Six Degree-of-Freedom Sphere Tracking for 3D Interaction

Derek Bradley
School of Computer Science
Carleton University, Canada
derek@derekbradley.ca

Gerhard Roth
Institute for Information Technology
National Research Council of Canada
Gerhard.Roth@nrc-cnrc.gc.ca

Abstract

Interaction with 3D objects using standard computer input devices such as a mouse and keyboard is often a difficult task. For this reason, *Tangible User Interfaces* (TUIs) are developed to allow more natural 3D interaction by manipulating physical objects in a familiar way. We present a new TUI system that includes a passive optical tracking method to compute the six degree-of-freedom pose of a sphere in a real-time video stream, and then apply the pose to a virtual object. The pose is accurately resolved under partial occlusions, allowing manipulation by hand without a tracking failure.

1 Introduction

Natural interaction methods that consist of manipulating real physical objects to control virtual entities are often called *Tangible User Interfaces* (TUIs). We present a new TUI system that operates by resolving the full six degrees-of-freedom (DOF) pose of a sphere from real-time video input, and then maps the pose to a 3D virtual object. We use a blue ball with red and green surface dots as the sphere to be tracked (see Figure 1). This approach is similar to that of Guenter et al. [3], who use colored dots to capture facial animation.

Figure 1: Sphere to be tracked for interaction.

Other vision-based TUIs have been developed by Fjeld and Voegtli [1], and Huang et al. [4] where a physical cube is visually tracked for 3D interaction. However, the cube tracking relies on locating planar markers on the sides of the cube which do not allow occlusion by a user's hand. As well, the planar tracking accuracy depends on the angle formed between the plane normal and the camera, so tracking can be inconsistent as the cube is manipulated. Our sphere tracking system provides a more consistent TUI that can handle partial occlusions, allowing the sphere to be manipulated by hand. Vision-based sphere tracking is not a new idea, however most of the previous methods resolve only three DOF, namely the position of the sphere and not the orientation. The most similar method to ours is that of Greenspan and Fraser [2], where a sphere dipole is tracked to provide a five DOF passive input device.

2 Sphere Tracking

Our sphere tracking method consists of a number of computer vision techniques and applications of 3D geometry. We break up the tracking into two steps: computing the sphere location and computing the sphere orientation.

2.1 Location

Locating the sphere in an input image consists of finding its circular projection and then computing the 3D location with respect to the camera. The blue color of the ball is used to find its projection. First a binary image is created by segmenting based on the hue that represents the blue color. Then contours in the binary image are computed and small contours are filtered out. Since the projection of a sphere is always a circle, the minimum enclosing circle of each contour is computed and the contour with the best ratio of pixel area to minimum enclosing circle area is chosen as the sphere projection. Figure 2 illustrates these steps to find the projection of the sphere.

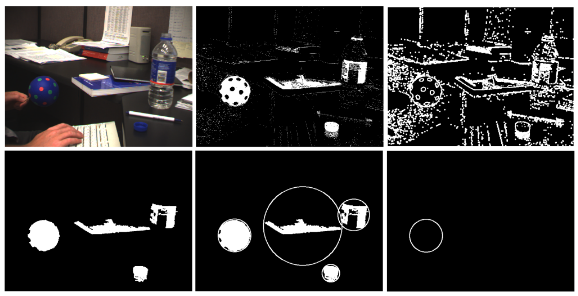

Figure 2: Locating sphere projection. From top left: Input; Binary image; All contours; Filtered contours (filled in); Minimum enclosing circles; Most circular contour.

When the sphere projection is found, the 3D location of the sphere (X, Y, Z) is computed using the intrinsic parameters of the camera. If the focal length of the camera is f, the principal point is (p_x, p_y), and the center of the sphere projection is (u, v), then

$$X = \frac{Z \cdot (u - p_x)}{f}, \quad Y = \frac{Z \cdot (v - p_y)}{f}.$$

So we have X and Y expressed in terms of Z. Now, we use the assumption of weak perspective projection [2] to compute Z as follows

$$Z = \frac{R \cdot f}{r},$$

where R is the radius of the sphere and r is the radius of the circular projection.

2.2 Orientation

The red and green dots on the surface of the sphere are used to compute its 3D orientation. First the projections of the dots are located on the image plane, similar to locating the blue projection of the sphere. In this case the projections are ellipses and we can restrict the search space to within the projection of the sphere that we have already computed (as shown in Figure 3). Then we compute the polar coordinates of the located dots assuming that the center of the sphere projection is the north pole. The two dots closest to the center of projection are selected, and the angle between them (through the center of the sphere) is calculated from the polar coordinates. This angle is then used to search through a pre-computed list of all pairs of dots to determine the set of all pairs that the selected dots could match. For each potential match, a virtual copy of the sphere is oriented to align the two chosen dots and a score is computed to indicate how well the other located dots align. The orientation with the highest score is chosen as the real sphere orientation.

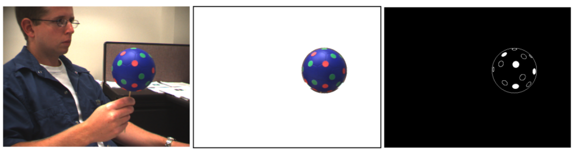

Figure 3: Locating the projection of surface dots. From left: Input; Restricted Input; Ellipses found for red (filled) and green (unfilled) dots.

3 Results and Applications

Our TUI system operates at an average of 15 frames per second on a Pentium 4 processor at 3.4Ghz using 640x480 color images. One of the main benefits of our system is the ability to handle partial occlusions. We choose the minimum enclosing circle for the sphere projection and also we do not require all of the surface dots to be visible. This allows a user to pick up and control the sphere by hand, even if their fingers occlude part of it. Figure 4 demonstrates the results of our system including occlusions, by augmenting a teapot using the pose of the sphere.

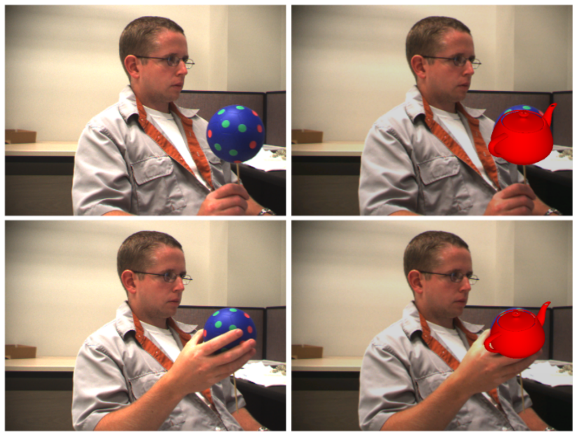

Figure 4: Regular tracking results (top) and occlusion handling results (bottom).

Applications of our TUI system include augmented reality, games, CAD systems, and virtually any interactive graphical application.

Acknowledgements

This work was supported by the Natural Sciences and Engineering Research Council of Canada.

References

[1] Morten Fjeld and Benedikt M. Voegtli. Augmented chemistry: An interactive educational workbench. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2002)*, pages 259–260, September 2002. Darmstadt, Germany.

[2] Michael Greenspan and Ian Fraser. Tracking a sphere dipole. In *16th International Conference on Vision Interface*, June 2003. Halifax, Canada.

[3] Brian Guenter, Cindy Grimm, Daniel Wood, Henrique Malvar, and Fredric Pighin. Making faces. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 55–66. ACM Press, 1998.

[4] Chun-Rong Huang, Chu-Song Chen, and Pau-Choo Chung. Tangible photorealistic virtual museum. *IEEE Computer Graphics and Applications*, 25(1):15–17, 2005.