JOURNAL OF INFORMATION PROCESSING SYSTEMS **JIPS**

An Embedded Multifunctional Media System for Mobile Devices in Terrestrial DTV Relaying

Jun Huang* and Haibing Yin**

Abstract

The paper presents a novel embedded multifunctional media sever (EMMS) for mobile devices to receive various media programs. Being different from other contemporary system research, the paper mainly studies how to design a reception solution for terrestrial digital television (DTV) on mobile devices and how to enable mobile devices can receive DTV program, enjoy video-on-demand (VOD), achieve video surveillance and relay Internet video program via local Wi-Fi simultaneously. In the system design, we integrate broadcasting-terrestrial DTV tuner, streaming media re-transmission system, VOD disk, video camera and access interface to the Internet into EMMS, which can either receive terrestrial DTV radio signals and demodulate out digital transport stream (TS), or can read streaming media bit-stream from VOD disk, surveillance camera or access interface to the Internet. The experimental results show the proposed system is stable and quality-efficient. Comparing with the other systems, the proposed system has the least packet loss rate and response time.

1. Introduction

Digital broadcasting terrestrial systems, such as DVB-T [1] in European, ISDB-T in Japan and DTMB [2] in China, etc., had been built up and popularized for application in the world today. At the same time, mobile digital TV systems, such as T-DMB [3] in Korea, ISDB-T in Japan and CMMB [4] in China, had also been applied in a wide range. There are three ways for terrestrial and mobile DTV program reception. The first is the dedicated portable receiver, such as receivers on public transport facilities. The second is specific set-top box receiver which is usually used at home. USB interface based DTV receiver is the third way. The dongle-type receiver terminal is connected with personal computer (PC) by USB bus to offer bit-stream to PC. Transport stream (TS) demultiplexer, video/audio decoding, and rendering are implemented by software running on PC.

Several DVB-T systems for digital terrestrial TV have been proposed in recent years, such as the front-end module, demodulator module, decoder module [5,6], and the whole receiver system [7,8].

In addition, there are some research reports on embedded multimedia home gateway [9] or media

server solution [10]. In [11], a new media server with DTV reception called Wi-Fi TV was presented, in which a small stand-alone TV tuner was firstly combined with home Wi-Fi router and then connected worldwide TV signals and streams of live TV to mobile devices. In some Blu-ray recorders, the ISDB-T DTV bit stream was allowed to be recorded and relayed to mobile devices by local Wi-Fi network, such as the JVC recorder [12]. In [13], a method was proposed to enable ATSC Mobile DTV standard based digital broadcasting in multiple mobile devices. The implemented solution used Wi-Fi network and supported iOS and Android OS based platforms. Similarly, a dongle-type ASTC broadcasting DTV receiver was showed in [14]. Pan et al. [15] presented a multimedia gateway system, which combined the function of digital television signal re-transmission and video monitoring innovatively.

Most traditional solutions such as dedicated DTV receiver, set-top box DTV receiver and dongle-type DTV receiver were designed for some specific application cases. The primary function of digital DVB-T ISDB-T in [14] is to watch and record digital terrestrial TV on PC or Laptop. In [16], the authors utilized the time-domain PN sequence in the TDS-OFDM system and only applied it to the DTMB system. An infrastructure only for indoor spaces application was presented in [17], which consisted of mobile spectrum sensors that were built by DTV receiver dongles interfaced with Android-based mobile devices and a cloud-based central server to manage such sensing devices. Zhu [18] designed a streaming media data forwarding platform only for compatible with multiple products in the field of video surveillance.

However, there are some limitations of the above systems or technologies. Firstly, most of these systems and technologies were developed mainly for some specific applications or designed for some technical problems, which were lack of integrity and versatility. Secondly, the traditional DTV receiver mainly uses a specific terrestrial DTV terminals to receive and renderer the DTV programs, and how to design a reception solution for terrestrial DTV on mobile devices is less studied, which greatly limited application of receiving and rendering the DTV programs on mobile devices, especially today in the era of the smart mobile phone; Thirdly, most of the digital broadcasting system mainly realized the DTV signal forwarding and reception, without considering integration with such as online video monitoring, VOD and other functions.

In respect of the above issues, the paper presents a novel EMMS for mobile devices in terrestrial DTV relaying, which has a deep research in re-transmission system design, EMMS hardware and software architecture. The main works and contributions are as follows.

1) Designing a re-transmission system for re-transmitting terrestrial digital television signals for smart mobile devices and presenting a method for acquiring and collecting tuning information and program-related information from broadcast signals, which make the system more versatile. The proposed method mainly follows. Firstly, the tuner transfers the radio frequency (RF) signal to the stream media server after demodulating the TS stream of the multichannel program. Secondly, streaming media server uses Live555 open source project to build the streaming media data connection between the gateway and mobile devices, through which the TS data flow can be transmitted based on RTSP. Finally, the Wi-Fi router is used to build the LAN and make multiple mobile devices can receive video data at the same time.

2) Proposing a novel embedded server solution for multifunctional applications on the same hardware platform and solving the receiving and rendering the DTV programs problems on mobile devices. Being different form conventional dongle-type or dedicated set-top box receiver, our solution adopts three ways to solve the potable problems. Firstly, in the input part of the system, the mobile TV

stick is employed to receive TV signal in the terrestrial broadcast digital television signal coverage and TV signal is transferred to the standard TS by demodulator. Secondly, in the re-transmitting part, embedded ARM developing board is used as re-transmitter and can provide rich peripheral equipment's and industrial interface, which is very suitable for the application in the portable re-transmitting system. At last, wireless LAN is used as output part to improve the portability of system.

3) Integrating online video monitoring, VOD and other functions into the EMMS, and enabling mobile devices can receive DTV program, enjoy VOD, achieve video surveillance or relay Internet video program via local Wi-Fi interface simultaneously.

The following parts are organized as follows. Section 2 describes the general re-transmitting system design of the proposed embedded media server and rendering solution on the mobile devices. Section 3 designs the hardware architectures of the proposed multifunctional media server and receiver system. Section 4 discusses the detail design of system software and some key technologies. The whole hardware prototype and verification systems are given in Section 5 and then final conclusion is drawn in Section 6.

2. Retransmission System Design based EMMS

Digital television retransmission system of EMMS implements re-transmitting terrestrial digital television signals for smart phone. Fig. 1 shows the architecture design of re-transmission system based proposed EMMS. As showed in Fig. 1, the terrestrial DTV receiver, such as DTMB or CMMB, is composed by a tuner and a demodulator. The tuner module receives terrestrial DTV RF signal, and generates a differential signal, from which MPEG-2 or AVS TS is generated by the demodulator. Then, TS stream is outputted to WIFI network adaptor which works as access point (AP) router and the TS bit-stream is relayed to mobile devices by IP multicast technology via the Wi-Fi AP router. DTV streaming server bridges the gap between DTV transmitter and mobile devices, and the mobile devices can receive the DTV TS stream freely. The video and audio decoding is implemented by software on the mobile devices. Fig. 2 presents the workflow of re-transmitting system.
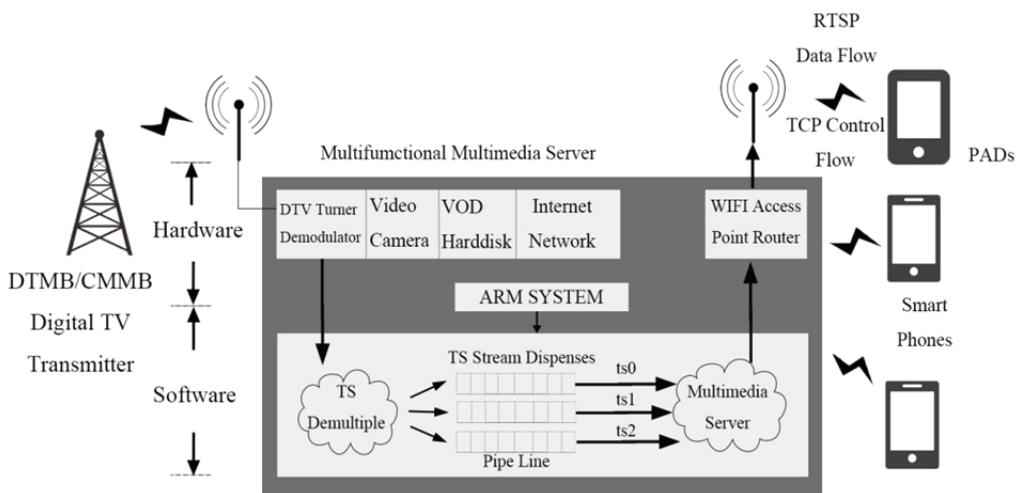

Fig. 1. Architecture of re-transmission system.

In order to effectively receive the terrestrial broadcasting RF signal from digital TV transmitting towers, the module employs the mobile TV stick as a signal receiving terminal in the streaming media retransmitting system. The tuner module in the mobile TV stick first receives digital TV RF signal, generates differential signal, then generates standard TS and transfers to a streaming media re-transmitter to process.

To meet requirements of user concurrent requests on the multiple mobile devices to the different programs, FIFO pipe transporting mechanism is used to solve multiple data stream distributed control transmission. When terrestrial DTV receives a carrier frequency signal, signal is processed by digital DTV tuner and demodulator, output standard TS containing multiple programs, which then transmit to TS demultiplexing module and the TS stream multiplex program is decomposed into the basic stream of multiple single program. Then multiple FIFO data pipeline is established and single program data stream is stored and distributed. The server is waiting for user request and performs the corresponding command. User can choose different program or the same channel at the same time based on the different requirements. If different users choose different programs, different FIFO data pipelines will transmit different TS stream. However, if many users request the same program at same time, it is necessary to copy data pipeline which stores this program and ensures the correct program stream transmission in multiple FIFO pipes. In the end, the multiple pipelines transmit data stream of multiple programs in parallel (TS1, TS2, TS3) to the media server of the system. There are several TV programs embedded in one TS bit-stream, which is demodulated from one channel under a certain RF carry frequency. To decrease the power dissipation and bandwidth of Wi-Fi data transmission, the server parses the TS stream and demultiplexes multiple TV programs in parallel, and offers simultaneous access for different mobile devices. Therefore, the multiple mobile devices can simultaneously be connected to the server and retrieve elementary stream (ES) of different DTV programs.
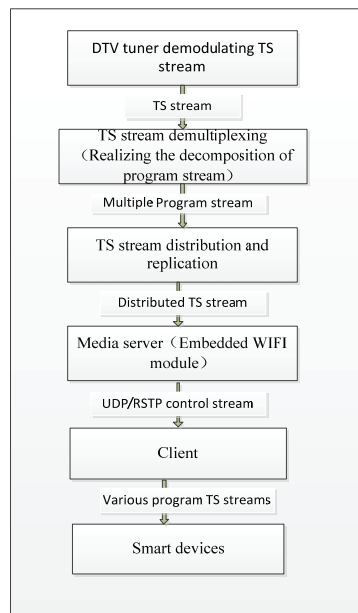

Fig. 2. Workflow of re-transmitting process.

In the design of free-network traffic transmission of demodulation to the flow free transport to the mobile devices, a platform of streaming media server is built based on ARM system, on which Wi-Fi hotspot is set up by the Wi-Fi adapter. When each mobile device user establishes a Wi-Fi connection, each user will be assigned to different wireless LAN IP address automatically. The server sets up a uniform resource locator (URL) to provide end users address to access data in server. Data transfer control between mobile devices terminal and server is based on RTSP protocol.

The clients of mobile devices can get program list information via the UDP protocol, which select the channel to send request information to streaming media server. When the multithread listener on server listen the connection request or change program request, it will adjust the tuning carrier frequency to get TS stream containing the current program, store the TS stream in the FIFO pipe and transfer to the streaming media server. At the same time, in order to establish the RTSP connection between server and client and transfer streaming data to the client, the media server can build 4–5 RSTP protocol flow transport connections at the same time and realize multiple client requests. Multiple client RSTP data connection can be concurrent transmission, share bandwidth Wi-Fi channel, and allow the multiple mobile devices terminal to watch digital TV programs almost simultaneously.

In addition, to fully unitize the resource of hardware platform of embedded server, the EMMS is also designed to read streaming media bit-stream from VOD disk, camera or access interface to the Internet. Various streaming bit-stream is similarly relayed to mobile devices via local Wi-Fi.

3. Hardware Architecture Design of EMMS

The EMMS hardware system mainly consists of ARM system, DTV reception module, Wi-Fi wireless AP router, and standard camera, hard disk and access interface to the Internet. In addition, surveillance camera, VOD hard disk and Internet network interface are all standard peripherals connected with ARM systems via USB, IDE interfaces etc. Fig. 3 shows the detailed hardware architecture of the system.
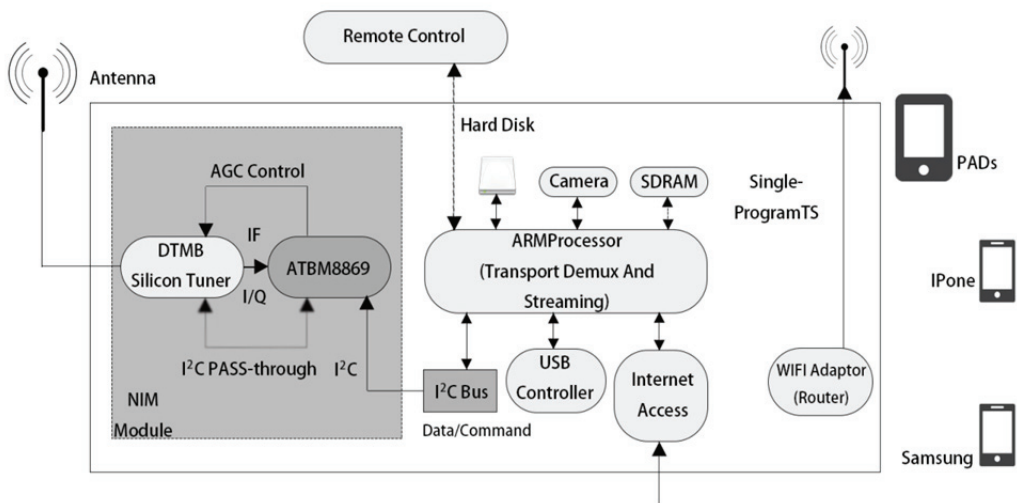

Fig. 3. Architecture of system hardware platform.

The broadcasting-terrestrial DTV tuner provides the selective tuning, demodulation, decompression, error correction, analog-to-digital conversion and other functions. Selective tuning is the process by which the RF of the television channel is selected by a receiver within a band of transmitted RF signals. The tuner usually performs the function of frequency-agile selection and unwanted out-of-band signals rejection. Demodulation means transforming the signal from the tuner into a usable signal that a media server use to produce images and sound without further consideration for the frequency it is transmitted. Since digital signals that are broadcast over the air are compressed, once they are received by the tuner, these compressed packets of digital data need to be decompressed. Our system uses lossy compression and while the decompressed data size is the same as the original compressed data size, although the data produced is not exactly the same as the original data fed into the system at the transmitting site, but it is close enough that most people cannot identify the differences. Error correction technology is also used by our tuner to insure any missing data can be corrected. The function of the demodulator is to convert the analog signal to a digital bit-stream, check errors and forward the bit-stream to the demultiplexer. The modulator is to deliver a signal to the return path tuner and the WIFI AP router acts as a router to relay the bit-stream to mobile devices using TCP/UDP/IP and multimedia communication protocols within the Wi-Fi signal covers.

In addition, video camera is connected to ARM to offer compressed surveillance for media server. The media server calls the V4L2 interface in Linux system to control the camera to capture the original YUV video data. To reduce the video data storage of system and save network bandwidth, the x264 encoder is adopted to compress the original YUV video data, which can reach 500 times compression effect. At last, to achieve real-time transmission, the pipeline blocking control is used in data transmission and mobile devices terminal can read the video data from media system synchronously.

VOD program bit-stream is retrieved from the VOD hard disks and then is relayed to media server. Similarly, the Internet networks enable ARM system to access digital multimedia stream from Internet webpage which is also relayed to the multimedia server. Various streaming bit-streams are relayed to mobile devices via local Wi-Fi AP.

4. Software Platform of the EMMS

The design of the software platform is based on ARM QY-IMX6S hardware and Linux operating system. According to system characteristics and application requirements, the objective of software platform is to support media forward transmission, DTV reception and rendering, DTV decoding and broadcasting and so on. The software framework consists of two parts as shown in Fig. 4, the right part is the program on the Android system of the mobile devices, and left is the software on the Linux system of the multifunctional server.

The design of the system adopts the hierarchical and sub-modular method. The software structure is divided into three layers as follows.

1) System software layer: The layer includes board support package level and the operating system kernel software module, which mainly provides standard drive interface for access to the hardware device for operating system and user tasks, packaging services of hardware resources and the operating system, and the unified access interface for the upper layer. The service supplying to the upper layer is implemented based on Hardware Abstraction Layer (HAL). By the HAL, the underlying hardware
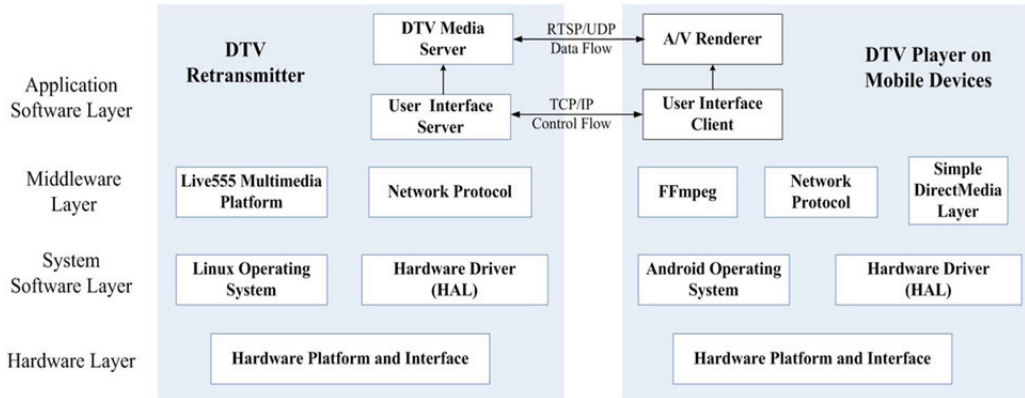
Fig. 4. Software architecture of the DTV streaming system.

platform can be map to multiple hard software entities and transparently be provided to the middleware system, which can eliminate influence of different hardware platform on the middleware implementation and provide a uniform functionality for the upper layer.

2) Middleware layer: The layer defines a number of APIs to provide various applications with high-level functionality for multifunctional multimedia sever, support a running environment of the applications for the mobile devices users. As show in Fig. 4, the middleware platform of the system consists of Live555 multimedia platform and network protocol. Live555 is a cross-platform open source project to provide solutions for streaming media [19], and is suitable to support the standard streaming media transfer protocol and realize the streaming, receiving and processing of variety encoding format data of audio and video. However, our system is to design an embedded multimedia server for mobile devices to realize transmission of multimedia data on embedded control board by communication protocol, and Live555 only can achieve transmission of locally stored audio and video, it is necessary to improve the source code of Live555 to realize the functions of our system. For example, since the real time data transmission of the embedded control board to the media server is done by the pipe, if the unmodified Live555 is used directly, Live555 process will cause blocked when reading the pipe files. Our main modification to Live555 is to open the pipe by set O_NONBLOCK flag. Function of Open (filename, O_RDONLY) is to open the pipe in read-only mode and will be blocked until the same pipe is open by a process in write mode. Open (filename, O_RDONLY|O_NONBLOCK) is to open the pipe in read-only unblocked status mode and the open call returns immediately. The read process will return 0 and the pipe cannot be blocked when there is no process to open the same pipe in write mode. By the some modifications of Live555 codes, transmission of the data in real time in our system can be realized.

3) Application software layer: The layer performs the function of DTV reception and rendering, such as A/V (audio and video) rendering and UI (user interface). A/V renderer function includes TS streaming de-multiplexing, decoding of video audio ES stream, and video audio playback. The program of A/V renderer mainly composed of master loop thread, decoding thread, playing thread, and so on. Master loop thread first gets TS data and de-multiplexes multiple programs, then forwards ES date of one way program to the video audio frame list waiting for decoding. Meanwhile, master loop thread listens to request and responds to the request, or calls some function according to the request message. Decoding thread is responsible for decoding to the video audio ES stream data. In the decoding process,

data of DTS (Decoding Time Stamp) and PTS (Present Time Stamp) is used for synchronization video audio data streams to realize video audio normal synchronous playback. Moreover, in A/V rendering, the filter graph model of DirectShow is used to manage the process of data streams. The procedure mainly follow these steps: first to inherit the base class of video filter, second to define unique identifier, then to develop member functions and last to manage video and audio streams.

Client UI design is developed by Qt, which helps users achieve program switching, frequency points switching, VOD, video surveillance, and so on.

4) DTV Player on Mobile Device: The module is composed of Android operating system, device drivers, communication protocols, FFmpeg decoder and so on. The configurations of Android system, device driver and network protocols are standard provided by mobile device providers. FFmpeg is an open source, free and cross-platform and supports more than 90 decoders and H.261/H.263/H.264 protocols [20]. Our system mainly uses its decoding function and main procedure is: firstly, `av_register_all()` completes the register of codec on FFmpeg system, then `av_open_input_file()` opens the multimedia file in the local or in the server, `av_find_stream_info()` finds out ID that corresponds to audio and video stream in multimedia file. Then `avcodec_find_decoder()` and `avcodec_open()` will complete the initialization of video decoder and audio decoder via ID. Finally `avcodec_alloc_frame()` allocates a buffer for each decoded video frame, and `avcodc_decode_video()` starts the decoding work. Once the FFMpeg library function completes the video and audio decoding, the decoded streaming data is converted in format and utilize platform independent SDL libraries to play. Initialization process of SDL includes following main steps: Call `SDL_CreateWindow()` to create a drawn window; Call `SDL_CreateRenderer()` to create a renderer; Call `SDL_CreateTexture()` to create a texture. After initialization of SDL, the output video frame data of FFmpeg can be used as input of texture. Fig. 5 shows DTV decoding and playing procedure diagram on the mobile devices.
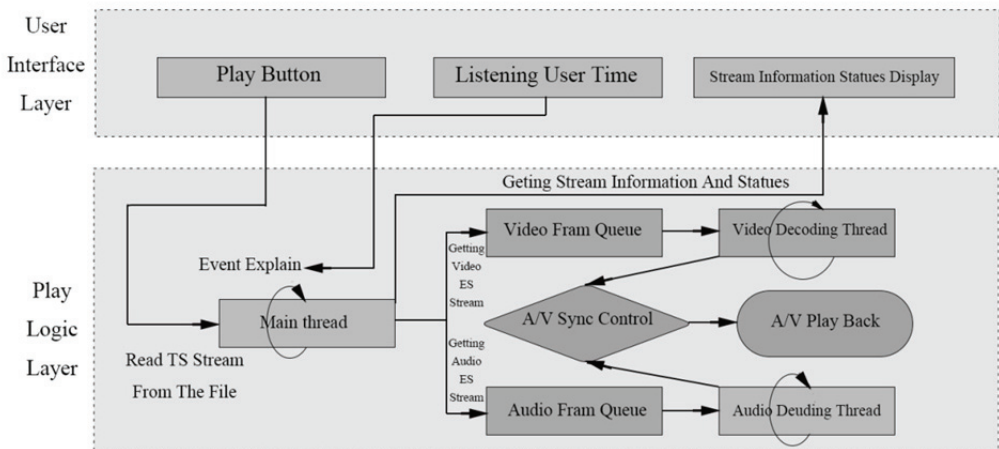

Fig. 5. DTV decoding and playing process diagram.

5) Communication protocols: Because the efficient compression can greatly reduce data volume and the A/V data packets in system are not required to arrive in order, our system adopts low latency and high transmission UDP bi-direction communication during data transmission. Multiple mobile devices can get efficiently compressed audio and video data on the server, including the list containing

information of paths and names. In the transmission of A/V streams (RTSP data streams) to mobile devices, a Wi-Fi hotspot by the WIFI network adapter is first established, and the source of video program is got by the list of mobile device. Then streaming media connection between smart device and media server is set up based on RTSP. As showed in Fig. 6, the client first initiates channel request based UDP and then transmits the channel information to the server in character string. After responding channel request from the client, the server reads the data in the pipes and establishes an RTSP connection to the client, then transfers streaming data to the client.

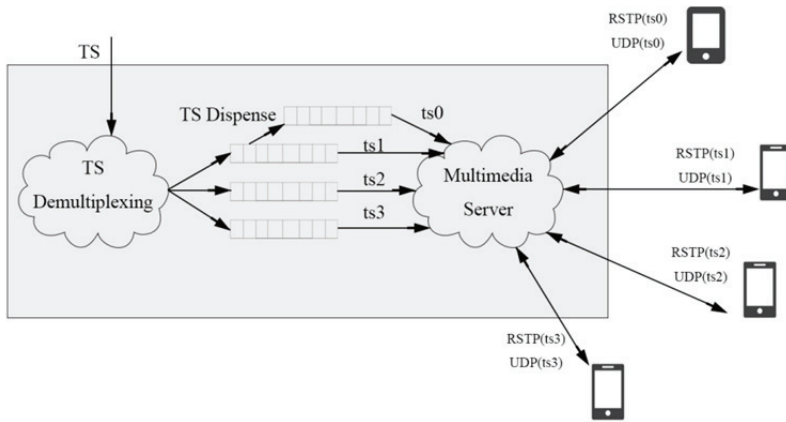**Fig. 6.** Communication flow between server and client.

5. System Evaluations

In order to verify the validity of the system, we set up a test platform prototype, as showed in Fig. 7, including a QY-IMX6S embedded motherboard, an AP-901M-A wireless routing module, a MyGica D680 mobile TV stick, a MN88436 demodulator, an Android phone and an Apple phone.

Fig. 7. System testing prototype.

First experiment is to test whether the user can receive the real-time terrestrial DTV program normally. The testing process is: first installing our independently developing APP software on the phone client; then booting up the demo board and running the required service processes on the target board. One of service processes is UDP Socket Server, which mainly responds user channel request. When user client sends a channel request, the UDP Socket Server first searches matching contents in program list and writes the found data into the pipe for reading. Another service process is Live555 Media Server process, which builds a stream media server and provides RTSP real-time streaming services for both the server and the client. In the test, the smart phone is connected to the router's RT5350 wireless LAN. Fig. 8(a) shows our developing APP interface on the phone client and Fig. 8(b) shows that the client can receive real-time terrestrial DTV program normally.

| (a) | (b) |

Fig. 8. (a) APP interface on the phone client. (b) Example of receiving DTV program.

In the EMMS testing process, we test respectively whether the EMMS can correctly identify the linked devices, whether running program is stable and whether system compatibility is good after system running a long time (1 hour, 6 hours, 12 hours, and 24 hours). Table 1 presents the testing results, which proves the performance stable after system running long time.

Table 1. Testing results of EMMS system running

Running time (hr)	ARM system	Demodulator	Stream server	Device identification
1	Normal	Normal	Normal	Normal
6	Normal	Normal	Normal	Normal
12	Normal	Normal	Normal	Normal
24	Normal	Normal	Normal	Normal

After completing the single-client test, the system stability with multiple clients also is tested. Be similar to the testing method of [17], our multi-client testing is simulated by 80 thread call algorithms on the server. Each client calls the algorithm via interface and transmits back the processed data respectively to the 80 clients running on the simulating client computer. In order to compare our system performance with other systems, we also test systems respectively presented by [15,17] with the same method. Table 2 shows the different packet loss rate and server response time at different client number of our system and other two systems.

Table 2. Performance comparison of different systems

Client no.	Proposed system		[15]		[17]	
	PLR (%)	RT (ms)	PLR (%)	RT (ms)	PLR (%)	RT (ms)
1	0	10	0	12	0	12
5	0	16	0	18	0	21
10	0	20	0	23	0	28
20	0	30	0	35	0.1	39
30	0	42	0.2	46	0.3	50
40	0	54	0.6	58	0.8	68
60	0.5	60	0.8	67	1.5	78
80	0.8	90	1.1	98	2	102
100	1.2	120	1.8	134	2.4	143

PLR=packet loss rate, RT=response time.

Comparing with performance of other two systems, we can find from Table 2 that the proposed system has the least packet loss rate and the least response time. When the number of client requests is not too large, server response time does not increase linearly with the increase of concurrent request threads and server works stably without packet loss in three systems. However, when the number of concurrent request threads increase to 80, the client frequent requests make server response time increase sharply and packets began to be dropped, that show the server throughput ability will exceed the load when client number increase too much.

In the second experiment, the function of rendered video and audio programs is verified. In the system evaluations, the Chinese DTMB digital TV system in Hangzhou City is used. The AVS-P2 video standard and MP2 stereo audio standard are used for multimedia source coding. In the terrestrial DTV system in Hangzhou City, there are 8-way CIF (Common Intermediate Format) TV channels and one-way high definition TV channel in one RF carry frequency. Fig. 9(a) shows the parsed program information, as well as the parameters of the compressed video and audio bit stream. Fig. 9(b) presents one-way DTV program renderer example. The experimental results showed the rendered video and audio programs are stable and quality-efficient.

(a)   (b)

Fig. 9. (a) User interface of parsed program information. (b) DTV rendering example.

Then, we verify the function of the VOD service, local video surveillance on the same verification prototype system. As show in Fig. 10(a), after user of Android phone inputs the IP address of the server, the server based UDP will display the play list and monitor list on the mobile terminal. The mobile terminal receives the data stream through RTSP connection. Fig. 10(b) presents the display screen of VOD on client, which can carry operations of pause, stop and so on.
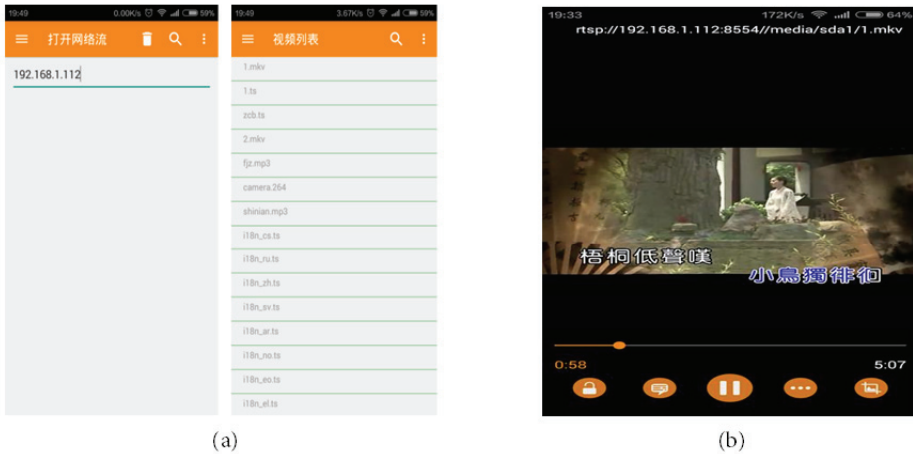

(a)   (b)

Fig. 10. (a) VOD example. (b) Effect of VOD on mobile devices.

Fig. 11 presents a testing case of local video surveillance of the system. The server outputs 15 fps CIF format H.264 video data. The clients of Android phone set up a connection with sever by WIFI network and implement local video preview and other operations. Furthermore, we also test DVD broadcasting services and the relayed and re-transmitted function of bit-stream retrieved from Internet to multiple mobile devices.
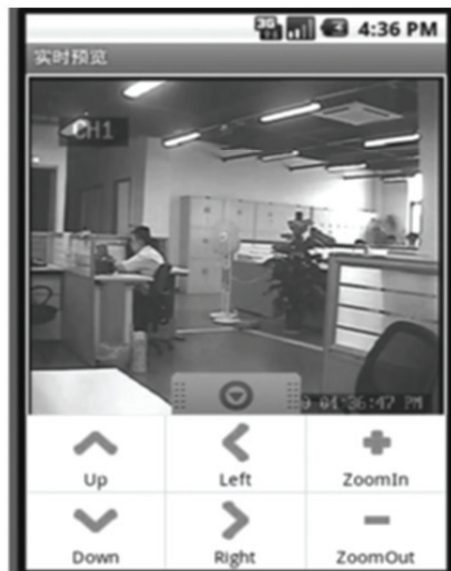

Fig. 11. User interface example of video surveillance.

6. Conclusion

This work proposes an embedded multifunctional server system, which has advantages such as small volume, high real time, free flow and little requirement for network bandwidth. The server works as a re-transmitter to receive the terrestrial DTV and relay the transport stream, and relays bits-stream from VOD hard disk, video surveillance camera to local mobile devices via local Wi-Fi network without Internet connection. The EMMS is designed to either receive the terrestrial DTV radio signals and demodulate out digital TS, or read streaming media bit-stream from VOD disk or camera or access interface to the Internet. The proposed solution is verified by experiments and testing result shows that our system can receive the real-time terrestrial DTV program normally and the rendered video and audio programs are stable, quality-efficient. In addition, our simulations also verify that the proposed system can implement VOD service, local video surveillance and DVD broadcasting services using mobile devices.

However, with advancement in hardware circuitry and mobile operating system, various features have been evolved in the existing smart mobile devices and the energy dissipation is quite difficult to control, which is becoming a big challenging factor. Low energy consumption optimization for the system solution will be our future research focus. Our initial plan is to design a novel application-aware and user-interaction aware energy optimization framework, which can optimize CPU and screen backlight energy consumption while maintaining a minimum acceptable level of performance.

Besides, since the clients are required to continuously initiate requests to the server and wait for the response, which may cause some additional overhead for the server when client number increase too much in the system, and the server throughput ability will exceed the load and cause server response time increase sharply and packets be dropped. In the next works, we will improve communicating way between server and client and place communicating control information in the protocol headers as far as possible.

Acknowledgement

References

[1] *Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television,* ETSI EN 300 744 V1.2.1, 1999.

[2] *Framing structure, channel coding and modulation for digital television terrestrial broadcasting system*, Chinese National Standards GB 20600-2006.

[3] T-DMB Standard [Online]. Available: http://www.t-dmb.org.

[4] J. J. Li, "A new standard of mobile multimedia broadcasting - CMMB," *Shanxi Electronic Technology (Chinese)*, vol. 2008, no. 4, pp. 84-85, 2008.

[5] C. C. Hsieh, C. H. Lin, and W. T. Chang, "Design and implementation of the interactive multimedia broadcasting services in DVB-H," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 4, pp. 1779-1787, 2009.

[6] J. S. Choi, J. W. Kim, D. S. Han, J. Y. Nam, and Y. H. Ha, "Design and implementation of DVB-T receiver system for digital TV," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 4, pp. 991-998, 2004.

[7] C. C. Wang, H. K. Lo, S. P. Lin, and R. Hu, "High-sensitivity and high-mobility compact DVB-T receiver for in-car entertainment," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 1, pp. 21-25, 2006.

[8] Y. W. Bai and M. H. Lin, "Design and implementation of low-price business card-size portable digital TV set through high integration DVB-T solution," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 4, pp. 1608-1614, 2007.

[9] J. Han, I. Han, and K. R. Park, "Service-oriented power management for an integrated multi-function home server. *IEEE Transactions on Consumer Electronics*, vol. 53, no. 1, pp. 204-208, 2007.

[10] H. Yoon and J. Kim, "Collaborative streaming-based media content sharing in WiFi-enabled home networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 4, pp. 2193-2200, 2010.

[11] Geniatech, "WiTV2: watch live TV on iPad, iPhone & Android devices," [Online]. Available: https://www.geniatech.com/product/witv-t2/.

[12] The SR-HD1700US DISC & HDD recorder [Online]. Available: http://pro.jvc.com/prof/attributes/features.jsp?model_id=MDL102286&feature_id=01.

[13] S. H. Park, Y. S. Park, S. B. Yu, and J. R, Choi, "Implementation of ATSC mobile DTV broadcasting for N-screen smart devices," in *Proceedings of 2012 IEEE International Conference on Consumer Electronics*, Las Vegas, NV, 2012, pp. 331-332.

[14] Newsky Tech Co. Ltd., "DVB-T ISDB-T MPEG4 digital TV tuner SDR radio" [Online]. Available: http://www.alibaba.com/product-detail/High-quality-DVB-T-ISDB-T_709519719.html.

[15] Z. Pan, H. Yin, and C. Yang, "Design and implementation of an intelligent terminal for receiving broadcast digital television programs," *Journal of China Jiliang University*, vol. 25, no. 3, pp. 323-327, 2014

[16] J. Song, Z. Yang, and J. Wang, "Digital and implementation of DTV receiver," in *Digital Terrestrial Television Broadcasting: Technology and System*. Hoboken, NJ: John Wiley & Sons, 2015, pp. 255-298.

[17] A. Chakraborty and S. R. Das, "Designing a cloud-based infrastructure for spectrum sensing: a case study for indoor spaces," in *Proceedings of 2016 International Conference on Distributed Computing in Sensor Systems,* Washington, DC, 2016, pp. 17-24.

[18] Z. X. Zhu, "Research and implementation of streaming media data forwarding platform in the field of video surveillance," Nanjing University, 2014.

[19] FFmpeg [Online]. Available: http://ffmpeg.org/.

[20] Wikipedia, "LIVE555" [Online]. Available: https://en.wikipedia.org/wiki/LIVE555.

Jun Huang https://orcid.org/0000-0003-0947-1936

He received his B.S. and M.S. degree in computer & application major from Hunan University and ChangJiang University in 1982 and 1992, respectively. He is currently with the College of Modern Science and Technology, China Jiliang University, Hangzhou as an associate professor of Computer Engineering. His research interests include network communication and multimedia processing.

Haibing Yin https://orcid.org/0000-0002-3025-0938

He received the Ph.D. degree from Shanghai Jiaotong University, Shanghai, China, in 2006. He is currently with the College of Information Engineering, China Jiliang University, Hangzhou as a professor of Electrical Engineering. His current research interests include image and video processing, and very large scale integration architecture design.