The Design and Implementation of an Augmented Reality Scheme Based on Netty Communication

Yongping Gao¹,², Jie Wang¹,² and Jingzhong Zhang³

¹Jiangxi Engineering Laboratory on Radioactive Geoscience and Big Data Technology, East China University of Technology, Nanchang, 330013, Jiangxi, China,
²Jiangxi Engineering Research Center of Nuclear Geoscience Data Science and System, East China University of Technology, Nanchang, 330013, Jiangxi, China
³Jiangxi ZhongYe Light Culture Technology Limited Company, Nanchang, 330006, Jiangxi, China
¹Email: ypgaoypgao@163.com

Abstract. This paper introduces a whole project of augmented reality interaction mode based on netty communication method, determines the main technical difficulties in the implementation process, and finds the solution by means of system architecture design and communication protocol development. In this paper, the netty framework is adopted to deal with network IO and business logic, and the communication between each terminal device, virtual environment and server is established through long connection. This paper adopts the netty framework to deal with network IO and business logic, establishes communication between each terminal device and virtual environment and server through long connection, so as to realize fast and natural real-time interaction between various interactive devices and other augmented reality devices and virtual environments under multiple users. It can also solve the problem of high concurrency of real-time communication between various devices and the virtual environment in augmented reality.

1. Introduction

In recent years, due to the rapid development of smart devices and information technology, and the promotion of computer vision and artificial intelligence, augmented reality technology has shown a strong momentum of development, has received increasing attention, and has played an important role, promptly triggered. Combined with education, medical, game lights, and other fields, and showing great potential. However, it's clear that there are many issues to be solved in Augmented Reality. From the perspective of interaction mode, the quality of human-computer interaction mode will greatly affect the user's interactive experience. AR technology has evolved to the present day, in terms of human-computer interaction, it is usually based on the interaction of traditional hardware devices, such as keyboard or handle, or voice-based interaction, or interaction based on motion recognition. But the more natural, real-time augmented reality interactive technology that supports multiple users remains to be studied.

Augmented reality system integrates and superposition and rendering of computer-generated virtual images and real scenes captured by the camera in real time, and display them in a space or picture, so as to build an augmented world of virtual and real fusion.

Interaction is a process in which two objects affect and influence with each other. The common way of human-computer interaction is through an external device to achieve the user interactions with the virtual object or the interaction through speech recognition and motion capture. The interaction
includes not only moving Objects, as well as the feeling of objects and the various forces, using the sense of touch, hearing, and smell device and interaction of the virtual scene. By creating sound effects, providing tactile, olfactory, and force perception and other kinds of perceptual perception to establish a natural channel matching with human senses, so as to present a perfect virtual world. But how are the various external interacting devices and sensing devices connected to the virtual objects? This requires the use of communication technology to achieve the connection between the device and the virtual world. Therefore, it is necessary to design a high-concurrency instant messaging application that can satisfy the device communication [1]. The choice of network communication technology is especially important. Not only requires stability and real-time performance, but also requires the ability to handle high concurrent communications, and can manage more hardware devices at the same time. Based on this requirement, this paper proposes to develop high concurrency server by JavaNIO based on Netty and to use the custom private protocol to interact devices and enhance communication between devices and the world. Based on this, the project provides a complete set of design solutions.

2. Design and Implementation of the Scheme

2.1. System Architecture
This project consists of terminal equipment, application server, business server and augmented reality server. Currently, the interactive devices in the augmented reality system mainly include hardware devices, voice recognition devices, or motion capture devices, and other terminal devices. such as sound, photoelectric and water, etc. The first function of the terminal module is to collect data information, such as collecting action information or voice information. The second is to accept the server to send the command and execute. Terminal devices are classified in the system, and different business servers are responsible for different types. It mainly solves the problem of preventing data from entering concurrent servers and affecting the overall performance of the system due to long data storage time. Data distribution to multiple business servers for processing operations can improve the corresponding speed of the system [2]. The application server is the core and brain of the whole system, where all the logic control is implemented. The interaction between each terminal device and the content of the augmented reality requires the establishment of a long connection link with the server, and then the communication between all the terminal devices and the augmented reality world is realized through the server.

2.2. System Design

2.2.1. Communication technology
Netty is a network programming framework and tool for NIO-based asynchronous communication provided by JBOSS. It is an improvement based on Epoll model and JavaNIO. It uses IO multiplexing technology to handle IO operations. It adopts IO multiplexing technology to handle IO operations and multiplexes multiple IO blocks to a select, so as to meet the connection requirements of multiple IO handled by a single thread. When dealing with large-scale concurrency, the Epoll model is used to process ready I/O connections, avoiding invalid polling. Its zero-copy technique can combine multiple ByteBuf references into one logical ByteBuf object, avoiding copies between various bytebufs [3]. Netty is one of the most popular NIO frameworks in the industry, and its robustness, functionality, performance, customizability, and scalability are all very good. Therefore, this project chooses netty as the communication framework of this project. The server-side sequence diagram is shown as follows:
2.2.2. The system mechanism

(1) Communication protocol

In order to realize the communication between each module in the system and the server, this project has developed its own communication protocol, namely the rules to be followed in the communication. Only when both parties “speak” according to this rule, can the other party understand and serve for it [4]. In addition to the business messages between the modules and the server, you need to carry other instructions for control and management. For example, the handshake request and response messages established by the link, heartbeat messages that monitor the availability of the link and so on are combined to form the communication protocol. The NIO of Netty is used for the customization and development of private protocols. The protocol stack of Netty is used for communication among internal modules. It is based on the TCP/IP protocol stack and adopts the flow method for data transmission.

Communication model:

Figure 1. Server-side sequence diagram

Figure 2. System communication model diagram
Table 1. Message definition table

| Field name | Field type | Field length | Field meaning                                      |
|------------|------------|--------------|---------------------------------------------------|
| SOURCE     | String     | longer       | ID of the message sender                           |
| TARGET     | String     | longer       | ID of the message recipient                       |
| PARAMER    | String     | longer       | For the request message, it is the parameter to the method, and for the response message, it is the return value |
| Separator  | String     | 1            | Symbol to separate each field                     |
| Terminator | String     | 1            | As a sign of the end of each message               |

(2) Heartbeat mechanism
Because this system requires frequent communication between the client and the server, if there is a data communication both sides each interaction, establish a connection, that is to say, Only one data is sent per connection, Will lead to frequent connection establishment, which is a waste of resources, and also need time to establish a connection, which causes the system to processing speed will reduce a lot. Therefore, this article uses a long connection mechanism. After each connection is established, it is opened continuously. All data interactions between client and server are communicated through this connection, a connection can continuously send multiple data packets. However, during the connection hold period, when there is no data interaction between the two parties, in order to detect the availability of the link, both parties need to send a link detection packet, that is, a heartbeat protocol. The implementation of the heartbeat mechanism is very simple. It directly uses Netty's IdleStateHandler mechanism. When the server and the client are in the idle state, the server detects the read at regular intervals. If no packets are detected for two consecutive times. The resource will be released, all the information of the client will be cleared, and waiting for the client to initiate connections. On the client side, the heartbeat detection of the IdleStateHandler is performed once every time. If the writing method is not called within a certain time interval, a heartbeat message packet is required to send to the server side.

(3) Break line reconnection mechanism
This project is based on network communication, but in the system running, the network status is unpredictable, the client should respond to the network changes in time, in order not to affect the operation of the system, minimize the impact, when the client loses the connection with the server, it is necessary to initiate a reconnection immediately to obtain a connection with the server. The first thing is to monitor the connection of the network. The client hangs on the closeFuture to listen for the close signal of the link. Once the channel is closed, the reconnection timer is created. After 5S, the connection is re-initiated until the server is connected.

2.2.3. System topology
From the perspective of the Internet of things, this project is divided into three layers from the bottom to the top. The first is the on-site machine. This level is mainly for the collection of information on the site, mainly including voice information, or action information and collection of various external device information. The second is the network layer, which uses the wireless and wired networks to encode, authenticate and transmit the collected data. Its main function is to realize the data transmission between the field machine and the host computer. The third is the upper computer, that is, the data application layer, On the one hand, relevant conclusions can be drawn through data analysis to support the decision of the server; on the other hand, it can receive the instructions passed by the server to control the lower equipment. The specific system topology diagram is as follows: mainly consists of hardware equipment, transmission network, client or data acquisition and control unit, server and so on.
2.2.4. Software architecture design
The system platform is divided into four modules, which are data receiving module, business processing module, data distribution module and log recording module. The data receiving module processes the data request from the client through the implementation of protocol adaptation interface, verifies the legality and validity of these data, and encapsulates the legitimate data message, and then delivers it to the business processing module. The service processing module parses the received data according to the corresponding rules and then implements specific business logic processing. For example, after the server receives the data from the client, if the data needs to be written into the database, the service processing module for data after certain processing and analysis, written to the database to achieve data persistence. The message of the data transmission module comes from the business processing module, which encapsulates and encodes the data to be sent according to the protocol rules, and then sends it to the corresponding client, and the client parses the received data and then performs corresponding processing on the lower computer. The logging module is mainly to meet the needs of software performance test and daily maintenance work. Different levels of log output (info, debug, warn, error) can be set according to the needs. You can set different levels of log output (info, debug, warn, error), etc., and output the log. You can set the log output mode, such as a console or a file. The purpose of logging is to facilitate the identification of problems in case of system failures or problems so as to help solve problems quickly [5]. The functional module diagram is shown as follows:

![System topology diagram](image-url)
3. Concluding

According to the requirements of fast, natural and real-time interaction between various interactive devices and virtual environments under multiple users in the augmented reality system, the paper proposed a processing method based on netty communication and augmented reality interaction mode, which realizes the network communication service that supports high concurrency and complex logic processing. The server data processing speed is fast, the response time is short, can satisfy the real-time corresponding request. At present, this solution has been successfully applied in some projects of a company. Users can use a variety of interactive devices and virtual content in the scene for real-time interaction, and combine multimedia, 3D, photoelectric sound effect and other means, as well as various physical mechanical devices to create a dreamlike and real environment. Customers feel very new and satisfied with the experience. Therefore, the prospect of market application is very broad.

4. Acknowledgment

The paper is sponsored by fund (fund id: JELRGBDT201707, National Natural Science Foundation of China 61662002, 61463003 & 11865002). Yongping Gao is the corresponding author.

5. References

[1] Wang Yu-Xi, Zhang Feng-Jun, Liu Yue. Research status and development trend of augmented reality technology [J]. Journal of science and technology review, 2018, 36 (10): 75-83.
[2] Chen Qiang, Wu Jia-Jia. Research and design of Internet of things server based on Netty [J]. Journal of Electronic technology and software engineering, 2018 (07): 34-35.
[3] Zhang He. Research and application of multi-user concurrent access technology based on Netty framework [D]. Henan University, 2017.
[4] Chen Rui-Lin, Zheng-Yu. Design and implementation of UDP protocol stack based on ARM9 [J]. Journal of Computer Technology and Development, 2013, 23 (05): 22-25.
[5] Gong Peng, Zeng Xing-bin. Design of data communication service system based on Netty framework [J]. Journal of Wireless communication technology, 2016, 25 (01): 46-49.