Usability Tests for Improvement of 3D Navigation in Multiscale Environments

Tathiane Mendonça Andrade, Daniel Ribeiro Trindade, Eduardo Ribeiro Silva, Alberto Barbosa Raposo, and Simone Diniz Junqueira Barbosa

Department of Informatics, Pontifical Catholic University of Rio de Janeiro
Caixa Postal 38.097, 22.453-900, Rio de Janeiro, RJ, Brazil
tandrade@inf.puc-rio.br, danielrt@tecgraf.puc-rio.br,
eribeiro@tecgraf.puc-rio.br,
{abraposo,simone}@inf.puc-rio.br

Abstract. The interest in virtual 3D environments has increased in the past years due to the popularization of the technology and the huge human ability to visually convey and grasp information. However, unlike the real world, 3D navigation, especially in multiscale environments, is no longer natural to humans, becoming confusing and resulting in unpleasant experiences. To improve the quality of the users' navigation in and across multiscale 3D environments, three techniques were developed, based on a structure called cubemap. The 3D application chosen to apply these techniques was the Petrobras 3D System for Integrated Visualization in Exploration and Production (SiVIEP). This paper describes and reports the evaluation of these three techniques, using usability tests, which were performed to validate the more adaptable solution, ensuring the efficiency of the proposed techniques in assisting and facilitating the task of 3D navigation.

Keywords: 3D Navigation, Multiscale Environments, Usability, Cubemap.

1 Introduction

The popularization of 3D systems has brought new means of manipulating three-dimensional objects available to a wide audience of users, from basic to expert, depending on the system. However, 3D systems may require more effort from users to understand the models presented on the screen [1]. Therefore, we need to carefully design the interaction between the user and the 3D system to have a high quality of use.

Besides users' lack of experience at navigating in 3D environments, we find poorly designed navigation tools and also the problem known as desert fog [2], condition where the surrounding environment does not provide sufficient information to enable users to make a decision.

This paper briefly describes and reports the evaluation of three techniques developed to improve the quality of the users' navigation in and across multiscale 3D environments. A *multiscale environment* allows to users to view objects at different scales, together with information at different levels of detail, from a single screw to an oil field spanning dozens of miles [3, 4]. The techniques are based on a structure

M. Kurosu (Ed.): Human Centered Design, HCII 2011, LNCS 6776, pp. 481–490, 2011.

called *Cubemap* [5], which has the purpose of providing information about the virtual environment at any given moment, like creating a representation of the environment in relation to the observer's position. Based on cubemap, the three techniques proposed in [6] and investigated in our usability tests are: *Fly with automatic speed adjustment*, *Collision detection and treatment*, and *Examine with automatic pivot point*.

The evaluation technique chosen was the usability test. The 3D application chosen to apply these techniques was the Petrobras 3D System for Integrated Visualization in Exploration and Production (SiVIEP), developed by the Computer Graphics Technology Group (TecGraf) / PUC-Rio. SiVIEP displays the objects of a petroleum field, integrating all of the related tasks, and allows changes in the field during the 3D visualization.

The goal of this study is to assess the efficiency of the navigation techniques, based on the cubemap structure, through usability tests with different user profiles and for different tasks.

This paper is organized as follows: In the next section, the cubemap concept is presented and the Navigation techniques are described. Section 3 presents SiVIEP and its features. Section 4 describes the creation, preparation and execution of the tests with the users. Finally, conclusions are discussed in Section 5.

2 Cubemap

According to McCrae et al. [5], cubemap aims to provide information about the virtual environment at any given moment. The method consists of rendering six images in six different directions from the viewer's position, each one corresponding to a side of the cube. Cubemap uses a field of view of 90º in camera perspective, allowing the combination of the resulting *frustums* to cover the entire environment located between the clipping planes.

Whenever the camera changes position or rotation, the cubemap is updated. The distances from the generated fragments to the viewer are calculated. The computed distance values are normalized in relation to the near and far planes, and stored in the alpha channel of the positions related to the fragments. This way, the cubemap provides a depth representation of the environment at every given moment without the need for any kind of preprocessing, which is a desirable feature in the case of dynamic scenes.

The cubemap update process requires six additional rendering steps, which adds considerable cost and can result in loss of performance. To prevent this from occurring we use a lower resolution rendering of the faces, since only an estimate needs to be obtained. A resolution of 64×64 was enough to reach the level of precision the users needed.

3 Navigation Techniques

This section presents the three navigation cubemap-based techniques developed to solve the aforementioned 3D navigation problems.

3.1 Fly with Automatic Speed Adjustment

In multiscale environments, the user is free to view the information from a simple screw up to oil fields with hundreds of miles. In this case, the navigation speed is directly related to the scale of the environment. When you navigate across an oil reservoir, for example, the speed should be faster than the one used to analyze the interior of a platform.

In certain environments, the scale does not change too much and is well known, allowing pre-computation of the values for the navigation's speed. However, these values can not always be estimated, either by incomplete models and units errors or by large scale variations.

To address this issue, the proposed method uses the distance values stored in cubemap to determine the scale and to adjust the speed of navigation. This allows us to estimate the appropriate speed and make smooth transitions between different scales.

3.2 Collision Detection and Treatment

Not allowing collisions can be a determining factor in virtual environments. In immersive environments, a collision with an object can cause the loss of immersion, causing discomfort to the user, especially in interaction that uses stereoscopy effects.

In other cases the lack of treatment of collision can cause the effect known as *desert fog* [2], especially for non-advanced users. For example, crossing walls in a closed environment or cutting through the ocean floor during navigation may cause the user to lose the necessary information to make further navigation decisions.

The information stored in cubemap determines the closest objects in the path taken by the observer. The distances are used to calculate a repulsive force that smoothly deflects the camera from obstacles.

3.3 Examine with Automatic Pivot Point

To inspect an object, two actions are required: 1) determine the object as the pivot point and 2) use the *examine* tool. This latter action allows the camera to rotate and zoom around the object of interest.

The location of the center of rotation is essential for the proper functioning of the *examine* tool. Among the problems with this interaction technique, there is the choice of a pivot point which is away from the user's field of view (Fig. 1a). Another factor is the center of rotation located in front or behind the object to be examined (Fig. 1b). In either case the result is not the one the user expected.

We observe that some users, even advanced ones, make mistakes when using these tools. The most common mistake is to forget to establish a new center of rotation. Even users who have not forgotten to accomplish this task reported that they felt uncomfortable about having to explicitly do it.

The proposed solution was to automatically determine the pivot point to be examined. For this, the point located in the center of the user's vision, information stored in the cubemap, is used as pivot point wherever the *examine* tool is activated. It only requires the user to point the center of his vision at the object of interest.

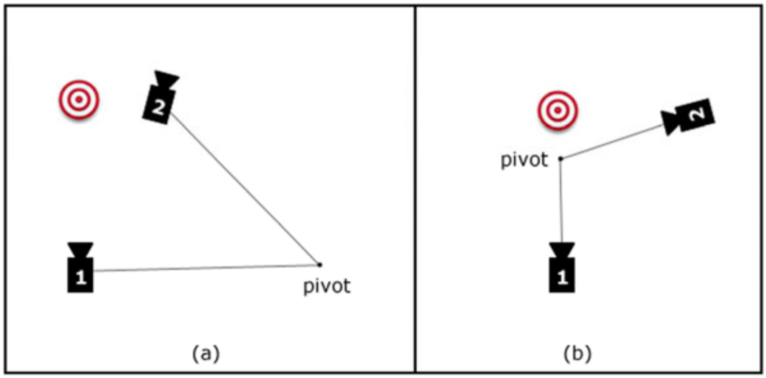

Fig. 1. Problems with the choice of inadequate pivot points. In both cases, when the camera moves from position 1 to position 2, the user loses sight the object being examined.

4 SiVIEP

The SiVIEP application was developed due to a need from Petrobras, a Brazilian Oil & Gas Company. The system interface is characterized by the presentation of geo-science and engineering models, which are represented by 3D objects under different scales and levels of detail (Fig. 2).

4.1 System Features

The SiVIEP application has a high complexity of requirements given to its composition of 3D objects, developed and designed with high-quality and close to reality.

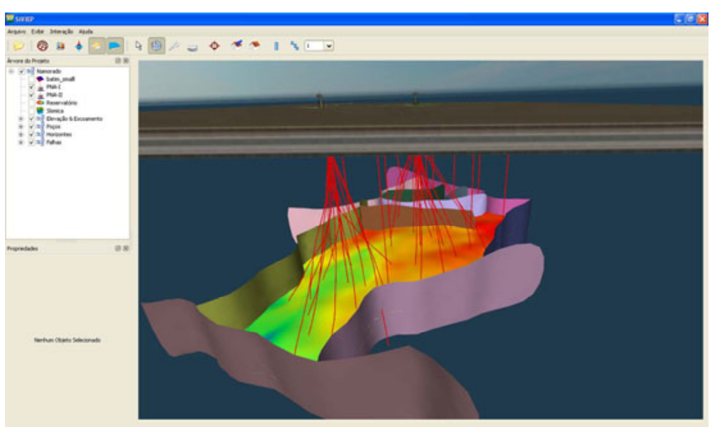

Fig. 2. SiVIEP interface example with a view of a petroleum field

SiVIEP users can navigate through the objects using tools like: *Fly* and *Examine*. The *fly* tool allows the camera to fly over the scene, at a specific speed. This tool can be controlled by the user by pressing the arrow keys of the keyboard to move the camera while guiding the direction of the motion with mouse movements. The *examine* tool allows the user to examine a certain object through drag movements using the left mouse button, which makes the camera rotate around the pivot point. The pivot point can be fixed by another tool of the system: *Change pivot point*.

5 Evaluation

The evaluation was planned according to the DECIDE *framework* [7], which presents a model to elaborate the evaluation of user–system interactions. After the preparation, the tests with the users were conducted to assess the following usability factors: ease of learning, efficiency, and user satisfaction with the system.

The results of the evaluation were analyzed and solutions were designed to solve the problems we had found.

5.1 Evaluation Planning

The evaluation was planned according to the guidelines of the DECIDE framework. The purpose of this framework is to ensure that all necessary steps in conducting an evaluation have been correctly applied correctly.

The test required selected users to have two different profiles, to demonstrate the different views and preferences between them. The two groups of users defined were: *advanced* users, with experience in the use of 3D visualization and 3D modeling applications, i.e., who use this type of software at least once a month; and *non-advanced* users, with little experience with 3D visualization applications, except for games, and who do not use 3D applications frequently. Some authors [8, 9] came to interesting conclusions about the behavior of non-advanced users. For example, they do not understand how a tool works or have a wrong understanding of it, causing them to expect a different behavior when using it. In some cases, they try to accomplish all tasks with a single instrument, even when they need a different set of tools for the task.

From the 16 participants of the tests, 6 were considered advanced, 8 non-advanced, 2 were used for the pilot tests, one from each group. They were between 19 and 30 years old, and 9 of them were male and 7 were females.

Before the interaction, we conducted a semi-structured interview with the users in order to detect the profile to which they belonged. After the interview, a short demonstration was presented to the user so that he/she could get an idea of how to navigate in SiVIEP. In some occasions, where individuals from the non-advanced group did not understand how to manipulate the controls and use the tools available, it was necessary to allow the user to quickly interact with the system, which did not interfere with the test results. Such situations had low occurrence in this test.

After the demonstration, we explained to the users how the test would be conducted and asked them to sign an informed consent form. Then, we applied the tasks in random order, to minimize the effects of system learning from one task to the other.

Each task was applied in the modes with and without the features of Multiscale navigation and inspection we were investigating. The users read the task description and were able to clarify any doubts about it. Because we wanted to test each cube-map-based navigation technique, we defined 3 tasks (A, B and C) for users to perform, each one involving one or more techniques.

The technique *Fly with Automatic Speed Adjustment* was to be used in all tasks because the user needs the *fly* tool to navigate through the objects to reach the location required by the tasks. Task A tested only the *Fly with Automatic Speed Adjustment* technique by establishing a situation where the user had to move from a specific side of the oil platform, traversing through a corridor on the same side and finally getting to the final point, which was the other platform located within his/her field of view. Task B primarily tested the *Examine with Automatic Pivot Point* technique: the user had to inspect four different points inside a specific oil platform, and to facilitate the task, the user could enable the *Change Pivot Point* tool followed by the *Examine* tool each time he/she passed by an inspection point. Finally, task C primarily tested the *Collision Detection and Treatment* technique: the user had to traverse an open and closed corridor inside the oil platform, and on this path there were platform objects as obstacles the user could or could not pass through during the interaction.

5.2 Test Execution

The user–system interaction was recorded in video and the users' comments in audio, both during the interviews and the test execution. The tests did not have a time limit. The observers took notes to record information that could not be captured by the recording equipment.

To execute each task, two different versions of SiVIEP were tested by each user: *automated*, in which the techniques to test were turned on so that the user did not have to worry about some navigation aspects; and *manual*, in which the techniques mentioned were turned off and some navigation parameters had to be explicitly set by the users. The versions were applied in a random order to reduce the learning effect of using the first version over the second. Data collected from the observation and the interviews were triangulated.

Primarily, pilot tests with two users were executed, one from the advanced group and the other from the non-advanced group. This pilot test was necessary and important because it revealed the material needed some changes in order to focus even more on the navigation techniques problem.

After executing each task, some questions were asked to the user, aiming to explore his/her opinion on the interaction with the system in each version (manual or automated).

At the end of the test execution, the user answered a questionnaire which consisted of the following statements about the interaction with SiVIEP, with a response scale ranging from 1 to 5, here identified as Q1, Q2, Q3, Q4 and Q5:

Q1: Using SiVIEP was (Hard .. Easy)

Q2: The system is similar to some other program you have used (Little .. A lot)

Q3: You performed the tasks successfully (None of them .. All of them)

Q4: What did you think of the manipulation and navigation of the model objects? (Hard .. easy)

Q5: In general, what do you think of your interaction with SiVIEP? (Poor experience .. Great experience)

5.3 Evaluation Results

This section describes the results of the usability tests and the problems found during the user–system interaction.

First, we asked the users' opinion about the beginning of the interaction. The users from the advanced group thought the beginning was easy, but it was even easier when testing the first task in the second version, because of the learning effect, and especially when the first version was the automated one. The users from the non-advanced group thought the beginning was difficult, but this situation got better when the first version tested was the automated one. According to most users from both groups, the automated version made the navigation simpler and avoided more errors.

For better comprehension of the results, they are organized in table 1, which contains the number of users, of each group, who preferred a specific version after doing each task.

In the advanced group, there was a difference in the users' preferences. In tasks A and C, the users preferred the automated version slightly: in the first case, because the automated version avoided some navigation and direction errors, and in the second case because it simulated the realism and provided a better sense of direction. The users who preferred the manual version, in these tasks, justified their choice by saying that they liked the automatic adjustment, but preferred to control the navigation speed. And, about the collision treatment, they thought that it was easier make task C without this technique due to the high speed this version can achieve when the camera is near the objects. Regarding task B, the users' opinions were divided. Those who chose the automated version justified their choice with comments like "this version helps me avoid some mistakes by distraction" since, in the manual version, the user can forget to select the center of rotation and get lost into the navigation field. On the other hand, the users who chose the manual version justified their choice by saying that they lost the vision of the center of rotation more easily, since the automatic adjustment set the pivot point automatically. But, they also felt lost during the navigation when forgot to set the pivot point manually and in their words "the screen jumps from a place to another into the navigation field".

Almost all users from the non-advanced group preferred the automated version of the navigation in tasks A and B given the facility to control and the precision of the manipulation. In task C, the users preferred the manual version given the speed with which they completed the task, since in this version there is not a collision treatment, and the speed is faster than the other version when the camera approaches the objects. However, four of the six non-advanced users complained about how difficult it was to control the camera in this version.

Table 1. Preference versions according to feedback from users of both groups

Task	Advanced Group		Non-advanced Group	
	Automated version	Manual version	Automated version	Manual version
A (fly)	4	2	7	1
B (examine)	3	3	7	1
C (collision)	4	2	2	6

Tables 2 and 3 present the questionnaire results of the advanced and non-advanced users' opinion about the navigation in SiVIEP, respectively. Individuals in the advanced group are herein called UA1, UA2, UA3, UA4, UA5 and UA6, whereas the individuals of the non-advanced group are identified as UN1, UN2, UN3, UN4, UN5, UN6, UN7 and UN8. To facilitate the reading, we are repeating the questions below:

Q1: Using SiVIEP was (Hard .. Easy)

Q2: The system is similar to some other program you have used (Little .. A lot)

Q3: You performed the tasks successfully (None of them .. All of them)

Q4: What did you think of the manipulation and navigation of the model objects? (Hard .. easy)

Q5: In general, what do you think of your interaction with SiVIEP? (Poor experience .. Great experience)

Table 2. Results of the questionnaire as applied to the users in the advanced group, where 1 is the worst response value and 5 is the best

	UA1	UA2	UA3	UA4	UA5	UA6	Avg
Q1	4	4	3	4	4	4	3,8
Q2	4	3	5	2	3	3	3,3
Q3	3	5	4	5	5	4	4,3
Q4	4	4	5	4	4	4	4,2
Q5	4	4	4	3	3	4	3,7

Table 3. Results of the questionnaire as applied to the users in the non-advanced group

	UN1	UN2	UN3	UN4	UN5	UN6	UN7	UN8	Avg
Q1	3	4	4	3	4	2	4	3	3,3
Q2	1	4	3	1	2	4	5	2	2,8
Q3	3	5	5	3	4	5	5	4	4,3
Q4	3	4	4	3	3	3	4	4	3,5
Q5	3	5	4	4	5	4	4	4	4,2

Some responses were not different than expected due to the users, such as the values obtained for questions Q2 and Q4, in both groups. The questions that had unexpected answers were Q1 and Q5: even though interaction with a 3D system is something new to the non-advanced users, they judged the interaction as easy and had a very good experience with it.

After answering the questionnaire, users could make suggestions for improvements of SiVIEP through written responses. The suggestions were very important to justify some answers and behaviors during the test. Two users from the advanced group and two users from the non-advanced group suggested an insertion of keyboard shortcuts during the interaction with the *fly* tool. Another suggestion was to momentarily increase the speed in the automated version, by means of the scroll wheel, thus customizing the behavior in this version. One user, from the non-advanced group, said that it was difficult to locate himself in the model, arguing thus for the definition of a target symbol to replace the mouse pointer, indicating the direction of the user navigation. Another user, from the non-advanced group, suggested the insertion of control tools for zooming.

6 Conclusions

This work presented a usability test of three navigation cubemap-based techniques developed to improve the *examine* tool, by providing a way to automatically determine the pivot point, and the *fly* tool, in which collision support and automatic speed adjustment in relation to the scale were implemented.

The usability tests were performed to assess the ease of learning and user satisfaction with the proposed techniques. The test consisted of three tasks, in which each one tested a technique proposed. They were performed by two user groups with different profiles.

The results indicate that the techniques presented here may improve the users' navigation experience. Usability tests confirmed that, from the 6 individuals of the advanced group and 8 of the non-advanced group, respectively, about 70% and 90% preferred the version of the application that included the proposed techniques. In face of their experience with 3D applications, users from the advanced group preferred the manual controls, most of the time, but acknowledged that the automatic adjustment minimized the mistakes that could happen and found the collision treatment and detection very important to the interaction. Users from the non-advanced group were surprised to see that they were able to interact well, most of the time, with a 3D system like SiVIEP, because that was their greatest concern before the interaction had begun. They preferred the automated version, most of the time, given the facility that this version offered for navigation and manipulation of objects.

The 3D navigation in multiscale environments is becoming more common, especially due to the popularization of technology and the need to develop 3D systems with increasing complexity in areas such as engineering, geoscience, games, and maps, to name a few. This popularization brings further problems to be solved in terms of navigation and user interaction, providing a lot of challenges to HCI researchers and practitioners. We believe the usability tests we have conducted shed some light in the problems different users experience during interaction and navigation in these environments.

Acknowledgments. All authors thank TecGraf, PUC-Rio for its support to their research work. Simone Barbosa also thanks CNPq for its support though grant #313031/2009-6. Alberto Raposo thanks FAPERJ for its support though grant #E-26/102.273/2009.

References

1. Andrade, T.M., Seixas, M.L.A., Raposo, A.B.: Uso de Técnicas da Engenharia Semiótica para Avaliação de um Sistema 3D na Área de Exploração e Produção de Petróleo. In: XXXVII Seminário Integrado de Software e Hardware – SEMISH 2010, pp. 425–439. SBC, Belo Horizonte (2010)
2. Jul, S., Furnas, G.W.: Critical zones in desert fog: aids to multiscale navigation. In: UIST 1998: Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology, pp. 97–106. ACM, New York (1998)

3. Perlin, K., Fox, D.: Pad: an alternative approach to the computer interface. In: SIGGRAPH 1993: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, pp. 57–64. ACM, New York (1993)
4. Bederson, B.B., Stead, L., Hollan, J.D.: Pad++: a zooming graphical interface for exploring alternate interface physics. In: UIST 1994: Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology, pp. 17–26. ACM, New York (1994)
5. Mccrae, J., Mordatch, I., Glueck, M., And Khan, A.: Multiscale 3d navigation. In: I3D 2009: Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games, pp. 7–14. ACM, New York (2009)
6. Trindade, D.: Técnicas de Navegação 3D Usando o Cubo de Distâncias. Tese de mestrado. Departamento de Informática. PUC-Rio, Rio de Janeiro (2010)
7. Preece, J., Rogers, Y., Sharp, E.: Interaction Design: Beyond Human-computer Interaction, 1st edn. John Wiley & Sons, New York (2002)
8. Fitzmaurice, G., Matejka, J., Mordatch, I., Khan, A., Kurtenbach, G.: Safe 3d navigation. In: I3D 2008: Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games, pp. 7–15. ACM, New York (2008)
9. Ware, C., Osborne, S.: Exploration and virtual camera control in virtual three dimensional environments. In: SIGGRAPH 1990: Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques, pp. 175–183. ACM, New York (1990)