Article

Adapt or Perish? Exploring the Effectiveness of Adaptive DoF Control Interaction Methods for Assistive Robot Arms

Kirill Kronhardt [1,*,†] , Stephan Rübner [1,*,†] , Max Pascher [1] , Felix Ferdinand Goldau [2] , Udo Frese [2] and Jens Gerken [1,*]

1   Human-Computer Interaction Group, Department of Media Informatics and Communication, Westphalian University of Applied Sciences, 45897 Gelsenkirchen, Germany; max.pascher@w-hs.de
2   German Research Center for Artificial Intelligence (DFKI) GmbH, 28359 Bremen, Germany; felix.goldau@dfki.de (F.F.G.); udo.frese@dfki.de (U.F.)
*   Correspondence: kirill.kronhardt@studmail.w-hs.de (K.K.); stephan.ruebner@studmail.w-hs.de (S.R.); jens.gerken@w-hs.de (J.G.)
†   These authors contributed equally to this work.

Abstract: Robot arms are one of many assistive technologies used by people with motor impairments. Assistive robot arms can allow people to perform activities of daily living (ADL) involving grasping and manipulating objects in their environment without the assistance of caregivers. Suitable input devices (e.g., joysticks) mostly have two Degrees of Freedom (DoF), while most assistive robot arms have six or more. This results in time-consuming and cognitively demanding mode switches to change the mapping of DoFs to control the robot. One option to decrease the difficulty of controlling a high-DoF assistive robot arm using a low-DoF input device is to assign different combinations of movement-DoFs to the device's input DoFs depending on the current situation (adaptive control). To explore this method of control, we designed two adaptive control methods for a realistic virtual 3D environment. We evaluated our methods against a commonly used non-adaptive control method that requires the user to switch controls manually. This was conducted in a simulated remote study that used Virtual Reality and involved 39 non-disabled participants. Our results show that the number of mode switches necessary to complete a simple pick-and-place task decreases significantly when using an adaptive control type. In contrast, the task completion time and workload stay the same. A thematic analysis of qualitative feedback of our participants suggests that a longer period of training could further improve the performance of adaptive control methods.

Keywords: assistive robotics; human–robot interaction (HRI); shared user control; augmented reality; virtual reality; visual cues

1. Introduction

Robotic solutions are becoming increasingly prevalent in many areas of our professional and personal lives and have started to evolve into collaborators [1,2]. A non-negligible number of people live with motor impairments, ranging from slight limitations to severe paralysis [3]. While a near-complete integration into professional and social life is the final goal, current assistive robotic technologies focus on performing activities of daily living (ADLs). These include tasks ranging from essentials such as eating and drinking to more complex behaviors such as grooming and activities associated with leisure time [4].

A general problem with assistive robotic solutions is finding suitable methods and technologies for controlling such robots. Assistive robotic devices are often characterized as having a large number of Degrees of Freedom (high-DoF). For example, a robotic arm with a simple gripper can freely operate in 3D space and move along Cartesian space as well as yaw, pitch, and rotate. This typically results in five to seven DoFs. Standard input devices, such as joysticks, only cover two DoFs. To control a high-DoF device with a low-DoF input device, mode switching is used. This means that at any point in time, the user has to select

a mode, which then maps the two DoFs of the input device to two of the total available DoFs of the robot while neglecting the others. While high-DoF input devices do exist, they are not often accessible for people with motor impairments.

Using a human–computer interface with a standard button-based mode switching setup, Herlant et al. discovered that more than one-sixth of the total execution time is spent changing the currently selected mode [5]. They showed that automatic mode switching leads to increased user satisfaction within a deterministic simulation environment and with a predefined goal.

Our latest research findings provide a proof-of-concept for a novel method of shared control of an assistive robot. We evaluated the idea within a 2D simulation environment [6]. The novel control method uses a Convolutional Neural Network (CNN) to adaptively generate DoF mappings based on camera data of the current situation. From a user perspective, this system can help the user choose an optimal mapping of available control DoFs for a low-DoF input device, either automatically or upon the user's request. In this paper, we build on this approach, focusing in particular on the user interface. Having an adaptive mapping of control DoFs to the input device can be challenging to understand and learn, which is why there is a need for visual feedback to convey that information to the user. The approach in our previous work included visual cues in the form of arrows. While the results are promising (see Section 2), the limitation of a 2D environment means that it is difficult to predict how this approach transfers to 3D. For example, certain DoF combinations might be more difficult to display with arrows in a 3D environment and lead to visual clutter.

The goal of this paper is to explore the proposed novel control method, as well as possible visual cues for the DoF mappings. In particular, we want to explore how the novel, adaptive control method performs in a 3D environment compared to the standard mode-switch approach with cardinal DoF mappings and whether changes in the visual cues have an impact on the performance of the adaptive control method.

We conducted a remote online study with 39 non-disabled participants, in which we compared three different control types with different DoF mapping behaviors and visual cues. These were *Classic* and *Double Arrow*, which used two arrows attached to the fingers as visual cues, and a visually reduced variant *Single Arrow*. *Single Arrow* only used one arrow through the middle of the gripper (see Section 3 for a detailed description of each control type).

The study was conducted inside a 3D Virtual Reality (VR) environment, utilizing Head-Mounted Displays (HMDs) for an immersive experience (see Section 4.3 for a complete description of the virtual environment). The participants repeatedly performed a simple pick-and-place task, controlling a virtual robot arm using the three control types (see Section 4.5 for a detailed description of the study design).

Due to the ongoing COVID-19 pandemic, we opted to recruit non-specific participants that had access to the required hardware (an *Oculus Quest* VR-HMD) to participate in our study. None of the recruited participants reported living with any motor impairments. We acknowledge this limitation and discuss how our findings can be transferred to the target group of people with motor impairments in Section 7.

As our main contribution, we present findings from our study, which compare our two adaptive control types with the standard mode-switch control type, explicitly focusing on task completion times, number of mode switches and workload. In addition, we contribute an extensive discussion of qualitative results from voice recordings of our participants, providing a deeper understanding of the benefits and challenges of each of the three control types.

2. Related Work

To assist people with physical or cognitive impairments, prior research often suggests possible solutions that use robots that automate specific tasks [7–10]. Assistive robots are found in a variety of designs. There are stationary robots specifically designed for

meal-assistance [11], socially assistive robots for elderly people and people with cognitive impairments [12], navigational robots for blind people [13], and many more examples, both in research and commercially. Besides stationary robots (e.g., fixed to a table) [14], there are also moving robots attached to mobile platforms [15,16] or mounted to the user's wheelchair [9].

To help people with motor impairments, assistive robot arms are widely used, both within the workspace and in performing ADLs [17]. Their flexibility allows for many different applications, such as feeding assistance [18], fetch and pick-up tasks [15], and cataloging of books [7].

Robotic assistance is generally well-received by people with motor impairments. Drolshagen et al. found that people with disabilities quickly accept working with robots, even if the robots are in close proximity [19]. Regarding ADLs specifically, Pascher et al. conducted an ethnographic study with 15 participants with tetraplegia, multiple sclerosis, Locked-In Syndrome, and similar diseases [20]. They found that people with motor impairments would prefer to perform ADLs themselves with the help of a robotic aid as opposed to with the help of another person. People with motor impairments want to "live more independently" and "gain increased autonomy".

However, automating ADLs, as suggested in research, can have unintended consequences. Pollak et al. conducted a study comparing manual and autonomous modes of collaboration with a collaborative robot (cobot) [21]. They found that using the manual mode in which the cobot would perform tasks only upon interaction with the participants decreased stress significantly. The participants felt "more capable of coping with and controlling the situation" than in the autonomous mode.

Similarly, Kim et al. conducted a study with subjects with spinal cord injuries using an assistive robot arm in either a manual or an autonomous mode [22]. They found that overall task completion times for manual and autonomous usage for trained participants were similar, but user satisfaction was higher in manual mode. This is despite the fact that autonomous usage decreased the effort necessary to perform tasks significantly. The authors call for more flexible interfaces to control assistive robot arms.

When interacting with robots that carry out movements, a study by Cleaver et al. showed that users generally prefer to have a visual representation of the robot's future movements. However, having this visualization does not significantly affect the performance when executing tasks using the robot [23]. When using a visual representation of robot motion intent, the most prominent solution is to show the robot's movement using arrows [24–26]. In addition, most of these approaches rely on Augmented Reality to overlay the visual representation on the user's real environment.

Heeding the call for more flexible interfaces, we proposed in our recent work an adaptive control concept for assistive robot arms that promises to allow users to be in control at all times while still providing them with more assistance during ADLs than the standard mode switch control concept [6]. In this proposed concept, a CNN interprets the video feed of a camera attached to the robot arm and adaptively outputs the most likely movement DoFs.

With current control concepts, users with low-DoF input devices, such as simple joysticks, can only move the gripper of an assistive robot arm in cardinal directions (i.e., movement and rotation around Cartesian X-, Y-, and Z-Axes). The user has to switch and choose between the provided mappings of input DoFs to some of the robot's DoFs. This may include the pairings of different DoFs of the robot that are less than ideal for the given situation, resulting in many time-consuming and mentally demanding mode switches. Additionally, in any given mode, an input on an axis of a low-DoF device would move the gripper only in the cardinal direction currently assigned to this input DoF. Combinations of multiple output DoFs (such as orbiting an object, which is the combination of rotation and translation) require more than one input DoF (e.g., both the X- and Y-Axes of a joystick) to be engaged simultaneously in such systems.

To solve this problem, we proposed a representation of these assignments of input DoFs to output DoFs in the form of a matrix similar to the one seen in Figure 1 in our previous work. Each row in that matrix represents a cardinal output DoF, while each column represents the input DoFs of an input device. The values in a column determine which movement the robot's gripper will perform if the input DoF is engaged. For example, an identity matrix would yield a behavior identical to the cardinal mode switch approach, as each input DoF is only mapped to one cardinal output DoF.

$$
\begin{matrix}
X - Axis \\
Y - Axis \\
Z - Axis \\
Roll \\
Pitch \\
Yaw \\
Gripper
\end{matrix}
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
0.5 & 0 & 0 & 0.5 & 0 & 0 & 0 \\
0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 \\
0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0.5 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 \\
0 & 0 & 0.5 & 0 & 0 & 0.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

Figure 1. Two different DoF mappings as matrices—(**Left**): classic control (one input DoF controls one cardinal output DoF); (**Right**): arbitrarily combined controls (one input DoF controls more than one cardinal output DoF at the same time).

This representation allows for combinations of multiple output DoFs for one input DoF. For example, if the first column contains a value of 0.5 in the first two rows, engaging the first input DoF would result in a diagonal movement along the XY plane of the robot's coordinate system (see the matrix on the right in Figure 1). According to the current situation, the proposed control concept adaptively fills this matrix to create the most useful combination of output DoFs.

We then conducted a small study with a 2D proof-of-concept simulation for our proposed control concept. A total of 23 participants used a "standard" and an "adaptive" control type for a simulated 2D robot that could drive forwards, sideways, rotate around its center, and close its fingers to move blue boxes to target red boxes (see Figure 2). This is the 2D equivalent of a simple pick-and-place task in 3D. Both control types switched modes after five seconds without user input.

The results of our study showed that, subjectively, the "adaptive" control was significantly faster but significantly more difficult than the "standard" control. "Adaptive" control also led to significantly shorter sequence execution times.

While these findings are promising, the concept requires further evaluation in 3D and in a more complex environment with devices that have more DoFs. We set out to do precisely that: evaluate the proposed concept of adaptive control in a more complex environment with a robot arm with seven DoFs.


Figure 2. The simulated robot with two out of the four cardinal DoFs (**left**) and two adaptive DoFs (**right**) [6].

3. Control Types for a 3D Environment

To compare the standard control type of switching between cardinal modes to the adaptive approach, we implemented three control types (see Figure 3) in a simulated 3D

environment (see Section 4.3). This simulated environment is meant to act as a proxy for a potential Augmented Reality (AR) implementation. There, users would control an assistive robot arm and see the visual feedback superimposed on the real world and robot via an AR-HMD device. Instead, in our 3D simulation, users wear an *Oculus Quest* VR-HMD, which superimposes the visual feedback directly in the computed 3D scene. An overview of the environment and the control types described in the following sections is provided as a video (see Video S1).

All three control types use arrows as visual cues. Specifically, the arrows show which direction the gripper will move if a user engages the corresponding input DoF. To allow the users to predict the robot's movement when engaging the input DoF with positive values (e.g., pressing the control stick up) and negative values (e.g., pressing the control stick down), the arrows have two heads. Each arrowhead points towards the corresponding movement direction.

(a)   (b)   (c)
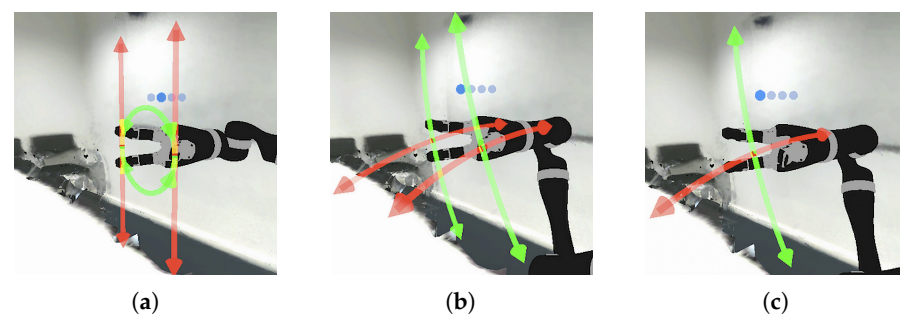
Figure 3. Visualization for the different control types: (**a**) Classic; (**b**) Double Arrow; (**c**) Single Arrow.

Using visual cues in 3D as opposed to 2D often causes visual obstruction, e.g., if the gripper is close to the table, the active DoF would lower the gripper towards the table. In that case, the arrows would clip through the table, making them partially invisible to the user. It would also be common that the robot's gripper itself obstructs parts of the arrows, making them harder to see and interpret. To eliminate these problems, the arrows were made translucent and are always rendered above all other objects yet shown at the correct depth as if looking through whatever is blocking them. This behavior is similar to viewing the scene through Augmented Reality glasses, which would overlay the arrows onto a real scene as opposed to showing the arrows as part of the real world that can be blocked by other real-world objects.

To more easily communicate the currently active mode, all control types show a blue indicator above the robot gripper consisting of four spheres, each representing a mode (see Figure 3). The sphere representing the currently active mode is darker and less translucent than the inactive ones, indicating how many modes are left to switch through before returning to the first.

3.1. Manually Designed DoF-Calculations

The focus of this study was to evaluate how adaptively changing DoF mappings would impact the participant's experience in a more complex 3D environment. While we proposed a CNN to perform these calculations in our previous work [6], there are other ways of calculating these DoF mappings. We developed a manually scripted method of calculating these DoF mappings for the specific task used in the study instead of training a CNN. This method generates a matrix with the same rules described in our previous work (see Figure 1) to represent DoF mapping, thus providing the possibility of equal movements as generated by a CNN trained on camera data. Since our primary focus is the participant's experience with the adaptively changing DoF mappings, we assumed that this approach would significantly decrease the possibility of unpredictable behavior while having little impact on the applicability of our findings to a system using a CNN. A detailed description

of the generated output values is presented in the description of the adaptive control types (see Sections 3.3 and 3.4).

This approach is akin to the widely used "Wizard of Oz" method, in which the output of a proposed system is instead provided by a human to test the user experience of that proposed system before finishing the implementation. In our case, we instead simulated the output of a complex CNN using a simpler system. As with "Wizard of Oz" experiments, our results should therefore be applied to the user experience with the system using a CNN, but the absolute performance measures may vary.

We developed three control types—*Classic*, *Double Arrow*, and *Single Arrow*—to function with different assistive robot arms and different input devices. To conduct the study, we decided to use the widely available stand-alone VR headset *Oculus Quest*. The *Oculus Quest* consists of the headset itself, and two motion controllers, one for each hand, with several buttons and a control stick each. Participants executed a simple pick-and-place task (see Section 4.6) in our VR environment using a virtual model of the *Kinova Jaco* robot arm using each of the control types (see Section 4.3 for a detailed description of the virtual environment and the VR setup).

3.2. Classic Control Type

The *Classic* control type implements the standard mode switch control type most commonly used to control assitive robot arms. This means that an input DoF always corresponds to a cardinal output DoF. Given the seven cardinal DoFs of the Jaco robot arm (X-Translation, Y-Translation, Z-Translation, Roll, Yaw, Pitch, Open/Close fingers) and two input DoFs (the X-Axis and Y-Axis on a motion controller's control stick) four modes are available to the users:

1. X-Translation + Y-Translation;
2. Z-Translation + Roll;
3. Yaw + Pitch;
4. Open/Close fingers + Nothing.

The last mode has no assigned output DoF for the X-Axis on the control stick to allow the users to learn an axis-to-action mapping.

Users can switch modes by pressing the A-Button on the right-hand motion controller. This allows them to perform the tasks at their own pace and assess the usefulness of a mode as long as they need to. Whenever the A-Button is pressed while the fourth mode is active, the first mode is selected again, allowing the users to cycle through modes at will.

Two arrows attached to the fingers of the gripper show the users which motion the gripper would perform, given a user's input on the respective input DoF. Red arrows represent the movement assigned to the Y-Axis of the control stick, and green arrows represent the movement assigned to the X-Axis of the control stick. As the motion controllers are also rendered in the virtual environment, we added a visual representation onto the control stick rendered in-game. A cross with one red axis and one green axis is shown on the motion controller to indicate which direction corresponds to which color. A blue sphere surrounds the A-Button to match it to the blue mode indicator (see Figure 4).
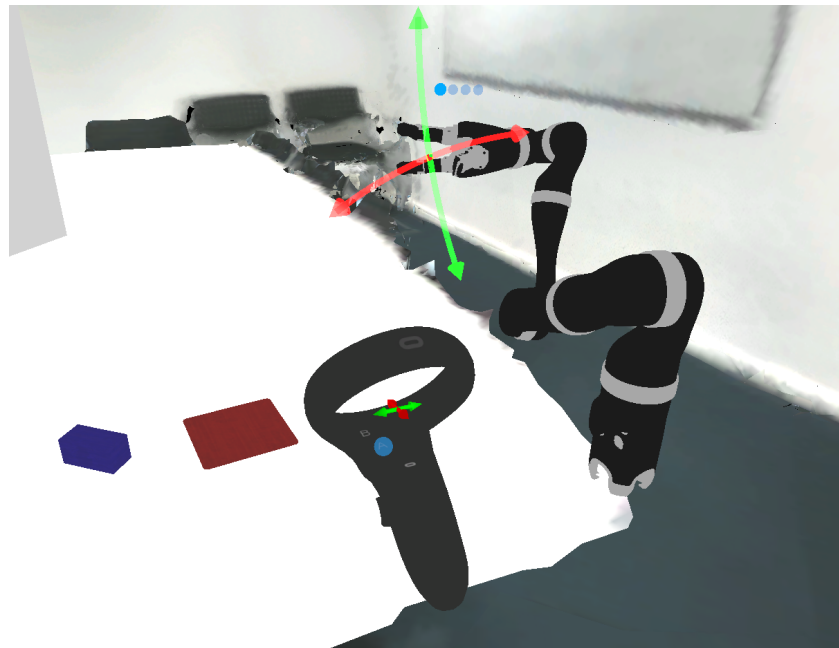
Figure 4. The virtual motion controller with directional indicators and the robot arm with matching arrows visualizing the currently selected mode.

3.3. Double Arrow Control Type

The *Double Arrow* control type implements the proposed adaptive control method using two arrows to show the position of the fingers if a user engages an input DoF. Therefore, each input DoF corresponds to a combination of cardinal DoFs determined based on the current situation. To ensure comparability with the *Classic* control type in regards to the number of mode switches necessary to return to the starting mode, four modes were developed. The modes are ordered by their complexity and usefulness to the users' goal of reaching the next target.

As in the *Classic* control type, two arrows attached to the fingers of the gripper show the users which motion the gripper would perform, given a user's input on the respective input DoF. Red arrows represent the movement assigned to the Y-Axis of the control stick, and green arrows represent the movement assigned to the X-Axis of the control stick.

The first mode assigns the Y-Axis of the control stick to a movement that both rotates and translates the gripper towards the next target simultaneously. More precisely, if the gripper is further than 10 cm away from the target, the movement is oriented towards a point 15 cm above the target. If the gripper is closer than 10 cm to the target, the movement is oriented towards the actual target. This ensures that the gripper tends to grasp and let go of objects from above, as opposed to trying to do so from the sides and thereby possibly crashing into the table. If the gripper is within reach of an object or target point where an object is supposed to be placed by the users, it also allows them to open and close the fingers. The X-Axis of the control stick in the first mode is assigned the same movement as the Y-Axis but rotated by 90° to allow for corrections perpendicular to the Y-Axis movement.

To provide users with more options, the second mode assigns the Y-Axis of the control stick to a linear translational movement towards the object and the X-Axis of the control stick to a rotational movement of the gripper towards the next target. Both of these assignments were chosen since only moving or only rotating are less likely to further the goal of the users. However, the individual movements themselves are still integral movements for coordinating the gripper orientation and some movement towards the goal. In the optimal case, this means that users would not need to use this mode, as both orientation and positioning would be taken care of simultaneously by the first mode.

The third mode assigns the Y-Axis of the control stick to the opening or closing the fingers, depending on whether an object was currently held or not. The X-Axis of the control stick has no assignment in this mode to ensure comparability with the *Classic* control type.

If users stop moving the gripper, they should always be able to move the same way they did before. To ensure this, the fourth mode always assigns the X- and Y-Axis of the control stick the same mappings that were last used to move the gripper. Otherwise, users who would want to assess if they had moved the robot far enough for their personal preference using a given mapping would have no possibility to correct their course.

The system calculates the next movement mappings whenever the users stop moving the robot. However, the system does not instantly assign the first mode to be active, as this would disrupt the users' flow of control (i.e., they might have stopped to asses the situation and then decided to continue with the DoF mapping they were using). Moreover, this would harm comparability to the *Classic* control type (as no automatic mode switches happen in that control type). This means that whenever the users stop moving, the blue mode indicator would show the fourth mode as being active, and a press on the A-Button would lead to the newly calculated first mode.

3.4. Control Type Single Arrow

During the development of *Classic* and *Double Arrow* we discovered that, while two arrows are a perfectly suitable visualization for a 2D environment, these arrows can result in a large amount of visual clutter during complex movement in 3D environments. We decided to develop a visualization that reduces visual clutter in a 3D environment and compare its usage to the *Double Arrow* control type.

Dubbed *Single Arrow*, the input-to-output DoF mappings are calculated in the exact same way as the mappings in *Double Arrow*. Switching between modes is also handled in the same way as in *Double Arrow*. However, the visualization changes from displaying two arrows at the tips of the fingers to displaying one arrow in the middle of the gripper, with a slight offset to allow certain movements to be displayed. This reduces visual clutter for all situations except when the fingers move.

4. Materials and Methods

We present a remote study with 39 participants to compare the proposed concept of adaptive control (in two variations) against the standard mode-switch control concept. In particular, we measured task completion times, the number of mode switches necessary to perform a task, the workload necessary to use the different control concepts via a NASA Raw-TLX (NASA Raw Task Load Index), and the participants' personal ranking of the three presented control types. Participants used their own *Oculus Quest* headset to perform a simple pick-and-place task using a virtual robot inside a realistic 3D environment.

4.1. Hypotheses

We propose the following hypotheses:

- Average Task Completion Time
    - *H1* Double Arrow *leads to lower task completion time than* Classic. The adaptive control of *Double Arrow* should significantly reduce the movements necessary to perform the task by combining different cardinal DoFs into one continuous movement, which otherwise would each have to be adjusted separately.
    - *H2* Single Arrow *leads to lower task completion time than* Double Arrow. Only using one arrow for each DoF mapping should reduce visual clutter. This should lead to a shorter processing time of the suggested movements, reducing the total time to execute a task.
- Average Number of Mode Switches

- *H3* Double Arrow *leads to fewer mode switches than* Classic. The adaptive control of *Double Arrow* should reduce the necessity to switch modes significantly. Since different DoFs are combined depending on the current situation, a change in position and rotation brings the robot arm closer to the target and can be performed without mode switches.
- *H4* Single Arrow *and* Double Arrow *need roughly an equal number of mode switches.* The behavior of the two adaptive control types is the same. Thus, while it might take participants longer to understand what movements they can perform with *Double Arrow* as opposed to *Single Arrow*, they should switch modes approximately as often in both control types.

- Workload
  - *H5* Double Arrow *leads to lower NASA TLX scores than* Classic. The adaptive control of *Double Arrow* calculates sensible movements to reach the next goal position and rotation. Thus, it should alleviate the participants from having to think of a sequence of movements to reach their goal, reducing workload. This is in contrast to the findings of our previous study, in which participants perceived the *Adaptive* control as more complex than the *Standard* control [6]. We expect the benefit of pre-calculated DoF combinations and the workload of developing a sequence of movements in cardinal DoFs to be higher in a 3D environment than in a 2D environment. Therefore, the workload for the adaptive control types should be lower than for *Classic* in 3D.
  - *H6* Single Arrow *leads to lower NASA TLX scores than* Double Arrow. Since we assume that reduced visual clutter leads to a shorter processing time for the suggested movements, the NASA TLX scores of *Single Arrow* should be lower.

4.2. Participants

In total, 39 people participated in our study (12 female, 26 male, 1 non-binary), which led to a data-set of 936 individual trials (8 per control type, 24 per participant). The age of participants ranged from ≤ 19 to 69, with 20 to 29 being the largest group with 22 participants. Four participants had prior experience with controlling an assistive robot arm, and no participants declared any motor impairments. All participants received EUR 10 as compensation unless they specifically denied the offer.

Due to the ongoing COVID-19 pandemic, we opted to perform a remote study using VR. We did not specifically search for participants with motor impairments because the potential target audience for people with VR setups at home that also have motor impairments appeared too small. There would not be enough time to gather enough participants in a realistic time frame. Instead, we searched for any participants that had access to the necessary equipment (an *Oculus Quest* headset, see Section 4.3) and were able to install our study software on their devices. With these non-specific participants, the performance measures for executing the tasks in our study with the different control types (see Section 4.6) can be compared relative to one another, even though they may not be representative of the intended target audience of such an assistive device. We acknowledge this limitation, which is further discussed in Section 7.

Participants were recruited via announcements in different social media communities relating to VR (e.g., r/OculusQuest: https://www.reddit.com/r/OculusQuest/, accessed on 3 January 2022), social media communities regarding assistive technologies (e.g., r/AssistiveTechnologies: https://www.reddit.com/r/AssistiveTechnology/, accessed on 3 January 2022), and platforms for acquiring participants specifically for XR studies (e.g., XRDRN: https://www.xrdrn.org/, accessed on 3 January 2022) among other more local announcements.

To ensure that VR sickness symptoms did not influence our results, the participants filled out the Virtual Reality Sickness Questionnaire (VRSQ) at the end of the study [27]. The VRSQ measures nine items on a four-point Likert scale and results in a value between 0 and 100, where 0 means no symptoms experienced and 100 means all symptoms were severe.

Reported values were low (Mean: 11.30, Std.-Dev.: 11.38), and none of the participants selected the "Severe" option for any of the items.

4.3. Apparatus

We designed a Virtual Reality environment based on a photogrammetry scan of a real room. The environment included a virtual model of the *Kinova Jaco* (Kinova Jaco robot arm: https://assistive.kinovarobotics.com/product/jaco-robotic-arm, accessed on 3 January 2022) robot arm attached to a table, a red target surface, a blue block, and two virtual screens—one for descriptions and questionnaires and one that would show example photos of the control types (see Figure 5). We decided to use a virtual model of a real robot arm (*Kinova Jaco*) to stay as close to a physical system as possible. Additionally, the *Kinova Jaco* robot arm is specifically designed and often used as an assistive device for people with motor impairments [5].

Figure 5. The virtual environment: description screen (**Left**); screen with example photos of the control types (not shown); *Kinova Jaco* with visualisation for control type *Single Arrow* (**Right**); table with blue block and red target (**Bottom**).

The virtual environment was created with the *Unreal Engine 4.26* and was developed to be deployed to the *Oculus Quest* VR headset. Participants had to either own or have access to such a headset and be able to install the study software on that headset using a computer (*Windows*, *macOS*, and *Linux* could be used). Although we tested our software on the original *Oculus Quest* hardware, we did not explicitly exclude the use of the newer and very similar *Oculus Quest 2* headset. The *Oculus Quest* consists of the VR headset and two motion controllers, one for each hand. Each motion controller has several buttons and a control stick. Participants controlled the robot using the right motion controller of the VR headset. In particular, the control stick of the motion controller moved the robot according to the currently active control type. This enabled the participants to control which DoFs were being used and how fast the robot would move. The A-Button was used to switch to the next mode cyclically, returning to the first mode when a mode switch was performed in the last mode.

To simulate the movement of the robot arm, the inputs did not move the joints of the robot as they would with a physical robot arm. Rather, the gripper of the virtual robot arm is moved in 3D space according to the inputs, and the arm of the robot is programmed to adopt a correct pose automatically. This was implemented using the physics system of the *Unreal Engine*.

4.4. Procedure

Participants were directed to a website with a brief introduction to the study, the duration of the study (around 30 to 45 min), the technical and non-technical prerequisites to participate in the study, and a description about what data would be collected during the study. Participants were informed that certain metrics and usage data, such as task completion times, will be recorded and sent to our servers during the study. They were also informed that they would need to fill out a short questionnaire after each condition of the study and that they would be able to record a short audio message after each condition. Lastly, participants were informed that cookies were being used on our website. Each participant gave informed consent by pressing a clearly labeled button to continue and start the study. After giving their consent, participants were instructed on how to install and open the study application and what to do when they were finished with the part of the study inside the VR headset. During the study, neither a video of the participants surroundings through the VR headsets external cameras and sensors nor a screen-recording was captured.

Next, the participants put on their VR headsets and opened our study application. They were greeted with a brief explanation of the study on a large virtual screen. Except for the questionnaires after each control type, any text that was available to read on that screen was also simultaneously read aloud as a prerecorded voice-over. The participants interacted with this screen via a common interaction method that was also used in the menus of the *Oculus Quest* headset: pointing a ray that originated from the motion controller towards the screen and using the trigger to confirm input.

After the study explanation, the participants were presented with a description of the first control type they would be using and the task they would be performing. This explanation was supplemented with an image on a second smaller virtual screen. The descriptions were written in a way that described how the gripper would move in relation to the current situation. We did not explicitly describe the intentions behind the different modes and their order in *Double Arrow* and *Single Arrow* (to provide ideally optimal mappings) to prevent possible biases. Otherwise, the participants might have been inclined to trust the adaptive mappings against their own judgment, thereby changing their behavior.

The explanation of each control type was followed by a series of trials of our pick-and-place task (see Section 4.6) the participants had to execute to progress through the study. For each control type, the task was performed once as a training trial and then eight more times for the same control type. During these eight trials, the task completion time and the number of mode switches performed was recorded.

After executing all trials for a control type, the participants were presented with the NASA Raw-TLX questionnaire to capture the participants' workload. Additionally, the participants could record a short audio message to point out additional things they felt were relevant during the execution of the trials. The recording of the audio message was optional. After filling out the questionnaire and optionally recording an audio message, the participants would continue with the next control type until they had executed all trials for all three control types.

Upon finishing the VR part of the study, participants received a unique code to be entered in a form on our website to complete the VRSQ [27] and our questionnaire. We asked the participants to report their demographic data and rank the control types presented in the VR section of the study. Lastly, participants left their contact information to receive the compensation.

4.5. Study Design

We used a within-subjects design with the control type as an independent variable with three levels: (1) *Classic*, (2) *Double Arrow*, and (3) *Single Arrow*. Each participant performed eight trials of a pick-and-place task for each of the three control types (see Section 4.6). Additionally, they performed one training trial for each control type to

familiarize themselves with the control type. The order of control types shown to the participants was fully balanced.

We measured three dependent variables for each control type: *Average Task Completion Time*, *Average Number of Mode Switches*, and *Workload via a NASA Raw-TLX questionnaire*.

Average Task Completion Time in seconds While participants executed each trial with the robot arm, the time to complete the task was measured for each participant. Then, the average task completion time for each control type was calculated across all participants.
Average Number of Mode Switches While participants executed each trial with the robot arm, each mode switch executed by pressing a button on the input device was counted and stored as the number of mode switches. Then, the average number of mode switches for each control type was calculated across all participants.
Workload via a NASA Raw-TLX questionnaire After completing all trials within each control type, the participants were asked to fill out a NASA Raw-TLX questionnaire to obtain information about the participants' perceived workload. The questionnaire consists of the following six criteria, which participants would rate on a scale of 0 to 100 in steps of 5: mental demand, physical demand, temporal demand, performance, effort, and frustration [28].

In addition, the participants could record a short description of their experiences in the form of a voice message, although this was not mandatory. The recorded voice messages were transcribed and analyzed by multiple researchers to identify underlying themes and common impressions the participants had while using the virtual robot arm (see Section 5.2). Participants also provided a personal ranking of the three control types in a questionnaire at the end of the study.

4.6. Task

Participants were asked to repeatedly place a blue block onto a red target using the assistive robot. Participants performed this task eight times per control type. We did not use two blocks per trial to reduce variability in our results. We decided to use a simple pick-and-place task instead of a specific ADL (e.g., drinking from a glass) since pick-and-place tasks are part of many ADLs. Moreover, a specific ADL might have caused problems with participants' preconceived notions of that task (e.g., they would approach the glass in a particular way, while the adaptive system would approach it differently). This would have possibly distracted them from evaluating the control types as a whole, which we wanted to avoid.

In each of the eight trials per control type, the position of the blue block changed to one of eight predefined positions around the red target surface. The order in which the positions were used in the eight trials was randomized for each participant and control type.

5. Results

We recorded both quantitative and qualitative data from the participants during the trials. This section presents the results of each section from our data analysis.

5.1. Quantitative Results

The recorded quantitative data for each trial included *task completion time* (in seconds) and *the number of mode switches*. For each control type, the quantitative data included the *NASA Raw-TLX* results and the *Rank* given to the control type by the participants (lower rank numbers are better). The used abbreviations and symbols are:

- IQR: Interquartile Range;
- SD: Standard Deviation;
- SE: Standard Error;
- p: *p*-value as an expression of the level of statistical significance;
- N: Sample Size;
- $\chi^2(2)$: Chi-Squared with two degrees of freedom;

- F: F-Statistic for the Repeated-Measures ANOVA;
- M: Mean;
- df: Degrees of Freedom for the calculation of χ^2 for the Friedman Tests.

5.1.1. Task Completion Time

For each participant, we averaged the task completion times (see Table 1) of the trials for each control type. In an exploratory analysis, we removed outliers that had average task completion times $\geq 2.2 * \text{IQR}$ of the mean task completion time in at least one control type [29] (see Figure 6). Four outliers were excluded this way, leaving 35 participants for analysis of task completion times. An inspection of QQ-plots found the resulting data-set to follow a normal distribution.

Table 1. Statistics for average task completion times (in seconds, N = 35).

|	**Classic**	**Double Arrow**	**Single Arrow**
Mean	47.41	42.62	44.04
Median	44.66	37.75	41.23
Std.-Dev.	12.55	19.28	22.24
IQR	14.03	24.33	31.68
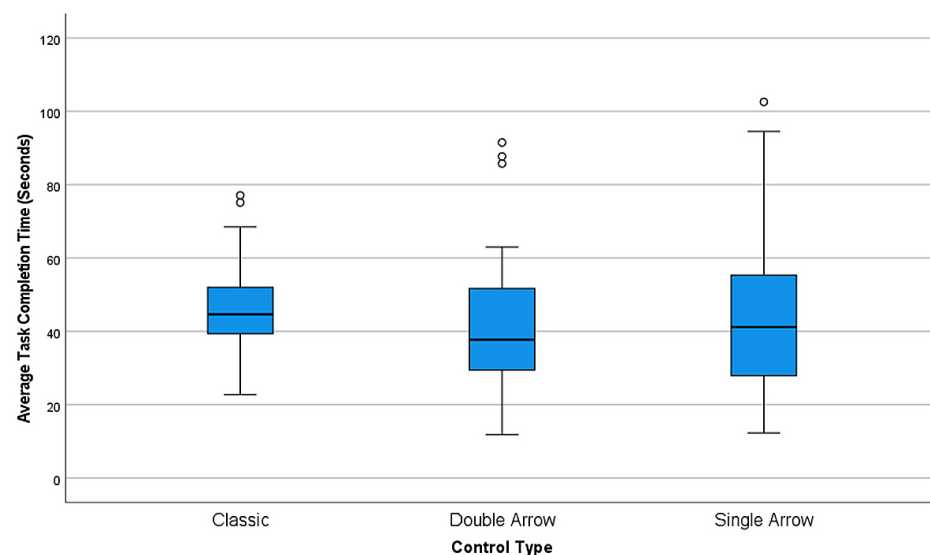
Figure 6. Boxplots for average task completion times.

To determine whether the control types had an effect on average task completion times, we performed a Repeated-Measures ANOVA (RM-ANOVA). However, we found no significant main effect ($F(2, 64) = 1.31$, $p = 0.28$).

In addition to the effect of control types, we examined whether the starting condition of a participant had an impact on task completion times. We included the starting condition as a between-subjects factor for the RM-ANOVA and discovered a significant interaction effect between the starting condition and the task completion times ($F(4, 64) = 8.86$, $p < 0.001$). Analyzing simple main effects, we discovered that the task completion times for *Classic* stayed roughly the same regardless of the starting condition. However, both adaptive control types heavily suffered when they were the starting condition (see Figure 7). A post hoc pairwise comparison (Estimated Marginal Means, Bonferroni adjusted) showed that task completion times for *Single Arrow* (M = 54.66 s, SE = 5.9) were significantly longer than those for *Double Arrow* (M = 33.74 s, SE = 4.6) if *Single Arrow* was the starting condition ($p = 0.001$). Conversely, task completion times for *Double Arrow* (M = 57.89 s, SE = 5) were significantly longer than those for *Single Arrow* (M = 37.82 s, SE = 6.41) if *Double Arrow* was the starting condition instead ($p = 0.002$). Another significant difference was found if *Single*

*Arrow* was the starting condition: *Classic* task completion times (M = 48.23 s, SE = 3.57) were longer than those of *Double Arrow* (M = 33.74 s, SE = 4.6) in that case (*p* = 0.013). The other comparisons yielded insignificant results.
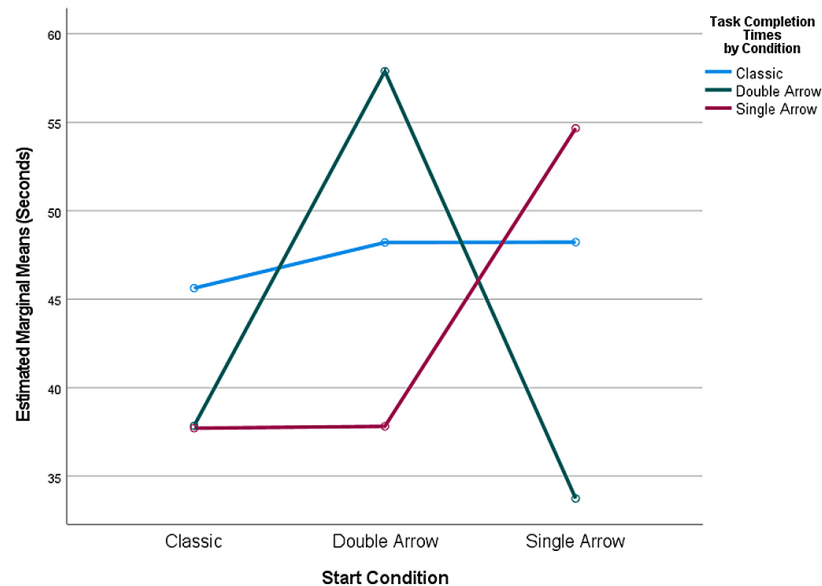
Figure 7. Estimated Marginal Means for average task completion times.

5.1.2. Mode Switches

To determine whether there were differences between the average number of mode switches between control types we used an RM-ANOVA. Due to a software error, mode switch data were only recorded correctly for 20 participants. We found a significant effect of control types on the average number of mode switches ($F_{(2, 38)}$ = 8.08, *p* = 0.001). Pairwise comparisons revealed that there were significant differences (*p* < 0.05) between the average number of mode switches for both adaptive control methods (*Double Arrow*: M = 12.93, SD = 3.91; *Single Arrow*: M = 14.23, SD = 5.15) and the *Classic* control method (M = 17.87, SD = 4.8). We found no significant difference between the average number of mode switches for *Single Arrow* compared to *Double Arrow* (*p* = 0.11, see Table 2 and Figure 8).
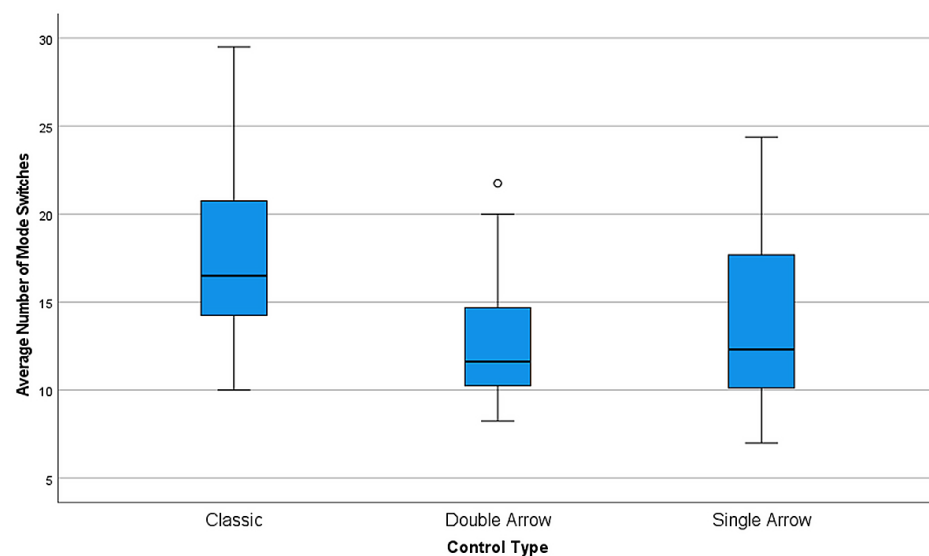
Figure 8. Boxplots for average number of mode switches.

Table 2. Statistics for average number of mode switches (N = 20).

	Classic	Double Arrow	Single Arrow
Mean	17.87	12.93	14.23
Median	16.50	11.63	12.31
Std.-Dev.	4.80	3.91	5.15
IQR	7.00	5.09	7.91

5.1.3. Workload and Rank

Each participant completed a NASA Raw-TLX questionnaire after completing the task with each control type, rating each dimension on a scale from 1 to 100. To evaluate whether there were any differences between the control types regarding workload, Friedman Tests were performed for both the overall NASA TLX value as well as the individual dimensions of the questionnaire. No significant differences were found for either the overall NASA TLX value ($\chi^2(2)$ = 5.33, p = 0.07) or the individual dimensions (see Table 3).

We also evaluated whether the users preferred one control type over the others. To do so, the participants ranked the control types after completing all tasks. A lower number means the participant ranked that control type higher. No significant differences were found for the ranks ($\chi^2(2)$ = 0.97, p = 0.65) (see Table 4).

Table 3. Statistics for individual NASA TLX Dimensions on a scale from 1 to 100 (df = 2, N = 39 for all Friedman Tests).

	Mental Demand	Physical Demand	Temporal Demand	Performance	Effort	Frustration
Classic (Mean)	53.33	30.26	36.92	32.05	48.59	41.41
Classic (Std.-Dev.)	24.64	21.67	21.07	20.48	24.84	24.52
Double Arrow (Mean)	56.28	28.21	40.38	38.97	52.82	43.08
Double Arrow (Std.-Dev.)	22.93	16.20	25.06	25.50	24.08	26.40
Single Arrow (Mean)	48.97	27.56	36.03	40.64	51.41	38.33
Single Arrow (Std.-Dev.)	24.69	22.94	20.56	26.61	23.25	26.34
Mean Ranks						
Classic	2.04	1.92	1.96	1.73	1.79	1.92
Double Arrow	2.21	2.17	2.18	2.15	2.18	2.17
Single Arrow	1.76	1.91	1.86	2.12	2.03	1.91
Friedman Tests						
χ^2	4.23	2.07	2.38	4.86	3.15	1.76
Exact Significance	0.12	0.37	0.31	0.09	0.21	0.43

5.2. Qualitative Results

Participants were asked to describe their experience with the control type they used in a voice message. They were asked to elaborate on the ease of controlling the robot, their understanding of movement directions, and the predictability of the next movement directions.

In total, 23 of the 39 participants recorded a message for all three control types. In addition, only four participants recorded voice messages for two of the three control types, and one participant just recorded a single voice message. This resulted in 26 voice messages for each control type.

Table 4. Statistics for NASA TLX on a scale from 1 to 100 and ranking on a scale from 1 to 3 (df = 2, N = 39 for all Friedman Tests).

	NASA TLX	Rank
Classic (Mean)	40.43	1.87
Classic (Std.-Dev.)	17.11	0.77
Double Arrow (Mean)	43.29	2.05
Double Arrow (Std.-Dev.)	15.32	0.86
Single Arrow (Mean)	40.49	2.08
Single Arrow (Std.-Dev.)	17.29	0.84
Mean Ranks		
Classic	1.85	1.87
Double Arrow	2.29	2.05
Single Arrow	1.86	2.08
Friedman Tests		
χ^2	5.33	0.97
Exact Significance	0.07	0.65

5.2.1. Thematic Analysis

The voice recordings were analyzed with the Thematic Analysis method described by Braun and Clarke [30]. This method was chosen because it has the flexibility to identify themes within the unstructured feedback from the recorded voice messages. Throughout the analysis, we identified themes related to our hypotheses, which gave us a better insight into how participants perceived their experience and success in executing the given tasks.

First, we transcribed the voice messages to be able to analyze them. Although most participants recorded their messages in English, a few recorded them in German. Some of the statements in the following chapters were therefore translated into English. Second, two of our researchers performed the Thematic Analysis using the six-phase method described by Braun and Clarke [30]. Each researcher read each transcribed voice message to become familiar with the participant's feedback. They then marked certain paragraphs and phrases to identify underlying topics related to our hypotheses that were relevant within multiple data-sets. Each marked phrase was assigned a short code describing its topic. We used the software *Obsidian* (Obsidian markdown note-taking software: https://obsidian.md, accessed on 3 January 2022) for managing and tagging the transcribed messages in a simple markdown text format with links and tags. Third, codes were organized and grouped into themes, and descriptive titles were assigned to each theme. For a visual representation, we developed visual thematic graphs; one of which is shown in Figure 9. Although some comments were related to several themes, we decided to sort them into the theme with the best fit. Fourth, themes were revised and evaluated by reading the related phrases and codes again to ensure that each theme was internally homogeneous. Fifth, both researchers worked together to refine the themes and compile them into a single thematic map presented in Figure 10. Sixth, a summary of the results was written based on the final thematic map.
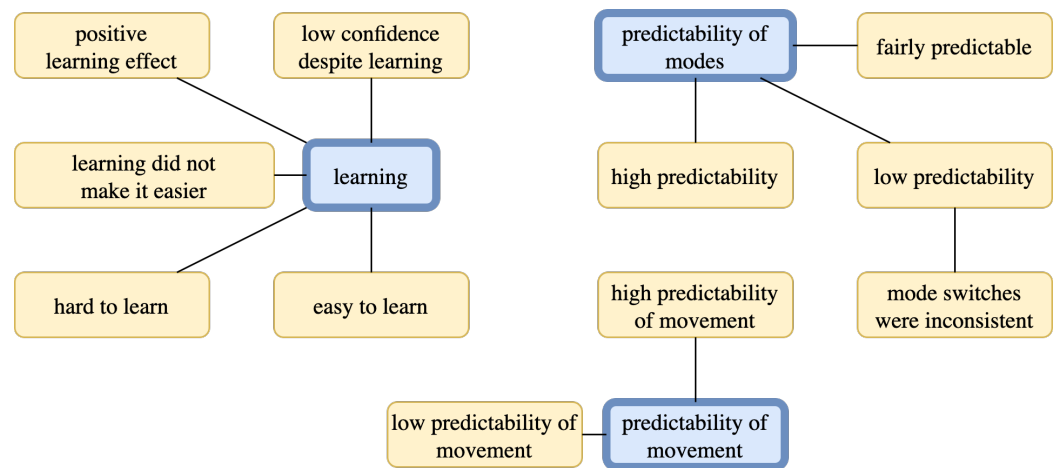
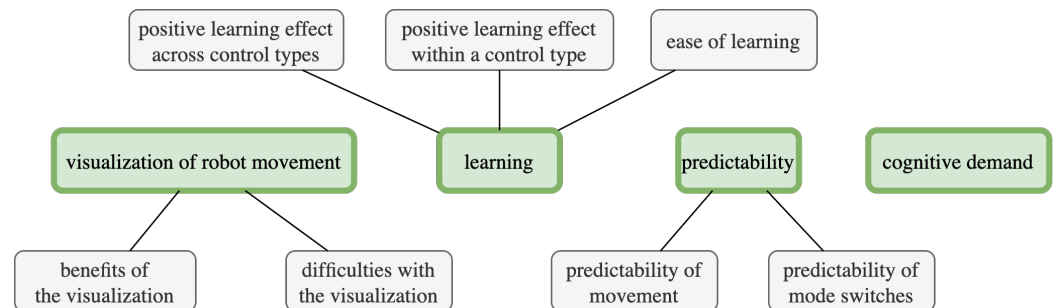**Figure 9.** Early thematic map with codes shown in yellow and themes shown in blue.

Figure 10. Final thematic map with themes shown in green and sub-themes shown in gray.

5.2.2. Results of the Thematic Analysis

We identified the following themes in the combined thematic map: *visualization of robot movement*, *cognitive demand*, *predictability of mode switching*, *predictability of movement* and *learning*. The excerpts from one participant's audio messages were marked with the participant's unique number (e.g., *P26* for the 26th participant out of the total 39 participants). Since participants often referenced the previous control types they used, we also added which control type they were referring to in brackets when citing them.

Visualization of robot movement: This theme comprises the difficulties and benefits of the visualization of the robot's movement. As expected when transferring over a visualization from a 2D environment to 3D, perspective was one source of errors across all three control types. *P5* stated, "Depending on the orientation of the robot arm, I could not see exactly which way the arrows were going." P4 added, "Sometimes moving the robot was a bit difficult because it just did not feel natural from different perspectives."

Regarding the control types *Double Arrow* and *Single Arrow*, many participants mentioned that the arrows are either hard to interpret or hard to see. Interestingly, the participants did not mention this problem with the *Classic* control type. Participants stated, "[...] the arrows that follow the change of the movement direction are a little more difficult and a little bit less intuitive to understand than the previous trial [control type *Classic*]" (*P9*), and "I think it is more difficult than the previous control type because it has more abstract movement [...]" (*P25*). Besides the curved arrows, many participants found it difficult to associate the differently colored arrows of the visualization with the different input DoFs across all three control types. *P31* made this clear after using the *Single Arrow* control type. They said, "The hardest part working with this method of motion was determining which direction pushing the analog stick would actually move the robot."

Across both adaptive control types, participants mentioned the helpfulness of the arrows. *P25* commented, "I think it was confusing at first, but those red and green arrows helped a lot to understand how the robot moved." After using the *Double Arrow* control

type, *P8* mentioned, "Controlling the robot was better than before [control type *Single Arrow*], because one could tell more easily where the arm would go, based on the multiple arrows". This suggests the possible benefits of having multiple arrows in the *Double Arrow* control type.

Cognitive demand: In this theme, we consolidated statements that describe a higher or lower cognitive demand while using a specific control type. Across all three control types, some participants mentioned a high cognitive demand. After using the *Single Arrow* control type, *P17* stated, "This one was more cognitively demanding than the previous one [control type *Classic*], maybe because this one did not have straight movement but a lot of rotational movements". *P18* found it to be "a bit confusing, but okay." Participants described the *Classic* control type as "confusing" (*P8*) and "counter intuitive" (*P18*). Using the *Double Arrow* control type, *P21* expressed the need to focus on the task and added, "I do not think you could do anything else while using this control method".

While mentions of lower cognitive demand were equally frequent in total, many participants found the *Classic* control type to be "easy" or "easy to understand" (*P6*, *P9*, *P25*, *P27*, among others). After using the *Classic* control type, *P39* added, "Here it was best to intuitively remember where each function was". This suggests a connection with the next two themes regarding predictability.

Predictability of mode switching: This theme describes the ability of the participants to anticipate the next set of movement combinations that the system provides when the participant executes a mode switch. Many of the difficulties participants had with the predictability were with the adaptive control types *Single Arrow* and *Double Arrow*. When using the *Double Arrow* control type, *P17* noted, "In this condition, I was not sure whether cycling through the different types of movements in there always were consistent. That was very confusing." We also identified this statement as an expression of an increase in cognitive demand. For the same control type, *P21* added, "I did not know which combination would be next when I pressed A". Using the *Double Arrow* control type, *P23* mentioned, "I could not predict the next movement, because I did not understand in which order the different movements are shown to me next." We think this participant confused the ever-changing nature of the adaptive suggestions with the different modes. Only a few participants mentioned difficulties with predicting the next mode in the *Classic* control type. *P37* said, "Predictability was uncertain as well, until the later moves where I had enough training to do it effectively." Additionally, many participants mentioned that they had to switch modes many times to find the proper movement they needed in a given situation, especially with the adaptive control types. Using the *Double Arrow* control type, *P3* stated, "So if I wanted it to go down I would have to switch through multiple modes [...]". Furthermore, using the *Double Arrow* control type, *P5* mentioned, "I had to click through many modes to find the movement that I thought would bring me closer to the block".

Mentions of good predictability were also spread across all three control types, although these were less common. For the *Classic* control type, *P39* stated, "It was very easy to understand and especially the predictability was the easiest here". Using the *Single Arrow* control type, *P37* mentioned, "The ease of understanding the movement was a lot easier as well. With some of the movement directions being easier to understand and predict before they show up." After executing the tasks with the *Double Arrow* control type, *P37* added, "It seemed more predictable and overall, a more optimum way of doing things".

Predictability of movement: In contrast to the previous theme, this theme is about predicting how and where the robot arm will move when using the currently selected mode. As visualization plays a big part when predicting the robot's movement, this theme is related to the first theme about visualization. Only a few participants mentioned the predictability of movement directly. After using the *Double Arrow* control type, *P4* said, "So I tried to do one thing and it would do a completely other thing. It felt really unnatural to try and get to the cube and even to pick it up". For the *Classic* control type, *P10* stated, "Because of the immediate predictability [...], it was much easier to control the robot and

to steer it into different vectors to approach the block in the different positions". Using the *Single Arrow* control type, *P10* added, "Therefore I could understand very well how it would move and how it would work out so I could reach the target".

Learning: This theme describes the participants' impression of their learning experience while using the different control types. Across all three control types, participants reported that they grew better at performing the tasks over time. For the *Classic* control type, *P26* stated, "Using this robot arm is pretty easy if you learn how to use them, [...]". After using the *Double Arrow* control type, *P25* mentioned, "The predictability of the next movement directions, I think, is easier as you practice with it, [...]". For the *Single Arrow* control type, *P39* said, "The more I practiced, the more confidence I got [...]".

As participants used the different control types, they noticed a learning effect even across the different control types. After finishing all trials of all control types, ending with the *Double Arrow* control type, *P25* said, "The predictability of the next movement directions, I think, is easier as you practice with it, [...]" After using the *Single Arrow* control type, *P33* stated, "Maybe I simply have more experience now, if I performed better in this task in any way".

Even though many participants felt that they needed more practice with the tasks so that they are easier to perform, some described that the process of learning felt relatively easy. When finishing the tasks with the *Classic* control type, *P16* stated, "It was quicker to get familiar with the system." *P33* expressed some difficulties with the *Double Arrow* control type but added, "At least it did not take long to notice a learning effect".

Additionally, we identified many instances where participants reported that they liked the second adaptive control type they used better than the one before, regardless of which control type came first and which came second. This also suggests that a learning effect is taking place. After using the *Double Arrow* and then the *Single Arrow* control types, *P27* stated, "I don't know what is the difference between double arrow and single arrow, but single arrow is much easier to control". For the *Double Arrow* control type, *P31* stated, "This method is a little bit easier to use than the second method [*Single Arrow* control type], but I think that was more a function of having a little bit more experience".

6. Discussion

Initially, our assumptions were that the overall task performance would be best when using the *Single Arrow* control type, followed by *Double Arrow*, and *Classic* would have the worst task performance. In comparison to the results of our previous study [6], the new results are not as pronounced in a realistic virtual 3D setting, at least not without considering the learning effects.

Regarding the task completion times, both Hypothesis 1 and Hypothesis 2 could not be substantiated. However, the interaction effect between the starting condition and task completion times suggests that, with time to learn, the adaptive control types could perform better than the *Classic* type. This is corroborated by participants' reports, as many participants said that their performance and understanding of the adaptive control types improved during the tasks. It is also worth noting that more participants experienced the second adaptive control type as "better" than the first, implying a learning effect not only for one control type but between control types.

Regarding mode switches, Hypothesis 3 and Hypothesis 4 could be substantiated by our results. From *Classic* to *Double Arrow*, we measured a significant reduction in the number of mode switches necessary to perform the task. In contrast, there was no significant difference between *Double Arrow* and *Single Arrow*. Interestingly, this contrasts the participants' opinions that they felt they had to switch many times to get to a mode that performed a movement they expected. However, this reduction in mode switches might be of higher benefit for people with motor impairments than for non-disabled people. Switching modes using a button requires a certain level of dexterity and causes the user to constantly divert their attention away from the original task, so more mode switches can cause more fatigue and time consumption, as explained by Herlant et al. [5]. The impact

of this difference in the number of mode switches on people with motor impairments can thus only be evaluated in a future study with participants with motor impairments.

Regarding workload, Hypothesis 5 and Hypothesis 6 could not be substantiated. This could have multiple reasons. For example, the participants expressed that the predictability of the adaptive control types was low and that they did not necessarily know how the robot would move, even with the arrows. These impressions, combined with the statements regarding positive learning effects and overall high cognitive demand, could mean that with increased exposure to the adaptive control types, users could have a lower workload than with *Classic*.

According to some participants, using visual cues in a 3D environment caused problems with perspective. This made it difficult for them to predict how the robot would move, even with the visual cues provided by the arrows. To mitigate this problem, our concept might be combined with a "digital twin" of the robot arm, which demonstrates the movement virtually before the real robot performs it physically [31].

To improve the overall predictability of the system, both regarding the suggested modes and the movements of the robot, a training mode could be implemented. In this mode, the users would be able to teach the system the way they want specific tasks to be performed [32]. This should increase predictability, as the participants would know the proposed movements will be (partially) based on their own instructions. In addition, Spatial Augmented Reality can help the user's understanding of the robot's perception, e.g., which object the robot assumes the user wants to interact with [33]. In combination with the already implemented visual cues, this can help the users predict the robot's movement more accurately.

After further research and refinement of our proposed control methods, they might allow assistive robot arms to help with ADLs that currently require the help of caregivers or more complex robots, such as dressing [34] or bathing [35]. The fact that the users always stay in control of the robot while the robot performs more fluent, natural movements could also allow people with motor impairments to use the robot in social situations, e.g., at the workplace [36].

7. Limitations

Our study did not specifically involve or focus on people with motor impairments. Thus, we need to discuss how our results can be transferred to this target group. First, the absolute performance measures cannot be generalized to this target group. Individual differences are usually high within people with motor impairments due to varying degrees of physical limitations [37]. However, the study did not aim to provide absolute results in terms of performance but rather an insight into the relative performance of the three different control types. Since they all rely on the same physical interaction concept, we believe that the way motor impairments might affect performance should be comparable for all three control types. Second, Augmented Reality is necessary to provide the user with the type of visual feedback we implemented for our study. We are aware from our prior research that current-generation AR-HMDs are often not accessible to people with motor impairments. AR-HMDs such as the *Microsoft HoloLens* are too heavy and conflict too often with health-supporting systems [38]. We conducted this research with the firm belief that future AR hardware solutions will cope with requirements for people with motor impairments. We acknowledge, however, that this might make the visual feedback designs inapplicable for real-world systems at this point in time or the immediate future.

Additionally, our study involved the use of the *Oculus Quest* system and the *Oculus Quest* Motion Controller as the only input device. In the real world, however, assistive robot arms can be controlled with a wide range of input devices depending on the abilities and preferences of the person using them. We specifically only used the most basic functionality of the Motion Controller (the control stick and one button) to ensure that the results are also applicable when using a different input devices with two input axes. It is still possible

that the use of different input devices might add more complexity to the overall usage of such a system.

Another limitation is the nature of our study being performed as a remote study. The level of control is limited for such a method, which means that the level of engagement of participants can vary. We addressed this limitation by keeping the duration of the study relatively short (30–45 min) and designing the task so that we could easily identify cases in which participants did not follow the study protocol. Our analysis further shows that only a few participants were identified as extreme outliers. In addition, the focus on one set of hardware devices made it possible to harmonize and control the kind of immersive experience that participants engaged with, further reducing potential biasing effects, such as low frame rates or other hardware-performance-related issues. Given the current COVID-19 pandemic, we believe that our study setup is sensible and still able to provide robust results. Still, we aim to replicate at least part of the study in a lab environment and with people with motor impairments in the future.

It is possible that our study does not provide insight into the quality of adaptive control through the means of a CNN. We simulated the adaptive control method to be able to have full control in the study. Otherwise, imperfect DoF mappings would have overshadowed the potential effects of the different visualizations, thus making it difficult to draw conclusions. As discussed, we believe that our approach significantly decreases the possibility of unpredictable behavior while having little impact on the applicability of our findings to a system using a CNN, as long as this CNN is able to perform at a high level of quality regarding the DoF mappings.

8. Conclusions

We conducted a study exploring and evaluating the user experience of an adaptive control concept for assistive robot arms in a realistic virtual 3D environment. Our results suggest a significant benefit of such an adaptive control concept regarding the necessary number of mode switches. However, task completion times and workload do not change when using an adaptive control concept without more intensive training.

By evaluating the interaction between the starting conditions and task completion times and applying a thematic analysis of qualitative data, we conclude that there could be a significant benefit of training that would reveal the potential of an adaptive control concept. Therefore, future work should consider longer training sessions before evaluating task completion times and workload. The targeted user group of assistive robot arms would use such devices not just once but daily and over extended periods and thus have more time to learn how to use the device. Therefore it is important to assess whether the adaptive control concept might have high cognitive demand in the beginning but is better than the *Classic* approach once the users are trained.

Our results seem to suggest that there is little to no difference between *Single Arrow* and *Double Arrow* regarding how well they convey the robots currently active DoF mapping to the users. However, an improved visualization could reduce the overall high cognitive demand users have experienced. Therefore, future work will also focus on different types of visualizations, which will not be restricted to MR-headsets and overlayed arrows but could (additionally) show the robot's future path using spatial Augmented Reality [39].

Future work should (whenever possible) include participants with motor impairments since their experience is vital in designing assistive technology [4]. The impact of a lower number of mode switches enabled by an adaptive control concept should be especially evaluated with people with motor impairments. This could significantly improve their execution of activities of daily living.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/technologies10010030/s1. Video S1: An Overview of the Environment and Control Types.

References

1. Bauer, A.; Wollherr, D.; Buss, M. Human—Robot Collaboration: A Survey. *Int. J. Hum. Robot.* **2008**, *5*, 47–66. [CrossRef]
2. Fong, T.; Nourbakhsh, I.; Dautenhahn, K. A survey of socially interactive robots. *Robot. Auton. Syst.* **2003**, *42*, 143–166.
3. Statistisches Bundesamt (Destatis). Pflegestatistik—Pflege im Rahmen der Pflegeversicherung—Deutschlandergebnisse—2019. 2020. Available online: https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Gesundheit/Pflege/Publikationen/Downloads-Pflege/pflege-deutschlandergebnisse-5224001199004.pdf (accessed on 3 January 2022).
4. Pascher, M.; Baumeister, A.; Schneegass, S.; Klein, B.; Gerken, J. Recommendations for the Development of a Robotic Drinking and Eating Aid—An Ethnographic Study. In Proceedings of the Human-Computer Interaction—INTERACT 2021, Bari, Italy, 30 August–3 September 2021; Ardito, C., Lanzilotti, R., Malizia, A., Petrie, H., Piccinno, A., Desolda, G., Inkpen, K., Eds.; Springer: Cham, Switzerland, 2021. [CrossRef]
5. Herlant, L.V.; Holladay, R.M.; Srinivasa, S.S. Assistive Teleoperation of Robot Arms via Automatic Time-Optimal Mode Switching. In Proceedings of the Eleventh ACM/IEEE International Conference on Human Robot Interaction, Christchurch, New Zealand, 7–10 March 2016; pp. 35–42.
6. Goldau, F.F.; Frese, U. Learning to Map Degrees of Freedom for Assistive User Control: Towards an Adaptive DoF-Mapping Control for Assistive Robots. In Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference, Corfu, Greece, 29 June–2 July 2021; Association for Computing Machinery: New York, NY, USA, 2021; PETRA 2021, pp. 132–139. [CrossRef]
7. Gräser, A.; Heyer, T.; Fotoohi, L.; Lange, U.; Kampe, H.; Enjarini, B.; Heyer, S.; Fragkopoulos, C.; Ristic-Durrant, D. A Supportive FRIEND at Work: Robotic Workplace Assistance for the Disabled. *IEEE Robot. Autom. Mag.* **2013**, *20*, 148–159. [CrossRef]
8. Foresi, G.; Freddi, A.; Monteriù, A.; Ortenzi, D.; Pagnotta, D.P. Improving mobility and autonomy of disabled users via cooperation of assistive robots. In Proceedings of the 2018 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 2–14 January 2018; pp. 1–2. [CrossRef]
9. Tsui, K.M.; Kim, D.J.; Behal, A.; Kontak, D.; Yanco, H.A. "I want that": Human-in-the-loop control of a wheelchair-mounted robotic arm. *Appl. Bionics Biomech.* **2011**, *8*, 127–147. [CrossRef]
10. Goldau, F.F.; Shastha, T.K.; Kyrarini, M.; Gräser, A. Autonomous multi-sensory robotic assistant for a drinking task. In Proceedings of the 2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR), Toronto, ON, Canada, 24–28 June 2019; pp. 210–216.
11. Tanaka, K.; Mu, S.; Nakashima, S. Meal-assistance robot using ultrasonic motor with eye interface. *Int. J. Autom. Technol.* **2014**, *8*, 186–192. [CrossRef]
12. McColl, D.; Louie, W.Y.G.; Nejat, G. Brian 2.1: A socially assistive robot for the elderly and cognitively impaired. *IEEE Robot. Autom. Mag.* **2013**, *20*, 74–83. [CrossRef]
13. Guerreiro, J.; Sato, D.; Asakawa, S.; Dong, H.; Kitani, K.M.; Asakawa, C. Cabot: Designing and evaluating an autonomous navigation robot for blind people. In Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility, Pittsburgh, PA, USA, 28–30 October 2019; pp. 68–82.
14. Maheu, V.; Archambault, P.S.; Frappier, J.; Routhier, F. Evaluation of the JACO robotic arm: Clinico-economic study for powered wheelchair users with upper-extremity disabilities. In Proceedings of the 2011 IEEE International Conference on Rehabilitation Robotics, Zurich, Switzerland, 29 June–1 July 2011; pp. 1–5.
15. Hashimoto, K.; Saito, F.; Yamamoto, T.; Ikeda, K. A field study of the human support robot in the home environment. In Proceedings of the 2013 IEEE Workshop on Advanced Robotics and its Social Impacts, Tokyo, Japan, 7–9 November 2013; pp. 143–150.
16. Fattal, C.; Leynaert, V.; Laffont, I.; Baillet, A.; Enjalbert, M.; Leroux, C. Sam, an assistive robotic device dedicated to helping persons with quadriplegia: Usability study. *Int. J. Soc. Robot.* **2019**, *11*, 89–103. [CrossRef]

17. Kyrarini, M.; Lygerakis, F.; Rajavenkatanarayanan, A.; Sevastopoulos, C.; Nambiappan, H.R.; Chaitanya, K.K.; Babu, A.R.; Mathew, J.; Makedon, F. A survey of robots in healthcare. *Technologies* **2021**, *9*, 8. [CrossRef]
18. Canal, G.; Alenyà, G.; Torras, C. Personalization framework for adaptive robotic feeding assistance. In Proceedings of the International Conference on Social Robotics, Kansas City, MO, USA, 1–3 November 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 22–31.
19. Drolshagen, S.; Pfingsthorn, M.; Gliesche, P.; Hein, A. Acceptance of industrial collaborative robots by people with disabilities in sheltered workshops. *Front. Robot. AI* **2021**, *7*, 541741. [CrossRef] [PubMed]
20. Pascher, M.; Baumeister, A.; Klein, B.; Schneegass, S.; Gerken, J. Little Helper: A Multi-Robot System in Home Health Care Environments. In Proceedings of the 2019 International Workshop on Human-Drone Interaction (iHDI) as Part of the ACM Conference on Human Factors in Computing Systems, Glasgow, UK, 5 May 2019.
21. Pollak, A.; Paliga, M.; Pulopulos, M.M.; Kozusznik, B.; Kozusznik, M.W. Stress in manual and autonomous modes of collaboration with a cobot. *Comput. Hum. Behav.* **2020**, *112*, 106469. [CrossRef]
22. Kim, D.J.; Hazlett-Knudsen, R.; Culver-Godfrey, H.; Rucks, G.; Cunningham, T.; Portee, D.; Bricout, J.; Wang, Z.; Behal, A. How Autonomy Impacts Performance and Satisfaction: Results From a Study With Spinal Cord Injured Subjects Using an Assistive Robot. *IEEE Trans. Syst. Man Cybern.—Part A Syst. Hum.* **2012**, *42*, 2–14. [CrossRef]
23. Cleaver, A.; Tang, D.V.; Chen, V.; Short, E.S.; Sinapov, J. Dynamic Path Visualization for Human-Robot Collaboration. In Proceedings of the Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction (HRI '21 Companion), Boulder, CO, USA, 8–11 March 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 339–343. [CrossRef]
24. Walker, M.; Hedayati, H.; Lee, J.; Szafir, D. Communicating Robot Motion Intent with Augmented Reality. In Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction (HRI '18), Chicago, IL, USA, 5–8 March 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 316–324. [CrossRef]
25. Shrestha, M.C.; Kobayashi, A.; Onishi, T.; Uno, E.; Yanagawa, H.; Yokoyama, Y.; Kamezaki, M.; Schmitz, A.; Sugano, S. Intent communication in navigation through the use of light and screen indicators. In Proceedings of the 2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Christchurch, New Zealand, 7–10 March 2016; pp. 523–524. [CrossRef]
26. Shindev, I.; Sun, Y.; Coovert, M.; Pavlova, J.; Lee, T. Exploration of Intention Expression for Robots. In Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction (HRI '12), Boston, MA, USA, 5–8 March 2012; Association for Computing Machinery: New York, NY, USA, 2012; pp. 247–248. [CrossRef]
27. Kim, H.K.; Park, J.; Choi, Y.; Choe, M. Virtual reality sickness questionnaire (VRSQ): Motion sickness measurement index in a virtual reality environment. *Appl. Ergon.* **2018**, *69*, 66–73. [CrossRef] [PubMed]
28. Hart, S.G. Nasa-Task Load Index (NASA-TLX); 20 Years Later. *Proc. Hum. Factors Ergon. Soc. Annu. Meet.* **2006**, *50*, 904–908. [CrossRef]
29. Hoaglin, D.C.; Iglewicz, B. Fine-Tuning Some Resistant Rules for Outlier Labeling. *J. Am. Stat. Assoc.* **1987**, *82*, 1147–1149. [CrossRef]
30. Braun, V.; Clarke, V. Using thematic analysis in psychology. *Qual. Res. Psychol.* **2006**, *3*, 77–101. [CrossRef]
31. Park, K.B.; Choi, S.H.; Lee, J.Y.; Ghasemi, Y.; Mohammed, M.; Jeong, H. Hands-free human–robot interaction using multimodal gestures and deep learning in wearable mixed reality. *IEEE Access* **2021**, *9*, 55448–55464. [CrossRef]
32. Kyrarini, M.; Zheng, Q.; Haseeb, M.A.; Gräser, A. Robot learning of assistive manipulation tasks by demonstration via head gesture-based interface. In Proceedings of the 2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR), Toronto, ON, Canada, 24–28 June 2019; pp. 1139–1146.
33. Pascher, M.; Kronhardt, K.; Franzen, T.; Gruenefeld, U.; Schneegass, S.; Gerken, J. My Caregiver the Cobot: Comparing Visualization Techniques to Effectively Communicate Cobot Perception to People with Physical Impairments. *Sensors* **2022**, *22*, 755. [CrossRef]
34. Zhang, F.; Cully, A.; Demiris, Y. Probabilistic real-time user posture tracking for personalized robot-assisted dressing. *IEEE Trans. Robot.* **2019**, *35*, 873–888. [CrossRef]
35. Zlatintsi, A.; Dometios, A.; Kardaris, N.; Rodomagoulakis, I.; Koutras, P.; Papageorgiou, X.; Maragos, P.; Tzafestas, C.S.; Vartholomeos, P.; Hauer, K.; et al. I-Support: A robotic platform of an assistive bathing robot for the elderly population. *Robot. Auton. Syst.* **2020**, *126*, 103451. [CrossRef]
36. Arévalo Arboleda, S.; Pascher, M.; Lakhnati, Y.; Gerken, J. Understanding human-robot collaboration for people with mobility Impairments at the Workplace, a thematic analysis. In Proceedings of the 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Naples, Italy, 31 August–4 September 2020; pp. 561–566.
37. Arévalo Arboleda, S.; Becker, M.; Gerken, J. Does One Size Fit All? A Case Study to Discuss Findings of an Augmented Hands-Free Robot Teleoperation Concept for People with and without Motor Disabilities. *Technologies* **2022**, *10*, 4. [CrossRef]
38. Arévalo Arboleda, S.; Pascher, M.; Baumeister, A.; Klein, B.; Gerken, J. Reflecting upon Participatory Design in Human-Robot Collaboration for People with Motor Disabilities: Challenges and Lessons Learned from Three Multiyear Projects. In Proceedings of the 14th PErvasive Technologies Related to Assistive Environments Conference—PETRA 2021, Corfu, Greece, 29 June 2021–2 July 2021. [CrossRef]
39. Watanabe, A.; Ikeda, T.; Morales, Y.; Shinozawa, K.; Miyashita, T.; Hagita, N. Communicating robotic navigational intentions. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 5763–5769. [CrossRef]