International Journal of Smart Home
Vol.7, No.5 (2013), pp.187-196
http://dx.doi.org/10.14257/ijsh.2013.7.5.19

Inverter Power Monitoring System Based on VRML Virtual Reality Technology

[1]Qing-yun Dai, [1]Jin-dou Wang, [1]Rui-Tao Liu
[1]Jun-hong Cheng, [1]Yi li and [2]Wei-yang Wu

[1]Dept. of Electric and Electronic Engineering
Shijiazhuang Vocational and Technology Institute, Shijiazhuang050081, China
[2]College of Electric Engineering
Yanshan University, Qinhuangdao 066004, China
dddqqyy@126.com dqy_2003@126.com

Abstract

A Inverter power monitoring system based on virtual reality is introduced. A good interactive 3D virtual scenes Based on the communication of VRML and java, can get real-time power state parameter of inverter from database. Based on the virtual scenes, a variety of monitoring functions is achieved.

***Keywords**: VRML; monitoring; interactive; java*

1. Introduction

Virtual reality is a man-machine environment technology supported by computer technical, by which human beings can interact with a computer and its complex data. It combines advanced computer technology with sensing and measurement technology, communication technology, simulation technology, microelectronic technology. The virtual reality technology using computer technology to generate a virtual environment that enables the user into it through a variety of sensing devices, then the user can direct interact with the environment naturally.

VRML (Virtual Reality Modeling Language), a file format to describe interactive three-dimensional objects and three-dimensional world, is a powerful 3D description tool of modeling and rendering graphical. As it appears, especially the development of standards VRML2.0 has brought considerable progress for Web-based virtual reality [1].

By using this language, 3-D graphic model could he quickly obtained without the environmental limitations and the good quality of image could be realized. Through the user interface of the graphical embedded simulation in JAVA applet and VRML, the communication of programmatic interface to a nale could be established by- 3D graphics, and the controlling model of various movements could he flexibly achieved[2].

In industrial, VRML plays an important role in design, develop and test products. Since the design, test and relevant experiments of products will extend the product development cycles and increase the product development costs, it is very important to shorten the product development cycle and reduce the development cost of products.

Virtual design, virtual test and virtual reality experiment can apparently shorten the time of the product design, test, and the experiment cycle and reduce the cost of product testing and

ISSN: 1975-4094 IJSH

experiment. We can build mathematical model to establish and test the products, and the modelof emulation by the computer can achieve the purposes and analyze the real physical tests [3].

This paper studies how to use the virtual reality technology to monitor the inverter power, and to provide a highly harmonious man-machine relationship. The realistic 3D interface allows the user to operate in control room, as operating a real object.

2. The Basic Configuration of the System

Monitoring system based on virtual reality technology consists of three parts: the PC monitoring module, DSP intelligent controller and inverter power module. Show in Figure 1 below.

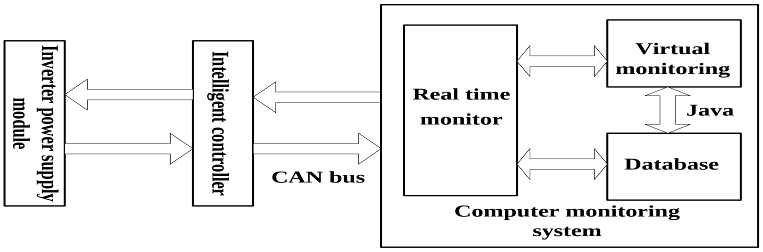

Figure 1. The Virtual Control Schematic Diagram

The function of each part is described as follows: (1) PC monitoring module: PC monitoring consists of two parts, including the real-time monitoring developed by visual basic 6.0 and virtual reality monitoring. Between them, virtual reality monitoring is focus of this discussion, which uses powerful graphics capabilities of VRML to draw a virtual three-dimensional scenes and achieve interaction with the database by communications between java and VRML. Then the status data of inverter power that real-time monitoring program stored into database is extracted and displayed in the virtual scene. In order to control the inverter power supply, control commands also can send to DSP intelligent controller. (2)DSP intelligent controller: the controller programmed by C language communicates with host computer via CAN bus. (3) Power inverter module: monitoring objects, on-site state data sources.

3. Communication between Java Program and VRML Scenes

Since it has advantages such as small file size, real-time shaders, full range of viewing, VRML becoming a new generation of 3D network technology has been widely used. But most of the VRML works still remain in the static display stage, lack of dynamic and interactive, mainly in the following points [4].

Firstly, the browser is more difficult to control. There is no accurate means of control and difficult to locate, we must use mouse or keyboard to change the perspective to watch the VRML world in almost all the browsers. Secondly, the VRML hardly communicate with the HTML file, just as plug-ins embedded in Web pages [5]. The VRML data can not transmit to HTML. HTML also can not read VRML data. Thirdly, VRML interaction is only limited to a mouse click or move，can do nothing for the require of input data from user.

Since there is no nodes to display decision logic when behavior changes, the ability to interact with the outside of VRML is not very strong. If you want to create a complex 3D interactive scene, it must be combined with other language to construct virtual scene which has interactive ability [6].

Fortunately, Java program can access the VRML browser interface and executive environment by API which support the VRML[7]. Java is a programming language very suitable for the development of cross-platform application of network. The combination between VRML and Java, can overcome some defects of VRML, provide more abundant, convenient human-computer interaction between the user and the VRML scene, and meet the demands of complex systems.

All support by java form additional packaging category, through which, java program can access the VRML scene, receive and send the event, achieve full control over the VRML scene[8]. VRML and java communicate in two ways, namely, the internal java script creation method (Java Script Authoring Interface -- JSAI) and external program interface (External Authoring Interface -- EAI).

3.1. Communication between Java Program and VRML by JSAI

The scene is limited in the node line to transfer the previous event along a route defined beforehand in VRML, so the procedures of control event, can only outside the scene. Fortunately, VRML provides a node to combine with the scene outside the program :Script script node[9].\

When the user instantiates a script node of VRML, it will have a user-defined sensor or a interpolator, and a programming interface. The interface is composed of three parts: a plurality of interface field, eventIn, Eventout. Then, no action perform in the node, also did not create any visible domain[10]. The interface only provides an equivalent entity which can through external field program control for the user. The user need use programming language script supported by VRML browser to complete the required behavior.

JSAI consists of three components: vrml, vrml.field, wrml.node. Use the JSAI, it can accomplish complex tasks by using powerful features of Java. Because of a class Script defined in the java vrml.node package, java program can access the script node. Methods defined in the SCript class include initialize(), getField(), getEventout(), getEventln(), proeessEvents(), proeessEvent(), eventProeessed(), shutdown(), toString()[11].

The vrml.node package consists of Script and Node. The script class is an abstract class. By the script class inheritance, java program and script node can establish contact so as to control the other nodes in VRML. VRML domain data types classes which representing the spatial position of the rotational direction, the time, the node and the other VRML domain form the vrml.field package[12]. For each VRML domain types, JSAI provide corresponding class represents. In JSAI, we can use Field class and constfield class to encapsulate domain object types. The former is used to encapsulate objects that can read and write, these classes provide get and set methods to read and write object value; the latter used to encapsulate those read only objects, these classes provide only a get method which can read the value of the object.

In order to realize the VRML and java communication, define domain field values of script nodes firstly, and then set route to link other nodes in VRML to script. The event which linked to script node will be passed to the eventln domain of script node when it happened. One eventIn field event which arrives at the script node will cause execution of the java program. The script node find java class which must inherit from the script class via the address specified in URL domain. Java classes call initialize() method firstly when it work. It must get values form eventin or field domain of script nodes in initialize( ) method, or return a value from java program to eventout domain in script nodes. GetField(), getEventln(), getEventout() are methods provided by script class:

```
getField(“fieldName”)；
getEventIn(“eventInName”)；
getEventOut(“eventOutName”);
```

GetField() method is used to return a reference value of the script node named fileName field. The reference value can be converted to the corresponding value of the java program. GetField () method is used to return a reference value of input events. GetField() method is used to return a reference value of eventoutfield event's output. As parameters, fieldname, eventinname and eventoutname should correspond to name defined in the script node. The corresponding subclasses of the Field class should be in accord with types defined in the script node. Subclass of the field class exists in vrml.field package, which provides getvalue() and setvalue() method, is used to convert VRML value to java value，also can return and set the VRML field to java value.

After initialization is completed, the processEvent() or processEvents() method provided by script class can capture the event transmit from the script node. PocessEvent() and processEvents() method are common entry points for all external events. Java program make corresponding processing through the substitution of these two methods on the event. getName() method from VRML package's Event class returns the name of the event (this name is same as the eventlnName that happened in the Script node). Vrml package event getname() method returns the name of the event to determine the occurrence of an event, so as to make the appropriate treatment. getValue() method of The Event class returns an instance. Constfield class, subclass of constfield class is a read-only class. It does not have setvalue () method, can only read eventin domain’s value of the script node[13].

All events dealt with, procedure calls the script class in the eventsprocessed() method. In the procedure can join the user wants to achieve animation, interaction effects. When leaving the virtual world, a variety of animation effects also can achieve. Remove script node or unloaded scene contains script node or be replaced by another scene. Uninstall script by calling the shutdown() method to finish the work.

3.2. Communication between EAI and VRML

EAI(External Authoring Interface) is a high-level java class package which can enhance the fusion and communication capability between VRML scenes and external environment. There are 51 classes and interfaces in EAI class package. They were divided into three package: vrml.external, vrml.external.field and vrml.external.exception. Among them, vrml.external packages are only two classes: browser and node, were used to obtain an instance of the browser and save the reference node in the scene. There are 51 classes and interfaces in vrml.external.field class package, respectively represent data type of EventIn and EventOut in VRML. There is also a event out listener (eventoutobserver.class) to listen out events that the scene interest in. 5 exception classes defined in the vrml.external.exception package, they are inherited from java.lang.runtime.exception class[14].

Browser class and Node class are mainly used in java program. Moreover, attribute classes (Eventln*, EventOut*) to process eventin domain and eventout domain, a variety of *Exception class，to complete the EAI description. The browser class is encapsulated in vrml.external package, which contains most method of Browser class refer to in 3.1.

In order to control the nodes in the VRML scene, the first thing to do is to get a browser instance, namely call the the static method getbrowser(), and then obtain the nodes in the VRML scene by getnode() method. After achieved a reference of an access node, we can send input event to that node and read the output event and listen event of the node.

JavaApplet access the scene mainly use vrml.external package encapsulated in the Browser class. To access the VRML scene using EAI in JavaApplet, need to get an instance of Browser first of all, it can be achieved by calling static methods getBrowser() of the Browser class. After the browser instance, to control the node in the VRML scene, we need to call getNode() method to obtain the entrance address of the node in the system. In VRML, only the node named by DEF structure can obtain node address when the external EAI program call related method. In a reference to be accessed node, it can be transmitted in event to the node, read and monitor out event of the node. Calling getEventIn() method can send an event to a node in VRML scene event entrance, thus changing the scene. Calling getEventOut() node method can read a domain value send by node in VRML scene export. Callback() method of interface class Eventoutobserver automatically invoked when out event occurs.

4. Connection with Database

Because VRML can not interact with the database directly, we use java to connect with database in this paper. JDBC (Java DataBase Connectivity) is a java application programming interface used to execute SQL statements, is composed of a set of classes and interfaces written in the java language. JDBC access the database through ODBC with the help of JDBC-ODBC bridge. A typical process that JDBC API access database is：create a driver use JDBC's Driver class firstly, and create a JDBC connection object by getconnection() method of drivermanager class. Then create a SQL statement through createsatement() method of JDBC's statement class [15]. The SQL statement will be executed by executequery() method and a resultset object will be created. Resultset object provides the SQL query result set.

5. System Implementation

Inverter power supply status data is displayed on the virtual power supply cabinet through interaction between Script and java. Shortcoming of VRML is made up, and interactive of the three-dimensional scene also become easier, after introducing the java control to VRML virtual reality scenes.

The following is some key code of connection with the database and update the power supply cabinet data procedure:

```
  Connection con;Statement sql;Resultset rs;
  // Connect to the database
  try{
     Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
          }
        catch(ClassNotFoundException e){}
        try{
          con=DriverManager.getConnection("jdbc:odbc:remotecon
          trol","susu","susu");
          sql=con.createStatement();
          rs=sql.executeQuery("SELECT * FROM state ");

          while(rs.isFrist()){
// Access to the data values in the corresponding data table


         int xuhao=rs.getInt("id");
         String xuhao1=String.valueOf(xuhao);
         Int model=rs.getInt("mod");
         String model1=String.valueOf(Ml);
         float v=rs.getFloat("voltage");
         String v1=String.valueOf(v);
         float el=rs.getFloat("current");
         String el1=String.valueOf(el);
         float ph=rs.getFloat("phrase");
         String ph1=String.valueOf(ph);
         float fr=rs.getFloat("frequency");
         String fr1=String.valueOf(fr);
         float ha=rs.getFloat("power");
         String ha1=String.valueOf(ha);
         float no=rs.getFloat("nopower");
         String no1=String.valueOf(no);
        }
      }
```

Because of the bridge of communication with the database, the change of the virtual scene information become more abundant and flexible. When VRML extract information from the database, the first thing to do is sending VRML instruction to the java applet through JSAI. Java applet access local database through JDBC data, and return data information to VRML. Finally, the browser interprets and executes changed information, and feedback to the virtual scene, caused the change of scene.

```
public void initialze(){
// Access reference value of eventOut field, set display
// initialization

     setStringBlock=(SFString)getEventOut("setStringBlock")
     ;
       textNode[0]=null;
       textNode[1]=null;
       textNode[2]=null;
       textNode[3]=null;
       textNode[4]=null;
       textNode[5]=null;
       textNode[6]=null;
       textNode[7]=null;
       textNode[8]=null;
}
……
     public void processEvent(Event e){
if(e.getName().equals("interval")==true){
//eventIn occurs


setAgent();
            }
}
```

Interactive 3D scenes shown in the following figure:

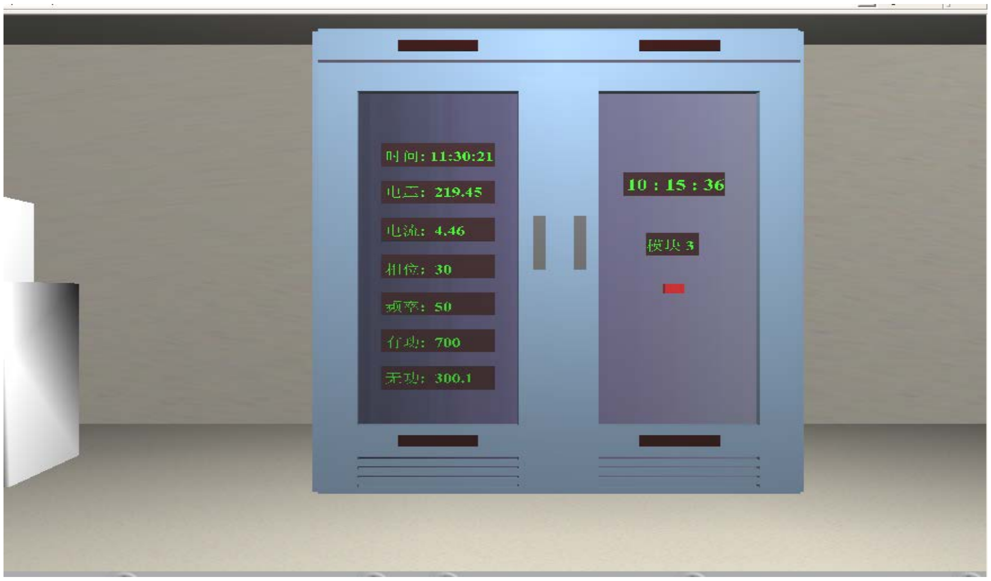

Figure 2. Virtual Power Supply Cabinet

6. Conclusions

This paper discusses several methods of VRML interaction with the outside world and the connection method VRML with the database. A virtual reality monitoring system of inverter power is proposed and implemented. Virtual reality technology has a considerable development in recent years, its importance has been recognized by more and more people, more and more widely applied. Application of virtual reality to on-site information monitoring system by using, is the current research focus in the field of automatic control. I believe that virtual reality technology will surely promote the development of modern automation technology, entering a new era.

Acknowledgments

This research was supported by National Natural Science Fund Projects (50237020).

References

[1] J.-S. Lee and P.-L. Hsu, "Remote Supervisory Control of the Human-in-the-loop System by Using Petri Nets and Java", IEEE Transactions on Industrial Electronics, vol. 50, no. 3, **(2003)**, pp. 431-439.

[2] Y. Yubiao and H. Hanwu, "A Study of VRML-JAVA Based Robot Motion Simulation", Mechanical Science and Technology, vol. 23, **(2004)**, pp. 120-122.

[3] M. A. Zhuang, "A Study of the Simulation for Robot Motion Based on VRML-JAVA", Hydromechatronics Engineering, vol. 12, **(2012)**, pp. 75.

[4] B. Haitao, "The virtual reality technology in automabile design study", Equipment Manufacturing Technology, vol. 10, **(2009)**, pp. 33-38.

[5] G. Sings and F. Thalmann, "Virtual Reality Softwareand Teehnology", Proceedings of the Conference, WorldSeientifiePublishing ComPany, Ine., **(1994)**, pp. 67-89.

[6] A. Lames, D. Rnadeau and J. L. MoreLand, VRML Resoure Book, **(1997)**.

[7] K. Jamsa, P. Sehmauder and N. Yee, VRML Programmer's Library, Publishing House of Electronic Industry, vol. 1, **(2000)**.

[8] D. R. Nadeau, Building Virtual Worldswith VRML, IEEE ComputerGraPhics and Applications, **(1999)**, pp. 18-29.

[9] G. Xiaoying, “Research on Three- freedom Robot Avatar Model and Its Application”, Jinan: Shandong University, **(2007)**.

[10] M. Rohrmeier, “Web based robot simulation using VRML”, Proceedings of the 2000 winter Simulation Conference. Orlando, **(2000)**, pp. 1525-1528.

[11] L. Dong and L. Yingming, “VRML-based application inchemical engineering virtual experiment”, The proceedings of 2011 2nd International Conference on Networking and Information Technology. Wuhan, **(2011)**.

[12] Y. Yanqing, S. Jianlong and N. Xiao, “Based on the VRML network interactive virtual reality modeling”, Journal of System Simulation, vol. 18, no. 10, **(2006)**, pp. 2827-2831.

[13] J. Kris, S. Phil and Y. Nelson, “VRNIL programmer's library”, Beijing: Publishing House of Electronics Industry, **(2000)**, pp. 122-123.

[14] S. Jia-quan and L. Qun, “Development of Three-dimensional Molecular Dynamics Simulation software Based on Components Technology”, Journal of System Simulation, vol. 20, no. 23, **(2008)**, pp. 6380-6386.

[15] L.-X. Zhu, Z.-L. Liang and L.-Y. Huang, “Numerical Simulation of Dynamic of Dynamic Response of Supple Nets”, China Ocean Engineering (S0890-5487), vol. 20, no. 3, **(2006)**, pp. 443-456.

Authors

Qing-yun Dai received the B.Eng degree in Electrical Engineering from BaoTou Institute of Iron & Steel in BaoTou China and the M.Eng degree in Computer Engineering from YanShan University in Qinhuangdao china. He is currently a lecturer in Shijiazhuang Vocational and Technology Institute in Shijiazhuang, China. His main research interests are in the areas of computer networks, and embedded technology.

Jin-dou Wang received the B.Eng degree in Electrical Engineering from Hebei University of Technology in Tianjin China. He is currently a associate professor in Shijiazhuang Vocational and Technology Institute in Shijiazhuang, China. His main research interests include power electronics and embedded technology.

Rui-Tao Liu received the B.Eng degree in Electrical Engineering from Hebei University of Technology in Tianjin China and the M.Eng degree in electrical Engineering from Huazhong University of Science and Technology in Wuhan china. He is currently a lecturer in Shijiazhuang Vocational and Technology Institute in Shijiazhuang, China. His main research interests include power electronics and embedded technology.

Jun-hong Cheng received the B.Eng degree in Electrical Engineering from Shijiazhuang University Of Economics in Shijiazhuang China. She is currently a lecturer in Shijiazhuang Vocational and Technology Institute in Shijiazhuang, China.Her main research interests include automatic control and embedded technology.

Yi li received the B.Eng degree in Electrical Engineering from Tianjin Normal University in Tianjin China. He is currently a lecturer in Shijiazhuang Vocational and Technology Institute in Shijiazhuang, China. His main research interests include computer control and embedded technology.

Wei-yang Wu received the B.Eng degree in Electrical Engineering from YanShan University in Qinhuangdao china. He is currently a professor and doctoral supervisor in College of Electric Engineering at Yanshan University, Qinhuangdao, China. His main research interests include high power pulse technology and application of high-power GTO.