Research on Data Synchronization Algorithm in 3D Virtual Scene

Guo Jianlong*
Training and Evaluation Center of Guangdong Power Grid Company
Guangdong 510520, China
E-mail: 1850508590@qq.com

Feng Weixia
Training and Evaluation Center of Guangdong Power Grid Company
Guangdong 510520, China

Hao Tengfei
Training and Evaluation Center of Guangdong Power Grid Company
Guangdong, 510520, China

Wen Manhua
Training and Evaluation Center of Guangdong Power Grid Company
Guangdong 510520, China

Abstract—**With the deepening in the combination of virtual reality technology and many fields, 3D virtual scenes have been maturing, which have brought a lot of convenience to modern applications. However, there are still many shortcomings and imperfections in today's 3D virtual scene system. For example, since interaction speed of the system is directly affected by a large number of models in virtual experiment scene, the experimental effect will be greatly reduced. Based on the virtual scene system, the data synchronization parallel operation method is used to realize multiple direction data transmission in this paper. In order to avoid the situation where that the data synchronization is not in place affects the virtual environment, the C/S architecture mode collaborative synchronization control strategy is adopted to ensure that the data received by each port is synchronized in real time.**

Keywords-3D Virtual Scene; Data Synchronization; Control Strategy

I. INTRODUCTION

The establishment of virtual experiment environment has always been a hot research topic in the field of network computing technology, and people’s thinking has been changed by the wide application of virtual reality technology. What’s more, virtual experiment system can be performed freely without the constraints such as time and space with virtual reality technology. Combined with a variety of computer technologies, virtual reality technology can enable experimental users to be immersed and get the most real feelings through the real-world three-dimensional simulation and expression from the designer's perspective[1-3].

The virtual experiment system combined with virtual reality technology has created a three-dimensional virtual experiment method, which makes microscopic experiments, dangerous experiments, and expensive experiments possible. Moreover, the virtual experiment system provides users with a platform for experimentation, technical exchange, joint research and collaborative work, which can not only assist scientific research, but also contribute to experimental teaching. In addition, Lianguan Shen et al. proposed the concept of multiple roles to distinguish operational rights, each of which provides specific permissions and services in 2007. Besides it, Jara et al. proposed a real-time collaborative virtual lab to realize simultaneous viewing among multiple people in different rotations in 2009, and VR technology has also been used to support various system environments for a long time. In addition, VR technology was used by Petrakou to realize virtual virtual interactive learning for college students. One of the recurring questions in VR-supported learning is how the technology empowers various scenarios, and in this regard, Miyata et al. proposed to establish interdisciplinary groups to promote the creativity of developing VR applications and laboratory skills. The goal of the virtual collaborative experiment platform is the collaboration based on network communication, which can not only meet the needs of the team members to cooperate in the completion of an experiment, but more importantly, it is performed under the premise of synchronous operation. However, the related research that the collaborative technology is directly applied to the virtual experiment environment is still lacking, and there are still problems that need further research and in-depth exploration in the perspective of coordination mechanism, authority control and management under the virtual experiment environment[4-6].

In view of the above problems, Unity platform is used in this paper to develop a three-dimensional virtual scene environment, and combined the computer-supported coordination mechanism with the virtual experiment system based on VR technology, data synchronization parallel operation method is adopted to realize multiple directional data transmission. In order to avoid the situation where that the data synchronization is not in place affects the virtual environment, the C/S architecture mode collaborative synchronization control strategy is adopted to ensure that the data received by each port is synchronized in real time.

II. Architecture Design on Network Collaborative Virtual Experiment

In the collaborative virtual experiment scenario, the system can support multiple experimental users to coordinate operations, transfer information to each other, and complete experiments according to the division of labor. The cooperative operation here means that when a member operates the experimental equipment, the operation action can be simultaneously displayed in the paintings of other members in the experimental group so that the experimental user can perform experiments while communicating as in traditional laboratory[7-9].

C/S architecture mode is adopted in this paper where most of the data storage and calculation of the system are performed in the server, and client user needs to establish a dedicated client menu interface. Moreover, C/S architecture is suitable for local area networks, but the requirements for network speed are relatively high. The collaborative virtual experiment system is mainly divided into experimental client which is developed on the Unity platform and mainly responsible for the construction of virtual experiment scenarios, model display, server data transmission as well as communication with other users and experimental server which mainly provides online user management.

A. Real-Time Synchronization Control Strategy

When a new experimental member participates in the experiment, the real-time synchronization control module will detect whether the current experiment is in progress, and if it is in progress, the real-time synchronization control module will send a request command to the teacher with the highest operation authority to notify the processing request. Meanwhile, the highest-privileged instructor needs to lock all current online users and send the current experimental status to the new user. The new client will parse the message and create the current experimental state object, which is synchronized with the experimental state of the collaborative group user. What’s more, time-based consistent ordering strategy is used in this paper to solve the problem that there is operational inconsistencies in timing which is caused by network delay.

A logical clock is created by all members in the collaborative group, which counts from 1 with a variable (Clock) with a period of 3 ms (since the LAN is used, the network delay is generally 1 ms). Then operation time of each member is sorted according to the logical clock so that the order of the operation sequences among all the experimental members is consistent, and if an operation is generated while sorting, a priority policy can be introduced. Moreover, according to the time of joining the collaborative group, the first joiner has high priority. Through these two methods mentioned above, the data received by each client is synchronized in real time. The collaborative communication architecture is shown in Figure 1.

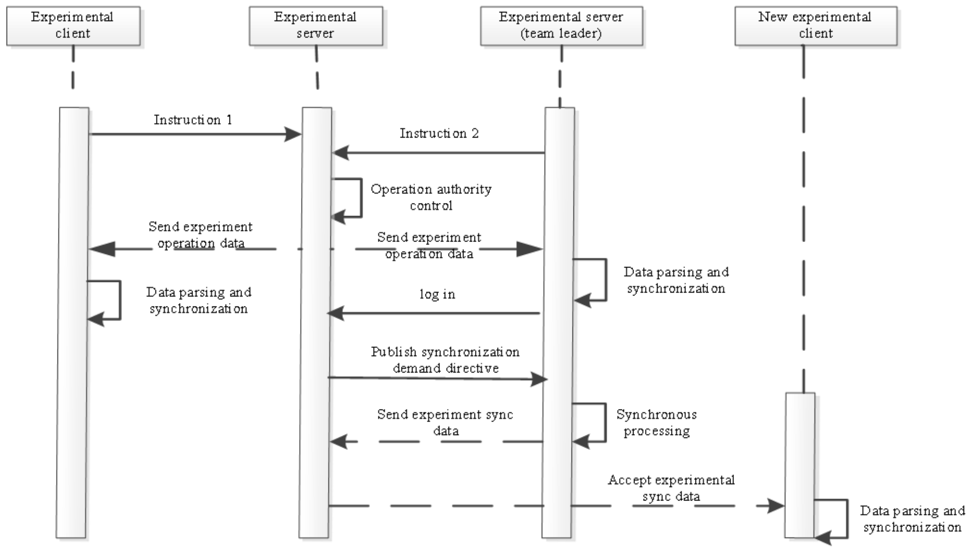

Figure 1. Client and Server Interaction Diagram

B. *Processing of Scene Data*

Since there may be differences in the format and type of scene data with different sources, and there may be a large amount of invalid data and erroneous data in the original data, it is necessary to properly process the scene data to form a truly available auto-driving car test scenario, and the key to scene data processing lies in the deconstruction and reconstruction of scene elements.

There are seven steps for scene data processing in German PEGASUS project. They are generating general environment descriptions, checking data formats, generating additional information, analyzing the degree of association among scenes, analyzing the possibility of scene occurrences, clustering logical scene data as well as calculating frequency distribution, and generating a specific test scenario based on the generated logical scenario.Besides it, Baidu Company in China proposed a three-step method of scene clustering including scene classification rule definition, scene label (element decomposition, quantization) and label clustering.

According to the existing typical scene data processing method, scene data processing flow is summarized and proposed in this paper, which is shown in Figure 2[10].

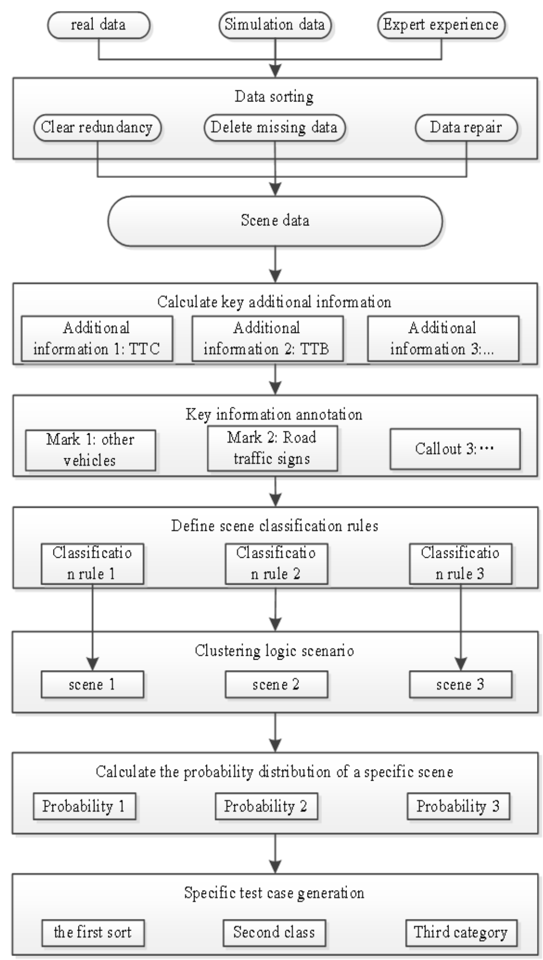

Figure 2. Scenario Data Processing Flow

Step 1: Collected scene data is cleaned, including clearing redundancy, deleting missing data, and repairing data where manual completion of key information or repair according to statistical rules of data can be performed[11-12]. What's more, following conditions should be met during the data cleaning. Maintain data integrity constraints should be maintained, and appropriate data cleaning rules should be developed to meet user needs. Then, the cleaning cost should be minimal under the premise of meeting all data quality requirements. Taking data repair as an example, the expression of the cleaning cost is

$$C_{ost}(t) = w(t)\sum_{A\in R} D_{istance}(t_A, t'_A) \quad (1)$$

$$C_{ost}(I) = \sum_{t\in I} C_{ost}(t) \quad (2)$$

where t refers to a single data tuple, $w(t)$ is the proportion of data tuple t in all data tuples, and I represents the sum of all data tuples. Besides it, $D_{istance}(t_A, t'_A)$ is the distance from element t_A to the repaired element t'_A where Damerau-Levenshtein distance algorithm is used.

Step 2: The cleaned data are organized to form a usable scene data set.

Step 3: The key additional information is calculated for the scene.For critical information that cannot be obtained directly from the sensor such as Time to Collision, Time Headway, and Time to Brake [13-14], calculations are required, .

Step 4: The key information in the scene elements is labeled. Annotation methods commonly used include semantic analysis based methods, semi-supervised learning based methods, and Bayesian learning based methods.

Step 5: The scene classification rules is defined. Distribution rules of the scene feature parameters are analyzed, and the scene classification rules are established according to the measured autopilot function requirements, such as the vehicle speed, the cut-in vehicle speed, and the cut-in position in the dangerous scene.

Step 6: Logical scenes are clustered. Scenes that meet the classification rules are clustered into corresponding logical scenes to clarify the parameter space of the scene elements, and clustering algorithms commonly used mainly include K-Means clustering, hierarchical clustering, and mixed Gaussian Model et al. As for clustering of auto test vehicle in test scenarios, LI et al. proposed a method for automatic clustering of traffic basic scenes based on K-Means[15].

Step 7: Probability distribution in the specific scene is calculated. Kernel density function of the logical scene is calculated according to the above scene data so as to facilitate the random generation of subsequent specific scenes. Supposing $x_1, x_2, \cdots, x_n$ is n scene sample point which is distributed independently and whose probability density function is f, the nuclear density D is estimated as

$$\hat{} f_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \quad (3)$$

$$K_h(x) = \frac{1}{h} K\left(\frac{x}{h}\right) \quad (4)$$

where K is a kernel function whose non-negative and the integral value is 1, h indicates smoothing parameter determined by the average integral squared error, and K_h refers to scaling kernel function.

Finally, the random generation of test cases is performed according to the specific scene probability distribution.

III. Data Transmission

A. Audio Data Transmission

A prototype system is developed in this paper where the online user volume is not very large at the same time, and it will not cause too much pressure on the server. Therefore, the single-service C/S architecture is adopted in which all the logic on the server side is completed on one server host.The implementation process of remote audio transmission under such a network framework can be roughly described as follows. The client 1 collects and serializes the audio signal, and sends it to the server. Through the real-time broadcast function of the server, the audio signal is synchronized to other clients that need to receive audio signals, such as client 2 and client 3, where the server acts as a data relay station. Finally, Client 2 and client 3, which receive the audio signal, will deserialize it and play it in real time.

The transmission of audio signals is mainly for the user to conveniently carry out remote real-time voice calls in this system. Moreover, the acquisition of client audio data needs to be done through using the Microphone where Unity provides functions such as acquiring a local microphone device, starting recording, stopping recording, obtaining recording samples as well as location and Audio Clip in the official Unity API. In addition, it is used to obtain the local recording device and control the establishment and shutdown of the audio input stream channel. Then, the acquired audio data will be stored as an instantiated object in the Audio Clip class and sent to the director or the intern by the server through the recording data compression. After the other end receives the data, it also decompresses the audio data and plays it out, which is completed by the Zlib compression algorithm.

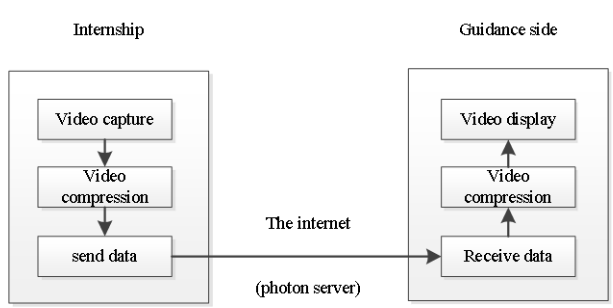

Figure 3. Video Transmission Process

C. *Virtual Scene Information Synchronization*

The synchronization of the virtual body posture information is included in the synchronization of the virtual scene information, and it is necessary to select an appropriate transmission protocol for transmitting the scene information to realize the information synchronization of the virtual

B. *Video Data Transmission*

The transmission of the video signal is mainly for the guidance doctor to see real-time pictures of the virtual patient and the real patient which are taken through the mixed reality helmet in internship. Additionally, obtaining the real patient image is for the guidance doctor to determine whether the real patient posture at the internship meets the requirements of knee operation. If it is not met, Leap Motion will adjust the virtual patient posture superimposed by hand, which can be observed by the practice doctor in real time through the mixed reality helmet. The intern only needs to adjust the real patient posture lying on the operating table to coincide with the virtual patient.

The general design idea of video transmission is shown in Figure 3.At the interns client, real-time video frame data in bitmap format is obtained from the Leap Motion API, and the video frame is compressed by Divx encoder which is a video compression technology that compresses video images to 10% and makes it ideal for compression and decompression in network video transmission.Then data are then transmitted to the Photon server and forwarded to the director waiting to be received through the server. Moreover, the director that receives the video frame also needs to decompress it in real time through DivX, and finally the video is displayed through the Oculus helmet.

scene. JSON is adopted as the transmission protocol of the virtual scene information synchronization link in this paper .

The scene information that needs to be synchronized in real time is mainly the action information of the Leap Motion virtual hand model and the virtual patient model with both hands in this system.Taking the Leap Motio virtual hand as an example, as shown in Fig. 4, the information to be synchronized is a spatial six-degree-of-freedom coordinate value in the local coordinate system among each joint point of the model with respect to the world coordinate. What's more, there are a total of 16 joint points whose information can be described by a composite JSON object in this hand model. .

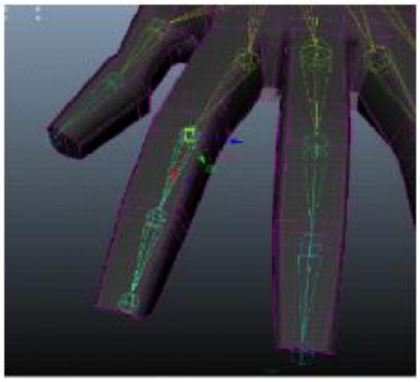

Figure 4. Leap Motion Virtual Hand Joint Point Local Coordinate System

The specific code is as follows: {

```
"thumb":
    [
    {"joint":"bonel","position":[40,210,30],"rotation":
        [90,61,105]},
    {"joint":"bone3","position":[81,25.4,30],"rotation":
        [21,35,76]},
    {"joint":"bone3","position":[36,21,38],"rotation":
        [66,24,18]},
    ],
"index":
    [
    {"joint":"bonel","position":[93,24,88],"rotation":
        [101,31.3,105]},
    {"joint":"bone2","position":[33,64,79],"rotation":
        [85,61.1,135]},
    {"joint":"bone3","position":[65,94,25],"rotation":
        [90,21,10]}
    ]
    ......
}
```

The JSON object whose outermost layer is a variate consisting of a finger name and a composite JSON array composed of three composite objects, each of which is stored separately is the transmission model of the virtual hand model data.The finger joint name, the spatial coordinate value in the virtual world coordinate system, and the rotation angles of the three coordinate axes around the world coordinate system X, Y, Zare shown in Table 1, and so is the data structure and data type in the joint point object.Meanwhile, due to the large content of the transmission model, only the JSON model data of the thumb and forefinger are listed in the above composite JSON object, and middle finger, ring finger, little finger, palm, and the forearm are omitted. What's more, synchronization between the practice side and the guidance side of the virtual scene data can be realized by the composite JSON object.

TABLE I. JSON OBJECT DATA STRUCTURE OF EACH JOINT POINT

Type of data	Name
string	Node name
double□	X-axis coordinate
	Y-axis coordinate
	Z-axis coordinate
double□	Rotation angle around the X axis
	Rotation angle around the Y axis
	Rotation angle around the Zaxis

IV. APPLICATION EXAMPLES AND ANALYSIS

As shown in Figure 5, in the remote collaboration guidance platform, the system automatically activates the built to detect the environment during the scene initialization phase when the collaboration group is created and interconnected, and the data are synchronized to the virtual scene of the console and client where the data need to be matched.

Figure 5. 3D Virtual Scene

Real person is observed from the perspective of the top of the platform, and superposition of false and real people is observed through a mixed reality helmet as well. Moreover, the guiding end adjusts the virtual pose by driving the Leap Motion virtual hand in the virtual scene, and the client has adjusted the spatial position of the virtual object to completely coincide, which indicates that the gesture has met the guiding standard. According to the data sampling analysis, it is as shown in Figure 6.

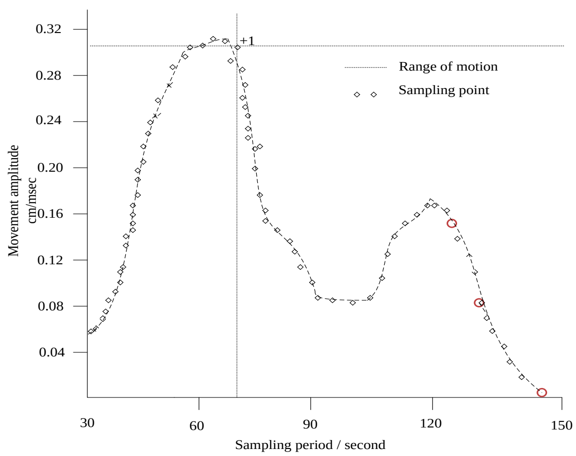

Figure 6. The data sampling analysis

The sampling points in the three-dimensional space scene are consistent with the real motion range, where the synergy rate reaches 99.9%.

V. CONCLUSION

Unity platform is used to build a virtual lab environment and VR technology is integrated to enhance the immersion of virtual experiment users in this paper. In order to realize the collaborative function requirements on the experimental system, not only collaborative communication mechanism of the experimental client but also the collaborative management and real-time synchronization mechanism of the experimental server is designed to realize the goal that multiple people can operate together and complete experiments in different Unity clients in this paper. In addition, it is also a new attempt to apply virtual reality technology to various fields that combines augmented reality technology, remote collaboration technology and virtual projects, which can be extended to areas such as industrial assembly manufacturing and medical surgery.

REFERENCES

[1] 3D scene real-time rendering technology based on LOD texture mapping [J]. Zhang Min, Liu Yongjiang. Journal of Minjiang University. 2017(05).

[2] Design and integration of intelligent manufacturing virtual experiment system[J]. Bai Ruifeng, Han Honghong, Yu Heyang, Fang Zhaohui. Experimental Technology and Management. 2016(06).

[3] Construction of virtual simulation experiment system based on Unity3D gas chromatograph[J]. Zhou Sijie, Yang Zeliang, Dong Zihe, Zhou Mingda, Zeng Dongming. Chromatography. 2016(06).

[4] Application and development trend of sensor technology in automatic control system [J]. Zhou Fangfang. Technology and enterprise. 2016(05).

[5] Digital gesture recognition based on LeapMotion [J]. Lin Shutan, Yin Changqing. Computer Knowledge and Technology. 2015(35).

[6] Research on 3DS MAX and Unity3D campus virtual roaming system[J]. Li Xu, Wang Shifa, Ci Weitao, Si Chunjing, Yao Jianghe. Journal of Tarim University. 2015(02).

[7] Texture mapping method for multi-view 3D reconstruction [D]. Zhu Linan. Zhejiang University 2018.

[8] High-quality texture mapping based on gap optimization [D]. Luo Likang. Zhejiang University 2018.

[9] Research and implementation of free gesture interaction based on Leap Motion in 3D model transformation [D]. Yan Siyang. Southwest Jiaotong University 2017.

[10] Research and development of remote virtual experiment teaching system for hydraulic excavator tooling control [D]. Liu Weiping. Harbin Institute of Technology 2017.

[11] Design and implementation of real-time production monitoring system based on virtual and real integration [D]. Liang Xingming. Hefei University of Technology 2017.

[12] Research on intelligent manufacturing unit monitoring system based on CPS [D]. Wang Xiang. Harbin Institute of Technology 2016.

[13] Design and implementation of high school physics virtual experiment based on Unity3D [D]. Zhang Li. Hunan Normal University 2015.

[14] Advanced Course in VR and AR Development [M]. People's Posts and Telecommunications Press, Wu Yafeng, 2017.

[15] Applying augmented reality to enhance learning: a study of different teaching materials[J]. Y.‐H. Hung, C.‐H. Chen, S.‐W. Huang. Journal of Computer Assisted Learning . 2017 (3).