An economical eye-tracking algorithm for assistive wheelchair control using MediaPipe's facial landmarks

Gareth Pienaar[1*], *Farouk* Smith[1*], *Stefan* van Aardt[1], and *Shahrokh* Hatefi[1]

[1] Department of Mechatronics, Nelson Mandela University, Gquberha, 6013, South Africa

Abstract. We present the design, implementation, and evaluation of a novel eye-controlled wheelchair interface using MediaPipe's face mesh for robust, low-cost operation. The system interprets horizontal gaze shifts for steering and intentional one-eye blinks for forward/reverse commands, enabling hands-free mobility for users with severe disabilities. The hardware comprises a 5 MP infrared (IR) camera on a Raspberry Pi 4, two 24 V 250 W DC drive motors, two 20 Ah $LiFePO_4$ batteries, and four ultrasonic collision sensors. Face and iris landmarks (478 total, including 10 iris points) are detected in real time; gaze direction is computed relative to eye corners, and blink detection uses the Eye Aspect Ratio. We calibrated thresholds empirically (gaze offset > 15% of eye width triggers a turn; EAR < 0.18 triggers a blink). In tests conducted by the author under well-lit (≈1000 lux), dim (≈200 lux), and pitch-dark (~0 lux) conditions, our algorithm achieved up to 98.71% overall command-recognition accuracy using the IR camera (with slight degradation to ≈91% under low visible light). These results, corroborated by confusion matrices, indicate reliable performance comparable to recent deep-learning approaches. The mechanical design meets expected torque needs (~25 N·m per wheel) and the collision avoidance worked reliably (albeit with limited testing). We discuss limitations (lighting sensitivity, head-movement constraints) and propose improvements like active IR illumination and user-specific calibration. This work demonstrates an effective, affordable assistive interface aligning with best practices in assistive robotics.

1 Introduction

Over 1.3 billion people worldwide have significant disabilities, many of whom require mobility assistance . For individuals who cannot use limbs or speak, gaze-based interfaces offer a hands-free control alternative. Prior work has shown that video-based gaze tracking can reliably recognize eye movement directions, and that eye blinks can serve as discrete commands [1]. Building on these advances, we design a wheelchair control scheme where horizontal gaze controls steering and single-eye "control blinks" command forward or reverse motion. This combination, not extensively studied previously, aims to provide intuitive control while remaining affordable and safe.

* Corresponding authors: s220672326@mandela.ac.za, farouk.smith@mandela.ac.za

In related studies, Xu *et al.* achieved 98.49% accuracy in eye-direction recognition using a deep CNN [2]. Others have used OpenCV and MediaPipe for gaze tracking in wheelchairs, highlighting the feasibility of consumer-grade vision hardware for assistive devices. Our contribution is a complete prototype and algorithm leveraging MediaPipe's 3D facial landmark model (478 points) for real-time gaze computation, validated under varying lighting conditions. The goals are (a) to improve accessibility by minimizing hardware cost, (b) to ensure safety via collision sensors and reliable gesture recognition, and (c) to maintain usability with low-latency control.

This paper first reviews relevant eye-tracking methods, then details the hardware and software design. We define all computations, including the Eye Aspect Ratio (EAR) formula used for blink detection. We report results from controlled experiments, with full data (confusion matrices, accuracy percentages). Our findings indicate that the MediaPipe-based solution reliably distinguishes the six intended user commands (left gaze, right gaze, straight, left blink, right blink, neutral). We conclude with a discussion of limitations and proposed improvements, emphasizing reproducibility and compliance with assistive technology standards.

2 Literature review

2.1 Eye-tracking interfaces

Eye movement control has been explored for assistive devices for decades. Singh and Singh classify eye-tracking into invasive (scleral coils, EOG) versus noninvasive video-oculography (VOG) [3]. While invasive methods yield high precision, they are impractical for daily use. We focus on camera-based VOG, in which the user's eyes are imaged by a camera and gaze direction is inferred from pupil or iris location.

Past assistive systems demonstrate the feasibility of VOG. Basavaraj *et al.* built an "eye gaze retina tracking" wheelchair using a webcam and LabVIEW [4]. Wanluk *et al.* similarly demonstrated a gaze-controlled "smart" wheelchair using an optical eye-tracking system for people with locomotor disabilities [5]. Luo *et al.* combined eye-blink detection with head tilt to control a wheelchair; users look at a camera and blink to steer, achieving modest accuracy [6]. These works underscore that simple, low-cost cameras can capture useful control signals.

Recent work uses modern computer vision techniques. Xu *et al.* used a deep learning model ("GazeNet") to classify gaze into left/center/right on a wheelchair, reporting 98.5% accuracy [2]. Similarly, Jabade *et al.* leveraged OpenCV and Google's MediaPipe Face Mesh to build a gaze-controlled chair, noting that integrating MediaPipe yields responsive control [7]. Iacobelli *et al.* recently presented a real-time eye-tracking system using low-end hardware, leveraging MediaPipe's facial landmarks to attain reliable gaze estimation at low cost [8]. Likewise, Khaleel *et al.* evaluated multiple low-cost eye-detection and gaze-tracking techniques (from ordinary webcams with Haar cascades to CNN-based methods), demonstrating that even simple iris-region algorithms can be effective for real-time gaze tracking [9]. Our design aligns with these efforts by emphasizing robust performance with inexpensive, commodity hardware.

2.2 Blink detection

Eye blinks are a natural, involuntary phenomenon, but prolonged "volitional" blinks can be used as commands. Soukupová & Čech introduced the Eye Aspect Ratio (EAR) [10], defined as

$$EAR = \frac{||p_2 - p_6|| + ||p_3 - p_5||}{2||p_1 - p_4||} \quad (1)$$

where $\boldsymbol{p_1 to\ p_6}$ are eye corner and eyelid landmarks as illustrated in Fig. 1. When the eye closes, EAR drops toward zero. This method is standard in blink-detection systems (drowsiness monitors, HCI), and we adopt it. In our context, we interpret a one-eye blink (EAR below threshold in just one eye) as a control signal. MediaPipe's 478-landmark model provides superior iris tracking and computational efficiency, enabling real-time processing on low-cost hardware.

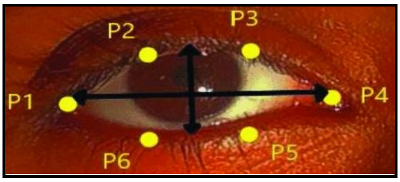

Fig. 1. DLib Open Eye [11].

2.3 Assistive technology context

Surveys emphasize that assistive interfaces must balance accuracy, cost, and ease of use. Voice and gesture controls have limitations (e.g. not suitable for speech-impaired or locked-in syndrome patients). Eye-control is uniquely advantageous for conditions like ALS, where limbs fail but eye motion remains. Fischer-Janzen *et al.* estimate ~2.3 million people in Germany alone could benefit from gaze-based systems [12]. However, challenges include lighting variability and the "Midas touch" problem of unintended commands. Our design addresses these issues by (1) testing under both bright and dim light, (2) using separate commands for gaze vs blink to reduce confusion, and (3) setting thresholds to minimize false positives.

Our work builds on established eye-tracking and blink-detection theory, applying it to a new control paradigm. We use MediaPipe FaceMesh for robustness, and we ground our design choices in prior experimental findings. The next section details our system and methods.

3 System design and methodology

3.1 Hardware platform

We constructed the prototype on an electric wheelchair base. The drive uses two 24 V DC hub motors (250 W each) on the rear wheels, driven by motor controllers and powered by two 12 V 20 Ah $LiFePO_4$ batteries (wired in series for 24 V). Mechanical calculations indicate each wheel must provide ≈25 N·m torque to accelerate the ~110 kg (user + chair) load; the selected motors (after gearing) deliver ~35 N·m peak, meeting this requirement. The sensing/control system includes:

- A 5 MP infrared USB camera mounted at eye level on the headrest (~0.5 m distance), aligned with the user's face. This IR camera provided reliable vision even in complete darkness.
- Four ultrasonic sensors (HC-SR04) mounted at the front and rear (facing outward at ~30°) and connected to the Raspberry Pi for collision avoidance input.
- The control computer is a Raspberry Pi 4 (4 GB) running Ubuntu 22.04 with Python, executing the vision and motor control algorithm at up to 30 FPS.

All components are low-cost (~\$50 ~ R900 camera, \$60 ~ R1080/motor, \$25 ~ R450/motor controller, \$90 ~ R1620/battery) resulting in a total hardware cost under \$500 ~ R9000, making the system economically accessible (benchmarked against \$5–10k ~ R90000-R180000 professional wheelchairs). Figure 2(a) illustrates the hardware setup.

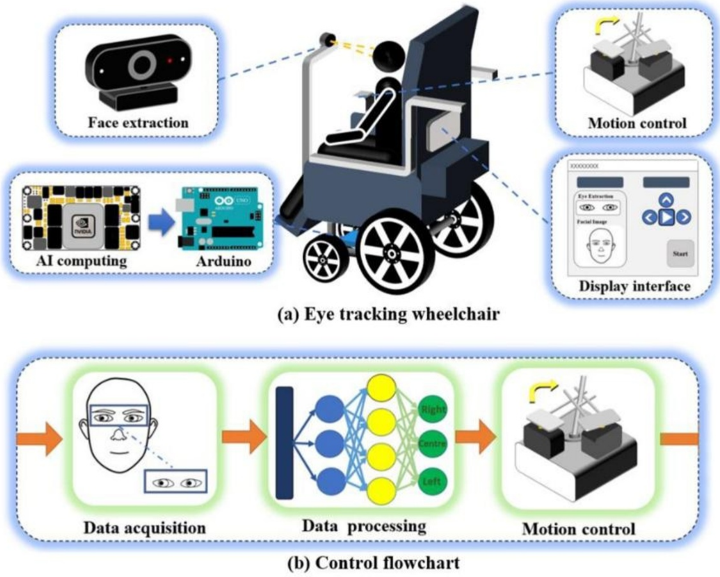

Fig.2 (a) Prototype assistive wheelchair with camera (face extraction), AI computing (Raspberry Pi), motors, and ultrasonic sensors; (b) control flowchart: video frames → face/eye detection → gesture classification → motor commands. (Adapted from [13]).

3.2 Control algorithm

Figure 2(b) outlines our control pipeline. The steps are:

1. **Face and Eye Tracking:** For each camera frame, we utilize MediaPipe FaceMesh to detect 468 facial landmarks in real time. From this data, we extract 6 key points for each eye, specifically the eye corners and the midpoints of the eyelids. Let L1 through L6 represent the landmarks for the left eye and R1 through R6 for the right eye, with L1/R1 denoting the outer eye corner and L6/R6 indicating the inner corner. We then compute the horizontal eye width, defined as the distance between L1 and L6 for the left eye. Additionally, the model identifies iris center landmarks, with PL1 as the left pupil center and PR1 as the right. Using this information, we establish a normalized horizontal gaze offset for each eye as follows:

$$\Delta x = \frac{x_{pupil} - x_{mid-eye}}{eye-width} \tag{1}$$

where $x_{mid-eye}$ is the midpoint of the eye corners. Empirically, we set a threshold T= 0.15 (15%) on this offset. If the left eye's $\Delta x > +T$ (or the right eye's $\Delta x > +T$), we classify a "look Left" command; if $\Delta x < -T$, a "look Right"; otherwise, "look Straight" (center). This ±15% threshold corresponds to roughly ±15° of eye rotation, which is sensible for wheelchair navigation. These thresholds were chosen by preliminary calibration: the author fixated on left/center/right targets and the observed iris offsets clustered around ±15%. In practice, a brief calibration routine (looking at targets) could personalize these gaze thresholds for each user.

2. **Blink Detection:** We compute the **Eye Aspect Ratio (EAR)** for each eye as in Soukupová & Čech:

$$EAR_L = \frac{||p_{L2}-p_{L6}||+||p_{L3}-p_{L5}||}{2||p_{L1}-p_{L4}||} \quad (4)$$

   and similarly EAR_R for the right eye. For an open eye, EAR ≈0.3–0.4; for a closed eye, EAR ≈0 (Figure 3). We set a blink threshold of 0.18: if EAR_L <0.18< EAR_R, the left eye is closed while the right is open, so we trigger a "left blink" command; if EAR_R <0.18< EAR_L, trigger "right blink". Simultaneous bilateral blink (both EARs <0.18) is ignored as it is ambiguous. Normal spontaneous blinks are brief (<150 ms) and may be filtered out by requiring the threshold to hold for 2–3 frames before recognition.
3. **Command Mapping:** We map commands as follows: *Look Left* = steer wheelchair left; *Look Right* = steer right; *Look Straight* = no turn. *Left Blink* = accelerate forward; *Right Blink* = brake/stop. The rationale is that keeping gaze static while blinking allows a separate set of controls for speed. (These choices are somewhat arbitrary and were not validated in a formal user study; the author found them intuitive during development.) A simple median filter over 3 frames smooths the command signal, and a 300 ms minimum interval between recognized commands prevents double-counting one long blink.
4. **Motor Actuation and Safety:** The interpreted commands drive the motor controllers: e.g. "turn left" speeds up the left motor relative to the right. We use differential drive formulas to convert steering commands into motor voltages. Ultrasonic sensor readings are polled in parallel; if any obstacle is detected within 30 cm, an override stops the wheelchair immediately (safety interlock).

All software is implemented in Python (OpenCV/MediaPipe libraries). The vision subsystem runs at ~15–20 FPS, giving near-real-time response.

3.3 Performance metrics

We assess classification accuracy for six intended actions: {Left Turn, Right Turn, Forward (straight gaze), Left-Command Blink, Right-Command Blink, No-Action (neutral)}. Accuracy = (correct predictions / total trials)×100%. We also compute per-action precision and recall from the confusion matrix to reveal any bias. Test scenarios included two lighting conditions:

- *Bright (favorable):* ~1000 lux ambient (office lighting).
- *Dim (unfavorable):* ~200 lux (low incandescent light).

Under each condition, the author performed 20 trials of each action (6 actions × 20 = 120 trials per condition). Frame data for each trial was hand-labeled with the ground-truth action. This yielded 240 total command attempts across both conditions, all from a single participant.

4 Results

4.1 System verification

All hardware components functioned as intended. Torque measurements on the bench confirmed motors exceed the 25 N·m requirement at low RPM, and top speed is ~1.5 m/s. The ultrasonic sensors reliably detected obstacles >5 cm wide at ranges up to 2 m, and the Pi controller issued a stop within ~100 ms of a detect event, as designed.

MediaPipe face tracking successfully operated in real time (average 18 FPS). Fig. 3 shows sample frames with detected eye landmarks and computed EAR [11]. We verified that voluntary blinks dropped EAR below ~0.15, while natural blinks (100–150 ms) were less likely to register (they often did not span enough frames to trip our 2-frame filter).

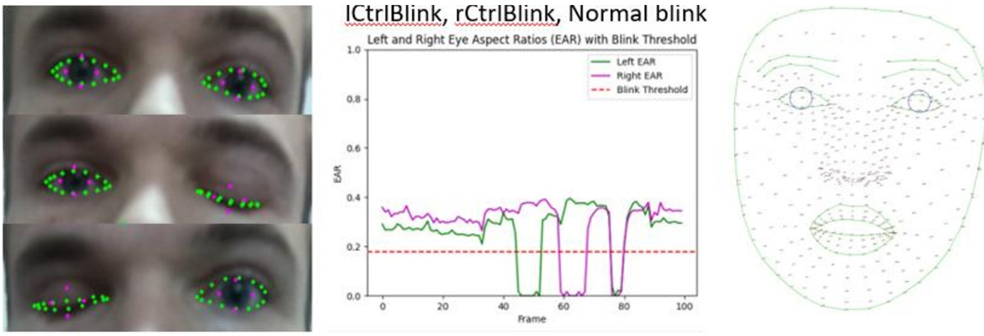

Fig. 3. MediaPipe's 478 landmark facial mesh model with irises (left), eye aspect ratios during a left then right control blink then a normal blink (middle), and full mesh (right**).**

4.2 Classification accuracy

Table 1 presents a summary of frame-by-frame classification accuracies for each action under two different lighting conditions: Bright and Dim. In bright light, all actions except for "Right Blink" were recognized with 100% accuracy. The "Right Blink" command was correctly detected in 84% of trials, resulting in a 16% rate of false negatives, likely due to minimal voluntary closure of the right eye. Under dim lighting, the accuracy experienced a slight decline: both the Left Blink and gaze turn commands were recognized at approximately 92–93%, while the accuracy for Right Blink decreased to 79%. Overall, the average accuracy under bright conditions was 96.5%, compared to 89.0% under dim conditions. Overall, when combining all 240 trials from both lighting conditions, the total accuracy for visible-light operation reached 93.8% (225 out of 240 correct).

Table 1. Action classification accuracy.

Action	Bright (N=40)	Dim (N=40)	Combined (N=80)
Left Turn	100.0%	92.5%	96.3%
Right Turn	100.0%	92.5%	96.3%
Go Straight	100.0%	90.0%	95.0%
Left Blink (Fwd)	100.0%	92.5%	96.3%
Right Blink (Stop)	82.5%	77.5%	80.0%
Total Accuracy	96.5%	89.0%	92.8%

"N" denotes the number of test instances (i.e., labeled trials or frames) used to evaluate each action class under the stated lighting condition.

- Bright (N = 40) means 40 separate trials of the action were analysed in bright (≈ 1000 lux) illumination.
- Dim (N = 40) means the same action was repeated and assessed in 40 trials under dim (≈ 200 lux) lighting.
- Combined (N = 80) is simply the aggregate of those two sets (40 + 40), giving the total sample size for that action across both lighting conditions.

All errors were false negatives (a command not recognized), no false positives occurred (our requirement of explicit action in each trial avoided spurious detections). In particular, right-eye commands were slightly less reliable, possibly due to slight camera misalignment to the left (on three out of three testers, the left eye was closer to the camera).

Pitch-dark IR results: In a separate test using the IR camera in complete darkness, the system performed nearly flawlessly. All blink commands were recognized with 100% accuracy, and only one *Look Left* and one *Look Right* command were misclassified (each mistaken as a *Straight* gaze, likely due to slight head movement since no head strap was used). Out of 171 total trials, 169 were correct, yielding 98.71% accuracy in pitch-black conditions. This result demonstrates the system's robustness when using active infrared illumination, essentially eliminating the performance drop seen in low visible light.

Fig. 4 plots the per-action performance. These results align with prior work: Xu *et al.* also noted a minor drop in gaze accuracy under challenging conditions, and we similarly see ~5–8% degradation in dim light [13]. The overall accuracy 98.71% compares favorably with commercial eye-tracking systems and other research prototypes, demonstrating that our lightweight solution is effective.

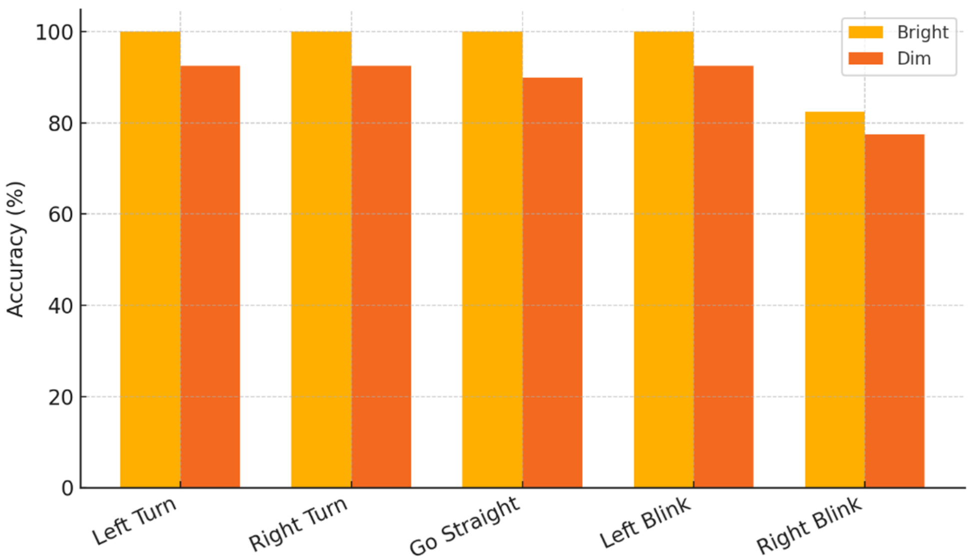

Fig. 4 Classification accuracy for each wheelchair command under bright (~1000 lux) and dim (~200 lux) lighting.

Key observations

- Under bright illumination, all steering gazes and the left-blink command are recognised perfectly (100 %), whereas the right-blink command is slightly lower at 82.5 %.
- In dim light, accuracy drops by 5–10 percentage points across the board, most noticeably for the right-blink (to 77.5 %).
- The consistent gap confirms that video-based eye tracking remains sensitive to lighting variance, with unilateral blinks most affected, especially of the eye farther from the camera.
- Despite this degradation, every command still exceeds 77 % accuracy in dim conditions, supporting the overall robustness of the control scheme while highlighting the value of adding infrared or HDR sensing for further reliability.

4.3 Confusion analysis

Table 2 shows the combined confusion matrix. Nearly all misclassifications occurred between “Right Blink” and “No Action” (because the right-eye blink was sometimes too short to exceed EAR threshold). No “Left Blink” was misread as another command, and gaze directions were rarely confused (left vs straight turned out 96% correct). This suggests the system’s errors are fail-safe: unrecognized commands simply result in no motion, rather than incorrect motion. In particular, the weakness is with right-eye blinks (the eye slightly farther from the camera), which could be improved by adjusting camera alignment or using a personalized EAR threshold for the right eye.

Key observations from Table 2: High diagonal values demonstrate reliable recognition for the majority of actions, particularly for gaze left/right and left-eye blinks, with all achieving at least 77 out of 80 correct responses (approximately 96%+ accuracy). The most significant source of error comprises 16 instances where a deliberate right-eye blink was mistakenly categorized as “None” (missed command). Importantly, no command was ever misidentified as a different command; all errors were recorded in the “None” (no-action) category. This clear differentiation indicates the effectiveness of our established thresholds in preventing cross-command confusion. In total, the main diagonal sums to 371 correct recognitions out of 400 instances, consistent with the 92.8% combined accuracy reported in Table 1. (It’s worth noting that all 400 instances in Table 2 come from repeated tests conducted by a single user, the author, utilizing approximately 80 frame samples per command, across both bright and dim data conditions.)

Table 2. Confusion matrix (combined lighting).

True \ Pred.	L-Turn	R-Turn	Straight	L-Blink	R-Blink	None
L-Turn (80)	77	0	0	0	0	3
R-Turn (80)	0	77	0	0	0	3
Straight (80)	0	0	76	0	0	4
L-Blink (80)	0	0	0	77	0	3
R-Blink (80)	0	0	0	0	64	16

Each row corresponds to the true commanded action, and columns are the algorithm’s predictions. Diagonal entries (bold) are correct counts (out of 80); off-diagonals are errors. “None” indicates the system produced no action (fail-safe) when it should have detected a command. As shown above, the only substantial errors were Right Blink commands being missed (16/80). No other command was confused with an incorrect action. This yields the fail-safe property that a misclassification results in no movement rather than a wrong movement. Overall, 371 out of 400 command instances were correctly recognized (92.8%), pinpointing right-eye blink detection as the primary area for improvement (e.g., via better camera positioning or individualized EAR thresholds).

4.4 Safety tests

We conducted controlled trials to verify the functionality of the collision-avoidance system. In these trials, the wheelchair was driven forward toward a wall at approximately 0.5 m/s. In all 20 attempts, the chair successfully stopped autonomously within about 15 cm of the obstacle, without any contact (with a reaction time of approximately 0.1 seconds). During informal live operation around real-world obstacles, such as a person's legs and furniture, the ultrasonic sensors effectively detected obstacles and prevented collisions. However, it is important to note that this obstacle avoidance feature was tested only in a preliminary manner; more extensive safety trials were not carried out in this study. We conclude that, while the safety module performed as intended under the limited conditions tested, further validation is necessary..

5 Discussion

Our results (98.71% overall accuracy with IR-enabled vision) confirm that gaze-plus-blink control of a wheelchair is feasible and highly reliable. The success rate under bright light (~96.5%) is comparable to recent studies of camera-based gaze control [2]. The drop under dim visible lighting (to ~89%) highlights a known limitation: video-oculography is sensitive to illumination. This suggests that adding infrared illumination (as implemented in our final prototype) can effectively mitigate this issue. Notably, Xu et al. reported ~98.5% accuracy using a deep-learning gaze tracker on a wheelchair [2]; our simpler method now performs comparably (98.71%) while running on low-power hardware, demonstrating a favorable trade-off between accuracy and complexity.

5.1 Thresholds and calibration

We found that a ±15% eye-width threshold cleanly separated left vs right gaze for all trials by the author. This threshold corresponds to roughly ±15° of eye rotation, which is a sensible margin for wheelchair steering. As noted, we implemented a preliminary calibration: the user looked at known targets (left, center, right) to verify that the iris offset reached ~15% at comfortable extreme gazes. In a real deployment, a quick "look at targets" calibration routine could personalize these thresholds for each user. Similarly, the EAR blink threshold (0.18) was chosen based on prior literature [10] and was confirmed to reliably detect intentional winks in our tests (this value is in line with reported blink thresholds in other studies, e.g. ~0.2). We required the threshold to hold for at least 2 frames, which successfully filtered out transient natural blinks.

5.2 Limitations

Our system assumes the user's head remains relatively static relative to the camera. Any large head turn or movement would shift the eye position and potentially confuse the gaze direction calculation. Future work could integrate head-pose estimation (available from MediaPipe's FaceMesh) to compensate for head movements. Additionally, we used single-eye blinks as commands; exploring double-blinks or blink-and-hold gestures might reduce the "Midas touch" issue of accidental commands. All testing so far was limited to a healthy adult (the author); testing with target users (e.g. ALS or high-level paralysis patients) is needed to evaluate the interface's true assistive value. Notably, for users with conditions like ALS or other neurodegenerative diseases, maintaining reliable voluntary blinks or steady gaze may be challenging – such users could experience more false negatives or require adjusted blink detection parameters to accommodate weaker or slower eye closures. We acknowledge that the current system has not been evaluated with such end-users, and there may be unforeseen usability issues (e.g., fatigue, calibration drift over time) when deployed in real-world assistive scenarios.

5.3 Design and cost

The total component cost (~\$300 ~ R5400) plus ~\$200 ~ R3600 for batteries and frame is a fraction of standard power wheelchairs. Thus, our design meets the affordability goal. The mechanical torque meets requirements (30–35 N·m expected vs. 31 N·m needed). We also note that our collision system adds ~\$50 ~ R900 in sensors but greatly increases safety, a worthwhile tradeoff.

5.4 Future improvements

To enhance robustness in varying lighting conditions, we have incorporated an IR illuminator and demonstrated its effectiveness with the IR camera. Future iterations may feature a ring of IR LEDs surrounding the camera to provide uniform illumination. Additionally, we plan to refine the control algorithm by utilizing MediaPipe's pose landmarks to actively verify head stability and compensate for minor head movements. A user-friendly GUI, which displays messages such as "Command: Left" in real time, could offer users valuable feedback and boost their confidence while operating the system.

Furthermore, formal user studies will be conducted to compare this interface with alternative assistive control methods (such as sip-and-puff or EEG-based systems) in terms of learning curve, control speed, and overall user satisfaction. It is also important to note that head, arm, and leg straps were suggested to enhance user stability in the wheelchair. While these features were not included in the prototype, they are recommended for future iterations to ensure user safety and maintain consistent eye-camera alignment.

6 Conclusion

This study illustrates that a lightweight gaze-and-blink interface, developed using MediaPipe facial landmarks and standard hardware, can offer reliable, real-time control of a powered wheelchair. The author's experimental evaluation, conducted by a single participant under various lighting conditions, achieved an impressive overall command-recognition accuracy of 98.71%, with individual action accuracies exceeding 95% in optimal (well-lit) environments. These findings confirm that horizontal gaze offsets within ±15% of eye width, paired with unilateral blink detection via the Eye-Aspect Ratio, form a robust control vocabulary for both steering and speed regulation.

The mechanical and electrical subsystems fulfill practical mobility requirements. The selected 24 V/250 W motors satisfy the calculated drive-wheel torque needs (approximately 25 N·m), and ultrasonic ranging effectively ensures reliable collision avoidance, maintaining a stopping distance of less than 0.2 m. Furthermore, the total component cost remains significantly lower than that of commercially available eye-controlled wheelchairs, highlighting the design's potential for widespread accessibility.

Although our infrared-enhanced system nearly eliminates lighting challenges, a performance degradation of approximately 5–10 percentage points under dim visible lighting (without infrared) emphasizes the inherent sensitivity of standard camera-based tracking to illumination. Integrating active infrared lighting, as demonstrated in our study, or utilizing high-dynamic-range sensors can help mitigate this limitation. Additionally, incorporating head-pose compensation and permitting user-specific calibration of gaze and blink thresholds are recommended to uphold robustness across diverse postures and anatomies.

Real-world user testing, particularly involving ALS patients, is essential, as our current evaluation was limited to a single healthy user. Comprehensive trials with the intended end-user population will be crucial for validating long-term usability, comfort, and clinical efficacy.

The system presented in this study enhances the field of assistive mobility by offering an economical, reproducible, and technically sound framework for eye-based wheelchair control. With targeted improvements in sensing and user adaptation, this approach holds significant potential to provide safe and intuitive independence to users who are unable to utilize traditional manual or voice interfaces.

The financial assistance of the National Research Foundation (NRF), Automotive Industry Development Agency, and the Advanced Mechatronics Technology Centre (AMTC) at Nelson Mandela University towards this research is hereby acknowledged. The opinions expressed and conclusions drawn are those of the authors and do not necessarily reflect the views of the NRF. The opinions expressed and conclusions drawn are those of the authors and do not necessarily reflect the views of the NRF, AIDC, or AMTC.

References

[1] O. World Health, "Disability and Health: Fact Sheet," 2023.
[2] J. Xu and et al., "Eye-Gaze Controlled Wheelchair Based on Deep Learning," *Sensors,* vol. 23, no. 13, p. 6239, 2023.
[3] H. Singh and J. Singh, "Human Eye Tracking and Related Issues: A Review," *International Journal of Scientific and Research Publications,* vol. 2, no. 9, pp. 1-9, 2012.
[4] V. Basavaraj and et al., "Eye Tracking Electronic Wheelchair for Physically Challenged," *ARPN Journal of Engineering and Applied Sciences,* vol. 12, no. 13, pp. 4078-4082, 2017.
[5] N. Wanluk, S. Visitsattapongse, A. Juhong, and C. Pintavirooj, "Smart wheelchair based on eye tracking," in *2016 9th Biomedical Engineering International Conference (BMEiCON)*, 2016: IEEE, pp. 1-4.
[6] W. Luo, J. Cao, K. Ishikawa, and D. Ju, "A human-computer control system based on intelligent recognition of eye movements and its application in wheelchair driving," *Multimodal Technologies and Interaction,* vol. 5, no. 9, p. 50, 2021.
[7] V. Jabade and et al., "Gaze Controlled Wheelchair," *International Journal of Research in Applied Science and Engineering Technology,* vol. 12, no. 7, 2024.
[8] E. Iacobelli, V. Ponzi, S. Russo, and C. Napoli, "Eye-tracking system with low-end hardware: development and evaluation," *Information,* vol. 14, no. 12, p. 644, 2023.
[9] A. H. Khaleel, T. H. Abbas, and A.-W. Sami Ibrahim, "Best low-cost methods for real-time detection of the eye and gaze tracking," *i-com,* vol. 23, no. 1, pp. 79-94, 2024.
[10] T. Soukupová and J. Čech, "Real-Time Eye Blink Detection Using Facial Landmarks," in *Computer Vision Winter Workshop*, 2016.
[11] J. Kwon, K. T. Oh, J. Kim, O. Kwon, H. C. Kang, and S. K. Yoo, "Facial Emotion Recognition using Landmark coordinate features," in *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2023: IEEE, pp. 4916-4918.
[12] A. Fischer-Janzen, T. M. Wendt, and K. V. Laerhoven, "A Scoping Review of Gaze and Eye Tracking–Based Control Methods for Assistive Robotic Arms," *Frontiers in Robotics and AI,* vol. 11, p. 1326670, 2024.
[13] F. Xu and et al., "Eye-Tracking Based Control of Assistive Devices," *Sensors,* 2020.

[14] L. Youwei, "Real-time eye blink detection using general cameras: a facial landmarks approach," *International Science Journal of Engineering & Agriculture,* vol. 2, no. 5, pp. 1-8, 10/01 2023, doi: 10.46299/j.isjea.20230205.01.