Exploring Orientation on a Table Display

Russell Kruger and Sheelagh Carpendale
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada T2N 1N4
{krugerj, sheelagh}@cpsc.ucalgary.ca

Abstract

With the recent move towards table displays, there has been an increasing interest in examining horizontal orientation. In order to effectively understand this issue, it is important to have a foundational understanding of orientation on physical tables as well as table dependent aspects of orientation. As an initial investigation into supporting rotation of digital objects on table displays, an examination of mobile controls was carried out. A test application consisting of a map with labels that could be freely rotated was implemented and examined, and from this a general framework for approaching orientation on table displays emerged.

Keywords

Human-computer interaction, information visualization, single display groupware, roomware, ubiquitous computing, orientation, table display.

1. Introduction

The research discussed in this paper falls under the sub-disciplines of computer science known as human-computer interaction and information visualization. Human-computer interaction is the study of how people interact with computers and the extent to which computers are developed for successful interaction with human beings. Information visualization is concerned with the various ways computers can assist in enhancing people's understanding of complex and abstract information.

A significant issue related to information visualization is the screen real estate problem. Today's society is experiencing information explosion. There are vast amounts of information available and capable of being generated. Moreover, there have been rapid advances in CPU processing power, computer memory and graphics capabilities; gains which have not been matched with proportional gains in display size. The end result is that the size of display screens is a significant limitation in the research of complex visual information spaces; there is simply too much information to represent in too small an area.

Large-scale displays have been examined as a possible solution to this problem. Although display size has not increased at a rate that has matched the performance of computer hardware, there have been some moderate advances in this area. Additionally, there has been an increasing move to incorporate displays into the surrounding environment, a concept referred to as ubiquitous computing. Associated with this is the idea of roomware, or the combining of information devices and physical objects (such as walls, chairs, and tables) in a room. A natural extension of these ideas is the incorporation of a display into a table.

As well, a table seems to be a natural medium for group collaboration. This type of collaboration involves groups of people exploring and examining the same visual field. A model for supporting such collaborative group work is single display groupware (SDG). SDG typically refers to computer programs that enable co-present users to collaborate via a shared computer with a single shared display and simultaneous use of multiple input devices.

2. Understanding Orientation on Tables

As part of an undergraduate honours thesis course, a table display known as the e-Table was designed, constructed and empirically evaluated. As a result of this empirical evaluation, it was determined that orientation of digital objects on the e-Table is a significant research issue. In order to effectively understand this issue, a foundational understanding of orientation on physical tables is required.

2.1 Orientation on a Physical Table

Part of the motivation for the creation of the e-Table was the belief that an electronic table could effectively support and enhance work done on physical tables. According to Tang [14] who investigated design activity on a physical table, orientation is both a problem and a potential resource. It is problematic in that people do not share the same orientation; hence, it is difficult for multiple individuals to share a common physical view and to add drawings in proper orientation relative to existing sketches. However, orientation can also be used as a resource. For instance, sometimes sketches are drawn facing a particular team member, so as to establish a context and audience for that drawing. Also, orientation can be used to indicate whether a drawing is personal (oriented towards the drawer and within one's own personal space) or shared (oriented towards others and outside one's personal space).

In addition, when using tables collaboratively, people often arrange their seating to allow for more than one person to view objects of interest at a slightly compromised orientation; for instance, by sitting kitty-corner and placing an object of interest so that it is oriented in the direction of the corner between them. As well, when people share or examine objects, they often use partial rotation of objects (rotation to non-90-degree angles) as opposed to strict 90-degree angle rotation.

2.2 Orientation on a Table Display

As with a physical table, the general problem of orientation on electronic tables is simply this: two or more people sitting at different sides of the table cannot view objects of interest at the same orientation simultaneously without having multiple copies of information. During the empirical evaluation of the e-Table, it was found that horizontal orientation was indeed problematic. In general, people felt far more comfortable dealing with oriented objects on a traditional upright display. Although it took some people longer than others to notice the effect that orientation had, all people in the study commented on it. Viewing text upside down on the e-Table concerned some people more than others; some people were so disturbed viewing text upside down that they tried to stand up and turn their head in an effort to view the text in its correct orientation. People also thought that viewing text that wasn't upside down but rather sideways was still more difficult than viewing text that was right side up. These effects were greatest for clearly oriented information, such as text. Objects that lack a predefined orientation, such as geometric shapes, may not be as strongly affected. Navigation of menus that are upside-down in particular was found to be quite difficult, due to the resulting change in ordering amongst elements of the menu.

2.3 Table Dependent Aspects of Orientation

The empirical study also indicated that orientation of objects on the screen can be affected by the physical size and design of the table. If a table is large enough to support the desired number of collaborators on one side of the table, then orientation is no longer an issue. Other seating arrangements, such as individuals sitting opposite or kitty-corner to one another result in different orientation paradigms. Rectangular tables with four distinct seating positions give rise to four distinct orientations, as do rectangular displays (regardless of the shape of the table itself, such as circular vs. square). People felt that if the e-Table's display was circular or square, then this may have resulted in interaction that was not as strongly affected by orientation. It should also be noted that the casing of the display itself can contribute to orientation: having display adjustment controls visible strongly implies one position as having the "correct" orientation.

As well, the input devices used may also affect orientation. Tables that require mice input somewhat constrain the number of distinct orientations to the number of sides of the display. This is because mice usually require a rectangular region with a definite top and bottom in which to operate. An input device such as a stylus or finger is not similarly affected.

3. Related Work

The problem of orientation on a table was first introduced and discussed by Tang [14]. Since this time, only cursory investigations of this issue have been made. The most pertinent work here is that of Fitzmaurice et al. [2], which introduces the concept of a rotating user interface. It discusses various design issues for rotating user interfaces, including the idea of self-rotating or self-orienting user interface elements, and distinguishes between widgets that are rotation sensitive (i.e. have a predefined orientation), and those that are rotation insensitive (i.e. have no predefined orientation). It suggests that the scope of creating rotating user interfaces is clearly challenging, and concludes that a new user interface paradigm and toolkit that supports rotation in its core architecture is necessary.

Additionally, Hinckley et al. includes a discussion of variables that are important to consider when changing orientation: these include affording quick, informal sharing of the display; minimal rotation overhead; and allowing an intuitive rotation mechanism that does not interfere with collaborative conversation [7]. Hancock discusses technology that allows horizontal displays to detect the location of its various users, thereby allowing for automatic orientation of digital objects [5]. Other papers discuss methods and devices that can be used to produce two-dimensional rotation, including a digital pen, a touch sensitive overlay that is gesture sensitive, and various augmented mice [1, 6, 8, 9].

4. Research Directions

Results of the empirical study involving the e-Table indicate that incorporating a mechanism to allow digital objects to change orientation so that individuals seated at different positions can view them in their correct orientation is something that is necessary to support on a digital table. However, there are numerous issues related to this that require attention. For instance, is rotation of individual objects on the display more preferable than rotation of entire windows or even the entire screen? What sort of mechanism can be used to allow for quick and intuitive rotation of objects? Is having multiple copies of the same information in different orientations a feasible solution? If change of orientation is supported, will this result in decreases in productivity or efficiency because of the constant requirement to reorient objects?

As an initial investigation into supporting rotation of digital objects on the e-Table, an examination of mobile controls was carried out. Mobile controls are currently being studied as part of a possible solution to the screen real-estate problem [11]. Allowing user interface elements, such as buttons, icons and other widgets to vary in size, position and orientation may allow interface designers to more effectively maximize use of display space. The direction taken in this research was to examine the orientation aspect of mobile controls, which necessarily involved positional considerations as well.

Due to the relative lack of research on horizontal orientation, as well as the seemingly myriad of questions surrounding this issue, it was decided that a prototype application be developed that incorporated rotation of discrete objects as a starting point. It is often the case that having a concrete example of a particular concept aids in positively solidifying and directing abstract notions and questions; as such, this approach seemed particularly justified. Additionally, given the need to address interaction concerns such as how best to control rotation, as well as the scope of creating a more generalized solution [2], a broader implementation was not possible within the time constrained nature of this research.

After the implementation was completed, an informal analysis of the rotation of discrete objects was carried out by members of the University of Calgary's Interaction Lab (iLab).

This discussion served to provide feedback on the approach taken, as well as directions for future research.

5. The Application

In order to allow for empirical investigation of the research directions given above, it was required that an appropriate test application be developed. Currently, all major operating systems do not support generalized rotation of user interface elements or controls. As a result, it was necessary to create an image that looked and behaved like a control and that could be freely rotated. An example of a standard control, a button, was chosen and implemented (see Figure 1).

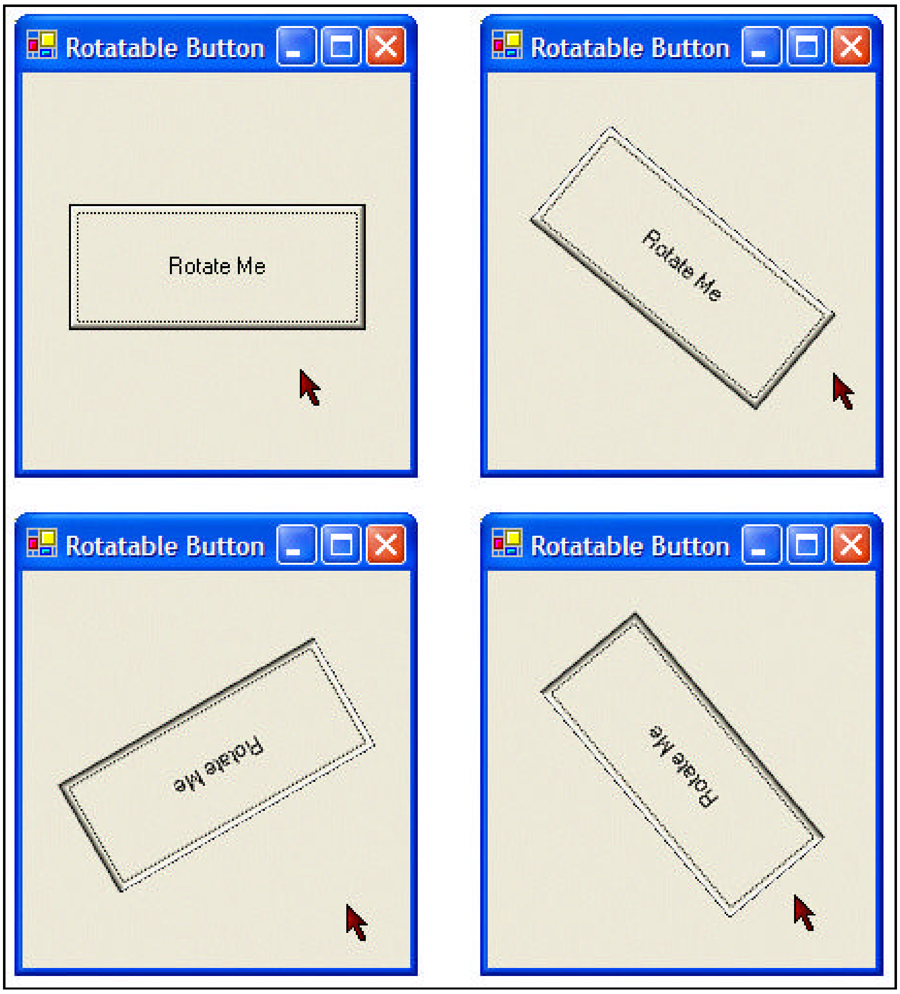

Figure 1: Rotatable Button

The button served as a proof-of-concept of a rotatable control. A point-in-polygon algorithm was implemented that enabled the button to determine mouse clicks at any angle of rotation. Several mechanisms were investigated to control the button's rotation including: right-clicking once to produce a clockwise rotation of a certain number of degrees; rotating the button clockwise while the right mouse button was held down; and using the middle scroll wheel to rotate the button. It was found that using the scroll wheel was markedly better than the other two schemes: it allowed the button to be rotated in either direction at any point in time; as well, the rate of rotation could more easily be manipulated.

Once it was determined that it was possible to create a rotatable control, the button was modified and integrated into a suitable application that could be used to examine the control. Suitable in this sense means that the application made use of discrete objects that could be rotated and whose rotation made sense, and that the application could be run on the e-Table. It was decided that a map satisfied these basic criteria. Maps are typically viewed on a horizontal surface, and oftentimes include labels that are aligned along roads, rivers, etc. A map of Canada including rotatable labels of the major cities was created. In order to allow for simultaneous user interaction, the map was made SDG aware by use of an SDG Toolkit created by Edward Tse at the University of Calgary. Support for two mice was incorporated, thereby allowing individuals seated opposite each other to examine the map simultaneously. Rotation of the labels was accomplished by clicking on the label of interest, and using the mouse's scroll wheel to rotate it. Labels could be rotated to non-90-degree angles as pictured in Figure 2 below.

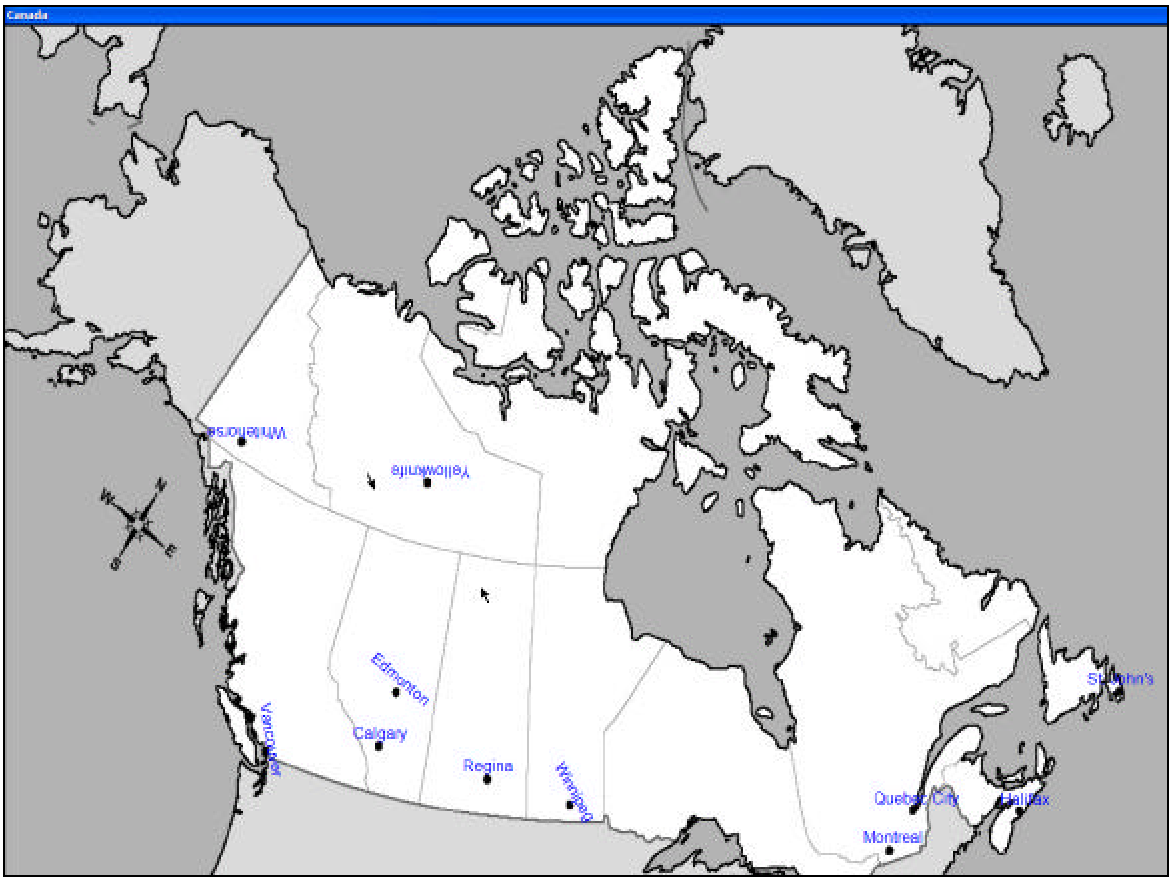

Figure 2: Map of Canada

6. Implementation Details

Both the rotatable button and the map were implemented using Microsoft's Visual Basic .NET, and were run on Windows XP. Visual Basic .NET was chosen as it can access the .NET graphics framework, which allows images and text to be drawn at various angles (see Figure 3 below for sample code).

```
'Responsible for drawing the map and labels
Private Sub Form1_Paint(ByVal sender As Object, ByVal e As System.Windows.Forms.PaintEventArgs)
Handles MyBase.Paint

        'Draws the Canada map
        Dim pic As New Bitmap("Canada.jpg")
        e.Graphics.DrawImage(pic, Form1.Left, Form1.Top, Form1.Width, Form1.Height)

        'Iterates through a collection of labels and draws each at the proper rotation
        Dim tempRotObj As RotatableObject     'Each label is a RotatableObject
        For Each tempRotObj In rotatableObjectCollection
                e.Graphics.TranslateTransform(tempRotObj.center.X, tempRotObj.center.Y)
                e.Graphics.RotateTransform(tempRotObj.angle)
                tempRotObj.Draw(e.Graphics)
                e.Graphics.ResetTransform()
        Next

End Sub
```

Figure 3: Sample Code

In terms of portability, use of the SDG Toolkit is restricted to Windows XP; hence so too is the application. Performance remains a significant issue: incorporation of multiple mice introduces significant response delays while rotating labels.

In order to detect mouse clicks at various angles of rotation, a point-in-polygon algorithm was implemented that computed the sum of the angles made between an input point (i.e. from a mouse click) and each pair of points making up the rectangular region bounding a label. If this sum came to 2 PI, then the test point was an interior point; if the sum came to zero then the test point was an exterior point. See Figure 4 below for implementation of this algorithm.

```
'Determines whether point is inside or outside a rectangular region defined by
'UpperLeftPoint, UpperRightPoint, LowerLeftPoint and LowerRightPoint
Public Function PointInObject(ByVal point As Point) As Boolean

        Dim angle As Double
        Dim p1, p2 As Point
        angle = 0

        p1.X = Me.UpperLeftPoint.X - point.X
        p1.Y = Me.UpperLeftPoint.Y - point.Y
        p2.X = Me.UpperRightPoint.X - point.X
        p2.Y = Me.UpperRightPoint.Y - point.Y
        angle = angle + Angle2D(p1.X, p1.Y, p2.X, p2.Y)

        p1.X = Me.UpperRightPoint.X - point.X
        p1.Y = Me.UpperRightPoint.Y - point.Y
        p2.X = Me.LowerRightPoint.X - point.X
        p2.Y = Me.LowerRightPoint.Y - point.Y
        angle = angle + Angle2D(p1.X, p1.Y, p2.X, p2.Y)

        p1.X = Me.LowerRightPoint.X - point.X
        p1.Y = Me.LowerRightPoint.Y - point.Y
        p2.X = Me.LowerLeftPoint.X - point.X
        p2.Y = Me.LowerLeftPoint.Y - point.Y
        angle = angle + Angle2D(p1.X, p1.Y, p2.X, p2.Y)

        p1.X = Me.LowerLeftPoint.X - point.X
        p1.Y = Me.LowerLeftPoint.Y - point.Y
        p2.X = Me.UpperLeftPoint.X - point.X
        p2.Y = Me.UpperLeftPoint.Y - point.Y
        angle = angle + Angle2D(p1.X, p1.Y, p2.X, p2.Y)

        If Math.Abs(angle) < Math.PI Then
                Return False
        Else
                Return True
        End If

End Function

'Returns the angle between two vectors on a plane.  The angle is from vector 1 to vector 2,
'positive anticlockwise.  The result is between -PI and PI.
Private Function Angle2D(ByVal x1 As Double, ByVal y1 As Double, ByVal x2 As Double, ByVal y2 As
Double)
        Dim dtheta, theta1, theta2 As Double

        theta1 = Math.Atan2(y1, x1)
        theta2 = Math.Atan2(y2, x2)
        dtheta = theta2 - theta1

        While dtheta > Math.PI
                dtheta = dtheta - (2 * Math.PI)
        End While

        While dtheta < -Math.PI
                dtheta = dtheta + (2 * Math.PI)
        End While

        Return dtheta
End Function
```

Figure 4: Point In Object Implementation

7. The Demonstration Session

After implementing the map, an informal demonstration session was held involving members of iLab at the University of Calgary. The purpose of this session was fourfold:

- To provide empirical feedback on the approach taken, the rotation of discrete objects.

- To brainstorm other possible approaches to the orientation problem.
- To elicit discussion on rotation interaction possibilities, as well as other issues related to orientation.
- To provide indication of possible future research directions.

Initially, members of iLab were informed of the research concerning orientation on the e-Table. They were then encouraged to work with the application in partners to get a feel for the interaction. The resulting discussion was recorded and later transcribed to allow for maximum exploration of the response.

Results of the discussion session indicated that even with rotation of city names, it was still difficult to examine the map upside down. The reason for this seemed to stem from the fact that other important details of the map, such as provinces and bodies of water, did not share the same orientation as the rotated labels. As well, ordering of provinces (i.e. from left to right) was different when viewing the map from the top. The fact that people are not used to being able to rotate elements of a display may also have played a role in this response.

8. A Framework for Approaching Orientation

As a result of feedback from the discussion session, it was determined that examination of alternative approaches to change of orientation was necessary. To provide structure to this exploration, it would seem necessary to derive a framework for approaching orientation on a table display. In what follows, a framework consisting of variations in orientation, variations in metaphor, and variations in interaction is introduced.

8.1 Variations in Orientation

There are several possible ways of supporting change of orientation on a table display. These include doing nothing, supporting full rotation of the display, supporting rotation of portions of the display, having multiple copies of information or multiple displays, as well as some combination of these approaches.

8.1.1 Do Nothing

The first, and easiest solution to implement is to provide no mechanism to rotate digital objects. The individuals using the table would be forced to adjust seating positions and/or head positions to view the data of interest. If the table was large enough to physically support all individuals on one side, then this may be a feasible solution. As well, it may also be the case that no other variations in orientation given below satisfactorily address the orientation issue.

8.1.2 Full Rotation

The second approach is to support full rotation of the display. This could either be accomplished physically, where the entire display or table could swivel to different orientations, or could be accomplished electronically, where the entire display contents could be programmatically reoriented. The physical nature of the table display is important to consider here. Some table displays, such as top-projected ones, may be particularly suited to the physical solution where the projector itself could rotate; others, such as stationary tables with embedded displays may be more suited to the electronic solution.

Additionally, full rotation of the display can either be restricted to 90-degree rotations, or allow for partial rotations as well. If partial rotations are allowed, then information contained in the corners may be lost when rotated, depending on the display size and shape (i.e. if the original displayed area is less than the total display size, or if the display is circular, then information may not be lost). 90-degree rotation on rectangular screens may be problematic, with the switch between portrait and landscape orientations being the area of concern.

Full rotation may be appropriate for examination of display contents that require the entire screen as context and that do not easily segment. However, this variation involves a "yours or mine" philosophy, thereby requiring support for appropriate sharing or turn-taking protocols.

8.1.3 Rotation of Regions

The third approach is to support partial rotation, or rotation of regions. There are two possibilities here: rotation of discrete objects and rotation of physically proximate regions, as well as some combination of these.

Rotation of Discrete Objects

As previously mentioned, this approach, involving changing the orientation of separate components on the display, was examined in this research. Although not particularly suited for the application studied, it may be beneficial in other situations. Additionally, an aspect of this variation not explored involves rotation of conceptually related discrete objects, such as groups of controls. Instead of rotating, say, individual icons on a toolbar, a possibility would be to rotate all icons on the toolbar, or the toolbar itself. Initial feedback from the discussion session indicated that people would probably prefer rotation of entire groups of related objects, as opposed to rotation of individual elements. This is partly due to the fact that ordering within the group would remain the same if the group were rotated together. Identical ordering seems important to maintain due to people's familiarity with and remembrance of sequences and actions based on a particular ordering of objects.

Rotation of Physically Proximate Regions

Part of the reason that people found rotation of discrete objects disorienting was the lack of a sufficient rotated context for the label. To address this problem, introducing rotation of physically proximate regions may be appropriate. The generalized solution here is a selectable rectangular region that could be freely rotated to whatever orientation was desired and moved around the display. Making the size of the region adjustable, as well as having an optional and varying contextual indication of its original position would add flexibility and functionality to this approach. The cognitive issue of contextual indication, or where the region's original position was and how the region relates or attaches to the other contents of the display, is one that will require significant consideration. Possibilities here include using a "twisted" connection from the region to its original position (see Figure 5 below), as well as outlining the original position and border of the rotated region with the same color. Having the region sizeable would allow individuals to select a context that was sufficient for the task being carried out. This approach would be especially appropriate for situations in which context is important, as well as situations involving individuals doing collaborative work that was slightly decoupled to allow different regions of the display to be worked on at the same time.

Figure 5: Twisted Connection

However, this approach has several potential drawbacks, a notable one being implementation. It is easy enough to create an image of a rotated region, but in order for individuals to interact with this area, it will be necessary to support functionality of the rotated region, potentially including buttons, nested menus, navigation and text entry. As well, the region may obscure portions of the display when rotated that were not originally obscured, and rotation of regions near the edge of the display may result in loss of information.

8.1.4 Multiple Copies of Information

Instead of rotating a single shared region, it is also possible to maintain multiple copies of information, each of which can be oriented a unique direction. The problem with having multiple copies of information is that collaboration involving more than a small number of individuals quickly becomes problematic. As well, having multiple copies of information may preclude gesturing during collaborative conversations as a means of directing a person's attention to a specific object or area of interest.

8.1.5 Multiple Displays

It is also possible to construct a larger table display composed of multiple smaller displays, each of which is oriented a different direction, as is done with GMD's ConnecTable [13]. Again, the problem with this scheme is that collaboration with a number of individuals is difficult if each is to have their own screen. As well, the potential advantages of the single display groupware model, as given in Stewart et al.'s paper [12], may not be realized.

8.2 Variations in Metaphor

A different approach to orientation on a table display involves variations in metaphor. Instead of using a traditional windows desktop metaphor and allowing for change of orientation of existing elements on the display, a new and differing paradigm can be used in its place. As with other variations, possibilities here range from limited modifications to more comprehensive alternatives.

8.2.1 Limited Variations

Instead of completely reworking the existing desktop metaphor, an alternative would be to use a related but unique metaphor that could be called the workbench. The idea here is that professionals such as carpenters or artists use tables and benches to support the gathering and ordering of their tools. When transitioned to a table display, the workbench would involve floating tool palettes, containing related controls, that could be freely relocated and manipulated. The underlying premise is that toolbars or controls for a table should be much more independent of existing windows to enable people to locate them as they desire, and reorient them as needed. The tool palettes would have sufficient context to maintain ordering of elements, and the relocatable nature of the toolbars would enable collaborators to share with relative ease.

Another possibility, somewhat more limited than the workbench metaphor, would be to use non-oriented icons, controls and other windows elements as much as possible to reduce the necessity of rotation. Some existing controls, such as color and shape related icons, are rotation independent. The idea here would be to devise original non-oriented controls, a partial example of which is FlowMenu [4]. FlowMenu is a radial menu consisting of 8 octants and a central rest area whose underlying shape and navigation is relatively orientation independent (see Figure 6 below).

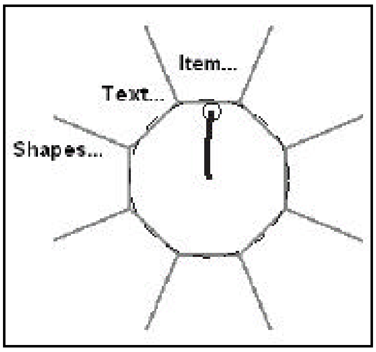

Figure 6: FlowMenu

8.2.2 Comprehensive Variations

More comprehensive variations involve gross changes to the existing desktop metaphor. An example of such an approach could be called fixed universe, in which objects in the middle of the display have a fixed orientation, and as they move to various sides of the display, they automatically rotate to the appropriate orientation. If an individual wanted to view a particular object, they could either drag it or have a collaborator push it towards them, and as the object approached, it would rotate to allow the individual to properly view it. The empirical study involving the e-Table indicated that people did in fact tend to gather objects in front of and oriented to them, while interacting with a middle portion whose orientation was fixed. Hence, this metaphor may be familiar to people.

An additional possibility here includes using a display with an electronic lazy suzy surrounding the edge of the screen, as is done with the Philips LIME project [10]. Digital objects could be placed on the lazy suzy and depending on where they were taken off, the orientation of the objects would change. As well, the representation of the object while on the lazy suzy would rotate appropriately as it traveled around the display.

8.3 Variations in Interaction

A third major approach to orientation on table displays involves variations in interaction. Specific areas of interest here include gesturing, rotation issues, and physical modes of interaction.

8.3.1 Gesturing

A possible substitute for oriented controls involves the use of gestures for input. Although gestures themselves may be somewhat oriented (i.e. a gesture may involve relative directions, such as moving down and to the left), since there exists technology for determining a user's position around a table [5], it would be possible for the table to respond appropriately to input at various locations. FlowMenu [4] is an example of gesture input that could be used with such a scheme.

8.3.2 Rotation Issues

A rotation issue of interest is that of self vs. other rotation. Prior to the empirical study involving the e-Table, it was thought that people would most often reorient objects of interest to themselves, as opposed to reorienting them for other people. However, during the course of the study it became clear that most of the time, people actually rotated objects for others to see. Hence, supporting other rotation should be carefully considered.

An associated issue is that of free vs. 90-degree rotation. Free rotation is more computationally intensive and may contribute to performance delays. As well, the quality of text and some images at non-90 degree angles can be significantly degraded. However, in the empirical study involving the e-Table it was found that individuals almost always did not rotate objects in strict 90-degree multiples. As well, it was thought that free rotation of objects would be especially required for creative endeavors.

A third rotation issue of interest is that of automatic vs. manual rotation. Incorporating a mechanism that allows objects of interest to be automatically oriented to either one's self or another individual may be a desirable option. This can be contrasted with more manual rotation mechanisms that require explicit input to direct rotation. Supporting both variations may be a suitable approach.

8.3.3 Physical Interaction

One of the reasons that people use physical tables for collaboration is that they can be used to hold objects on their surface. As such, tables offer a tactile quality that other mediums such as whiteboards do not. In order to take advantage of this quality of physical tables, a table display could make use of physical widgets or phidgets [3]. Ideas here include incorporating physical buttons, sliders, and dials off the screen that could be used to exact rotation of individual screen elements or the entire display. During the empirical evaluation of the e-Table, it was noted that having physical controls adjacent to the display (the display adjustment controls) was somewhat awkward: hence, placing the physical controls at a slight distance from the display may be appropriate.

9. Future Directions

It is difficult to gauge the relative strength of the various approaches to orientation introduced above without first viewing examples of each. Due to the lack of operating system support for change of orientation, implementing proof-of-concepts of each approach will no doubt be time and effort intensive. As such, it is recommended that one specific application be chosen and implemented for a table display that effectively addresses the orientation problem for that particular application. From this, it is thought that more general guidelines and principles will emerge, as was the case for the implementation done in this research. Waiting for operating system support before implementing an application is not a feasible approach, as integration of such support is a major endeavor and would no doubt require a compelling proof-of-concept. In terms of picking a specific application, due to the relative lack of software designed for table displays it is difficult to know the type of application that would be appropriate and successful on a table. A choice that involves support for multiple collaborators and that lacks strongly oriented components may be appropriate. Possibilities here include concept mapping software, a surveillance application, or optimization software, such as scheduling, where human input is required.

10. Conclusion

Orientation on table displays is a significant and potentially limiting problem in supporting collaboration on these displays. Possible approaches to this issue include investigating variations in orientation, variations in metaphor, and variations in interaction. Each approach offers unique advantages and potential drawbacks. At this point, it remains to be determined if table displays can effectively support multiple individuals working at different orientations on a shared task. The recommend approach is to chose and implement an application for a table display that effectively addresses orientation concerns, and from this glean generalized guidelines for approaching this issue.

11. Acknowledgements

The authors are very grateful to Mike Boyle who answered numerous technical questions, and Edward Tse who provided the SDG Toolkit. Also, the comments provided by members of the Interactions Lab at the University of Calgary were greatly appreciated.

12. References

[1] K. Evans, P. Tanner, and M. Wein. Tablet-based valuators that provide one, two, or three degrees of freedom. In *Computer Graphics*, 15(3):91–97, 1981.

[2] G. Fitzmaurice, R. Balakrishnan, G. Kurtenbach, and B. Buxton. An exploration into supporting artwork orientation in the user interface. In *CHI*, pages 167–174, 1999.

[3] S. Greenberg, and C. Fitchett. Phidgets: Easy development of physical interfaces through physical widgets. In *UIST*, pages 209–218, 2001.

[4] F. Guimbretiere, and T. Winograd. FlowMenu: Combing command, text, and data entry. In *UIST*, pages 213–216, 2000.

[5] M. Hancock. A feed forward neural network for determining a user's location. Technical Report, December 2001.

[6] C. Herot, and G. Weinzapfel. One-point touch input of vector information for computer displays. In proceedings of *Computer Graphics and Interactive Techniques*, pages 210–216, 1978.

[7] K. Hinckley, J. Pierce, M. Sinclair, and E. Horvitz. Sensing techniques for mobile interaction. In *UIST*, pages 91–100, 2000.

[8] K. Hinckley, M. Sinclair, E. Hanson, R. Szeliski, and M. Conway. The videomouse: a camera-based multi-degree-of-freedom input device. In *UIST*, pages 103–112, 1999.

[9] G. Kurtenbach, G. Fitzmaurice, T. Baudel, and W. Buxton. The design of a GUI paradigm based on tablets, two-hands, and transparency. In *CHI*, pages 35–42, 1997.

[10] Living Memory: Connecting Communities. A video produced by Philips Design, 2000, Royal Philips Electronic N.V.

[11] M. McGuffin, and R. Balakrishnan. Acquisition of expanding targets. In *CHI*, pages 57–64, 2002.

[12] J. Stewart, B. Bederson, and A. Druin. Single display groupware: A model for co-present collaboration. In *CHI*, pages 286–293, 1999.

[13] P. Tandler, T. Prante, C. Muller-Tomfelde, N. Streitz, and R. Steinmetz. ConnecTables: Dynamic coupling of displays for the flexible creation of shared workspaces. In *UIST*, pages 11–20, 2001.

[14] J. Tang. *Listing, drawing, and gesturing in design: A study of the use of shared workspaces by design teams.* PhD thesis, Stanford University, 1989.