

Study on Remote Monitoring System of 3D Scanning and Measuring Machine

WANG WEN □ LU KEQING □ CHEN ZICHEN
 Institute of Advanced Manufacture Engineering
 Zhejiang University
 ZheDa Road 38#, Hangzhou, 310027, Zhejiang
 P.R.CHINA

Abstract:-Combing with remote monitoring and control technology with production management ideas, a remote monitoring and management system of 3D laser scanning machine is presented in this paper. The operation model, system structure is studied and the theory to realize the system function is also given. With the net programming technology, the prototype system is successfully developed, furthermore, the key technologies were discussed. Finally the effect of the system is testified.

Key-words:- Scanning and measuring machine, Remote monitor and control, Socket, Screen share, Database

1 Introduction

The rapid development of Reverse Engineering stimulates the usage of Three-Dimensional Laser Scanner (3DLS), which has a good performance nowadays. However, most 3DLSes today are lack of network function. This disability brings in many problems: Unable to capture the status of 3DLS from distance, not mention controlling the system remotely; Unable to arrange tasks for distributed 3DLS dynamically; Excessive demand for technicians as every 3DLS needs monitoring separately; Unable to share the information among each 3DLS. As a result, the efficiency of these equipments isn't satisfied, but its managing cost is high.

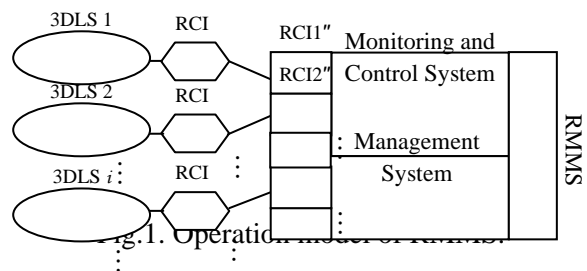
Remote Monitoring and Management System (RMMS) of 3DLS, based on computer network technique and production management ideas, integrates the 3DLSes in different locations to make the best use of resources.

*Sponsored by Provincial Key Project of Science and Technology of Zhejiang □ 2003C11023 □ 2003C21031 □

2 System Structure

2.1 Operation Mode of RMMS

According to its function, RMMS can be divided into two subsystems: Remote Monitoring & Control System as well as Management System. The operation mode of RMMS is shown in Fig.1.



The controller of 3DLS is a kind of Open Numerical Control System based on PC, which has no function about network. However, attribute to its opening architecture, we can embed network interfaces inside the existing NC System to add its ability to network. Therefore, a remote communicate interface (RCI) is developed inside 3DLS for connecting to RMMS .

As shown in Fig.1, the 3DLS will send a request to RMMS through RCI after it started. As soon as the RMMS detected the request, it will create a new interface named RCI^i which responsible for connection, then add RCI^i to the interface list. After the connection is successfully established, RMMS will distribute each communication channel within RCI^i an independent thread, therefore, all the communication channels will never interfere with each other. RMMS will also receive a message from 3DLS i when it close, subsequently, RMMS will close all the threads related to RCI^i . And then delete RCI^i from the interface list.

match these four interfaces.

The communication channels between sub-interfaces and modules are established by Socket. Take Stimulant Panel and the sub-interfaces of Real-time Monitoring & Control as an example, the process of creating a communication channel between them is shown in Fig.3.

The other three communication channels can be established in the same way, all the connecting sockets created by RMMS will be added to the interface list, that waiting for communication command.

2.2 Structure Mode of RMMS

As illustrated in Fig.2, the RCI of 3DLS consists of four sub-interfaces including Real-time Monitoring & Control, Screen Sharing, Video and Audio as well as File Transfer. Correspondingly, the Monitoring & Control Subsystem of RMMS has four modules to

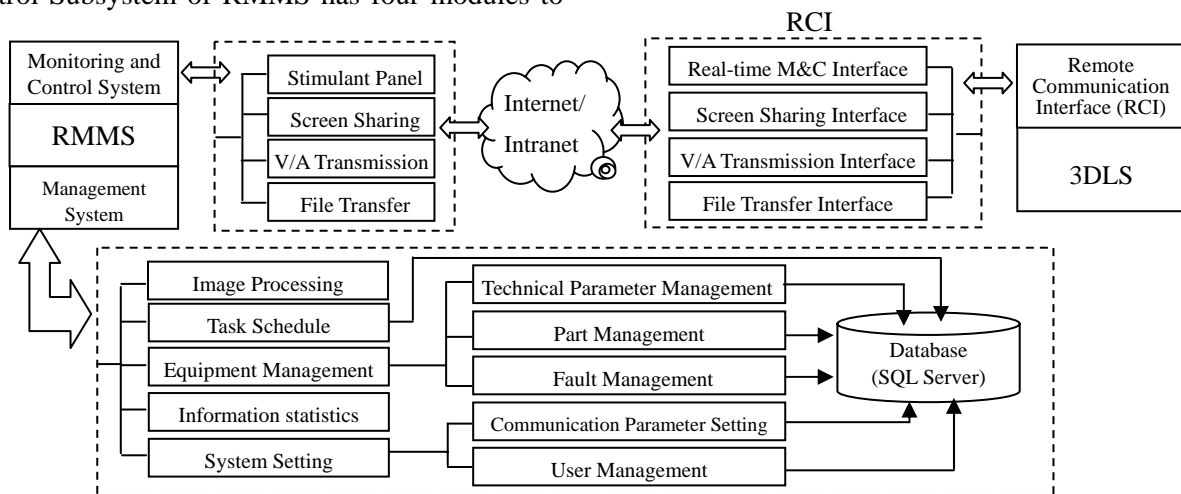


Figure 2. Structure of RMMS

Equipment Management is the key module of RMMS' Management Subsystem. It is a miniature expert system, which will be enriched when varieties of fault cases are recorded by users. Furthermore, it is capable of summarizing related knowledge from the recorded cases. After a period of time, the knowledge it possess will be enough to diagnose the fault types, fault positions, even propose some suggestions for repair when one 3DLS breakdown. As a result, the diagnostic

knowledge, maintain methods which possessed by the separately 3DLS are fully shared through this module. The other modules of Management Subsystem are shown in Fig. 2.

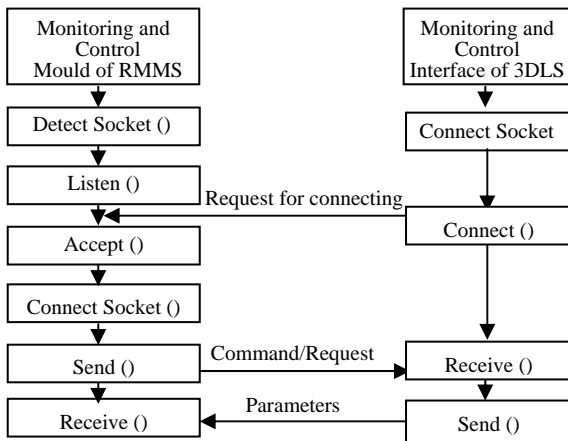


Fig.3. Process of creating a communication.

3 Implementation of RMMS

3.1 Network Architecture

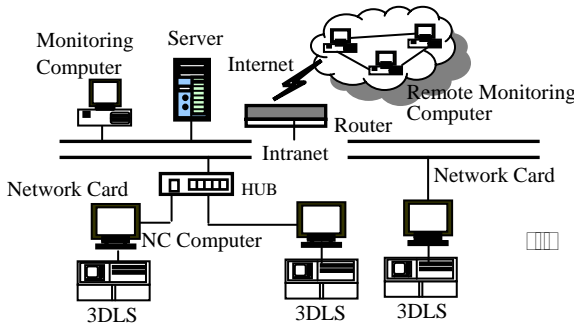


Fig.4. Network Structure of. RMMS

Making use of Intranet as well as Internet resources, adopting Visual C++ as the developing tool, the system is successfully finished. As shown in Fig. 4, the Numerical Control Systems run at the NC computers while RMMS runs at the monitoring computer or remote monitoring computers. RMMS communicates with 3DLSes through Intranet or Internet.

3.2 Analysis of Key Technology

3.2.1 WinSock Technology

The system uses Class WinSock, derived from Class CSocket, to transfer data. It's very convenient to use Class CSocket and its derived classes, because object established by CSocket can transfer data cooperating with CSocketFile Object and CArchive Object. Meanwhile, the data transferring functions

of Class CSocket needn't be directly deployed by user. In the sending end, users only need to insert the sending data into the created CArchive Object. Then CArchive Object will transfer the data to CSocketFile Object, where the socket is responsible for sending the data. In the receiving end, the process goes on the opposite way. The transferring process is simple but effective.

3.2.2 Screen Sharing Technology

The principle of Screen Sharing is: In every cycle of the timer, the screen image of monitored computer would be saved to bitmap document, which will be compressed and sent to the RMMS. RMMS would refresh the interface when receiving new image, to make screen monitoring possible.

The image cut from screen is very big, while a 24-bit image can reach to 2M bit or more. According to this, the sharing window of RMMS couldn't have refreshing continuously. RMMS will compress the image by Huffman Coding, and decode it in the receiving end, which largely reduce the data for transferring.

It's not possible that every part of the screen will change from time to time. In the monitored end, a complete screen bitmap can be transferred first, from then on, only the changed part needs to be transferred. We use screen segmentation to realize only changed region transferring. The user could set the grid for the screen, which divides it to many regions. Then every region is monitored and if any change happens, it will be sent.

3.2.3 Definition of Remote Control Protocol

RMMS can control the Numerical Control System from distance by Sharing Screen or Simulating Panel. Remote control works by means of the Order Information Sequence. RMMS self-defines a Remote Control Protocol (RCP), which defines the coding rules for data transferring between RMMS and 3DLS. The information transferring process in RCP is by coding and decoding. In the sending end, the information are coded and then transferred by Socket to the other end, where the package was decoded and used for remote control.

There is an example when the user clicked the button located in coordination (100,200) of Sharing Window. When the clicking happened, RMMS generated the following code:

Source	Type	Operation	Position
Screen	ButtonEvent	OnClick	100□200

This code was packaged and sent to 3DLS Numerical Control System. RMMS received the information and decoded it: The information is from Screen Sharing Channel, Controlling Button Event, Single Clicking Action, Location is X=100, Y=200. Then RMMS drives the operating system of the Numerical Control Computer to do a real clicking on the button, therefore the remote control had been executed.

3.2.4 Database Technology

Management Subsystem is realized by database support. The system adopts SQL Server2000 database, which has full support from Internet and rich interface programming tools, as well as good elasticity, reliability and extensibility. It uses ADO to access the database, while ADO is based on OLE DB and has the merits of easy to use, fast accessing speed, different database accessing ability and extensibility. RMMS Management Subsystem uses ADO Object and SQL language to manipulate the database, which realizes the functions of the system.

4 Conclusion

RMMS, developed by the referenced proposals, when connected with 3DLS cluster, has an operation interface as Fig. 5, 6.

By the analysis of the system model and experimental data, we can draw the following conclusions:

1. RMMS can provide the real-time status of

3DLS (raw data and Video/Audio information) and control it by Simulating Panel or Sharing Screen from distance. Its Management Subsystem can analyze the parameters of the 3DLS cluster and share the fault diagnosis results.

2. RMMS can improve the management of equipments, share the information and recourses of 3DLS cluster, promote production efficiency and provide the facility for Agility Manufacture.

3. RMMS is extensible. By new setting on the panel, RMMS can monitor other Numerical Control System, which makes it a wide range of application.

References:

- [1] ZHANG Rongtao, SUN Yu and ZHANG Jun. Research on Remote Distributed Intelligent Monitoring, Diagnosis, Maintenance System for Special Engineering Vehicle, China Mechanical Engineering, 13(12), 2002:105-108
- [2] LI Hongsheng, HE Linsong and SHI Tielin. Research of B/S-Based Remote Diagnostics Expert System, Journal of WuHan University of Technology, 21(4), 1999:39-41
- [3] ZHAN Hongfei, GU Xinjian, SHEN Zuzhi. Research on the Web-based Enterprise Remote Collaborative Diagnosis System, 15(2), 2004:146-149
- [4] M. J. Wooldridge. Agent-Based Software Engineering. IEEE Transactions on Software Engineering, 144(1), 1997:26~37
- [5] George Shepherd, Programming with Microsoft Visual C++.NET, Microsoft Press, Washington, 2003 (in U.S.A.)



Figure 5. Main interface of RMMS.

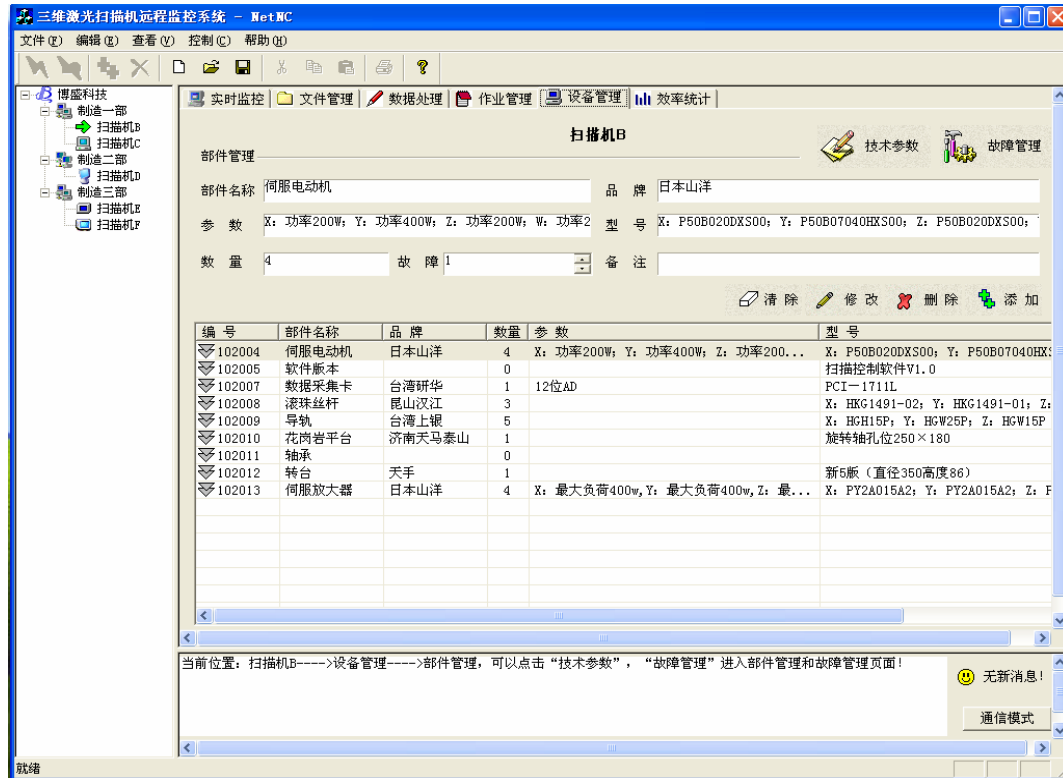


Figure 6. Interface of Equipment Management.