Implementation and Application of Video Distribution Technology Based on OpenGL

IOP Conf. Series: Materials Science and Engineering **563** (2019) 042093 doi:10.1088/1757-899X/563/4/042093

Implementation and Application of Video Distribution Technology Based on OpenGL

Jiangeng Li[1,2], **Minjie Fan**[1,2], **Guangsheng Wang**[1,2], **Rihui Sun**[1,2]

[1] Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China;
[2] Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing 100124, China;

lijg@bjut.edu.cn

Abstract. The panoramic video contains a wealth of information that vividly reproduces the surrounding scene. However, the larger video information puts forward higher requirements for data transmission, interactivity and real-time. All the video information directly shows that the phenomenon of stagnation and black screen is easy to occur, and the user's traffic is greatly consumed, so that the payment cost of the terminal is large. In a traditional live video broadcast, the information displayed by the client is information with a fixed perspective. The client has no right to select the information, but only displays the obtained information. In order to solve the above problems, this paper has improved the traditional live broadcast mode. In this new live broadcast mode, the host side synthesizes the collected surrounding scene into a super wide-angle video. Then with the OpenGL technology, the host side performs the texture mapping, projection and other steps on the panoramic video. The host then distributes the video to the client. As a result, the client not only gets undistorted video, but also interacts with the server in real time. This approach greatly enhances the visual experience of the client.

1. Introduction

In recent years, along with the rapid development of Internet technology, the content of streaming media videos has become increasingly rich. All kinds of high-definition and ultra-high-definition video services are widely spread in the crowd and are gradually accepted by the public.

Sujan Ghimire et al. [1] introduced the latest media technology, using multiple GoPro cameras to create 360° video. K. Huang et al. [2] completed the design of a 360-degree panoramic video system. Gaoang [3] et al. proposed a hybrid streaming media distribution model. Liu Yuan et al.[4] proposed a lightweight transmission system to control network traffic and resource partitioning, and improve resource utilization. Wang Xinxiang has made improvements in load balancing [5]. Zhang Hanbing [6] combined cloud computing and big data processing architecture to solve the problems encountered in distribution. Liu Yulong [7] realized the realization of distributed management system for 4k streaming media fusion content.

For the OpenGL (Open Graphics Library) technology open graphics library or open graphics library, is a cross-language, cross-platform application programming interface (API) for rendering 2D, 3D vector graphics. Dave Shreiner [8] OpenGL Programming Guide: The official guide to learning OpenGL. Li Wenrui [9] designed and implemented a new image processing unit (IPU) based on OpenGL image processing algorithm. Zhou Fan [10] designed and implemented a panoramic video display system that can realize interaction based on OpenGL technology. Li Tingting [11] and others

based on the theory of three-dimensional images, based on OpenGL technology to achieve an interactive virtual roaming system; Yang Wei [12] and others explored OpenGL technology in the display of three-dimensional objects; ZhenHuan Zhou et al. [13] studied the 3D reconstruction technology of medical images based on OpenGL. Karsten Pedersen [14] greatly simplifies the development process by using OpenGL technology, Jian-Fei Chai [15] realized the 3D visualization monitoring system of the production line with OpenGL technology. Burkhard C [16] built a 360-degree virtual campus based on OpenGL, giving people visiting the site an immersive experience.

2. Introduction to OpenGL technology and related content

2.1. Introduction to OpenGL

OpenGL is essentially an image rendering API that has cross-platform features and can communicate directly with the GPU. OpenGL is designed as a modern, platform-independent interface, We can implement the OpenGL interface on a variety of different graphics hardware systems, or completely through software, without considering the system and window system. OpenGL itself does not contain any functions that perform window tasks or handle user input.

The technology is closely linked to the C language, with powerful portability and high-performance image rendering capabilities. OpenGL is a direct mode API, and each command produces some immediate effect based on the current rendering state. In OpenGL, use the glEnable() and glDisable() functions to enable and disable certain features of rendering.

Because of its independence from the platform, it defines its own data type. These data types will be mapped to regular C data types, these C data types can also be used directly in the program.The structure of a graphics processing system of a complete window system is generally shown in Figure 1. OpenGL is powerful, and it involves a lot of knowledge points and APIs. In the following pages, we will mainly introduce the more commonly used projection knowledge in OpenGL. The distribution operation of this article is a clever solution to the problem of distortion by means of projection.

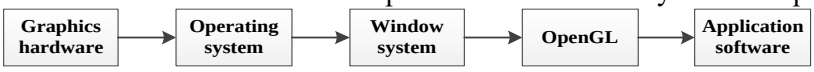


Figure 1. Hierarchy of OpenGL graphics processing systems

2.2. Introduction to OpenGL

The coordinate processing in OpenGL (Figure 2) includes model transformation, visual transformation, projection transformation, viewport transformation and the like.

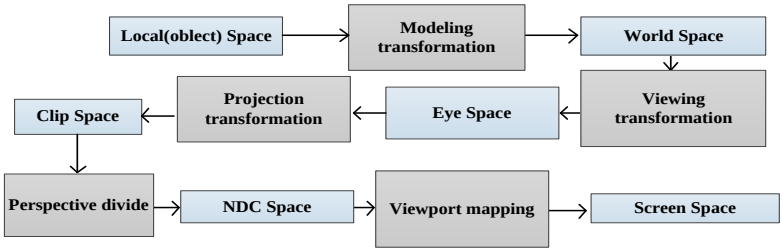


Figure 2. Coordinate processing in OpenGL

There are many ways to project, there are two main methods used in OpenGL, namely perspective projection and orthographic projection (Figure 3). In this paper, perspective projection is used.

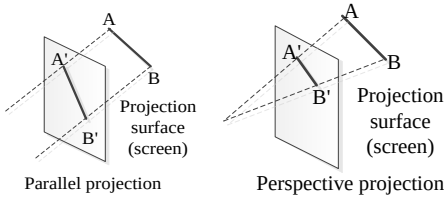


Figure 3. Parallel projection and perspective projection sketch

3. Introduction to related technologies of video transmission and processing

3.1. Streaming media technology

Streaming media technology is a technology that transmits multimedia data, such as voice and video, in real time over the network in a streaming manner. Streaming media communication protocols are based on UDP. For example, the RTP/RTCP protocol, RTP is a real-time transport protocol, which is a transport layer protocol designed by the IETF (Internet Engineering Task Force) for real-time multimedia applications such as interactive voice, video and analog data. Its purpose is to provide time information and stream synchronization to provide end-to-end services for multimedia applications. The RTP protocol consists of two protocols: the real-time transport protocol RTP and the real-time control protocol RTCP (real-time control protocol). The RTP protocol is responsible for transmitting data packets with real-time information. The RTCP protocol is the control part of the RTP protocol.

3.2. System overall design

The real-time playback system for panoramic video uses the C/S architecture, where C refers to the client and S refers to the server. The choice of this architecture stems from the strong ability of server control and easy interaction with the client. On the other hand, it is also convenient for future expansion of functions. Real-time playback of panoramic video not only meets the requirements of real-time playback on the server side, but also sends the video in real time according to the needs of the client. Therefore, the design of the system should fully meet the requirements of parallelism and stability. The general process of the system is as follows (Figure 4):

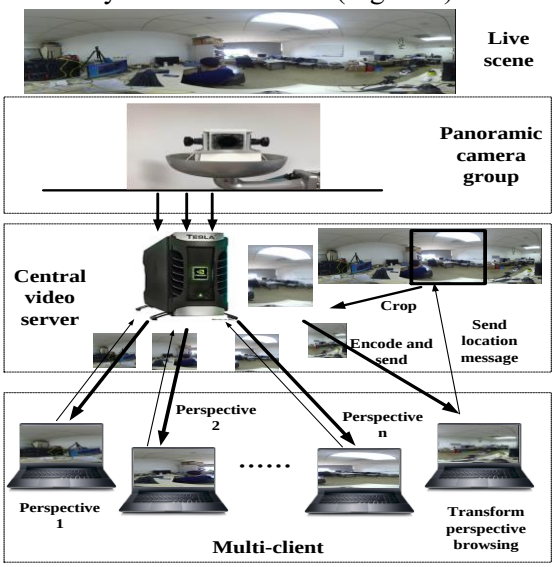


Figure 4. System architecture

The panoramic camera is responsible for collecting information from surrounding scenes and transmitting the captured video to the video server. The splicing of the video is completed on the server, and is sent to the client through the network through steps of cropping, encoding, and the like. The client's work is relatively simple, including sending location information, calling libVLC for display, and so on.In this system, the transmission of video data uses the UDP protocol, and the interaction of the location information uses the TCP protocol.

For the trimming part in the above image, we added OpenGL technology before this part of the operation. Below we will talk about the specific implementation.

The common projection format for panoramic video is ERP. The method is to project the panoramic video sphere by latitude and longitude coordinates onto a two-dimensional plane coordinate

system, and the generated image has an aspect ratio of 2:1. The method adopted in this paper is different from the previous one (Figure 5). After the video source is obtained on the server side, this method performs OpenGL texture mapping and projection before distributing.

Figure 5. Mapping schematic

For the OpenGL part, we mainly introduce two big blocks: one is to calculate the vertex coordinates and texture coordinates of the ball, and the other is to render this part, including initialization, setting projection matrix, real-time rendering, picture update and redraw. The details will be described below.

1.Calculating the vertex coordinates of the ball and the calculation of the vertex coordinates:
the calculation of the vertex coordinates and texture coordinates of the ball can be said to be the key to the panorama.

2.Rendering part: s

(1)Initialization: Initialization includes initialization of the queue, creation of textures, calculation of the coordinates of the ball's vertices and texture coordinates, linear filtering and other parameter settings. Also, you need to disable clipping, otherwise you won't be able to get inside the sphere because you can see the panoramic video inside the sphere.

(2) Set the projection matrix part: The projection uses a perspective projection so that it can enter the interior of the sphere. Here the angle is set to 45, you can set it yourself, but the effect is not too good.

(3) Rendering: The rendered data is retrieved from the queue to generate a new texture during rendering. Rendering uses the glDrawArrays() function, using the GL_TRIANGLES parameter. Using this parameter does not need to be considered a lot for calculating the vertex coordinates and texture coordinates of the ball.

(4) Screen update and redraw: Here, the OpenGL timer is used to make a timed update of the picture, thereby realizing the effect of video playback.

In addition, the mouse control is added to process the mouse click.

Through OpenGL technology, we project the panoramic view of the plane into a spherical image, and then use the perspective projection to capture the real-time video in the window. The user can drag the mouse in the operable area of the client to get the view that I want to see. So for the client, the size of the video that is always captured at any moment, always the size of the OpenGL window captured, is always an area in the panoramic video. The location where the window appears, that is, the location of the captured panoramic video, is determined by the client. Whenever the viewer requests to change the area shown, the client sends a location message to the server recording the new area that the viewer is requesting to view. The server receives this location message, moves the window to the new location, and transmits the panoramic video area on the new location to the client. In this way, when the remote viewer requests to change the observation area, the client sends the location message, the server immediately changes the position of the window according to the location message, and transmits the video area of the new location to the client through the network, and the client plays the video of the new location area. come out. In this way, the remote audience controls his sights, as if sitting at the scene, free to watch any area of the scene, immersive. The key points involved in this are roughly as shown in the Figure 6 (the clipping part shown in the figure, we capture the OpenGL mapping to the window on the desktop):

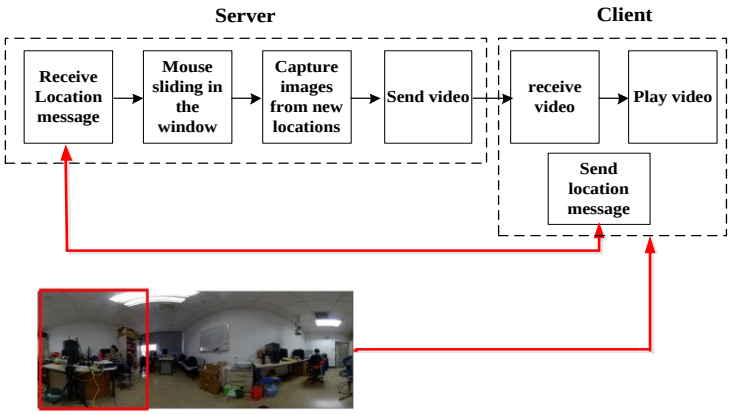


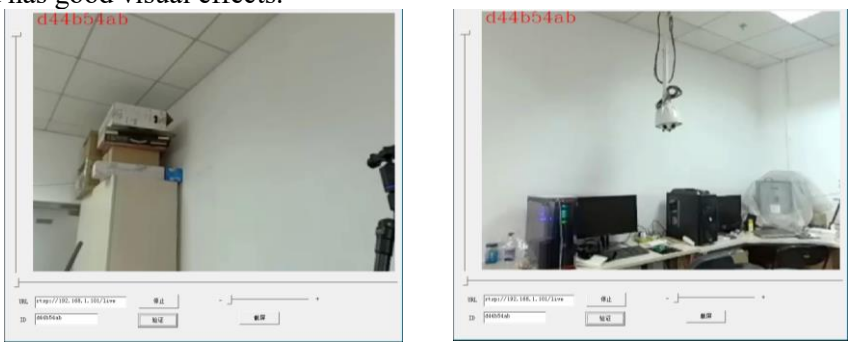
Figure 6. Sight control flow chart

4. Display effect and analysis

Table 1. Experimental configuration table

Hardware name configuration	Configuration
Operating system	Win 7 64 bit
Graphics card model	NVIDIA GeForce GTX 1080
CPU	I7-7700k

For the display of the results (Figure 7), we used the designed PC-side playback software, which can meet the needs of users to view any position. At the same time, the video played is real-time, without distortion, and has good visual effects.



(a) (b)

Figure 7. PC software play screenshot

5. Conclusion

With the development of computer network technology, people's demand for the network has also increased accordingly. The panoramic video contains a wealth of information that vividly reproduces the surrounding scene. However, when the network side directly transmits large video information under the condition that the network bandwidth is limited, the client is prone to jamming and black screen. This article uses OpenGL technology to send the corresponding video information to the client according to the client's request. The results show that the client can not only get distortion-free video, but also interact with the server in real time.

References

[1] Sujan Ghimire 2016 Production of 360° video: Introduction to 360° video and production guidelines Helsinki Metropolia University of Applied Sciences (Helsinki: Helsinki Metropolia University of Applied Sciences Press)
[2] Huang K C, Chien P Y, Chien C A, Chang H C and Guo J I 2014 A 360-degree panoramic video system design *Technical Papers of 2014 Int. Symp. on VLSI Design, Automation and Test* pp 1–4
[3] Gao A, Xu S, Li Z and Li M 2013 P2P streaming media distribution model and its application in video conference system *J. Southwest Jiaotong University* pp 88-93
[4] Liu Y and Wang R 2014 Research and implementation of P2P network content distribution centralization *J. Xi'an University of Arts and Science:* Natural Science Edition **17** 105-9
[5] Wang X 2013 Research and design of streaming media distribution system based on P2P Science and Technology Plaza pp 113-6
[6] Zhang H 2016 Research and application of key technologies of streaming media distribution based on cloud computing Graduate School of Chinese Academy of Sciences (Shenyang: Shenyang Institute of Computing Technology Press)
[7] Liu Y 2017 Research and implementation of converged content distribution management system for 4k streaming media Southeast University (Nanjing: Southeast University Press)
[8] Shreiner D, Sellers, G, Kessenich J M and Liceakane B 2013 *OpenGL Programming Guide: The Official Guide to Learning OpenGL Version 4.3* Addison-Wesley
[9] Li W 2017 Design and implementation of image processing unit based on OpenGL2.0 standard Xidian University (Xi'an: Xidian University Press)
[10] Zhou F, Zhang W and Yang F 2018 Interactive panoramic video display system based on OpenGL LCD and Display **33** pp 808-15
[11] Li T, Yu Q 2018 Development of interactive virtual roaming system based on OpenGL *Computer Technology and Development* **28** pp 131-3
[12] Yang W and Li W 2018 Discussion on the display process of 3D objects in OpenGL *Information Record Materials* pp 106-7
[13] Zhou Z and Wen X 2018 3D reconstruction of medical image based on OpenGL *J. Computers* **29** 249-60
[14] Pedersen K, Gatzidis C and Tang W 2018 OpenGL| D-an alternative approach to multi-user architecture *Trans. Computational Science* **32** 57–74
[15] Chai J, Hu X, Qu H, Li M, Xu H, Liu Y and Yu T 2018 Production line 3D visualization monitoring system design based on OpenGL *Advances in Manufacturing* **6** 126–35
[16] Liu Y and Zhang H 2018 Three-dimensional campus 360-degree video encoding VR technology based on OpenGL *Multimedia Tools and Applications* pp 1–9