Exploring the Use of Tangible User Interfaces for Human-Robot Interaction: A Comparative Study

Cheng Guo
University of Calgary, 2500 University Drive NW.
Calgary, Alberta, Canada, T2N 1N4
cheguo@cpsc.ucalgary.ca

Ehud Sharlin
University of Calgary, 2500 University Drive NW.
Calgary, Alberta, Canada, T2N 1N4
ehud@cpsc.ucalgary.ca

ABSTRACT

In this paper we suggest the use of tangible user interfaces (TUIs) for human-robot interaction (HRI) applications. We discuss the potential benefits of this approach while focusing on low-level of autonomy tasks. We present an experimental robotic interaction test bed to support our investigation. We use the test bed to explore two HRI-related task-sets: robotic navigation control and robotic posture control. We discuss the implementation of these two task-sets using an AIBO™ robot dog. Both tasks were mapped to two different robotic control interfaces: keypad interface which resembles the interaction approach currently common in HRI, and a gesture input mechanism based on Nintendo Wii™ game controllers. We discuss the interfaces implementation and conclude with a detailed user study for evaluating these different HRI techniques in the two robotic tasks-sets.

ACM Classification Keywords

H5.2 [Information interfaces and presentation]: User Interfaces – Interaction Styles

Author Keywords

Human-Robot Interaction, Tangible User Interface, Gesture Input

INTRODUCTION

Over the last few decades a large variety of robots have been introduced to numerous applications, tasks and markets. They range, for example, from robotic arms that are used in space station assemblies to explosive ordnance disposal robots dispatched on battlefields. Depending on the task difficulty and the complexity, the interaction techniques used by the human operator to control a robot may vary from simple mouse clicks to complicate operations on a specialized hardware.

When performing tasks, the human operator may need to break down high-level commands such as "pick up that object" into a sequence of low-level discrete actions that the robot can perform, and then translate each action to a key or switch on the user interface to trigger the appropriate action on the robotic platform. The necessity to perform high-level task planning and management through the composition of low-level actions is not ideal. Depending on the low-level set of interactions, the overall experience can be unnatural, confusing and can cause task failure and endanger the robot in case of critical tasks. As the level of task difficulty increases, it is ideal if the operator spends more time on high-level problem solving and task planning than on low-level robot operations. Intuitive interfaces, well mapped to specific human-robot interaction (HRI) tasks, can allow users to focus on tasks goals rather than on the micro-scale operations needed to accomplish these goals. We believe that orthodox input devices such as keyboards and joysticks can often hinder higher-level interactive tasks as their physicality (layout of keys and buttons) is limited and cannot always be mapped intuitively to a large set of robotic actions.

The aforementioned problem can be tackled by searching for natural and intuitive input methods for robotic interfaces, with one possible avenue being the use of gestures. Studies have shown that children begin to gesture at around 10 months of age [18] and that humans continue to develop their gesturing skills from childhood to adolescence [17]. This natural skill coupled with speech enables us to interact and communicate with each other more effectively. In contrast, moving a mouse and typing on a keyboard, which are arguably not difficult to learn, are acquired skills that are not as innate as performing gestures with our hands and arms. Also, the generic nature of the mouse and keyboard cause them to be inappropriate for certain tasks, which can break the flow of users' cognitive engagement with the task, negatively impacting performance [8]. Can specialized gesture controlled input devices offer more efficient mappings from human to robot than the prevalent keyboard, joystick and mouse interface for a certain set of HRI tasks?

Tangible user interfaces (TUIs) exploit embodied interaction [6], coupling physical objects with computerized qualities, and ideally empowering users with simple and natural physical interaction metaphors. Intuitive, efficient spatial mappings underlie the design of tangible user interfaces [19, 23]. TUIs make effective use of the affordances [19] of physical objects which can directly represent their functionality. The shape, size and weight along with other physical properties of a physical object imply the way we interact with it. By taking the advantage of the affordances of physical objects we may design a set of physical robotic interfaces directly mapped to the

physical aspects and the potential functionalities of robotic platforms. Furthermore, the spatial orientation and the position of a physical object in relation to its surroundings can reveal additional information and provide interaction insight and task awareness to the manipulator. When controlling a robot, maintaining good situational awareness [7] is crucial to the operator. If a physical object can be transformed into a tool for controlling a robot, then the orientation and position of the object in the physical space can be utilized to provide additional information about the status of a robot. We see great potential in using TUI-based mediators for supporting more natural human-robot interfaces.

To explore the possibilities of applying TUIs to HRI, we utilized the Nintendo Wii™ controllers, the Wiimote and Nunchuk, as generic TUI for capturing human postures. The Wiimote can be viewed as a generic 3D TUI, similarly to the view of the mouse as a generic and very successful 2D TUI [23]. The Wiimote can also be seen as a gestural interface [3], arguably representing a gestural/TUI duality which is a key to its commercial success. We believe that the Wiimote supports simple tangible interaction metaphors and techniques that could not have been as successful with pure gestural interaction (that is, with no physical embodiment, and with no physical interface to hold and manipulate).

To utilize the power of the Wiimote, we used it as a robotic interface for fusing dynamic human postures and gestures with robotic actions. (Figure 1) In order to assess the quality and effectiveness of the Wiimote and Nunchuk as robotic interface, we designed an experimental test bed that allowed us to test them against a generic input device – a keypad, with a robot that has 0% autonomy and 100% intervention ratio [29]. Our experimental test bed was based on a Sony AIBO™ robot dog which the user had to control through a variety of tasks. We used the test bed to conduct a user study investigating the advantages and drawbacks of each interaction method in practical HRI tasks.

In this paper we briefly present related TUIs efforts and other instances of gesture input interfaces in the field of HRI. We describe in detail our Wiimote and Nunchuk interaction technique implementation, the baseline keypad interface and the robotic test bed. We present our experimental design, the comparative user study and its results. We discuss the findings and their implications on using gestures in Human-Robot Interaction tasks vis-à-vis a more orthodox keyboard-based approach.
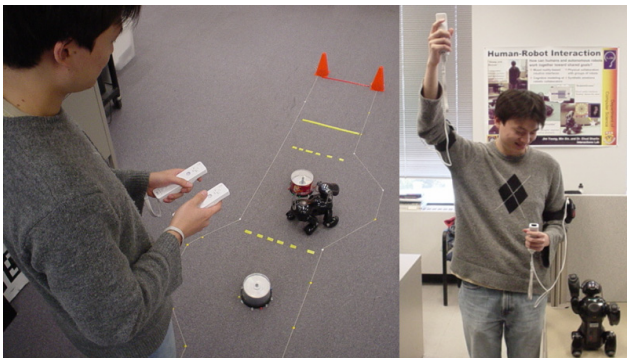
RELATED WORK

HRI is a relatively new sub area of study in HCI. A large amount of effort in the field of robotics has been spent on the development of hardware and software to extend the functionality and intelligence of robotic platforms. Compared to the substantial increase in the variety of robots and their capabilities, the techniques people use to command and control them remain relatively unchanged. As robots are being deployed in more demanding situations, the need for meaningful and intuitive interaction techniques for controlling them has raised a considerable amount of attention among HRI researchers. In terms of interface design, Goodrich and Olsen's [11] work provided a general guide to HRI researchers on how to design an effective user interface. Drury, Yanco and Scholtz had defined a set of HRI taxonomies [29] and conducted a thorough study [28] on how to improve human operators' awareness of rescue robots and their surroundings. To broaden the view of HRI researchers in interface design, Richer and Drury [22] had summarized and formed a video game-based framework that can be used to characterize and analyze robotic interfaces.

Yanco and Drury defined and detailed sets of robotic interfaces terminologies and definitions in an effort to classify the different HRI approaches explored in the domain [29]. Specially, they defined that a robot's autonomy level can be being measured as the percentage of task time in which the robot is carrying out its task on its own. In correspondence, the amount of intervention required for a robot to function is measured as the percentage of task time in which a human operator must be controlling the robot. These two measures, autonomy and intervention, sum up to 100% [29]. In this paper, we are mainly focused on interactions with robot of low autonomy level.

The notion of tangible user interfaces [13] is based on Fitzmaurice et al.'s earlier Graspable User Interfaces effort [10]. Fitzmaurice and Buxton have conducted an experiment which allowed users to use "Bricks" [9] as physical handles to direct manipulate virtual objects. Their study has shown that *a space-multiplex input scheme with specialized devices can outperform a time-multiplex (e.g., mouse-based) input design for certain situations*. [9] Later, Ishii and Ullmer proposed the term Tangible User Interfaces [13] and addressed the importance of both the foreground interaction which consists of using physical objects to manipulate virtual entities and the background interaction which happens at the periphery to enhance users' awareness using ambient media in an augmented space. In our research, we focus on the essence of TUIs which is defined by Ishii as "*seamless coupling everyday graspable objects with the digital information that pertains to them*" [13]. Moreover, we want to select a TUI that has a tight spatial mapping [23] with robotic actions. Spatial mapping can be defined as "*the relationship between the object's spatial characteristics and the way it is being used*" [23]. A "good" TUI for HRI should take advantage of

Figure 1. Wiimote, Nunchuk and AIBO

its physical and spatial characteristics to reflect the physical state or function of robots.

Another quality of TUIs that can make them an interesting choice for HRI tasks is I/O unification, or the natural coupling of action and perception space [23]. TUIs, like any physical object, can allow the user to perceive and act at the same place and at the same time. This quality is often lost in orthodox interfaces that usually separate action space (e.g. mouse) from perception space (e.g. display). By capturing this natural quality of physical objects, TUIs can allow the user to be more attentive and focus on the task at hand.

Beaudouin-Lafon [4] discusses measures for the mapping of physical controllers to their use in digital applications. He defines "degree of integration" as the ratio between the degrees of freedom of the controller to the degrees of freedom of the entity being controlled. "Degree of compatibility" is defined as the similarity between the actions performed on the controller to the reaction of the entity being controlled [4]. Although Beaudouin-Lafon does not discuss mapping between TUIs to robotic tasks, we believe the measures introduced in his work can be adapted to HRI and we use them later in this paper to evaluate the different mappings we implemented.

Using gestures to interact with robots is not a new idea. A significant amount of work has been done using either vision based [12] or glove based mechanisms [24] to capture human arm and hand gestures. Among these efforts, we found that Korenkamp et al.'s [15] work is somewhat similar in approach to the gesture recognition technique we used. Their paper presented a vision-based technique to monitor the angles between a person's forearm and upper arm to predict the gesture that the person is performing. Korenkamp et al.'s efforts were based on a single arm-only interaction, mapped to two main robotic actions: "stop" and "point-at-target". For our approach, we used the Wiimote and Nunchuk to detect the rotation angle of a person's shoulder and elbow joints in relation to the arm rest position. Moreover, our system supports simultaneous movements of two arms with eight different gesture-to-action mappings.

Another interesting approach to the integration of the human body as a robotic input device is exoskeletons system [5, 14]. Kazerooni et al.'s [14] research was focused on augmenting the human body with robotic arms to extend the physical strength of an individual. The human operator wore a robotic arm to directly apply *mechanical power and information signals* [14] to the robot. By measuring the dynamic contact force applied by the human operator, the robotic limbs are able to amplify that force for performing heavy duty tasks that normal human strength would not be able to perform. The Robotnaut project [5] uses similar concept but a different approach to interact with robots. Bluethmann et al. adopted a master-slave system approach which requires the human operator to wear gloves equipped with Polhemus trackers for detecting arm and hand positions [5]. The Robonaut operator remotely controls the Robonaut from a distance without physically touching it. Both of the interaction techniques mentioned above allow human operators to directly manipulate a robot that is either collocated or remotely located.

To extend the notion of TUI to the field of HRI, two projects have demonstrated the potential of using physical objects to manipulate robots. The Topobo toy application [21] allows kids to assemble static and motorized plastic components to dynamically created biomorphic forms. The system is able to replay motions created by twisting and pulling the motorized components to animate users' creations. By combining physical input and output within the system itself, Topobo allows kids to learn about mechanics and kinematics through rapid trial-and-error. [21] A pioneering effort that utilizes physical object for controlling a robot is presented by Quigley et al. [20]. In one of the studies presented in their paper the authors suggest the use of a physical icon (Phicon [13]) to directly manipulate the roll and pitch angle of a mini-unmanned aerial vehicle (mini-UAV).

SYSTEM DESIGN AND IMPLEMENTATION

In order to explore the possibility of using gestures for HRI, we were looking for a robotic platform that would allow us to have full and flexible control in lab settings. The robot should be able to response to both high level commands (such as walking or turning) and low-level commands (such as rotate a specific joint by a certain number of degrees) to match the meaning of both abstract gestures (such as arbitrary hand gestures used in a speech) and specific gestures (such as teaching others a specific movement by demonstrating a similar gesture). Moreover, we were searching for an anthropomorphic or zoomorphic robot that resembles the human skeletal structure to a degree in order to achieve an intuitive mapping between the user interface and the robot. In search for robots that satisfy the above criteria, we found that the AIBO robotic dog can be a suitable platform for our studies. The AIBO is a zoomorphic robot that resembles parts of the human skeletal structure. For instance, the AIBO has "shoulder" and "elbow" joints on its forelegs which act similarly to human's shoulder and elbow joints. By using the Tekkotsu framework [25], developers can gain full control over the low-level actuators and high-level body gestures and movements of the AIBO.

To evaluate the usability of gesture input for HRI in contrast with a generic input device we have designed two interaction techniques for manipulating an AIBO in a collocated setup. One of the interaction techniques supports human gesture input through a Wiimote and Nunchuk interfaces, another input technique uses a keypad as the basis for interacting with the AIBO. During the selection of TUIs, the Nintendo Wiimote came to our attention. The Wiimote clearly differentiates itself from other generic controllers in terms of the interaction style. Instead of pressing buttons, the Wiimote allows players to use motions such as, swing, shake and thrust to interact with the virtual objects on the TV screen. Players feel more immersed and satisfied when using the Wiimote due to the fact that virtual entities in games react to their physical inputs. Although the Wiimote does not qualify as a highly specialized TUI, we believe it can be categorized as a generic 3D tangible user interface due to its ability to capture physical input and to

interact with digital entities. Due to its generality, we think that the Wiimote's basic physical affordances are a good, fundamental starting point for exploring the use of TUIs in HRI tasks. Success in mapping a robotic task to a tangible/gestural interaction via a simple TUI will point to the great potential of better and more elaborate TUIs in more complex HRI tasks. In order to utilize the power of Wiimote and apply it to control an AIBO, we used a PC equipped with both Bluetooth and 802.11b wireless network adapter to act as a mediator to translate and transmit the command from the Wiimote to the AIBO.

Another interface that we selected for representing the generic input device is an OQO 02 Ultra-Mobile PC (UMPC) with an onboard thumb keyboard. The OQO 02 is a scaled down version of a regular desktop PC. It has built-in wireless network adapter that can be used to communicate with an AIBO. The OQO-based "button-press and key-to-action mapping" interaction style represents a common interaction technique in current HRI.

When designing the interface we had to deal with a short (about 500ms) latency issue, resulting from the wireless network transmission and the robot's electromechanical startup time. To maintain the fairness of the experiment, the underlying controlling code for both techniques was identical. Thus, the amount of lag the participants experienced was the same using both techniques, unless there were random wireless transmission spikes.

Wiimote & Nunchuk Input
The Wiimote consists of a D-pad, a speaker, four LEDs and eight buttons. It communicates with the Wii via Bluetooth wireless link. A complete 3-axis accelerometer [1] located inside the Wiimote measures a minimum full-scale range of ± 3g with 10% sensitivity. It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration. [1] An extension port is located on the bottom of the Wiimote to allow peripherals such as a Nunchuk to be attached. The Nunchuk has an analog stick and two buttons and uses the same accelerometer on the Wiimote to support motion sensing.

Accelerometer
In order to understand the Wiimote's motion sensing capability, we need to examine its acceleration measuring mechanism. According to the Data sheet [1] of the ADXL 330 accelerometer:

"The sensor is a polysilicon surface micromachined structure built on top of a silicon wafer. Polysilicon springs suspend the structure over the surface of the wafer and provide a resistance against acceleration forces. Deflection of the structure is measured using a differential capacitor that consists of independent fixed plates and plates attached to the moving mass… Acceleration deflects the moving mass and unbalances the differential capacitor resulting in a senior output whose amplitude is proportional to acceleration."

In other words, the sensor does not measure the acceleration of the Wiimote, but rather the force exerted by the test mass on its supporting springs. [27] When the Wiimote is at rest on a flat surface the accelerometer reading is 1 g (approximately 9.8 m/s^2) due to gravity. When it is in a free fall motion, the reading is close to zero. These facts implies that one, we can only derive a relatively accurate measuring of the pitch and roll angle of the Wiimote when it is reasonably still. This is because when the Wiimote is accelerating (e.g. when a user is swinging the Wiimote), the acceleration value sensed by the Wiimote is due to the force exerted by the user rather than the pulling of gravity. Thus, the tilting angle derived based on this force does not represent the current position of the Wiimote. Two, the accelerometer cannot detect the rotation angle around the gravitational axis. For instance, when the Wiimote is facing up (e.g. the A button is facing upward) and resting on a flat surface, the Z-axis (Figure 2) of the accelerometer is parallel to the direction of gravity. Thus, it does not matter how we orient the Wiimote on the surface, the acceleration value sensed on the Z-axis always remains the same. This means we lose one degree of freedom when one of the axes of the accelerometer aligns with the direction of the gravity.

Figure 2. The coordinate system of Wiimote

Due to the constraints associated with the accelerometer and the unavailability of a motion analyzing package, we are left with the choice of measuring pitch and roll angles for recognizing arm and hand gestures. In our experiment, we want to allow users to use large arm movements for controlling an AIBO, because large movements are easier to distinguish when processing the Wiimote data and also easier to memorize by the user. Therefore, we decided to rely on only using the pitch angle of the Wiimote and Nunchuk to predict arm positions. In this case, we use the Wiimote and Nunchuk as a one degree of freedom input devices to measure the rotation angle of a person's elbow and shoulder joint in relation to the arm rest position.

In order to access the acceleration value sensed by the Wiimote and Nunchuk we used Brian Peek's C# library [16] for acquiring the accelerometer readings. To covert the raw acceleration value into rotation angles, we enter the calibrated raw acceleration values into the following equation, where the variable a_y denotes the calibrated acceleration value along the Y-axis: [26]

$$\text{Pitch} = \text{asin} (a_y / 1)$$

OQO 02 Thumb Keyboard
The generic input device that we have chosen is the thumb keyboard on an OQO 02 UMPC. The OQO 02 is equipped with a 1.5 GHz VIA C7M ULV CPU, 1GB DDR2 SDRAM and a 60GB HDD. It runs on Windows® XP Tablet PC Edition 2005. It supports both 802.11 a/b/g and Bluetooth network standards. The input devices on the OQO 02

include a dedicated mouse, a backlit thumb keyboard with a total of 58 keys (including function keys, letter keys and a number pad) and a digital pen. The letter keys on the thumb keyboard follow the QWERTY keyboard layout. The OQO 02 can be either powered by a removable lithium-ion polymer battery or an AC charger. For our comparative study we used the thumb keyboard only for controlling an AIBO.

EXPERIMENTAL DESIGN

To compare and better understand how well people can learn and utilize the aforementioned techniques when controlling a robot, we designed an experimental test bed based on two tasks for comparing the techniques in terms of speed, accuracy and subjective preferences of the participants. Our goal was to explore the benefits and drawbacks associated with each interaction technique, and to try to point out which technique supports a more effective, intuitive and rich user experience when interacting with a robot.

Pilot Study

Before the full user study, we had conducted a pilot study to test the usability of both interaction techniques and the experiment fairness under different conditions. Our pilot study included 8 participants recruited from our lab. We found that our posture recognition technique does not suit well with people who have large body size. Thus, we changed our system to allow for a more flexible range of input. However, misrecognition still occurred during the pilot study. To minimize the impact of this problem on the participants' task completion time, we modified the underlying software component that supports the interaction to automatically record the time when each posture command is triggered. The examiner also used the same software to manually log the time when a correct posture is preformed by pressing a button on a keyboard. A video tape recorder is used for backup purposes, capturing the entire experiment for replay and time synchronization purposes.

To enable participants to navigate the AIBO, we initially used the "W, A, S and D" key mapping on the OQO keypad for the navigation test. However, in this particular key arrangement, users only need to use their left thumb for most of the movements they need to perform. On the other hand, with the Wiimote technique, users have to use both hands with equal amount of effort to navigate the AIBO. To balance the single hand interaction technique with an asymmetric bimanual [2] interaction technique we revised the key mapping of the keypad interface (the revised mapping is explained in detail in the next section).

USER STUDY

Participants

For the comparative user study, we recruited twenty participants (16 males and 4 females) from the University of Calgary; each participant was paid $10 for taking part in the experiment. Ages ranged from 18 to 29 ($M = 21.75$, $SD = 3.05$). All of the participants reported to use computer keyboard everyday. Among all of the participants, eighteen people were right-handed, one person was left-handed and one person was ambidextrous. All of the participants indicated that they have some sort of computer game experience. Fifteen participants reported to play computer games on a daily or weekly basis. Seventeen participants indicated that they "often" or "very often" use computer keyboard to play games. Six participants reported no prior experience playing the Nintendo Wii. Out of the fourteen people who had previous experience with the Wii only three participants reported to play it on a weekly basis. The other 11 indicated playing either "Monthly" or "Rarely".

Task and Procedure

Our experiment was designed for two different tasks, *robotic navigation* and *robotic posture*, each with two difficulty levels. The participants were asked to perform both tasks with both interaction techniques. Thus, in total, participants had to go through four sub experiments in order to complete the study. The order of techniques was counterbalanced among participants by alternating the tasks order, thus ten participants started with the Wiimote Interface and ten participants started with the OQO interface. The experiment was conducted following a written protocol. Participants were asked to start with one interaction technique to complete both *navigation* and *posture* tasks and then switch to the other technique and repeat the two tasks. During the experiment, each participant was asked to complete four sets of questionnaires after each trial and, once finished, to complete a post-study questionnaire which was followed up with a non-structured interview. Each study took around 60 min. to complete.

To allow participants to learn and practice each interaction technique and to familiarize themselves with the tasks a practice trial was administrated before the full experiment started. The administrator demonstrated the interaction techniques and presented guidelines on how to complete the tasks. Then, the participants would try out the interaction technique until they felt proficient and comfortable to move on.

The main dependent measure in the experiment was the task completion time. In addition, we recorded the number of errors that the participants made with each interaction technique.

Task 1 – Navigation

Description

In this task, the participants were asked to navigate the AIBO through an obstacle course (Figure 3). The obstacle course is 262 cm in length and 15.3 cm in width. The goal of this test is to see how well both interaction techniques support user control in a fairly delicate robotic navigation task. We provided the user with eight different navigation control actions: *walk forward, stop, walk forward while turning left, walk forward while turning right, rotate left, rotate right, strafe left and strafe right*.

To motivate the participants to use all actions, we designed two routes for the task. For the easier route, participants were not forced to use any particular action during the course of the obstacle course and were allowed to use any

combination of actions they want. However, for the harder route, participants were forced to use rotation and strafing in addition to walking and turning in order to complete the obstacle course successfully. A dotted yellow line on the course (Figure 3) indicated the starting point of the strafing action. The solid yellow line indicated the starting point of the *rotate right* action. In order to finish this task, the participants were asked to complete the easier route first followed by the harder trail.

Before the start of the experiment, we reminded the participants to complete the obstacle course as fast as possible, and try to make as few errors as possible. An error in this task is defined as hitting obstacles, navigating the AIBO out of the route boundary or failure to perform required actions at the specified locations. If a participant navigated the AIBO out of the route boundary, then she/he had to navigate it back to the route and continue on. If a participant failed to perform the required action at certain locations during the trial the examiner had to physically move the AIBO back to that location and ask the participant to try again. This error correction mechanism could have introduced a variable amount of time into the task completion time depending on how fast the examiner moves the AIBO back to the right location. We emphasized the "penalizing" implications of this set of errors to participants and were pleasantly surprised to see that none of the experimental trials required the administrator to physically move the AIBO or to manually correct any out-of-bound navigation errors.
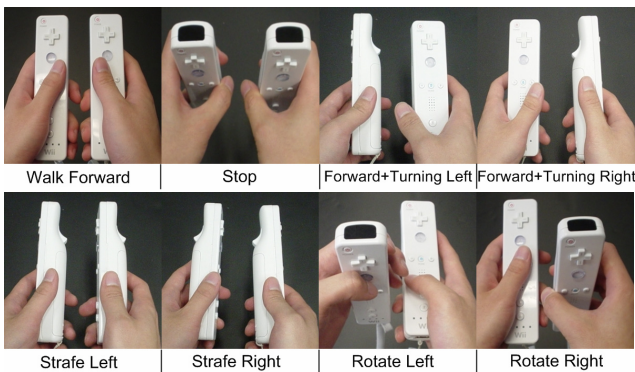

Figure 3. The obstacle course

Figure 4. The Wiimote interaction technique for controlling the movement of the AIBO

Figure 5. Key – Movement Mapping

Interaction Techniques

In this task, the function mapping for the Wiimote interface is presented in Figure 4 and the mapping for the keypad interface is presented in Figure 5. The gesture mapping for the Wiimote controller was developed based on horseback riding techniques metaphor. The participants were told to think of the pair of Wiimotes as a neck rein on the AIBO. For instance, pulling both Wiimotes back will stop the AIBO; pulling the right Wiimote only will rotate the AIBO to the right, etc.

Due to the nature of the task, the gesture-to-robot action mapping is somewhat indirect. In this case, the participants are not controlling a single joint of the AIBO but rather the spatial kinematics of the robot when navigating it through the obstacle course. This implies a non-ideal "degree of integration", and a weaker "degree of compatibility" [4] for the gesture-to-robot action mapping in this task. However, we can argue that the horseback riding metaphor provides efficient and intuitive mechanism for dealing with this abstract mapping.

The keypad-to-robot action mappings are: Forward – W + 2, Stop – S + 5, Forward + Turning Left – A + 2, Forward + Turning right – W + 6, Strafe Left – A + 4, Strafe Right – D + 6, Rotate Left – S + 2, and Rotate Right – W + 5. (the plus sign means pressing and holding the keys on both side of the sign).

The Data collected from this task was analyzed using a 2 x 2 within-subjects ANOVA for the following factors:

- *Technique*: Wiimote, Keypad
- *Difficulty*: easy, hard.

Task 2 - Posture

Description

This task is used to examine the usability of both interaction techniques for low-level robot control. In this task, we asked the participants to perform twelve different postures with the forelegs of the AIBO. We displayed an image of the AIBO with a posture on a computer screen. Then the participants were asked to control the AIBO to imitate that posture. In the experiment setup, we have pre-defined four different postures for each foreleg of the AIBO. (Figure 6) We selected ten postures out of the sixteen possible combined postures using both forelegs. Then, we divided them into two groups of postures which can be chained together to create gesture sequences (Figure 9). The only difference between these groups of postures is that in order to transform from one posture to another within a group, the participants have to manipulate either one foreleg or both forelegs of the AIBO to complete the transition. We define the group of postures that require only one arm movement during the transition as the easier set, and the other group as the harder set. For the experiment, the participants were
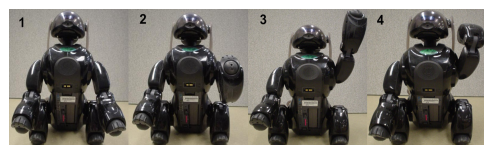

Figure 6. The possible postures for each foreleg of the AIBO

asked to perform the easier set first followed by the harder set.

Similar to task 1, we measure the task completion time and the number of errors. The task completion time in this task is defined as the time that elapsed since a new posture image was displayed on the screen till the time the participants completed the correct robotic posture. Completion time was measured automatically by the software according to the user sensed gestured, with a manual measurement for backup. The error in this case is defined as performing a posture that is different from the posture displayed on the screen. If a participant fails to perform the correct posture, then he/she needs to correct themselves. The time it takes the participants to think and correct their postures is also taken into account as part of the task completion time. Since the harder posture set requires the participants to move both forelegs of the AIBO, the actions can be preformed either sequentially or simultaneously. In this case, we did not constrain the participants to any of the input styles, allowing them to gesture either sequentially or simultaneously, as long as they feel it is the fastest and most intuitive way to complete the postures.

Interaction Techniques

For this task, the function mapping for the Wiimote interface is presented in Figure 7 and the mapping for the keypad interface is presented in Figure 8.

For the gesture input technique, the participants directly adjust the position of the forelegs of the AIBO using their own arms. Compare to the navigation task, the gesture-to-robot action mapping in this case has an almost prefect degree of integration, and a high degree of compatibility [4].

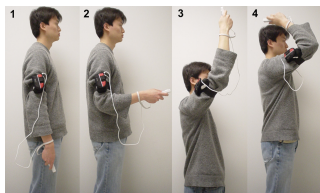For the keypad-to-robot action mapping, the four letter keys

Figure 7. Arm posture input. These postures correspond to the four AIBO postures showed in Figure 6.

Figure 8. Key-Posture Mapping.

Figure 9. Posture 1-6 is the easier posture group. Posture 7-12 is the harder posture group.

on the OQO control the right foreleg of the AIBO. The four number keys control the left foreleg of the AIBO. By pressing either X or 8, the AIBO will perform Posture 1 (Figure 6) with either its right foreleg or left foreleg. By pressing either Z or 9, the AIBO will perform Posture 2. By pressing either A or 6, the AIBO will perform Posture 3. By pressing either Q or 3, the AIBO will perform Posture 4 (Figure 6).

The Data collected from this task was analyzed using a 2 x 2 within-subjects ANOVA for the following two factors:

- *Technique*: Wiimote/Nunchuk, Keypad
- *Posture*: posture 1 to 12 (Figure 9).

RESULTS

Task 1 - Navigation

Task Completion Time
A 2 x 2 (*Technique* X *Difficulty*) ANOVA, with repeated measures on both factors, revealed no significant *Technique* X *Difficulty* interaction ($F_{1,19} = 1.54$, $p = 0.23$), which suggests that performance with the techniques is not substantially influenced by the difficulty level. There was a significant main effect for *Technique*, $F_{1,19} = 12.19$, $p < .001$, indicating that overall task completion time for the Wiimote technique ($M = 43.2s$, $SD = 6.9s$) was 10% faster than for the keypad technique ($M = 48.5$ s $SD = 6.7s$) (Figure 10). As we expected, the main effect of *Difficulty* was significant, $F_{1,19} = 115.61$, $p < .001$, with the mean jumping from $M = 38.7s$, $SD = 4.6s$ for the easy trail to $M = 53.0s$, $SD = 8.1s$ for the hard trail.

Error
A two-way ANOVA was used to determine if there were differences on the number of errors (dependent variable) participants made using the Wiimote and keypad techniques when performed the navigation task under different difficulty levels. The result of the ANOVA showed no significant *Technique* X *Difficulty* interaction ($F_{1,19} = 0.03$, $p = .87$), which suggests that the number of errors made using different techniques is not significantly influenced by the difficulty level. There was a significant main effect for *Technique*, $F_{1,19} = 9.81$, $p < .01$, indicating the errors that participants made using the Wiimote technique ($M = 0.35$, $SD = 0.4$) is 43% less than using the keypad technique ($M = 0.83$, $SD = 0.6$). The result also showed a marginally significant main effect for *Difficulty* ($F_{1,19} = 3.96$, $p = .06$), with mean varying from $M = 0.43$, $SD = 0.4$ for the easy trail to $M = 0.75$, $SD = 0.6$ for the hard trail.
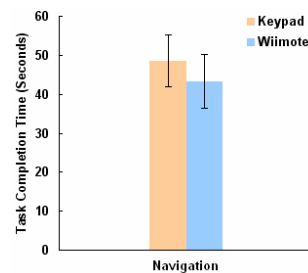

Figure 10. Mean Task Completion Time for Navigation Task.

Task 2 – Posture

Task Completion Time

A 2 x 12 (*Technique* X *Posture*) ANOVA on the task completion time for the posture task showed a significant *Technique* X *Posture* interaction effect ($F_{11,209} = 8.43$, $p < .001$), which means that the *Technique* effect varies with *Posture* or vice versa.

On the average, there was a significant effect for *Technique* ($F_{1,19} = 67.37$, $p < .001$), with mean times reducing from 2.2s ($SD = 0.4$s) with keypad, to 1.5s ($SD = 0.3$s) with Wiimote/Nunchuk; On the average, a 32% reduction in task completion time between the two conditions. On the average, pairwise comparisons showed that there was a significant difference ($p < .05$) between the techniques for posture 1, 2, 7, 8, 9, and 10. But, there was on significant difference for the other postures. (Figure 11) Also, on the average, the test showed a significant effect for *Posture* ($F_{11,209} = 27.77$, $p < .001$).
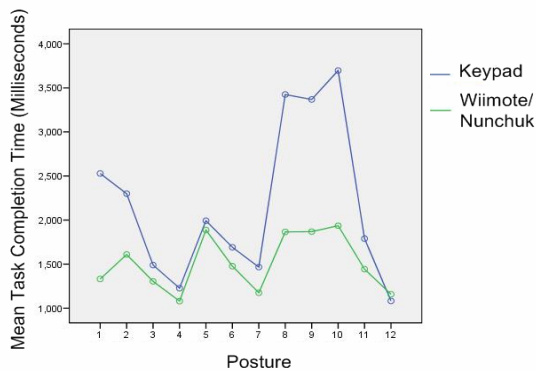

Figure 11. Pairwise comparisons of the mean task completion time for each interaction technique according to posture.

Error

For the keypad interface, participants had made 1.5 ($SD = 1.2$) errors on average for both difficulty levels. However, none of the participants had made any errors using the Wiimote/Nunchuk interface. As anticipated, a paired t-test showed a significant difference ($t_{19} = 7.44$, $p < .001$) between the techniques.

DISCUSSION

The results presented in the previous section point to the Wiimote and the Wiimote/Nunchuk interfaces outperforming the keypad interface in terms of task completion time in both the *robotic navigation* and the *robotic posture* tasks. The differences between the interfaces, although statically significant, are a little underwhelming in their magnitude.

When attempting to explain this for the navigation task, we should consider that both interaction techniques use a set of abstract key and gesture combinations to represent specific robot movements. Since none of the participants have prior experience with these input methods, they have to learn and memorize the mappings of both techniques in order to navigate the AIBO. This abstract mapping between the user interface and the robot action added an extra layer of

cognitive load for the participants to process during the experiment. Although pressing buttons should not be slower than performing gestures, the study showed that the participants finished the obstacle course quicker with gesture input than with button input. We believe that although both interfaces require the participants to think about the abstract mapping before carrying out any actions, the Wiimote interface provides a slight advantage.

When using the Wiimote, participants do not need to focus on their hands while performing a posture. They are naturally aware of the spatial location of their hands. For the keypad interface, we observed that the participants have to constantly shift their attention back and forth between the keypad and the AIBO to look for the buttons they want to press and to confirm if they triggered the intended action. The consequences of shifting attention constantly between the interface and the AIBO may result in action overshoot (for example, overturning a corner) and can break the continuity of the task when participants have to stop the AIBO before they decide which action to take for the next step. This practical separation of action and perception spaces [23] is perhaps the reason for the slower task completion time when using the keypad.

Another possible reason for the faster task completion time when using the Wiimote/Nunchuk in the navigation task may be the zoomorphic rein-like mapping we used. While the mapping offered in this condition is not ideal (see our previous discussion of its degrees of integration and compatibility) the mapping does afford a simple, and arguably intuitive interaction metaphor.

Although the study results indicate that gesture input is faster for the navigation task, we are not suggesting it would always be a better solution than button input for this type of tasks. As we mentioned earlier in the pilot study section, the keypad mapping that we used was arguably not the most intuitive mapping we can come up with. A "W, A, S, D" key configuration would probably be more intuitive to use since it requires less key combinations and is a commonly used mapping in computer games for navigational tasks. However, we believe that our results demonstrate that when participants are limited to use asymmetric two-hand interaction techniques to control a robot, gesture input tends to be more intuitive to use than button input.

For the navigation tasks we did not expect that there would be a significant difference between the numbers of errors participants made using the different techniques. However, the data showed the opposite. Participants made 43% more errors with the keypad interface than with the Wiimote interface. Many participants felt that this was due to the small key size and the less intuitive mapping between buttons and robot actions.

For the posture tasks, we can see that on average there was a significant difference in task completion time between the postures that required two arms movement and the ones that only required one arm movement. By observation, we found that when the participants were using the Wiimote/Nunchuk interface, they were extremely engaged

and focused on the computer screen that displayed the posture images. However, when the participants used the keypad interface, they often looked at the computer screen first, and then focus on the keypad to find the right button to press. This attention shifting problem slowed down the participants' task completion time and can again be associated with the separation between action and perception space created by the keypad.

Most participants felt they were simply mimicking the postures on the computer screen when using the Wiimote/Nunchuk interface, but they felt the keypad interface required them to "act". Following, we believe that the intuitiveness of gesture input had definitely reduced the cognitive load of associating user inputs with zoomorphic robotic actions.

In addition, gesture input tends to support simultaneous input compared to button input. As one of the participants commented, "I could do both hands (both arm movements) at the same time without a lot of logical thinking (with the Wiimote/Nunchuk interface), where with the keyboard I had to press one (button) and the other (button) if I was doing two hand movements at the same time. Although they would be in succession, they would not be at the same time."

It is worth to point out that even though posture 1 and 2 only required single arm movements, there was a significant difference between the task completion times of both techniques. In our opinion, we think this is perhaps due to the participants not being fully trained at the beginning of the study. Thus, they tend to make more mistakes with the first few postures. This may also imply that the Wiimote/Nunchuk interface was easier to learn compared to the keypad interface and can be utilized faster.

Subjective Ratings
We also asked the participants to rate the intuitiveness of both input techniques and indicate their preferred techniques for both tasks. Figure 12 and 13 shows the results of participants' ratings.

After the study, we asked the participants who preferred to use the keypad for the navigation task about their subjective reasoning. All of them responded that they are more familiar with the keypad interface because of related computer game experiences. However, their performance indicates they completed the navigation task when using the keypad slower than when using the Wiimote interface. One of the participants commented, "I have to think harder when I use the keyboard, and this kind of mental overhead coupled with the lag time just makes it feel harder."

For the participants who preferred to use the keypad for the posture task, their reasoning was that they can easily memorize the key-action mapping since there were only four postures for each arm and the buttons associated with both arms are symmetrical on the keypad layout. As one of the participants stated, "With so few postures available, the keyboard was just as easy as the Wiimote." We agree with this participant's comment. We believe that if we provided extensive training to all of the participants using the keypad

interface, they would eventually outperform the Wiimote/Nunchuk interface in terms of task completion time. However, we think that the gestural TUI control method would prevail if we increase the number of degrees-of-freedom and postures to an amount that participants cannot easily memorize, or if we deal with an interaction task that cannot afford intensive training.

During the experiment, many participants asked whether the Wiimote interface supports gradual motion sensing. The consensus indicates that people expect gesture interface to be capable of sensing and reacting to gradual changes of

| Q1	I found the keypad technique is easy to learn for the Navigation task.
	I found the Wiimote technique is easy to learn for the Navigation task.
Q2	I had difficulty remembering how to perform certain movements with the keypad technique for the Navigation task.
	I had difficulty remembering how to perform certain movements with the Wiimote technique for the Navigation task.
Q3	I found the keypad technique is easy to learn for the Posture task.
	I found the Wiimote technique is easy to learn for the Posture task.
Q4	I had difficulty remembering how to perform certain postures with the keypad technique for the Posture task.
	I had difficulty remembering how to perform certain postures with the Wiimote technique for the Posture task.
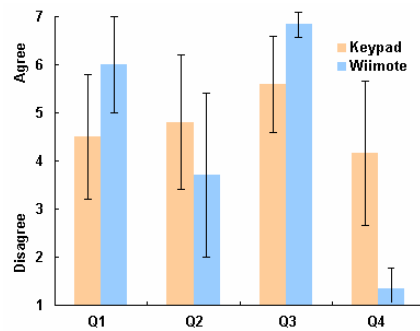
Figure 12. Mean ratings on post-study questionnaire. The rating scale ranges from 1 (strongly disagree) to 7 (strongly agree).
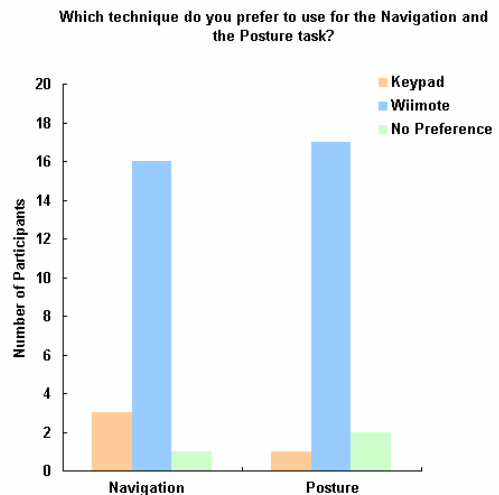

Figure 13. Users' preference for each interaction technique

body motion. However, in order to maintain the fairness of the comparative study, we implemented the Wiimote interface as a state machine to match the limitation of the keypad interface.

CONCLUSION

We have introduced a new interaction technique which utilizes simple generic 3D TUIs (based on the Nintendo Wiimote and Nunchuk) to capture human arm and hand gesture input for human-robot interaction. To evaluate this technique, we have conducted a comparative user study which compares the Wiimote/Nunchuk interface with a traditional input device – keypad in terms of speed and accuracy. We employed two tasks for our study: the posture task utilized a direct mapping between the TUIs and the robot, and the navigation task utilized a less direct, more abstract mapping. The result of our experiment provides some evidence that a gesture input scheme with tangible user interfaces can outperform a button-pressing input design for certain HRI tasks. We have observed a significant decrease in both task completion time and the number of mistakes participants made for both the navigation and posture tasks. The follow-up questionnaire revealed that a significant majority of the participants chose the Wiimote/Nunchuk interface as their preferred technique for controlling an AIBO in both tasks.

In future work, we hope to improve the Wiimote/Nunchuk interaction technique to analyze continuous human arm and hand gestures to extend our abilities in controlling anthropomorphic and zoomorphic robots. We believe more elaborate TUIs would afford intuitive mapping for much more delicate HRI tasks. We also intend to explore the possibility of mapping a large set of TUIs as physical manipulators for a large group of robots.

ACKNOWLEDGMENTS

REFERENCES

1. ADXL330, Analog Devices. http://www.analog.com/ UploadedFiles/Data_Sheets/ADXL330.pdf
2. Balakrishnan, R., Hinckley, K. Symmetric bimanual interaction. In *Proc. CHI 2000*, ACM Press (2000), 33-40.
3. Baudel, T., Baudouin-Lafon, M. Charade: Remote Control of Objects Using Free-Hand Gestures. *Communication of the. ACM*, vol. 36, (1993), no. 7, pp. 28-35.
4. Beaudouin-Lafon, M. Instrumental interaction: an interaction model for designing post-WIMP user interfaces. In *Proc. CHI 2000*, ACM Press (2000), 446-453
5. Bluethmann, W., Ambrose, R., Diftler, M., Askew, S., Huber, E., Goza, M., Rehnmark, F., Lovchik, C. and Magruder, D. Robonaut: A robot designed to work with humans in space. Autonomous Robots (2003), 14(2-3): 179-197.
6. Dourish, P. Where the Action is: The Foundations of Embodied Interaction. MIT Press, Cambridge, MA, USA, 2001.
7. Drury, J. L., Scholtz, J., Yanco, H.A. Awareness in human-robot interactions. In *Proc. IEEE SMC 2003*.
8. Faisal, S., Cairns, P., and Craft, B. Infoviz experience enhancement through mediated interaction. In *Proc. ICMI 2005*, ACM Press (2005), 3-9.
9. Fitzmaurice, G.W., Buxton, W. An empirical evaluation of graspable user interfaces: towards specialized, space-multiplexed input. In *Proc. CHI 1997*, ACM Press (1997), 43-50.
10. Fitzmaurice, G.W., Ishii, H., and Buxton, W. Bricks: Laying the Foundations for Graspable User Interfaces. In *Proc. CHI* 1995, ACM Press (1995), 317-324.
11. Goodrich, M., and Olsen, D. Seven principles of efficient human robot interaction. In *Proc. IEEE SMC 2003*, 3943–3948.
12. Hasanuzzaman, Md., Zhang, T., Ampornaramveth, V., Kiatisevi, P., Shirai, Y., and Ueno, H. Gesture Based Human-Robot Interaction Using a Frame Based Software Platform. In *Proc. IEEE SMC 2004*, 2883-2888.
13. Ishii, H. and Ullmer, B. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In *Proc. CHI 1997*, ACM Press (1997), 234-241.
14. Kazerooni, H. Human-Robot Interaction via the Transfer of Power and Information Signals. IEEE Trans. on System and Cybernetics (1990) Vol.20, No. 2, 450-463.
15. Kortenkamp, D., Huber, E., and Bonasso, R. Recognizing and interpreting gestures on a mobile robot. In *Proc. AAAI 1996*.
16. Managed Library for Nintendo's Wiimote, http://blogs. sdn.com/coding4fun/archive/2007/03/14/1879033.aspx
17. McNeill, D. Hand and Mind: What Gestures Reveal about Thought. University of Chicago Press (1992).
18. Messing, L., Campbell, R. Gesture, speech, and sign. New York: Oxford University Press (1999), 227.
19. Norman, D.A. The Psychology of Everyday Things. BasicBooks, 1988.
20. Quigely, M., Goodrich, M., Beard, R. Semi-Autonomous Human-UAV Interfaces for Fixed-Wing Mini-UAVs. In *Proc. IROS 2004*.
21. Raffle, H., Parkes, A. Ishii, H. Topobo: A Constructive Assembly System with Kinetic Memory. In *Proc. CHI 2004*, ACM Press (2004), 647 – 654.
22. Richer, J., Drury J.L. A Video Game-Based Framework for Analyzing Human-Robot Interaction: Characterizing Interface Design in Real-Time Interactive Multimedia Applications. In *Proc. HRI 2006*, ACM Press (2006), 266-273.
23. Sharlin, E., Watson, B.A., Kitamura, Y., Kishino, F. and Itoh, Y. On Tangible User Interfaces, Humans and Spatiality. Personal and Ubiquitous Computing, Springer-Verlag (2004).
24. Sturman, D., Zeltzer, D. A Survey of Glove-Based Input. *IEEE CG&A* (1994), Vol. 14, No. 1, 30-39.
25. Tekkotsu framework, http://www.cs.cmu.edu/~tekkotsu/
26. Wiimote Motion Analysis, http://www.wiili.org/index.php/Motion_analysis
27. Wiimote Motion Sensor, http://www.wiili.org/index.php/Wiimote#Motion_Sensor
28. Yanco, H.A., Drury, J.L. and Scholtz, J. Beyond Usability Evaluation: Analysis of Human-Robot Interaction at a Major Robotics Competition. Journal of Human-Computer Interaction (2004), Vol 19, Numbers 1 and 2, pp. 117-149.
29. Yanco, H.A., Drury, J. Classifying Human-Robot Interaction: An Updated Taxonomy. In *Proc. IEEE SMC 2004*