ISBN: 978-981-18-3959-7
2022 the 12th International Workshop on Computer Science and Engineering (WCSE 2022)
doi: 10.18178/wcse.2022.06.004

Design and Implementation of Mobile Surveillance System

Shaowei Li [1]and Cairong Xiang [1 +]

[1]School of Artificial Intelligence, Jianghan University, China

Abstract. In order to expand the video monitoring range and flexibility of a single camera, a remote-control mobile video monitoring system is designed. The system is composed of four modules. The smart car based on the Arduino system is equipped with a camera to receive user instructions for mobile video acquisition; the embedded Linux system realizes real-time video data acquisition through v4l2 interface. Meanwhile, it sends the data to the forwarding server through the network, on the other hand, it forwards the control commands from users to the smart car; the server is used to forward the video to the client and forward the user control instructions to the Linux system. Android based mobile terminal presents monitoring video and provides user control interface. Compared with the existing system, the new system can achieve monitoring without blind spot by using one single camera.

Keywords: mobile monitoring, V4L2 interface, server, mobile terminal

1. Introduction

Traditional video surveillance systems have problems such as high cost of use and complicated wiring. It has become an urgent need to design a video surveillance system with excellent performance, flexible use and low cost. Literature [1] proposed a remote video monitoring system based on the C/S mode, which uses specific hardware to complete the collection and encryption of video data, and transmits the data to a remote server through a specific network link.; Huang Xin, Liang Yangyang [2] Embedded system based on ARM+Linux and combined with V4L2 interface completes the collection of video data. At the same time, a web server is installed in the system to realize remote video access based on web; Chen Guojun, Zhuo Xuanyu [3-5] and others proposed a video surveillance mode of ARM+Linux+mobile smart car, which can realize the monitoring of different scenes by a single camera by controlling the movement of the smart car; Bu Zhenjiang et al. [6-9] proposed a mobile video monitoring client based on Android. The video data is obtained in real time by connecting to the streaming media server, and the received video data is displayed on the Android mobile phone client at the same time; Kai Zhou, You Hongyuan [10] designed a mobile video surveillance system based on Raspberry Pi, but the collected data is only It can be sent to a device with a designated IP; the Android-based video surveillance system proposed by J. Azeta [11] and others can control a smart car equipped with a camera through a mobile phone, but the mobile phone and the smart car control system must be in the same WIFI network .

Based on the analysis of the above-mentioned existing video surveillance systems, this article proposed a mobile remote video surveillance system. The camera in the system is no longer fixed at a certain position, but installed on a mobile smart car that can be operated remotely, which greatly increases the video surveillance range; using the V4L2 interface and socket technology in the Linux system, it is extremely convenient The video data collection and transmission; use the powerful processing capabilities of the desktop to realize data forwarding, storage and client connection, which facilitates the access of any network device to the video surveillance system; the application running on the Android/desktop system allows the user to the video can be accessed at all times, and the movement of the smart car can be controlled. The system improves the flexibility of video data collection, at the same time, simplifies the deployment of cameras, improves system scalability, and realizes remote access to the monitoring system from the mobile terminal.

+ Corresponding author. Tel.: +86 18171385977.
E-mail address: 55440881@qq.com

2. System Model

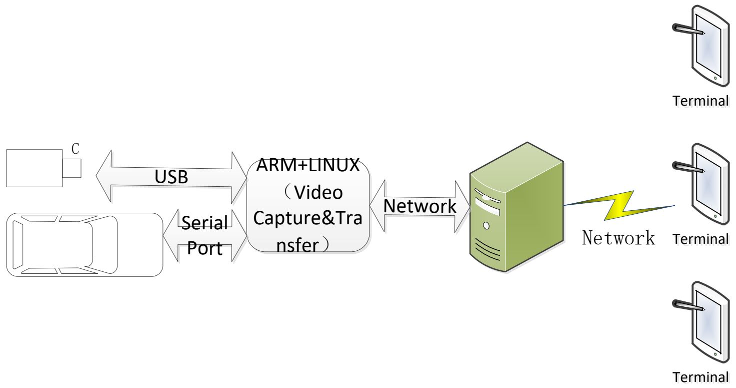

Fig. 1 System structure diagram.

As can be seen from Figure 1, the entire system is composed of a vehicle-mounted mobile camera, a video capture system, a server, and a client terminal. Its composition and functions are as follows: 1. The camera is mounted on a crawler smart car, and the Arduino system receives user control instructions through the serial port to control the movement of the smart car; 2. The ARM system with embedded Linux realizes the functions of video data collection, forwarding and receiving control commands;3. The data server is responsible for the function of connecting the video acquisition system and the client; 4. The APP running on the Android terminal is connected to the data server through the network, displays the received video data in real time and provides a user control interface to operate the movement of the smart car.

3. Key Technologies

3.1. Protocol for Commands

Table 1 lists the control command protocol used in the system. The instructions are sent from the client terminal via the server to the video acquisition system through the TCP channel to realize the control of vehicle movement, the adjustment of camera parameters, and user login.

From the control protocol in table 1, it can be seen that the camera parameters only include brightness and contrast adjustment, and the value range is "0-99". Except for the 0x10 at the end, which is a hexadecimal number, the other parts are represented by ASCII code.

Table 1: Control commands

Commands	Description
$CMD,VEH,0,0x10	Forward
$CMD,VEH,1,0x10	Backward
$CMD,VEH,2,0x10	Left
$CMD,VEH,3,0x10	Right
$CMD,VEH,4,0x10	Stop
$CMD,CAM,0,***,0x10	Set Brightness to ***
$CMD,CAM,1,***,0x10	Set Contrast to ***
$CMD,LOGIN,0x10	Login
$CMD,LOGOUT,0x10	Logout

3.2. Vehicle Control System Based on Arduino

The smart vehicle is driven by the motors, and the steering of the vehicle is controlled by the differential speed of the left and right wheels, and its movement is controlled by the instructions of the host computer.

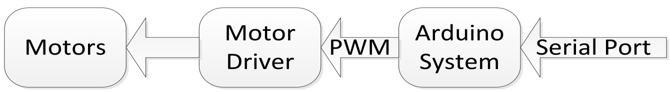

Fig. 2: Control flow for smart vehicle.

Figure 2 shows the model of the smart vehicle motion control structure. The Arduino system and the ARM-based video capture and forwarding system are connected through a serial port and exchange data; once the Arduino control board receives the instruction from the video acquisition system, it will generate PWM signals to control the work of the left and right motor drivers.

3.3. Video Data Splitting and Transmission

For the embedded ARM+Linux system used in this system, the maximum data transmission volume of UDP protocol data packets at one time is about 50KB. For some devices, it is impossible to transmit one frame of collected data to the server at a time. Since the video capture and the server are in the same local area network, the data transmission rate is high and the error rate is low, so this system uses the simple protocol shown in table 2 to mark the beginning of the frame data and divide the data into fixed-size blocks.

It can be seen from table 2 that the data between the two data headers is a complete image data frame; the size m of the data block is determined in practical applications according to system resources and network transmission speed.

Table 2: Protocol for video data splitting

	Description
Header: $HDR,0x10	Packet Header, First Data Block is Coming
Data Block Size	mKB
Last Data Block Size	S-n*mKB

3.4. Network and Data Transfer

According to the system network structure shown in figure 3, the data transmission between the various modules in the system (except the Arduino system that controls the smart vehicle) is completed through the TCP/UDP network.

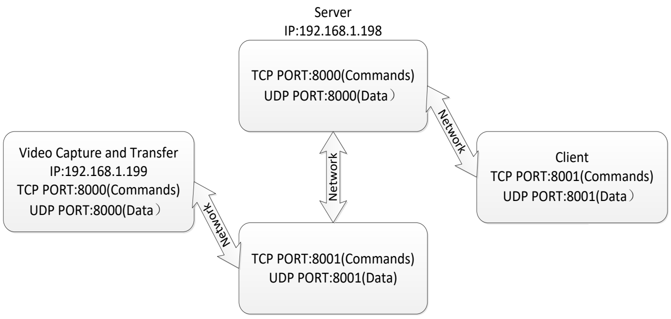

Fig. 3: Network structure.

1) The ARM-based video capture and forwarding system uses TCP protocol to monitor and receive instructions from the client;
2) The client uses TCP and UDP protocol (8001 port) to connect to the server, send control commands and obtain video data at the same time;
3) The function of the server is mainly used for the forwarding of instructions and video data. The server uses the linked list structure shown in Figure 4 to maintain client information.

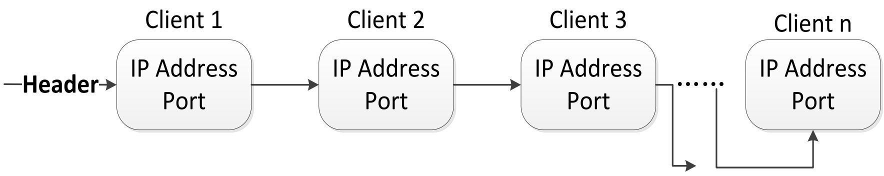

Fig. 4 Chain structure for clients

4. System Design

4.1. Parameters for Video Capture

Since the embedded Linux system is installed in the video acquisition system, the V4L2 interface can be used to conveniently complete the video data acquisition and camera parameter adjustment. According to the Linux development document, combined with the video data collection and parameter setting workflow shown in figure 5, the video data collection can be completed quickly.

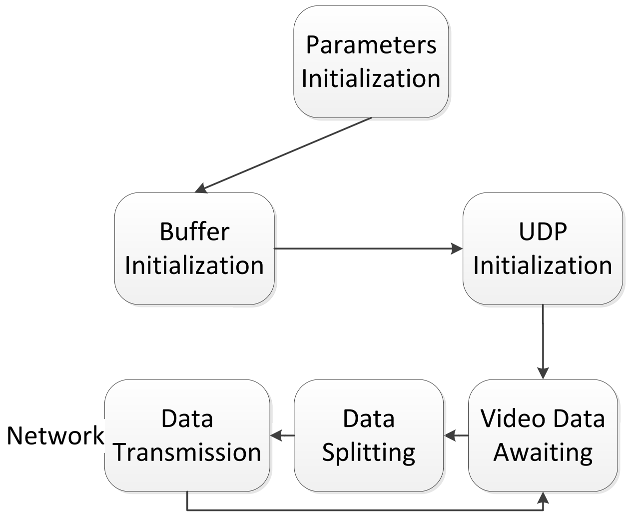

Fig. 5: Working flow for video capture

The captured video parameters mainly include image resolution, video format and other information, and the parameter information is stored in the v4l2_format data structure; buffer initialization requires specifying the number of buffers, buffer types, capture modes, and memory area usage. The parameters are stored in the v4l2_requestbuffers data structure. According to the selected hardware, the system selects the parameters listed in Table 3 to complete the collection of video data.

Table 3: Parameters for Video capture

Parameters	Value
Resolution	640×480
Format	MJPEG
Memory Access	Mapping
Number of Buffers	4
Size of Data Block	50KB

4.2. Data Storage and Display

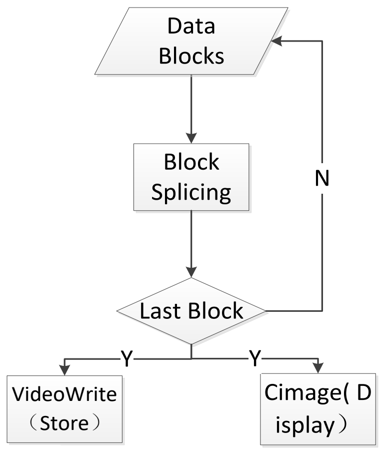

Fig. 6: Video storage and display.

The server locally creates two frame buffers for receiving data blocks. When the server receives the data, it will choose to store it in the free buffer. At this time, the server will respond to the data in the other buffer. The effective data is stored and displayed. The above-mentioned operation process is shown in detail in figure 6.

After the connection, customer related information is displayed in the client login interface shown in Figure 7. It should be noted that the client can only connect to one of the channels as needed.

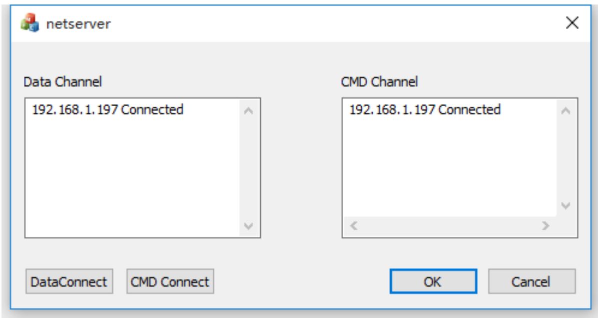

Fig. 7: Client login on server.

4.3. Client Control and Display

According to the program workflow mentioned above, Unity is used as a mobile terminal development tool, so that the program has cross-platform characteristics and can run on operating systems such as Android, IOS, Windows, and Linux.

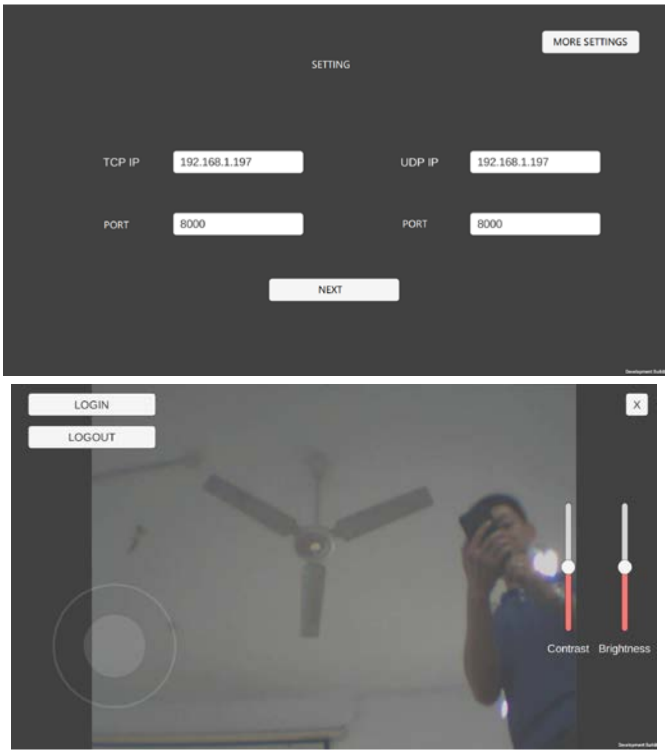

Fig. 8: Operation interface for mobile terminals.

In this system, the program uses the Android system as a test platform, installs the obtained APK file into the mobile client terminal, and obtains the mobile terminal display interface shown in figure 8. The user enters the server IP address and port in the first configuration interface, and after completion, you can enter the second video display and operation interface; in the video display and operation interface, the user clicks LOGIN to complete the login, and the system can realize the video Display and control the sending of commands.

5. System Implementation

In order to verify the practicability of the system and analyze the operating efficiency of the system at the same time, the collection part of the system (crawler with video collection system), data server and mobile client are all deployed in the laboratory of Jianghan University. The crawler-type mobile remote video surveillance system shown in figure 9.

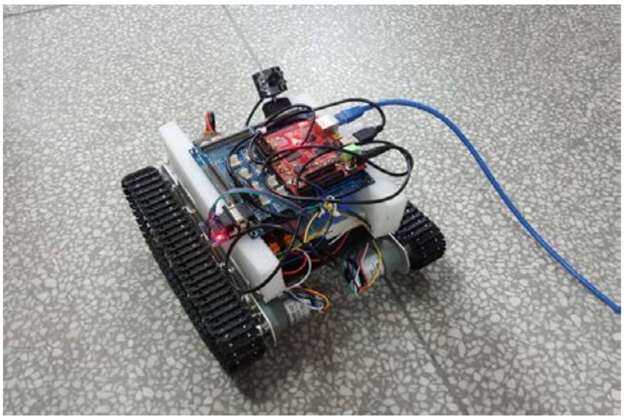

Fig. 9: Mobile surveillance system.

6. Acknowledgements

I would like to acknowledge the invaluable help of my colleagues, who have given my constant consultant in my paper writing, and have supported me throughout the whole process of the papers; and appreciate financial support from Jianghan University (Fund No. 2021yb053).

7. References

[1] LIU Yan, WANG Xue － mei, TAO Jun et al. The Design and Implementation of Remote Network Video － monitoring System Based on C/S Mode[J]. Journal of Changchun Normal University, 2019,38(6):30-34.

[2] Huang Xin,Liang Yangyang. Embedded Video Monitoring System Based on V4L2[J]. INSTRUMENTATION, 2017,24(5): 46-50.

[3] Chen Guojun, Tang Xueqi, Wu Yuwan. Mobile surveillance vehicle Based on ARM and Linux[J]. Computer Knowledge and Technology, 2016,12(34): 81-83

[4] Zhuo Xuanyu, Tang Lin. Wifi Surveillance Vehicle Based on STM32[J]. Technology and Information, 2018(3):98

[5] Zhang Lu, Yao Xiangguo. Mobile Surveillance Vehicle Based on Wifi[J]. Technology Application, 2018(6):88-90.

[6] BU Zhen-Jiang, YANG Hai-Bo, JIA Jun-Ying. Mobile Video Monitoring Terminal Based on the Android System[J]. Computer System Applications, 2017,26(4): 275-279

[7] LIU Jiangping， ZHANG Lei. The Research of Mobile Phone Video Surveillance Based on Android[J]. Journal of Inner Mongolia Agricultural University (Natural Science Edition), 2017,38(6):68-72

[8] Wang Yan. Design of Mobile Video Surveillance Client Based on Android Terminal[J]. Microcomputer Application, 2019,35(7): 105-107

[9] Wu Yaqin, Lu Zhangyichan, Zhang Hongna et al. Design and Implementation of Surveillance System Based on Smartphone[J]. Intelligent Processing and Application, 2018(2): 85-87

[10] Kai Zhou, You Hongyuan. A Smart Ammunition Library Management System Based on Raspberry Pie[J]. Procedia Computer Science, 2020,166:165-169

[11] J. Azeta; C.A. Bolu; D. Hinvi et.al. An Android Based Mobile Robot for Monitoring and Surveillance[J]. Procedia Manufacturing, 2019,35: 1129-1134