arXiv:2610.38569v1 [cs.RO] 28 Oct 2026

A Multi-Layer Navigation System for Smart Wheelchairs with a Limited Input Interface

Piero Simonetto [1], Anna Polato[1], Francesco Pasti[1],Muhammad Rameez ur Rahman [2], Sebastiano Vascon [2,3], Nicola Bellotto[1], Emanuele Menegatti[1,4], Stefano Tortora[1], Luca Tonin[1,4]

***Abstract*— Smart wheelchairs are robotic devices that gather information from the surroundings and allow people to safely drive through various environments. They usually rely on the interpretation of high-level user's intentions, typically decoded through human-robot interface (HRI) or brain-machine interface (BMI). However, navigating in an unknown environment and accurately understanding user commands remain significant challenges. Misinterpretation of commands or the environment can lead to user frustration, distrust of the system, or even unsafe situations. In this paper, we present a novel Multi-Layer Navigation System (MLNS) for a smart wheelchair that enables navigation in unknown environments while interpreting and safely contextualizing user's intentions even in the case of limited and faulty commands. The stack relies on a Virtual Sensors-Driven Potential Fields (ViSPF) method for obstacle detection that accounts for the wheelchair's actual size, a Model Predictive Control (MPC) module for trajectory generation, and an open-vocabulary computer vision algorithm for target identification. Finally, a system based on Priority Queue (PQ) was implemented for target selection. We validated our system on a real smart wheelchair with experiments involving 10 participants without disabilities, including scenarios with faulty input interfaces. The results demonstrate the robustness and effectiveness of our system in real-world environments, proving its potential for practical applications in assistive technologies.**

I. Introduction

Assistive Technology (AT) refers to any device and software designed to maintain and improve the functional capabilities of an individual, enabling their inclusion and participation in all areas of society. Recently, several studies have focused on the development of intelligent powered wheelchairs and on implementing appropriate human-robot interface (HRI) to enable people with severe motor or cognitive disabilities to control these systems [1], [2], [3].

This study was partially carried out within the PNRR research activities of the consortium iNEST funded by the European Union Next-GenerationEU (PNRR ECS_00000043) and within the project PRINPNRR22 EasyWalk (Prot. P2022XJ7TE). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

[1]Piero Simonetto, Anna Polato, Francesco Pasti, Stefano Tortora, Emanuele Menegatti, Luca Tonin are with Department of Information Engineering, University of Padova, Italy {piero.simonetto, anna.polato.2, francesco.pasti }@phd.unipd.it, {stefano.tortora, nicola.bellotto, emanuele.menegatti, luca.tonin }@unipd.it

[2] Muhammad Rameez ur Rahman [2], Sebastiano Vascon are part of the Ca' Foscari University of Venice {muhammad.rahman,sebastiano.vascon}@unive.it

[3] Sebastiano Vascon are part of the European Centre for Living Technology

[4]Emanuele Menegatti and Luca Tonin are also with the Padua Neuroscience Center, University of Padova, Italy

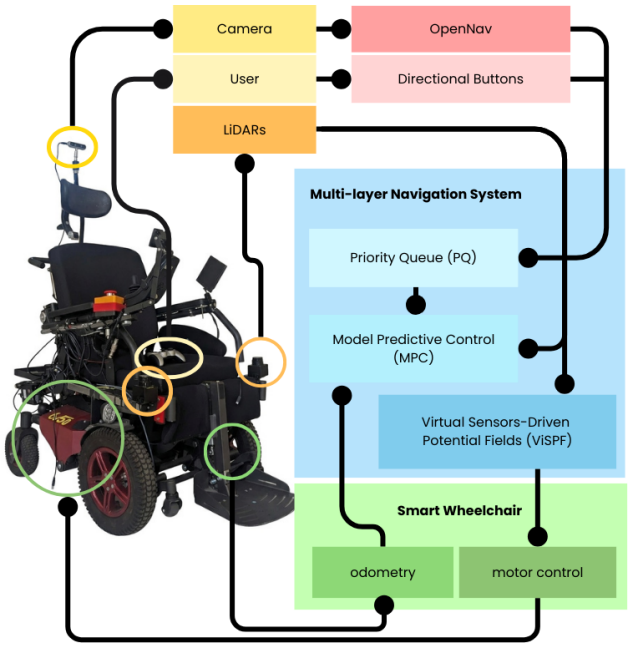


Fig. 1. The proposed Multi-Layer Navigation System (MLNS) for a smart wheelchair semi-autonomous navigation. Given the user and OpenNav inputs, a PQ combined with MPC and ViSPF controls the navigation of the wheelchair.

Researchers have focused on hardware and software solutions that provide alternative interfaces to the traditional joystick and that allow end-users to communicate their intentions to the navigation system without using their upper-limbs. The most common approaches rely on deep learning techniques for detecting and interpreting user commands—e.g., hand gestures [4], [5], eye movement [6], and voice commands [7]. Other methods are based on the use of a joystick controlled by the lips [8], tongue [9], or head-neck [10]. Recently, similar commands have been generated by decoding brain activity using brain-machine interface (BMI) [2].

Despite these advances, the integration of robotics in the assistive technology field is still at its infancy. Indeed, many solutions remain at the research or early validation stage, often demonstrating incremental improvements under controlled laboratory conditions. Their robustness and generalizability in real-world assistive scenarios are still limited. In practice, HRI for AT are highly sensitive to noise, environmental variability, sensor occlusions, and user-specific motor and cognitive fluctuations. As a consequence, they typically support discrete, high-level commands (e.g.,

left, right, stop, move) rather than continuous, intention-rich control signals. Although such abstractions simplify command decoding, they strongly constrain the interpretation of user intent. This limitation becomes particularly critical in navigation tasks involving close interaction with objects or people, where fine-grained adjustments, continuous modulation, and context awareness are required. In these situations, even small misclassifications or latency-induced errors may propagate into large trajectory deviations, reducing usability and increasing cognitive load. Therefore, understanding how to integrate such a limited input interface in the control of AT devices is not merely a performance optimization, but a prerequisite for effective navigation.

Furthermore, safety is a priority in these applications. Indeed, over the last few decades, many obstacle avoidance algorithms for smart wheelchairs have been evaluated. Among these, Artificial Potential Fields (APFs) have gained particular attention due to the ability to generate smooth trajectories and the straightforward implementation where attractive and repulsive forces are combined to guide the robot away from obstacles while steering it toward free spaces [11].

In this work, we introduce a novel semi-autonomous navigation system for smart wheelchairs to operate in unknown environments with limited user input while maintaining adaptability and safety. The proposed architecture combines reactive obstacle avoidance, predictive trajectory planning, and computer vision algorithms for target identification within a framework designed for assistive mobility.

The remainder of the paper is structured as follows. Section II reviews related works, identifies the limitation of current approaches, and defines the aim of this research. Section III presents the assistive platform and the developed navigation system, as well as the experimental protocol. Section IV reports the results of the experiments followed by discussion, future work, and conclusions in Sections V-VI.

II. Related Works

As mentioned above, research in HRI for AT must address limitations that extend beyond technical performance. Repeated misinterpretations, unpredictable behaviors, or lack of transparency can compromise user trust in both the interface and the assistive system as a whole [12]. To mitigate the impact of noisy or unreliable user inputs, several works have shifted toward fully autonomous navigation paradigms [13], [14]. In these systems, the user typically specifies only a high-level destination, while the robot autonomously manages local motion planning and obstacle avoidance. Although this strategy reduces cognitive load and increases robustness against command misinterpretation, it also reduces direct user involvement in the navigation process, potentially affecting perceived agency and acceptance. Empirical evidence suggests that many users prefer semi-autonomous or shared-control strategies, where assistance enhances safety without removing control authority [15]. Within this framework, several approaches rely on adaptive assistance mechanisms in which the level of autonomy is modulated according to the user's needs or explicit requests [16], [17].

In these systems, machine intervention is triggered in the presence of risk, instability, or degraded performance. Some shared-control architectures move beyond event-based arbitration and instead maintain a continuous contribution from both the user and the autonomous system. Rather than alternating authority, these approaches blend human intent and autonomous intelligence in real time, establishing a cooperative interaction [18], [19]. A critical requirement in such frameworks is the reliable estimation of user intention. In this direction, Casado et al. propose an intention-prediction method based on a generative diffusion model for short-term trajectory planning in a shared-control smart wheelchair paradigm [20]. The technique estimates user intention by jointly modeling user inputs and environmental information, producing motion plans that reflect both contributions.

Similarly, several works propose an alternative shared-control strategy based on Bayesian inference [21], [22], [23]. More advanced solutions work directly with neural signals to decode user's intentions by exploiting BMI-driven shared control to drive smart wheelchairs or telepresence robots [2], [24], [25]. Their probabilistic formulation explicitly accounts for uncertainty in intention decoding, reinforcing the central role of intention prediction in cooperative navigation.

In the context of assistive technologies and smart wheelchairs, classical reactive collision avoidance strategies remain widely adopted. The most investigated approaches are the Vector Field Histogram (VFH) [26], the Dynamic Window Approach (DWA) [27], [28], and APFs. While VFH and DWA provide structured velocity selection mechanisms, APFs remain particularly attractive in assistive robotics due to their intuitive formulation, smooth motion generation, and low computational cost. However, APFs methods are also affected by well-known limitations, extensively analyzed in foundational works such as [29]. These include local minima, where attractive and repulsive forces balance out, oscillatory behavior in narrow corridors or doorways, sensitivity to parameter tuning, and susceptibility to sensor noise. Several assistive robotics works have attempted to adapt and enhance APFs formulations to overcome these limitations. For example, in [2] authors implemented a APFs-based navigation strategy in a BMI-driven smart wheelchair and validated it with end-users. Although the system demonstrated feasibility in controlled environments, oscillations in narrow passages emerged due to sensor noise and the intrinsic limitations of the APFs formulation. Other works have adapted APFs-based navigation to shared-control frameworks, demonstrating their feasibility in hands-free interfaces and reporting improvements in safety and user experience metrics, including reduced perceived cognitive workload [30], [31]. A more modern direction involves integrating predictive control mechanisms into the navigation stack. MPC has been adopted as a trajectory planner in smart wheelchairs [32].

MPC is particularly suitable for tracking trajectories that resemble natural user motion patterns [33], while explicitly handling system constraints. Still, its deployment in assis-

tive platforms presents several structural limitations. The wheelchair is usually modeled as a unicycle robot, which reduces computational complexity and facilitates real-time implementation, but it neglects relevant dynamic effects. As a result, prediction accuracy may be degraded, potentially affecting trajectory optimality and tracking performance. As a consequence, collision avoidance is encoded in the distance between the obstacle and the center of the robot, ignoring its real geometry. This may create incoherence between the full range of feasible paths and the set computed by the collision avoidance system. Finally, the computational time strictly depends on the spatial configuration, density, and geometry of the obstacles in the environment. In cluttered scenarios or constrained passages, a higher number of active constraints can significantly increase the complexity of the optimization problem and slow down convergence, compromising real-time execution.

The problem of accurately representing the wheelchair's physical footprint has been addressed through simplified geometric hardware-based approximations. Most approaches model the platform as either circular or elliptical [34], [35] to facilitate analytical treatment and reduce computational complexity. Alternatively, some systems rely on multiple proximity sensors distributed along the perimeter of the wheelchair (e.g., ultrasonic arrays) to approximate its occupied space during navigation [31], [36]. One of the earliest works explicitly addressing wheelchair geometry in navigation is [37], which proposed modeling the platform through a discrete set of representative points and introduced the use of five control points for computing repulsive forces within a VFH-based framework.

Motivated by the limitations discussed above, this work addresses three main gaps in current assistive navigation systems. First, we introduce an enhanced ViSPF formulation that explicitly accounts for the actual geometry and footprint of the wheelchair, overcoming the simplifications associated with center-based collision constraints and circular approximations, without compromising real-time performance. Second, we propose a multi-layer navigation architecture that integrates ViSPF and MPC to leverage their complementary strengths while addressing the computational burden typically associated with MPC-based navigation, improving real-time feasibility even in cluttered environments. Third, we develop an HRI framework based on a PQ mechanism and a computer vision module to improve intention inference and environmental awareness. Finally, we experimentally evaluated the robustness of the proposed approach under noisy input conditions characteristic of multiple common HRIs for AT. The system was implemented and tested on a real smart wheelchair platform in semi-autonomous assistive scenarios.

III. Materials and Methods

A. Customized powered wheelchair

In this work, we customized a commercially available powered wheelchair (18.68E HI-LO Electric, Vassilli s.r.l., Padua, Italy) as shown in Figure 1. The velocity and position of the wheelchair were tracked using two Kübler KIS40 encoders attached to the front wheels by belt and the resulting odometry was computed according to [38]. Two Hokuyo URG-04LX-UG01 LIDARs were mounted in front of the wheelchair to detect obstacles, while a RealSense D435i camera, placed above the user's head, captured the environment with a field of view similar to the user's perspective. The user controlled the wheelchair using a Logitech F710 joystick. All components were connected to a Lenovo Thinkpad T15g Gen 2, 11th Gen Intel Core i9-11950H @ 2.60 GHz x 16, 32 GB RAM, NVIDIA GeForce RTX 3080. We used ROS Noetic 1.16 to provide a common interface for custom components. An R-Net chipset (Curtiss-Wright Corporation, Davidson, United States) was used to connect the R-Net CAN-bus network of the wheelchair to the external laptop.

B. Multi-Layer Navigation System (MLNS)

The architecture of MLNS is reported in Figure 1. It is composed of three main elements: a custom ViSPF layer, to ensure proactive and safe navigation in real-time, a MPC layer used to build smooth trajectories, and a PQ layer to fuse user's commands and decisions autonomously taken by the intelligent navigation system. The PQ relies on visual information provided by OpenNav [39], a computer vision module composed of YOLO-World [40] and MobileSAM [41] and designed for real-time open vocabulary 3D object detection. The navigation stack computes the velocity command in a real-time reconstructed local map that tracks the possible obstacles. The map was used by both the MPC and ViSPF control layers. Since the wheelchair occludes its backside to the lasers, a spatio-temporal voxel plugin [42] was used to track obstacles in this area. By construction, the field of view of the laser in this setup did not ensure that the space right behind the wheelchair was empty. For this reason, we decided to inhibit direct reverse commands.

C. Virtual Sensors-Driven Potential Fields (ViSPF) Layer

To ensure a suitable clearance between the obstacles and the robot's entire footprint, and to prevent collisions with any part of it, we modified the APFs layer to reproduce some *Virtual Sensors (VS)*. In this new method, reported in Figure 2, instead of modeling the wheelchair as a unique point, each vertex of the rectangle area of the wheelchair p_i, $i \in \{1, 2, 3, 4\}$ is subject to a linear and angular velocity based on its position with respect to the nearest obstacles (repellers) around the wheelchair, with an approach similar to that reported in [2]. During navigation, the obstacle positions are provided in polar coordinates in p_i centered reference frame. In order to set an upper limit on the computation time, the angular values were discretized into $N = 36$ sectors. Linear velocity is calculated as

$$v_{l,i}(t) = V_{TMAX}(1 - e^{-\frac{d_{O,i} - D_F}{\tau}}) \tag{1}$$

where V_{TMAX} is the maximum velocity (0.2 m/s), D_F is the safety distance. $d_{O,i}$ is the distance detected between the nearest obstacle and p_i in its range of motion and τ is

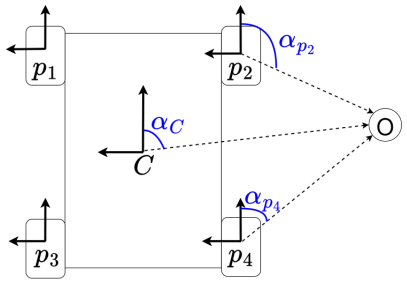


Fig. 2. Graphical representation of the proposed ViSPF. The wheelchair is represented with a rectangle surrounded by its four wheels. C represents the center of the wheelchair and $p_{\{1,2,3,4\}}$ the new VS.$\alpha_{\{C,p_1,p_2,p_3,p_4\}}$.

a smoothing parameter. The resulting linear velocity is the minimum among the four calculated velocities. At the same time, the angular velocity is calculated as

$$v_{\alpha,i}(t) = \sum_{k=1}^{N} \lambda_k (\alpha_{p_i} - \alpha_C) e^{-\frac{(\alpha_{p_i} - \alpha_C)^2}{2\sigma_k^2}} \tag{2}$$

where N is the number of angular sectors and λ_k depends on the distance (d_k) of the nearest obstacle in sector k from the center of the wheelchair:

$$\lambda_k = e^{-\frac{d_k - D_F}{\tau}} \tag{3}$$

and $\alpha_{p,i}$ and α_C are the angles of the obstacle with respect to the point p_i and the center of the robot C. σ_k represents the strength of the detected repeller. Recalling that $\alpha = 0$ implies moving in a straight direction, the subtraction $(\alpha_{p_i} - \alpha_C)$ defines the sign of the velocity value.

The final angular velocity is computed as the sum of all components:

$$v_\alpha(t) = \sum_{i=1}^{4} v_{\alpha,i}(t) + v_a \tag{4}$$

with v_a the attractor's angular component.

This method allows navigation to adapt to the wheelchair's shape and dimension and ensures precision in collision avoidance.

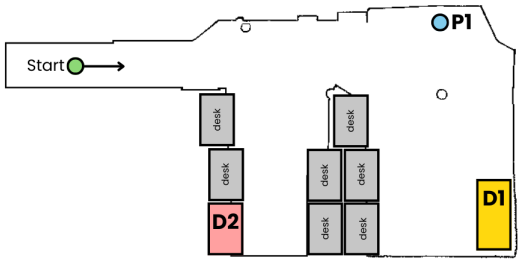


Fig. 3. Map of the laboratory where the experiment took place. The green point is the starting position, Target *D1* and *D2* are desks, and Target *P1* is a person.

D. MPC and trajectory generation

A MPC is exploited to generate a trajectory from the current wheelchair position to a given target (see next section). Our implementation of the MPC is based on the problem defined in Eq. 3 of [33]. In this formulation, obstacles are identified by clustering the wheelchair's surroundings and representing them as discrete points at a distance that accounts for the wheelchair's physical dimensions. To ensure that the solver generates reliable solutions, specific constraints have been implemented: the maximum number of obstacles is limited to 50, and the planning horizon is restricted to the next 30 seconds of motion.

Conversely to traditional approaches—where the MPC is used to directly provide the command velocity—the generated path is sub-sampled at fixed intervals of 0.3 m and the resulting samples are considered as attractors for the ViSPF layer. This solution ensures a safe control even if environmental dynamics change faster and prevents the classical oscillations of the artificial potential fields.

E. Target selection: Priority Queue (PQ)

MPC requires a target to model a path. The choice of target is expected from the PQ that periodically selects the current goal among the available ones. We provide the following two implementations. PQ *A* exposes two targets: the one sent by the user and the one sent by default by the system. The user's input is the first choice and overcomes the default behavior. PQ *B* exposes three targets: The user's one, the default's one, and a target provided by the computer vision module OpenNav. The OpenNav target stands in the middle, it overcomes the default behavior but follows the user's input.

Regarding the type of target, the user sends it via a limited input interface based on the joystick described below. OpenNav sets a target on a surrounding object, the closest to the moving direction of the wheelchair, as described above. Eventually, the default target point is set in front of the wheelchair, 2 meters ahead, every 4 seconds.

In this work, we experimentally evaluated both PQs, investigating how OpenNav can influence their usability and navigation itself in the presence of misinterpreted commands.

F. Limited input interface

The MLNS is tested on experiments in which the user communicates to the robot through a limited input interface. More precisely, the user can choose only between two turning directions (i.e., left and right) as in the case of head-neck joystick. The direction is selected through the directional buttons of a joystick and transformed into a point in the space around the wheelchair. Furthermore, we introduced a minimum refractory period of 3 seconds between consecutive user commands to emulate anti-tremors filters usually implemented in this kind of interfaces. Finally we designed a perturbation process that reverts the user's command with a fixed probability of 0.2 to simulate wrong button presses.

G. Experiments

The semi-autonomous navigation system of the smart wheelchair was evaluated by 10 users without disabilities (3 females, 7 males, 27 ± 3 years old). Users sat on the wheelchair and were asked to navigate from a fixed starting point to a target destination (identified by a common object) by sending commands to the wheelchair using the joystick directional buttons.

Each user was asked to navigate from a starting point A to one of the targets reported in Figure 3 (i.e., two desks and one person identified respectively by letters *D1*, *D2*, *P1* on the map). The users autonomously decided when the target was reached by stopping the navigation.

Each user performed 12 navigation runs for each target, 4 for each of the three following modalities based on PQ policies (Section III-E):

- Baseline (*B*): user's input priority (PQ *A*), no perturbations;
- Without OpenNav (*WO*): user's input priority (PQ *A*), with perturbations;
- With OpenNav (*ON*): user's input or OpenNav priority (PQ *B*), with perturbations.

In the first modality *B*, users had full control of the wheelchair and no errors were introduced in the input interface. Conversely, in *WO* and *ON* users evaluated the system with a faulty input interface and without and with the OpenNav module, respectively. During navigation, participants were blinded to the active condition, which was randomized across trials. In *ON*, OpenNav was configured to detect the "desk" for Target *D1* and *D2* and the "person" for Target *P1*.

IV. Results

First, it is worth to mention that the MLNS allowed the wheelchair to move in a cluttered real environment without any crashes and among 360 trials conducted by 10 participants, the system achieved a 99.72% task completion rate, with only one interrupted run (Target *D2*, approach *ON*) after user request.

Figure 4A presents the normalized heatmap aggregating all user trajectories (central panel). The white dashed lines represent the average *B* trajectories for each target. The spatial distribution highlights the dominant motion paths emerging across trials, providing a global view of path consistency and dispersion. Figures 4B-D illustrate the results separately for each target (*D1*, *D2*, and *P1*) and for each modality (*B*, *WO*, and *ON*). Heatmaps reveal clear differences in spatial dispersion. *WO* trajectories exhibit greater variability, with occasional deviations far from the nominal path. These paths correspond to runs in which perturbations in user commands led to unintended deviations. In contrast, the implementations of *B* and *ON* result in a visibly reduced dispersion and a more concentrated flow of trajectories toward the target. This suggests that perception-driven assistance effectively constrains the set of feasible motion strategies, promoting more repeatable and stable navigation behavior, similar to the baseline (i.e., without input perturbations).

To further quantify the efficiency of the system and consistency of the generated paths, Figure 5 depicts the distribution of all paths length for each target and navigation modality. For target *D1*, the modality *ON* shows a statistically relevant decrease ($\sim 0.6m$, $p < 0.05$, Bonferroni corrected) in the paths length compared to *B*. Furthermore, across all targets, the modality *WO* displays significantly larger ranges and more pronounced outliers compared to both other methods ($p < 0.05$, Bonferroni corrected). These results indicate that while the *B* and *ON* pipelines maintain a steady and predictable motion path, the *WO* configuration is prone to occasional substantial deviations and less efficient trajectories, as confirmed by the higher standard deviations reported in Table III.

TABLE I

Mean and standard deviation of completion time per target and method

	B	*WO*	*ON*
D1	92.1 ± 17.3	108.4 ± 44.1	91.5 ± 31.5
D2	69.9 ± 15.5	74.0 ± 65.2	77.3 ± 38.3
P1	77.7 ± 13.8	81.7 ± 39.2	77.7 ± 23.9

TABLE II

Mean and standard deviation of number of commands per target and method

	B	*WO*	*ON*
D1	4 ± 3.5	5.3 ± 5.4	3.1 ± 7.6
D2	3.2 ± 1.8	4.7 ± 9.4	5.8 ± 6.1
P1	3.1 ± 4.5	3.9 ± 6.1	2 ± 2.9

Figure 6 presents the distributions of completion time (top) and the number of user commands (bottom) for the three methods (*B*, *WO*, *ON*). In targets *D1* and *P1*, *ON* exhibits a lower mean completion time and a narrower inter-quartile range compared to *WO*, indicating not only faster execution, but also reduced variability between runs. This suggests that perception-driven assistance promotes more consistent performance between users and trials. A similar trend is observed in the distribution of user commands: *ON* generally reduces the number of corrective inputs required, reflecting a lower interaction burden and a smoother control process. For target *D2*, the differences between the two assisted conditions are less pronounced. The mean completion times are comparable between *ON* and *WO* modalities (77.3 vs. 74.0 s, Table I), and the average number of commands is slightly higher in the *ON* modality (5.8 vs 4.7, Table II).

V. Discussion

The proposed MLNS showed robust performance in an unknown environment, effectively mitigating the well-known limitations of both MPC and APFs approaches. With the ViSPF method that provides reactive obstacle avoidance and MPC that allows predictive trajectory optimization, their integration enables the system to take advantage of the complementary strengths of both algorithms. The execution of user commands relies on the MPC process, which introduces computation-dependent latency. During this short planning interval, the wheelchair maintains the last MPC calculated trajectory while the ViSPF layer continues to enforce local collision avoidance constraints, with a processing rate of 50 Hz. This layered interaction ensures continuous safety and ensures a stable and smooth navigation experience. In addition, the cooperation between the predictive MPC layer and the proposed reactive ViSPF layer overcomes the classical oscillatory behaviors typically observed with pure APFs-based navigation, especially in confined environments

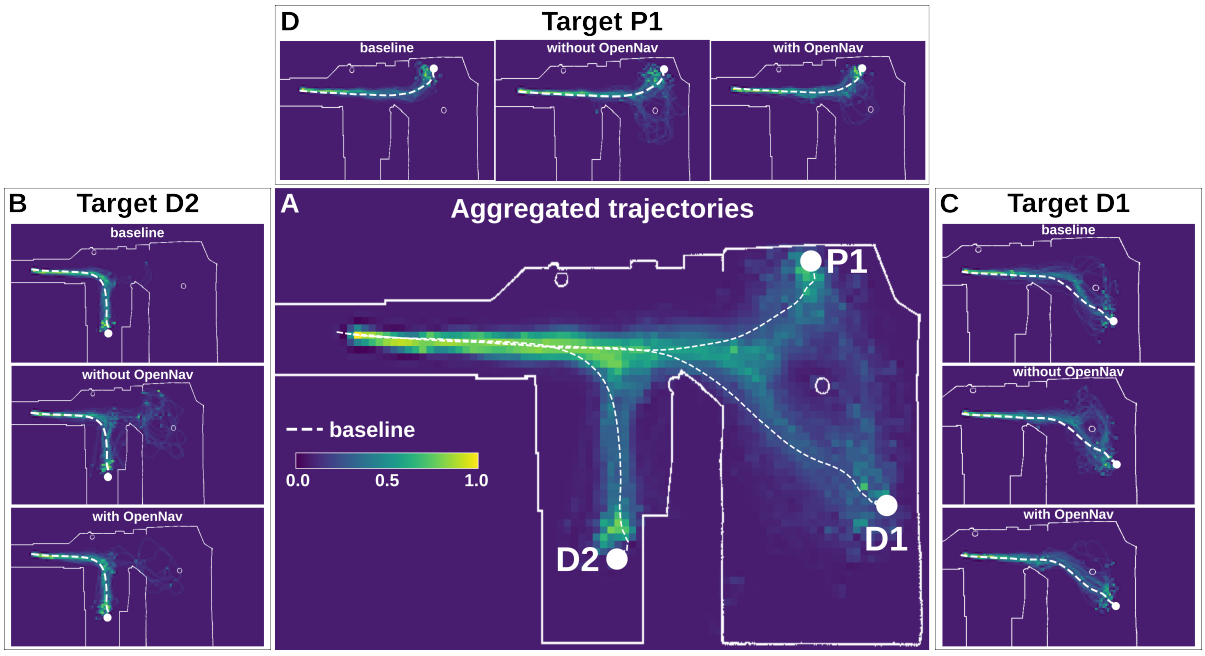


Fig. 4. A) Normalized heatmaps of the aggregated trajectories produced during the experiments (center). B-D) Normalized heatmaps of the trajectories divided per target and modality. White dashed lines represent the average Baseline path for each target. Heatmap represents a color gradient of the crossed positions, from the more (in yellow) to the less (in blue) common.

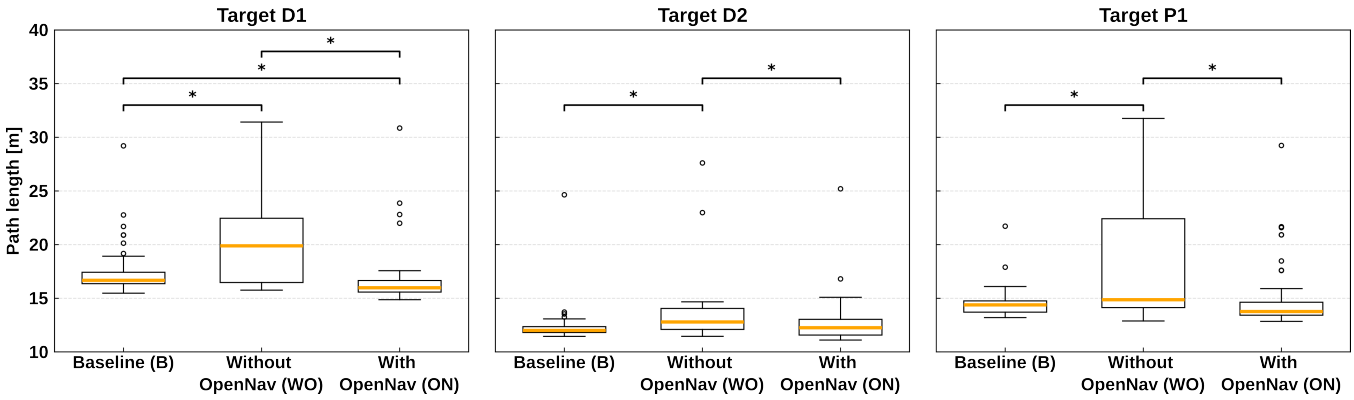


Fig. 5. Distribution of the paths length traveled for each target and navigation modality. Horizontal brackets indicate pairs with a statistically significant difference (Kruskal-Wallis with Dunn's post-hoc test, $p < 0.05$)

TABLE III

MEAN AND STANDARD DEVIATION OF LENGTH OF THE PATHS PER TARGET AND METHOD

	B	*WO*	*ON*
D1	16.6766 ± 2.5074	19.8891 ± 7.1318	15.9831 ± 4.2086
D2	12.0068 ± 2.0267	12.7959 ± 10.3488	12.2596 ± 5.7333
P1	14.3877 ± 1.5002	14.8688 ± 6.6185	13.7696 ± 3.2384

such as narrow corridors or door passages. Furthermore, the introduction of VS (Section III-C) enables the experimental trials to be conducted without any collisions while ensuring that the entire footprint of the smart wheelchair maintains a consistent safe clearance from the surrounding objects and walls. A video of the above-mentioned navigation experience is available in the paper attachments.

The integration of OpenNav into PQ proved to be an effective design choice. Across all tasks, the trajectories generated with *ON* remain closer to the *B* than those obtained with *WO*, indicating stronger alignment with the user-driven paths of *B* (not affected by perturbations). In general, the pipeline *WO* exhibits greater spatial dispersion, reflecting increased variability in navigation behavior. For targets *D1* and *P1*, the efficiency gains introduced by OpenNav are evident. Compared to the corresponding configuration *WO*, users require fewer corrective commands and achieve shorter completion times, while maintaining trajectory profiles that closely resemble the baseline. This suggests that OpenNav reduces the interaction effort without compromising user intent. Nevertheless, target *D2* requires additional discussions. The task consists of approaching a table located at the end of a corridor, with additional tables positioned along both sides. Two critical phases characterize this scenario: the initial turn required to enter the corridor and the selection of the target table. The corridor entry maneuver is strongly dependent on

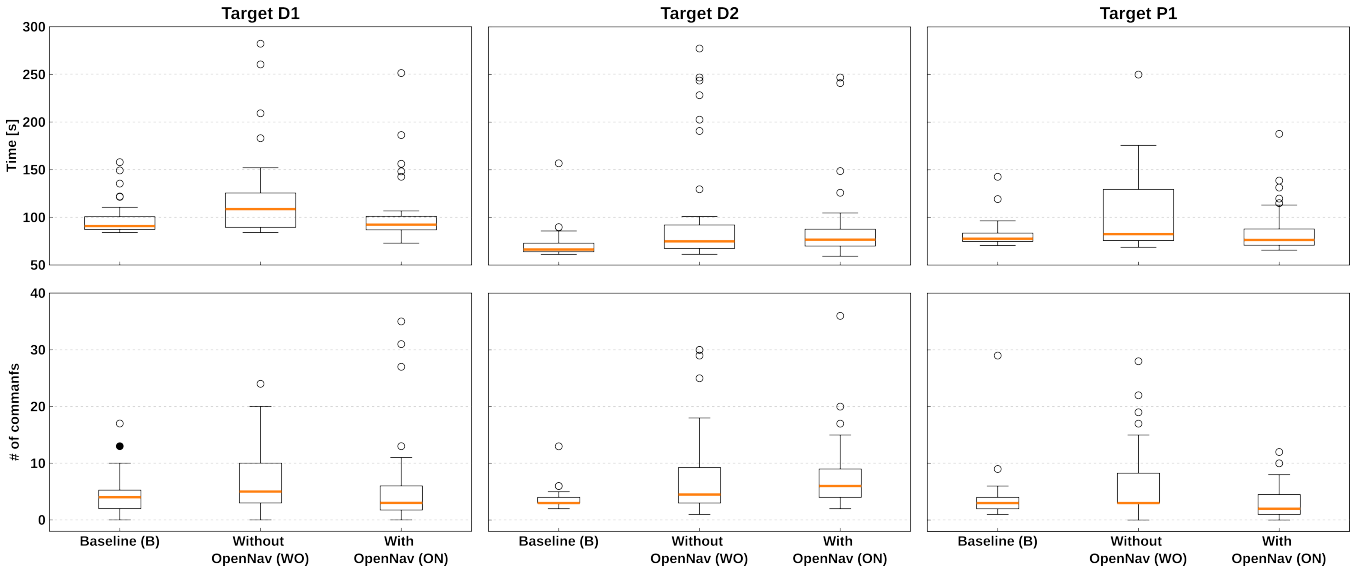


Fig. 6. Distribution of the completion time (top) and number of commands sent by the user (bottom) of each target.

the initial user command in combination with the collision avoidance system. When this first input is inaccurate, due to the limited input interface, users may deviate significantly from the nominal path and require substantial corrective actions to realign with the corridor entrance. This behavior is consistent with the outliers observed in Figure 4, as both *ON* and *WO* configurations, affected by perturbations, exhibit some trajectories that diverge markedly from the baseline. Regarding table selection, the presence of multiple nearby tables may lead OpenNav to prioritize the closest feasible target. In such cases, users may issue additional commands to override or refine the system's choice and continue toward the intended table. This interaction dynamic highlights a balanced human-robot synergy. Within this framework, the human component becomes essential in resolving environmental ambiguities. This dynamic explains the slightly higher number of commands and longer execution times observed for target *D2* under the OpenNav condition (Section IV), and is a clear reflection of the shared-control paradigm adopted in this study and of the collaborative nature of the system.

Overall, OpenNav is particularly beneficial during the early and mid phases of navigation, when the user is still far from the target and can assess and, if necessary, refine the system's choice. Its benefit decreases during the final approach to the target locations, when users typically prefer to manually adjust the wheelchair's orientation for precise alignment.

VI. Conclusions

In this paper, we present and experimentally validate a novel multi-layer navigation framework for smart wheelchairs that integrates a new Virtual Sensors-Driven Potential Fields layer, Model Predictive Control, and OpenNav computer vision algorithm within a unified architecture. The proposed system enables the smart wheelchair to plan motions in space accounting for the entire footprint of the wheelchair. Moreover, we introduce and evaluate two priority queue-based pipelines to regulate a hierarchy of control commands while preserving user authority at the highest level. The PQs are designed to evaluate the system in the case of a limited input interface. We tested the setup with a real device and with 10 users in order to verify the performance. Experimental results indicate that users were able to navigate the environment successfully, while relying on the assistive system, suggesting effective human–robot cooperation under constrained control conditions. The user maintains control of the wheelchair, with the only exception of a not reachable direction. In this case, the system tends to carry the user to the nearest safe position. The configuration incorporating OpenNav demonstrated clear advantages in tasks involving distant targets, enabling simpler and more direct trajectories, shorter completion times, and requiring fewer corrective inputs. Future studies will include validation on a larger population of users and in more complex and heterogeneous environments, in order to evaluate the system performance under realistic assistive conditions and to further assess its robustness and adaptability.

References

[1] S. Desai, S. S. Mantha, and V. M. Phalle, "Advances in smart wheelchair technology," in *2017 International Conference on Nascent Technologies in Engineering (ICNTE)*, 2017, pp. 1–7.

[2] L. Tonin, S. Perdikis, T. D. Kuzu, J. Pardo, B. Orset, K. Lee, M. Aach, T. A. Schildhauer, R. Martínez-Olivera, and J. d. R. Millán, "Learning to control a bmi-driven wheelchair for people with severe tetraplegia," *Iscience*, vol. 25, no. 12, 2022.

[3] J. Leaman and H. M. La, "A comprehensive review of smart wheelchairs: Past, present, and future," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 4, pp. 486–499, 2017.

[4] A. Bhawarthi, N. Lokhande, S. Lokhande, R. Lole, T. Lonkar, V. Loya, and B. Lunawat, "Hand gesture controlled wheelchair using image processing," in *2024 IEEE Symposium on Industrial Electronics & Applications (ISIEA)*, 2024, pp. 1–6.

[5] M. S. Sadi, M. Alotaibi, M. R. Islam, M. S. Islam, T. Alhmiedat, and Z. Bassfar, "Finger-Gesture Controlled Wheelchair with Enabling IoT," *Sensors*, vol. 22, no. 22, p. 8716, Nov. 2022.

[6] J. Xu, Z. Huang, L. Liu, X. Li, and K. Wei, "Eye-Gaze Controlled Wheelchair Based on Deep Learning," *Sensors*, vol. 23, no. 13, p. 6239, Jul. 2023.

[7] A. I. Iskanderani, F. R. Tamim, M. M. Rana, W. Ahmed, I. M. Mehedi, A. J. Aljohani, A. Latif, S. A. L. Shaikh, M. Shorfuzzaman, F. Akther, T. Palaniswamy, and M. T. Vellingiri, "Voice Controlled Artificial Intelligent Smart Wheelchair," in *2020 8th International Conference on Intelligent and Advanced Systems (ICIAS)*. Kuching, Malaysia: IEEE, Jul. 2021, pp. 1–5.

[8] M. A. José and R. de Deus Lopes, "Human–computer interface controlled by the lip," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 302–308, 2015.

[9] M. Mohammadi, H. Knoche, M. Gaihede, B. Bentsen, and L. N. S. Andreasen Struijk, "A high-resolution tongue-based joystick to enable robot control for individuals with severe disabilities," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*. Toronto, ON, Canada: IEEE, Jun. 2019, pp. 1043–1048.

[10] H. Zhang, B.-C. Chang, Y.-J. Rue, and S. K. Agrawal, "Using the Motion of the Head-Neck as a Joystick for Orientation Control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 2, pp. 236–243, Feb. 2019.

[11] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on systems, Man, and Cybernetics*, vol. 19, no. 5, pp. 1179–1187, 1989.

[12] D. Firmino De Souza, S. Sousa, K. Kristjuhan-Ling, O. Dunajeva, M. Roosileht, A. Pentel, M. Mõttus, M. Can Özdemir, and Gratšjova, "Trust and Trustworthiness from Human-Centered Perspective in Human–Robot Interaction (HRI)—A Systematic Literature Review," *Electronics*, vol. 14, no. 8, p. 1557, Apr. 2025.

[13] F. Leishman, V. Monfort, O. Horn, and G. Bourhis, "Driving assistance by deictic control for a smart wheelchair: The assessment issue," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 1, pp. 66–77, 2013.

[14] L. F. Manta, C. F. Pana, D. Cojocaru, I. C. Vladu, D. M. Patrascu-Pana, and A. Dragomir, "APF-based Control for Obstacle Avoidance in Smart Electric Wheelchair Navigation," in *2021 22nd International Carpathian Control Conference (ICCC)*. Velké Karlovice, Czech Republic: IEEE, May 2021, pp. 1–6.

[15] C. Yang, H. Patel, M. Kleiman-Weiner, and M. Cakmak, "Preserving Sense of Agency: User Preferences for Robot Autonomy and User Control across Household Tasks," in *2025 34th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. Eindhoven, Netherlands: IEEE, Aug. 2025, pp. 1595–1602.

[16] V. Schettino and Y. Demiris, "Improving generalisation in learning assistance by demonstration for smart wheelchairs," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5474–5480.

[17] L. Devigne, F. Pasteau, T. Carlson, and M. Babel, "A shared control solution for safe assisted power wheelchair navigation in an environment consisting of negative obstacles: a proof of concept," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Bari, Italy: IEEE, Oct. 2019, pp. 1043–1048.

[18] L. Devigne, V. K. Narayanan, F. Pasteau, and M. Babel, "Low complex sensor-based shared control for power wheelchair navigation," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, South Korea: IEEE, Oct. 2016, pp. 5434–5439.

[19] A. V. Nguyen, L. B. Nguyen, S. Su, and H. T. Nguyen, "Shared control strategies for human - Machine interface in an intelligent wheelchair," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. Osaka: IEEE, Jul. 2013, pp. 3638–3641.

[20] F. E. Casado, R. C. Quesada, and Y. Demiris, "Navigating Uncertainty: Diffusion-Based User Intention Estimation for Wheelchair Assistance," *IEEE Transactions on Robotics*, vol. 42, pp. 80–97, 2026.

[21] X. Deng, Z. L. Yu, C. Lin, Z. Gu, and Y. Li, "A Bayesian Shared Control Approach for Wheelchair Robot With Brain Machine Interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 1, pp. 328–338, Jan. 2020.

[22] H. T. Trieu, H. T. Nguyen, and K. Willey, "Shared control strategies for obstacle avoidance tasks in an intelligent wheelchair," in *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Vancouver, BC: IEEE, Aug. 2008, pp. 4254–4257.

[23] H. Trieu, K. Willey, and H. Nguyen, "Adaptive shared control strategies based in the Bayesian recursive technique in an intelligent wheelchair," in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Minneapolis, MN: IEEE, Sep. 2009, pp. 7118–7121.

[24] R. Leeb, L. Tonin, M. Rohm, L. Desideri, T. Carlson, and J. d. R. Millán, "Towards independence: A bci telepresence robot for people with severe motor disabilities," *Proceedings of the IEEE*, vol. 103, no. 6, pp. 969–982, 2015.

[25] R. Leeb, S. Perdikis, L. Tonin, A. Biasiucci, M. Tavella, M. Creatura, A. Molina, A. Al-Khodairy, T. Carlson, and J. d.R. Millán, "Transferring brain–computer interfaces beyond the laboratory: Successful application control for motor-disabled users," *Artificial Intelligence in Medicine*, vol. 59, no. 2, pp. 121–132, 2013.

[26] D. Bell, J. Borenstein, S. Levine, Y. Koren, and J. Jaros, "An assistive navigation system for wheelchairs based upon mobile robot obstacle avoidance," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*. San Diego, CA, USA: IEEE Comput. Soc. Press, 1994, pp. 2018–2022.

[27] S. P. Levine, D. A. Bell, and Y. Koren, "NavChair: An example of a shared-control system for assistive technologies," in *Computers for Handicapped Persons*, G. Goos, J. Hartmanis, J. Leeuwen, W. L. Zagler, G. Busby, and R. R. Wagner, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, vol. 860, pp. 136–143, series Title: Lecture Notes in Computer Science.

[28] R. Adamek, M. K. Bugeja, S. G. Fabri, and R. Grepl, "Enhancing the Obstacle Avoidance Capabilities of a Smart Wheelchair," in *2022 20th International Conference on Mechatronics - Mechatronika (ME)*. Pilsen, Czech Republic: IEEE, Dec. 2022, pp. 1–7.

[29] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. Sacramento, CA, USA: IEEE Comput. Soc. Press, 1991, pp. 1398–1404.

[30] Y. Chen, M. Mansouri, C. Xiao, Z. Wang, E. T. Hsiao-Wecksler, and W. R. Norris, "Enabling Shared-Control for A Riding Ballbot System," in *2025 IEEE 21st International Conference on Automation Science and Engineering (CASE)*. Los Angeles, CA, USA: IEEE, Aug. 2025, pp. 858–865.

[31] D. Ashley, K. Ashley, R. Alqasemi, and R. Dubey, "Semi-autonomous mobility assistance for power wheelchair users navigating crowded environments," in *2017 International Conference on Rehabilitation Robotics (ICORR)*. London: IEEE, Jul. 2017, pp. 1025–1030.

[32] E. Ceravolo, M. Gabellone, M. Farina, L. Bascetta, and M. Matteucci, "Model predictive control of an autonomous wheelchair," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9821–9826, 2017.

[33] D. De Lazzari, P. Simonetto, N. Turcato, L. Tonin, and R. Carli, "Nonlinear model predictive control of a bmi-guided wheelchair for navigation in unknown environments," in *2024 European Control Conference (ECC)*. IEEE, 2024, pp. 3582–3587.

[34] S. Chatzidimitriadis, P. Oprea, M. Gillham, and K. Sirlantzis, "Evaluation of 3D obstacle avoidance algorithm for smart Powered Wheelchairs," 2017.

[35] H. Kong and Q. Zhang, "Path planning for collision avoidance based on artificial potential field with vehicle dimension constraint," in *2022 41st Chinese Control Conference (CCC)*, 2022, pp. 5371–5376.

[36] C. Diao, S. Jia, G. Zhang, Y. Sun, X. Zhang, Y. Xue, and X. Li, "Design and realization of a novel obstacle avoidance algorithm for intelligent wheelchair bed using ultrasonic sensors," in *2017 Chinese Automation Congress (CAC)*. Jinan: IEEE, Oct. 2017, pp. 4153–4158.

[37] S. Levine, D. Bell, L. Jaros, R. Simpson, Y. Koren, and J. Borenstein, "The NavChair Assistive Wheelchair Navigation System," *IEEE Transactions on Rehabilitation Engineering*, vol. 7, no. 4, pp. 443–451, Dec. 1999.

[38] R. Xu, "Path planning of mobile robot based on multi-sensor information fusion," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, p. 44, 2019.

[39] M. R. u. Rahman, P. Simonetto, A. Polato, F. Pasti, L. Tonin, and S. Vascon, "Opennav: Efficient open vocabulary 3d object detection for smart wheelchair navigation," in *Computer Vision – ECCV 2024 Workshops*. Cham: Springer Nature Switzerland, 2025, pp. 372–387.

[40] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," *arXiv preprint arXiv:2401.17270*, 2024.

[41] Y. Liu, Y. Zhao, X. Zhang, X. Wang, C. Lian, J. Li, P. Shan, C. Fu, X. Lyu, L. Li *et al.*, "Mobilesam-track: Lightweight one-shot tracking and segmentation of small objects on edge devices," *Remote Sensing*, vol. 15, no. 24, p. 5665, 2023.

[42] S. Macenski, D. Tsai, and M. Feinberg, "Spatio-temporal voxel layer: A view on robot perception for the dynamic world," *International Journal of Advanced Robotic Systems*, vol. 17, no. 2, 2020.