Mid-Air Text Input Techniques for Very Large Wall Displays

Garth Shoemaker, Leah Findlater, Jessica Q. Dawson, and Kellogg S. Booth

Department of Computer Science
University of British Columbia
2366 Main Mall, Vancouver, BC, Canada
{garths, lkf, ksbooth}@cs.ubc.ca

ABSTRACT

Traditional text input modalities, namely keyboards, are often not appropriate for use when standing in front of very large wall displays. Direct interaction techniques, such as handwriting, are better, but are not well suited to situations where users are not in close physical proximity to the display. We discuss the potential of *mid-air* interaction techniques for text input on very large wall displays, and introduce two factors, *distance-dependence* and *visibility-dependence*, which are useful for segmenting the design space of mid-air techniques. We then describe three techniques that were designed with the goal of exploring the design space, and present a comparative evaluation of those techniques. Questions raised by the evaluation were investigated further in a second evaluation focusing on distance-dependence. The two factors of distance- and visibility-dependence can guide the design of future text input techniques, and our results suggest that distance-independent techniques may be best for use with very large wall displays.

KEYWORDS: text input, wall displays, interaction techniques

INDEX TERMS: H.5.2 [User Interfaces]: Input devices and strategies

1 INTRODUCTION

Designers of very large interactive display environments (Figure 1) have yet to establish a standard model for interaction. Traditional devices (keyboard and mouse) and onscreen metaphors (windows, icons, menus, and pointers) long established for desktop use are often not ideal for the form factors of new wall and table displays. An ongoing effort in the research community has been to investigate different interaction possibilities, with the goal of designing better methods for input and manipulation. The development of these methods will play a critical role in the evolution of large displays from research systems to universally adopted tools.

In this paper we consider English language text input techniques for very large wall displays. Text input is one of the primitives of interaction identified by Foley et al. [6], and must be supported by any general purpose interactive system, even if that system is not used for lengthy uninterrupted text input. Researchers recognize the limitations of keyboard input, the most significant being that keyboards are designed for use while sitting stationary at a desk or a table, but users of wall displays are often standing or walking. Alternative approaches have been investigated, including pen-based text input, where a user writes as on a whiteboard. This method, while natural, has the significant drawback that a user must be within physical reach of the display surface in order to write. Users of large wall displays are often not within physical reach of the display surface [21, 23].

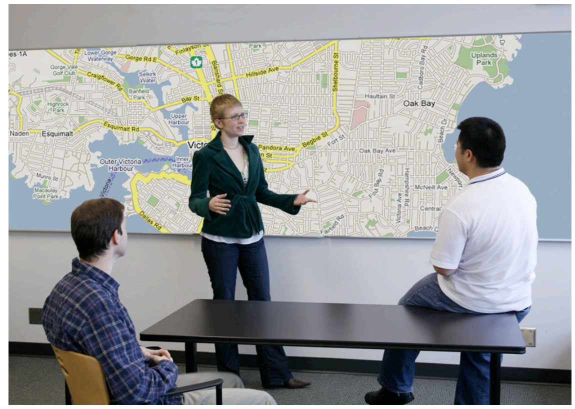

Figure 1. A mockup of collaboration around a large wall display.

To address this problem we identified and investigated the under-explored design space of *mid-air* text input techniques, those that can be used by a standing, mobile user. In this context, techniques that allow for input independent of visibility or distance to the display might be particularly important. Thus, we segment this design space along the axes of *visibility-dependence* and *distance-dependence*, both of which are useful for categorizing candidate interaction techniques.

We developed three text input techniques that differ in terms of distance- and visibility-dependence: Circle, QWERTY, and Cube. For each technique the user manipulates a handheld device in mid-air, at some distance from the display surface. We conducted a controlled experiment to compare the techniques in terms of speed and accuracy for a text input task. Results suggest that the QWERTY technique is suitable for adoption, and that the other two techniques have potential for future development but need improvements to be competitive with the QWERTY technique.

A second, follow-up experiment was performed with the goal of developing a deeper understanding of the distance-dependence factor of interaction techniques. We evaluated Circle and QWERTY techniques, which we hypothesized would be distance-independent and –dependent, respectively, at two different distances. Results showed that the factor of distance had a significant impact on performance. This highlights the need for further work on distance-independent techniques.

The primary contributions of this paper are three-fold. First, we investigate the design space of mid-air text input techniques for large wall displays, and identify important factors segmenting this space. Second, we report a comparative analysis of three candidate text input techniques representative of the design space, and we provide conclusions regarding the usefulness of those techniques. Third, we report a second experiment that highlights the importance of distance-independence when designing interaction techniques for large wall displays.

Graphics Interface Conference 2009
25-27 May, Kelowna, British Columbia, Canada

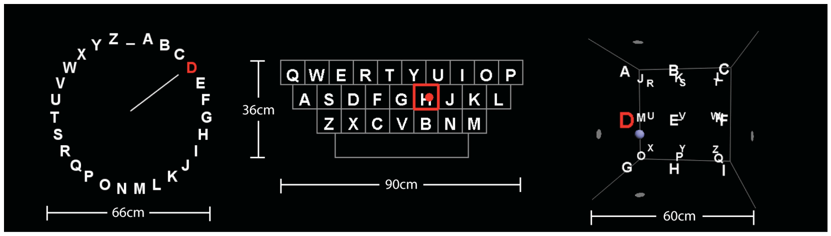

Figure 2. Text input interaction techniques as used in Experiment 1. From left to right: Circle Keyboard, QWERTY Keyboard, Cube Keyboard.

2 Related Work

Relevant related work falls into the categories of large display interaction and text input. The large display work serves to define the special requirements of these environments, while the substantial body of text input research serves as a foundation for text input considerations in general.

2.1 Large Display Interaction

Large physical surfaces can provide valuable inspiration in the design of large electronic surfaces. Traditional whiteboards are widespread in modern workplaces, and have been shown to be important tools for collaborative information sharing and visualization [4]. It has been argued that the 19th century replacement of personal slates by large shared blackboards was an important innovation in the domain of education [3]. This, combined with quantitative evidence that large display surfaces can enhance performance [5], makes a strong argument that large interactive wall displays have a useful role to play.

Consistent with these conclusions, recent work on Shadow Reaching [20], Soap [2], and freehand pointing [24] has begun looking at mid-air techniques that allow for relatively unconstrained movement on the part of the user. The Shadow Reaching work demonstrates the use of full-body shadows for interaction, whereas the findings of the work on Soap and freehand pointing are about device-based and device-less pointing, respectively.

2.2 Text Input

Text input techniques in a variety of domains can serve as inspiration for the design of techniques for large wall displays.

2.2.1 Large Wall Display Text Input

Large wall displays are physically similar to whiteboards, so there is an understandable tendency for designers to employ pen-based handwriting for text input, as was done in Flatland [16] and Tivoli [18]. Handwriting is not always the best choice for text input, however. Writing input speed, known to be at best around 20 wpm [1], is worse than many mechanized techniques. Furthermore, recognition algorithms for the purpose of digitizing and archiving written data are error-prone. As a result of these limitations, researchers have considered alternatives to handwriting input. Pavlovych and Stuerzlinger evaluated text entry performance using direct touch input with a variety of keyboard layouts [17]. They found that a standard QWERTY layout resulted in a mean text entry rate of 17.6 wpm, which is roughly comparable to hand writing performance. Magerkurth and Stenzel took a different approach, supporting text input from a small personal input device, with feedback provided on a shared display [14].

2.2.2 Small Display Text Input

Small handheld devices, such as phones and personal digital assistants (and hybrids of the two) are increasingly being used for text-heavy tasks, such as writing email and web browsing. Techniques for small displays are relevant to large displays because users of both systems are often standing and moving, and successful approaches in one domain may apply to the other. But it is difficult to incorporate a full QWERTY layout keyboard into a small display. As an alternative to this layout, many small display devices employ disambiguation techniques such as T9 or multitap. T9 and multitap are widely deployed, but performance by any except highly expert users is poor when compared to standard keyboard input. Other disambiguation approaches have been explored. TiltText, as an example, uses tilt information from an accelerometer to filter how characters are selected by button presses on the keypad [27].

More recently, stylus and touch-based text input approaches have gained popularity on small mobile devices. In this context, MacKenzie and Zhang showed that soft keyboards can provide impressive performance with either standard QWERTY or other optimized character layouts [13]. Similarly, a number of 2D gesture techniques have been shown to hold promise [15, 22, 28].

2.2.3 Other Text Input Approaches

Many innovative methods for text input are not easily categorized as being either small or large display techniques. Dasher, one such technique, makes use of a continuous gesture through a 2D landscape of characters generated by a predictive model [26]. Another gestural technique developed by Liu et al. [9] allows users to input text by tracing letters in mid-air with their fingers. There has also been work done on device-specific text input techniques. Input using game controllers has been explored [7], as have techniques that employ chording keyboards [8].

3 Design Space

As we have already noted, many large display text input techniques require the user to be within physical reach of the display, but in many cases users of such displays are frequently at a distance. In contrast, techniques that allow for free body movement within the space near, but not at, the display while interacting have received relatively little attention. These are termed *mid-air* techniques. They can be used while standing or walking, and at any practical distance from the display.

We have identified two properties related to mid-air techniques that are important to consider. The first property is *distance-dependence*. The physical action of a *distance-dependent* technique changes as the distance between the user and the display changes, whereas action of a *distance-independent* technique is invariant with distance. As an example, pointing using a ray-casting model is a *distance-dependent* technique: as the user moves farther from the display, the effect of pointing motions is magnified. We hypothesized that large displays will benefit from the development of *distance-independent* techniques, because these techniques will not constrain the movement of users within space.

The second property is *visibility-dependence*. A *visibility-dependent* technique requires that the user refer to visible feedback during use, whereas a *visibility-independent* technique does not. For example, touch-typing is *visibility-independent*, but

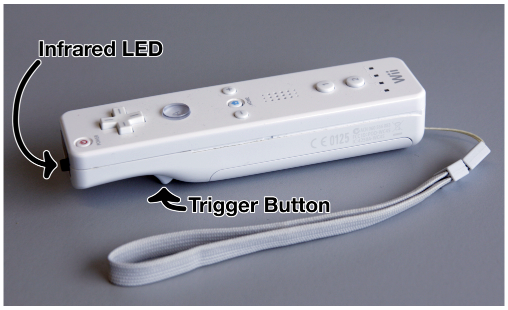

Figure 3. Wiimote altered with infrared LED for 3D position tracking.

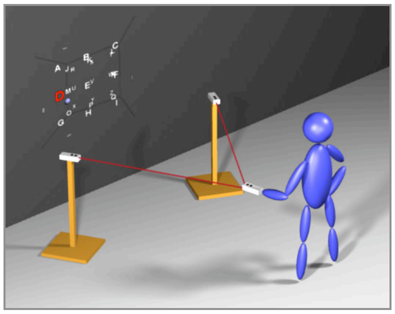

Figure 4. Triangulating position of hand-held Wiimote using 2 fixed Wiimotes on stands. Red lines indicate vectors from detected IR light source location to fixed Wiimotes.

input on touch screens (such as an iPhone) is often visibility-dependent because there is no tactile feedback available to guide input. We assume that all techniques provide visual feedback through the display of the characters that have been entered. We are concerned here only with visual feedback provided during the entry action itself. We hypothesize that large display use will benefit from the development of *visibility-independent* techniques because these will allow the user to focus on the data being manipulated, rather than on the mechanics of the interaction.

The remainder of this paper describes our exploration of the design space of *mid-air* text input techniques for large wall displays, with a special focus on *distance-dependence*.

4 Candidate Interaction Techniques

Based on the properties of distance- and visibility-dependence, we designed three text input techniques for very large wall displays (shown in Figure 2). These techniques represent contrasting points in the design space: *Circle* is distance-independent but visibility-dependent, *QWERTY* is both distance- and visibility-dependent, and *Cube* is distance-independent and visibility-independent. All three techniques employ a handheld button-equipped pointing device for input, and provide visual feedback on the large display. We used Wiimotes for our implementations, but other devices such as magnetically tracked devices or laser pointers could be used for at least some of these techniques.

4.1 Circle Keyboard Technique

In the Circle technique, letters of the alphabet are shown in a circular arrangement (Figure 2). A pointer line radiating from the center of the circle indicates the currently highlighted letter. Using the handheld device, the user moves the pointer to highlight the desired character before pressing a button to select that character. The angle of the pointer is defined by intersecting a ray cast from the handheld-device with the display surface. The pointer line radiates from the center of the circle towards the point of intersection.

Because character selection is based on angle, input response is invariant with user distance from the display. Thus the technique is distance-independent. On the other hand, the technique is most certainly visibility-dependent. This is because of the small angle available to individual characters when a reasonable-sized alphabet is displayed. For 26 letters and 1 space evenly distributed around the circle each character spans only 13 1/3 degrees. A user almost certainly requires visual feedback in order to aim the controller within a tolerance of 13 1/3 degrees.

Inspiration for the technique came both from a technique developed for touch wheel text input [19], and from a similar technique used in the Nintendo Wii Game “Super Monkey Ball.” This prior work, plus the property of being distance-independent, indicated the Circle technique would be worth investigating.

4.2 QWERTY Keyboard Technique

The QWERTY technique displays a standard keyboard layout and a dot cursor (Figure 2). The user controls the cursor by pointing the hand-held device at the display, and selects a character by hovering over it with the cursor and pressing a button.

The QWERTY technique is distance-dependent. As a user moves farther away from the display, the motion of the cursor is magnified, and the size of an individual button shrinks in motor space. The technique is also visibility-dependent, as the user almost certainly requires visual feedback during character selection in order to aim the cursor at individual keys.

One significant advantage of the QWERTY technique is the familiarity it holds for users. It is beneficial for users to be able to adapt very quickly to a new technique, and to reach optimal performance levels in a short time.

4.3 Cube Keyboard Technique

Visual feedback for the Cube technique is a 3D cube, subdivided into a 3×3×3 matrix of sub-cubes, within which are displayed the 26 letters of the English language, and the space character (Figure 2). A dot cursor is also shown inside the cube. Movement of the hand-held device in 3D space maps directly to the 3D motion of the cursor within the cube. When the user moves the cursor into a character sub-cube, that character is highlighted in red. To input a character, the user then presses a button.

The Cube walls are “hard” in the sense that a ballistic movement of the controller in the direction of the desired character will cause the cursor to “stick” to the side of the cube. Such an impenetrable border results in a reduced Fitts’s index of difficulty and enhanced performance, as described by Walker and Smelcer [25].

The Cube technique is distance-independent: cursor movement is relative only to the device itself, the size of the physical interaction space does not decrease with distance from the display. We hypothesized that the technique is visibility-independent because we expected that users can learn the locations of the various letters over time and select them without feedback.

The Cube technique can be considered a 3D extension of gesture-based 2D input techniques for pen-based computing, such as Venolia and Neiberg’s T-Cube technique [22].

Users of our early Cube prototypes reported difficulty in selecting characters that were occluded by other characters. To alleviate this problem our technique rotates the cube automatically as the cursor moves. As the cursor moves to the left, for example, the cube rotates about the vertical axis such that characters on the left side of the cube are more visible. Although we pilot tested other approaches, such as transparency and highlighting, we did not implement these in the final version because rotation was more effective and seemed to adequately solve the problem of occluded characters.

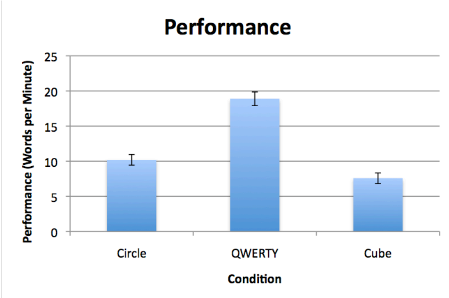

Figure 5. Performance for the three techniques in words per minute. Error bars represent standard error. N = 12

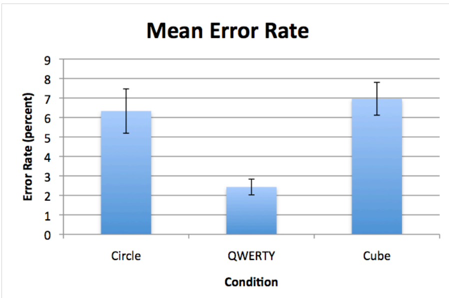

Figure 6. Mean error rates (percentage) for all three conditions. Error bars represent standard error. N = 12

4.4 Wiimote Tracking Details

Prototypes for each of the techniques use a Wiimote from a Nintendo Wii video game system as the input device. Wiimotes possess excellent ergonomics, built-in buttons, wireless Bluetooth connectivity, and relatively long battery life. The main disadvantage is the lack of true 6 degree-of-freedom tracking. Position information is obtained either through built-in 3-axis accelerometers, or from an infrared camera on the front that can track several infrared light sources.

The Circle and QWERTY techniques utilize the Wiimote's infrared camera for 2D pointing information. The Cube technique, however, requires that the 3D position of the Wiimote be known. To accomplish this we modified the Wiimote to act as an infrared light emitter, and used two additional Wiimotes as stationary infrared cameras. As seen in Figure 3, an infrared LED powered directly from the Wiimote power supply protrudes from the front of the device. The two fixed Wiimotes, mounted on stands, are then used to triangulate the 3D location of the manipulated Wiimote (Figure 4). Using the known position, orientation, and field-of-view of the two sensing Wiimotes, and the two sensed relative 2D positions of the hand-held Wiimote's LED, we calculate the 3D position of the hand-held Wiimote.

5 Experiment 1: Exploring Mid-Air Techniques

We conducted a controlled experiment to compare performance of the Circle, QWERTY, and Cube techniques for text input on a large display. By evaluating three techniques that provide contrasting points in the design space, this study provides a better understanding of mid-air input techniques in general and the properties of distance-dependence and visibility-dependence in particular.

5.1 Methodology

We followed a standard laboratory approach for the controlled experiment as an initial exploration of the design space.

5.1.1 Conditions

The experimental conditions were Circle, QWERTY and Cube, as described in the previous section.

5.1.2 Task and Apparatus

The experimental task was to enter a set of English phrases as quickly and accurately as possible. Target phrases were shown one at a time above the text input feedback mechanism. As each character was entered correctly it appeared under the target phrase to provide visual feedback. Participants had to correctly enter each character before continuing on to the next character. Errors in input caused the Wiimote to vibrate.

The phrase set was a randomly ordered version of that used by MacKenzie and Soukoreff [11]; the task was based closely on that used by Wigdor and Balakrishnan [27]. The same ordering of phrases was used in each session. Thus, each participant typed the same phrases, in the same order, for each of the three experimental conditions.

The experimental room contained a very large wall display approximately 4.9m × 2.4m (16' × 9') in size. Only a small portion of the display was used for the text entry task, to simulate an isolated operation in a collaborative environment. Participants stood 2.44m (8') from the display. For the Circle and QWERTY conditions an infrared light source was placed in front of the display to support the pointing functionality of the hand-held Wiimote. For the Cube condition, two Wiimotes on stands were used to measure the 3D position of the user-held Wiimote.

The software, written in Java, ran on a Microsoft Windows XP computer. The software managed all interactive components and logged all timing and error data. Bluetooth support was developed using a combination of the BlueCove implementation of the Java JSR-82 specification, a WIDCOMM Bluetooth stack, and custom Wiimote communications code.

5.1.3 Procedure

Each experimental condition for a particular participant was administered on a different day, in a separate one-hour session. For each session, the participant completed as many task blocks as possible in 50 minutes, where a task block consisted of 10 predefined phrases from the larger phrase set. During 3-minute breaks between blocks, the participant completed a distractor task (building a puzzle). The distractor task provided a mental and physical break from the primary task. This is consistent with real-world large wall display use, where it is unlikely that there will be lengthy, uninterrupted text entry. After each session, the participant completed a questionnaire for that condition.

At the beginning of the first session the participant was given a pre-questionnaire to collect demographic information. At the end of the third session, the participant completed a post-questionnaire that asked for rankings of the techniques, and comments.

5.1.4 Participants and Experimental Design

Twelve participants (3 female) were recruited through on-campus advertising. All participants were right-handed, although handedness was not a criterion for selection, and all were regular computer users (4+ hours weekly). Although a firm command of English was required of all participants, degree of fluency varied. When asked how long they had lived in English speaking countries, answers ranged from 1.5 years to 31 years (whole life).

The design was a single-factor within-subjects design. Order of presentation was fully counterbalanced across subjects.

5.1.5 Measures

Performance was measured using the standard words-per-minute metric, calculated as $60\times(|T|-1)/(5\times S)$, where T is the string entered, and S is the completion time in seconds [12]. Because users had to correctly enter a character before moving on to the next one, speed contained an implicit error penalty. For completeness, however, we also calculated error rates as the percentage of all character events that were errors.

The pre-questionnaire collected demographic information and computer experience. Questionnaires for each condition collected preference data using a 5-point Likert scale based on the NASA Task Load Index, as well as comments. Finally, a post-questionnaire collected comparative rankings on overall preference, speed, and difficulty, as well as qualitative comments.

5.1.6 Hypotheses

Experiment 1 was largely exploratory. We did not form any formal hypotheses. We did, however, expect that results would aid in determining the usefulness of mid-air text input techniques in general, as well as the three techniques in question specifically, and to help gauge the importance of distance-dependence and visibility-dependence as design factors.

5.2 Results

We ran a repeated measures ANOVA on the dependent variables of speed and errors. A Bonferroni adjustment was applied to all pairwise comparisons.

5.2.1 Performance

As shown in Figure 5, the average input speed in words per minute was 10.2 for Circle, 18.9 for QWERTY, and 7.6 for Cube. A one-way RM ANOVA showed a significant main effect of technique ($F_{2,22}$=291.556, p<.001). We ran pairwise comparisons to compare between techniques. QWERTY was faster than both Circle (p<.001) and Cube (p<.001). Circle was also faster than Cube (p=.001).

5.2.2 Error Rates

Mean error rates by condition were 6.3% for Circle, 2.4% for QWERTY, and 7.0% for Cube (Figure 6). A one-way RM ANOVA showed that technique significantly impacted error rate ($F_{2,22}$=55.590, p<0.001). Pairwise comparisons showed that participants made fewer errors with QWERTY than with either Circle (p=.009) or Cube (p<.001).

5.2.3 Subjective Measures

A summary of results from participants' subjective ratings of the three conditions is shown in Figure 7. Results were fairly consistent across perceived speed, difficulty, and overall preference. Users found the QWERTY technique to be the easiest to use, followed by the Circle and then the Cube technique.

Results from the post-questionnaire, asking for rankings on speed, difficulty, and overall preference are shown in Figure 8. It was clear that the QWERTY technique was favoured over the other two techniques.

5.2.4 Comments

Free-form written comments provided important insight into the different techniques. The most consistent feedback stated a preference for the QWERTY technique. Other interesting comments included the following:

- *"My ranking may be biased towards 'QWERTY Keyboard' model as I am usual [sic] to its use in daily life."*

This reveals an awareness of the biasing effect that familiarity with standard keyboards may have had on the user's performance. This is a potential confound, which we discuss later.

The Circle technique garnered a combination of positive and negative comments:

- *"Rotation alone was easier to manage than [rotation] + translation"*

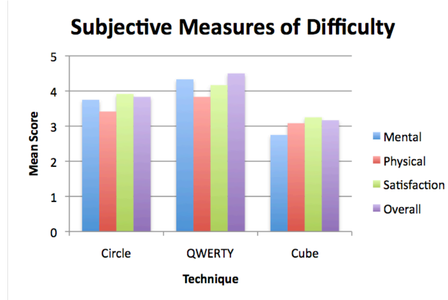

Figure 7. Mean scores from a NASA TLX based questionnaire. Ratings are on a scale of one to five (longer bars are better). *N* = 12

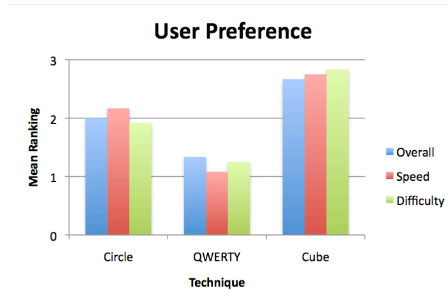

Figure 8. Mean scores from user rankings of the three techniques from best to worst (1=best, 3=worst, shorter bars are better). *N* = 12

- *"Seemed to require more accuracy than QWERTY technique"*
- *"I think if the sensor area was bigger, it would be easier"*

The first comment suggests that the approach has potential. Unfortunately, the second comment confirms our fear that the angular accuracy required to select an individual character was a problem. The last comment revealed an unexpected shortcoming in our implementation of the technique. The field of view of the Wiimote IR camera is limited, and several users reported problems caused by the Wiimote losing sight of the IR light source.

Despite the relatively poor showing of the Cube technique in terms of performance, several participants had positive comments that suggest that the technique might have potential:

- *"The Cube keyboard could be a great input method if some modifications were made..."*
- *"...it probably has the most potential for speedup of all the methods..."*
- *"smallest range of motion / potentially fastest method"*

5.3 Discussion

The QWERTY technique was significantly faster and had fewer errors than either the Circle or the Cube techniques. QWERTY performance of 18.9 wpm is competitive with both handwriting and pen-based typing. Combined with the positive subjective feedback on QWERTY, these results suggest that adapting the status quo QWERTY keyboard layout for mid-air interaction is a viable strategy for text input with large wall display systems.

The performance results for the Circle and Cube techniques are less encouraging. Although Circle was faster than Cube, the speeds for both techniques were low enough that they do not appear to be competitive with other techniques. Experiment 1 therefore does not provide evidence that Circle or Cube are appropriate for deployment, although it is useful to note that subjective scores for the two techniques are largely neutral to positive (Figure 7), indicating some potential.

There are reasons beyond performance, however, to continue exploration of the Circle and Cube techniques. Experiment 1 was

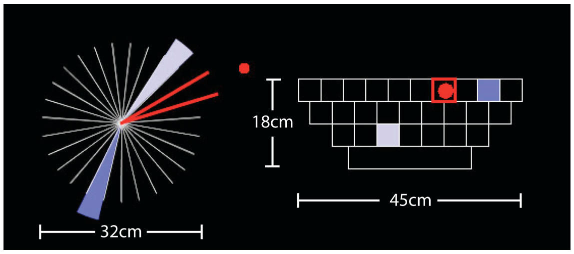

Figure 9. The Circle Keyboard (left) and QWERTY Keyboard (right) interfaces as used in Experiment 2.

exploratory and was not designed to isolate the impact of either visibility- or distance-dependence. The Cube technique holds promise because of its property of being both a visibility- and distance-independent gesture technique. We hypothesized that the visibility-independence could result in high levels of performance, if users are given enough time to practice. This hypothesis was supported by comments from participants in the experiment. Unfortunately the amount of time to achieve such performance may be very long, requiring a longitudinal experiment. We leave further investigations into the Cube technique to future work.

The Circle technique was faster than the Cube technique in Experiment 1. We chose to further evaluate it in Experiment 2, focusing on its property of distance-independence. We predicted that performance with Circle would not degrade as distance increases, whereas performance with QWERTY would. Experiment 2 also addressed the potential confound of keyboard layout familiarity. The QWERTY technique character layout was well-known to users, but the Circle character layout was novel. By removing the confound of keyboard layout and allowing users of the Circle technique to perform in the terminal Fitts's law stage of performance [10] we hoped to gain a more accurate and promising assessment of potential expert performance.

6 EXPERIMENT 2: INVESTIGATING DISTANCE-DEPENDENCE

Experiment 1 provided us with insight into performance for Circle, QWERTY and Cube. However, it was not designed to isolate either visibility-dependence or distance-dependence due to wider differences between the three techniques. The goal of Experiment 2 was to determine how performance of Circle, hypothesized as being distance-independent, and QWERTY, hypothesized as distance-dependent, differ as a user's distance from the display increases. An additional limitation of Experiment 1 was that previous experience with QWERTY keyboard layouts likely biased results in favour of QWERTY. Because of this, we modified the task to be a targeting task, simulating the terminal Fitts's stage of pointing in character entry.

6.1 Methodology

The experimental setup was largely similar to that of Experiment 1. We highlight only the differences here.

6.1.1 Conditions

The first factor was input *technique*. We tested the Circle and QWERTY techniques from Experiment 1, with the difference that the "keys" were unlabelled for both techniques, and target keys were instead highlighted in white (see Figure 9). A few minor graphical changes for the Circle technique were introduced, based on user feedback obtained in Experiment 1.

The second factor was *distance*. Participants interacted while standing either 2.74m (9') or 5.49m (18') from the large display.

6.1.2 Task and Procedure

The experimental task was to press highlighted keys on a blank (unlabeled) virtual keyboard as quickly and accurately as possible. Two keys were always highlighted: a white key indicated the current target, and a blue key indicated the next target. The purpose of the blue key was to allow the participants to plan ahead. The sequence of highlighted keys corresponded to the same input phrases as used in Experiment 1, although participants were not aware of the phrases because the keys were unlabeled.

The experiment was designed to fit in a single one-hour session. For each condition, participants completed four task blocks of 75 character input events, for a total of 300 character inputs for each condition. Between blocks was a 20 second break. Between conditions, participants filled out a questionnaire and worked on a distractor task for a total break of five minutes.

6.1.3 Participants, Measures, and Experimental Design

Sixteen new participants (8 female) were recruited through on-campus advertising. Fifteen of the participants were right-handed, and all used their dominant hand throughout the experiment. All but two participants were regular computer users (4+ hours weekly). Participants were compensated $10 for participating, and the fastest 50% of participants received an extra $10.

Performance and error data was collected in the same manner as for Experiment 1. Participants again filled out pre-questionnaires, questionnaires for each condition, and a post-questionnaire.

The experiment was a 2×2 within-subjects experimental design. The factors were technique (Circle and QWERTY) and distance (9 feet and 18 feet). Order of presentation was counterbalanced using a Latin square for the four combinations of technique and distance.

6.1.4 Hypotheses

The following hypotheses are motivated by the discussion in Section 3 on the impact of distance-dependence:

H1. Performance and error rates

1. Relative to Circle, QWERTY performance will decrease as distance increases.
2. Relative to Circle, QWERTY error rates will increase as distance increases.
3. Circle performance will not change with distance.

H2. Preference

1. Circle will be rated better relative to QWERTY at the larger distance than the shorter one.

6.2 Results

We ran a 2×2 repeated measures ANOVA (technique × distance) on each of the main dependent variables of speed and error rate. A Bonferroni adjustment was applied to all pairwise comparisons.

6.2.1 Performance

The average input speed for Circle was 11.6 wpm at 9 feet and 10.1 wpm at 18 feet, and for QWERTY it was 14.5 wpm at 9 feet and 10.3 wpm at 18 feet (Figure 10). An ANOVA showed significant main effects of technique ($F_{1,15}$=62.748, p<.001) and distance ($F_{1,15}$=244.573, p<.001). A significant interaction of technique × distance ($F_{1,15}$=12.935, p=.003) was also found.

To understand how distance and input type impacted performance differently, we conducted post-hoc pairwise comparisons on the interaction effect. Distance had a significant impact on both QWERTY performance (p<.001) and on Circle performance (p=.002). At 9 feet, QWERTY was faster than Circle (p<.001), but performance with QWERTY also degraded more quickly with distance and there was no difference found between the two techniques at 18 feet.

6.2.2 Error Rate

Mean error rates by condition were calculated as the percentage of all character entry events that were incorrect (Figure 11). The

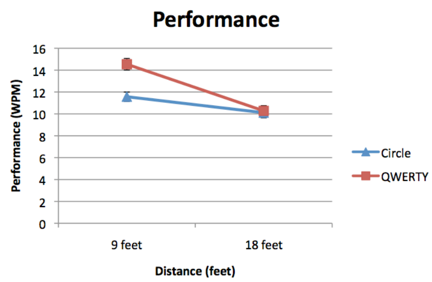

Figure 10. Performance for the four conditions in words per minute. Error bars represent standard error. *N* = 16

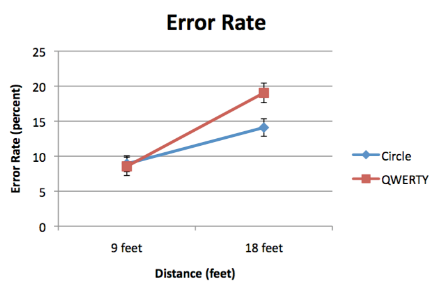

Figure 11. Mean error rates (percentage) for the four conditions. Error bars represent standard error. *N* = 16

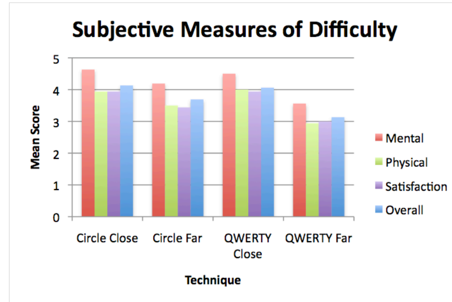

Figure 12. Mean scores from a NASA TLX based questionnaire. Ratings are on a scale of one to five (longer bars are better). *N* = 16

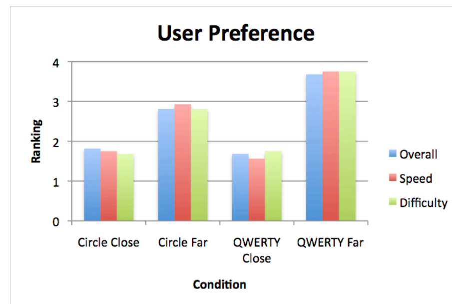

Figure 13. Mean scores from user rankings of the three techniques from best to worst (1=best, 3=worst, shorter bars are better). *N* = 16

average error rate for the Circle technique was 8.9% at 9 feet and 14.1% at 18 feet, and for the QWERTY technique was 8.5% at 9 feet and 19.0% at 18 feet. An ANOVA showed significant main effects of technique ($F_{1,15}$=11.792, p=.004) and distance ($F_{1,15}$=78.026, p<.001). A significant interaction of technique × distance was also found ($F_{1,15}$= 18.766, p=.001).

To understand how distance and input type impacted error rates differently, we conducted post-hoc pairwise comparisons on the interaction effect. Distance had a significant impact on error rates for both QWERTY (p<.001) and Circle (p<.001). These results showed the relative degradation of QWERTY as distance increases: at 9 feet there was no difference in error rate between Circle and QWERTY, but at 18 feet QWERTY had more errors than Circle (p<.001).

6.2.3 Subjective Measures

Questionnaires administered after each condition collected subjective ratings and comments. Responses to the TLX-based Likert questions are summarized in Figure 12.

Participants ranked the four conditions from best to worst in terms of perceived speed, difficulty, and overall preference (Figure 13). A Friedman test showed that technique significantly impacted rankings on all measures, including speed ($\chi^2_{(3,N=16)}$=30.675, p<.001), difficulty ($\chi^2_{(3,N=16)}$=27.675, p<.001), and overall preference ($\chi^2_{(3,N=16)}$=25.350, p<.001). Pairwise comparisons using Wilcoxon Signed Ranks Tests found no difference between Circle and QWERTY at 9 feet on any of the measures. At 18 feet, however, Circle was perceived to be faster (z=–2.231, p=.026) and less difficult (z=–2.349, p=.019) than QWERTY, and was preferred overall (z=–2.018, p=.044).

6.3 Summary and Discussion

We summarize our results according to our hypotheses:

H1.1 Relative to Circle, QWERTY performance will decrease as distance increases. *Supported*.

H1.2 Relative to Circle, QWERTY error rates will increase as distance increases. *Supported.*

H1.3 Circle performance will not change with distance. *Not supported.*

H2.1 At the larger distance Circle will be rated better relative to QWERTY than at the shorter distance. *Supported*.

The quantitative results of Experiment 2 successfully answered a number of questions raised by Experiment 1. First, QWERTY performance decreases significantly as the user moves away from the display, and is therefore distance-dependent. Contrary to our hypothesis, however, Circle performance also decreased as distance increased. Thus, although the Circle technique is invariant with distance in motor space, it is not entirely distance-independent. The significant interaction showed, however, that Circle performance decreases less than QWERTY performance as distance increases. In fact, we observed a crossover where Circle performance surpasses QWERTY performance: at 18 feet performance in wpm was no different for the two techniques, but errors were greater for QWERTY. These results support the hypothesis that techniques possessing a degree of distance-independence, in this case the Circle input technique, have value when applied to systems where users may be interacting from different locations in a room.

Considering the QWERTY technique, the argument that it is unsuitable for interaction at large distances is strengthened by the error data and subjective responses from participants. The error rate for the QWERTY 18 foot condition was the highest of all conditions. Perhaps more telling, that condition was subjectively ranked as the worst technique by almost all users. Comments related to the QWERTY 18 foot condition included: "*…it wants very much concentration, mentally and physically!*" and "*The slight shake of the hand makes pointing a tough task.*" These further highlight the limitations of distance-dependent techniques.

The second question raised by Experiment 1, whether the confound of key layout familiarity had any effect on performance of the Circle technique, appears to have been answered in the negative. While it is not appropriate to perform a statistical comparison of results between the two experiments, performance for both the labeled (Exp. 1) and unlabelled (Exp. 2) keyboards was in the range of 10-12 wpm. This suggests that in order to improve user performance, we will need to improve the design of the technique. It is unlikely that increased user familiarity with the key layout over time will result in large performance gains.

7 Conclusions and Future Work

Large wall displays are well suited to interaction techniques that can be used while outside of physical reach of the display. Unfortunately, most relevant work on text input has focused on touch techniques. We addressed this gap by investigating the use of mid-air text input for large wall displays.

We developed three mid-air text input techniques: Circle, QWERTY, and Cube, which combine input using a hand-held device with visual display feedback. The techniques differ in regards to their distance- and visibility-dependence. An experiment comparing the techniques showed that QWERTY performed significantly better than the other techniques, and is appropriate for deployment. However, there was also evidence that the Circle and Cube techniques hold promise.

A second experiment provided answers to questions raised by the first. Results showed that performance of the Circle technique degrades more gracefully than that of the QWERTY technique as distance between user and display increases, yet it is not entirely distance-independent. This means that the Circle technique may be better than QWERTY for use in large rooms, such as lecture halls. More generally this result suggests that the class of distance-independent techniques holds promise. In addition, we found that eliminating pre-existing knowledge of key layout from the evaluation of the QWERTY vs. Circle keyboards made little difference in performance.

We plan to further refine the techniques. The Circle technique's main limitation, namely difficulty in accurate character selection, might be dealt with using a detail-in-context approach to magnify the motor-space extent of characters. Possible refinements of the Cube technique include improving navigation on the depth axis by varying transparency. Alternate character layouts also need to be looked at. It would be interesting to investigate the use of a Cube-like technique on mobile devices, as its visibility-independence may serve well in that context.

More generally, we plan to investigate how mid-air text input techniques might function inside a larger working system. Mode switching and collaborative functioning are two factors that will be of significant interest in real-world scenarios.

8 Acknowledgments

Funding for this research was provided by the Natural Sciences and Engineering Research Council of Canada through operating and equipment grants, scholarships, and the NECTAR strategic research network, and by Defence Research and Development Canada.

9 References

[1] R.N. Bailey. *Human performance engineering: Designing high quality, professional user interfaces for computer products, applications, and systems*, 3rd ed. Prentice Hall PTR, 1996.

[2] P. Baudisch, M. Sinclair and A. Wilson. Soap: a pointing device that works in mid-air. In *Proceedings of UIST '06*, pages 43-46.

[3] B. Buxton. Surface and tangible computing, and the "small" matter of people and design. In *IEEE International Solid-State Circuits Conference Digest of Technical Paper*s (2008), pages 24-29.

[4] M. Cherubini, G. Venolia, R. DeLine and A.J. Ko. Let's go to the whiteboard: How and why software developers use drawings. In *Proceedings of CHI '07*, pages 557-566.

[5] M. Czerwinski, G. Smith, T. Regan, B. Meyers, G. Robertson and G. Starkweather. Toward characterizing the productivity benefits of very large displays. In *Proceedings of Interact '03*, pages 9-16.

[6] J.D. Foley, V.L. Wallace, and P. Chan. The human factors of computer graphics interaction techniques. *IEEE Comput. Graph. Appl.* 4, 11(1984), pages 13-48.

[7] T. Koltringer, P. Isokoski and T. Grechenig. Two-stick: writing with a game controller. In *Proceedings of Graphics Interface '07*, pages 103-110.

[8] S. Lee, S.H. Hong and J.W. Jeon. Designing a universal keyboard using chording gloves. In *Proceedings of CUU '03*, pages 142-147.

[9] Y. Liu, X. Liu and Y. Jia. Hand-gesture based text input for wearable computers. In *Proceedings of ICVS '06*, p. 8.

[10] I.S. MacKenzie, H. Kober, D. Smith, T. Jones, and E. Skepner. Letterwise: prefix-based disambiguation for mobile text input. In *Proceedings of UIST '01*, pages 111-120.

[11] I.S. MacKenzie and R.W. Soukoreff. Phrase sets for evaluating text entry techniques. In *Extended Abstracts of CHI '03*, pages 754-755.

[12] I.S. MacKenzie and K. Tanaka-Ishii. *Text Entry Systems*. Morgan Kaufmann, 2007.

[13] I.S. MacKenzie and S.X. Zhang. The design and evaluation of a high-performance soft keyboard. In *Proceedings of CHI '99*, pages 25-31.

[14] C. Magerkurth and R. Stenzel. A pervasive keyboard – separating input from display. In *Proceedings of PERCOM '03*, page 388.

[15] B. Martin. VirHKey: a VIRtual Hyperbolic KEYboard with gesture interaction and visual feedback for mobile devices. In *Proceedings of MobileHCI '05*, pages 99-106.

[16] E.D. Mynatt, T. Igarashi, W. K. Edwards and A. LaMarca. Flatland: new dimensions in office whiteboards. In *Proceedings of CHI '99*, pages 346-353.

[17] A. Pavlovych and W. Stuerzlinger. An analysis of novice text entry performance on large interactive wall surfaces. In *Human-Computer International*, 2005.

[18] E.R. Pedersen, K. McCall, T.P. Moran and F.G. Halasz. Tivoli: an electronic whiteboard for informal workgroup meetings. In *Proceedings of CHI '93*, pages 391-398.

[19] M. Proschowsky, N. Schultz and N.E. Jacobsen. An intuitive text input method for touch wheels. In *Proceedings of CHI '06*, pages 467-470.

[20] G. Shoemaker, A. Tang and K.S. Booth. Shadow Reaching: A new perspective on interaction for large displays. In *Proceedings of UIST '07*, pages 53-56.

[21] A. Tang, M. Finke, M. Blackstock, R. Leung, M. Deutscher and R. Lea. Designing for bystanders: Reflections on building a public digital forum. In *Proceedings of CHI '08*, pages 879-882.

[22] D. Venolia and F. Neiberg. T-Cube: a fast, self-disclosing pen-based alphabet. In *Proceedings of CHI '94*, pages 265-270.

[23] D. Vogel and R. Balakrishnan. Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. In *Proceedings of UIST '04*, pages 137-146.

[24] D. Vogel and R. Balakrishnan. Distant freehand pointing and clicking on very large, high resolution displays. In *Proceedings of UIST '05*, pages 33-42.

[25] N. Walker and J.B. Smelcer. A comparison of selection time from walking and pull-down menus. In Pro*ceedings of CHI '90*, pages 221-226.

[26] D.J. Ward, A.F. Blackwell and D.J.C. MacKay. Dasher: a data entry interface using continuous gestures and language models. In *Proceedings of UIST '00*, pages 129-137.

[27] D. Wigdor and R. Balakrishnan. TiltText: using tilt for text input to mobile phones. In *Proceedings of UIST '03*, pages 81-90.

[28] J.O. Wobbrock, B.A. Myers and J.A. Kembel. Edgewrite: a stylus-based text entry method designed for high accuracy and stability of motion. In *Proceedings of UIST '03*, pages 61-70.