2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Improving Generalisation in Learning Assistance by Demonstration for Smart Wheelchairs

Vinicius Schettino and Yiannis Demiris

Abstract— **Learning Assistance by Demonstration (LAD) is concerned with using demonstrations of a human agent to teach a robot how to assist another human. The concept has previously been used with smart wheelchairs to provide customised assistance to individuals with driving difficulties. A basic premise of this technique is that the learned assistive policy should be able to generalise to environments different than the ones used for training; but this has not been tested before. In this work we evaluate the assistive power and the generalisation capability of LAD using our custom teleoperation and learning system for smart wheelchairs, while seeking to improve it by experimenting with different combinations of dimensionality reduction techniques and machine learning models. Using Autoencoders to reduce the dimension of laser-scan data and a Gaussian Process as the learning model, we achieved a 23% improvement in prediction performance against the combination used by the latest work on the field. Using this model to assist a driver exposed to a simulated disability, we observed a 9.8% reduction in track completion times when compared to driving without assistance.**

I. INTRODUCTION

Smart robotic wheelchairs are, from a hardware perspective, powered wheelchairs with added sensors and a computer [1], [2]. Using the sensors to collect information about its surroundings, the robot[1] is able to automatically modify the user input, sharing control of navigation. The goals of the application usually fall within providing safer or faster navigation, or reducing the user's cognitive workload. While some studies estimate that the majority of powered wheelchair users could benefit from this sort of technology [3], those that experience difficulties with motor control of upper limbs are certainly among the ones that could profit the most. Unfortunately, there are a myriad of conditions (cerebral palsy, Parkinson's disease, amyotrophic lateral sclerosis, Erb's palsy, etc.) that can lead to the manifestation of these difficulties. The symptoms include loss of dexterity, tremors, muscle weakness and spasms and can affect each individual in a unique way. As a consequence, it becomes difficult to derive hard-coded policies capable of satisfyingly providing assistance for all these individual needs.

When robots need to execute custom tasks that are difficult to be programmatically designed but can readily be performed by humans, Learning by Demonstration [4] can usually be applied. If the task is related to providing assistance to humans, it then falls within Learning Assistance by Demonstration [5] (LAD), a subset of Learning by Demonstration. In the context of smart wheelchairs, we can leverage the

Authors are with the Personal Robotics Laboratory, Department of Electrical and Electronic Engineering, Imperial College London, United Kingdom; v.schettino17@imperial.ac.uk; y.demiris@imperial.ac.uk

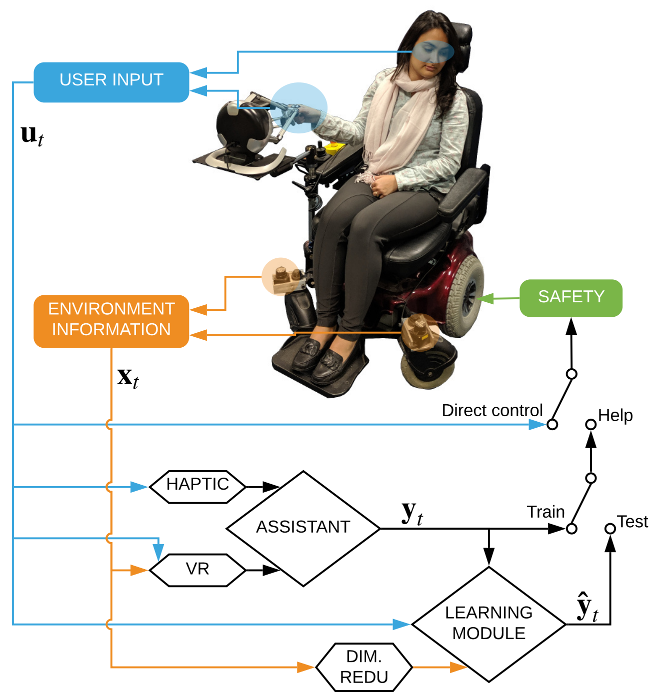

Fig. 1. System architecture used for Learning Assistance by Demonstration in smart wheelchairs. A remote human assistant provides teleoperated demonstrations of how to help a person with driving difficulties. To infer the driver's intention in real time, the assistant is aided by haptic and virtual reality interfaces, allowing easier interpretation of the multimodal information registered from the environment and the user. A machine uses the demonstrations to learn how to automate the customised assistive policy.

tacit knowledge of professionals like physiotherapists, nurses and caretakers, who are familiar with the user and their condition, and ask them to provide demonstrations of how to help that specific individual. Then, a machine learning model can be used to approximate a mapping function between input data (sensor readings and driver commands) and the provided demonstrations, as depicted in Figure 1. If learning is successful and the model is capable of generalising, the robot should be able to autonomously assist the driver, without need of further intervention from the demonstrator[2].

The question of generalisation, however, still lacks deeper exploration. A basic premise of LAD is that, after collecting demonstration data, the learned policy should be able to help the person using it in their daily lives. In the case of

[1]The terms *robot*, *wheelchair* and *smart wheelchair* are used interchangeably throughout the text

[2]The terms *assistant*, *remote operator* and *demonstrator* are used interchangeably throughout the text

smart wheelchairs, this translates to the assistive policy being generic enough to still be useful outside of the training area. Nevertheless, we are not aware of any study investigating the validity of this premise. Furthermore, some characteristics of the data naturally available for this scenario pose considerable generalisation concerns for machine learning models. First, there is need of a triadic interaction between two human agents, one of them disabled, and a robot, which limits the amount of data samples that can be collected in a reasonable training time. Second, the data needed to represent environment information with acceptable accuracy is of high dimensionality. These two factors combined can potentially lead to models that overfit the assistive policy to spatial features that are only present in the training course. If performance is then tested on the same course, it might yield apparently good results, but without being useful for the driver outside of the training area.

In this paper, we collected data from a driver exposed to a simulated disability and an expert demonstrator using our custom teleoperation and learning platform for smart wheelchairs. Then, using different obstacle courses for training and testing, we evaluated the generalisation performance of a variety of learning models and investigated ways of improving it. Finally, we used track completion time as a metric for assessing assistive power of both human help and machine-generated help.

II. RELATED WORKS

A. Assistive policies for smart wheelchairs

Usage of recorded driving demonstrations to aid in the derivation of assistive policies for smart wheelchairs has previously been explored. In [6], the variability in angular speed commands issued by drivers when travelling through different paths is used to determine, at each point in time, how much weight should be given to autonomous driving in a shared control scheme. In [7] driving demonstrations are instead used to learn how to predict the drivers intended goal locations in a known map. In both cases, local path planning can be generated by using off-the-shelf techniques traditionally available in mobile robotics.

Other works used the expert demonstrations to directly learn reference navigation behaviours. In [8], this was done by forming a lookup table composed of recorded sensor data and the associated control command. Afterwards, any sensor readings from the robot can be matched against the closest input on this table and the corresponding control action is used as reference for that particular scenario. In [9], the driving demonstrations are used to simultaneously perform short-term intention estimation and provide local path planning.

In common, all these works use *driving* demonstrations from *expert* drivers in an attempt to generate better policies for helping a user to navigate. However, should the assisted user operate very differently from the expert, due to an incapacitating disability for example, it can be unfeasible to match their actions, thus reducing the usefulness of the expert driving behaviour for intention estimation or local path planning.

B. Learning Assistance by Demonstration

By contrast, a more direct approach is proposed with Learning Assistance by Demonstration. Instead of using the experts for, ultimately, learning how to drive, they are directly tasked with generating demonstrations of *how to help the driver*. The concept was first introduced in [5], where separate online learning modules were used for predicting when (classification) and how (regression) to help the driver. The system was tested both in simulation and with a paediatric wheelchair, with users being exposed to a simulated disability. In [10], the work was expanded to include haptic pairing [11] between driver and demonstrator and to use Mixtures of Experts as the learning modules, with Online Infinite Echo-State Gaussian Process [12] as the basic units. It was observed that the simulated disability hindered driving performance, measured by lap completion time, but human-help was able to bring it back to base-line level (when the simulated disability was not present). Additionally, it was observed that machine-generated help yielded similar performance improvements.

In these works, however, assistive demonstrations and real-time testing of machine-help were performed on the same track. From this, the authors noted that the robot's 'when-to-help' signal was mostly triggered at the same locations were the demonstrator provide assistance during training. While this indicates that the learned policy correctly responds to spatial features of the environment, it is also possible that the model was overfitting to these features. This would lead to degraded performance when the driver actually needed to use the system; that is, outside of the training course. Accordingly, the authors note in their conclusions that "additional experiments are needed to resolve outstanding issues such as model generalisability (in particular, to test how well policies learned in one location extend to other environments)".

The works in [13], [14] moved away from assessing assistive power and only explored how well the machine could learn to replicate the demonstrator's actions. Thus, no simulated disability was imposed to drivers. Other differences were continuously providing assistive signals (no 'when-to-help' module), offline training with a regular Gaussian Process model and use of a single lap for recording assistive demonstrations (one-shot learning). In this case, different tracks were used for training and testing. However, the tracks were still part of the same obstacle course, thus retaining the possibility of overfitting to spatial features. Again, in [13] the authors observed that generalisation could not be fully demonstrated and note as future work the importance of using completely different trajectories for training and testing.

III. METHODOLOGY

To test the generalisation capability of LAD for smart wheelchairs, we devised a data gathering experiment using our custom assistive teleoperation system (see section III-A). The data collected was used to learn how to automate the assistant's actions and, using a different scenario, the generalisation performance of the learned policy was tested (III-B). In an attempt to improve it, multiple combinations of dimensionality-reduction techniques and machine learning models were evaluated (III-C).

A. System overview

The smart wheelchair used is equipped with two laser-scanners on the front and one on the back. The system used for assistive teleoperation is described in details in [15]. It consists of using of various modalities of interfaces for intuitively representing the driving scene to the assistant. The goal is to allow the remote operator to make more accurate and timely inferences about the driver's intentions.

The first interface conveying driver's input information is haptic pairing. For this, two identical joysticks with force-feedback capability are used, one for the remote operator and one for the wheelchair. The joysticks' end-effector positions are matched at a high rate, thus providing the assistant with a haptic feeling as if they were holding the same joystick as the person on the wheelchair. The second interface is eye-gaze estimation [16]. A webcam on the wheelchair provides a video stream of the driver's face and RT-GENE [17] is used to perform online estimations of the gaze direction. The yaw angle of this estimation is represented to the assistant by accordingly turning the head of a virtual mannequin. This mannequin sits on a wheelchair which moves on a map built online using the third interface, SLAM, which is primarily used for explaining the surroundings of the wheelchair. But since the human assistant is naturally capable of scene understanding (for example, the assistant understands the different actions reasonably available when the wheelchair is in the middle of a corridor or facing a doorway), it can also be used for inferring the driver's intention. By exerting force on their joystick, the assistant can overcome the haptic pairing forces and generate alternative velocity command signals. A teleoperation control signal is used to decide who conducts the wheelchair at each point in time, the driver or the remote operator. In any case, the command velocities pass through a safety layer before being sent to the wheelchair, in order to avoid collisions against obstacles. This safety measure does not steer the wheelchair away from obstacles, it merely slows down the vehicle if it gets too close to an obstacle, leading to a halt at approximately 20 cm.

Previous work in LAD for smart wheelchairs employed different strategies for sharing control of navigation between driver and demonstrator. While in [10] the demonstrator would take over control of the wheelchair in order to provide assistance, in [13] navigation was continuously shared between both human agents. However, during our initial trial runs, the driver reported feeling uncomfortable with either option, due to unwillingly having to relegate control of the vehicle, at times being driven to points contrary to their goals. This is in accordance with smart wheelchairs literature, where lack of control is reported as a potential user-rejection factor [18], [19].

To improve this, we explored a new paradigm for 'when-to-help' by directly letting the driver manage the teleoperation control signal. Hence, we ask the demonstrator to continuously provide assistive commands and let the driver define when assistance is needed, which is done by simply pressing a button on their joystick. With this new setup, the drivers always know when they are relinquishing control and can quickly regain it if the provided assistance is not in accordance with their intentions. Furthermore, in a LAD scenario, this paradigm has the added benefits of eliminating the need to learn 'when-to-help' [5], while also precluding spurious assistive signals from constantly interfering with the drivers normal operation [14].

To learn how to automate the assistive actions provided by the demonstrator, the scheme illustrated in Figure 1 is used. Environment information $\mathbf{x}_t$ and user input $\mathbf{u}_t$ data, captured by the wheelchair's sensors, are fed to the assistant, which is aided by the haptic and virtual reality modules for intuitively making use of the information. Based on this, the assistant continuously provide demonstrations $\mathbf{y}_t$ of how to assist that specific driver. If the assistant's actions are modelled by a mapping function $\mathbf{y}_t = f(\mathbf{u}_t, \mathbf{x}_t) + \theta_t : \mathcal{U}, \mathcal{X} \rightarrow \mathcal{A}$, where θ represents noise due to inference of intention and other random factors; the machine task can then be formally defined as learning a regression function $\hat{\mathbf{y}}_t = \hat{f}(\mathbf{u}_t, \mathbf{x}_t)$ such that $\hat{\mathbf{y}}_t \approx \mathbf{y}_t$ for the entire set $\{\mathcal{U}, \mathcal{X}\}$. Specifically, in this work we are exploring if the approximation still holds for the elements of $\mathcal{X}$ not seen during training.

B. Data collection and testing

To address the possible overfitting concerns previously mentioned, a special data gathering and testing procedure was devised, as illustrated in Figure 2. In a controlled and wheelchair-safe environment, three physically different scenarios, or obstacle courses, were built. The first two courses were used to compose a training dataset. It is important to have distinct courses composing the training data because we are specifically seeking methods that are able to generalise to a variety of environments and situations, instead of overfitting to local spatial features (for example, whenever approaching obstacle X, provide an assistive signal to turn left). Each course was marked with predefined target locations and an endless list of goals was generated by randomly sampling the target numbers. The driver was then asked to follow the list, driving to the goals in sequential order for around 10 minutes. During this, the assistant was continuously providing assistance commands, which were recorded along with the laser-scan data and the driver's gaze estimation, command velocities and teleoperation control signal.

The third scenario was used only for testing performances. In this course, the driver was given two finite lists with 10 goals each and asked to finish the tracks as fast as possible. The same kind of data was recorded and this formed the test set. To assess the assistive potential of human help, the tracks

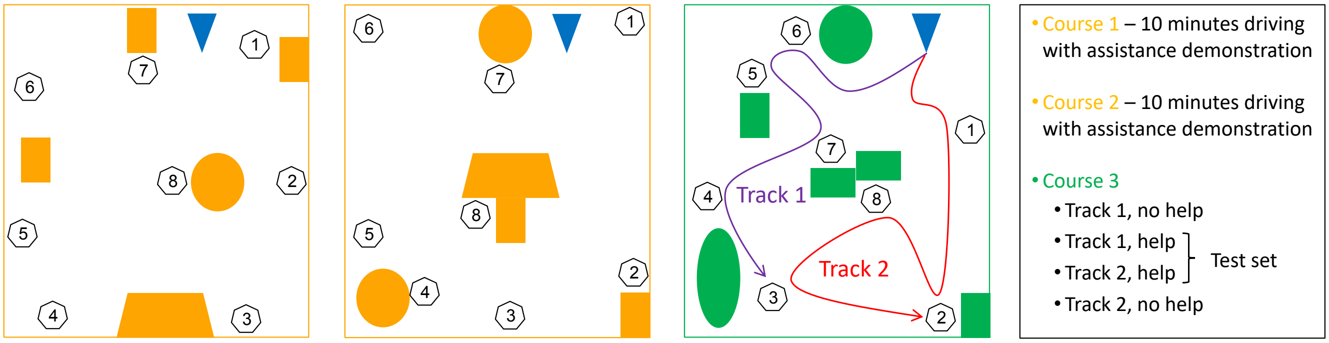

Fig. 2. Illustration of experimental procedure used. In orange, two different obstacle courses with marked goal positions are used for gathering a training dataset. The driver is asked to continuously drive to randomly picked goal numbers, while environment and user information are recorded, along with the provided assistive demonstrations. To test the generalisation capability of the learning model, a third, unseen, course is used. Here, finite tracks are employed and track completion time is used as a metric to test assistive potential.

were also run without the demonstrator being available. To correct for learning effects, the order in which the assistant was available or not was switched between the tracks.

The data collected was used to train the machine learning models. Afterwards, we repeated the same procedure on the third obstacle course, except that the assistive signals were automatically being generated in real-time by the machine. This was done to evaluate the assistive potential of the learned model and happened on a separate day, about a month latter, thus mitigating possible learning effects by the driver.

Since our goal is to learn *personalised* assistive policies, the same driver-demonstrator pair was used for the entire experiment. Furthermore, the driver, an able-bodied subject, was exposed to a simulated disability. The reasoning for this is to test the LAD concept in cases where it is more likely to be used, that is, with drivers that would have difficulty in manoeuvring the wheelchair without any assistance. While the ideal would be to test the system directly with disabled subjects, it would also be ill-advised to expose the target population to a still experimental system. Thus, as in previous work on the field [5], [10], the simulated disability was the compromise found. For easier and fairer comparison against these works, the same disability was used, which consists of a difficulty in executing right-turn motions with the joystick. This kind of symptom could be manifested, for example, in people suffering from Erb's palsy, which can lead to a loss of supination power on the forearm [20]. The disability was simulated using the force-feedback capability of the joystick, by automatically exerting a high opposing force whenever the driver tried to turn to the right.

After recording the data, the entire dataset was time synchronised to 4 Hz using a nearest-sample approach, except for the control signal, which used a nearest-previous-sample approach due to its naturally low-changing frequency. Each sample is composed of the target signal and 363 features - 360 readings from the laser-scans, the estimated yaw angle of the driver's eye-gaze and the driver's linear and angular command velocities. The target signal is formed by the demonstrator's angular velocities. For simplicity, and because it is not related to the given disability, the demonstrator's linear velocity is not mapped, with the driver's linear velocity directly bypassed to the robot. Pre-processing analysis showed that some of the samples on the training set had to be discarded due to problems with the gaze-estimation so, in the end, the training dataset was composed of $\sim$3000 samples.

C. Improving generalisation

Teleoperation platform: Several considerations took place in an attempt to improve generalisation performance. First, by using a teleoperation platform we can eliminate the correspondence problem [4], common in robot Learning by Demonstration applications. Additionally, the demonstrator could only observe the scene and driving actions through the 'eyes' of the robot. This is different from previous work on the field, where the demonstrator had either a direct or camera view of the scene available. By only providing the demonstrator and learning module with the same amount of information, we avoid noise from 'unexplainable' demonstrations (for example, if the demonstrator provides assistive signals to avoid an obstacle that they can see but is not be captured by the wheelchair's sensors). These considerations are done to facilitate the machine learning task, specially in a scenario were the amount of data available for training is restricted.

Pre-processing of laser data: A common problem with laser-scan samples is that many of the readings can be corrupted with invalid values. In our system, this happens if the reading is out of the sensor's maximum range (an Inf value is returned) or if the reading falls within the footprint of the wheelchair (a NaN value is returned). To cope with this and avoid discarding samples, first the NaN values are converted to Inf and then all readings are inverted. This results in readings close to the robot, perhaps the most relevant ones, having a high value and invalid readings having a zero value.

Dimensionality reduction: The relatively high number of features, due to the 360 channels on the scan samples, can lead to overfitting problems. Therefore, it is prudent to test the effects of applying different dimensionality reduction algorithms to this data. The simplest method for this is spatial subsampling, which was applied by all previous work on the field. Keeping an uniform angular distance between readings on the circular scan, tests were performed keeping 256, 64, 16 and 4 readings from each sample. Additionally, tests were also performed keeping the full scan sample and completely removing it.

However, due to the natural characteristics of the readings generated by a moving laser scanner, this simplistic feature selection mechanism can deteriorate the quality of input data. As a more powerful approach, we propose the use of Autoencoders [21]. We experimented with this using unsupervised training and up to four fully-connected layers, resulting in encodings of sizes 256, 64, 16 or 4. For all layers, logistic sigmoid was used as the encoding and decoding transfer functions and both weight and sparsity regularisation was applied. As an alternative, Principal Component Analysis (PCA) [22] was also considered, keeping enough principal components to explain 99, 90, 75 and 50% of the observed variance of the scan data. Post-analysis showed that this corresponded to respectively 229, 59, 14 and 5 components.

Learning algorithms: Finally, we explored the impact that different machine learning algorithms have on the generalisation capability of LAD. All previous work on the field [5], [10], [13], [14] used variants of Gaussian Process (GP) [23] as learning models. When the number of features is not very high, GPs can be an interesting option due to their ability to directly capture uncertainty in the input space, being trainable with relatively small datasets and allowing one to readily insert prior knowledge into training. However, the choice of the kernel function is important. While Squared Exponential kernel has previously been used, we noticed that it is overly smooth to capture the variations observed in the target signal, shown in Figure 3. Hence, we opted for also evaluating performance with the Matérn 3/2 kernel. To better deal with the cases where the number of features is high, we further tested this kernel with Automatic Relevance Determination [23].

For completeness, different families of common machine learning models were also compared, namely: Linear Regression (LR) [24], Support Vector Machine (SVM) [25], Random Forest [26] and Gradient Boosting (Boost) [27]. Linear regression models were trained with Lasso, Ridge or no regularisation. For Lasso and Ridge, cross-validation was used to determine appropriate values for the regularisation coefficient. SVM models were trained with linear, gaussian and polynomial kernels and cross-validation was used to determine appropriate values for insensitive band, box constraint, kernel scale and, in the last case, polynomial order parameters. Gradient Boosting for regression is implemented using shallow trees and mean-squared error as the loss function. For both Random Forest and Gradient Boosting models, cross-validation optimisation was used to choose appropriate values for the number of learning cycles used and the minimum leaf size of each tree on the ensemble. For Gradient Boosting the learning rate of shrinkage is also optimised. For hyperparameter tuning, 5-fold cross-validation with MSE as the loss function was always used and bayesian optimisation applied for more efficient search of minima. Additionally, the maximum number of objective function evaluations and maximum training time was the same for all models.

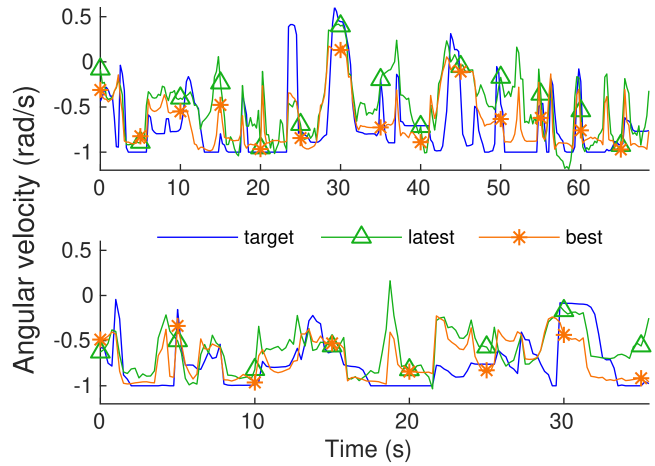

Fig. 3. Comparison of human provided assistive signal against machine predictions for both test tracks (best viewed in colour)

All the dimensionality reduction techniques and machine learning models were implemented and trained with Matlab and, where omitted, default parameters were used.

IV. RESULTS AND DISCUSSION

Table I shows Root Mean Squared Error losses for all combinations of dimensionality reduction techniques and models discussed in the previous section. Highlighted in green is the combination used by the latest related work [14] and in orange is the best observed performance. Using data collected from the test tracks when assistance was being requested, Figure 3 compares the responses predicted by these combinations against the target signal (angular velocities provided by the demonstrator). To assess assistive performance, the best observed combination was used for making real-time predictions on the wheelchair. The driver that participated in the data-gathering experiment was asked to return and was allowed as much time as wanted to familiarise themselves with the new driving mode. Afterwards, the same testing procedure on the third course was repeated; except with assistance being automatically generated by the machine. Lap completion times for all runs on both test tracks are shown in Table II.

Discussion

As can be seen from Table I, a Gaussian Process model using the Matérn 3/2 kernel combined with Autoencoders for reducing the dimension of laser data down to 16 features attained the best prediction performance, with a 23% improvement against the combination used by the latest work on the field. Although not shown here, we also observed moderate improvements due to the inclusion of gaze-estimation as the predictor and to the use of different weights for samples were assistance was being requested.

TABLE I

RMSE FOR DIFFERENT COMBINATIONS OF DIMENSIONALITY REDUCTION TECHNIQUES AND MACHINE LEARNING MODELS

	LR	LR Lasso	LR Ridge	GP SqrExp	GP Matérn3/2	GP ARD Matérn3/2	SVM Lin	SVM Gauss	SVM Poly	Random Forest	Boost	Average
FullScan	0.4143	0.3209	0.6483	0.4155	0.4137	0.4405	0.3941	0.4255	0.4207	0.3956	0.3466	0.4214
SubSamp256	0.3893	0.4840	0.3353	0.4541	0.4445	0.4390	0.4125	0.3931	0.3444	0.3595	0.3277	0.3985
SubSamp64	0.3799	0.3697	0.3792	0.3947	0.3925	0.3829	0.3907	0.5306	0.4843	0.3566	0.3473	0.4007
SubSamp16	0.3659	0.3647	0.3651	0.3788	0.4162	0.4593	0.3662	0.4297	0.4227	0.3430	0.3587	0.3882
SubSamp4	0.3329	0.3330	0.3329	0.3600	0.3528	0.3370	0.3357	0.4820	0.3330	0.3418	0.3361	0.3525
NoScan	0.3214	0.3214	0.3214	0.3085	0.3058	0.3048	0.3291	0.3337	0.3824	0.3794	0.3321	0.3309
PCA99	0.3831	0.3983	0.4536	0.7630	0.4771	0.4587	0.4115	0.4391	0.3598	0.5585	0.3506	0.4594
PCA90	0.3639	0.3590	0.3793	0.3922	0.3890	0.3912	0.3951	0.4486	0.3507	0.4013	0.3460	0.3833
PCA75	0.3659	0.3660	0.3659	0.3511	0.3554	0.3589	0.3825	0.6660	0.4661	0.4531	0.3823	0.4103
PCA50	0.3513	0.3516	0.3514	0.3556	0.3975	0.4029	0.3429	0.7010	0.3332	0.3546	0.3373	0.3890
Autoenc256	0.3824	0.5003	0.3494	0.3550	0.3832	0.3921	0.4119	0.3847	0.4341	0.3834	0.3392	0.3923
Autoenc64	0.3396	0.3288	0.3375	0.3389	0.3282	0.3358	0.3366	0.3202	0.3335	0.3282	0.3332	0.3328
Autoenc16	0.3425	0.3218	0.3368	0.2924	0.2914	0.2942	0.3285	0.3296	0.3265	0.3366	0.3289	0.3208
Autoenc4	0.3410	0.3214	0.3214	0.3172	0.3081	0.3276	0.3342	0.3218	0.3124	0.3442	0.3313	0.3255
Average	0.3624	0.3672	0.3770	0.3912	0.3754	0.3804	0.3694	0.4433	0.3788	0.3811	0.3427	0.3790

TABLE II

COMPLETION TIMES AND IMPROVEMENTS FOR BOTH TEST TRACKS

	1st day		2nd day	
	No help	Improvement due to human help	No help	Improvement due to machine help
Track 1	281 s	-0.3%	388 s	15.5%
Track 2	279 s	3.2%	379 s	4.0%

From Table II we first notice a difference in lap completion times between the two days when data was collected, in spite of the same tracks and disability being used. This seems to be related to the driver presenting different physical and/or mental states between both days. While on the first day the driver presented a more aggressive driving behaviour, on the second day they seemed more subdued by the limitation imposed by the disability. Although not anticipated, these variations in motor function and attitude are common with many real disabilities [28], [29], [30].

To correct for this variation, the improvements in track completion times are calculated on a day-by-day basis. Following this analysis, we observed that the assistance provided by the demonstrator did not lead to significant improvements in lap completion times. Contributing to this is the possibility that, by overcoming the simulated disability, the driver was already navigating near their best possible performance, hence not allowing much room for improvement. Additionally, when the demonstrator needs to remotely infer the intention of the driver, their assistive behaviour naturally becomes more erratic, thus reducing its usability.

This unpredictability in the assistant behaviour also leads to a more noisy target function, therefore increasing the difficulty of the machine learning task. Notwithstanding, we observed that the policy learned by our model was able to generate helpful assistive commands in real time. On the second day of the experiment, when the driver exhibited a slower driving behaviour and thus could potentially take more advantage of the custom assistance, machine help led to an average improvement in track completion time of 9.8%. We note that, by training with multiple demonstration samples, the learning module is able to filter-out part of the inference of intention noise, thus generating a more consistent assistive policy, which benefits the driver.

Limitations

A limitation of this work is related to the disability simulation. In an attempt to more realistically mimic a real human disability, we did not constrain the velocity commands that were sent to the robot and merely imposed opposing forces at the joystick during right turns. However, due to hardware limitations of the joystick, the driver is able to overcome the simulated disability, should they exert enough force. This made our experiment more susceptible to the interference of "human factors", such as differences in attitude, which complicates analysis. Notwithstanding, we posit that with real disabilities factors like tiredness, distraction and learning should play similar roles.

V. CONCLUSIONS

In this work we dealt with assisted driving for smart wheelchairs and explored if helping policies automatically generated from human demonstrations are able to generalise to different physical scenarios, not seen during training. We observed that it is possible and that the use of more sophisticated learning models and dimensionality reduction techniques can help in this sense. Seeking a higher user satisfaction, We also explored a new paradigm for "when-to-help" by switching to "ask-for-help" instead of taking over control or continuously sharing navigation; which also alleviated the machine learning problem. Our future work shall be in the direction of exploring how to further improve the assistive power of these personalised helping machines. Additionally, we intend to study how to better model the driver's disability.

ACKNOWLEDGMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 - and by CEFET-MG as a PhD scholarship to VS, and a Royal Academy of Engineering (RAEng) Chair in Emerging Technologies to YD.

REFERENCES

[1] J. Leaman and H. M. La, “A Comprehensive Review of Smart Wheelchairs: Past, Present, and Future,” *IEEE Transactions on Human-Machine Systems*, vol. 47, pp. 486–489, 8 2017.

[2] R. C. Simpson, “Smart wheelchairs: A literature review,” *The Journal of Rehabilitation Research and Development*, vol. 42, no. 4, p. 423, 2005.

[3] R. C. Simpson, E. F. LoPresti, and R. A. Cooper, “How many people would benefit from a smart wheelchair?,” *Journal of rehabilitation research and development*, vol. 45, pp. 53–71, 12 2008.

[4] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, “A survey of robot learning from demonstration,” *Robotics and Autonomous Systems*, vol. 57, pp. 469–483, 5 2009.

[5] H. Soh and Y. Demiris, “When and how to help: An iterative probabilistic model for learning assistance by demonstration,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3230–3236, IEEE, 11 2013.

[6] A. Goil, M. Derry, and B. D. Argall, “Using machine learning to blend human and robot controls for assisted wheelchair navigation,” in *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, pp. 1–6, IEEE, 6 2013.

[7] T. Matsubara, J. V. Miro, D. Tanaka, J. Poon, and K. Sugimoto, “Sequential intention estimation of a mobility aid user for intelligent navigational assistance,” in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, (Kobe), pp. 444–449, IEEE, 8 2015.

[8] H. Chow and Y. Xu, “Learning Human Navigational Skill for Smart Wheelchair in a Static Cluttered Route,” *IEEE Transactions on Industrial Electronics*, vol. 53, pp. 1350–1361, 6 2006.

[9] J. Poon, Y. Cui, J. V. Miro, T. Matsubara, and K. Sugimoto, “Local driving assistance from demonstration for mobility aids,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5935–5941, IEEE, 5 2017.

[10] H. Soh and Y. Demiris, “Learning Assistance by Demonstration: Smart Mobility With Shared Control and Paired Haptic Controllers,” *Journal of Human-Robot Interaction*, vol. 4, p. 76, 1 2015.

[11] K. J. Kuchenbecker, “Haptics and Haptic Interfaces,” in *Encyclopedia of Robotics*, pp. 1–9, Berlin, Heidelberg: Springer Berlin Heidelberg, 2018.

[12] H. Soh and Y. Demiris, “Spatio-Temporal Learning With the Online Finite and Infinite Echo-State Gaussian Processes,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, pp. 522–536, 3 2015.

[13] A. Kucukyilmaz and Y. Demiris, “One-shot assistance estimation from expert demonstrations for a shared control wheelchair system,” in *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 438–443, IEEE, 8 2015.

[14] A. Kucukyilmaz and Y. Demiris, “Learning Shared Control by Demonstration for Personalized Wheelchair Assistance,” *IEEE Transactions on Haptics*, vol. 11, pp. 431–442, 7 2018.

[15] V. Schettino and Y. Demiris, “Inference of user-intention in remote robot wheelchair assistance using multimodal interfaces - In Press,” in *IEEE International Conference on Intelligent Robots and Systems*, 2019.

[16] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, “MPIIGaze: Real-World Dataset and Deep Appearance-Based Gaze Estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 162–175, 1 2019.

[17] T. Fischer, H. J. Chang, and Y. Demiris, “RT-GENE: Real-Time Eye Gaze Estimation in Natural Environments,” in *European Conference on Computer Vision*, pp. 339–357, 2018.

[18] D. Kairy, P. Rushton, P. Archambault, E. Pituch, C. Torkia, A. El Fathi, P. Stone, F. Routhier, R. Forget, L. Demers, J. Pineau, and R. Gourdeau, “Exploring Powered Wheelchair Users and Their Caregivers' Perspectives on Potential Intelligent Power Wheelchair Use: A Qualitative Study,” *International Journal of Environmental Research and Public Health*, vol. 11, pp. 2244–2261, 2 2014.

[19] P. Viswanathan, E. P. Zambalde, G. Foley, J. L. Graham, R. H. Wang, B. Adhikari, A. K. Mackworth, A. Mihailidis, W. C. Miller, and I. M. Mitchell, “Intelligent wheelchair control strategies for older adults with cognitive impairment: user attitudes, needs, and preferences,” *Autonomous Robots*, vol. 41, no. 3, pp. 539–554, 2017.

[20] G. J. Tortora and B. Derrickson, *Principles of anatomy and physiology*. Wiley, 15th ed., 2016.

[21] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th international conference on Machine learning - ICML '08*, (New York, New York, USA), pp. 1096–1103, ACM Press, 2008.

[22] H. Abdi and L. J. Williams, “Principal component analysis,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, pp. 433–459, 7 2010.

[23] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. MIT Press, 2006.

[24] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2006.

[25] L. Wang, ed., *Support Vector Machines: Theory and Applications*, vol. 177 of *Studies in Fuzziness and Soft Computing*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005.

[26] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[27] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer Series in Statistics, New York, NY: Springer New York, 2009.

[28] B. Bruguerolle and N. Simon, “Biologic Rhythms and Parkinson's Disease: A Chronopharmacologic Approach to Considering Fluctuations in Function,” *Clinical Neuropharmacology*, vol. 25, pp. 194–201, 7 2002.

[29] S. M. A. Lamers, L. Bolier, G. J. Westerhof, F. Smit, and E. T. Bohlmeijer, “The impact of emotional well-being on long-term recovery and survival in physical illness: a meta-analysis.,” *Journal of behavioral medicine*, vol. 35, pp. 538–47, 10 2012.

[30] N. Bellamy, R. B. Sothern, J. Campbell, and W. W. Buchanan, “Circadian rhythm in pain, stiffness, and manual dexterity in rheumatoid arthritis: relation between discomfort and disability.,” *Annals of the rheumatic diseases*, vol. 50, pp. 243–8, 4 1991.