# When It Gets More Difficult, Use Both Hands – Exploring Bimanual Curve Manipulation

Russell Owen, Gordon Kurtenbach, George Fitzmaurice, Thomas Baudel, Bill Buxton

Alias
210 King Street East
Toronto, Ontario M5A 1J7, Canada
E-mail: russello@acm.org; {gf | gkurtenbach}@alias.com; baudel@ilog.fr; bill@billbuxton.com

*Abstract*

In this paper we investigate the relationship between bimanual (two-handed) manipulation and the cognitive aspects of task integration, divided attention and epistemic action. We explore these relationships by means of an empirical study comparing a bimanual technique versus a unimanual (one-handed) technique for a curve matching task. The bimanual technique was designed on the principle of integrating the visual, conceptual and input device space domain of both hands. We provide evidence that the bimanual technique has better performance than the unimanual technique and, as the task becomes more cognitively demanding, the bimanual technique exhibits even greater performance benefits. We argue that the design principles and performance improvements are applicable to other task domains.

*Key words: two-handed input, bimanual input, curve editing, task integration, divided attention.*

## 1 Introduction

There have been various studies concerning interaction techniques which utilize two handed input [2-12, 16-18, 20, 21]. Three potential advantages of two handed input motivate these studies:

1. *Time-motion* The hope is that bimanual input through reduced time-motion will allow users to perform tasks faster. This reduced time-motion can be attributed to several things. First, some operations can be accomplished in parallel (such as simultaneously moving and scaling a rectangle). Another time-motion saving may result from a reduction in the amount of task switching. For example, two endpoints of a line can be moved without having to move the cursor back and forth between them.

2. *Existing skills*. Another potential advantage of two handed input is using everyday two handed actions as metaphors for computer interactions. The intention is to use these metaphors to assist in learning an interaction style. For example, the Toolglass/Magic Lenses interaction technique [6] is based on the metaphor of holding a painter's pallet in one hand while drawing or picking with the other hand. This is consistent, in general, with the everyday bimanual skills used in the physical world, as outlined by Guiard's asymmetric division of labor in bimanual action [15].

3. *Expressiveness*. This advantage is much more intangible than the previous two concepts. Informally, we define expressiveness as the ability of the interaction technique to allow the user to rapidly explore solutions and browse the data being operated on. Thus expressiveness is linked to the ability to quickly manipulate, perceive, and evaluate the transformed data. In other words, greater expressiveness increases the iteration speed and range in exploring the solution space. We give evidence in this paper, that when tasks require more interactive manipulation of the data, two-handed input can result in better performance than one-handed input.

It can be argued that two-handed input simply increases the bandwidth of input and therefore will by definition result in better performance than one-handed input. However, we have found that in practice this is not generally true. A classic criticism of two handed input has been the "tapping the head and rubbing stomach" argument. That is, two handed input is too much of a burden on the user. Kabbash et. al. [20] has shown that two hands are not always better than one. In the case where a user is working on a complicated task, two-handed input may further complicate the task by forcing the user to coordinate the actions of their hands.

In our research we have concluded that in order for two-handed input to be effective it must be designed carefully. One obvious and very general two handed design is to have two input devices (for example, two mice, one for each hand) and two cursors [10, 25, 28]. Both time-motion and existing skills advantages seem plausible with this approach. Tasks may take less time because one cursor does not have to "run around and do everything" and the notion of two cursors is metaphorical to our two hands.

However, in practice the two cursor design can perform poorly. In many cases, the user's job boils down to serially switching their focus between two cursors and serially operating each input device. The gain (not having to move one device around to do everything) may be outweighed by the cost of keeping track of and switching between two different activities.

One reason for the failure of this design is the divided attention problem (see [22] for additional information on divided attention). When there are multiple sources of information (for example, two cursors) we must make choices about what to attend to and when. This results in additional "switching costs" which in many cases outweigh the benefits [23].

We have also noticed that other two handed designs work very well and almost transparently, for example, scaling and moving a rectangle by controlling its two opposite corners [12, 9]. This case, we believe, is the opposite of the divided attention problem. The operations of moving and scaling are visually integrated and can be conceptually chunked as one task (match the current rectangle to the target rectangle) therefore switching costs are minimized.

Merging one or more sub-tasks into one integrated conceptual task is fairly common. For our purposes, we specifically define conceptual integration as when the user perceives and prefers to think of the operations of the two hands not as separate activities but as a single activity. For example, holding a nail and hitting it with a hammer can be conceptually integrating into the single activity of "hammering a nail". This is similar to Guiard's perspective on bimanual tasks [15]. Similarly, Jacob et.al. provided evidence of conceptual integration effecting computer interaction [19].

Integration can also happen in other ways besides visually and conceptually. The way in which we manipulate the input devices can be integrated or divided in the motor domain. For example, in the two cursors, two mice approach, the input domain is split into two different coordinate systems: one coordinate system for the left (the left side of your desk) and one for the right (the right side of your desk). However, if two absolute mice were used with the same physical coordinate system, we would say the input space was integrated.

The difference between the one-handed and two-handed techniques will not be completely explained by differences in time-motion. Instead, the difference may consist of one-handed time motion costs and two-handed cognitive benefits (see Figure 1).

We hypothesize that the cognitive benefits of the two handed techniques can be attributed to epistemic action. Some cognitive psychology research proposes that motor activity can be classified as either epistemic or pragmatic action [24]. Epistemic actions are performed to uncover information that is hidden or hard to compute mentally. Moreover, these physical actions make internal cognitive computation easier, faster and more reliable. For example, we sometimes use our fingers when we count. A second example notices that novice chess players sometimes physically move a chess piece, temporarily, to its new position to assess the move and possible counter-moves by an opponent. The notion is that epistemic actions can improve cognition by reducing the (1) memory, (2) number of steps and (3) probability of errors involved during mental computation. In contrast, pragmatic actions are physical actions whose primary function is to bring the user closer to the goal by physical manipulation, i.e, a particular goal cannot be accomplished in any way without this particular physical action.

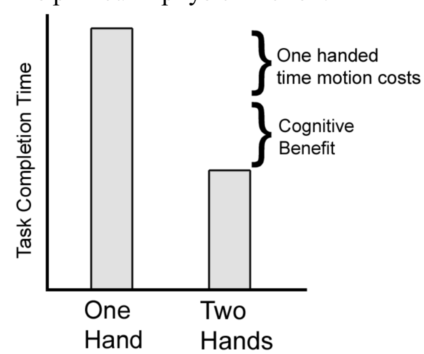


*Figure 1. A model for the performance differences between one-handed input and two-handed input.*

We believe two-handed interaction techniques offer more opportunity for epistemic rather than pragmatic actions compared to one handed techniques. This is because two- handed input inherently provides more bandwidth and therefore it is possible to manipulate data more rapidly to uncover information that is hidden or hard to compute mentally. We speculate that this ultimately results in better performance (faster task completion times and/or fewer failures).

## 2 Experiment

To test these beliefs about two-handed input, we performed an experiment which involved curve editing by manipulating control vertices (CVs). This task is commonly found in computer graphics applications and can be challenging (see Foley and van Dam [13] for more detail on curve representations). The difficulty of the task arises from the relationship between the CVs and their effect on the curve being controlled. Figure 2 show examples of how the locations of the CV interact to produce unintuitive resulting curves. While the effect of a CV on a curve is learnable, most users find it very hard to get a particular shape of a curve without some trial and error in placing the CVs.

The task in our experiment requires a subject to manipulate two CVs until the curve being manipulated (the response curve) matches a target curve. The response curve was displayed superimposed over the target curve.

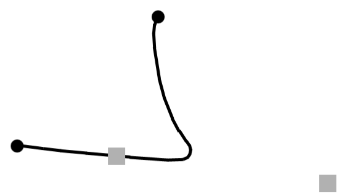

*Figure 2. Examples of how unintuitive locations of controlling vertices (squares) are needed to produce certain curves.*

We selected this task for several reasons:

1) We are interested in using two-handed input to enhance curve editing. While the task does not truly reflect a practical way to perform curve editing (for example, most curve editing involves the manipulation of more than 2 CVs), we hope that what we learn about two-handed input from this simple curve editing task can be applied to more realistic curve editing situations and different problem domains.

2) We believe that for this simple curve editing task we can create a two-handed interaction technique that supports conceptual, visual and input space integration, thus avoiding some of the pitfalls of two-handed input.

3) We believe that curves which are difficult to match will require high amounts of manipulation and exploration. We believe that this task is a strong candidate to bring out the advantages of two-handed input.

### 2.1 Set-up

The experiments were run on an SGI Indy using a 20 inch color monitor. Subjects sat at a fixed distance of 100cm from the screen. An 18x25 inch multi-device Wacom digitizing tablet was used as the input device. For the one handed case a stylus was used in the right hand. For the two handed case the stylus was held in the right hand and a small puck (approximately 1" in diameter) was held in the left hand. The wacom tablet is capable of sensing both the stylus and puck simultaneously (see Figure 3) The experiment program was written using the X input extensions and OpenGL graphics library.

### 2.2 Procedure

The task involved presenting a target curve to the user then asking them to match (overlay) the target curve with the response curve by manipulating the control vertices of the response curve. The endpoints of the response curve start at the endpoints of the target curve and require no manipulation. To complete the task the subject manipulated the two CVs which control the shape of the response curve until the curves match within a fixed error for 1/3 of a second. To eliminate the chance that the user would start a trial in random position that was close to a match position, the beginning of each trial was started with the user by placing their hand(s) in a home position and pressing down with the stylus to signal the start of the trial.

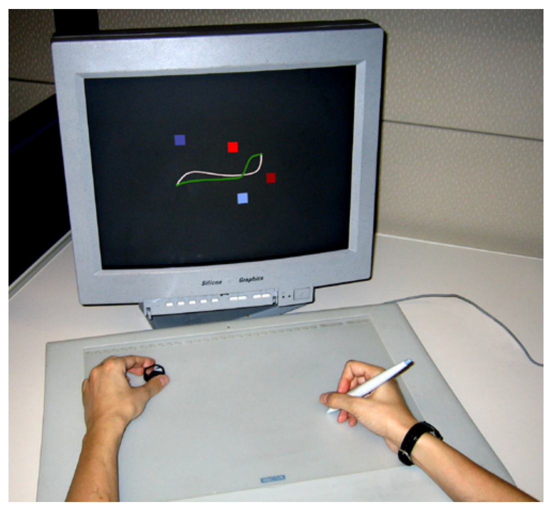

*Figure 3: The experiment setup.*

The two CVs were distinguished by color, one was red the other blue. When two hands were used to manipulate the CVs, the red CV was assigned to the right hand and the blue CV to the left.

### 2.3 Conditions

From this basic configuration we tested three input configuration conditions:

*One hand*. In this case CVs could be manipulated one at a time using the stylus in the right hand. A cursor tracks the stylus position on the tablet when the stylus is close to the tablet surface. Applying pressure to the stylus when the cursor was over a CV allowed subjects to "drag" a CV. The entire surface of the tablet was mapped to the screen of the workstation.

*Two hands, integrated device space*. In this case no CV selection was necessary since the cursors for the two input devices were automatically attached to the CVs at the beginning of the trial. The entire surface of the tablet was mapped to the screen of the workstation. Both the left and right hands operated in the same coordinate system so when the left and right hands where in the same place physically, the CVs where in the same place on the screen.

*Two hand, non-integrated device space*. The two handed non-integrated case was identical to the previous two handed case except that the tablet was logically segmented into two halves. The position of the left hand relative to the left half of the tablet was mapped to the entire workstation screen. Similarly, the position of the right hand relative to the right half of the tablet was mapped to the entire workstation screen (see Figure 4).

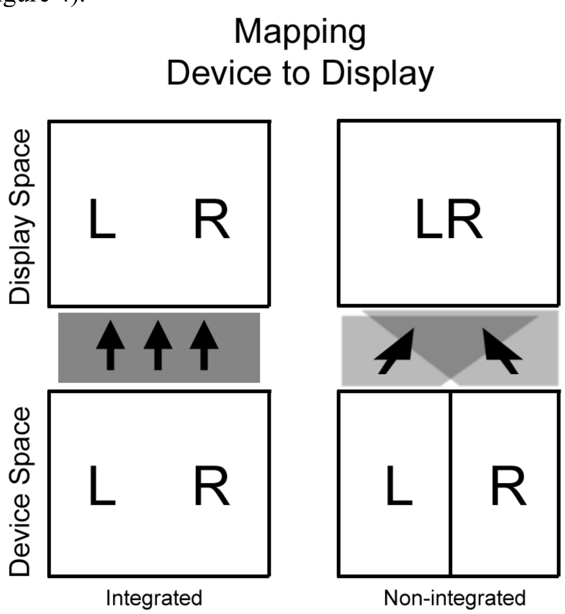


*Figure 4: The differences between integrated and non-integrated task spaces.*

These three conditions were crossed with two output configurations: with and without target CVs visible (see Table 1). The two output conditions represent two degrees of difficulty (low and high). That is, it is easier to solve the task when the feedback of the target CVs are visible compared to the more difficult condition of not having the target CVs visible.

*Target CVs visible*. In this condition the CVs for the target curve were displayed. These CVs, like the CVs for the response curve where also color coded in red and blue, but in a deeper hue. Thus matching curves in this condition simply involved moving the red CV over the dark red CV and the blue CV over the dark blue CV (see Figure 5).

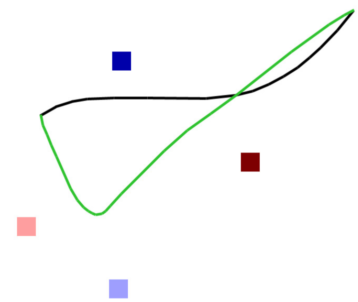

*Figure 5: View of the curve editing task with target CVs.*

*No target CVs visible*. In this condition, the target CVs were not displayed. Matching in this condition became a matter of manipulating the response CVs until the response curve matched the target curve (see Figure 6).

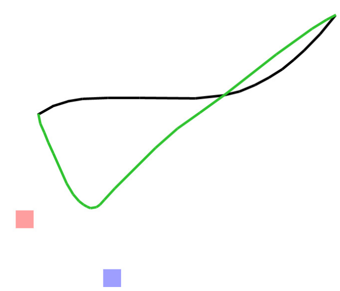

*Figure 6: View of the curve editing task without target CVs.*

| | CV targets | no CV targets |
|---|---|---|
| one handed | Condition 1 | Condition 2 |
| two handed integrated | Condition 3 | Condition 4 |
| two handed non-integrated | Condition 5 | Condition 6 |

*Table 1: The conditions of the experiment.*

## 2.4 Design

A total of 6 employees from Alias were run as subjects. They were pre-screened for color-blindness. All subject were right handed.

A within subject, repeated measures design was used. Thirty trials were run for each condition. Each 30 trials used the same ordered set of target curves. This resulted in 6 subjects x 3 input conditions x 2 levels of difficulty x 30 trials = 1080 data points. Conditions presentation ordering for each subject was counterbalanced.

The dependent variable of task completion time was measured for each trial. Subjects' subjective impressions were collected in an exit interview and survey. All input device events data (pen up, pen down, pen moved) were logged to allow separation of selection and movement time in the one handed condition.

The calculation of error between curves that matched the user's perception of curve closeness was a non-trivial task. We used the curve error function developed by Baudel [5] (see Appendix for a description of the curve matching error algorithm).

### 2.5 Hypotheses

Before running the experiment, we defined the following four hypotheses.

**H1:** The two handed conditions will yield faster task completion times than the one handed conditions.

**H2:** Even when switching costs are removed from the task completion times for the one-handed conditions, H1 will still hold.

In the one handed case subjects are forced to move CVs one at a time, switching back and forth between the two CVs (switching costs). In terms of time motion mechanics switching costs appears to be the major difference between the one-handed and two handed techniques. However, we believe that there are additional cognitive benefits in the two-handed condition which will further reduce task completion times. Therefore, even when switching costs are factored out, the two-handed technique will still outperform the one-handed technique.

**H3:** Assuming that task completion times degrade in the more difficult tasks (i.e., the no target CV visible condition), the amount of degradation will be greater for the one handed conditions than for the two handed conditions. In other words, we expect two-handed techniques to exhibit more of an advantage as the task become more difficult.

**H4:** Integrated device space conditions will yield faster task completion times than non-integrated device space conditions. The integration of the device spaces will allow subjects to utilize their sense of relative limb position in the placement of the CVs. This should result in reduced task performance time.

## 3 Results

We performed an analysis of variance on the performance data. Hypothesis 1 and 3 were confirmed while hypothesis 2 and 4 were not confirmed. Figure 7 shows the results of the six conditions in terms of overall mean task completion performance.

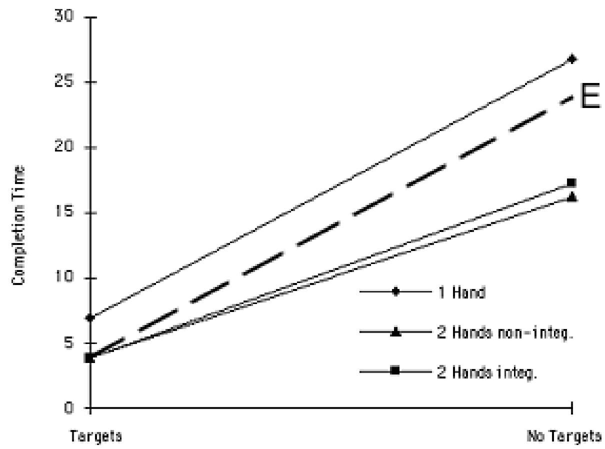


*Figure 7: Task completion times for the six conditions. Dotted "E" line represents expected performance.*

As we predicted, the absence of CV targets had a major effect on task performance $F(1, 5) = 19.475$, $p < 0.01$. Thus we conclude that curve matching without the targets was a significantly more difficult task than curve matching with the target CVs.

H1 (two-hands faster than one) was found to be true. Task completion times were significantly shorter in all the two handed conditions ($F(2, 10) = 13.685$, $p < 0.05$). On average the two handed condition was approximately 40% faster than the one handed.

No evidence was found to support H2 (two hands still faster, even after subtracting switching times in one-hand condition). For each trial in the one handed case, the task completion time was decomposed into stylus up time (the CV switching time) and stylus down time (CV manipulation time). We then subtracted switching time from the total time to complete the trial. However, analysis of this derived data showed no difference between the one handed and the two handed techniques. We believe that part of the problem here is uncertainty of whether the effect does not exist or that our simple (pen up time) calculation of switching time is too excessive or not detailed enough to measure an effect. This will require more experimentation.

H3 (the advantage of two hands becomes more pronounced when the task becomes more difficult) was found to be true. An interaction was found between input configuration (one hand, two-hands integrated, two-hands non-integrated) and output configuration (target CVs, no target CVs) $F(2, 5) = 4.534$, $p < .05$.

Figure 7 shows the expected 2 handed performance (dotted line labeled "E") relative to the one handed performance for the more difficult (no target CVs) condition. This was generated by considering the performance difference between one-hand and two-handed conditions for the "easy" condition (target CVs visible) and keeping that difference constant for the "hard" (no target CVs visible) condition. That is, we expect the same performance increase in the two-handed conditions for the hard condition. However, the two-handed conditions' performance was significantly faster. Most notably, the magnitude of difference between one-hand and two hands in the easy condition (target CVs) versus the difference in the hard condition (no target CVs) is approximately a factor of three.

H4: (integrated faster than non-integrated). We found no significant differences in task completion time between an integrated and non-integrated device space.

## 4 Discussion

The most telling result was the decreased task completion time for the two handed case as compared to the one handed case. As Kabbash et. al. [20] points out not all two handed tasks give superior performance. If the feedback requires the splitting of visual attention between multiple targets the two handed technique can actually be worse than the one handed. In our experiment the two control points had the potential for splitting attention, however we believe the curve feedback provided a visual integration of the task. We believe this avoided the divided attention pitfall and allowed the two handed technique to demonstrate its advantage.

Although this experiment did not explore the issues of conceptual integration, we believe the task exhibited conceptual integration. Earlier, we defined conceptual integration as when the user perceives and prefers to think of the operations of the two hands not as separate activities but as a single activity. For example, a subject may have thought of the task not as the process of manipulating each CV but as the process of reshaping a curve with two hands. In our post-experiment survey we asked subjects if they ever matched the curve by watching only the curve, not the target CVs, even when the target CVs were displayed. While we had experienced this phenomena ourselves as we had a great deal of practice, none of our subjects reported a similar phenomena. Perhaps this can be attributed to subjects not being extremely experienced with the curve matching task. A longitudinal study may help to isolate learning effects and the issue of conceptual integration requires further investigation.

We did not find any evidence of epistemic action since H2 (two hands still faster, even after subtracting switching times from one-hand condition) was not supported. This could be due to an inaccurate measure of switching time. Essentially we defined switching time as the time the subject spent not dragging a CV (the time spent with the pen up). However, a subject might not only be using this time to switch to the other CV but also to think about the problem, rest or visually acquire the other CV. Thus this time may be an over-estimate of pure time motion switching costs and result in the cognitive effects being masked.

One should also note that switching time is an unavoidable cost in the one handed technique (just as deciding which hand to move is an unavoidable cost in the two handed technique). An attempt could be made to optimize switching time (for example, a key that the subject presses to switch between CVs). However, despite how small the cost, switching time will always be present in the one handed technique.

Our results support the notion that two handed input can be effective for difficult tasks. As the task became more difficult in the experiment the relative advantage of the two handed case increased. Leganchuck et. al. [26] have also found evidence of this. Furthermore, the majority of our subject reported preferring the two handed technique over the one-handed technique.

We found no performance difference between integrated and non-integrated input space with absolute input mappings. This may be due to several reasons: First, because the hand positions were always represented on the screen, hand position was never ambiguous. Thus there was no advantage in sensing the relative hand position in the shared case. Second, the results may also be confounded by collisions or avoidance of collisions between the hands during the task. The benefit of sensing relative hand position may have been canceled out by the cost of hand collisions in the shared case.

Although our experiment showed no significant differences between integrated and non-integrated absolute device spaces, there are practical reasons to choose either approach. If non-integrated device space is used either the device resolution can be split in two, or separate devices used. The first reduces the resolution of the input device and physical size of the input space. The second, consumes twice as much physical ("footprint") space. These problems are not present with an integrated device space scheme. While device collisions are possible, interaction techniques can be designed to minimize collisions.

Our two-handed task was not very complete in the sense that it did not require any selection action by the subject. We have found in experimental prototypes that selection with two devices can be problematic. Two basic schemes can be used: automatic and manual. In the automatic case, the user picks something with one hand and the system automatically assigns another object to the other hand. Alternately, a manual scheme requires the user to explicitly select each object.

Both of these approach have positive and negative issues. Automatic assignment saves the user time in selection (only one item needs to be selected), however the major problem is that the system must determine a proper secondary selection and this is often difficult to predict. The major problem with manual selection is that the user is required to pick with their non-dominant hand and this can be difficult. Furthermore, both of these approaches are susceptible to divided attention. Supporting selection in two-handed tasks is a topic for future research.

## 5 Conclusions

We suggest three design principles for effective two handed input. First, the task should be visually integrated. The task must not promote divided visual attention between the activities of the two hands. Second, the task should be conceptually integrated. That is, the user must be able to conceptualize the operation performed by the two hands as a single operation. Third, the task should employ integrated device spaces.

Our experimental task was designed around these principles and demonstrated significant advantage over a one-handed technique. Thus we have shown that two-handed interactions can be designed which do not suffer from the "tapping the head and rubbing stomach" syndrome.

Our experiment demonstrates the benefits of two-handed interaction over one-handed interaction for increasingly difficult tasks. While we were not able to isolate the specific reason (e.g., switching cost vs. cognitive benefits) for the performance difference for these two conditions, our data shows an increased performance benefit of the two-handed condition when the task become more difficult. However, we found no evidence of improvement if the task employed integrated device spaces. Nevertheless, we believe the two-handed interaction benefit is attributed to the expressive nature of the two-handed technique allowing users to perform epistemic actions to explore the solution space more rapidly compared to the one-handed condition.

## Acknowledgements

We greatly acknowledge Wacom Technology Corp. for assistance in utilizing the multi-device mode on their tablets. Azam Khan provided valuable feedback and assistance in producing the document. Beverly Harrison and Shumin Zhai provided us with help in the experimental design and analysis. Steve Spencely provided comments on requirements on curve editing tasks. We also would like to thank Alias employees who served as subjects in our experiment and supplied valuable comments.

## References

[1] Bartels, R., H., Beatty, J., C., Booth, K., S., Bosch, E., G., Jolicoeur, P., (1993) Experimental Comparison of Splines Using the Shape-Matching Paradigm, *ACM Transactions on Graphics*, 12(3), 179-208.

[2] Ravin Balakrishnan, R., Fitzmaurice, G.W., Kurtenbach, G. and Singh, K. (1999) Exploring interactive curve and surface manipulation using a bend and twist sensitive input strip, *Proceedings of ACM Symposium on Interactive 3D Graphics 1999*, 111-118.

[3] Balakrishnan, R. and Hinckley, K. (2000). Symmetric bimanual interaction. *Proceedings of ACM CHI 2000 Conference on Human Factors in Computing Systems*, 33-40.

[4] Balakrishnanm R. and Kurtenbach, G. (1999) Exploring bimanual camera control and object manipulation in 3D graphics interfaces, *Proceedings of ACM CHI 1999 Conference on Human Factors in Computing Systems*, 56-62.

[5] Baudel, T., (1994) A Mark-Based Interaction Paradigm for Free-Hand Drawing, *Proceedings of ACM Symposium on User Interface Software and Technology (UIST'94)*, 185-192

[6] Bier, E., A.,Stone, M., C., Fishkin, K., Buxton, W., Baudel, T., (1994) A Taxonomy of See-Through Tools. *Proceedings of ACM CHI 1994 Conference on Human Factors in Computing Systems*, 358-364.

[7] Butler, C.G. and Amant, R.S. (2004) HabilisDraw DT: a bimanual tool-based direct manipulation drawing environment. *Extended Abstracts of ACM CHI 2004 Conference on Human Factors and Computing Systems*, 1301-1304.

[8] Buxton, W., & Myers, B. A., (1986) A study in two-handed input. *Proceedings of ACM CHI 1986 Conference on Human Factors in Computing Systems*, 321-326.

[9] Didier Casalta, D., Guiard, Y. and Beaudouin Lafon, M. (1999) Evaluating two-handed input techniques: rectangle editing and navigation, In *Extended Abstracts ACM CHI '99 Conference on Human Factors in Computing Systems*, 236-237.

[10] Cutler, L.D., Frohlich, B., 8z Hanrahan, P (1997), Two-handed direct manipulation on the responsive workbench. *Proceedings of ACM 1997 Symposium on lnteractive 3D Graphics*, 107-114.

[11] Fitzmaurice, G.W. and Buxton, W. (1997) An empirical evaluation of Graspable User Interfaces: towards specialized, space-multiplexed input, *Proceedings of ACM CHI 1997*, 43-50.

[12] Fitzmaurice, G.W., Ishii, H. and Buxton, W.A.S. (1995) Bricks: laying the foundations for graspable user interfaces, *Proceedings of ACM CHI 1995 Conference on Human Factors in Computing Systems*, 442-449.

[13] Foley, J., van Dam, A., Computer Graphics Principles and Practice Second Edition, Addison-Wesley, New York, 1990.

[14] Grossman, T., Balakrishnan, R., Kurtenbach, G., Fitzmaurice, G., Khan, A. and Buxton, B. (2002) Creating principle 3D curves with digital tape drawing, Proceedings of CAM CHI 2002, 121-129.

[15] Guiard, Y. (1987). Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. In *Journal of Motor Behavior*, 19(4), 486-517.

[16] Hinckley, K., Pausch, R., Proffitt, D., and Kassell, N.F. (1998) Two-handed virtual manipulation, *ACM Transactions on Computer-Human Interaction (TOCHI)*, 5 (3), 260-302.

[17] Hinckley, K., Czerwinski, M. and Sinclair, M. (1998) Interaction and modeling techniques for desktop two-handed input, *Proceedings of ACM UIST 1998 User Interface Software and Technology*, 49-58.

[18] Hinckley, K., Pausch, R., Goble, J.C. and Kassell, N.F. (1994) Passive real-world interface props for neurosurgical visualization, *Proceedings of CHI 1994 Conference on Human Factors in Computing Systems*, 452-458.

[19] Jacob, R. J. K., Sibert, L., McFarlane, D. et Mullen, P. (1994) Integrality and Separability of Input Devices, *ACM Transactions on Computer-Human Interaction*, 1(1), 3-26.

[20] Kabbash, P., Buxton, W., and Sellen, A., (1994) Two-Handed Input in a Compound Task. *Proceedings of ACM CHI 1994 Conference on Human Factors in Computing Systems*, 417-423.

[21] Kabbash, P., MacKenzie, I.S. & Buxton, W. (1993). Human performance using computer input devices in the preferred and non-preferred hands. *Proceedings of ACM InterCHI 1993*, 474-481.

[22] Kahneman, D., (1973) Attention and Effort, Englewood Cliffs, NJ, Prentice-Hall

[23] Kantowitz, B., A., (1991) Effects of Response Symmetry upon Bi-Manual Rapid Aiming. *Proceedings of the Human Factors Society 35th Annual Meeting.* 1541-1545

[24] Kirsh, D. and Maglio, P. (1994) On Distinguishing Epistemic from Pragmatic Action. *Cognitive Science*, Vol. 18., 513-549.

[25] Kurtenbach, G., Fitzmaurice, G., Baudel, T. and Buxton, B. (1997) The design of a GUI paradigm based on tablets, two-hands, and transparency, *Proceedings of ACM CHI 1997 Conference on Human Factors in Computing Systems*, 35-42.

[26] Leganchuk, A., Zhai, S.& Buxton, W. (1998). Manual and Cognitive Benefits of Two-Handed Input: An Experimental Study. *ACM Transactions on Human-Computer Interaction*, 5(4), 326-359.

[27] Oliviero, A., and Scarpetta, R., (1981) A new approach to contour coding. In *Computer Graphics Image Processing*, 15(1), 87-92.

[28] Zeleznik, R.C., Forsberg, A.S., and Strauss, P.S. (1997) Two pointer input for 3D interaction, *Proceedings of ACM 1997 Symposium on Interactive 3D Graphics*, 115-120.

## Appendix

Our experiment requires the definition of a distance function between curves that matches visual perception. This is not as straightforward as it may appear, as many commonly used functions, such as computing the area between the two curves, or taking the maximum of the minimum distances between each point of the two curves can exhibit a lot of cases in which two visibly different curves have a very close distance. Bartels [1] use a distance function defined by Oliviero and Scarpetta [27]. This distance is adequate for matching somewhat distant curves, but requires a few trial and errors to precisely adjust its parameters. Instead, we used our own distance function, extracted from [5]. We argue that this distance reflects well the perceptual notion of distance within curves. Finally, it is fast to compute.

Assuming that T, the target curve and M, the matching curve are arc-length parametric function (i.e. the parameter for a given curve) varies in the range [0..L] where L is the length of the curve. Assuming that d is a distance between two points (in our case, d is the euclidean distance squared). Then we define the distance between T and M by the following formula:

$$\mathrm{Min}\left(\int_0^1 d\left(T\left(L_T(t)\right), M\left(L_m(t)\right)\right)\, dt, \int_0^1 d\left(T\left(L_T(t)\right), M\left(L_m(1-t)\right)\right)\, dt\right).$$

The minimum of the two values means that we don't care about the relative directions in which the T and M curves were traced: M and T can be a good match even though M was traced in the reverse direction of T. The integral formula means that we compute the average value of all the distances between individual points on each curve. When applied to a discrete context of curves displayed on the screen, its distance is roughly equivalent to the number of pixels by which the two curves differ over their whole length (i.e. choosing a threshold of 10 means that there are about 10 pixels of difference between the two curves over their whole length).