Research Article

Qing Liu, Liye Peng*, and Kang Shang

Nonlinear remote monitoring system of manipulator based on network communication technology

https://doi.org/10.1515/nleng-2022-0246
received March 26, 2022; accepted August 18, 2022

Abstract: In order to study the nonlinear remote monitoring system of the manipulator based on the network communication technology, a research method of the nonlinear remote monitoring system of the manipulator based on the network communication technology is proposed. In this article, in the Visual C++ environment, the transmission control protocol/internet protocol technology is used to build a remote monitoring system for the robotic arm, and it mainly realizes the function of the robotic arm running according to the trajectory in the remote. When storing video, you need to specify the ID of the video source, *i.e.*, the synchronization source in the real-time transport protocol data packet header identifies the robot, which can monitor more clearly. The remote monitoring technology is widely used in the modern automation industry, which not only further extends the human space activity ability, but also removes the human from the dangerous and complex working environment. Therefore, the remote monitoring technology has broad application prospects.

Keywords: network communication technology, remote monitoring, robotic arm, track planning, nonlinearity

1 Introduction

In the new era of the new century, Internet technology has developed rapidly and has penetrated into different application fields, such as the current automation field. In order to save manpower and time better, remote monitoring has been highly valued and widely concerned by engineers [1]. A robotic arm is a robotic device that can have multiple functions. Through the remote control of the robotic arm, not only can people work in a dangerous environment, but also can replace people to complete the tasks in places that people cannot reach [2,3]. Now, remote robotic arms have achieved outstanding performance in space exploration, telemedicine, tele-education, *etc.*, and are constantly exploring new fields to serve the public in their daily life. Therefore, it has very important research significance and good development prospects to combine the monitoring method of the robotic arm with the modern Internet level. The network robot arm has appeared in various aspects such as space, industry, ocean, and battlefield. In terms of remote manufacturing, in order to achieve industrial automation, improve labor efficiency, and reduce the burden on workers, the use of remote-controlled robotic arms can play a very good role. Remote operation enables people to operate these devices anywhere; in the field of entertainment, it is also possible to collect various sounds and images at the far end and can obtain more information and entertainment from more ways; in addition, network robots can also be used in remote health monitoring to measure various data of people's daily life. Then the data are sent to the data analysis center through the network. If abnormal, then targeted treatment is carried out [4]. Remote video surveillance system is a remote monitoring system with computer technology as the core, combining with advanced multimedia technology, network communication technology, and digital image compression technology. Remote video surveillance integrates a variety of leading technology levels in the new century, and has a forward-looking development prospect. Now the types of robotic arms are becoming more and more diverse and intelligent, and they are constantly exploring new fields. When the operator is far away from the robotic arm, by adding video surveillance, it is more convenient to monitor and control the remote robotic arm

* **Corresponding author: Liye Peng,** School of Network and Communication, HeBei Polytechnic Institute, Shijiazhuang 050091, China, e-mail: pengliye6@163.com
Qing Liu: School of Artificial Intelligence and Big Data, HeBei Polytechnic Institute, Shijiazhuang 050091, China, e-mail: liuqing652@126.com
Kang Shang: School of Artificial Intelligence and Big Data, HeBei Polytechnic Institute, Shijiazhuang 050091, China, e-mail: shangkang18@126.com

in real time. Particularly in dangerous, complex, and harsh environments, using video technology to supervise and control the current status of work applications can better play the control role of long-distance video technology. With the advancement of science and technology, remote video surveillance has penetrated into almost all industries as an important tool. Therefore, this article takes the traditional manipulator as the background, and details its work in the long-distance video control technology. This subject is of extraordinary significance for the promotion and application of advanced networked automatic control technology [5].

2 Literature review

Remote monitoring is to separate the controller and the controlled object in physical locations and use certain communication means to realize the transmission of control information, video information, and feedback information. Generally, when the operator is far away from the controlled object, remote video surveillance technology is often used [6]. The remote video monitoring system is a remote monitoring system with computer technology as the core, combined with advanced multimedia technology, network communication technology, and digital image compression technology. It is an organic combination of traditional monitoring systems, computer information technology, and network technology. It is an organic whole composed of related electronic equipment and transmission media such as video capture, transmission, monitoring, and system control. It can transmit the monitoring information of the monitoring site to other computers in the network through the computer network, and integrate it with the information management system to achieve the purpose of remote monitoring. Generally, the remote monitoring system includes two parts: local and remote monitoring, which is composed of remote monitoring site and monitoring center. The remote monitoring site and monitoring center are connected by network communication lines [7,8]. Many scholars have carried out various research studies on the topic of remote video surveillance. In terms of long-distance supervision and control of conventional robotic arms, foreign research levels are relatively more active and advanced. A long time ago, some foreign research institutions have created various robot control sites, mainly through the network to effectively control the robot, to achieve some simple operations. Postolache *et al.* completed the Mercury project, which is the first robot to realize remote control based on a web browser, which allows users to log in to the network and remotely operate the robot to complete the excavation task with only a computer keyboard [9]. Hassan *et al.* used the network to manipulate the robot to achieve a series of tasks on the platform. The advantage of this system is that it provides users with a 3D graphical interface and reinforces the significance of the user interface [10]. The Drinking Maiden developed by Majumder and Deen utilizes camera technology and industrial robots to allow people to see the objects in the museum in all directions. There is also a remote system for robotic telescopes, which enables remote users to observe images of celestial objects, compare the images obtained by the telescope with the images in the NASA celestial object library, and then send them to users through the network [11]. The Puma Paint system established by Roy *et al.* is to combine the painting industrial robot with the Internet, so that the robot can choose four colors of red, yellow, green, and blue to draw the specified image in the browser. Monitoring technology has stepped up the pace of research and made some progress [12]. Tom *et al.* used contemporary network technology to remotely monitor the operation of robots as required, and made a good breakthrough for the application of this technology in our country. Although there is still a certain gap between our country's remote control robot technology and the world's advanced technology, many gratifying progress has also been made [13]. The People's Liberation Army Navy General Hospital successfully applied medical robots to perform brain surgery on patients in the operating room, and the experts who gave orders to the robot directed every step of the operation from another building thousands of meters away. The operation used a teleoperated telemedicine robot system jointly developed by the Robotics Institute of Beihang University and the Naval General Hospital. Harbin Institute of Technology takes deep space exploration, dangerous, and extreme environment operations as the background, and carries out delay transmission characteristics, virtual environment modeling, network coordination control, and multi-sensor information fusion for network remote technology under large space delay and Internet transmission delay. Based on the research of related theories and key technologies, a web-based teleoperated robot system and an Internet-based multi-operator multi-robot remote control system have been established. Some high-tech companies are also very optimistic about the prospect of network robots being applied to the service industry and entertainment industry [14]. Mellit *et al.* proposed the design scheme of the network-based robotic arm remote video surveillance system, deeply studied the key technologies needed for the system platform construction, emphatically discussed the design and implementation of each part of the system, and analyzed and discussed for the problems encountered in this process [15].

Based on this research, this article proposes a nonlinear remote monitoring system based on network communication technology. In the Visual C++ environment, transmission control protocol/internet protocol technology builds the remote monitoring system of the robot arm, and mainly realizes the function of the robot arm running according to the track in the long distance. When storing the video, it is necessary to indicate the ID of the video source, *i.e.*, the synchronization source (SSRC) identification robot in the real-time transport protocol (RTP) data header, which can be monitored more clearly. Finally, on the basis of completing the remote video surveillance system, some improvements are proposed to improve the real-time performance and security of the system.

3 Research methods

3.1 Robotic arm control system

The control object of this system is the GRB400 four-degree-of-freedom manipulator produced by Gu Hi-Tech (Shenzhen) Co., Ltd. The four degrees of freedom include three horizontal degrees of freedom and one vertical degree of freedom. The manipulator uses an AC servo motor as the drive source, directly drives the four kinematic joints through the harmonic reducer, and the incremental photoelectric encoder coupled with the motor positions of each axis of the manipulator.

The entire robotic arm system adopts an open motion control architecture. Its hardware consists of three parts: the local computer, the multi-axis motion controller of the manipulator, and the manipulator body. The robotic arm system adopts a hierarchical structure. The local computer, as the main controller, runs the control software platform, provides the user interface and the running trajectory planning, sends control commands, and detects the working state [16,17]. The multi-axis motion controller, as the underlying controller, adopts the open-loop control method to servo-correct the motion of the manipulator. At the same time, the real-time position information of the mechanical joints is fed back. The use of this hierarchical structure design makes each level in the motion control system independent of each other. The connection diagram of the entire robotic arm control system is shown in Figure 1.

3.2 Dynamic analysis of robot arm

3.2.1 Kinematics solution

When the nonlinear kinematics of the manipulator is the positive solution problem, the data in the coordinate space of the manipulator joint are given. Then use a certain relationship to obtain the data of the robot arm in the Cartesian coordinate space. Correspondingly, the kinematics inverse solution problem of the manipulator is to give the coordinates of the manipulator in the rectangular space. According to the corresponding relationship, the joint coordinates of the robot arm are obtained. Joint coordinates are used in the trajectory planning or motion

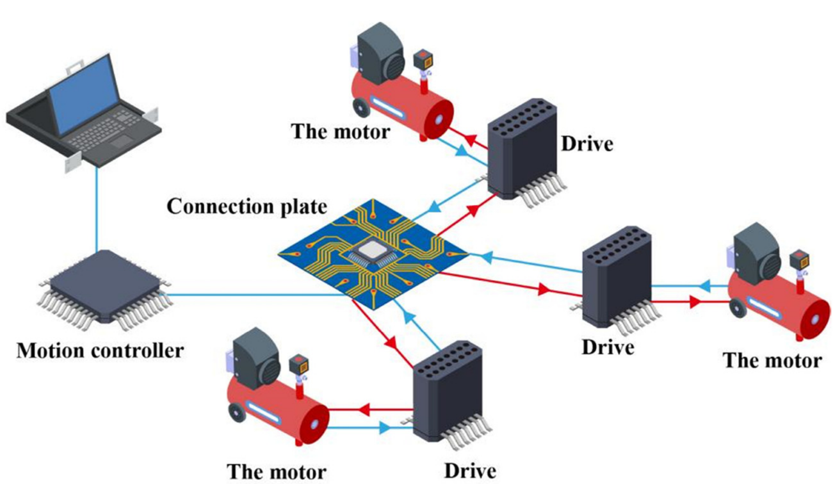

Figure 1: Schematic diagram of the robotic arm control system.

analysis of the robotic arm. Therefore, the kinematic inverse solution occupies a very important position in the system development of the manipulator. The state of the manipulator joint needs to be measured by an encoder, which uses an incremental encoder, and not an absolute encoder. Therefore, every time you run the program, you first need to return the robotic arm to the zero position. When converting the robot arm in the Cartesian coordinate space and the joint coordinate space, our assumption is that joints 1 and 2 are on a straight line, and their joint coordinates are 0°, which coincide with the X axis, which is convenient for solving. Figure 2 is a schematic diagram of coordinate conversion.

3.2.2 Correct solution to kinematics

The mathematical expression is shown in the following formula:

$$\begin{cases} x = l_1 \cos q_1 + l_2 \cos(q_1 + q_2) \\ y = l_1 \sin q_1 + l_2 \sin(q_1 + q_2). \end{cases} \tag{1}$$

Here x is the X-axis coordinate value of the end of joint 2 in the Cartesian coordinate space; y is the Y-axis coordinate value of the end of joint 2 in the Cartesian coordinate space; q_1 is the coordinate value of joint 1 in the joint coordinate space; q_2 is the coordinate value of joint 2 in the joint coordinate space; l_1 is the length of joint 1, which is 200 mm; and l_2 is the length of joint 2, which is 200 mm.

Knowing the joint coordinate values q_1 and q_2 of joints 1 and 2, and substituting them into Eq. (1), the tail of joint 2 of the manipulator can be solved, *i.e.*, the coordinates of the coordinate point of the tail of the gripper in the XY plane. The above is the process of solving the positive solution of the nonlinear kinematics of the manipulator. The relevant parameters of the manipulator are shown in Table 1.

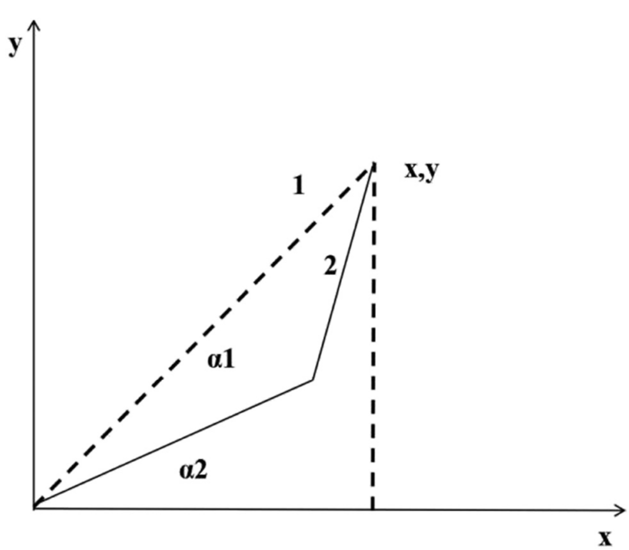

Figure 2: Schematic diagram of coordinate conversion.

4 Analysis of results

In the remote control part of the robotic arm, we adopted a client/server model. Instead of connecting directly with the actual robotic arm, the client uses the server to indirectly control the robotic arm through the network. In this way, the movement of the robotic arm can be controlled correspondingly through the server, so that it can complete the operation task synchronously. Therefore, the main function of the robotic arm remote control software is to allow the server and the client to communicate with each other. The following describes in detail the knowledge required to realize network communication. The simulation environment for the remote control of the robotic arm includes the following:

1. Import the 3D model of the robotic arm
2. Forward/inverse kinematics simulation
3. Collision detection
4. Trajectory planning
5. Forward/inverse dynamics simulation
6. Motion control

4.1 Design of client program

The client must first establish a connection with the server, and then initialize the remote robotic arm through the server. After the robotic arm is successfully initialized, the motion of the robotic arm can be remotely controlled on the client. By sending control commands to the manipulator to make the manipulator perform

Table 1: Parameters of each joint of the manipulator

	Joint 1	Joint 2	Joint 3	Joint 4
Connecting rod length (mm)	200	200		
Range of motion (degrees)	−10 to 190	0–100	−47 to 40	360

the motion, the commands will be displayed in the edit box [18].

Code when sending:

```
voidCZXXDlg::OnBtnSend()
{
//TODO:Addyourcontrolnotificationhandlercodehere
DWORDdwIP;
CStringstrSend;//get the number of bytes to send
WSABUFwsabuf;
DWORDdwSend;//receive the number of bytes actually sent
intlen;//the number of characters used to store in the cstring object
SOCKADDR_INaddrTo;//define address structure
((CIPAddressCtrl*)GetDlgItem(IDC_IPADDRESS1))->GetAddress(dwIP);//IDC_IPADDRESS1 is the ip of the ip address control, first get the pointer that the ip address control points to cwnd // and then call GetAddress to get the ip address
addrTo.sin_addr.S_un.S_addr=htonl(dwIP);//convert to network byte order
addrTo.sin_family=AF_INET;
addrTo.sin_port=htons(2000);
GetDlgItemText(IDC_EDIT_SEND,strSend);//get the number
of bytes to send
len=strSend.GetLength();//the number of characters used to store in the cstr object
wsabuf.buf=strSend.GetBuffer(len);//GetBufferCan convert cstr object to char* return
wsabuf.len=len+1;//add one more to make the receiver end with a /0
SetDlgItemText(IDC_EDIT_SEND,“”);//clear the content of the send edit box
if(SOCKET_ERROR==WSASendTo
(m_socket,&wsabuf,1,&dwSend,0,
(SOCKADDR*)&addrTo,sizeof
(SOCKADDR),NULL,NULL))
//WSASendTo The function is used to send &wsabuf as a pointer to the wsa structure
{
//1 is the actual number of bytes sent (SOCKADDR*) &addrTo is the address structure pointer
MessageBox(“Failed to send data!”);
return;
}
}
```

The implementation code of the message response function is:

```
voidCZXXDlg: OnSock(WPARAMwParam,LPARAMlParam)
{
switch(LOWORD(lParam))//Use the LOWORD macro to take out the low word of lParam to determine which network event occurs.
{
caseFD_READ://Determine whether the network read event FD_READ has occurred,
//Because multiple network events can be requested at the same time
WSABUFwsabuf;///Define the structure variable of WSABUF
WSABUFwsabuf1;
wsabuf.buf=newchar[200];
wsabuf1.buf=newchar[200];
wsabuf.len=200;// its length
wsabuf1.len=200;
DWORDdwRead;//Define a variable to receive the actual read data
DWORDdwFlag=0;//The definition flag is initialized to 0
SOCKADDR_INaddrFrom;//Define the address information received by the address structure variable
intlen = sizeof(SOCKADDR);//Initialize with address structure length
CStringstr;
CStringstrTemp;
if(SOCKET_ERROR==WSARecvFrom
(m_socket,&wsabuf,1,&dwRead,&d
wFlag,(SOCKADDR*)&addrFrom,&len,NULL,NULL))
//WSARecvFrom This function receives the datagram and stores the source address
//The first parameter is the socket descriptor and the second parameter is the structure pointer to wsabuf
//The third parameter is the number of structures provided, here is 1
//(SOCKADDR*)&addrFrom,&len, is the address structure pointer
{
MessageBox(“Failed to receive data!”);//WSARecv FromWSARecvFrom function returns if there is an error
SOCKET_ERROR
//Because this function generally does not fail when called in response to a read event
//The judgment here is to unify the code structure
//delete[]wsabuf.buf;
return;
}
str.Format(“%s”,wsabuf.buf);//strout Only the words after the colon are received
str.Format(“Service-Terminal%s:%s”,inet_ntoa
(addrFrom.sin_addr),wsabuf.buf);
str+=“\r\n”;//add a carriage return
```

GetDlgItemText(IDC_EDIT_RECV,strTemp);//Get the previous data in the receive edit box

str + = strTemp;//make new data plus old data

SetDlgItemText(IDC_EDIT_RECV,str);//Put data into receive edit box

break;

}

}

4.2 Realization of remote video surveillance

Video monitoring is the core module of the remote monitoring system, which is responsible for the remote real-time reproduction of the on-site work of the robotic arm. The quality of video monitoring directly affects the quality of the system. The “client/server” model is widely used in video surveillance. In this mode, the image captured by the camera is read into the computer through the video capture card, which realizes the video capture process on the server side. The collected video is used in two ways, one way is to store it for local viewing and the other way is to compress it and send it to the client. The system uses MPEG-4Codec encoder to compress the video, and after the video data are compressed, it is packaged by RTP protocol, and then sent to the client in the multicast network. After the video data are received by the client, first remove the RTP header, arrange the order, and then decode the video image by the decoder.

4.3 Software implementation of server-side video

The server side of the video surveillance mainly implements functions such as video capture, video encoding, video transmission, and video storage. When the system is developed, the development kit provided by the video card manufacturer can be called to capture the video stream into a file or play buffer. The development kit is in the form of a dynamic library.

4.3.1 Video capture

Using the video capture card development kit, the sequence of functions to be called is shown in Table 2.

When exiting the video capture system, *i.e.*, closing the video surveillance system, the function used is shown in Table 3.

Table 2: Video capture functions

Order of calling functions	Functions	Effect
1	BOOLYH81-PrepareAll()	Used to initialize the entire environment as a whole, this function is used before all function calls
2	BOOLYH-SetVideoFormat(DWORDnChannel , DWORDFoemat)	Used to set the video format, can be used to determine whether the camera is working
3	BOOLYH81-OpenDevice(DWORDnChannel)	Used to open a certain device
4	BOOLYH81-SetVideoHWnd(HWNDhWnd)	Set the video window to put the video into a new window to start the video preview
5	BOOLYH81-StartVideoPreview (DWORDnChannel , RECTWinRECT)	

Table 3: Video exit function

Function prototype	Effect
BOOLYH81-StartEncode(DWORDnChannel)	For compression programming, start compression stream
BOOLYH81-SetEncodeSettings(DWORDnChannel , ENCODE-SETTINGSEncodeSttings)	Used to set encoding settings
BOOLYH81-StopEncode(DWORDnChannel)	Used to stop compression encoding

4.3.2 Video coding

The purpose of setting the video encoding module is to encode and compress the YUV format video stream data sent by the video capture module. In this way, the image will take up a lot less space, which is convenient for transmission on the Internet. The algorithm adopted by this system is to optimize the MPEG-4 video compression algorithm, to collect and compress the high-quality video quality into low-code stream data in real time. The functions used are shown in Table 4.

On the premise of ensuring the bandwidth, increasing the data transmission packet can reduce the transmission delay. Therefore, 32K per packet is selected for compression encoding.

4.3.3 Video sending

Sending the collected video data to the network by multicast is the responsibility of the video sending module. After the video is sent, the client can receive the video and monitor the working condition of the robotic arm in real time. After the video data are obtained in the video capture and compression card, a certain buffer must be set up to store the data. The collected video stream data are continuously pushed into the buffer. Once the buffer is full, the data sending thread is started, the video stream data are popped up, and then sent out by IP multicast. At the same time, the buffer is emptied after receiving the new data. The cycle repeats until the capture or video sending process is stopped.

4.4 Software implementation of video surveillance client

The client of the video surveillance system needs to go through several processes of receiving video data, decoding the video, playing back the video, and storing the video [19,20].

4.4.1 Video data reception

There are three cameras on the robotic arm for real-time video recording, so the video received by the client also has three channels. The multi-thread technology is used to receive and play these three channels of video at the same time, and the system uses four channels of video capture and transmission. In order to better receive and play the video, four user interface threads are established. The process of receiving and playing video is shown in Figure 3.

Receiving RTP video data packets from IP network is the function of video receiving module. If the multi-channel video sent by the server is received at the same time, it can only be viewed after receiving. It is also necessary to decode the received video and mark the unrecorded video data. The SSRC in the RTP packet header information is the main basis for identification [21,22].

```
    //create an IP group socket
    MultiSock=WSASOCKET(AF_INET, SOCK_DGRAM,
IPPRORO_UDP,
    (LPWSAPROTOCOL_INFO)NULL, 0,
    WSA_FLAG_MULTIPOINT_C_LEAF|
WSA_FLAG_MULTIPOINT_D_LEAF);
    //set the server data buffer to 32K
    constintPACKET_SIZE_32K=4;
    status=YH_ServerSetBuffer
(intnPackSize,intnPackNum,boolbFullSize);
    //connect to a server to join a multicast group
    YH_ClientConnectServer(char*pIp, intnPort);
    stDestAddr.sin_family=PF_INET;
    stDestAddr.sin_port=htons(201);
    stDestAddr.sin_addr.s_addr=inet_addr("226.1.1.1");
    hNewSock=WSAJoinLeaf(MultiSock,(PSOCKADDR)
&stDestAddr,sizeof(stDest
```

Table 4: Video encoding functions

Function prototype	Effect
BOOLYH81-StartEncode (DWORDnChannel)	For compression programming, start compression stream
BOOLYH81-SetEncodeSettings (DWORDnChannel , ENCODE-SETTINGSEncodeSttings)	Used to set encoding settings
BOOLYH81-StopEncode (DWORDnChannel)	Used to stop compression encoding

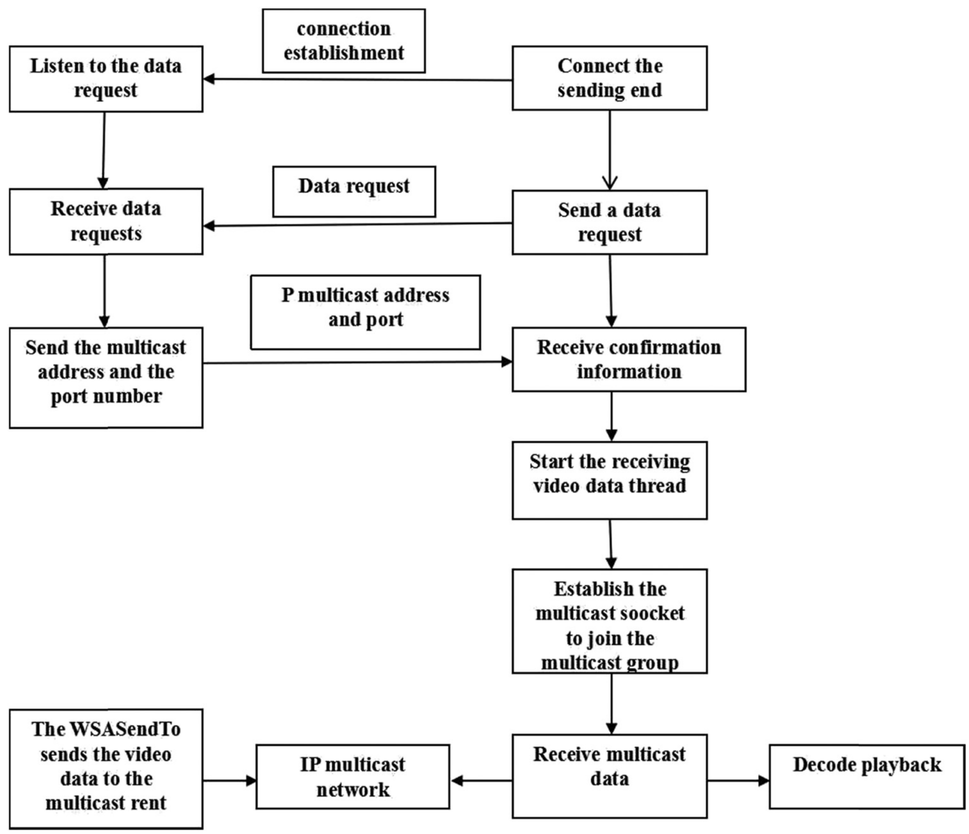

Figure 3: The process of receiving and playing video.

Addr),NULL,NULL,NULL,NULL,JL_RECEIVER_ONLY);

//receive multicast data and store it in the buffer stWSABuf

Status=WSARecvFrom(MultiSock, &stWSABuf, 1, &cbRet, &cbRet, &dFlag,

(structsockaddr*)&stSrcAddr, &iLen, NULL, NULL);

4.4.2 Video decoding and playback

The received video or locally stored video is decoded to form an image frame stream, which can be played in the video playback module. It is the function of the video playback module to convert the image in YUV format into an image in RGB format, which is a necessary way to realize image reproduction [23,24]. The functions used for video decoding and playback are shown in Table 5.

4.4.3 Video storage

After the video image data are demultiplexed, several video sources will appear. When storing the video, it is necessary to specify the ID of the video source, *i.e.*, the SSRC identifier in the RTP data packet header. The related functions are shown in Table 6.

Table 5: Video decoding functions

No.	Function	Effect
1	YH81-Creat()	Create a playback/network receiver
2	YH81-OpenFile()	Open play file
3	YH81-Play()	Start playing the file, or resume normal playback
4	YH81-Stop()	Stop play
5	YH81-Pause()	Pause playback
6	YH81-GetCurrentFrameRate()	Get the frame rate of the current code stream programming

Table 6: Video storage functions

No.	Function prototype	Effect
1	YH81-ReadPacket (DWORDnChannel, UCHAR&PacketBuffer, DWORD&ReceivedPacketSize)	Get packets
2	YH81-GetSystemHeader (DWORDnChannel, UCHAR**pSystemHeader, int&Length, UCHAR*pData)	Add a file header to the saved file, please write this header before saving a new file each time
3	BOOLYH81-StopReadPacketMsg (DWORDnChannel)	Stop getting packets

5 Conclusion

The realization process of the remote video monitoring system of the degree-of-freedom manipulator discusses the functions realized by the manipulator body, network communication, video monitoring, and measures to improve the performance of the system. The hardware structure of the GRB four-free manipulator involved in the system is introduced, mainly the communication problem in the C/S mode. The basic communication function and remote control function are realized.

Funding information: The authors state no funding involved.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and approved its submission.

Conflict of interest: The authors state no conflict of interest.

References

[1] Antonin J, Zheng R, Tsutomu K, Kimihiko N. Effect of drowsiness on mechanical arm admittance and driving performances. IET Intell Transp Syst. 2018;12(3):220–6.
[2] Jia Y, Zhang X, Wang Z, Wang W. Intelligent calibration of a heavy-duty mechanical arm in coal mine. Electronics. 2020;9(8):1186–96.
[3] Jiang A, Yao X, Zhou J. Research on path planning of real-time obstacle avoidance of mechanical arm based on genetic algorithm. J Eng. 2018;2018(16):1579–86.
[4] Chen W, Li Z, Cui X, Zhang J, Bai S. Mechanical design and kinematic modeling of a cable-driven arm exoskeleton incorporating inaccurate human limb anthropomorphic parameters. Sensors. 2019;19(20):4461–71.
[5] Cheng LC, Huang DP, Ling ZY, Geng Y, Da Q, Lv CL. Study on mechanical properties of steel on transmission tower under icing load. Key Eng Mater. 2018;773:255–9.
[6] Grubic T. Remote monitoring technology and servitization: exploring the relationship. Comput Ind. 2018;100(1):148–58.
[7] He L, Hao L, Qiao W. Remote monitoring and diagnostics of pitch-bearing defects in an MW-scale wind turbine using pitch symmetrical-component analysis. IEEE Trans Ind Appl. 2021;57(4):3252–61.
[8] Honeycutt WT, Kim T, Ley MT, Materer NF. Sensor array for wireless remote monitoring of carbon dioxide and methane near carbon sequestration and oil recovery sites. RSC Adv. 2021;11(12):6972–84.
[9] Postolache O, Alexandre R, Geman O, Hemanth DJ, Khanna A. Remote monitoring of physical rehabilitation of stroke patients using IoT and virtual reality. IEEE J Sel Areas Commun. 2020;39(2):562–73.
[10] Hassan MK, Desouky AE, Elghamrawy SM, Sarhan AM. A hybrid real-time remote monitoring framework with NB-WOA algorithm for patients with chronic diseases. Future Gener Comput Syst. 2019;93(Apr):77–95.
[11] Majumder S, Deen MJ. Smartphone sensors for health monitoring and diagnosis. Sensors. 2019;19(9):2164–74.
[12] Roy K, Jana S, Ghosh SK, Mahanty B, Mandal D. 3D MOF assisted self-polarized ferroelectret: an effective auto-powered remote healthcare monitoring approach. Langmuir. 2020;36(39):11477–89.
[13] Tom M, Prabha R, Wu T, Baltsavias E, Schindler K. Ice monitoring in Swiss lakes from optical satellites and webcams using machine learning. Remote Sens. 2020;12(21):3555–65.
[14] Chen Y, Tan K, Yan S, Zhang K, Zhang H, Liu X, et al. Monitoring land surface displacement over Xuzhou (China) in 2015–2018 through PCA-based correction applied to SAR interferometry. Remote Sens. 2019;11(12):1494.
[15] Mellit A, Benghanem M, Herrak O, Messalaoui A. Design of a novel remote monitoring system for smart greenhouses using the internet of things and deep convolutional neural networks. Energies. 2021;14(16):5045–55.
[16] Fei X, Long F, Li F, Ling Q. Multi-component fusion temporal networks to predict vehicle exhaust based on remote monitoring data. IEEE Access. 2021;9:42358–69.
[17] Jerwood K, Lowy P, Deeming L, Kariuki BM, Newman PD. Remote control: Stereoselective coordination of electron-deficient 2,2′-bipyridine ligands to Re(I) and Ir(III) cores. Dalton Trans. 2021;50(45):16459–63.
[18] Junior A, Munoz R, Quezada A, Neto A, Albuquerque V. Internet of water things: a remote raw water monitoring and control system. IEEE Access. 2021;9(99):35790–800.
[19] Si H, Sun C, Chen B, Shi L, Qiao H. Analysis of socket communication technology based on machine learning algorithms

under TCP/IP protocol in network virtual laboratory system. IEEE Access. 2019;7:80453–64.
[20] Li P, Xiao Z, Chen Z, Wang Y, Wu X, Li D, et al. The research on communication technology matching of power terminal communication access network. Procedia Comput Sci. 2019;155:785–90.
[21] Bian D, Kuzlu M, Pipattanasomporn M, Rahman S, Shi D. Performance evaluation of communication technologies and network structure for smart grid applications. IET Commun. 2019;13(8):1025–33.
[22] Sharma S, Roy SD, Kundu S. Secrecy outage of a multi-relay cooperative communication network with accumulation of harvesting energy at relays. IET Commun. 2019;13(18):2986–95.
[23] Kulchin YN, Voznesenskii SS, Gamayunov EL, Golik SS, Subbotin EP. Photonic methods and technologies for monitoring the ocean and atmosphere. Quantum Electron. 2020;50(5):475–88.
[24] Livi LF, Catani J. A new remote monitor and control system based on Sigfox IoT network. Rev Sci Instrum. 2021;92(9):094705.