Collaborative Hybrid Virtual Environment

Leonardo Pavanatto Soares* — School of Informatics, PUCRS
Thomas Volpato de Oliveira† — School of Informatics, PUCRS
Vicenzo Abichequer Sangalli‡ — School of Informatics, PUCRS
Márcio Sarroglia Pinho§ — School of Informatics, PUCRS
Regis Kopper¶ — School of Engineering, Duke University

Abstract

Supposing that, in a system operated by two users in different positions, it is easier for one of them to perform some operations, we developed a 3D User Interface (3DUI) that allows two users to interact together with an object, using the three modification operations (scale, rotate and translate) to reach a goal. The operations can be performed using two augmented reality cubes, which can obtain up to 6 degrees of freedom, and every user can select any operation by using a button on the keyboard to cycle through them. To the cubes are assigned two different points of view: an exocentric view, where the user will stand at a given distance from the object, with a point of view similar to the one of a human being; and an egocentric view, where the user will stand much closer to the object, having the point of view from the object's perspective. These points of view are locked to each user, which means that one user cannot use both views, just the one assigned to his ID. The cameras have a small margin of movement, allowing just a tilt to the sides, according to the Oculus's movements. With these features, this 3DUI aims to test which point of view is better for each operation, and how the degrees of freedom should be separated between the users.

1 Introduction

The Collaborative Hybrid Virtual Environment is a system that allows two users to perform manipulation operations (translation, rotation or scaling) to the same object together, by separating the degrees of freedom that each user can control. It was developed as a submission to the 3DUI Contest 2016 where the proposed problem was to develop a simply learned collaborative manipulation 3D User Interface (3DUI), to allow multiple users to perform those operations, in order to solve different tasks that required modifications in 6 or more degrees of freedom.

In the early stages of development, we focused on understanding how we could benefit from the existence of multiple users. We already had developed an environment where two users could apply 6 degrees of freedom to opposite sides of an object and it would generate an interpolation, like what happens in the physical world. We discussed it, and figured out that this approach would not be the best for the proposed tasks, since we are manipulating small objects and need to perform different operations.

However, separating the degrees of freedom seemed to be more related to the tasks. We found out that giving different views for each user could make a big difference on how the tasks are solved. If all users had the same view, it would not be much different than one user completing the task alone. But different views would add the possibility that depending on which view the user has, he could be able to perform the operations more efficiently, and could perform more complex operations.

2 Related Work

Pinho [3] discussed the simultaneous manipulation of a virtual object by multiple users, including methods of combining simultaneous user actions and the separation of degrees of freedom between both users. Duval [2] proposed a different approach, where the object would be manipulated as if each user was handling the extremity of a skewer. As stated before, we decided to follow the approach of separating degrees of freedom instead of users grabbing the object on opposite sides. Aguerreche [1] compares and evaluates both approaches and a third one, using a Collaborative Tangible Device (CTD) with handles, which is manipulated by the users. It was concluded that the users felt the CTD as the most realistic of the three methods. In the presented paper, we will combine the idea of separation of degrees of freedom and the use of a physical object, a cube, as an input device similiar to the one being displayed to the user, trying to make the interaction with the system feel more realistic.

3 User Interface Architecture

We built the system with two users, each one with a different concept of visualization: one uses an egocentric view, while the other uses an exocentric view.

In the egocentric view, the user stands with the focus on the object. While wearing a Head Mounted Display (HMD), the user can tilt his head to have a different point of view and, therefore, perform localized manipulations better. In the demonstration video, the user with this view performed the translation operation.

The exocentric view shows the object far from the user. In this way, the user has an overview of the scene and object, from a human-size point of view. In the video, the user with this view performed the scale and rotate operations.

Even though the system allows two points of view, each one is attached to a user and cannot be changed. The freedom of movement of the user is also limited: one cannot perform broad movements, but can tilt the camera to the sides, using the movement sensors of the Oculus as indicators for its intensity.

All operations are available for both users, they just need to press a key on the keyboard to change it. However, both users cannot apply the same operation at the same time. Note that different operations can still be applied together by different users.

3.1 Hardware

The developed 3DUI uses one input device for each user: an augmented reality cube which, combined with a webcam, will give us 6 degrees of freedom, as shown in Figure 1. This cube, also shown in Figure 1, has a handle to indicate the orientation of the device, since the user will utilize the HMD and will not be able to see it physically while performing the operations.

*e-mail: leonardo.pavanatto@acad.pucrs.br
†e-mail: thomas.volpato@acad.pucrs.br
‡e-mail: vicenzo.sangalli@acad.pucrs.br
§e-mail: marcio.pinho@pucrs.br
¶e-mail: regis.kopper@duke.edu

Figure 1: HMDs, webcams and AR cubes.

3.2 Software

The software has two main components, which were developed using the Unity Engine and the ARToolKit SDK. The component developed using the ARToolKit is responsible for obtaining data from the cube and sending it to the main program (i.e. translation and rotation).

The Unity software is the main program, which is responsible for the physics simulation, rendering and interpolation of the operations performed by both users. It was developed using the Photon Server API, a networking asset that simplifies the connection and synchronization between Unity programs. This API also allows the user to remotely connect to this system using a feature called Photon Cloud. There will be a higher latency, but depending on the situation, the amount of delay is totally acceptable.

By using these features, the 3DUI allows users to perform two of the three operations (i.e., translation, rotation and scaling) on the same object at the same time. These operations are interpolated in a server, tested for the existence of collisions, and updated on each client.

4 Applying the Operations

Our system uses two augmented reality cubes, both with 6 degrees of freedom, to perform the three operations separately. We use the AR cube rotation as a trigger to the object operations - when you rotate the AR cube, the virtual cube rotates (or scales) in the same axis. The more the AR cube rotates, the faster the virtual object spins or scales. In Figure 3 there is an example of three possible movements using the pitch axis: (a) no physical rotation; (b) an ω rotation that makes the object spin at an ω' velocity; (c) a θ rotation, being ($\theta > \omega$), which makes the object spin at a θ' velocity, where ($\theta' > \omega$').

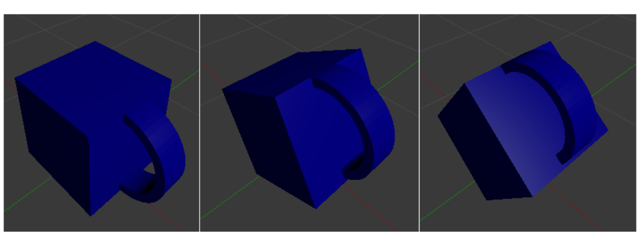

Figure 2: AR Cube performing a rotation operation using the pitch axis.

The translate operation works using a trigger too. There is an imaginary cube, and when the AR cube is inside this imaginary cube, the virtual object is stationary. The virtual object starts to move when the AR cube leaves the imaginary cube. The more the AR cube moves out of the imaginary cube, the faster the virtual object translates in the virtual environment. In Figure 4 there are examples of three possible movements in the roll axis: (a) the AR cube is inside the imaginary cube; (b) α% of the AR cube is outside the imaginary cube, so the virtual object moves back with an α' velocity; (c) β%, being ($\beta > \alpha$), of the AR cube is outside of the imaginary cube, therefore, the virtual object moves back with a β' velocity, where ($\beta' > \alpha$') .

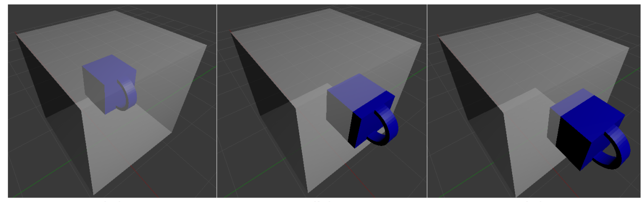

Figure 3: AR Cube performing a translation operation using the roll axis.

The users utilize the advantages that each point of view has to offer to solve the problem, since one of them is really near the cube and the other one is away from it. They will choose any operation by cycling through them using a button on a keyboard and collaborating with each other to perform the tasks demanded by the program.

5 Conclusion and Future Work

In this paper we presented a 3DUI where a single virtual object can be manipulated simultaneously by two users performing different operations. We used two agumented reality cubes as our main input devices, with the perspective of improving the realism of the experience to the user, while also trying to make it easier to use. We perform these operations using different points of view (egocentric and exocentric).

In the future, we plan to investigate the best combination of points of view (egocentric and exocentric) and operations (translate, rotate, scale), to determine how the degrees of freedom should be separated for a better performance. We will also research possibilities to enhance the freedom of movement to the user.

Acknowledgements

This work has been partially funded by the Brazilian Federal Agency CAPES through the Brazilian Scientific Mobility Program (BSMP).

References

[1] L. Aguerreche, T. Duval, and A. Lécuyer. Comparison of Three Interactive Techniques for Collaborative Manipulation of Objects in Virtual Reality. In *CGI 2010 (Computer Graphics International)*, Singapour, Singapore, June 2010.

[2] T. Duval, A. Lecuyer, and S. Thomas. Skewer: A 3d interaction technique for 2-user collaborative manipulation of objects in virtual environments. In *Proceedings of the 3D User Interfaces*, 3DUI '06, pages 69–72, Washington, DC, USA, 2006. IEEE Computer Society.

[3] M. S. Pinho, D. A. Bowman, and C. M. Freitas. Cooperative object manipulation in immersive virtual environments: Framework and techniques. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '02, pages 171–178, New York, NY, USA, 2002. ACM.