Low Cost 3D Rotational Input Devices: the stationary *Spinball* and the mobile *Soap3D*

Marcus Tönnis, Florian Echtler, Manuel Huber, Gudrun Klinker

Fachgebiet Augmented Reality
Technische Universität München, Fakultät für Informatik

{toennis, echtler, huberma, klinker}@in.tum.de

Abstract

This paper introduces a new approach enabling intuitive input of rotational data with low-cost optical sensors. Two devices are built, one for desktop applications and a mobile wireless device for everywhere use. The desktop variant, called Spinball makes the virtual trackball a real device. The mobile device, called Soap3D uses a two-layered approach for device casing, enabling closed hand control. Benefits and construction are illustrated and results of expert reviews are discussed.

1. Introduction

For input of 3D rotations several systems exist. An example for a mapping from mouse input to rotations is the widely used ARCBALL [6] method. Here, the user has to click and drag a point on an imaginary sphere around the object, which then rotates to follow the described circle segment. This principle suffers from the fact that a mouse offers only two degrees of freedom, whereas a rotation has three degrees of freedom. The second option is to use a dedicated rotation input device, e.g. the 3Dconnexion Spacemouse [1]. Such devices are also not fully intuitive, as they only produce rotation velocities. Specialized and intuitive input devices are, e.g., the ones presented by Takahashi et al. [7] and Kim et al. [5]. A similar approach is also presented with the GlobeFish [4] by Fröhlich et al.

The two devices presented in this paper are based on low cost optical sensors and enable. First, the *Spinball*, which follows the principles of Takahashi, Kim, Fröhlich et al., enables intuitive unconstrained 3D rotation in desktop, powerwall or table-top environments. Second, the *Soap3D* exhibits the same capabilities but focuses on use in mobile applications. Both devices incorporate technology form optical mice and thus provide a low cost alternative to many other systems.

2. The *Spinball*

Our first device is called is the *Spinball* (see Figure 1), which is inspired by the concept of the trackball.

2.1. Construction

The *Spinball* enables the user to control the rotation of a virtual object by turning a sphere inside the device in the same way as the object is supposed to rotate.

The core elements of the *Spinball* are two optical computer mice. They are placed beneath the sphere which comprises the actual interaction element. On the basis of requirements for good haptic experience and sufficient texture for tracking, we found beechwood to be a suitable and readily available choice for the sphere.

With respect to the placement of the mice, we found that it is sufficient to simply orient the two mice in parallel with a slight tilt towards the center so that both sensors touch the sphere. While Kim et al. [5] stated the optimal placement to be at an angle of 90 between the two mice, our experiments have shown that even ad-hoc placement of the sensors is completely sufficient

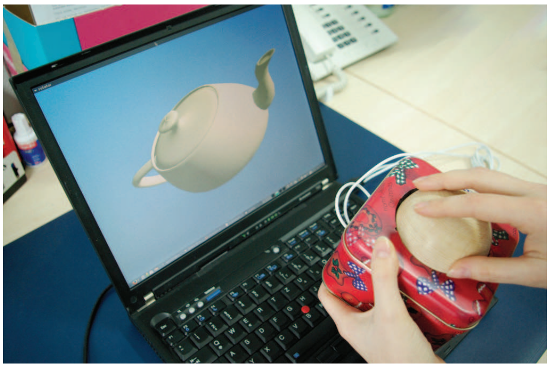

Figure 1. User interface of TimeMenu node

to provide an intuitive user experience, as long as the sensors are not exactly opposite or very close to each other.

In our prototype, we mounted the two mice into a small box with a circular hole in the lid, through which the sphere can be inserted and guided. The entire cost of materials summarizes to €13.20.

As shown in Figure 2, every mouse sensor produces two measurement values of the relative movement of the sphere surface. These measurements are delivered in terms of an integral number of mouse units, or ”ticks”. From these vectors $\Delta 1 = (dx1, dy1)$ and $\Delta 2 = (dx2, dy2)$, a relative rotation in terms of three angles can be derived.

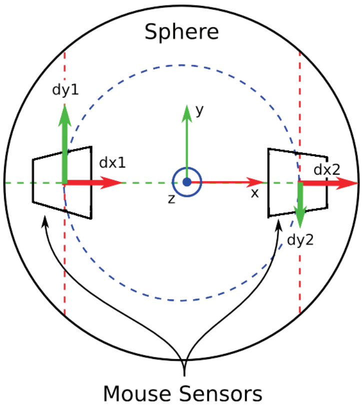

Figure 2. Top view of raw data vectors and coordinate systems. The two rectangles are the optical sensors which touch the sphere tangentially

First, it is necessary to determine the conversion factor from mouse ”ticks” to radians. To this end, the path lengths for rotations around each axis have to be known. The paths which are completed when the sphere performs a full rotation are drawn dashed in Figure 2 in the color of the corresponding axis. As these paths are additionally dependent on sensor placement, we found that an accurate conversion is possible by counting the ticks for one full manual rotation of the sphere. This step is done once for each axis, resulting in three tick counts n_x, n_y, n_z.

Although still inaccurate due to unavoidable jitter, the three resulting factors

$$f_x = \frac{2\pi}{n_x}, f_y = \frac{2\pi}{n_y}, f_z = \frac{2\pi}{n_z}$$

provide a sufficiently precise mapping from real to virtual rotation. This calibration step only needs to be performed once for a given sphere diameter and sensor placement and does therefore not concern the end user. After the conversion factors are known, the three angles can now be calculated. Rotation around the y axis can simply be assumed as

$$r_y = f_y \frac{dx_1 + dx_2}{2},$$

averaging the two delta-x values.

Rotation around the x and z axis is slightly more complex, as this data is mixed within the two delta-y measurements. We can assume these two measurements to be composed from a common component *a* from rotation around the x axis and an inverse component *b* from rotation around the z axis in such a way that $dx_1 = a + b$ and $dy_2 = a - b$. This leads to the following results:

$$r_x = f_x \frac{dy_1 + dy_2}{2} \quad \text{and} \quad r_z = f_z \frac{dy_1 - dy_2}{2}$$

Now that all three relative angles are known, we can use them to update the object rotation. While the most straightforward way would be to simply collect three absolute rotation angles and apply them to the object, this leads to known problems such as gimbal lock. Therefore, we first convert the relative rotation to a quaternion (see, e.g., [3]) and premultiply this result with a second quaternion storing the absolute rotation. This quaternion can then easily be converted into a matrix representation which, in turn, can be loaded into a 3D graphics toolkit such as OpenGL or DirectX.

2.2. Results

We have tested the *Spinball* with a demo application (see also Figure 1) which displays the well-known teapot model. We let several users evaluate the *Spinball* by simply telling them to flip the teapot upside down using the sphere. No further instruction was necessary, and rotation in other directions was intuitively understood.

While no quantitative evaluation was conducted yet, subjective impressions suggest that the quality of the calibration as described in the previous section is not highly relevant. Even when one of the conversion factors had an estimated error of about 25 % due to imprecise movement during calibration, this was not immediately noticeable to the user. One explanation for this might be that the largest rotation which can be achieved without releasing the sphere is slightly less than 180. It seems to be difficult to perceive discrepancies between the haptic feedback from rotating the sphere and the visual feedback from the virtual object within this range.

3. The Soap3D

For mobile environments where the user is standing or walking, we propose the *Soap3D* as an appropriate input device. The device is inspired by the work of Baudisch et al. [2] who developed an input device for conventional 2D desktop applications. They put a wireless optical mouse into a soap-like case surrounded by a hull of flexible fabric. Turning the soap inside the fabric delivers the same positional information to the computer as if a mouse would be moved on a flat surface. The wireless communication enables the users to use the mouse in mid-air. Handling tests showed that interaction with this device is easy to learn and that fast and high accuracies in positioning can be reached with even short training.

Our *Soap3D* device combines the approach by Baudisch et al and the concept of the *Spinball* for 3D rotation by incorporating two optical mouse sensors that track the flow of the fabric. The *Soap3D* is referenceless and does not suffer from jitter through larger rotations where the the user usually has to perform clutching. Fig. 3 illustrates the handling in two pictures.

(a) Initial Grip to the Soap3D (b) Grip after a turn

Figure 3. Usage of the Soap3D. The user can rotate the device in the hand without clutching

3.1. Construction

We basically inverted the setup of the *Spinball*. Instead putting the two optical mouse sensors in the outer case, we equipped the hard case interior with the two optical sensors. The prototype was constructed in two iterations.

First Prototype

The first version consists of a soap like hard plastic case of dimensions 8.0 x 5.6 x 2.4 cm. This size fits reasonably well into a hand, but was conjectured that a smaller case is easier to interact with. Also the case was smooth enough to glide well in the fabric hull. Like in the *Spinball* setup we used two off-the-shelf optical mice, albeit wireless ones. These mice were placed in non-colinear positions inside the case. To fit these into this small form factor, the removal of any non-essential parts of the mouse PCBs was necessary as well as modification to the power distribution. To reach a non-colinear mounting of the second mouse sensor, we detached the optical sensor from the circuit board and connected it with flexible wires. Another interesting problem encountered was that due to the close proximity of the mice radio interference was essentially muting one mouse completely. This was only circumvented by choosing two different mice operating in two sufficiently distant parts of the radio spectrum (27MHZ and 2.4GHZ). The entire cost of the materials summarized to €57.55.

Building an improved Version

The goal of the second design was to completely overcome the radio interference problem by using a single radio link for both mice. At the same time the size of the device should be reduced. The heart of the device consists of two Avago ADNS- 5020 integrated optical flow sensors. These are mounted with sufficient illumination to the side and the bottom of the case. The data from both optical sensors is collected by an Atmega644 MCU and sent via a Bluetooth link to the host computer. The radio-communication is handled by a Parani ESD-200 Serial-to-Bluetooth adapter module due to its small footprint and ease of integration.

Using this custom made hardware we were able to reduce the size of the PCB to 3.4 x 3.3 cm and in consequence, the size of the case to

5.6 x 5.2 x 2.4 cm . This resulted in an casing with an almost quadratic footprint. The resulting closed-up device can be seen in Fig. 4 and the assembly contained in Fig. 5.

Figure 4. The case of the Soap3D V2 now almost 3 cm shorter

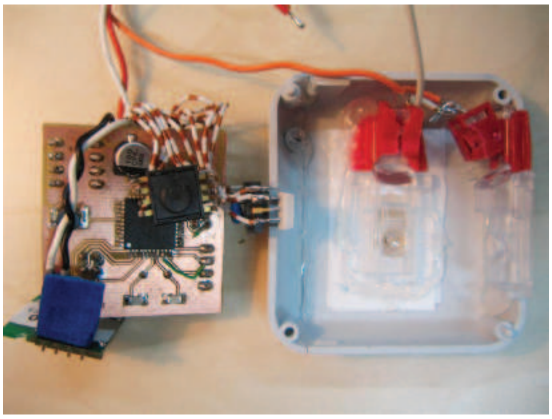

Figure 5. The interior of the Soap3D V2. Left: two optical sensors (on brown-white wires), the custom circuit board and the Parani wireless link (blue). Right: half of the case containing the optics and the LEDs

The entire cost of the materials summarized to €69.75. The Serial-to-Bluetooth adapter is the main cost factor, but is expected to be replaced by components for about €6.00, effectively halving the overall cost to €33.15.

3.2 Interaction

We gave both devices to some reviewers for examination. The reviewers used three interaction techniques. Using the *closed hand interaction*, people kept the hull of fabric stationary in their hand and turned the inside case. With the *clutching interaction*, users dragged the fabric on one side of the device in one direction and the fabric at the opposite side of the device in the opposite direction. The inside case almost remained in the same absolute position, while the fabric moved. The *two-handed interaction* puts both flat hands on opposite sides of the device and moves the hands in opposite directions. The two first techniques are described by Baudisch et al [2] as soap interaction and the belt interaction. Baudisch et al described a third technique, using the thumb directly on top of the sensor for finegranular positioning. Naturally this interaction does, due to the existence of two sensors, not operate properly with the *Soap3D*.

Using the Larger First Prototype

With the closed hand interaction, the reviewers started with the easiest rotation, the one around the long axis of the *Soap3D*. Even if the device in its construction is relatively large, such rotations were quite easy to perform. To perform rotations around the second longest axis, the users had to change the grip to the device – otherwise it is too long to reach around with the hand and the fingers. The rotation around the shortest axis also is not easy to handle. The device is too large to apply turning forces with the fingers.

With the clutching interaction the same order as with the closed hand interaction was determined but with less difference between the different axes. All rotations around all axes could be performed, but still the one around the shortest axis was not too easy to perform.

With the two-handed interaction almost all rotations were equally easy to perform. Only having the hands on the smallest sides for a rotation around the shortest axis required some stabilization of the case to avoid flipping sidewards.

A combination of two first interaction techniques allows reasonable rotation of the device in all directions.

Using the Smaller Second Prototype

Closed hand rotations around the two long axes were almost similar to handle. Rotations around the short axis were easier to handle. Clutching interaction also improved for the shortest axis. The fabric generated a lower resistance force easing the rotation. The smaller size of the device also eased handling between the fingers. The two-handed interaction interestingly was almost unchanged. The smaller size of the device had no influence on handling rotations around the two long axes. Rotating around the short axis still required stabilization of the device inside the hands.

4. Conclusion

In this paper, we have presented the *Spinball*, a stationary tangible rotation input device and the *Soap3D*, a mobile rotation input device. Consisting mainly of two optical mice, both devices are an inexpensive means for rotational input. All users which reviewed the devices immediately understood their functionality and had no problems to control a displayed 3D object.

While the concept for the *Spinball* is not at all new, the device built with optical mouse sensors provides an intuitive low cost alternative for environments where the case can be placed. While we did not yet conduct a formal evaluation, we believe the *Spinball* to be superior to most mouse-based rotation input methods. We will verify these claims in a rigidly controlled study.

The *Soap3D* is an extension to Baudisch's 2D Soap but is a proof that optical mouse sensors can be used in an inside-out approach to generate rotational input. The review showed that the two versions are already usable, but are still too large. After further decreasing the size, we will extend the device with position tracking and will test the device in a user study. For this new device we assume that *closed hand interaction increases smoothness of rotation input with concurrent decrease of positional jitter*.

5. References

[1] 3DConnexion Spacemouse. http://www.3dconnexion.com/.

[2] P. Baudisch, M. Sinclair, and A. Wilson. Soap: a mouselike pointing device that works in mid-air. In *User Interface Software and Technology Symposium*. ACM Press New York, NY, USA, 2006.

[3] N. Bobick. Rotating Objects Using Quaternions. In *Game Developer Journal*, volume 2, pages 21–31, 1998.

[4] B. Fröhlich, J. Hochstrate, V. Skuk, and A. Huckauf. The GlobeFish and the GlobeMouse: Two New Six Degree of Freedom Input Devices for Graphics Applications. In *Proc. of CHI*, pages 191–199, 2006.

[5] M.-S. Kim, J.-K. Seong, D.-E. Hyun, K.-H. Lee, and Y.-J. Choi. A Physical 3D Trackball. In *9th Pacific Conference on Computer Graphics and Applications*, page 134,Washington, DC, USA, 2001. IEEE Computer Society.

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

[7] T. Takahashi and M. Kuzuya. 3-D Input Device Using a Ball Rotation Interface. In *HCI International: Ergonomics and User Interfaces -Volume I*, pages 397–401, Hillsdale, NJ, USA, 1999. L. Erlbaum Associates Inc.