Satellite Visualization Tool Based On the ArcGIS Engine and OpenGL

Wu Hao[1], Li Ziyang[2], Hu Jian[3], Tang Lingli[4], Li Chuanrong[5]
Academy of Opto-electronics, Chinese Academy of Sciences, Beijing, China, 100094

The satellite resource visualization tool provides a large and clear view about the moving and working status of the satellite, it is an important part of the satellite mission operation system. ArcGIS Engine is a collection of GIS components that can be embedded, allow the user to add 2D and 3D world map to the applications. The way of using ArcGIS Engine for satellite resource visualization was discussed in this article, and since it doesn't provide enough methods for visualizing details, a way of extending ArcGIS Engine by OpenGL is also be discussed. The architecture of the visualization tool built with ArcGIS Engine and OpenGL was presented, which was used in the HJ-1A/B, SJ-9, ZY-3, ZY-01-C ground systems.

I. Introduction

Satellite visualization tool is quit a common software in the satellite ground system, from the visualization scene in the command hall, everyone could know the working status of the status. The satellite visualization tool always contains many functions, such as satellite track prediction, visualization, satellite motion control, and so on. Of these functions, the most important part is the visualization, it determined the performance and effects of the tool. For the earth resource satellite, the visualization part has many special needs:

- The need to render 2D and 3D scene at the same time.
- The need of loading different types of map file.
- The need of supporting custom graphics elements of different shapes.
- The need of fast rendering speed.

During these needs, rendering 2D and 3D scene at the same time is the base need, the 2D scene provides a full extent view of the world, and the 3D scene provides a more real view of the earth, all the visualization graphics are drawn on them. Map files are loaded for geography information, when the resource satellite was observing, we should know where it is. Custom drawing is the most important part, for the visualization needs many different shapes of geometry to render the scene. Also the rendering speed is for the effects, satellite are moving very fast, and everything changes all the time, so low rendering speed may reduce the credibility of the visualization.

II. ArcGIS Engine

ArcGIS Engine is a collection of GIS components and developer resources that can be embedded, allowing you to add dynamic mapping and GIS capabilities to existing applications or build new custom mapping applications.[1]

Developers use ArcGIS Engine to deploy GIS data, maps, and geoprocessing scripts in desktop or mobile applications using application programming interfaces (APIs) for COM, .NET, Java, and C++. ArcGIS Engine provides a set of visual controls that allow developers to easily deploy applications, of these controls, the mapcontrol provides a 2D map view, and the globecontrol provides a 3D globe view, which can satisfy the need of visualizing 2D and 3D world.

The ArcGIS Engine have its own way of rendering geometries, you must add a new layer, build a geometry element, set its displaying symbol, and then add it to the new layer, after refreshing, the element will appear. This way can rendering most types of geometry, but also has its disadvantages. The first one is low speed, especially

[1] Assistant Researcher, Academy of Opto-Electronics, Chinese Academy of Science. wuhao@aoe.ac.cn.
[2] Associate Researcher, Academy of Opto-Electronics, Chinese Academy of Science. zyli@aoe.ac.cn.
[3] Research Fellow, Academy of Opto-Electronics, Chinese Academy of Science. jhu@aoe.ac.cn.
[4] Research Fellow, Academy of Opto-Electronics, Chinese Academy of Science. lltang@aoe.ac.cn.
[5] Research Fellow, Academy of Opto-Electronics, Chinese Academy of Science. crli@aoe.ac.cn.

when many elements are added; the second is difficult way of maintaining, for example when updating an element, a query operation is always needed; the third is that not all the types of geometry are supported, some complex geometries can not be built dynamically.

In order to solve these problems, the dynamic display mode is added since the version 9.3, it provides a mechanism for synchronized refreshing that allows moving objects, such as maps and graphics, to be displayed with fast refresh-rate technology. The dynamic display mode can be implemented in two ways, the one is in the dynamic layer, and the other is by OpenGL. The dynamic layer way separated the displaying data in to different layers, thus make it difficult to do operations on satellites. While the OpenGL way can provide bottom operations that can control the satellite model directly, so it is better for satellite visualization.

III. OpenGL embedding of ArcGIS Engine

OpenGL (Open Graphics Library) is a standard specification defining a cross-language, cross-platform API for writing applications that produce 2D and 3D computer graphics. The interface consists of over 250 different function calls which can be used to draw complex three-dimensional scenes from simple primitives. It provides a fast, high quality and cross platform solution for 2D and 3D visualizations.[2]

The visualization part of the ArcGIS Engine is developed with OpenGL, so it is quit compatible for OpenGL embedding. In the satellite visualization tool, the MapControl and the GlobeControl are used to render the 2D and 3D scenes, and the OpenGL codes should be embedded at these two controls. For the mapcontrol, the OpenGL can be embedded in its AfterDynamicDraw method; for the globecontrol, that is the AfterDraw method. In these methods, you can write direct OpenGL codes like the common developing with OpenGL.[3]

Since the ArcGIS Engine support different programming languages, .NET, Java, C++ are supported, while the OpenGL is implemented with C++, but there are many projects of transplanting OpenGL to other languages, such as the Tao.OpenGL for C#, JOGL for Java and so on. With these projects, the OpenGL can also be implemented and embedded in the ArcGIS Engine through other program languages.

IV. Potential of ArcGIS Engine and OpenGL

The ArcGIS Engine support many formats of images and the OpenGL can define different types of geometry, combining with them can meet all the requirements in the satellite visualization. Compared with the other visualization tools, the tool based on the ArcGIS Engine and OpenGL can display the whole process of satellite image acquisition, including satellite motions, sensor working plans, and satellite image observing followed by the working plans.

V. Satellite Visualization Tool

A. Structure

The satellite visualization tool was consists of several modules: data provider, resource management, motion control, and 2D and 3D visualization scenes, its structure was shown in the Figure 1. The data provider module manage and provide the data of the visualization needed, mainly for the satellite track data, also include the attitude data, working plans, target areas, sun positions and so on. Resource management module manage and provide the data for visualization, include the map files, satellite models, ground station images and so on, it contains an extended interface, resources like the new satellites, sensors, ground stations or others can be imported through it. The motion control module contains a global timer, which used for controlling the refresh events of 2D and 3D scenes, and the user's interface was provided for the interaction. The 2D scene and 3D scene modules are the main visualize module, they change the visualization data from the data provider module and resource management module and the motion commands from the motion control module into visual elements, and provide user's interface for the user to build the custom visual screens.

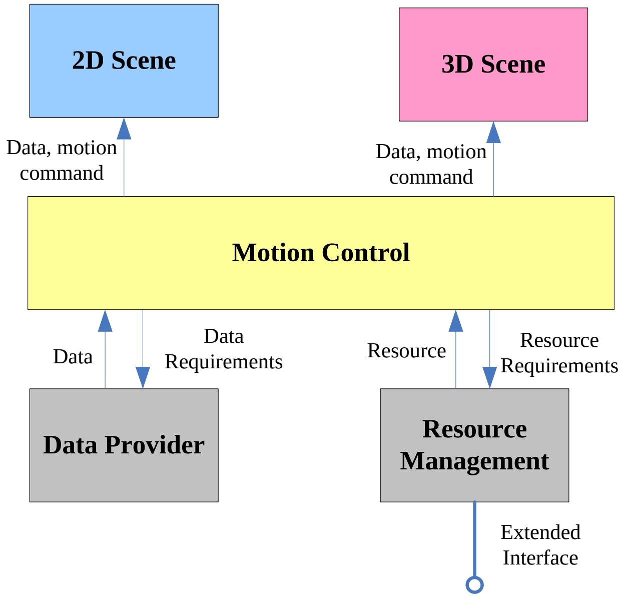

Figure 1 satellite visualization tool structure

B. Visualization Scene

Satellites are moving around the earth, so the first step of visualizing satellite is to build a scene that contains the earth, considering the visualization effects, both 2D and 3D scenes are required. There are many controls in the ArcGIS Engine, the MapControl can provide 2D scene, and the GlobeControl can provide 3D scene, these scenes are like a kind of platform that can display other graphics elements.[4] All of the visualization motions are based on these two scenes, partly by the API of ArcGIS Engine, partly by OpenGL. The data that stay the same when visualizing, such as the world map, satellite image and so on, are processed by the API of ArcGIS Engine; while the data that change all the time, such as the satellite track, model, sensor coverage and so on, are processed by the OpenGL.

1) Raster Data

In order to make the scene more real, we need to fill the blank scenes that built by MapControl and GlobeControl with earth pictures, that is to load raster data into 2D and 3D scene. The ArcGIS Engine have very good support with raster data, like the jpeg, tiff, raw and so on, data files can be loaded directly, also we can have many different kind of satellite images loaded, which can make the visualization more real.

2) Vector Data

The world map should be displayed too, for we need to know where the satellite can be observed, and the bound line of the country or province are usually stored in vector data files. The ArcGIS Engine support many types of vector data, such as shapefile, geodatabase and so on. Also the vector data can be labeled with the area name, then we can have a world map with details.

3) DEM

The DEM (data elevation model) can describe the elevation of the earth, thus we can see the surface of the earth in the 3D scene. The GlobeControl support the DEM data and can set it exaggeration parameter, which make the terrain more obvious.

4) Online Maps

As the development of web map service (WMS) and tile map service (TMS), the online maps are becoming more and more common. The ArcGIS Engine support many kinds of online maps, such as the ArcGIS online maps, bing maps, google maps, open street maps and so on. These online maps can be loaded as a normal layer in the 2D and 3D scene, with the online maps, the visualization scene could be more real and the map data can be updated as soon as possible.

5) Data Management

So far we have loaded several kinds of data into the visualization scene, a good management is needed to make them work together. The ArcGIS Engine display all the data as layers, we can have different display effects with ordering layers. In order to display the scene as we need, set the visual scale of every layer, and then we can show or hide one layer at the appropriate time, such as hide the province bounder layer and show the country bounder layer at the world view, make them reverse when at the country view. Besides, a layer control was provided for the user to control the layer hide or show. In the 3D scene, set the DEM layer to the bottom and set its exaggeration parameter, drape the other layer on it, then all the layers were well managed and a whole view of earth was built.

Figure 2 and Figure 3 Shows the 2D and 3D visualization scenes, the map data are from the website of ESRI.

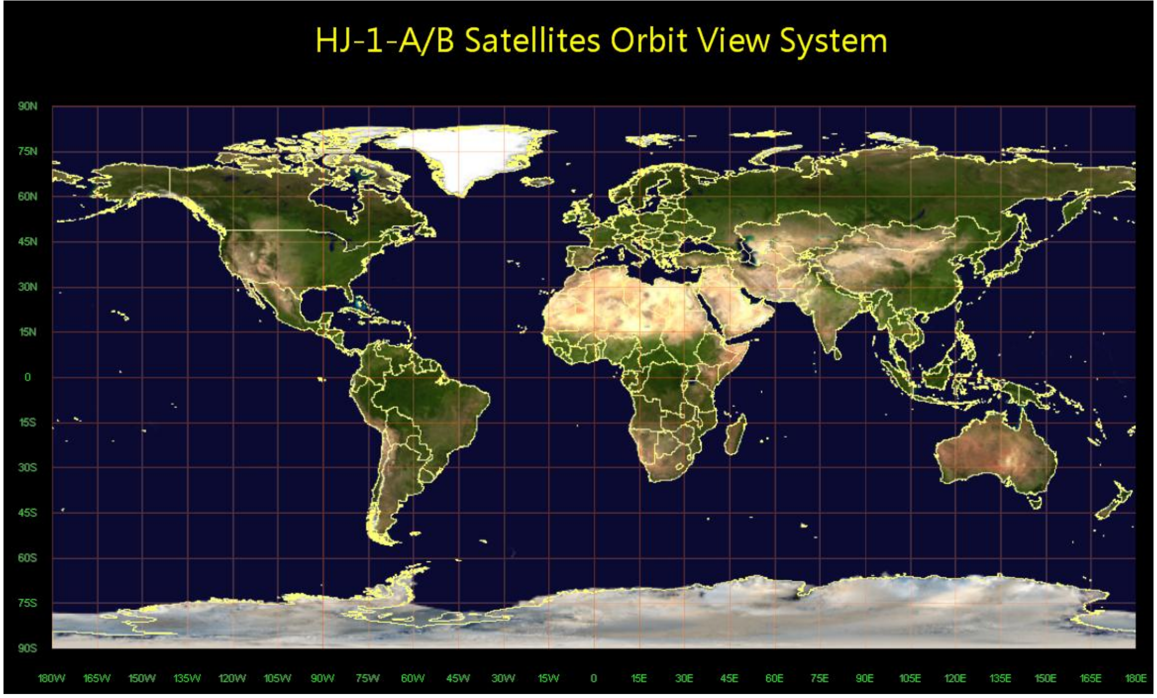

Figure 2 2D visualization scene

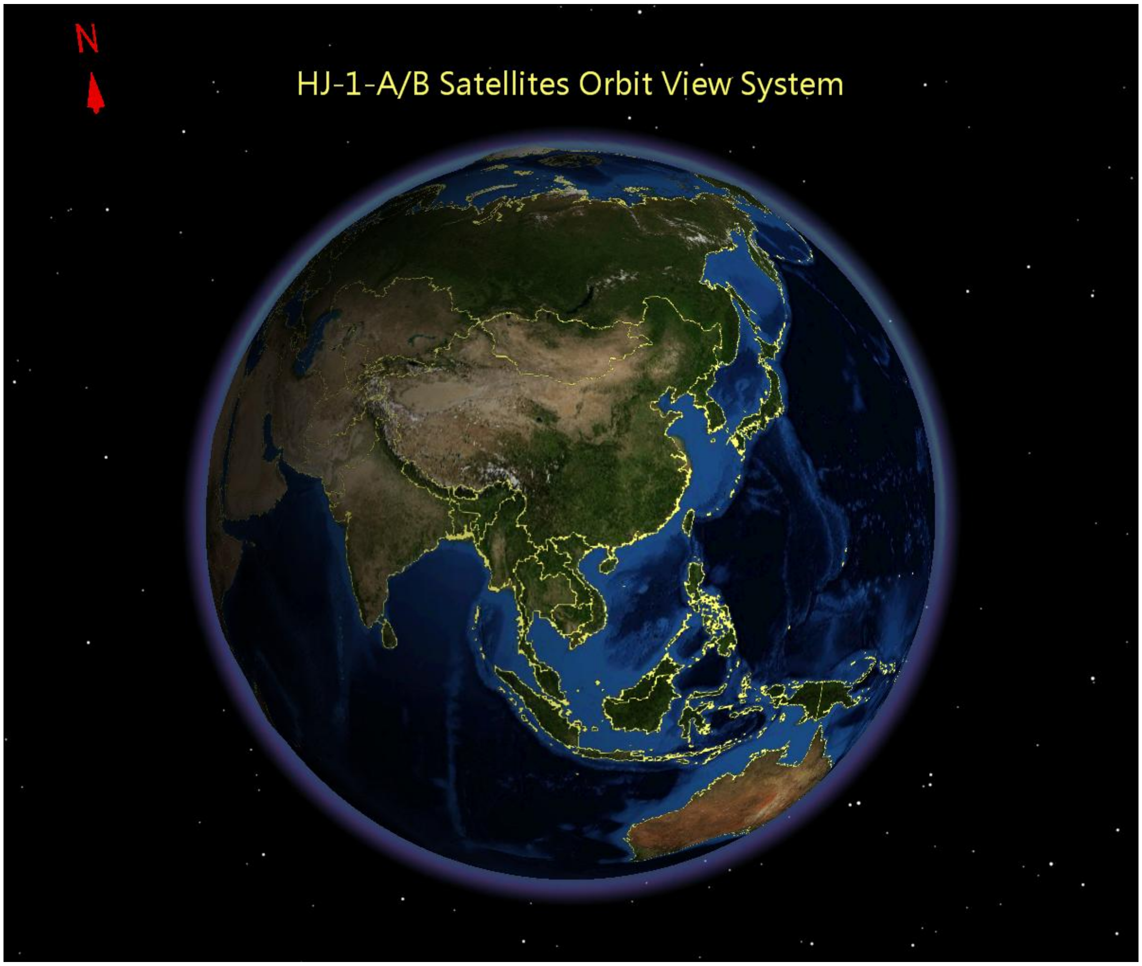

Figure 3 3D visualization scene

C. Custom Drawings

Once the 2D and 3D earth scenes have built, we can have our custom drawings to perform the satellite working status. Several contents should be displayed, including satellite model, orbit, sensor coverage, ground station, sun, moon, stars, information, other animations and so on. These elements are drawn with OpenGL, since the data are dynamic and require high speed.

1) Satellite Model

Satellite model is used to show the position of satellite, in the 2D view, an image of the satellite is used, the picture should have a transparent background so as not to affect the map; in the 3D view, a 3ds model should be provided, which make the satellite more real in the 3D view. OpenGL support the picture and the 3ds model very well, we can create display lists and show them in each scene.

2) Satellite orbit

The satellite orbit is always be described as a line, while drawing a line is quit simple in the OpenGL, with the given satellite coordinates, we can draw it both on the 2D and 3D scene.

3) Sensor Coverage

Every resource satellite carries observing sensors, the coverage of the sensor should be displayed so as to know where they were observing. Satellite sensors have many types,[5] such as the CCD, HSI, SAR, different types of sensor have different ground shapes when observing, Figure 4-6 shows the ground shapes of common sensors.

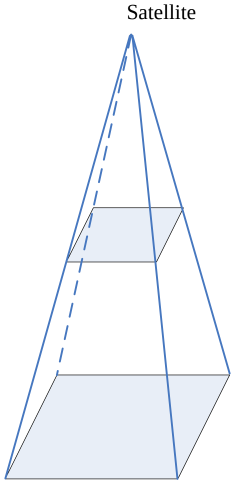

Figure 4. Frame imaging.

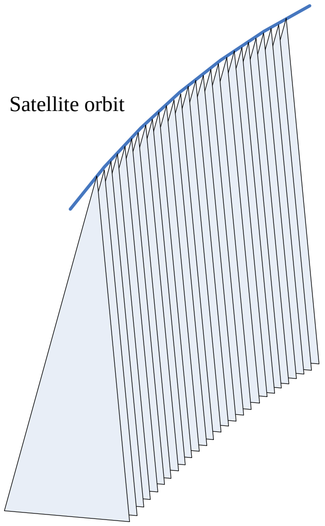

Figure 5. Linear array push-broom imaging.

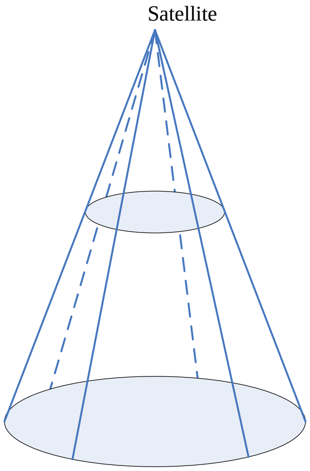

Figure 6. Antenna beam imaging.

With the geometry type of the coverage shape, different sensor coverage can be rendered. We can render the frame imaging type can be rendered as pyramid; the linear array push-boom imaging as triangle; the antenna beam imaging as cone, and the way of rendering will be automatically chosen. Since the satellite coordinates and the observe angle of the sensor can be obtained, the shape of the coverage area can also be computed.

Sensor observing path should also be displayed, which indicates the area that can be observed by satellite. This is a bit complicated in the 3D scene, since the path is the curved surfaces, and can not be described with equations, thus was not supported by the OpenGL. But there are many ways to solve it, one of them is rasterizing, divide the paths into small pieces and draw them on the 3D scene.

4) Ground Station

The ground station is very important to the satellite, in the satellite visualization tool, the position of the station and the receiving area should be displayed. Like the satellite, the position of the station can be displayed as 3ds model and image, while the receiving area can be displayed as circle or part of sphere, also we should know that the bounder of the receiving area is associated with the orbit of the satellite and the antenna of the station. So it is always changing when setting for different satellites.

5) Sun, moon, terminator

Terminator is also important for the satellite, different working modes should be set when crossing the terminator. The terminator can be described as a line, but in order to mark the day and night, it must be described as polygon in the 2D scene, and part of sphere in the 3D scene, then fill it with black color, also the transparency should be set so as not to cover the part of the earth.

For the sun, drawing it as the 3ds model is not a good way, since the light view of the sun can not be described in the 3ds model, a schematic drawing of the sun is better, so does drawing the moon.

6) Text and Label

Text and label are needed to mark the graphics elements, it is quite simple to draw them on the 2D scene, while in the 3D scene there need some changes. The labels should not rotate with the earth so as to face the user, so they must be drawn as the "billiard image", when the earth rotated, the label rotated on the opposite direction, then the text will always be seen.

7) Animation

The OpenGL support many kinds of motions, we can design some animations for the satellite to show its working status, such as sensor observing, communication, satellite rotation and so on. Take the communication animation as an example, when the satellite reached the receiving area of the ground station, draw a serial of circle, change the count of the circle and made it as an animation.

8) Custom drawing management

The custom drawings are drawn in the pipeline of the OpenGL, the former drawings will be covered by the latter, so appropriate drawing order is needed. In the satellite visualization tool, custom drawing elements are arranged by their distance to the center of the earth.

Figure 7 shows the final rendering effect of the 3D scene.

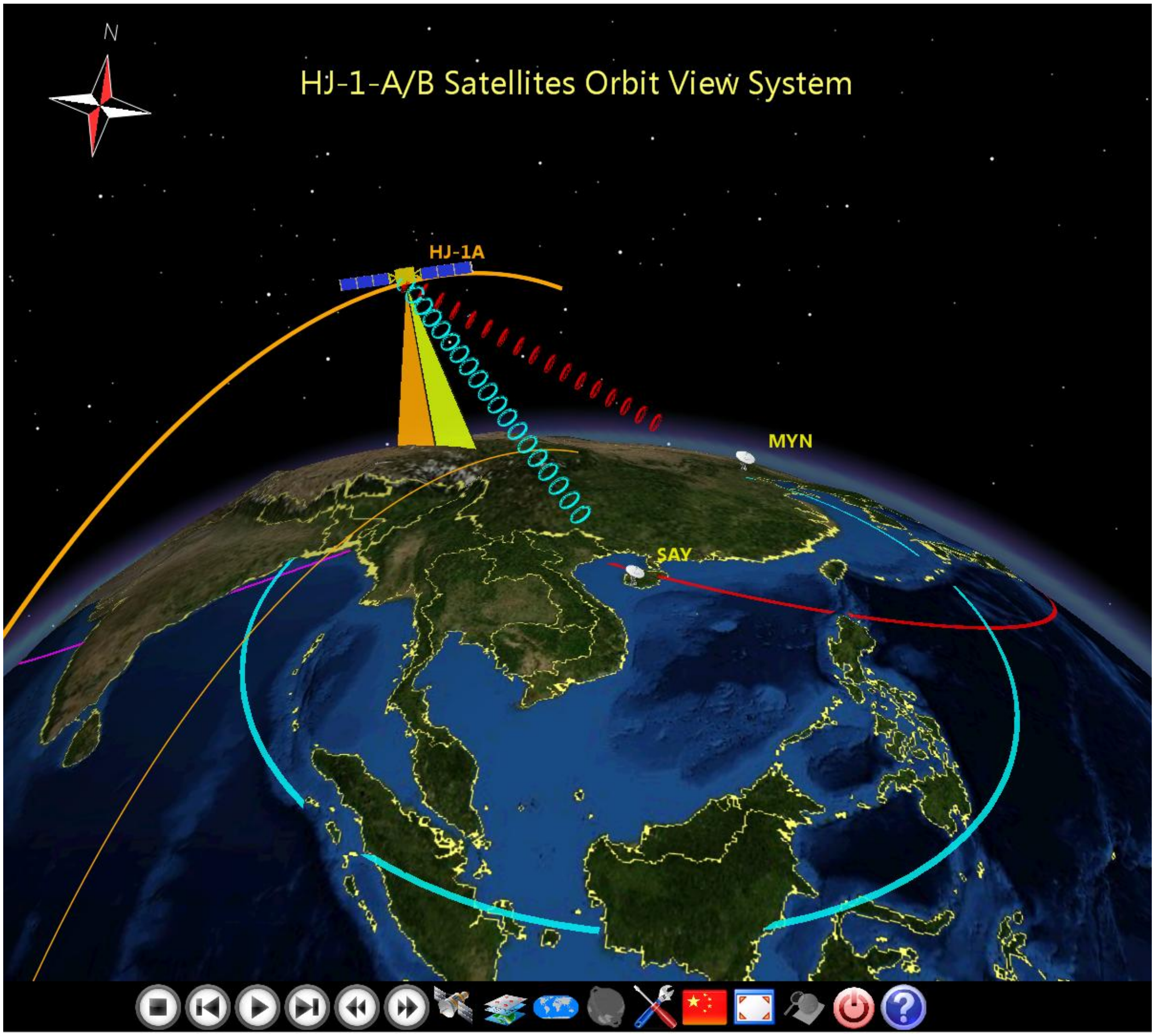

Figure 7 3D scene rendering effect

D. Data Provider and Resource Management

Data provider and resource management works together and supply the visualization data, the data provider computes the satellite trace and other kind of data, by default the satellite trace was computed by the TLE of satellite, but loading satellite trace data file is supported, this can be switched by the satellite type. The resource management all the resources of the satellite visualization tool in the database, including satellite, sensor, ground station parameters, satellite working plans, satellite models, images and so on. Any appropriate requirements for the resource will be met, also an extension interface was provided for extending resources, a new satellite or sensor can be added through this interface.

E. Motion control

Satellite motions are decided by the working plans and the operations of the user, in the satellite visualization tool, three kind of working modes are set in the satellite visualization tool, real mode, which simulates the real situation of the satellites; replay mode, which simulates the satellite situation at the time set by user, and the moving speed and direction of the satellite can also be set; preview mode, which displays the satellite track in the time span set by user.

The motion control read the working plans of the satellites, accepts the inputs from the user, then generates the data requirements for visualization, after receiving data, send them to the visualization scenes. The 2D and 3D visualization scenes are displayed as the front and back, only the front one could be seen. A timer was used to control the refreshing frequency, and only the front scene will be redrawn.

VI. Conclusion

We have described the satellite visualization tool based on the ArcGIS Engine and OpenGL, the visualization tool is common to many space science missions and have many implementations. The ArcGIS Engine has its advantages at map files and satellite data visualization, while the OpenGL is an additional function for its custom drawings, the combination of each other is a good solution for the earth resource satellite visualization. This visualization tool has good appearance and effects, and was used in the HJ-1A/B, SJ-9, ZY-3, ZY-01-C ground systems.

Future plans include the improvements of rendering speed, especially when the satellite number is more than 10. And so far the satellite visualization tool can only display the satellite working status, the process of launching and returning could not be visualized, this is very useful for the space station that returns back to the earth, such as the TIANGONG-1, so this function should be extended for the future.

Acknowledgments

The research described in this paper was carried out at the Earth observation Technology Application Department (ETA), Academy of Opto-Electronics, Chinese Academy of Science, led by Li Ziyang (ETA) and Hu Jian (ETA), and under a contract with Euan Macinnes.

References

[1]"ArcGIS Engine Overview" *ESRI products* [online database] ,URL: http://www.esri.com/software/arcgis/arcgisengine/index.html [cited Jul 2007]

[2]"OpenGL" *Wikipedia* [online database], URL:http://en.wikipedia.org/wiki/OpenGL

[3]"Implementing custom globe layers with dynamic feed " *ESRI developer Network (EDN)* [online database],URL: http://resources.esri.com/help/9.3/arcgisengine/dotnet/b0e91ce8-c180-47dc-8323-06cac5d77064.htm [cited Jul 2007]

[4]Wu Hao, Li Chuanrong, Li Ziyang, Hu Jian, "Visual Simulation of Satellite Track by ArcGIS Engine", Microcomputer Information, Vol. 26, No.4-1, 2010, pp.170, 172.

[5]Hu Yasi, Meng Xin, Li Ligang, "A Satellite Coverage Computing Model for Multi-sensor," Remote Sensing Technology and Application, Vol. 24, No.4, Aug.2009, pp. 537, 542.