

Augmented Reality for Web - A new interaction method without markers

Luis F. A. Silva and Frutuoso G. M. Silva

Instituto de Telecomunicações, University of Beira Interior, Portugal

ABSTRACT

In the new age of omnipresent internet, browsers are accessible on every device in everyday's life. Thus it is also needed new interaction methods to facilitate the interaction through browser. Augmented reality is one of the techniques that is available now via browser. In this paper we present a new method of interaction without physical markers for augmented reality via browser. The first part of the paper describes the use of augmented reality via browser using physical markers and its limitations. In the second part, we propose a solution to solve the limitations of augmented reality using physical markers, i.e. we propose a new interaction method to manipulate virtual objects without using physical markers. Finally, we show a simple augmented web game based on the new interaction method proposed.

I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality

1. Introduction

The Internet is an excellent media to present ideas and products. Today, the Augmented Reality (AR) via web browser is used to publish all kind of information in Internet. However, AR had a weak growth until 2001 [ABB*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[®] and Avatar[®] and among others. 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 [ABB*01], we have an augmented reality application when we add virtual objects to a real environment, i.e. objects created by computer are inserted in a real environment where it is captured by a camera. But the augmentation is conventionally in real time and allows user interaction. In short, in AR the users have the power to interact in real time with virtual objects, which gives the perception that these objects are in the real world.

In general, in most AR applications the objects are added to the real environment according to the physical marker captured by camera [LWL04], between other methods. Thus, the physical marker is used to positioning and orienting the virtual objects in AR applications.

Several toolkits are available for the development of augmented reality applications, where the OSGART [LGSB06] library is a reference in the area. But all of them use physical markers to put and manipulate the virtual objects.

Recently, Zhou *et al.* [ZDB08] presented a study about the trends in augmented reality, tracking, interaction and dis-

play, 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 in the context of tangible interfaces. But in this context, they identify only one method based on finger tracker. Unfortunately, this study does not include any reference to AR applications via browser. However, Liarokapis *et al.* [LMW*04] presented an educational application that allows users to interact with 3D Web contents using virtual and augmented reality.

In this paper we describe a new method of interaction without physical markers for AR via web browser. Besides, we present a simple web game based on the new interaction method developed to show their potentialities. Section 2 presents an introduction to augmented reality based on physical markers. Section 3 describes the new interaction method proposed that work without physical markers. Section 4 describes an augmented web game developed to show the potentialities of our interaction method. Section 5 presents some results of usability tests made for our interaction method when compared with the use of physical markers. Lastly, section 6 presents some conclusion and future work.

2. AR with physical markers

In AR applications normally the virtual objects are placed in the real scene according to a reference marker and the interaction is based on direct manipulation of this physical marker. Figure 1 shows an example of these kind of markers. By the segmentation of the captured image it is possible to find the marker and then put the virtual object in the position of the marker. Thus the markers have specific characteristics for a better contrast with the environment to facilitate

the segmentation process. Hence normally are used square markers in black and white with symbols inside, helping for better identification of them.

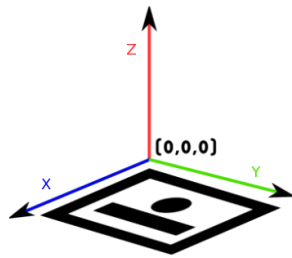


Figure 1: A possible marker in the axes.

The position and the orientation of the virtual object is calculated based on the position of the marker in the captured image, if it is equivalent to predefined marker in application. The marker allows user interaction with virtual object as shown in Figure 2. Thus moving the marker the user moves the reference position in virtual environment, consequently leads to the calculation of new coordinates of virtual object. Then the object is moved to new coordinates and a new image will be presented to the user. So the calculation of the positions (x, y, z) of virtual object is done as following:

- The coordinate z , is calculated across the distance of the marker to webcam, i.e., the size of the marker;
- The x and y coordinates are calculated through the values of the pixels where the marker appears;
- The orientation of the object is calculated in relation to position of the camera in the virtual environment (see [JMC93]), which make possible to map or re-create a 3D environment from a captured image.

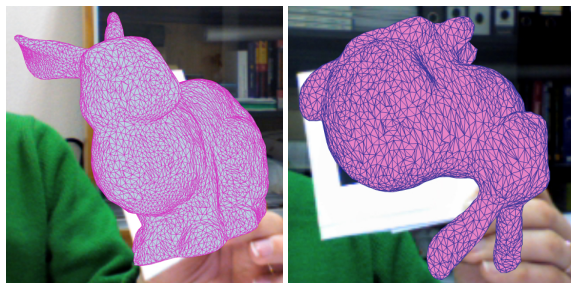


Figure 2: Direct manipulation of the object using the physical marker.

Therefore the user can rotate the virtual object by rotating the physical marker. But the rotation is limited in two axis (e.g. in X and Y axis, according to Figure 1). Only the rotation in Z axis is completely free (i.e. the rotation in the axis perpendicular to screen). Hence it is not possible to see the bottom of the virtual object because it is not possible to rotate 90 or more degrees in X axis (or in Y axis).

To avoid this limitation we proposed a new interaction method to manipulate virtual objects in an augmented reality applications via browser, that is described in next section.

3. The new interaction method

One of the objectives of our work was simplify the interaction in AR. Thus we have been removed physical markers,

being the interaction made only with the body of the user (i.e., hands and head). But the new method of interaction will have to allow positioning and orienting of the virtual model in the space.

The alternative chosen as reference point for positioning the model was the use of the face user. This means that is possible to put the virtual model in the scene according to the position of the user face. However, this alternative does not allow orienting the model as we can do with markers. To give the user an alternative to orient the model we developed a solution based on user gestural. For that, we divided the screen in several areas allowing the rotation of the model based on motion detection for each area. The areas of interaction are predefined, see [SS10] for details. Thus 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 strategy 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).

The system makes the following calculus for positioning and orienting the virtual object based in face and motion detection:

- The z coordinate is calculated based on the user's face, i.e. the distance that he/she is from the webcam;
- The x and y coordinates are calculated based on the position of the user's face in the environment (i.e., the position of the face in each frame);
- The orientation of the object is calculated based on detection of the motion in predefined areas of the screen (e.g., using gestures).

In short, the face of the user is used as reference point to positioning the object in virtual environment and the user gestures to interact with the virtual object (e.g. to rotate it). To prevent a conflict between the functions of face detection and movement detection, it was considered a priority rule to had success. First, the system considers the face detection phase and if this function return a new detection, the movement detection phase is aborted. If the face detection is not significantly changed it is taken into account the movement detection phase.

Normally, the face is presented in all frames but it is only considered a new detection when it has changed the position relatively to last detection. Besides 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.

Figure 3 shows an example of the use of the system without physical markers. 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. This system was introduced in [SS10], but the methods for face and movement detection are not described yet. Thus



Figure 3: Manipulating virtual object by gestures.

face and movement detection will be explained in detail in the following sections.

3.1. Face detection

There are a lot of algorithms that can be applied for face detection but we needed a rapid one. Then our choice was the algorithm proposed by Viola and Jones [VJ01] which enables a rapid detection of objects through Boosted Cascade of Simple. This algorithm includes three parts, the Integral Images, AdaBoost and the Cascade Classifier.



Figure 4: Basic Haar Features.

The algorithm is based on application of Features, which are useful to detect patterns in images. For example, the Features can be tags with geometric characteristics as shown in figure 4. These Features will replace pixel-by-pixel search and thus the possibility of areas of different sizes of analysis (i.e., scalable in image).

The implementation of the technique for analysis of specific characteristics on images was presented by Papageorgiou *et al.* [POP98]. They proposed a Feature with two adjacent rectangles, but Viola and Jones extend it for more adjacent rectangles. In summary, the Features allow to find a specific region in an image, by contrast or nearness of shades of the analyzed areas, that is not possible in the analysis of single pixels.

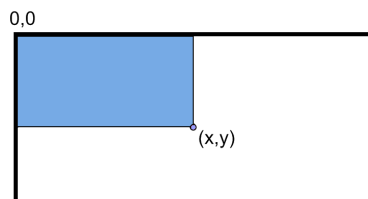


Figure 5: Summed Area Tables representation.

Integral Images are sub-representations of the full image where estimates the Summed Area Tables (see figure 5)

based on the work of Crow [Cro84]. This calculation is done with an array I of values corresponding to the intensities of all pixels $p(x',y')$, which are to the left and above the pixel (x,y) in Integral Image analyzed. The formula for this result is follows:

$$I(x,y) = \sum_{x' \leq x, y' \leq y} p(x',y') \quad (1)$$

The method of Lienhart and Maydt [LM02] was not used, which allows the Integral Images with 45 degrees can be calculated. The method that we implemented is less effective on detection because it uses a limited set of Features, but on the other hand, it is more efficient computationally.

The second step is the choices of Integral Images with the help of AdaBoost classifier. This classifier is a learning machine [FS95] with a objective of selecting the images closer to the Feature for main analysis. This classifier reduces the number of sub-images (50% according to Viola and Jones) considered for search, increasing the performance of the classification method. Briefly, this classifier uses a simple function AB_j :

$$AB_j(x,y) = \begin{cases} 1 & \text{if } F_j(x) < \sigma_j \theta_j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Where F_j represents the Feature, θ_j the threshold for the classification of the Feature and σ_j defines what will be the orientation of the signal in the *if* rule.

The third step uses the classifier individually and sequentially for each sub-image. It is called the Cascade of Classifiers and uses the threshold to control the detection complexity, which will affect the cost and accuracy. Thus if the threshold is low, it will pass less sub-images with possibility of success for a next stage where the analysis is more specific. This process continues until last state, leaving just the sub-images with more interest, i.e this technique rejects irrelevant images between states, which are not taken into account in the next ones, as the background image. The face detection method was implemented guided by the OpenCV Library [Ope]. For more details see also [VJ01].

3.2. Movement detection

The movement detection is the second way of interaction with the application, which enables rotating the virtual objects. Several techniques for motion detection are available [EHD00, WXF*04, ZK04, YC09]. But the technique chosen consists in frame subtraction because it be simple and fast method. It looks for the variations between the current and previous image. This technique implies a stable camera (e.g., laptop webcam) in order for the capture not to register its own movements. The subtraction of frames is calculated analyzing the value of the pixels for two instants of time, as shows the equation 3.

$$R = |L_2(x_2, y_2, t_2) - L_1(x_1, y_1, t_1)| \quad (3)$$

In equation 3, the R is the modulus of the difference, between the lightness of the frames L_2 and L_1 in coordinates (x,y) , for two instants of time, t_1 and t_2 .

Basically this will walk-through all pixels of the two images to detect changes of lightness. Then the results with the value zero are those with little or no light, i.e. no movement in this area. On the other hand, values greater than zero show a variation between two images, which corresponds to movements in these areas. However, this technique is influenced by light conditions and by noise in captured images. In capture of images can appear problems due to noise or failures because the capture depends of the type of webcam. We can improve the method using some kind of auto-adjustable value to clear that noise based on the characteristics of the webcam. The improvement of the algorithm can also be made using other techniques as contours calculation [CXS*07], color algorithms [MRG99] or background removal [ZK04].

4. Augmented Game

To show the potentialities of our approach we develop a web game using augmented reality without the use of physical 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 gestures of the user.

The main idea of the game is to find the letters of a word. For that, we have a virtual cube that has a single letter on each face, and the user has to rotate the cube to find the correct letters of the word (see Figure 6). The word is presented below the cube using several plans, one for each letter of the word to find. These plans are initialized with a question mark (i.e. '?'), which is replaced for the correct character when it appears in the front face of the cube. Therefore the user needs to rotate the cube to find the correct letters of the word, but only one face have the correct letter for each position.

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 using only the four areas makes the game more easy because is more easy to maintain always the front face of the cube aligned with user.

The game generates a visual feedback to help user every time he activates the rotation of the cube. This feedback helps the user to understand which type of movement was activated. The feedback is created using a set of arrows which are moving in the direction of the rotation, as shown in the second and third pictures of Figure 6. The game has also 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 6.

The position and size of the cube depends of the position and size of the user's face, respectively as explained in previous section. However, when the face of the user disappears from the image, the cube stays in the last position (as shown in Figure 6).

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

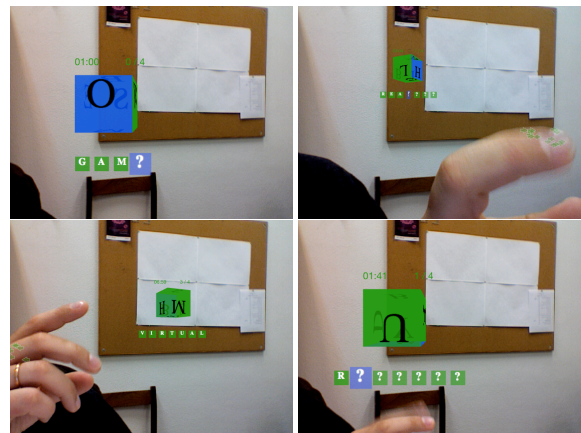


Figure 6: Playing the augmented web game by gestures.

5. Usability tests

To evaluate the new interaction method we developed some tests with several users, comparing interaction with markers with interaction without physical markers. For that, we made two tests with users, in the first (see Figure 7), a simple application with markers was compared with a similar one without physical markers (i.e. using our interaction method). In the second test (see Figure 8), the users used two applications, one with the augmented game and other with a simple 3D model, both using our interaction method. Our system is available in the web to be tested, i.e. to evaluate its performance online (see <http://webar.it.ubi.pt/>).



Figure 7: First test with two systems: with physical markers and without (i.e. using gestures).

The majority of the users involved in tests were students and they had between 10 and 30 years old, i.e. eighty percent of them and only twenty percent had more than 30 years old. Sixty percent of the users never had used augmented reality applications and only twenty percent had tested augmented reality applications via web.

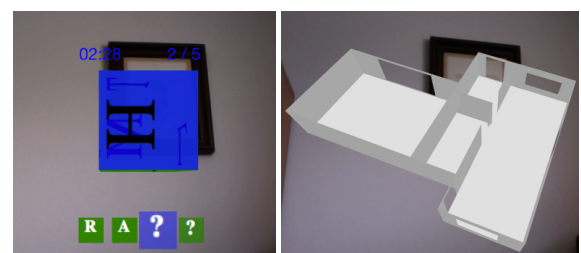


Figure 8: Second test with our interaction method.

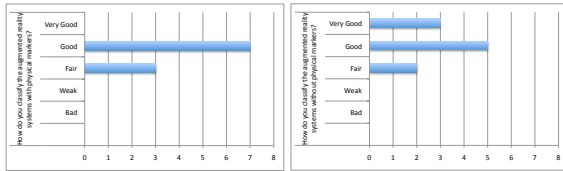


Figure 9: How do you classify the AR systems - Q1: with physical markers? Q2: without physical markers?

After each session the users answered to a questionnaire about their experience with the two systems (i.e. with physical markers and without). The first two questions are about how to classify the augmented reality systems with physical markers and without markers. As we can see in Figure 9 seventy percent of the users consider the augmented reality systems with physical markers good and only thirty percent consider it fair. But for the augmented reality systems without physical markers, eighty percent of the users consider it good or very good, i.e. only twenty percent consider it fair, as you can see in Figure 9.

The second two questions are about how to classify the interaction for each system. For augmented reality systems with physical markers only forty percent of the users consider it good. But for augmented reality systems without physical markers sixty percent of the users consider it good (see Figure 10).

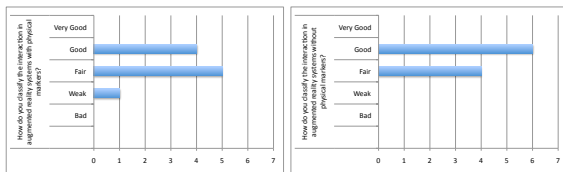


Figure 10: How do you classify the interaction in AR systems - Q3: with physical markers? Q4: without physical markers?

The other question evaluates the ease of use of each system. For augmented reality systems with physical markers, sixty percent of the users consider the ease of use good or very good, while ninety percent of the users consider the ease of use of the augmented reality systems without physical markers good or very good (see Figure 11).

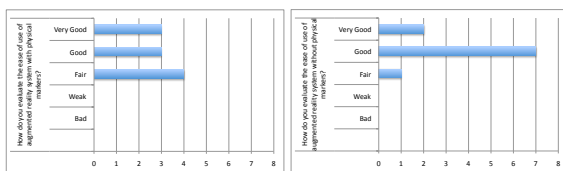


Figure 11: How do you evaluate the ease of use of AR systems - Q5: with physical markers? Q6: without physical markers?

Finally, we inquired the users how they rate the new method of interaction in two questions: first how they evaluate the method to put virtual objects based on the head position, and second how they evaluate the manipulation of the virtual object using gestures. In the first case, ninety percent

consider good or very good the method used to put the virtual object in scene based on head position. In second case, also ninety percent of the users consider good or very good the method used to manipulate the virtual object based on gestures.

When we asked the users which was the system that they liked the most to test, ninety percent chosen the system without physical markers.

In short, our new method of interaction for augmented reality applications via browser without the use of physical markers obtained always better results than the system with physical markers. Besides our method to manipulate the virtual object has more degrees of freedom when compared with a physical marker to manipulate the virtual object. Hence it is not possible to compare the performance for the two methods because they are not comparable due to different degrees of freedom. Thus, it is not possible developed tests of performance for a specific tasks because the two methods have different degrees of freedom. But of course, that rotating the virtual object using a physical marker (i.e., rotation in Z axis) is faster than in our system.

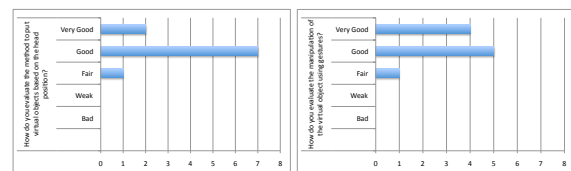


Figure 12: How do you evaluate the method to - Q7: put virtual objects based on the head position? Q8: manipulate virtual objects using gestures?

6. Conclusions and future work

The generality of AR applications are dependent of physical markers to run. But we presented a new approach to run without physical markers. The new interaction method proposed, which is used in our augmented web game, showed that this new way of interaction without physical markers is useful and can be a simple solution for augmented reality applications in web browser. Our method to manipulate the virtual object has more degrees of freedom when compared with a physical marker to manipulate the virtual object. Besides, it showed that is possible to have augmented reality applications without physical markers with more functionalities than applications that use physical markers.

Moreover these type of applications has the ability to be used via web browser by any user that has the Flash Player plugin. Thus, this new interaction method can be used in any type of devices, for example in mobile devices like smart phones or tablets.

In future, we want to see how this type of applications will run using a web API for creating interactive 3D applications in browser (e.g., WebGL). In addition, we want to test our method for mobile devices due to its simplicity/performance.

Acknowledgements

This work has been partially supported by the Fundação para a Ciência e a Tecnologia (FCT) and the Instituto de Telecomunicações.

municações (IT), through the MOGGY project, whose reference is PTDC/EIA/70830/2006.

[ZK04] ZANG Q., KLETTE R.: Robust background subtraction and maintenance. In *17th International Conference on Pattern Recognition* (2004), pp. 90–93. 3, 4

References

- [ABB*01] AZUMA R., BAILLOT Y., BEHRINGER R., FEINER S., JULIER S., MACINTYRE B.: Recent advances in augmented reality. *IEEE Computers & Graphics* 21, 6 (2001), 34–47. 1
- [Cro84] CROW F. C.: Summed-area tables for texture mapping. In *International Conference on Computer Graphics and Interactive Techniques* (1984), pp. 207–212. 3
- [CXS*07] CHAOHUI Z., XIAOHUI D., SHUOYU X., ZHENG S., MIN L.: An improved moving object detection algorithm based on frame difference and edge detection. In *Fourth International Conference on Image and Graphics (ICIG 2007)* (2007), pp. 519–523. 4
- [EHD00] ELGAMMAL A., HARWOOD D., DAVIS L.: Non-parametric model for background subtraction. In *6th European Conference on Computer Vision* (2000), pp. 751–767. 3
- [FS95] FREUND Y., SCHAPIRE R.: A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational Learning Theory: Eurocolt* (1995), pp. 23–37. 3
- [JMC93] JANIN A. L., MIZELL D. W., CAUDELL T. P.: Calibration of head-mounted displays for augmented reality applications. In *IEEE Virtual Reality Annual International Symposium* (1993), pp. 246–255. 2
- [LGSB06] LOOSER J., GRASSET R., SEICHTER H., BILLINGHURST M.: Osgart - a pragmatic approach to mr. In *5th IEEE/ACM International Symposium on Mixed and Augmented Reality* (2006). 1
- [LM02] LIENHART R., MAYDT J.: An extended set of haar-like features for rapid object detection. In *IEEE ICIP 2002* (2002), pp. 900–903. 3
- [LMW*04] LIAROKAPIS F., MOURKOUSSIS N., WHITE M., DARCY J., SIFNIOTIS M., PETRIDIS P., BASU A., LISTER P. F.: Web3d and augmented reality to support engineering education. *World Transactions on Engineering and Technology Education* 3, 1 (2004), 11–14. 1
- [LWL04] LIAROKAPIS F., WHITE M., LISTER P.: Augmented reality interface toolkit. In *IEEE Proceedings of the Eighth International Conference on Information Visualisation* (2004), pp. 761–767. 1
- [MRG99] MCKENNA S. J., RAJA Y., GONG S.: Tracking colour objects using adaptive mixture models. *Image and Vision Computing* 17, 3 (1999), 225–231. 4
- [Ope] OPENCV: Opencv library. <http://sourceforge.net/projects/opencvlibrary/>. 3
- [POP98] PAPAGEORGIOU C., OREN M., POGGIO T.: A general framework for object detection. In *International Conference on Computer Vision* (1998), pp. 555–562. 3
- [SS10] SILVA L. F. A., SILVA F. G. M.: Augmented reality without markers. In *Actas do Interação* (2010), pp. 189–192. 2
- [VJ01] VIOLA P., JONES M. J.: Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Vision and Pattern Recognition* (2001), pp. –. 3
- [WXF*04] WANG Z., XU Y., FORD J., MAKEDON F. S., ZHENG Z., GAO L., PEARLMAN J. D.: An adaptive approach for image subtraction. In *26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (2004), pp. 1818–1820. 3
- [YC09] YU Z., CHEN Y.: A real-time motion detection algorithm for traffic monitoring systems based on consecutive temporal difference. In *Proceedings of the 7th Asian Control Conference* (2009), pp. 1594–1599. 3
- [ZDB08] ZHOU F., DUH H. B. L., BILLINGHURST M.: Trends in augmented reality tracking, interaction and display: A review of ten years of ismar. In *7th IEEE/ACM International Symposium on Mixed and Augmented Reality* (2008), pp. 193–202. 1