2009 First International Workshop on Education Technology and Computer Science

# Collabrative Education UI in Augmented Reality

## From Remote to Local

Yuan Yao, Di Wu
Rapid Manufacture Engineering Center
Shanghai University
Shanghai, China
(Yaoyuan, Wudy)@shu.edu.cn

Yu Liu
College of Software
Dalian University of Technology
Dalian, China
psodream@126.com

***Abstract*—The goal of this paper is to provide a easy way for manipulating 3D graphical object in remote education system. We describe a remote collaborative framework running in an augmented reality environment. It provides a nature operation interface for manipulating graphics object in the web browser or applications from different locations. A Jabber like protocol is designed to translate geometry information and transformation status which reduce the total amount of data transmission. We will discuss the structure of this method, and demonstrate its usability with practical applications.**

***Keywords-remote collaboration; augmented reality; ARSGF***

## I. INTRODUCTION

Internet base remote education is becoming an important part of modern education, and brings a lot convenience for people. However it still depended on simple communication methods, such as web pages, broadcast videos and instant message tools. They cannot help us realize an interactive education process.

Developing a remote collaborative system is a tedious work. The soft engineering must reconstruct different experiment contexts and processes in an entire virtual scene. Flash, WPF are the most popular techniques to build similar systems. Due to the complexity of copy real word to virtual, the development circle of these projects is not easy to be controlled. In addition the operation interface required the users to spend certain time to learn.

Augmented reality(AR) technologies create an environment where virtual objects are superimposed on a predominantly real world scene. The main function of AR, enabling users to see virtual objects and a real world scene together, has attracted the attention of researchers and developers in several fields. Many physical tools are designed which make the virtual object in the real environment can be touched or even be controlled using a way that with the people's habits. In this regard, AR based education and demonstration system is more attracting.

In this paper we will discuss the method for building AR based remote collaboration system. First, an AR application framework is presented. Follow this we design a jabber like protocol to encapsulate geometry information and operation message from different end users. Finally, we will demonstrate a remote multi-user education application to show how the framework and protocol simply the design process.

## II. RELATED WORKS

Many toolkits, such as ARToolkit[1], ARtags[2], ARToolkitPlus[3] and metaio[4], provide low-level functions for building AR applications. While some extensions of these approach, such as OSGAR[5]and DART[6]were proposed to give more high-level tools for rapid creating AR applications in the past years.

Haller et al.[7]firstly discussed building AR application with a Scene Node Library. They integrated ARToolKit with openSG or openSceneGraph to realize a culling effect between real object and virtual object. OSGAR is also based on openSG to correct the registration errors from tracking devices. Piekarski et al.[8] developed Tinmith-evo5 to support mobile AR games. Because all the architectures began to consider of the scene graph management, they can provide a more simple, familiar and intuitive environment for application developers.

DART is built on top of Macromedia Director, a widely used multimedia development environment. Thus it supports a rapid transition from storyboards to working experience using script and the experiential part of a design can be tested early and often. Barakonyi et al.[9] presented an AR based video conference system which share scene graph information between two remote users. An active-passive model is used to refresh operation status in both sides. Grasset et al.[10] discussed the tangible, transitional and collaborative user interfaces in their OSGARToolKit framework. These architectures begin to focus on the remote collaboration operation in AR applications.

Collaborative Augmented Reality involves graphics and image processing, human computer interaction, Computer-supported collaborative work and other disciplines and fields, is a typical cross-emerging field of research. Different users have different perspectives, expressions and interactions when they enter into the mixed virtual-real environment. It requires collaborative augmented reality system to provide multi-faceted environment. In general, users work together in a certain moment, one can only describe a certain sequence of video direction and scope of the real environment, which makes video sequences based on single video sequence is difficult to satisfy users to work together. There must be multi-

978-0-7695-3557-9/09 $25.00 © 2009 IEEE
DOI 10.1109/ETCS.2009.409

video sequence work together to ensure that all users in the augmented reality environment obtain the same understanding and perception of the space, that is, to ensure that the AR scene is consistency between different users. Thus, establishment of unanimous AR scene is becoming the focus of current research.

## III. THE BASIC FRAMEWORK

A high-level scene graph based toolkit named ARSGF(Augmented Reality Scene Graph Framework) is presented first, which is a software framework for rapid developing augmented reality applications. It uses scene graph to manage content loading, graphics rendering, data tracking and interaction. A set of encapsulated classes are used to deal with the base functions in AR applications. Figure 1 shows the main components in ARSGF toolkit. There are mainly five components in this toolkit. The function of each part is listed as follows:

- Scene manager: holding the whole information of each part. It supports communication among different nodes and manages the rendering process. It is the kernel of the framework.
- Tracking manager: getting tracking data from vision tracking process. In ARGSF we use ARToolkit to get the registration information. Tracking manager can analyze the real illumination information from ARTookit markers. This information can help scene graph manager create virtual lights to simulate the real lights and get a consistent illumination effect.
- Graphics node: holding the graphics information. Developers can flexibly derive from these basic nodes and construct special one.
- Collaboration manager: supporting communication with other scene graph managers through the net interface. It can also convert graphics node action to VRML messages, thus we can control VRML action in an Internet browser through a tangible interface.
- Content loader: it is a group tools that support loading different types of media, such as 3D studio max, VRML, and many type of videos and convert them to ARSGF internal format.

There are mainly three basic types of node in ARSGF toolkit: status node, static node and tracking node. Status node maintains the status information in the scene graph. It is always not displayed. Static node has constant position in the camera view space. It is used to present the information that has a constant position, such as status text and simulation windows. The tracking node is the most important graphics node. It can get position from tracking manager and load graphics content from content loader. Developers can extend this node to satisfy their own requirements.

ARSGF toolkit provides four types of sub node derived from tracking node. They are tool node, light node, real object node and virtual object node. Developers can construct new interaction tools based on tool node. The users can specify the XML based configuration file to construct the tools rather than designing on coding level. Figure 3 shows the different tools designed on tool node.

The light node is a special node in ARSGF. It can create virtual light with default value. Then it gets tracking data from tracking manager and adjusts its position and intension during the run time.

Real object node is often not displayed. It holds the real object's geometry information for object culling and shadow casting. The virtual object node provides basic function for virtual graphics registration and display.

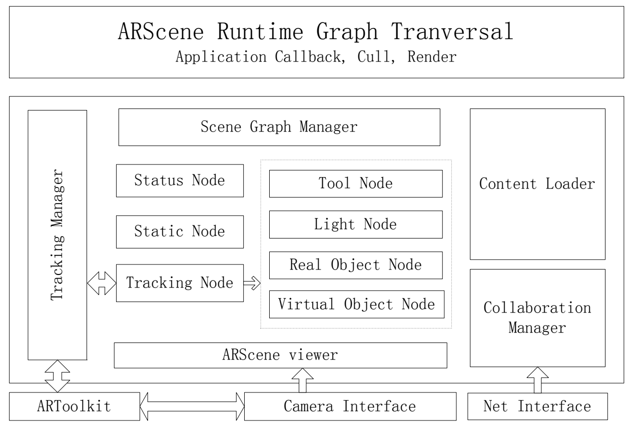


Figure 1. ARSGF framework architecture

ARGSF provides a simple way of rapid prototyping AR applications. These supports for developer are list as follows in three aspects:

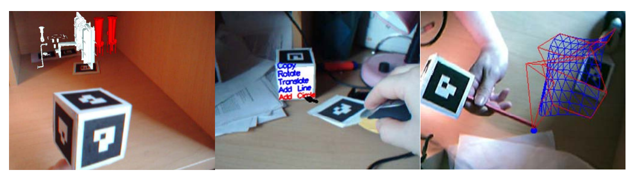

Figure 2. Interaction Tools

### A. Interaction Tools

Four types of interaction tools are provided in ARSGF toolkit as tool nodes. These interaction tools (e.g. select tool, mouse tool and grasp tools) are easy to be added into application. Developers can simply define these types of tools and add them to the scene graph. The tracking data from tools is achieved in scene graph by name. New properties can be added easily to these tools for special requirements.

### B. Content

The content authoring is the most important part for rapid AR application prototype. In addition to support content loading from multiple media sources, our framework also supports building complex graphics structure in a more controllable way. Developers can write their own graphics node with draw function and call back function. Thus support more flexible construction method.

### C. Collaboration

AR application supports collaboration in the local space intuitively. Our framework supports sending/receiving scene graph status with an XML based protocol. In addition to share virtual object information among remote AR applications, it also sup-ports manipulating VRML object in the Internet browser through AR application.

## IV. THE COLLABRATION LAYER

### A. Analysis

An AR based collaboration system including different operators from several independent workstations. Each operator changes the share scenes and affects other participants in the interactive process. In ARSGF, such a change in the share scene graph includes the following three aspects:

#### 1) Process

ARSGF cooperative mode is created by the master process. The other processes can enter or quit the collaboration at any time. This operation could change the sharing state of participate in the work process.

#### 2) Interactive event

The events generated by operators, such as mouse event, keyboard event, gestures, voice or menu command, are explained by the AR system to change the content of the shared scene.

#### 3) Transformation

Position changing caused by marker moving is the most common transformation in the shared scene. The marker moving is a continuous event and it is necessary to set a fixed frequency synchronous

### B. XSGSTP

Collaborative AR application requires the system maintain the consistency of the shared scene. Event inspired by the above three aspects will produce enormous amount of data that should be shared in collaborative work space. In order to achieve real-time synchronization of the scene, an XML based scene graph state transmission protocol(XSGSTP) is used in our framework. Which encapsulate different event type, scene state and transformation matrix in an incomplete description of the XML string. In XSGSTP the information packages are written into the semantic unit between <Stream> and </ Stream> tag, using Client-Server ways to communicate. This method of delivery data is very similar to jabber. XSGSTP includes 3 types of information unit: 1) the coordinates unit, 2) the event unit and query unit.

```
< ML >
    <Marker visible=”true” id=”1”>
    <Matrix value=”2.0 23.0 5.0 0.0  1.0 4.5 6.0 0.0
5.0 0.0 1.0 0.0 2.3 5.2 4.5 1.0”/>
</Marker>
```

The coordinate unit is used to delivery registration changing event among each of the process. This kind of package requires a certain refreshing frequency in order to hold a consistent of the scene graph. The coordinate unit is encapsulated into <ML> </ ML> tags, which contains a set of positioning values that marked by <Marker></Marker> tag. The above item shows the format of the coordinate unit, which is consist of multiple marker registration information.

Event unit use multicast method to transfer all those client processes involved in the collaboration. It uses the pair of <EL> and </ EL> tags to send a list of events to all the AR clients. Thus the events can be the synchronized between different remote processes. There are three types of event unit in ARSGF: 1)hardware input event, 2)menu event and 3) fiducial mark visible event. All above event types are defined by EventType tag. Therefore, the practicable event type can be extended by the requirement of corresponding application. The following example gives the left button down event triggered by a mouse pen tools.

```
<EL>
  <Event  type=”mouse” target=”node1”
                          status=”LeftButtonDown” >
     <Marker visible=”true” id=”1”>
     <Matrix value=”1.0 63.0 5.0 0.0 1.0 4.5 6.0 0.0
                         5.0 0.0 1.0 0.0 2.3 5.2
                         4.5 1.0”/>
   </marker>
 </Event> …
</EL>
```

Inquiry unit use the request-response mechanism to share information among the remote AR clients. Its message body is packed into <IQ> </ IQ> tags. Inquiry unit can be used in many types of context. For example, the AR collaboration process allows new collaborators registered in the same shared scene graph. Because of XSGSTP is not a graphics transmission protocol, the new participant will sent a request to the service process to obtain a complete list of state information. IQ Metadata can contain multiple sets of corresponding types of tags and request a list of information and event information list, which can be used to reconstruct a shared scene graph.

```
  <IQ type="request" targetUser="user1" object="12"
objectType="node" />
  <IQ type=“response” targetUser="user2" object=""
objectType="node" >
     <ML >  …  </ML>
     <EL”>   <Event type=”mouse” target=”AR Menu
” status=”create”/>   …
     </EL>
  </IQ>
```

## V. THE APPLICATION

In order to demonstrate the usability of the framework we redesign the Augmented Chemistry(AC)[11] system based on the ARSGF architecture. AC is an application that utilizes a tangible user interface (TUI) for organic chemistry education.

This system is a successful AR educational project with wonderful AR effect. It is developed by Morten Fjeld et al. We recoded the base functions of this system and added two new collaborative properties in it.

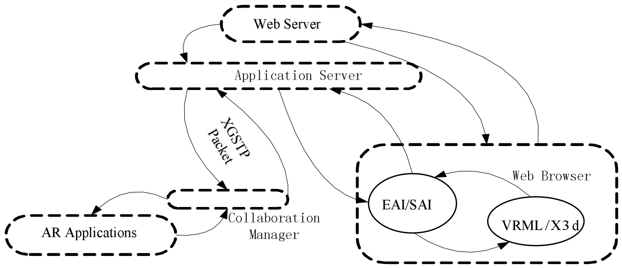


Figure 3. The AR-AR and AR-Web collaboration mode

Figure 3 shows the collaboration mode in our demonstration. In this application two type of client can be used together. One is the pure AR application and the other application is running in a VRML browser that embeds in a web browser.

### A. AR-AR remote collaboration.

The AR-AR collaboration mode runs on the following steps. Firstly, An AR application is start, which creates a master process. Then the second AR application begin running with a client mode. It connects to the application server created by the first application and send inquiry unit. The received information is used to initiate the scene graph. Figure 3 shows the scenery: first user creates one ball-and-stick model and the second user register in and creates the others.

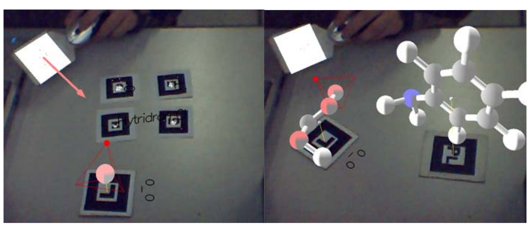

Figure 4. AR-AR collaboration demo

### B. AR-Web collaboration

Compared to AR application the web 3D applications do not need any hardware device. Traditionally they are developed by VRML with EAI interface or direct by java3D. But it is not easy to control and interact with. AR-Web collaboration can give a more nature way to manipulate the 3D object in web browsers. We provide the second collaboration mode to support this operation.

The AR-Web interaction also uses XSGSTP packet share scene graph information between AR systems and Web systems to exchange data. We use VRML developed the molecular assembly scene, which includes only two atomic VRML prototype and one molecular assembly platforms. The application uses the Java interface to control the scene and exchange XSGSTP packet with application server. Compared to IEEE DIS protocol XSGSTP uses quaternion to transmit rotation operation thus it not need to re-calculate the coordinate between AR scene and VRML.

Figure 5. AR based web 3D controller

## VI. CONCLUSION

This paper introduces a new high-level framework for developing remote collaborative AR application. It support AR-AR and AR-Web collaboration mode which can help develops build. The software framework is still in development stage. We are hoping to improve the stability of the fiducial mark and add more XSGSTP unit to test its usability.

## ACKNOWLEDGMENT

This system was supported by the national natural science foundation of China(60703029) and Shanghai outstanding young teachers fund. We thank Benedikt for providing the wonderful source code of the chemical AR project. And we also thank our colleagues for their valuable comments and constructive criticism on the system.

## REFERENCES

[1] Kato, H., P. Lamb, ARToolkit. http://artoolkit.sourceforge.net 2000.

[2] M. Fiala, Comparing ARTag and ARToolkit Plus fiducial marker systems, Proc. IEEE International Workshop on Haptic Audio Visual Environments and their Applications, 2005, pp. 148-153.

[3] W. Daniel and S. Dieter, ARToolKitPlus for Pose Tracking on Mobile Devices, Proceedings of 12th Computer Vision Winter Workshop (CVWW'07), 2007

[4] metaio, Unifeye software platform, http://www.metaio.com/, 2008

[5] E.M. Coelho, et al., OSGAR: A Scene Graph with Uncertain Transformations, Proc. Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality , 2004, pp. 6-15.

[6] B. MacIntyre, et al., DART: A toolkit for rapid design exploration of augmented reality experiences, Proc. Proceedings of the 17th annual ACM symposium on User interface software and technology, ACM New York, NY, USA , 2005, pp. 197-206.

[7] M. Haller, et al., Combining ARToolKit with Scene Graph Libraries, IEEE International Workshop on Haptic Audio Visual Environments and their Applications, 2002, pp. 148-153

[8] W. Piekarski and B.H. Thomas, An Object-Oriented Software Architecture for 3D Mixed Reality Applications, in Symposium on Mixed and Augmented Reality. 2003. IEEE Computer Society Washington, DC, USA , 2003, pp. 247.

[9] I. Barakonyi, et al., Remote collaboration using Augmented Reality Videoconferencing, Proc. Proceedings of Graphics Interface 2004, Canadian Human-Computer Communications Society School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada , 2004, pp. 89-96.

[10] R.L. Grasset, et al., OSGARToolKit: tangible + transitional 3D collaborative mixed reality framework, ACM International Conference Proceeding Series, vol. 157, 2005, pp. 257-258.

[11] M. Fjeld, et al., Tangible user interface for chemistry education, Proc. Conference on Human Factors in Computing Systems , ACM New York, NY, USA , 2007, pp. 805-808.