

Direct Interaction with Large-Scale Display Systems using Infrared Laser Tracking Devices

Kelvin Cheng, Kevin Pulo

School of Information Technologies
The University of Sydney
NSW 2006, Australia

{kcheng, kev}@it.usyd.edu.au

Abstract

Existing large scale display systems generally adopt an indirect approach to user interaction. This is due to the use of standard desktop-oriented devices, such as a mouse on a desk, to control the large wall-sized display. By using an infrared laser pointer and an infrared tracking device, a more direct interaction with the large display can be achieved, thereby reducing the cognitive load of the user and improving their mobility. The challenge in designing such systems is to allow users to interact with objects on the display naturally and easily. Our system addresses this with *hotspots*, regions surrounding objects of interest, and *gestures*, movements made with the laser pointer which triggers an action, similar to those found in modern web browsers (e.g. Mozilla and Opera). Finally, these concepts are demonstrated by an add-in module for Microsoft® PowerPoint® using the NaturalPoint™ Smart-Nav™ tracking device.

Keywords: Direct Interaction, Large Scale Display, Infrared Laser Pointer, Infrared Tracker System, Presentation control, Human Computer Interaction, Information Visualisation

1 Introduction

Large scale display systems spanning an entire wall are widely used in many modern information technology facilities especially for non-interactive purposes such as presentations. Where they are used interactively, the user interaction devices typically consist of a standard keyboard and mouse. However, there are a number of reasons why these devices are less than optimal for large displays. From the outset, in 1968, Douglas Engelbart developed the mouse to provide a way for users to interact with their personal computer (Engelbart and English, 1968). It was never designed to be used in a large display environment. It is expected that systems designed from the ground up for large scale display environments will perform better than their desktop counterparts.

Imagine a user giving a talk to an audience while their presentation slides are being shown on a large screen

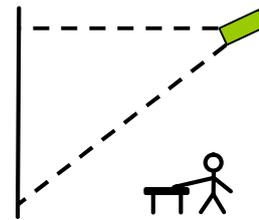


Figure 1: A schematic representation of existing large-display interactive system.

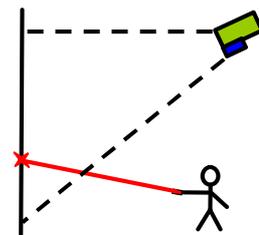


Figure 2: An example of direct interaction with large-display.

(figure 1). To use the mouse, the user needs to place it on a desk or a flat surface. This constrains the user to stay within arm's reach of the table and thus reducing the mobility of the user. To interact with the system, the user moves the mouse horizontally across the table, while the cursor on the display moves vertically. The user also needs to spend a small fraction of time considering how their mouse movements will be mapped onto the large display, although this effect is reduced with practice. Yet another problem is the need to turn around every time to see where the pointer is on the large display. Thus the mouse is not optimal for interacting with large displays.

A better approach is a system that allows for direct interaction between the user and the object seen on the large display. For example, pointing at the display using fingers or a laser pointer (figure 2), or to rotate objects by twisting, pushing or turning of the hands. In general, we need a device that allows the user to interact directly with the display, without the need for an intermediary device. Such systems are more natural and easier to use.

Over the years, various novel input devices have been developed to address this problem, with varying degrees of success. A comprehensive overview is outlined in (Jacob, 1996). We now consider some devices that provide a somewhat more direct approach than the mouse.

The *lightpen* is one of the first pointing devices produced (Myers, 1998). It works by pressing the pen against a CRT display which is operated by a switch on the pen and allows it to capture light produced from the screen. However, it is not suitable for prolonged use due to its poor ergonomics, thus it was eventually replaced by the mouse. In addition, it provides an indirect method to interact with the computer.

Light gun technology has been used extensively, especially in the computer gaming industry. Although the accuracy is adequate, the major drawback is that it must be used with a CRT display. Thus large-scale displays cannot be used since images are provided by a data projector.

The *Polhemus FasTrack* and *Logitech 3D Mouse* belong to a category of industrial strength tracking systems, primarily designed for 3D motion tracking, such as found in applications like *CavePainting* (Keefe et al., 2001). The former uses electromagnetic signals to track the position in space of a receiver, while the latter uses ultrasonic signals. They are also used for CAD object manipulation and Virtual Reality, but are typically cumbersome and expensive.

The *Logitech Spaceball*, *Gyration Gyromouse* and *Interlink RemotePoint* represent another category of input devices designed for personal use as a mouse replacement. The Spaceball is a device with a ball-shaped controller mounted on top. It allows users to push, pull and twist the ball in order to manipulate on-screen objects. It is designed for 3D model manipulation and provides a more natural movement for the user. The Gyromouse is based on a technology called GyroPoint that use gyroscopes to detect angular movements of the device. The RemotePoint allow users to roll their thumb around a soft rubber pointing button fitted onto a handheld device. The advantages over the previous set include their ability to be wireless, more affordable and natural, although the user still interacts through an intermediary device.

Another category of input devices tracks the position of the head of the user and moves the mouse cursor on the display correspondingly. This is primarily developed to provide full mouse control to people who cannot use their hands but have good head control. Devices in this category consist of the Synapse Head Tracking Device, Origin Instruments HeadMouse and NaturalPoint Smart-Nav.

Eye tracking devices follow the movement of the pupils and can in principle be used for cursor control (Sibert and Jacob, 2000). This technology is beneficial within a confined, controlled space but there is currently no evidence to support its use on a large display. Another promising input system is voice-recognition and although recognition accuracy is improving, these systems are primarily command-based (Jacob, 1996), thus indirect.

Touch screens provide an excellent solution to the problem of direct interaction and provides high precision (Sears and Shneiderman, 1991) but again, they do not scale well to large displays as manufacturing such large touch display is not feasible (even with technology such as *DiamondTouch* (Dietz and Leigh, 2001) or *Smartskin* (Rekimoto, 2002)). Even if it were possible, the user cannot reach the top of the display nor pace across from side to side. The same can

be said of specialised *whiteboards*, such as *MimioMouse* (Mimio, 2002), *Xerox Liveboard* (Elrod et al., 1992) and *Flatland* (Mynatt et al., 1999), which allow the user to press against the wall at specific location on the display with a pen-like device.

One device that is attracting an increasing amount of research is the laser pointer. These have the advantages of mobility, direct interaction, and being comparably inexpensive with the notable disadvantages of lag and the instability with the human hand. Many studies into these systems have been carried out. Dwelling is a popular technique for interaction (Kirstein and Muller, 1998), and Olsen investigated the effect of lag with this method (Olsen and Nielsen, 2001), which led to a discussion on the use of visible and invisible laser pointers (Cavens et al., 2002). The problem of hand jitter was presented in (Myers et al., 2002, Peck, 2001).

This paper focuses on methods for interacting with large displays using a more direct approach as explored during the course of a computer science honors project. The method chosen is an infrared laser pointer and associated tracking device where its use in the area of large-scale display systems will be examined. Section 2 presents the terminology for such systems. The interaction paradigms for large display direct interaction are then discussed in section 3. The details of our LasIRPoint implementation will be given in section 4. Finally, section 5 presents an evaluation of this system.

2 Terminology

This section provides an understanding for the definition of each element involved in the systems we examined. Since these terms have different meanings to different people in different domains, here we present how we have defined them.

Space. A logical space for an input or output device. We consider only bounded 2d planes.

Display. Any sensory output device. For example, monitor, 3d sound, haptic, cave. We will only deal with 2d planar monitor type displays.

Large-Scale Display. We are only interested in displays that are larger than the size of a person. This presents the problem of reachability – such displays are too tall for the user and too wide, out of arm's length.

Pointing Device. A user input device (hardware) which can be used to locate a particular point in space. For example, mouse, laser pointer, Gyropoint, Headtracker, 3d tracker.

Latency. The time between when the user moves the pointing device, and when the display updates to reflect this change (if any update is necessary).

Interaction. Refers to the cycle between the user observing the output display and using their input device to make changes, and the system's response to these changes.

Direct vs Indirect Interaction. The amount of physical disparity between the pointing device space and the output

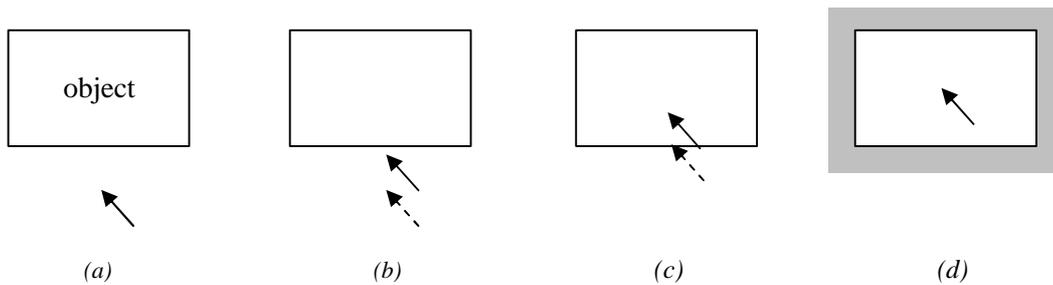


Figure 3: (a) Initial state where the black arrow represents the position of the invisible laser beam, and is not currently selecting the object. (b) To select the object, the user moves their laser beam towards the object. However, due to latency, the system is not yet aware of these changes, and so keeps the original beam position, illustrated with a dotted arrow. (c) As the pointer moves inside the object, the system detects the crossing of the boundary by the laser beam. (d) The system reacts to this crossing and highlights the object, while the laser beam of the pointer stops at the centre of the object.

space, and the wrapping between them. For a touch screen, the input space is the same as the output space, giving direct interaction. Whereas for a mouse, the input space is the flat horizontal surface of a table and the output space is the vertical surface of a monitor. We can see that these two spaces do not overlap, giving an indirect interaction. Although indirect interaction requires a higher level of cognition, it is reduced with practice.

Pointer Instability. The unsteadiness of our hands affects the exact location of the laser beam on the screen. This effect increases as the distance between the wall and the user increases.

3 Interaction Paradigm

3.1 Challenges Addressed

Pointer Instability. A potential problem arises since our hands are unsteady (Peck, 2001), making the input device unstable and reducing the accuracy of the cursor position on screen. The problem of hand jitter is significant with the laser pointer because it is held in mid-air by the user without any support (Peck, 2001). Such jitter results in a slight difference away from the target and is therefore significant (Myers et al., 2002). In addition, this jitter increases when the user moves further away from the screen (Peck, 2001). A moving average filter was proposed by Myers et al. (2002) but the accuracy did not improve significantly. We therefore aim to reduce this effect so that the interaction method can be as accurate as possible.

Latency. It is anticipated that some amount of lag will be involved due to slow system response. Video based tracking systems typically work between 25 and 50 frames per second. This corresponds to latency between 20 and 40 milliseconds simply from video frame updates. In addition there will be time taken to determine the position of the pointing device and computing the on screen cursor position. All of these cumulate to confuse and disorient the user, since the cursor is not where they expect to be.

Selection. Most input devices require a button for clicking so that items on the screen can be selected. However, a laser pointer does not have a button for selecting objects.

An obvious solution is to add a button to the laser pointer as experimented by Cavens et al. (2002). Unfortunately this requires too much engineering. Furthermore, in most cases, a cable is required for such solution to work. One button in most laser pointers that should not be overlooked is the on/off switch. It is possible to use this switch to act as a button (Kirstein and Muller, 1998). However as suggested by Peck (2001) when the button is released, the beam often moves away from the target before it goes off, therefore not providing a good indication of the intended selection. Dwell clicking is a possible candidate for replacing a button click. However, the major disadvantage of such technique is that it takes at least 2 seconds to make a dwell selection (Peck, 2001). In light of these issues, we choose to avoid clicking as much as possible and instead provide a paradigm for direct interaction without clicking.

3.2 Proposed Solution

We address the above challenges by hiding the on screen cursor and laser pointer, not requiring clicking, and using *hotspots* and *gestures*. Each of these will now be discussed in detail.

By not displaying the on-screen cursor, the problems of hand jitter and high latency are hidden from the user. This can also be achieved by having an invisible infrared laser pointer. This is desirable as the user will not be distracted by these, allowing them to focus instead on the on-screen objects.

However, these policies introduce other problems. The most important is how users can select a particular object without an on-screen cursor and without a button for clicking. We solve this through the use of hotspots and gestures to perform selections.

Hotspots. Hotspots are areas around objects which change appearance when the pointer enters them. Such changes include highlighting with a coloured background. This provides a mechanism for objects to be selected without clicking. Technically, the hotspot resides inside a bounding box around an object. When the pointer hits the boundary of this bounding box, the object will be selected (figure 3). As a result of the unavoidable lag, the system detects the crossing of the boundary after the pointer

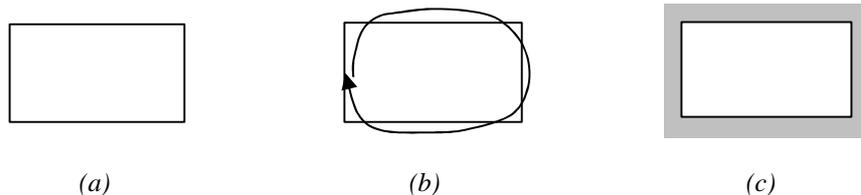


Figure 4: (a) Initial state of an object. (b) To select the object, a circling gesture is used. (c) The final state of the object after selecting.

actually crosses it. Therefore when the system detects the hotspots activation, the user's pointer is likely to be near the centre of an object. This is ideal because people tend to point towards the centre of an object, rather than the edges. Thus highlighting when a hotspot is entered will counteract the effect of the lag. The object reverts to its original appearance when the user points away from it.

Gestures. Gestures are natural movements of the hand (as indicated by the path traced by the pointer) which the system recognises, allowing an action to be performed. Such methods have been used successfully in web browsers such as Mozilla and Opera. The idea here is to also use gestures to select objects by circling around the object (figure 4). Another possible use of gestures is for navigation. It is natural to move forward to the next piece of information by using a left to right sweeping gesture and move backward by a right to left gesture.

4 Implementation

The proposed solution discussed in the previous section is evaluated with an implementation in a presentation environment. Application to other domains is also possible, such as large graphs manipulation where users can select a particular node or a group of nodes by circling them. However, existing techniques such as hotspots and gestures may not be enough to interact with the visualization on display adequately. In such situation, alternative techniques need to be discovered, for example providing a mechanism for dragging and droppings nodes or clusters of nodes so that they can be moved around the display as necessary.

In our presentation environment, we have implemented these ideas as a software add-in to Microsoft PowerPoint called *LasIRPoint* which uses an infrared tracking device *Smart-Nav* which was developed by NaturalPoint (figure 5). It is a hands-free infrared mouse replacement device designed to allow computer users with disabilities to control a mouse pointer by moving their head, hand, or fingertip. It works by using an infrared camera to track the movements of small dots of reflective material which are



Figure 5: Infrared laser tracking device *Smart-Nav*.

illuminated by infrared LEDs. Alternatively, an active source of infrared illumination can be used, such as an infrared LED light. This technology gives good results at a fraction of the cost of more professional alternative mouse and movement tracking systems as discussed in section 1. As a result, this input device is used for the evaluation.

The software implementation has used Visual Basic for Application (VBA) for PowerPoint and the ActiveX tracker camera API supplied by NaturalPoint to control and receive data from the *Smart-Nav* tracking camera.

There are three important features in our implementation. The first is that objects on a PowerPoint slide will be highlighted temporarily when the pointer moves inside the bounding box of an object. When the pointer leaves the object, it will revert to its original appearance. This makes use of the hotspot technique. The second is the gesture based selection where the user makes a circling action with the laser pointer and the object being circled will be highlighted permanently (even when the pointer leaves). Selected objects can be un-highlighted by making the same circling gesture again. Finally, the user can move between slides in the slideshow by dwelling on either of two arrows situated in the bottom corners of the display.

4.1 Highlight

Each PowerPoint slide consist of user defined set of *shapes*. A shape can be a title box, textbox, rectangle, circle or any other shapes supported by PowerPoint. In our system, these shapes are referred to as *original shapes*. For each original shape, we create a *highlight box*, initially invisible (that is, 100% transparent). These boxes are needed so that highlighting can be performed independent to the shape of the object, making it much easier and more consistent to modify the state of the highlights.

When the cursor moves into the bounding box of a shape, the highlight box of that shape will be highlighted with grey (figure 6). To ensure that the highlight stands out

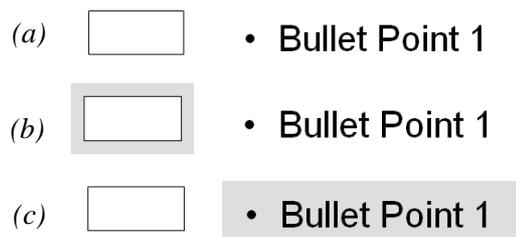


Figure 6: (a) A screen shot illustrating the initial state of a rectangle and a textbox. (b) The rectangle is being selected. (c) The textbox is being selected.

(especially with a solidly coloured original shape) the highlight boxes are wider and taller than the bounding box.

4.2 Gestures

To recognise the circling gesture, we need a gesture library such as LibStroke (Willey, 1997). In our paradigm, we receive a continuous stream of coordinates, and thus the start and end coordinates of any gestures are not as clearly defined as they are in such gesture libraries. In addition, if the laser pointer is assumed to be switched on at all times, the start and end coordinates are recorded when the laser pointer leaves the viewing area of the camera, which is highly unsatisfactory. Another possibility is if the user is allowed to switch the laser pointer on and off at will, we cannot be certain when such an action takes place. It is therefore possible that more than one gesture has been performed during that time. Due to these restrictions we have decided not to use these libraries, but to develop our own method. The only gesture that needs to be recognised is the circling action, which simplifies our problem.

The solution is simple. The bounding box is separated into four rectangles evenly divided at the centre (figure 7a). These rectangles are termed “gesture boxes”. To select an object, the user needs to point their laser pointer to all four gesture boxes of that object in order. Further more, their pointer needs to stay within the bounding box of the shape for the duration of the gesture (figure 7b). When successful, the colour of the highlight box will change to yellow, indicating that the shape is highlighted permanently (figure 7d).

When circling, users tend to make a loop around the

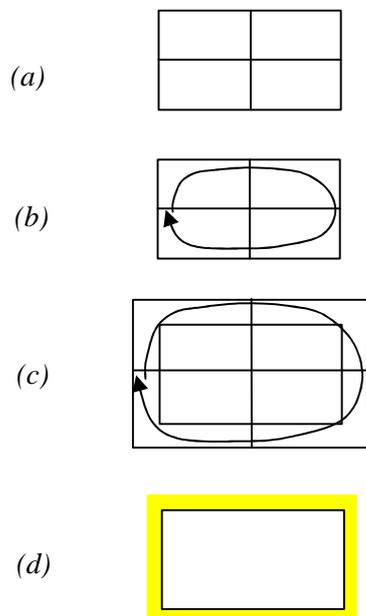


Figure 7: (a) The bounding box of an object is divided into four “gesture boxes”. (b) To select the object, all four boxes must be in the path of the gesture. (c) To make it easier to perform gestures, an external box is implemented, allowing users to go outside the boundary of the object. (d) A successful selection of the object.

outside of an object rather than inside. Therefore, we have created yet another bounding box named “external box” for each object. This box is significantly larger than the original bounding box by around 20 pixels in width and height (figure 7c). With this configuration, they will only be deemed to have left the shape when they have also left the external box, making it easier to perform a successful gesture.

4.3 Navigation

Initially, we intended to use sweeping gestures for changing to the next and previous slide. However, these gestures are not feasible for the following reasons. Navigation gestures rely on the start and end coordinates; the system will need to see if they lie approximately on a straight line and determine if it spans the whole width of the display. As a result, a gesture library is required and as mention previously this is not easy to achieve. Second, after implementing gesture boxes, we realised that the circling action may interfere with the sweeping action. What happens when a shape is as wide as the whole screen? During a circling gesture, parts of it may appear to be sweeping actions. This causes a large amount of false slide changes. Therefore, we have used dwell clicking instead, similar to active regions as proposed by Sukthankar, Stockton and Mullin (2000). We have added arrows to the bottom corners of the display to allow users to navigate between slides. A right arrow is placed at the bottom right of the display and a left arrow at the bottom left. To move to the next slide, users point to the right arrow and dwell until the system reacts after a set period of time t . Similarly, pointing to the left arrow moves to the previous slide.

4.4 Technical difficulties and limitations

While all efforts have been made to improve the efficiency of the implementation, there nevertheless still exist some unavoidable technical difficulties which are out of our control. Two major limitations have been identified. Smart-Nav is designed for use by individuals at a distance of less than approximately 2 metres; it has a low resolution of 256 x 256 pixels. This is marginally adequate for the application on a large display at a distance of around 3 metres. This alone does not pose a great concern to our implementation. The more serious issue is the fact that the camera has difficulties when tracking small objects smoothly. In some cases, the camera may lose tracking altogether, often caused by bursts of frames between periods of inactivity. This is a hardware issue and is therefore outside the scope for this paper.

5 Evaluation

Intuition and Introspection. Testing were done extensively during and after the development of the system. The inability to capture the infrared pointer smoothly is a problem as it contributes to the lag of the system. However, because the laser pointer and the cursor on screen are hidden, users will have no knowledge of how much lag that is actually present. Lag is therefore not a major issue to the user. On another issue, it is actually quite difficult to select the objects using the circling gesture as it is extremely easy to accidentally leave the external boxes.

Therefore, these boxes may need to be expanded or perhaps an altogether different gesture technique used.

Informal evaluation. An open day was held by the Information Visualisation Research Group at the School of Information Technologies. As members of the group, we set up a demonstration for anyone interested to try the system and for us to see what their thoughts are. It was a very useful session and a lot of helpful feedback was received. It was done as an informal observation since a full HCI study was beyond the scope of the project.

- **“Where is it?”** -The issue most frequently commented on was the inability to see the position of the cursor. Users are left with no choice but to guess where it is or search for it by moving the pointer. Most users commented that they would very much like to see a red dot shown on the screen or some sort of cursor shown from within the system. A computer controller cursor would have latency issues, so perhaps a visible laser pointer could be used to provide this visual feedback.

- **Mouse case confusion** - Another cause for concern is the fact that the infrared laser pointer is enclosed in the case of a computer mouse (figure 8). Users tend to continuously click and release the mouse button as they would when they are using a computer mouse. This tremendously decreases the smoothness of the system. Some suggested that it might be better if it was enclosed in a pen like case (such as a typical red laser pointer). One user followed on to say that it would be good if it could be used in a shooting type gaming environment, where game players can aim and shoot using this laser pointer. This could be a good replacement to the light gun technology as it does not require any cabling. In addition, the light gun requires a CRT as the display type and is therefore inappropriate for large displays.



Figure 8: An infrared laser pointer enclosed in a mouse case.

- **Occlusion** - It was observed that users tend to walk up to the display wall obscuring both the tracking camera and the projector. This is usually because they do not know where they are pointing. They have the intuition that they will know the pointer location more accurately when they move closer to the display. Most of the time it is the hand holding the pointer that is in the way rather than the whole body.

- **“Cool!”** - Most people said that this system is “cool” and would definitely use it in their own presentation if it

was available to them. They liked the idea of using the laser pointer as an alternative to the mouse for presentations and having it actively interact with the on screen objects. They found it entertaining to circle the objects and see them change appearance. However, this is only true if the problems limiting the usefulness of this particular implementation are resolved.

6 Conclusion and Future work

6.1 Conclusion

After evaluating the system, it can be seen that the paradigm and the system presented does indeed improve on conventional mouse-based systems. Hotspots and gestures play a crucial role in replacing the traditional mouse click to select on screen objects. However, due to technical difficulties and limitations in our implementation, the system did not perform perfectly. These techniques show promise for future development. If combined with improved hardware and a more robust implementation, an effective, usable and more direct system would result.

6.2 Potential future applications of this system

The following is a list of applications that seem promising for future uses of our interactive paradigm.

Graph movement: By using the laser pointer to select individual nodes or clusters of nodes, graphs can be directly manipulated in 2D or 3D space using such actions as rotation and transformation. In such a scenario, hotspots and gestures may be used for selection. As the laser beam enters the bounding box of one particular node, that node will be selected. To select a cluster of nodes, a circling gesture may be used. Extending the idea further, a freeform shape may be drawn so that all nodes residing in its interior will be selected. Manipulation can be achieved through dwell clicking so that the selected nodes can be rotated, for example. The nodes will then follow the laser pointer and rotate around a fixed point, such as their centre of mass. Such techniques are often found in image editors. It is also envisioned that the selected nodes on a graph can be dragged around on the display. Again, using dwell clicking, nodes can then follow the position of the laser pointer around the display. This provides a mechanism for interacting with large graphs on a large screen at a distance.

Focus and context: Laser pointers can be used to point directly at an area of interest, focusing in on this area by magnifying it in some way. Particular techniques of potential include the Document Lens (Robertson and Mackinlay, 1993), Fisheye View (Furnas, 1986) and the Hyperbolic Browser (Lamping, 1995). The area around the position of the laser pointer acts as the focus of the visualisation. As the pointer moves around the display, the focus moves accordingly.

Selective dynamic manipulation: Gestures can be used to select arbitrary object sets (Chuah et al., 1995). These object sets can be manipulated and their appearance changed so that different tasks can be performed. In

addition, gestures may be used to perform various operations on the selected object sets.

Collaboration: Multiple users can interact with the large display simultaneously using multiple laser pointers (Chen and Davis, 2001). Multiple objects can therefore be selected. However, this introduces the problem of tracking multiple laser pointers. There is also the problem of handling an object when it is selected at the same time by two users.

See-through tools: By using two laser pointers and gestures, users can use the Toolglass paradigm to select application tools interactively (Bier et al., 1993).

Flow menus: Direct manipulation, menu selection and text entry are merged so that they can be performed without the use of a button click. This is expected to be extremely useful in laser pointer systems (Guimbretière and Winograd, 2000). It uses gestures made by the laser pointer to deduce the command being selected. It also uses the concept of hotspots to allow the system to know which command wedge has been entered.

6.3 Future work

Video Camera instead - We have chosen to use an infrared camera which tracks objects using hardware, SmartNav, because the cost was much less than a video camera. We expected that an infrared tracking camera would simplify the image processing. However, the limitation of the tracker is a major contributor to the poor performance of the system. It is the inability of the tracker to smoothly track a single small dot, and its limited resolution that caused unstable tracking. Instead a consumer handheld video camera or other CCD camera can be used to track the position of the infrared dot (Kirstein and Muller, 1998, Olsen and Nielsen, 2001, Sukthankar, Stockton and Mullin, 2000). Additional processing would be required to capture the laser dot in each frame; however, it would provide higher resolution, which increases the quality of the tracking.

Automatic Keystone calibration - Keystone correction was not implemented into the LaserIRPoint system because the issue of distortion was dominated by limitations in the tracker. However, we need to manually adjust the height and angle of the tracker every time before it is used, so that the bounded tracking camera observes the same area as the output display. This process can be avoided if an automatic keystone correction calibration module is implemented, alleviating the user's need to precisely setup the camera's position. Such system was implemented by Sukthankar, Stockton and Mullin (2000). Users of such systems will only need to have the camera facing approximately the correct direction, and the calibration required is computed by the system. This both saves time and produces a much more accurate mapping between the camera and the output display.

Visible laser pointer - From the evaluation, we can see that most users would like to visual feedback on where they are pointing. One solution is to attach a visible red laser to the infrared laser pointer and allow it to be switched on and off by the user. This should increase the

performance of the user (Cavens et al., 2002). In addition, we could add an on/off switch to the infrared pointer so that it can stay on continuously rather than having to press for some duration.

Acknowledgements

We would like to thank Judith Dawes from the Department of Physics at Macquarie University for her invaluable technical expertise with lasers. We would also like to thank Peter Eades for his many useful discussions and encouragements and Masahiro Takatsuka for help on the final paper submission.

References

- Bier, E. A., Stone, M. C., Pier, K, Buxton, W., & DeRose, T. D. (1993): "Toolglass and magic lenses: The see-through interface", *Proceedings of SIGGRAPH '93*, p.73- 80.
- Cavens, D., Vogt, F., Fels, S., and Meitner, M. (2002): "Interacting with the big screen: pointers to ponder", *ACM SIGCHI '02 Conference on Human Factors in Computing System, Companion Proceedings*, p.181-182.
- Chen, X. and Davis, J. (2001): "LumiPoint: Multi-User Laser-Based Interaction on Large Tiled Displays", Technical report, Stanford University.
- Chuah, M. C., Roth, S. F., Mattis, J. and Kolojejchick, J. (1995): "SDM: Selective Dynamic Manipulation of Visualizations", *Proceedings of UIST'95 Conference*, p.15-17.
- Dietz, P.H. and Leigh, D.L. (2001): "DiamondTouch: A Multi-User Touch Technology", *ACM Symposium on User Interface Software and Technology (UIST)*, p.219-226.
- Elrod, S., Bruce, R., Gold, R., Goldberg, D., Halasz, F., Janssen, W., Lee, D., McCall, K., Pedersen, E., Pier, K., Tang, J., and Welch, B. (1992): "Liveboard: A large interactive display supporting group meetings, presentations and remote collaboration", in *Proceedings of ACM CHI'92 Conference*, pp. 599-- 607.
- Engelbart, D. and English, W., "A Research Center for Augmenting Human Intellect", *Reprinted in ACM SIGGRAPH Video Review, 1994.*, 1968. 106
- Furnas, G.W. (1986): "Generalized fisheye views," *SIGCHI'86 Conference on Human Factors in Computing Systems*, ACM Press, p. 16-23.
- Guimbretière, F. and Winograd, T. (2000): "FlowMenu: Combining Command, Text, and Data Entry", *Proceedings of the 13th annual ACM symposium on User interface software and technology 2000*, p.213-216
- Jacob, R. J. (1996): "Human-Computer Interaction: Input Devices", *ACM Computing Surveys*, 28(1): p.177-179.
- Keefe, D., Acevado, D., Moscovish, T., Laidlaw, D., and LaViola, J. (2001): "CavePainting: A Fully Immersive 3D Artistic Medium and Interactive Experience",

Proceedings of the 2001 Symposium on Interactive 3D Graphics, pages 85-93.

- Kirstein, C. and Muller, H. (1998): "Interaction with a projection screen using a camera-tracked laser pointer", in *Multimedia Modeling; MMM '98 Proceedings*, p.191-192
- Lamping, J., Rao, R., and Pirolli, P. (1995): "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies", *Proceedings of CHI '95*, ACM Press, p.401-408
- MimioMouse, <http://www.mimio.com/> Accessed 3 Oct 2002.
- Myers, B. A. (1998): "A Brief History of Human Computer Interaction Technology", *ACM Interactions*, 5(2), p. 44-54.
- Myers, B.A., Bhatnagar, R., Nichols, J., Peck, C.H., Kong, D., Miller, R., and Long, A. C. (2002): "Interacting At a Distance: Measuring the Performance of Laser Pointers and Other Devices", *ACM CHI 2002 Conference on Human Factors in Computing Systems*, p.33-40.
- Mynatt, E.D., Igarashi, T., Edwards, W.K., and LaMarca, A. (1999), "Flatland: New Dimensions in Office Whiteboards", in *Proceedings of CHI'99*. pp 346--353.
- Olsen Jr, D.R. and Nielsen, T. (2001): "Laser Pointer Interaction", *ACM CHI'2001 Conference Proceedings: Human Factors in Computing Systems*, Seattle, WA: p.17-22.
- Peck, C. (2001): "Useful Parameters for the Design of Laser Pointer Interaction Techniques", in *Proceedings of CHI 2001*, p.461-462
- Rekimoto, J. (2002): "SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces", *ACM CHI 2002 Conference on Human Factors in Computing Systems*, p.113-120.
- Robertson, G. G. and Mackinlay, J. D. (1993): "The Document Lens", In *Proceedings of UIST '93*, pages 101-108.
- Sears, A., Shneiderman, B. (1991): "High Precision Touchscreens: Design Strategies and Comparisons with a Mouse", *International Journal of Man-Machine Studies*, 34 (4), 1991, p. 593-613.
- Sibert, L. E. and Jacob, R. J. K. (2000): "Evaluation of Eye Gaze Interaction", *Proceedings of the CHI 2000*, pages 281-288.
- Sukthankar, R., Stockton, R., Mullin, M. (2000): "Self-Calibrating Camera-Assisted Presentation Interface", *Proceedings of International Conference on Automation, Control, Robotics and Computer Vision*.
- Willey, M. (1997): "The Design and Implementation of a Stroke Interface Library", *IEEE Region 4 Student Paper Contest*.