

4-7-2009

Comparing the Use of Tangible and Graphical Programming Languages for Informal Science Education

Michael S. Horn
Tufts University

Erin T. Solovey
Tufts University

R. Jordan Crouser
Tufts University, jcrouser@smith.edu

Robert J.K. Jacob
Tufts University

Follow this and additional works at: https://scholarworks.smith.edu/csc_facpubs

 Part of the [Computer Sciences Commons](#)

Recommended Citation

Horn, Michael S.; Solovey, Erin T.; Crouser, R. Jordan; and Jacob, Robert J.K., "Comparing the Use of Tangible and Graphical Programming Languages for Informal Science Education" (2009). Computer Science: Faculty Publications, Smith College, Northampton, MA.
https://scholarworks.smith.edu/csc_facpubs/146

This Conference Proceeding has been accepted for inclusion in Computer Science: Faculty Publications by an authorized administrator of Smith ScholarWorks. For more information, please contact scholarworks@smith.edu

Comparing the Use of Tangible and Graphical Programming Languages for Informal Science Education

Michael S. Horn, Erin T. Solovey, R. Jordan Crouser, Robert J.K. Jacob

161 College Ave.

Tufts University, Computer Science

Medford, MA 02155 USA

617-627-2225

{michael.horn, erin.solovey, jordan.crouser}@tufts.edu, jacob@cs.tufts.edu

ABSTRACT

Much of the work done in the field of tangible interaction has focused on creating tools for learning; however, in many cases, little evidence has been provided that tangible interfaces offer educational benefits compared to more conventional interaction techniques. In this paper, we present a study comparing the use of a tangible and a graphical interface as part of an interactive computer programming and robotics exhibit that we designed for the Boston Museum of Science. In this study, we have collected observations of 260 museum visitors and conducted interviews with 13 family groups. Our results show that visitors found the tangible and the graphical systems equally easy to understand. However, with the tangible interface, visitors were significantly more likely to try the exhibit and significantly more likely to actively participate in groups. In turn, we show that regardless of the condition, involving multiple active participants leads to significantly longer interaction times. Finally, we examine the role of children and adults in each condition and present evidence that children are more actively involved in the tangible condition, an effect that seems to be especially strong for girls.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

Author Keywords

Tern, Tangible User Interfaces, museums, informal science education, children, education, programming languages

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, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2009, April 4–9, 2009, Boston, MA, USA.

Copyright 2009 ACM 978-1-60558-246-7/08/04...\$5.00



Figure 1. The Robot Park exhibit is a permanent installation at the Boston Museum of Science. Visitors use our tangible programming interface, called Tern, to control a robot on display.

INTRODUCTION

An often-stated goal in modern education is that learning should be intrinsically motivated. In other words, students themselves should be the principle driving force behind their own academic explorations [2,26,28]. This applies both to formal learning settings such as classrooms and to informal learning settings such as after school programs and museums. In this paper, we describe a tangible user interface for computer programming that is part of a permanent exhibit at the Boston Museum of Science. We created this exhibit to provide a hands-on learning experience for children and to introduce concepts of computer programming and robotics. Our decision to use a tangible interface was motivated by the desire to create an engaging exhibit that encourages self-guided, collaborative interaction. To evaluate this design decision, we conducted a study comparing the use of tangible computer programming and mouse-based computer programming in this environment. The study involved observations of 260 museum visitors and interviews with 13 family groups. Our results show that the tangible language and the graphical

language are equally easy for visitors to understand. However, the tangible language offers several significant advantages from an informal science learning perspective. Among these, the tangible interface is more inviting and provides better support for active collaboration. Furthermore, the use of the tangible interface results in a more child-focused activity. In other words, children seem to be more actively involved and self-motivated. Parents, in turn, take on more of a supporting role and less of an instructional role. While there is still much work to be done in this area, we hope that this study will provide concrete evidence that tangible interaction can be an effective way to promote intrinsically motivated educational activities for children.



Figure 2. The Robot Park sign will light up and move when visitors program the robot to move to a special target area at the back of the exhibit platform.

Tangible Programming: Formal and Informal Education

Our work on tangible programming languages was originally inspired by informal observations of teachers in classrooms over a period of two years who were grappling with the challenge of incorporating computer-based learning activities into their curriculum. From these observations, we recognized that one of the most important and difficult responsibilities of a teacher is to maintain a positive and productive learning environment in the classroom. Modern desktop computers, designed primarily as single-user productivity tools for businesses, are inappropriate for many classroom applications, often making the teacher's job more difficult. In the case of computer programming, students are required to leave their normal work space, crowd around a limited number of desktop computers, and share single user input devices. In addition, teachers are often apprehensive about the use of technology in their classrooms. Not only are they concerned about what exactly students are doing behind the screens of multi-media computers connected to the Internet, but teachers may also feel a sense of loss of control and self-doubt about their own proficiency with technology [1]. Our initial prototype tangible programming language [12] sought to address some of these issues and improve computer programming activities in the classroom for students and teachers alike.

This work led to discussions and an eventual partnership with program directors at the Museum of Science who were interested in creating hands-on computer programming and robotics activities. The result of this partnership was the creation of a permanent exhibit in Cahners ComputerPlace at the Museum [13]. This exhibit has been open to the public since October 2007, and, in its first year, it was used by approximately 20,000 people.

In informal science learning settings, educational priorities are often different from those of classrooms. Without a teacher or curriculum requirements to guide activities, exhibits are designed to be self-directed and to allow visitors to create knowledge through hands-on exploration [2, 17, 43]. There is less concern for curriculum standards or students' performance on tests and quizzes. In a sense, helping to shift children's attitudes and preconceptions is at least as important as conveying high-level science and technology concepts. In addition, most people visit museums in social groups as part of family visits or school field trips [35]. While many computer and mouse-based exhibits may be engaging to one person, they are often detrimental to the social group as a whole [10, 14]. Despite these differences, many of the design considerations for tangible programming in classroom settings such as cost, durability, simplicity, apprehendability, and robustness, are equally important in museum settings [2, 16, 40].

Exhibit Overview

The Robot Park exhibit allows museum visitors to control an iRobot Create™ robot by creating computer programs using a tangible programming language called *Tern* that we developed [12]. Programs created with *Tern* consist of chains of wooden blocks shaped like jigsaw puzzle pieces (Figure 5). These blocks represent actions for the robot to perform, such as SHAKE, BEEP, or TURN LEFT; control-flow structures such as a WHILE loop and a REPEAT loop; and robot sensor values such as a bump sensor, and an infrared light detector. Visitors press a button to compile their programs, which are converted into digital instructions using a reliable, low-cost computer vision system that we developed [11]. Visitors' programs are then transmitted wirelessly to the robot through a Bluetooth connection. The robot, in turn, immediately begins to act out its instructions. The implementation details for the exhibit were described in [13].



Figure 3. Visitors' programs are sent to an iRobot Create™ robot through a wireless Bluetooth connection. The large red horseshoe magnet on the robot activates the Robot Park sign shown in Figure 2.

To help engage visitors, the exhibit includes a built-in challenge activity. By programming the robot to drive to a

special target at the back of the robot platform, visitors can cause the *Robot Park* sign (Figure 2) to light up and move.

BACKGROUND

Educational Programming Languages

As computers play a larger and larger role in our everyday lives, familiarity with computers and programming is becoming essential, and many researchers are exploring methods to introduce computer programming to children [20]. Research has shown that learning how to program computers can have a positive and measurable effect on children's achievement, not only in math and science, but also in language skills, creativity, and social-emotional interaction [3,5,9]. Of course, the decades of research involving computer programming in schools is diverse, and much depends on the age of the students, the context of the computer programming activities, and the ways in which the activities are integrated in with the broader curriculum [5].

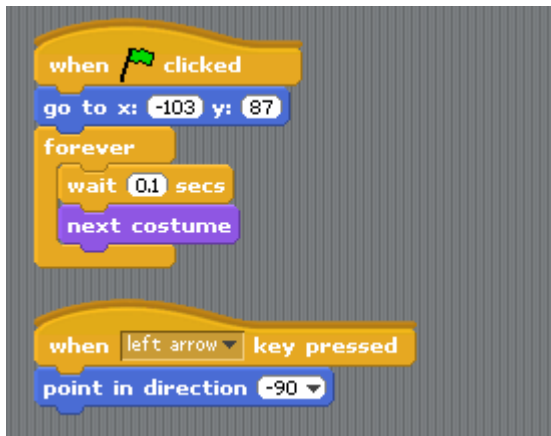


Figure 4. A screen shot from the Scratch programming language from the Lifelong Kindergarten group at the MIT Media Lab [29]. Scratch is one of several recent educational programming languages to adopt a puzzle piece graphical metaphor.

Since the 1960s a large number of programming languages and systems targeted at novice users have been created [20]. Notable recent languages include PicoBlocks, Scratch [29], Alice [6], and Storytelling Alice [21]. Many recent languages have also adopted a puzzle piece metaphor, whereby programs are constructed by connecting interlocking visual elements. Tern follows this metaphor as well, although in our case the puzzle pieces are physical blocks rather than icons on a screen.

Tangible Interfaces for Education

Research in tangible user interfaces (TUIs) has expanded our definition of what it means to interact with computers. Much of the research conducted with tangible user interfaces has focused on education [25]. Perhaps this is not surprising given that many of the benefits of moving interfaces into the physical world seem especially beneficial for children in the classroom. Jacob et al. propose that

tangible interfaces have potential advantages for users because they build directly on existing knowledge and experience from the real world (such as an understanding of *naïve physics*, *body awareness and skills*, and *social awareness and skills*) [19]. However, Marshall points out a general lack of empirical evidence supporting the value of tangible interfaces for educational use [22], although some recent studies have sought to begin building both theoretical and empirical foundations in this area [27,31,37,42,44]. As an example, Stringer et al. report on a tangible interface called Webkits [37] that was the result of a series of prototypes developed through ethnographic work in classrooms. This study documents how the use of tangible interfaces in classrooms can augment curriculum and facilitate group work. Here, we add to this effort, providing both quantitative and qualitative evidence of benefits of tangible interaction in informal education settings.

Tangible Programming

Tangible programming is an idea that has been around since at least the 1970s when Radia Perlman created the Slot Machine device to allow young children to create physical Logo programs. Since that time, a variety of tangible programming languages have been created. Some involve *program by example* systems (such as StoryKits [24], Topobo [27], and Curlybot [8]), while others allow for the construction of physical algorithmic structures (such as AlgoBlocks [38], Digital Construction Set [23], Electronic Blocks [41], and Tern [12]). Researchers have begun to explore the exciting potentials of programming in and with the physical world. Some ideas that have been generated include the blending of physical space and digital programming [7, 24], robots that are also embodied algorithmic structures [33, 41], the incorporation of found or crafted materials into algorithmic expressions [36], or the integration of physical activity and play with programming [36, 32].



Figure 5. Our tangible programming interface, Tern, consists of a collection of wooden blocks shaped like jigsaw puzzle pieces.

While this work is pioneering, there is minimal evidence that tangible systems offer benefits compared to their onscreen counterparts. Some of these systems provide a unique programming experience that has no reasonable

graphical comparison. Others, however, could be compared to onscreen systems through controlled experiments, and we believe that the results of such experiments could offer useful direction to the field of tangible interaction as a whole.

COMPARATIVE STUDY

We conducted a between subjects study to compare the effectiveness of a tangible and a graphical programming interface for the Robot Park exhibit at the Boston Museum of Science. The study consisted of observations of museum visitors and an analysis of logs generated by the exhibit computer. In order to provide context for our quantitative data, we also interviewed 13 family groups who had used the exhibit. Based on our prior formative evaluation of this exhibit [13], we were interested in the following six questions.

Inviting: Does the type of interface affect how likely visitors are to interact with the exhibit?

Apprehensible: Does the type of interface affect whether or not visitors are able to develop an understanding of how the exhibit works? That is, are they able to figure out how to create programs and send them to the robot?

Active Collaborative: Does the type of interface affect how well visitors are able to interact with the exhibit in groups? Does the exhibit support simultaneous active participants?

Engaging: Does the type of interface affect how long visitors interact with the exhibit?

Programs: Does the type of interface affect either the number or the complexity of programs that visitors create to control the robot?

Child-Focused: Does the type of interface affect how children interact with the exhibit and with other members of their social group?

Method

During the evaluation, visitors used the exhibit without interacting with researchers and without help from the museum staff. Quantitative data involving visitor behavior were logged automatically by the exhibit computer as well as manually by the researchers. During periods of observation, researchers sat ten feet away from the exhibit and watched visitors' interactions with the exhibit. A sign was posted next to the exhibit, notifying visitors that the exhibit was being observed. For this study, we were interested in observing *family groups*, which we define as groups of individuals consisting of at least one child and one adult who visit the museum together. To increase our chances of observing family groups, we conducted all of our observations over a period of three weeks on weekend days. Visitors were not recruited to participate in the study; rather, participants were people who happened to visit

Cahners ComputerPlace at the Museum on one of our observation days.

Conditions

We observed museum visitors in two independent conditions, tangible and graphical. In the tangible condition (TUI), visitors interacted with the exhibit using the Tern wooden block interface described above. In the graphical condition (GUI), we replaced the wooden blocks with a single standard, two-button computer mouse. To allow for mouse-based programming in the GUI condition, we created a comparable visual programming language (Figure 6). We were careful to make the visual and the tangible languages as similar as possible, and we offered an identical set of blocks in the two versions. In addition, to help ensure an intuitive graphical language, we modeled our mouse-based interaction conventions on the popular Scratch programming language [29].

All other aspects of the physical exhibit installation remained the same. We set up only one interface for visitors to use on a given observation day, and we alternated conditions on subsequent days. Furthermore, the two researchers who collected data for the study spent roughly equal amounts of time observing each condition.

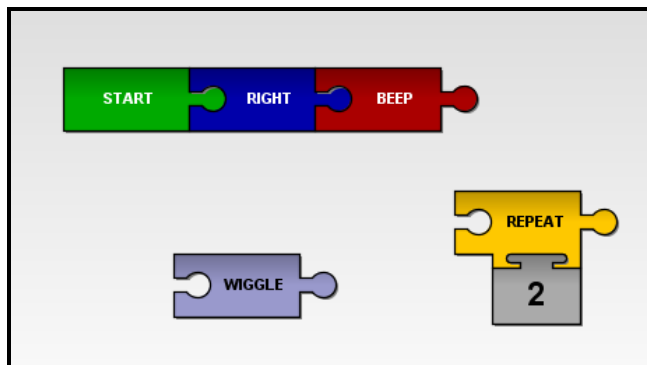


Figure 6. We created a graphical programming language equivalent to Tern using the same jigsaw puzzle metaphor and an identical set of blocks.

We chose to compare the tangible language to a system with a single mouse because this still seems to be the predominant form of interaction in schools, after school programs, and museums. In the future, we would like to broaden our comparison to include multi-touch or multi-mice systems. For now, we will note areas where we think that alternate screen-based systems might be advantageous in our discussion of the results.

Participants

We observed a total of 260 individuals at the Museum of Science (108 for the GUI condition and 152 for the TUI condition). Of these, 104 of the participants were children (47 for the TUI condition and 57 for the GUI condition). We defined a child as an individual 16 years old or younger. However, for these observations we did not interview visitors, so our participant ages are estimates.

Based on these estimates, observed children were between 2 and 16 years old (46 were approximately 8 years old or younger; 34 were between the ages of 9 and 12; and 24 were between the ages of 13 and 16). As we mention above, we were interested in observing family groups. In the GUI condition there were 25 total groups, 16 of which contained at least one parent and one child. In the TUI condition, there were 39 total groups, 18 of which contained at least one parent and one child.

For family group interviews, we recruited thirteen child-parent pairs. Of the children, six were girls and seven were boys. The ages ranged from 5-16 years of age, with an average age of 9.

Quantitative Data

To measure the *inviting* quality of the exhibit, we kept a tally of the number of people who noticed (looked or glanced at) the exhibit while within a five foot radius of the installation. Of the people who noticed the exhibit, we recorded the number of people who touched the exhibit with their hands. The time that a visitor first touched the exhibit was recorded as the start of a session. Session data were recorded on a per-group basis.

To measure *apprehendability*, we noted whether or not a social group was able to develop an understanding of how the exhibit worked. In other words, did visitors understand that pressing the run button causes the robot to execute a program? For our purposes, programming the robot one time was not sufficient evidence of understanding. Instead, we required evidence that visitors were purposefully putting pieces together to create more than one program. We recognized that it might be possible for a visitor to understand how the exhibit works without compiling more than one program; however, for the purposes of comparison, we felt that it was more important to avoid false positives than false negatives.

To determine the extent to which the exhibit supports *collaboration*, we compared the number of active participants to the number of passive participants for each interaction session. An *active participant* is someone who touches or interacts with the exhibit in some way, while a *passive participant* is a visitor who simply observes other members of his or her social group using the exhibit. Visitors often switch between active and passive roles during an interaction session; however, for this study, any hands-on interaction during a session classified a participant as active. We recognize that collaboration is a complicated concept with many shades of meaning. Our measure is not comprehensive, but we feel the results are still worth noting.

To measure *engagement*, we recorded the duration of each interaction session. This was recorded as the time the first group member started interacting with the exhibit to the time that the last group member left the exhibit. This method is based on prior studies of engagement with

interactive elements in museums [17]. Like collaboration, we recognize that session length is a narrow definition of engagement, a phenomenon that has intellectual, physical, emotional, and social aspects; however, museum research has shown that session length correlates well with physical, intellectual, and social aspects of engagement [17].

To analyze visitor *computer programs* we programmed the exhibit computer to log every program compiled by participants. This was in the form of a screen shot for the GUI condition and an image captured by the digital camera for the TUI condition (Figure 9). In analyzing these logs we were interested in three questions: does the type of interface affect (1) the number of programs created by visitors per session; (2) the length of programs created; and (3) the complexity of programs created?

To determine the extent to which the exhibit is *child-focused*, we noted which member of each family group initiated an interaction session and whether that person was a parent, a child, or a parent and child together. We also analyzed the data for differences between children and adults for each of the other measures described above.

Qualitative Data

After collecting the quantitative data, we returned to the museum on several additional weekend days to conduct interviews with family groups. In all, we interviewed thirteen parent/child pairs who had used the exhibit for more than five minutes. In some cases there was more than one child in a family group that we interviewed. In these cases, we interviewed the child who seemed most involved with the activity. After initially agreeing to participate, each family group was given time to read and sign a consent form. The interviews took about fifteen minutes to complete. Of the children who participated in the interviews, twelve had never programmed before, and one had programmed with LEGO Mindstorms at a camp. All reported that they used a computer at least once a week at school or at home. Four of the groups had used the GUI condition and nine had used the TUI condition prior to participating in the interview.

In the interviews, we gathered background information and impressions from the visitors, and we asked them to work together to complete a few simple programming tasks. We also introduced the visitors to the interface (either GUI or TUI) that they had not been using before the interview. After trying out the second interface, we asked the children to fill out a short questionnaire about the two interfaces. Eleven out of the thirteen children answered these questions, while the two remaining children left before finishing the questionnaire. For children who could not read, we read the questions out loud.

RESULTS

Inviting

Based on our prior formative evaluation of the exhibit, we expected the tangible interface to be highly inviting. We

hoped that the use of familiar objects (such as wooden blocks) would transform an unfamiliar and potentially intimidating activity like computer programming into an inviting and playful experience.

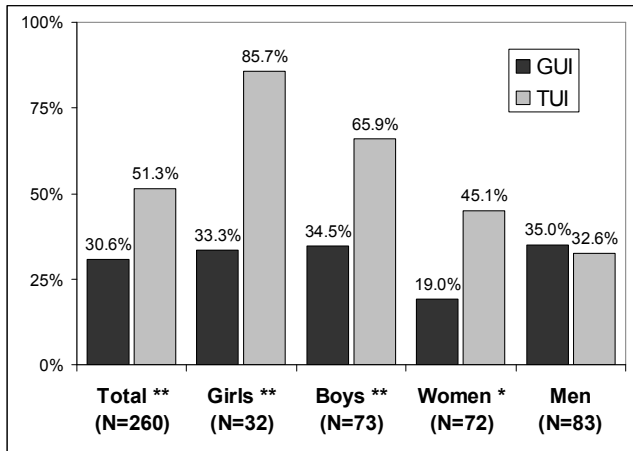


Figure 7. Percentage of visitors who interacted with the exhibit after noticing it (* $p < .05$, ** $p < .01$).

Our results, shown in Figure 7, indicate that the choice of interface matters a great deal. Overall, visitors were significantly more likely to try the exhibit with tangible blocks rather than with a mouse (based on a two-tailed z-test). This was especially true for children and for girls in particular. For the graphical system, 33.3% of girls who noticed the exhibit also tried it. This number rose to 85.7% when the tangible system was presented, an increase of over 50%.

To gain some insight into this effect, during the family group interviews, we asked the children which interface they thought was more fun. Seven of the visitors who were interviewed reported that the blocks were more fun than the mouse; one said that the mouse was more fun; and two reported that they were the same (1 subject had no response).

Apprehensible

Before conducting this study, our hypothesis was that the graphical condition would be easier for visitors to understand than the tangible condition. This was because we felt that, in general, visitors would be very familiar with graphical user interfaces. On the other hand, we thought that visitors would have a difficult time linking actions performed with wooden blocks to the resulting actions of the robot, especially since there are no electronic components in the wooden blocks or the programming surface.

Despite our expectations, the results of our study showed that there was no significant difference between the two conditions. Of the 25 groups that we observed in the graphical condition, 18 (72%) successfully developed an understanding of how the exhibit worked. In the tangible condition, 29 out of the 39 (74%) groups were successful.

Typically, visitors would give up if they were not able to figure out the exhibit within the first two minutes of interaction (only two groups overall spent more than two minutes before giving up).

Correspondingly, out of the family groups we interviewed, four reported that the blocks were easier to use and five reported that the mouse was easier to use (with one reporting that they were the same, and one not responding to this question).

Active Collaborative

Tangible interfaces are often claimed to improve support for collaborative interaction. We expected that this would be true for our tangible system as well, if for no other reason than the tangible interface consists of multiple input devices that can be manipulated independently, while the graphical condition consists of only a single input device.

For the purposes of this study, we define *active collaboration* as simultaneous active participation, and we measure it by comparing the number of active and passive participants in each group. The average number of active participants per group in the graphical condition was 1.32 (SD=0.48), while the average in the tangible condition was 2.0 (SD=1.05). This difference is statistically significant (one-tailed t-test, $p < 0.01$). Similarly, the ratio of active to passive participation in the GUI condition was 1.18, while the ratio in the TUI condition was 3.55. Figure 8 shows the average number of active participants in each condition by age and gender. The difference between conditions was especially large for children (based on a one-tailed z-test).

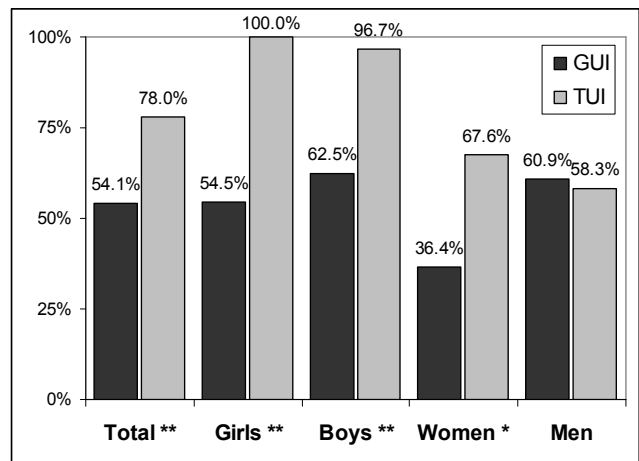


Figure 8. Percentage of participants who were active (out of active and passive participants combined) in each condition (* $p < .05$, ** $p < .01$).

This result was reflected in our interviews with family groups. More children said that they would prefer to use the blocks for working with friends or family (7 blocks, 3 mouse, 1 no response). However, the responses were evenly split when asked which interface would be preferable for working alone (4 answered the blocks, 4 the mouse, 1 the same, 1 had no opinion, 1 didn't answer).

Engaging

We measured engagement in terms of the overall session length for each group. The average session length was 4.02 minutes for the graphical condition (SD=4.87) and 5.03 minutes for the tangible condition (SD=5.84). The variance for session length was high in both conditions, and a two-tailed t-test showed no significant difference between the two means. To put these results into perspective, recent research on engagement in science museums has found average session lengths of 3.3 minutes for successful, engaging exhibits [17].

| | GUI (N = 25) | TUI (N = 39) |
|------------------------------|-------------------|--------------------|
| Single Active Participant | 2.55 (SD=2.09) | 1.77 (SD=1.82) |
| Multiple Active Participants | 7.13 (SD=7.42) | 6.65; (SD=6.47) |

Table 1. Average session lengths for groups with a single active participant and groups with multiple active participants for both conditions. The differences within both conditions are statistically significant ($p < 0.05$).

We did, however, observe a significant difference in average session length between groups with only a single active participant compared to groups with multiple active participants (see Table 1). This suggests that for engagement, the type of interface might be less important than actively involving multiple participants. Thus, multiple mice (e.g. [34]), large touch-enabled displays (e.g. [30]), and tangible blocks might all be equally effective in engaging visitors because they all provide good support for multiple active participants.

Computer Programs

Using the exhibit computer logs, we analyzed the actual computer programs that visitors created during the first two days of observations (one day for each condition). This included 13 groups in the GUI condition and 20 groups in the TUI condition. Prior to conducting the study we hypothesized that visitors would create longer and more complex programs with the tangible interface because we felt that it was easier to manipulate and rearrange physical blocks than to manipulate icons on a computer screen with a mouse. However, despite this hypothesis, we found no significant differences between the conditions.

Number of Programs

The average number of programs created per group in the GUI condition was 4.85 (SD=7.09). The average in the TUI condition was 7.19 (SD=5.72). This difference was not significant. These results include groups that compiled no programs. If we omit the groups with zero programs the averages were closer together (7.88 for GUI vs. 8.39 for TUI).



Figure 9. An example screen shot from a program created in the tangible condition. The digital photograph has been cropped for clarity. The field of view of the digital camera includes the entire programming surface.

Program Length

We measured program length in terms of the number of blocks (or statements) included. The average program length in the GUI condition was 8.06 statements (SD=3.04), while the average program length in the TUI condition was 9.13 statements (SD=5.71). Again, this difference was not statistically significant.

Program Complexity

To measure complexity, we assigned a complexity score from 1-4 to each program compiled. Programs with a score of 1 contained only action blocks (no flow of control statements). Programs with a score of 2 contained at least one control block but no parameter values, while programs with a score of 3 contained a single control flow block with a parameter value. Finally, programs with a score of 4 contained multiple control flow blocks with parameters. We found no significant differences in complexity levels between conditions.

Child-Focused

Finally, we examined the roles of children and adults under both conditions and found evidence that the tangible system seems to encourage more active participation on the part of children. For our measure of *inviting*, we note that children were much more likely to try the tangible interface, whereas for adult men, there was no significant difference between the conditions (Figure 7). We also noticed an increase in the percentage of *active participation* (compared to *passive participation*) that was especially large for children (Figure 8). This data combined with qualitative observations suggests that parents tended to take on more of a supporting role (offering advice and suggestions from the sidelines) rather than an instructional role. For example, during one of the interviews in the GUI condition, a seven-year old girl worked with her father to complete the programming tasks. The father had control of the mouse during this session. When we introduced them to the tangible interface, the girl

immediately took over the job of creating programs, while her father looked on and offered advice.

Finally, considering data from family group sessions (omitting adult-only or child-only groups), 17 out of 18 sessions in the TUI condition were either initiated by a child or initiated by a parent and child simultaneously. In the GUI condition, on the other hand, 11 out of 16 sessions were initiated by a child or parent and child simultaneously. The remaining sessions were initiated by an adult. This difference is statistically significant using a two-tailed z-test ($p < 0.05$).

DISCUSSION

Overall, on the six measures, the tangible interface was more *inviting*, more supportive of *active collaboration*, and more *child-focused* than the mouse-based interface. We also found that the tangible and graphical interfaces were equivalently *apprehensible* and *engaging*, and the resulting visitor *programs* were not significantly different. For the measure of engagement, we noted that regardless of the condition, session times were longer when more than one participant was actively involved. This suggests that other types of interfaces designed to support collaboration (such as multi-touch or multi-mouse systems) might be equally effective in encouraging engagement.

Girls and Programming

In this study, we observed that girls were significantly more likely to use the exhibit in the tangible condition than in the graphical, mouse-based condition. It has been noted that as technology becomes more pervasive in our society, it is important that it supports all members of society [1,21]. Although women and girls currently are under-represented in the field of computer science [1, 39], Kelleher et al. [21] point out that performance and interest in programming has been shown to depend on previous experience programming and time spent programming, rather than on gender. Thus, researchers and educators have been developing methods and tools to motivate girls to learn about computers and to provide them with positive learning experiences with programming (for example Storytelling Alice [21] and Lillypad Arduino [4]). Furthermore, Inkpen et al. demonstrate the importance of technology's ability to support collaboration to engage girls. In this case the technology was multiple mice connected to a single computer. We feel that our results provide strong evidence that tangible programming languages might be another approach to create more gender-neutral computer programming activities for both formal and informal education.

Active Collaboration and Sharability

The tangible condition proved to be more supportive of active collaboration than the graphical, mouse-based condition. Hornecker, Marshall, and Rogers outline rudimentary components of *sharable interfaces* [15] that identify many of the key features that we felt contributed to

the success of the tangible interface for our exhibit. A sharable interface, in this case, refers to the support for multiple, co-present collaborators around a common task, and it includes components of *entry points* and *access points*. Entry points entice people to interact with a system. In our case, the wooden blocks seemed much more effective for this purpose than a computer mouse. This is perhaps because the blocks are familiar and playful objects. But we suspect that this is only part of the story. It could also be that the blocks are non-threatening objects presented in a novel and curious context.

The tangible interface was much better at luring people into socially-motivated interaction. With the graphical interface, on the other hand, an observer might feel equally motivated to the activity but also may feel unable to do so without taking the mouse away from the active user. Hornecker et al. [15] describe the *honey-pot effect* as a social entry point, whereby passive observers are drawn into active participation by watching their family and friends interact with the exhibit. The honey-pot effect, however, is most effective when coupled with easy access to the system for multiple participants as in the tangible interface.

Sharable interfaces also provide access points, which can refer to both perceptual access and manipulative access [15]. The tangible interface is clearly superior in terms of manipulative access because it offers many objects that can be independently manipulated in a meaningful way by multiple participants. We also suspect that the tangible interface provides better perceptual access as well. This is simply because the *display space* that bounds the interaction is larger and more visible. The tangible interface is manipulated on a large horizontal table top that can be viewed at any angle, while the graphical interface is displayed on a vertical computer screen that can only be viewed from limited angles. Thus, it is both easier for one collaborating active participant to understand the actions of another, and for passive participants to understand the actions of active participants.

CONCLUSION AND FUTURE WORK

In this paper we presented a study comparing tangible and graphical interaction for use with a science museum exhibit. Our results provide concrete evidence that thoughtfully designed tangible interfaces can offer several significant advantages over the standard single-mouse graphical interfaces in the context of informal science education. Among these advantages, tangible interfaces can be more inviting and more conducive to collaborative interaction. Furthermore, in this case, the tangible interface is better at encouraging children to take an active role in exploring and learning, an effect that seems especially strong for girls.

Our work on tangible programming is ongoing. One limitation of our results is that our comparison included only a tangible interface and a single-mouse graphical interface. Thus, we are interested in comparing tangible interfaces to a broader spectrum of input devices for use in

informal science learning such as multi-touch surfaces or multi-mice systems. Our results on engagement suggest that these alternative multi-user devices might have similar benefits to tangible interaction. In addition, we are beginning a three year research project, funded by the National Science Foundation, on the use of tangible programming languages in early elementary school classrooms. The primary focus of this work will be on the developmental appropriateness of programming concepts for younger children in light of new, more appropriate technology.

ACKNOWLEDGMENTS

We thank our collaborators at the Tufts University Developmental Technologies Research Group and the Human-Computer Interaction Laboratory including Rachael Fein, Audrey Girouard, Leanne Hirshfield, Emily Lin, and Orit Shaer. Angela Chang and Jamie Zigelbaum from the MIT Media Lab also offered advice and support. Work at the Boston Museum of Science was the result of collaboration with Taleen Agulian, Lucy Kirshner, Dan Noren, and Bill Rogers. Mary Murray from the Massachusetts College of Art and Design provided refreshing insights and designs for our tangible programming language, and Daniel Ozick contributed software expertise to the design of the iRobot interface. Finally, we thank the National Science Foundation for support of this research (NSF Grants IIS-0414389 and DRL-0735657). Any opinions, findings, and conclusions or recommendations expressed in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

1. AAUW, *Tech-Savvy: Educating Girls in the New Computer Age*. American Association of University Women Educational Foundation, Washington, DC, USA, 2000.
2. Allen, S. Designs for Learning: Studying Science Museum Exhibits That Do More Than Entertain. *Science Education*, 88 (S1), Wiley Periodicals (2004), S17-S33.
3. Bers, M. (2008). *Blocks to Robots: Learning with Technology in the Early Childhood Classroom*. Teachers College Press, NY, NY.
4. Buechley, L., Eisenberg, M., Catchen, J. and Crockett, A. The LilyPad Arduino: Using Computational Textiles to Investigate Engagement, Aesthetics, and Diversity in Computer Science Education. In *Proc. CHI 2008*, ACM Press (2008).
5. Clements, D.H. The future of educational computing research: the case of computer programming. *Information Technology in Childhood Education*, (1999), 147-179.
6. Conway, M., Pausch, R., Gossweiler, R., and Burnette, T. 1994. Alice: a rapid prototyping system for building virtual environments. In *Proc. of CHI '94*. ACM Press (1994), New York, NY, 295-296.
7. Fernaeus, Y. and Tholander, J. Finding design qualities in a tangible programming space. In *Proc. of CHI '06*. ACM Press (2006), New York, NY, 447-456.
8. Frei, P., Su, V., Mikhak, B., Ishii, H. Curlybot: designing a new class of computational toys. In *Proc. of CHI 2000*, ACM Press (2000), 129-136.
9. Haugland, S.W. The Effect of Computer Software on Preschool Children's Developmental Gains. *Journal of Computing in Childhood Education*, 3(1), (1992), 15-30.
10. Heath, C., vom Lehn, D., Osborne, J. Interaction and interactivities: collaboration and participation with computer-based exhibits. *Public Understanding of Science 14*, Sage Publications (2005), 19-101.
11. Horn, M. TopCode: Tangible Object Placement Codes. <http://hci.cs.tufts.edu/topcodes/>
12. Horn, M. and Jacob, R.J.K. Designing Tangible Programming Languages for Classroom Use. In *Proc. First International Conference on Tangible and Embedded Interaction TEI'07*, (2007).
13. Horn, M., Solovey, E. T., Jacob, R.J.K. Tangible Programming and Informal Science Learning: Making TUIs Work for Museums, In *Proc. IDC 2008 Conference on Interaction Design for Children* (2008).
14. Hornecker, E. and Buur, J. Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. In *Proc. of CHI 2006*, ACM Press (2006), 437-446.
15. Hornecker, E., Marshall, P., Rogers, Y. From entry to access -- how sharability comes about. In *Proc. Designing Pleasurable Products and Interfaces DPPI'07*, ACM Press (2007).
16. Hornecker, E. and Stifter, M. Learning from Interactive Museum Installations About Interaction Design for Public Settings. In *Proc. Australian Computer-Human Interaction Conference OZCHI'06*, (2006).
17. Humphrey, T. and Gutwill, J.P. *Fostering Active Prolonged Engagement: The art of creating APE exhibits*. Exploratorium, (2005).
18. Inkpen, K., Booth, K.S., Klawe, M., and Upitis, R. Playing Together Beats Playing Apart, Especially for Girls. In *Proc. Conference on Computer Support for Collaborative Learning CSCL'95*, Lawrence Erlbaum (1995), 177-181.
19. Jacob, R.J.K., Girouard, A., Hirshfield, L.M., Horn, M.S. Shaer, O., Solovey, E.T., and Zigelbaum, J. Reality-Based Interaction: A Framework for Post-WIMP Interfaces. In *Proc. CHI 2008*, ACM Press (2008).
20. Kelleher, C. and Pausch, R. 2005. Lowering the barriers to programming: A taxonomy of programming

- environments and languages for novice programmers. *ACM Comput. Surv.* 37, 2 (Jun. 2005), 83-137
21. Kelleher, C., Pausch, R., and Kiesler, S. Storytelling alice motivates middle school girls to learn computer programming. In *Proc. of CHI 2007*. ACM Press (2007).
 22. Marshall, P. Do tangible interfaces enhance learning? In *Proc. First international conference on tangible and embedded interaction TEI'07*, ACM Press (2007), 163-170.
 23. McNeerney, T.S. From turtles to tangible programming bricks: explorations in physical language design. *Personal and Ubiquitous Computing (2004) 8*, Springer-Verlag (2004), 326-337.
 24. Montemayor, J., Druin, A., Farber, A., Simms, S., Churaman, W., and D'Amour, A. Physical programming: designing tools for children to create physical interactive environments. In *Proc. CHI 2002*, ACM Press (2002), 299-306.
 25. O'Malley, C. and Stanton Fraser, D. Report 12: Literature review in learning with tangible technologies. Technical report, NESTA Futurelab (2004).
 26. Papert, S. *Mindstorms: Children, Computers, and Powerful Ideas*, Basic Books, (1980).
 27. Parkes, A., Raffle, H., and Ishii, H. Topobo in the wild: longitudinal evaluations of educators appropriating a tangible interface.
 28. Resnick, M., Bruckman, A., and Martin, F. Pianos not stereos: creating computational construction kits. *Interactions*, 3(6), (1996).
 29. Resnick, M. Sowing the seeds for a more creative society. *Learning and Leading with Technology*, International Society for Technology in Education (2007), 18-22.
 30. Rizzo, F. and Garzotto, F. "The Fire and The Mountain": Tangible and Social Interaction in a Museum Exhibition for Children. In *Proc. IDC '07*, ACM Press (2007).
 31. Rogers, Y., Scaife, M., Gabrielli, S., Smith, H., and Harris, E. A conceptual framework for mixed reality environments: designing novel learning activities for young children. *Presence*, 11(6), MIT Press (2002), 677-686.
 32. Scharf, F., Winkler, T., and Herczeg, M. Tangicons: Algorithmic Reasoning in a Collaborative Game for Children in Kindergarten and First Class. In *Proc. Interaction Design and Children IDC'08*. (2008).
 33. Schweikardt, E. and Gross, M.D. The robot is the program: interacting with roBlocks. In *Proc. Conference on Tangible and Embedded Interaction TEI'08*, (2008).
 34. Scott, S.D., Shoemaker, G.B.D., and Inkpen, K.M. Towards seamless support of natural collaborative interactions. In *Proc. Graphics Interface*. Montreal, Canada, (2000), 103-110.
 35. Serrell, B. *Exhibit Labels: An Interpretive Approach*. AltaMira Press (1996).
 36. Smith, A. Handcrafted physical syntax elements for illiterate children: initial concepts. In *Proc. Interaction Design and Children IDC'08*, ACM Press (2008).
 37. Stringer, M., Rode, J.A., Toye, E.F., Blackwell, A.F., and Simpson, A.R. The Webkit Tangible User Interface: A Case Study of Iterative Prototyping. *Pervasive Computing*, 4(4), IEEE Computer Society (2005), 35-41.
 38. Suzuki, H. and Kato, H. Interaction-level support for collaborative learning: Algoblock—an open programming language. In *Proc. CSCL '95*, Lawrence Erlbaum (1995).
 39. Vegso, J. Drop in CS Bachelor's Degree Production. *Computing Research News*, 18(2), (2006).
 40. West, R. The Economics of Interactivity. *Curator* 47(2), AltaMira Press (2004), 213-223.
 41. Wyeth, P. How young children learn to program with sensor, action, and logic blocks. *Journal of the Learning Sciences*, 17(4), Routledge (2008), 517-550.
 42. Xie, L., Antle, A.N., and Motamedi, N. Are tangibles more fun? Comparing children's enjoyment and engagement using physical, graphical and tangible user interfaces. In *Proc. Tangible and Embedded Interaction TEI'08*, (2008).
 43. Zheng, S., Bromage, A. Adam, M. and Scrivener, S.A. Surprising Creativity: A Cognitive Framework for Interactive Exhibits Designed for Children. In *Proc. Creativity & Cognition*, ACM Press (2007), 17-26.
 44. Zuckerman, O., Grotzer, T. and Leahy, K., Flow blocks as a conceptual bridge between understanding the structure and behavior of a complex causal system. In *Proc. ICLS'06*, (2006).