



Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

The application of FFMPEG technology in supervisory control system

Wu-sheng Tang, Jie Yu and Li-hong Tian

College of Mechanical and Vehicle Engineering, Scientific Research Administrative Dept., Chang Chun University,
Chang Chun City of JiL in Province, China

ABSTRACT

The article briefly introduces the developing process of FFMPEG technology, achieved function through the application of FFMPEG technology, and main application field of FFMPEG technology at present. Through examples, the article illustrates the mode of establishing thread based on DVR with VC program, the method of achieving supervisory control to signal channels by adopting FFMPEG process, and realization of multichannel signals through FFMPEG technology. The conversion from the standalone mode to the network mode can effectively lower the bit stream of video information, reduce demand for network bandwidth, and enhance the flexibility to watch surveillance video for clients. The client can easily check video, so long as it supports flv formats.

Key words: FFMPEG, Bit Stream, DVR, FLV

INTRODUCTION

Ffmpeg is an open source cross-platform audio and video solving technology which is widely applied in recent years, a complete open source solving scheme which integrates the function of recording, conversion and audio/video codec as a whole. Based on Linux operating system, development of Ffmpeg can also be compiled and used in most operating system[1]. Ffmpeg supports more than 40 kinds of coding such as MPEG、DivX、MPEG4、AC3、DV、FLV, etc. and 90 kinds of decoding such as AVI、MPEG、OGG、Matroska、ASF[2], etc.; Open source players such as TCPMP、VLC、MPlayer, etc. are all applied in Ffmpeg. It can quickly achieve the conversion of audio/video format, obtain and save data from audio/video sources as well[3].

II. APPLICATION OF FFMPEG TECHNOLOGY

A. Application Field of FFMPEG

FFMPEG applies to the conversion of surveillance video. With the development of video surveillance and codec technology, surveillance video applies to all fields stage by stage, representing mode of surveillance video also transforms original standalone mode into network modes gradually, so long as there is network, the real-time surveillance video can be watched through computer[4]. But then, video information need consume a great deal of network bandwidth in the process of network transmission, so lowering video bit stream becomes hot technology nowadays, FFMPEG can convert standardized h264 video signals into flv, it can not only partly lower bit stream of video information and then reduce network bandwidth, but also enhance the flexibility that clients watch surveillance video. The client just supports flv, the video can be checked[5].

B. Parameter Table of FFMPEG

TABLE I. Parameter Table of FFMPEG[6]

Number	Parameter	Illustration
1	-b	Set bit rate , default 200kb/s
2	-r	Fps, set frame frequency, default 25 frame
3	-s	Size, set size of frame, format is WXH, default 160X128. The following abbreviation can also be directly used
4	-bt	Tolerance, set video bit rate tolerance kbit/
5	-maxrate	Bitrate, set the maximum video bit rate tolerance
6	-sameq	use the same video quality as sources (VBR)
7	-re	use local frame frequency to read data, mainly use to simulate capture device

C. Implementation Sample of FFMPEG

The threading is established according to the channels of DVR through VC procedures. That is, the channel number of supervisory control signals decide the number of ffmpeg procedures[7].

Ffmpeg

```
-re
-i "3.flv"
-qscale 3
-s 720*360
-aspect 16:9
-r 25
-threads 4
-vcodec flv
-acodec libmp3lame
-ar 44100
-ab 128k
-ac 2
-f flv rtmp://192.168.0.128/MyStream/stream123
```

D. Code Examples and Running Interface of Starting up Multichannel FFMPEG Proceeding by Using VC

```
HANDLE hThread = NULL;
for(int i = 1; i < streamCount + 1; i++){
FfmpegStruct ffmpegS;
ZeroMemory(&ffmpegS, sizeof(FfmpegStruct));
strcpy(ffmpegS.app , strApp);
ZeroMemory(&streamBuf, sizeof(streamBuf));
itoa( i, streamBuf, 10 );
strcpy(ffmpegS.streamName , strPreSchoolCode);
strcat(ffmpegS.streamName, "_");
strcat(ffmpegS.streamName , streamBuf);
ZeroMemory(&portBuf, sizeof(portBuf));
_itoa( atoi(strStartPort) + i, portBuf, 10 );
strcpy(ffmpegS.port , portBuf);
hThread = (HANDLE)_beginthreadex(NULL,0, (unsigned (__stdcall*) (void
*))ffmpegThread,&ffmpegS,0,&pid);
Sleep(100);
printf("\n No.%d ffmpeg proceeding has start up, the threading number is: %d\n",i, pid);
}
```

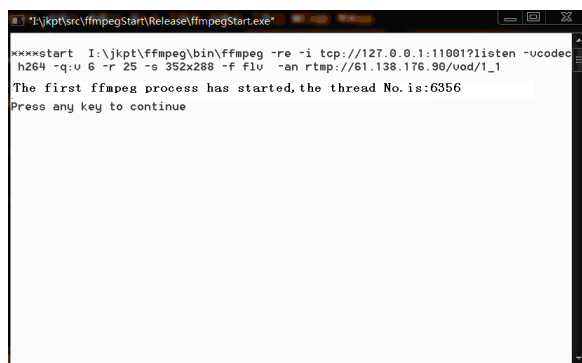
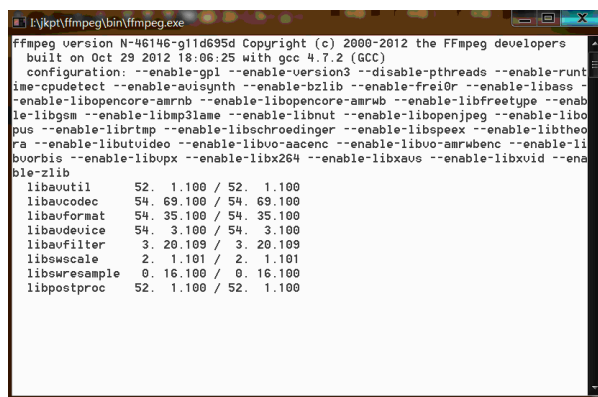


Figure 1. Screen to start up multichannel Ffmpeg

E. Code Examples and Running Interface of transferring ffmpeg procedures by Using VC

ffmpegThread , thread codes are as following:

```
void ffmpegThread(void *arg){
pFfmpegStruct pff = (FfmpegStruct*)arg;
char destination[1024];
ZeroMemory(&destination, sizeof(destination));
strcpy(destination, "start ");
strcpy(destination, g_Path);
strcat(destination, " -re -i tcp://");
strcat(destination, g_SvrIP);
strcat(destination, ":");
strcat(destination, pff->port);
strcat(destination, "?listen ");
strcat(destination, "-vcodec ");
strcat(destination, g_vcodec);
strcat(destination, " ");
strcat(destination, g_bitrate);
strcat(destination, " -r ");
strcat(destination, g_r);
strcat(destination, " -s ");
strcat(destination, g_s);
strcat(destination, " -f ");
strcat(destination, g_f);
strcat(destination, " -an rtmp://");
strcat(destination, g_RED5SvrIP);
strcat(destination, "/");
strcat(destination, pff->app);
strcat(destination, "/");
strcat(destination, pff->streamName);
printf("\n****%s\n", destination);
system(destination);
}
```



The screenshot shows the command prompt output for the Ffmpeg process. It displays the version (N-46146-g11d695d), copyright (c) 2000-2012, and build date (Oct 29 2012 18:06:25). It lists various configuration options such as --enable-gpl, --enable-version3, --disable-pthreads, etc. Below the configuration, it shows a list of libraries and their versions, including libavutil, libavcodec, libavformat, libavdevice, libavfilter, libswscale, libswresample, and libpostproc.

Figure 2. Startup Screen of Ffmpeg Process

F. Process and Code examples that DVR Links FFMPEG

The achievement of this step mainly uses monitoring information which is obtained through the method of NET_DVR_RealPlay_V30 in SDK supplied by DVR manufacturer, and delivers monitoring information to listening port of FFMPEG through SOCKET.

```
// transfer SDK of dvr
IRealPlayHandle1 = ZLNET_RealPlayEx(IUserID, 0, NULL, RealPlayType);
if(IRealPlayHandle1 < 1){
printf("NET_DVR_RealPlay_V30 the first channel signal is abnormal! \n");
}
cbRec=ZLNET_SetRealDataCallBackEx(IRealPlayHandle1, g_RealDataCallBack_V30_1, 1000, 0x00000001);
if (!cbRec){
goto ERR;
}
```

```
// codes of callback function
Void CALLBACK g_RealDataCallBack_V30_1(LONG IReallHandle, DWORD dwDataType, BYTE *pBuffer,
DWORD dwBufSize, LONG IParam, DWORD dwUser)
{
if(send (sClient1, (char *)pBuffer, dwBufSize, 0) < 0){
printf("-----the first decoding procedure breaks down! ");
//connect to server
ret = connect(sClient1, (struct sockaddr *)&saServer, sizeof(saServer));
if (ret != SOCKET_ERROR)
{
send (sClient1, (char *)pBuffer, dwBufSize, 0) ;
printf("-----the first decoding procedure work normally! ");
}
}
Sleep(delayedMsecCount);
}
```

CONCLUSION

Technology of FFMPEG is a set of open source computer programs which can be used to record and convert digital audio and video into stream[8]. It comprises the leading audio and video encoding library Libavcodec in currently. FFMPEG is under the development of Linux, but it can compile in most operating system including Windows[9]. It has the function of video screenshots and watermarking to video. It can easily achieve interconversion of many kinds of video formats, convert video recording avi format into flv format adopted by video website now; lower the network bandwidth which are largely consumed during the process of video bit stream transmission[10]; and enhance the flexibility to watch Surveillance video for clients. The client can easily check video, so long as it supports flv formats. Technology of FFMPEG applies to the system of video surveillance and mobile video surveillance.

Acknowledgement

The work is Changchun City funded science and technology project (2013319)

REFERENCES

- [1]Yang Lijun, *Modern Electronic Technology* [J] **2011**.8(15)P147-149.
- [2]Chen Chu, *Computer and Communication* [J] **2012**.6 P55-57.
- [3]Yang Jinling, Design and Development of mobile video surveillance system, Master Dissertation of Jinan University [M] **2011**.5.
- [4]Tan Yan, *Computer and Communication* [J] **2011**.3,Vol.7(12) P2912-2913.
- [5]Jiang Zhifeng, *Microcontrollers & Embedded System* [J] **2008**(1) P69-71.
- [6]Wu Zhangshun, Zhang Xun, *Journal of Hangzhou Dianzi University*, **2006**, 26 (3) : 30-34.
- [7]Wu Yue, HuiJuan Shi, *Electronic Design Engineering* [J] **2013**.12P185-187.
- [8]HU Cheng , REN Ping an , LI Wen li, *Computer Technology And Development* , [J] **2011**.10P85-91.
- [9]Ding Feng, Diao Ming, *Applied Science and Technology*, [J] 2013.4P11-14.
- [10]KIM W J. *IEEE Transactions on Consumer,Electronics*, **2010**, 56(2) : 3-10.