

Article

# Investigating Learning Assistance by Demonstration for Robotic Wheelchairs: A Simulation Approach

Vinícius Barbosa Schettino <sup>1</sup>, Murillo Ferreira dos Santos <sup>1,\*</sup> and Paolo Mercorelli <sup>2</sup>

<sup>1</sup> Department of Electroelectronics, Federal Center of Technological Education of Minas Gerais (CEFET-MG), Leopoldina 36700-001, Brazil; viniusb@cefetmg.br

<sup>2</sup> Institute for Production Technology and Systems (IPTS), Leuphana Universität Lüneburg, 21335 Lüneburg, Germany; paolo.mercorelli@leuphana.de

\* Correspondence: murillo.ferreira@cefetmg.br

## Abstract

A major challenge for robots that provide physical assistance is adapting to the needs of different people. To overcome this, personalised assistive models can be created by observing the demonstrations of help provided by an assistant, a setting known as Learning Assistance by Demonstration (LAD). In this work, the case of robotic wheelchairs and drivers with hand control disabilities, which make navigation more challenging, was considered. To better understand LAD and its features, a simulator capable of generating repeatable examples of the triadic interactions between drivers, robots, and assistants was developed. The software is designed to be modular and parametrisable, enabling customisation and experimentation with various synthetic disabilities and scenarios. This approach was employed to design more effective data collection procedures and to enhance learning models. With these, it is shown that, at least in simulation, LAD can be used as follows: for different disabilities; to help consistently; to generalise to physically different environments; and to create customised assistive policies. In summary, the results provide further evidence that LAD is a viable approach for efficiently creating personalised assistive solutions for robotic wheelchairs.



Academic Editor: Giovanni Muscato

Received: 11 August 2025

Revised: 19 September 2025

Accepted: 25 September 2025

Published: 28 September 2025

**Citation:** Schettino, V.B.; Santos, M.F.d.; Mercorelli, P. Investigating Learning Assistance by Demonstration for Robotic Wheelchairs: A Simulation Approach. *Robotics* **2025**, *14*, 136. <https://doi.org/10.3390/robotics14100136>

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** assistive robotics; learning by demonstration; learning assistance by demonstration; robotic wheelchair

## 1. Introduction

Robotic wheelchairs are powered wheelchairs with added sensors and a computer [1,2]. The sensor data are used to assist in navigation, usually to achieve safer or faster driving or to reduce the driver's cognitive workload [3–5]. This can occur in the form of autonomous navigation [6] or reactive approaches, such as collision or obstacle avoidance [7,8], which can help in preventing loss of residual skills [9].

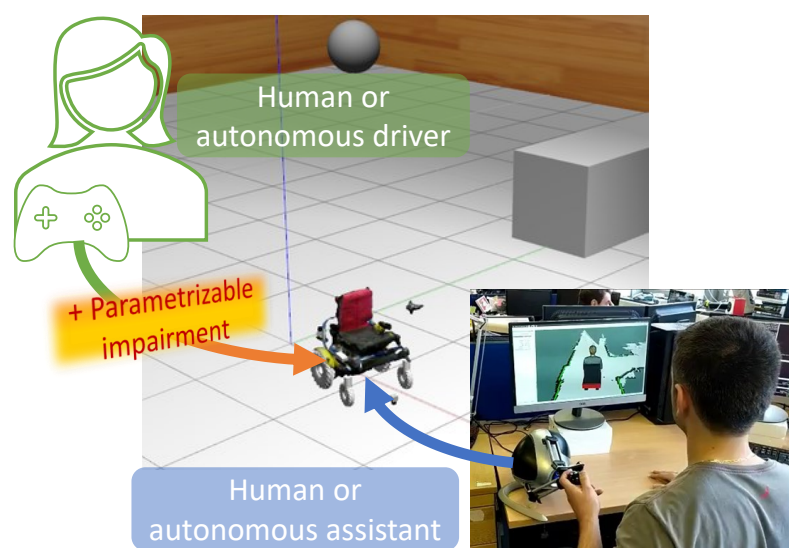
A great deal of research has been carried out with these robots, also known as smart wheelchairs. However, the focus is usually on drivers who have complete hand control to manipulate a joystick [10,11], or no control, in which case, alternative input interfaces are needed [12–16]. Less attention has been given to those who, besides needing a wheelchair for personal mobility, also struggle with hand control impairments. Although significant, this application is challenging due to the large variations in the type of support that may be needed. For example, besides difficulty in walking, cerebral palsy can lead to muscle weakness or spasms in the upper limbs, whereas Parkinson's disease usually leads to

tremors of the hand [17]. In each case, a custom assistive policy should be employed, as using a generic solution can lead to more frustration than support [18,19].

When personalised assistive solutions are needed, Learning Assistance by Demonstration (LAD) can be employed [20], which is a subset of Learning by Demonstration (LbD) [21,22]. With LAD, demonstrations of help are provided by a human assistant (ideally, someone familiar with the driver and their disability, such as caretakers or physiotherapists), recorded, and then used to train an algorithm that maps user input and sensor data to the assistance that should be offered. If good generalisation is achieved, this mapping can be used as a personalised assistive policy, allowing the robot to help the driver autonomously.

In a previous work, Schettino and Demiris [23] used a custom wheelchair teleoperation platform to run preliminary tests with LAD, achieving encouraging results [24]. Validating this technique on a broader scale, however, is complicated by the limited number of driver–assistant interactions that can be recorded due to cost, time (especially for healthcare workers [25]), and space constraints. Additionally, human drivers are naturally susceptible to factors such as concentration, fatigue, and mood, which act as confounding variables in experimentation. To temporarily circumvent these issues, simulations can be employed, speeding up research development and allowing a more comprehensive understanding of LAD before conducting larger trials with humans.

Therefore, in this work, a simulator was developed, as depicted in Figure 1, which reproduces the experience of navigating a wheelchair with a hand control impairment:



**Figure 1.** Simulation environment, where drivers with artificial hand control impairments can interact with a robotic wheelchair and a remote assistant. Humans can take up the roles of driver and/or assistant, or the full triadic interaction can be autonomously simulated. Configurable impairments allow for the quick testing of LAD performance against different driving behaviours.

Human drivers and assistants can share control of a robotic wheelchair, or the entire triadic interaction can be simulated without human intervention. Repeated runs with the latter setup were used to assess how variations in data collection, model architecture, and training procedures affect assistive performance. The results enabled advancement in the development of the LAD models, with a special focus on improving generalisation. Lastly, this platform was used to verify whether the claimed features of LAD hold true, such as generalisation to unseen environments, applicability to generic disability types, and creation of personalised assistive models. Although fundamental in motivating the usage of LAD, the validity of these claims had not been demonstrated before.

The text is organised as follows: Section 2 reviews the works most closely related to this study. Section 3 describes the software developed for simulating indoor wheelchair navigation. Section 4 details the architecture of the model used for learning from the demonstrations of assistance, as well as the hyperparameter optimisation procedure and data preprocessing techniques employed. Section 5 discusses the experiments that were conducted to deepen the understanding of LAD and its application to robotic wheelchairs. Conclusions and future work are presented in Section 7. A preprint of the work presented here, especially Sections 4 and 5, can be found in the first author's PhD thesis [26].

## 2. Background

### 2.1. Target Population

In this manuscript, the term “individuals with hand-control impairments” refers to people who experience motor control deficits that reduce their ability to perform voluntary, precise hand or wrist actions needed to operate joystick-based or other manual wheelchair interfaces. Examples include, but are not limited to, action tremors (e.g., essential tremors or Parkinsonian tremors); involuntary spasms or spasticity (e.g., following spinal cord injury, stroke, or cerebral palsy); restricted range of motion or joint mobility; and reduced grip strength or coordination. These impairments can be heterogeneous (varying in frequency, amplitude, and temporal characteristics) and may be intermittent or progressive, which complicates the use of static controller settings and motivates adaptive solutions.

### 2.2. Limitations of Commercial Solutions

Commercial-powered wheelchair controllers provide useful, commonly applied options, such as dead zone (deadband) adjustment and reduction in sampling or control update frequency to mitigate involuntary micro-movements. While these options can help reduce the incidence of false or jittery commands, they have important trade-offs. Increasing a joystick dead zone reduces sensitivity to small intentional movements and can make fine manoeuvring difficult; additionally, lowering sampling frequency or applying heavy temporal smoothing reduces responsiveness and introduces latency, degrading the user's control over rapid voluntary commands. Moreover, these settings are typically static and require manual re-tuning by clinicians or technicians; they do not automatically adapt to intra-session or day-to-day changes in symptom severity. Taken together, these limitations show a gap between what current commercial settings can achieve and the needs of users with variable or mixed motor impairments.

### 2.3. Related Works

Using demonstrations to teach robots how to provide physical assistance is an interesting concept, offering a direct approach to creating custom assistive policies. It has been employed in rehabilitation training [27–29] to teach the execution of new movements [30,31] and to simplify the control of a robotic arm [32–34].

The concept of shared control has been explored in robotic wheelchairs. Goil et al. [35] proposed an algorithm for the collaborative control of an assistive semi-autonomous wheelchair. The objective was to blend human and robot controls using machine learning to learn task variability from demonstration examples. The results showed that the algorithm was effective in learning task variability from demonstration examples, allowing for safe traversal of challenging driving scenarios, such as doorway navigation. The conclusion was that the approach provides a promising solution for assisted wheelchair navigation.

Matsubara et al. [36] proposed a framework for intelligent navigational assistance in mobility aids. The objective was to estimate the user's intention sequentially to provide intelligent assistance. The results demonstrated that the framework could effectively

estimate the user's intention, leading to improved navigational assistance and achieving an accuracy of ~80% in near real-time. The conclusion was that the proposed framework enhances the effectiveness of intelligent navigational assistance in mobility aids.

Poon et al. [37] presented a framework for local driving assistance from demonstration for mobility aids. The objective was to provide active short-term navigation assistance by learning from demonstrations. The results indicated that the framework could provide effective local driving assistance, improving the mobility aid's responsiveness to user inputs. The conclusion was that learning from demonstrations enhances the effectiveness of local driving assistance in mobility aids.

Casado and Demiris [38] evaluated federated learning from demonstration for active assistance to smart wheelchair users. The objective was to develop a model for active assistance using federated learning from demonstrations. The results showed that the model could provide effective active assistance, improving the smart wheelchair's performance. The conclusion was that federated learning from demonstration is a viable approach for developing active assistance models for smart wheelchairs.

Bozorgi and Ngo [39] presented a roadmap from shared autonomy to full autonomy for human-in-the-loop mobile robot navigation systems. The objective was to incorporate joint perception and action to enhance the practicality and applicability of mobile robot navigation. The results indicated that the proposed framework could enhance the practicality and applicability of mobile robot navigation systems. The conclusion was that incorporating joint perception and action is essential for advancing mobile robot navigation systems towards full autonomy.

All these works, in common, attempted to use expert driving demonstrations to generate policies that assist with navigation. However, suppose the assisted person operates differently from the expert, due to a hand control impairment, for example. In such a case, it can be infeasible to match the expert and non-expert inputs, thus reducing the usefulness of the assistive policy.

To circumvent this, LAD was proposed in Soh and Demiris [20]: "LAD augments LbD by focusing on the assistive element ... Instead of deriving a policy for 'how-to-drive', it extracts a policy for 'how-to-help-a-user-drive'." LAD was further developed in Soh and Demiris [40], where subjects exposed to a simulated hand control impairment controlled a robotic wheelchair with a joystick. An assistant observed the scene and intervened as needed, providing demonstrations of how to help that particular driver. The demonstrations were recorded and used to train a learning model, which should, subsequently, be capable of autonomously assisting the driver.

The concept was also explored in Kucukyilmaz and Demiris [41] and Kucukyilmaz and Demiris [42]. Still, these seminal works did not demonstrate or fully study some of the key features that make LAD an attractive approach: Can LAD models trained on a training course be used outside of it? Can these models significantly and consistently help drivers? Can they be used with generic disabilities? Are they indeed offering personalised assistance?

The question of generalisation to unseen environments was better explored in Schettino and Demiris [23] since the assistive performance was tested in a separate location from where the training data were recorded. Although promising results were observed, experiments were carried out considering only a single disability type and assistant-driver pair. To fully validate LAD, more statistically robust experiments are needed. This, however, is an expensive and time-consuming endeavour, and before taking this road, it is essential to understand better the features and limitations of LAD. One way to achieve this is through simulations, which can accelerate research development and eliminate confounding variables from experimentation, thereby enhancing the analysis of results.

Many research groups have taken this approach to accompany the study of assistive robotics, developing both domain-specific [43–45] and general-purpose [46,47] simulators. The literature also shows extensive usage of simulators in research with both regular [48–50] and robotic wheelchairs [51–53]. But their purpose is to be used by humans without simulating the driver's input. In particular, no open-source simulator was found that can autonomously mimic the triadic interaction between a driver, their wheelchair, and a remote assistant.

For example, in Morère et al. [51], a robotic wheelchair simulator was used to test the effectiveness of combining haptic feedback and obstacle avoidance. In Devigne et al. [52], a simulator was used to test a simplified obstacle avoidance algorithm based on the readings of low-cost sonar sensors. A similar setup was used in Di Gironimo et al. [54], where the usability of different joysticks for controlling a robotic arm mounted on a powered wheelchair was tested. All these simulators were, in common, developed for use by humans. This is useful for allowing new ideas to be tested more quickly and safely.

However, sometimes it is also constructive to simulate the human. For example, in Reddy et al. [43], a model-free reinforcement learning algorithm for shared control was explored. Using a simulated navigation environment and a simulated impaired driver allowed the authors to extensively test their method and make the necessary improvements before moving on to user studies. More recently, a general-purpose simulator for assistive robotics was developed [46]. It allows for the simulation of six different manipulation tasks to assist with activities of daily living, including scratching, bed bathing, drinking water, feeding, dressing, and arm manipulation.

Additionally, four different types of robots can be employed. The simulator is primarily designed for reinforcement learning applications, and the human is typically simulated in a passive pose, i.e., without active movement, but still constrained by realistic joint movements. In Erickson et al. [47], the simulator was taken one step further, allowing people to control the virtual human by using virtual reality headsets and controllers. This enabled the researchers to smoothly transition from simulated humans to real people using a virtual environment, and finally to test their assistive robots on a physical platform safely.

#### 2.4. Summary

In summary, previous works have shown that demonstrations can be successfully employed in rehabilitation and movement training, robotic arm control, and wheelchair navigation. However, approaches based on expert demonstrations often fail to accommodate the variability introduced by hand control impairments, limiting their applicability for personalised assistance. LAD has emerged as a promising alternative. LAD is designed to learn a personalised assistive policy from user demonstrations. This allows the system to generate a tailored solution that understands and adapts to the nuanced complexities of a user's unique motor control limitations, offering a significant advantage over the manual configuration of commercial systems or one-size-fits-all algorithms.

Still, existing studies remain constrained to a few disability types, small-scale experiments, and controlled environments, leaving open questions regarding their robustness, generalisation, and ability to provide truly personalised support. Furthermore, although simulators have been extensively used in assistive robotics, no open-source platform currently exists that autonomously reproduces the whole triadic interactions that occur between drivers, assistants, and wheelchairs. These gaps motivate the present work, which develops a dedicated simulator, explores improved data collection and learning strategies, and validates the key motivations behind LAD.

### 3. Simulator Design and Implementation

A robotic wheelchair simulator provides a controlled platform for systematic experimentation with software, control algorithms, and learning techniques, enabling rapid iteration while reducing both development time and experimental costs. It also facilitates the generation of synthetic datasets for training learning algorithms and the evaluation of navigational strategies in scenarios that would be unsafe or impractical with real users, such as extreme perturbations or rare collision events.

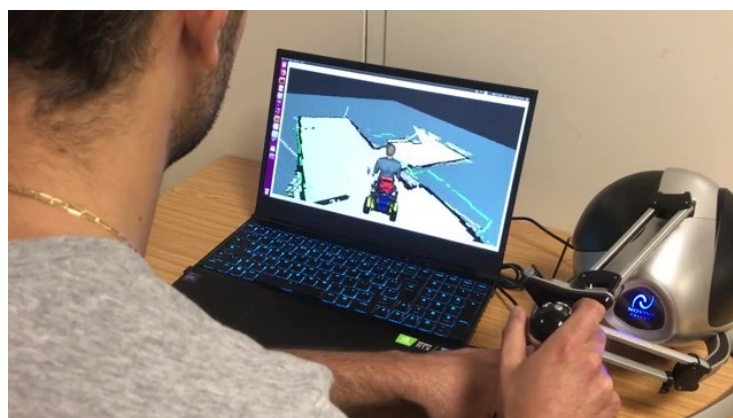
Importantly, this simulator is not intended to substitute experiments with human drivers, but rather to serve as an initial validation and risk mitigation tool. By simulating user behaviours (including artificially induced noisy or distorted control inputs), the platform allows researchers to detect failure modes in assistive policies before deployment. Policies that exhibit instability, unsafe trajectories, or poor responsiveness under these simulated conditions can thus be refined or discarded, ensuring that subsequent testing with real users is both safer and more efficient. In this way, the simulator functions as a critical intermediary step, bridging theoretical development and practical, user-centred evaluation.

#### 3.1. Wheelchair Navigation

The simulator is built on top of Gazebo (<http://gazebo.org>, accessed on 24 September 2025), which handles both the Robot Operating System (ROS) interface and the physics simulation [55]. The wheelchair model is based on a real robotic wheelchair used in the Personal Robotics Lab (PRL) (<http://www.imperial.ac.uk/personal-robotics/>, accessed on 24 September 2025), which uses velocity command inputs to actuate two back wheels with differential drive and also has two caster wheels at the front for stabilisation.

The number and type of sensors available also mimic the PRL wheelchair: three planar laser scanners covering 360 degrees around the robot and an Inertial Measurement Unit (IMU). The characteristics of the sensors and the noise present in their readings are modelled based on the nominal values defined by the manufacturers. The relative position of the sensors was calibrated using an optical motion capture system. The model graphics were created by scanning the real wheelchair using 3D technology. To reduce the computational load, the collision properties were simplified to a cube covering the entire model. The ROS interface for the simulated wheelchair is identical to the real one, thus allowing for a seamless transition when testing assistive navigation algorithms.

A person can control the simulated navigation using a regular joystick, as shown in Figures 1 and 2 (a joystick with force feedback capability can also be used, but the force feedback mechanism was not used for this work):



**Figure 2.** Dedicated visualisation window for a remote assistant (who can interact in the same simulation as the driver).

To increase the variability of data that can be collected in simulation and to better test the robustness of assistive models, a few fictitious worlds were created. These worlds were designed as abstract generalisations of common indoor environments, as depicted in Figure 3:



**Figure 3.** Custom, simulated environments were created for collecting synthetic data, but other general-purpose worlds (<https://github.com/aws-robotics>, accessed on 24 September 2025) can also be easily incorporated into the simulator.

It is noted that third-party users of the simulator can easily expand this initial set of wheelchair and world models to accommodate specific needs.

### 3.2. Hand Control Impairments

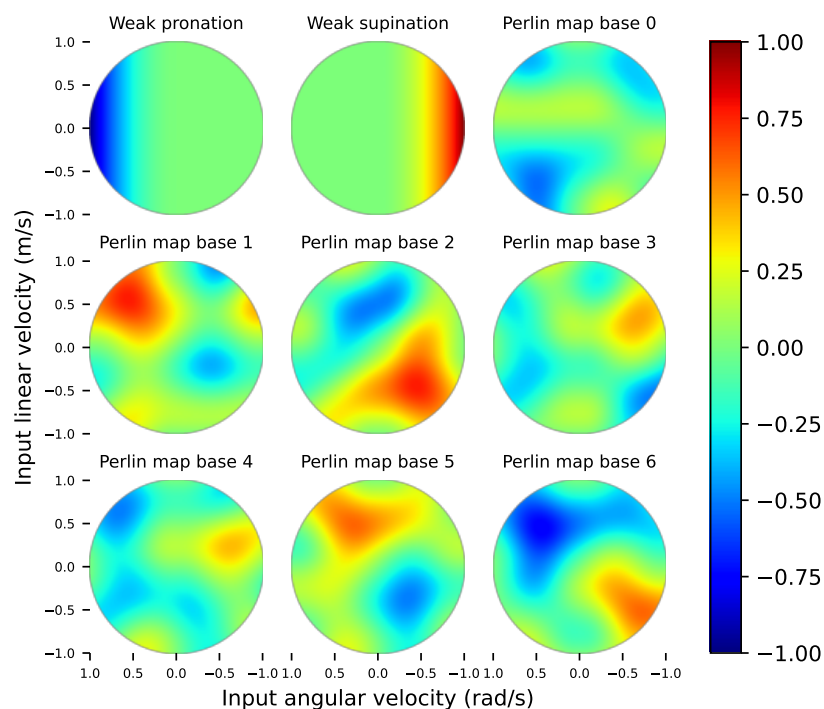
The focus here was on exploring the use of LAD to assist drivers who struggle with hand control impairments. These impairments hinder dexterous joystick manipulation, occasionally rendering wheelchair navigation impractical. The simulator allows people with such impairments to safely attempt navigation with a virtual wheelchair before moving on to tests with a physical system.

From a developer's perspective, however, it might be helpful to simulate the hand control impairment. An example use case would be to continuously test the efficacy of an assistive policy without requiring the impaired person to drive the wheelchair continuously. In this case, the developer could test drive the wheelchair with the simulated disability over-imposed and assess the impact of the assistive policy. Another application would be to quickly test whether an assistive policy developed for one type of impairment could be helpful for other types.

In this sense, simulated hand control impairments were also included in the software, where the mapping drivers' standard control signals simulate them to noisy and distorted ones. The distortions are created using 2D maps, where linear and angular velocity commands are used as input, and the output is a distortion value that can be superimposed, making navigation harder. For example, if a driver suffers from weak arm supination movements (a common post-stroke symptom) and thus struggles with right hand turns, a map like Figure 4-left could be employed to roughly simulate this impairment. In this case, whenever the driver tries to turn right (negative angular velocity), a positive value is added to their command, limiting their maximum turn rate.

Rarely, however, is an impairment clean. Usually, the movement restraints are more complex and vary from person to person. To generate more natural simulations of impairments, Perlin noise was employed [56]. This is an algorithm traditionally used in the film industry to compose natural-looking, computer-generated textures (e.g., fire and smoke). These textures are multidimensional, pseudo-random noise maps, where the algorithm's

parameters control the level and interplay of details. Examples of these maps are also shown in Figure 4:



**Figure 4.** Examples of distortion maps that can be imposed over a driver’s input commands to simulate a disability. Colour values indicate what was added to the driver’s desired angular (or linear) velocity. Weak pronation: a synthetic example, where the driver would have difficulty in performing left hand turns. More natural distortion maps can be procedurally generated using Perlin noise. Perlin Map Base 5: a driver that tends to overshoot when doing forward-left or backward-right turns (used in the experiment in Section 5.3). Perlin Map Base 6: a driver that pulls to the right when attempting to go straight (used in the experiment in Section 5.4).

Because Perlin noise is only pseudo-random, the same output can always be expected for a given input (impairments that vary over time, although common, are not yet supported). And if a different impairment must be simulated, the distortion map can be deterministically changed simply using a different seed (known as base in the Perlin algorithm). This allows different disabilities to be procedurally generated and easily switched between.

In addition to the distortion mapping, white noise can also be added to the driver’s input, thus emulating both the deterministic and random components of a generic impairment. The Perlin base and the relative levels of deterministic and random distortions can be easily adjusted, even during simulation.

It is noted that, currently, this simulator does not aim to mimic any specific real-world hand control impairment. Instead, it offers the simulation of generic disability types, which can be used to test the robustness of an assistive policy.

### 3.3. Triadic Interactions

For some individuals, hand control impairments may not be debilitating enough to prevent wheelchair navigation entirely, but they can still pose challenges in specific situations. For example, a driver might navigate open areas with ease but struggle with dexterity when passing through narrow doorways. In such cases, a remote assistant may be called upon to help via a teleoperation platform [24]. This assistance can be provided

either sporadically or as part of wheelchair training. Due to its relevance, this interaction feature is incorporated into the simulator.

An assistant can join the same simulation session as the driver and provide alternative driving commands using a separate joystick. Since the assistant is assumed to be operating from a remote location, a dedicated visualisation interface is provided, allowing them to observe the driving scene solely through the wheelchair’s onboard sensors, as shown in Figure 2.

Because both the driver and assistant can generate control commands concurrently, a shared control strategy is required. This strategy includes an “ask-for-help” mechanism, where the driver decides when to relinquish control of the wheelchair by pressing a button on their joystick [23]. This technique aims to enhance user satisfaction by granting the driver as much autonomy as desired [18,57].

### 3.4. Repeatable Interactions

A challenging aspect of running experiments with robotic wheelchairs is the behavioural variability that drivers may exhibit over short periods due to factors such as fatigue and learning effects. These variations act as experimental noise, potentially obscuring trends in the data that could distinguish effective from ineffective assistive models. The typical solution to this issue is to conduct large-scale human trials, where confounding effects tend to cancel each other out. However, before committing to such studies, it can be beneficial to test assistive models more systematically. This is possible through the use of simulated drivers, which behave consistently across trials. Our simulator features this capability through a fundamental autonomous driver that can follow predefined waypoints while avoiding obstacles.

To implement this functionality, localisation, path planning, and local control are combined to simulate a driver without hand control impairments. The process begins by creating 2D maps of the environment using Simultaneous Localisation And Mapping (SLAM). The localisation and path planning modules then use these maps to guide the wheelchair to predefined target locations, while the local control module ensures obstacle avoidance. All algorithm parameters are tuned specifically for the modelled wheelchair to guarantee smooth and realistic navigation. Once the unimpaired autonomous driver is in place, an impaired version can be simulated by applying the method described in Section 3.2, which involves adding distortion and random noise to the control signal.

The assistant can be simulated simply by using the original (pre-distortion) control signals. Since local control is used to follow designated trajectories, the assistant’s commands remain effective for aiding navigation, even when the impaired driver deviates from the planned path.

Figure 5 illustrates the various modes of operation available in the simulator:

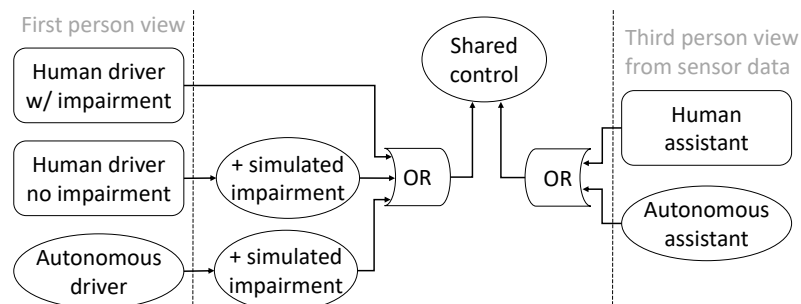
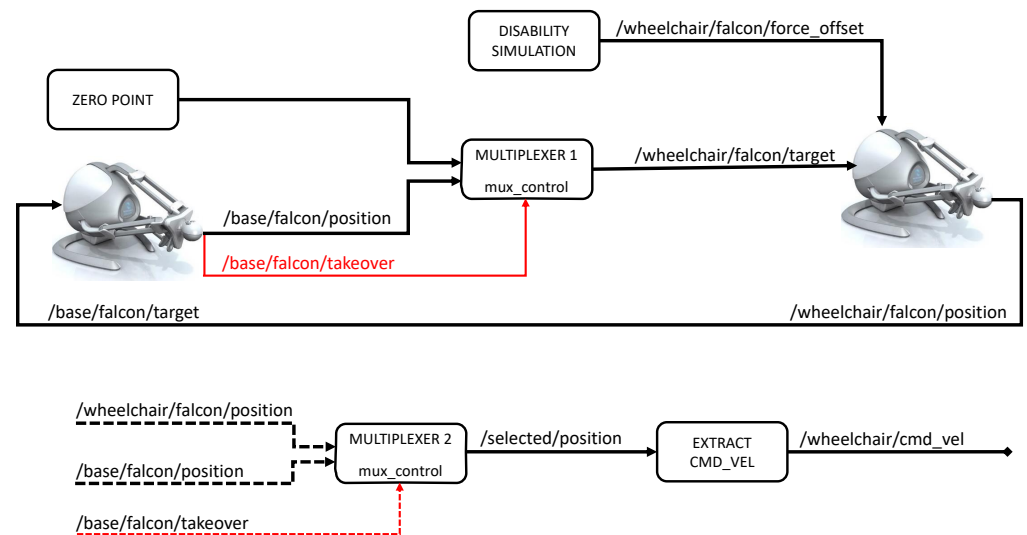


Figure 5. Possible modes of operation when using the simulator.

For instance, a researcher could have a human driver with a real hand control impairment operate the virtual wheelchair independently to better understand the functional

limitations imposed by the disability. Alternatively, a researcher could drive the virtual wheelchair themselves under simulated impairment conditions to evaluate an autonomous assistive policy. Another option is to allow both an autonomous driver and an autonomous assistant to share control of the virtual robot. This final setup was used to simulate the triadic interaction required for LAD, enabling repeatable and controlled experiments.

To provide more information of the developed software, Figure 6 illustrates the shared control scheme:



**Figure 6.** Illustration of the shared control architecture, which integrates driver input, disability simulation, and remote assistance through multiplexers, enabling teleoperation, autonomous takeover, and mixed control strategies.

The architecture is based on a shared control scheme, in which multiplexers manage the signals coming from the driver, the simulated disabilities, and the remote assistant. This modular design enables the simulation of various assistance strategies, including direct teleoperation, autonomous takeover, and mixed control modes. The software is fully implemented in ROS and designed to be extensible, allowing for the easy integration of new modules and interaction modes. For reproducibility and to support future research, the complete source codes are open source ([https://github.com/vbschettino/wheelchair\\_assist\\_simulator](https://github.com/vbschettino/wheelchair_assist_simulator), accessed on 24 September 2025, <https://github.com/vbschettino/lad>, accessed on 24 September 2025).

## 4. Learning to Assist

In a LAD context, triadic interactions are used to collect training data. This data is then used to fit a learning model that automates the assistance previously demonstrated by the human assistant. In this section, the neural network architecture developed for learning from demonstration is described, along with the data preprocessing techniques adopted.

### 4.1. Data Preprocessing

During training, the laser scan readings and the command velocities from both the driver and the assistant are recorded (during testing, the wheelchair's position on the map and the planned paths are also recorded, but only for performance evaluation—see Section 5.1). These data sources operate asynchronously and at different frequencies, where a unified sampling rate of 10 Hz (matching the slowest source (the laser scanners)) is taken and uses a nearest sample strategy to synchronise the data streams. During inference, incoming data is similarly buffered and synchronised to the same rate.

A common issue with laser scan data, even in simulation, is the presence of invalid readings when values fall outside the scanner's nominal minimum or maximum range. Additionally, the laser data is high-dimensional, with up to 720 channels used to cover a full 360-degree scan. To address both problems, a "valid average" approach is needed. Neighbouring channels are grouped, and the average of valid values within each group is computed. This not only reduces the dimensionality of the laser data, but also simplifies the machine learning task. It was found that reducing the number of channels to 72 provided a good balance between resolution and learnability. If all channels in a group are invalid, the average is replaced with the scanner's maximum range.

Next, the values are inverted and normalised to the [0, 1] range (so that nearby obstacles yield values close to 1, while distant ones result in values near 0). Although this constitutes a non-linear transformation, it was observed that it facilitates the learning process in converging on better solutions.

The command velocities from both the driver and the assistant are derived from their joystick inputs. These signals are low-dimensional, naturally bounded, and free of invalid values, so they are used directly for training without further preprocessing. Since the simulated impairment affects only the angular component of the driver's velocity commands, the target signal is defined solely as the angular velocity commands issued by the remote assistant.

#### 4.2. Model Design

In this application, the learning model takes as input the command velocities from the driver and the laser scan readings from the wheelchair, and it uses the assistant's command velocity as the target signal. In an earlier work (Schettino and Demiris [23]), an autoencoder was used to reduce the dimensionality of the laser scan input, and a Gaussian process model was used to predict the assistive control signal. In the present study, after evaluating several alternatives, a fully neural network-based approach was chosen, as it yielded the best performance in preliminary tests [26]. This section provides a concise description of the model architecture and the procedure adopted for hyperparameter optimisation.

A schematic representation of the model architecture is shown in Figure 7:

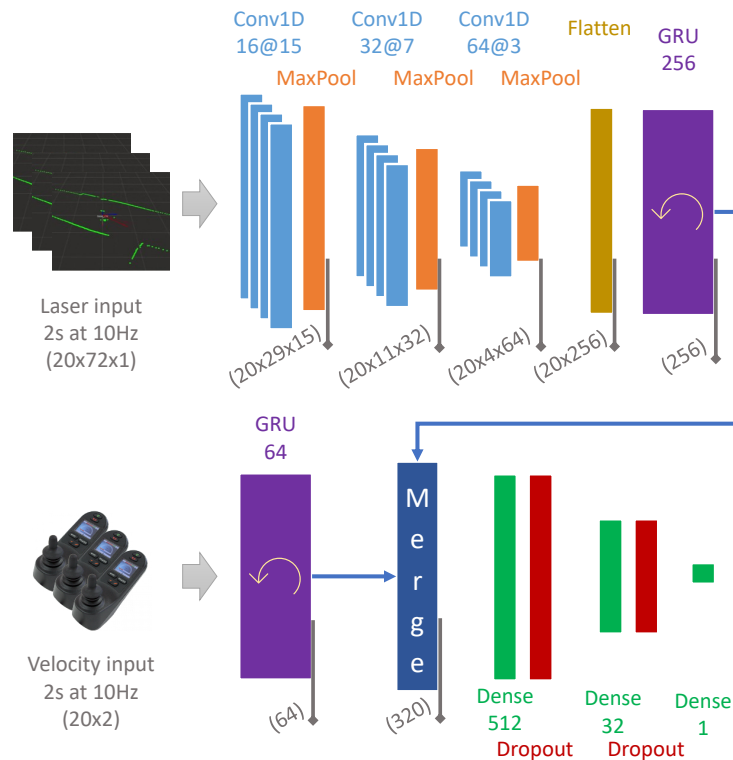
After preprocessing, the laser scan readings are processed through a series of 1D convolutional layers interleaved with MaxPooling layers. Although this approach requires more memory, it was observed that using a relatively large kernel size (15) in the first convolutional layer significantly improved performance. This enables the model to detect coarse environmental features, such as doorways, corners, and extended walls, in corridors. As the signal progresses deeper into the network, the kernel size is reduced and the number of filters is increased, allowing the model to extract finer-grained spatial details.

Following the convolution and pooling stages, the laser data is compressed and flattened before being passed through a recurrent layer, which captures temporal patterns as the wheelchair moves. The recurrent layer operates on a time window of 20 samples, equivalent to 2 s at the 10 Hz sampling rate.

The driver's command inputs, consisting of linear and angular velocity components, follow a separate recurrent pathway, also with a 2 s window. After temporal processing, both input pathways are merged and passed through a sequence of fully connected layers interleaved with dropout layers to produce the predicted assistive signal.

The model was implemented and trained using TensorFlow. The Adam optimiser is used for backpropagation, with a batch size of 128 samples and Mean Squared Error (MSE) as the loss function (penalising larger deviations more strongly than absolute error). ReLU is used as the activation function in the convolutional and fully connected layers. In contrast,

the recurrent layers use hyperbolic tangent as the main activation and as a sigmoid function for the recurrent step. Early stopping is applied to mitigate overfitting.



**Figure 7.** Architecture of the neural network model developed using simulation data and hyperparameter optimisation.

### Hyperparameter Optimisation

A multi-stage process was employed to arrive at the final network architecture. Initially, a heuristic approach was taken, selecting hyperparameters commonly found in related literature. This was followed by manual exploration, where the effects of substantial changes in hyperparameters to identify general performance trends were evaluated. The objective function employed was validation loss, specifically the Mean Squared Error (MSE) between the predicted and true outputs on the validation set. Once plausible ranges were identified, it was used with automated methods for fine tuning.

For instance, multiple training runs were performed on the same base architecture, varying the number of dense layers (from one to five), to assess the impact on performance. This helped eliminate suboptimal hyperparameter choices and to reduce the search space, and it was guided by general heuristics for neural network design.

Automatic optimisation was then applied. Due to the high computational cost and the large number of possible hyperparameter combinations, a grid search was not feasible. Instead, the Hyperband algorithm was used, accelerating the search process by terminating unpromising runs early and reallocating resources to better-performing configurations [58]. However, the goal was not exhaustive optimisation but to identify trends in hyperparameter combinations that improve model performance.

Once the search space was sufficiently narrowed, Bayesian optimisation was used to select the final hyperparameter values [59]. This method leverages Bayesian inference to balance the exploration and exploitation within the search space, enabling more efficient convergence to optimal configurations than random or exhaustive search strategies.

Taking the optimisation process into account, the final set of hyperparameters were obtained, as reported in Table 1:

**Table 1.** Hyperparameters selected for automatic optimisation.

Name	Type	Range/Values	Description
conv_blocks	Int	1–3	Number of convolutional blocks
conv_block_style	Choice	1, 2, 3, 4	Conv block type: 1 = Conv, 2 = Conv-Conv, 3 = Conv-Pool, 4 = Conv-Conv-Pool
conv_filters1	Int	16–48 (step 16)	Filters in 1st convolutional block
conv_filters2	Int	48–96 (step 16)	Filters in 2nd block (if conv_blocks $\geq$ 2)
conv_filters3	Int	64–128 (step 32)	Filters in 3rd block (if conv_blocks = 3)
conv_ksize1	Int	7–15 (step 2)	Kernel size in 1st block
conv_ksize2	Int	7–9 (step 2)	Kernel size in 2nd block (if conv_blocks $\geq$ 2)
conv_ksize3	Int	3–5 (step 2)	Kernel size in 3rd block (if conv_blocks = 3)
conv_batchnorm	Boolean	True/False	BatchNorm after conv block
conv_drop	Boolean	True/False	Dropout after conv block
conv_drop_rate	Float	0.1–0.3 (step 0.1)	Dropout rate (if conv_drop is True)
rnn_type	Choice	SimpleRNN, LSTM, GRU	Type of recurrent layer
rnn_layers	Int	1–2	Number of recurrent layers
rnn_units_scan	Int	16–512 (step 248)	RNN units for scan input
rnn_units_vel	Int	48–90 (step 16)	RNN units for velocity input
rnn_batchnorm	Boolean	True/False	BatchNorm after RNN layers
merge_batchnorm	Boolean	True/False	BatchNorm after merging scan/vel paths
dense_layers	Int	1–2	Number of dense layers before output
dense_units1	Int	256–768 (step 256)	Units in first dense layer
dense_units2	Int	8–32 (step 8)	Units in second dense layer (if dense_layers = 2)
dense_batchnorm	Boolean	True/False	BatchNorm after dense layer
dense_drop	Boolean	True/False	Dropout after dense layer
dense_drop_rate	Float	0.1–0.3 (step 0.1)	Dropout rate after dense layer
learning_rate	Float	0.001–0.005 (step 0.001)	Learning rate for Adam optimiser

## 5. Investigating Features of LAD

In this section, the data collected from simulated drivers and assistants are used to investigate various aspects of LAD, right after introducing the evaluation metrics.

### 5.1. Metrics

The MSE was used as the main metric to compare the predictive performance of different models. The error was computed on a test course by comparing the driving commands generated by the autonomous assistant to the corresponding model predictions. However, a lower predictive loss does not necessarily imply better assistive performance. For instance, a model that slightly mispredicts during straight-line driving in open areas but performs accurately during narrow doorway navigation is arguably preferable to one that performs accurately during straight-line driving in open areas but mispredicts during narrow doorway navigation. Yet, such nuance may not be reflected in the total predictive error, especially when the training dataset is biased towards straight-line navigation (a common scenario).

For this reason, predictive and assistive performance were distinguished: the former is helpful for rapid model comparison, while the latter is the primary metric of interest from the driver’s perspective.

To evaluate assistive performance more comprehensively, it was necessary to consider multiple dimensions. An assistive model that increases navigation speed but also causes frequent collisions may not be beneficial to the user. Therefore, four distinct metrics were employed:

- Time to complete a lap: the total time required to complete one lap of a dedicated test course;

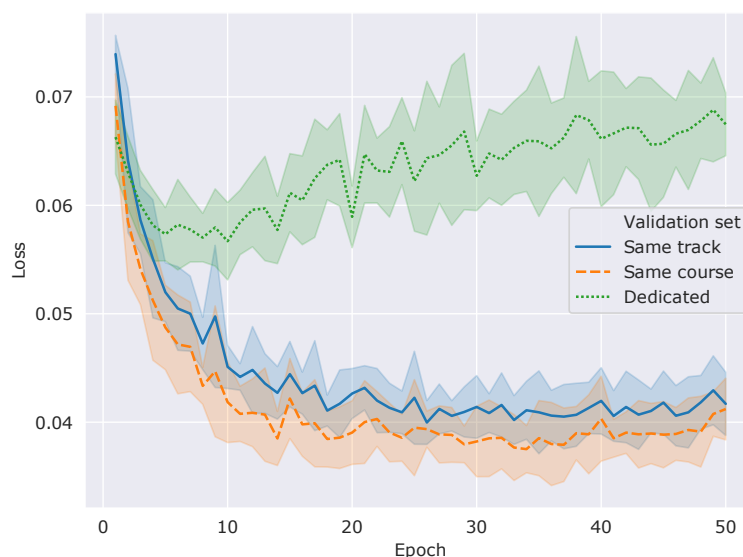
- Average distance from the planned path: the absolute distance between the wheelchair's current position and the closest point on the planned path, sampled every 0.1 s and averaged over the lap;
- Fraction of time spent clearing collisions: the ratio of time spent in autonomous collision recovery behaviour to the total lap time;
- Number of instructor interventions: If the autonomous driver remains stuck for more than 10 s, a human instructor supervising the test intervenes to guide the wheelchair back to the planned path manually. This metric counts the number of such interventions per lap.

Predictive performance results were averaged over 10 training runs with different random seeds. Assistive performance results were averaged over five laps on the test course.

## 5.2. Generalisation

This section explores the issue of model generalisation to unseen environments. Although this capability has been demonstrated in the related field of autonomous cars [60], assistive robotics presents a distinct challenge, as significantly less training data is typically available. This limitation is especially true for LAD, where data collection must be performed on a per-user basis and requires the full attention of an expert assistant.

It started by contrasting the performance testing procedures used in previous work with our approach. First, data on a training obstacle course were collected, and then three different validation sets were built, all of equal size. The first set consisted of a second lap on the same training course, following the same trajectory (similar to the approach in [40]). The second set consisted of a run on the same course, but following a different trajectory, as seen in Kucukyilmaz and Demiris [42]. The third validation set was recorded on a dedicated obstacle course that was physically distinct from the one used for training. Then, it fit the LAD model to the training data and, as training progresses, evaluated the predictive loss on all three validation sets. The results are shown in Figure 8:



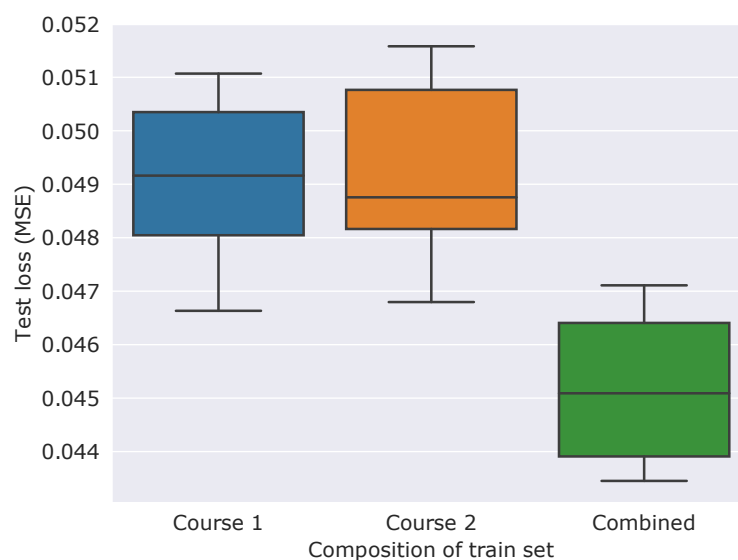
**Figure 8.** Impact of using a separate course to collect data for a validation set. A single model was trained, and at each epoch, its performance was tested against three distinct validation sets. Only the set recorded on a separate course (green line) indicated overfitting. Without this, previous approaches (blue and orange lines) would be unable to apply early stopping to improve generalisation.

As is shown, the first two validation sets (blue and orange) indicated good performance, even though the model was actually overfitted. In realistic scenarios (where the

wheelchair must operate in previously unseen environments), this assistive model would likely perform poorly. This overfitting occurred because, as training progressed, the model increasingly relied on spatial features (from the laser scanners) specific to the training environment. At inference time, when spatial features differ, the model fails to produce appropriate assistive signals.

This is not merely an issue of measuring performance. The use of a suitable validation set is crucial for proper model regularisation via early stopping, particularly when training deep neural networks. Therefore, this experiment demonstrates how a simple adjustment in the data collection procedure can significantly improve predictive performance.

Next, it assessed how collecting training data in multiple environments affects generalisation. Figure 9 presents the predictive performance of our LAD model when trained on three different datasets:

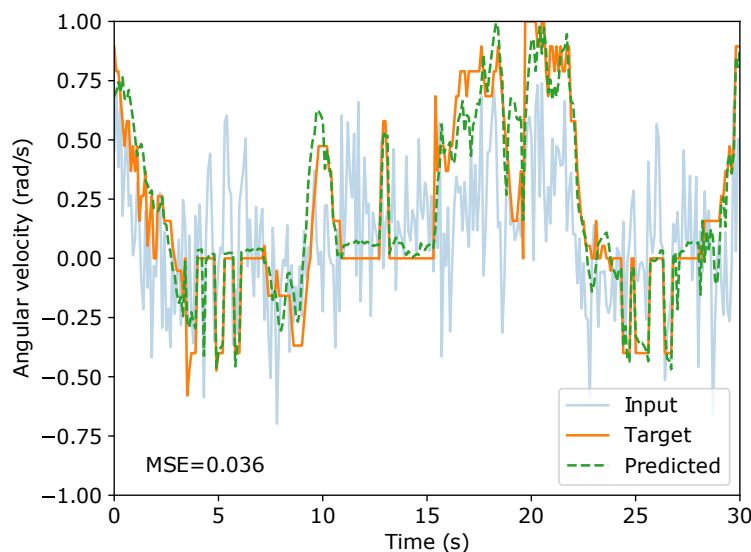


**Figure 9.** Impact of using multiple courses for training data collection. The ‘Combined’ dataset includes assistive demonstrations from physically distinct courses, as opposed to repeated laps on the same course. This approach improves performance when predicting assistance in an unseen environment.

All datasets were of the same size, but the first two (blue and orange) were derived from a single obstacle course. This strategy, employed in previous LAD studies, again leads to overfitting. Instead, following the principle of improved data quality [61], we set it to combine the data from both courses to encourage the model to learn more generic environmental representations. As expected, this approach yielded better results when tested in a novel obstacle course. Once again, a simple modification in the data collection process can enhance generalisation.

Finally, Figure 10 presents a plot of the angular velocity commands recorded while the simulated driver navigated a dedicated test course not seen during training.

The plot compares the driver’s control signal (input), the assistant’s control signal (target), and the model’s prediction. The assistive model used here was trained with the improved data collection and preprocessing strategies described above. Note how the predicted velocity correctly tracked the target signal trends while filtering out most of the noise and distortion from the input.



**Figure 10.** Example plot showing the input, target, and predicted velocity signals on a dedicated test course. The predicted signal successfully followed the target’s trends while filtering the noise and distortion from the driver’s input.

### 5.3. Assistive Performance

Predictive performance helps compare different model architectures and training procedures, especially when the goal is to enhance generalisation. However, for the driver, the important aspect is how and to what extent the final model can assist with navigation. Hence, this section discusses different facets of LAD’s assistive benefits.

Moreover, five distinct simulation worlds were employed (Figure 3), with three used for training, one for validation, and one for testing. The proposed methodology facilitates a straightforward transition between environments, as new scenarios can be seamlessly incorporated by following the same procedure described in this work.

In this set of experiments, Base 5 from the Perlin map (Figure 4) was employed to model the driver’s impairment, ensuring that the assistance provided by the LAD model was evaluated under consistent and repeatable distortion conditions.

Then, as shown in Table 2, the driving performances for three different simulated drivers were recorded: one without a hand control disability, one with it, and one with a driver with the same disability but autonomously aided by the LAD model. Driving performance was then reported using the metrics described in Section 5.1:

**Table 2.** Assistive performance results (reported as average (standard deviation)) when comparing a driver without hand control disabilities, with a disability, and with the same disability but aided by a custom LAD model. The recovery (0–100%) shows how much, on average, the performance gap between ‘No disability’ and ‘Disability’ was recovered by using LAD. LAD led to a positive impact across all metrics.

	No Disability	Disability	LAD	Recovery
Time to complete a lap (s)	179.3 (4.8)	325.7 (23.8)	247.9 (11.6)	53.2%
Avg. dist. from planned path (cm)	8.7 (0.3)	23.0 (1.6)	16.2 (1.1)	47.5%
Frac. of time clearing collisions (%)	2.0 (0.8)	23.3 (1.4)	3.6 (0.8)	92.4%
Num. of instructor interventions	0.0 (0.0)	3.2 (1.2)	0.2 (0.4)	93.8%

As expected, the imposition of impairment had a significant negative impact on all measures. The disability makes driving more erroneous, which increases the average distance from the planned path. It can also lead to under or overshooting turns, which

increases the chance of collision with obstacles. In turn, this leads to a surge in the number of collision clearance manoeuvres required and the frequency at which the wheelchair becomes stuck. All of these contribute to an average increment of 82% in the time it takes for the simulated driver to complete a lap on the test course.

However, providing this driver with autonomous assistance from a LAD model improved all (measured) aspects of navigation. In two of the metrics, the number of human interventions needed and the relative amount of time spent clearing collisions, the driver was capable of nearly recovering their original performance (last column of Table 2). While the performance recovery was not as pronounced for the other two metrics, the results were still significant. To put it in context, these results suggest that a wheelchair equipped with LAD autonomous assistance would enable this driver to navigate 24% faster, while also having a far smaller chance of collisions.

#### 5.4. Robustness

The primary motivation for using LAD is that it offers a straightforward approach to creating custom assistive models that can aid individuals with various disabilities. Hence, testing if this promise holds is paramount, but it has not been performed before. Previous work in this field [20,23,40] carried out experiments with a single disability type. But with the approach proposed in Section 3, it can be tested whether LAD is indeed robust and applicable to more general types of impairments.

We began by repeating the experiment of Section 5.3 for a second driver, who had a different simulated disability (distortion map at the right of Figure 4). The results are shown in Table 3:

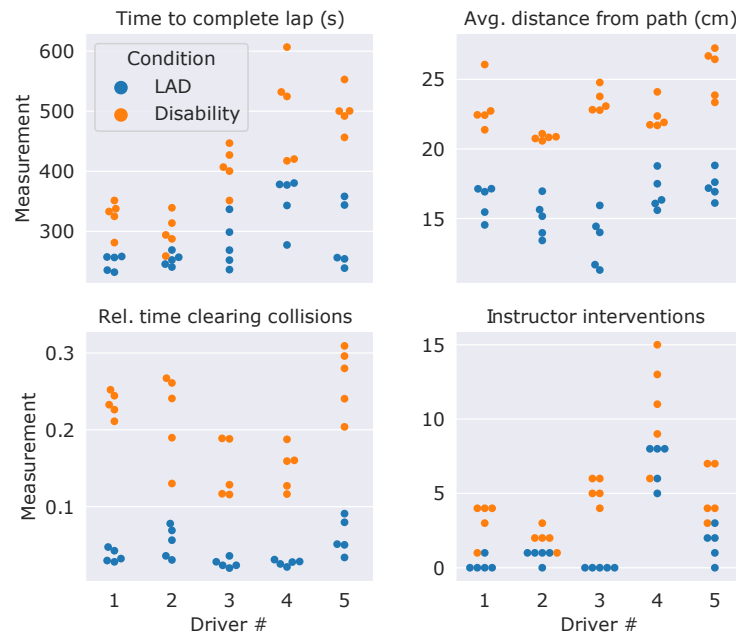
**Table 3.** Assistive performance results for a driver with a different disability. LAD still helped the driver without requiring any hyperparameter tuning. The experiment was repeated for more disabilities, as shown in Figure 11.

	No Disability	Disability	LAD	Recovery
Time to complete a lap (s)	179.3 (4.8)	298.7 (26.9)	252.8 (9.8)	38.4%
Avg. dist. from planned path (cm)	8.7 (0.3)	20.8 (0.2)	15.0 (1.3)	48.0%
Frac. of time clearing collisions (%)	2.0 (0.8)	21.8 (5.2)	5.4 (1.8)	82.8%
Num. of instructor interventions	0.0 (0.0)	2.0 (0.6)	0.8 (0.4)	60.0%

As can be seen, the LAD model was again capable of significantly helping across all metrics, even though a different driver was operating and the same model architecture and hyperparameters were kept. However, a smaller part of the gap to the original performance was recovered in this case. This highlights a limitation of LAD: as a data-centric approach, the same level of improvement cannot be expected for all types of disabilities. The experiment was conducted with another three drivers.

The results are summarised in Figure 11, where it is shown that, in all cases, LAD provided useful assistance, indicating the robustness of the approach.

Then, in the experiments summarised above, five different disabilities were considered, corresponding to Perlin Map Bases 0, 2, 5, 6, and 99. The results demonstrate that the proposed LAD methodology performs consistently well across multiple scenarios and impairment types. Importantly, by following the methodology presented here, additional disabilities and environments can be easily integrated into the evaluation pipeline, making the framework extensible and adaptable to future studies.



**Figure 11.** Assistive performance metrics for drivers with five different hand control disabilities, compared with and without LAD assistance. Each dot represents a lap on a test course. LAD consistently improved performance for all tested metrics and disabilities, indicating the robustness of the approach.

5.5. Personalisation

This section examines whether LAD models can provide personalised assistance, that is, whether they are tailored to the unique needs of each user and their specific impairments (or if they are merely offering generic help). Personalised assistance could potentially be replaced by simpler alternatives, such as low-pass filtering of driver inputs or off-the-shelf obstacle avoidance algorithms [62].

To investigate this, both the simulated drivers from Tables 2 and 3 were used to re-run the test course with LAD assistance. However, this time, each driver was assisted by the model trained on the other driver’s data, where Driver 1 received assistance from the model trained for Driver 2, and vice versa. Table 4 presents the resulting performance, showing the percentage improvement (or deterioration) relative to each driver’s performance without assistance. Negative values indicate a drop in performance.

**Table 4.** Personalisation experiment. The improvements achieved when using a custom model versus a generic one were compared. The results show that LAD generates personalised assistive solutions.

Time to Complete a Lap (s)			Avg. Dist. from Planned Path (cm)		
	Model 1	Model 2		Model 1	Model 2
Driver 1	<b>23.9%</b>	−35.5%	Driver 1	<b>29.4%</b>	−47.1%
Driver 2	−31.2%	<b>15.3%</b>	Driver 2	−42.7%	<b>27.8%</b>
Frac. of time clearing collisions (%)			Num. of instructor interventions		
	Model 1	Model 2		Model 1	Model 2
Driver 1	<b>84.6</b>	58.9	Driver 1	<b>93.8%</b>	−31.2%
Driver 2	42.3	<b>75.2</b>	Driver 2	−10.0%	<b>60.0%</b>

The results clearly show that using the incorrect LAD model leads to a significant deterioration in navigation performance. In many cases, the drivers performed worse than they did without assistance. This supports the claim that the models are indeed personalised as they are optimised for the specific driving patterns and limitations of

individual users. These results also indicate that simple, generic solutions can be insufficient for individuals with hand control impairments and reinforce the fundamental motivation for using LAD to create personalised assistive models.

However, subjective observation of these mismatched runs reveals that most of the degradation occurred due to the model's difficulty in correctly inferring the driver's intention. The assistance frequently guided the wheelchair into incorrect rooms, deviating from the planned path. Despite this, the models still managed to offer some degree of collision avoidance, especially in narrow passages, such as doorways. This is reflected in the metric "Fraction of time clearing collisions", where performance did not degrade as severely as when using the incorrect model.

This finding aligns with an intuitive understanding of the problem. The simulated disability only affected a subset of the model's inputs (namely the driver's velocity commands). The other input, the laser scan data, depended solely on the environment. While velocity inputs convey intent (e.g., wanting to turn left or right), the laser scans constrain the range of viable options (e.g., not going straight into a wall). This means that although the mismatched models failed at intention prediction, they still provided practical reactive safety assistance.

These results suggest that both models independently learned to use the laser data to avoid obstacles. This highlights an opportunity for transfer learning, where a shared feature representation (such as obstacle avoidance via laser scans) could be reused across users, potentially improving training efficiency. This line of research remains open for future exploration [63,64].

## 6. Discussion

The experiments reported previously provide evidence that the proposed simulator and LAD models advance the state of the art in assistive robotics. Regarding generalisation, this approach extends the previous investigations by Soh and Demiris [40,65] and Kucukyilmaz and Demiris [42], where assistance in limited environments was validated. The results presented here confirm that, with improved data collection procedures, LAD can maintain performance even in unseen courses, an essential requirement for real-world deployment.

In terms of assistive performance, the presented findings align with earlier studies showing that demonstrations can improve wheelchair navigation [23,35,36,65]. However, while prior work has only validated LAD on a small scale or with restricted scenarios, the present results systematically quantify improvements across multiple performance metrics, confirming consistent benefits of assistance.

Concerning robustness, earlier LAD studies evaluated only one type of impairment [23,40,65]. By contrast, the experiments reported here demonstrate that LAD remains effective across several distinct simulated disabilities, albeit with variable recovery rates. This supports the claim that LAD offers a general strategy for creating customised assistive models.

Finally, the personalisation experiments showed that models optimised for one driver fail when applied to others, leading in some cases to worse results than no assistance at all. This corroborates the intuition behind personalised models and is consistent with previous work on shared control adaptation [42]. Such evidence highlights the importance of tailoring LAD models to each user's driving behaviour and impairment profile.

Taken together, these results showed that the simulator developed here not only validates the core motivations behind LAD, but also provides a research platform to investigate open challenges. Nonetheless, it is important to acknowledge that the simulator simplifies several aspects of human-robot interaction. Therefore, while indicative, the results should be interpreted with caution and complemented by future studies with real users.

Furthermore, some remarks about limitations can be highlighted. The primary goal of the simulator developed in this work was to enable rapid experimentation with variables that impact LAD models. It is not intended to fully replicate the complex nuances of human–human interactions mediated through a robotic platform. As such, the results presented in the section before, while indicative, should be interpreted with caution.

Another limitation is that the physical models of the wheelchair and its sensors are simplified. Although this enables efficient experimentation, it restricts the fidelity of the simulated interactions. In addition, scalability has been assessed only for a few simulated drive-increasing situations; further validation is needed to confirm robustness in larger-scale scenarios considering the methodology proved here.

Finally, the simulator is being adapted for support with hardware-in-the-loop experiments. This limits its immediate applicability to real-time testing and integration with physical platforms. These limitations present opportunities for future work, particularly in enhancing the fidelity of the dynamic models, extending scalability to larger user groups, and expanding compatibility with robotic hardware for experimental validation.

## 7. Conclusions

This work advances the understanding and validation of LAD as a viable approach for enabling robots to provide personalised assistance. To this end, a custom simulator was developed; one that is capable of mimicking the experience of driving a robotic wheelchair while affected by hand control impairments, as well as of autonomously reproducing the whole triadic interaction required by LAD. The simulator enabled the collection of extensive data through repeated runs across varied environments, which, in turn, allowed for the exploration of alternative data preparation techniques, model architectures, and training procedures. These experiments led to improved generalisation capabilities (a critical requirement for real-world assistive applications).

With the improved model, several features typically attributed to LAD could be examined in greater depth. The results demonstrate that, at least in simulation, LAD models are capable of generalising to previously unseen environments while still offering meaningful navigational assistance. Moreover, the findings confirm that these models can support a variety of impairment types, reinforcing one of the central arguments for using LAD: the ability to create personalised assistive solutions tailored to the specific needs of each user.

In summary, it is also possible to highlight the contributions that this work brought out: (i) the development of a modular and extensible simulator for wheelchair navigation and triadic driver–robot–assistant interactions (which is to be released as open-source software); (ii) the investigation of new data collection procedures and learning models, resulting in improved training setups for LAD; and (iii) the validation of fundamental motivations for using LAD, demonstrating its ability to generate personalised assistive policies, handle generic disabilities, and generalise beyond training environments.

### *Future Works*

The limitations discussed above highlight several directions for future research, all aimed at bridging the gap between simulated and real-world assistive wheelchair operation. Enhancing the fidelity of the simulator, particularly in modelling wheelchair dynamics and hand control impairments, will be crucial to ensure that the behaviours learned in simulation are relevant and transferable. Additionally, assessing the scalability of the framework in scenarios with larger user groups and multiple assistants will help guarantee that the system remains robust under more complex and varied conditions. Integrating hardware-in-the-loop also represents a key step, allowing the simulator to interact di-

rectly with physical devices and real-time controllers, thus providing a practical bridge between simulation and deployment. These improvements are essential to ensure that the assistive policies developed in a virtual environment can be reliably applied to actual wheelchair systems.

Building on these advancements, the next step involves evaluating the developed LAD models with real human users, for example. This process will begin with controlled simulator-based experiments that include participants with diverse impairment profiles, providing a safe and adjustable environment to test and refine the policies. Once validated in simulation, the learned policies will be implemented on a real robotic wheelchair through real-time control interfaces. Furthermore, suppose a policy identifies that a user tends to understeer when navigating tight corridors. In such a case, the system can subtly augment the turning input to improve trajectory tracking while preserving the user's sense of control. Similarly, if the system detects hesitations or inconsistent input patterns, it can provide adaptive speed modulation or predictive path assistance to maintain safety and comfort. By blending user inputs with autonomous corrections in real time, the wheelchair can respond dynamically to the user's intentions while mitigating potential errors.

This staged approach ensures that the benefits observed in simulation (such as improved navigation efficiency, safety, and user comfort) are effectively translated to real-world scenarios. Ultimately, these steps will support the development of assistive systems that are both adaptive and responsive and capable of providing personalised assistance that accommodates individual user behaviours and impairments.

**Author Contributions:** Conceptualisation, V.B.S.; methodology, V.B.S.; validation, V.B.S. and M.F.d.S.; formal analysis, V.B.S.; investigation, V.B.S.; writing—original draft, V.B.S.; writing—review and editing, M.F.d.S. and P.M.; resources, V.B.S., M.F.d.S. and P.M.; supervision, P.M.; project administration, V.B.S.; funding acquisition, V.B.S., M.F.d.S. and P.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was financially supported by the Federal Center of Technological Education of Minas Gerais (CEFET-MG) and by the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES).

**Data Availability Statement:** The data used to support the findings of this study are available from the corresponding authors upon request.

**Acknowledgments:** The authors would like to thank Yiannis Demiris for his guidance during the development of this work. The authors would also like to thank the Personal Robotics Lab and its members for providing the necessary infrastructure and helpful discussions.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Leaman, J.; La, H.M. A Comprehensive Review of Smart Wheelchairs: Past, Present, and Future. *IEEE Trans. Hum.-Mach. Syst.* **2017**, *47*, 486–489. [[CrossRef](#)]
2. Sivakanthan, S.; Candiotti, J.L.; Sundaram, S.A.; Duvall, J.A.; Sergeant, J.J.G.; Cooper, R.; Satpute, S.; Turner, R.L.; Cooper, R.A. Mini-review: Robotic wheelchair taxonomy and readiness. *Neurosci. Lett.* **2022**, *772*, 136482. [[CrossRef](#)]
3. Narayanan, V.K.; Spalanzani, A.; Babel, M. A semi-autonomous framework for human-aware and user intention driven wheelchair mobility assistance. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, 9–14 October 2016; pp. 4700–4707. [[CrossRef](#)]
4. Lei, Z.; Tan, B.Y.; Garg, N.P.; Li, L.; Sidarta, A.; Ang, W.T. An Intention Prediction Based Shared Control System for Point-to-Point Navigation of a Robotic Wheelchair. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8893–8900. [[CrossRef](#)]
5. Sezer, V. An Optimized Path Tracking Approach Considering Obstacle Avoidance and Comfort. *J. Intell. Robot. Syst.* **2022**, *105*, 21. [[CrossRef](#)]

6. Burhanpurkar, M.; Labbe, M.; Guan, C.; Michaud, F.; Kelly, J. Cheap or Robust? The practical realization of self-driving wheelchair technology. In Proceedings of the 2017 International Conference on Rehabilitation Robotics (ICORR), London, UK, 17–20 July 2017; pp. 1079–1086. [[CrossRef](#)]
7. Sanders, D.A. Using Self-Reliance Factors to Decide How to Share Control Between Human Powered Wheelchair Drivers and Ultrasonic Sensors. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 1221–1229. [[CrossRef](#)] [[PubMed](#)]
8. Udupa, S.; Kamat, V.R.; Menassa, C.C. Shared autonomy in assistive mobile robots: A review. *Disabil. Rehabil. Assist. Technol.* **2023**, *18*, 827–848. [[CrossRef](#)] [[PubMed](#)]
9. Kim, D.J.; Hazlett-Knudsen, R.; Culver-Godfrey, H.; Rucks, G.; Cunningham, T.; Portee, D.; Bricout, J.; Wang, Z.; Behal, A. How Autonomy Impacts Performance and Satisfaction: Results From a Study with Spinal Cord Injured Subjects Using an Assistive Robot. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2012**, *42*, 2–14. [[CrossRef](#)]
10. Erdogan, A.; Argall, B.D. The effect of robotic wheelchair control paradigm and interface on user performance, effort and preference: An experimental assessment. *Robot. Auton. Syst.* **2017**, *94*, 282–297. [[CrossRef](#)]
11. Teodorescu, C.S.; Carlson, T. AssistMe: Using policy iteration to improve shared control of a non-holonomic vehicle. In Proceedings of the 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9–12 October 2022; pp. 941–948. [[CrossRef](#)]
12. Bastos-Filho, T.F.; Cheein, F.A.; Muller, S.M.T.; Celeste, W.C.; de la Cruz, C.; Cavalieri, D.C.; Sarcinelli-Filho, M.; Amaral, P.F.S.; Perez, E.; Soria, C.M.; et al. Towards a New Modality-Independent Interface for a Robotic Wheelchair. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2014**, *22*, 567–584. [[CrossRef](#)]
13. Wästlund, E.; Sponseller, K.; Pettersson, O.; Bared, A. Evaluating gaze-driven power wheelchair with navigation support for persons with disabilities. *J. Rehabil. Res. Dev.* **2015**, *52*, 815–826. [[CrossRef](#)]
14. MacIel, G.M.; Pinto, M.F.; Da Júnior, I.C.; Coelho, F.O.; Marcato, A.L.; Cruzeiro, M.M. Shared control methodology based on head positioning and vector fields for people with quadriplegia. *Robotica* **2022**, *40*, 348–364. [[CrossRef](#)]
15. Kutbi, M.; Li, H.; Chang, Y.; Sun, B.; Li, X.; Cai, C.; Agadakos, N.; Hua, G.; Mordohai, P. Egocentric Computer Vision for Hands-Free Robotic Wheelchair Navigation. *J. Intell. Robot. Syst.* **2023**, *107*, 10. [[CrossRef](#)]
16. Kundu, A.S.; Mazumder, O.; Lenka, P.K.; Bhaumik, S. Hand Gesture Recognition Based Omnidirectional Wheelchair Control Using IMU and EMG Sensors. *J. Intell. Robot. Syst.* **2018**, *91*, 529–541. [[CrossRef](#)]
17. Ropper, A.H.; Adams, R.; Victor, M.; Samuels, M.A. *Adams and Victor's Principles of Neurology*; McGraw Hill: New York, NY, USA, 2005.
18. Kairy, D.; Rushton, P.; Archambault, P.; Pituch, E.; Torkia, C.; El Fathi, A.; Stone, P.; Routhier, F.; Forget, R.; Demers, L.; et al. Exploring Powered Wheelchair Users and Their Caregivers' Perspectives on Potential Intelligent Power Wheelchair Use: A Qualitative Study. *Int. J. Environ. Res. Public Health* **2014**, *11*, 2244–2261. [[CrossRef](#)]
19. Padir, T. Towards personalized smart wheelchairs: Lessons learned from discovery interviews. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, Milan, Italy, 25–29 August 2015; pp. 5016–5019. [[CrossRef](#)]
20. Soh, H.; Demiris, Y. When and how to help: An iterative probabilistic model for learning assistance by demonstration. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3230–3236. [[CrossRef](#)]
21. Zheng, B.; Verma, S.; Zhou, J.; Tsang, I.W.; Chen, F. Imitation Learning: Progress, Taxonomies and Challenges. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 6322–6337. [[CrossRef](#)]
22. Ravichandar, H.; Polydoros, A.S.; Chernova, S.; Billard, A. Recent Advances in Robot Learning from Demonstration. *Annu. Rev. Control Robot. Auton. Syst.* **2020**, *3*, 297–330. [[CrossRef](#)]
23. Schettino, V.; Demiris, Y. Improving Generalisation in Learning Assistance by Demonstration for Smart Wheelchairs. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 5474–5480. [[CrossRef](#)]
24. Schettino, V.; Demiris, Y. Inference of user-intention in remote robot wheelchair assistance using multimodal interfaces. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 4600–4606. [[CrossRef](#)]
25. WHO. *Global Strategy on Human Resources for Health: Workforce 2030*; Technical Report; World Health Organization (WHO): Geneva, Switzerland, 2016.
26. Schettino, V. Learning to Assist in Triadic Human-Robot Interaction. Ph.D. Thesis, Imperial College London, London, UK, 2021. Available online: <http://hdl.handle.net/10044/1/97948> (accessed on 8 August 2025).
27. Najafi, M.; Adams, K.; Tavakoli, M. Robotic learning from demonstration of therapist's time-varying assistance to a patient in trajectory-following tasks. In Proceedings of the 2017 International Conference on Rehabilitation Robotics (ICORR), London, UK, 17–20 July 2017; pp. 888–894. [[CrossRef](#)]

28. Lauretti, C.; Cordella, F.; Guglielmelli, E.; Zollo, L. Learning by Demonstration for Planning Activities of Daily Living in Rehabilitation and Assistive Robotics. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1375–1382. [[CrossRef](#)]
29. Fong, J.; Rouhani, H.; Tavakoli, M. A Therapist-Taught Robotic System for Assistance During Gait Therapy Targeting Foot Drop. *IEEE Robot. Autom. Lett.* **2019**, *4*, 407–413. [[CrossRef](#)]
30. Ewerton, M.; Rother, D.; Weimar, J.; Kollegger, G.; Wiemeyer, J.; Peters, J.; Maeda, G. Assisting Movement Training and Execution With Visual and Haptic Feedback. *Front. Neurobotics* **2018**, *12*, 24. [[CrossRef](#)]
31. Meccanici, F.; Karageorgos, D.; Heemskerk, C.J.M.; Abbink, D.A.; Peternel, L. Probabilistic Online Robot Learning via Teleoperated Demonstrations for Remote Elderly Care. In *Advances in Service and Industrial Robotics (RAAD 2023)*; Springer Nature: Cham, Switzerland, 2023; Volume 135, pp. 12–19. [[CrossRef](#)]
32. Losey, D.P.; Srinivasan, K.; Mandlekar, A.; Garg, A.; Sadigh, D. Controlling Assistive Robots with Learned Latent Actions. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 378–384. [[CrossRef](#)]
33. Losey, D.P.; Jeon, H.J.; Li, M.; Srinivasan, K.; Mandlekar, A.; Garg, A.; Bohg, J.; Sadigh, D. Learning latent actions to control assistive robots. *Auton. Robot.* **2022**, *46*, 115–147. [[CrossRef](#)]
34. Qiao, C.Z.; Sakr, M.; Muelling, K.; Admoni, H. Learning from Demonstration for Real-Time User Goal Prediction and Shared Assistive Control. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 3270–3275. [[CrossRef](#)]
35. Goil, A.; Derry, M.; Argall, B.D. Using machine learning to blend human and robot controls for assisted wheelchair navigation. In Proceedings of the IEEE International Conference on Rehabilitation Robotics, Seattle, WA, USA, 24–26 June 2013; pp. 1–6. [[CrossRef](#)]
36. Matsubara, T.; Miro, J.V.; Tanaka, D.; Poon, J.; Sugimoto, K. Sequential intention estimation of a mobility aid user for intelligent navigational assistance. In Proceedings of the 2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Kobe, Japan, 31 August–4 September 2015; pp. 444–449. [[CrossRef](#)]
37. Poon, J.; Cui, Y.; Miro, J.V.; Matsubara, T.; Sugimoto, K. Local driving assistance from demonstration for mobility aids. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 5935–5941. [[CrossRef](#)]
38. Casado, F.E.; Demiris, Y. Federated Learning from Demonstration for Active Assistance to Smart Wheelchair Users. In Proceedings of the 2022 IEEE/RJS International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022; pp. 9326–9331. [[CrossRef](#)]
39. Bozorgi, H.; Ngo, T.D. Beyond Shared Autonomy: Joint Perception and Action for Human-In-The-Loop Mobile Robot Navigation Systems. *J. Intell. Robot. Syst.* **2023**, *109*, 20. [[CrossRef](#)]
40. Soh, H.; Demiris, Y. Learning Assistance by Demonstration: Smart Mobility With Shared Control and Paired Haptic Controllers. *J. Hum.-Robot Interact.* **2015**, *4*, 76. [[CrossRef](#)]
41. Kucukyilmaz, A.; Demiris, Y. One-shot assistance estimation from expert demonstrations for a shared control wheelchair system. In Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Kobe, Japan, 31 August–4 September 2015; pp. 438–443. [[CrossRef](#)]
42. Kucukyilmaz, A.; Demiris, Y. Learning Shared Control by Demonstration for Personalized Wheelchair Assistance. *IEEE Trans. Haptics* **2018**, *11*, 431–442. [[CrossRef](#)] [[PubMed](#)]
43. Reddy, S.; Dragan, A.D.; Levine, S. Shared Autonomy via Deep Reinforcement Learning. In Proceedings of the Robotics: Science and Systems 2018, Pittsburgh, PA, USA, 26–30 June 2018.
44. Kapusta, A.; Erickson, Z.; Clever, H.M.; Yu, W.; Liu, C.K.; Turk, G.; Kemp, C.C. Personalized collaborative plans for robot-assisted dressing via optimization and simulation. *Auton. Robot.* **2019**, *43*, 2183–2207. [[CrossRef](#)]
45. Clegg, A.; Erickson, Z.; Grady, P.; Turk, G.; Kemp, C.C.; Liu, C.K. Learning to Collaborate From Simulation for Robot-Assisted Dressing. *IEEE Robot. Autom. Lett.* **2020**, *5*, 2746–2753. [[CrossRef](#)]
46. Erickson, Z.; Gangaram, V.; Kapusta, A.; Liu, C.K.; Kemp, C.C. Assistive Gym: A Physics Simulation Framework for Assistive Robotics. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 10169–10176. [[CrossRef](#)]
47. Erickson, Z.; Gu, Y.; Kemp, C.C. Assistive VR Gym: Interactions with Real People to Improve Virtual Assistive Robots. In Proceedings of the 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Naples, Italy, 31 August–4 September 2020; pp. 299–306. [[CrossRef](#)]
48. John, N.W.; Pop, S.R.; Day, T.W.; Ritsos, P.D.; Headleand, C.J. The Implementation and Validation of a Virtual Environment for Training Powered Wheelchair Manoeuvres. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 1867–1878. [[CrossRef](#)]
49. Arlati, S.; Colombo, V.; Ferrigno, G.; Sacchetti, R.; Sacco, M. Virtual reality-based wheelchair simulators: A scoping review. *Assist. Technol.* **2020**, *32*, 294–305. [[CrossRef](#)]

50. Vailland, G.; Grzeskowiak, F.; Devigne, L.; Gaffary, Y.; Fraudet, B.; Leblong, E.; Nouviale, F.; Pasteau, F.; Breton, R.L.; Guegan, S.; et al. User-centered design of a multisensory power wheelchair simulator: Towards training and rehabilitation applications. In Proceedings of the 2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR), Toronto, ON, Canada, 24–28 June 2019; pp. 77–82. [\[CrossRef\]](#)
51. Morère, Y.; Hadj Abdelkader, M.; Cosnuau, K.; Guilmois, G.; Bourhis, G. Haptic control for powered wheelchair driving assistance. *IRBM* **2015**, *36*, 293–304. [\[CrossRef\]](#)
52. Devigne, L.; Babel, M.; Nouviale, F.; Narayanan, V.K.; Pasteau, F.; Gallien, P. Design of an immersive simulator for assisted power wheelchair driving. In Proceedings of the IEEE International Conference on Rehabilitation Robotics, London, UK, 17–20 July 2017; pp. 995–1000. [\[CrossRef\]](#)
53. Devigne, L.; Pasteau, F.; Carlson, T.; Babel, M. A shared control solution for safe assisted power wheelchair navigation in an environment consisting of negative obstacles: A proof of concept. In Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 6–9 October 2019; pp. 1043–1048. [\[CrossRef\]](#)
54. Di Gironimo, G.; Matrone, G.; Tarallo, A.; Trotta, M.; Lanzotti, A. A virtual reality approach for usability assessment: Case study on a wheelchair-mounted robot manipulator. *Eng. Comput.* **2013**, *29*, 359–373. [\[CrossRef\]](#)
55. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An open-source Robot Operating System. *ICRA Workshop Open Source Syst.* **2009**, *3*, 5.
56. Perlin, K. An image synthesizer. *ACM SIGGRAPH Comput. Graph.* **1985**, *19*, 287–296. [\[CrossRef\]](#)
57. Viswanathan, P.; Zambalde, E.P.; Foley, G.; Graham, J.L.; Wang, R.H.; Adhikari, B.; Mackworth, A.K.; Mihailidis, A.; Miller, W.C.; Mitchell, I.M. Intelligent wheelchair control strategies for older adults with cognitive impairment: User attitudes, needs, and preferences. *Auton. Robot.* **2017**, *41*, 539–554. [\[CrossRef\]](#)
58. Li, L.; Jamieson, K.; DeSalvo, G.; Rostamizadeh, A.; Talwalkar, A. Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization. *J. Mach. Learn. Res.* **2018**, *18*, 1–52.
59. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian optimization of machine learning algorithms. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; Volume 4, pp. 2951–2959.
60. Yurtsever, E.; Lambert, J.; Carballo, A.; Takeda, K. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access* **2020**, *8*, 58443–58469. [\[CrossRef\]](#)
61. Belkhale, S.; Cui, Y.; Sadigh, D. Data Quality in Imitation Learning. In Proceedings of the Advances in Neural Information Processing Systems 36 (NeurIPS 2023), New Orleans, LA, USA, 10–16 December 2023; Volume 36, pp. 80375–80395.
62. Dicianno, B.E.; Cooper, R.A.; Coltellar, J. Joystick Control for Powered Mobility: Current State of Technology and Future Directions. *Phys. Med. Rehabil. Clin. N. Am.* **2010**, *21*, 79–86. [\[CrossRef\]](#)
63. Niu, S.; Liu, Y.; Wang, J.; Song, H. A Decade Survey of Transfer Learning (2010–2020). *IEEE Trans. Artif. Intell.* **2020**, *1*, 151–166. [\[CrossRef\]](#)
64. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A survey on deep transfer learning. In Proceedings of the Artificial Neural Networks and Machine Learning—ICANN 2018, Rhodes, Greece, 4–7 October 2018; Volume 11141, pp. 270–279. [\[CrossRef\]](#)
65. Soh, H.; Demiris, Y. Towards Early Mobility Independence: An Intelligent Paediatric Wheelchair with Case Studies. In Proceedings of the IROS Workshop on Progress, Challenges and Future Perspectives in Navigation and Manipulation Assistance for Robotic Wheelchairs, Vilamoura, Portugal, 12 October 2012.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.