

# Virtual Object Position Manipulation Using ARToolKit

Hoger Mahmud Hussien  
University of Human Development  
Department of Computer Science  
[hoger.mahmud@uhd.edu.iq](mailto:hoger.mahmud@uhd.edu.iq)

**Abstract**— In this paper the outcome of a project is presented that aims to modify and improve one of the most widely used Augmented Reality tools. Augmented reality (AR), is a fast growing area of virtual reality research.

Augmented Reality (AR) is a newly emerging technology by which user's view of the real world is augmented with additional information from a computer model. ARToolKit is one of the most widely used toolkits for Augmented Reality applications. The toolkit tracks optical markers and overlays virtual objects on the markers. In the current version of the toolkit the overlaid object is stationary or loops regardless of the optical target position, this means that the overlaid object cannot be animated or changed based on the movement of the optical target.

The aim is to improve the toolkit, therefore a design solution to modify it were designed and implement so that users can manipulate the position of the overlaid virtual object, through movements of the optical target. The design solution focuses on developing a mathematically based links between the position of the optical target and the overlaid virtual object.

To test the solution test cases were developed and the results show that the design solution is effective and the principal idea can be used to develop many applications in different sectors such as education and health.

**Keywords**— AR (Augmented Reality)

## I. INTRODUCTION

Over the past few decades, AR researchers have explored a wide variety of application domains for AR technology, most recently they have been focusing on the use of AR in education and entertainment settings [1] [2]. However the primitive state of AR technology and the lack of sophisticated multi-tasked AR tools have made it difficult for the technology to be widely available and used.

Tracking, also called Position and Orientation Tracking or Position Tracking and Mapping [3], is used in many kind of applications (e.g. robotic, VR/AR) where the orientation and the position of a real physical object is specified using the Cartesian coordinates  $x,y,z$ . ARToolkit is one the most widely used technology for tracking and processing optical target "Camera (optical) based tracking for augmented reality is nowadays frequently done using AR-Toolkit" [4]. ARToolKit is a library that scans video images for known optical

markers and computes the pose of the markers with respect to the camera [5].

The position and orientation of an optical target is determined by analyzing the projection of 2-dimensional images or by computing the sweep-beam angles of a given target [6]. In AR cameras are usually used as the optical sensors, video and optical technologies are the two main choices to combine Real and virtual objects [7]. For the purpose of this project optical based technology (camera) is used to do the tracking. In the current ARToolKit software the overlaid object is stationary or loops regardless of the camera or optical target position, this means that the overlade object cannot be animated or changed based on the optical target movement. Animation can only be achieved when it is preset in the graphic file. In this project a modification the ARToolKit is propose and tested so that the position of the virtual object can be manipulated by the user, through movement of the optical target. The ARToolKit modifications enable users to move virtual objects produced by the toolkit, in the  $(x, y, z)$  and  $(-x, -y, -z)$  directions, through the movement of the optical marker. The findings of this project will contribute towards improving one of the main tools that deliver AR technology and in result improve the usability of AR technology. The design modification is tested using a number of test cases and the results show that it is an effective approach to achieve virtual object manipulation in ARToolKit.

In recent years researchers of Augmented Reality has increased despite the fact that it is a newly emerging technology. The wide range applicability of the technology has made it attractive to many different sectors such media companies and mobile device manufacture. Not only expanding areas of applications deserve attention, but more important is refining the tools that deliver the technology, so that the tools' performance reflects the true value of the technology. ARToolKit nowadays is one of the most widely used AR tools, improving the performance and flexibility of this particular toolkit will have a positive impact on AR technology as a whole, and also reduces the usability limitations of the toolkit.

II. RELATED WORK

The creation of an interactive virtual world involves two major tasks; the first one is identifying and modelling the virtual object and the other is describing the interaction and behaviours of the virtual object. The approach that is normally used for modelling 3D geometries is to use 3D creator tools which are widely available; this has paved the way for many different attempts to model virtual world geometry within virtual world environments see [8] [9] [10] [11]. As for the behaviours of the virtual objects within virtual environments, the attempts to define them have been few. There are a number of researches to define and create the virtual environment such as [12] that suggests a conceptual virtual reality system called “Lingua Graphica” that uses programming languages to create the environment but it was never implemented.

III. MODIFICATION DESIGN

ARToolKit is a type of interface known as “tangible AR interfaces” which combine virtual and physical objects and it let users interact with virtual objects by manipulating corresponding physical objects [17]. From this we can say that the two main pillars of tangible AR Interface are:

1. Virtual objects
2. Corresponding physical objects

The virtual object is what the users aim to compute and interact with which can be displayed in various forms e.g. 2D images, texts and 3D geometries. The physical objects are sensed and used as input to visualise overlaid corresponding virtual object. 4

Since users interact with virtual objects via physical objects that is used as input by ARToolKit there must be logical connections between both pillars of the interface. The logical connection is the proposed solution which is mathematically created to enable users manipulate virtual objects. From this it’s clear that there are three types of components that need to be combined to build an interactive AR model, the physical object, the virtual object and the logical connection between them. Each component has a set of properties that represent the state of that component. Each of these properties differ between the component types, for instance the position property of a physical object has a vector type value that represents a three dimensional coordinates X, Y, Z known as Cartesian Coordinates see figure 1. These three values can be read but can not be modified freely, since it is determined by the position of the physical object, all the values collected in transformation matrix.

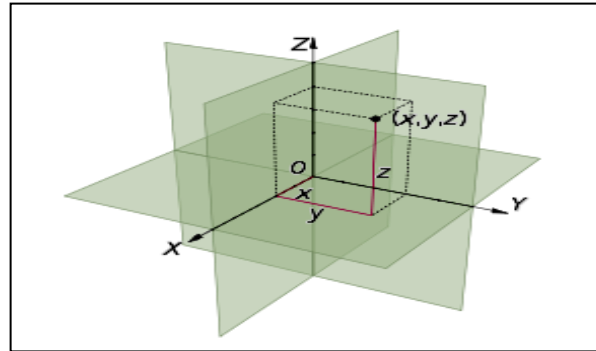


Fig.1 (Cartesian Coordinates) [18]

In the current version of ARToolKit the physical object (optical target) position determine the values of the transformation matrix computed by the tool. The values are used to determine the position of the virtual object and then draw it, this corresponds to step 3 and 4 of the activity diagram, see figure 2.

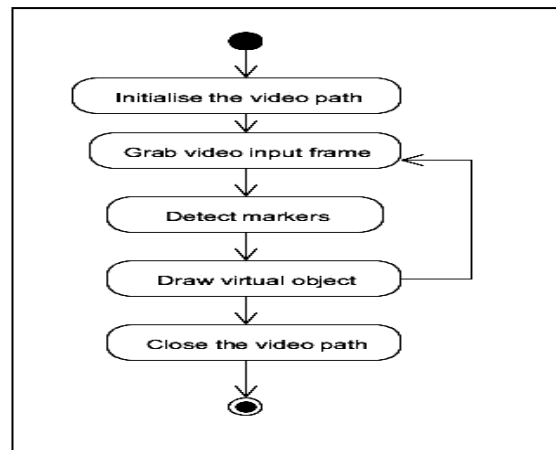


Fig.2 (Current ARToolKit Activity Diagram)

If a logical link established that is able to read and update the transformation matrix, users can manipulate virtual objects by changing the values of the physical transformation matrix relative to the camera (optical sensor). After the establishment of the logical link, to achieve the desired manipulation, ARToolKit will perform different activities to redraw the position of the virtual object every time there is a change in the transformation values. Because ARToolKit uses OpenGL functions [19] to set the virtual camera position and render virtual objects, OpenGL function such as glTranslate (x, y, z), will be used to provide the link needed to change the position of virtual objects see figure 3.

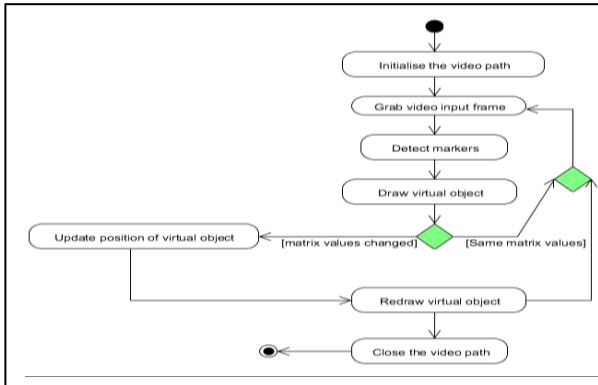


Fig.3 (Modified ARToolkit Activity Diagram)

IV. IMPLEMENTATION OVERVIEW

To manipulate the virtual object through the movement of the optical marker, mathematically based links (operations) is established between the virtual object and the matrix values. The 3x4 matrix values contain the optical marker position relative to the optical sensor (camera). These mathematical operations are placed inside the code in a way that they are executed before the virtual objects are drawn. The mathematical operations, manipulate the matrix output values, and store them in different float variables that are declared in the program for this purpose. These values are then used to change the x, y, z values of the OpenGL functions that are used to manipulate the position of the virtual objects.

V. CASE DEVELOPMENT AND TESTING

To show the efficiency and capability of the modified toolkit, an example was constructed consisting of a ball created in the Maya 3D graphic development software, and then a globe texture was assigned to it, to produce a globe like model see figure 4. The model was then imported and placed in the appropriate folder of the ARToolkit and a random pattern (optical target) was assigned to it. For the modified toolkit to be viewed as successful the test cases in table 1 should produce the described expected results during testing. Table 1 also shows the actual result for each of the test cases.

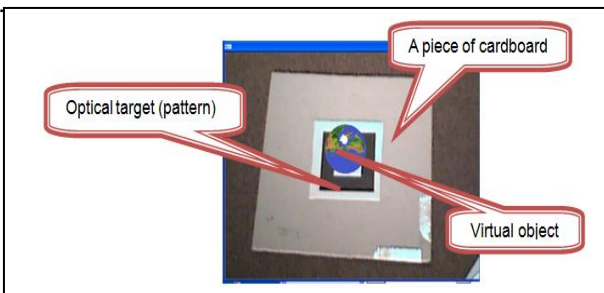










Fig.4 (Case developed)


VI. TEST CASES AND RESULTS

A number of test cases have been designed to test the modification and the result is shown in table 1.

TABLE 1

No	Test Cases	Expected results	Actual results
1	Tilt the physical object (optical target) in the +X direction.	As the physical object is being tilted, the virtual object (globe) should move in the +X direction.	
2	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe).	
3	Tilt the physical object (optical target) in the -X direction	As the physical object is being tilted, the virtual object (globe) should move in the -X direction.	
4	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe).	
5	Tilt the physical object (optical target) in the +Y direction.	As the physical object is being tilted, the virtual object ( globe) should move in the +Y direction	
6	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe)	
7	Tilt the physical object (optical target) in the -Y direction.	As the physical object is being tilted ,the virtual object ( globe) should move in the -Y direction	

8	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe)	
9	Tilt the physical object (optical target) diagonally, in the +X, +Y combination direction	As the physical object is being tilted, the virtual object (globe) should move in the +X, +Y direction.	
10	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe).	
11	Tilt the physical object (optical target) diagonally, in the +X, -Y combination direction.	As the physical object is being tilted ,the virtual object ( globe) should move in the +X,-Y direction	
12	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe).	
13	Tilt the physical object (optical target) diagonally, in the -X, -Y combination direction	As the physical object is being tilted, the virtual object (globe) should move in the -X, -Y direction.	
14	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe)	
15	Tilt the physical object (optical target) diagonally, in the -X, +Y combination direction.	As the physical object is being tilted ,the virtual object ( globe) should move in the -X +Y direction	

16	Move the physical object (optical target) back to its original position.	As the physical object (optical target) is being moved back to its original position so should the virtual object (globe).	
----	--	--	---

## VII. CONCLUSION

Augmented Reality is the new technology of the future; however one of the things that have limited the commercial use of AR applications is the lack of sophisticated tools that can be used to develop complex applications. Improving ARToolKit which is one of the tools that are widely used for AR application developments was the focus of this project.

The aim of the project was to enable users directly manipulate the position of a virtual object through the movement of the optical target, something users of the toolkit could not do in the current version. To achieve this, the toolkit was extensively investigated, to identify limitations as well as the way that the toolkit works. A solution was then designed to modify the toolkit, in order to enable users directly manipulate virtual objects. The idea behind the design was to use an aspect of the toolkit that could be changed through direct users' action, and use the changes to manipulate the virtual objects. After investigating the source code, it was decided that the camera matrix which represents the position of the optical target is something that can be used for this purpose, and through this the virtual object can directly be manipulated.

After the design stage of the project, it was implemented and a case study was then developed to test and evaluate the modified toolkit. The test results show that the modification has enabled users to manipulate the position of the virtual object. Users are able to move virtual objects in all directions freely as well as stopping the virtual object in a position of their choice. The principle idea behind the modification is something with great potentials for future ARToolKit applications. It has paved the way for other developers to improve on the principle idea behind the modification.

## VIII. REFERENCES

- [1] Kangdon Lee, "Augmented Reality in Education and Training," TechTrends, vol. Volume 56, , no. Number 2, April 2012.
- [2] Manuel Contero, David C. Pérez-López and Mario Ortega Mariano Alcañiz, "Augmented Reality Technology for Education," NewtechOpen, 2010.
- [3] [Online].  
[http://www.ercim.eu/publication/Ercim\\_News/enw53/foursa.html](http://www.ercim.eu/publication/Ercim_News/enw53/foursa.html)
- [4] [Online]. <http://www.rm.is.ritsumeai.ac.jp/MR-PreVizProject/workshop.htm>.
- [5] Mark Billingham, Ivan Poupyrev Hirokazu Katoa. (2013, September) tinmith.net. [Online].  
<http://www.tinmith.net/lca2004/ARToolkit/ARToolkit2.33doc.pdf>

- [6] Ribo Miguel. (2001, September) [Online]. [http://www.augmentedreality.pl/publications/TR\\_VRVis\\_2001\\_025\\_Fu1l.pdf](http://www.augmentedreality.pl/publications/TR_VRVis_2001_025_Fu1l.pdf)
- [7] David Sickinger. (2004, February) cs.auckland.ac.nz. [Online]. [http://www.cs.auckland.ac.nz/~burkhard/Reports/2003\\_SS\\_DavidSickinger.pdf](http://www.cs.auckland.ac.nz/~burkhard/Reports/2003_SS_DavidSickinger.pdf)
- [8] A. Davidson, S. Hench and T. M.Olano J. Butterworth, "3DM: A Three Dimensional Modeler Using a Head-Mounted Display," in Proceedings of Symposium on Interactive 3D Graphics, Cambridge, Massachusetts, 1992, pp. 135-138.
- [9] J.Liang and M. Green, "JDCAD: A Highly Interactive 3D Modeling System," in Computer & Graphics, 1994, pp. 499-506.
- [10] M. R. Mine, "ISSAC: A Meta-CAD System for Virtual Environments," in Computer-Aided Design, 1997, pp. 547-553.
- [11] G. Wesche and H. Seidel, "'FreeDrawer – A Free Form Sketching System on the Responsive Workbench," in ACM Symposium on Virtual Reality Software and Technology (VRST 2001), 2001, pp. 167-174.
- [12] R. Stiles and M. Pontecorvo, "Lingua Graphica: A Visual Language for Virtual Environments," in Proceedings of IEEE Workshop on Visual Languages, 1992, pp. 225-227.
- [13] A. Steed and M. Slater, "Dataflow Representation for Defining Behaviours within Virtual Environments," in Proceedings of Virtual Reality Annual International Symposium, 1996, pp. 163-167.
- [14] G. J. Kim and C. M. Park G. A. Lee, "Modeling Virtual Object Behavior within Virtual Environment," in Proceedings of ACM Symposium on Virtual Reality Software and Technology (VRST 2002), 2002, pp. 41-48.
- [15] D. S. Tan, M. Billinghurst, H. Kato, H. Regenbrecht and N. Tetsutani I. Poupyrev, "Developing a Generic Augmented Reality Interface," IEEE Computer, vol. 35, no. 3, pp. 44-50, March 2002.
- [16] M. Billinghurst, I. Poupyrev, K. Imamoto and K. Tachibana H. Kato, "Virtual Object Manipulation on a Table-Top AR Environment," in Proceedings of the International Symposium on Augmented Reality (ISAR 2000), Munich, German, 2000, pp. 111-119.
- [17] Hirokazu Kato, Ivan Poupyrev Mark Billinghurst. (2002) hitl.washington.edu. [Online]. <http://www.hitl.washington.edu/publications/r-2002-61/r-2002-61.pdf>
- [18] [Online]. [http://en.wikipedia.org/wiki/Cartesian\\_coordinate\\_system](http://en.wikipedia.org/wiki/Cartesian_coordinate_system)
- [19] (2013, October) OpenGL. [Online]. <http://www.opengl.org/>