

# Design of H.265 Streaming Media Player System Based on QT

Yu-Hua Wu<sup>1, a</sup>, Min Zhang<sup>1, b</sup>, Ting Yang<sup>1, c</sup>, Shi-Jun Li<sup>d, \*</sup>

<sup>1</sup>Jilin Agricultural University, Changchun 130118, Jilin, China

<sup>a</sup>awuyuhua@163.com, <sup>b</sup>1165339875@qq.com, <sup>c</sup>627533884@qq.com, <sup>d</sup>lsj0883@sina.com

\*Shi-Jun Li

**Keywords:** H.265, streaming media player, FFmpeg, codec.

**Abstract.** With the continuous development of video services, video applications gradually in order to meet the high resolution, high frame rate, high compression development demand, put forward an H.265 streaming media player system based on QT. Under the QT environment, the use of FFmpeg codec library for video file processing, while the use of SDL multimedia displays library combined with multi-thread display video files. Experimental result shows that the player can play a clearly and smoothly H.265 video files, and can support 4K high-quality video playback.

## 1 Introduction

Video data plays an important role in information interaction. According to the research, it is compared with other information, the human brain accepts 70% of the information is from the visual information[1]. However, with the development of mobile terminals and network technology, people's demand for video is growing, according to research shows that the video data will occupy 75% of the mobile data in 2020[2]. In order to provide users with high-quality video information, and reduce the video file data transmission bandwidth, the ITU-T VCEG put forward H.265 standard after H.264 standard. Compared to H.264 video files, H.265 standard video files play the same quality video files only need to occupy the original half of the bandwidth. While the H.265 video encoding standard supports 4K and 8K ultra-high-definition video, which means that people can use mobile devices to watch high-quality video. Therefore, the combination of FFmpeg and SDL library to develop the H.265 player based on QT can not only play H.265 video, and has good cross-platform feature[4], while allowing users to better experience high-quality video information.

## 2 Overall System Design

This design is based on multi-thread technology, using FFmpeg as a codec library, using SDL library output audio and video information; the same time under the QT environment, to achieve the player design. Because Qt, FFmpeg, SDL has a cross-platform feature, so the player has a good cross-platform capability, can work in different environments. The overall system framework is shown in Fig .1

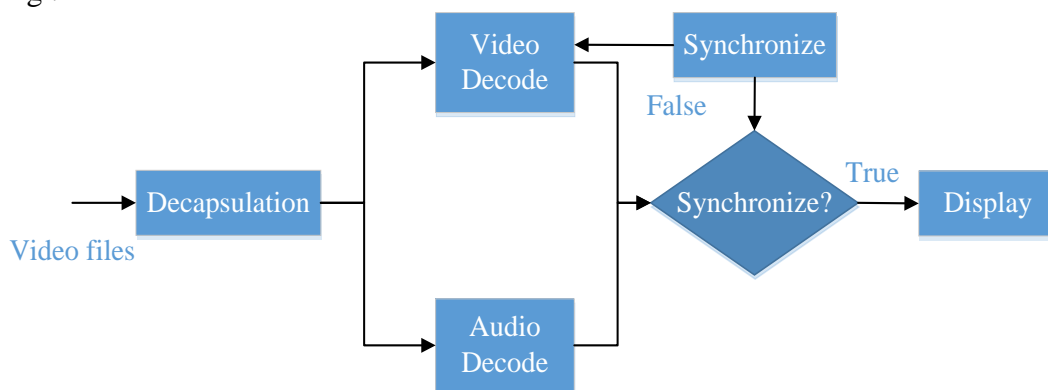


Fig. 1 Working flow chart of the player

### 3 FFmpeg Decoding Process

#### 3.1 FFmpeg Introduction

FFmpeg is a free cross-platform audio and video stream processing scheme, because its follow the LGPL or the GPL license, is a free development of software, allows you to modify[5].

It provides a powerful audio and video processing capabilities, including video capture, video format conversion, video recording and so on. Libavcodec is a dedicated audio and video codec library, because many of its codec are developed from scratch, so it can improve the quality of the program can be transplanted and video codec. Such as VLC, MPlayer and other famous open-source players are used FFmpeg library[6].

#### 3.2 FFmpeg Decoding Process

A multimedia file that contains audio files and video files information, but due to the use of compressed format is different, so you need to select the corresponding audio and video decoder[7]. And FFmpeg has the feature function, can can provide decoding operation for audio and video data of different compression formats.

FFmpeg decoding process: first read the video package file, read the stream information from the video package file, according to the corresponding video stream file format to find the corresponding decoding. Then decodes the one frame data to obtain the original audio data PCM or the original video data YUV. Decoding process is shown in Fig .2.

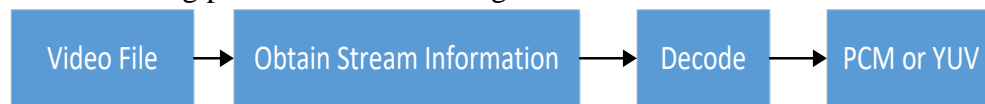


Fig. 2 Decoding process

FFmpeg decoded data structure is shown in Fig .3. The AVFormatContext structure mainly contains the information of the video file encapsulation format, AVInputFormat structure that corresponds to each video format information, structure information such as FLV and MKV format corresponding to the AVStream structure, including audio or video video format in the information flow, the AVCodecContext structure includes the audio or video stream format and corresponding decoder information stored in the AVPacket structure is a frame compression encoding of the data, the AVFrame structure is a frame storage pixel data compression the original video data decoded or the original audio sample data.

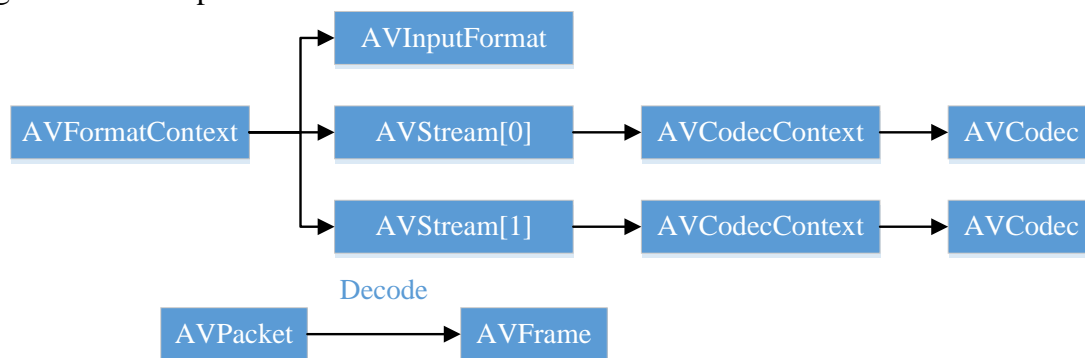


Fig. 3 FFmpeg decoded data structure

### 4 SDL Display Design

#### 4.1 FFmpeg Introduction

Simple DirectMedia Layer(SDL) is a set of open source cross-platform multimedia development library, while providing a wealth of control image, sound, input and output functions for the development of applications to provide great convenience. SDL can support multiple operating systems, with good cross-platform features, such as support for Linux, Windows, MacOS and other operating systems.

SDL is internally written in C language, which encapsulates the underlying interactive work of complex audio and video to the form of the library for users to use, while providing a large number of other language programming interfaces, it can be combined with the development of FFmpeg. But also for games, audio, and video applications, users need not know the hardware information, only need to use SDL audio and video processing API function interface, you can achieve audio and video processing, simplifying the audio and video processing Difficulty. The structure of the SDL is shown in Fig .4.

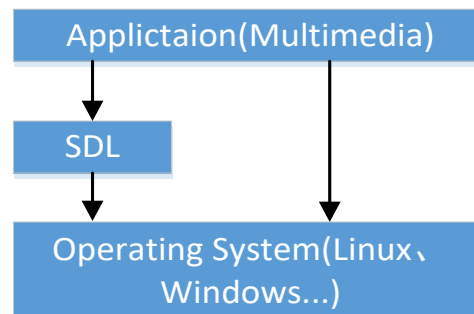


Fig. 4 The structure of the SDL

#### 4.2 FFmpeg Display Process

Through FFmpeg decoded audio and original data frame information, and then through the SDL library for display and playback[8]. SDL multimedia library mainly contains three structures are: SDL\_Window display window, SDL\_Renderer rendering device operation, SDL\_Texture texture settings. SDL display process is shown in Fig .5.

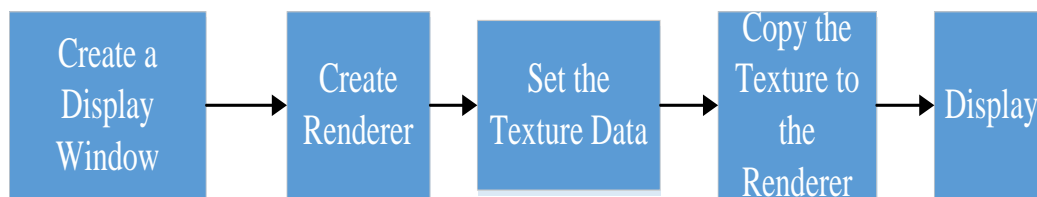


Fig. 5 SDL display process

#### 4.3 Multi-threaded design

When the video player needs to adjust the size of the user needs to adjust the size of the video display window, and so on, design a multi thread processing operation. The multi-threaded operation can improve the use of the CPU path and the utilization of system resources, the player display window interface as the main thread of the program, but also will accept the user interface size adjustment of the player information.

Multi-threading technology used in the player, mainly through the SDL multimedia display module in the multi-thread technology to achieve[9]. In SDL multimedia library through the use of SDL\_CreateThread to create multiple threads, through the use of multi-threading technology can move the player window and change the size of the player window.

### 5 System Test

This article uses the FFmpeg 3.0 and the SDL2.0 version to carry on the development, runs with the PC end I3 processor, the frequency is 2.3GHz, the 8G memory system. Test video files for the same content of H.264 format and H.265 format video files. By viewing the file size found H.265 standard format video file size 13.9M, and H.264 standard format video file size is 141M. Player test results is shown in Fig .6.



Fig. 6 Player test results

## 6 Conclusions

Through the above test, you can see the same section of the video file, H.265 encoding standard video file size is much smaller than the H.264 encoding standard. The result shows that, in combination with FFmpeg open source codec library and SDL multimedia library, the player developed in QT environment can support H.265 format video player, can clearly play H.265 format video files. And the H.265 format video file quality is significantly higher than the H.264 format of the video file is conducive to the user experience high-quality video.

## Acknowledgments

The authors wish to express their gratitude to the projects: *Design of H.265 Streaming Media Player System Based on QT from Education Department of Jilin Province*, *Jilin Province Economic Structural Adjustment Leading Fund Special Project* (No. 2014Y108) and *Changchun City Science and Technology Plan Project* (No. 14nk029), *Jilin Province Science and Technology Development Program Funded Project* (No. 20160623016TC), *Education Department of Jilin Province's 12<sup>th</sup> Five-Year Science and Technology Research Project* (2015175), for their generous support of this work. At the same time, the other authors also thank the corresponding author, Shijun LI, for his assistance in submitting this paper.

## References

- [1] LIN Fuzong, Multimedia Technology Foundation[M]. Beijing: Tsinghua University Press, 2002, in Chinese.
- [2] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015–2020 White Paper[R]. 2016.
- [3] DONG Wenhui, NING Jinhui, ZHANG Qian, PAN Rong. Overview of H.265 Standard and Its Application[J]. Radio & TV Broadcast Engineering, 2015,12:60-64, in Chinese.
- [4] QIN Yan. The Research of The Video Format Conversion Technology Base on The Project of FFMPEG[J]. Computer Knowledge and Technology, 2011,12:2912-2913, in Chinese.
- [5] PING Qingqing. The Design of Client Software in Mobile Video Surveillance on Windows Platform[D]. XIDINA University, 2013, in Chinese.
- [6] ZENG Hao, FANG Yuan. Implementation of video transcoding client based on FFMPEG[J]. Advanced Materials Research, 2013, 756-759:1748-1752, in Chinese.

- [7] XIN Changchun, LOU Xiaoping, LV Naiguang. The FFmpeg encoding and decoding in remote monitor system[J]. Electronic Technology, 2013,01:3-5, in Chinese.
- [8] WU Wenxiang, SHAO Shi. Design of The Embedded AVS Video Player Based on ARM Processor[J]. Computer Applications and Software, 2010,01:247-249, in Chinese.
- [9] LI Ke, LI Lu, LAN Shiyong. Multiple Video Streams Real-time Conversion and Playback Based on FFmpeg and SDL[J]. Computer Technology and Development, 2014,04:65-68, in Chinese.