.0
MOUTH (Device Toggle)	96	100	96.0
Average (Overall)	579	600	96.5

The effect of lighting on gesture recognition performance by testing the same 20 participants and gesture set under three controlled indoor lighting conditions: normal illumination (approx. 500 lux), dim illumination (<200 lux) and bright/direct illumination (>1500 lux) was measured. Illuminance was measured with a lux meter directly in front of each participant at around face level, while the camera setup

was fixed and the face-to-camera distance was kept constant at 40–45 cm. Each gesture was performed 5 times by each participant in each illumination condition. The overall recognition accuracies summarized in Table 6 are to be 96.5% under normal illumination, 85.8% under dim illumination and 81.5% under bright/direct illumination. The accuracy of the STOP dropped from 97.0% in normal illumination to 80.0% in dim illumination and 75.0% in bright/direct illumination. Table 6 provides the corresponding 95% confidence intervals. The results indicate that facial-landmark recognition performs better under non-nominal illumination. However, the camera exposure and gain settings were not recorded separately. This is a limitation of the present controlled indoor evaluation. As summarized in Table 6, the overall recognition accuracies were 96.5%, 85.8%, and 81.5%, corresponding to error rates of 3.5%, 14.2%, and 18.5% under normal, dim, and bright/direct illumination, respectively. The corresponding 95% confidence intervals are also reported in Table 6. Consider other factors that may depend on the user and the environment. Only 2 of the 20 participants wore eyeglasses; thus, the present data are insufficient to demonstrate robustness across different lens types, frames, glare, or reflections. Fatigue due to prolonged use and limited head mobility were not evaluated separately and may reduce landmark visibility or gesture amplitude, especially for users who cannot exceed the fixed $\pm 12^\circ$ head-tilt threshold. Moreover, as the testing was conducted at a relatively fixed distance of 40–45 cm from the camera, the effects of camera displacement, viewing angle variation, and motion-induced vibration are not quantified. These factors require further evaluation with the intended user population in more variable indoor and outdoor operating conditions.

Table 5. Confusion matrix of gesture classification results

Actual\Predicted	Left	Right	Forward	Backward	Stop	Mouth-Toggle	Total
LEFT	97	0	2	1	0	0	100
RIGHT	0	96	3	1	0	0	100
FORWARD	0	0	98	0	2	0	100
BACKWARD	0	0	0	95	5	0	100
STOP	0	0	2	1	97	0	100
MOUTH-TOGGLE	0	0	0	4	0	96	100
Total	97	96	105	102	104	96	600

Table 6. Gesture-recognition accuracy under different illumination conditions with 95% confidence intervals

Gesture Class	Normal = 5 00 lux	Dim < 200 lux	Bright/Direct > 1500 lux
LEFT	97.0 (91.5–99.0)	88.0 (80.2–93.0)	83.0 (74.5–89.1)
RIGHT	96.0 (90.2–98.4)	87.0 (79.0–92.2)	82.0 (73.3–88.3)
FORWARD	98.0 (93.0–99.4)	89.0 (81.4–93.7)	85.0 (76.7–90.7)
BACKWARD	95.0 (88.8–97.8)	85.0 (76.7–90.7)	80.0 (71.1–86.7)
STOP	97.0 (91.5–99.0)	80.0 (71.1–86.7)	75.0 (65.7–82.5)
MOUTH-TOGGLE	96.0 (90.2–98.4)	86.0 (77.9–91.5)	84.0 (75.6–89.9)
Overall Accuracy	96.5 (94.7–97.7)	85.8 (82.8–88.4)	81.5 (78.2–84.4)
Error Rate	3.5%	14.2%	18.5%

Additionally, the precision, recall and F1-score have been computed through Eqs. (4)–(6) [53] for class-wise evaluation, where TP, FP and FN stand for true positives, false positives and false negatives, respectively. The LEFT, RIGHT and MOUTH classes achieved a precision of 1.000, indicating that they did not have any false-positive classifications, as shown in Table 7.

The precision values for the FORWARD and

BACKWARD commands were 0.933 and 0.931 respectively, mainly caused by small confusion between classes caused by natural variation of head orientation. The recall of all the gesture classes was from 0.950 to 0.980. The STOP class achieved a precision of 0.933, a recall of 0.970, and an F1-score of 0.951. The recalculated macro-averaged precision, recall, and F1-score were 0.966, 0.965, and 0.965, respectively, which affirm the balanced and reliable gesture-

classification performance across the entire 600-trial dataset.

Table 7. Precision, recall, and F1-score evaluation of gesture classification

Gesture	TP	FP	FN	Precision	Recall	F1-Score
LEFT	97	0	3	1.000	0.970	0.985
RIGHT	96	0	4	1.000	0.960	0.980
FORWARD	98	7	2	0.933	0.980	0.956
BACKWARD	95	7	5	0.931	0.950	0.941
STOP	97	7	3	0.933	0.970	0.951
MOUTH (Device Toggle)	96	0	4	1.000	0.960	0.980
Macro Avg.	—	—	—	0.966	0.965	0.965

$$\text{Precision} = \frac{T_p}{T_p + F_p} \quad (4)$$

$$\text{Recall} = \frac{T_p}{T_p + F_n} \quad (5)$$

$$F_1 - \text{score} = \frac{2T_p}{2T_p + F_n + F_p} \quad (6)$$

5.3 Software command latency

Python timestamping (`time.time()`) was used to measure the software command latency. A timestamp t_1 was taken at the time of the input-frame acquisition, and t_2 at the time of the completion of the corresponding MCP4725 DAC command write. The latency was computed as $t_2 - t_1$. The reported latency includes frame acquisition, MediaPipe facial-landmark inference, feature extraction, gesture-decision processing, and DAC command output [54]. Thus, the mean software latency of 67.5 ms is the total processing time until the DAC interface (not to be confused with the subsequent electromechanical response of the wheelchair). Vision processing and command output components were not independently timed. For temporally validated commands such as triple-blink STOP and MOUTH-TOGGLE, the gesture-validation duration requirement is considered separately from the post-validation software response. The average directional command latency was 67.5ms, summarized in Table 8 indicating the responsiveness of the software command generation under the tested conditions. These values should not be seen as the entire mechanical response time of the wheelchair.

5.4 Wireless command-delivery evaluation

The reliability of the wireless control subsystem was tested using the HC-12 transceiver pair operating at 433 MHz [55]. The wireless link of the HC-12 was tested in controlled indoor conditions of 10, 20, 30, 40, and 50 m. Each distance had 10 command transmissions, so a total of 100 transmissions were made. We have seen 100% packet delivery ratio (PDR) in the tested conditions; all commands were successfully received. For the 100 controlled indoor transmissions, there was no packet loss for the distances of 10-50 m. The observed PDR was 100% under the tested conditions, as summarized in Table 9. This result is specific to the current experimental setup and does not guarantee general wireless-link reliability. We do not report mean and maximum link latencies because the packet length, transmission rate, obstruction and interference

conditions, and wireless communication latency were not independently characterized.

Table 8. Software command latency to digital-to-analog converters (DACs) update

Command	Mean Latency (ms)	SD (ms)	Measurement Boundary
FORWARD	67.0	8.0	Frame acquisition → DAC write
BACKWARD	67.0	8.0	Frame acquisition → DAC write
LEFT	65.0	7.0	Frame acquisition → DAC write
RIGHT	65.0	7.0	Frame acquisition → DAC write
STOP (Triple Blink)	69.1	9.0	Validated event → DAC write
MOUTH-TOGGLE	69.0	9.0	Validated event → command output
Overall	67.5	7.5	Software response only

Table 9. Observed HC-12 command delivery under controlled indoor conditions

Distance (m)	Successful Transmissions	Total Transmissions	Observed PDR (%)
10	20	20	100
20	20	20	100
30	20	20	100
40	20	20	100
50	20	20	100
Overall	100	100	100

5.5 Prototype implementation and operational-state verification

The proposed system was physically integrated with the powered wheelchair to check the hardware compatibility, facial-command execution, emulation of joystick with DAC, emergency-stop operation and control of an auxiliary device under controlled indoor environments. The implementation described here is a prototype-level command-interface verification as opposed to a full route-based mobility-safety validation. The overall hardware assembly is shown in Figure 4, which includes the Raspberry Pi 5 controller, UGREEN webcam, two MCP4725 DAC modules, HC-12 wireless communication units, Arduino Nano receiver, LCD1602 display, buzzer feedback circuit and PC817 isolation interface. The prototype was successfully interfaced with the original joystick-control electronics of the wheelchair used in this study. This compatibility is platform specific and should not be generalized to other powered-wheelchair controllers without electrical characterization and recalibration. Figure 5 shows the practical operating conditions of the implementation system, but these conditions should not be considered as eight independent gesture classification classes. The classification experiment evaluates six control events: FORWARD, BACKWARD, LEFT, RIGHT, STOP and MOUTH-TOGGLE. The CENTER condition is the neutral navigation region and is generated when the head orientation is maintained within the pre-defined decision thresholds, so that it does not require a separate intentional gesture. The auxiliary subsystem is a two-state finite-state mechanism

which operates independently. A sustained mouth-opening gesture of at least 2 s causes a MOUTH-TOGGLE event that sets the auxiliary device from OFF state to ON state if its previous state was OFF, and from ON state to OFF state if its previous state was ON. Therefore, Device ON and Device OFF are not two separate gesture classes, but two output states of the same toggle command. The practical implementation confirmed the successful operation of the navigation, neutral, emergency-stop and auxiliary-device transitions. The quantitative classification results in Tables 4-6 are only for the six recognized control events.

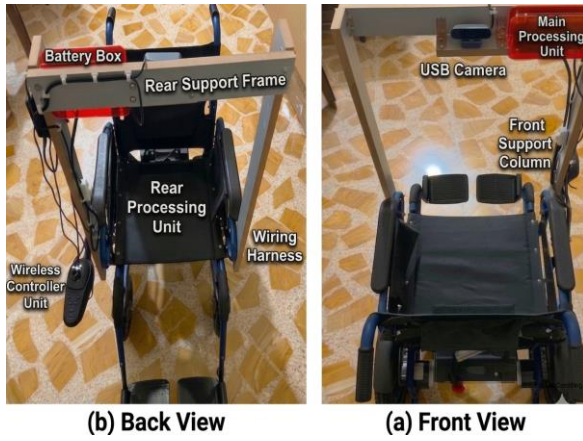


Figure 4. Practical implementation of the proposed vision-based wheelchair control system: (a) front view, (b) back view

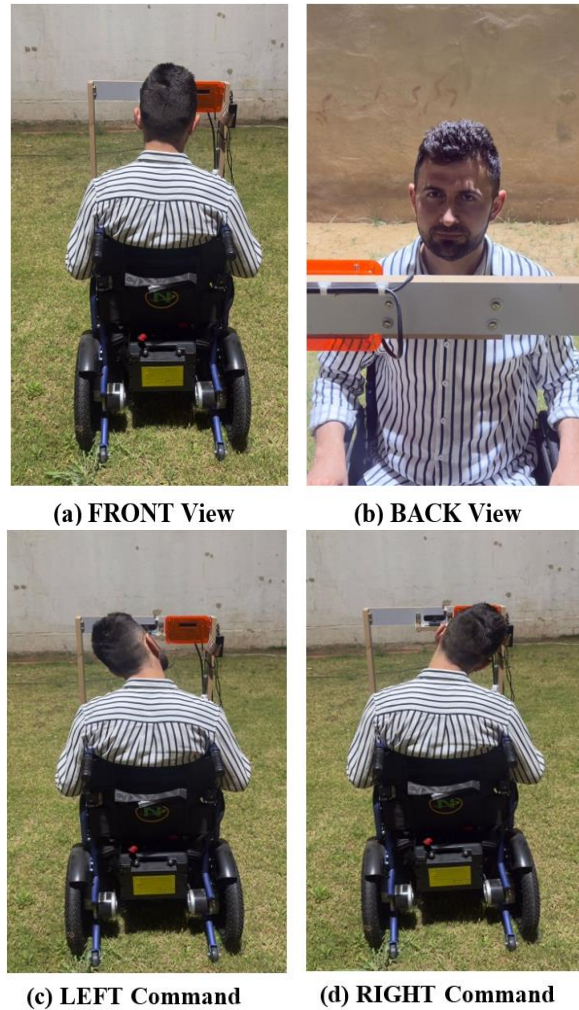


Figure 5. Practical operational conditions of the proposed facial-gesture control system: (a) prototype front/neutral view, (b) prototype rear/neutral view, (c) LEFT command, (d) RIGHT command, (e) FORWARD command, (f) BACKWARD command, (g) triple-blink STOP command, and (h) sustained mouth-opening gesture used to toggle the auxiliary device between ON and OFF states

The practical implementation confirmed successful execution of the defined operational states under the controlled experimental conditions. The experimental evaluation achieved an overall accuracy of 96.5% for gesture recognition and a mean software command latency of 67.5 ms from frame acquisition to completion of the DAC output update. In the wireless evaluation, 100 controlled indoor transmissions over distances of 10-50 m resulted in no packet loss and PDR of 100% under the tested conditions. Figures 4 and 5 show the gesture-recognition, command-generation, emergency STOP and auxiliary-device control functions of the integrated vision-based prototype in controlled indoor conditions.

The evaluation was performed on recognition accuracy, software command response, operational-state execution, and wireless reliability. Quantitative route-level measures not independently measured in the current experimental protocol are straight-path tracking, turn trajectory, stopping distance, route-completion success, and dynamic unintended-command rate. Likewise, the reported STOP latency is the software response to the validated STOP event and is not inclusive of the electromechanical stopping distance of the wheelchair. Thus, the present findings can be viewed more as a validation of the proposed prototype command interface, as opposed to full validation of mobility safety or clinical navigation.

As shown in Figure 6, the LCD1602 display offers real-time visual feedback on the physical operational states of the proposed system. The display continuously shows the

recognized wheelchair navigation command (FORWARD, BACKWARD, LEFT, RIGHT, STOP) and the status of the wireless auxiliary device control function (Device ON/OFF). The messages displayed are updated just after the gesture recognition and the command execution, so that the users can verify the correctness of the detected action before and during the wheelchair movement. The experimental results showed that the LCD interface was synchronized with the control algorithm and displayed all operating states correctly and without any noticeable delay. This visual feedback mechanism improves user confidence, enables monitoring of the system and improves overall safety.



Figure 6. Real-time LCD1602 feedback display for wheelchair navigation and auxiliary device control states

Table A1 summarizes a comparison of the proposed system with representative assistive wheelchair control approaches in

terms of input modality, wearable requirement, processing platform, control functions, reported performance, and safety or auxiliary features. Current systems utilize various modalities including eye gaze, head motion, hand gestures, and wearable sensing, with different performance and implementation requirements. Conversely, the proposed prototype combines contact-free head, blink and mouth input with joystick signal emulation, fail-safe stopping and wireless auxiliary-device control in a single embedded platform. While several previous systems report higher recognition accuracies for specific control modalities, the current work emphasizes multimodal functional integration with an overall accuracy of 96.5% and a mean software latency of 67.5 ms.

6. CONCLUSIONS

This paper developed and experimentally evaluated a prototype wheelchair command interface based on vision that integrates head orientation, eye-blink, and mouth-opening gestures using a Raspberry Pi 5 and MediaPipe Face Landmarker, with dual MCP4725 DACs for joystick-signal emulation and HC-12 communication for auxiliary-device control. 20 able-bodied subjects performed 600 gesture trials in controlled indoor conditions with a global recognition accuracy of 96.5%, individual command accuracies of 95–98%, and macro-averaged precision, recall and F1-score of 0.966, 0.965 and 0.965, respectively. The illumination evaluation resulted in overall accuracies of 96.5%, 85.8%, and 81.5% for normal (500 lux), dim (<200 lux), and bright/direct (>1500 lux) illumination respectively. This shows a decrease in recognition performance under non-nominal lighting. The average latency of the software commands was 67.5 ms between the acquisition of a frame and the completion of updating the DAC output. No packet loss was observed in 100 controlled indoor HC-12 transmissions over 10–50 m, resulting in an observed PDR of 100% under the tested conditions. The neutral fail-safe output and STOP-priority logic provide safe operation of the command interface. Compatibility with the NOUR AL DALIL NR 6081-46 platform was specifically tested for wheelchair use. Generally, the results support the technical feasibility of the proposed multimodal command interface under the evaluated controlled indoor conditions, while noting illumination sensitivity and the need for further validation. Future work will concentrate on evaluation with intended users with severe motor disabilities, long-term operation, larger illumination and camera-position ranges, eyeglasses and facial occlusion, motion-induced camera vibration, and dynamic wheelchair routes with turns, stopping and obstacles to evaluate real-world usability and mobility safety before wider practical deployment.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the Technical Engineering College/Mosul, Northern Technical University, Iraq, for providing the facilities, technical resources, and academic environment that supported the completion of this research. The authors also appreciate the valuable guidance and support received from the Department of Computer Techniques Engineering throughout the development and implementation of this work.

REFERENCES

- [1] Euldji, Bonanno, M., Saracino, B., Ciancarelli, I., et al. (2025). Assistive technologies for individuals with a disability from a neurological condition: A narrative review on the multimodal integration. *Healthcare*, 13(13): 1580. <https://doi.org/10.3390/healthcare13131580>
- [2] Ayalew, S.G., Gebeyehu, M.A., Kebede, G.A., Mhret, T.G. (2026). Design and development of assistive device: A dual-language voice-controlled smart wheelchair to enhanced mobility. *SSRN*, p. 6019479. <https://doi.org/10.2139/ssrn.6019479>
- [3] Fouad, A.I., Ismail, A.K., Ali, A.N., et al. (2025). Evolution of wheelchair technology: A comprehensive overview of history, disabilities, types and control mechanisms. *Advanced Sciences and Technology Journal*, 2(2): 1-20. <https://doi.org/10.21608/astj.2025.347587.1046>
- [4] Ukaegbu, U.F., Houshmand, S., Hammond, L., Adams, K., Andersen, J., Rouhani, H. (2026). Navigation paradigms for non-invasive BCI-controlled wheelchairs: A systematic review. *Progress in Biomedical Engineering*, 8(2): 022008. <https://doi.org/10.1088/2516-1091/ae5563>
- [5] Burke, K.M., Arulanandam, V., Scirocco, E., et al. (2025). Assistive technology in ALS. *American Journal of Physical Medicine & Rehabilitation*, 104(8): e115-e124. <https://doi.org/10.1097/phm.0000000000002742>
- [6] Therar, H.M., Mohammed, E.A., Ali, A.J. (2020). Biometric signature based public key security system. In 2020 International Conference on Advanced Science and Engineering (ICOASE), Duhok, Iraq, pp. 1-6. <https://doi.org/10.1109/icoase51841.2020.9436615>
- [7] Hu, Y., Wu, R., Li, G., Shen, Z., Xie, J. (2025). Facial features controlled smart vehicle for disabled/elderly people. *Electronics*, 14(6): 1088. <https://doi.org/10.3390/electronics14061088>
- [8] Atulan, G., Chowdhury, K.R., Nusrat, F., Schoen, M.P. (2025). Evolution and emerging trends in intelligent wheelchair control: A comprehensive review. *Machines*, 14(1): 33. <https://doi.org/10.3390/machines14010033>
- [9] Jalayer, R., Jalayer, M., Orsenigo, C., Tomizuka, M. (2026). A review on deep learning for vision-based hand detection, hand segmentation and hand gesture recognition in human-robot interaction. *Robotics and Computer-Integrated Manufacturing*, 97: 103110. <https://doi.org/10.1016/j.rcim.2025.103110>
- [10] Griffiths, T., Judge, S., Souto, D. (2024). Use of eye-gaze technology feedback by assistive technology professionals: Findings from a thematic analysis. *Disability and Rehabilitation: Assistive Technology*, 19(7): 2708-2725. <https://doi.org/10.1080/17483107.2024.2338125>
- [11] Alwali, M., Pambudi, S.F., Suciningtyas, L., Yuniarno, E.M. (2025). Markerless facial reconstruction motion capture using triangulation method. *Journal on Advanced Research in Electrical Engineering*, 9(2): 95-101. <https://doi.org/10.12962/jaree.v9i2.456>
- [12] Wickramasingha, K.B., Herath, H.M.K.K.M.B., Karunasena, G.M.K.B., Madhusanka, B.G.D.A. (2026). Smart assistive control of wheelchairs using unobtrusive eye movement recognition. In 2026 IEEE International Research Conference on Smart Computing and Systems Engineering (SCSE), Kelaniya, Gampaha, Sri Lanka, pp. 1-6. <https://doi.org/10.1109/scse70081.2026.11499827>
- [13] Palumbo, A., Ielpo, N., Calabrese, B., et al. (2024). An innovative device based on human-machine interface (HMI) for powered wheelchair control for neurodegenerative disease: A proof-of-concept. *Sensors*, 24(15): 4774. <https://doi.org/10.3390/s24154774>
- [14] Abdulkareem, H.J., Gharghan, S.K., Mutashar, S. (2025). Intelligent wheelchair control system using real-time head movement detection. *Mathematical Modelling of Engineering Problems*, 12(7): 2373-2386. <https://doi.org/10.18280/mmep.120717>
- [15] Divakar, R., Nithesh Aravind, C.M., Abhishek, T., Ram Bharath, S., Ganesan, M. (2026). Eye movement-based wheelchair control with emotion detector and obstacle detection. In *Communications in Computer and Information Science*, Raipur, India, pp. 222-240. https://doi.org/10.1007/978-3-032-14908-4_19
- [16] Tran, V.D., Nguyen, C.T. (2025). Design of a wheelchair control system based on hand gesture recognition using ResNet18. *Journal of Technical Education*. <https://doi.org/10.54644/jte.2025.1813>
- [17] Saichoo, T., Siribunyaphat, N., Sahoh, B., Efendi, M.A., Punsawad, Y. (2026). Electroencephalography-based brain-computer interface system using tongue movement imagery for wheelchair control. *Sensors*, 26(7): 2211. <https://doi.org/10.3390/s26072211>
- [18] Chakraborty, G.S., Bortty, J.C., Barman, J.M., Das, J., Polok, S.D. (2026). Design and implementation of a smart IoT-based wheelchair system: A novel approach to assistive technology for mobility and health monitoring. *Journal of Applied Research and Technology*, 24(2): 274-288. <https://doi.org/10.22201/icat.24486736e.2026.24.2.2909>
- [19] Ahmadi, A., Jahromi, M.N., Mostafavi, M.A., Morales, E., Sabo, N. (2026). Assistive navigation technologies for inclusive mobility: Identifying key environmental factors influencing wheelchair navigation through a scoping review. *ISPRS International Journal of Geo-Information*, 15(2): 75. <https://doi.org/10.3390/ijgi15020075>
- [20] Ritu, K.J., Mahmud, S.S., Bhuiyan, S., Ohidujjaman, Hasan, M. (2025). A facial gesture-controlled wheelchair for individuals with complete disabilities. In 2025 2nd International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM), Gazipur, Bangladesh, pp. 1-6. <https://doi.org/10.1109/NCIM65934.2025.11159844>
- [21] Haddoun, A., Djabri, D., Saidani, M., Benbouzd, M. (2025). Development and evaluation of a head-controlled wheelchair system for users with severe motor impairments. *MethodsX*, 15: 103485. <https://doi.org/10.1016/j.mex.2025.103485>
- [22] Pienaar, G., Smith, F., van Aardt, S., Hatefi, S. (2025). An economical eye-tracking algorithm for assistive wheelchair control using MediaPipe's facial landmarks. *MATEC Web of Conferences*, Pretoria, South Africa, 417: 10001. <https://doi.org/10.1051/mateconf/202541710001>
- [23] Yousif, K.S., Mohammed, E.A., Al-Hammoshi, M. (2026). Eye tracking and gaze estimation: A systematic review of deep learning methods and educational applications. *ITEGAM-JETIA*, 12(60): 1536-1547. <https://doi.org/10.5935/jetia.v12i60.4438>

- [24] Wameed, M., Alkamachi, A.M. (2023). Hand gestures robotic control based on computer vision. *International Journal of Intelligent Systems and Applications in Engineering*, 11(2): 1013-1021.
- [25] Masud, U., Abdulaziz Almolhis, N., Alhazmi, A., Ramakrishnan, J., Ul Islam, F., Razzaq Farooqi, A. (2024). Smart wheelchair controlled through a vision-based autonomous system. *IEEE Access*, 12: 65099-65116. <https://doi.org/10.1109/access.2024.3395656>
- [26] Bakouri, M., Alqarni, A., Alanazi, S., Alassaf, A., AlMohimeed, I., Alqahtani, T. (2025). Analysis of autonomous wheelchair navigation technologies in the past five years: A systematic review. *International Journal of Online and Biomedical Engineering*, 21(3): 56-83. <https://doi.org/10.3991/ijoe.v21i03.52269>
- [27] Neelima, K., Haricharan, S., Vamshi, B., Pranav, V., Abhinay, C. (2026). Design and development of a prototype smart wheelchair control system for enhanced accessibility and safety. *International Transactions on Electrical Engineering and Computer Science*, 5(1): 35-43. <https://doi.org/10.62760/iteecs.5.1.2026.173>
- [28] Guirguis, C.A., Ching, L.M., Tung, J.K. (2026). Computer vision and facial landmark detection in dermatology: A proof-of-concept study. *Experimental Dermatology*, 35(3): e70233. <https://doi.org/10.1111/exd.70233>
- [29] Matsuzaka, Y., Yashiro, R. (2023). AI-based computer vision techniques and expert systems. *AI*, 4(1): 289-302. <https://doi.org/10.3390/ai4010013>
- [30] Suma, K.G., Patil, P., Sunitha, G., Mantri, V.P., Kale, N.D. (2024). Computer vision and its intelligence in Industry 4.0. In *Advances in Computational Intelligence and Robotics*, pp. 119-142. <https://doi.org/10.4018/979-8-3693-5271-7.ch007>
- [31] Ryan, C., O'Sullivan, B., Elrasad, A., et al. (2021). Real-time face & eye tracking and blink detection using event cameras. *Neural Networks*, 141: 87-97. <https://doi.org/10.1016/j.neunet.2021.03.019>
- [32] Kielty, P., Ryan, C., Shariff, W., Lemley, J., Corcoran, P. (2025). Event-based multi-task facial landmark and blink detection. *IEEE Access*, 13: 45609-45622. <https://doi.org/10.1109/access.2025.3549910>
- [33] Hobbs, T., Ali, A. (2025). Smart home control using real-time hand gesture recognition and artificial intelligence on Raspberry Pi 5. *Electronics*, 14(20): 3976. <https://doi.org/10.3390/electronics14203976>
- [34] Tello, R.J.M.G., Bissoli, A.L.C., Ferrara, F., Müller, S., Ferreira, A., Bastos-Filho, T.F. (2015). Development of a human machine interface for control of robotic wheelchair and smart environment. *IFAC-PapersOnLine*, 48(19): 136-141. <https://doi.org/10.1016/j.ifacol.2015.12.023>
- [35] Karim, A.B., Haq, A.ul, Noor, A., Khan, B., Hussain, Z. (2022). Raspberry Pi based voice controlled smart wheelchair. In *2022 International Conference on Emerging Trends in Smart Technologies (ICETST)*, Karachi, Pakistan, pp. 1-5. <https://doi.org/10.1109/icetst55735.2022.9922929>
- [36] Alhajim, D., FAHAMA, H.S. (2024). Enhancing and securing a real-time embedded face recognition system using Raspberry Pi. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, 16(1): 92-104. <https://doi.org/10.29304/jqscsm.2024.16.11438>
- [37] Ahmed, G.T., Mohammed, E.A. (2026). Implementation of autonomous firefighting robot for real-time fire detection and suppression. *AIP Conference Proceedings*, 3393(1): 23003. <https://doi.org/10.1063/5.0320238>
- [38] Sadi, M.S., Alotaibi, M., Islam, Md.R., Islam, Md.S., Alhmiedat, T., Bassfar, Z. (2022). Finger-gesture controlled wheelchair with enabling IoT. *Sensors*, 22(22): 8716. <https://doi.org/10.3390/s22228716>
- [39] Weiss, A., Ben-Hanan, U., Salih, A. (2025). Dynamics mimicking—Enabling robot operation for wheelchair users. *IEEE Sensors Journal*, 25(11): 18901-18908. <https://doi.org/10.1109/jsen.2025.3559666>
- [40] Perez, E., Soria, C., Nasisi, O., Bastos, T.F., Mut, V. (2011). Robotic wheelchair controlled through a vision-based interface. *Robotica*, 30(5): 691-708. <https://doi.org/10.1017/s0263574711000919>
- [41] Teyssier, M., Koelle, M., Strohmeier, P., Fruchard, B., Steimle, J. (2021). Eyecam: Revealing relations between humans and sensing devices through an anthropomorphic webcam. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, Yokohama Japan, pp. 1-13. <https://doi.org/10.1145/3411764.3445491>
- [42] Tambakhe, S., Suryawanshi, V., Khadatkar, G., Belsare, K.S. (2022). Real-time paralysis patient health monitoring system using wireless communication. *YMER*, 21(5): 235-245.
- [43] Kuznetsov, O., Atzeni, G., Arnesano, M., Randieri, C., Smirnov, O. (2025). Secure IoT-based smart wheelchair system: From implementation to security enhancement strategy. In *Security and Privacy of Cyber-Physical Systems*, pp. 225-257. <https://doi.org/10.1201/9788743801306-8>
- [44] Islam, M.A., Hasib, A., Emon, T.M., Hasan, K.T. (2025). An AI-IoT based smart wheelchair with gesture-controlled mobility, deep learning-based obstacle detection, multi-sensor health monitoring, and emergency alert system. In *2025 IEEE 2nd International Conference on Computing, Applications and Systems (COMPAS)*, Kushtia, Bangladesh, pp. 1-6. <https://doi.org/10.1109/COMPAS67506.2025.11381815>
- [45] Fereidouni, S., Sheikh Hassani, M., Talebi, A., Rezaie, A.H. (2022). A novel design and implementation of wheelchair navigation system using Leap Motion sensor. *Disability and Rehabilitation: Assistive Technology*, 17(4): 442-448. <https://doi.org/10.1080/17483107.2020.1786734>
- [46] Balaji, S., Varsha Sree, R.S., Elma, K.J. (2025). Smart wheelchair based on gesture control with autonomous navigation. In *2025 7th International Conference on Inventive Material Science and Applications (ICIMA)*, Namakkal, India, pp. 803-811. <https://doi.org/10.1109/icima64861.2025.11074049>
- [47] Lee, Y., Jang, S., Bae, H.B., Jeon, T., Lee, S. (2024). Multitask learning strategy with pseudo-labeling: Face recognition, facial landmark detection, and head pose estimation. *Sensors*, 24(10): 3212. <https://doi.org/10.3390/s24103212>
- [48] Neto, E.N.A., Barreto, R.M., Duarte, R.M., et al. (2012). Real-time head pose estimation for mobile devices. In *Lecture Notes in Computer Science*, Natal, Brazil, pp. 467-474. https://doi.org/10.1007/978-3-642-32639-4_57
- [49] 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>
- [50] Gumus, C., Yenilmez, F., Ozlem, K., et al. (2026). A multidirectional textile interface for remote control using dynamic area-based capacitance modulation. *Advanced Robotics Research*, p. e202500169. <https://doi.org/10.1002/adrr.202500169>
- [51] Hou, L., Latif, J., Mehryar, P., et al. (2024). An autonomous wheelchair with health monitoring system based on Internet of Thing. *Scientific Reports*, 14(1): 5878. <https://doi.org/10.1038/s41598-024-56357-y>
- [52] Choi, J.H., Choi, H.T., Kim, K.T., Jung, J.S., Lee, S.H., Chang, W.D. (2025). Gesture classification using a smartwatch: Focusing on unseen non-target gestures. *Applied Sciences*, 15(9): 4867. <https://doi.org/10.3390/app15094867>
- [53] Muntean, M., Militaru, F.D. (2023). Metrics for evaluating classification algorithms. In *Education, Research and Business Technologies: Proceedings of 21st International Conference on Informatics in Economy (IE 2022)*, Bucharest, Romania, pp. 307-317. https://doi.org/10.1007/978-981-19-6755-9_24
- [54] Emad, A. (2024). Head gesture-controlled robotics wheelchair design and implementation. *International Journal of Artificial Intelligence and Emerging Technology*, 7(2): 18-27. <https://doi.org/10.21608/ijaiet.2025.346009.1010>
- [55] KP, D. (2020). Wireless transceiver module HC-12 based automatic water-level monitoring and control system. *IRJASH*, 2(10): 24-28. <https://doi.org/10.47392/irjash.2020.184>
- [56] Dahmani, M., Chowdhury, M.E., Khandakar, A., et al. (2020). An intelligent and low-cost eye-tracking system for motorized wheelchair control. *Sensors*, 20(14): 3936. <https://doi.org/10.3390/s20143936>
- [57] Somawirata, I.K., Utaminingrum, F. (2023). Smart wheelchair controlled by head gesture based on vision. *Journal of Physics: Conference Series*, 2497(1): 012011. <https://doi.org/10.1088/1742-6596/2497/1/012011>
- [58] Xu, J., Huang, Z., Liu, L., Li, X., Wei, K. (2023). Eye-gaze controlled wheelchair based on deep learning. *Sensors*, 23(13): 6239. <https://doi.org/10.3390/s23136239>
- [59] Higa, S., Yamada, K., Kamisato, S. (2023). Intelligent eye-controlled electric wheelchair based on estimating visual intentions using one-dimensional convolutional neural network and long short-term memory. *Sensors*, 23(8): 4028. <https://doi.org/10.3390/s23084028>

NOMENCLATURE

AI	Artificial Intelligence
DAC	Digital-to-Analog Converter
EAR	Eye Aspect Ratio
FPS	Frames Per Second
GPIO	General Purpose Input/Output
HC-12	433 MHz Wireless Communication Module
I ² C	Inter-Integrated Circuit
MCP4725	12-bit Digital-to-Analog Converter
UART	Universal Asynchronous Rx-Tx

Greek symbols

Δy	Vertical displacement between nose and eye-center landmarks, mm
θ	Head tilt angle used for lateral direction estimation, degree (°)
θ_{tilt}	Inter-ocular roll angle for left/right command detection

Subscripts

X	X-axis joystick control signal
Y	Y-axis joystick control signal
cmd	Generated control command

APPENDIX

Table A1. Comparative analysis of representative assistive wheelchair control systems

Ref.	Input Modality	Intended User/Applicat ion	Wearab le	Platform/Met hod	Control Functions	Main Reported Results	Safety/Auxiliary Function
[13]	Eye tracking + MR	Powered-wheelchair users	✓	HoloLens 2, MR-HMI	Wheelchair navigation	WST score: 84.3 ± 4.5; circuit time: 78.1 ± 8.5 s Accuracy: 99.8%;	HMD-based interface
[14]	Head movement	Mobility-impaired users	X	Facial landmarks + Raspberry Pi	Directional navigation + Stop	precision: 99.7%; recall: 99.7%; F1: 99.75% Training accuracy: 99.36%; validation accuracy: 100%	Stop-based control
[16]	Hand gestures	Assistive wheelchair control	X	ResNet18 + Jetson Nano	Six motion commands	Control accuracy/latency: NR	NR
[18]	IoT/multimodal sensing	Assistive mobility + monitoring	---	IoT-based embedded system	Mobility + health monitoring	24 studies reviewed;	Health monitoring
[19]	Environmental/navigation data	Accessible wheelchair	—	GIS + AI/ML review	Accessible route planning		Accessibility assessment

		navigation				sidewalks: 96%; ramps: 63%; curb cuts: 54%	
[20]	Facial gestures	Hands-free wheelchair control	✗	OpenCV + dlib + Arduino	F/B/L/R + speed control	Quantitative accuracy/latency: NR	No obstacle/wireless control
[21]	Head motion	Severe motor impairment	✓	MPU-6050 + Arduino	F/B/L/R/Stop	Directional accuracy: >90%; optimal safe speed: 1.87 km/h	Neutral/Stop state
[22]	Eye gaze + blink	Assistive wheelchair control	✗	MediaPipe + Raspberry Pi 4	Steering + Fwd/Stop	Accuracy: 96.5% (bright), 89.0% (dim), 98.71% (IR-dark)	Ultrasonic obstacle interlock
[56]	Eye gaze	Motorized-wheelchair users	✓	CNN + IR eye tracking	Fwd/Left/Right/Stop	Classification accuracy: 99.3%; processing: 99 FPS; prediction: ≈1.57 ms	Ultrasonic obstacle stop
[57]	Head gestures + vision	Hands-free smart-wheelchair users	✗	Camera + mini PC; Haar-Cascade face and nose detection	Brake, accelerate, turn right, turn left	Accuracy: 75–90% depending on gesture and camera distance (30–60 cm).	Smooth PWM-based acceleration/deceleration
[58]	Head gestures	Hands-free wheelchair control	✗	Camera + Haar Cascade	Motion control	Accuracy=75–90%	PWM-based smooth control
[59]	Head gestures	Hands-free wheelchair mobility	✗	Vision + Haar Cascade	Brake, accelerate, left/right	Accuracy detection =75–90%	Smooth motor-speed control
This work	Head + blink + mouth	Assistive wheelchair interface*	✗	MediaPipe + Raspberry Pi 5 + dual MCP4725	F/B/L/R/Stop + auxiliary control	Overall accuracy: 96.5%; mean software latency: 67.5 ms	Fail-safe Stop + HC-12 auxiliary control