Collaboration with DiamondTouch

Stephen G. Kobourov, Kyriacos Pavlou, Justin Cappos, Michael Stepp,
Mark Miles and Amanda Wixted

Department of Computer Science
University of Arizona

Abstract. We study the performance of collaborative spatial/visual
tasks under different input configurations. The configurations used are a
traditional mouse-monitor, a shared-monitor with multiple-mice, and a
multi-user input device (DiamondTouch). Our experiments indicate that
there is a significant variation in performance for the different configura-
tions with pairs of users, while there is no such variation with individual
users. The traditional configuration is not well-suited for collaborative
tasks, and even after augmenting it to a shared monitor with multiple-
mice it is still significantly inferior to the multi-user input device.

1 Introduction

The traditional mouse-monitor-keyboard configuration was designed for a sin-
gle user and it does not lend itself easily to collaborative spatial/visual tasks.
When more than one users interact with one computer, typically there is one
user interacting and the others are "back-seat drivers." Moreover, augmenting
a mouse-monitor setup to a shared monitor with multiple mice and keyboards
is also unlikely to significantly improve performance in collaborative tasks, com-
pared to input devices designed for collaborative use.

The DiamondTouch table [2] is a touch-sensitive input device which supports
concurrent operation by up to four users. The table detects multiple and simul-
taneous tactile events and can distinguish between each user's touch. The users
interact with the table by placing their finger(s) on the touch-sensitive surface
while sitting or standing upon a receiver pad, which closes a low signal circuit.
The DiamondTouch table offers advantages for applications that benefit from
collaboration, as well as applications that allow interaction through more than
one contact point at a time. These properties make it especially attractive for
design, architecture and 3D modeling applications.

One of the earliest uses of the DiamondTouch table was for a collaborative
game, TetraTetris [1]. In the early days of touchscreen research and development,
Schneiderman listed some of the desirable qualities of being able to use one's
fingers directly on the display screen [4]. The relationship between group size
and table size in collaborative work has also been studied [3].

We compared the user-pairs performance of spatial/visual problems using
three different input configurations: one-mouse and one-monitor, two-mice and
one-monitor, and the DiamondTouch table. Despite the familiarity of the tra-
ditional mouse interactions, the DiamondTouch setup yielded the best perfor-
mance thus confirming that it offers a significant advantage for collaborative

work on spatial/visual problems. To ensure that the results are not due to other differences between the DiamondTouch and the mouse-monitor configurations, we also tested individual users. There was no significant difference in individual performance on the mouse-monitor versus the DiamondTouch configurations.

2 Experimental Setup

In order to compare the different hardware configurations for collaborative work on spacial/visual problems, we presented user-pairs with a series of three graph problems (crossings removal) with increasing difficulty. We measured the time required to solve each problem, and recorded the mean time to completion over the three problems. We performed the experiments on three input configurations:

One-mouse and one-monitor: The first configuration was the standard one-mouse and one-monitor setup. In all the experiments we used the same computer (Dell Pentium 4 desktop running Windows XP) with a 52cm diagonal LCD display. Typically, the users ended up with one dominant user manipulating the input with the other one in a "back-seat driver" role.

Two-mice and one-monitor: The second configuration was a two-mice and one-monitor set-up, allowing both users to simultaneously interact with the system. Note that just attaching a second mouse the the computer is not a good solution as it defeats the purpose of collaboration. When two mice are attached to the same computer, one "steals" the cursor from the other, whenever movement is detected. Therefore, we modified our system to allow two mouse-cursors, each independently controlled by its own mouse, using the CPNMouse project drivers (`http://cpnmouse.sourceforge.net`).

DiamondTouch table: The DiamondTouch hardware used in this study has a surface with a 79cm diagonal and a 4:3 aspect ratio. The table is connected through a USB cable to a Dell Pentium 4 desktop PC running Windows XP. All images, which normally appear on the display monitor, are routed to a video projector that

Fig. 1. Physical setup.

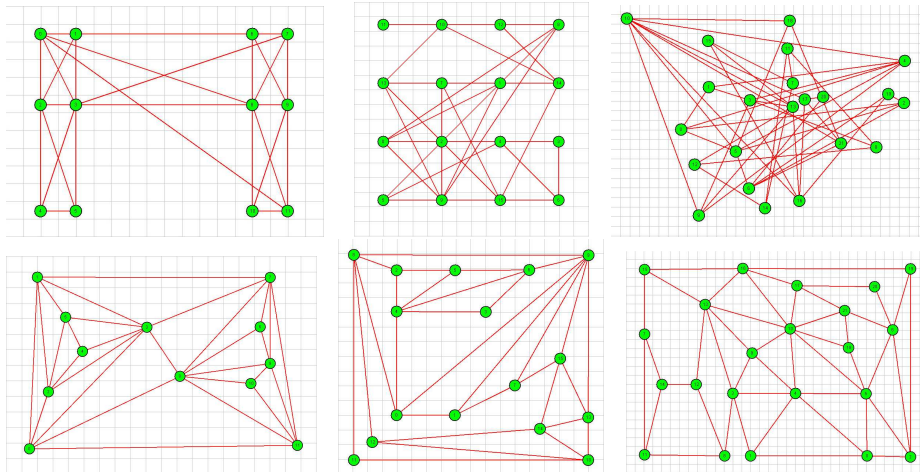projects them onto the table surface with the aid of a mirror; see Fig. 1.

Fig. 2. Each column contains an input graph and one of the possible plane drawings.

The task: Each of the three problems featured a 2-dimensional straight-line representation of a planar graph with crossings; the task was to "untangle" this graph by moving the vertices until a representation without crossings was found. Since there is no unique planar representation for the test graphs, any planar representation arrived to by the users was considered a valid solution. All graphs were constructed and solved using custom-built software which allows for easy click-and-drag manipulation of graphs. The GUI was identical in all configurations, except that in the two-mice case there were two mouse-cursors on the screen. Clicking on a node (or placing a finger on it) selects the node and dragging the mouse (finger) moves the node. The graphs are shown in Fig. 2.

The test subjects: Our users were college students who were proficient computer users. The test subjects did not necessarily have knowledge of graphs or graph theory. All test subjects were initially given a short training session (less than 5 minutes) using an easy graph problem, to allow them to familiarize themselves with the experimental setup, the software and the task at hand. The assignment of the user-pair to one of the three hardware configurations was done at random and in all three configurations user-pairs sat side by side. The users were presented with the same series of graph problems and were asked to collaborate in solving the problems. Times for each of the three graph problems were recorded and their mean was taken.

3 Experimental Results

The seven user-pairs who worked on the DiamondTouch table averaged 302 seconds per problem. The eight user-pairs who worked on the two-mouse and one-monitor configuration averaged 464 seconds per problem. Finally, the eight user-pairs who worked on the one-mouse and one-monitor configuration averaged

Result Summary			
Test type	Sample size	Mean	Standard Deviation
one-mouse (pairs)	8	480	223
two-mice (pairs)	8	464	119
DiamondTouch (pairs)	7	302	119
one-mouse (individuals)	9	536	313
DiamondTouch (individuals)	9	558	197

Fig. 3. Summary of experimental results.

480 seconds per problem. We used the robust two-sample t-test to verify that there is a statistically significant difference in the population means; see Fig. 3. The test yielded a p-value of 0.011, so at a 95% confidence level the hypothesis that the DiamondTouch setup outperformed the two-mouse and one-monitor setup for the collaborative untangling of graphs, was accepted.

There is a possibility that the better performance of the user-pairs on the DiamondTouch table is the result of the display size, or display orientation, or speed of touch-interactions compared to mouse-interactions. In an effort to eliminate these possibilities, we also compared the performance of single users on the DiamondTouch table versus the traditional one-mouse and one-monitor setup. We conjectured that the improvement in the case of collaborative use on the DiamondTouch table will not be seen in single-user interactions. With this in mind, we performed the same experiments as in the case of user-pairs and found that in the case of individual users, there was no significant difference in the mean completion time. The nine single-users who worked on the DiamondTouch table averaged 558 seconds per problem, while the nine single-users who worked on the one-mouse and one-monitor configuration averaged 536 seconds per problem. Again, using the two-sample t-test (p-value = 0.567) we verified that at a 95% confidence level there is no statistically significant difference in the performance of the DiamondTouch table versus the traditional mouse-monitor setup for the individual untangling of graphs.

References

1. C. Collberg, S. G. Kobourov, S. Kobes, B. Smith, S. Trush, and G. Yee. Tetratetris: An application of multi-user touch-based human-computer interaction. In *9th Intl. Conf. on Human-Computer Interaction (INTERACT)*, pages 81–88, 2003.
2. P. Dietz and D. Leigh. Diamondtouch: A multi-user touch technology. In *14th ACM Symp. on User Interface Software and Technology (UIST)*, pages 219–226, 2001.
3. K. Ryall, C. Forlines, C. Shen, and M. Ringel-Morris. Exploring the effects of group size and table size on interactions with tabletop shared-display groupware. In *ACM Conference on Computer Supported Cooperative Work*, pages 284–293, 2004.
4. B. Shneiderman. Touchscreens now offer compelling uses. *IEEE Software*, 8(2):93–94, 107, 1991.