Interface-Aware Assistance for 7-DoF Robot Arm Teleoperation: Case Studies on Feasibility

Mahdieh Nejati Javaremi, Larisa YC Loke, and Brenna Argall

Northwestern University, Shirley Ryan AbilityLab
m.nejati@u.northwestern.edu

Abstract. This paper details the development, implementation and experimental evaluation of an interface-aware, task-agnostic assistance system for shared human-robot teleoperation, specifically applied to a 7-DoF robotic arm. The system addresses a limitation of current shared-control methods by considering the impact of control interfaces on user input precision and the robot agent's incomplete understanding of the human's policy. The approach is evaluated in empirical case studies involving participants with spinal cord injuries. The personalized assistance system improves safety and reduces cognitive load.

1 Introduction

Individuals with severe motor impairments face limited control interface options, posing considerable challenges in operating devices designed to enhance their quality of life. The challenges in operating assistive devices are compounded by interface and physical constraints, device complexity, task complexity, and control fidelity requirements. Incorporating robotics autonomy in assistive devices can improve physical independence for those with motor impairments, but challenges arise when human intentions are misaligned with the signals measured by the robots through the interfacing device [10]. Discrepancies can occur between the intended signals from the human and those received by the autonomy through the interface due to various factors such as lack of skill and symptoms of neuromuscular injury such as spasticity or tremors [2, 21]. Fatigue, stress, and complex mappings between robot actions and interface actions can worsen these discrepancies [8].

Shared-control assistance, often applied to task-level commands such as robot joint or mobile base velocities, does not generally consider the physical interface operation when interpreting user commands [1, 14]. Discrepancies between user-intended signals and those measured through the interface can, however, affect the performance of shared control systems [16]. Probabilistic robotic techniques have been designed to handle sensor measurement noise and actuation uncertainty in autonomous robots operating outside controlled lab settings [19] but do not handle noise in the human interface signal. We posit that in human-robot systems with human control signals, it is important to also consider an additional source of uncertainty that contributes to these discrepancies—the *uncertainty and noise associated with human interaction with the control interface.* This paper contributes the following:

1. An algorithmic formulation of interface-aware, task-agnostic shared control assistance, that relaxes key assumptions in our seminal interface-aware framework [9].
2. Experiment evolution from 3-DoF toy example in simulation to real-hardware 7-DoF robotic arm application.
3. Case study analyses of our approach with spinal cord injured end-users.

2 Background and Related Literature

We present a summary of related research on robot-aided assistance, mapping control interface signals to robots, and interface noise handling.

Robot-Aided Assistance via Shared-Control Shared control—also called assistive teleoperation, assist-as-needed, or mixed initiative control in various domains— is of importance in human-robot interaction because it enables the distribution of control between the human and autonomous agent to improve overall team performance and safety without taking away the human's full agency [15, 14]. The specific shared control framework is often domain and application specific [1], and spans a wide range of interactions and arbitration topologies between the human and autonomous partners.

In signal-level arbitration, low-level signals are provided by both the human and autonomy to control the same robotic device. A common approach is to define a blending mechanism that decides how and when to incorporate signals from both partners, and typically depends on various requirements such as the task context [4] or the preferences of the human partner [7]. Alternatively, in policy methods the human and autonomy actions are not blended, and rather the autonomous agent action is optimized given the human command [3, 10].

Mapping Control Interface Signals to Robots Assistive robots designed to aid in daily living activities predominantly operate in the $SE(3)$ space, possessing a 6-dimensional task space. However, control interfaces tailored for individuals with severe motor impairments often are constrained in the number of independent control signals they can generate. The three primary interfaces include the 2-axis or 3-axis joystick, the 1-D or 2-D head-mounted switch array, and the 1-D sip/puff device. To control high-dimensional assistive robots with these constrained interfaces, users encounter a fundamental disparity between the low-dimensional control command space and the robot's high-dimensional motion space. To bridge this gap, control interfaces often utilize modal mappings, wherein mode switching enables a single interface action to govern multiple robot motion dimensions [20]. This indirect mapping can impact the difficulty of controlling such robots with physically accessible control interfaces.

Handling Uncertainty in Interface Signals Prior work in handling interface signal uncertainty is primarily concerned with non-commercial interfaces that rely on physiological signals such as EMG or EEG [22, 11]. Although the relationship between intended human motion and produced human motion has been extensively studied in other domains [13], it has received limited attention within the field of robotics [12]. Prior work in handling the uncertainty in user input typically filters or smooths high-frequency noise in continuous input signals [17, 18]. To our knowledge, no robotics work other than [9] uses models of the stochasticity in user input to infer the human's intentions and subsequently provide appropriate corrections.

3 Technical Approach

We define **interface-awareness** as an *explicit* and *deliberate* representation of the uncertainty in measured user commands. Figure 1 shows the sources of noise in a typical shared control system. Classically, two primary sources of onboard noise and uncertainty are modeled—noise in sensor measurements ϵ_s and uncertainty in actuation ϵ_a [19]. We identify distinct interactions in manual teleoperation affecting user input uncertainty, necessitating interface awareness: (1) the *sensorimotor noise* (ϵ_h) due to various sources such as cognitive overload and neuromuscular injury and (2) the noise from the *interfacing system* (ϵ_i) arising from the physical mechanism of how commands are issued and how control mode switches occur. We posit that modeling how physical interface-activation actions are mapped to task-level robot actions, and how the control signal is then altered through the interface, will help the autonomy to *reason about deficiencies* in human teleoperation in order to *improve the quality* of robot operation.

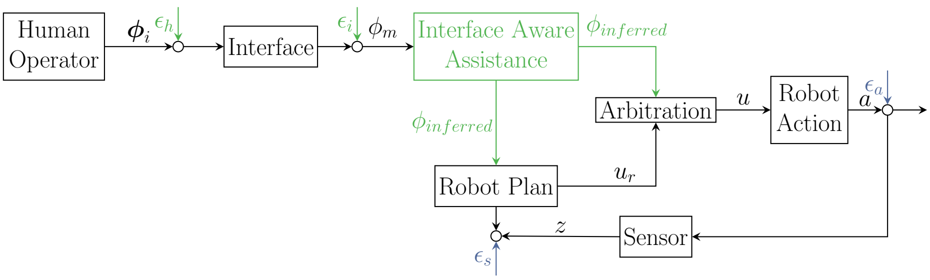

Fig. 1. Interface-aware assistance pipeline. Inferred interface signal $\boldsymbol{\phi}_{inferred}$ is used instead of the raw human signal $\boldsymbol{\phi}_m$. Noise and signals specific to interface-awareness are highlighted in green.

In our framework, we define $\boldsymbol{\phi} \in \mathbb{R}^{n_\phi}$ as the action that actuates the physical interface, which we refer to as the *interface-level action* (e.g., joystick deflection). Φ is the set of all interface-level actions, and n_ϕ is the dimensionality of the interface. Let $\boldsymbol{\phi}_i^t \in \Phi$ denote the unobservable *intended interface-level action* initiated by the user that aims to achieve the task-level action $\mathbf{a}^t \in \mathcal{A}$, while $\boldsymbol{\phi}_m^t \in \Phi$ is the actual *measured* *interface-level action.* $p\big(\mathbf{a}^t|\mathbf{x}^t\big)$ is the *control policy* the user follows during task execution, where x denotes the state. $p\big(\boldsymbol{\phi}_i^t|\mathbf{a}^t\big)$ is the user's *internal model* of the true mapping from task-level action primitives to the intended interface-level physical actions (e.g., to rotate the robot clockwise, deflect the joystick 90°). $p\big(\boldsymbol{\phi}_m^t|\boldsymbol{\phi}_i^t\big)$ is the model of stochastic deviations of the *measured* interface-level actions from the *intended* interface-level actions and is the *user input distortion model* (e.g., 80° measured for an intended 90° deflection).

We build upon our theoretical work [9], in which the model required knowledge of the ground-truth human control policy $p(\mathbf{a}^t|\mathbf{x}^t)$—easily accessed in simulations, but challenging to acquire in real-world scenarios, especially when human optimization includes physiological considerations such as motor impairments. Here we hypothesize that within an interface-aware assistance scheme, using models of a user's stochastic interface interaction and internal mapping from interface to robot control—easier to construct and task-agnostic—can improve task performance even *without* knowledge of their policy $p(\mathbf{a}^t|\mathbf{x}^t)$.

Algorithm 1 Task-Agnostic Interface-Aware Assistance

1: **function** Interface Aware Assistance(t, $\boldsymbol{\phi}_m^t$)
2: $\quad x \leftarrow y(f(\boldsymbol{\phi}_m^t))$ ▷ Forward project $\boldsymbol{\phi}_m^t$
3: $\quad x_{\text{safe}} \leftarrow$ Check Safety(x)
4: $\quad$ **if** $x_{\text{safe}} = True$ **then**
5: $\quad\quad \boldsymbol{\phi}_c^t \leftarrow \boldsymbol{\phi}_m^t$
6: $\quad$ **else**
7: $\quad\quad \mathbf{a}_e^t \leftarrow$ Estimate Action(Δt, $\boldsymbol{\phi}_m^t$)
8: $\quad\quad \boldsymbol{\phi}_i^t, H \leftarrow$ Infer Intended Command($\boldsymbol{\phi}_m^t$, $\mathbf{a}_e^t$) ▷ Algorithm 1 in [9]
9: $\quad\quad$ **if** $H \leq \epsilon$ **then** ▷ uncertainty is low
10: $\quad\quad\quad \boldsymbol{\phi}_c^t \leftarrow \boldsymbol{\phi}_i^t$ ▷ correct user commands
11: $\quad\quad$ **else**
12: $\quad\quad\quad \boldsymbol{\phi}_c^t \leftarrow \mathbf{0}$ ▷ block commands
13: $\quad$ **return** $\boldsymbol{\phi}_c^t$

Algorithm 1 details our task-agnostic interface-aware assistance algorithm. The method initially projects the current measured interface command into the future (Algorithm 1, line 2) to determine the robot's end effector pose if the interface command were to be applied. If the resulting robot state is deemed unsafe (end effector is in collision with known objects in the environment), we estimate the most probable task-level action (line 7, and Algorithm 2) and infer the intended interface-level action (line 8) utilizing our knowledge of $p(\boldsymbol{\phi}_i|\mathbf{a})$ and $p(\boldsymbol{\phi}_m|\boldsymbol{\phi}_i)$. We compute a confidence in this inference as the normalized entropy H of the distribution $p(\mathbf{a}^t|\boldsymbol{\phi}_m^t)$, computed as in the seminal theoretical work [9], and correct the measured interface-level action if the entropy H is below a preset threshold ϵ (ϵ=0.9 in our hardware implementation).

Algorithm 2 outlines our approach to identify the most probable task-level action, a_t, without prior knowledge of the user's policy. The inputs to the *Estimate Action* function are the projection time-step and the current measured interface-level action. The algorithm first computes the action given the interface level command using the inverse of the mapping function of interface commands to robot actions (line 2). The next steps of the algorithm handle probability vectors representing the likelihood of actions based on the current observations

Algorithm 2 Action Estimation with Domain Knowledge

1: **function** ESTIMATE ACTION($\Delta t, \boldsymbol{\phi}_m^t$)
2: $\quad \mathbf{a}_t \leftarrow f^{-1}(\boldsymbol{\phi}_m)$
3: $\quad P \leftarrow$ ones(length($\mathcal{A}$)) ▷ Initialize a vector of ones with the same length as $\mathcal{A}$
4: $\quad P[a_t] \leftarrow 0$ ▷ Set the unsafe action to cold
5: $\quad P \leftarrow \frac{P}{\sum P}$ ▷ Normalize the vector
6: $\quad P' \leftarrow$ zeros(length($\mathcal{A}$))
7: $\quad P'[a_{t-1}] \leftarrow 1$ ▷ Sets prior action to hot
8: $\quad$ **if** $\Delta t > T_{long}$ **then**
9: $\quad\quad P_{\mathbf{a}} = P_{\mathbf{a}} - \eta P'$ ▷ η is a weighting factor
10: $\quad$ **else if** $\Delta t < T_{short}$ **or** $\boldsymbol{\phi}^{t-1}$ is *motion* **then**
11: $\quad\quad P_{\mathbf{a}} = P_{\mathbf{a}} + \eta P'$
12: $\quad P \leftarrow \frac{P}{\sum P}$
13: $\quad \mathbf{a}_{estimate}^t \leftarrow argmax(P)$
14: $\quad$ **return** $\mathbf{a}_{estimate}^t$

of the environment and domain knowledge of interface usage characteristics [16]. We introduce a vector P initialized with ones (representing a uniform probability across all possible actions in $\mathcal{A}$ (line 3). We assume any unsafe state and its correlating interface-level action as unintended. Therefore, the action deemed unsafe (obtained from the interface level command) is assigned a zero probability (line 4), making it 'cold'. To ensure valid distribution representations, P is normalized (line 5). A new one-hot vector P' is introduced in line 6, hot at prior action a_{t-1} (line 7). Temporal interface input characteristics are utilized to adjust the probability vector P given the prior action using P', using a weighting factor η, based on the time difference between the current and previous input (lines 8-11). For longer time intervals (greater than T_{long}), the algorithm decreases the weight of the prior action. For shorter intervals, or if the previous command indicates motion, the algorithm increases the weight. η determines the strength of the adjustment. Following the adjustments, the vector is once again normalized. Finally, the action with the highest probability is returned as the estimated action (line 15). Essentially, to perform action estimation Algorithm 2 integrates domain knowledge, in the form of safety considerations from sensor input, and temporal dynamics of interface use.

4 End-User Case Studies

We executed our task-agnostic, interface-aware algorithm on robot hardware, conducting two case studies with motor-impaired participants. This constitutes the first implementation of an interface-aware framework on real hardware and without presupposing complete knowledge of human's control policy, contrasting prior simulation work with strict human policy and trajectory constraints [9].

Hardware and Materials We utilized a sip/puff interface to operate a 7-DoF Kinova JACO robotic arm in this study. The sip/puff, a discrete 1D interface, offers four interface-level actions—hard puff, hard sip, soft puff, soft sip—that map to different robot control actions depending on the current control mode m—the robot's operational subspace. To control the JACO, hard puff/sips correspond to mode-switching commands, while soft puff/sips denote positive/negative motion within a control dimension. The robot's state is defined to be $[x, y, z, \text{roll}, \text{pitch}, \text{yaw}, \text{gripper}]$ and the mode m.

Fig. 2. Top: Study setup. Participants use a sip/puff interface to teleoperate a robotic arm to retrieve a cup from a shelf and place it on the table. Bottom: The sip/puff interface.

Study Task The study incorporated a representative constrained manipulation task, featuring elements common in Activities of Daily Living (ADL), including constrained grasping and object placement [6]. The task involved navigating narrow spaces to grasp a cup from a shelf and place it in a confined area (Figure 2)—chosen for its relevance and its known difficulty for both human teleoperation of an $SE(3)$ robotic arm and robotic path-planners [5].

Study Procedure The study consisted of three phases of training and testing. In Phase 1, participants learned to use a sip/puff interface to issue control inputs, with testing involving screen prompts for different interface actions. This data was then used to create a personalized model for each participant to capture their unique input signal distortions. We utilized histogram binning to construct the personalized models, leveraging its simplicity, visual clarity, and insight into human variability and internal models. This non-parametric approach avoids assumptions about distribution forms, aptly capturing authentic human patterns and enhancing outlier robustness.Phase 2 trained participants on how to operate a robot arm using the sip/puff interface, including the range of achievable motions and mode switching commands. Testing involved showing participants a control action on the robot and asking for the corresponding interface-level action, with this data used to construct a personalized model (using the same modeling technique in Phase 1) of each participant's internal control mapping. Phase 3 had participants use the sip/puff to control the robot arm for an Activites of Daily Living inspired task, initially with a free exploration period, followed by testing both *with* personalized task-agnostic interface-aware assistance and *without* any assistance. Post-testing, participants filled out a NASA Task Load Index questionnaire to assess mental workload and shared their preferences regarding the operation of the robot.

5 Results

Participant Profiles and Performance Analysis The details of each case study participant and their experiences are presented with quantitative result metrics presented in Table 1.

Case 1 The participant, a 40-year-old woman with a cervical-level spinal cord injury, has been a daily powered wheelchair user with a right-handed joystick interface for 20 years, and has some experience and comfort with robotic devices. Her training proceeded smoothly with signs of improvement, but phase 2 testing for building the internal model introduced confusion, leading to a noisy internal model map and stochastic input model. Despite this, she successfully completed the task with and without interface-aware task-agnostic assistance, albeit with collisions in the latter case. The participant reported a lower NASA-TLX score, indicating lower mental workload, when using the assistance system.

Case 2 The second participant, a 58-year-old man with a cervical-level spinal cord injury, is not a powered wheelchair user but is accustomed to new technology and enjoys video gaming. Despite initial struggles, the participant's high confidence resulted in perceived control over the robot and interface, with a low NASA-TLX score indicating reduced mental workload. He experienced numerous collisions without assistance and succeeded only once per assistance condition. The assistance system identified many unintended commands, resulting in a significant number of mode switches. However, the assistance system reduced task completion time via quicker automatic mode switch corrections.

Table 1. Hardware Trial Metrics for Participants 1 & 2

Participant	Assistance condition	Trial success	collisions	# mode switches	Task time (s)	TLX score
Case 1	None	[1, 1]	[0, 2]	[34, 49]	[119, 167]	52.3
Case 1	Ours	[1, 1]	[0, 0]	[33, 61]	[124, 149]	49.5
Case 2	None	[1, 0]	[5, 6]	[135, 126]	[455, 528]	8.8
Case 2	Ours	[0, 1]	[0, 0]	[171, 206]	[350, 351]	8.6

6 Analysis and Interpretation

The initial results from the case studies reveal intriguing outcomes. The case studies also elucidate the disparities that can arise when transitioning from simulation-based toy experiments to real-world hardware setups.

Main Experimental Insights. Even without any knowledge of the task at hand, our task-agnostic interface-aware assistance demonstrated a promising improvement in performance metrics, especially for Case 2. Here we elaborate and expand on the key insights and guidance for future work.

Simulation vs. Real-World Discrepancies: *Observation:* There was a notable difference between tasks in Phase 1 and 2, designed to gather data to build personalized participant models, and the ADL evaluation task in Phase 3. The real hardware presented challenges in effectively crafting these models. Testing methods differed from natural robot interaction, causing confusion during the transition from training to testing tasks. This emphasizes the importance of ensuring training and testing methods mirror real-world device operation. *Action Item:* Ensure that training and testing tasks closely emulate real robot operations, avoiding abstractions like verbal selections. Refine the test procedures to capture genuine user interactions and reduce potential confusion.

Case Differentiation: *Observation:* The performance variation between the two cases highlighted both human variability and cognitive adaptability. Case 1 showed a stronger internal model, due to prior experience with similar devices. Case 2 showcased adaptability, possibly enhanced by familiarity with video gaming. *Action Item:* Condition on user background when analyzing performance and adaptability.

Performance Improvements and Plateaus: *Observation:* Case 2 witnessed a marked (30%) reduction in task completion time. This underscores the system's potential in aiding users unfamiliar with the interface. Case 1's times, however, seemed to reach a performance plateau. *Action Item:* Investigate ways to further enhance task performance, including how to address performance plateaus.

Safety Improvements: *Observation:* A noticeable advantage was the enhancement in safety, with the assistance system preventing all collisions observed during unassisted operations. *Action Item:* Maintain and further improve the safety features of the system, keeping real-world applications in mind.

User Feedback and System Limitations: *Observation:* Participants perceived the robot as obstructive, stemming from the algorithm's conservative safety measures. This led to confusion, stemming in part from an incomplete understanding of the assistance system operation. *Action Item:* Enhance system transparency via communication and feedback mechanisms. Optimize for a reduction in user confusion during the design of safety conditions.

Cognitive Load: *Observation:* Differences in TLX scores among participants indicated varying cognitive load assessments, possibly influenced by individual backgrounds, such as experience with video gaming. *Action Item:* Consider diverse participant backgrounds when evaluating cognitive load and devise strategies to mitigate any potential load.

Mode Switches: *Observation:* There was an unexpected increase in mode switches with the active assistance system, which might be due to either corrective mode switches by the system or inaccurate action estimates. *Action Item:* Investigate the user-system interaction dynamics to understand the cause behind increased mode switches and rectify any identified issues.

Improvement Strategies. Based on the insights and guidance above, we propose the following steps to improve user interaction with our assistive system.

System Transparency: Communication is key. Ensure users understand the system's assistance, especially during complex robotic arm movements. Introduce subtle cues to help participants grasp the robot's intentions.

<u>**Model Building and Human Training:**</u> Adapt models from simulation for the real world. Use more realistic methods to capture personalized distribution, update data collection methods, and adopt audio cues to enhance training. Enable direct engagement with the robot arm during training, to offer a more intuitive learning experience.

7 Conclusion

We presented the hardware implementation and task-agnostic evolution of our interface-aware robotic assistance framework, demonstrated through a feasibility study where two spinal cord injured individuals evaluated its potential and limitations in an ADL manipulation task. The case studies revealed the system's potential and the challenges of moving from simulation to real-world settings, guiding future refinements in methodology and design.

8 Acknowledgment

This material is based on work supported by the National Science Foundation under Grant CMMI-2208011.

Bibliography

[1] David A Abbink, Tom Carlson, Mark Mulder, Joost CF de Winter, Farzad Aminravan, Tricia L Gibo, and Erwin R Boer. A Topology of shared control systems—finding common ground in diversity. *Transactions on Human-Machine Systems*, pages 1–17, 2018.

[2] Icek Ajzen, Thomas C Brown, and Franklin Carvajal. Explaining the discrepancy between intentions and actions: The case of hypothetical bias in contingent valuation. *Personality and Social Psychology Bulletin*, pages 1108–1121, 2004.

[3] Alexander Broad, Todd Murphey, and Brenna Argall. Operation and imitation under safety-aware shared control. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 905–920. Springer, 2018.

[4] Tom Carlson, Robert Leeb, Ricardo Chavarriaga, and José del R Millán. Online modulation of the level of assistance in shared control systems. In *2012 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3339–3344. IEEE, 2012.

[5] Marco Costanzo, Simon Stelter, Ciro Natale, Salvatore Pirozzi, Georg Bartels, Alexis Maldonado, and Michael Beetz. Manipulation planning and control for shelf replenishment. *IEEE Robotics and Automation Letters*, pages 1595–1601, 2020.

[6] Peter F Edemekong, Deb L Bomgaars, Sukesh Sukumaran, and Shoshana B Levy. Activities of daily living. StatPearls Publishing, 2021.

[7] Ahmetcan Erdogan and Brenna D Argall. Prediction of user preference over shared-control paradigms for a robotic wheelchair. In *2017 International Conference on Rehabilitation Robotics*, pages 1106–1111. IEEE, 2017.

[8] A Aldo Faisal, Luc PJ Selen, and Daniel M Wolpert. Noise in the nervous system. *Nature Reviews Neuroscience*, pages 292–303, 2008.

[9] Deepak Gopinath*, Mahdieh Nejati-Javaremi*, and Brenna Argall. Customized handling of unintended interface operation in assistive robots. In *IEEE International Conference on Robotics and Automation*, pages 10406–10412, 2021.

[10] Kris Hauser. Recognition, prediction, and planning for assisted teleoperation of freeform tasks. *Autonomous Robots*, pages 241–254, 2013.

[11] Reva E Johnson, Konrad P Kording, Levi J Hargrove, and Jonathon W Sensinger. EMG versus torque control of human–machine systems: Equalizing control signal variability does not equalize error or uncertainty. *IEEE transactions on neural systems and rehabilitation engineering*, pages 660–667, 2016.

[12] Rebecca P Khurshid and Katherine J Kuchenbecker. Data-driven motion mappings improve transparency in teleoperation. *Presence*, pages 132–154, 2015.

[13] Simon Little, James Bonaiuto, Gareth Barnes, and Sven Bestmann. Human motor cortical beta bursts relate to movement planning and response errors. *PLoS biology*, page e3000479, 2019.

[14] Dylan P Losey, Craig G McDonald, Edoardo Battaglia, and Marcia K O'Malley. A review of intent detection, arbitration, and communication aspects of shared control for physical human–robot interaction. *Applied Mechanics Reviews*, 2018.

[15] Selma Musić and Sandra Hirche. Control sharing in human-robot team interaction. *Annual Reviews in Control*, pages 342–354, 2017.

[16] Mahdieh Nejati Javaremi, Michael Young, and Brenna D Argall. Interface operation and implications for shared-control assistive robots. In *IEEE 16th International Conference on Rehabilitation Robotics*, pages 232–239, 2019.

[17] Vangelis P Oikonomou, A Tzallas, Spiros Konitsiotis, D Tsalikakis, and D Fotiadis. The use of kalman filter in biomedical signal processing. *Kalman Filter: Recent Advances and Applications*, 2009.

[18] Robin S Sharp. Driver steering control and a new perspective on car handling qualities. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, pages 1041–1051, 2005.

[19] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. Probalistic robotics. *Kybernetes*, 2006.

[20] Hylke A Tijsma, Freek Liefhebber, and Just L Herder. A framework of interface improvements for designing new user interfaces for the manus robot arm. In *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.*, pages 235–240. IEEE, 2005.

[21] Robert J Van Beers, Patrick Haggard, and Daniel M Wolpert. The role of execution noise in movement variability. *Journal of Neurophysiology*, pages 1050–1063, 2004.

[22] John Williamson, Roderick Murray-Smith, Benjamin Blankertz, Matthias Krauledat, and K-R Müller. Designing for uncertain, asymmetric control: Interaction design for brain–computer interfaces. *International Journal of Human-Computer Studies*, pages 827–841, 2009.