



Javelin: an access and manipulation interface for large displays*

Zhen-kun ZHOU, Jiang-qin WU^{†‡}, Yin ZHANG, Da-wei XIE, Yue-ting ZHUANG

(School of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China)

[†]E-mail: wujq@zju.edu.cn

Received Sept. 14, 2010; Revision accepted Oct. 10, 2010; Crosschecked Sept. 14, 2010

Abstract: We describe a user interface and interaction technique, named ‘Javelin’, designed for large display environments. It provides quick access to random screen regions and manipulation methods for screen widgets which are difficult or impossible to reach. It consists of a dynamic global thumbnail, a touchpad widget that drives the screen cursor, and a teleport widget in which interactions are transferred to its target screen region. Javelin can be easily integrated into many programs to optimize their interaction performance in large screens. The experiment and user study show that Javelin can extend user access field and enhance widget manipulation in large displays.

Key words: Large display, User interface, Interaction method, Screen widget, Remote target acquisition, Human-computer interaction (HCI)

doi:10.1631/jzus.C1001008

Document code: A

CLC number: TP391

1 Introduction

In recent years, large display devices have been widely used in our daily life, such as for public news broadcasting (Peltonen *et al.*, 2008) or scientific mass information visualization (Wei *et al.*, 2000). With improvements in the manufacturing process, larger screen size and higher display resolution are now available to maximize the visual bandwidth, which helps to demonstrate more and more information to users simultaneously. A longitudinal study (Ball and North, 2005) shows that large displays are helpful in launching more application windows while keeping all of them viewable.

The rapidly decreasing cost of large display devices, e.g., LCD (liquid crystal display) walls or multi-projectors tiled walls, has encouraged researchers in related research fields to construct their own high-resolution wall-sized displays (Schikore *et al.*, 2000). More recently, with improvements in tac-

tile interaction techniques, especially the multi-touch techniques, large displays now have additional interaction capability and appear in new application scenarios, such as school class education, group collective work (Morris *et al.*, 2010), as well as game playing (O’Hara *et al.*, 2008; Stødle *et al.*, 2008).

The research community has been interested in human-computer interaction (HCI) on large screens. In this field, especially for large wall-sized displays, there are several major usability issues (Badillo *et al.*, 2006), such as interaction cursor tracking (Khan *et al.*, 2005), large screen target acquisition (Bezerianos and Balakrishnan, 2005), task management (Malik and Laszlo, 2004), GUI (graphic user interface) widgets, and touch sensation (Wilson, 2004).

In this paper, we introduce Javelin. The name ‘Javelin’ indicates that its design focuses on interactions with the remote part of large screen display space, i.e., the areas out of user arm length. Javelin provides quick random screen region access, global screen display context thumbnail, and remote widget manipulation. Compared with the prior work mentioned above, the main contribution of Javelin is as follows: (1) Javelin introduces dynamic thumbnails, which provide global context awareness with remote

[‡] Corresponding author

* Project supported by China Academic Digital Associative Library (CADAL)

© Zhejiang University and Springer-Verlag Berlin Heidelberg 2010

screen region interaction, diminishing a user's physical movement; (2) Javelin is an easy-to-integrate library, which can enhance the user interaction performance of many existing applications in large display environments; and, (3) Javelin provides a multiple-instance interface for group collective work.

This paper is organized as follows: we first articulate the development motivations and our design considerations. Second, we introduce the components of Javelin and the operations it supports. Some experimental applications integrated with Javelin are described. Third, the experiment and user study results are described. Finally, we conclude our work and discuss potential directions for future work.

2 Motivations

2.1 Hardware

We are planning to build a large wall-sized display system for academic research. The configuration is as shown in Fig. 1.

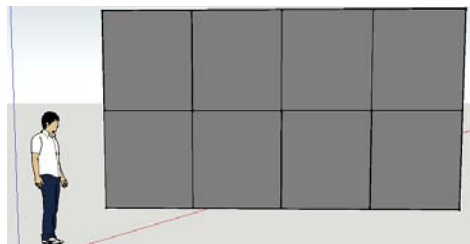


Fig. 1 A display wall made up of eight modular LCD display cells

The display wall, with a size of 5.1 m×2.1 m, can provide up to 4096×1532 pixels resolution display. Because of the limitation of our indoor decoration, only the lower four cells are configured to support touch interaction

A user interface for this 'heterogeneous' touch interactive configuration is considered to make better use of the display space.

2.2 Software

We have developed some rich Internet applications (RIAs) on Adobe Flex platform. These programs should be transplanted from the single monitor environment to the wall-sized display environment. In the process of transplanting, the programs' GUI and interaction method should be optimized. We need a solution to minimize design and programming work.

Moreover, in large screen and multi-display environments (MDEs), the display space management and screen widget manipulation, as well as interaction and display context redirection, are worth researching.

3 Related works

There are many scenarios in which interaction from a distance to the display screen is ideal, such as watching movies, navigating high-resolution satellite images, mass data visualization, and other jobs that require the user to maintain an overview of display content. In some other scenarios, users prefer to stay up close to the display screen and focus on the 'local' contents, which provide high-resolution and more detailed visual information.

In the up-close-to-screen environment, touch is considered as a simple, reasonable, and robust interaction method. Recently, a lot of research work has been conducted on multi-point touch sensitive display (Malik, 2007; Moscovich, 2007). It is not simple, however, to design the user interface and interaction, since in some circumstances, users have to step back to have an overview of the display context (Badillo *et al.*, 2006). A mini map of global content, i.e., thumbnail, could benefit if its screen size is appropriate. In Wordlets (Elvins *et al.*, 2001) and Worlds in Miniatures (Stoakley *et al.*, 1995), similar ideas were proposed. For the Javelin system, we designed a widget called 'mini map' to show a dynamic thumbnail of the global display (Fig. 2).

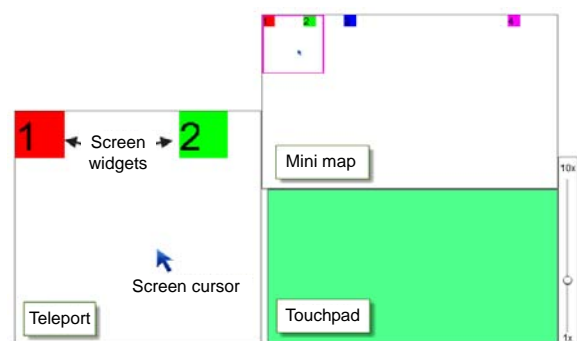


Fig. 2 Layout of Javelin

In a high-resolution GUI, the fingers on the screen may obscure the underlying content. Moreover, the 'fat finger' problem may make it difficult for some users to select very small widgets on the screen. There

have been various research attempts to address these problems (Albinsson and Zhai, 2003; Benko *et al.*, 2006).

Screen cursor behavior has also been investigated. In a touch sensitive display, finger movement drives cursor movement, and tap screen simulates a mouse click, while one drags widgets, e.g., icons, by performing ‘drag and drop’ operations. In multi-touch sensitive displays, more than one finger is involved to provide a rich set of interactions. SDmouse (Matejka *et al.*, 2009) introduced a technique that uses multi-finger to emulate full mouse functions, such as tracking the cursor, three-button operations, and chording. In detail, SDmouse uses touch points to drive screen cursor movement, and to recognize finger relative positions to emulate mouse operation, e.g., right-clicking. In Javelin, the touchpad widget acts as an on-screen mouse simulator, and magnifies finger movement to drive the screen cursor movement. The magnification factor is adjustable by users. In a display space where widgets reside sparsely, the larger factor is preferred.

In a large screen display, there are some regions and some widgets which are difficult or impossible to access and manipulate. Visual touchpad (Malik and Laszlo, 2004) divides a vision-based touchpad into two parts. One region is mapped to the entire screen and the other region is mapped to a small area. Interaction on the first region makes a user-interested screen part change, and in a certain screen part area, interaction with the second region manages cursor movement. The visual touchpad provides the user with a random screen region access capability. It is not optimal, however, for large wall-sized touch sensitive displays, and it assumes users sit at a distance from the display screen. The Frisbee (Khan *et al.*, 2004) provides a portal to another part of the screen. It consists of two circular widgets: ‘teleport’ and ‘target’. The target is a circular widget, working as a visual feedback for users to see which part of the screen is currently being remotely operated on. The teleport widget receives user interaction and translates this to the corresponding content in a target circle, as if the operations were done directly to the remote content. The main contribution of Frisbee is the idea of ‘hyperlink’ between the screen parts. The Javelin system adopts a similar concept, and extends the single ‘teleport-target’ interface to multiple ‘teleport-

cursor’ portals, which enables multiple-object interactions, as well as collective work by a group of participants.

4 Javelin design and components

We present Javelin for the following problems:

1. Quick random screen space access.
2. Remote widget acquisition and manipulation.
3. Interactions between the remote area and local workspace.

The three problems, as well as our motivations, lead to some design principles listed below:

1. Application and scenario generality. We want the Javelin interface to work as a library or plug-in so that it will be easy to deploy Javelin in many existing RIAs, and to minimize the programming work.

2. Reduce user body movement. The user stands close to the large screen and his/her body movement, such as step aside or back/forward, should be reduced. Usually, the user steps backward to see the overview of the screen context. The body movement is exhausting and sometimes it diverts the user’s attention and interrupts his/her thought.

3. Support multiple participants. Javelin should support collective work among team members.

To meet the above requirements, we designed and implemented the Javelin user interface as follows.

As shown in Fig. 2, Javelin interface consists of four main widgets: a screen cursor, touchpad area, a global screen context dynamic thumbnail ‘mini map’, and a viewport teleport which is represented as a block in the mini map.

4.1 Touchpad

Similar to the SDmouse (Matejka *et al.*, 2009), the touchpad is an on-screen mouse simulator. In the touchpad area, user finger movement drives the screen cursor movement in the screen display space. Different types of clicks, drag-drops, and other mouse-like operations are supported. The recognized mouse behavior is transferred to the screen cursor, and replayed as if the cursor triggered the mouse event at its screen position.

Fig. 3 shows the screen cursor icons and the touchpad widget.

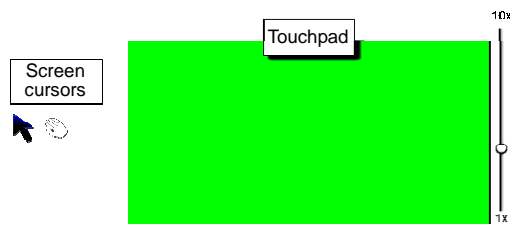


Fig. 3 Cursor and touchpad

Two kinds of cursors were used to indicate tracking and dragging states. On the right side of the touchpad is a vertical slider, whose value indicates the movement magnification factor, ranging from 1 to 10. User finger movement distance on the touchpad is magnified by the factor, and drives the screen cursor movement

The initial value of the magnification factor is the ratio of screen resolution width to the touchpad width. The larger the magnification factor, the smaller the distance the user finger needs to move and the lower the cursor accuracy will be.

In a user study, 40% of participants thought that the magnification factor vertical slider was useless. Another 30% of the participants, however, highly appreciated the design. As the small widgets in the thumbnail were difficult to select, the vertical slider was maintained.

4.2 Mini map

The mini map widget provides a dynamic global thumbnail of the whole-screen display content. At the beginning of every frame, the mini map is updated. Programmers can configure which screen content is included in the mini map. For example, Javelin interface instances in the display space are excluded from the capture list of the mini map.

In our plan, the wall-sized display system will provide a full screen resolution of 4096×1532. We choose the top-left region as an example. The background of Fig. 4 is the screen part.

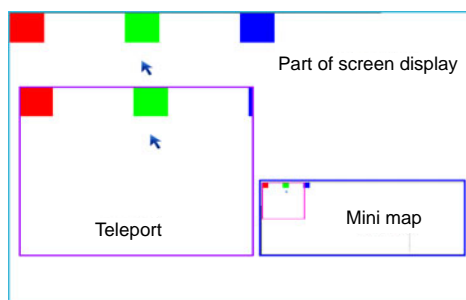


Fig. 4 Top-left screen, the mini map, and the teleport
In the mini map widget, the global thumbnail updates to reflect screen context changes. The cursor is also captured

With mini map in view, users do not need to step backward to overview the screen context. In the analysis of user studies, the mini map view reduces body movement and helps users to concentrate.

In our user study, some participants suggested that the touchpad and mini map should be integrated. They thought that the touchpad space could be mapped to a full display space, and its background could be the screen context thumbnail. User interaction with snapshots of widgets in the combined thumbnail causes the corresponding widget to be manipulated. This has the following reasons: the initial value of the magnification factor caused the touchpad widget to be mapped to the full screen space. However, if we adjust the factor according to practical scenarios, the touchpad-mapped screen spaces do not keep the same size as the screen space. For this reason, the layout of Javelin remains unchanged.

Clicking on the mini map or dragging the square causes the inner content in teleport to change (Fig. 5). This design enables a quick random access to any region of the screen.

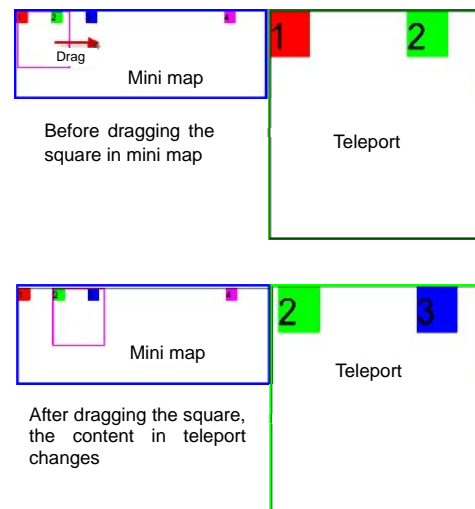


Fig. 5 Teleport and screen regions

In the mini map, the square indicates the part of the screen context that the teleport widget has targeted

4.3 Teleport

The teleport widget is a viewport for screen content redirection (Tan *et al.*, 2004) from the remote screen region to the local workspace. It is also a user input redirection (Johanson *et al.*, 2002) widget that transfers the user in teleport interactions to the target

remote region, as if interacting directly in a remote region.

For example, in Fig. 6, the teleport is targeted to a remote screen area containing a drop-down-menu widget. If we click the menu in teleport, the click event is transferred and applied to the drop-down menu in the remote area, as if we directly clicked it.

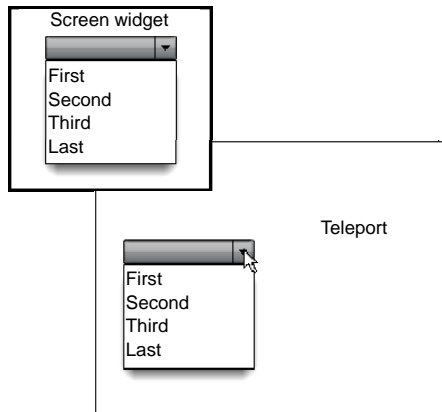


Fig. 6 Teleport: content and interaction redirection

Drag and drop operations are common in GUI environment. However, in a large display application scenario, considering the huge display space size, the movement distance is a notable issue. We adapted the operation of drag between screen space and teleport.

The key idea of drag between screen space and teleport is allowing the dragging to span remote region and the local workspace. For example, if a user wants to move an icon nearby to a remote area that he/she cannot reach, the user will target the teleport to the corresponding remote area, select the icon in the screen space, drag it across the border of the teleport, and drop the icon in the teleport widget. Then the icon will appear in the remote screen area (Fig. 7).

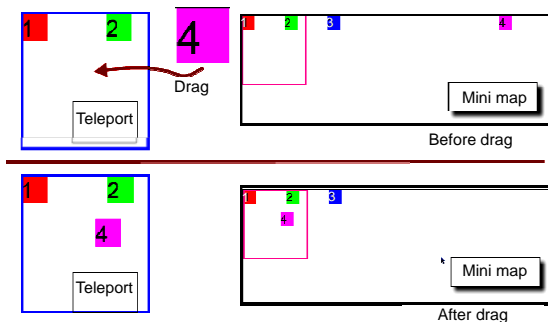


Fig. 7 Drag operation with Javelin

Furthermore, Javelin supports drag between teleports. Multiple Javelin instances help to manage more than one screen area. The drag between teleports enables multiple-object manipulation.

4.4 Multiple Javelin instances

One Javelin instance provides a view and management interface for a remote region of screen. With multiple instances, it will be easy for a user to handle more than one area simultaneously. The targeted regions may be out of user's arm length, or just on the other screen display.

Multiple Javelin instances also provide for collective work environment support to group members. It has been proposed that participants prefer to sit across from each other when casually conversing or competing, but choose to sit side by side for collaboration (Sommer, 1969). Compared with MDEs, the large screen display environment with a high resolution enables group members with side-by-side positions enough local workspace and an explicitly public shared space. For every single user, with Javelin instances, a member is able to focus on his/her private work and have an overview of group task status (via the mini map); groups of people can share their work results, and exchange their ideas concerning one task. Furthermore, they can work together to finish common tasks simultaneously (by having teleports targeting a common area).

5 Applications with Javelin

There are numerous programs worth being transplanted to large display environments. However, in the transplanting process, a lot of design and programming work is required. The transplanting is always time-consuming and laborious.

The Javelin is designed to be a program level library that helps to minimize the time and labor cost in transplanting applications. Temporally, we have a Javelin library implemented in Adobe Flex platform.

To evaluate to what extent the Javelin as a library is able to enhance user interactions of existing applications in large screen display environments, we experimentally integrated Javelin with two Flex RIAs.

5.1 CReader with Javelin

CReader is an RIA book reading application that we designed for Internet browsers. In the application, the user navigates resources in an image wall composed of book covers in a digital library. Clicking one book cover, a real book similar widget arises, in which scanned images of the book act as book pages. Users can activate multiple book widget instances for comparative reading, and drag the book widgets to any position on the screen.

The CReader can also work on large screens. The prior design is for single medium-sized monitors. Thus, the full screen space, especially the buttons above the book widget, is designed to be interactive. As introduced in Section 2.1, in our large wall-sized display system, only the lower half of the screen space is touch sensitive. Thus, we integrated the CReader system with Javelin to solve the problem. The result is shown in Fig. 8. Interacting within the touchpad, users can use the cursor in the whole screen space. All the interactions that CReader supports are also available through Javelin interface.

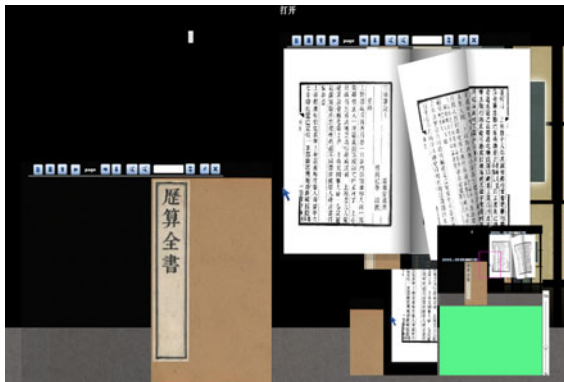


Fig. 8 CReader with Javelin

The CReader with Javelin is an example showing what the function of touchpad can do to extend the interaction field to the whole display space. The task to integrate the CReader with Javelin costs one programmer 1.5 h.

5.2 FunnyDraw with Javelin

FunnyDraw is an online open source Flex-based drawing application. With FunnyDraw, users can draw some simple diagrams, such as triangle, circle, and rectangle, to express workflow, mind map, as well as basic math concept.

In FunnyDraw, the control buttons, such as the ‘create a triangle’ button, and drop down menus, are placed in the top and left areas of the screen. For a user of the large screen, these areas are hard to reach. With the help of Javelin, the control widgets are ‘hyperlinked’ using teleport widget (Fig. 9). The user can select diagram elements from the teleport widget and drag them to his/her local workspace, without stepping aside or stretching the arm to click command widgets.

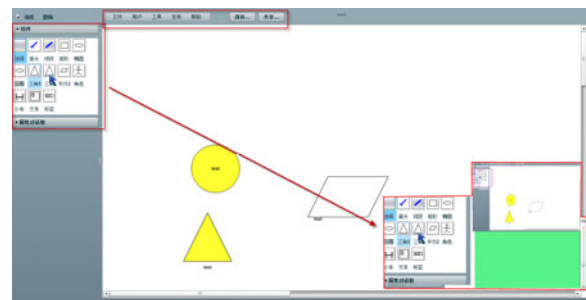


Fig. 9 FunnyDraw with Javelin

The task to integrate FunnyDraw with Javelin costs the same programmer 0.5 h. The reason for different time spent on the two applications lies in their different display layers. In Javelin, the dynamic thumbnail of the screen context updates every frame. For CReader, as a 3D application, the time spent on configuring the mini map capturing is the main time-consuming process.

6 Experiments

Javelin provides quick access to screen display spaces and interaction methods for contents out of user arm length. In practice, the Javelin decreases the bodily physical movement required, extends the arm interaction field, and speeds up the screen widget acquisitions.

To evaluate its benefits, we chose the frequent operation pick and drop as a test project. The time spent on finishing the task and body movement were chosen as the measuring standards.

6.1 Experiment design

The Javelin provides a number of functions to help users interact with widgets across the screen space. We designed the experiment to discover

Javelin's advantages in the following two aspects: (1) body movement distance, and (2) widget acquisition and manipulation.

We used a Wiimote multi-touch whiteboard to simulate the large interactive screen (Lee, 2007). The whiteboard consists of a front projector with its curtain, a Wiimote mounted above, and an infrared ray (IR) pen in the user's hand. The projector provides 1.5 m×2.3 m size, 1024×768 resolution displays. The Wiimote recognizes the movement of the IR pen in the user's hand, and communicates with the computer via Bluetooth. With the IR pen, the user can move the cursor, select the widget, and do other mouse functions in the interactive whiteboard system. The system delay of this whiteboard system, i.e., time between user operations and the mouse event recognized, is negligible.

We invited 10 participants (seven male and three female) to take part in our experiment. All of them were right handed, between the ages of 20 and 30. They all had experience in using the interactive whiteboard. The test program GUI is as shown in Fig. 10.

The task reflects some representative computer operations, such as copy and paste and moving icons to folders.

In this experiment, the pick and drag operations via two interaction manners were conducted and compared.

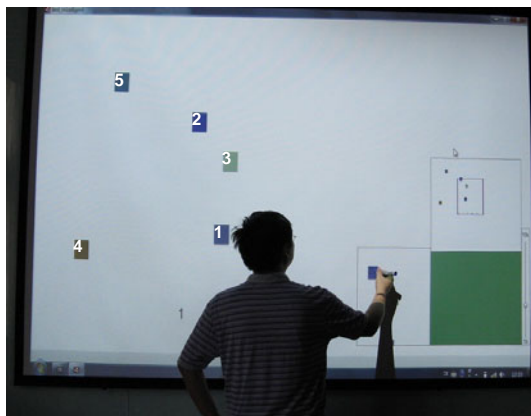


Fig. 10 Pick and drag via Javelin experiment

Several colored boxes with large font digital labels (i.e., 1, 2, 3, ...) reside randomly on the screen. The same digits are duplicated and arranged closely at the bottom screen. The participants' task is to pick the colored boxes, drag it to the corresponding digit at the bottom, and drop it on its position. If the box is not placed correctly, the box will return to its original position

The direct-screen-manipulation manner requires the user to pick and drag operations directly in screen space. The operation consists of searching the box, approaching movement, stretching arm, tapping/clicking, and moving the box to its target position. The Javelin method is to pick and drag widgets via the Javelin interface. The Javelin manipulation requires the user to finish the task with the help of Javelin. The Javelin widget is placed near the bottom labels of the workspace. Initially, teleport is targeted to the top-left screen region. Users can use the mini map to change the teleport content, manipulate the boxes inside teleport, and move them to their corresponding positions.

We chose four different numbers of boxes, 5, 10, 15, and 20, to investigate how screen content thickness affects interaction speed. For each participant, the experiment was divided into three procedures.

1. Direct screen manipulation. The user was required to perform four trials directly pick and drag operations on 5, 10, 15, and 20 boxes respectively. For each trial, the number of user body movements was observed and recorded. The time spent on every trial was also recorded.

2. Javelin manipulation. After a 10-min rest, the participant was required to perform another four trials with Javelin manipulation. The number of body movements, as well as the time spent, was also recorded.

3. User survey. At the end, a user survey was conducted in the form of a face-to-face interview, which is designed to collect participant feedback.

Before the experiment, we had several hypotheses. First, with Javelin, the time spent on each object manipulation is shorter than that with direct manipulation. In other words, Javelin decreases average manipulation time. Second, with Javelin, the body movement is decreased compared to direct manipulation. Third, as the number of boxes grows, and the screen display space becomes denser, the participant will spend more time on accessing every object.

6.2 Experiment results

We measured the time spent in each trial (Fig. 11). From the data shown in Fig. 11, we can calculate the average time spent on each box, as shown in Fig. 12. The body movement frequency records are shown in Fig. 13.

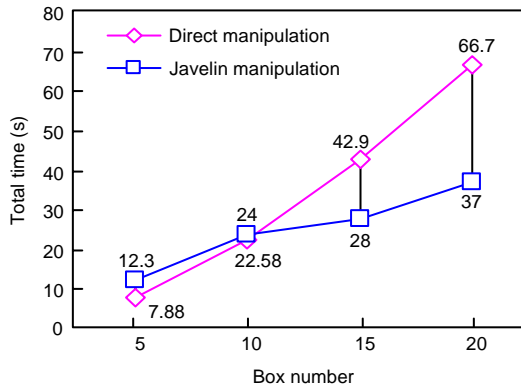


Fig. 11 Time spent in each trial

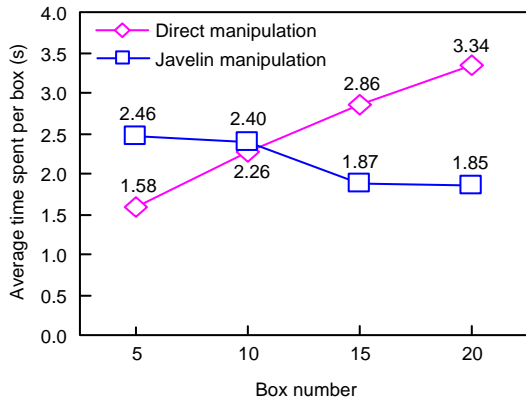


Fig. 12 Average time spent per box

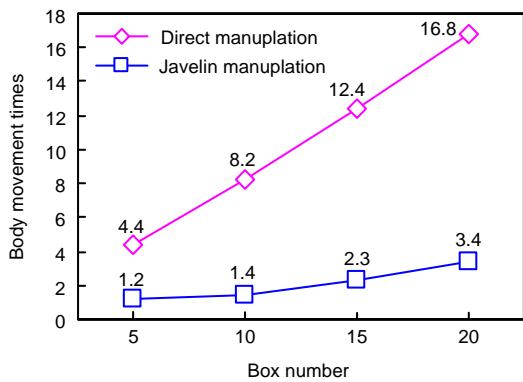


Fig. 13 Body movement times in each trial

In the first four trials using direct manipulation, the time spent per trial and user body movement roughly increased linearly with the number of boxes. In Fig. 12, as the box number increased, the average operation time on each box also increase. This result confirms the third hypothesis mentioned in Section 6.1.

In our observations, for direct manipulation trials, users finished their tasks as follows: search, click, drag, and place the boxes in the target position. The search step took about 35% of the total time.

Almost all the participants manipulated the boxes one by one and in numerical order. Only one female participant did her task area by area. Her time spent on body movement was much less than average, but also showed the same trend as shown in Fig. 12.

For Javelin manipulation, the time spent increased with the number of targets, while the body movement remained low.

In Fig. 11, the time spent with Javelin in each trial was on the whole less than that in direct manipulation. However, in Fig. 12, when the box number was low, the average manipulation time using Javelin was higher than direct manipulation. Only on the condition that the box number was larger than 10, was the average time using Javelin reduced. The first hypothesis was confirmed.

In our observation, the participants finished their task with Javelin as follows: search for the target in the mini map, click the mini map to change the region of teleport, drag the widget out of teleport, and drop to the correct position.

Since in the mini map widget the boxes and their digit labels were easily identified, the participants seldom needed to step back to obtain an overview, and the body movement remained low. The second hypothesis was confirmed.

Finally, in Fig. 11, when the box number was small, it took much more time to use Javelin than directly dragging and dropping the widget. The data stressed the mini map access time in Javelin.

The mini map interaction and teleport content replacement operations were observed to take a significant percentage of the time. However, the operations required in our experiment did not grow linearly as more boxes became involved, though there was a potential relationship with the screen display size.

6.3 User survey

Our participants gave us a lot of valuable suggestions, such as those mentioned in Section 4.2.

Most participants thought that Javelin decreased the physical exertion involved. They were interested in the teleport widgets and gave good evaluations of these.

7 Conclusions and future work

In this paper, we introduced a new user interface, called 'Javelin'. It is designed to provide quick random screen region access, remote widget acquisition, and manipulation, as well as interaction between a remote widget and a local workspace. The two integrated programs, as well as the experiments, show that Javelin is easily integrated as a program library into existing applications, with user interaction enhanced and optimized in a large display environment.

In the future, we plan to investigate several improvements for Javelin:

1. Adapting the touchpad movement magnification factor. In Javelin, the touchpad movement magnification factor controls cursor movement speed. Currently, its value is initialized to be the quotient of screen resolution width and touchpad width. Users can actively adjust the value in different scenarios. However, in experiments, we found users seldom adjusted the vertical slider.

Prior work on screen cursor movement and target acquisition (Guiard *et al.*, 2004; Khan *et al.*, 2005) has been proposed. The magnification factor should be adjusted automatically according to screen content and the distance from the cursor to the interactive widgets.

2. Initial setup for teleport. In Javelin, teleport is an important component as a hyperlink port to a remote area. The initial position of its target is worth studying. With user behavior analysis and prior application-related knowledge, the initial target region of teleport will be optimized.

References

- Albinsson, P., Zhai, S., 2003. High Precision Touch Screen Interaction. Proc. SIGCHI Conf. on Human Factors in Computing Systems, p.105-112. [doi:10.1145/642611.642631]
- Badillo, B., Bowman, D.A., McConnel, W., Ni, T., Silva, M., 2006. Literature Survey on Interaction Techniques for Large Displays. Available from <http://eprints.cs.vt.edu/archive/00000925/> [Accessed on Oct. 20, 2006].
- Ball, R., North, C., 2005. Analysis of user behavior on high-resolution tiled displays. *LNCS*, **3585**:350-363. [doi:10.1007/11555261_30]
- Benko, H., Wilson, A.D., Baudisch, P.M., 2006. Precise Selection Techniques for Multi-Touch Screens. US Patent 20070247435A1.
- Bezerianos, A., Balakrishnan, R., 2005. The Vacuum: Facilitating the Manipulation of Distant Objects. Proc. SIGCHI Conf. on Human Factors in Computing Systems, p.361-370. [doi:10.1145/1054972.1055023]
- Elvins, T.T., Nadeau, D.R., Schul, R., Kirsh, D., 2001. Worldlets: 3-D thumbnails for wayfinding in large virtual worlds. *Pres. Teleoper. Virt. Environ.*, **10**(6):565-582. [doi:10.1162/105474601753272835]
- Guiard, Y., Blanch, R., Beaudouin-Lafon, M., 2004. Object Pointing: a Complement to Bitmap Pointing in GUIs. *Proc. Graphics Interface*, **62**:9-16.
- Johanson, B., Hutchins, G., Winograd, T., Stone, M., 2002. PointRight: Experience with Flexible Input Redirection in Interactive Workspaces. Proc. 15th Annual ACM Symp. on User Interface Software and Technology, p.227-234. [doi:10.1145/571985.572019]
- Khan, A., Fitzmaurice, G., Almeida, D., Burtnyk, N., Kurtenbach, G., 2004. A Remote Control Interface for Large Displays. Proc. 17th Annual ACM Symp. on User Interface Software and Technology, p.127-136. [doi:10.1145/1029632.1029655]
- Khan, A., Matejka, J., Fitzmaurice, G., Kurtenbach, G., 2005. Spotlight: Directing Users' Attention on Large Displays. Proc. SIGCHI Conf. on Human Factors in Computing Systems, p.791-798. [doi:10.1145/1054972.1055082]
- Lee, J.C., 2007. Johnny Chung Lee>Projects>Wii. Available from <http://johnnylee.net/projects/wii/> [Accessed on Aug. 11, 2008].
- Malik, S., 2007. An Exploration of Multi-Finger Interaction on Multi-Touch Surfaces. PhD Thesis, University of Toronto, Toronto, Canada.
- Malik, S., Laszlo, J., 2004. Visual Touchpad: a Two-Handed Gestural Input Device. Proc. 6th Int. Conf. on Multimodal Interfaces, p.289-296. [doi:10.1145/1027933.1027980]
- Matejka, J., Grossman, T., Lo, J., Fitzmaurice, G., 2009. The Design and Evaluation of Multi-Finger Mouse Emulation Techniques. Proc. 27th Int. Conf. on Human Factors in Computing Systems, p.1073-1082. [doi:10.1145/1518701.1518865]
- Morris, M.R., Lombardo, J., Wigdor, D., 2010. WeSearch: Supporting Collaborative Search and Sensemaking on a Tabletop Display. Proc. ACM Conf. on Computer Supported Cooperative Work, p.401-410. [doi:10.1145/1718918.1718987]
- Moscovich, T., 2007. Principles and Applications of Multi-Touch Interaction. PhD Thesis, Brown University, Providence, RI, USA.
- O'Hara, K., Glancy, M., Robertshaw, S., 2008. Understanding Collective Play in an Urban Screen Game. Proc. ACM Conf. on Computer Supported Cooperative Work, p.67-76. [doi:10.1145/1460563.1460576]
- Peltonen, P., Kurvinen, E., Salovaara, A., Jacucci, G., Ilmonen, T., Evans, J., Oulasvirta, A., Saarikko, P., 2008. It's Mine, Don't Touch!: Interactions at a Large Multi-Touch Display in a City Centre. Proc. 26th Annual SIGCHI Conf. on Human Factors in Computing Systems, p.1285-1294. [doi:10.1145/1357054.1357255]
- Schikore, D.R., Fischer, R.A., Frank, R., Gaunt, R., Hobson, J., Whitlock, B., 2000. High-resolution multiprojector

- display walls. *IEEE Comput. Graph. Appl.*, **20**(4):38-44. [doi:10.1109/38.851748]
- Sommer, R., 1969. Personal Space. The Behavioral Basis of Design. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA.
- Stoakley, R., Conway, M.J., Pausch, R., 1995. Virtual Reality on a WIM: Interactive Worlds in Miniature. Proc. SIGCHI Conf. on Human Factors in Computing Systems p.265-272. [doi:10.1145/223904.223938]
- Stødle, D., Hagen, T., Bjørndalen, J.M., Anshus, O.J., 2008. Gesture-based, touch-free multi-user gaming on wall-sized, high-resolution tiled displays. *J. Virt. Real. Broadcast.*, **5**(10):0009-6-15001.
- Tan, D.S., Meyers, B., Czerwinski, M., 2004. WinCuts: Manipulating Arbitrary Window Regions for More Effective Use of Screen Space. Conf. on Human Factors in Computing Systems, p.1525-1528. [doi:10.1145/985921.986106]
- Wei, B., Silva, C., Koutsofios, E., Krishnan, S., North, S., 2000. Visualization research with large displays. *IEEE Comput. Graph. Appl.*, **20**(4):50-54. [doi:10.1109/38.851750]
- Wilson, A.D., 2004. TouchLight: an Imaging Touch Screen and Display for Gesture-Based Interaction. Proc. 6th Int. Conf. on Multimodal Interfaces, p.69-76. [doi:10.1145/1027933.1027946]

Journals of Zhejiang University-SCIENCE (A/B/C)

Latest trends and developments

These journals are among the best of China's University Journals. Here's why:

- *JZUS (A/B/C)* have developed rapidly in specialized scientific and technological areas.
JZUS-A (Applied Physics & Engineering) split from *JZUS* and launched in 2005
JZUS-B (Biomedicine & Biotechnology) split from *JZUS* and launched in 2005
JZUS-C (Computers & Electronics) split from *JZUS-A* and launched in 2010
- We are the first in China to completely put into practice the international peer review system in order to ensure the journals' high quality (more than 7600 referees from over 60 countries, <http://www.zju.edu.cn/jzus/reviewer.php>)
- We are the first in China to pay increased attention to Research Ethics Approval of submitted papers, and the first to join CrossCheck to fight against plagiarism
- Comprehensive geographical representation (the international authorship pool enlarging every day, contributions from outside of China accounting for more than 46% of papers)
- Since the start of an international cooperation with Springer in 2006, through SpringerLink, *JZUS*'s usage rate (download) is among the tops of all of Springer's 82 co-published Chinese journals
- *JZUS*'s citation frequency has increased rapidly since 2004, on account of DOI and Online First implementation (average of more than 60 citations a month for each of *JZUS-A* & *JZUS-B* in 2009)
- *JZUS-B* is the first university journal to receive a grant from the National Natural Science Foundation of China (2009-2010)