

Analysis of Natural Gestures for Controlling Robot Teams on Multi-touch Tabletop Surfaces

Mark Micire, Munjal Desai, Amanda Courtemanche,
Katherine M. Tsui, and Holly A. Yanco
University of Massachusetts Lowell, Department of Computer Science
One University Avenue, Lowell MA 01854, USA
{mmicire, mdesai, acourtem, ktsui, holly}@cs.uml.edu

ABSTRACT

Multi-touch technologies hold much promise for the command and control of mobile robot teams. To improve the ease of learning and usability of these interfaces, we conducted an experiment to determine the gestures that people would naturally use, rather than the gestures they would be instructed to use in a pre-designed system. A set of 26 tasks with differing control needs were presented sequentially on a DiamondTouch to 31 participants. We found that the task of controlling robots exposed unique gesture sets and considerations not previously observed, particularly in desktop-like applications. In this paper, we present the details of these findings, a taxonomy of the gesture set, and guidelines for designing gesture sets for robot control.

Author Keywords

Human-robot interaction, human-computer interaction, robot control, multi-touch, tabletop interface, gestures

ACM Classification Keywords

H.5.2. User Interfaces: Interaction styles, I.2.9 Robotics, J.7.a Command and control

INTRODUCTION

Multi-touch interfaces have been shown to be useful for geospatial and temporally grounded tasks. Some of the successful applications include geographical information system visualization [1, 15, 16], command and control [2, 6, 11, 12, 17], and multi-dimensional data analysis [3]. In the last few years, the command and control of robots has emerged as a potentially useful application of multi-touch technology [8, 10, 14, 18]. In 2001, the US Department of Defense put forth the goal for one-third of deep strike aircraft to be unmanned by 2010 and for one-third of all military ground combat vehicles to be unmanned by 2015 [21]. As such, there is an increasing need to control multiple robots in homogeneous and heterogeneous teams, which could include ground, air, and underwater robots. Even if the robots prove

capable in autonomous operations, the robots will still require a command and control structure for oversight and tasking.

Historically, multi-touch interfaces have used carefully designed gesture sets and user interface (UI) elements. The gesture sets are often tailored around detectability and repeatability. These requirements vary depending on the enabling multi-touch technology and the capabilities of the touch sensor mechanisms. Despite the best intentions of the system designers, often the detectability of a gesture is at odds with its ease of learning. In an ideal setting, a naive user should be able to begin interacting with the multi-touch interface quickly, naturally, and without explicit instructions. In the case of command and control for military operations or disaster response, ease of learning is especially crucial since the commanders typically do not have an abundance of time to learn new user interfaces and must be able to quickly achieve operational proficiency.

To maximize the ease of learning for a command and control interface for teams of autonomous robots, this work uses a different approach to gesture design. The research presented in this paper aims to find the most natural gestures for controlling robot teams, regardless of detectability or input technology. We designed an experiment in which the participant was presented with a number of tasks that had varying numbers of robots and the need for different types of control. With no other visual user interface elements such as menus and windows, the participant was asked how they would express the task to the robot(s). The result is a unique look at how users wish to control multi-agent teams.

RELATED WORK

Most prior work in multi-touch gesture design has taken the approach of letting human-computer interaction (HCI) experts design gesture sets, and then conducting user studies to verify whether these sets are, in fact, natural or easy to use (e.g. [13, 20]). However, this method may not produce the most natural gesture set for naive users. Wobbrock et al. [22] conducted a study to create a natural gesture set by designing around unbiased user input. They presented participants with tasks to perform with one and two hands, and no prompting as to what was an acceptable gesture. They found that users used an arbitrary number of fingers to perform many tasks, so differentiating gestures based on the number of fingers on the tabletop may be a poor choice. Additionally, they found that users preferred using one hand rather than two. Their work

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ITS '09, November 23-25 2009, Banff, Alberta, Canada
Copyright ©2009 978-1-60558-733-2/09/11... \$10.00

was focused on tasks such as word processing and managing documents. As their study confirmed, this domain is heavily influenced by desktop computing and WIMP (window, icon, menu, and pointing device) paradigms; our research goal was to determine if this would also be the case for command and control tasks.

Epps et al. [4] conducted a similar study, using a combination of multi-touch and computer vision inputs to allow users to interact with the system while not in contact with the board. Users were asked to perform common desktop computing tasks on the tabletop, but most employed largely off-the-surface, 3D gestures to accomplish these tasks. The authors concluded that users prefer using their index finger only, and that for more complicated tasks there is a strong need for the ability to gesture off the surface. The current state of the art for off-the-table gestures require range sensing or camera technologies that are sensitive to ambient light, changes in light, and visual noise. The complexity of setting up a secondary gesture detection system is also prohibitive for the field requirements of the command and control domain.

Koskinen et al. [9] examined the possibility of improving power station control rooms using multi-touch tables, instead of the single-touch displays that are currently used. Their study investigated the natural gesture space for this environment by asking users to demonstrate how they would perform common mouse-driven tasks on a multi-touch screen. They concluded that, in general, users prefer single-hand and single-finger interaction over more complicated gestures, and that users preferred gestures that required less contact with the screen. However, we believe that the study most likely introduced a bias: when asked to perform a well-known task which involves a mouse, users are likely to be biased towards using a single finger.

Wu et al. [23] proposed a systematic procedure for designing multi-touch interaction. They identified three major issues that were not addressed in with previous design attempts: incorporating multi-finger and multi-hand gestures into an environment which has been traditionally pointer-based, occlusion issues, and access of areas on the surface which are physically uncomfortable to reach. They defined gestures as having three phases: registration, relaxation, and reuse. Gestures would be registered either statically or dynamically, and, after being registered, the user would not be constrained to maintaining the same hand position (relaxation). In different contexts or when performing different tasks, users could reuse a previous gesture to operate differently, perhaps by using some sort of gestural cue to change tools. In our study, we focus on the initial registration phase, in order to provide the greatest ease of learning.

EXPERIMENT DESIGN

Our goal was to determine the gestures participants would use naturally. The tasks were designed to illicit responses from the participants that were free-form and not constrained by pre-determined graphics and visual feedback. Care was taken to avoid standard UI window conventions such as menus, title bars, and buttons. In this regard, the experiment can be

considered analogous to the paper prototype method often used in early user interface designs [19].

For this experiment, we use a thirty-two inch Mitsubishi DiamondTouch by Circle Twelve and a Dell 5100MP projector. The DiamondTouch was securely fastened to a round wood tabletop with a large steel base, ensuring that participants could interact naturally and rest their arms without accidental movement. The projector was fastened to the ceiling with a custom mount and front reflecting mirror for image adjustment. An overhead camera was secured on a custom camera mount that aligned the camera field of view with the projected image and tabletop.

Thirty-one people participated in the study; each received a movie ticket. The average age was 27.5 years ($SD=10.1$), and nine of the participants were female. All of the participants had some experience with computers. Seventeen participants reported playing video games for an average of 7.8 hours per week ($SD=7.4$). Nine of the video game players reported that they played real time strategy (RTS) games such as StarCraft, Civilization, and Sins of a Solar Empire.

All but two participants reported prior experience with touch screen or stylus-based devices. Eighteen had experience with some type of touch screen phone; sixteen of these were the Apple iPhone. Sixteen participants had used a Palm OS stylus device and thirteen had experience with tablet-based PCs.

Each participant was first briefed on the experiment and introduced to the physical robot, an ActiveMedia Pioneer 2Dx, that would be iconically depicted in the experiment. After answering any questions, the participant completed an informed consent form and a demographic survey. The participant was then presented with the tasks and asked to “think aloud” [5] while completing them. For each task, the participant was presented with a slide with a written description of the task at the top of it; the experimenter also stated the task. Participants were asked to use their finger, fingers, hand, or hands on the tabletop to express how they would command the robot(s) to complete the tasks; there was no time limit on responses. We videotaped the participants’ interactions and commentary using the overhead camera and logged the movements using custom software. The experimenter also took notes. In addition to the think aloud protocol, the participant was asked to talk about any aspects of the interface that they would expect if the interface were active. These could include, but were not limited to, menus, dialog boxes, or interface features.

Twenty-six slides were presented sequentially which showed either one robot, two robots, or two teams of 8 robots each (shown in Table 1). The tasks on the slides introduced a range of desired interactions for robot command and control. Some tasks required very little interaction, while others forced the participant into situations where multiple hands or complex UI strategies were required. Twenty-four showed top-down views of the robot(s), while two showed 3D views. The slides are shown in Table 1. The omitted slides were visually identical and had small changes in the required task (detailed below).

The first three tasks involved only one robot and provided a simple starting point for the participants' understanding of the experiment and talk aloud method. Task 1 displayed only one robot and one labeled area; the participant was instructed to move the robot to the labeled area. Building on the previous task, Task 2 added a wall between the robot and the destination, requiring the participant to either express that the robot needed to go around the wall or make the assumption that the robot was capable of finding a path around the wall on its own. Task 3 extended this one step further by displaying an impassable "swamp" area on the left side. Again, the participant needed to express that the robot should travel around the wall, but not through the swamp.

Two robots were controlled in Task 4 through Task 9. Tasks 4 and 5 asked the participant to command each robot to a separate area. Task 5 asked the participant to command the robots at the same time, encouraging some type of group gesture, multi-touch interaction, or command queuing. Tasks 6 and 7 extended this idea by having the participant command both robots to the same area. This variant was explored since it could allow the participant to use group selection and a single destination gesture. Like Task 5, Task 7 asked the participant to perform the task for both robots at the same time. Tasks 8 and 9 displayed the robots in an area to the left and required the participant to move them to different locations on the screen. If the participant was using multi-handed gestures, this sequence created an arm crossing situation. Again, Task 9 required concurrent actions.

Task 10 asked the participant to simply command the robot to move forward and continue moving. Since there is no destination, this task asks the participant to form a gesture that had no predicate. Tasks 11 and 12 asked the participant to label the robot and rotate the robot respectively.

Groups of robots were displayed in Tasks 13 through 24. In Task 13, the participant was asked to give each of the teams of robots a name. This task required the participant to think about grouping and group selection. Tasks 14 and 15 then asked the participant to have a robot team face a specific area on the map, which extended the group selection gesture to now include a group action. Task 15 was identical, but asked for the team to face the direction at the same time. In Task 16, the team on the right side of the screen was then required to move to Area B. This iteration required group selection, position change, and destination designation. Task 17 then took the previous task one step further and asked the participant to command the team on the right to Area B, and back. This combined action required some sort of command queuing since there were two destinations.

Task 18 was unique since the participant needed to sort the robots and label them by color. Then for Tasks 19 and 20, the participant needed to maneuver the robot groups to various areas and paths on the map. Tasks 21 through 23 explored the gestures to describe map translation, rotation, and zoom. Task 24 asked the participant to command the robots into a line formation, which required a hybrid between group selection and the need for independent movement of robots.

The final two slides were rendered in 3D to explore how participants would control the viewpoint of the robot using only a 2D tabletop surface. In Task 25, the view was from the rear and slightly above the robot; the participant was asked to rotate to a forward view of the robot from the same angle. Task 26 displayed the robot from above; the participant was asked to adjust the view so the robot was seen from behind.

TAXONOMY OF USER DEFINED GESTURES

We began data analysis by looking for large-scale patterns of movement in the overhead video. We refined our description of the patterns of interaction by isolating components of actions using open and axial coding from grounded theory [7]. After several iterations of group discussions and individual video analysis, clear patterns were seen across the majority of the participants. We coded instances of participants' gestures to the consensus of the gesture classifications. Inter-rater reliability was established using Cohen's Kappa statistic ($\kappa=0.74$ excluding chance, $\kappa=0.76$ if chance was not factored out). The data set provided a total of 3197 gestures over 31 participants and 26 tasks.

We identified five groups of interaction classifications for these 26 tasks: selection, position, rotation, viewpoint, and user interface elements.

Selection gestures were used to select a robot, multiple robots, other objects of interest in the environment. For example, a common occurrence of selection was when the participant tapped on a robot (selecting the robot to move) and then dragged their finger on the path that they would like the robot to follow. The initial finger tap on the robot would be classified as a selection.

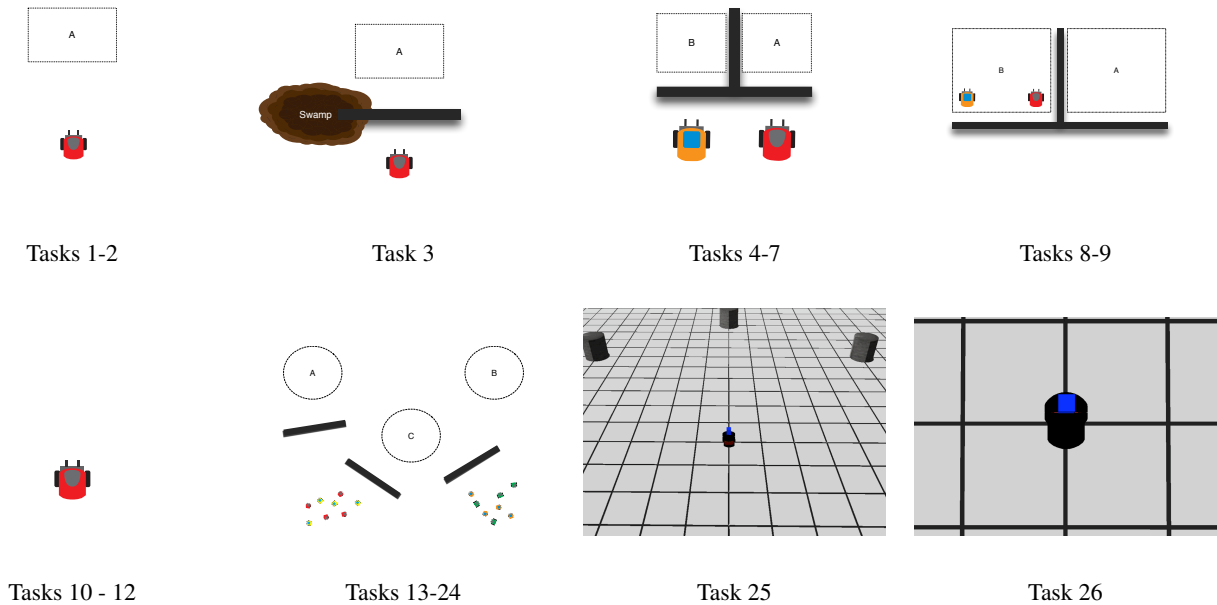
Position gestures indicated a desired change in location of the robot or some object. In the previous example, the drag movement of the finger providing the path would be classified as a drag representing a position change.

Rotation gestures expressed rotation of robots, objects, or the map. Many of the tasks required the reorientation of the robots to face areas of the map. In the simplest of cases, participants used a two finger rotation gesture over the robot image. Other participants creatively used single finger gestures where the rotation was outside of the center of mass of the robot.

Viewpoint gestures were used to change the view by either moving the world or changing the position of the camera. Most commonly, either case was achieved using a single finger on each hand to reorient the screen. Several participants came up with unique and unexpected gestures to accomplish the tasks.

User Interface Elements included commonly used interface elements like buttons, menus, virtual keyboards, handwriting recognition, and voice recognition. This classification was used when the participant described some additional interface element outside of the touch gesture space. Most often this was a verbal statement like, "I would expect to have a menu

Table 1. Illustration and descriptions of some of the 26 tasks performed by participants. Similar tasks have been omitted from this table due to size constraints; full descriptions are given in the text.



here,” while pointing at the area that would contain the menu.

The grouping of these user defined gestures does not imply that gestures across groups are not interrelated or would not be mixed in sequences. In fact, sequencing of multiple groups of gestures was important for many of the tasks. For example, the simple task of moving a robot from its resting position to another area of the screen might require selection (e.g., tap), position (e.g., drag), and then another selection (e.g., double tap). Another method for directing the robot may be simply to select the robot (e.g., tap) and then select the destination (e.g., tap), expecting the robot to determine the best path.

Although the grouping of the gestures is important, a developer should be careful to not use these high level groupings to drive state-based inspection of the gesture intent. In the first example, the selection (tap) can be thought of as the subject of the action, the position (drag) as the verb, and the selection (double tap) as the predicate. This grammar can drive the state of the gesture recognition in this simplistic case. Unfortunately, the grammar quickly falls apart in the second example where only the noun (tap) and predicate (tap) are provided; the verb is omitted from the sentence analogy and must be inferred.

RESULTS AND DISCUSSION

The data set produced 3197 coded gestures using the gesture taxonomy described in the prior section. For each gesture, we recorded the selected object (if applicable), the gesture type, and the destination of the gesture. Since participants had varied levels of verbosity and gesturing, we normalized the data for each task by dividing the examined feature by the total number of gestures used by the participant in the task. This scalar could then be equally compared to other

participants that might have given duplicate explanations or extended talk-aloud narratives. Unpaired two-tailed *t*-tests assuming equal variance with $\alpha = 0.05$ were used to determine significance. The results of the coding are shown in Figure 1. Of particular interest is the low percentage of participants expressing a desire for voice recognition (1.3%) and keyboards (1.5%).

The display provided no visual or audio feedback to the participants; while this eliminated any potential biasing, it also removed any indications that the participant might be providing inconsistent or nonsensical input. We observed that this lack of feedback resulted in tendency for participants to leave their fingers or hands engaged with the tabletop while thinking. These pauses resulted in the coding of more “press and hold” gestures than would have been seen if the interface reacted to the gestures.

Selection

From an overall gesture space perspective, the selection classifications followed several expected trends (shown in Figure 1). Tap accounted for 49% of the selection gestures. This large percentage was expected since this gesture is roughly analogous to the “click” of the mouse in the window user interfaces. Lasso was the second most occurring selection gesture (20%) primarily due to the natural action of drawing an ellipse around the objects to be selected. There was a cluster of selection techniques totaling 27% that included double taps, 2-finger select, press and hold, sequence select, and *n*-finger select. These are also analogous to mouse click methods. The low percentage of bounding box gestures at 1.5% and palm select at 1% is notable since these gestures have been used in several tabletop applications in the past. Overall, the selection results indicate that there is a bias to

Table 2. Taxonomy of user generated gestures based on 3197 gestures over 31 participants and 26 tasks.

	Name	Description
Selection	Tap	Single finger taps object to be selected (See Sequence select for multiple taps)
	Double tap	Single finger double taps object to be selected (See Sequence select for multiple taps)
	Lasso	Single finger draws line encompassing objects to be selected
	Meta	Object selected with some external modifier (e.g. Ctrl, Alt)
	Sequence select	Robots selected in a serial fashion (Supersedes Tap and Double Tap)
	Press & hold	Object touched for a duration of longer than 1 second
	Bounding box	Opposite corners of bounding box are shown with fingers
	Palm	Palm of hand placed on object or objects
	2-finger select	Two fingers on the same hand simultaneously used for selection (Supersedes Tap)
	<i>n</i> -finger	More than two fingers on the same hand used simultaneously for selection (Supersedes Tap)
Position	Drag	Single finger slides across surface to robot destination with immediate lift at end
	Drag & hold	Single finger slides across surface to robot destination with finger hold greater than one second at end
	Waypoint	Tap sequence providing waypoints for robot to follow ending at destination
	Pinch & move	Two finger pinch and then position change to robots' destination
	Flick	One or more fingers placed on robot and finger tip(s) accelerated rapidly in direction of movement
	Path to edge	Finger placed on object and dragged to the edge of screen in direction of movement
	Arrow	Vector gesture terminating in an arrowhead
	Direction segment	Like drag, but smaller segment (vector) not terminating at goal
	Palm drag	Palm placed on object and dragged
	2-finger drag	Two fingers on the same hand are simultaneously used for drag
<i>n</i> -finger drag	More than two fingers on the same hand used simultaneously to perform drag	
Rotation	Finger rotate	Finger placed on object and finger tip rotated
	Pinch & rotate	Two finger pinch and then rotation change
	Off center rotation	Finger placed on object outside of center of mass and rotated
	C-style rotation	Finger begins in the center of the object, extends outward, and begins rotation
	Palm rotation	Palm placed on object and rotated
	2-finger rotation	Two fingers from the same hand placed on the object and fingers rotated.
<i>n</i> -finger rotation	More than two fingers on the same hand used simultaneously to perform rotation	
Viewpoint	Pinch	Thumb and finger(s) converging using one hand
	Rev. pinch	Thumb and finger(s) diverging using one hand
	Finger pinch	Two or more fingers converging using two hands - one or more finger per hand
	Rev. finger pinch	Two or more fingers diverging using two hands - one or more finger per hand
	Vanishing point	Hands placed on side parallel to each other and then angled outward
Elements	Menu selection	Menu appears with more than one object property or action
	Button selection	A button selected by pressing on it, allowing for object modification or action
	Keyboard	A keyboard appears for annotation
	Handwriting	Handwriting modifies object
	Voice recognition	Voice recognition modifies object
Widget	A widget verbally described and interacted via specialized functionality	

classic mouse “click” paradigms, but a gesture to circle or lasso the object would appear to be a very natural alternative.

We found an effect with selection gestures that has implications for gesture registration: in situations with one or two robots, most participants had no explicit selection step. The participant would simply gesture a drag or waypoint for the robot to follow; the selection was implied through the source of the gesture. The movement started at the robot and ended at the destination with no explicit selection of either. In contrast, with a group of three or more robots, the participant would select the robot group explicitly and then gesture the drag or waypoints that the group was to follow. The implications of this finding are important for gesture registration because, although the task is the same, the start of the gesture is different depending on the number of robots being considered for the task.

For tasks that had two robots, there was statistical significance between individual selections (e.g., tap or double tap) and group selection gestures (e.g., sequence select, lasso, and *n*-finger select). Participants used significantly fewer group select gestures ($\bar{X}=0.13$) than individual select gestures ($\bar{X}=0.95$), ($t(60)=3.0, p<0.004$). Participants found it

easier to individually select each of the two robots than to use a gesture such as a lasso for a group selection. We had expected to see lasso used more often for selection of two robots, since this is the way that it would be accomplished in the mouse-driven case; however, participants preferred using individual selections in the case of two robots.

In tasks with three or more robots, participants used significantly more group select gestures ($\bar{X}=2.75$) than individual select gestures ($\bar{X}=0.43$), ($t(60)=6.5, p<0.001$) and the use of group selects was significant against all of the other selection gestures.

Multi-hand and Multi-finger Gesturing

All participants in our study used a multi-handed gesture without explicit prompting by the experimenters. Slightly fewer (90%) used gestures that involved multiple fingers on the same hand; the other 10% used two hands for at least one of the tasks, but never more than one finger on each hand. This result contradicts other studies that found that most users preferred using a single finger on a single hand for their interactions [4, 9]. Instead, we found an almost unanimous willingness to use multi-hand and multi-finger gestures. In the other studies, the tasks performed were largely desktop-

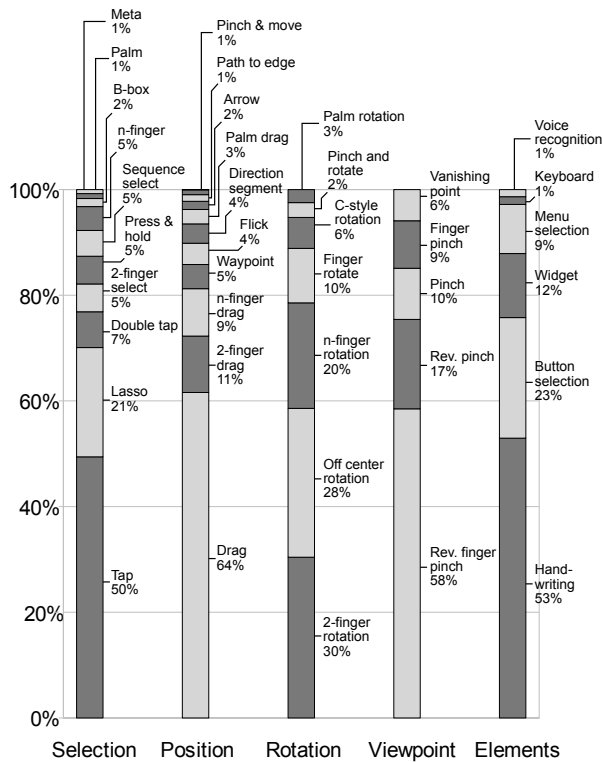


Figure 1. Normalized percentages of the gestures expressed by the participants in each of the classification groups.

computing tasks. Since our task domain is outside of the traditional desktop paradigm, we had predicted that we would see different results. As the data above confirms, we did indeed see this difference. This result is important because it shows that when given unfamiliar tasks, users are more likely break away from mouse-driven paradigms to take advantage of multi-touch capabilities.

We also observed that the number of contact points on the multi-touch surface was influenced by the number of objects requiring manipulation in the task. For tasks that required the selection of two robots, 42% of the participants (13 of 31) used two fingers to select the robots, one finger per robot, rather than use a “group select” gesture. The two fingers used for the robot selection were most often on the same hand if the robots were close to one another and on different hands if the robots were farther apart. Participants used two hands to gesture significantly less often when there were three or more robots ($\bar{X}=0.58$) as compared to tasks where there were one or two robots ($\bar{X}=4.76$), ($t(60)=5.6, p<0.001$). For these tasks with three or more robots, participants tended to perform a group select, using a single hand.

We also observed that the type of task influenced the number of contact points, particularly in cases where prior computer usage introduced a bias. For example, the analogy of dragging an item from one location to another is ubiquitously used in WIMP interfaces. We found that when asked to move a robot or a group of robots, participants continued to use this

dragging paradigm, with single finger drag (64%), 2-finger drag (11%), and *n*-finger drag (9%) as the most used position gestures. While all three gestures convey the same intent, they use differing numbers of fingers. Given that 20% of the movement gestures used two or more fingers to accomplish the same task as a single finger drag, a gesture set for robot control must be designed to allow these gestures to be used interchangeably. This finding that the number of fingers is not significant to the use of the gesture is consistent with [22]. We believe that it is the shared context of dragging an item on a screen that leads to this agreement between the studies.

In contrast to position gestures, rotation gestures showed a tendency toward multi-finger gestures. The single finger rotation classification only accounted for 10% of the total number of rotation gestures. Since rotation is not a very common movement in WIMP interfaces, participants had fewer biases towards a mouse-like single point gesture. These multi-finger rotations follow the same motion that one would perform with physical objects on the table. Since the robot icons were analogues to physical objects in the world, the off-center rotation is a natural response and how one might rotate a robot in the physical world.

Only nine participants used some type of gesture involving their palm, constituting 1.85% of all the gestures. One of the reasons for the low usage of palm based gestures could be the limited flexibility when the palm is in contact with the board.

Handwriting

We observed that the participants tended towards handwriting for annotation tasks when given the freedom to gesture without constraint. 87% of the participants (27 of 31) expected to be able to use handwriting for labeling objects. Only 13% of the participants (4 of 31) described using onscreen keyboards, and 6.5% of the participants (2 of 31) used both keyboards and handwriting. We found this surprising since most ATMs, iPhones, and tablet-based PCs use on screen keyboards as the primary text input method. We had hypothesized that ubiquitous keyboard usage would create a bias toward the description of virtual keyboards or keypads. However, since participants could be free-form in all of their other gestures, they expected to be able to use handwriting and have it be recognized by the system or stored as an image label in their handwriting. Even further emphasizing this free form expectation, many participants did not write on the robot itself: there is an expectation of locality. 92.5% of the participants (25 of 27) that used handwriting gestures performed the gesture on the ground and only 7.4% (2 of 27) of the participants performed the handwriting gesture directly on the target robot(s).

We also observed that 10% of the participants (3 of 31) used both of their hands for handwriting, meaning that these participants were using non-dominant hands for handwriting. The participants abbreviated the labels for the robots in these cases, substituting “L” for “Team Left” and “R” for “Team Right.” Watching the video for this effect, we believe that the ambidextrous handwriting was due to the simplistic abbreviation and the placement of the annotation on the tabletop.

Objects on the left side of the screen were annotated with the left hand and objects on the right by the right hand. This is important since, if implemented, handwriting recognition may need to work correctly with possibly degraded registration from multi-handed handwriting. Alternatively, handwriting may need to be stored as a label, without recognition, particularly in the case of writing with the non-dominant hand.

Gesture Usage as a Function of Prior Experience

Unsurprisingly, we found that the gestures that people wanted to use often correlated to their prior experience with computers, computer gaming, and multi-touch devices. Since the mainstream adoption of personal computers in the 1990's, we expected that desktop computing and WIMP paradigms would affect how the participants commanded the robot(s). The use of computer gaming has also become pervasive. In the past few years, personal electronics have begun to incorporate multi-touch technology, such as the iPhone.

Every participant had used computers, meaning that they had been exposed to common mouse paradigms. Despite the fact that 90% (28 of 31) of the participants used more than two fingers for gestures in their tasks at some point, multi-finger gestures only constituted 8.9% of all the gestures, indicating that the use of WIMP interfaces may be heavily influencing the expectations of the participants. Drag, 2-finger drag, and n -finger drag were the most used position gestures, totaling 84%. While these are natural responses, their use is also encouraged by the ubiquitous use of dragging in mouse paradigms.

We found that for tasks that were uncommon in mouse paradigms, the gesture space did not show such influence. For example, rotation is not extremely common in window interfaces outside of graphic design programs. The single finger rotation and the off-center rotation classifications accounted for 10% and 28% of the gestures respectively, where as the 2-finger rotation and n -finger rotation accounted for 40%.

Since most computer applications have some type of menu system, we expected to see participants describing the use of drop down or on-screen menu systems. However, this was not the case as only 29% of the participants (9 of 31) expressed the need for menus during some point in the experiment. Similar effects were noted for the use of buttons (32% of participants; 10 of 31) and specialized widgets (41% of participants; 13 of 41). Participants that played games expressed the desire for significantly more "widget" gestures ($\bar{X}=0.41$) than those participants that did not play games ($\bar{X}=0.13$), ($t(29)=2.2$, $p=0.0304$), which may be attributed to the fact that games tend to have many custom widgets to perform certain activities.

We believe that iPhones have biased the "pinch" since 53.8% of the zoom gestures were some form of pinch gesture. This is not surprising, but indicates that established gestures cannot be ignored regardless of their "correctness" (or lack thereof) from a HCI perspective. Combined pinch gestures include "pinch," "reverse pinch," "finger pinch," and "reverse finger pinch." Participants that had prior experience using

the iPhone used significantly more combined pinch gestures ($\bar{X}=0.68$) than those participants that had no experience using the iPhone ($\bar{X}=0.39$), ($t(29)=3.9$, $p<0.001$).

Participants that played RTS games had fewer combined pinch gestures ($\bar{X}=0.68$) than those participants that did not play RTS games ($\bar{X}=0.48$), ($t(29)=2.3$, $p<0.028$). The prior experience of these participants might have influenced them against using pinch gestures, since most RTS games are played using a mouse and a keyboard and do not have any form of pinch gesture.

CONCLUSIONS

The past decade has seen a surge in the use of robots in consumer and military applications, with millions of home robots and thousands of military robots now deployed. As robots become more commonplace in large teams, the need to manage their actions as individuals and groups becomes important. When designing an interface for multiple robots, an overhead view of the working area is more useful than the single robot's eye camera view provided by most interfaces designed for single robot control. This overhead view is similar to a map, for which multi-touch computing has been demonstrated to be an effective control method. The addition of robots to map-based multi-agent control creates the need to be able to task individual robots and groups of robots. Our experiment was designed to determine the gestures that people would find the most natural for a variety of tasks in this domain. The assumption was made that the robots would have the ability to be very autonomous in the execution of the specified tasks; without a great deal of autonomy in each individual robot, it would not be feasible to control a group of more than a dozen robots.

We found that the prior experience of the participants introduced some bias into the gestures that they wanted to use. In particular, selection and movement gestures were heavily influenced by standard mouse paradigms. Additionally, we saw that participants who had used iPhones used significantly more pinch gestures for zooming, while people who have spent many hours playing real time strategy (RTS) games expect to have similar controls in the multi-touch domain. However, we also found that when presented with unfamiliar tasks, users are willing to break away from standard mouse-driven paradigms and use multi-touch capabilities freely.

This research has identified several guidelines for designing gesture sets for the robot control domain:

1. If we want the gesture set to be easy to learn, biases introduced by mouse driven interfaces will need to be carried over to the multi-touch domain.
2. To account for individual biases, caused by the use of devices such as the iPhone or a great deal of time playing computer games, gesture sets could include multiple gestures for the same capabilities. One could even imagine a small set of questions to be asked of a user that would customize the gesture set to their experiences.
3. Users expect to provide multiple levels of instruction to

the robot. This includes providing a start and destination, providing way points, and providing an explicit path.

4. If we use free-form gestures (such as lasso) for robot selection and movement, then there will be an inherent user expectation for free form labels (symbols) and handwriting recognition (instead of virtual keyboard).
5. The grammar expressed by the users' gestures are not always complete and may not include a explicit selection step. For example, "Robot A should move to Area B" may be expressed as a drag starting near Robot A and ending in Area B with no explicit selection of the robot itself.

The next step in this research is to create the gesture set and link them to the actual control of robots. Through this next step, we will be able to discover if additional gestures or capabilities are needed, particularly for collaborative control of multiple robots.

ACKNOWLEDGEMENTS

This work was funded in part by the Microsoft Corporation. Thanks to Adam Norton.

REFERENCES

1. Benko, H., Ishak, E., and Feiner, S. Collaborative mixed reality visualization of an archaeological excavation. In *Third IEEE and ACM Int. Sym. on Mixed and Augmented Reality* (2004).
2. Bjørneseth, F., Dunlop, M., and Strand, J. Dynamic positioning systems: usability and interaction styles. In *Proc. of the 5th Nordic Conf. on Human-Computer Interaction: Building Bridges* (2008).
3. Cuyppers, T., Schneider-Barnes, J., Taelman, J., Luyten, K., and Bekaert, P. Eunomia: Toward a Framework for Multi-touch Information Displays in Public Spaces. In *Proc. of HCI* (2008).
4. Epps, J., Lichman, S., and Wu, M. A study of hand shape use in tabletop gesture interaction. In *Conf. on Human Factors in Computing Systems* (2006).
5. Ericsson, K. and Simon, H. Verbal reports as data. *Psychological Review* 87, 3 (1980), 215–251.
6. Esenther, A. and Ryall, K. RemoteDT: Support for Multi-Site Table Collaboration. In *Proc. Int. Conf. Collaboration Technologies* (2006).
7. Glaser, B. and Strauss, A. *The discovery of grounded theory*. Aldine de Gruyter New York, 1967.
8. Kato, J., Sakamoto, D., Inami, M., and Igarashi, T. Multi-touch interface for controlling multiple mobile robots. In *Proc. of the 27th Int. Conf. extended abstracts on Human factors in computing systems* (2009), 3443–3448.
9. Koskinen, H., Laarni, J., and Honkamaa, P. Hands-on the process control: users preferences and associations on hand movements. In *Conf. on Human Factors in Computing Systems* (2008).
10. Micire, M., Drury, J., Keyes, B., and Yanco, H. Multi-touch interaction for robot control. In *Proc. Int. Conf. on Intelligent user interfaces* (2009).
11. Nóbrega, R., Sabino, A., Rodrigues, A., and Correia, N. Flood Emergency Interaction and Visualization System. In *Proc. Int. Conf. on Visual Information Systems: Web-Based Visual Information Search and Management* (2008).
12. Regal, R. and Pacetti, D. Extreme C2 and Multi-Touch, Multi-User Collaborative User Interfaces. In *13th Int. Command and Control Research and Technology Sym.* (2008).
13. Rekimoto, J. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. In *Conf. on Human Factors in Computing Systems* (2002).
14. Sakamoto, D., Honda, K., Inami, M., and Igarashi, T. Sketch and run: a stroke-based interface for home robots. In *CHI '09: Proceedings of the 27th Int. Conf. on Human factors in computing systems*, ACM (2009), 197–200.
15. Schöning, J., Hecht, B., Raubal, M., Krüger, A., Marsh, M., and Rohs, M. Improving interaction with virtual globes through spatial thinking: helping users ask "why?". In *Proc. Int. Conf. on Intelligent User Interfaces* (2008).
16. Schöning, J. and Krüger, A. Multi-Modal Navigation through Spatial Information. In *Proc. Int. Conf. on GIScience* (2008).
17. Scotta, A., Pleizier, I., and Scholten, H. Tangible user interfaces in order to improve collaborative interactions and decision making. In *Proc. of Urban Data Management Sym.* (2006).
18. Seifried, T., Rendl, C., Perteneder, F., Leitner, J., Haller, M., and Stacey, D. CRISTAL, Control of Remotely Interfaced Systems using Touch-based Actions in Living spaces. (August 03 - 07, 2009).
19. Snyder, C. *Paper prototyping: The fast and easy way to design and refine user interfaces*. Morgan Kaufmann Pub, 2003.
20. Tse, E., Shen, C., Greenberg, S., and Forlines, C. Enabling interaction with single user applications through speech and gestures on a multi-user tabletop. In *Proc. of the Working Conf. on Advanced Visual Interfaces* (2006).
21. Washington, DC. Floyd D. Spence National Defense Authorization Act for Fiscal Year 2001. (2000).
22. Wobbrock, J., Morris, M., and Wilson, A. User-defined gestures for surface computing. In *Conf. on Human Factors in Computing Systems* (2009).
23. Wu, M., Shen, C., Ryall, K., Forlines, C., and Balakrishnan, R. Gesture registration, relaxation, and reuse for multi-point direct-touch surfaces. In *Proc. of IEEE TableTop-the Int. Workshop on Horizontal Interactive Human Computer Systems* (2006).