Research on the Wireless Video Surveillance System Based on Linux

Xiao-jun Liu¹, Hong-yuan Wen¹, Xiang-zhen Zhou¹ and Di Xu¹

¹School of Intelligent Manufacturing, Taizhou Institute of Sci.& TECH., NUST., Taizhou, 225300, Jiangsu Province, China
93429706@qq.com

Abstract. With the new development of embedded technology, image compression and wireless transmission technology, wireless video surveillance system based on embedded system has become the tendency of development in the field of video surveillance. This paper combines the character of embedded system with wireless transmission technology, researches and designs a solution of wireless video surveillance system based on Linux. The embedded video surveillance system uses C/S framework. The server based on ARM11 and Linux is designed to process video image including capture, preview, compression and transmission. The PC as a client is built to receive, decode, display and setup the video.

1. Introduction
With the establishment and proposal of various wireless communication standards and the increasingly mature related technologies, it is possible to develop an embedded video monitoring system based on wireless transmission. Traditional network video monitoring system usually adopts wired transmission for video, which has disadvantages such as difficult wiring, poor flexibility, high cost and difficult maintenance. Wireless video monitoring just makes up for these shortcomings [1]. This paper proposes a wireless transmission video system based on Linux, based on wireless standard protocol of IEEE 802.11b/g, which can display and control video collected by remote terminal [2].

2. Overall structure of the system
This system adopts the two-layer structure of client-server model design. Embedded video monitoring system based on C/S structure is mainly divided into video monitoring server and video monitoring customer service end, which communicate through wireless network [3]. Video surveillance server includes embedded microcontroller processor, USB camera, the wireless network card and other peripheral hardware and auxiliary equipment, and carry on the hardware platform embedded operating system and run the corresponding application, including Bootloader, inner nuclear file system level software and root file system composition, good environment for the upper applications run; The upper layer of the application sequence to achieve video image acquisition, preview, compression and transmission functions; Video monitoring client is an ordinary PC, which installs Linux system and runs the corresponding client program under the virtual machine environment to realize the functions of image reception, display and parameter setting. The overall structure of the system is shown in figure 1.
Hardware platform
(Embedded microprocessor, USB camera, wireless network hardware platform card, LCD touch screen, etc)

Bootloader
System kernel, driver

Application
(video acquisition, transmission, etc.)

User level
software platform

File system
System level

WLAN network

Video acquisition server

Video monitors the client

Application (video receiving, displaying, parameter setting, etc.)

Common operating system (Linux virtual)
The PC

Figure 1. Overall structure of the system.

3. System hardware design
According to figure 1, the hardware design is mainly the hardware platform of video acquisition server, and its hardware structure is shown in figure 2. Among them, the embedded microcontroller processor is the core, take the program instruction execution, Program (Bootloader and kernel file, the root file system, etc.) of the storage space, the LCD for the preview image is used, the Wi-Fi module is used to send and receive data, RS232 serial port is used to connected to the PC, in the early stages of the program development, used as a debug port, SDRAM memory is equivalent to the desktop, as the program runs used to store temporary data.

The camera used in the system adopts Tianmin "small wasp" camera, with the maximum resolution of 640x480 and USB2.0 interface. The photosensitive element is CMOS, which supports capturing and storing static photos (BMP, JPG format).

Video acquisition server and video monitoring client adopt Wi-Fi wireless communication mode. This system adopts WM-G-MR-09 Wi-Fi module produced by Feiling company, adopts SDIO interface, and the data transmission rate reaches 54Mbps, fully meeting the requirements of video transmission.

In terms of storage, the system integrates 2GB NandFlash and 256MB SDRAM, which can fully meet the storage and operation of system programs.
4. System software design
The client part of the software is based on the graphical interface application development, does not involve the underlying part of the system; the video frequency acquisition server is a typical embedded system. The software includes system level and user level.

4.1. Server system-level software migration
The transplantation of system-level software mainly includes three parts: Bootloader, kernel and root file system. The Bootloader adopted in this paper is U-boot (Universal boot Loader), because this Bootloader already supports various embedded CPU architectures, and its compilation process can use graphical interface to facilitate configuration and migration. Kernel transplantation is to generate the final usable kernel image file by configuring and compiling the original Linux kernel file according to the specific target board hardware environment. What is emphasized here is that the corresponding camera and WLAN wireless network card model support are needed to configure the kernel. Making a root file system is all about creating function directories and adding necessary function files to those directories. This article Uses the Busybox tool to create the root file system. Busybox includes simple utility commands that make it easy to create YAFFS2 file systems.

4.2. Video server software design
1) Video image acquisition based on V4L2
V4L2 is the interface specification for video device drivers on Linux systems. It is a set of predefined functions designed to provide application developers with the ability to access a set of routines based on a piece of software or hardware without having to access the source code or understand the details of the internal workings. This paper encapsulates a set of V4L2 interface, including video acquisition device initialization, cancellation of camera head device, acquisition and setting of camera parameters, as well as start, terminate and process the acquisition of images, user control and other related functions. The video acquisition process based on V4L2 is shown in figure 3.

![Figure 3. Collection flow chart of API function video](image_url)
2) Encoding and decoding based on JPEG video

The original image data obtained from the camera is relatively large, which needs to be compressed and then transmitted to the client. S3C6410 processor with JPEG hardware encoder is easy to use, high efficiency.

The following is a description of the encoding and decoding process of video image:

(1) JPEG video encoding

- First call `jpg_enc_init` to initialize video encoding, `jpg_enc_create` to create JPEG encoding objects.
- Call `jpg_enc_yuyv_frame` to create an encoded video frame structure, where the buffer holds the address of the input buffer used to store the data to be encoded. This data must be in cbcr4:2:2 format, otherwise the encoding will fail. The width, height and pixel of video image need to be passed in corresponding parameters to allocate space. Generally, it is set as width *height * BPP. In this paper, the width *height * BPP is 480, 272 and 255 bit color images, which can be set as 480, 272, 3.
- Before coding, `jpeg_set` function can be used to set relevant parameters, such as image height, image width, image compression rate.
- After successful encoding, the output buffer address can be obtained by calling `jpg_enc_get_outbuf`, and the content is written to the file as a complete JPEG image.
- Finally, `jpg_enc_free` is called to free the object resources.

The JPEG video coding flow is shown in figure 4.

(2) JPEG video decoding

- First call `jpg_dec_init` to initialize, `jpg_dec_create` to create the JPEG encoding function.
- Call `jpg_dec_frame` to create an encoding framework, where `out_buf` stores the address of the input buffer obtained, the same address is used to store the data to be decoded, or the JPEG image file data can be read, and then directly into the buffer with `jpeg_mem_src`.
- Before decoding, you can also set some parameters, such as encoding format `out_color_space`, etc.
- The `jpeg_start_decompress` function can then be called to begin decoding, which sets `output_width`, `output_height`, and `linesize` for the output image.
- YUV conversion to JPEG format.
- After successful decoding, call `jpg_dec_get_outbuf` to get the address of the output buffer, which contains image data that can be directly used for display, and its format is determined during encoding. Call `jpg_dec_get_framsiz` to get the data.
- Finally, call `jpg_dec_free` to free the resource.

The JPEG video decoding flow is shown in figure 5.
4.3. Video client software design
The client is mainly used to send connection requests to the server. After receiving the information returned by the server, it establishes connection with the server and receives the video image frame sent by the server and displays it on the monitoring interface. The video image display process is shown in figure 6. First, create the client object and carry on the related initialization work, such as thread creation; Then according to the input IP address and PORT number to the server to issue a connection request; after the connection is established, the receiving thread will receive video image data from the server through the network. Then the video data is decompressed and displayed in the graphical display area. The program continues to receive video data and extract the display until the user disconnects.
5. Test results
After the system development is completed, the test platform is built to verify the performance. Figure 7 is a preview of the server LCD screen. Figure 8 shows the video monitoring interface of the client, which is divided into four areas: video monitoring area, information area, photo button, parameter setting switch button and video parameter setting area. Video monitoring area displays real-time image information collected by the server; the information area describes related image information and auxiliary information. The parameter setting area can adjust the relevant parameters of video image in real time, such as brightness and contrast.
6. Conclusion
After testing, the whole system runs normally and reliably, video server can correctly identify the camera and wireless network card, realize stable video collection, preview and transmission functions. Client monitoring software can complete real-time video display, video image is clear and smooth, but also has parameter adjustment and snapshot function. The design cost of the whole system is low, the mobility is good, the use is flexible, to meet the design requirements of this paper.

References
[1] Zong Wenguang. Application of 3G wireless video monitoring system in power rush repair [J]. Science and technology communication, 2010, (21): 166-167.
[2] Yuan Xindi, Li Qiusheng. Embedded Wireless Video Data Acquisition System Based on RT5350 [J]. Journal of Anhui University(Natural Science Edition) , 2014, 38(06): 60-65.
[3] Zeng Xinhui. Design of wireless video monitoring system for android client [J]. Modern electronics technique, 2016, 39(12): 6-9+13.
[4] Zhao Chunyuan, Li Meng, Han Huishan, Liu Fucai. Design and implementation of wireless video monitoring system based on ARM9 [J]. Computer engineering and design, 2012, 33(02): 529-534.
[5] Xu Lin. Research and implementation of embedded wireless video monitoring system based on ARM11 [D]. Wuhan: Wuhan university of science and technology, 2012.