# HydraScope: Interaction and Application of Wall Sized Display

*Hong Wu*
*Björn Hartmann, Ed.*

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2012-149
http://www.eecs.berkeley.edu/Pubs/TechRpts/2012/EECS-2012-149.html

June 1, 2012

Acknowledgement

We thank Prof. Bjoern Hartmann (EECS department, U.C. Berkeley) for his continuous
support and guidance. We also thank Viraj Kulkarni and Yun Jin for their efforts on
HydraScope project.

# HydraScope: Interaction and Application of Wall Sized Display

Hong Wu

University of California at Berkeley

**Abstract.** High resolution wall sized displays are becoming increasingly popular due to falling price. A lot of wall sized displays are set in the public space such as shopping malls and airports. However, there are few applications which allow users to interact with the contents on the wall sized displays. In this paper, we first introduce and compare four remote interaction control techniques, involving smart phones and Microsoft's Kinect controller. A user study of these interaction techniques revealed that a remote touchpad controller outperforms other interaction techniques in term of speed and accuracy. The second part of the paper focuses on developing applications for multi-display walls.Developing custom software for wall-sized displays today is time- and resource-intensive. We develop HydraScope, a framework for generating the applications by reusing and transforming the existing applications. This paper focuses on the design and implementation of one of the applications for HydraScope. The wall search application enables users to have a big picture of the search results and automatically open the links for users. A user study with ten participants reveals that the searching becomes more efficient by using HydraScope's search application.

**Keywords:** high resolution wall sized displays, smart phone, Microsoft's Kinect controller,touchpad remote control and search server application

## 1 Introduction

We combine standard LCD displays to create high resolution wall sized displays, as shown in Figure 1. Due to their decreasing price, wall sized displays become more

**Fig. 1.** Developers are working on the large wall sized displays. The wall sized displays can connect with laptops and smart phones.

popular in both research and commercial environments. For example, astrophysicists, biochemists, artists and crisis management centers can exploit large displays to visualize big images or data. Moreover, wall sized displays appear in our daily life such as at shopping malls and airports, where they are used to attract the attention of the audience. However, such displays can offer not only big data visualization but also rich collaboration to make the displays interactive with users. Nowadays, most of the wall sized displays can just show pictures or videos which do not actively involve the audience. The motivation of this paper is to make the wall sized application more interactive and appealing. There are two major aspects we consider in designing the solutions for wall sized displays.

First, new interaction techniques are necessary for large displays. Interaction with wall-sized displays is significantly different from that with smaller displays. Smaller displays, typically attached to a single computer, can be operated with traditional input devices like keyboards and mice. However, large wall-sized displays pose a different set of trade-offs than smaller displays. Users need to move freely in front of the large displays while interacting with them. Three of our new interaction techniques are based on using sensors from a mobile phone and the fourth one combines mobile phones with natural gestures detected using a Microsoft's Kinect controller. The dramatically incresing of smart phones provides new opportunities for interactive technique. Smart phone become cheaper and affordable and people carry them every day. Android smart phone provides rich interaction based on its multiple sensors, including accelerometer and touch screen [5], which provided acceleration, orientation, and touch position. To further free user from any kinds of hand holding devices, Kinect seems a good solution. Microsoft released the SDK of its Kinect controller [6] which can track the movement of human body by using structured infrared light patterns and computer vision. In this paper, we present four techniques to perform the simple task of selecting an object and dragging it to another location on a wall sized display. The results of the user study show that the touchpad controller based on smart phone achieves the best accuracy and speed among the four techniques.

Second, new applications need to be built for the wall sized displays. It is very expensive to develop particular applications from scratch for wall size displays. In order to reuse existing applications, we develop a system called HydraScope which can reuse PC applications and extend them to wall size displays. A process in background synchronizes the data between different displays. As there are multiple displays with only one synchronizing process, it is similar to Hydra, a gigantic monster with multi-heads from Greek legend. This paper discusses the details of one server application search, which show the search results and open the links automatically from google search. We pick the best remote control technique touchpad and develop an user interface to reserve the selected windows. A user study shows that our wall sized displays with remote control is better than the small display with traditional input devices in term of usability.

In this paper, Section 2 illustrates the related works in interactive technique and server applications. Section 3 describes the architecture, hardware and software we use in the project. Section 4 demonstrates the four interaction techniques, followed by a user study indicating the best remote control technique. Section 5 shows the search server application and its related control interface. Section 6 discusses the future work, followed by the conclusion in Section 7.

## 2 Related Work

### 2.1 Interaction Technique

A lot of recent research effort has gone into developing new interaction techniques that substitute mouse and keyboard input. Khan et al. [3] proposed a new widget and interaction technique, known as a Frisbee, for interacting with areas of a large display that are difficult or impossible to access directly. Boring et al. [2] proposed using mobile phones to interact with distant unreachable displays by taking a live video of the display on the mobile phone. Malacria et al. [4] proposed clutch-free panning and integrated pan-zoom control on touch-sensitive by drawing circles clockwise or counterclockwise. Nancel et al. [1] have studied and evaluated different families of location independent mid-air interaction techniques for pan-zoom navigation on wall-sized displays. Baudisch et al. [7] propose a way to move the target icon near the original items to reduce the distance of moving. Haller et al. [8] combines traditional collaborative tools with digital media which involves tangible tool palettes or digital pie menu. Guimbretire t al, [9], proposed a technique based on the metaphor of a whiteboard. Users can provide input using a pen-like device and the screen is split into different pages which can be dynamically moved or resized .Users can write on the screen and the system translates handwritten notes to text. For inputting images, the users can drag the file URL to an application on their mobile devices or laptops and the image would then appear on the screen.

### 2.2 Achitecture

Gjerlufsen et al., [11], introduced a system Shared Substance, which can interact between multiple input and output devices. Shared Substance can be performed by several users simultaneously. The users of the system include astrophysicists, biochemists, physicists and neuroscientists. Shared Substance can share the images or screens from different devices to the 32-screen wall. It can also display two brain images on each screen, totally 64 brain images on the wall. Johanson et al., [12], proposed a system Pointright is a system which enabled multiple input devices to interact with multiple displays in a seamless manner. Pointright decouples the display and the applications that are being show by the display from the underlying computer that is actually executing the application. With this model, any user could use any other input devices to interact with this display. Wigdor et al., [13], proposed a collaborating system WeSpace which solve the problems on privacy and equality. Our system also allow user to interact with each other but we do not need to develop the applicaitons on the system from scratch.

## 3 System Overview

In this section, we will review the software architecture of HydraScope.

### 3.1 System Architecture

The Hydrascope system consists of three components which are mobile remote controllers, network and a view synchronization server with displays servers.

Mobile remote controllers are written in two versions, one in native Android API and another in JavaScript. The version written in native Android API is used to pick up the best interactive technique while the version written in JavaScript is connected to the synchronization server.

Mobile phones communicate with servers by network protocol Open Sound Control (OSC) and socket.io.

View synchronization server transfer messages among display serves to synchronize the view and data by using NodeJS and Google Chrome extension. Node.js is a software system for scalable Internet applications, notably web servers. Google Chrome Extensions can be added into the Google Chrome browser as a plugin to run some small applications.

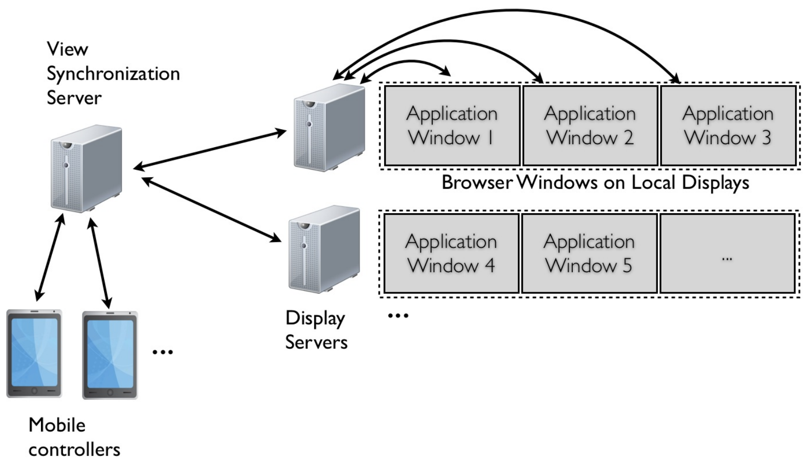



**Fig. 2.** The system architecture of the whole system. Mobile phone send messages to system server which sychronizes data with display server. Display servers control the contents shown on the screens.

Inside every server view node, as shown in Figure 3, HydraScope contains Instant Interface Manager (IIM) clients and server. When IIM client receive a message from IIM server or input manager, IIM client send a message to its IIM server. After that, IIM synchronizes its own clients and send another message to synchronization server. We exploit Google Chrome Extension as IIM and the existing applications are put as target pages.

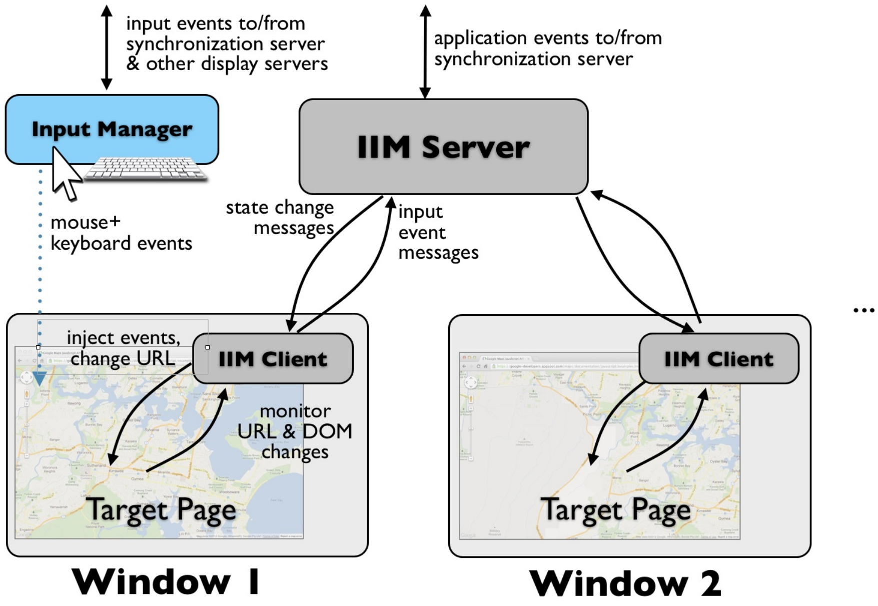



**Fig. 3.** The architecture inside a single display server node

### 3.2 System Hardware

Our system setup, as seen in Figure 4, includes a display wall built by 6 Samsung 460UTN-UD monitors in a 3x2 configuration with a HD resolution of 5760x2160 pixels. Each monitor offers resolution of 1920x1080 pixels. This wall is driven by a single Mac Pro - 2.4 GHz, Quad-Core Intel Xeon, 8 GB RAM and two ATI Radeon HD 5770 GPUs. A Mac Mini drives a single monitor with a resolution of 1600x1200.

A Microsoft's Kinect controller connected to a Windows 7 machine provides gesture input. In order to interact with the wall, we can use laptop, Android phones and iPhone.

## 4 Interaction Technique

We present and evaluate four different interaction techniques, three of which rely on the touch and accelerometer sensor of mobile phones, for selecting and dragging objects on

**Fig. 4.** The hardware setup in the project

wall-sized displays. The fourth technique combines mobile phones and Kinect. When using the smaller screen of the mobile phone to control the much larger display wall, an important factor that demands due consideration is the control-display ratio. A smaller control-display ratio affords fine granularity in interaction which, in our case, enables the user to finely control the pointing location on the wall. This, however, comes at a trade-off of speed of motion. A large control-display ratio affords faster movement across the large display wall but comes at the expense of accuracy. We reason that an effective interaction technique should offer the user a choice between fine grained and coarse grained interactions. This is especially important for wall-sized displays as the penalty incurred for the wrong choice of granularity increases with display size. Hence, all our interaction techniques are designed to use the tilt data from the accelerometer of the mobile phone to manipulate the control-display ratio.

At the top of the interface of the phone app, shown in Figure 5, there is an address bar contains the IP address of the server. Through Wi-Fi and the open source library

Open Sound Control (OSC), the smart phone app can communicate with the applications on the server.

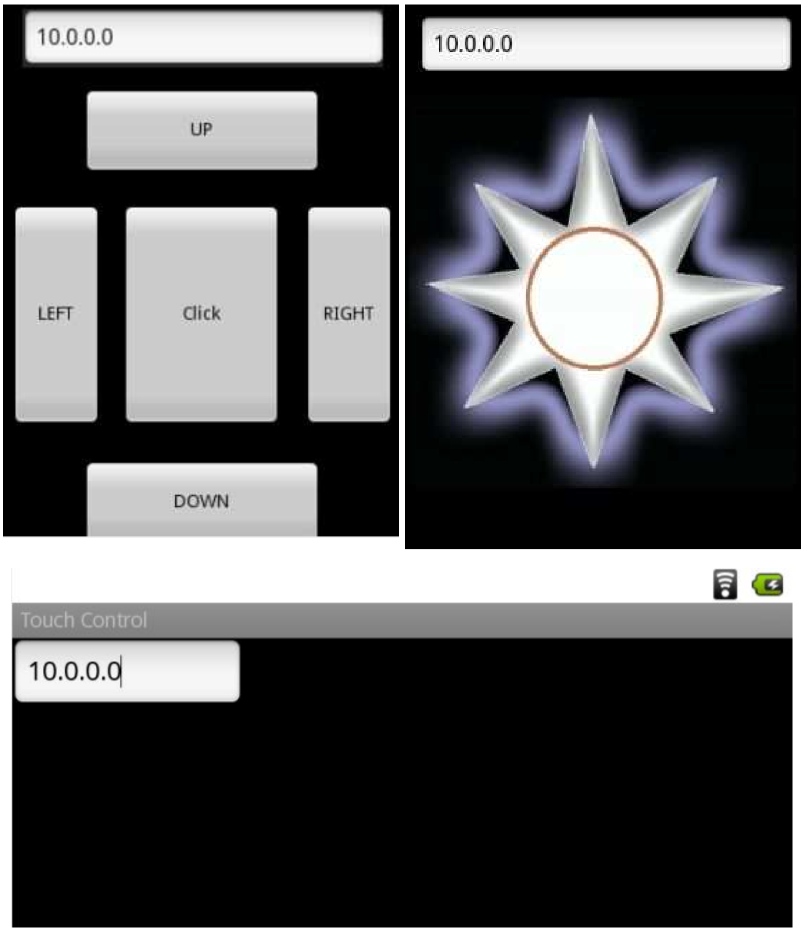



**Fig. 5.** Mobile user interface. Upper left: 4-way remote controller. Upper right: Omni-directional remote controller. Lower: Touchpad remote controller

### 4.1 4-way remote controller

The 4-way remote controller contains four buttons for four directions and a click button in the center. The interface is very straightforward. Four buttons can move the cursor in four directions while the center button simulate a click. The drawback of this design is that the cursor can move either horizontal or vertical rather than any arbitrary angles. The interface is shown in Figure 5 upper left. As with all the other three interactions, tilting the phone increases the control-display ratio. The ratio is calculated by accelerometer.

### 4.2 Omni-directional remote controller

Omni-directional remote controller is an improvement to the 4-way remote controller. The user can move the cursor in any arbitrary angles by touching the area outside the red circle. The touch is inside the red circle will be a click. The interface, shown in Figure 5 upper right, is self explained and user instantly knows the cursor can move in any directions. If user's click is inside the red circle, it is counted as a click.

### 4.3 Touchpad remote controller

The design of touchpad remote controller is different from the above two remote controllers. There are no button on the interface, shown in Figure 5 lower. We use relative mapping for mapping the touch input to on-screen coordinates. User can move cursor or drag an image by pressing and remaining the figure on the smart phone. If user touches the screen of the smart phone and release from the screen fast, it appears as a click.

### 4.4 Mobile phone coupled with Kinect

As it is harder to achieve both speed and accuracy, we combine Microsoft's Kinect and smart phone controller to achieve better performance in both speed and accuracy. The Kinect is used for quickly placing the image roughly in the area pointed to and the phone is then used to fine place the fine. Kinect SDK provides 20 skeleton points for the whole human body and can track at most 2 people at the same time. Kinect uses infrared ray to determine the depth. User can switch between two mode: roughly place and fine tune. Kinect is used in roughly place while Smart phone is used in fine tune.

**Fig. 6.** User tests mobile phone coupled with Kinect.

For Kinect, the hand closer to the Kinect is associated with cursor. After placing the cursor to the roughly correct area, user can switch to fine tune and further place the cursor at the exact place. Figure 6 shows a user testing this technique.

### 4.5 Evaluation and Results

10 users, 8 males and 2 females, in the age group 20-30 years of age, participated in our user study. All of them were right handed and were daily computer users. User needs to select an highlighted image and move into a targeted rectangle. We compare the result with different image size and various movement distance, shown in Figure 7.

*Selection time and movement time*

In accordance with the Fitts law [10], the size of the image to be selected affects the selection time while distance of the target box from the image affects the movement

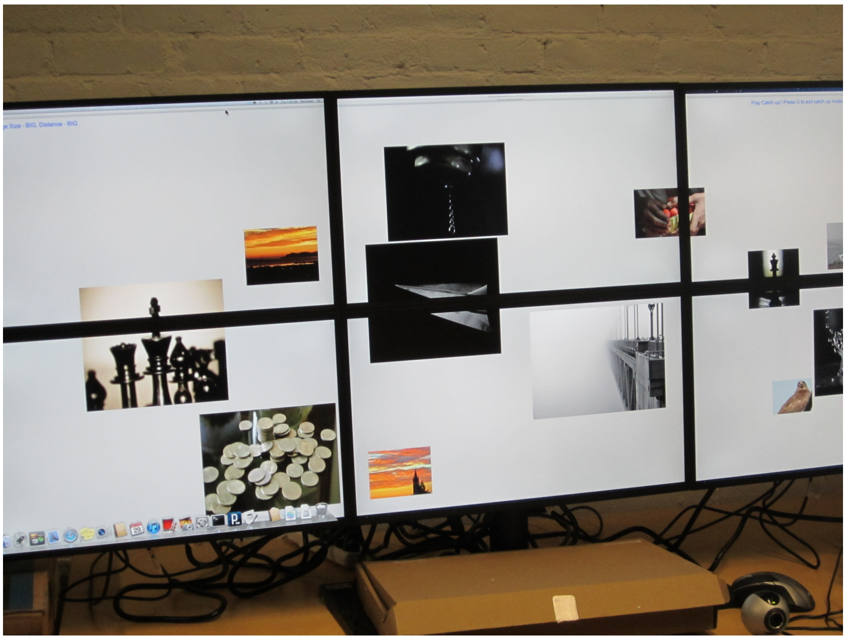


**Fig. 7.** Test application canvas with different size of images

time required to drag the image. As expected, shorter distances and larger image sizes translate to shorter task completion times.

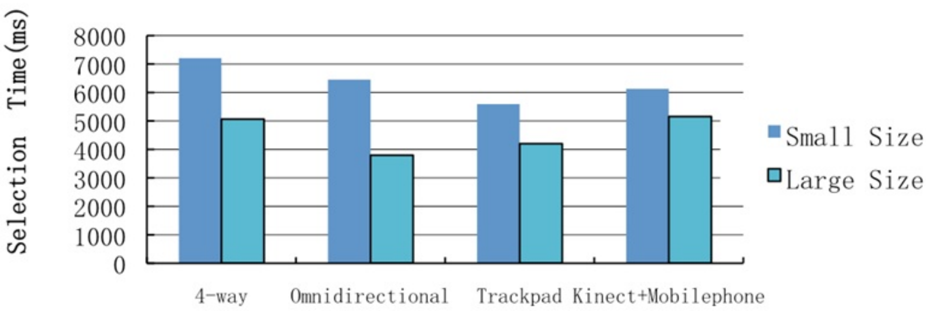



**Fig. 8.** Selection time of four interaction techniques based on two image sizes.

Figure 8 illustrates the selection time of different image sizes by using the four interaction techniques. Smaller images lead to higher selection times for all the techniques. For small images, touchpad interaction technique outperforms the others but for larger images, the omni-directional technique is faster than the touchpad.

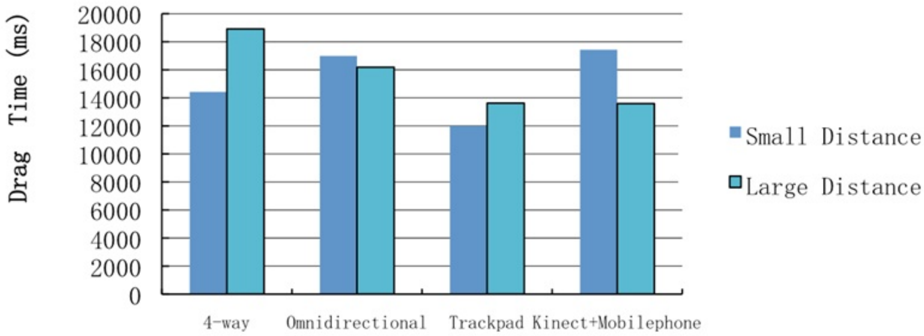


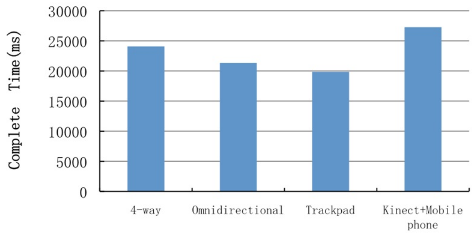

**Fig. 9.** Movement time of four interaction techniques based on two distances.

Figure 9 shows the movement times for different distances. Surprisingly, for the omni-directional remote control and mobile phone coupled with Kinect, smaller distances took longer times than larger distances! The Kinect implementation is good for coarse-grained interaction but its not very well suited for tasks that demand accuracy. There is also a noticeable latency in our Kinect implementation which might have caused this anomaly.

*Overall time and incorrect placements*

Overall time is the total test completion time and it sums up the selection time and movement time for each technique regardless of the image size and the distance. Figure 10 left shows the total test completion times. The touchpad interaction works the fastest while the Mobile phone coupled with Kinect is the slowest. Omni-directional is faster than 4-way remote control as we expected.

Figure 10 right shows the average number of incorrect placements for each interaction. As far as accuracy is concerned, the 4-way remote control technique performed the best while the Kinect was the worst.



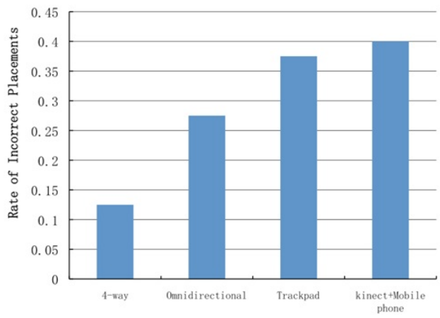




**Fig. 10.** Left: Task complete time per task of four interaction techniques. Right: Number of incorrect placements per task of four interaction techniques

*Open questions and qualitative data*

The qualitative data collected from the participants gives insights into some interesting observations. As seen in Figure 11, the touchpad is the most preferred technique since it is easy to use and very tractable. Though omni-directional remote control is faster than the 4-way remote control, users much preferred the 4-way to the omni-directional technique. Users claimed the omni-directional technique was harder to use. One of the reasons might be that users are more familiar with up/down/left/right keys from using keyboards and joysticks. Another reason is that unconstrained motion in 2D space requires more cognitive effort than orthogonally constrained motion.

To sum it up, the touchpad technique was the most widely preferred and also the most efficient one. However, for applications where a higher degree of accuracy is desired, the 4-way remote control is more preferred. Although the omni-directional remote gives users more freedom and more speed and accuracy of movement, users tend to dislike it because it requires more cognitive effort. Users found the Mobile phone coupled with Kinect the most exciting, and also the easiest to learn.

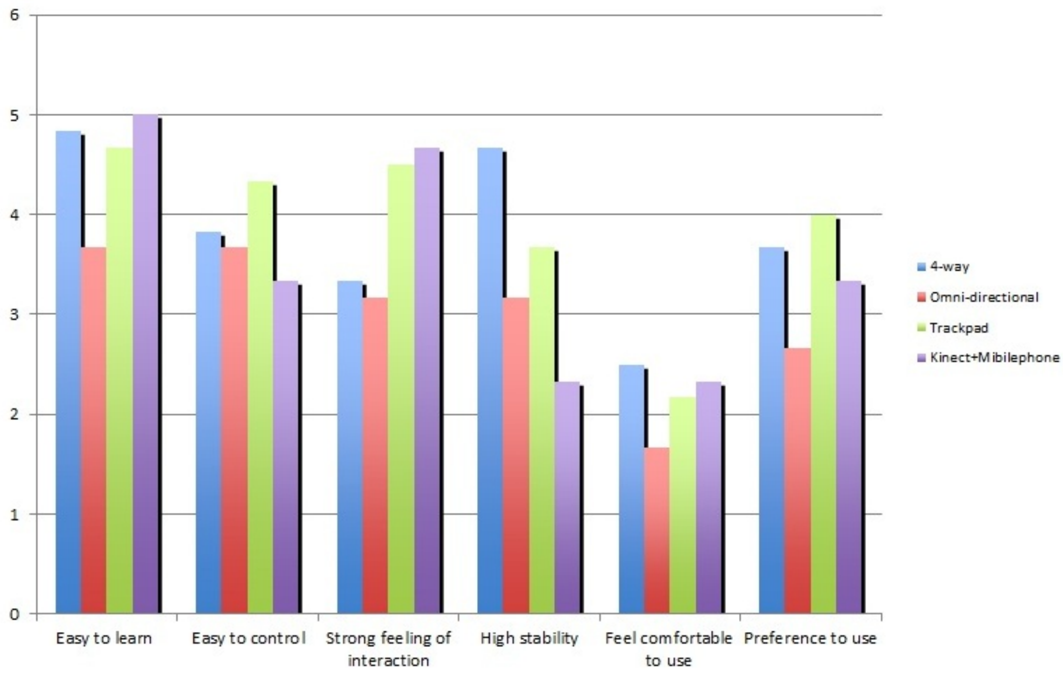



**Fig. 11.** Qualitative date from users for four interaction techniques. 0 - not agree at all. 5 - strongly agree

### 4.6 Discussion

The techniques were designed keeping in mind the large size and high fidelity of these displays which afford a different set of trade-offs than smaller personal displays. Based on our user study, we compared these interaction techniques and presented our analysis on the merits and demerits of each technique.

Our evaluation consisted of both measurements conducted through live testing and qualitative feedback through open ended questions. Since the users were unfamiliar with the techniques, we developed a game which they were asked to play before the measurements. This helped them get familiarized with the techniques faster and in a more exciting way.

There are multiple ways to combine gesture input from the Kinect for coarse interaction with fine grained input from the phone. Our future work will consist of developing and evaluating more techniques which use natural gesture detection. We also intend to test different approaches towards changing the control-display ratio using tilt data from the accelerometer of the mobile phone.

## 5 Server Application

The server applications are built in order to reuse the existing applications and adopt them into wall sized displays. The data inside the application need to be sychronized. We have search, map, stock, hackpad and presentation server application. This paper will mainly focus on the search applicaiton, which is built upon Google search.

### 5.1 Mobile User Interface

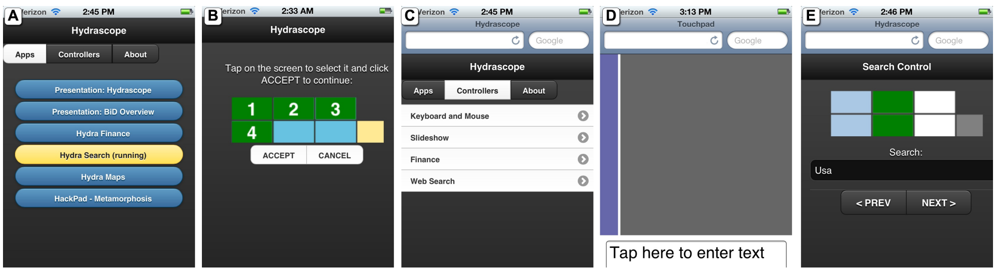



**Fig. 12.** (A) HydraScope allow user to select the application. (B) select the screen to run the application. (C) choose the specific input method. (D) use the touchpad to control the cursor and textbox to input the character. (E) A specific interface for search application.

HydraScope introduce mobile phone interfaces which allows user to choose the server application. In Figure 12(A), user selects the applications to run and the selected application is highlighted as yellow. After that, user need to choose the windows to run the selected application, shown in Figure 12(B). In Figure 12(C), user can select a specific remote controller to interact with wall sized displays. Figure 12(D) shows the scroll, cursor move and text input control interface. After launching the search application, user can press ’next’ to update the result window and open the next batch of links. User can select windows and lock them to disrupt the update on those windows. User can also input the key word in the text bar shown in Figure 12(E).

### 5.2 Search Applicaiton

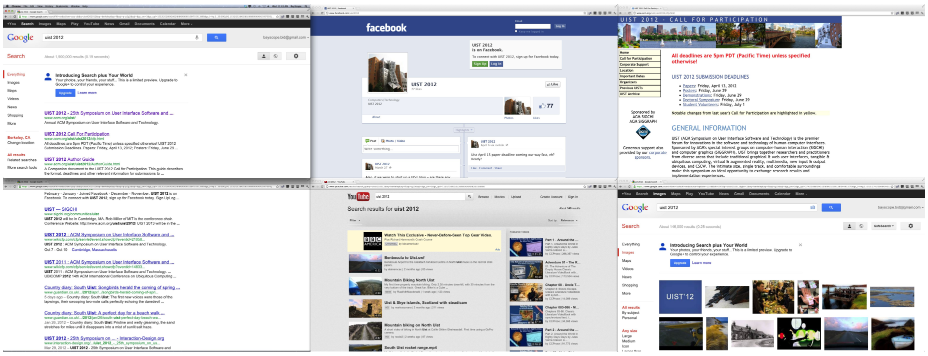


**Fig. 13.** Screen shot of the search application

Search applications, shown in Figure 13 allows users to quickly go through the result set. The left two screens illustrate the results return by Google. These two screens are synchronized by the scroll bar position to form an extendable long window. The rest four screens shows the top four links in the search results, which saves the time of opening the links and shows more information to users. Users can switch to the next four links by just pressing ’next’ button on the user interfaces, shown in Figure 13. Our controller can also let user to lock the windows they are interested in and keep it even when search for another query.

In our implementation, the mobile controllers send the message to the background page of Google Chrome Extension. The background page analyses the message and send another message to the JavaScript page behind each web page to synchronize the data and actions.

### 5.3 Evaluation and Results

To evaluation the applications, we recruit 10 users and form them into 5 pair. Each pair need to collaborate with each other to figure out the answer of questions. The test set requires user to search the answer of the questions in different conditions. The first condition is using keyboard, mouse and a regular size display. In the second condition, users switch to our search application in wall sized display but still using keyboard and mouse. In the third condition, users can just use our smart phones controller on wall sized display.

The question for users to find answer includes

Question1: Which continent has the most French speakers in the world? For this task, users should use keyboard and mouse to find the right answer by google search on single display.

Question2: China recently completed a tunnel that goes beneath the Pearl River. How many minutes would it take you to go through it by train? For this task, users are asked to use keyboard, mouse and mobile phone on six displays.

Question3: Did your brains frontal lobe or temporal lobe have more to do with planning out which movie you'd like to see this evening? For this task, users are asked to only use mobile phone on six displays.

Users strongly agreed that the search application on wall sized displays shown more information and give a good picture of the whole domain of the questions. It decrease the time to open the links and will be more powerful with the increasing of the displays. Users collaborate with each other well. For instance, one use the interface of search, input keywords and click next and lock screens. The other one use touch pad and scroll bar to scroll up and down to find useful information. In term of the mobile phone controller, some user adopted the technique very fast while others took some time to get used to the controller. Our application will be benefit if we combine the keyboard and mouse and search interface together.

### 5.4 Discussion

HydraScope provides the framework and API for software developer to easily extend their application for single display to wall sized displays. One outside developer spent only three hours to extend an existing application to fit for wall sized displays. The application reviews cyclists' performance within a particular time period, shown in Figure 14. In the wall sized displays application, user can select the team's name to view their performance. User can also input a particular time period to synchronize the viewing of all the team.


**Fig. 14.** Outside developer is using HydraScope to extend an existing current application to a wall sized application.

## 6 Conclusion

HydraScope is the framework and API to quickly expand existing single display application to wall sized display application. In order to better fit into the scenario of wall sized display, HydraScope introduce new interaction technique by using smart phone and Kinect. Through the user study, Touchpad technique achieves the fastest speed and the most accurate. We applied Touchpad technique into HydraScope remote controller and developed search server applications to automatically open the links for user. Outside developer also benefited from HydraScope as it is easy to develop wall sized application.

## 7 Acknowledgements

We thank Prof. Bjoern Hartmann (EECS department, U.C. Berkeley) for his continuous support and guidance. We also thank Viraj Kulkarni and Yun Jin for their efforts on HydraScope project.

## References

1. M. Nancel, J. Wagner, E. Pietriga, O. Chapuis and W. Mackay: Mid-air Pan-and-Zoom on Wall-sized Displays. In Proc. CHI 2011 pp. 177 186.
2. S. Boring, D. Baur, A. Butz , S. Gustafson, P. Baudischl: Mobile Interaction Through Video. In Proc. CHI 2010, pp. 2287 2296
3. A. Khan, G. Fitzmaurice, D. Almeida, N. Burtnyk, G. Kurtenbach: A remote control interface for large displays. In Proc. UIST 2004, pp. 127-136.
4. S. Malacria, E. Lecolinet, Y. Guiard: Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: the cyclostar approach, in Proc. CHI 2010.
5. http://developer.android.com/reference/android/hardware/Sensor.html
6. http://www.xbox.com/en-US/kinect
7. P. Baudisch, E. Cutrell, D. Robbins, et al: Drag-and-pop and Drag-and-Pick: Techniques for accessing remote screen content on touch- and pen-operated systems. In Proc. Interact 03, pages 5764, 2003.

8. M. Haller, J. Leitner, T. Seifried, et al: The NiCE Discussion Room: Integrating Paper and Digital Media to Support Co-Located Group Meetings. In Proc. CHI, 2003
9. F. Guimbretire, M. Stone and T. Winograd: Fluid Interaction with High-resolution Wall-size Displays. In Proc. UIST, 2001
10. Fitts law, http://en.wikipedia.org/wiki/Fitts's_law
11. T. Gjerlufsen, C. Klokmose, J. Eagan, C. Pillias and M.l Beaudouin-Lafon: Shared Substance: Developing Flexible Multi-Surface Applications. In Proceedings of Human Factors, 2011, pp. 3383-3392
12. B. Johanson, G. Hutchins, T. Winograd, and M. Stone: PointRight: experience with flexible input redirection in interactive workspaces. In Proceedings of UIST, 2002, pp. 227-234.
13. D. Wigdor, H. Jiang, C. Forlines, M. Borkin, and C. Shen: WeSpace: the design development and deployment of a walk-up and share multi-surface visual collaboration system. In Proceedings of CHI, 2009, pp. 1237-1246.