Intelligent Human-Robot Interface Design

Thesis submitted in partial fulfillment

of the requirements for the degree of

"Doctor of Philosophy"

by

Guillaume Doisy

Submitted to the Senate of

Ben-Gurion University of the Negev

October 2014

Beer Sheva

Intelligent Human-Robot Interface Design

Thesis submitted in partial fulfillment

of the requirements for the degree of

"Doctor of Philosophy"

by

Guillaume Doisy

Submitted to the Senate of

Ben-Gurion University of the Negev

Approved by the advisors: Prof. Yael Edan Yael Edan

Prof. Joachim Meyer

Approved by the Dean of The Kreitman School of Advanced Graduate Studies __________

October 2014

Beer Sheva

This work was carried out under the supervision of

Prof. Yael Edan

Prof. Joachim Meyer

In the

Department of Industrial Engineering and Management

Faculty of Engineering Sciences

Research-Student's Affidavit when Submitting the Doctoral Thesis for Judgment

I Guillaume Doisy, whose signature appears below, hereby declare that

(Please mark the appropriate statements):

✓ I have written this Thesis by myself, except for the help and guidance offered by my Thesis Advisors.

Date: 13/11/2014 Student's name: Guillaume Doisy Signature: Doisy

Acknowledgments

To my advisors, **Prof. Yael Edan** and **Prof. Joachim Meyer**, who not only guided, trusted and motivated my research during these years but also made me feel welcome in Israel from the very first moment I landed. To Yael's family who represents the Israeli hospitality at its best.

To the INTRO project, and all its people, without whom I would have never entered the research community. A very special thanks to **Dr. Aleksandar Jevtic**, **Dr. Guido Schillaci** and soon to be doctors **Maria Elena Giannaccini** and **Saša Bodiroža**, my fellow ESR, friends, travel companions and warm hosts during my secondments in Berlin, Bristol and Biarritz. To their supervisors, **Prof. Verena V. Hafner** and **Prof. Sanja Dogramadzi**, who welcomed me in their labs.

To all the wonderful people of the BGU industrial engineering department. In particular to **Yossi Zahav, Nissim Abuhatzira, Noam Peles, Albina Bulgaro, Ayelet Atias** and **Efi Vitzrabin** without whom as a non-israeli non-hebrew speaker I would have not survived long in the holy land. Thanks to **Prof. Yisrael Parmet** for his statistically significant help.

To my dear friends **Cécile, Zip, Zat, Leslie, Wiwi, Baptiste, Elise, Thomas, Titi, Pierre, Asaf, Pauline, Audrey, Merav, Daniel, Tania** and **Antoine** who heard me complaining and saw me enjoying so many times during these years spent between a basement, a plane, a desert and the sea.

Finally, thanks to my family, my parents and my sisters Sophie, Laure and Claire for their unconditional support and their scientific inspiration.

This research was supported by the European Community's Seventh Framework Programme under the INTRO ITN project (grant agreement no. 238486). Partial support was also provided by the ABC Robotics Center funded by Helmsley Charitable Trust and by the Rabbi W. Gunther Plaut Chair in Manufacturing Engineering, both at Ben-Gurion University of the Negev, and the French Embassy in Israel.

Tables of Contents

Abstract

The use of automation, and more specifically the use of robots, is increasing in industry and recently also in people's daily life activities. The presence of personal robotics in our homes should lead to important societal changes in upcoming years, with personal robots assisting our aging societies in health care, therapy and rehabilitation but also serving as entertainers and household staff. However, current robotic technology is still limited. Fully autonomous robots, capable of performing new tasks in complex and unstructured, unknown and changing environments like our homes do not exist so far. Humans are highly flexible and can easily adapt to changing conditions, but they are far less accurate and reliable compared to robots. Hence, it is advantageous for robots and humans to collaborate, with each benefitting from the specific capabilities of the other. A critical component of successful human-robot collaboration is the interface, and this is particularly problematic for learning robots.

This thesis focuses on two research directions related to interface design aiming to increase the efficiency of human-robot collaboration: 1) understanding the design of the human-robot interface for robots with learning capabilities and 2) developing and testing novel intuitive human-robot interfaces based on state-of-the-art technological advances. The work is divided accordingly in two main parts: Part A and Part B.

In Part A different aspects related to interface design for learning robots were investigated. The focus was on how the interface design influences the user interaction with a robot with behavior which evolves over time in a changing environment. The research is organized in three user experiments in which different interaction conditions were tested and compared with the goal of extracting guidelines for future designers of human-robot interfaces. The first two experiments were conducted in two specially developed computer simulations with 42 and 96 participants accordingly, whereas the last one was conducted with a real mobile robot with 48 participants.

The first experiment focused on the impact of the number of changes of environmental conditions and the type of feedback provided about the learning behavior: the usefulness of warning the user about changes affecting the learning and of showing previews of the learned behavior was tested.

In the second experiment, a simplistic form of automation was used: a binary warning system, but its characteristic (i.e. its sensitivity) varied over time and the users' responses to these changes, and in particular the two dimensions of trust, compliance and reliance, were studied with different feedback conditions.

Finally, the third experiment, still in the context of a learning robot in a changing environment, looked at the level of automation at which the automation gained from the learning should be applied.

In Part B, a more practical approach of interface development was taken. Novel interfaces and algorithms were created using recent advances in the field of sensors and in particular the release of cheap RGB-D sensors like the Microsoft Kinect.

Four new interaction modalities were created and then evaluated: person following, pointing control, direct physical interaction and camera head control for remote robot teleoperation. First, a mobile robotic platform dedicated to person following was created from a customized generic differential drive robot. Then using this platform, person tracking and person following algorithms were developed and practically tested to achieve a robust person tracking behavior in complex office environments. Next, using the platform developed before, two more interaction modes were developed and added: a pointing control interface and a direct physical interface. These three interfaces were tested in usability experiments in two robot navigation control tasks which took place in a test home apartment with 24 participants. Finally, a new non-intrusive method based on the use of a Kinect sensor for controlling the orientation of the camera through the operator's head orientation in a robot teleoperation task was developed and compared to more classical interfaces. The performance of the interfaces tested was also evaluated against user familiarity with the system with 36 participants.

Part A: Interface design for learning robots

This research started to tackle a largely unexplored domain: interface design for learning robots. Learning algorithms for robots are quickly gaining in maturity but the question of how they should be implemented on a user perspective remains mostly unanswered. This work aimed to begin to bridge that gap. The results of the three experiments conducted show that the human interaction with a learning robot is not trivial and is sensitive to many parameters. The following list summarizes the knowledge gathered during these experiments as a list of guidelines:

- In the context of online robot learning in changing environments, providing information to help the user understand the validity of the robot's learned behavior is very important for the user and to the whole system performance.
- Not every type of information is beneficial. The simplistic thought that the more information is provided, the better will be the performance is false. Too much information, even perfectly accurate, can degrade the performance.
- The best way to inform users of changes in the environment is brief and contextualized notifications.
- Giving the ability to the user to see the future actions of the automation gained from learning is disturbing and counter-productive.
- The sensitivity of these results to the number of changes tested appears to be limited.
- In the context of an automated system with changing characteristics, providing feedback about changes in the characteristics in the form of notifications or continuous information is beneficial.
- The user adaptation to changes is slightly faster when the feedback is provided in the form of continuous information.
- Not informing the user about changes of characteristics of the system leads to situations of under or over trust directly impacting the performance.
- Previous experience has an impact on the user response to new changes of system characteristics: users who experienced a positive change in terms of performance before have more trouble detecting and reacting properly to new negative changes.

- Users are relatively robust to misleading or fake information about changes, they are able to discard them and react in the same way as if they received no information, and not worst.
- In the context of online robot learning in a changing environment, for which the users receive notification when it changes, the level of automation of the learning of the robot has an important effect on the performance of the task learnt, on the way users make use of the learning and on the performance of a parallel task.
- Depending on the level of automation at which the learning is applied and the situation, performance with learning can be worse than without learning. In particular, applying the learning in form of suggestions or approvable suggestions presents very little to no advantages when compared to no learning.
- Users make the best use of a learning robot when they can use it in the form of switchable automation, i.e. when they can switch between fully manual and fully autonomous.

This list of guidelines constitutes a basis for the design of future human-robot interfaces for learning robots, but there is still a lot to explore in this area. Results need to be generalized to more application scenarios and different learning algorithms. The reflection on the level of autonomy of the learning needs to be extended to more learning implementations modes, a more adaptive control than the basic switchable automation tested in this research should be achievable. Moreover, the level of autonomy at which the system is learning, i.e. the control the user has on what the system learns and when, should be considered, even if the resulting complexity would be probably hard to handle for the user. In this research learning was always activated, the user had no control on this aspect.

Part B: Advanced human robot interfaces

This research focused on the creation and development of novel approaches for making robots more usable for end users. A person following platform and accompanying algorithms were developed. Then a pointing control interface building upon the person following platform was designed. Next a direct physical interface enabling the control of a mobile robot by directly pushing it was created. All these 3 interfaces were then compared in set of user experiments. It turns out that the direct physical interface is the easiest and most intuitive one to use, but if a contact less interface is required, the following interface is the best. The pointing interface, even if it is appealing in terms of novelty revealed to be harder to use.

Additionally, a novel interface for remote robot teleoperation was created and tested: using the operator head movements to control the orientation of the camera of the distant robot. This interface proved to be more intuitive for novice users, however as the user practice of the system progressed, the more classical control interface based on joysticks showed better performance.

From these experiments it emerges that new sensor technologies permits the development of a lot of new human-robot interaction modalities and offer different creative approaches for interface designers. Overall we observe that robot control by body movements (e.g. person following, pointing control or head control of camera orientation) presents the advantage of being appeling for novices and more intuitive for users that have no experience. Moreover they are contact free which can be a requirement in some specific application (e.g. sterile medical environments). However, we noted that more classical interfaces are still better in terms of performance if the user has enough time to practice and get familiar with the system. Future work should focus on developing interfaces that are intuitive but also offer a margin of progression big enough to reach with time high levels of performance. Additionally, we tested a direct physical interaction modality which appeared promising both in terms of intuitivity and performance, but the feasibility of its implementation in more general applications needs to be investigated.

Keywords: Human-Robot Interaction; Interface Design; Level of Control; Robot Learning; Natural Interfaces; Warning Systems; Performance Evaluation.

This thesis is in part based on the following publications:

Refereed Journal Papers

Doisy, G., Meyer, J., Edan, Y.2014. The Impact of Human-Robot Interface Design on the Use of a Learning Robot System. *IEEE Transactions on Human-Machine Systems.* Volume 44, Issue 6.

Submitted Refereed Journal Papers

Doisy, G., Ronen, A., Edan, Y. Non-Invasive Robot Camera Head Control for Teleoperation: Performance and Workload Assessment. *Applied Ergonomics.* Corresponding to Chapter 8.

Jevtic, A., Doisy, G., Parmet, Y., Edan, Y. 2014. Comparison of Novel Interfaces for Mobile Indoor Robot Control: Direct Physical Interaction, Person Following and Pointing Control. *IEEE Transactions on Human-Machine Systems.* Corresponding to Chapter 7.

Journal Papers to be Submitted

Doisy, G., Meyer, J.,Edan, Responses to Warnings and the Effect of Feedback in a Simulated Robot-Control Task. Corresponding to Chapter 4.

Doisy, G., Meyer, J.,Edan, Y. The effect of the level of automation of the learning on the use of learning robot system. Corresponding to Chapter 5.

Peer Reviewed Conference Papers:

A. Jevtić, G. Doisy, S. Bodiroža, Y. Edan, and V. V. Hafner, 2014, “Human-robot interaction through 3D vision and force control,” in *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction - HRI ’14*, 2014, pp. 102–102.

G. Doisy and J. Meyer, 2013, “Responses to Warnings and the Effect of Notifications in a Simulated Robot-Control Task,” in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1594–1598.

S. Bodiroza, G. Doisy, and V. V. Hafner, 2013, “Position-invariant, real-time gesture recognition based on dynamic time warping,” in *Proceedings of the 2013 ACM/IEEE International Conference on Human-Robot Interaction - HRI ’13*, pp. 87–88.

Doisy, G., Jevtić, A., Bodiroža, S. 2013. “Spatially unconstrained, gesture-based human-robot interaction”, in *Proceedings of the 2013 ACM/IEEE International Conference on Human-Robot Interaction - HRI ’13*, pp. 117-118.

Doisy, G. 2012. “Sensorless collision detection and control by physical interaction for wheeled mobile robots”, in *Proceedings of the 2012 ACM/IEEE International Conference on Human-Robot Interaction - HRI ’12*, pp. 121-122.

Doisy, G., Jevtic, A., Lucet, E.,Edan, Y. 2012. Adaptive person-following algorithm based on depth images and mapping. In *Proceedings of the IROS Workshop on Robot Motion Planning*. (pp. 43-48).

G. Doisy and J. Meyer, 2011, “Expectations regarding the interaction with a learning robotic system,” in *Proceedings of the Workshop of the 2011 HRI Conference on Expectations in Intuitive Human-robot Interaction*, 2011, pp. 41–44.

Doisy, G., Meyer, J.,Edan, Y. 2011. Toward an ecological approach to interface design for teaching robots. In *Towards Autonomous Robotic Systems* (pp. 420-421). Springer Berlin Heidelberg.

Conference Publications:

Doisy, G., Meyer, J, Edan, Y. 2013. Interface Design for Learning Robots, *4th Israeli Conference on Robotics*, Tel Aviv, Israel. Abstract.

Doisy, G., Jevtić, A., Bodiroža, S.,Edan, Y. 2013. Spatially Unconstrained Natural Interface for Controlling a Mobile Robot, *4th Israeli Conference on Robotics*, Tel Aviv, Israel. Abstract.

1. Introduction

1.1. Description of the problem

The use of automation, and more specifically the use of robots, is increasing in industry and recently also in people’s daily life activities. The presence of personal robotics in our homes should lead to important societal changes in upcoming years (Gates 2007), with personal robots assisting our aging societies in health care, therapy and rehabilitation but also serving as entertainers and household staff (Fong et al. 2003).

However, current robotic technology is still limited. Fully autonomous robots, capable of performing new tasks in complex and unstructured, unknown and changing environments like our homes do not exist so far. Humans are highly flexible and can easily adapt to changing conditions (Rodriguez & Weisbin 2003), but they are far less accurate and reliable compared to robots. Hence, it is advantageous for robots and humans to collaborate, with each benefitting from the specific capabilities of the other. A critical component of successful human-robot collaboration is the interface, and this is particularly problematic for learning robots as pointed out later.

This thesis focuses on two research directions related to interface design aiming to increase the efficiency of human-robot collaboration: 1) understanding the design of the human-robot interface for robots with learning capabilities and 2) developing and testing novel intuitive human-robot interfaces based on state-of-the-art technological advances.

A. Human-Robot interface design for online learning robots

A robot learning a behavior by interacting with a human operator is a special case of human-robot collaboration. There are two main applications of human-robot learning that must be distinguished since they have different consequences on the interaction. In the first robot learning application, the learning occurs during the development process of the robot. Through learning, the robot's capabilities are extended before it is used by the end-user. The learning or training is done by the developers of the robot, hence they have an extensive understanding of what the robot is learning and how learning is achieved. They generally have enough time to test and edit what the robot has learned before the desired behavior is reached, but once the development is finished, the robot's behavior is fixed, and no additional learning occurs. Therefore, in this case, users will not experience interaction with the learning process, and there is no need to consider additional interaction issues, compared to those that exist with a robot that does not learn.

In the second application of robot learning, the robot learns from the end-user while performing tasks. Here learning takes place in the real world, and it is used on-line to continuously increase the robot's autonomy and adaptive capabilities. The user usually has no knowledge about the internal processes in the robot, and it is generally difficult to test and to modify what the robot has learned. This raises several issues about the user interaction with the robot and the learning process.

It has been shown that when users teach a robot, it is unlikely that they properly understand what the robot is learning, even for simple tasks (Saunders et al. 2007). Hence, the user won't be able to predict how the robot will react, because its behavior becomes inconsistent as it is modified by the learning process. This leads to a mismatch between the user expectations about the robot and the robot's actual behavior. This is caused by the lack of feedback available during the learning process: the user can compare his or her expectations and the result of the teaching only after the teaching is complete and applied for the first time. Eventually this mismatch may lead to a loss of trust in high levels of automation of the robot, and the user may choose to operate at a lower level of automation which will reduce the overall performance. Thus, the feedback must support proper user expectations, and it should be provided continuously by the interface. The interface should be able to communicate to the user sufficient information for the user to understand the constraints of the learning process.

The interface between the robot and the human can shape the way the user acts on the robot and its learning process. Moreover, it can provide all the relevant information about the current state of the robot, of the learning process and of the surrounding environment. Hence, the interface has the potential to overcome the understanding and expectation issues arising from the learning capabilities of the robot. But this requires solving a complex problem: how should the human-robot interface be designed to support robot learning? What information should be provided, and when and how should this information be presented? Providing too much information and too many interaction options will confuse the user. Providing too little information and too few interaction options won't allow the user to understand the robot behavior or to deal with complex situations. When users have too little information and too few interaction options they will not be able to understand the robot behavior or to deal with complex situations, such as changes in the environmental conditions. Even with the correct information and the proper interaction options the problem is not solved, since the way the information and the interaction options are presented can influence greatly how they are perceived by the user.

B. Advanced Human-Robot interfaces

High-end service robots of today already have the capacity and ability to perform meaningful tasks in several application domains. Despite this fact, very few robots of this kind can be found in real commercial settings. One of the reasons for this limited use is the complexity of the human-robot interaction for non-expert users (Rouanet et al. 2013). Recent technological development led to the release of cheap and efficient sensors, such as the Microsoft Kinect with new algorithms for human body segmentation (Schwarz et al. 2012; Shotton et al. 2011), dramatically extending the perception capabilites of robots. The use of this kind of sensor was quickly accepted by the scientific community and influenced a wide range of application domains such as object detection (Camplani & Salgado 2012), person tracking (Matthias Luber et al. 2011) , SLAM (Endres et al. 2012), 3D surface reconstruction (Izadi et al. 2011), and gesture recognition (Suarez & Murphy 2012), among others. Additionally, new algorithms and hardware are emerging in the field of physical human-robot interaction. Direct physical interaction (DPI), also known as physical human-robot interaction (pHRI), allows the user to influence the robot behavior through a physical contact with it (Chen & Kemp 2010). Originaly intended for safety, the maturation of these techniques permits the creation of robots controllable directly by moving them physically.

These new techniques potentially enable the creation of radicaly new natural interfaces solving the issue of the complexity of the usage of robots by non-experts by creating a seamless interaction between the human and the machine. However, little is known about their usability, how to design a functional interface with them and how they perform in the long run compared to classic interfaces. New concepts and ideas need to be developed and their usability tested to advance toward the interfaces of the future.

1.2. Research Objectives

The objective of this research is to enhance human-robot collaboration by studying different aspects of the interface design. Two directions are proposed to reach this goal:

A. Understanding the effect of parameters of the human-robot interface on systems with online robot learning of new unknown tasks, when the robot is controlled by non-expert users. This problem is expected to depend on characteristics of the learning process, the task, the user and the environmental constraints. More specifically, this research focuses on understanding the interaction with the learning process, and in particular the level of control the user should have on the learning process and how this control should be presented to the user; and the kind of information, or feedback, the user should receive from the interface during the learning process.

B. Developing and testing the usability of novel natural interfaces based on the latest technological developments in the fields of sensors, robot perception and control, in real world conditions.

1.3. Research Significance

Robot learning could greatly improve human robot collaboration in everyday life and can increase autonomy and flexibility. However, the complexity of current robotic learning algorithms does not allow end users to fully understand how to use them. For this reason advanced learning techniques are most of the time manipulated by developers or expert users and if not, are often not optimally used. To introduce learning robotic systems into everyday life, robot learning must be transparent so that end users can use and understand it.

A critical component of successful human-robot collaboration is the interface, and this is particularly problematic for learning robots. We state that an improved interface can improve learning and can enable better, faster and easier implementation of more complex tasks. Moreover, the following critical specific issues in real world robot learning situations can be overcome with an appropriate interface:

1. The limited possibility to test the learned behavior,
2. The difficulty for the user to understand the learning process, and
3. The risk of a mismatch between user expectations and robot behavior.

Alternatively, advanced natural human-robot interfaces can facilitate the usage of robots for non-expert users and help to increase their acceptance in human society. Designing robots that are easier to understand and to control will enable future personal robots to become as common and trivial to use as today’s smartphones. New technologies, algorithms and sensors promise to enable this paradigm change. However, novel natural interfaces emerging from new technologies are not necessarily more intuitive and better to use than classical ones. There is a need to characterize their usability, their advantages and their drawbacks and to analyze for which situation they are best suited.

1.4.Research Contribution and Innovations

This dissertation presents one of the first attempt to systematically study the interface design for learning robots from a user perspective. Several experimental platforms, in chapters 3 to 5 of part A, using either simulations or real robots, were developed to conduct user studies and evaluate the effect of the parameters influencing the human-robot interaction. From the results of the experiments and their analysis, we extracted the first set of guidelines on how to design an interface for users that interact with a robot with learning capabilities.

Additionally, in the second part (Part B, chapters 6 to 8), we developed four novel innovative interface modalities to control mobile robots and evaluated their usability:

- A person following interface that enables a robot to robustly follow a person in complex environments.
- A pointing interface that permits the control of a robot by pointing to a desired location.
- A direct physical interface that allows the control of a robot by simply pushing it.
- A non-invasive robot camera head control interface for robot teleoperation.

2. Scientific Background

2.1. Levels of automation

The different levels at which a human operator can control an automatic process are defined and classified as levels of automation (LOA). Several taxonomies have been proposed of LOAs (Endsley 1987; Kaber & Endsley 2004; Sheridan & Verplank 1978) with the most common taxonomy defining ten levels from fully manual to fully autonomous with intermediate levels (Sheridan & Verplank 1978).

Online robot learning can be seen as a way to increase robot autonomy. It raises the question of the level of automation at which the robot should work during and after learning. Moreover, the learning process itself can be viewed as a different automatic process than the control of the robot. Thus, the level of automation at which the robot is learning should also be considered.

Human collaboration implies that an automatic process is not perfectly autonomous and requires human intervention. As long as human collaboration is needed, working at high levels of automation leads to human-out-of–the-loop (OOTL) issues. These issues cause delays in the detection of system errors (Wickens & Hollands 1999; Wiener & Curry 1980), longer system recovery times and poor response accuracies (Wickens & Kessel 1977). In (Endsley & Kiris 1995), the authors identify three major mechanisms causing the lack of situational awareness responsible for the human OOTL issues:

1. Changes in vigilance and complacency associated with monitoring,
2. Assumption of a passive role instead of an active role in controlling the system,
3. Changes in the quality or form of feedback provided to the human operator.

To solve these issues, research focused on human-centered automation (Billings 1997; Sheridan 2006). The idea is to keep the human in the loop by using intermediate LOAs. However, which LOA is the best to use is a complex question. It has been showed that this depends on the task, the environment, the user and the system characteristics (Bechar et al. 2009). Another approach to human centered automation is the area of adaptive automation (AA) or dynamic function allocation (DFA), where the control is dynamically allocated between the human and the machine. Different AA strategies to switch the control exist:

1. Critical events: switches are triggered by the occurrence of events critically impacting system goals (e.g., malfunction (Hilburn et al. 1993)).
2. Performance measures: switches are triggered by degradations in human performance below a criterion measure (Parasuraman et al. 1993; Tkach et al. 2011).
3. Psycho-physiological assessment: real-time assessment of operator workload (using, for example, physiological measures like electroencephalogram (EEG) signals or heart-rate variability) are used to decide when to switch (Byrne & Parasuraman 1996; Pope et al. 1995).
4. Behavior modeling: switches occur to achieve predetermined patterns of overall system functioning(Rouse et al. 1986).

2.2.Adaptive user interfaces

Adaptive User interfaces (AUI) are defined in (Rothrock et al. 2002) as systems that autonomously adapt their displays and available actions to the user's current goals and abilities by monitoring the user status, the system task, and the current situation. In case of online robot learning, the possible actions the robot can perform vary with the state of the learning process. In order to be accessible to the user, these learned actions should be available through the interface. For instance, in the case of a robot learning online how to navigate in a house, the interface should adapt and expose the locations learned, e.g., once the robot has been driven to the kitchen, the interface should let the user select the action “Go to the kitchen”. Therefore, user interfaces for learning robotic systems can be seen as adaptive user interfaces. Hence, user interfaces for robot learning can benefit from the research done on AUIs.

An important point in AUI research is to understand when adaptivity should be implemented. Translated to the perspective of our user interfaces for learning robots, this point could be to understand when the learned actions should be implemented. Most studies in AUI compare different adaptation methods and conclude on their performance (Findlater & McGrenere 2004; Findlater & McGrenere 2008; Gajos et al. 2006; Tsandilas & Schraefel 2005). However in (Gajos et al. 2006), the authors tried to understand what affects the success or failure of an AUI. They found that predictive accuracy and frequency of adaptation impact user performance. Research on the benefits and costs of adaptive user interfaces (Lavie & Meyer 2010)identified four additional factors impacting performance:

1. Level of adaptivity. The level of adaptivity is the level of automation at which the interface adapts or changes according to the situation, from no adaptation to full adaptation with intermediate mixed-initiative levels. The level of adaptivity of a system interface is different from the level of automation at which the system operates. For instance a system operating at a high level of automation, i.e. almost autonomously, could have an interface with a low level of adaptivity, i.e., the interface stays the same over time and doesn't change according to the situation. The impact on performance of the level of adaptivity depends on the accuracy of the adaptation. As long as the system doesn't need user interaction, full adaptation leads to the best performance. But in case of unfamiliar situations which require the user to override the system, the mixed-initiative levels show better performance.
2. Proportion of routine versus non-routine situations (such as tasks that must be performed). Non-routine situations cause the adaptation to fail and can lead to bad suggestions or bad automation performance according to the level of adaptivity (respectively mixed-initiative or full adaptation). The higher the proportion of non-routine situations, the lower will be the benefits from adaptation, and adaption may actually even have greater costs than benefits.
3. User characteristics. Different user groups may benefit more or less from adaptation. For instance, the performance of older people benefits more from adaptation when the adaptation is correct (for routine tasks) but their performance decreases faster when the proportion of non-routine task increase.
4. Task characteristics. The more difficult a task is for the user, the more the user can benefit from adaptation.

2.3. Interfaces and robot learning

Robot learning through interaction with humans requires efficient machine learning algorithms and clear, intuitive human-robot interaction systems. Robot machine learning algorithms have been the subject of important research, notably on imitation learning, learning by demonstration and social learning (Argall et al. 2009; Calinon et al. 2007; Nehaniv & Dautenhahn 2007; Thomaz & Breazeal 2007; Thomaz & Breazeal 2008). Despite extensive research in human-robot interaction (Fong et al. 2001; Scholtz 2003; Yanco et al. 2004), the interaction with learning robots remains largely unexplored. Yet, understanding the human teacher/robotic student relationship is of great importance (Thomaz & Breazeal 2008), and the interaction scenario is an important part of the system design (Calinon & Billard 2007). Furthermore (Saunders et al. 2007) showed that when humans teach a robot, it is unlikely that they understand what the robot is learning, even for simple tasks.

This lack of understanding of what the robot is learning is a major usability issue when robot learning is done online. With robot learning it is difficult for the user to predict how the robot will react, because the robot behavior becomes inconsistent as it is modified by the learning process. There is then a mismatch between the user expectations about the robot and the robot's actual behavior. Eventually this mismatch may lead to a loss of trust in the robot, and to a misuse of automation which will reduce the overall performance of the system (Dzindolet et al. 2003; Lee & See 2004). In a stable human-machine system, operators' trust and expectations converge toward stable and close to optimal values when the operator gains experience with the system (Madhavan & Wiegmann 2007; Bailey & Scerbo 2007). However, when the system is learning and the automated process change over time, the user cannot see immediately the result of his teaching actions: the user can compare his or her expectations and the result of the teaching only after the teaching is complete and applied for the first time. Thus, a feedback system must support proper user expectations, and it should be provided by the interface. The interface should communicate sufficient information for the user to understand the constraints of the learning process.

In (Rouanet et al. 2013), the authors also identified the interface and its feedback as critical and argued that their importance is paramount when it comes to deploying learning robot systems outside the laboratory for the use by non-expert users. They studied the impact of various interfaces on the efficiency of robot learning (Rouanet & Danieau 2011; Rouanet et al. 2009; Rouanet et al. 2010; Rouanet et al. 2013). In the context of teaching a robot to recognize new visual objects, they demonstrated the superiority of an artifact-based interface which uses a mediator object (here a smartphone) to show what the robot perceives and hence what it is actually learning. Similarly, in a navigation teaching task, Crick et al. (Crick et al. 2011) showed that to understand the learning process, the user should be able to see the world through the eyes of the robot. In other words, the interface should explicitly show what the robot understands from its sensors and its algorithms. With this goal in mind, gesture based interfaces (Chaudhary et al. 2011; Mitra & Acharya 2007), as demonstrated in (Rouanet et al. 2013), or voice interfaces (Cohen et al. 2004; Pieraccini 2012), are inferior because they cannot help the user resolve the ambiguity about what the robot perceives. Hence, we claim that screen-based interfaces are currently the most suitable interfaces to study the interaction with a learning robot, because the interaction with the user can be accurately constrained and controlled. These interfaces can also display interactive relationships between environmental features and computational processes, and they can display an enhanced reality in the form of virtual reality or augmented reality. Augmented reality interfaces which are not screen-based, like projective augmented reality interfaces, should provide the same advantages (Krevelen & Poelman 2010), but the technology is not mature yet, is not flexible enough to be considered for use and is expensive.

2.4. Warning systems, Trust, Reliance and Compliance

Warning systems are simple but wildly used in numerous application such as security alarms, proximity or collision avoidance alert in aviation or monitoring systems in health care. In such systems, it is possible to distinguish between two dimension of trust (Meyer 2001; Meyer 2004): compliance, referring to the operator responding as if there was a hazard when a warning is displayed; and reliance, referring to the operator responding as if the system was safe when no warning is displayed. The independence of reliance and compliance responses has been the subject of some recent research (Dixon & Wickens 2006; Dixon et al. 2007; Rice 2009; Bahner et al. 2008; Rice & McCarley 2011; Wiczorek et al. 2012; Meyer et al. 2013). In (Meyer 2001) different independent variables affected compliance and reliance differently: only reliance changed over time whereas compliance remained constant; and only compliance was affected by the display form. In (Meyer 2004) the author provided an analysis demonstrating that for rational decision makers compliance and reliance should be independent. In (Dixon et al. 2007) and (Rice & McCarley 2011) the authors showed an asymmetrical effect of automation false alarms and automation misses on compliance and reliance. Finally, in (Wiczorek et al. 2012) and in (Meyer et al. 2013), the authors showed results supporting the fact that compliance and reliance should be distinguished, but that these two dimensions of trust are not entirely independent.

2.5.Direct Physical Interaction

Direct physical interaction (DPI), also known as physical human-robot interaction (pHRI), allows the user to influence the robot behavior through a physical contact with it (Chen & Kemp 2010). This specific form of human-robot interaction (HRI) has been applied for different purposes.

Originally, the primary application of physical human-robot interaction was safety, in order for a robot and a human to share the same workspace without the risk of traumatic injury. For example, comparison of the force generated by the robot's actuators with the values predicted by a dynamic model of the robot allows detection of the force created from a physical contact (i.e. a disturbance in the model), as shown by a case of interaction with a robotic arm in (De Luca & Mattone 2005; De Luca & Mattone 2004). Also, in (Haddadin et al. 2008; De Luca et al. 2006) a robotic arm is capable of detecting a collision and stopping its motion in order to prevent an injury to the human operator or a damage to the robot. The same authors further show that this method can be applied on the same hardware to prevent soft tissue injuries such as human skin cuts caused by a knife or a similar sharp tool, which is manipulated by the robotic arm (Haddadin et al. 2010). Furthermore, DPI has also been used in manipulation of robotic arms, generally through impedance control (Hogan 1984), which enabled the exploration of new natural techniques for human-robot contact (Hale & Pollick 2005), human-robot cooperation (Ikeura & Inooka 1995), object transfer (Edsinger & Kemp 2007), teleoperation (Love & Book 2004), or kinesthetic learning by imitation (Mulling et al. 2013).

But the use of DPI is not limited to robotics arms and its benefits have been explored in manipulation and navigation tasks of mobile robots proving it to be more intuitive than classical gamepad interfaces (Chen & Kemp 2011). State-of-the-art methods propose the use of external force sensors or torque sensors mounted on the robot (Takeda et al. 2007), or compliant joints located on the robot upper part in order to measure the contact forces, as it has been done with the robots like Cody (Chen & Kemp 2011; Chen & Kemp 2010; Jain & Kemp 2009), PR2 (Wyrobek et al. 2008), Justin (Fuchs et al. 2009) and IRL-1 (Ferland et al. 2013). However, these methods limit the surface on the robot body where physical interaction can take place. Though, in (Doisy 2012), the author presents a proof of concept in which an indoor mobile robot can be controlled without the need of external sensors, by applying force to any part of the robot body.

2.6. Person Following

The idea of a person-following robot is not new and it has been applied to robot companions, smart shopping carts, transporters, walking assistants, and so forth. Two challenging tasks constitute the person-following behavior, namely robot navigation and person tracking. While the robot navigation has been thoroughly researched so far (Kruse et al. 2013), the lack of affordable and powerful sensors and efficient person-tracking techniques limited its application to person following.

While some rely on smart environments (Najmaei & Kermani 2011), most object and person detection and tracking methods use vision-based techniques (Jia et al. 2009; Moeslund et al. 2006); however, they are sensitive to illumination changes that can degrade segmentation results (Liu & Fujimura 2004). Laser rangefinders (LRFs) provide accurate distance measurements and they are generally used to detect the legs of the person (Martinez-Otzeta et al. 2009; Alvarez-Santos et al. 2012). But the legs can easily be confused with tables and chairs, so they must be filtered out by mapping the environment. Some authors propose filtering (Schulz et al. 2003; Gu & Veloso 2009) or sensor fusion techniques (Kobilarov et al. 2006; Bellotto & Hu 2009; Spinello et al. 2010; Motai et al. 2012) in order to improve tracking performance. The use of stereo vision cameras for person tracking has also been reported in literature (Satake & Miura 2010), in combination with LRF (Martínez-Otzeta et al. 2010), or LRF and color-image segmentation (Yoshimi et al. 2006; Calisi et al. 2007).

Recent release of affordable depth sensors for indoor applications such as Microsoft Kinect led to development of new algorithms for human body segmentation (Schwarz et al. 2012; Shotton et al. 2011). The Kinect can provide depth images at the rate of 30 fps allowing real-time object segmentation, which is based on distance gradient and insensitive to variable lighting conditions. This technology was quickly accepted by the scientific community and influenced a wide range of application domains such as object detection (Camplani & Salgado 2012), person tracking (M Luber et al. 2011), SLAM (Endres et al. 2012), 3D surface reconstruction (Izadi et al. 2011), and gesture recognition (Bodiroza et al. 2013), among others.

The depth sensors simplify the problem of indoor person tracking and allow development of more efficient person-following robots (Pradhan 2013; Pucci et al. 2013).

2.7.Pointing control

Pointing is recognized as one of the most intuitive gestures for indicating a location or an object of interest (Wachs et al. 2011). The idea of using this gesture for robot control appeared early (Cipolla & Hollinghurst 1996), and also proved to be an accepted way of interaction for specific category of lay users such as the older adults (Beer et al. 2012). Various combinations of sensors and algorithms can be used to track the pointing motion. Smart devices that are manipulated by the user proved to provide accurate tracking results, as shown in the case of laser pointers (Kemp et al. 2008; Suzuki et al. 2005; Beer et al. 2012; Shibata et al. 2011), a mobile phone (Koceski et al. 2012) or other devices that were specifically developed for this application such as the XWand(Wilson & Shafer 2003) or the WorldCursor (Wilson & Shafer 2003). However, the need to hold or wear a device to perform pointing is neither practical nor intuitive.

Extensive research in image and video processing for pointing gesture recognition has been done with systems comprising of one or more cameras (Wren et al. 1997; Azarbayejani & Pentland 1996; Darrell et al. 2000; Jojic et al. 2000; Cipolla et al. 1994; Nickel & Stiefelhagen 2007). The proposed techniques lack accuracy and fixed-cameras systems spatial constraints limit the application in a mobile-robot scenario. As for the person-following interface, the pointing interface can benefit from the skeleton-tracking ability of the depth sensors. The novel algorithms for human-body segmentation (Schwarz et al. 2012; Shotton et al. 2011) provide improved speed and accuracy and have already shown benefits for hand tracking, arm tracking, posture recognition (Diego-Mas & Alcaide-Marzal 2014) and pointing recognition (Suarez & Murphy 2012).

The pointing target is derived from the position of the arm and hand joints using their 3D location obtained from a depth sensor input. Various combinations of joints shown to provide good results in pointing target recognition such as the hand and the elbow, the hand and the shoulder, or the hand and the head (Droeschel et al. 2011). Alternatively, in (Bodiroža et al. 2013) it was proposed to allow the robot to learn the relation between pointing gestures and control commands.

2.8. Interfaces for remote teleoperation of mobile robots

Remote control of robotic platforms through teleoperation has been used for years in many applications: telepresence(Jouppi 2002), exploration of remote and hostile locations (Burke et al. 2004), or augmentation of human perception and power (Turro et al. 2001). However, teleoperation is subject to important human factors issues (Chen et al. 2007; Murphy 2004; Tittle et al. 2002; Voshell et al. 2005). Notably, the experience of the user, the context of use of the robot, the visual information available and the interaction modality have strong effects on the user performance (Casper & Murphy 2003; Scholtz et al. 2004; Woods et al. 2004). Also, there is a strong influence of the operator's situational awareness (Drury et al. 2003; Yanco & Drury 2004), the operator's understanding of the robot's close and far surroundings. In a teleoperated system using the feedback of an embedded camera, the “keyhole” or “soda straw” effect is a major factor degrading situational awareness (Voshell et al. 2005; Casper & Murphy 2003).

Hence, significant effort has been put in finding a solution for the operator to make better sense and use of the robot camera. The limited field of view of the camera has quickly been identified as a major issue (Alfano & Michel 1990) and to overcome it some studies suggested the use of wide angle lens on the camera (Scribner & Gombash 1998; Eliav et al. 2011).

However, when receiving the video feedback of a camera equipped with a wide angle lens, users have been reported to over-estimate the speed of the objects in their surroundings and therefore to reduce their speed (Scribner & Gombash 1998). Additionally, the optical distortion caused by a wide angle can increase the risk of operator motion sickness as well as the cognitive workload to operate the robot (Chen et al. 2007). Alternatively, it has been suggested to use multiple cameras, like in (Keyes & Yanco 2006) where a rear-facing camera was added; or in (Voshell et al. 2005) where 5 cameras (one pointing straight and the four others pointing 45° in each direction: up, down, left, right) were used to create *Perspective Folding*. But with multiple cameras there is a risk of cognitive tunneling (Thomas & Wickens 2000)that occurs when the operator's attention is captured by a single camera output and the feedbacks from the other views are ignored. Another suggestion which was widely adopted is to allow the decoupled motion of the camera orientation from the movements of the robot (Hughes et al. 2003); i.e., to mount the robot camera on a pan-tilt (or pan alone) mechanism and to provide the operator with independent control of the mechanism. But this solution presents the risk of degrading further the operator's situational awareness instead of increasing it if not carefully used (Nielsen et al. 2005). It appeared that when using a controllable camera orientation, some users have trouble detecting when the camera is not aligned with the forward direction of the robot which can provoke dramatic collision and loss of the system (Drury et al. 2003). To overcome these problems of unnoticed misalignment of camera orientation two main strategies can be found in the literature.

The first approach, described in (Nielsen et al. 2007; Nielsen et al. 2005; Ricks et al. 2004), is a specific form of ecological design for teleoperation. The problem of misalignment is solved by constructing a 3D virtual exocentric view of the robot. Inside this view the output of the camera can be displayed according to the current orientation of the pan-tilt mechanism. Hence, the operator can directly visualize the position of the camera with respect to the robot, as illustrated in Figure 1. However, the process required to construct the 3D view is complex to set up and needs reliable localization in addition to an accurate range sensor, limiting its usage for simple robotic hardware. Nevertheless, this approach has proven to provide advantages in terms of performance and operator workload as compared to classical interfaces.

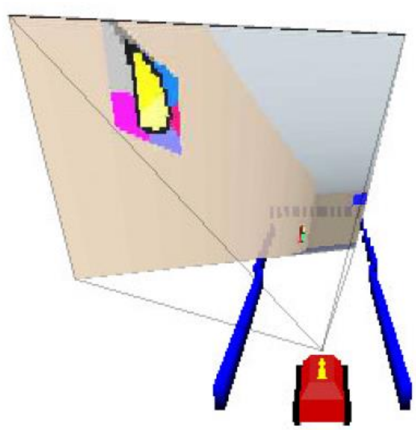

Figure 1: Visualizing the orientation of the pan-tilt camera using the 3D interface, from (Nielsen et al. 2007).

A second approach consists of controlling the orientation of the video feedback through the operator's head orientation: conscious of their head orientation, operators are then aware when there is a misalignment between the forward robot direction and the video feedback. In (Zalud 2006) for instance, the operator uses virtual reality goggles capable of tracking the head orientation, which is then used to control the pan-tilt mechanism. However, no comparison was made with classical camera orientation control. Moreover, the latency between the head orientation, the actual movement of the pan-tilt mechanism and the update of the video image in the head mounted display is likely to provoke motion sickness, discomfort and degradation of perceptual capabilities typical of virtual reality display with high latency (Mania et al. 2004; Allison et al. 2001). This latency issue was overcome by the use of a similar head tracking virtual reality goggles in conjunction with omnidirectional cameras, like in (Fiala 2005). The operators can then choose their video view orientation though their head movements with very little delay since there is no need to wait for a mechanical device to move and an image the be transmitted. However, head tracking virtual goggles and see-through techniques are promising but heavy to wear and exhausting for the operator and a significant minority of the population still cannot use them without experiencing motion sickness with the current state of the technology. Additionally, visual displacement caused by see-through systems deteriorates visio-motor performance due to sensory conflict (Biocca & Rolland 1998; Cobb 1999; Smyth 2000)and no user studies proved significant improvement of robot teleoperation performance with such systems compared to classical control modes.

Part A: Interface Design for Learning Robots

This section investigates different aspects related to interface design for learning robots. The focus was on how the interface design influences the user interaction with a robot with behavior which evolves over time in a changing environment. The research is organized in three user experiments in which different interaction conditions were tested and compared with the goal of extracting guidelines for future designers of human-robot interfaces. The first two experiments were conducted in two specially developed computer simulations, whereas the last one was conducted with a real mobile robot.

1) The first experiment focused on the impact of **the number of changes of environmental conditions** and **the type of feedback provided** about the learning behavior: the usefulness of warning the user about changes affecting the learning and of showing previews of the learned behavior was tested.
2) In the second experiment, a simplistic form of automation was used: **a binary warning system, but its characteristic (i.e. its sensitivity) varied over time** and the users' responses to these changes, and in particular the two dimensions of trust, compliance and reliance, were studied with different feedback conditions.
3) Finally, the third experiment, still in the context of a learning robot in a changing environment, looked at the **level of automation** at which the automation gained from the learning should be applied.

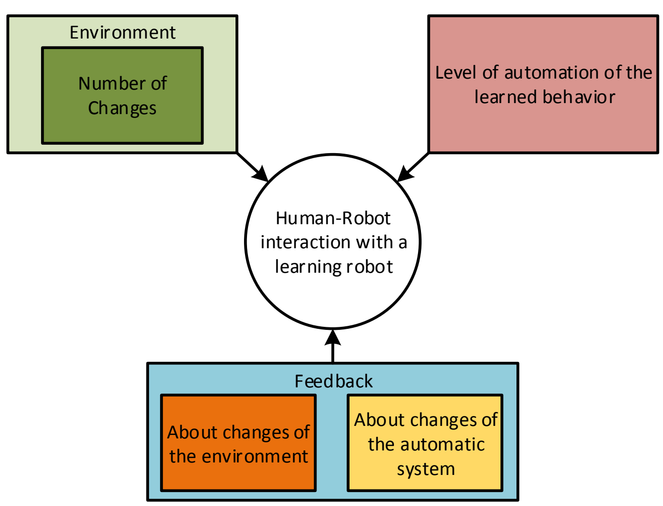

Figure 2: Tested parameters influencing the human-robot interaction with a learning robot

3. The effect of feedback and environmental changes on the use of a learning robot system

3.1. Overview

This experiment focuses on investigating the influence of different forms of feedback on the system (robot+user) performance in an environment where conditions can change over time. The hypothesis is that performance can be improved if users have a better understanding of the state of robot learning. We first looked at the effect of providing feedback about environmental changes in the form of notifications, expecting that it will help the user to adapt the learning of the robot to these changes and improve the performance of the learned behavior and the whole system. Then, we studied the effect of adding previews about the behavior the robot learned, given that there were notifications about changes, to see if users could make use of additional information, trust and use the learned behavior of the robot more and improve performance.

3.2. Methods

A. Participants

Participants were 42 undergraduate (15 females, 27 males) engineering students, aged from 23 to 29 years, without previous experience with the system. They were recruited using the mailing list of the engineering department. They received 30 NIS (New Israeli Shekels, about 8 dollars) for their participation in the experiment and could get a bonus of 100 NIS (about 26 dollars) according to their performance. The recipients of the bonus were determined with a lottery, with each point in the cumulative score serving as a lottery ticket. Thus, the higher a subjects' cumulative score in the two sessions, the more virtual lottery tickets they received. The subjects received instructions on how to use the system in the form of an interactive tutorial.

B. Experimental platform

To study the impact of the interface design in a controlled environment, we developed an experimental platform in the form of a video game software simulating a robotic waiter task. The software was developed in C# and WPF for PC. It was run on each station of a computer laboratory which allowed conducting the experiment with up to 10 participants in parallel.

C. Task

Participants were asked to play the "game" and to finish with the highest score. They conducted two tasks in parallel. The left side of the screen displays the main task and its related controls, and the right side displays the secondary task(Figure 3). The bottom right part shows the remaining time (time left) before the end of the session and the score (computed from the performance in both tasks). A session of the experiment lasts 18 minutes. The initial score is 100.

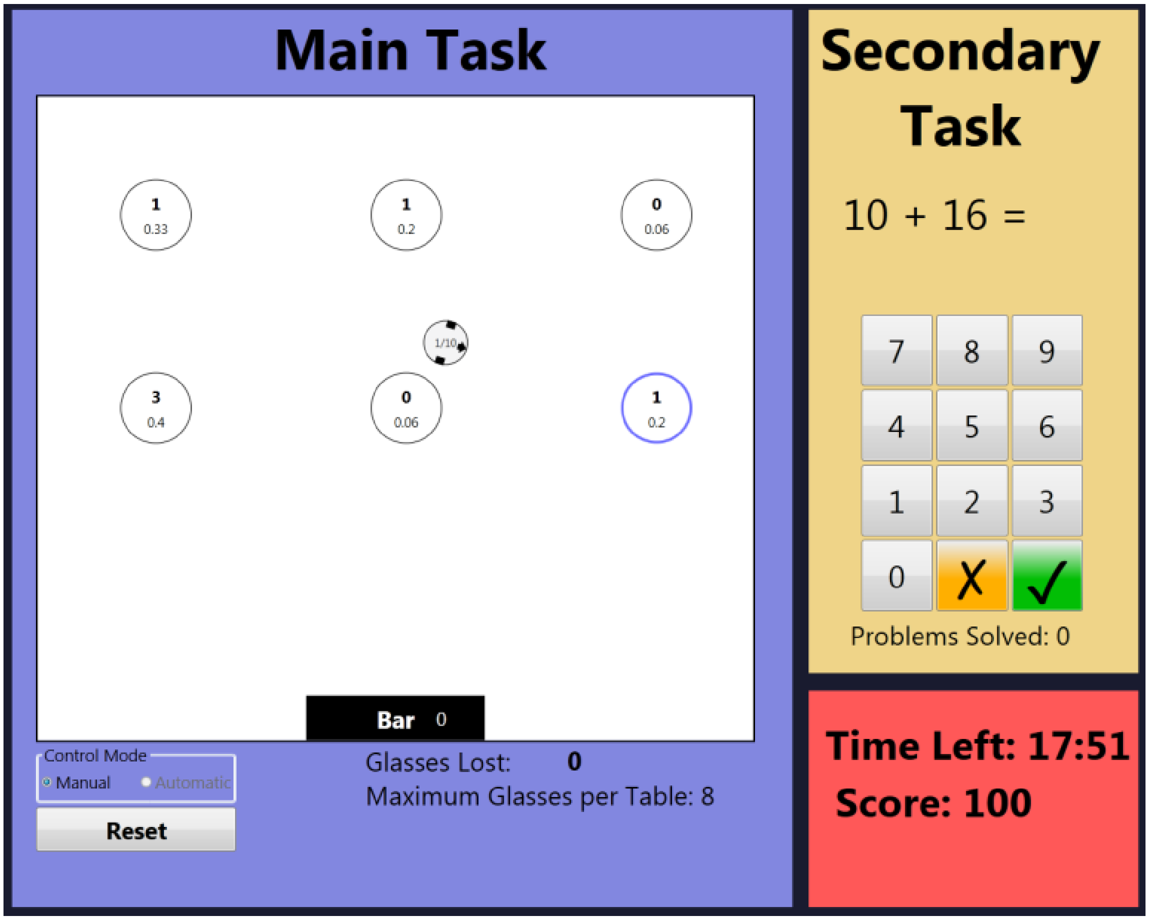

Figure 3: Experimental platform.

The scenario for the main task is to control a robot waiter that is able to learn. The robot collects empty glasses which are left by customers on each of the 6 tables of the restaurant. The tables and the robot tray have limited capacities: a table can hold a maximum of 8 drinks and the robot tray cannot carry more than 10 glasses. The robot brings empty glasses to the bar to avoid a situation when a table is full, so that no additional glasses can be deposited. This is considered an undesirable event because one score point is lost each time an empty glass cannot be added to a table. The number of glasses increases on the tables at a specific rate, which may differ between tables. The upper, bold number in each table is the current number of empty glasses, and the bottom number is the current rate of arrival of empty glasses (i.e. new empty glasses per second); on the robot a counter displays the number of glasses on the tray and its capacity and an alert message appears when it is full (seeFigure 4).

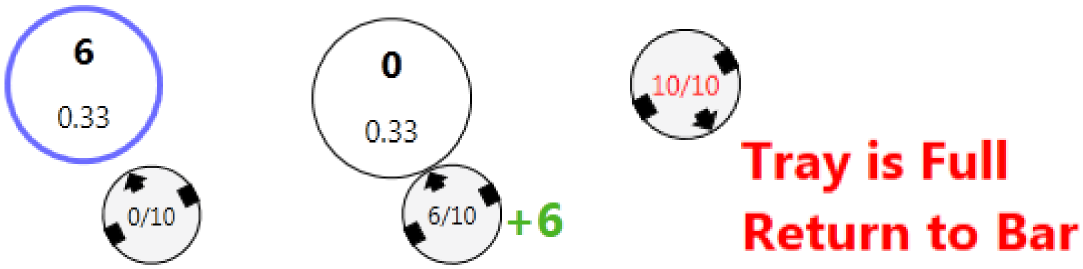

Figure 4: Robot and a table before (left) and after (midle) a transfer of drinks. Robot tray is full alert message (right).

In the manual mode the user controls the robot destination, one of the six tables or the bar, by clicking on it. A blue circle indicates a chosen destination (Figure 4). If it is a table, glasses are transferred, as indicated by an animation (Figure 4), to the robot tray until all glasses are transferred or the robot tray is full. When the robot reaches the bar, the robot tray is emptied.

In the manual mode the robot learns at which frequency the user visits the tables. Thus, the robot does not learn the sequence at which tables are visited, but rather the relative frequency of visits to each table, i.e. how often on average the user moves the robot to each table compared to the other tables. Note that this is not equivalent to the number of visits at a table per time unit. The user can choose to reset what the robot has learnt at any time. After two minutes of use of the manual mode without a reset, the automatic mode becomes available, and a notification of its availability is displayed. From then on the user can choose to switch to the automatic mode.

In the automatic mode, the robot automatically visits each table according to the frequency it has been visiting the different tables under the control of the user. At the moment the automatic mode is switched on, a sequence is generated from the visit count of each table during learning. This sequence reproduces the relative frequency at which each table was visited and spaces two visits to the same table as regularly as possible. The sequence is then repeated in a loop. When the robot tray is full, the sequence is paused, the robot automatically returns to the bar to empty the tray, and then resumes the sequence. The user can choose to switch back to the manual mode at any time. After a switch to the manual mode the robot continues to learn the frequency at which the tables are visited. Also, if the user chooses to reset what the robot has learnt, the robot will learn from scratch which table to visit, but the user will have to wait again 2 minutes to be able to use the automatic mode.

In the secondary task, the participant had to add two two-digit numbers. They gained one score point for each correct answer and a wrong answer had no effect. The secondary task was designed as to create an incentive for the user to teach the robot properly and to use the automatic mode: because the participants could respond to the secondary task only by using the mouse, this task was much easier when the robot was in the automatic mode.

D. Design

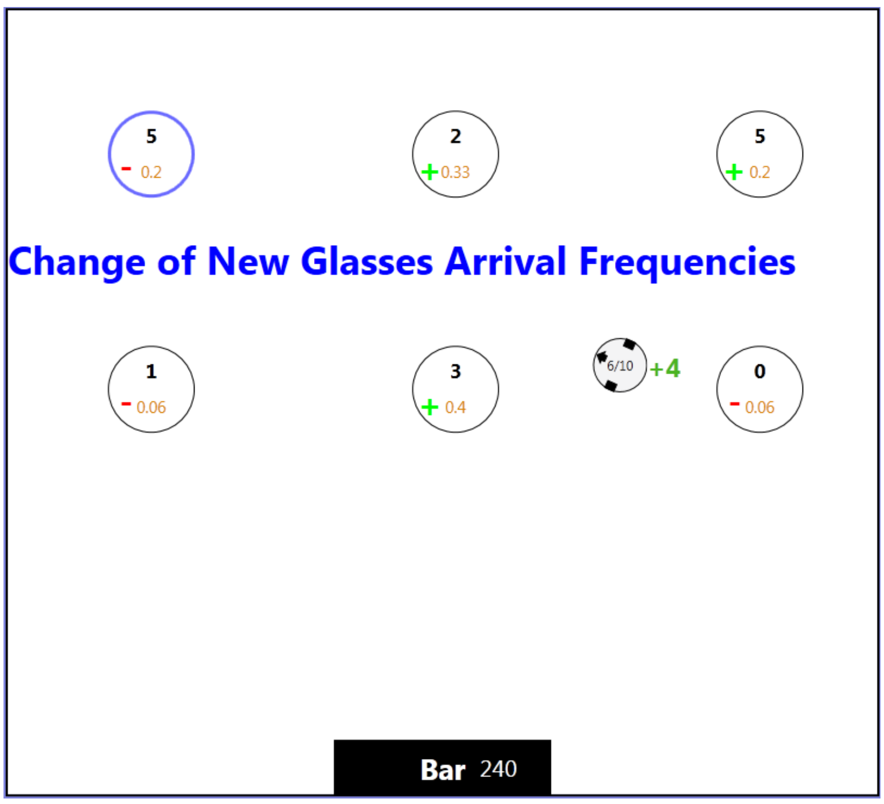

Figure 5: Notification of change (the frequencies of all tables have changed).

A fully counterbalanced mixed within- and between-subject experimental design was used with 14 subjects in each of the three conditions. The within-subject variable was the *number of changes* of environmental conditions with two levels (*2 changes*per session and *4 changes* per session) and the experimental session (2 sessions, which were run consecutively).

The arrival rate of empty glasses at each table was an environmental condition that could change. Changes occurred at predetermined times during the experiment and consisted of a permutation of the rates between the tables. Such changes impact the validity of the learned behavior: if the automatic behavior performed well before a change (i.e. the robot visited each table at the appropriate frequency), then after a change the performance with the automatic behavior will drop, because it will not be adjusted to the new rates of arrival.

The experiment was designed, so that one factor differed between pairs of groups (A-B, B-C), allowing us to evaluate the influence of this factor. In all analyses the number of changes (2 or 4) was a within-subject variable.

In group A participants received *no preview* about the learned behavior and *no information* about the validity of the learned behavior.

In group B participants received *notifications about changes* impacting the validity of the learned behavior and *no preview* of the learned behavior. These notifications appeared whenever the empty glass arrival rate at a table changed. They consist of the display of a blue text saying "Change of New Glasses Arrival Frequencies" and of animations causing the new frequency to blink on each table where a change occurred. The animation also indicates if the change is positive or negative (Figure 5).

Participants in group C received a *preview* of the learned behavior and *notifications about changes* impacting the validity of the learned behavior. In the automatic mode the preview indicated the three next destinations of the robot (bar or tables) through lines showing the future path of the robot (Figure 6).

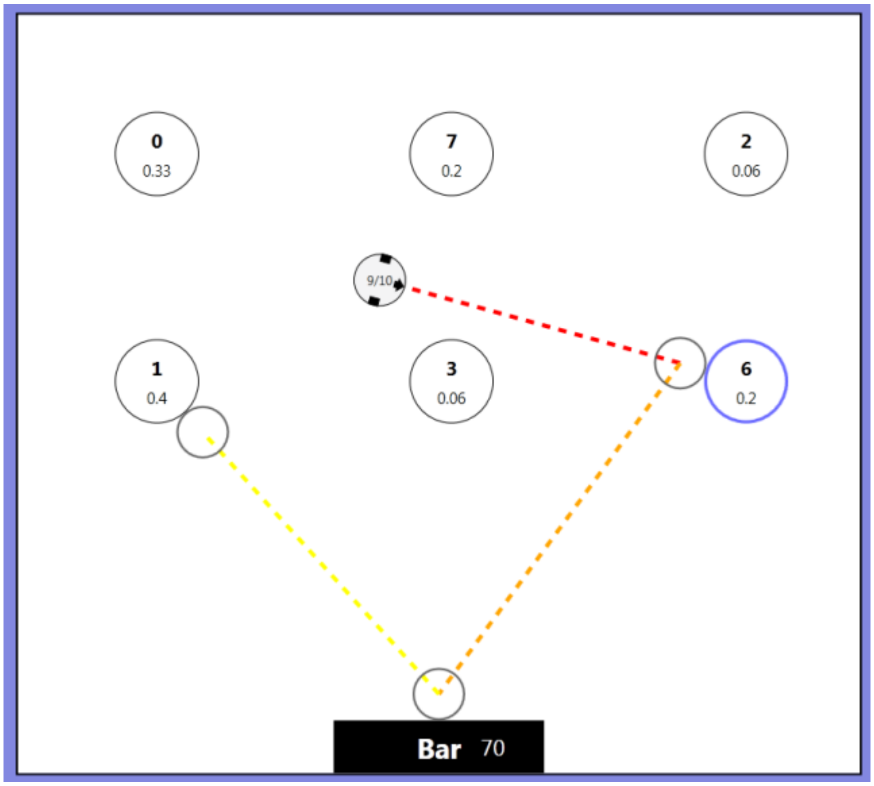

Figure 6: Preview of the robot learned behavior.

See Table 1for the configuration of the factors tested for each group.

Group	Subgroup	Presence of preview of the learned behavior	Information about the validity of the learned behavior	Number of changes of environmental conditions during a session: Session 1	Session 2
A	1	No preview	No information	2	4
	2			4	2
B	1	No preview	Notification about changes	2	4
	2			4	2
C	1	Preview	Notification about changes	2	4
	2			4	2
	2			4	2

Table 1: Group factors configuration

E. Dependent measures

Evaluation included objective and subjective measures. With the following eight metrics we measured performance, how the participants made use of the learning capabilities of the robot, what triggered a switch, and what were their reactions to changes:

1) *Number of glasses lost.* This is the performance measure in the main task, with more glasses lost indicating lower performance.

2) *Number of math problems solved.* This is the performance measure in the secondary task. The more math problems were solved, the better the performance in the secondary task.
3) *Score*. The overall performance score, as measured through the final score displayed at the end of the session:

   Score = 100 + Number of math problems solved - Number of glasses lost
4) *Use of automation*. The percentage of time the subject used the learned robot automatic mode relative to the time the learned robot automatic mode was available.
5) *Number of switches*. The total number of switches between the automatic and the manual modes.
6) *Number of resets*. The number of times the subject performed a reset of the learning of the robot.
7) *Number of glasses lost before a switch from the automatic mode to the manual mode*. The average number of glasses lost in a time period starting 10 seconds before a switch from the automatic mode to the manual mode, and ending when the switch occurs. This metric helps to characterize to which extent a switch to the manual mode is triggered by a decrease in performance of the main task, i.e. an increase in the number of glasses lost.
8) *Appropriate response to a change of environmental constraints.* The percentage of changes of environmental conditions followed in the next 10 seconds by a switch to the manual mode when the automatic mode is used. Since a change of environmental condition makes previously valid learned behavior obsolete, an appropriate response to such a change is to quit the automatic mode to avoid a decrease in performance.

Additionally, subjective evaluations of workload were collected using the NASA-TLX questionnaire (each with a 100-points range with 5-point steps):

1) Mental Demand
2) Physical Demand
3) Temporal Demand
4) Performance
5) Effort
6) Frustration

F. Data analysis

Two mixed design ANOVAs were conducted, one for each of the pairs of conditions:

- A and B: to determine the effect of a notification.
- B and C: to determine the effect of a preview when a notification was given.

These ANOVAs were used to evaluate the effects of the factors on the 8 objective dependent measures defined in the previous section.

To evaluate how the subjective workload metrics are impacted, a different one-way ANOVA was conducted for the two pairs of conditions as the questionnaire was completed by the subjects after their two experimental sessions; hence the effect of the Number of Changes could not be evaluated.

For the objective dependent measures the effect is considered significant at or below $\alpha = 0.05$. For the subjective dependent measures, the effect is considered significant below $\alpha = 0.1$.

G. Procedure

Before starting the experiment, participants signed a consent form. The scoring system and the different elements of the interface were described in detail. The explanation they received about the robot learning is that it will "reproduce the frequency at which they visited the tables". Then they had the opportunity to try the system for 2 minutes to become familiar with the commands. During the 2 minutes trial the automatic mode was not available.

After the trial ended, the subjects were asked to start the first session. After the first session ended they were offered a 10 minutes break and could then start the second session. Next they had to complete the NASA-TLX questionnaire and were paid.

3.3. Results

A. The effect of notifications

The influence of providing *notifications about changes* impacting the validity of the learned behavior compared to providing *no information* was evaluated by comparing results of groups A and B. Mean values of each measures and significant differences between the groups are shown in Figure 7 and Figure 8.

Providing *notifications about changes* compared to *no information* significantly increases the average score (from 138.1, SD=91.5 to 203.9, SD=60.4, F(1,24)=6.94, *p*=.015). It reflects better performance in both the main task (less number of glasses lost) and the secondary task (more math problems solved) (Fig. 5). Taken separately the changes in the two tasks are not statistically significant, but combined in the Score metric, they are.

With *notifications about changes* the subjects use the automatic mode on average significantly more compared to with *no information* (55.9%, SD=28.9 of the time compared to 74.4%, SD=19.3, F(1,24)=4.90, *p*=.037). This can explain the improved performance in the secondary task: the secondary task is much easier to complete when no manual control is needed in the main task.

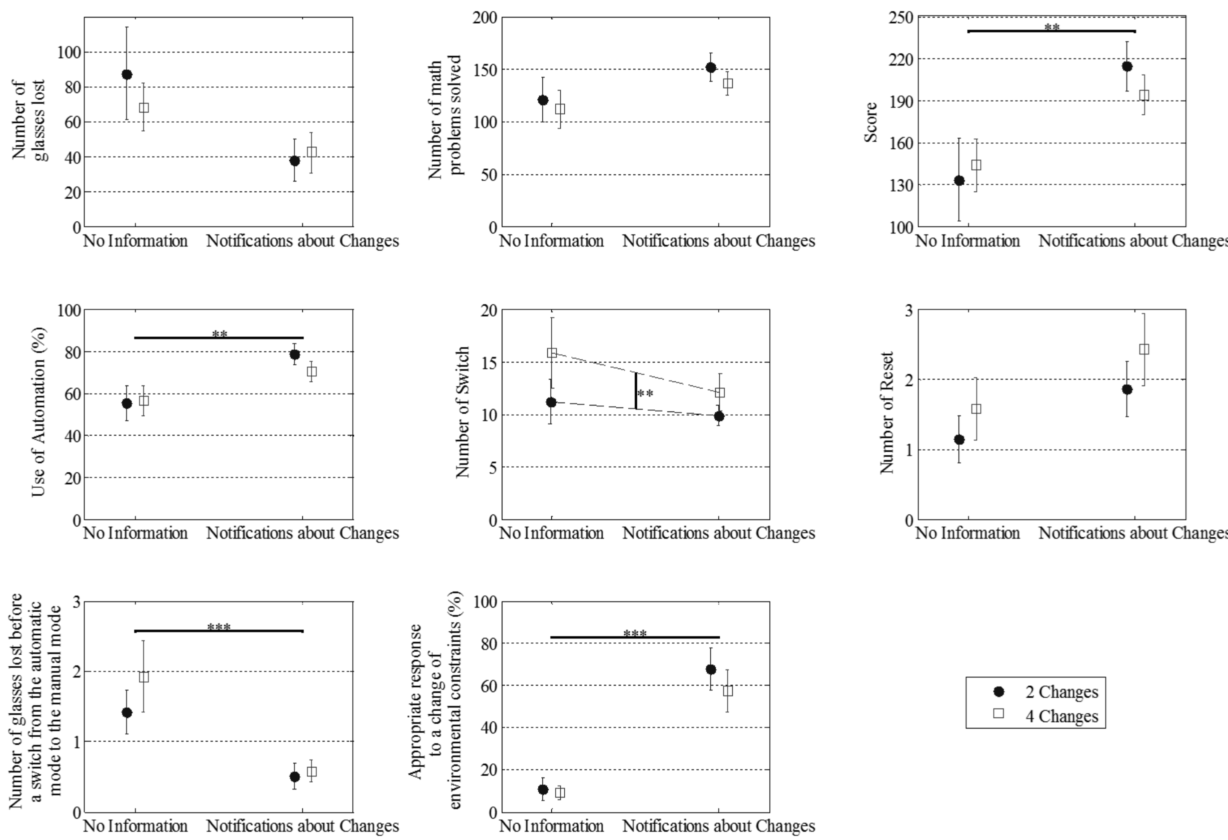

Figure 7: Effect of the presence of notifications on the general measures. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 10%, 5% and 1%. Error bars represent the standard error of the mean.

With *notifications about changes* the subjects react on average significantly better to a change of the conditions than with *no information* (62.5% ,SD=36.9 of appropriate responses compared to 9.8%, SD=17.1, F(1,24)=44.06, *p*<.001). This result indicates that with ***notifications about changes*** the subjects are able to understand better that the current learned behavior of the robot is not adapted any more to the environmental condition and that they take the appropriate action to avoid a decrease in performance in the main task. This leads to better performance in the main task compared to with ***no information***.

There is also a significant impact of providing *notifications about changes* on the average number of glasses lost before a switch from the automatic mode to the manual mode (0.54, SD=0.63, glasses with *notifications about changes* compared to 1.67, SD=1.5, glasses with *no information,* F(1,24)=11.86, *p*=.002). This suggests that with *no information* the subjects tend to switch from automatic mode to manual mode in a reactive manner. They switch when they see the performance decrease, i.e. when they see the loss of more glasses. With *notifications about changes*, they switch to the manual mode before the performance of the main task starts to decrease.

Having 4 changes per session instead of 2 changes increases, as expected, the number of switches (from 12.7, SD=6.1, to 17.5, SD=10.1, F(1,24)=4.613, *p*=.042), because with more changes, more modifications of the learning of the robot are needed. However, the other metrics remain unchanged.

In the subjective measures, it appears that providing *notifications about changes* significantly reduces subjects' workload compared to providing *no information* (from 61.1, SD=11.7 to 53.8, SD=8.7, F(1,24)=3.36, *p*=.079). **Subjects' subjective performance is also significantly better when providing *notifications about changes*** (lower is better) **(**34.3 ,SD=19.4 compared to 48.2, SD=18.7, F(1,24)=3.72, *p*=.066), which correlates with the objective increase in the score metric when *notifications about changes* are present.

There was no significant interaction between the Number of Changes and the Presence of notifications.

These results show that subjects benefit much from notifications *about changes* of environmental constraints compared to receiving *no information*. Their overall performance (score) increased, their understanding of the system improved and their workload diminished. Additionally, the impact of the number of changes remains limited.

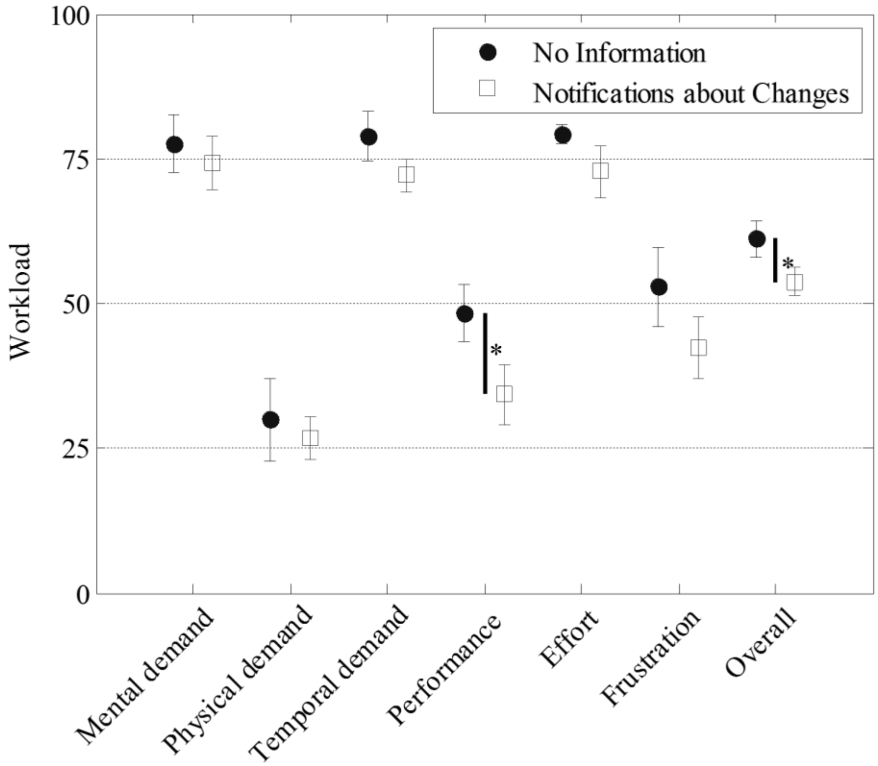

Figure 8: Effect of the presence of notifications on the 6 dimensions of workload of the NASA-TLX questionnaire, the overall workload is represented by horizontal lines. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 10%, 5% and 1%. Error bars represent the standard error of the mean.

B. The effect of previews

By comparing the results of group B and group C, the influence of the presence of *preview* of the robot learned behavior compared to *no preview* can be evaluated. Mean values of each measure and significant differences between the groups are shown in Figure 9 and Figure 10.

The presence of a *preview* significantly reduces subjects' score (from 203.9, SD=60.4 without a preview to 125.1, SD=58.5 with preview, $F(1,24)=20.53$, $p<.001$). This score reduction is mainly due to a significant decrease in the performance of the secondary task (143.8, SD=45.9, math problems solved with a preview compared to 69.6, SD=50.1, with no preview, $F(1,24)=16.844$, $p<.001$).

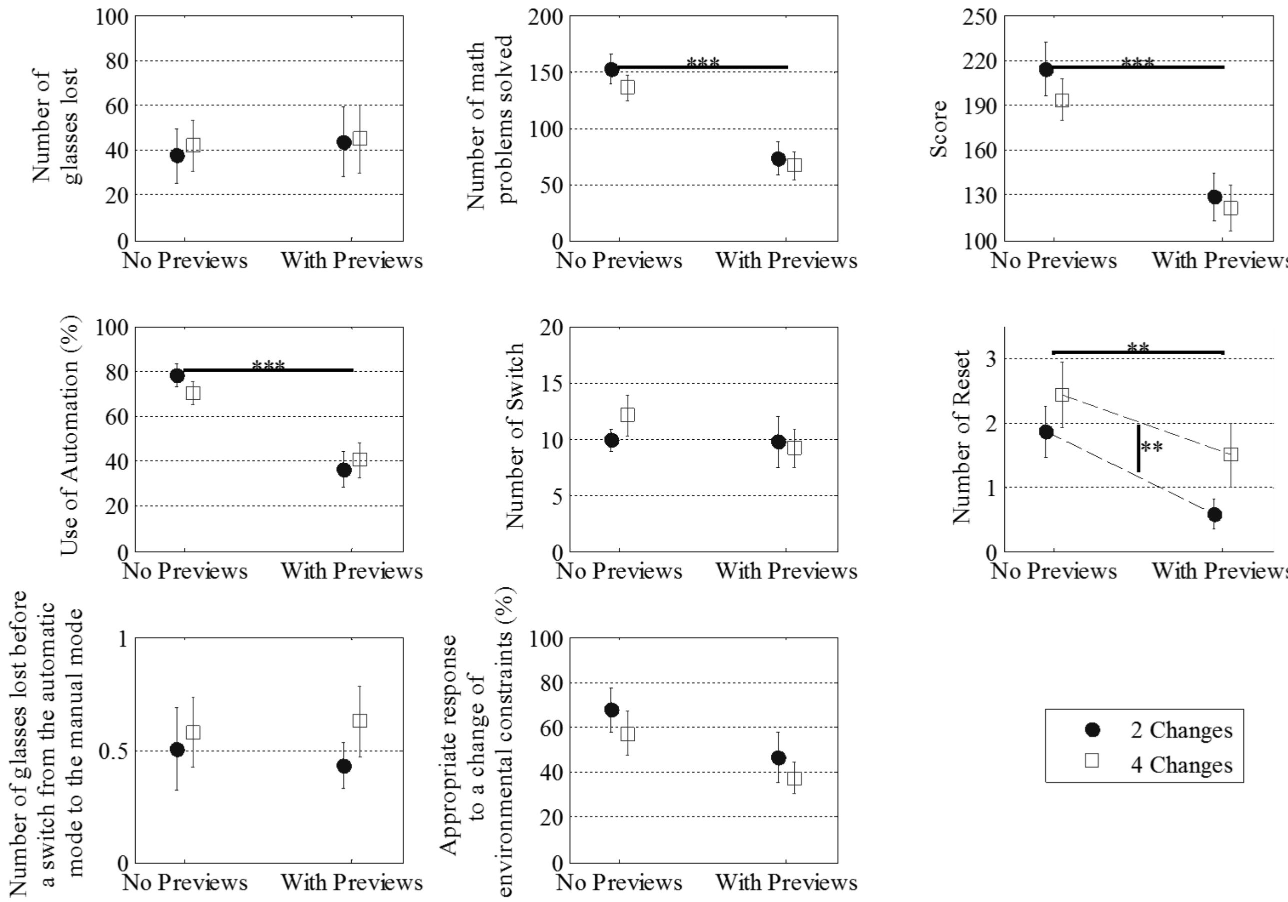

Figure 9: Effect of the presence of previews on the general measures. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 10%, 5% and 1%. Error bars represent the standard error of the mean.

The use of automation was significantly reduced by the presence of a *preview* (from 74.4%, SD=19.3 to 38.5%, SD=28.6, F(1,24)=21.04, p<.001). This can explain the decrease in performance in the secondary task: when using more the manual mode in the main task, subjects had less time to answer the math problems of the secondary task.

It appears that subjects performed on average less resets of the learning in the presence of a *preview* (1.04, SD=1.5, compared to 2.14, SD=1.6, F(1,24)=4.32, p=.049). This effect can be partially explained by the lower use of automation: since the subjects use the automation less, they are less likely to put effort in maintaining an efficient automatic mode by performing resets.

Having 4 changes per session instead of 2 changes increases the number of resets (from 1.21, SD=1.37 with 2 changes to 1.96, SD=2.93 with 4 changes, F(1,24)=8.468, p<.001). The same tendency was observed in the first analysis, but in this case it is significant. This effect can be explained by the fact that with more changes, more adaptation of the learning is required.

In the subjective measures, the presence of a *preview* led to a significant increase in the Physical Demand (from 26.8, SD=13.4, to 40.7, SD=26.9, $F(1,24)=2.999$, $p=.09$), but has no significant effect on the other subjective metrics.

The interaction between the two factors *Number of Changes* and *Presence of notifications* was not significant.

The results indicate that the presence of previews of the robot learned behavior does not modify the subjects' reaction to a change of environmental constraints. However, the preview lowers subjects' willingness to give the control to the automatic mode. Apparently seeing the preview lowers subjects' trust in the automatic mode. This lesser use of automation with a preview is reflected by degraded performance in the secondary task and hence by a lower overall score. Additionally, the impact of the number of changes remains limited.

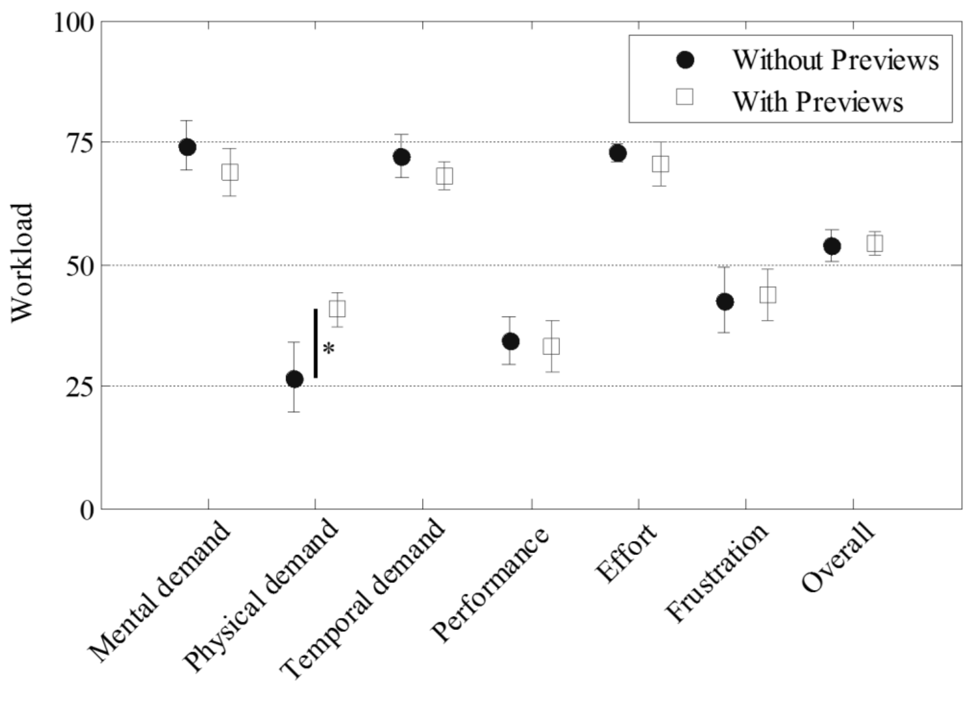

Figure 10: Effect of the presence of previews on the 6 dimensions of workload of the NASA-TLX questionnaire, the overall workload is represented by horizontal lines. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 10%, 5% and 1%. Error bars represent the standard error of the mean.

3.4. Discussion and conclusion

The results indicate that in the context of online robot learning in changing environments, **providing information to help the user understand the validity of the robot's learned behavior is very important for the user and to the whole system**. However, **not every type of information is beneficial**. It is necessary to find the right balance between increasing the system performance and not confusing the user. The simplistic thought that the more information is provided, the better will be the performance turns out to be false. Moreover, with too much information, even if the information is perfectly accurate, the performance proved to be worse than when compared to no information at all.

The best way to improve performance among the tested settings was to **provide the users only with brief and contextualized notifications about changes**. This **increased their overall performance and reduced their workload**. Moreover, the users reacted better to a change in the environment and were able to take the appropriate action to avoid a decrease in performance when a change occurred, realizing that the learned automatic robot behavior should not be trusted anymore. They tended to switch to the manual mode to avoid a loss of performance and were likely to reset the learned behavior so that the robot begins to learn a new behavior, more appropriate to the current conditions. These good results can be explained by the limited additional workload needed to process these notifications. Moreover, with notifications users did not have to monitor constantly the main task to detect changes, which were difficult to perceive, resulting in reduced measured workload.

In contrast, **when adding previews in the presence of notifications, the overall performance dropped dramatically,** mainly due to the decrease in performance in the secondary task, the main task performance remaining stable. Interestingly **the preview also reduced the use of automatio**n, which explains why the secondary task performance decreased so much. With more information about how the robot in the automatic mode will act in the future, users trusted it less and were less willing to use the automatic mode. Thus, the previews were not able to convey any useful information to the users, and moreover seemed to disturb them and to make the situation more complex to understand, leading to a lower level of performance than when no information was provided at all. However, with no measured impact on any subjective workload metrics but the physical demand, the negative effect of adding previews can hardly be explained by them causing cognitive overload due to their complexity. There are several possible explanations for this counterintuitive negative effect. First, the preview is a continuous stimulus which changes constantly and draws much attention to the main task when the automatic mode is chosen. This defeats the purpose of the automatic mode, which is to reduce users' attention needed for the main task to allow them to work on the secondary task. Second, because the preview is only present when the automatic mode is active, it is possible that it is incorrectly interpreted as an alert, and it introduces a bias toward the manual mode in order to make the signal disappear.

The sensitivity of these results to the number of changes appears to be limited. Moreover, the number of changes did not significantly modify the way the users reacted to a change, nor their level of performance. **Users seemed to have developed a strategy to cope with changing conditions**, and this strategy remains the same, irrespective of the number of changes of conditions, 2 or 4, during a session. However, it is possible that this result will be affected by the number of changes - when the cost of the increased number of switches and resets is higher than the benefit from having access to an automatic mode.

4. Responses to warnings and the effect of feedback about changes in a simulated robot-control task

4.1. Overview

This experiment focuses on the evaluation of operator responses to a warning system with changing characteristics (i.e. with changing sensitivity over time). In particular the operator’s trust in the warning system through its two dimensions, reliance and compliance was analyzed.

The operator's response was studied in four feedback conditions: 1) *Without Feedback* about the changes of the warning system, 2) with feedback about the changes in forms of *Notifications,* 3) with feedback about the state of the warning system sensitivity in forms of *Continuous Information*, and 4) with *Misleading Notifications*, informing the operators of a change in the opposite direction to the actual change of the warning system.

Each of these four conditions were tested with two warning system sensitivity settings: 1) the operators interacted first with a warning system with a low sensitivity, which then increased to a high sensitivity, before going back to the original low sensitivity, thus denoted as the Low-High-Low sensitivity setting; and 2) the operators interacted first with a warning system with a high sensitivity, which then decreased to a low sensitivity, before going back to a high sensitivity, thus denoted as the High-Low-High sensitivity setting.

4.2. Method

A. Participants

96 participants were recruited among the members of the Bristol Robotics laboratory and the Industrial Engineering Department of the Ben-Gurion University of the Negev. Each was randomly assigned to one of the 8 experimental conditions. They received 30 NIS (New Israeli Shekels, about 8 dollars) for their participation in the experiment and could get a bonus of 100 NIS (about 26 dollars) according to their performance. The recipients of the bonus were determined with a lottery, with each point in the cumulative score serving as a lottery ticket. Thus, the higher a subjects’ cumulative score in the two sessions, the more virtual lottery tickets they received.

B. Experimental platform

The experiment was conducted on a PC with a 15.6” HD monitor. The experimental program was written using C# and WPF 4. Each participant sat at the computer and interacted with the experiment using only a mouse.

C. Task

Participants were asked to play a decision making game. Their goal was to end the game with the highest score, knowing that the two highest scores will receive a prize (i.e. in this experiment a cake). They played a succession of trials in which a robot carried plates from a kitchen to a restaurant. In each trial, participants chose with a slider to load between 0 and 10 plates on the robot tray, which is the load, L. To validate a choice of L the participants then pressed a “Go” button. There was no time limit on making the choices. For each trial either the path of the robot to the restaurant was free and the participant was awarded with a number of points L equal to the number of plates loaded on the robot, or the path of the robot was crossed by people. The probability of people crossing the path of the robot was $P_p = 0.5$, and the probability of the path being clear was$P_c = (1 - P_p) = 0.5$. Participants were informed that there was a 50% chance that people will cross the path. When people crossed the path, the robot had to brake to avoid a collision. There were two possible outcomes of such a maneuver:

1) Plates fell from the robot and the participants lost a number of points equal to L.
2) The robot avoided the people smoothly without any fall of plates and reached the restaurant. In this case participants received L points.

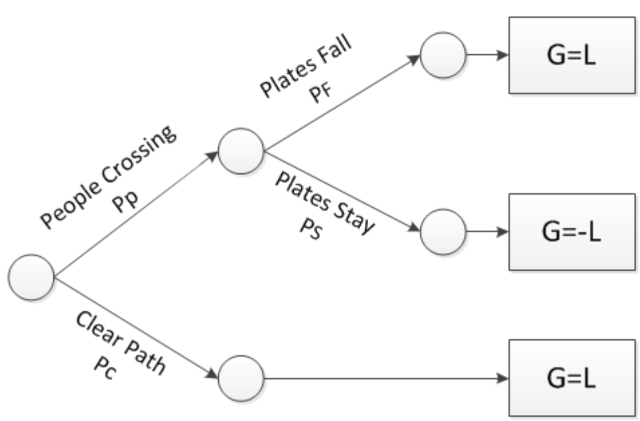

Figure 11: Possible outcomes in terms of gain G of a trial

The probabilities that the plates fell or stayed on the robot were proportional to the number of plates loaded L: $P_F = \frac{L}{10}$ and $P_S = (1 - P_F) = \frac{10-L}{10}$. See Figure 11 for the possible outcomes of a trial. Hence the expected value for a trial is:

Ev

$= Ev_{People\ Crossing} * P_p + Ev_{Clear\ Path} * P_c$

$= \left[Ev_{Fall} * P_F + Ev_{Stay} * P_S\right] * P_p + Ev_{Clear\ Path} * \left(1 - P_p\right)$

$= [-L * P_F + L * (1 - P_F)] * P_p + L * \left(1 - P_p\right)$

$= -2 * P_F * P_p * L + L$

$= -\frac{1}{5} * P_p * L^2 + L$

Figure 12 shows which L value a rational decision maker has to choose to maximize Ev in terms of P_p. For P_p values below 0.25 the best choice of L is 10, and it is never an optimal decision to choose a L value lower that 2.5, independently from the value of P_p. Thus, for $P_p = 0.5$, the optimal choice of L is 5, which is the default slider position of choosing L at the beginning of each trial.

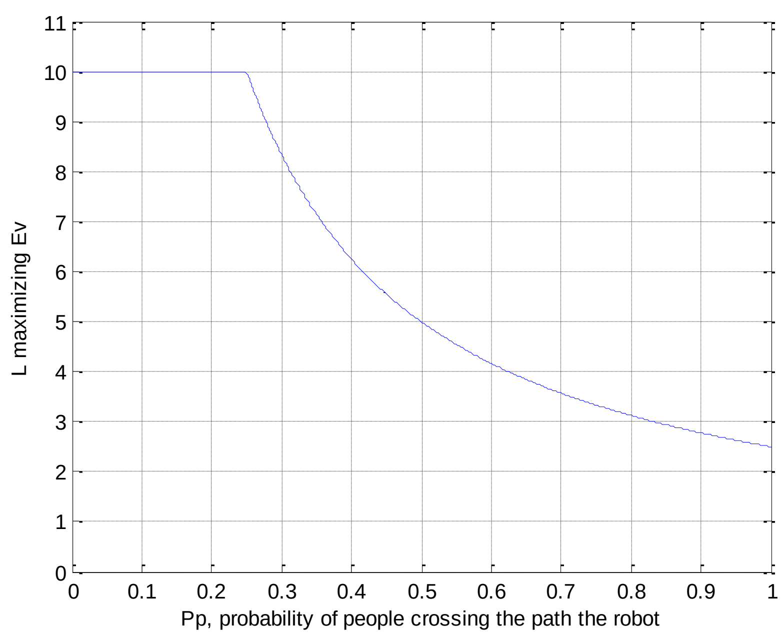

Figure 12: L values maximizing Ev in terms of Pp

Participants received additional information about the risk of people crossing the robot path from a warning system displayed in the form of a rectangle above the slider for choosing L (Figure 11). This alert system has two possible outputs:

1) A warning is issued in the form of amessage “Movement Detected” in the rectangle with a red background.
2) No warning is issued, and the message “No Movement Detected” is written in the rectangle with a green background.

It is possible to assess to which extent the participant relied or complied with the warning system by measuring the difference between the participant choice of L with and without a warning; the default L value was set to 5.

For each participant the experiment was divided in three blocks of 100 trials. The sensitivity of the warning systems changes between the first block and the second block, and between the second block and the third block, according to the sensitivity setting, see Table 1. For each experimental block of each participant, the order of the trials with and without warning and with and without people crossing was individually randomized. The random distribution is ensured by creating a list representing the trials order with a number of trials with or without people crossing and in presence or not of a warning which respect the probabilities and the conditional probabilities associated with the sensitivity setting of each experimental block defined inTable 2. Then the elements of this list are shuffled using the Fisher-Yates-Durstenfeld shuffle algorithm.

Low-High-Low sensitivity setting							
Experimental Block	**d'**	**p_p**	**p_w**	**$p_{p/w}$**	**$p_{p/nw}$**	**OL/w**	**OL/nw**
1	1	0.5	0.5	0.69	0.31	3.62	8.07
2	3			0.93	0.07	2.69	10
3	1			0.69	0.31	3.62	8.07

High-Low-High sensitivity setting							
Experimental Block	**d'**	**p_p**	**p_w**	**$p_{p/w}$**	**$p_{p/nw}$**	**OL/w**	**OL/nw**
1	3	0.5	0.5	0.93	0.07	2.69	10
2	1			0.69	0.31	3.62	8.07
3	3			0.93	0.07	2.69	10

Table 2: Warning system sensitivity (d'), probabilities for people crossing (p_p) and for display of warning indicator (p_w), conditional probabilities for people crossing given that awarning is displayed ($p_{p/w}$) and for people crossing given that no warning is displayed ($p_{p/nw}$), and optimal choice of L given that a warning is displayed (OL/w) and given that nowarning is displayed (OL/w) ; for the three different experimental blocks and for the low-high-low sensitivty setting (left) and the High-low-high sensitivity setting (right)

D. Design

A between-subject experimental design was used with 8 groups (4 feedback conditions * 2 warning system sensitivity settings).

For the first feedback condition, *Without Feedback* of change of the warning system was displayed, the transition between two experimental blocks is indiscernible. However, for the second feedback condition, a *Notification* describing the change in the warning system was displayed between blocks 1 and 2; and between blocks 2 and 3. See Figure 13. The notifications appeared for 10 seconds with an animated colored arrow and masking the slider and the “Go” button.

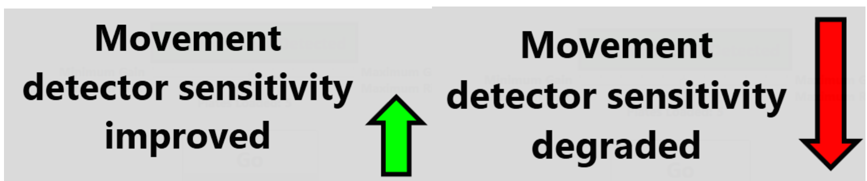

Figure 13: Notification displayed between block 1 and 2 and between block 2 and 3 for the secondand fourth experimental group according to the direction of the sensitivity change.

For the third feedback condition, *Continuous Information*, information about the level of sensitivity of the warning was constantly displayed below the score. It consists of a title, “Detector Sensitivity”, and the level of the sensitivity, either “LOW” or “HIGH”, see Figure 14. This information was updated each time the sensitivity changes (i.e. between blocks).

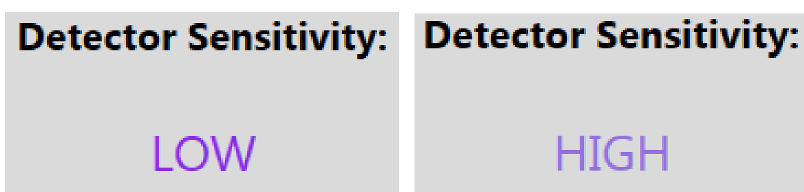

Figure 14: Continuous information about the sensitivity of the warning system as displayed in the third experimental conditions. Low level (left) and High level (right).

For the fourth feedback condition, for which the participants received *Misleading Notifications*, the same notifications as for the second feedback condition were displayed between the experimental block, but their polarity was inversed: if the sensitivity of the system increased, the notification displayed “Movement detector sensitivity degraded”, and if the sensitivity of the system decreased, the notification displayed “Movement detector sensitivity improved”.

E. Dependent measures

To assess to which extent the participant relied or complied with the warning system, the participant choice of *L* was recorded for each trial of the three experimental block. More specifically it was divided in two sub-measures: the choice of *L* given that there was a warning and the choice of *L* given that there was no warning; in order to evaluate how much the participants relied or complied on the warning system.

F. Data Analysis

For all settings, a Linear Mixed Models analysis (McCulloch & Searle 2000) was conducted for the operator's choice of L with contrast analysis with the presence of warning and block number as within-group fixed effects and the experimental condition as a between-group fixed effect.

G. Procedure

Before starting the experiment, participants signed a consent form. The scoring system and the different elements of the interface were described in detail. At the end of the experiment they were paid.

4.3. Results and Discussion

For all conditions and all experimental conditions, **the effect of the presence of warning has strong effect of the operators' choice of L** ($p<0.001$). The sections below describe how this effect is influenced by the experimental conditions.

A. Effect of accurate notifications

a) Low-High-Low setting

When the sensitivity of the alarm system is low in the first block, high in the second block and low again in the third block, *without feedback* about the change of the sensitivity of the warning system, the operators' chosen value of L in the presence of warnings remains constant between the first block and the second block, and between the second block and the third block, see Figure 15. Thus, **the compliance remains unchanged regardless of positive or negative**

changes in the sensitivity of the system when no notifications about these changes are provided. In contrast, when no warning is displayed, the chosen value of *L* significantly increases between the first and the second block (from 7.25, SD=1.06, to 8.11, SD=1.35, $p<0.001$), and remains constant between the second and the third block, see Figure 15. **This supports a difference between reliance and compliance as changes of warning system sensitivity provoked a modification of the participants' response to the no-warning state of the system, but it did not modify their reaction to warnings**. Thus, here participants increased their level of reliance on the warning system following its increase in sensitivity between the first and the second block. However, they did not adjust their level of reliance between the second and the third block when the sensitivity went back to the initial low level, resulting in over-reliance in the third block. **In this case the participants adjusted their level of reliance as if they understood that the warning system could improve over time but not worsen**.

In the presence of *notifications*, the situation is different (Figure 15). Between the first block and the second block, both the operators' level of compliance ($p<0.01$) and reliance ($p<0.001$) in the warning system significantly increased; i.e. the chosen value of *L* when a warning is displayed significantly decreased between the blocks (from 4.04, SD=1.05, to 3.22, SD=1.58, $p<0.01$), and when no warning is displayed significantly increased (from 7.42, SD=1.14, to 8.92, SD=1.05, $p<0.001$). Similarly, between the second block and the third block, the operators' levels of compliance ($p<0.001$) and reliance ($p<0.001$) significantly decreased, returning to values not significantly different from the ones in the first block, 7.15, SD=1.6, and 4.49, SD=1.36, respectively. **Hence, with notifications, the participants properly adjusted their levels of trust to the change of warning system sensitivity**. It can be noted that in the third block, the level of reliance is significantly different between the conditions with and without notification ($p>0.05$). With notifications it is properly adjusted and without notifications there is over-reliance.

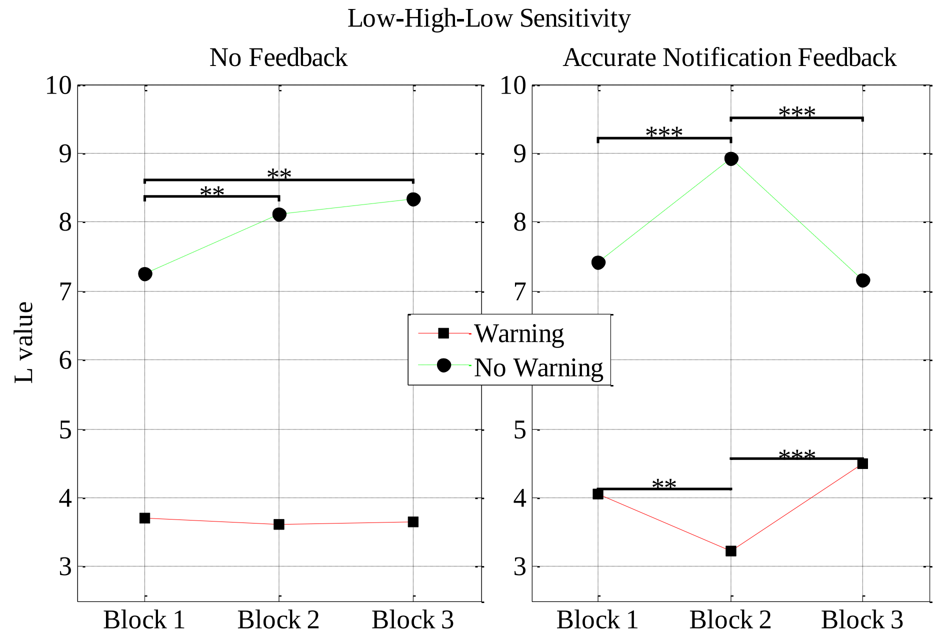

Figure 15: Effect of warning and notification on the operators' choice of L in the Low-High-Low setting.

b) High-Low-High setting

Without feedback, when the sensitivity of the alarm system is high in the first block, low in the second block and high again in the third block, **both the level of compliance and the level of reliance did not change significantly, showing no adaptation to the changes of system sensitivity** (Figure 16). For the three blocks **the operators' level of compliance and reliance on the warning system was high**, and hence appropriate for the first and third blocks. But no adaptation, either in terms of reliance or compliance, to the decrease of the system sensitivity is seen during the second block, when the sensitivity went from high to low. Again operators acted as if a decrease of the warning system performance was not possible.

However, **similar adaptation as with the Low-High-Low sensitivity setting is seen in presence of *notifications***: between the first block and the second block, both the operators' level of compliance and reliance in the warning system significantly diminished following the decrease of the warning system sensitivity (L values in presence of warning increased from 2.79, SD=1.17, to 3.67, SD=1.33, $p<0.01$, and in presence of no warning decreased from 9.16, SD=0.63, to 8.21, SD=1.24, $p<0.01$). Between the second and third block, both the operators' level of compliance and reliance increased, adjusting to the increase of the warning system sensitivity (L values in presence of warning decreased from 3.67, SD=1.33, to 2.97, SD=1.06, $p<0.05$, and in presence of no warning increased from 8.21, SD=1.24, to 9.77, SD=0.24, $p<0.001$). These results show that **when operators receive notifications about positive or negative changes of the warning system sensitivity, they adapt their levels of reliance and compliance to the new characteristics of the system, independently from the initial or previous sensitivity setting**.

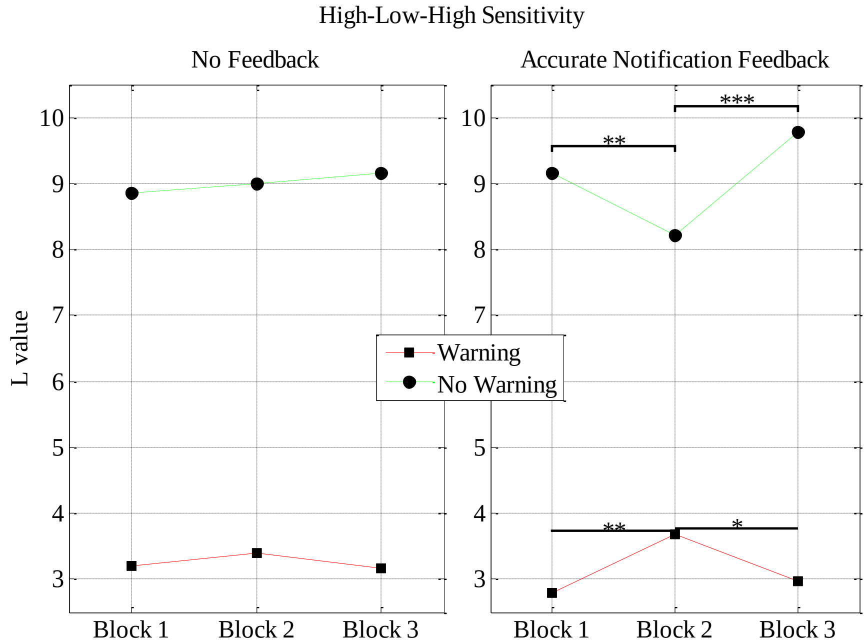

Figure 16: Effect of warning and notification on the operators' choice of L in the High-Low-High setting.

4.4.Effect of the type of feedback

a) Low-High-Low setting

With the Low-High-Low sensitivity setting, no significant difference is seen between the type of feedback provided (for both *notifications* or in the form of *Continuous Information*), see Figure 17. **With both types of feedback the operators adapted similarly their level of compliance and reliance to the change of system sensitivity.** With *Continuous Information*, in presence of warning, the L values were 3.92, SD=0.79, in the first block, then decreased significantly ($p<0.01$) to 3.12, SD=1.27, in the second block and increased significantly ($p<0.001$) to 4.28, SD=0.86, in the third block. With *Continuous Information*, in presence of no warning, the L values were 7.02, SD=1.55, in the first block, then increased significantly ($p<0.001$) to 9.18, SD=0.86, in the second block and decreased significantly ($p<0.001$) to 7.51, SD=1.71, in the third block.

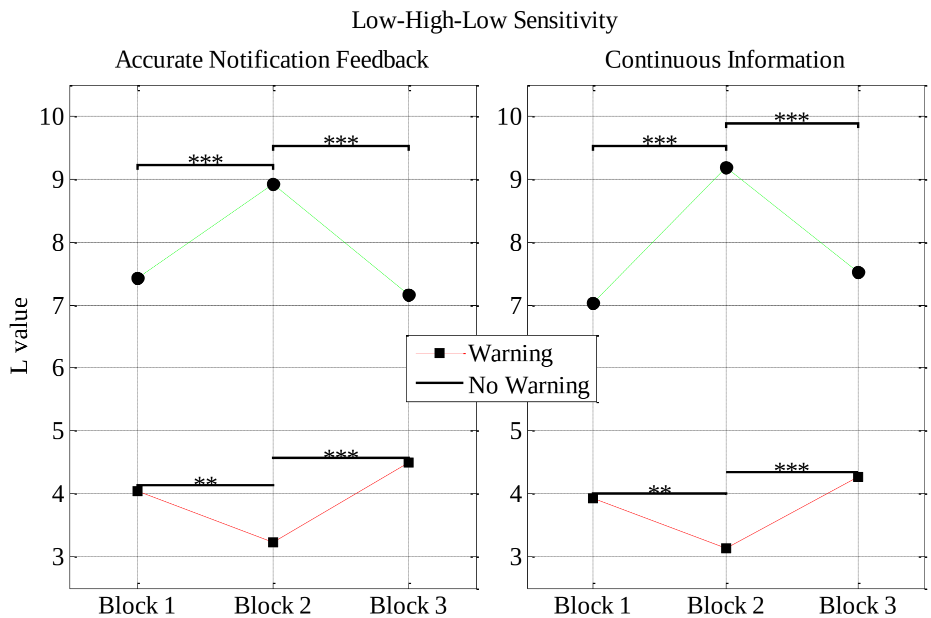

Figure 17: Effect of warning and type of feedback on the operators' choice of L in the Low-High-Low setting.

b) High-Low-High setting

However, with the High-Low-High sensitivity setting, providing the operators with *notifications* and *Continuous Information* about the changes of the warning system sensitivity has similar results for the first and the third blocks, see Figure 18; but **the adaptation to the decrease of sensitivity taking place in the second block is significantly stronger with *Continuous Information*** for both the reliance ($p<0.01$) and the compliance ($p<0.05$). Both the levels of compliance and reliance are lower in the second block with *Continuous Information* than with notifications. With *Continuous Information*, in presence of warning, the L values were 3.32, SD=1.09, in the first block, then increased significantly ($p<0.001$) to 4.89, SD=1.65, in the second block and decreased significantly ($p<0.001$) to 3.65, SD=1.6, in the third block. With *Continuous Information*, in presence of no warning, the L values were 8.93, SD=1.10, in the first block, then decreased significantly (p<0.001) to 7.09, SD=0.67, in the second block and increased significantly (p<0.001) to 9.23, SD=0.86, in the third block.

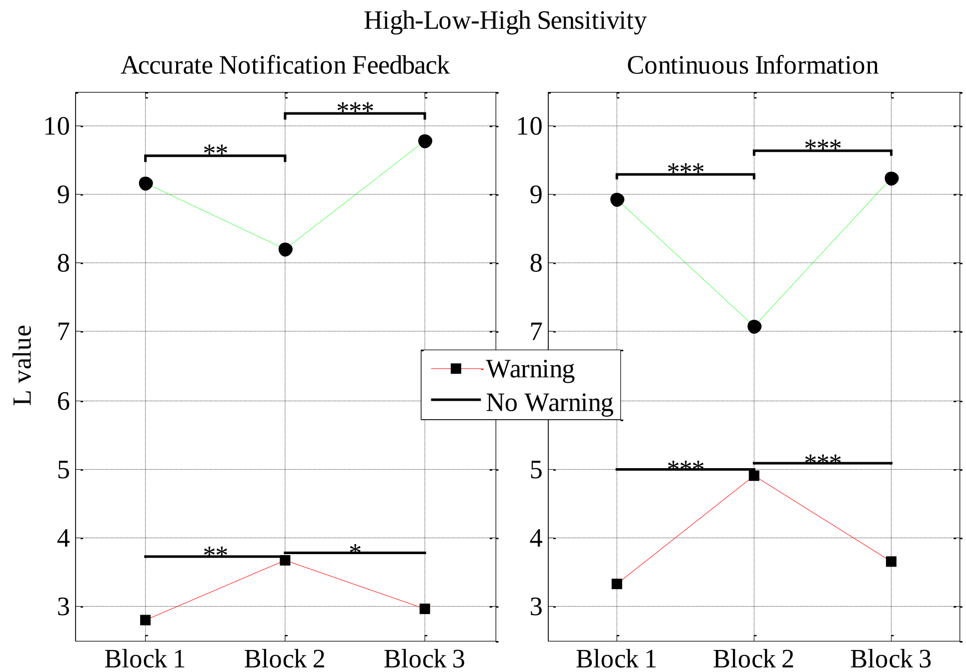

Figure 18: Effect of warning and type of feedback on the operators' choice of L in the High-Low-High setting.

4.5.Effect of misleading notifications

For both the Low-High-Low and the High-Low-High sensitivity settings, **presenting the operators *misleading notifications* has the same effect as providing *no feedback* about the change of warning system sensitivity**, see Figure 19 and Figure 20. Their levels of compliance and reliance for all blocks are not significantly different than levels for when *no feedback* is provided:

For the Low-High-Low setting, with warning from the first to the third block: 3.84, SD=1.16, 3.44, SD=1.07, and 3.83, SD=1.04; without warning 7.68, SD=1.28, 8.35, SD=1.27 and 8.61, SD=1.07. For the High-Low-High setting, with warning from the first to the third block: 3.46, SD=0.89, 3.73, SD=1.05, and 3.77, SD=1.13; without warning 9.27, SD=0.84, 9.10, SD=1.01 and 9.05, SD=0.73.

These results show that **the operators were able to properly discard and ignore the misleading notifications**. However, they were not able to use them as a cue that something changed, in this case the sensitivity of the warning system. It is possible that the operators followed first the misleading notifications and corrected their response seeing the performance dropping. Hence, averaged over the 100 trials of an experimental block the response appears unchanged as compared to the settings with no notification. Nevertheless, as it will be seen in next section, this is not the case.

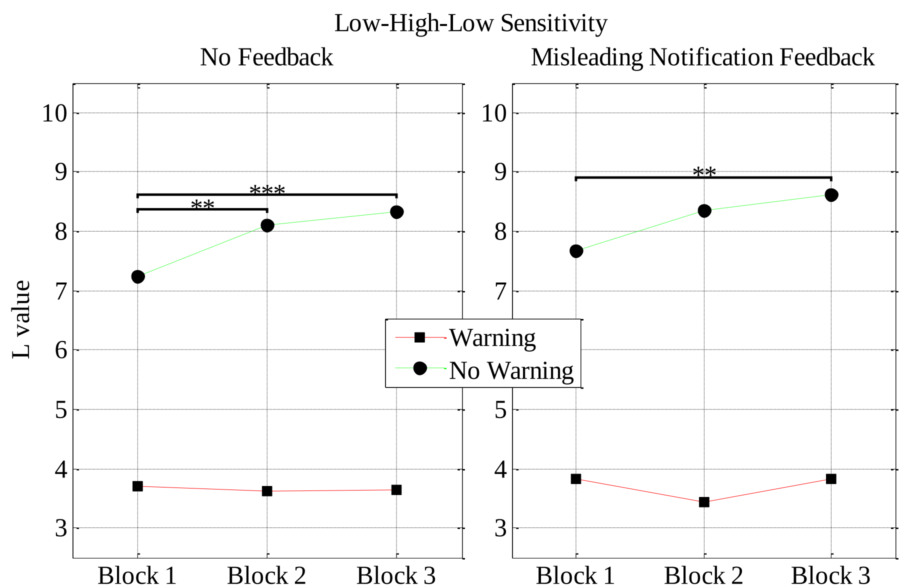

Figure 19: Effect of warning and misleading notifications on the operators' choice of L in the Low-High-Low setting.

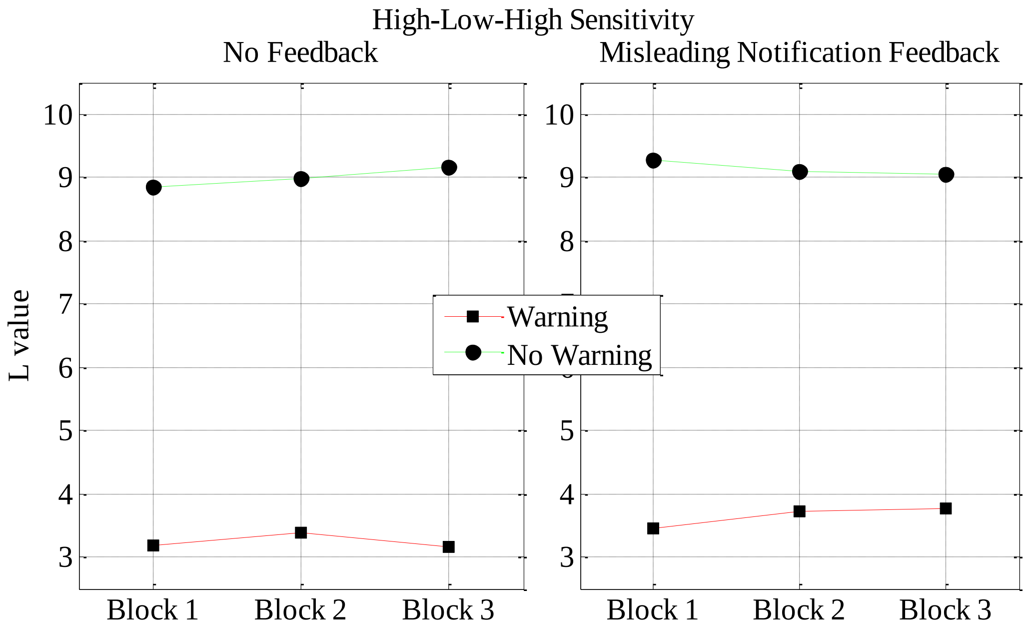

Figure 20: Effect of warning and misleading notifications on the operators' choice of L in the High-Low-High setting.

4.6. Detailed response to changes

By increasing the resolution of the analysis achieved by dividing each block into smaller sub-blocks, it is possible to visualize the evolution of the operators compliance and reliance within a block and hence to obtain a better picture of the effect of the two changes of system warning sensitivity taking place between Blocks 1 and 2, and between Blocks 2 and 3. Here each block of 100 trials has been divided in 10 sub-blocks of 10 trials, and for each sub-lock the average choice of L value with and without warning has been calculated. Results are displayed for all conditions in Figure 21 and Figure 22.

a) Block 1

In the *Low-High-Low* sensitivity setting, both reliance and compliance levels seems relatively stable. However, with the *High-Low-High* sensitivity setting, the level of reliance is increasing as the operators gain experience with the system whereas the level of compliance is remaining stable. **No influence of the feedback condition is seen in the first block for both sensitivity settings**, which is reasonable as feedback is given only at the transition between the blocks, apart for the *Continuous Information* condition.

b) Block 2

Between Block 1 and Block 2 the system warning sensitivity changed. **When proper feedback is given** in form of accurate *Notification* or *Continuous Information*, as seen before, **the operators successfully adapted both their levels of compliance and reliance to the new system warning sensitivity almost immediately after the change, for both sensitivity settings and with a slightly faster or stronger adaptation for the *Continuous Information* condition**.

For the other two conditions, *No Feedback* and *Misleading Notifications*, it is first interesting to note that the operators' responses were similar and that the *Misleading Notifications* were immediately discarded. **Operators gradually adapted their level of reliance over the length of the block**, which was not visible for the High-Low-High sensitivity setting when using only one measure for the whole block in section 4.5 on Figure 20. However, **this adaptation seems to be not present for the level of compliance**, or to a much smaller extend, in both sensitivity settings.

c) Block 3

Between Block 2 and Block 3 the system warning sensitivity changed again, going back to the same sensitivity as in Block 1. Similarly as for Block 2, **when proper feedback is given** in form of accurate *Notification* or *Continuous Information*, **the operators successfully adapted both their levels of compliance and reliance to the new system warning sensitivity** almost immediately after the change, for both sensitivity settings.

For the other two conditions, *No Feedback* and *Misleading Notifications*, as in Block 2, the operators' response was similar. In the same way as for Block 2, **a gradual adaptation of the level of reliance is observed for the High-Low-High sensitivity setting** which was not visible when using only one measure for the whole block in section 4.5 on Figure 20. However **this gradual adaptation of the level of reliance is not present for the Low-High-Low sensitivity setting**. Without proper feedback, when exposed first to an increase of system warning sensitivity, operators had trouble realizing that the sensitivity could change again negatively (i.e. decrease). Additionally, as for Block 2, without proper feedback, the level of compliance remains almost stable.

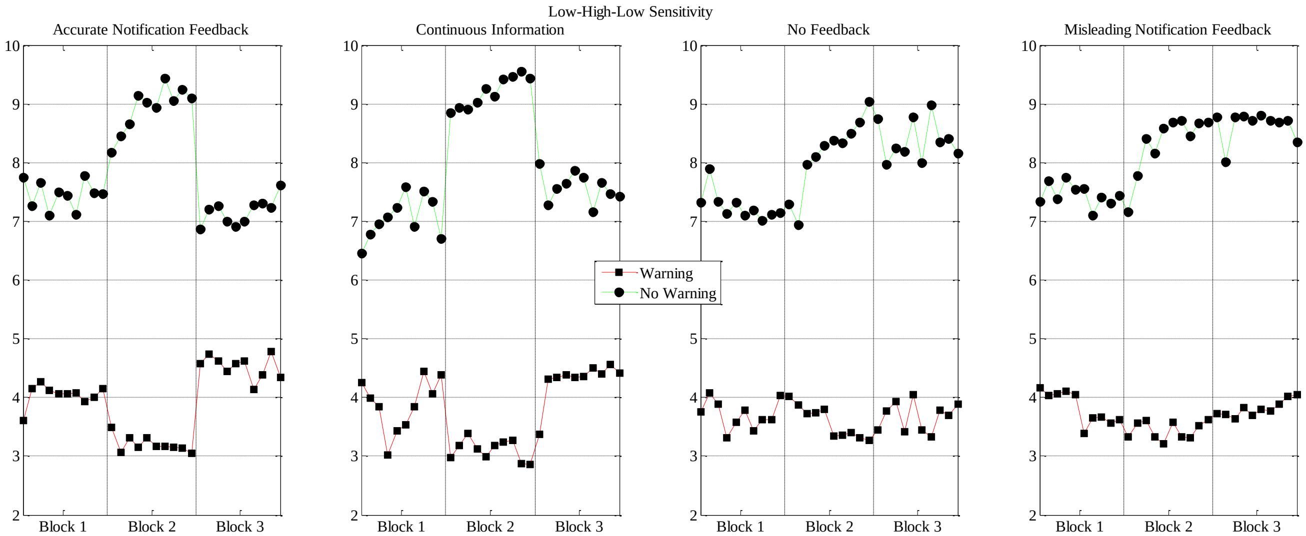

Figure 21: Evolution of operators' choice of L in the Low-High-Low setting for all tested conditions.

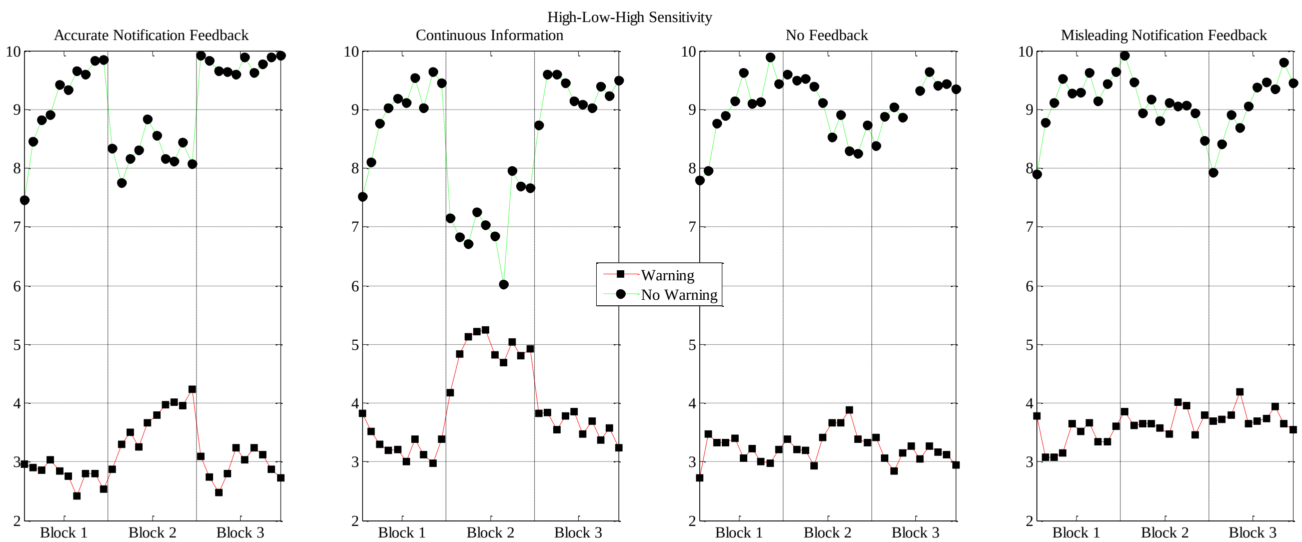

Figure 22: Evolution of operators' choice of L in the High-Low-High setting for all tested conditions.

4.7. Conclusion

These results demonstrated the **benefit of providing feedback about changes in the characteristics of an automated system in the form of *notifications* or *Continuous Information* to the operator in human-machine collaboration**. In this experiment the automated system was a warning system. **With *notifications* or *Continuous Information* the operator successfully and quickly adjusted the two dimensions of its trust, compliance and reliance, to both positive and negative changes of system sensitivity. The adaptation was slightly faster when the feedback was provided in the form of *Continuous Information***.

In the absence of proper feedback, this adaptation is either slow or inexistent, thus creating a temporary or permanent situation of over-reliance, under-reliance, over-compliance or under-compliance. **When exposed to a first change of system sensitivity, operators slowly and gradually adapted their reliance but do not adapt their compliance**. Interestingly, when exposed to a second change of system sensitivity opposite to the first one, their response varied depending on the direction of these changes: if the first one decreased the sensitivity and the second one increased it, they gradually adapted their reliance after the second change too. However, if the first change increased the sensitivity and the second decreased it, no adaptation of reliance is observed after the second change. In both cases there was no adaptation of compliance, which highlights the difference between the two dimensions of trust, reliance and compliance. **These results show the importance of previous experience in the operator's response and the bias provoked by a positive change**: after a sensitivity improvement, operators have more trouble detecting a sensitivity degradation when provided with no feedback.

Additionally, **when misleading notifications were present, operators behaved exactly as if no feedback was given. They were capable of immediately discarding the wrong notifications.** However, they were not able to use them as a cue that the warning sensitivity changed.

5. The effect of the level of automation of the learning on the use of learning robot system

5.1. Introduction

After studying the effect of providing feedback about the environmental changes, the effect of the number of changes, and the effect of providing feedback about the system changes, this experiment focuses on the influence of the level of automation at which a learned behavior is applied. Contrary to the two previous experiments, it was conducted using a setup comprising a real mobile robot. Participants were recruited and assigned to four different groups corresponding to four different settings of level of automation: *no learning*, *suggestions*, *approvable suggestions* and *switchable automation*. They were asked to control a robot in a decision making task in parallel of solving a mathematical problem task. Their performance and workload were assessed and compared using various metrics.

5.2. Methods

A. Participants

Participants were 48 undergraduate engineering students, without previous experience with the system. They were recruited using a mailing list. They received either 30 NIS (New Israeli Shekels, about 8 dollars) or 1 bonus point at the final exam of the automation course for their participation in the experiment and could get a bonus of 100 NIS (about 26 dollars) according to their performance. The recipients of the bonus were determined with a lottery, with each point in the cumulative score serving as a lottery ticket. Thus, the higher a participant’s score, the more virtual lottery tickets he received. The participants received instructions on how to use the system in the form of an interactive tutorial.

B. Experimental platform

To study the impact of the control over the learning in a controlled environment, the experiment was conducted in the lab. The experimental platform consisted of a Pioneer LX mobile robot (see Figure 23) running ROS (Robot Operating System) with the navigation stack and a control interface running on a remote computer developed in C# and WPF for PC. The control interface and the robot were communicating over a wifi network through web sockets using the ros-bridge package on the robot side. Participants sat at the control station made of the computer running the control interface and a 24 inches monitor with a 1920x1080 pixels resolution. This station was overlooking the lab and hence enabled the participants to see the 6 by 8 meters area where the robot was evolving.

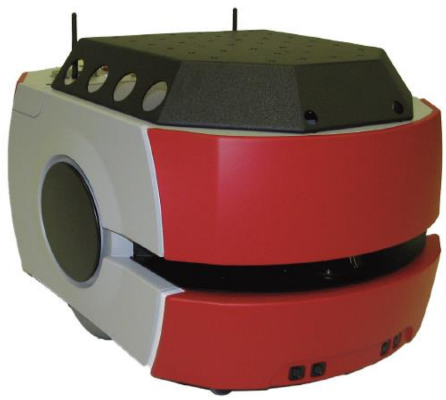

Figure 23: Pioneer LX mobile robot platform

C. Task

Participants were asked to try to finish with the highest score. They conducted two tasks in parallel on the control station. The left side of the screen displays the main task and its related controls, and the right side displays the secondary task (Figure 24). The bottom right part shows the remaining time (time left) before the end of the session and the score (computed from the performance in both tasks). The initial score is 100. The experiment was 30 minutes long and virtually divided in 3 blocks of 10 minutes which are referred below as block 1, block 2 and block 3.

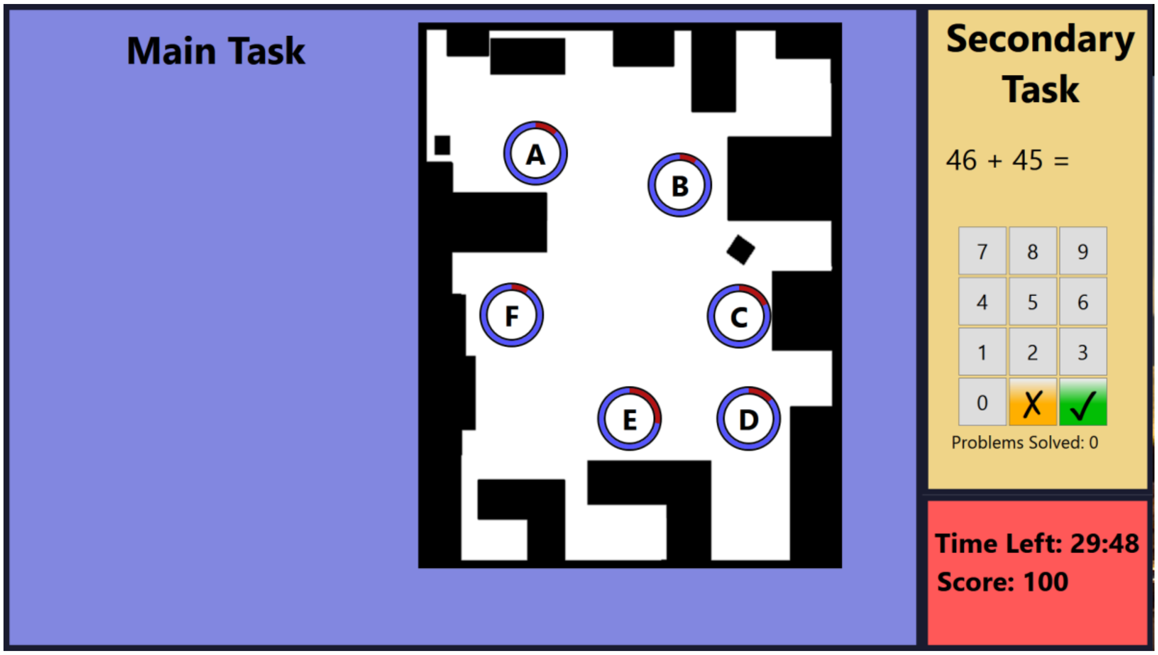

Figure 24: Experiment Interface

In the main task the participants had to control the mobile robot through the remote interface. The interface provided feedback of the position of the robot on a 2D map of the lab where the experiment was taking place. The user could also directly see the physical robot. The task of the user was to send the robot the different zones marked on the lab floor, numbered from A to F. Zones had to be visited often enough by the robot. The scenario was that these zones represented points of interest that the robot had to monitor by visiting. The maximal visit period is different for each zone and changes twice during the experiment, hence each experimental block has a different maximal visit period setting (see Table 3). When the maximal visit periods change, a notification informing the participant of the change is displayed for 10 seconds and the whole main task area on the interface blinks in orange.

If the time separating two visits to a zone is superior to the maximum visit period, 2 points are removed from the score. For each zone the time left to visit it before the loss of 2 points can be visualized through the progression of a red arc circle inside the blue circle marking a zone: when the red arc circle forms a circle, it means the time ran out (see Figure 25). When it happens a small animation shows “-2” in red next to the zone and the zone blinks in red.

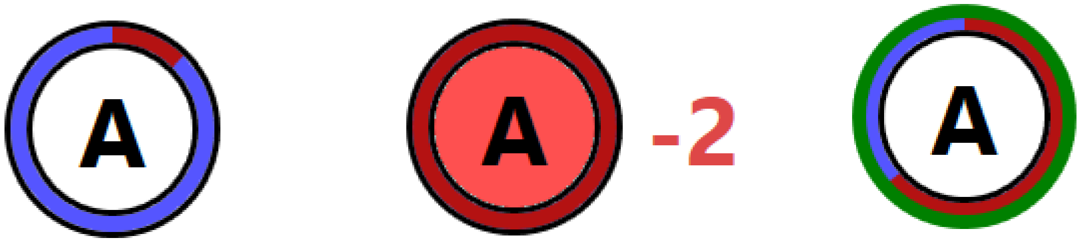

Figure 25: Symbol for a zone (left), with the red arc circle representing the progression of time; animation signaling the loss of 2 points when the robot did not visit the zone in time (center); and green highlighting when a zone is designated as a target by the participant (left).

To send a robot to a desired zone, the participant had to click on it, then the robot automatically computes its route and moves to the target. While the robot is moving toward a zone, the targeted zone is highlighted in green (see Figure 25) on the interface and no other zone can be targeted. Once the robot reaches the target, the time before a loss of point for this zone is reset to zero (and the red arc circle too) and the time is frozen until the robot leaves. When the robot is ready to go to another zone, the green highlighting disappear and a “ding” sound is played.

Zone	**Maximal visit period (seconds)**		
	Block 1	**Block 2**	**Block 3**
A	90	80	120
B	120	90	80
C	60	40	120
D	80	120	60
E	40	120	90
F	120	60	40

Table 3: Maximal refresh period zone configuration for each experimental block

In the secondary task, the participants had to add two two-digit numbers. They gained one score point for each correct answer and a wrong answer had no effect. The secondary task was designed as to create an incentive for the user to be as efficient as possible in the main task (and hence use the automation when possible).

D. Design

A between-subject design was used with 48 participants divided between 4 experimental conditions corresponding to four levels of automation of the learning of the robot.

a) No automation

In the first condition, *no automation* is available to the participants. No automation, reset or learning information or options are displayed on the interface.

b) Robot suggestions

In the second condition, the systems learns from the manual choices of zone done by the user the frequency at which each zone was visited. After 2 minutes of manual interaction and if at least 2 zones have been visited, the system shows a notification informing the participant that suggestions of which zone to visit are now available. From this point, a *suggestion* is displayed in the form of “Go to zone X” below the map of the lab, see Figure 27. The participant can reset the learning of the system by clicking on the reset button, then the system will need another 2 minutes before been able to provide suggestions again. When *suggestions* are displayed, the system improves its learning only if the zone clicked is not the suggested one.

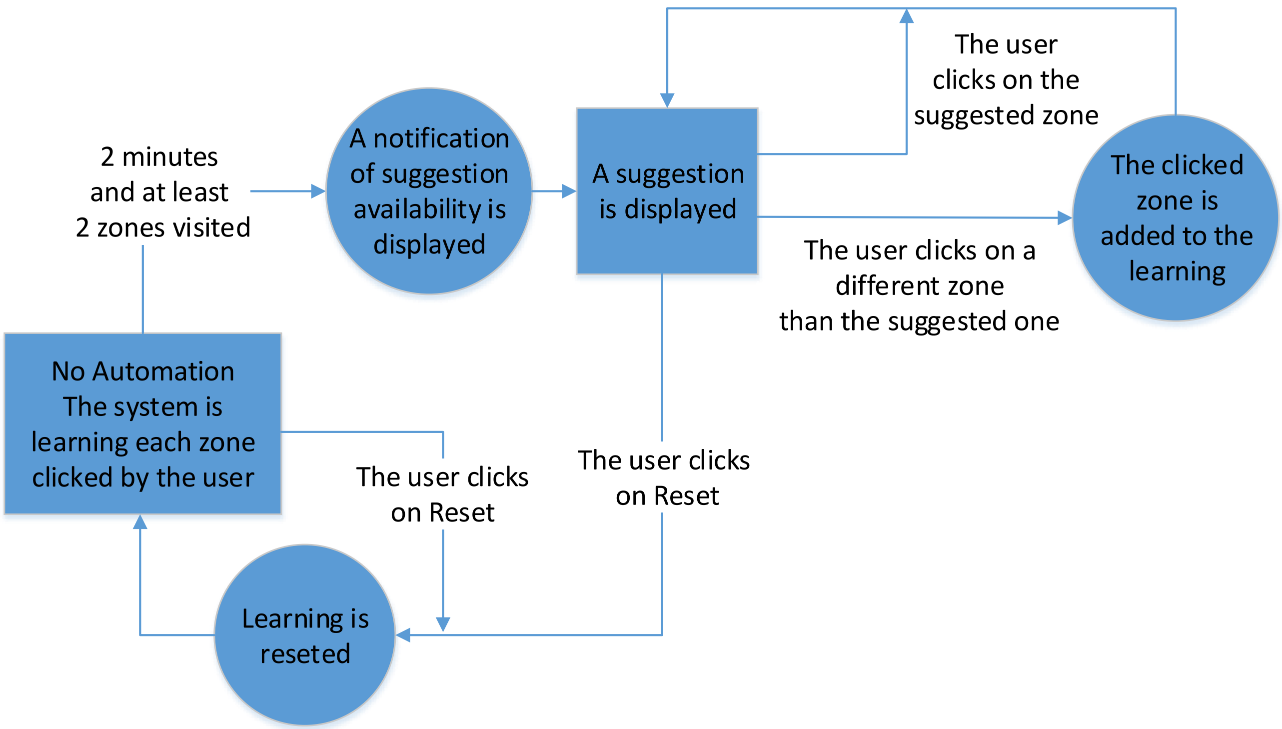

Figure 26: Suggestions experimental condition

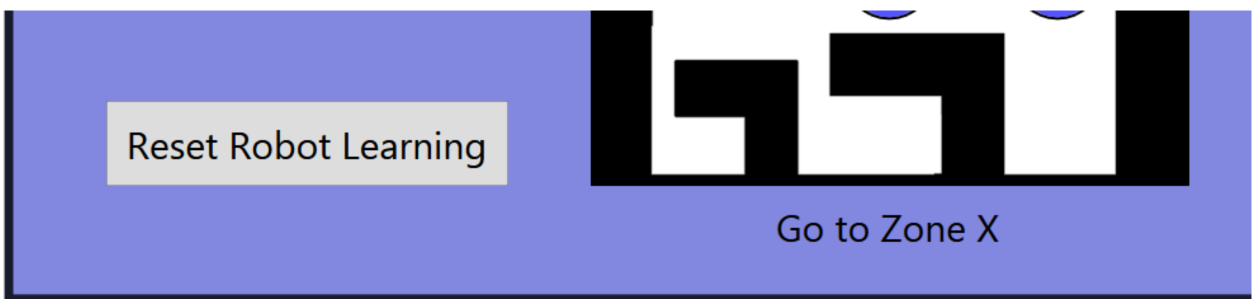

Figure 27: Lower part of the main task interface when suggestions are available in the second experimental condition; ”X” in the suggestion text can be A, B, C, D, E or F.

c) Approvable suggestions

The third condition is similar to the second condition but the participants have the possibility in addition to approve the suggestions by clicking on the suggestion which is now a button, see Figure 29. If the suggestion button is clicked, the robot goes automatically to the suggested zone. Alternatively, a suggestion can be approved by pressing the Enter key on the keyboard.

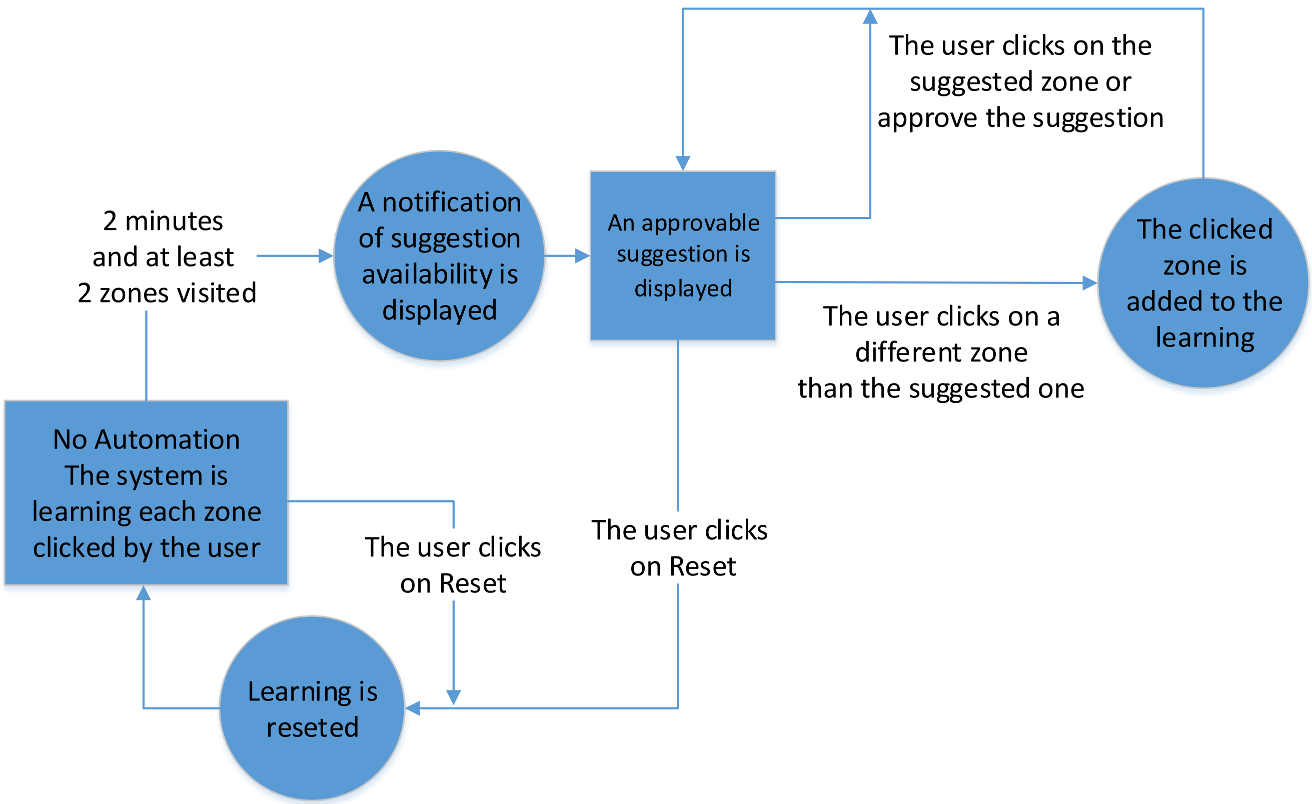

Figure 28: Approvable suggestions experimental condition

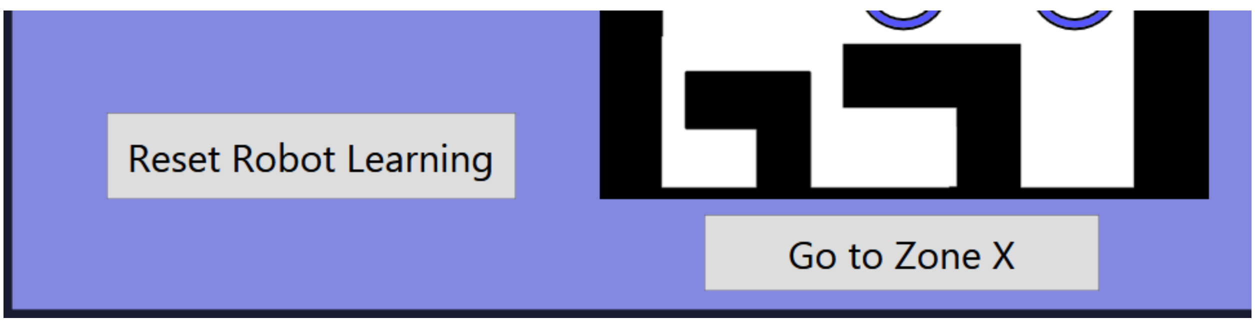

Figure 29: Lower part of the main task interface when approvable suggestions are available in the third experimental condition; ”X” in the suggestion button text can be A, B, C, D, E or F.

d) Switchable automation

In the fourth condition, the system learns the frequency of the zone visits as in the second and third condition. After 2 minutes of learning, a notification indicates that the automatic mode of the robot is now available and the “Automatic” part of the control mode switch is activated. As long as the user does not click on the “Automatic” switch, the system continues to learn.

The user can then click on the “Automatic” switch, which makes the robot visit the different zones automatically according to what it learned before.

The participant can click at any time on the Reset button to reset the learning.

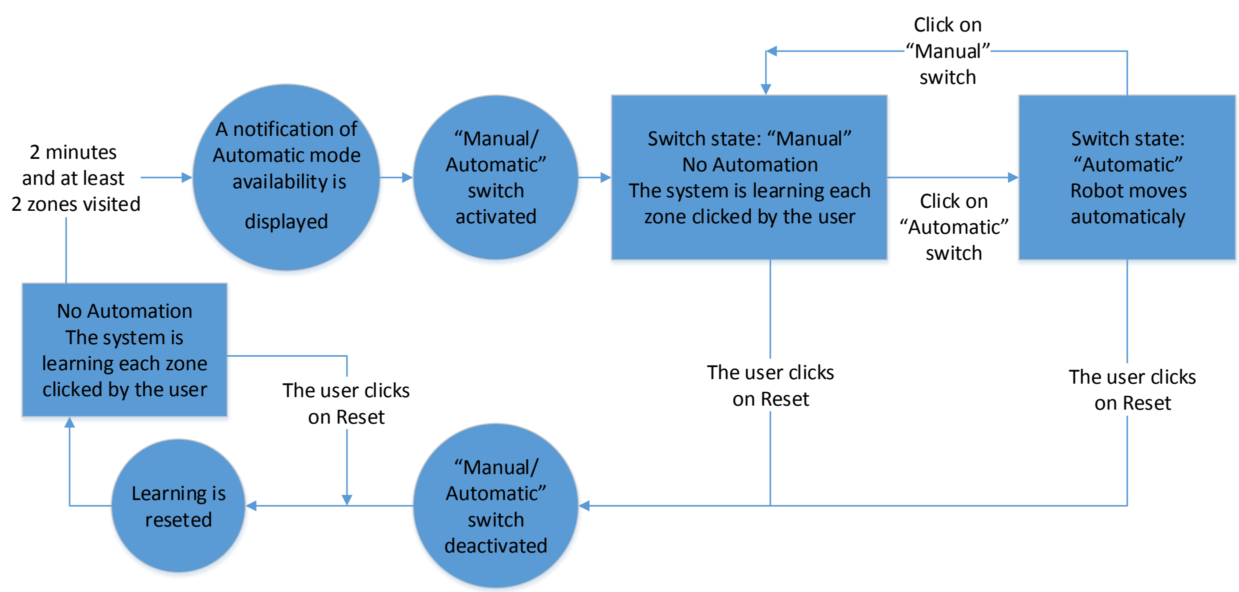

Figure 30: Switchable Automation experimental condition

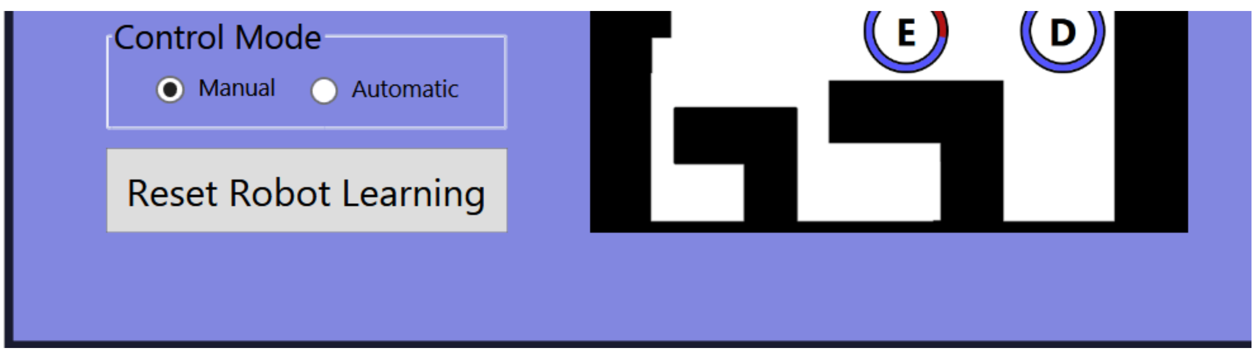

Figure 31: Lower part of the main task interface when the automatic mode is available in the fourth experimental condition

E. Dependent measures

Evaluation was conducted for both objective and subjective measures. With the following metrics we measured performance and how the participants made use of the learning capabilities of the robot:

1) *Number of failures in the main task.* This is the performance measure in the main task, the number of times the participant failed to make the robot visit a zone within the time constraint, the lower the better.
2) *Number of math problems solved.* This is the performance measure in the secondary task. The more math problems were solved, the better the performance in the secondary task.
3) *Score*. The overall performance score, as measured through the final score displayed at the end of the session:

   *Score = 100 + Number of math problems solved – 2 * Number of glasses lost*
4) *Use of automation*. The ration of the number of zone reached as a result of a suggestion or of the automatic mode, over the total number of zone reached while the suggestions or the automatic mode were available. Not applicable for the *No learning* condition.
5) *Number of resets*. The number of times the subject performed a reset of the learning of the robot. Not applicable for the *No learning* condition.
6) *Increase in heart rate relative to baseline*. Heart rate (HR) may be used as an indication of the physiological state of the participants and may be indicative to workload levels, fatigue and physiological strain (Roscoe 1993; Roscoe 1992; Turner & Carroll 1985), such as in (Harriott & Zhang 2011). Hence, it was used in this research to objectively measure workload. It was measured using a Polar H7 chest sensor in a resting state (baseline) and during each trial. The variation in percentage between the baseline and each trial was then calculated and used as a measure.

Additionally, subjective workload was assessed through the raw NASA-TLX questionnaire (Hart & Staveland 1988), similarly as in (Nielsen et al. 2007). Each measure was defined with a 100-points range with 5-point steps:

1) Mental Demand
2) Physical Demand

3) Temporal Demand
4) Performance
5) Effort
6) Frustration

F. Data analysis

An ANOVA test was used to evaluate the effect of the control over the learning on the dependent measures defined in the previous section. When a significant effect was found, a post-hoc LSD pairwise comparison was run to compare each condition. For the objective dependent measures the effect is considered significant at or below $\alpha = 0.05$. For the subjective dependent measures, the effect is considered significant below $\alpha = 0.1$.

G. Procedure

The participants were first asked to wear the heart rate sensor. Once equipped with the heart rate sensor, they had to read and sign a consent form informing them about the conditions of the experiment. They were then asked to relax and were presented with a 5 minutes long video unrelated to the experiment showing relaxing pictures of cascades and rivers. The last 2 minutes of the video were used to determine their heart rate baseline.

Then they watched a demonstration of the system for 3 minutes where all the element of the interface were explained. The explanation they received about the robot learning is that it will "reproduce the frequency at which they visited the zones". They were informed that something could change during the experiment but not what.

After the demonstration the participants could start the experiment. At the end of it they had to complete the raw NASA-TLX questionnaire and were paid.

5.3.Results

A. Overall performance

The overall participants' performance during the experiment (i.e. their Score) was significantly impacted by the level of automation of the learning they could use ($F(3,44)=6.72$, *p*=.001), see Figure 32. **Participants from the group which had access to *switchable automation* performed overall significantly better than participants from any other group**: they scored an average of 330 points, SD=127.1, compared to 237 points, SD=48.8, $p<0.05$, for the group with *no automation*, 161 points, SD=98.6, $p<0.001$, for the group with simple *suggestions*, and 221 points, SD=81.5, $p<0.05$, for the group with *approvable suggestions*. There was no significant difference in terms of overall performance between the two groups receiving suggestions. However, the group with *no automation* performed significantly better ($p<0.05$) than the group with simple *suggestions* and had no significant difference with the group with *approvable suggestions*.

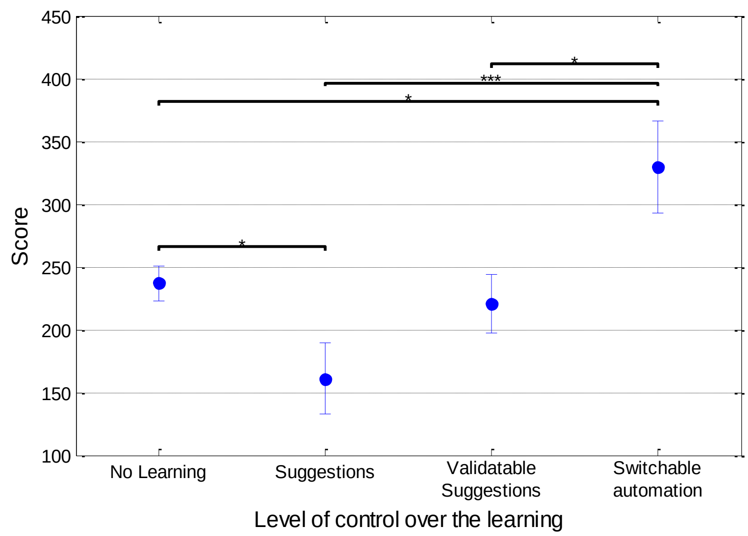

Figure 32: Effect of the level of automation of the learning of the robot on the overall performance (i.e. Score). Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1% and 0.1%. Error bars represent the standard error of the mean.

B. Main task performance

The overall performance being a composite metric of the performance in the main task and in the secondary task, analyzing the latter two is also of interest. As we will see, **the level of automation of the learning of the robot** does not have the same effect on the two tasks. It **significantly impacts the performance in the main task** ($F(3,44)=9.382$, $p<0.001$), see Figure 33. Performance progress (and number of failures decreased) as the level of automation of the learning of the robot increase: 37.6 failures, SD=2.84, with *no automation*, 36.58 failures, SD=6.61, with simple *suggestions*, 30.83 failures, SD=4.71, with *approvable suggestions*, and 21.6 failures, SD=14.19, with *switchable automation*. The group with *switchable automation* performed significantly better than all other groups ($p<0.001$), the group with *approvable suggestions* performed better than the group with *no automation* ($p<0.05$), but no significant difference was found between the group with *no automation* and the group with simple *suggestions*.

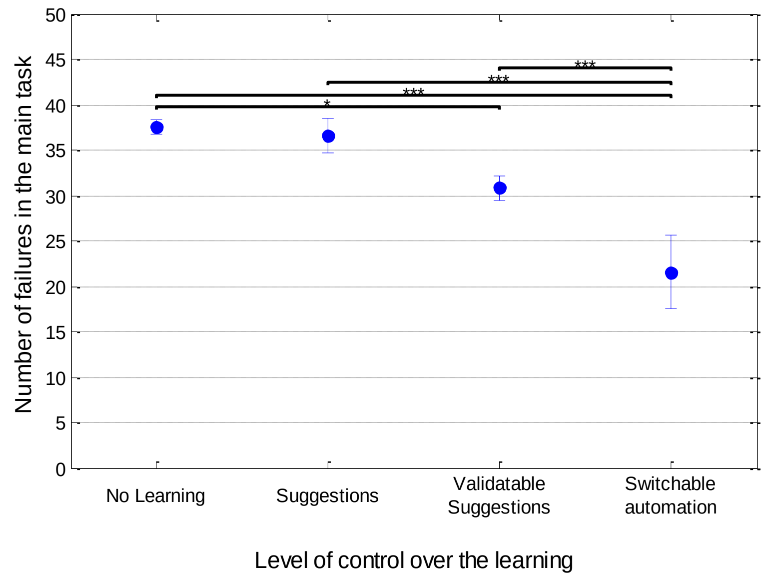

Figure 33: Effect of the level of control over the learning on the performance in the main task (i.e. number of failure in the main task), lower is better. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1% and 0.1%. Error bars represent the standard error of the mean.

C. Secondary task performance

The performance in the secondary task (i.e. the number of math problems solved) was significantly impacted by the level of automation of the learning of the robot ($F(3,44)=4.346$, $p=0.01$), see Figure 34. **The group with *switchable automation* performed significantly better than the other two groups with learning**, 273 problems solved, SD=132.2, compared to 182.50 problems solved, SD=82.1, $p<0.05$, with *approvable suggestions* and 134.2, SD=104.3, $p<0.01$, with simple *suggestions*. However there was no significant difference between the group with *switchable automation* and the group with *no automation*, the later having a significantly higher number of problem solved than the group with simple *suggestions* (212.1, SD=46.5, compared to

134.2, SD=104.3, p<0.05). There was also no difference between the two groups with suggestions.

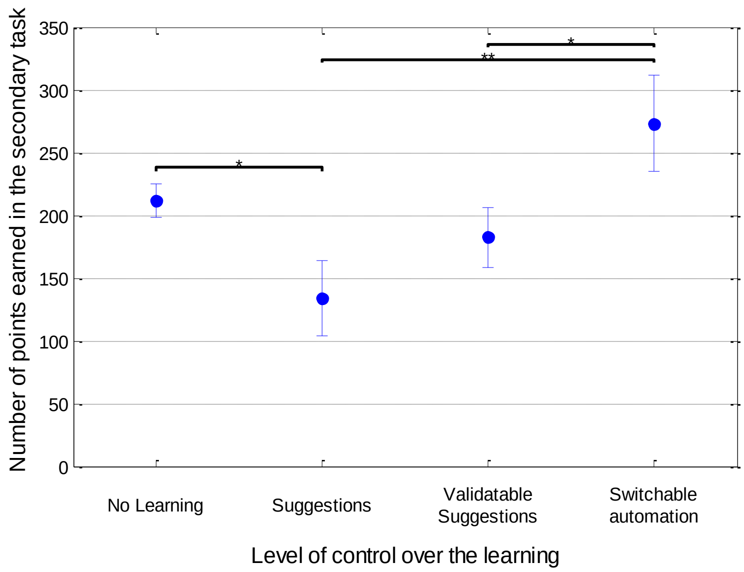

Figure 34: Effect of the level of control over the learning on the performance in the secondary task (i.e. number of math problems solved). Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1% and 0.1%. Error bars represent the standard error of the mean.

D. Use of automation

The two metrics reflecting how the participants made use of the learning capabilities of the system, the use of automation and the number of reset, concern only the groups for which the learning was available: *simple suggestions, approvable suggestions* and *switchable automation*. **The use of automation was significantly impacted by the level of automation of the learning of the robot** ($F(2,33)= 13.558$, $p<0.001$), see Figure 35. The group with *switchable automation* used significantly more the automation then the two other groups: 61.9%, SD=8.19, compared to 31.4%, SD=1.6, p<0.001, with *approvable suggestions*, and 29.9%, SD=1.5, p<0.001, with simple *suggestions*. However, no difference was found between the latter two groups.

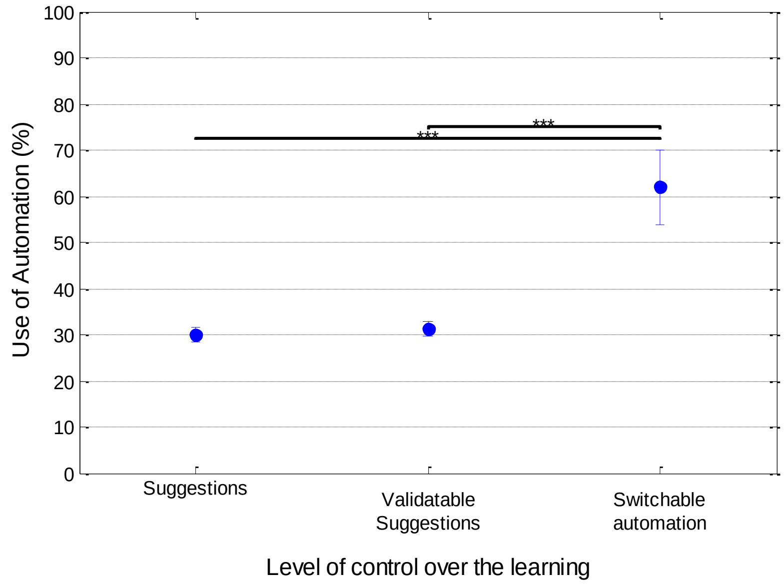

Figure 35: Effect of the level of automation of the learning of the robot on the use of automation. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1% and 0.1%. Error bars represent the standard error of the mean.

No significant effect of the level of automation of the learning of the robot was found on the number of resets of the learning performed by the participants ($F(2,33)= 0.646$, $p=0.531$), see Figure 36.

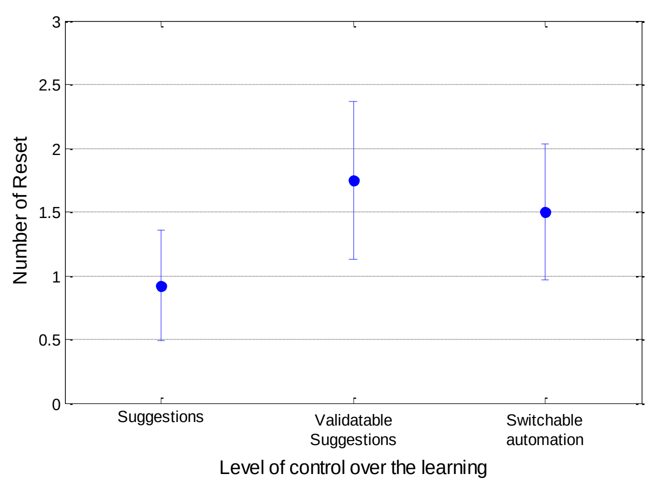

Figure 36: Effect of the level of automation of the learning of the robot on the number of reset. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1% and 0.1%. Error bars represent the standard error of the mean.

E. Subjective workload

Over the 6 dimensions of workload tested with the raw NASA-TLX questionnaire, **only the mental demand revealed to be significantly affected by the level of automation of the learning of the robot** (F(3,44)=2.528, *p*=0.07), see Figure 37. The mental workload appeared to be lower for the participants from the groups with *approvable suggestions* (42.08, SD=21.15) and *switchable automation* (44.17, SD=16.77) than for the participants from the groups with *no automation* (59.17, SD=26.01) and simple *suggestions* (58.33, SD=12.49).

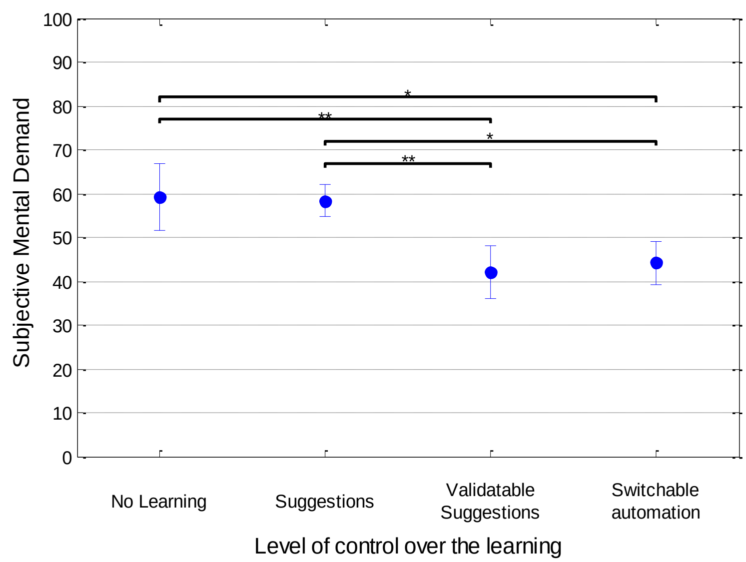

Figure 37: Effect of the level of automation of the learning of the robot on the subjective mental demand. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 10%, 5% and 1%. Error bars represent the standard error of the mean.

F. Heart rate increase relative to baseline

The increase of heart rate relative to the baseline is significantly affected by the level of automation of the learning of the robot (F(3,44)=3.264, *p*<0.05), see Figure 38. The three groups *no learning*, simple *suggestions* and *approvable suggestions* are not significantly different. However the participants of the group with *switchable automation* had a significantly higher relative hear rate, 7.2%, SD=0.97, than the participants of the groups with simple *suggestions*, 2.65%, SD=0.95, p<0.01, and with *approvable suggestions*, 3.73%, SD=1.49, p<0.05. However no significant difference appear between the group with *no automation*, 4.98%, SD=0.79, and any other group.

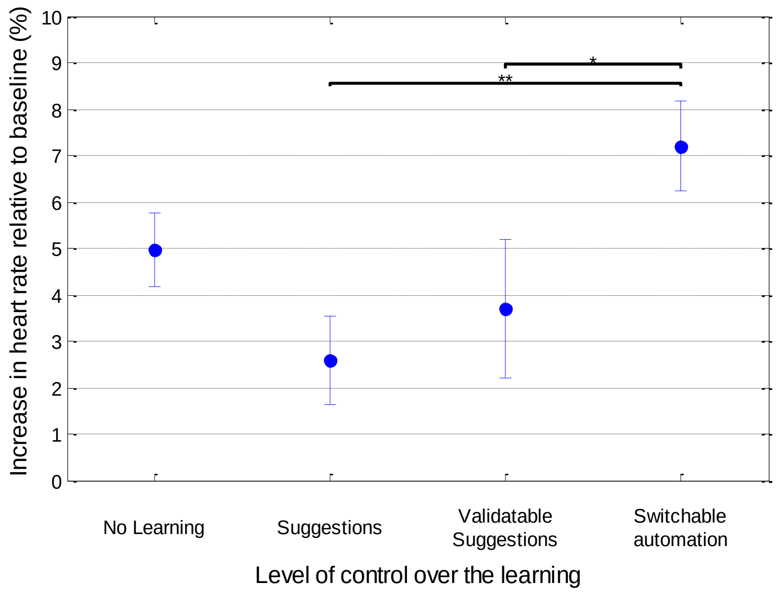

Figure 38: Effect of level of automation of the learning of the robot on the increase of heart rate relative to baseline. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1% and 0.1%. Error bars represent the standard error of the mean.

5.4.Discussion and Conclusion

The results indicate that **in the context of online robot learning in changing environments, the level of automation of the learning of the robot has an important effect on the performance of the task learnt, on the way users make use of the learning and on the performance of a parallel task**. In fact, adding the possibility for the robot to learn a task is not necessarily beneficial and depends on the way the user could make use the automation gained from the learning.

In the setting with the highest level of automation of the learning of the robot, *switchable automation*, for which the participants could chose to switch between a fully autonomous behavior of the robot or a fully manual mode, the overall performance was the best. The participants made the best use of the automation, choosing to activate the automatic mode 62% of the time it was available. Hence, they had more time to dedicate to the secondary task which explains why they performed the best there. Interestingly they also outperformed the participants with the lower levels of automation of the learning in the main task, meaning that **the highest level of automation did not only enable them to perform better in the secondary task, but**

also boosted their performance in the main task. However, their increase in heart rate was higher than the other groups, indicating a higher level of stress or workload. This could be explained by the participants' higher activity in the secondary task (more math problems solved). However, this is somewhat contradictory with the result of the subjective mental demand metric which indicate a lower workload for the group with the switchable automation than for the groups with no learning and simple suggestions.

It appears that the group with the simple *suggestions* setting is the one which performed the worst. Overall, its performance was lower than with *no automation*. Going into the details of the main and secondary task, we observe that participants solved less math problems when receiving simple *suggestions* than when having access to *no automation*, and in the main task there is no difference in terms of performance between the condition with *no automation* and the condition with simple *suggestions*. Additionally, thus no different than with *suggestions*, the group with simple *suggestions* had a higher subjective mental demand than with higher levels of automation of the learning, but not a significantly different increase of heart rate. These poor results of the group with simple *suggestions* compare to *no automation* can be explained by the additional attention required to train and proceed the suggestions for very little benefit: the users still have to manually locate and click on the suggested zone while checking if the suggestion is relevant. Hence, here it is better to have no learning at all than presenting the user with simple learned *suggestions*. However it would be of interest to study in the future the influence of the reliability of these suggestions.

The group with *approvable suggestions* seems to have a slight advantage compared to the group with *no automation*. No difference is seen in terms of overall performance and performance in the secondary task, but the setting with *approvable suggestions* had a better performance in the main task and a lower subjective mental demand. This small gain in the main task and the lower mental demand can be understood if hypothesizing the subjects approved the suggestions as long as they did not see too much failures in the main task, hence saving time and workload on the decision process of where to send the robot.

These results highlight the fact that when interacting with a learning robot in a changing environment, the level of automation at which is applied the learning has a big impact. We found that the performance in the learned task increase with the level of automation of the learning: the highest the level of autonomy at which the learning of the robot could be applied by the participants in the main task, the better the performance was. However, when the participants have a secondary task to complete in parallel, exploiting the learning with the lowest level of automation, simple *suggestions*, had a negative impact and the overall level of performance was below the participants that did not use learning at all. Using the highest level of automation of the learning was still the best solution. Though in this experiment, the participants were informed through notifications of the changes in the environment. Hence, one can argue that without notifications, as seen in the Chapter 3, the performance of the *switchable automation* group would be lower, which could maybe make the setting with no learning superior in this specific case.

Part B: Advanced Human Robot Interfaces

In this part, a more practical approach of interface development was taken. Novel interfaces and algorithms were created using recent advances in the field of sensors and in particular the release of cheap RGB-D sensors like the Microsoft Kinect.

Four new interaction modalities were created and then evaluated: person following, pointing control, direct physical interaction and camera head control for remote robot teleoperation. In a first section, a mobile robotic platform dedicated to person following was created from a customized generic differential drive robot. Then using this platform, person tracking and person following algorithms were developed and practically tested to achieve a robust person tracking behavior in complex office environments (chapter 6). Next in a second section (chapter 7), using the platform developed in the first section, two more interaction modes were developed and added: a pointing control interface and a direct physical interface. These three interfaces had then their usability tested in two robot navigation control tasks which took place in a test home apartment with 24 participants. Finally, in a third section (chapter 8), a new non-intrusive method based on the use of a Kinect sensor for controlling the orientation of the camera through the operator’s head orientation in a robot teleoperation task was developed and compared to more classical interfaces. The performance of the interfaces tested was also evaluated against user familiarity with the system. 36 students participated.

6. Adaptive Person-Following Algorithm Based on Depth Images and Mapping

6.1.Introduction

Person following by a mobile autonomous robot includes two tasks, person tracking and safe robot navigation. Two person-following algorithms that use depth images from a Microsoft Kinect sensor for person tracking are developed and tested in this experiment. The first one, the path-following algorithm, reproduces the path of the person in the environment. The second one, the adaptive algorithm, uses in addition a laser range finder for localization and dynamically generates the robot's path inside a pre-mapped environment, taking into account the obstacles locations. The Kinect was mounted on a pan-tilt mechanism to allow continuous person tracking while the robot followed the generated path. The two algorithms were tested and their performance compared in a series of trials where the robot had to follow a person walking in an environment with obstacles.

6.2.Methodology

A. Algorithms

Two person-following algorithms are developed and compared: a *path-following algorithm* and an *adaptive algorithm*. These two algorithms use other algorithms to control the robot and to track the position of the person and estimate its position (described in section 6.3).

B. Hardware

The two algorithms were implemented on a generic differential drive mobile platform, a Robosoft robuLAB10, with two propulsive wheels and two castor wheels, which comes with basic navigation functions. The robuLAB10 was customized with a rigid structure including three tubes and a tray for laptop PC (Figure 39). On the top of this structure a TRACLabs Biclops pan-tilt mechanism (PT-M) and a Kinect sensor were added. For navigation purposes, the base is equipped with a SICK S300 LRF, which is positioned at the height of 0.24mand provides distance measurements of up to 30min an angular field of view of 270°.

The pan-tilt mechanism has a tilt range of 120 deg and a pan range of 360deg with a maximum angular velocity of 170deg/s and a maximum angular acceleration of $3000\ \mathrm{deg/s^2}$. The precision of the angular position measurements is ± 0.01 deg. The mechanism can support a maximum payload of 4kg which is more than the weight of the Kinect sensor. In all experiments, the tilt value was set to 0deg and person tracking was performed only in the horizontal plane, using the pan axis of the pan-tilt mechanism only. The communication between the laptop PC and the mechanism is maintained via a USB port with a data transfer rate of up to 416kbps.

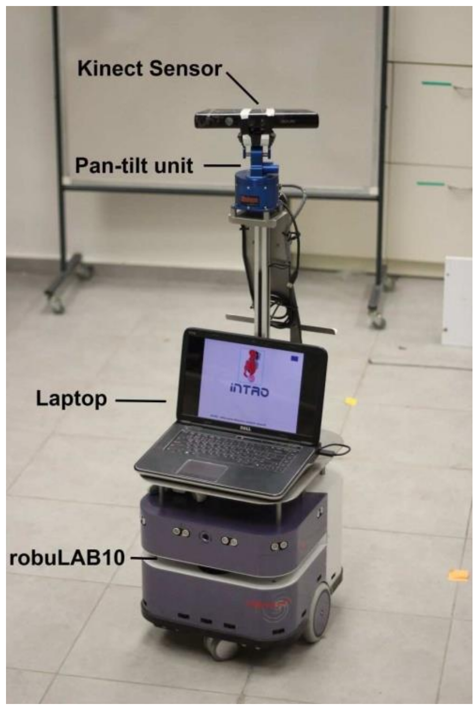

Figure 39: RobuLAB10 robotic platform with Biclops pan-tilt mechanism, Kinect sensor, and laptop PC.

The Kinect sensor is equipped with an infrared light projector, a depth sensor, a RGB camera, and a multi-array microphone. It also has a motorized tilt that was disabled and was used only for sensor positioning. The depth sensor range is from 0.8mto 6mwith the vertical viewing angle of 43° and horizontal viewing angle of 57 deg. It provides depth images at the resolution of 640×480 pixels at the maximal frame rate of 30fps. The Microsoft Kinect SDK provides person detection and person joints position tracking features up to $4m$.

The laptop PC used in this work is powered by an Intel quad-core i7 Q740 CPU with 4 GB of RAM.

C. Experimental setup

Two sets of experiment were conducted. The first set focused on the performance evaluation of the path-following person following algorithm and the second set focused on the adaptive person following algorithm. In all experiments, the person was instructed not to assist the robot and to walk at a constant speed along a marked path on the ground, regardless of the robot's tracking and/or following performance. This marked path on the ground makes the person travel around obstacles as seen in Figure 41 and Figure 42.

D. Performance analysis

The following performance metrics were used for each trial of each experimental setups to evaluate the proposed person-following algorithms: 1) Path-completion ratio: the length of the ground path from the person start point to the closest point of the robot end point, divided by the total length of the ground path, 2) Number of loss-of-track events: number of events when tracking of the person was lost in a single trial; loss of tracking is defined when no position estimation is provided by the Kinect SDK for a period longer than 500ms, and 3) Robot path length to person path length ratio: the distance travelled by the robot divided by the distance travelled by the person.

For each set of experiment, 10 trials were conducted. For the path-following algorithm evaluation, the error between the person's path and robot's path was computed in addition to the metrics described above. This path error is calculated by resampling robot path data to regular space interval of 1cm and calculating for each resampled point of the robot path the closest distance to the ground path followed by the person.

6.3. Algorithms

A. Robot control

The robuLAB10 platform uses RobosoftrobuBOX open source library. The robuBOX is based on the Microsoft Robotics Developer Studio (MRDS) and written in C#. Its most important component is the Core, which contains the definitions of robots actuators and sensors. All other components interact through these definitions either by implementing or using them. For robot navigation three robuBOX features were exploited, namely the obstacle collision detection, the differential-drive controller and the path follower.

The obstacle collision detection feature uses the LRF distance measurements and applies two parameters to control the robot's motion. At distances between 0.3mand 1mfrom an obstacle the robot speed is reduced proportionally to the distance value. The robot is finally stopped at the distance of 0.3mfrom the obstacle. The distances are calculated within the robot frame with its origin in the point P_m located at mid-distance of the actuated wheels.

The differential-drive controller is used to set robot's linear and angular speeds. The wheels' velocities are derived from these values by the robot's low-level controller.

The path follower feature allows the robot to follow a list of path points that are added to the buffer and executed sequentially. The path follower implements Morin-Samson's path following with no orientation control (Morin & Samson 2008). We consider a path $\mathcal{C}$ in the plane of motion, as illustrated on Figure 40.Let us define three frames $\mathcal{F}_0$, $\mathcal{F}_m$, and $\mathcal{F}_s$, as follows. $\mathcal{F}_0 = \{0, \vec{i}, \vec{j}\}$ is a fixed global frame, $\mathcal{F}_m = \{P_m, \overrightarrow{i_m}, \overrightarrow{j_m}\}$ is a frame attached to the mobile robot with its origin in the point P_m, and $\mathcal{F}_s = \{P_s, \overrightarrow{i_s}, \overrightarrow{j_s}\}$, which is indexed by the path's curvilinear abscissas, is such that the unit vector $\overrightarrow{i_s}$ tangents $\mathcal{C}$. The control point P is attached to the robot chassis, with the coordinates (l_1, l_2) expressed in the basis of $\mathcal{F}_m$. In the experiments the following values were set: $l_1 = 0.15m$ and $l_2 = 0$.

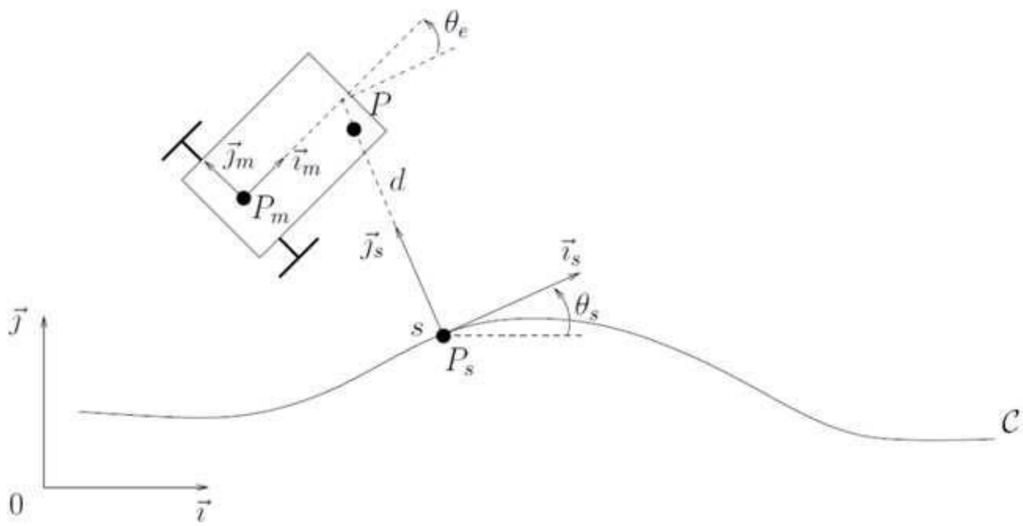

Figure 40: Representation of the path in the robot motion plane (Morin & Samson 2008)

To determine the equations of motion of P with respect to the path $\mathcal{C}$ let us define d as the distance between P and $\mathcal{C}$, and $\theta_e = \theta_m - \theta_s$ as the angle characterizing the orientation of the robot chassis with respect to the frame $\mathcal{F}_s$. Where θ_m is the orientation of the robot chassis in the global frame $\mathcal{F}_0$.The control objective is to stabilize the distance d at zero. For that, the following feedback control law was applied:

$$u_2 = u_1 \left(\frac{\tan \theta_e}{l_1} - k_0 \cdot d \right) \qquad (1)$$

Where u_1 and u_2 represent the intensities of the robot's longitudinal and angular velocity, respectively, and k_0 is a constant. The detailed proof that d exponentially converges to zero when u_1 is constant and $\theta_e \in (-\pi/2, \pi/2)$ can be found in (Morin & Samson 2008). The following values were set to $u_1 = 0.5 m/s$ and $k_0 = 20$. As a measure of precaution, the maximal heading error was set to $\theta_{e,max} = 60^{\circ}$, which initiates a recovery procedure that stops the robot and sends it to the last path point in the buffer.

B. Person tracking and position estimation

Tracking of person's skeleton joints is performed for each depth-image frame in the Kinect SDK, using no temporal information (Shotton et al. 2011).The algorithm uses the variation in depth to find different body parts and applies Random Decision Forests to compute estimated joint positions. It is also able to distinguish between two different persons. The 3D position of the head joint outputted by the algorithm was used to estimate the ground X and Y position of the person. This allows keeping track of the person position in presence of obstacles small in height causing an occlusion of the lower body parts. The outputted person ground position estimation is in the frame of reference of the Kinect sensor. It must be converted in the global frame of reference in order to be used by the path-following algorithm.

To calculate the position estimation in the global frame three direct orthonormal frames of reference were considered:

1) The fixed global frame $\mathcal{F}_0 = \{0, \overrightarrow{i_0}, \overrightarrow{j_0}\}$.
2) The frame attached to the robot $\mathcal{F}_m = \{P_m, \overrightarrow{i_m}, \overrightarrow{j_m}\}$. P_m is at the center of the robot and both $\overrightarrow{i_m}$ and $\overrightarrow{j_m}$ are in the horizontal plane; $\overrightarrow{i_m}$ is pointing in the forward direction of the robot.
3) The frame attached to the Kinect sensor $\mathcal{F}_k = \{P_k, \overrightarrow{i_k}, \overrightarrow{j_k}\}$. P_k is at the center of the Kinect sensor and both $\overrightarrow{i_k}$ and $\overrightarrow{j_k}$ are in the horizontal plane; $\overrightarrow{i_k}$ is pointing in the forward direction of the sensor.

P_m in $\mathcal{F}_0$, denoted $P_{m(\mathcal{F}_0)}$, and the angle between $\vec{i}$ and $\overrightarrow{i_m}$, denoted $\theta_{m(\mathcal{F}_0)}$, are known from odometry. P_k in $\mathcal{F}_m$, denoted $P_{k(\mathcal{F}_m)}$, is known from the hardware configuration of the robot: $P_{k(\mathcal{F}_m)} = (-0.08, 0)$. The angle between $\overrightarrow{i_m}$ and $\overrightarrow{i_k}$, denoted $\theta_{k(\mathcal{F}_m)}$, is given by the pan axis position measurement of the pan-tilt mechanism. The position of the person in $\mathcal{F}_k$, denoted $Person_{(\mathcal{F}_k)} = (X_{Person(\mathcal{F}_k)}, Y_{Person(\mathcal{F}_k)})$ is given by the output of the Kinect sensor. The angle between the forward direction of the Kinect sensor, $\overrightarrow{i_k}$, and the person, denoted $\theta_{Person(\mathcal{F}_k)}$, can be calculated:

$$\Theta_{Person(\mathcal{F}_k)} = \tan\left(\frac{Y_{Person(\mathcal{F}_k)}}{X_{Person(\mathcal{F}_k)}}\right) \qquad (2)$$

The position of the person in $\mathcal{F}_m$, denoted $Person_{(\mathcal{F}_m)} = (X_{Person(\mathcal{F}_m)}, Y_{Person(\mathcal{F}_m)})$ can be calculated:

$$Person_{(\mathcal{F}_m)} = Person_{(\mathcal{F}_k)} * \begin{pmatrix} \cos\left(\theta_{k(\mathcal{F}_m)}\right) & \sin\left(\theta_{k(\mathcal{F}_m)}\right) \\ -\sin\left(\theta_{k(\mathcal{F}_m)}\right) & \cos\left(\theta_{k(\mathcal{F}_m)}\right) \end{pmatrix} + P_{k(\mathcal{F}_m)} \quad (3)$$

The angle between the forward direction of the robot, $\overrightarrow{i_m}$, and the person, denoted $\theta_{Person(\mathcal{F}_m)}$, can be calculated:

$$\Theta_{Person(\mathcal{F}_m)} = \tan\left(\frac{Y_{Person(\mathcal{F}_m)}}{X_{Person(\mathcal{F}_m)}}\right) \quad (4)$$

Finally, the position of the person in $\mathcal{F}_0$, denoted $Person_{(\mathcal{F}_0)} = (X_{Person(\mathcal{F}_0)}, Y_{Person(\mathcal{F}_0)})$, can be calculated:

$$Person_{(\mathcal{F}_0)} = Person_{(\mathcal{F}_m)} * \begin{pmatrix} \cos\left(\theta_{m(\mathcal{F}_0)}\right) & \sin\left(\theta_{m(\mathcal{F}_0)}\right) \\ -\sin\left(\theta_{m(\mathcal{F}_0)}\right) & \cos\left(\theta_{m(\mathcal{F}_0)}\right) \end{pmatrix} + P_{m(\mathcal{F}_0)} \quad (5)$$

C. Pan-tilt mechanism control

In order to make the Kinect sensor always point in the direction of the person tracked, a control law of the pan axis of the pan-tilt mechanism was developed. The output of this control law is an angular speed command of the pan axis, denoted $\dot{\theta}_{k(\mathcal{F}_m)(Command)}$.

A first approach to compute the speed command was to implement a P-controller using the angular position of the person in the Kinect frame, $\theta_{Person(\mathcal{F}_k)}$, as the measurement and a 0° angle as the target, $\theta_{Person(\mathcal{F}_k)(Target)}$.

$$\dot{\theta}_{k(\mathcal{F}_m)(P-control)} = K_{p(Pan)} * error = K_{p(Pan)} * \left(\theta_{Person(\mathcal{F}_k)(Target)} - \theta_{Person(\mathcal{F}_k)}\right) \quad (6)$$

$\theta_{Person(\mathcal{F}_k)}$ is given by equation (2) and $\theta_{Person(\mathcal{F}_k)(Target)} = 0^\circ$.

Then the angular speed command is set equal to the output of the P-controller:

$$\dot{\theta}_{k(\mathcal{F}_m)(Command)} = \dot{\theta}_{k(\mathcal{F}_m)(P-control)} \quad (7)$$

We used $K_{p(Pan)} = 4s^{-1}$. This first approach using equation (6) for computing the speed command is able to maintain the sensor in the direction of the tracked person when the robot is not moving. However, when the robot is moving, the system is not reactive enough to keep track of the person. Loss of tracking happens when the robot is rotating or turning. To compensate for the robot rotation, a second approach was developed. Information from the odometry pose estimation is used to calculate the angular speed of the robot in $\mathcal{F}_0$, denoted $\dot{\theta}_{m(\mathcal{F}_0)}$, from two successive measurements of the robot orientation in the global frame: $\theta_{m(\mathcal{F}_0)(t-1)}$ and $\theta_{m(\mathcal{F}_0)(t)}$.

$$\dot{\theta}_{m(\mathcal{F}_0)} = \frac{\theta_{m(\mathcal{F}_0)(t)} - \theta_{m(\mathcal{F}_0)(t-1)}}{T(t) - T(t-1)} \qquad (8)$$

Where $T(t)$ and $T(t-1)$ are the time of the current measurement of angular speed and the time of the previous measurement of angular speed, respectively.

Using the additive inverse of the angular speed yields a robot rotation compensation speed command.

Finally, the speed command to send to the pan axis of the pan-tilt mechanism is calculated by summing the output of the P-controller and the robot rotation compensation speed command:

$$\dot{\theta}_{k(\mathcal{F}_m)(Command)} = \dot{\theta}_{k(\mathcal{F}_m)(P-control)} + \dot{\theta}_{k(\mathcal{F}_m)(Counter-rotation)} \qquad (9)$$

This approach using equation (9) is the one used in this work.

This algorithm requires an estimation of the person position. In case of a loss of tracking, the recovery procedure continues to apply the last pan axis speed command for 500 ms and then setting the pan axis to its neutral position, $\theta_{k(\mathcal{F}_m)} = 0^{\circ}$, while waiting for a new person position estimation.

D. Path-following algorithm

The principle of the path following algorithm is to make the robot take the same path as the person it follows. It uses the succession of person position estimations in $\mathcal{F}_0$, denoted $Person_{(\mathcal{F}_0)}$, and is calculated from equation (5) to generate a set of points to send to the robot path follower previously described in the robot control section. However, the $Person_{(\mathcal{F}_0)}$ points cannot be directly sent to the robot path follower. They are too noisy when the robot is moving, as described in the experimental results.

Hence the $Person_{(\mathcal{F}_0)}$ points are first filtered:

- Points which imply that the person accelerates faster than 1 g are ignored.
- Points which imply that the person moves faster than 1.5m/s are ignored.
- Jitter reduction of 15cm radius is applied: if a point is not farther than 5cm from the previous point, it is ignored.

Then the path connecting the succession of points is smoothed using a moving average technique of span 5. Finally, as the robot path follower needs a path with points separated by an interval of 2cm to properly work, points are interpolated by using uniform cubic B-splines. This also ensures further smoothing of the path. After filtering, smoothing and interpolation, the output point, denoted $Person_{(\mathcal{F}_0)(Filtered)}$ is sent to the robot path follower.

E. Adaptive algorithm

The idea of the adaptive algorithm is to continuously re-compute the best path for the robot to go to the person taking into account the obstacles in the environment. Hence, if a shorter way than the path the person took to go to its current position exists, the robot will be able to use it. The optimal path is computed using an implementation of the Karto library which uses the Monte Carlo Localization algorithm (Thrun et al. 2005).

The adaptive algorithm uses the filtered and smoothed person position estimation, $Person_{(\mathcal{F}_0)(Filtered)}$, described in the previous section. Each time a position estimation is received, it is compared to the last position estimation used to generate the robot path. If the distance that separates these two position estimations is superior to 50 cm, a new path using the last position estimation is computed and sent to the robot. This approach is needed in order to limit the frequency of the re-computation of the path which, when too high, saturates the computer and makes the robot oscillate and change its course too often.

6.4.Results and discussion

A. Path-following algorithm

For each of the 10 trials the robot was able to follow the person until the end of the path (Table 4). The average 0.9 loss-of-track events per trial did not affect the performance of the following thanks to the efficiency of the tracking recovery procedure. Figure 3 illustrates this success and shows both the robot's and the person path's close to each other, along with the obstacle setups from a typical trial.

The path taken by the person is reproduced accurately with an average path error of 11.04 cm, a standard deviation of 7.34 cm and a maximum error of 40.27 cm (Table 4). The agility and accuracy of this method are fully understood when comparing the results with the 40 cm width of our robot. Thanks to this accuracy it is possible to perform person following in an environment with obstacles without the need of detecting and actively avoiding the obstacles.

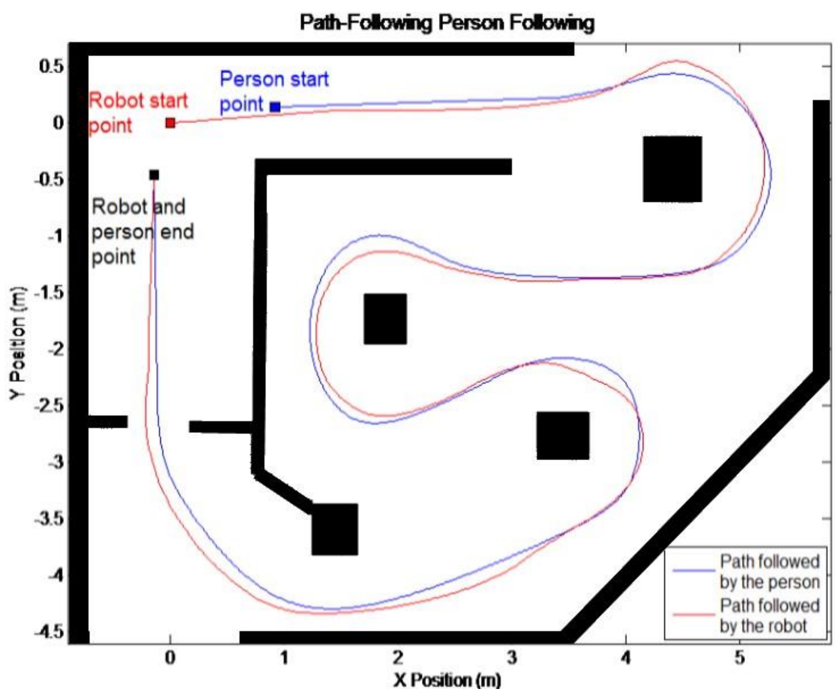

Figure 41: Person and robot paths of a sample trial of the evaluation of the path-following algorithm.

Path-following algorithm	Average	Max	Min	Standard Deviation
Path-completion success ratio [%]	100	100	100	0
Number of loss-of-track events per trial	0.9	2	0	0.88
Robot path length to person path length ratio [%]	100.5	103.7	97.6	1.6
Path Error [cm]	11.04	40.27	0	7.64

Table 4: Experimental results of the evaluation of the path-following algorithm

However, when comparing the distance covered by the human and the robot, it appears that they are nearly the same. This is due to the principle of this algorithm: the path taken by the person is accurately followed and hence is not optimal; in case of a possible shorter path, it will not be taken by the robot.

B. Adaptive algorithm

In terms of the path-completion ratio, the adaptive algorithm performed as well as the path-following algorithm with 100% completion for all the trials; and similarly it was not affected by the nearly same average 1.1 loss-of-track events per trial. Figure 42 illustrates this success, but it shows also how the adaptive algorithm enables the robot to take a shorter path when it can. Over the 10 trials the distance travelled by the robot was 70.9% of the distance travelled by the person, with a maximum of 82.1%, a minimum of 57.6% and a standard deviation of 6.5%.

Hence, the adaptive algorithm presents the advantage of minimizing the distance travelled by the robot compared to the path-following algorithm. However, this requires a prebuilt map of the environment.

Adaptive algorithm	Average	Max	Min	Standard Deviation
Path-completion success ratio [%]	100	100	100	0
Number of loss-of-track events per trial	1.1	3	0	1.1
Robot path length to person path length ratio [%]	70.9	82.1	57.6	6.5

Table 5: Experimental results of the evaluation of the adaptive algorithm

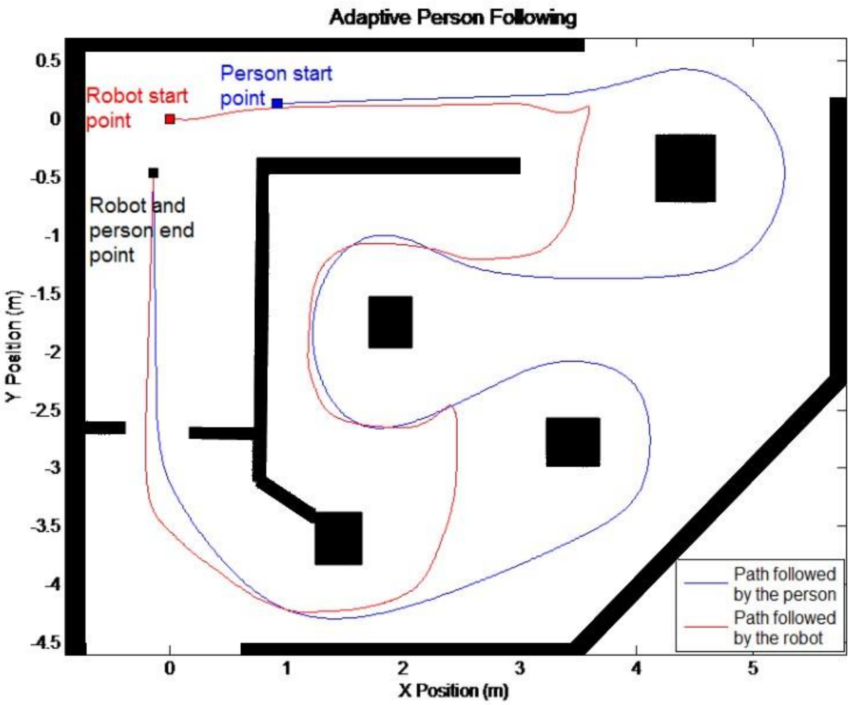

Figure 42: Person and robot paths of a sample trial of the evaluation of the path-following algorithm.

6.5. Conclusions and future work

Two person-following algorithms that use depth information from a Kinect sensor were presented. Both use the Kinect sensor mounted on a pan-tilt mechanism for 360-angle tracking and implement path generation from a sequence of estimated person's positions. The path following algorithm generates sequentially a path that reproduces the path taken by the person using each new updated position of the person. On the other hand, the adaptive algorithm recomputes from scratch the shortest path to the person each time the person has moved more than 50 cm. **Both person following algorithms were equally successful in following the person with a 100% path completion ratio**. However, **the adaptive algorithm minimized the distance travelled by the robot**: it travelled on average 29.1% less than the person it followed, whereas the path-following algorithm made the robot travel on average 0.5% more. Yet which algorithm is best to use is subject to discussion. The adaptive algorithm minimizes the distance travelled but presents the important constraint of needing a-priori information about the environment. This can be an advantage in situations where the cost of travel of the robot is expensive or in situations where the maximum speed of the robot is inferior to the walking speed of the person followed.

Future work should focus on path optimization without a-priori information. The case of the robot standing in the way of the person was not investigated in this work. Hence algorithms must be developed to adapt the path of the robot in order not to block the way of the person when she/he changes suddenly of direction. Furthermore, strategies to recover from complete occlusions from other persons or walls should be improved.

7. Comparison of novel interfaces for mobile indoor robot control: direct physical interaction, person following and pointing control

7.1. Introduction

In this experiment three novel natural interfaces for controlling navigation of a mobile robot in an indoor environment were created and developed. The first interface is based on direct physical interaction requiring from a human user to push the robot in order to displace it. Two other interfaces exploit a 3D vision-based human skeleton tracking and allow the user to navigate the robot by walking in front of it or by pointing towards a desired location. Performance and workload evaluation was conducted for two different navigation tasks. In the first task, the subjects were asked to navigate the robot between different rooms in the testing apartment. The second task evaluated navigation in the same environment through a set of waypoints, which were exact locations marked on the apartment floor.

7.2. Methodology

A. Apparatus

The interfaces were implemented on a customized Robosoft's Kompai robot shown in Figure 43. The robot's base is robuLAB10, a generic differential drive mobile platform with two propulsive wheels and two castor wheels that comes with basic navigation functions. A rigid structure was added on top of the platform, including three tubes and a tray for a laptop PC. On the top of this structure a TRAC Labs Biclops pan-tilt mechanism and a Kinect sensor were added. For navigation purposes, the base is equipped with a SICK S300 lase-range finder (LRF), which is positioned at the height of 0.24m and provides distance measurements of up to 30m with an angular field of view of 270°.

The pan-tilt mechanism has a tilt range of 120° and a pan range of 360° with a maximum angular velocity of 170°/s and a maximum angular acceleration of 3000°/s^2. The precision of the angular position measurements is 0.01°. The mechanism can support a maximum payload of 4kg which is more than the weight of the Kinect sensor. In this work, the tilt value was set to 0° and person tracking was performed in the horizontal plane, using only the pan axis. The communication between the laptop PC and the pan-tilt mechanism is maintained via a USB port with a data transfer rate of up to 416kbps.

The Kinect sensor provides depth measurement from 0.8m to 4m with a vertical viewing angle of 43° and the horizontal viewing angle of 57°. It provides depth images at the resolution of 640x480 pixels and at the maximal frame rate of 30 fps. The Microsoft Kinect SDK provides person detection and skeleton joints tracking features.

The laptop PC used in this work is powered by an Intel quad-core i7 Q740 CPU with 4 GB of RAM.

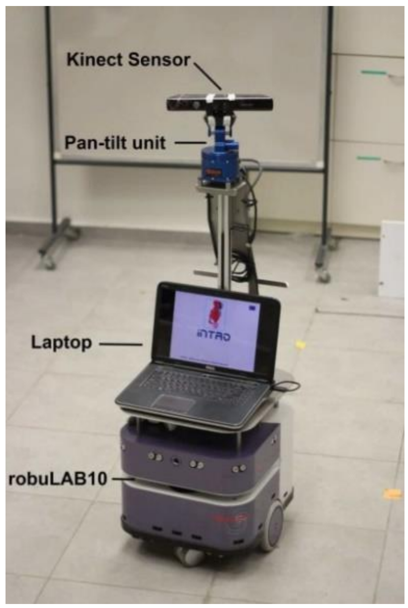

Figure 43: Robot Platform

B. Experimental environment

The experiments were performed in the apartment at Robosoft premises, which are fully furnished to have the functionality of a common home environment, as shown in Figure 44. The apartment consists of a lobby with a bathroom, one large room and a kitchen which is separated from the room by a bar table. The floor is uniformly covered with a carpet allowing the robot to smoothly displace itself around the apartment.

Figure 44: Test environment: the apartment at Robosoft premises

C. Robot navigation task

Three interfaces (detailed in 2.4) were compared and tested for two different navigation tasks inside the apartment. In both tasks the robot starting location was the same. In the first task, the subjects were instructed to navigate the robot through three different areas that were clearly marked on the apartment floor: the square in the center of the living room, the lobby, and the kitchen. In the second task, the goal was to navigate the robot through a set of three waypoints, marked as exact locations on the apartment floor. A robot-generated map of the apartment with the resolution of 1.67cm/pixel is shown in Figure 45. The size of the apartment was 6.55m x 5.2m. The Karto library that implements the Monte Carlo Localization algorithm (Thrun et al. 2005) was used to generate the map from the LRF readings, but also to provide robot localization and path planning with both static and dynamic obstacle avoidance. On the map, the placement of the furniture is displayed in grey, the starting robot location is displayed as the blue cross with the label "0", and the target areas and waypoints are shown as red lines and crosses, respectively, with labels "A", "B" and "C" representing the square in the center of the living room, the lobby and the kitchen, respectively. The waypoints' coordinates are given in centimeters assuming 0-waypoint as the origin of the coordinate system: A (31, 0), B (385, -110), C (420, 90). The Euclidean distances between the subsequent waypoints are: 0A = 155cm, AB = 252.83cm, BC = 203.5 cm, and CA = 279.85 cm.

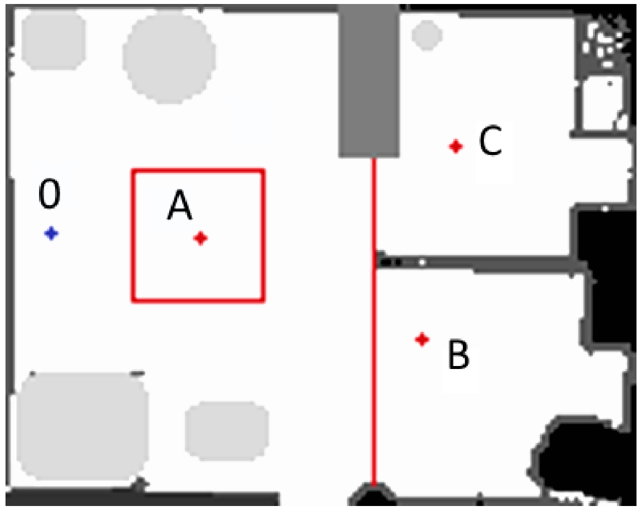

Figure 45: Karto-generated map of the apartment with marked target areas and waypoints

a) Area navigation

The area navigation task was used to evaluate the difficulty in using each of the three interfaces for robot navigation in the apartment. In each experiment, the robot was placed at the starting point “0” and navigated by the subject through a set of areas in the following order: 0-A-B-C-A. The robot was considered to be inside an area once its central axis crossed the area’s border line (red lines on the map in Figure 45). The subjects were advised to navigate the robot a bit.

b) Waypoint navigation

The waypoint navigation task was used to evaluate the accuracy in using each of three interfaces for robot navigation inside the apartment. As in the Task 1, in each experiment, the robot was placed at the starting point “0” where it was further navigated by the subjects, but this time through a set of waypoints in the following order: 0-A-B-C-A. The subjects could stop the robot at any distance from a waypoint, and this distance was later used to evaluate the navigation accuracy.

D. Robot control interfaces

a) Direct Physical Interaction

The DPI interface allows the user to navigate the robot by pushing and pulling it around the apartment. When the users push the robot, they experience a light resistance due to the implemented friction compensation control mode. Hence, the robot manipulation requires very little effort. The total friction torque generated by the friction forces on each motor was measured experimentally for wheel speeds from 0 to 13.3 rad/s corresponding to ground speeds from 0 to 1m/s. For angular speed from 0 to $\dot{\theta}_l$= 2.4 rad/s, (corresponding to a ground speed of 0.18 m/s), the friction is approximately proportional to the wheel angular speed. From $\dot{\theta}_l$ it does not increase with speed anymore and stays approximately constant.

The friction torque can be approximated by the following formulas:

$$\tau_f = \frac{C_f}{\dot{\theta}_l} * -\dot{\theta} \; when \; \dot{\theta} < \dot{\theta}_l$$

$$\tau_f = C_f * -\frac{\dot{\theta}}{\|\dot{\theta}\|} \; when \; \dot{\theta} > \dot{\theta}_l$$

Where τ_f is the friction torque in N.m, C_f is the friction coefficient determined experimentally of 0.85 Nm, and $\dot{\theta}$ is the wheel angular speed in rad/s.

The following control law is applied:

$$\tau_R = -\,\tau_{f_R} * 0.8$$

$$\tau_L = -\tau_{f_L} * 0.8$$

where τ_R, τ_L, τ_{f_R}, and $\tau_{f_L}$are right and left wheels command torques and friction torques. The control law loop runs at 200 Hz. The 0.8 factor is present to ensure stability and to keep the virtual friction non-zero.

b) Person Following

In the person following robot control mode, the user walks in front of the robot and the robot follows it; the user leads it to thedesired location in the apartment. The robot smoothly follows the user at a safe distance so any physical contact between the robot and the user is prevented. The user could stop the robot at any time by raising their left hand above the level of the left elbow, and restart the robot motion by putting the left hand back to the position below the level of the left elbow.

Person following is achieved through uninterrupted user tracking. The integration of the pan-tilt mechanism on top of which the Kinect sensor is mounted enables decoupled motion of the sensor and the robot and extends the Kinect's horizontal detection range. The position of the user obtained from the Kinect is fed to the visual control module of the pan-tilt mechanism, which then ensures that the sensor always faces the user. The position of the pan-tilt mechanism and the position of the robot on the map are used to compute the position of the user on the map. This position is fed back to the robot navigation module that executed the person-following behavior. Detailed description of the tracking, control and person-following algorithms are given in section 6, Experiment B1. The person following algorithm used was the person following one..

c) Pointing control

In the pointing control mode, the user navigates the robot by pointing with the right hand at a desired location on the floor. The robot is stopped by raising the left hand above the level of the left elbow. By lowering the left hand back to the position below the left elbow the robot is restarted and continues to move.

The desired destination is computed from the intersection of the ground floor plane with a line passing through the right hand and the right elbow joints, whose locations are obtained from the Kinect sensor. The joints locations are transformed beforehand from the Kinect frame of reference to the map frame of reference, using the same method as in the previous section. Similarly, the user-tracking algorithm is identical as for the person following.

E. Performance and workload measures

The following metrics were used to assess the performance and the workload of the users.

1) Completion time. The intermediate and total time needed for the subject to drive the robot through the set of waypoints or areas.
2) Accuracy. Accuracy was measured as distance between the waypoint and where the subject stopped the robot, calculated from the robot localization data. This metric was used only for the waypoint navigation task.
3) Raw NASA-TLX questionnaire. Participants completed a computerized version of the questionnaire after each trial. The raw NASA-TLX enables the collection of six

dimensions of workload ranging from 0 to 100 (Hart & Staveland 1988), and was used to assess the subject workload when controlling the robot, similarly as in (Nielsen et al. 2007).

F. Experimental Design and Subjects

A mixed between- and within-subject design was used in the experiments. 24 subjects, 9 females and 15 males, aged from 22 to 58 years (average 37.2, SD=11.3) were divided in two groups. The navigation task type was the between-subject variable: waypoint navigation task (group A) and area navigation (group B). The control mode was the within-subject variable: each subject completed the navigation task using each of the three control modes once. The possible order effect was counter-balanced by permuting the order of the control modes used between the subjects.

G. Procedure

The subjects performed the experiments alone. The operator was present in the apartment, but did not interfere during the task execution. Before each experiment, the subjects were given a short presentation about the robot features and abilities. They were explained that the robot can be controlled using three modes, namely pushing, following and pointing, and that they will perform three trials, one with each control mode. They were also assured that the robot speed is limited and that no harm will happen either to them or to the robot.

The order of the control modes was permutated between the subjects to avoid any learning bias. Before each trial, the subject was informed about the procedure that consisted of: 1) a demo of the interface by the operator, 2) a trial by the subject, 3) the experiment, and 4) the filling of a questionnaire about the performed experiment. The goal of the experiment in terms of the speed and accuracy, depending on the task at hand, was described to the subject and presented as a competition with other subjects in order to motivate them to perform at their best abilities.

At the beginning of the experiment, the robot was placed at its starting position. For the following and the pointing control modes the robot was started by the operator from the GUI on the robot-mounted laptop PC. Only in case of the pushing control mode the subjects had access to the GUI on the laptop PC and they started the robot themselves. During the task execution, the subjects were instructed to stop the robot at the target areas (task 1) or waypoints (task 2). For the pointing and the following control modes, the subjects could stop the robot by raising their left hand above the level of their left elbow; this action was detected by the Kinect sensor and it would store the robot location in the apartment. By lowering the hand back to the position below the level of the elbow would restart the robot and the experiment could continue. For the pushing control mode, the subjects would physically stop the robot and then click on the GUI to store its location. An occasional loss of tracking would activate the recovery procedure that navigates the robot to the starting point and position. The subjects could interrupt this procedure at any time by standing in front of the robot within the detection range of the Kinect sensor; this would restore tracking and allow the subjects to continue with the trial.

7.3.Results and discussion

Out of the 72 trials conducted by the 24 subjects, 6 were uncompleted by 5 different subjects. These failures occurred in 3 pointing control trials, 1 person following trial and 2 direct physical interactions trials.

Since task completion times have a skewed distribution they were log-transformed to achieve normality. Then, a Linear Mixed Model analysis (McCulloch & Searle 2000) was conducted on all the metrics with the control mode (pointing control, person following and direct physical interaction) as the within-group fixed effect and the task type (waypoint and area navigation task) as the between-group fixed effect, except for accuracy which was analysed using a Linear Mixed Model too but with only one factor: the control mode. Participants were included as a random effect to account for individual differences among participants and the correlations among repeated measures within participants. LMM analysis was employed rather than ordinary ANOVA with repeated measures due to the fact that there were missing values and in order to utilize the information of those observations without the need for supplementary data.

When necessary post-hoc pairwise comparisons were conducted using the Least Significant Difference method.

A. Completion time

The analysis conducted on the log of the completion timesreveals that **there is no significant effect of the task type on the completion time,** F(1,61)=0.451, p=0.504. However, the effect of the control mode on completion time was significant, F(2,61)=84.874, p<0.001. Subjects completed the tasks faster when using the DPI control mode (37 seconds, SD=11.8), slower when using the pointing control mode (160 seconds, SD=68), and had intermediate completion times when using the person-following control mode (103 seconds, SD=52). Post-hoc pairwise comparisons confirm that the difference between each control mode is significant, with p<0.001 for the three pairs. Results shown in the Figure 46 below are displayed in the form of the raw completion times, not the log of the completion times.

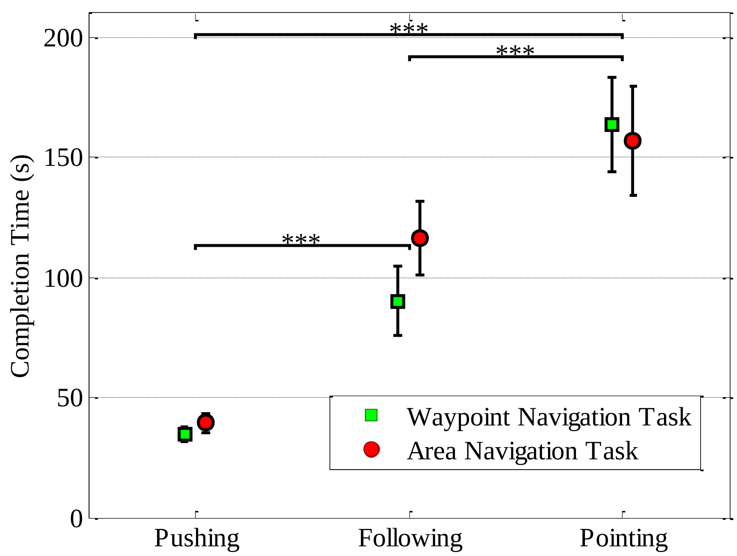

Figure 46: Effect of the control mode and task type on the task completion time. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1%and 0.1%. Error bars represent the standard error of the mean.

B. Accuracy

Analysis shows (Figure 47) that **when using the DPI control mode subjects were significantly more accurate than when using the person-following control mode** (0.12m, SD=0.027, compared to 0.25m, SD=0.31, p<0.05). No significant difference was found with the pointing control mode however.

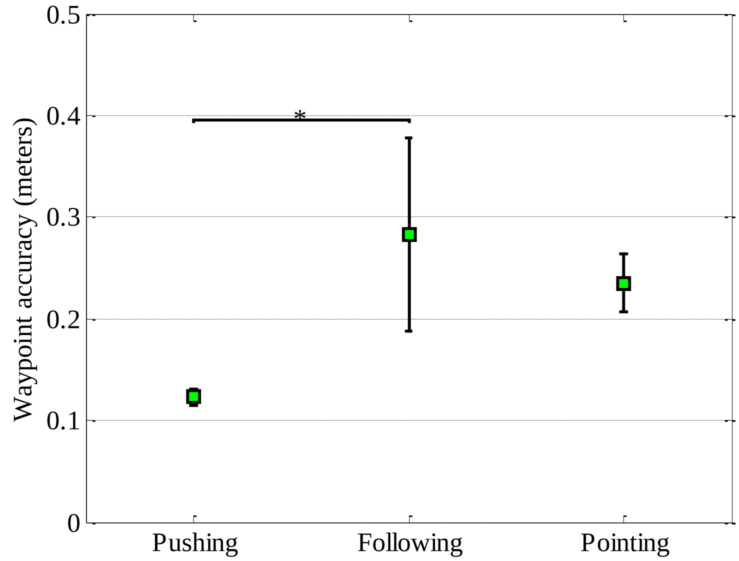

Figure 47: Effect of the control mode and task type on the waypoint accuracy. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1%and 0.1%. Error bars represent the standard error of the mean.

C. Raw NASA-TLX questionnaire

a) Overall Workload

The overall workload is the average of the six dimensions of workload measured with the raw NASA-TLX questionnaire.

Analysis reveals (Figure 48) that **the control mode had significant influence on the overall workload**, $F(2,68)=11.948$, $p<0.001$. However, there is no significant effect of the task type on the overall workload, $F(1,68)=2.208$, $p=0.142$.

These overall workload results are coherent with the completion time results. The DPI control mode appears to be the easiest to use (workload of 30.4, SD=10.9, for the waypoint navigation task and 28.8, SD=7.3, for the area navigation task) the pointing control mode the (workload of 46.5, SD=16.5, for the waypoint navigation task and 39.2, SD=13.9, for the area navigation task), and the person-following control mode induced an intermediate workload compared to the two other control mode (workload of 39.2, SD=10.6, for the waypoint navigation task and 33.9, SD=16.2, for the area navigation task). Post-hoc pairwise confirms that the difference between each control mode is significant.

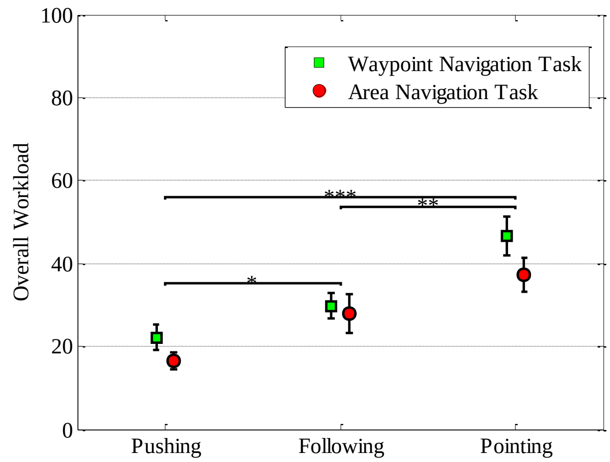

Figure 48: Effect of the control mode and task type on the overall workload. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1%and 0.1%. Error bars represent the standard error of the mean

b) Detailed workload dimensions

The control mode had significant effect on the four workload dimensions, as shown in Figure 49: Effect of the control mode and task type on the 6 dimensions of workload. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1%and 0.1%. Error bars represent the standard error of the mean.: Mental Demand, $F(2,68)=18.642$, $p<0.001$, Performance, $F(2,68)=8.324$, $p<0.001$, Effort, $F(2,68)=7.274$, $p<0.001$ and Frustration, $F(2,68)=13.117$, $p<0.001$. The effect of the control mode was the same on these four dimensions: the DPI control mode appears to be the less demanding, whereas the pointing control mode is the most demanding and the person-following control mode workload dimensions scores in between. Pairwise comparisons differences were checked and they are significant on all pairs apart between DPI and person following for Effort and Frustration.

Two workload dimensions (Physical Demand and Temporal Demand), were not impacted by the control mode (respectively $F(2,68)=0.041$, $p=0.959$ and $F(2,68)=1.149$, $p=0.323$).

Out of the 6 workload dimensions, none were significantly impacted by the task type: Mental Demand, $F(1,68)=0.648$, $p=0.424$, Physical Demand, $F(1,68)=0.760$, $p=0.386$, Temporal Demand, $F(1,68)=0.014$, $p=0.905$, Performance, $F(1,68)=1.424$, $p=0.237$, Effort, $F(1,68)=2.353$, $p=0.130$, and Frustration, $F(1,68)=3.464$, $p=0.067$.

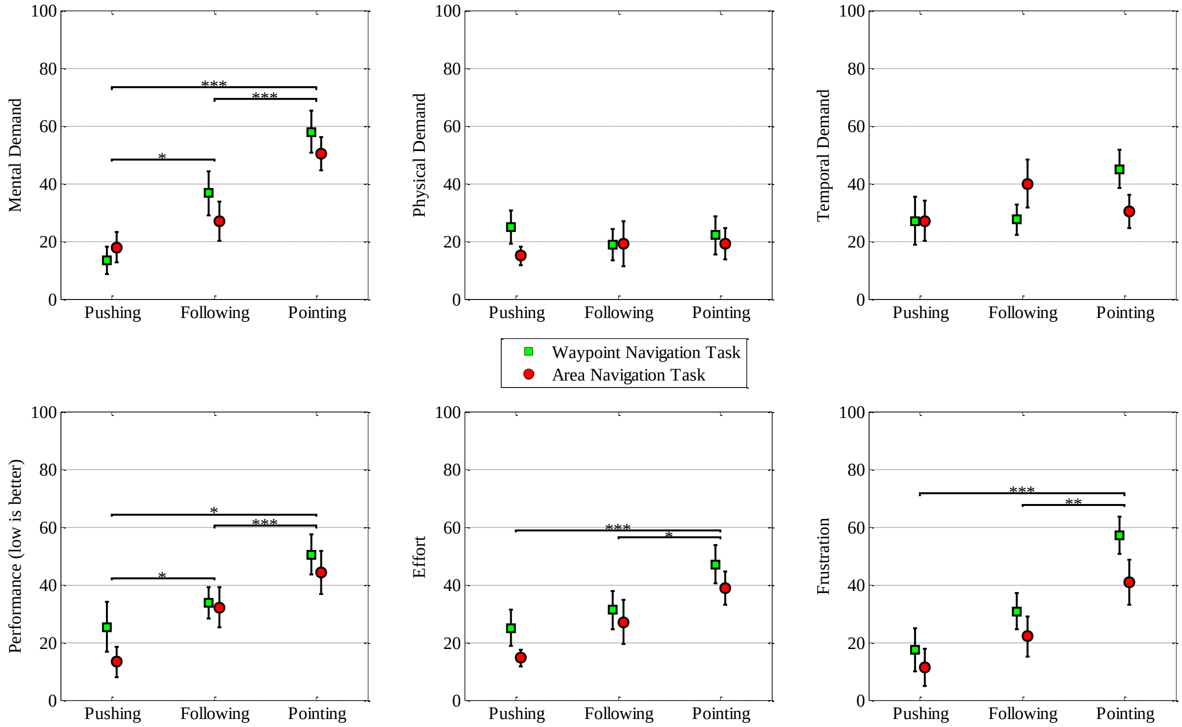

Figure 49: Effect of the control mode and task type on the 6 dimensions of workload. Significant effects are marked with stars: *, ** and *** respectively represent the significance level of 5%, 1%and 0.1%. Error bars represent the standard error of the mean.

7.4.Conclusions

Three novel interfaces for control of an indoor mobile robot in two different navigation tasks were developed and compared.**The effect of the control mode was statistically significant for almost all the variables measured, and remarkably consistent**. **The direct physical interaction is systematically better than the two indirect control modes. Subjects completed the tasks faster, with more accura**cy, less mental demand, less effort, less frustration and had the feeling they performed better. This result highlights the advantage of robot physical control in terms of performance and workload compared to contactless interfaces. Since a direct physical interface creates the illusion that the robot is a passive object and enables the user to directly manipulate it, this way of controlling a robot is necessarily more intuitive. Though, despite the absence of difference in physical demand measured here, the physical involvement needed for direct physical interaction cannot be suitable for all applications and there are scenarios where human-robot contact is unwanted. Therefore, the advance of contactless human-robot interfaces should be further developed.

In this experiment , **between the two contactless interfaces tested, the person-following control mode appears to be systematically better: subjects completed the task faster, with less mental demand, less effort, less frustration and had the feeling they performed better**. Pointing control could be better in terms of accuracy, but the difference was not statistically different. The advantage of the person-following interface over pointing can be explained by the fact that once started, the subjects don't need to actively control the robot, it is a "start and forget" technique, they just have to walk, knowing that the robot is following their steps. Whereas, with the pointing control, the subjects had to constantly take care of the robot control requiring more effort.

Direct physical interaction was the best interface modality in the experiment performed. If contactless control is needed, for instance when the hands of the operator are busy, the person-following interface is the best. Yet one could argue that this result could vary depending on the robustness of the person-following algorithm and the environmental situation. For instance, in a complex and dynamic environment, it is more likely that the robot will lose track of the followed operator and therefore active pointing control would be preferred. Still, person-following algorithm robustness is a technological issue, and when properly working like in this experiment, it presents definitive advantages over other interfaces. We believe that in terms of workload and in the context of service robots, it is better than conventional robot interfaces. Future work will focus on the comparison of these novel interfaces with classical robot control modalities in various scenarios.

For all control interfaces, surprisingly, no significant effect of the task type was found on any of the metrics measured, both objective and subjective metrics. One can argue that the two navigation tasks tested, waypoint navigation and area navigation, were similar, but the waypoint navigation required more precision in the control of the robot. When completing the navigation task with the added constraint of passing through a waypoint, it was expected that the subjects' workload would increase. But this was not the case, and for none of the six measured dimensions significant difference was noted. This result shows that controlling the robot with accuracy is not more costly for the three interface modalities tested here. It would be relevant to test in the future if the same result would be obtained when a person is using a classical gamepad interface.

8. Non-invasive robot camera head control for teleoperation: performance and workload assessment

8.1.Introduction

This experiment aims to evaluate a new method for controlling the orientation of the camera through the operator's head orientation in robot teleoperation tasks. Specifically, a new interface was created that use head-tracking in a non-invasive way, without immersive virtual reality devices was combined with joystick or hand gesture robot control, and compared in terms of performance and workload to a classical robot teleoperation interface. Additionally, the effect of the user experience and the way performance and workload evolved through consecutive trials was investigated.

8.2.Methodology

36 industrial engineering students, 21 males and 15 females between 22 and 28 years old, with no previous teleoperation experience were recruited through email. Participants received a compensation of 30 NIS (about 8 USD) for their participation and were told prior to the experiment that they could potentially win a bonus of 100 NIS (about 26 USD) through roulette wheel selection depending on their performance. The higher the score of a subject was, the more virtual lottery tickets she/he received. Among all the virtual lottery tickets distributed, one was chosen to attribute the 100 NIS bonus.

Participants were split into three groups of twelve students each for the three experimental conditions A, B and C which differed in the way the robot camera orientation and the robot movements were controlled, as described in section C.

The exploration task was conducted four times to study the impact of experience on performance.

A. Apparatus

The robot used in the experiment was a differentially driven iRobot Create with a custom structure holding a pan-tilt camera orientation mechanism (see Figure 50). The two servos constituting the pan-tilt mechanism are Dynamixel AX-12A. The camera is a PlayStation Eye which offers low latency and a field of view of 75°. An Asus Eee PC running a Microsoft Robotic Developer Studio node is embedded on the robot and controls its movements and the pan-tilt mechanism position, and additionally receives the video from the camera through USB connection. The camera is used with the CL-eye driver and stream the video feed to the control station at a rate of 30 images per second and at a resolution of 320×240 pixels.

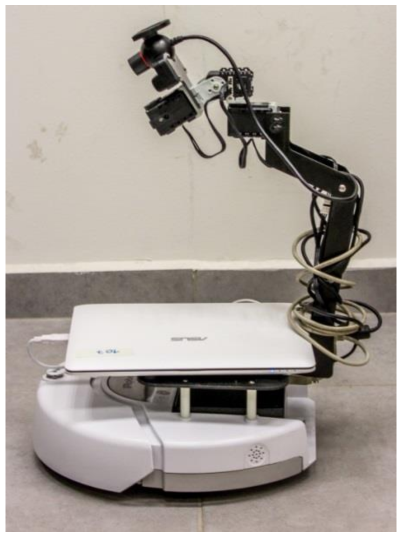

Figure 50: iRobot Create used for the experiments

The control station in front of which the participants sat comprises a 24 inches monitor displaying in full screen the video stream from the robot camera and a powerful PC computer running Microsoft Robotic Developer Studio to remotely communicate and control the robot through Wi-Fi. The computer was connected to a wireless Xbox 360 controller (see Figure 53) and a Kinect sensor used for the participants to control the robot and the orientation of its camera.

B. Teleoperation exploration task

The participants were asked to control and guide the robot through a maze from its entrance to its exit (see Figure 51). They were asked to complete the task as quickly as possible while minimizing the number of times they touched obstacles. They knew in advance that their score and their chance to win the bonus prize would be determined accordingly. By moving the robot camera orientation up from its default position, the participants had the possibility to see above the walls of the maze. The exit was marked by a pole higher than the walls that was viewable from any point of the maze when pointing the robot camera toward it (see Figure 54).

Controls included: 1) control of the robot movements, in the form of its linear and angular speeds, and 2) control of the orientation of the robot's camera.

The different experimental conditions were formed by the different combinations of the control modes for these two parts (Table 6).

Experimental Condition	**Robot movements control**	**Camera orientation control**
A	Xbox 360 controller	Xbox 360 controller
B	Xbox 360 controller	Head orientation
C	Hands gesture	Head orientation

Table 6: Experimental conditions control modes

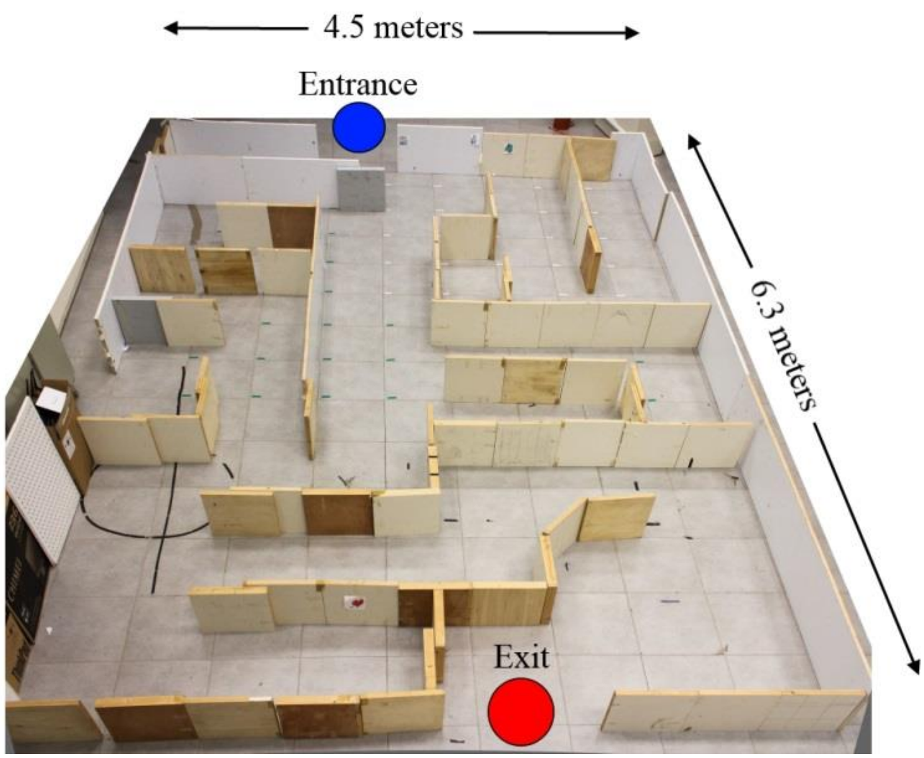

Figure 51: Maze used in the experiment

C. Robot control

The movements of the robot were controlled either by using an Xbox 360 controller or by using a hand gesture interface. The robot linear speed could be set from -1 to 1 m/s and its angular speed from -1.5 to 1.5 rad/s.

a) Conditions A and B: Xbox 360 controller

In conditions A and B the robots movements were controlled using an Xbox 360 controller (see Figure 53). The left analogic stick controls the angular speed: pressing left rotates the robot anti-clockwise and right clockwise. The left and right analogic triggers control respectively the negative and positive linear speed. All inputs are proportional meaning that a lighter push on the stick or a trigger results in a lower angular or linear speed.

b) Condition C: Hand gesture interface

In condition C the movements of the robot are controlled through the movements of the operator hands which are tracked using the Kinect. The left hand moves a bar on the left side of the screen which activates or deactivates the motors: when the left hands make the bar enter the bottom left area of the screen, the rectangle representing this area turns from red to green and it displays "Control On" instead of "Control Off". When the left hand makes the bar go out from the area its colors go back to red and displays "Control Off", and the robot motors are deactivated. When the operator's left hand is left of her/his shoulder the bar is inside the control area, when it is right of her/his shoulder, it is outside the control area.

The right hand is used to control the linear and angular speed of the robot. When the operator moves its right hand in a vertical plane, it moves a red dot on a cross on the screen. The position of this dot is used to set the desired angular and linear speeds of the robot which are applied only when the motors are activated by the left hand. The vertical position of the dot relative to the center of the cross sets the linear speed, and its horizontal position the angular speed.

Figure 52: Hand gesture interface

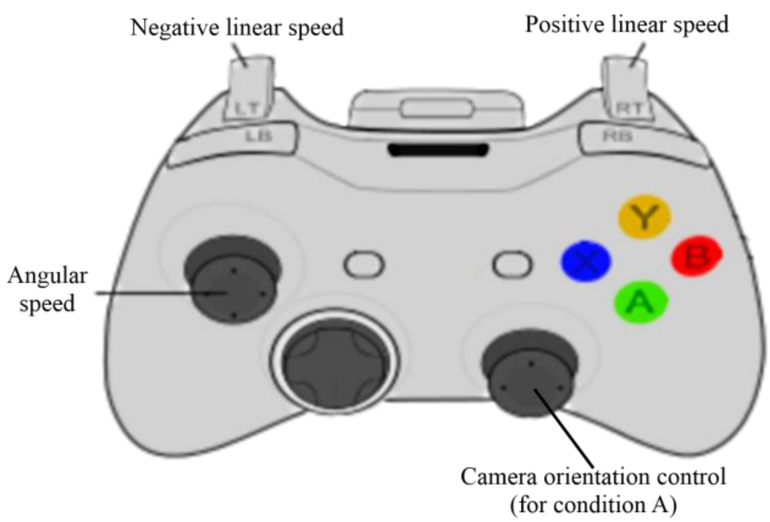

Figure 53: Xbox controller and inputs assignment

D. Camera orientation control

The orientation of the camera was controlled either by using an Xbox 360 controller or by using the orientation of the operator's head. In both modes the default orientation of the camera was a 0° pan angle relative to the forward direction of the robot and a -30° tilt angle relative to the floor (see Figure 54). The minimum/maximum pan angle allowed was -90°/+90°, and the minimum/maximum tilt angle allowed was -84°/0°.

Figure 54: Robot camera output for the extreme left, right, up, down and neutral position of the pan-tilt mechanism.

a) Condition A: Stick control

In the condition A the orientation of the camera was controlled by the position of the right stick of the Xbox 360 controller (see Figure 53): its orientation is mapped to the position of the stick with a 20 points resolution on each axis (e.g., when the stick is in the neutral position, the camera is in its neutral orientation; when the stick is pushed 50% to the left and 20% up, the camera is pointing 45° left and 41° down).

b) Conditions B and C: Head control

In the conditions B and C the orientation of the camera is controlled by the orientation of the participant's head measured with the Kinect sensor. In order for the participants to be able to still see the control station monitor while moving the camera to its extreme orientations, the participants head orientation was mapped to the camera orientation with a 2.5 ratio. In other words, a head orientation of 10° to the left provoked an orientation of the camera of -25° on the pan axis. As a result the participants never needed to move their head away from its neutral position more than 36° on the pan axis and 22° on the tilt axis. In order to avoid unwanted camera movements, a jitter reduction of 2° was applied to the tracking of the head orientation and head orientations smaller than 5° around the neutral orientation on both axes were not considered.

E. Head orientation tracking

The participant head orientation tracking was performed remotely in a non-invasive way using a Kinect sensor positioned 50 cm behind and 30 cm above the monitor of the robot control station. An algorithm fuses the RGB and depth data from the sensor to detect the participants' faces and track their orientation on the three axes. This algorithm is part of the Face Tracking SDK working with the Kinect for Windows SDK (Anon n.d.).

F. Performance and workload measures

The following measures were used to assess performance and workload:

1) Completion time. The time needed to drive the robot from the entrance to the exit of the maze, in seconds.
2) Number of collision. The number of times the robot touched a wall during a trial.
3) Use of the pan-tilt mechanism. The total displacement on both pan and tilt axes of the robot camera, in degrees.
4) Heart rate. Heart rate of the participants was measured using a Polar CS600X chest sensor in a resting state (baseline) and during each trial. The variation in percentage between the baseline and each trial was then calculated and used as a measure.
5) Raw NASA-TLX questionnaire. Participants completed a computerized version of the questionnaire after each trial. The raw NASA-TLX enables the collection of six dimensions of workload ranging from 0 to 100.

G. Procedure

The participants were first asked to wear the heart rate sensor. Once equipped with the heart rate sensor, they had to read and sign a consent form informing them about the conditions of the experiment. They were then asked to relax and were presented with a 5 minutes long video unrelated to the experiment. The last 3 minutes of the video were used to determine their heart rate baseline. Then, they received an explanation about the control of the robot and its camera and had the opportunity to see and to train to teleoperate the robot for about 3 minutes in an open space. They never had the possibility to see the maze other than through the video feedback of the camera during the trials. After each trial, they had to complete the raw NASA-TLX questionnaire and had 3 minutes to rest before starting the next experiment. After the four trials they received their compensation and signed a receipt.

H. Data analysis

A two-factor mixed design ANOVA with the subject variable as the Trial Number (1 to 4) and the between subject variable set as the experimental condition (Control mode A, B or C) was conducted.

8.3. Results

A. Task completion time

The task completion time was significantly impacted by the trial number, $F(3,32)=84.818$ $p<0.001$, reflecting that the **participants needed less time to complete the task as they gained experience with the system** (see Figure 55) with an average reduction of 55% between the first and the last trial.

The experimental condition (control mode) had a significant effect on the completion time, $F(2,33)=4.480$, $p<0.001$ maximum difference of 39%. However, since there was also a significant interaction between the experimental condition and the trial number, $F(6,30)=20.471$, $p<0.001$ additional analysis was necessary. Post-hoc pairwise comparisons were conducted for each trial between the three conditions, see Table 7for the P-values of these tests.

Trial Number	Conditions		P-Value
1	A	B	**,046**
		C	**,010**
	B	C	,000
2	A	B	,642
		C	**,003**
	B	C	**,011**
3	A	B	**,001**
		C	**,000**
	B	C	**,001**
4	A	B	**,000**
		C	**,000**
	B	C	,540

Table 7: P-values of least significant difference pairwise comparisons of the completion time between the experimental conditions

These results reveal that during the first trial, the participants controlling the robot with the Xbox controller and the camera with their head (condition B) performed the best with the smallest task completion time (233 seconds).The participants controlling both the camera and the robot with the Xbox controller (condition B), performed second best (290 seconds), and the group using hand gestures and head orientation (condition C) performed significantly worse (346 seconds). However, which group performed the best evolved with the trial number and the participants' experience. While group C had significantly the higher task completion time for all trials, apart for the last one where it was not different from group B; group A and group B had their trend reversed: during the second trial their performance was not significantly different, and during the third and last trial, group A was better than group B, opposite to the results of the first trial.

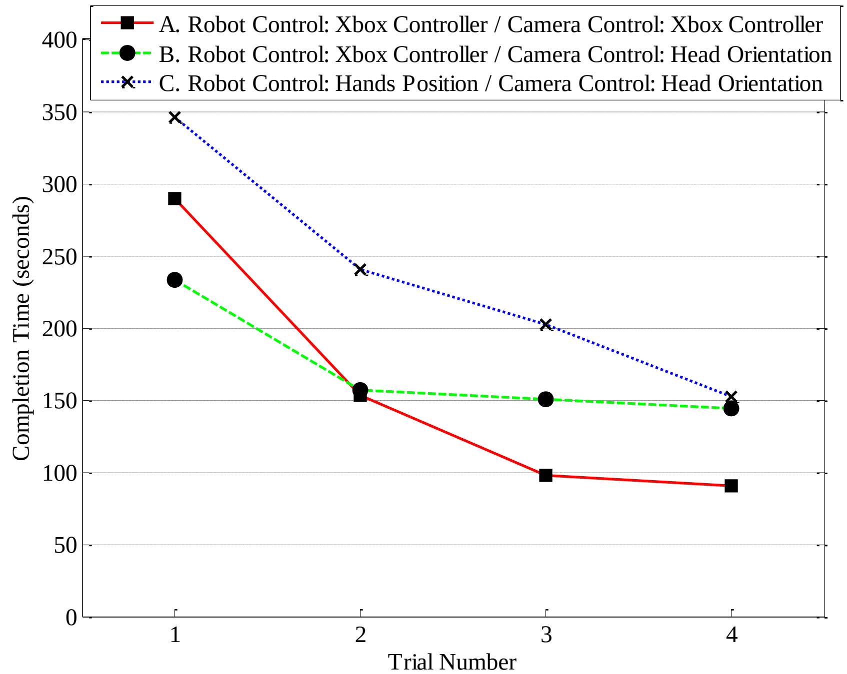

Figure 55: Effect of trial number and experimental condition on task completion time.

B. Number of collisions

The number of collisions was significantly affected by the trial number, $F(3,32)=26.332$, $p<0.001$, it was reduced with participants getting more experience (see Figure 56) reducing the number of collisions in average by 60%.

The experimental condition (control mode) had also a significant effect on the number of collisions, $F(2,33)=9.262$, $p<0.01$, however there was in addition a significant interaction between the experimental condition and the trial number, $F(6,30)=4.288$, $p<0.001$, requiring additional analysis. Post-hoc pairwise comparisons were conducted for each trial between the three conditions, see Table 8 for the P-values of these tests.

Results reveal that during the first trial, the performance regarding the number of collisions corresponded to the completion time performance, with best results received for group B followed by group A and group C resulted with worst. However, from the second trial, there was no significant differences anymore between groups A and B, and from the third trial and beyond there was no difference between the three groups. In fact, because of the low number of collisions, it is difficult to find significant differences.

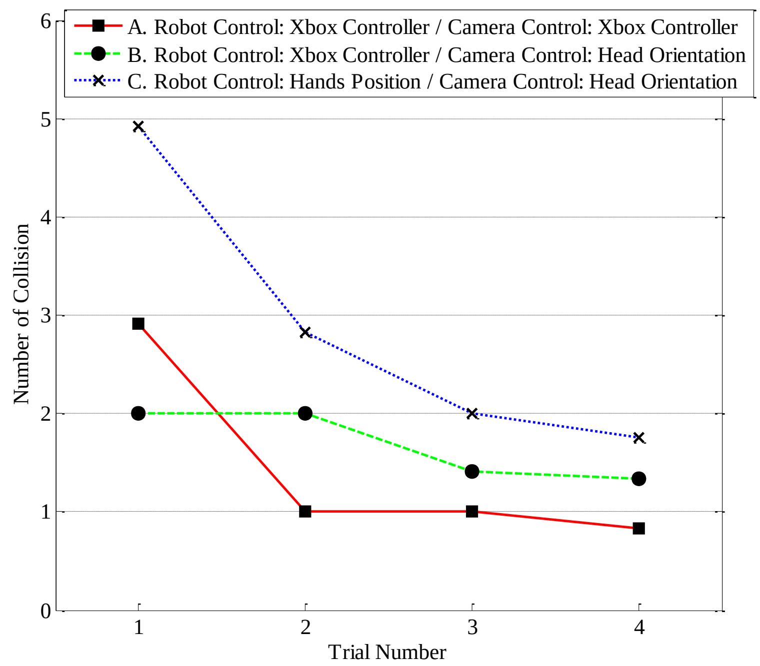

Figure 56: Effect of the trial number and control mode on the number of collision.

Trial Number	Conditions		P-Value
1	A	B	**,034**
		C	**,000**
	B	C	**,000**
2	A	B	,078
		C	**,002**
	B	C	,139
3	A	B	,398
		C	,053
	B	C	,239
4	A	B	,402
		C	,129
	B	C	,484

Table 8:P-values of least significant difference pairwise comparisons of the number of collision between the experimental conditions

C. Use of the pan-tilt mechanism

Both the trial number and the control mode affected the use of the pan-tilt mechanism, $F(3,32)=5.053$, $p<0.01$, and $F(2,33)=7.295$, $p<0.01$, respectively. But no interaction was found between these two factors. As illustrated in Figure 57, the use of the pan-tilt mechanism decreased with the experience of the user. The participants using only the Xbox controller moved less the mechanism than the two other groups, and the participants using hands gesture and head orientation moved it the most, independently from the trial number.

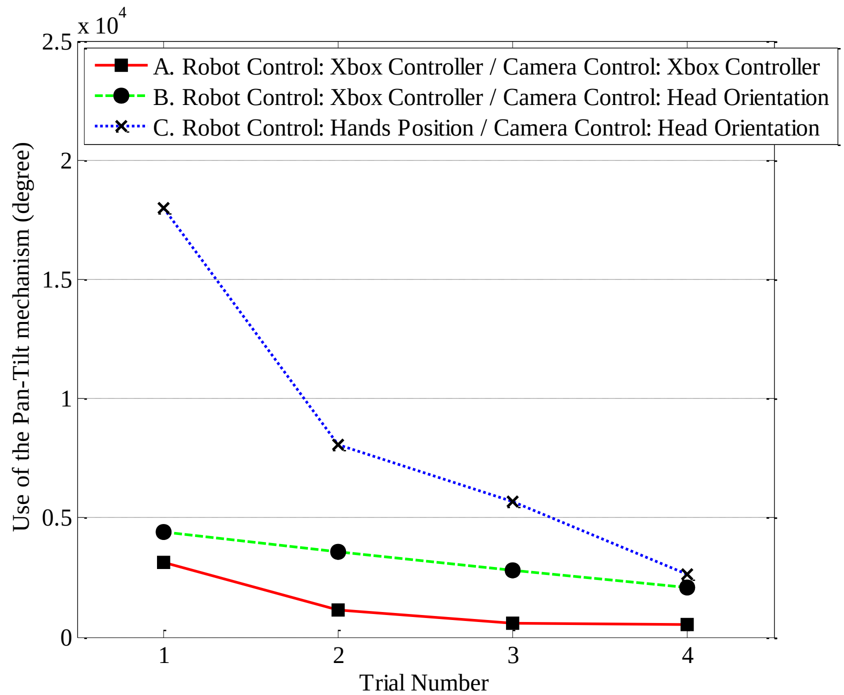

Figure 57: Effect of trial number and control mode on the use of the pan-tilt mechanism

D. Heart Rate

Heart rate was impacted by the trial number, $F(3,32)=2.806$, $p<0.05$, **the participants relative heart rate decreased with their experience of the system** apart between the first and second trial for the two conditions that involve physical movement (conditions B and C), where the relative heart rate increased (see Figure 58). The control mode had a significant effect, $F(2,33)=9.262$, $p<0.001$. However, the presence of a significant interaction between the control mode and the trial number, $F(6,30)=3.078$, $p<0.01$, required additional analysis. Post-hoc pairwise comparisons reveal that during the first trial, no significant difference can be found between the three conditions, but then the relative heart rate for the condition using hands gesture and head movement (condition C) was consistently higher than the one using only the Xbox controller (condition A).

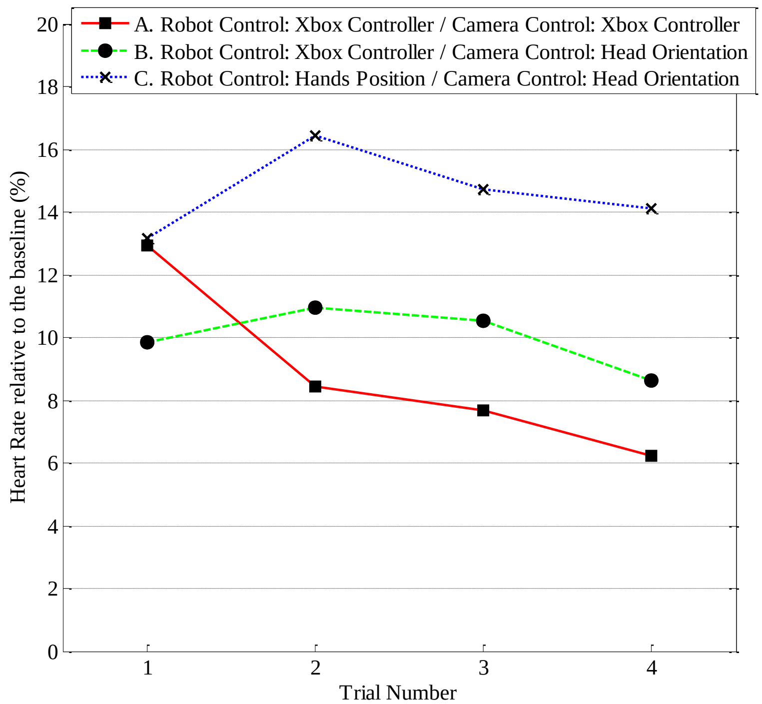

Figure 58: Effect of trial number and control mode on the participants' relative heart rate

Trial Number	Conditions		P-Value
1	A	B	,247
		C	,933
	B	C	,215
2	A	B	,353
		C	**,005**
	B	C	**,046**
3	A	B	,395
		C	**,042**
	B	C	,219
4	A	B	,438
		C	**,015**
	B	C	,085

E. Raw NASA-TLX questionnaire

Significant effect of the factors tested was found only on two dimensions of workload, Mental Demand and Physical demand.

Mental Demand and Physical Demand were affected by the control mode, F(2,33)=7.319, p<0.01 and F(3,22)=7.842, p<0.01, respectively. The condition A had the lowest values and the condition C the highest.

Only Mental Demand was impacted by the trial number, F(3,32)=11.48, p<0.01. It reduced with participants' experience for all conditions by an average of 29%.

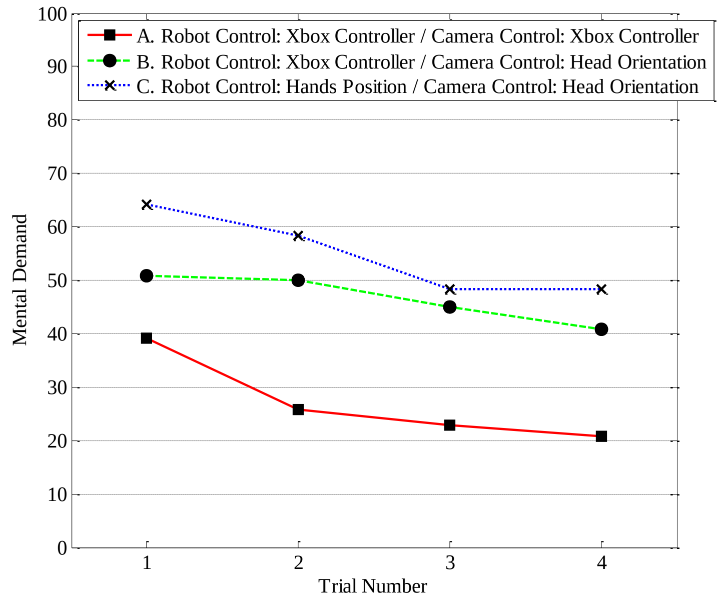

Figure 59: Effect of trial number and control mode on the participants' Mental Demand

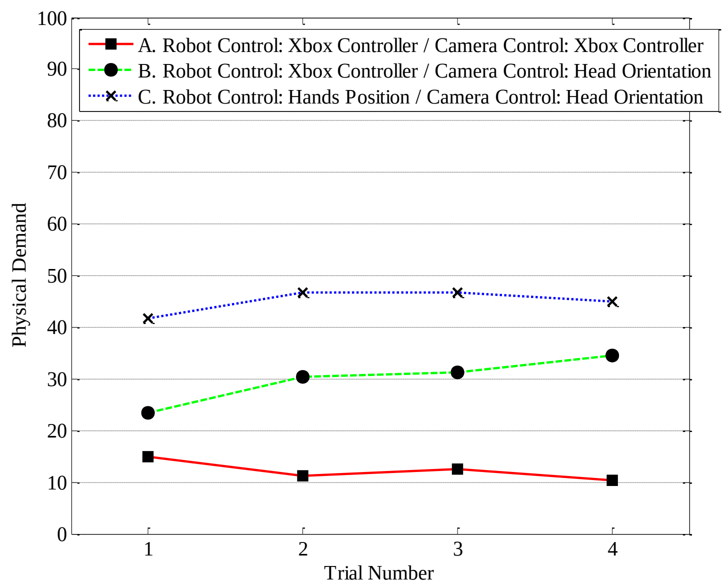

Figure 60: Effect of trial number and control mode on the participants' Physical Demand

8.4. Conclusions

In this experiment we developed and studied an unconventional form of camera orientation control for teleoperation: using head orientation without a head mounted display instead of a classical controller. With the rise of affordable sensors enabling the use of such non invasive techniques, this control scheme is likely to be more and more considered. Moreover, we used the first iteration of Kinect, which provides basic head tracking with important limitations in terms of accuracy, latency and range compared to the soon to be released new version. The fact that the user does not need to wear any sensor, or heavy virtual reality goggles, is what makes this technique very attractive. Furthermore, the technology of virtual reality head mounted display is seeing a dramatic acceleration of its development, with devices such as the Oculus Rift being released for a fraction of the price of current systems. It may be considered again as well for controlling a remote camera orientation since most of the issues of latency and motion dizziness appear to be on the way to be solved.

Hence, we studied the relevancy of using the user's head orientation to control the orientation of the remote robot camera orientation how it evolves with user experience.

This way of controlling the robot camera orientation was either combined with classical or gesture control of the robot movements, and compared to classical control of the camera orientation with a game controller joystick.

Results show that in terms of performance **using head orientation to control the camera combined with joystick control of the robot movements is better and more intuitive for users with no experience. However, as soon as the users gain experience with the system the advantages of using head orientation for the control of the camera over classical joystick control disappears.** When combined with robot movements, hands gesture control, camera control through head orientation revealed to be less intuitive and harder, but this degraded performance compared to other conditions decreases with user experience.

In terms of workload, heart rate measurements suggest that there is no difference between head orientation and joystick control of the camera, when combined with joystick control of the robot movement. However, **when using hands gestures, workload appears to be much higher**. This is consistent with Mental Demand and Physical Demand subjective measurements showing that the use of hands gesture for robot control combined with head orientation control of the camera is more demanding. With these last measurements, classical control of the camera and the robot with the Xbox controller appears less demanding than the two other conditions. This may be due to the sensitivity of the control mode, which requires that the head stays perfectly still to maintain a constant camera orientation, and requires very accurate head movements to control the camera because of the 2.5 ratio between the head and camera orientation.

Additionally, when the orientation of the camera was controlled with the head, users tend to move it more than with the stick control. It is unclear if these movements were voluntary or a consequence of natural small head movements.

9. Conclusions and Recommendations

9.1. Part A: Interface design for learning robots

This research started to tackle a largely unexplored domain: interface design for learning robots. Learning algorithms for robots are quickly gaining in maturity but the question of how they should be implemented on a user perspective remains mostly unanswered. This dissertation aimed to begin to bridge that gap. After conducting and analyzing the results of the three experiments, it is clear that the human interaction with a learning robot is not trivial and sensitive to many parameters. Hence, creators of human-robot interfaces for learning robots should be aware of the inherent complexity in terms of interaction of such systems, even with the simplest form of learning. Following is a list of recommendations or guidelines for designers from the knowledge gathered during these experiments:

- In the context of online robot learning in changing environments, providing information to help the user understand the validity of the robot's learned behavior is very important for the user and to the whole system performance.
- Not every type of information is beneficial. The simplistic thought that the more information is provided, the better will be the performance is false. Too much information, even perfectly accurate, can degrade the performance.
- The best way to inform users of changes in the environment is brief and contextualized notifications.
- Giving the ability to the user to see the future actions of the automation gained from learning is disturbing and counter-productive.
- The sensitivity of these results to the number of changes tested appears to be limited.
- In the context of an automated system with changing characteristics, providing feedback about changes in the characteristics in the form of notifications or continuous information is beneficial.
- The user adaptation to changes is slightly faster when the feedback is provided in the form of continuous information.
- Not informing the user about changes of characteristics of the system leads to situations of under or over trust directly impacting the performance.

- Previous experience has an impact on the user response to new changes of system characteristics: users who experienced a positive change in terms of performance before have more trouble detecting and reacting properly to new negative changes.
- Users are relatively robust to misleading or fake information about changes, they are able to discard them and react in the same way as if they received no information, and not worst.
- In the context of online robot learning in a changing environment, for which the users receive notification when it changes, the level of automation of the learning of the robot has an important effect on the performance of the task learnt, on the way users make use of the learning and on the performance of a parallel task.
- Depending on the level of automation at which the learning is applied and the situation, performance with learning can be worse than without learning. In particular, applying the learning in form of suggestions or approvable suggestions presents very little to no advantages when compared to no learning.
- Users make the best use of a learning robot when they can use it in the form of switchable automation, i.e. when they can switch between fully manual and fully autonomous.

This list of guidelines constitutes a basis for the design of future human-robot interfaces for learning robots, but there is still a lot to explore in this area. Results need to be generalized to more application scenarios and different learning algorithms. The reflection on the level of autonomy of the learning needs to be extended to more learning implementations modes, a more adaptive control than the basic switchable automation tested in this research should be achievable. Moreover, the level of autonomy at which the system is learning, i.e. the control the user has on what the system learns and when, should be considered, even if the resulting complexity would be probably hard to handle for the user. In this research learning was always activated, the user had no control on this aspect.

9.2.Part B: Advanced human robot interfaces

This research focused on the creation and development of novel approaches for making robots more usable for end users. A person following platform and accompanying algorithms were developed. Then a pointing control interface building upon the person following platform was designed. Next a direct physical interface enabling the control of a mobile robot by directly pushing it was created. All these 3 interfaces were then compared in set of user experiments. It turns out that the direct physical interface is the easiest and most intuitive one to use, but if a contact less interface is required, the following interface is the best. The pointing interface, even if it is appealing in terms of novelty revealed to be harder to use.

Additionally, a novel interface for remote robot teleoperation was created and tested: using the operator head movements to control the orientation of the camera of the distant robot. This interface proved to be more intuitive for novice users, however as the user practice of the system progressed, the more classical control interface based on joysticks showed better performance.

From these experiments it emerges that new sensor technologies permits the development of a lot of new human-robot interaction modalities and offer different creative approaches for interface designers. Overall we observe that robot control by body movements (e.g. person following, pointing control or head control of camera orientation) presents the advantage of being appeling for novices and more intuitive for users that have no experience. Moreover they are contact free which can be a requirement in some specific application (e.g. sterile medical environments). However, we noted that more classical interfaces are still better in terms of performance if the user has enough time to practice and get familiar with the system. Future work should focus on developing interfaces that are intuitive but also offer a margin of progression big enough to reach with time high levels of performance. Additionally, we tested a direct physical interaction modality which appeared promising both in terms of intuitivity and performance, but the feasibility of its implementation in more general applications needs to be investigated.

References

Alfano, P.L. & Michel, G.F., 1990. Restricting the field of view: perceptual and performance effects. *Perceptual and Motor Skills*, 70(1), pp.35–45.

Allison, R.S. et al., 2001. Tolerance of temporal delay in virtual environments. In *Proceedings IEEE Virtual Reality 2001*. IEEE Comput. Soc, pp. 247–254.

Alvarez-Santos, V. et al., 2012. Feature analysis for human recognition and discrimination: Application to a person-following behaviour in a mobile robot. *Robotics and Autonomous Systems*, 60(8), pp.1021–1036.

Anon, Face Tracking. Available at: http://msdn.microsoft.com/en-us/library/jj130970.aspx [Accessed September 5, 2013].

Argall, B.D. et al., 2009. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5), pp.469–483.

Azarbayejani, A. & Pentland, A., 1996. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. In *Proceedings of 13th International Conference on Pattern Recognition*. IEEE, pp. 627–632 vol.3.

Bahner, J.E., Elepfandt, M.F. & Manzey, D., 2008. Misuse of Diagnostic Aids in Process Control: The Effects of Automation Misses on Complacency and Automation Bias. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 52(19), pp.1330–1334.

Bailey, N.R. & Scerbo, M.W., 2007. Automation-induced complacency for monitoring highly reliable systems: the role of task complexity, system experience, and operator trust. *Theoretical Issues in Ergonomics Science*, 8(4), pp.321–348.

Bechar, A., Meyer, J. & Edan, Y., 2009. An Objective Function to Evaluate Performance of Human–Robot Collaboration in Target Recognition Tasks. *IEEE Transactions on Systems Man and Cybernetics Part C Applications and Reviews*, 39(6), pp.611–620.

Beer, J.M. et al., 2012. "Commanding Your Robot" Older Adults' Preferences for Methods of Robot Control. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 56(1), pp.1263–1267.

Bellotto, N. & Hu, H., 2009. Multisensor-based human detection and tracking for mobile service robots. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, 39(1), pp.167–81.

Billings, C., 1997. *Aviation Automation, The search for the Human-Centered Approach*, Mahwah, NJ: Lawrence Erlbaum.

Biocca, F.A. & Rolland, J.P., 1998. Virtual eyes can rearrange your body: adaptation to visual displacement in see-through, head-mounted displays. *Presence: Teleoperators Virtual Environments*, 7(3), pp.262–277.

Bodiroža, S. et al., 2013. Relation of Motion Control and Gestures Through Self-Exploration. In *Proceedings of the 1st Workshop on Robotics Challenges and Vision, Robotics: Science and Systems Conference*. pp. 21–24.

Bodiroza, S., Doisy, G. & Hafner, V.V., 2013. Position-invariant, real-time gesture recognition based on dynamic time warping. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, pp. 87–88.

Burke, J.L. et al., 2004. Moonlight in Miami: Field Study of Human-Robot Interaction in the Context of an Urban Search and Rescue Disaster Response Training Exercise. *Human–Computer Interaction*, 19(1-2), pp.85–116.

Byrne, E.A. & Parasuraman, R., 1996. Psychophysiology and adaptive automation. *Biological Psychology*, 42(3), pp.249–268.

Calinon, S. & Billard, A.G., 2007. What is the teacher's role in robot programming by demonstration? Toward benchmarks for improved learning. *Interaction Studies*, 8(3), pp.441–464.

Calinon, S., Guenter, F. & Billard, A., 2007. On Learning, Representing, and Generalizing a Task in a Humanoid Robot. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, 37(2), pp.286–298.

Calisi, D., Iocchi, L. & Leone, R., 2007. Person following through appearance models and stereo vision using a mobile robot. In *Proceeding of the International Workshop on Robot Vision*. pp. 46–56.

Camplani, M. & Salgado, L., 2012. Efficient Spatio-Temporal Hole Filling Strategy for Kinect Depth Maps. In *IS&T/SPIE Int. Conf. on 3D Image Processing (3DIP) and Applications*.

Casper, J. & Murphy, R.R., 2003. Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center. *IEEE transactions on systems, man, and cybernetics. Part B (Cybernetics)*, 33(3), pp.367–85.

Chaudhary, A. et al., 2011. Intelligent Approaches to interact with Machines using Hand Gesture Recognition in Natural way: A Survey. *International Journal of Computer Science and Engineering Survey*, 2(1), pp.122–133.

Chen, J.Y.C., Haas, E.C. & Barnes, M.J., 2007. Human performance issues and user interface design for teleoperated robots. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(6), pp.1231–1245.

Chen, T.L. & Kemp, C.C., 2011. A Direct Physical Interface for Navigation and Positioning of a Robotic Nursing Assistant. *Advanced Robotics*, 25(5), pp.605–627.

Chen, T.L. & Kemp, C.C., 2010. Lead me by the hand: evaluation of a direct physical interface for nursing assistant robots. , pp.367–374.

Cipolla, R., Hadfield, P. & Hollinghurst, N., 1994. Uncalibrated stereo vision with pointing for a man-machine interface. *Proceedings of the IAPR Workshop on Machine Vision Applications (MVA'94)*.

Cipolla, R. & Hollinghurst, N.J., 1996. Human-robot interface by pointing with uncalibrated stereo vision. *Image and Vision Computing*, 14(3), pp.171–178.

Cobb, S.V.G., 1999. Measurement of postural stability before and after immersion in a virtual environment. *Applied Ergonomics*, 30(1), pp.47–57.

Cohen, M.H., Giangola, J.P. & Balogh, J., 2004. *Voice User Interface Design*, Boston, MA: Addison-Wesley Professional.

Crick, C. et al., 2011. Human and robot perception in large-scale learning from demonstration. In *Proceedings of the 6th ACM/IEEE International Conference on Human-robot interaction*. pp. 339–346.

Darrell, T. et al., 2000. Integrated Person Tracking Using Stereo, Color, and Pattern Detection. *International Journal of Computer Vision*, 37(2), pp.175–185.

Diego-Mas, J.A. & Alcaide-Marzal, J., 2014. Using Kinect™ sensor in observational methods for assessing postures at work. *Applied ergonomics*, 45(4), pp.976–85.

Dixon, S.R. & Wickens, C.D., 2006. Automation Reliability in Unmanned Aerial Vehicle Control: A Reliance-Compliance Model of Automation Dependence in High Workload. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(3), pp.474–486.

Dixon, S.R., Wickens, C.D. & McCarley, J.S., 2007. On the Independence of Compliance and Reliance: Are Automation False Alarms Worse Than Misses? *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 49(4), pp.564–572.

Doisy, G., 2012. Sensorless Collision Detection and Control by Physical Interaction for Wheeled Mobile Robots. In *Proceeding of the 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI 2012)*. Boston.

Droeschel, D., Stückler, J. & Behnke, S., 2011. Learning to interpret pointing gestures with a time-of-flight camera. In *Proceedings of the 6th international conference on Human-robot interaction - HRI '11*. New York, New York, USA: ACM Press, p. 481.

Drury, J.L., Scholtz, J. & Yanco, H.A., 2003. Awareness in human-robot interactions. In *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics*. pp. 912–918.

Dzindolet, M.T. et al., 2003. The role of trust in automation reliance. *International Journal of Human-Computer Studies*, 58(6), pp.697–718.

Edsinger, A. & Kemp, C.C., 2007. Human-Robot Interaction for Cooperative Manipulation: Handing Objects to One Another. In *RO-MAN 2007 - The 16th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, pp. 1167–1172.

Eliav, A. et al., 2011. Advanced methods for displays and remote control of robots. *Applied ergonomics*, 42(6), pp.820–9.

Endres, F. et al., 2012. An Evaluation of the RGB-D SLAM System. In *Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA 2012)*. IEEE.

Endsley, M.R., 1987. The application of human factors to the development of expert systems for advanced cockpits. *Proceedings of the Human Factors Society 31st Annual Meeting*, pp.1388–1392.

Endsley, M.R. & Kiris, E.O., 1995. The Out-of-the-Loop Performance Problem and Level of Control in Automation. *Human Factors*, 37(2), pp.381–394.

Ferland, F. et al., 2013. Taking your robot for a walk: Force-guiding a mobile robot using compliant arms. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, pp. 309–316.

Fiala, M., 2005. Pano-presence for teleoperation. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 3798–3802.

Findlater, L. & McGrenere, J., 2004. A comparison of static, adaptive, and adaptable menus. *Proceedings of the 2004 conference on Human factors in computing systems CHI 04*, 6(1), pp.89–96.

Findlater, L. & McGrenere, J., 2008. Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces. *Proceeding of the twentysixth annual CHI conference on Human factors in computing systems CHI 08*, p.1247.

Fong, T., Nourbakhsh, I. & Dautenhahn, K., 2003. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42(3-4), pp.143–166.

Fong, T., Thorpe, C. & Baur, C., 2001. *Collaborative control: a robot-centric model for vehicle teleoperation*. Carnegie Mellon University.

Fuchs, M. et al., 2009. Rollin' Justin - Design considerations and realization of a mobile platform for a humanoid upper body. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, pp. 4131–4137.

Gajos, K.Z. et al., 2006. Exploring the design space for adaptive graphical user interfaces. In *Proceedings of the working conference on Advanced visual interfaces AVI 06*. AVI '06. ACM Press, p. 201.

Gates, B., 2007. A Robot in Every Home. *Scientific American*, 296(1), pp.58–65.

Gu, Y. & Veloso, M., 2009. Effective Multi-Model Motion Tracking using Action Models. *The International Journal of Robotics Research*, 28(1), pp.3–19.

Haddadin, S. et al., 2008. Collision detection and reaction: A contribution to safe physical Human-Robot Interaction. In *Proceeding of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 3356–3363.

Haddadin, S., Albu-Schäffer, A. & Hirzinger, G., 2010. Soft-tissue injury in robotics. In *Proceeding of the 2010 IEEE International Conference on Robotics and Automation*. IEEE, pp. 3426–3433.

Hale, J.G. & Pollick, F.E., 2005. "Sticky Hands": Learning and Generalization for Cooperative Physical Interactions With a Humanoid Robot. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 35(4), pp.512–521.

Harriott, C.E. & Zhang, T., 2011. Predicting and validating workload in human-robot teams. *Proceedings of the 20th Conference on Behavior Representation in Modeling and Simulation*, pp.162 – 169.

Hart, S.G. & Staveland, L.E., 1988. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In P. A. Hancok & N. Meshkati, eds. *Human Mental Workload*. Advances in Psychology. Elsevier, pp. 139–183.

Hilburn, B. et al., 1993. Operator versus computer control of adaptive automation. In R. S. Jensen & D. Neumeister, eds. *Proceedings of the Seventh International Symposium on Aviation Psychology*. Department of Aviation, The Ohio State University, p. 31.

Hogan, N., 1984. Impedance Control: An Approach to Manipulation. , pp.304–313.

Hughes, S. et al., 2003. Camera control and decoupled motion for teleoperation. In *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics*. pp. 1339–1344.

Ikeura, R. & Inooka, H., 1995. Variable impedance control of a robot for cooperation with a human. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*. IEEE, pp. 3097–3102.

Izadi, S. et al., 2011. KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. pp. 559–568.

Jain, A. & Kemp, C.C., 2009. Pulling open novel doors and drawers with equilibrium point control. In *2009 9th IEEE-RAS International Conference on Humanoid Robots*. IEEE, pp. 498–505.

Jia, Z., Balasuriya, A. & Challa, S., 2009. Vision Based Target Tracking for Autonomous Land Vehicle Navigation: A Brief Survey. *Recent Patents on Computer Science*, 2(1), pp.32–42.

Jojic, N. et al., 2000. Detection and estimation of pointing gestures in dense disparity maps. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*. IEEE Comput. Soc, pp. 468–475.

Jouppi, N.P., 2002. First steps towards mutually-immersive mobile telepresence. In *Proceedings of the 2002 ACM conference on Computer supported cooperative work - CSCW '02*. New York, New York, USA: ACM Press, p. 354.

Kaber, D.B. & Endsley, M.R., 2004. The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theoretical Issues in Ergonomics Science*, 5(2), pp.113–153.

Kemp, C.C. et al., 2008. A point-and-click interface for the real world: Laser designation of objects for mobile manipulation. , pp.241–248.

Keyes, B. & Yanco, H.A., 2006. Camera placement and multi-camera fusion for remote robot operation. In *IEEE International Workshop on Safety, Security and Rescue Robotics*.

Kobilarov, M. et al., 2006. People tracking and following with mobile robot using an omnidirectional camera and a laser. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, pp. 557–562.

Koceski, S., Koceska, N. & Kocev, I., 2012. Design and Evaluation of Cell Phone Pointing Interface for Robot Control. *International Journal of Advanced Robotic Systems*, 9, p.1.

Krevelen, D.W.F. Van & Poelman, R., 2010. A Survey of Augmented Reality Technologies, Applications and Limitations. *The International Journal of Virtual Reality*, 9(2), pp.1–20.

Kruse, T. et al., 2013. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12), pp.1726–1743.

Lavie, T. & Meyer, J., 2010. Benefits and costs of adaptive user interfaces. *International Journal of Human-Computer Studies*, 68(8), pp.508–524.

Lee, J.D. & See, K.A., 2004. Trust in Automation: Designing for Appropriate Reliance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 46(1), pp.50–80.

Liu, X. & Fujimura, K., 2004. Hand gesture recognition using depth data. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.* IEEE, pp. 529–534.

Love, L.J. & Book, W.J., 2004. Force Reflecting Teleoperation With Adaptive Impedance Control. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, 34(1), pp.159–165.

Luber, M., Diego Tipaldi, G. & Arras, K.O., 2011. Place-dependent people tracking. *The International Journal of Robotics Research*, 30(3), pp.280–293.

Luber, M., Spinello, L. & Arras, K.O., 2011. People tracking in RGB-D data with on-line boosted target models. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 3844–3849.

De Luca, A. et al., 2006. Collision Detection and Safe Reaction with the DLR-III Lightweight Manipulator Arm. In *Proceeding of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 1623–1630.

De Luca, A. & Mattone, R., 2004. An adapt-and-detect actuator FDI scheme for robot manipulators. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*. IEEE, pp. 4975–4980 Vol.5.

De Luca, A. & Mattone, R., 2005. Sensorless Robot Collision Detection and Hybrid Force/Motion Control. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, (April), pp.999–1004.

Madhavan, P. & Wiegmann, D.A., 2007. Effects of Information Source, Pedigree, and Reliability on Operator Interaction With Decision Support Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 49(5), pp.773–785.

Mania, K. et al., 2004. Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity. In *Proceedings of the 1st Symposium on Applied perception in graphics and visualization - APGV '04*. New York, New York, USA: ACM Press, p. 39.

Martinez-Otzeta, J.M. et al., 2009. Laser Based People Following Behaviour in an Emergency Environment. In *Proceedings of the 2nd International Conference on Intelligent Robotics and Applications (ICIRA '09)*. pp. 33–42.

Martínez-Otzeta, J.M. et al., 2010. People following behaviour in an industrial enviroment using laser and stereo camera. *Trends in Applied Intelligent Systems*, 6098, pp.508–517.

McCulloch, C.E. & Searle, S.R., 2000. *Generalized, Linear, and Mixed Models*, Hoboken, NJ, USA: John Wiley & Sons, Inc.

Meyer, J., 2004. Conceptual issues in the study of dynamic hazard warnings. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 46(2), pp.196–204.

Meyer, J., 2001. Effects of Warning Validity and Proximity on Responses to Warnings. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 43(4), pp.563–572.

Meyer, J., Wiczorek, R. & Gunzler, T., 2013. Measures of Reliance and Compliance in Aided Visual Scanning. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 56(5), pp.840–849.

Mitra, S. & Acharya, T., 2007. Gesture Recognition: A Survey. *IEEE Transactions on Systems Man and Cybernetics Part C Applications and Reviews*, 37(3), pp.311–324.

Moeslund, T.B., Hilton, A. & Krüger, V., 2006. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2-3), pp.90–126.

Morin, P. & Samson, C., 2008. Motion control of wheeled mobile robots. In *Handbook of Robotics*. Springer, pp. 799–826.

Motai, Y., Kumar Jha, S. & Kruse, D., 2012. Human tracking from a mobile agent: Optical flow and Kalman filter arbitration. *Signal Processing: Image Communication*, 27(1), pp.83–95.

Mulling, K. et al., 2013. Learning to select and generalize striking movements in robot table tennis. *The International Journal of Robotics Research*, 32(3), pp.263–279.

Murphy, R.R., 2004. Human–Robot Interaction in Rescue Robotics. *IEEE Transactions on Systems Man and Cybernetics Part C Applications and Reviews*, 34(2), pp.138–153.

Najmaei, N. & Kermani, M.R., 2011. Applications of artificial intelligence in safe human-robot interactions. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, 41(2), pp.448–59.

Nehaniv, C.L. & Dautenhahn, K., 2007. *Imitation and Social Learning in Robots, Humans and Animals*, Cambridge University Press.

Nickel, K. & Stiefelhagen, R., 2007. Visual recognition of pointing gestures for human–robot interaction. *Image and Vision Computing*, 25(12), pp.1875–1884.

Nielsen, C.W., Goodrich, M.A. & Ricks, R.W., 2007. Ecological interfaces for improving mobile robot teleoperation. *IEEE Transactions on Robotics*, 23(5), pp.927–941.

Nielsen, C.W., Goodrich, M.A. & Rupper, R.J., 2005. Towards facilitating the use of a pan-tilt camera on a mobile robot. In *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication*. pp. 568–573.

Parasuraman, R. et al., 1993. Adaptive function allocation reduces performance costs of static automation R. S. Jensen & D. Neumeister, eds. *Proceedings of the Seventh International Symposium on Aviation Psychology*, pp.178–185.

Pieraccini, R., 2012. *The Voice in the Machine: Building Computers That Understand Speech*, The MIT Press.

Pope, A.T., Bogart, E.H. & Bartolome, D.S., 1995. Biocybernetic system evaluates indices of operator engagement in automated task. *Biological Psychology*, 40(1-2), pp.187–195.

Pradhan, N., 2013. Mobile Robot Navigation for Person Following in Indoor Environments.

Pucci, D., Marchetti, L. & Morin, P., 2013. Nonlinear control of unicycle-like robots for person following. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 3406–3411.

Rice, S., 2009. Examining single- and multiple-process theories of trust in automation. *The Journal of general psychology*, 136(3), pp.303–19.

Rice, S. & McCarley, J.S., 2011. Effects of Response Bias and Judgment Framing on Operator Use of an Automated Aid in a Target Detection Task. *Journal of Experimental Psychology: Applied*, 17(4), pp.320–331.

Ricks, B., Nielsen, C.W. & Goodrich, M.A., 2004. Ecological displays for robot interaction: a new perspective. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 2855–2860.

Rodriguez, G. & Weisbin, C.R., 2003. A New Method to Evaluate Human-Robot System Performance. *Autonomous Robots*, 14(2), pp.165–178.

Roscoe, A.H., 1992. Assessing pilot workload. Why measure heart rate, HRV and respiration? *Biological Psychology*, 34(2-3), pp.259–287.

Roscoe, A.H., 1993. Heart rate as a psychophysiological measure for in-flight workload assessment. *Ergonomics*, 36(9), pp.1055–1062.

Rothrock, L. et al., 2002. Review and reappraisal of adaptive interfaces: toward biologically inspired paradigms. *Theoretical Issues in Ergonomics Science*, 3(1), pp.47–84.

Rouanet, P. et al., 2013. The Impact of Human–Robot Interfaces on the Learning of Visual Objects. *IEEE Transactions on Robotics*, 29(2), pp.525–541.

Rouanet, P. & Danieau, F., 2011. A robotic game to evaluate interfaces used to show and teach visual objects to a robot in real world condition. In *Human Factors*. HRI '11. ACM, pp. 313–320.

Rouanet, P., Oudeyer, P.-Y. & Filliat, D., 2009. An integrated system for teaching new visually grounded words to a robot for non-expert users using a mobile device. In *9th IEEE-RAS International Conference on Humanoid Robots*. IEEE, pp. 391–398.

Rouanet, P., Oudeyer, P.-Y. & Filliat, D., 2010. Using mediator objects to easily and robustly teach visual objects to a robot. In *ACM SIGGRAPH*. New York, New York, USA: ACM Press.

Rouse, W.B., Geddes, N.D. & Curry, R.E., 1986. An architecture for intelligent interfaces - Outline of an approach to supporting operators of complex systems. *Human-Computer Interaction*, 3(2), pp.87–122.

Satake, J. & Miura, J., 2010. Stereo-Based Multi-person Tracking Using Overlapping Silhouette Templates. In *Proceeding of the 2010 20th International Conference on Pattern Recognition*. IEEE, pp. 4304–4307.

Saunders, J., Otero, N. & Nehaniv, C.L., 2007. Issues in Human/Robot Task Structuring and Teaching. In *RO-MAN - The 16th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, pp. 708–713.

Scholtz, J. et al., 2004. Evaluation of human-robot interaction awareness in search and rescue. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*. IEEE, pp. 2327–2332 Vol.3.

Scholtz, J., 2003. Theory and evaluation of human robot interactions. In *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*. IEEE, p. 10 pp.

Schulz, D. et al., 2003. People Tracking with Mobile Robots Using Sample-Based Joint Probabilistic Data Association Filters. *The International Journal of Robotics Research*, 22(2), pp.99–116.

Schwarz, L.A. et al., 2012. Human skeleton tracking from depth data using geodesic distances and optical flow. *Image and Vision Computing*, 30(3), pp.217–226.

Scribner, D.R. & Gombash, J.W., 1998. *The Effect of Stereoscopic and Wide Field of View Conditions on Teleoperator Performance.*,

Sheridan, T.B., 2006. *Handbook of Human Factors and Ergonomics* 3rd ed. G. Salvendy, ed., Hoboken, NJ, USA: John Wiley & Sons, Inc.

Sheridan, T.B. & Verplank, W.L., 1978. *Human and computer control of undersea teleoperators*, Cambridge University Press.

Shibata, S., Yamamoto, T. & Jindai, M., 2011. Human-robot interface with instruction of neck movement using laser pointer. In *2011 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, pp. 1226–1231.

Shotton, J. et al., 2011. Real-time human pose recognition in parts from single depth images. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011)*. Colorado Springs, CO, USA: IEEE, pp. 1297–1304.

Smyth, C.C., 2000. Indirect Vision Driving with Fixed Flat Panel Displays for Near Unity, Wide, and Extended Fields of Camera View. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. pp. 541–544.

Spinello, L., Triebel, R. & Siegwart, R., 2010. Multiclass Multimodal Detection and Tracking in Urban Environments. *The International Journal of Robotics Research*, 29(12), pp.1498–1515.

Suarez, J. & Murphy, R.R., 2012. Hand gesture recognition with depth images: A review. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, pp. 411–417.

Suzuki, T., Ohya, A. & Yuta, S., 2005. Operation direction to a mobile robot by projection lights. In *IEEE Workshop on Advanced Robotics and its Social Impacts, 2005*. IEEE, pp. 160–165.

Takeda, T., Hirata, Y. & Kosuge, K., 2007. Dance Step Estimation Method Based on HMM for Dance Partner Robot. *IEEE Transactions on Industrial Electronics*, 54(2), pp.699–706.

Thomas, L.C. & Wickens, C.D., 2000. *Effects of Display Frames of Reference on Spatial Judgments and Change Detection*,

Thomaz, A.L. & Breazeal, C., 2007. Robot learning via socially guided exploration. In *IEEE 6th International Conference on Development and Learning*. IEEE, pp. 82–87.

Thomaz, A.L. & Breazeal, C., 2008. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence*, 172(6-7), pp.716–737.

Thrun, S., Burgard, W. & Fox, D., 2005. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*, The MIT Press.

Tittle, J.S., Roesler, A. & Woods, D.D., 2002. The remote perception problem. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 46(3), pp.260–264.

Tkach, I., Bechar, A. & Edan, Y., 2011. Switching Between Collaboration Levels in a Human–Robot Target Recognition System. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(6), pp.955–967.

Tsandilas, T. & Schraefel, M.C., 2005. An empirical assessment of adaptation techniques. *CHI 05 extended abstracts on Human factors in computing systems CHI 05*, pp.2009–2012.

Turner, J.R. & Carroll, D., 1985. Heart Rate and Oxygen Consumption during Mental Arithmetic, a Video Game, and Graded Exercise: Further Evidence of Metabolically-Exaggerated Cardiac Adjustments? *Psychophysiology*, 22(3), pp.261–267.

Turro, N., Khatib, O. & Coste-Maniere, E., 2001. Haptically augmented teleoperation. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*. IEEE, pp. 386–392.

Voshell, M., Woods, D.D. & Phillips, F., 2005. Overcoming the keyhole in human-robot coordination: simulation and evaluation. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 49(3), pp.442–446.

Wachs, J.P. et al., 2011. Vision-based hand-gesture applications. *Communications of the ACM*, 54(2), p.60.

Wickens, C. & Hollands, J., 1999. *Engineering Psychology and Human Performance* 3rd ed., Upper Saddle River, NJ: Prentice Hall.

Wickens, C.D. & Kessel, C., 1977. The effects of participatory mode and task workload on the detection of dynamic system failures. *IEEE Transactions on System Man and Cybernetics*, 9, pp.24–34.

Wiczorek, R., Meyer, J. & Guenzler, T., 2012. On the Relation Between Reliance and Compliance in an Aided Visual Scanning Task. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 56(1), pp.253–257.

Wiener, E.L. & Curry, R.E., 1980. Flight-deck automation - Promises and problems. *Ergonomics*, 23(10), pp.995–1011.

Wilson, A. & Shafer, S., 2003. XWand. In *Proceedings of the conference on Human factors in computing systems - CHI '03*. New York, New York, USA: ACM Press, p. 545.

Woods, D.D. et al., 2004. Envisioning Human–Robot Coordination in Future Operations. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 34(2), pp.210–218.

Wren, C.R. et al., 1997. Pfinder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7), pp.780–785.

Wyrobek, K.A. et al., 2008. Towards a personal robotics development platform: Rationale and design of an intrinsically safe personal robot. In *2008 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2165–2170.

Yanco, H.A. & Drury, J., 2004. “Where am i?” acquiring situation awareness using a remote robot platform. In *2004 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, pp. 2835–2840.

Yanco, H.A., Drury, J.L. & Scholtz, J., 2004. Beyond Usability Evaluation: Analysis of Human-Robot Interaction at a Major Robotics Competition. *Human-Computer Interaction*, 19(1), pp.117–149.

Yoshimi, T. et al., 2006. Development of a Person Following Robot with Vision Based Target Detection. In *Proceeding og the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 5286–5291.

Zalud, L., 2006. ARGOS - System for Heterogeneous Mobile Robot Teleoperation. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 211–216.

Appendices

Appendix A: Experimental Material for *The effect of feedback and environmental changes on the use of a learning robot system*

Software

The developed simulation software C# source used for this experiment is included in the attached CD in the following folder: *Appendices\A. The effect of feedback and environmental changes on the use of a learning robot system\Software*

Raw results

The metrics measured during the experiment and used for the analysis are included in the CD in the form of an excel format: *Appendices\A. The effect of feedback and environmental changes on the use of a learning robot system\Raw results\Metrics.xlsx*

Appendix B: Experimental Material for *Responses to warnings and the effect of feedback about changes in a simulated robot-control task*

Software

The developed simulation software C# source used for this experiment is included in the attached CD in the following folder: *Appendices\B. Responses to warnings and the effect of feedback about changes in a simulated robot-control task\Software*

Raw results

The metrics measured during the experiment and used for the analysis are included in the CD in the form of an excel format: *Appendices\B. Responses to warnings and the effect of feedback about changes in a simulated robot-control task\Raw results\Metrics.xlsx*

Appendix C: Experimental Material for *The effect of the level of automation of the learning on the use of learning robot system*

Software

The experiment apparatus comprised to computer. One windows computer running the control interface, the log system and the robot communication software. Its C# source is given is the following folder: *Appendices\C. The effect of the level of automation of the learning on the use of learning robot system\Interface Software*

The other computer was embedded on the Pioneer LX robot and was under Linux Ubuntu running ROS with the ROSARIA, Navigation and Rosbridge stacks.

Raw results

The metrics measured during the experiment and used for the analysis are included in the CD in the form of an excel format: *Appendices\C. The effect of the level of automation of the learning on the use of learning robot system\Raw results\Metrics.xlsx*

Appendix D: Experimental Material for *Adaptive Person-Following Algorithm Based on Depth Images and Mapping*

Software

A Windows computer was connected to a Robulab10 robot platform. The computer was running MRDS and the source code of the services written to control the robot, the pan-tilt mechanism and the Kinect sensor are included in the CD in the following folder*: Appendices\D. Adaptive Person-Following Algorithm Based on Depth Images and Mapping\Software*.

Appendix E: Experimental Material for *Comparison of novel interfaces for mobile indoor robot control: direct physical interaction, person following and pointing control*

Software

The code source of the developed MRDS C# services used in this experiment can be found in the CD on the folder: *Appendices\E. Comparison of novel interfaces for mobile indoor robot control direct physical interaction, person following and pointing control\Software*.

Raw results

The metrics measured during the experiment and used for the analysis are included in the CD in the form of an excel format: *Appendices\E. Comparison of novel interfaces for mobile indoor robot control direct physical interaction, person following and pointing control \Raw results\Metrics.xlsx*

Appendix F: Experimental Material for *Non-invasive robot camera head control for teleoperation: performance and workload assessment*

Software

The needed libraries and the code source of the developed MRDS C# services used in this experiment can be found in the CD on the folder: *Appendices\F. Non-invasive robot camera head control for teleoperation performance and workload assessment\Software*.

Raw results

The metrics measured during the experiment and used for the analysis are included in the CD in the form of an excel format: *Appendices\F. Non-invasive robot camera head control for teleoperation performance and workload assessment\Raw results\Metrics.xlsx*