Augmented Reality Without Markers

Luis F. A. Silva   Frutuoso G. M. Silva
Instituto de Telecomunicações
Universidade da Beira Interior
Rua Marquês D'Ávila e Bolama, Covilhã, Portugal
m2086@ubi.pt, fsilva@di.ubi.pt

Abstract
This paper introduces an augmented reality application for web that works without markers. We propose a solution to manipulate virtual objects without using markers, i.e. we replaced the physical marker by user face. For this we had to develop also a method to manipulate the virtual models (e.g. to rotate models) based on user gesture allowing doing the same as the marker. Finally we present a simple augmented web game showing our approach without the use of markers.

Keywords
Augmented reality without markers, Augmented web game, Gestural interface

1. INTRODUCTION

The constant growth of the Internet, where a variety of new ideas and information are published every day, make it in a excellent media to present information. Recently, Augmented Reality (AR) has been used also as a way to publish information in Internet. But we can see that the augmented reality had a weak growth until 2001 [Azuma 01], but recently is visible a large adherence to AR, particularly in marketing and dissemination of products and concepts. It's worth highlighting as examples the marketing of the latest movies, Star Trek® [1], Avatar® and among others by AR. Sales are another area where the onset of AR has been noticed, where brands such Citroën® (C3 Picasso) or Nissan® (Cube) are using AR to advertise their models.

According to Azuma[Azuma 01], we have an augmented reality application when we add virtual objects to a real environment, i.e. the objects created by computer (i.e. virtual objects) are inserted in a real environment where the real environment is normally captured by a camera. But the augmentation is conventionally in real time and allowing user interaction. Because what distinguishes AR from a simple 2D/3D scene placed over an image, captured by a camera, is the interactivity. In other words, in AR the users have the power to interact in real environment with virtual objects, thus giving a perception that these objects are in their world.

In general, in most applications of AR the objects are added to the real environment according to the marker captured by digital camera [Liarokapis 04b] or by geographic coordinates provided by GPS [Narzt 06], between other methods. But most AR applications are using markers to positioning and orienting the virtual objects.

Recently, Zhou *et al.* [Zhou 08] presented a study about the trends in augmented reality, tracking, interaction and display, in last ten years. In this study, the authors say that hand gesture recognition is one of the most natural ways to interact with an AR environment but this in the context of tangible interfaces. In the context of gestural interfaces, they identify only one method based on finger tracker. Unfortunately, in this work we not find any reference to augmented reality applications for web browser. However, Liarokapis *et al.* [Liarokapis 04a] presented an educational application that allows users to interact with 3D Web content using virtual and augmented reality.

In this paper we describe a web augmented reality application that works without the use of markers. This application have been developed to web, and it is available for those that have a web browser and a webcam. Section 2 describes our approach that works without markers to manipulate the virtual objects. In section 3 is presented the augmented web game developed to show the potentialities of our approach. Lastly, section 4 presents some conclusion and future work.

2 AR without Markers

One of the objectives of our work is simplify the interaction, for that reason we have been removed markers, being the interaction made only with the body of the user (hands and head). The alternative chosen as reference point for the model was the use of the face user. Thus it is possible to put the virtual model in the scene according to the position of the face user. However, this alternative not al-

[1] see www.experience-the-enterprise.com

lows to rotate the model as we can do with markers. Hence we divided the screen in several areas allowing the rotation of the model based on motion detection for each area (as shown in Figure 1).

Then application makes the following calculus based in face and motion detection for each frame:

- Calculation of z co-ordinate is based on the user's face, i.e. the distance that he/she is from the webcam;
- Calculation of x and y is based on the position of the face user in the environment (i.e. similarly with the markers);
- The orientation of the object is made by detecting motion of the user in different areas of the screen.

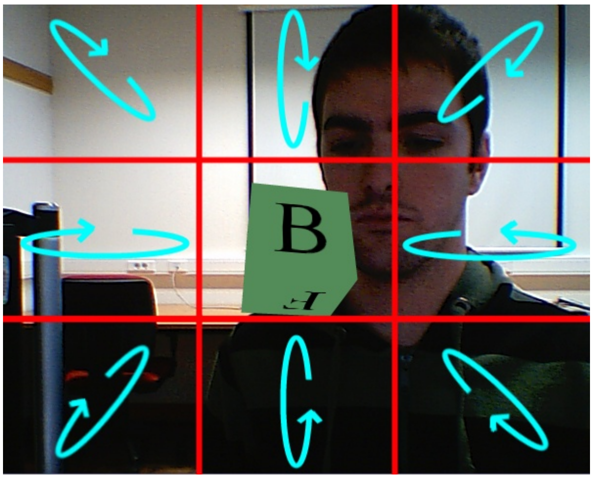

Figure 1. Interaction areas for rotating the object based on motion detection.

In short, the face of the user will be used as reference point to positioning the objects in virtual environment and the motion detection to interact with virtual object (e.g. to rotate). To prevents a conflict between the functions of face detection and movement detection, it was considered a priority rule to establish a functional hierarchy that had success. So, first is considered the face detection phase and if this function returns a new detection, the motion capture is aborted. If the face detection is not significantly changed it's taken into account the motion capture phase. Normally,

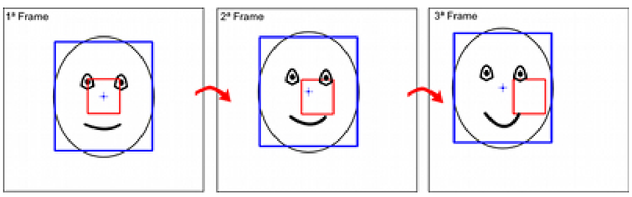

Figure 2. Example of the threshold in a sequence of frames.

the face is presented in all frames but it is only considered a new detection when it is in a different position from last detection. Thus was created a threshold to the position of the face, when this threshold is exceeded it is considered a new detection. This threshold is calculated based on the size of the face and the screen resolution (see Figure 2).

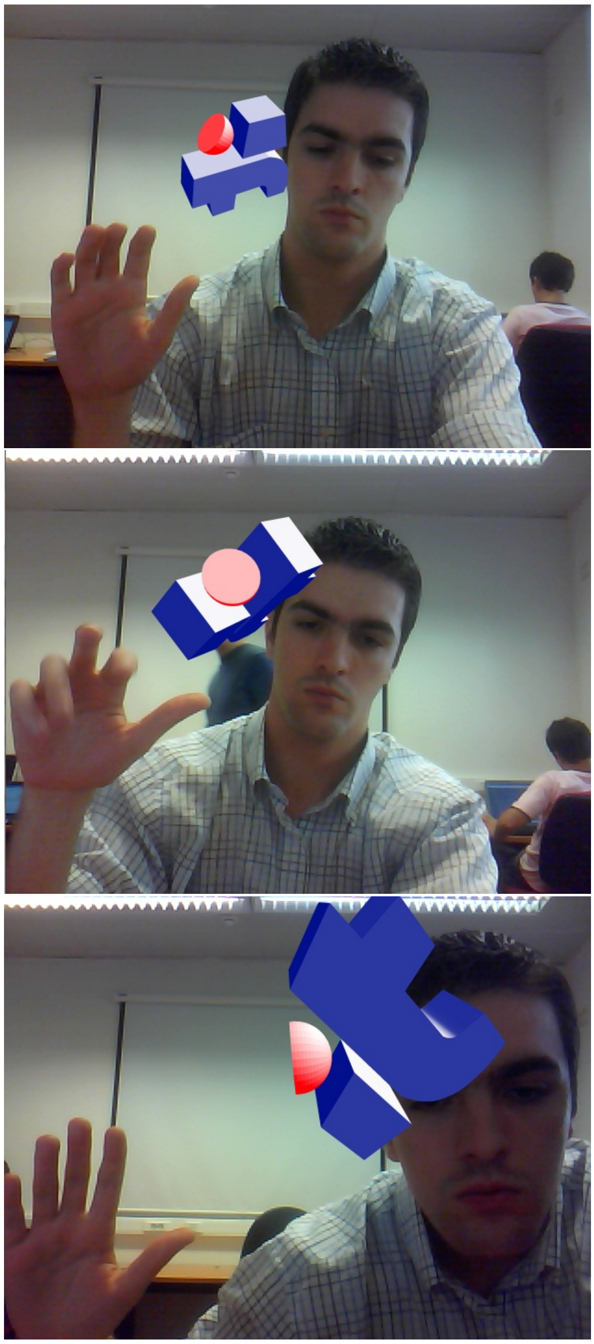

Figure 3. Rotating the object

The areas of interaction are pre-defined. So if is detected movement in right area of the screen, the virtual object will perform a rotation in y-axis (rotation to the right). The rotation in x-axis is activated if is detected movement in top or bottom area of the screen. It is also possible make a rotation simultaneously on two axes x and y, which will be triggered if there is a movement in one corner of the screen. Note that the motion detection is restricted to small areas, i.e. it is not allowed in large areas. This prevents that large movements in area are considered, because most of these situations are not orders of rotation. Only the movements detected for these small areas will be considered as instructions for the application (e.g. a simple movement of the fingers in these areas). Figure 3 shows some examples of the use of the application. In this example the virtual object is a 3D logo of the Institute of Telecommunications (IT) and the user can rotate it based on movements in predefined areas.

3 Augmented Web Game

To show the potentialities of our approach we develop a web game using augmented reality without use of markers. As explained in previous section we use the user face as reference to put the virtual model in the scene and its manipulation is based on gesture of the user.

The main ideia of the game is to find the letters of a word. For that, we have a virtual cube, which has a single character on each face, and the user has to rotate the cube to find the characters of the word (see Figure 4). The word is presented below the cube using several plans, one for each character of the word to find. These plans are initialised with exclamation character (i.e. '?'), which is replaced for the correct character when it appears in front face of the cube. Thus the user need to rotate the cube to find the characters of the word. The rotation of the cube is available in two axis, x and y, and it can be activated by the user movements only in four special areas, namely in upper, bottom, left and right area of the image. Note that it is possible use the eight areas of interaction with the model, as described in previous section, but we use only four to simplify the game (i.e. using only four areas is more easy to maintain the front face of the cube aligned with user).

When the user activates a rotation of the cube, the game generates a visual feedback to help him. The feedback helps user to understand which type of movement was realised. This feedback is created using a set of arrows that are moving in the direction of the rotation, as shown in the second and third pictures of Figure 4. The game has a timer clock and shows the number of words that are available yet. This information is displayed above the cube, as shown in first picture of Figure 4.

The position and size of the cube depends of the position and size of the face of the user, respectively as explained in previous section. However, when the face of the user disappears of the image, the cube stays in the last position of it (as shown in Figure 4).

The game uses a XML configuration file with the words of the game. Thus, it is possible to change the words of the game by changing this XML file. This simple game shows that our approach is viable and can be adopted in other applications on web.

4 Conclusion and future work

The generality of augmented reality applications are dependent of markers to run. But we presented a new approach to run without markers. Our augmented web game

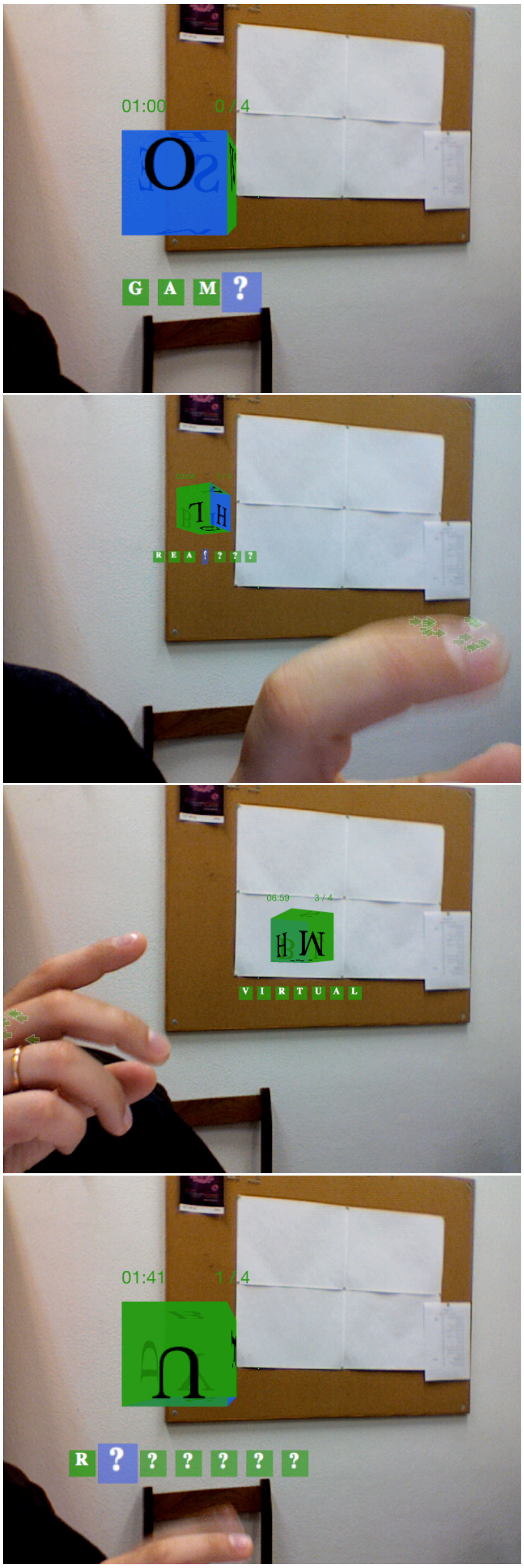

Figure 4. Augmented web game

showed that this new way of interaction without markers is useful and can be a simple solution for augmented reality applications in web browser. Therefore it is possible have augmented reality applications without physical markers with similar functionalities to applications that use physical markers. Note that the application was developed in Actionscript and compiled for Adobe Flash. Therefore the application has the ability to be used via web browser by any user that have the Flash Player plugin.

In future, we want to make an evaluation of the new way of interaction without markers with a group of users. Besides, we want to see how this type of applications will run using a web API for creating interactive 3D applications in browser, as WebGL [WebGL] from Khronos Group.

Acknowledgments

The first author would like to thank the grant given by Instituto de Telecomunicações to support his work.

References

[Azuma 01] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre. Recent advances in augmented reality. *IEEE Computers & Graphics IEEE*, 21(6):34–47, 2001.

[Liarokapis 04a] Fotis Liarokapis, Nikolaos Mourkoussis, Martin White, Joe Darcy, Maria Sifniotis, Panos Petridis, Anirban Basu, and Paul F. Lister. Web3d and augmented reality to support engineering education. *World Transactions on Engineering and Technology Education*, 3(1):11–14, 2004.

[Liarokapis 04b] Fotis Liarokapis, Martin White, and Paul Lister. Augmented reality interface toolkit. In *IEEE Proceedings of the Eighth International Conference on Information Visualisation*, pages 761–767, 2004.

[Narzt 06] Wolfgang Narzt, Gustav Pomberger, Alois Ferscha, Dieter Kolb, Reiner Müller, Jan Wieghardt, Horst Hörtner, and Christopher Lindinger. Augmented reality navigation systems. *Univers. Access Inf. Soc.*, 4(3):177–187, 2006.

[WebGL] WebGL. Webgl - opengl es 2.0 for the web. Khronos Group, http://www.khronos.org/webgl/.

[Zhou 08] F. Zhou, H. B. L. Duh, and M. Billinghurst. Trends in augmented reality tracking, interaction and display: A review of ten years of ismar. In *7th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2008)*, pages 193–202, 2008.