

Research on dynamic synchronization of multi channel three dimensional visual simulation system

Yongqiang YAO^{1,2}, Yaojie CHEN^{1,2}, Yating XUE^{1,2}, Xingliang ZHU^{1,2}

¹ College of Computer Science and Technology, Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, Wuhan University of Science and Technology, Wuhan 430065, Hubei, China;

² College of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, Hubei, China

yaosir1993@qq.com

Abstract—In order to guarantee the synchronization of multi-simulation entities in the multi-channel 3D visual simulation system, the dynamic synchronization method is of great significance. By improving the traditional bucket synchronization algorithm, the paper presents and implements the frame bucket lock synchronization algorithm, which effectively solves the problem of image ghosting and tearing that may occur when the moving entity traverses the critical region of the three-dimensional virtual simulation system of multi-channel ship driving. Ensure the critical channel screen seamless stitching. By introducing the RakNet network engine to ensure the reliability of data transmission, a motion smoothing algorithm with cubic spline interpolation is proposed, and an effective method to deal with the smoothing of the ship is given. The experimental results show that the proposed algorithm is feasible and feasible to solve the multi - channel synchronization problem of multi - athletic entities in three - dimensional ship driving virtual machine, and the display effect is better.

Keywords—dynamic synchronization method; multi-channel synchronization; frame bucket locking synchronization algorithm; motion smoothing algorithm

I. INTRODUCTION

Multi-channel display system refers to the use of one or more display channels together to generate a complete view of the screen, to provide users with a highly immersive virtual environment^[1], such as 360° three-dimensional ring screen. Compared with the general single-channel display, it has a larger display size, more display content and better visual effects. When displaying the same frame of the scene, each channel needs to work together, unified to accept the server scheduling, together to complete the rendering task. Multi channel synchronous display of traditional display system often only focus on the static scene, when the scene also includes a moving entity, often appear adjacent to the channel screen dislocation, deviation and other issues. Therefore, the stable and reliable dynamic synchronization technique is an important foundation for ensuring the synchronization of moving entities in visual simulation.

In literature [2], it presents an improved barrel synchronization algorithm based on C/S structure, to eliminate the delay synchronization error to achieve the client rendering system, and put forward in the client cache mechanism is introduced to solve the network jitter problem. In literature [3], the data transmission control information is used to control the static

scene and the dynamic entity solution respectively. The static data and the dynamic data are separated, and the feedback method of the adaptive waiting time is proposed to control the three-dimensional scene of each channel synchronous refreshing.

With the rapid development of virtual simulation technology, more and more dynamic scenes to join, in the rich display at the same time, but also increasingly highlight the traditional synchronization method have been unable to meet the complex real-time synchronization of real-time synchronization of the scene.

This paper studies the dynamic synchronization problem of multi-channel 3D visual simulation system. Through the introduction of the improved CMSS system architecture to optimize the system structure, for the same simulation entities (such as ships) in the adjacent channel display screen ghosting problem, proposed and implemented frame barrel lock synchronization algorithm. By introducing the RakNet network engine to optimize the network performance, aiming at the jitter problem caused by the inconsistency of the simulation state of the simulated entity in the complex scene, the motion smoothing algorithm based on the cubic spline interpolation is proposed to smooth the moving entity, and finally in the three-dimensional virtual simulation system which solves the dynamic synchronization problem of multi - channel multi - motion ship.

II. MULTI - CHANNEL VISUAL SIMULATION SYSTEM - CMSS ARCHITECTURE

Generally speaking, the traditional Client/Server model^[4] can effectively reduce the pressure of the client, but the real-time performance is poor, and the Master/Slave network data transmission is small, the simulation entity of response speed of state changes is extremely fast, but it also increases the calculation pressure of each node . This paper put forwards a dual architecture of CMSS based on synchronous data update and visual assignment. CMSS architecture visual simulation system structure shown in Fig.1,The data synchronization is consisted of M/S model composed of each rendering node, and the view synchronization is consisted of C/S model, each rendering node and the visual server through the 100M Ethernet connection to form a local area network. The synchronization data mainly refers to the ship position information, and the scene synchronization refers to the channel scene information synchronization.

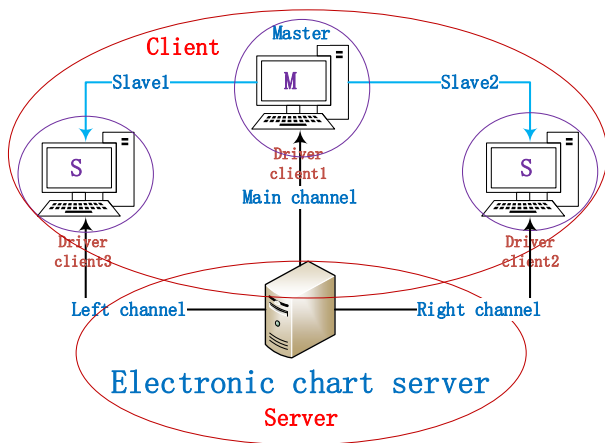


Fig.1. The overall system structure diagram.

In this paper, three-channel ship driving three-dimensional virtual simulation system is regarded as an example to study. In the three-dimensional ship simulation system, the computer running the electronic sea chart program is regarded as the visual server, and the other three computers running the 3D ship visual simulation system developed by the OSG 3D rendering engine^[5] respectively. The specific responsibilities of the modules in the above structure can be divided into the following three parts:

- 1) The master node receives the operator's interworking information (e.g., the orientation of the ship, the view information) along with the synchronization control information to the control information flow to all slave nodes.
- 2) The slave node is responsible for receiving the control information and parsing the synchronization information flow.
- 3) The visual server is responsible for distributing the scene packets to all nodes.

In the process of data exchange between three-channel ship driving three-dimensional virtual simulation system, each node of the system will use frame bucket synchronization algorithm to achieve the dynamic synchronization of scene data, and adopt the motion smoothing algorithm based on cubic spline interpolation to achieve the smooth display of moving entity.

III. VIEW DYNAMIC SYNCHRONIZATION TECHNOLOGY

A. Frame bucket synchronization algorithm

The bucket synchronization algorithm^[6-7] was proposed by C.Diot and L.Gautier, which was first used in multiplayer online games MiMaze for DIS-based simulation based on serverless P2P architectures. In this paper, some of the ideas of bucket synchronization algorithm are borrowed, and the synchronization algorithm of frame bucket locking is proposed based on the idea of online game synchronization.

The frame-by-lock synchronization algorithm divides the system timeline into a number of equally spaced bucket^[8]. The bucket is used to store the control commands and location up-

date data waiting to be sent to the Slave. The master is responsible for maintaining the changes in the bucket. At the end of each Bucket cycle, the Master broadcasts the bucket to all Slaves, and the Slave receives the bucket and fetches the instructions from it, verifying it after execution. In order to reduce the impact of network latency, master control instructions generated by the user operation will be in accordance with certain rules to the corresponding bucket. Which barrel bucket synchronization algorithm in the bucket synchronization strategy to ensure that the scene synchronization, locking strategy to ensure data synchronization^[9]. The predefined parameters of the frame bucket synchronization algorithm are shown in Table I below:

TABLE I. THE NUMBER OF SCANNING LINES UNDER THE TWO METHODS OF COMPARISON

Parameter definition	Parameter Description
OrderStart	Instruction generation time
AvgDelay	Average delay time
GatherStart	Collect the instruction start time
ActionStart	Instruction execution start time

Frame bucket synchronization algorithm specific steps are as follows:

- 1) Master accepts the control instructions generated by the user's operation, if $orderStart + avgDelay \leq gatherStart$ will deliver the instruction to the bucket that is currently collecting the instruction.
- 2) When the actionStart arrives, the master will determine whether the current bucket has the ship position information, and if so, broadcast the control instruction in this bucket to all the slaves, otherwise the broadcast lock status information, the system enters the locked state until the master receives the new The location information, the party broadcasts the unlock state information, and updates the bucket cycle, the new bucket cycle is the sum of the current bucket cycle and the lock time.
- 3) Every other bucket cycle, Slave receives the latest control instruction information from Master. When actionStart arrives, Slave updates the visual data and keeps the scene synchronized.

Frame bucket synchronization algorithm The total flow shown in Fig.2(a) and Fig.2(b):

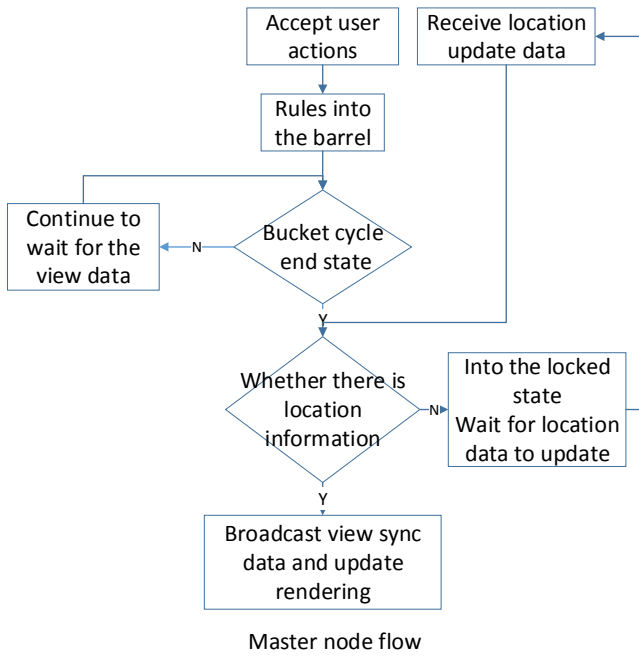


Fig.2(a). Frame bucket locking synchronization algorithm Master node flow.

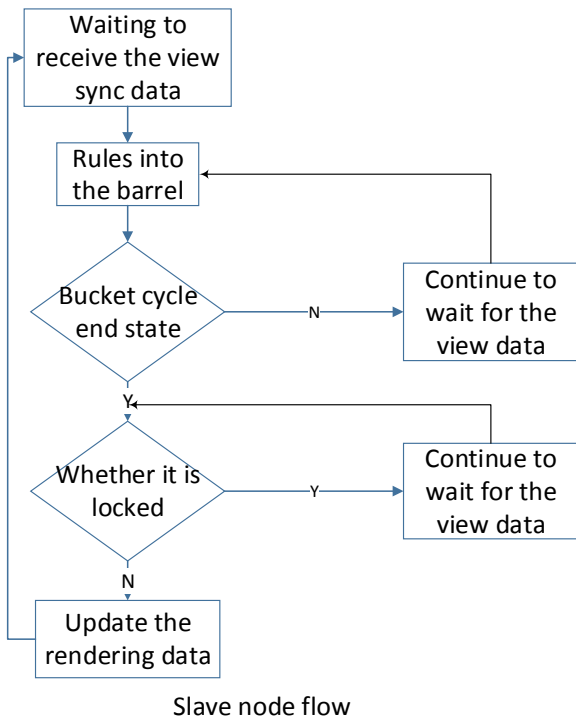


Fig.2(b). Frame bucket locking synchronization algorithm Slave node flow.

B. RakNet Network Engine

Although the use of UDP protocol to transmit data can ensure that the visual host rendering the same DDU (command data unit) static scene at the same time, but it will still appear when the screen jitter Phenomenon when the moving target (such as the movement of the ship) through multiple channels, after comparing the frame rate between adjacent channels to

confirm the phenomenon is caused by the unreliability UDP protocol.

By introducing the most popular RakNet network engine in the field of game development area, RakNet can realize high-reliability UDP communication and cross-platform high-level network programming, which effectively reduces the packet loss probability when the network is transmitted and ensures the data of each channel's consistency requirements^[10], and it provides the algorithm which can compress the network data, that reduces the time when transmit the large quantities and complex task in the three-dimensional virtual simulation system, which increased system's stability and high Availability, it provides a high quality network service guarantee for the visual scene dynamic synchronization.

C. Dynamic Smoothing Algorithm

After the network engine optimization, the client (composed of each rendering node) from the visual server to receive the location data already has a certain reliability, but there will be a small probability of deviation exists. We found that the ship was operating normally under static conditions in the three-channel ship driving three-dimensional virtual simulation system, but there was an intermittent random shake in the ship's movement. After several tests, it was found that the problem was due to the fact that the new data sent by the visual server was intermittent with the current frame data. Therefore, it is necessary to smooth the position data of the jumping ship and eliminate the influence of the synchronous display of the moving ship on the scene system^[11].

In view of the stable and better smoothness of cubic spline interpolation, it can meet the streamlined design of ship in practical application, and the interpolation path is more real and natural. Therefore, this paper proposes a motion smoothing algorithm based on cubic spline interpolation^[12].

The cubic spline interpolation is the second order continuous differentiable cubic polynomial function of the sample. Let the function $f(x)$ give a function value at $n+1$ mutually $x_i (i=0,1,\dots,n)$ different points $f(x_i) = y_i$ where if the function $s(x)$ satisfies the condition $a=x_0 < x_1 < \dots < x_n = b$

- 1) $s(x)$ is a triplet polynomial on each subinterval $[x_{j-1}, x_j] (j=1, 2, \dots, n)$
- 2) $s(x)$ has a continuous derivative up to the second order $S(x) \in C^2[a, b]$ on each inner node $x_j (j=1, 2, \dots, n-1)$, $s(x)$ is called the cubic spline function on the node x_1, x_2, \dots, x_n .

If the interpolation condition is satisfied on all nodes:

$$S(x_j) = f(x_j) = y_j (j = 0, 1, \dots, n) \quad (1)$$

Then called $s(x)$ for the cubic spline interpolation function^[13].

In this paper, we consider the case of the first derivative value of the two endpoints, and the endpoint value is set to 0 in the actual experiment. According to the definition of the cubic spline interpolation, the spline interpolation is obtained by using the two-point Hermite interpolation formula. The expression for the function in $[x_{k-1}, x_k]$ is:

$$\begin{aligned}
 s(x) = & m_{k-1} \frac{(x-x_k)^2 - (x-x_{k-1})}{h_{k-1}^2} \\
 & + m_k \frac{(x-x_{k-1})^2 - (x-x_k)}{h_{k-1}^2} \quad (2) \\
 & + \frac{y_{k-1}(x-x_k)^2 [2(x-x_{k-1}) + h_{k-1}]}{h_{k-1}^3} \\
 & + \frac{y_k(x-x_{k-1})^2 [2(x-x_k) + h_{k-1}]}{h_{k-1}^3}
 \end{aligned}$$

Where $h_{k-1} = x_k - x_{k-1}$ is determined, so that the cubic spline interpolation function on each interval can be determined by determining the first derivative value $m_k (k=0,1,2,\dots,n)$ of each node^[14]. In the multi-channel ship driving three-dimensional virtual simulation system, the ship in the navigation of the simulated interpolation experiment, because the visual server is based on the actual parameters of the ship simulation of the location coordinates ShipOld (shipdata.x, shipdata.y), so here only need ShipNew(Shipdata.x, shipdata.y), you can get the corresponding step length of the interpolation point, Fig.3 for the system smoothing process flow chart, According to formula(2) in Matlab simulation of the interpolation point shown in Fig.4:

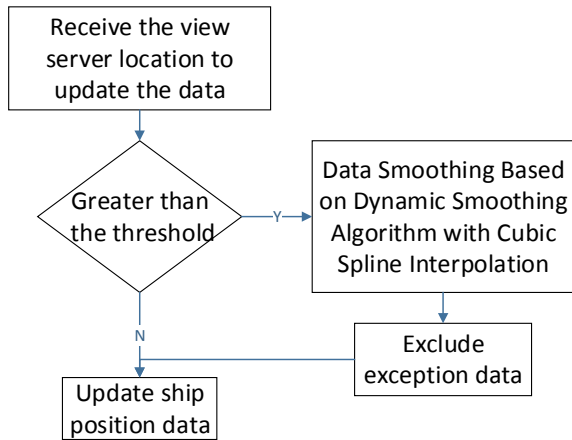


Fig.3. Interpolation smoothing process.

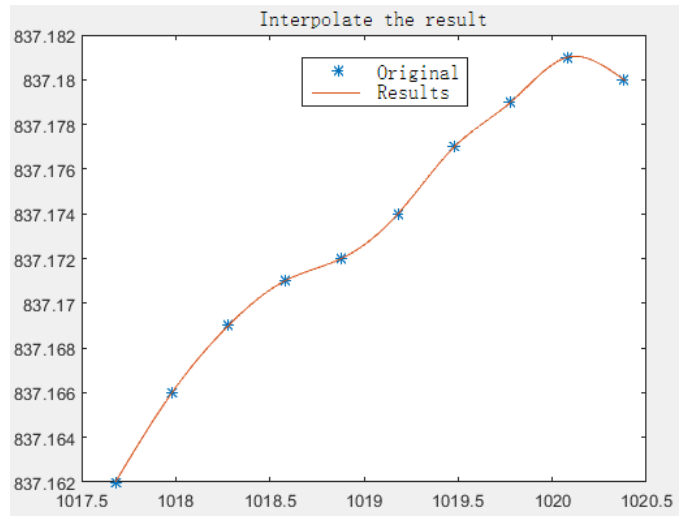


Fig.4. Matlab interpolation simulation.

The cubic spline interpolation's smoothing transition between the two data points ensures that the ship's trajectory is smooth, and the smooth trajectory of the simulated vessel is shown in Fig.5.

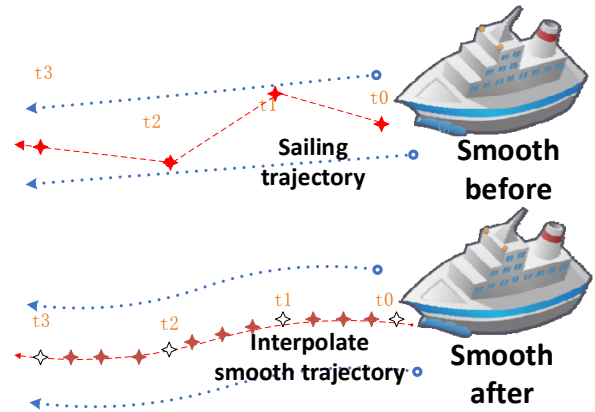


Fig.5. Simulation of ship navigation simulation after interpolation smoothing algorithm.

IV. EXPERIMENTAL RESULTS

Frame rate refers to the times when the computer updated in a second, which is an important criterion to evaluate the merits of a display system^[15]. Therefore, the display system joining the synchronization module ensure that the screen synchronization display also need to ensure the size of the frame rate. The frame rate is positively related to the advantages and disadvantages of the synchronization method proposed in this paper.

In this paper, four ordinary PC computers were used as the master-slave rendering node and the visual server to build the three-dimensional virtual simulation system. The two groups of experiments were designed to verify the effectiveness of the frame-bucket synchronization algorithm. Before and after each channel frame rate comparison experiment, and moving objects before and after the increase in frame rate changes in the ex-

periment. Fig.6(a)(b)(c)(d) is the display effect that the three-channel ship driving virtual simulation system is in the wall-mounted LG display screen.



(a) left channel



(b) right channel



(c) main channel



(d) Panorama renderings

Fig.6(a). Simulation diagram of the Left side of the ship driving simulator.
 Fig.6(b). Simulation diagram of the right side of the ship driving simulator.
 Fig.6(c). Simulation diagram of main channel of ship driving simulator.
 Fig.6(d). Simulation diagram of three - channel panorama of ship driving simulator.

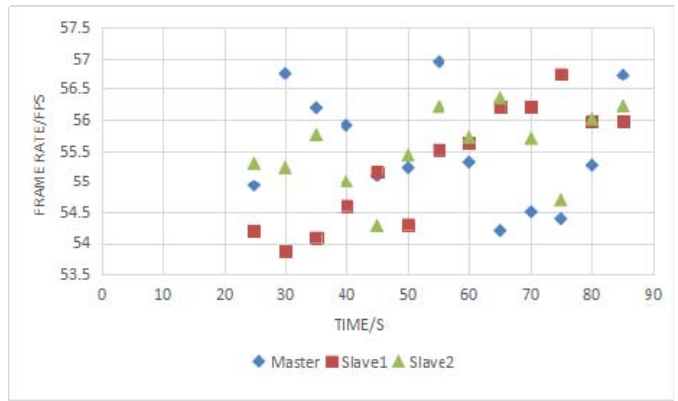


Fig.7. The frame rate is not used when synchronizing each frame.

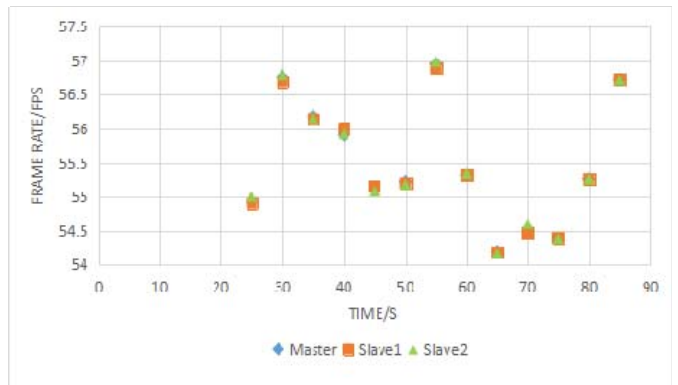


Fig.8. Frame rate for each channel when synchronization is used for frame bucket synchronization

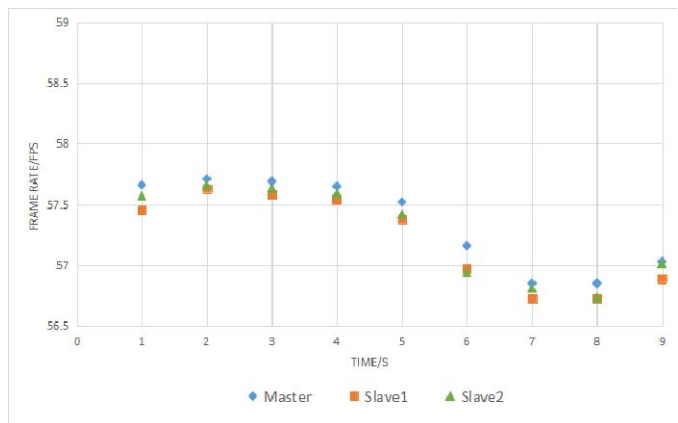


Fig.9. Increase the frame rate of each channel of the moving vessel when the synchronization method is not used.

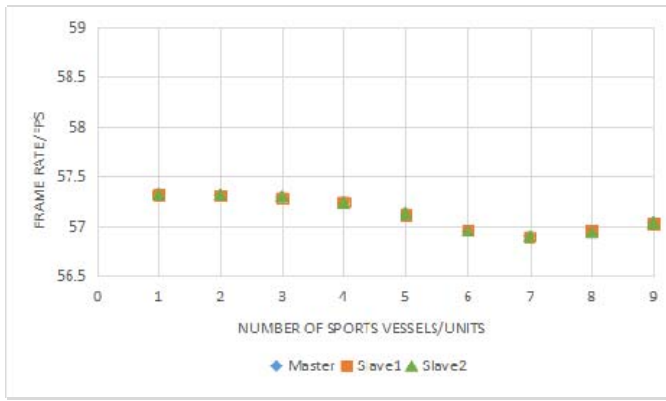


Fig.10. Increase the frame rate of each channel of the moving vessel when the synchronization method is used.

It can be seen from Fig.7 and Fig.8 that, the three-channel frame rate is not exactly the same and having a certain difference without the use of frame-lock synchronization algorithm, it will appear the split and ghosting phenomenon when a sports ship through the adjacent channel, the same movement of the ship in the adjacent channel display screen is not synchronized, the phenomenon disappears when use frame barrel lock synchronization algorithm. It can be seen from Fig.9 and Fig.10 that when the moving vessel increases, the number of triangular elements involved in rendering increases, and when the frame is not used to lock the synchronization algorithm, the frame rate difference of each channel is large, which resulting in inconsistent picture smoothness. Frame barrel lock synchronization algorithm, the channel screen fluency consistent, the overall display screen is better.

V. CONCLUSION

In this paper, a synchronous algorithm of bucket bucket is proposed to solve the problem of ghosting and tearing in the multi - channel ship driving 3D virtual simulation system when the moving entity through the adjacent channel. By introducing the RakNet network engine to solve the unreliability of traditional network communication, a motion smoothing algorithm based on cubic spline interpolation is proposed to solve the stochastic jitter problem caused by the discontinuity of the discontinuous motion state of simulated entity. On the basis of theoretical argumentation, the experimental results are simulated by simulating the relevant experiments in the virtual environment, which verifies the feasibility and effectiveness of the above methods, gives the system solutions of the problems and improves the The multi-channel multi-motion entity shows

the quality of the picture at the time of synchronization, and also extends the use of multi-channel synchronization technology in complex scenes.

REFERENCES

- [1] REN Shao-feng, Study and Implementation of Roaming System Based on Multi-channel Curved Stereo Projection [D]. Wuhan: Huazhong University of Science and Technology, 2012.
- [2] WANG Dan-xia. Real-time generation technology of multi-machine distributed collaborative 3D vision and its application [D]. Sichuan: Sichuan University, 2006.
- [3] LUO Min, LI Zi-Yang, FENG Lei, et al. Study on Multi-channel 3D Dynamic Display Synchronization Method [J]. Geospatial Information, 2016, 14 (3): 48-51.
- [4] GUO Yu-kun, WANG Zhong-hong. Comparison and Selection of B / S and C / S Patterns in Application [J]. Information and Communication, 2014, 2 (2): 102-102.
- [5] DENG Qing-ying, LI Guo-hui, WANG Bao-qi et al. Design and Implementation of a Flight Simulator Vision System [J]. Ordnance Industry Automation, 2016, 35 (8): 75-79.
- [6] Gilmore J.A state management and persistency architecture for peer-to-peer massively multi-user virtual environments[J]. Acm Sigmulti-media Records, 2013, 5(1):13-14.
- [7] Ravindran K, Mukhopadhyay S, Sidhanta S, et al. Managing shared contexts in distributed multi-player game systems[C]. Sixth International Conference on Communication Systems and Networks. IEEE, 2014:1-8.
- [8] Xiao Chao; Yang Hongyu. Research on Synchronization Algorithm of Multi-channel Projection Display System [J]; Journal of Sichuan University (Engineering Science Edition); 2012, 44 (4): 51-53.
- [9] YANG Hong-yu, YOU Zhi-sheng, ZHANG Jian-wei, YU Bo. Study on DR algorithm in interactive simulation of spatial and ground co-distributed [J]. Journal of Computer Research and Development, 2000, 22 (8): 2-4.
- [10] Li Jianwei, Chen Xin, Huang Zhouxiang et al. Design and Implementation of Extensible MMORPG Game Framework [J]. Computer Technology and Development, 2012, 22 (2): 2-4.
- [11] ZHAO Lan-fei, XI Zhi-hong. High Dynamic Range Imaging Method Based on Cubic Spline Interpolation [J]. Journal of Computer Aided Design and Graphics, 2015, 5 (7): 1279-1281.
- [12] WU Mei-jun, FU Jin-xin. Design of Ship Outer Plate Profile Based on Cubic Spline Interpolation [J]. Journal of Normal University, 2016, 36 (9): 29-31.
- [13] Fan Danyang, Yun Weiguo. Design and Implementation of Multi-channel Visual Simulation System Based on PC Cluster [J]. Computer Engineering and Applications, 2015, 51 (18): 204-208.
- [14] LI Dong. Three-dimensional distribution of cloud water in precipitation process based on multi-satellite multi-channel monitoring [D]. Donghua University, 2015.
- [15] LUO Min. Study on Synchronization Strategy of Multi-channel 3D Scene Rendering [J]. Journal of Beijing Information Science and Technology University (Natural Science Edition), 2015, (5): 48-51.