The Influence of Input Device Characteristics on Spatial Perception in Desktop-Based 3D Applications

Alexander Kulik* Jan Hochstrate† André Kunert‡ Bernd Froehlich§

Virtual Reality Systems Group, Faculty of Media
Bauhaus-Universität Weimar

ABSTRACT

In desktop applications 3D input devices are mostly operated by the non-dominant hand to control 3D viewpoint navigation, while selection and geometry manipulations are handled by the dominant hand using the regular 2D mouse. This asymmetric bi-manual interface is an alternative to commonly used keyboard and mouse input, where the non-dominant hand assists the dominant hand with keystroke input to toggle modes. Our first study compared the keyboard and mouse interface to bi-manual interfaces using the 3D input devices SpaceTraveller and Globefish in a coarse spatial orientation task requiring egocentric and exocentric viewpoint navigation. The different interface configurations performed similarly with respect to task completion times, but the bi-manual techniques resulted in significantly less errors. This result is likely to be due to better workload balancing between the two hands allowing the user to focus on a single task for each hand. Our second study focused on a bi-manual 3D point selection task, which required the selection of small targets and good depth perception. The Globefish interface employing position control for rotations performed significantly better than the SpaceTraveller interface for this task.

Index Terms: H.5.2. [User Interfaces]: Input devices and strategies—Evaluation/methodology;

1 INTRODUCTION

The majority of 3D graphics applications are still desktop-based, which is largely based on ergonomic reasons. The physical support for the operating hand on the desktop surface efficiently reduces fatigue. While there is a broad variety of interfaces for immersive virtual environments, desktop-based 3D applications are mostly managed with the familiar mouse and keyboard set-up. The operation of such 3D applications requires a lot of mode changes if only a 2D mouse and a keyboard are used. In this case, the non-dominant hand assists the dominant hand with keystroke input to toggle modes. This approach has two major drawbacks: The workload distribution between both hands is very unbalanced and integral 3D manipulations need to be separated into a sequence of 2D actions. Jacobs et al. [10] and Hinckley et al. [8] already showed that the second issue affects performance if integral 3D interaction is required.

The SpaceMouse™ is one of the few specialized 3D input devices that has gained respectable acceptance among users. The 3D input device Globefish [5] separates translational and rotational input by its hardware design (figure 1), while the SpaceMouse and its smaller descendant SpaceTraveller™ use integrated elastic six degree of freedom (DOF) rate control. Such 3D input devices are mostly operated by the non-dominant hand to control 3D viewpoint navigation, while selection and geometrical manipulations are handled by the dominant hand using the regular 2D mouse pointer. However, there is no scientific evidence if and why such an asymmetric bi-manual interface configuration is a good choice.

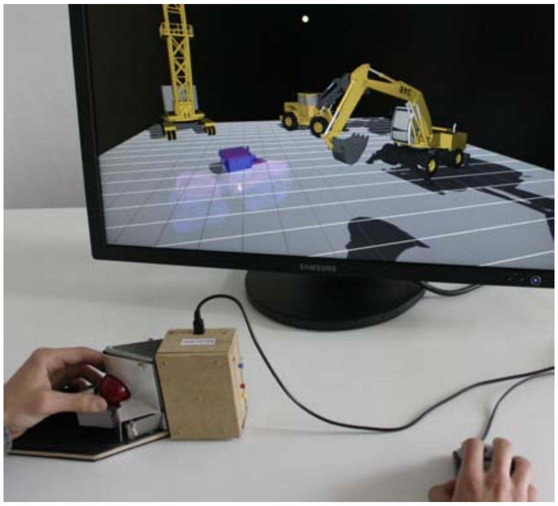

Figure 1: The Globefish input device for CAD and DCC applications

In a first step we analyzed the interaction requirements of desktop 3D applications. Based on our observations we implemented two scenarios and evaluated them by user studies. The first study compared regular keyboard and mouse input to two-handed input using a 3D input device and a mouse. The task focused on coarse spatial orientation in egocentric and exocentric viewpoint navigation. We constrained the required degrees of freedom such that they could be directly provided by the 2D mouse. Our results indicate that the input device configuration does not have much influence on the time efficiency in coarse spatial orientation tasks. However, bi-manual techniques resulted in significantly less errors. We argue that this observation is due to better workload balancing between the two hands allowing the user to focus on a single task for each hand. Our second study compared the two 3D device concepts SpaceTraveller and Globefish in a bi-manual 3D point selection task. The task required high selection accuracy and good depth perception, which had to be achieved through 6-DOF view point navigation. The results of this study revealed significant benefits for the position-controlled rotational input provided by the Globefish device.

2 RELATED WORK

Hinckley et al. [8] demonstrated that using appropriate input devices to control 3D rotations can be very beneficial. Compared to mouse-driven techniques like the Virtual Sphere [4] and Arcball

---

*e-mail: kulik@uni-weimar.de
†e-mail: jan.hochstrate@medien.uni-weimar.de
‡e-mail: andre.kunert@medien.uni-weimar.de
§e-mail: bernd.froehlich@medien.uni-weimar.de

[15], the average task completion times were reduced by up to 36% with orientation-tracked handheld devices in a 3D orientation alignment task. Balakrishnan et al. [2] instead described a number of advantages of the mouse-based input even for 3D interaction. From this analysis and the sustaining demand for more integrally provided degrees of freedom, they derived the concept of the Rocking Mouse. The device is a variation of the mouse that enables simultaneous control of four degrees of freedom instead of only two. A comparison to basic mouse functionality in a 3D positioning task demonstrated important advantages for the device. The results of both studies support the theory of Jacob et al. [10] that simultaneously required degrees of freedom of a task should also be integrally incorporated into appropriate input devices.

Devices like the SpaceMouse provide six integral degrees of freedom for the integral control of 3D rotation and translation. The device employs rate control for both tasks, which allows moving virtual objects around an unlimited workspace through minimal deviations of the input handle. While this is necessary to control 3D translations without lifting the arm from the supporting desk, rotational input could also be provided with position control as in the case of the Globefish device. A comparative user study revealed the superiority of this approach for 3D object manipulation over the fully integrated 6-DOF design of the SpaceMouse, providing only elastic rate control [5]. The separation of rotational and translational input seemed to be beneficial for the chosen 3D docking task which helps to explain the results, but many observations and studies also indicate advantages of position control over rate control.

Shumin Zhai showed that both techniques may perform equivalently, if used with the appropriate devices, but he also found higher training requirements for rate control[19, 18]. In an experiment using a one-dimensional scrolling task, Hinckley et al. [7] observed advantages for position control to operate short range movements and that this range can essentially be extended with software techniques such as pointer acceleration. Kunert et al. [13] validated this interaction of distance and technique in a 6-DOF manipulation task. Since rotations are cyclic, they rarely exceed the range where position control is beneficial. Thus, it is not surprising that Kim et al. [12] observed position-controlled object rotations with an isotonic 3D trackball to be more efficient than operating the same task with the rate-controlled SpaceMouse input.

In practice such desktop-based 3D devices are operated by the non-dominant hand to control reference objects or navigation, while the more frequent pointer interaction is assigned to the dominant hand. The efficiency of such a workload distribution in similar contexts has often been demonstrated [6, 11, 3, 14, 9]. Previous studies on specialized 3D interaction devices in desktop environments only analyzed performance of operations of the dominant hand. We aim to close this gap, by analyzing input device performance with the non-dominant hand in compound bimanual tasks as they are typically found in common 3D applications like games, CAD and DCC.

3 INTERACTION REQUIREMENTS OF DESKTOP-BASED 3D APPLICATIONS

3D games as well as 3D modeling in engineering and art are highly successful 3D graphics applications. Most often they are operated by devices known from 2D applications, with the mouse as the primary input tool. On closer inspection of the requirements of those applications we found that this adaptation of an existing infrastructure is not inappropriate. The major interaction task performed is the selection of trigger buttons, tools and objects. Using image plane selection techniques, even selection in 3D space can essentially be simplified to controlling only two degrees of freedom, for which a regular 2D mouse is perfectly suited. Since the task is the most frequent one and requires high precision, it is naturally assigned to the dominant hand [6].

To extend the applicability of the device, the non-dominant hand assists with keystroke input to toggle modes and tool selection shortcuts. If navigation and manipulation are required however, interaction becomes more difficult. All adjustments of fine-grained parameters are necessarily assigned to the mouse device. Many efficient techniques have been developed to map 3D interaction through 2D input [4, 15, 16], but this approach also increases the workload of the dominant hand and thus its physiological fatigue. To cope with this problem one can assign more sophisticated input devices to the non-dominant hand. This can be another pointing device [17] or a specialized 3D navigation and manipulation device like the SpaceMouse. The latter is more common and appreciated by many professional users. In our experiments we analyze the impact of controller design for such two-handed interaction with respect to the specific requirements of egocentric navigation within virtual environments and exocentric object examination.

4 DESKTOP-BASED 3D INPUT DEVICES

The Spacemouse is a desktop-based input device that enables users to control motion direction and velocity of 3D rotations and translations with small deviations of an elastically suspended handle. Users often comment on its comfortable operation but also about difficulties to work accurately with the device. Linear motion can hardly be induced without rotational input and vice versa. Furthermore, rate controlled input is not as simply reversible like position controlled input, which was pointed out by Balakrishnan et al. [2]. The device is offered in a number of different designs that vary in the size of the handle as well as the number of additional input buttons. In the described experiments we used the SpaceTraveller with a handle of 45 mm in diameter.

Figure 2: The Spacetraveller 3D input device

The Globefish is a novel sensor concept for desktop based 3D interaction. For our experiments we used prototypical implementations of the device. It consists of a 3DOF trackball which is elastically suspended within a surrounding frame. The trackball, suitable for being precisely held from two opposing sides, can be slightly moved in all spatial directions to induce translational input. Rotational input is generated by simply rotating the sphere, while translations are induced against an elastic counterforce similar to the Spacemouse device. For the experiments we used two different prototypes of the device. Both consist of a Trackball with 40mm in diameter and provide elastic counterforces similar to those of the SpaceTraveller. The design of both emphasizes rotational input around the vertical and the lateral axis as well as translational input in depth direction, since those are the most frequently required degrees of freedom. Both offer ergonomics that afford a comfortable hand posture similar to writing with a pen. The device prototypes differ in resolution and positioning of the rotation sensors and also they differ slightly regarding their ergonomics. While the more recent prototype used for the first study on coarse spatial orientation exposes the trackball towards the operating hand (figure 4), the trackball of the antecedent prototype is laid open at its top (figure 3). Though we believe that the recent design is slightly better, we also tested with the previous prototype due to technical issues. Regarding the results of the experiments we argue that the differences between both are negligible.

Figure 3: The Globefish device prototype I

Figure 4: The Globefish device prototype II

5 USER STUDY ON A SPATIAL ORIENTATION TASK

3D graphics applications allow for viewpoint motion to handle occlusion problems and to make larger workspaces accessible. Typically, it is not possible to position the viewpoint as such that all relevant objects in the scene are visible. Thus, egocentric viewpoint rotation as well as exocentric rotations around an object of interest are frequently required. The difference between both is the rotation pivot. The viewpoint itself defines the center of rotation within the egocentric case, enabling the user to look around in the virtual environment. In the exocentric case the pivot refers to an external object in order to encircle it.

Accordingly, our study included alternating subtasks of exocentric and egocentric viewpoint motion.

5.1 Task

We have chosen a very simple environment for our task. It consisted of a cubic room and a much smaller cube inside that room. The faces of both were displayed in six different colors. The small cube was aligned with the cubic environment and placed at its center (figure 5). For visual orientation, we assigned black to the top, a light gray to the bottom and saturated colors to the remaining four faces of the room and the cube.

The task started in exocentric mode with the viewpoint inside the cubic room, but still distant to its center where the small cube was previously located. The viewpoint was automatically oriented towards the center and could only be moved in two dimensions (head and pitch) encircling the smaller cube. The exocentric navigation technique is similar to rotating an object in front of the view (object manipulation) except two important differences: 1. Not a specific object is manipulated, but the whole environment remains consistent while the viewpoint is rotated around an external pivot. 2. The environment in front of the view therefore appears to rotate in the opposite direction, as the viewpoint is moved by the user's input. In practice, both methods can be found to support object examination. We decided to conduct the test with exocentric viewpoint navigation instead of an object manipulation technique to maintain consistency between the egocentric and the exocentric orientation subtask.

Our daily experience with spatial orientation in the real world is largely influenced by gravity effects. Humans are therefore most familiar with wayfinding in environments that are more or less horizontally aligned. To prevent loops that would easily lead to disorientation, we limited the arc motion around the lateral axis to $+/-60^\circ$. The participants were instructed to unveil numbers behind each of the cube's six faces and memorize the highest one. Mouse pointer input was controlled with the dominant hand to enable the selection of the target faces, which triggered randomized numbers between 10 and 99 to appear on the selected surface patch. The exocentric subtask could then be finished through double clicking on the face with the highest number displayed. Thereafter, an animated transition moved the viewpoint to the center of the scene, where the small cube was previously located.

The second phase of the task, starting after the automated viewpoint transition to the center of the scene required the user to control egocentric viewpoint rotations. Only rotations around the vertical (head) and the lateral (pitch) axis were enabled. To avoid disorientation, rotations around the lateral axis were again limited to $+/-60^\circ$. Here, the users had to search for smaller squares located at the edges of the four surrounding vertical walls. In contrast to the exocentric subtask, we excluded targets at the top and the bottom face of the surrounding cube, since finding these would have caused more difficulties. To involve also rotations around the lateral axis, the target squares were situated at different heights. Two opposing walls showed the squares at their lower edge and the other two at their upper edge. These squares were sized as such they appeared in a comparable size on the image plane as the surface patches of the target cube in the exocentric subtask. To ensure good visual contrast, target squares had the same color as the opposing wall segment of the surrounding room (figure 6). Again, participants were asked to unveil randomized numbers (ranging from 10 to 99) behind these targets through mouse pointer selection. In the egocentric condition four targets had to be unveiled before the users could decide on the highest displayed number among them and then select it with the mouse pointer. After finishing the task by double clicking with the mouse pointer on the respective highest number, an animated transition moved the viewpoint again to the starting position for the next exocentric subtask.

For both subtasks, a carefully designed distribution algorithm assured that the required rotation angles (0°, 90°, 180°, 270°) to move from the last unveiled target square to the one with the highest number were randomized and that they added up to the same amount of required motion for each block of trials.

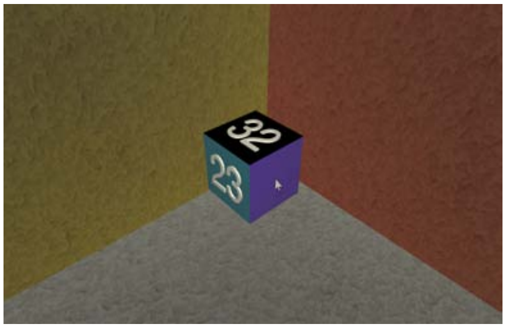

Figure 5: During the exocentric subtask, the viewpoint had to be turned around a cube in the center of the scene to explore the object from each side and select its six faces in order to unveil a randomized number. Thereafter, the respective highest number had to be confirmed with a double click operation.

Figure 6: During the egocentric subtask, the viewpoint was located at the center of the cubic environment. Quadratic target patches could be found on the surrounding four walls that had to be selected in order to unveil a randomized number. As in the exocentric subtask, the respective highest number had to be confirmed thereafter.

5.2 Apparatus

The study was conducted on a desktop workplace with the user seated in front of a table providing a support for the input devices and the graphics display (20” wide-angle LCD with a resolution of 1680x1050 px). The test application was running at 60 Hz. In all conditions, the user’s dominant hand manipulated the virtual pointer with a regular mouse device. A linear transfer function with a control-display gain of 10 was assigned to the mouse pointer. The large target faces allowed for such an amplifying transfer function. Three input device configurations were used for viewpoint control.

In two device conditions the non-dominant hand controlled a 3D input device: either the SpaceTraveller or the Globefish device. Since the navigation subtask also involved only the control of 2DOF, we included a basic mouse/keyboard condition in the tests. In this condition, pointer manipulation as well as viewpoint control were assigned to the mouse (using the same linear transfer function) and operated by the dominant hand. This approach obviously required mode changes. The participants had to hold the CTRL-button on the keyboard to trigger viewpoint navigation mode. The graphical pointer remained visible and also button clicks were enabled in this mode. Thus, it was possible to accomplish the whole task while remaining in viewpoint navigation mode.

The elastic SpaceTraveller was used with rate-control and a non-linear transfer function for high precision at low velocities, while still enabling rapid movements with larger deviation of the device’s handle. Both isotonic devices, the 2D mouse and the 3D trackball of the Globefish, were used with position control and linear transfer functions. For the 3D trackball, we implemented an isomorphous mapping since we assumed that users may benefit from a congruent relation between the amount of input motion and the resulting rotation on the screen.

5.3 Hypothesis

In a comparative study on the impact of rate control and position control on document scrolling performance, Hinckley et al. [7] found advantages for position control. They further demonstrated that such scrolling tasks can be modeled with Fitts’ Law. Andersen argued, that this finding is only true, if the target position is known beforehand. In contrast, he found a linear relationship between the time required for scrolling a document and the distance to be covered, if the target distance is unknown [1]. He assumed that in this case, rate control might be better suited, since it facilitates motion with a constant velocity.

In our case, only short moves had to be accomplished and since the environment was very simple, we assumed that users would be quite conscious about the required amount of rotation to find the next target for selection. Position control provides proprioceptive cues on the amount of motion input induced. The counterforce of elastic devices for rate control on the other hand provides haptic information on the motion velocity, which is more indirect regarding the goal to reach a certain position. Thus we hypothesized that position control with both isotonic input devices (Globefish and mouse) would result in better performance compared to elastic rate control with the SpaceTraveller, because users can benefit from proprioception. We expected the mouse to show the best performance, since it is well suited for this specific task and most users are really proficient with the device.

We also assumed that additionally employing a sophisticated input device by the non-dominant hand would increase time efficiency and/or accuracy due to the distribution of workload.

5.4 Design and Procedure

After a short training session to accommodate to the task environment, three blocks of 12 trials were recorded with one device condition. As described above, each trial included an exocentric and an egocentric subtask. Short breaks between the subsequent blocks helped to minimize fatigue. Thereafter, the same procedure was applied to the other two device conditions. The order of devices was fully permuted between six independent groups. After completing the test with the three device configurations, the participants were asked to rate the three tested input devices on a five point Likert scale. Overall, the experiment lasted about one hour.

5.5 Participants

Twenty-four volunteers, aged between 19 and 34 years, took part in the first study. All of them were students from various disciplines including humanities, engineering and fine arts. Eleven among them were male and thirteen were female. All except one were right handed. We adapted the input device mappings to the left handed person. Half of the participants reported that they were familiar with 3D applications, ten reported to have had only marginal experience and two of them were complete beginners. Fifteen had no experience with using devices like the SpaceMouse, the other nine already had used such 3D input devices.

5.6 Results and Discussion

Having the device conditions fully permuted between six independent groups, we first tested on possible differences between groups, but could not find any significant effect. Thus, all data was collapsed and entered into a 3 (devices) x 2 (subtasks) x 3 (blocks) analysis of variance, using Bonferroni adjustment for α in post-hoc comparisons.

Regarding task completion times, we found significant effects for the factors *subtask* ($F_{1,23} = 187.518$, $p < .001$) and *block* ($F_{2,46} = 211.344$, $p < .001$). Since both subtasks were quite different regarding their operation, it was expected, that the respective task completion times would also differ significantly (8.38 s in the exocentric and 11.17 s in the egocentric condition). The exocentric subtask was much easier, since the visual focus was fixed on the target object that defined the center of rotation. Thus, it could not be visually lost. Continuous learning can be observed over the three blocks. Post-hoc comparisons showed that the improvements between subsequent blocks are all significant (all $p < .001$). Average task completion times decreased from (11.30 s) over (9.24 s) to finally (8.78 s).

Device produced no main effect. This indicates, that all three tested device conditions are comparable in terms of time efficiency. In detail though we found the *device* condition to interact significantly with *block* ($F_{4,92} = 3.61$, $p < .01$) and even stronger with *subtask* ($F_{2,46} = 27.81$, $p < .001$).

The interaction between device and block (figure 7) seems to result from a different learning behavior for the mouse. Both input conditions involving motion input from the non-dominant hand expose smaller and more consistent learning effects than the mouse condition. For the mouse condition we observed the strongest performance gain from the first to the second block. In average over both *subtask* conditions the mouse showed worst performance during the first block (11.44 s), but showed the best performance in the following two (8.76 s and 8.19 s). This indicates that with the mouse participants required learning only for the task, since they were already highly proficient with the device. Post hoc comparisons on the interaction of device and block using the Tukey-Test revealed one significant effect, namely an advantage of the mouse over the Globefish during the last block.

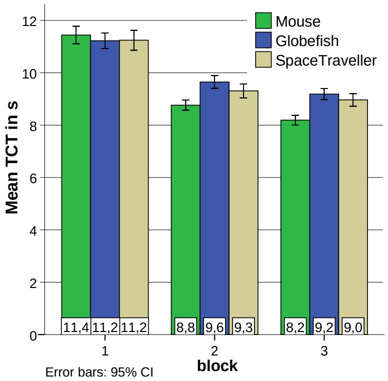

Figure 7: Mean task completion in seconds, sorted by *device* and *block*.

A Tukey test on the interaction of device and subtask (figure 8) revealed a significant advantage of the mouse to both other devices in the egocentric viewpoint subtask (both $p < .01$), but no further significant effects. This subtask dependency can also be observed in subjective ratings (from 1=best to 5=worst). While users preferred the mouse for egocentric rotations (Mouse: 2.20, SpaceTraveller: 2.41, Globefish: 2.54), they favored the 3D input device conditions for the exocentric subtask (SpaceTraveller: 1.54, Globefish: 2.08, mouse: 2.20). Though we did not find a significant interaction between device, block and subtask, we observed stronger training effects for the mouse in the exocentric condition. Some users also reported of having been confused from using the device with an exocentric navigation technique. The mouse is commonly used to move the on-screen pointer, which is more closely corresponding to the egocentric rotation subtask.

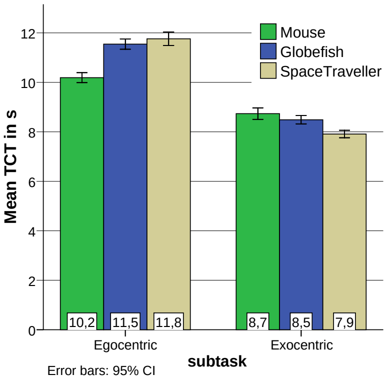

Figure 8: Mean task completion times in seconds, sorted by *device* and *subtask*.

We also recorded click errors per trial. All kinds of useless clicks were counted as such (repeated selection, empty space selection, selection of a wrong target face). The number of click errors was divided by the number of necessary selections to compute the relative click error rate. An analysis of variance (using Bonferroni adjustment for α) revealed significant main effects for *device* ($F_{2,36} = 19.37$, $p < .001$), *block* ($F_{2,36} = 14.73$, $p < .001$) and *subtask* ($F_{1,18} = 93.71$, $p < .001$) as well as interaction effects of *device* with *block* ($F_{4,72} = 4.68$, $p < .05$) and *device* with *subtask* ($F_{2,36} = 5.75$, $p < .05$). The egocentric subtask resulted in a significant higher errors rate (11.92%) compared to the exocentric (4.57%) subtask. Accurate pointing was easier in the latter condition since the selection target remained in the center of the screen and thus only little movements of the mouse pointer were required.

Block effects revealed a significant improvement from the first (9.67%) to the following blocks (7.85% and 7.21%). This is an expected learning effect.

More interesting are the differences of devices. While we did not find many differences between the three tested *device* conditions regarding time efficiency, both two-handed techniques provided higher accuracy. Using only the mouse device to the dominant hand with mode changes resulted in 10.5% click error rate. Employing an additional 3D input device to the non-dominant hand resulted in a significant lower click error rate: 7.59% for the SpaceTraveller and 6.64% for the Globefish (both $p < .01$).

A Tukey-test on the interaction of *device* with *subtask* revealed that only in the egocentric condition the mouse showed significant lower accuracy than both other *device* conditions (figure 9). Examining the interaction of *device* with *block* we found particularly strong learning effects for the mouse condition. The accuracy drawback of the mouse condition could efficiently be compensated through learning the task (figure 10). A Tukey-test showed no further significant differences between devices in the last block of trials. In the second block only the difference between the Globefish and the mouse was found to be significant ($p < .05$).

Our hypothesis on the superiority of position- over rate control could not be proved regarding the task completion times of the study. We suggest that the expected advantages could not be found, because the navigation task did not require much accuracy. Instead of targeted moves, participants rather scanned the environment continuously while trying to move fast. Following Anderson [1] this strategy is well supported by rate control. In fact, we observed that with the SpaceTraveller users tended to select moving targets with the mouse pointer while turning the viewpoint continuously. With position control instead motion input is necessarily interrupted by clutching operations to compensate the limitations of the physical input space. The Globefish with an isomorphous transfer function

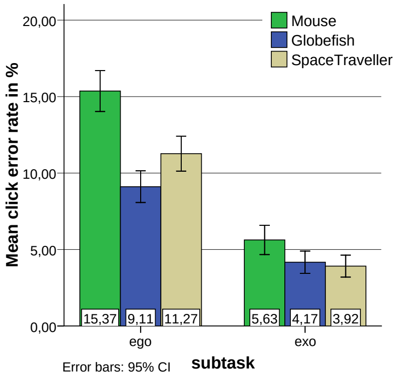

Figure 9: Mean click error rate sorted by *device* and *subtask*.

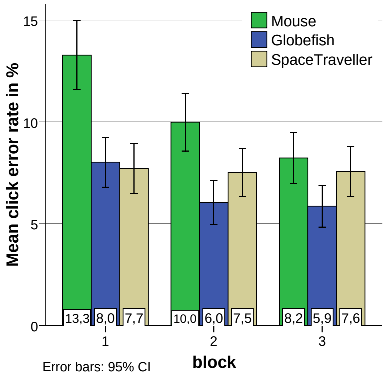

Figure 10: Mean click error rate sorted by *device* and *block*.

required even more clutching compared to the mouse. We assume that an accelerated transfer function could have improved its performance in our experimental task.

Estimated benefits of the mouse device could only be found in time efficiency. In terms of accuracy instead, we found advantages for two-handed interaction. Employing an additional input device for the non-dominant hand seems to enhance accuracy. The position-controlled Globefish however, showed larger advantages in that regard. We observed that with position control users worked more sequentially. Parallel input from both hands could rarely be recorded in the Globefish condition, but users altered between navigation input from the left hand and pointer input from the right hand. With the elastic rate controller instead, users tended to control viewpoint navigation and object selection in parallel. The results of our study indicate that sequential operation might decrease time efficiency, but improves accuracy.

6 USER STUDY ON DEPTH PERCEPTION THROUGH MOTION PARALLAX

To gather spatial orientation by looking around in a virtual scenery is an important task of many 3D applications but definitely not the only one relevant. Through interviewing and observing professionals working with CAD and DCC tools, we identified another important aspect of spatial navigation techniques: depth perception.

Most workplace set-ups for such applications do not provide a stereoscopic display. Thus, to obtain depth information users need to rely on perspective as well as on motion parallax resulting from movements of the viewpoint and the manipulated objects. When working with unknown geometries and especially with organic shapes, perspective is not a very reliable cue. Motion parallax instead is very robust, but permanently requires viewpoint or object motion. Since usually no head tracking is provided, this can be achieved with input devices like those described in chapter 4. Particularly challenging with respect to depth perception is the interaction with point clouds defining the shape of complex 3D objects. Since the respective applications display control points with a fixed size on the image plane, the user cannot rely on differences in size to distinguish their depth in 3D space.

6.1 Task

We implemented an evaluation scenario based on spatial control point selection. A number of yellow colored points were distributed on all six faces of a translucent cube, situated in front of the viewpoint. The cube could be translated and rotated in three dimensions without constraints using a 3D input device operated by the non-dominant hand. To complete the task, only small amounts of rotation were required, but we provided full 6-DOF interaction functionality to allow for different user strategies. The distance of the cube from the viewpoint had an important influence on the distance of the selection points in screen space. Keeping the object far away from the viewpoint resulted in short motion amplitudes for the mouse pointer between the points, but respectively harder differentiability between them. Moving the cube closer had the opposite effect. Reproducing the design of common software packages the square shaped points were visible and selectable from any point of view. They always maintained a fixed size of 2x2mm on the screen independent of their respective distance.

An arrow-shaped pointer, controlled by the mouse device in the dominant hand was used for point selection. Selected points switched color from yellow to red. One trial consisted of selecting four points defined by a surrounding square frame on one of the cube's six faces (figure 11). The task was completed by having the four points correctly selected. Incorrectly selected points had to be deselected to complete the task. Hardly any perspective ever showed only the relevant selection points within the borders of the turquoise-colored frame. To recognize, which of the surrounded points really belonged to the relevant group, users needed to slightly turn the cube in most cases. Thus depth information to recognize the points located on the same surface as the framing rectangle could be obtained through motion parallax.

Each trial started with the appearance of the target frame, stretching over a quarter of one of the cube's faces. The distribution of selection points and the position of the target frame were randomized. Successful selection of the four points finished a trial and initiated the next one, starting with the appearance of a new target frame and the repositioning of all selection points.

6.2 Apparatus

The experiments were conducted on a desktop set-up with the user seated in front of a 22 CRT display (resolution: 1920x1200 px), which showed monoscopic visual stimuli. The test application was running at a frame rate of 96Hz.

As in the study before, the pointing device assigned to the dominant hand might also be used for 3D manipulations if incorporating mode changes. We decided to compare only 6-DOF devices in that study, that were operated by the non-dominant hand. We did not include the mouse condition in the experiment since the actual application context in CAD and DCC involves full 3D rotations and translations, which are not as well supported by the mouse device [10, 8]. The specific task of our study could also be accomplished with only little rotational input, but testing the devices with full 6-DOF functionality allowed us to observe whether spuriously induced translations might be a problem for that kind of task.

For technical reasons we were required to use and earlier prototype of the Globefish device in that study. The employed mouse and SpaceTraveller devices as well as the respective transfer functions

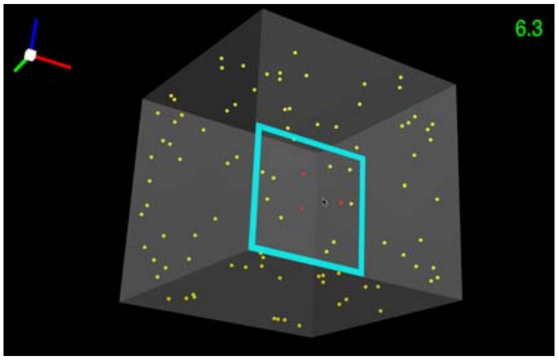

Figure 11: A translucent cube covered with control points was presented to the participants of the study on depth perception. The four points framed by the turquoise colored frame had to be selected with the mouse pointer. Only through motion parallax the respective depth of these points could be recognized. Participants thus had to rotate the cube at least slightly using a 3D motion controller to the non-dominant hand.

were the same as in the previous study, except that we decreased the control display gain for mouse input to seven for higher accuracy.

All elastic input with the SpaceTraveller device and translational input with the Globefish device were mapped to rate-controlled translations of the cube using a non-linear transfer function as in the study before. In contrast to the first study, position-controlled rotation inputs were now mapped to the virtual object using an accelerated transfer function. Previous experiences with the device indicated that an isomorphous mapping is less beneficial. We thus used a power function as described in [5] to enable precision as well as rapidity.

6.3 Hypothesis

We assumed that position control is the preferable choice to achieve spatial perception of geometric shapes through motion parallax, even if controlled with manual input instead of head tracking. The required relative motion between object and viewpoint is minimal but the task requires high accuracy. Employing the Globefish to rotate objects of interest with position control rather than rate control as with the elastic SpaceTraveller should therefore result in higher time efficiency as well as higher interaction accuracy in the chosen task on spatial point selection.

6.4 Design and Procedure

To gain insights into expert performance on such tasks, we conducted three sessions on three consecutive days with each user and device, thus trying to ensure sufficient training on the task as well as on the tested devices. In each session, three blocks had to be completed with each device. The order of devices was balanced between two user groups.

Since the task was rather difficult for many participants, every test session included a training block for each device. Short breaks interrupted the blocks consisting of 16 trials. Overall, one session lasted about half an hour each day. After the third session a questionnaire was handed to the participants, asking them to report experiences during the tests and assess the tested devices on a five point Likert scale.

6.5 Participants

Twenty-two volunteers, aged between 20 and 33 years, participated in the study. All were students from different disciplines. Sixteen were male and six were female. All were right-handed. Sixteen participants reported to have experience with certain variations of the SpaceMouse device. Eighteen reported to be familiar with 3D computer games, seven reported experience with VR-systems and four among them were used to work with CAD or DCC tools. Four participants had no experience with 3D applications.

6.6 Results and Discussion

Data was entered into a 2 (device) x 3 (session) x 3 (block) analysis of variance (using Bonferroni adjustment for α in post-hoc tests) with order of devices as between-subjects factor. The order of the devices produced neither a main nor an interaction effect.

Regarding task completion times, the Globefish (4.75 s) significantly outperformed ($F_{1,20} = 15.66$, $p < .001$) the SpaceTraveller (5.04 s). From our observations we conclude, that the advantages of the Globefish stem from different interaction strategies with both devices. While rate control with the Spacetraveller encourages concurrent two-handed input, position control fosters sequential input which seems to enhance accuracy.

The Globefish also features a stronger distinction between translational and rotational input than the Spacetraveller. Accordingly we recorded twice as much simultaneous 6-DOF input in the Spacetraveller condition than in the Globefish condition, but this did not result in observable disadvantages. In either case only small displacements were applied to the cube and user's did not seem to have issues with keeping the cube inside the field of view.

A main effect on time efficiency was also found for *session* ($F_{2,40} = 8.47$, $p < .001$). Significant learning occurred between the first and the following two sessions ($p < .05$), but post-hoc comparisons did not show a significant difference between the last two sessions (figure 12). Learning effects became significant only between sessions (days) due to the training block performed before each session with both devices. No main or interaction effect on *block* could be obtained.

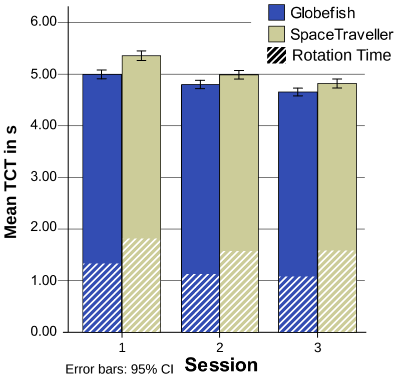

Figure 12: Mean task completion times in seconds, sorted by device. The hatched partitions of the bars at the bottom illustrate the fraction of time, when the cube was rotated. The larger partitions correspond to the sum of idle times and sole user input for object translations and point selection.

Additionally, we analyzed selection errors and click errors per trial. The marginal differences between device conditions on selection errors (number of incorrect point selections) indicated that users were always able to discriminate the spatial location of points to select. Regarding click errors that were counted when clicking outside a target, we only found one main effect on *device* ($F_{1,20} = 10.13$, $p < .01$). Rate-controlled rotations with the SpaceTraveller resulted in a significantly higher click error rate (24%) than using position control with the Globefish (20.75%).

As in the study before we observed that users tended to aim at selection points, while still turning the cube with rate-controlled input. This strategy of continuous object rotation, practically providing continuous depth cues through motion, resulted in the more complicated selection of moving targets. Sequential operation, as observed in the usage of the Globefish seems to be a better strategy.

Recorded operation sequences support this observation. We found that users spent about 41% more time on object rotations in case of the SpaceTraveller condition (figure 12 - Rotation Time). A much larger amount of time was dedicated to point selection than for turing the object in order to achieve depth perception. Note that both tested conditions differed mainly in the employed methods for object rotation. The huge performance difference with respect to that operation thus underlines the superiority of the Globefish device for that kind of task. Subjective rankings (from 1=best to 5=worst) confirmed the preference of the Globefish (1.45) over the SpaceTraveller (1.73) for the spatial point selection task.

7 CONCLUSIONS AND FUTURE WORK

We conducted two experiments on spatial navigation in desktop-based 3D applications. The results of the spatial overview task revealed two major results. Pure mouse-based interfaces provide comparable performance as using additional 3D navigation devices if only the most relevant degrees of freedom for viewpoint orientation need to be controlled. On the other hand we found significant benefits in accuracy through better balanced bi-manual input when using 3D input devices, which users of 3D modeling tools should take into consideration.

The experiment on depth perception indicated advantages in spatial perception for position-controlled over rate-controlled rotation input. The Globefish device performed significantly better than the SpaceTraveller with respect to time efficiency and accuracy. Although our first study suggests that for coarse spatial navigation rate control is well suited, we observed considerable advantages for the position-controlling Globefish in cases where accurate 3D object manipulations or exocentric viewpoint navigation were required.

The characteristics of input devices will remain a relevant research direction for the 3D UI community, since it is the sensors and displays that constitute the tangible reality of virtual 3D environments. For the further development of 3D interfaces we need to thoroughly analyze which factors contribute to the usability of a tool such that the user can concentrate on the task rather than on operating the input device and interface. The relationship of the mechanical characteristics of an input device (shape, size, operation methods, etc.) to its usage in real-time graphics applications is a major aspect of that research. We suggest that the perception of the operation of physical devices provides users with important cues that can help to distinguish interaction states of highly adaptable virtual environments. For example, switching from position control to rate control should be always accompanied by swapping the mechanical device characteristics from isotonic to elastic. We argue that future interaction devices should more closely relate to their specific use and allow interactive adaptations of their tangible characteristics.

ACKNOWLEDGEMENTS

We thank the participants of our studies as well as Susanne Günther and Sascha Gärtner for their tremendous support in implementing the evaluation tasks.

REFERENCES

[1] T. H. Andersen. A simple movement time model for scrolling. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1180–1183, New York, NY, USA, 2005. ACM.

[2] R. Balakrishnan, T. Baudel, G. Kurtenbach, and G. Fitzmaurice. The rockin'mouse: integral 3d manipulation on a plane. In *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 311–318, New York, NY, USA, 1997. ACM.

[3] R. Balakrishnan and K. Hinckley. The role of kinesthetic reference frames in two-handed input performance. In *UIST '99: Proceedings of the 12th annual ACM symposium on User interface software and technology*, pages 171–178, New York, NY, USA, 1999. ACM.

[4] M. Chen, S. J. Mountford, and A. Sellen. A study in interactive 3-d rotation using 2-d control devices. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 121–129, New York, NY, USA, 1988. ACM.

[5] B. Froehlich, J. Hochstrate, V. Skuk, and A. Huckauf. The globefish and the globemouse: two new six degree of freedom input devices for graphics applications. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 191–199, New York, NY, USA, 2006. ACM.

[6] Y. Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, 19:486–517, 1987.

[7] K. Hinckley, E. Cutrell, S. Bathiche, and T. Muss. Quantitative analysis of scrolling techniques. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 65–72, New York, NY, USA, 2002. ACM.

[8] K. Hinckley, J. Tullio, R. Pausch, D. Proffitt, and N. Kassell. Usability analysis of 3d rotation techniques. In *UIST '97: Proceedings of the 10th annual ACM symposium on User interface software and technology*, pages 1–10, New York, NY, USA, 1997. ACM.

[9] A. Huckauf, A. Speed, A. Kunert, J. Hochstrate, and B. Froehlich. Evaluation of 12-dof input devices for navigation and manipulation in virtual environments. In *INTERACT*, pages 601–614, 2005.

[10] R. J. K. Jacob, L. E. Sibert, D. C. McFarlane, and J. M. Preston Mullen. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.*, 1(1):3–26, 1994.

[11] P. Kabbash, W. Buxton, and A. Sellen. Two-handed input in a compound task. In *CHI '94: Conference companion on Human factors in computing systems*, page 230, New York, NY, USA, 1994. ACM.

[12] M.-S. Kim, J.-K. Seong, D.-E. Hyun, K.-H. Lee, and Y.-J. Choi. A physical 3d trackball. In *PG '01: Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*, page 134, Washington, DC, USA, 2001. IEEE Computer Society.

[13] A. Kunert, A. Kulik, A. Huckauf, and B. Froehlich. A comparison of tracking- and controller-based input for complex bimanual interaction in virtual environments. In B. Froehlich, R. Blach, and R. van Liere, editors, *EG IPT-EGVE 2007*, page 4352, 2007.

[14] A. Leganchuk, S. Zhai, and W. Buxton. Manual and cognitive benefits of two-handed input: an experimental study. *ACM Trans. Comput.-Hum. Interact.*, 5(4):326–359, 1998.

[15] K. Shoemake. Arcball: a user interface for specifying three-dimensional orientation using a mouse. In *Proceedings of the conference on Graphics interface '92*, pages 151–156, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[16] G. Smith, T. Salzman, and W. Stuerzlinger. 3d scene manipulation with 2d devices and constraints. In *GRIN'01: No description on Graphics interface 2001*, pages 135–142, Toronto, Ont., Canada, Canada, 2001. Canadian Information Processing Society.

[17] R. C. Zeleznik, A. S. Forsberg, and P. S. Strauss. Two pointer input for 3d interaction. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 115–ff., New York, NY, USA, 1997. ACM.

[18] S. Zhai. *Human Performance in Six Degree of Freedom Input Control, Ph.D. Thesis*. Univ. of Toronto, 1995.

[19] S. Zhai. User performance in relation to 3d input device design. *Proceedings of the SIGCHI conference on Human factors in computing systems*, 32(4):50–54, 1998.