Eurographics Italian Chapter Conference (2010)
E. Puppo, A. Brogni, and L. De Floriani (Editors)

Monitor-Based tracking system for Wide Augmented Reality Environments

G.M. Re[1], G. Caruso[1], P. Belluco[1] and M. Bordegoni[1]

[1]Politecnico di Milano
Department of Mechanical Engineering, Milan, Italy

Abstract

In Augmented Reality (AR) applications, the technological solutions used for tracking objects in the real environment are generally related to the conditions of the environment and to the application purposes. The selection of the most suitable tracking technology satisfies the best compromise among several issues including performance, accuracy and easiness of use. This paper describes an AR tracking approach based on a marker-based tracking and its development for wide environment by displaying itself on a monitor. This solution allows us to improve the visibility of the marker and the tracking, thanks to a dynamic control of marker's dimensions. The technical features and performances of our approach have been assessed by several testing sessions focused on comparative analyses.

Categories and Subject Descriptors (according to ACM CCS): Multimedia Information Systems [H.5.1]: Artificial, augmented, and virtual realities—; Pattern Recognition [I.5.2]: Design Methodology—Pattern analysis; Pattern Recognition [I.5.5]: Implementation—Interactive systems;

1. Introduction

In recent years Augmented Reality (AR) technologies have allowed researchers to develop several applications in various areas as the medical one, engineering, military, cultural heritage, games, and many more. However, there are still some open issues related to the AR research field [FDB08]. In particular, these open issues concern the tracking, the interaction and the visualization. The selection of the most suitable technology to be used for developing an AR system depends on the purpose of the AR application and on the environments that could influence it.

The tracking has been largely investigated in AR community since several tracking solutions based on different approaches have been developed. AR applications have distinct precision's requirements, the metrological quality of tracking is an important feature to be considered during the development of an AR system. In particular, the wideness of the environment is a discriminating factor to the choice of a tracking solution. Most of the AR applications have been developed with tracking systems that well operate in a little workspace. Consequently, in an application addressed to wide environment, the metrological quality would decrease or even could be impossible to detect the position if it was developed with a common tracking solutions for narrow spaces.

With this work we have tried to overcome some of the limits of those technologies by using an innovative tracking approach. In particular, our objective was to develop a marker-based application that can adapt itself according to the camera's position in order to use it in a large environment. In this paper, we describe the implementation and the testing of our AR tracking approach. The proposed solution has been evaluated by performing some testing sessions, so as to demonstrate the potentiality and the feasibility of our approach.

2. Related Works

In simplest and most economic AR applications, fiducial markers coupled with Computer Vision (CV) algorithms are widely used. One of these, which has been used for this work, is ArToolKit [KB99]. However, even if this solution supplies a fairly good tracking for the AR visualization, its

DOI: 10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2010/153-158

precision is limited to a small range [WPT02] [DJR04]. More often users hold the markers of these applications in hand indeed, thereby the range results to be approximately half meter. If the marker were placed in a wider tracking space, its distance to the camera would be greater than a "marker-in-hand" situation and, consequently, the pose estimation would be not accurate or even difficult. Increasing the marker's size, the tracking quality could rise up. The user could work in a wider space, but it could be impossible for him to get near to the marker in order to visualize the digital object to a closer position. This is due because a big marker is not visible by a camera if their relative distance does not allow the camera to frame the entire pattern. A solution to overcome this issue is to place many different markers in a space, especially on the walls and on the ceiling, obtaining a wide tracking area. Unfortunately, this requires a careful preparation of the room. The exact position of each marker in the space has to be known; consequently, the user has to spend much time for setting up the environment.

The solution of use nested markers [TKO06] overcomes the difficulty to track large markers if the camera is too much close to them, but at the same time it introduces further problems. Indeed, for its intrinsic structure, a nested marker is a not robust and requires a precise calibration and a more complex detection.

Another good solution based on CV algorithms is the marker-less tracking technique. This recognizes elements of the real scene and can define a sort of 3D map of point features [CEMC03, KM07, Oku04]. In this way, it is possible to have a working application in unknown spaces. However, this kind of tracking technique cannot be used for wide environments. Moreover, a marker-less system needs to find features in the image to detect the camera's position, so it is not possible to use it in wide and neutral spaces.

In those applications where the accuracy and precision of the tracking is fundamental, sensor-based tracking systems as magnetic, acoustic, inertial, optical and/or mechanical sensors are used. Often CV algorithms and sensor-based tracking are also merged in order to further improve the tracking quality [FEBS07]. All these tracking solutions operate with prior knowledge about the user's environment. In fact, it is required to set some coordinate systems in the real environment before executing the AR application. For this reason, they are little versatile and need a lot of time to be set up and for the system calibration.

There are other works as [EHH04] and [WHLW09] that use the video projection in two different way in order to increase the working space of the Augmented Environment. Unfortunately, both of them have big limitations due to the complexity of the systems, and the impossibility of employment in large spaces since if the plane of projection is far, the projected image quality decreases considerably.

One of the most important features to be addressed when using applications based on CV and common camera is the illumination. As a matter of facts, in these ones the user has to control the lightning source in order to help the recognition of markers or space's features.

A good solution to solve the illumination condition, which has been used and developed in our work as well, is to visualize the marker on a display. [HLLK06] and [RWB02] represent markers on the displays of a mobile phones and PDAs. However, since the mentioned displays are little, the tracking accuracy is not high and the working area is narrow above all.

Starting from these existing solutions, we have employed a large display to visualize the marker. In common AR applications, the markers used are squared and the length of their sides is often less than 10cm. Visualizing it on a monitor, we can increase its dimensions until a full screen size, allowing us to have a better tracking further as well. Obviously, a big marker is not always advised because it cannot be detected when we are close to it with the camera. Since the marker is represented by means of a PC, it is possible to change its dimensions according to the position of the camera, obtaining the best dimension for each frame dynamically.

3. The system

Our proposed solution is grounded on a system consisting of a laptop and an Ultra-Mobile PC (UMPC) for visualizing a wide AR Environment. The entire application has been developed to have a continuous data's exchange between these devices. In particular, the data are used for the detection of the user's point of view in a fixed reference system and the control of the marker's size.

The user holds the UMPC in hand and he can see the Augmented Environment in its display if the marker is correctly visible by its built-in camera. The UMPC used is the Sony UX1XN, which has the following features: 4.5″ touch screen, two built-in cameras, front: 0.3 Megapixels and back: 1.3 Megapixels, Intel Core 2 Solo processor U2200 1.2*GHz*, Win Vista OS. The main AR application runs on the UMPC and uses the back built-in cameras of the UMPC for framing the real scene at 30*fps* while the touch screen is used to interact with the AR environment.

The fiducial marker, which has been used for the tracking, is shown on the laptop's monitor. The function of the laptop is also to maintain the communication with the UMPC, in order to know the distance between themselfs and consequently correct the marker's size. This communications are established through a wireless TCP/IP connection.

3.1. System Architecture

One of the most important aspects of an AR application is that the virtual part has to match, as much as possible, with the real world. In particular, it is expedient that there are not delays or a jerky visualization due to processes, which are

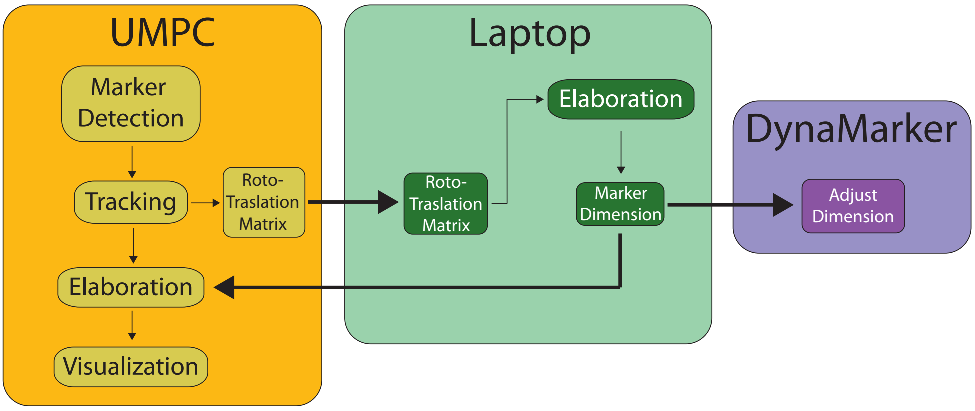


Figure 1: *Architecture of the system and the data flow between the devices.*

too slow to be in real time, or due to not-synchronized information. For these reasons, the continuous data flow between the UMPC and the PC has to be guaranteed.

The laptop plays the role of mediator of this application, as shown in Figure 1. In particular, the tasks carried out by the laptop are to elaborate the data, coming from UMPC, in order to change the marker's dimension, whereas the UMPC manages the marker tracking and the visualization of the AR environment.

The UMPC sends to the laptop the information related to the position and the orientation of the marker at every frame (30Hz) through a wireless connection with a TCP/IP protocol. The laptop receives the roto-translation matrix, calculates the distance between camera and marker, then modifies the sizes and notifies the UMPC about the variation of the zoom and its new value. The tracking system of the UMPC, which is informed about the change of size, can be always updated using the new dimensions, in order to constantly have a metrological correspondence.

3.2. The DynaMarker

The tracking of the marker from different distances is a well-known problem, because the precision of the tracking decreases with the increasing of the distance between camera and marker [WPT02] [DJR04].

Since the marker is visualized on the screen of the laptop, its dimensions can be changed with a modification of the zoom of its image on the display. For this reason, in order to provide a better tracking in a wider environment of workability, the dimensions of the marker have to change according to the relative distance between marker and camera. Since the size of the marker can change dynamically, we called this kind of marker DynaMarker.

In common applications, the AR tracking libraries are able to provide a good and stable detection of the marker's position with dimensions less than $10cm \times 10cm$. Usually the marker is placed in the hand of the user, so the distance between camera and marker is generally short. From these considerations, we have imposed that the size of the marker has to be equal to $8cm$ if it is placed at a distance of a meter from the camera. The marker changes following a linear trend, which is proportional to the distance. Consequently, the size s can be obtained by the following expression:

$$s[cm] = \frac{8[cm] \cdot d[m]}{1[m]} \quad (1)$$

where d is the distance from camera to marker.

4. Performances of the System

The DynaMarker is different to the common ones used in marker-based systems because it is displayed on a video. For this reason, its own visibility is less sensitive to the variation of the illumination and it is possible to modify the dimensions in order to make it better recognizable and traceable. To compare the different characteristics of the DynaMarker in relation to those proper of the common one, we have carried out some experiments.

4.1. Visibility Check

The first test aimed at checking the visibility of both markers through the camera at different intensities of light. Since the unique factor that had to weigh upon the experiment was the light intensity, the illumination of the environment has been controlled by means of professional lamps. In Figure 2 is shown the testing system.

The need of a controlled illumination is due to problems that could rise in environments where the light is direct. These problems are the reflections and the shades that can alter the experiments. In cases of direct illumination, we know

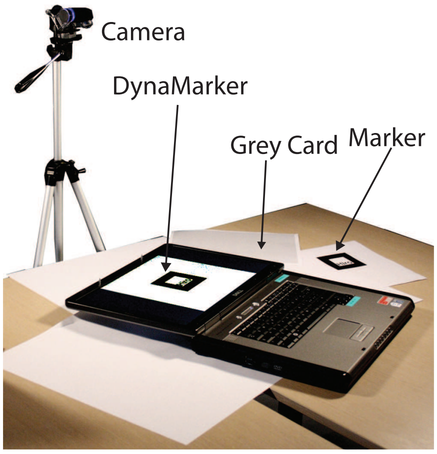


Figure 2: *The set-up for the recognition test.*

that reflections are common on an AR application or when a user watches a screen. The normal marker could create reflections that are caused by the quality of paper or by the ink used to print it. When we use the DynaMarker, instead, the reflections could be created to the incident light on the display. It is possible to overcome this issue by means of special papers and antiglare screens or a correct illumination. In our case, we chose to create a diffuse lighting for the test in order to avoid accidental shades as well.

To calculate the value of the light intensity we measure the value of every pixel in a gray scale that provides data of the image luminosity in a range from 0 to 255. This is useful, in our case, since during the first part of the marker recognition procedure there is a binarization of the image, which was set up to a threshold value of 100 (on the scale values from 0 to 255). Consequently, some details of the image, as the marker and its pattern, could disappear depending on their luminosity values on the gray scale. In order to evaluate if the scene was overexposed, underexposed or with a correct illumination, we have employed a neutral test card, with dimensions $20 \times 28cm$ and 18% of reflectance across the visible spectrum.

If the illumination is correct, the value of gray is 209 and it decreases by reducing the light. On the contrary, the gray card is visualized like a white one if the illumination is too much intense.

We have positioned the two markers in the diffuse-light environment one at time. Their position is fixed and we have placed them at a distance of 50*cm* from the camera, which is locked. The model of the camera used for the experiment is a Live! Cam Optia Pro by Creative.

By comparing the two markers, we have acquired an image of the gray card at every different set-up of light. Then, we have calculated the mean values of the markers white region, in order to evaluate how these values are higher than the threshold established before, and the probability of the marker recognition by the ARToolKit algorithm is detected.

As shown in Table 1, the common marker is slightly brighter than the DynaMarker at high levels of light intensity. This is probably due to the display, which is less sensitive to the reflecting light for constructive reasons, while the paper reflects the majority of the incident light. Actually the camera captures two different type of light from the two markers. In the case of normal marker it catches the light that is reflected by the paper, while in the second one the majority of light is produced by the laptop's display. Moreover, the contrast between black and white in the display is not enhanced as in the marker printed on paper. At high levels of luminosity, the common marker is slightly better recognizable; this is probably due to the pattern that is more contrasted and, consequently, well defined. For this reason, the threshold probably less erodes the details of the pattern, determining higher levels of probability of recognition, as shown in Table 2. On the contrary, the common marker is not usable in condition of low light, because there is not sufficient illumination, which is necessary to make the pattern recognizable. The marker's data cannot be taken out after threshold. In this condition, the DynaMarker is visible, since it is less sensitive to lighting conditions, so it is possible to recognize it and consequently detects its position.

Table 1: *Average values of the Grey Card, the white paper and the Display for the laptop at different light intensity. The scale of values is from 0 to 255.*

Grey Card	Paper	Display
25	29,61	109,87
75	90,54	114,85
125	151,58	143,97
175	202,59	197,70
225	251,97	233,70

Table 2: *Coefficient values of the marker and DynaMarker recognition at different intensity of light. The light intensity was measured by means of the value of the gray card (value scale from 0 to 255)*

Grey Card	Paper	Display
25	0	0,54
75	0	0,69
125	0,77	0,69
175	0,82	0,72
225	0,86	0,81

4.2. Tracking Check

The second test aimed at analyzing the performance increase during the tracking with the DynaMarker and in particular at validating the precision of the marker tracking by means of known simple movements of the camera in space while the marker is positioned in a precise point.

For this test, we have locked the camera on the end-effector of a robotic arm (Figure 3), in order to have a fixed

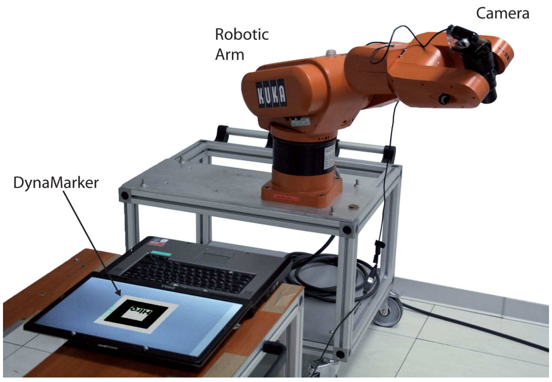


Figure 3: *The set-up for the tracking test.*

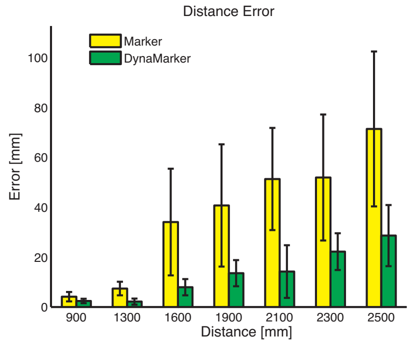


Figure 4: *Trend of the tracking error, in millimeters, at different values of distance.*

and stable support that allows the repeatability of the tests for obtaining a statistical result with a good sample.

We have carried out different tests by moving the robotic arm at different relative distances from the markers. The tests consist of simple straight displacements of the robotic harm that moves away the camera from the marker. The working distance of these tests ranges from 80*cm* to 2.5*m*.

We have positioned the marker in a precise point of the space that has been measured with reference to the Robot coordinate system, in order to relate all the measurement to a unique system. The values of the roto-translation matrix between camera and marker have been determined at every frame by the ArToolKit algorithm while the robotic arm was moving.

For every one of them, it has been found the gap between the position of the camera by the ARToolKit tracking and the position of the camera calculated by the data from the robotic arm. Figure 4 shows the trend of tracking error at increasing distances. The error values of the common marker grow considerably with the distance, until 2.5 meters, where the marker is not always traceable, since sometimes it is not recognized. The DynaMarker, on the contrary, does not already feel the effect of distance, because it changes its dimensions according to the distance in a linear way. After 2.3 meters, also the error with the DynaMarker raise up: this is due to the marker that achieves the maximum size to be completely visualized on the display. Since the marker is not able to further increase, the error tends to grow up.

Since the tests were based on simple displacements along straight lines, the angles between camera and marker should be constant. In order to verify the tracking quality, we checked this data as well. In Figure 5 it is possible to notice how the error in angles detection raise up for long distances, in particular the yaw angle in the common marker tracking.

5. Discussion

In section 4.1 we asserted that the DynaMarker is more visible than a common marker in environments with low lighting conditions. Even if the paper is slightly better in high-enlighten spaces, it is impossible to detect it if it is illuminated by weak light sources, while the DynaMarker allows us to detect itself.

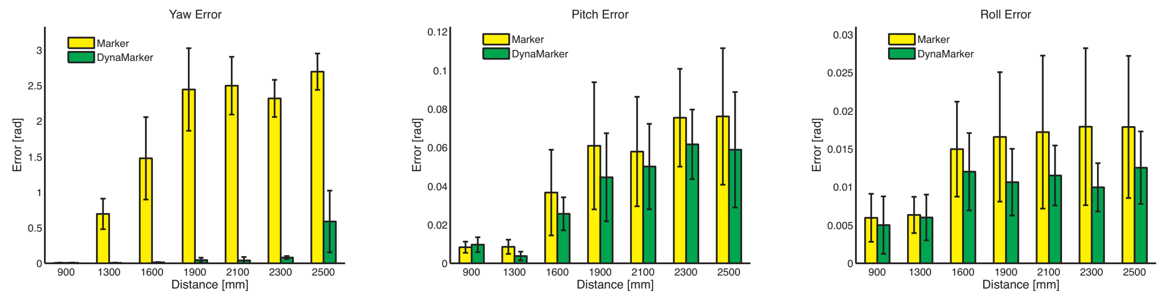
Thanks to the dynamic change of the marker's size, we improved the results of the pose estimation of ArToolKit. In particular, the system is able to detect the DynaMarker also at bigger distances in respect to a common marker.

The tests show us that, using DynaMarker, it is possible to calculate the position without a significant increase of tracking error up to 2.3*m*. Moreover, the calculated angles give us another important information about the better tracking quality obtained by use of DynaMaker. Actually they should be constant, but the high standard deviation, which was noticed in the tracking test with marker, points out that the system is not able to determine a stabile solution. The result of tracking with the common marker at long distances is a staggering visualization of the digital object. The user can see the Augmented Environment, but the virtual part is fluctuating.

To sum up, the DynaMarker supplies a better pose estimation as regards to the common marker until 2.3*m*. After this distance, which corresponds to the maximum marker dimension visualizable on the display, also the pose estimation of the DynaMarker starts to be less precise.

6. Conclusion

In this paper we have described the implementation and the validation of our tracking approach, which has been developed for being used in wide AR environments. The system is based on a marker-based tracking, where the marker is visualized on a monitor and it is possible to modify its dimensions dynamically and according to the distance from the camera. The marker, which we called DynaMarker, is



(a) (b) (c)

Figure 5: *Trend of the orientation error, in radians, at different values of distance.*

shown on a laptop's screen and it is used for the tracking by an UMPC that manages the AR environment and communicates with the laptop thought a wireless connection.

Thanks to this configuration, we have been able to provide a good tracking for the UMPC in an wider environment than a common marker-based system. In addition, the DynaMarker provides a tracking improving because in this way the observer can see objects near and far from the marker.

The proposed tracking technique was validated through different testing sessions by analyzing the main features of the system singularly. At first we validated the use of the marker showed on the screen of the laptop by comparing its performance with a classical printed marker. Subsequently, we tested the tracking system by analyzing its pose estimation improvement as regards the common one.

Globally the results of the whole testing sessions are encouraging, because it is possible to extend the tracking area of a marker-based application using a single marker displayed on a screen without loosing the metrological quality of the pose estimation. We are aware of several problems that have to be solved yet, but we claim that the foundation of our tracking approach will be certainly useful to further developments in the field of wide AR environments. At the moment, we are developing our approach mounting the DynaMarker on a mobile robot in order to enlarge the tracking area.

References

[CEMC03] COMPORT A. I., ÉRIC MARCHAND, CHAUMETTE F.: A real-time tracker for markerless augmented reality. In *Proc. of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality* (2003), IEEE Computer Society. 2

[DJR04] DANIEL F. A., JOACHIM B., RALF D.: Accuracy in optical tracking with fiducial markers: An accuracy function for artoolkit. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality* (2004), IEEE Computer Society. 2, 3

[EHH04] EHNES J., HIROTA K., HIROSE M.: Projected augmentation - augmented reality using rotatable video projectors. In *ISMAR '04: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 26–35. 2

[FDB08] FENG Z., 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 ISMAR.* (2008), pp. 193–202. 1

[FEBS07] FISCHER J., EICHLER M., BARTZ D., STRAER W.: A hybrid tracking method for surgical augmented reality. *Computers & Graphics 31*, 1 (2007), 39 – 52. 2

[HLLK06] HONG Y., LEE S., LEE Y., KIM S.: Mobile pointing and input system using active marker. *Mixed and Augmented Reality, IEEE / ACM International Symposium on 0* (2006), 237–238. 2

[KB99] KATO H., BILLINGHURST M.: Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *2nd International Workshop on Augmented Reality* (San Francisco, USA, October 1999). 1

[KM07] KLEIN G., MURRAY D.: Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)* (Nara, Japan, November 2007). 2

[Oku04] OKUMA T.: A natural featurebased 3d object tracking method for wearable augmented reality. In *Proc. of Advanced Motion Control (AMC'04)* (2004), pp. 451–456. 2

[RWB02] REGENBRECHT H., WAGNER M., BARATOFF G.: Magicmeeting: A collaborative tangible augmented reality system. *Virtual Reality 6*, 3 (2002), 151–166. 2

[TKO06] TATENO K., KITAHARA I., OHTA Y.: A nested marker for augmented reality. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches* (New York, NY, USA, 2006), ACM, p. 152. 2

[WHLW09] WENG D., HUANG Y., LIU Y., WANG Y.: Study on an indoor tracking system with infrared projected markers for large-area applications. In *VRCAI* (2009), pp. 239–245. 2

[WPT02] WAYNE P. M., PIEKARSKI W., THOMAS B. H.: Measuring artoolkit accuracy in long distance tracking experiments. In *ART02, 1st International Augmented Reality Toolkit Workshop* (Darmstadt, Germany, September 29 2002). 2, 3